



37) ๓๓
๑๐ ๑๖๘

ระบบวัดปริมาณน้ำฝนระยะไกลผ่านคลื่นวิทยุ
WIRELESS RAIN TELEMETERING SYSTEM

โดย

นายชัยวัฒน์ สมบูรณ์ลาภ 38013055

นายมงคล จันทร์ประพาฬ 38013068

นายสารสิทธิ์ วิชัยทัตะ 38013081

วัน เดือน ปี..... 17. ค.ค. 2541
เลขทะเบียน..... 039051
เลขเรียกหนังสือ..... 1.๑๐1๙๒ ศ.๑๒๒.๖

อาจารย์ที่ปรึกษา
ผ.ศ.นิภา ลีสารุจิ

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต
สาขาวิศวกรรมโทรคมนาคม
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
ปีการศึกษา 2540

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้ง 039051

ปริญญานิพนธ์ปีการศึกษา 2540

ภาควิชาวิศวกรรมโทรคมนาคม

คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เรื่อง ระบบวัดปริมาณน้ำฝนระยะไกลผ่านคลื่นวิทยุ

WIRELESS RAIN TELEMETERING SYSTEM

จัดทำโดย

- | | |
|---------------------------|----------|
| 1. นายชัยวัฒน์ สมบูรณ์ลาภ | 38013055 |
| 2. นายมงคล จันทร์ประภาพ | 38013068 |
| 3. นายสารสิทธิ์ วิชัยทัตะ | 38013081 |


.....อาจารย์ที่ปรึกษา
(ผศ.นิภา ลีลาวัจ)

เอกสารนี้เป็นเอกสารที่ส่งมอบไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ระบบวัดปริมาณน้ำฝนระยะไกลผ่านคลื่นวิทยุ

WIRELESS RAIN TELEMETERING SYSTEM

โดย นายชัยวัฒน์ สมบูรณ์ลาภ	38013055
นายมงคล จันทร์ประพาฬ	38013068
นายสารสิทธิ์ วิชัยทัศะ	38013081

อาจารย์ที่ปรึกษา ผศ. นิภา สีสารุจิ

บทคัดย่อ

โครงการฉบับนี้ได้นำเสนอการจัดทำระบบวัดปริมาณน้ำฝนระยะไกล ผ่านคลื่นวิทยุด้วยการประยุกต์สำหรับการใช้งานไมโครคอนโทรลเลอร์ กับเครื่องมือวัดที่จะแสดงผลการวัดลักษณะ ทางกายภาพที่ต้องการ ได้แก่ เครื่องมือวัดน้ำฝนแบบคานกระดก โดยจะทำงานร่วมกับไมโครคอนโทรลเลอร์ ตระกูล MCS - 51 เบอร์ 8032 และทำการบันทึกผล เทียบเวลาจริง จากนั้นส่งข้อมูล ไปยังเครื่องคอมพิวเตอร์โดยผ่านทางคลื่นวิทยุ จากนั้น นำข้อมูลที่ได้ไปทำการวิเคราะห์ เพื่อแสดงผลเป็นกราฟบนคอมพิวเตอร์

ABSTRACT

This project presents a design of the long distance rain telemetering system via radio wave. It is applied for micro-controller and measurement tool will show the desired physical characteristics. The tipping rain gauge will sent the signal through Micro-controller MCS - 51 with number 8032 , then it will record the results by comparing with the real time. After that it will sent the data through computer by using the radio wave . Finally, the computer analyzes the data and displays .

สารบัญ

เรื่อง	หน้า
บทที่ 1 บทนำ	1
บทที่ 2 ทฤษฎีและหลักการ	2
2.1 การวัดหยาดน้ำฟ้าและการระเหยของน้ำ	2
2.1.1 ข้อกำหนดโดยทั่วไป	2
2.1.2 หน่วยที่ใช้ในการวัด	2
2.1.3 การวัดจำนวนน้ำฝน	2
2.1.4 การติดตั้งเครื่องมือ	3
2.1.5 เครื่องวัดปริมาณน้ำฝนแบบคานกระดก	4
2.2 บอร์ดไมโครคอนโทรลเลอร์รุ่น แอนด-32	6
2.2.1 คุณสมบัติของบอร์ดแอนด-32	6
2.2.2 การอ้างอิงตำแหน่งพอร์ต 8255 ของบอร์ดไมโครคอนโทรลเลอร์	6
2.2.3 การอ้างอิงตำแหน่งพอร์ตแอลซีดีโมดูลของบอร์ดไมโครคอนโทรลเลอร์	7
2.2.4 นาฬิกาเวลาจริงและปฏิทิน	7
2.3 การอินเตอร์รัปต์	9
2.3.1 โครงสร้างการอินเตอร์รัปต์	11
2.3.2 การควบคุมการอินเตอร์รัปต์	12
2.3.3 การจัดการอินเตอร์รัปต์	13
2.4 การรับส่งข้อมูลแบบอนุกรม	14
2.4.1 อัตราความเร็ว	14
2.4.2 บิตเริ่มต้น	15
2.4.3 พาร์ตีบิต	16
2.4.4 บิตหยุด	17
2.4.5 การรับและส่งข้อมูลแบบอนุกรมผ่าน 8032	18
2.5 มาตรฐาน RS 232C	20
2.5.1 ลักษณะของสัญญาณ RS 232C	21
2.5.2 การกำหนดจุดต่อของ RS 232C	21
บทที่ 3 การออกแบบและการสร้าง	25
3.1 การเชื่อมต่อของคีย์บอร์ดกับบอร์ดไมโครคอนโทรลเลอร์	25
3.2 การเชื่อมต่อของเครื่องวัดปริมาณน้ำฝนกับไมโครคอนโทรลเลอร์	26
3.3 การเก็บข้อมูล และพื้นที่ในการเก็บข้อมูล	26
3.3.1 รหัสที่ใช้ในการเก็บข้อมูลและลักษณะของข้อมูลที่เก็บ	26

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.3.2 การเก็บข้อมูลปริมาณน้ำฝน	26
3.3.3 ความสามารถในการเก็บข้อมูล	26
3.4 โฟลว์ชาร์ตของโปรแกรม 8032	28
3.4.1 โฟลว์ชาร์ตของโปรแกรมหลัก	28
3.4.2 โฟลว์ชาร์ตของโปรแกรมรับคีย์บอร์ดค่าตัวเลข	29
3.4.3 โฟลว์ชาร์ตของโปรแกรมการติดต่อกับ DS1202	30
3.4.4 โฟลว์ชาร์ตของโปรแกรมการรับคำสั่งจากคีย์บอร์ดควบคุมการทำงาน	31
3.4.5 โฟลว์ชาร์ตของโปรแกรมการรับสัญญาณจากเครื่องวัดปริมาณน้ำฝน	32
3.4.6 โฟลว์ชาร์ตของโปรแกรมการบันทึกข้อมูลปริมาณน้ำฝน	32
3.4.7 โฟลว์ชาร์ตของโปรแกรมการส่งข้อมูลจาก RAM ไปยังเครื่องคอมพิวเตอร์	33
3.4.8 โฟลว์ชาร์ตของโปรแกรมย่อยการส่งค่าออกพอร์ตอนุกรมทุกๆ 1 นาที	33
3.4.9 โฟลว์ชาร์ตของโปรแกรมการส่งข้อมูลปริมาณน้ำฝนไปยังเครื่องคอมพิวเตอร์	34
3.4.10 โฟลว์ชาร์ตของโปรแกรมหลัก รับข้อมูลจากบอร์ดไมโครคอนโทรลเลอร์ มาแสดงและจัดเก็บ	35
3.5 ภาค FSK Modulation	36
3.6 วงจรเครื่องส่ง	37
3.7ภาครับสัญญาณคลื่นวิทยุ (FM Receiver)	38
3.8 ภาค FSK Demodulation	41
3.8.1 พื้นฐานของเฟสล็อกกลูป	41
3.8.2 สถานะของการทำงานของเฟสล็อกกลูป	42
3.8.3 การออกแบบ FSK Dmodulator โดยใช้ XR 2211	43
บทที่ 4 การทดลองและผลการทดลอง	44
4.1 ทดลองการเชื่อมต่อของ คีย์บอร์ดไมโครคอนโทรลเลอร์	44
4.2 ทดลองป้อนโปรแกรมลงบอร์ดไมโครคอนโทรลเลอร์และเชื่อมต่อบอร์ดไมโครคอนโทรลเลอร์กับจอ LCD และคีย์บอร์ด	45
4.3 ทดลองการเชื่อมต่อของเครื่องวัดปริมาณน้ำฝนกับ บอร์ดไมโครคอนโทรลเลอร์	46
4.4 ทดลองสวิตช์ควบคุมการทำงานของบอร์ดไมโครคอนโทรลเลอร์	48
4.5 ทดลองโปรแกรมเอ็กซ์ทอลก์ที่ใช้แสดงรูปแบบการส่งข้อมูลออกทางพอร์ตอนุกรมโดยรวมผลการบันทึกข้อมูลของปริมาณน้ำฝนด้วย	50
4.6 การทดลองชุดรับส่งข้อมูลแบบดิจิตอล	51
4.7 การทดลองโปรแกรมคอมพิวเตอร์ของการรับข้อมูลแสดงกราฟข้อมูลปริมาณน้ำฝน	54
บทที่ 5 บทวิจารณ์และสรุป	59
ภาคผนวก	

สารบัญรูปภาพ

เรื่อง	หน้า
รูปที่ 2.1 แสดงเครื่องวัดปริมาณน้ำฝนแบบคานกระดก	4
รูปที่ 2.2 กลไกของเครื่องวัดน้ำฝนแบบคานกระดก	5
รูปที่ 2.3 แสดงอัตราการผิดพลาดของเครื่องวัดปริมาณน้ำฝนแบบคานกระดก	5
รูปที่ 2.4 แสดงไคอะแกรมเวลาของการส่งข้อมูลเพื่อติดต่อกับ DS 1202 แบบไบท์เดี่ยว	8
รูปที่ 2.5 แสดงบล็อกไคอะแกรม DS 1202	10
รูปที่ 2.6 แสดงให้เห็นถึงแผนภาพการทำงานของโปรแกรม	10
รูปที่ 2.7 แผนภาพแสดงโครงสร้างระบบอินเตอร์รัปต์ของ 8051	12
รูปที่ 2.8 แสดงการส่งข้อมูลแบบอนุกรม ซึ่งจะทำการส่งข้อมูลที่ละบิต	14
รูปที่ 2.9 แสดงสัญญาณของข้อมูลที่ถูกส่งไปตามสายสัญญาณ	15
รูปที่ 2.10 แสดงการเพิ่มบิตที่เริ่มต้นเข้าไปก่อนหน้าบิต D0 ในกรณีที่บิต D0 เป็น "1" และ "0" ตามลำดับ	16
รูปที่ 2.11 แสดงการเพิ่มพาริตี เป็น "1" ลงไปในข้อมูลแต่ละไบท์	17
รูปที่ 2.12 แสดงการเพิ่มพาริตี เป็น "0" ลงไปในข้อมูลแต่ละไบท์	17
รูปที่ 2.13 แสดงการเพิ่มบิตต่าง ๆ ที่ใช้สำหรับป้องกันการผิดพลาดที่จะเกิดขึ้นในการรับส่งข้อมูลแบบอนุกรม	18
รูปที่ 2.14 แสดงการใช้ RS 232 C เชื่อมต่ออุปกรณ์	21
รูปที่ 2.15 แสดงพอร์ตและขาที่ใช้งาน	22
รูปที่ 3.1 แสดงบล็อกไคอะแกรมของวงจร	25
รูปที่ 3.2 วงจรการเชื่อมต่อคีย์บอร์ดกับ 8255	25
รูปที่ 3.3 แสดงการเชื่อมต่อของเครื่องวัดปริมาณน้ำฝนกับไมโครคอนโทรลเลอร์	26
รูปที่ 3.4 แสดงไพล์ชาร์ตของโปรแกรมหลัก	28
รูปที่ 3.5 แสดงไพล์ชาร์ตของการรับคำสั่งจากคีย์บอร์ดค่าตัวเลข	29
รูปที่ 3.6 แสดงไพล์ชาร์ตของการติดต่อกับ DS1202	30
รูปที่ 3.7 แสดงไพล์ชาร์ตของการรับคำสั่งจากคีย์บอร์ดควบคุมการทำงาน	31
รูปที่ 3.8 แสดงไพล์ชาร์ตของการรับสัญญาณจากเครื่องวัดปริมาณน้ำฝน	32
รูปที่ 3.9 แสดงไพล์ชาร์ตของการบันทึกข้อมูลปริมาณน้ำฝน	32
รูปที่ 3.10 แสดงไพล์ชาร์ตของการส่งข้อมูลจาก RAM ไปยังเครื่องคอมพิวเตอร์	33
รูปที่ 3.11 แสดงไพล์ชาร์ตของโปรแกรมย่อยการส่งค่าออกพอร์ตอนุกรมทุกๆ 1 นาที	33
รูปที่ 3.12 แสดงไพล์ชาร์ตของการส่งข้อมูลปริมาณน้ำฝนไปยังเครื่องคอมพิวเตอร์	34
รูปที่ 3.13 แสดงไพล์ชาร์ตการทำงานของโปรแกรมหลักในการรับข้อมูลที่คอมพิวเตอร์	35
รูปที่ 3.14 FSK MODURATION	37
รูปที่ 3.15 FM Transmitter Circuit	38

เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เรื่อง	หน้า
รูปที่ 3.16 Circuit FM Receiver	40
รูปที่ 3.17 บล็อกไดอะแกรมการทำงานของเฟสล็อกถู	41
รูปที่ 3.18 วงจรใช้งานจริง XR 2211 FSK Decoder	42
รูปที่ 4.1 สัญญาณที่เข้าไปยังขาของพอร์ต 8255 เมื่อทำการกดสวิทช์	44
รูปที่ 4.2 แสดงสัญญาณที่เข้าไปยังขาของพอร์ต 8255 เมื่อทำการกดสวิทช์หลายครั้งติดต่อกัน	44
รูปที่ 4.3 แสดง DEMO หน้าจอเมื่อป้อนโปรแกรมลงบอร์ดไมโครคอนโทรลเลอร์	45
รูปที่ 4.4 แสดงหน้าจอก่อนการป้อนค่าขนาดของอุปกรณ์ที่ใช้รับปริมาณน้ำฝน	45
รูปที่ 4.5 แสดงหน้าจอการป้อนค่าขนาดของอุปกรณ์ที่ใช้รับปริมาณน้ำฝน	45
รูปที่ 4.6 แสดงหน้าจอก่อนป้อนค่าเริ่มต้นของ วัน,เดือน,ปี และเวลา	45
รูปที่ 4.7 แสดงหน้าจอเมื่อป้อนค่า วัน,เดือน,ปี และเวลาเรียบร้อยแล้ว	46
รูปที่ 4.8 แสดงสัญญาณที่ขาอินเตอร์รัปต์ขณะที่คานกระดกของเครื่องรับมีการเกาะหลายครั้ง	46
รูปที่ 4.9 แสดงการเปลี่ยนแปลงหน้าจอ LCD เมื่อมีสัญญาณเข้ามาที่ขาอินเตอร์รัปต์	47
รูปที่ 4.10 แสดงการเปลี่ยนแปลงข้อมูลบนจอ LCD	47
รูปที่ 4.11 แสดงรูปสัญญาณ ที่ บันทึกได้ บนกระดาษกราฟ	47
รูปที่ 4.12 แสดงข้อมูลที่เครื่องไมโครคอนโทรลเลอร์บันทึกได้ โดย ทำการส่งข้อมูลขึ้นบนคอมพิวเตอร์	48
รูปที่ 4.13 แสดงหน้าจอ LCD เมื่อมีการกดคีย์ลบข้อมูลในเมโมรี่	48
รูปที่ 4.14 แสดงการทำงานของโปรแกรม การลบข้อมูลในเมโมรี่ โดยจะมีการตรวจสอบรหัสผ่าน	49
รูปที่ 4.15 แสดงการทำงานการลบข้อมูลในเมโมรี่ตรวจสอบรหัสผ่านถูกต้อง	49
รูปที่ 4.16 แสดงการทำงานการลบข้อมูลในเมโมรี่ตรวจสอบรหัสผ่านไม่ถูกต้อง	49
รูปที่ 4.17 แสดงหน้าจอ LCD หลังจากมีการกดปุ่มควบคุมการบันทึกข้อมูลของปริมาณน้ำฝน	50
รูปที่ 4.18 แสดงตัวอย่างโปรแกรมเอกซ์ทอลก์ที่ใช้แสดงผลการบันทึกข้อมูล	50
รูปที่ 4.19 แสดงความถี่ของสัญญาณที่ส่งออกจากเครื่องส่ง	51
รูปที่ 4.20 แสดงสัญญาณอินพุตที่ป้อนให้เครื่องส่ง	52
รูปที่ 4.21 แสดงความถี่ OUTPUT ที่ IC TA8122AN	52
รูปที่ 4.22 แสดงสัญญาณเอาต์พุตของวงจร FSK Modulation เพื่อป้อนลอจิก “1”	52
รูปที่ 4.23 แสดงสัญญาณเอาต์พุตของวงจร FSK Modulation เพื่อป้อนลอจิก “0”	53
รูปที่ 4.24 แสดงสัญญาณเอาต์พุตของวงจร FSK Demodulation เทียบกับเอาต์พุตของ FSK Modulation เมื่อป้อนลอจิก “1”	53
รูปที่ 4.25 แสดงสัญญาณเอาต์พุตของวงจร FSK Demodulation เทียบกับเอาต์พุตของ FSK Modulation เมื่อป้อนลอจิก “0”	54
รูปที่ 4.26 ผลที่หน้าจอคอมพิวเตอร์เมื่อเปิดโปรแกรมการรับข้อมูล	54
รูปที่ 4.27 ผลที่หน้าจอคอมพิวเตอร์เมื่อเปิดไฟล์เพื่อแก้ไข เก็บข้อมูล และแสดงกราฟ	55

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เรื่อง	หน้า
รูปที่ 4.28 ผลที่หน้าจอคอมพิวเตอร์เมื่อเก็บไฟล์ซึ่งอยู่ในรูปแบบค่าเบส	55
รูปที่ 4.29 ผลที่หน้าจอคอมพิวเตอร์เมื่อรับข้อมูลจากพอร์ตอนุกรม	56
รูปที่ 4.30 แสดงการเปรียบเทียบการรับข้อมูลจากเครื่องกับที่หน้าจอคอมพิวเตอร์	56
รูปที่ 4.31 ผลที่หน้าจอคอมพิวเตอร์เมื่อแสดงตารางค่าเบส เพื่อขอดูและแก้ไข	57
รูปที่ 4.32 ผลที่หน้าจอคอมพิวเตอร์เมื่อแสดงการเลือกกราฟที่จะดู	57
รูปที่ 4.33 ผลที่หน้าจอคอมพิวเตอร์เมื่อแสดงผลกราฟปริมาณน้ำฝนในเวลา 1 ชั่วโมง	58



สารบัญตาราง

เรื่อง	หน้า
ตารางที่ 2.1 ตารางแสดงไบท์คำสั่งในการเขียนและอ่านค่านาฬิกา	9
ตารางที่ 2.2 สัญญาณอินเตอร์รัปต์ของ 8051 ทั้ง 5 ประเภท	12
ตารางที่ 2.3 แสดงบิตต่าง ๆ ภายในรีจิสเตอร์ IE	13
ตารางที่ 2.4 แสดงตำแหน่งอินเตอร์รัปต์แวกเตอร์ของสัญญาณอินเตอร์ต่าง ๆ	13
ตารางที่ 2.5 ค่าที่ต้องนำไปไว้ในรีจิสเตอร์ไทม์เมอร์ 1 เมื่อใช้อัตราความเร็วมาตรฐานต่าง ๆ	19
ตารางที่ 2.6 ย่านของแรงดันไฟฟ้าที่ใช้ในสัญญาณ RS 232C	21
ตารางที่ 2.7 คุณลักษณะโดยย่อของสัญญาณ RS232C	24
ตารางที่ 3.1 ข้อมูลโดยรวมของไอซี TA 8122 AN	39
ตารางที่ 3.2 ข้อมูลในการรับในแบบเอเอ็ม	39



บทที่ 1

บทนำ

ในทางอุตุนิยมวิทยา ลักษณะทางกายภาพของธรรมชาติที่เกิดขึ้น เป็นปัจจัยสำคัญอย่างมากในการศึกษาในด้านนี้ และ พยายามหาวิธีที่เหมาะสมในการบันทึกลักษณะทางกายภาพเหล่านี้ไว้เพื่อ ใช้ในการพยากรณ์ลักษณะต่าง ๆ และปรากฏการณ์ทางธรรมชาติที่อาจเกิดขึ้น อาทิเช่นการวัดความเร็วลม , การวัดอุณหภูมิ , การวัดปริมาณน้ำฝน เป็นต้น โดยจะมีการนำเอากระบวนการทางวิศวกรรมเข้ามาประยุกต์ใช้กับงานเหล่านี้

โครงการนี้ได้นำเอาเทคโนโลยีทางวิศวกรรมอย่าง ไมโครคอนโทรลเลอร์ ตระกูล MCS 51 มาประยุกต์ใช้กับเครื่องวัดปริมาณน้ำฝนแบบคานกระดก ซึ่งสามารถเก็บปริมาณน้ำฝนนอกสถานที่ เครื่องวัดปริมาณน้ำฝนจะให้สัญญาณมาเป็นพัลส์ แล้วส่งไปยังบอร์ดไมโครคอนโทรลเลอร์เพื่อเก็บข้อมูลเหล่านี้ไว้ในหน่วยความจำในการเก็บข้อมูลจะทำการเก็บข้อมูลเทียบกับเวลาจริง โดยใช้คีย์บอร์ดต่อกับไมโครคอนโทรลเลอร์เพื่อทำการตั้งเวลาในตอนเริ่มทำงาน และมีการแสดงผลบนจอแอลซีดี ซึ่งบนจอแอลซีดีจะแสดงข้อมูลตามเวลาจริง ผลที่เก็บไว้ในหน่วยความจำจะถูกส่งออกไปโดยเข้ารหัสก่อนส่งต่อไปยังเครื่องส่งและส่งออกให้เครื่องรับ เครื่องรับทำการถอดรหัสออกมาและส่งต่อไปยังเครื่องคอมพิวเตอร์ จากนั้นข้อมูลจะผ่านโปรแกรมในการวิเคราะห์เพื่อแสดงผลออกมาเป็นกราฟ

บทที่ 2

ทฤษฎีและหลักการ

2.1 การวัดหยาดน้ำฟ้าและการระเหยของน้ำ (MEASUREMENT OF PRECIPITATION AND EVAPORATION)

2.1.1 ข้อกำหนดโดยทั่วไป

จำนวนทั้งหมดของหยาดน้ำฟ้า (precipitation) ซึ่งตกลงมาจากบรรยากาศสู่ผิวพื้นผิวโลกใช้วัดเป็นความลึก (depth) หรือความหนาของหยาดน้ำฟ้าบนพื้นระดับเรียบ โดยสมมุติว่า หยาดน้ำฟ้าไม่มีการระเหยหรือซึมออกไป และถ้าเป็นจำนวนหิมะหรือน้ำแข็งจะต้องไม่ละลายเสียก่อน และการวัดหิมะก็ต้องวัดเป็นความลึกของหิมะที่เพิ่งตกใหม่ๆ บนพื้นระดับเรียบ สำหรับประเทศไทยเราไม่เคยปรากฏว่ามีหิมะตกเลย ส่วนลูกเห็บมีตกบ้างเป็นครั้งคราว โดยมากเมื่อมีพายุฝนฟ้าคะนองอย่างรุนแรง ดังนั้น เราจึงต้องการวัดแต่ปริมาณของน้ำฝนเพียงอย่างเดียว และโดยที่ประเทศไทยเป็นประเทศกสิกรรม การตรวจวัดหาค่าของปริมาณน้ำฝนจึงมีความสำคัญเป็นอย่างมาก กรมอุตุนิยมวิทยาจึงมีโครงการที่จะขยายข่ายสถานีฝน ให้หนาแน่นยิ่งขึ้น เท่าที่สามารถจะทำได้ตามกำลังงบประมาณที่จะอำนวยให้

2.1.2 หน่วยที่ใช้ในการวัด (Units of measurement)

ดังได้กล่าวแล้วว่า การวัดจำนวนน้ำฝนนั้น ใช้วัดเป็นความสูงของน้ำฝนที่ตกลงมาในเนื้อที่จำกัดอันหนึ่ง โดยคิดเสียน้ำฝนนั้นไม่มีการระเหยหรือไหลซึมไปไหน โดยที่เรารู้เนื้อที่ของปากถัง ดังนั้นเมื่อเราต้องการวัดความสูง (หรือความลึก) ของน้ำฝน เราก็สามารถใช้ไม้บรรทัดขึงวัดความสูงได้ หรือจะทำเป็นแก้วสำหรับตวงน้ำฝนต่างหากก็ได้

หน่วยที่ใช้วัดจำนวนน้ำฝนที่ใช้กันอยู่ทุกวันนี้วัดเป็น มิลลิเมตร การอ่านค่าของจำนวนน้ำฝนต้องวัดให้ได้ใกล้เคียงถึง 0.2 มิลลิเมตร สำหรับจำนวนน้ำฝน 10 มิลลิเมตร หรือน้อยกว่า แต่ถ้าจำนวนน้ำฝนมากกว่านี้ ยอมให้ได้ค่าใกล้เคียง 2% ของจำนวนฝนทั้งหมด ในบางประเทศใช้หน่วยที่วัดเป็นนิ้ว ซึ่งเทียบค่าได้ 1 นิ้ว เท่ากับ 25.4 มิลลิเมตรก็นับว่าถูกต้องดี

2.1.3 การวัดจำนวนน้ำฝน (Rainfall measurement)

2.1.3.1 เครื่องวัดฝน (Rain gauges)

แรกเริ่มในการวัดจำนวนน้ำฝนนั้น กาสเทลลิ ได้เป็นผู้คิดสร้างเครื่องวัดขึ้นในประเทศอิตาลี เมื่อปี ค.ศ. 1639 เครื่องประกอบด้วยแก้วทรงกระบอก มีเส้นผ่านศูนย์กลางปากกระบอกประมาณ 5 นิ้ว และลึก 9 นิ้ว ต่อมาจึงได้มีการวิวัฒนาการเรื่อยมาจนถึงขนาดทำให้สามารถจดบันทึกรายงานได้ด้วยตัวเอง (self-recording) เครื่องวัดฝนธรรมดาที่ใช้เป็นประจำวันเป็นรูปทรงกระบอก และมีกรวยต่อลงไปยังที่รองรับภายใน ขนาดของถังรับน้ำฝนภายนอกไม่สำคัญ แต่เนื้อที่ของบริเวณที่รับน้ำฝนนั้นควรจะอยู่ระหว่าง 200 ถึง 500 ตารางเซนติเมตรเป็นเหมาะสมที่สุด เส้นผ่านศูนย์กลางของถังรองรับน้ำฝนภายในควรเท่ากับ 1/10 ของเส้นผ่านศูนย์กลางของปากถังภายนอก โดยทั่วไปที่ใช้กันอยู่มีขนาดเส้นผ่านศูนย์กลางของปากถังภายนอก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เป็น 8 นิ้ว 6 นิ้ว และ 5 นิ้ว อย่างไรก็ตามไม่ว่าจะใช้ถึงขนาดใดก็ตาม เครื่องวัดฝนที่ใช้ประจำวันจะต้องเป็นไปตามหลักเกณฑ์ที่สำคัญดังต่อไปนี้คือ

-ขอบของปากถังด้านนอกต้องคม โดยทำให้ด้านในเป็นแนวตั้งตั้งตรง ส่วนด้านนอกลาดเอียงลงเป็นแนวชันมาก

-ต้องรู้เนื้อที่ช่องปากถังให้ใกล้เคียง 0.5% และเนื้อที่นี้จะต้องคงที่อยู่ตลอดไป

-ถังภายนอกต้องออกแบบให้กันการกระเซ็นของน้ำฝนทั้งเข้าและออก การนี้ทำได้โดยทำให้ตัวถังเป็นแนวตั้งลึกลงไปพอสมควร และแนวลาดของกรวยต้องอย่างน้อย 45 องศา

-ถังรองรับภายในควรทำเป็นคอแคบๆ เพื่อป้องกันไม่ให้น้ำฝนระเหยออกไป เนื่องจากเรดิเอชัน

2.1.3.2 การวัดฝน (Rain measure)

การวัดฝนตามวิธีธรรมดาโดยทั่วไปมีอยู่ 2 วิธีคือ ใช้แก้วดวงอย่างหนึ่งและใช้ไม้บรรทัดหยั่งวัด แก้วที่ใช้ดวงต้องเป็นแก้วใส และมีสัมประสิทธิ์ของการขยายตัวต่ำ และต้องบอกให้ชัดเจนด้วยว่าใช้กับเครื่องวัดฝนขนาดใด ขนาดของเส้นผ่านศูนย์กลางของแก้วดวงต้องไม่โตกว่าประมาณ 1/3 ของปากถังของเครื่องวัดฝน และถ้าจะให้ดีกว่านี้ให้เล็กกว่าขนาดที่กล่าวนี้

สเกลที่แบ่งไว้บนแก้วดวงต้องขีดให้ชัดเจน โดยทั่วไปจะขีดไว้ทุกๆ 0.2 มิลลิเมตร และเส้นจำนวนเต็มของสเกลต้องเขียนเลขกำกับไว้ด้วย ถ้าจะให้ดีกว่านี้ขีดสเกลให้อ่านได้ถึง 0.1 มิลลิเมตร ในการวัดเพื่อให้ได้ค่าใกล้เคียงที่สุดควรอ่านค่าอย่าให้อัตราผิดเกินกว่า 0.02 มิลลิเมตร เมื่อจำนวนฝนน้อยกว่า 2 มิลลิเมตร

เพื่อที่จะให้ได้ค่าที่ถูกต้องแม่นยำจริงๆ เมื่อมีจำนวนฝนน้อย เขาจึงทำกระบอกแก้วดวงตอนก้นสุดให้เรียวเล็กลง ในการตรวจทุกครั้งต้องถือแก้วดวงให้ได้แนวตั้งจริงๆ เพื่อว่าระดับน้ำฝนในแก้วดวงจะได้อยู่ในแนวนอนตามขีดสเกล ทั้งนี้เพื่อเป็นการป้องกันอัตราผิดเนื่องจากการเหลื่อม (parallax errors) เพื่อช่วยในการนี้ควรขีดสเกลเฉพาะเลขหลักใหญ่ซ้ำอีกทางด้านที่อยู่ตรงข้าม

ไม้บรรทัดหยั่งวัดทำด้วยไม้สีดา (Cedar wood) หรือวัสดุที่เหมาะสมอื่นเช่น ไฟเบอร์กลาส ซึ่งมีคุณสมบัติไม่ดูดน้ำ ขีดสเกลทำไว้อย่างน้อยทุกๆ 10 มิลลิเมตร อัตราผิดสูงสุดของขีดสเกลต้องไม่เกิน 0.5 มิลลิเมตร

ในการวัดด้วยไม้บรรทัดหยั่งวัดนี้ ถ้าสามารถทำได้ให้เทียบกับการวัดด้วยกระบอกแก้วดวงด้วยก็จะได้

2.1.4 การติดตั้งเครื่องมือ (Exposure)

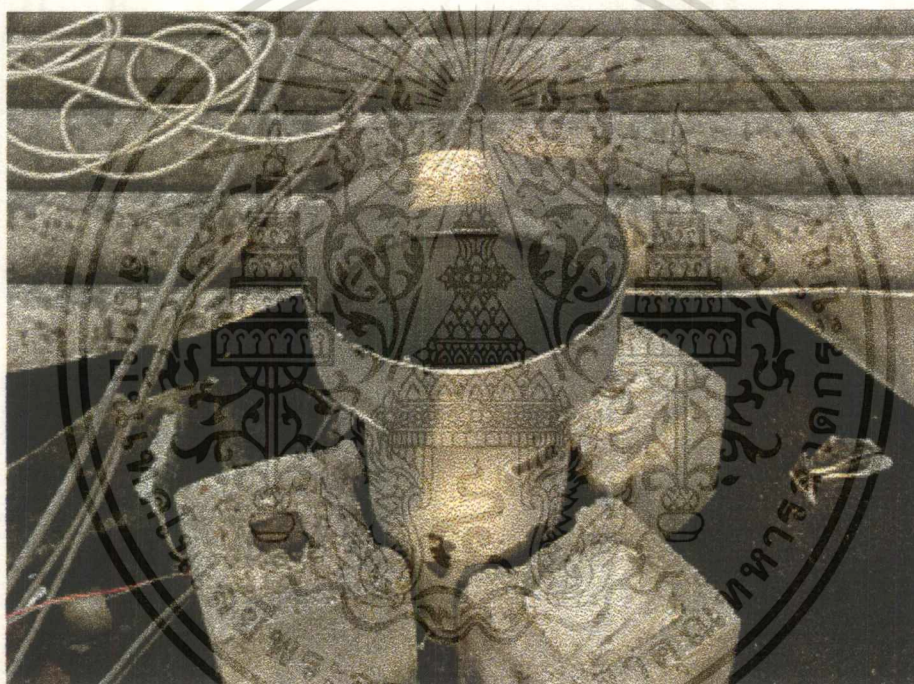
ปรากฏว่าจำนวนน้ำฝนที่จะเข้าไปในถังนั้นขึ้นอยู่กับลมประการหนึ่ง วิธีการในการติดตั้งเครื่องมือประการหนึ่ง และความสูงของบรรดาสิ่งที่อยู่แวดล้อมอีกประการหนึ่ง ค่าที่เปลี่ยนแปลงเหล่านี้เป็นเหตุมาจากลม และส่วนใหญ่เป็นลมหวนรอบๆ ถังวัดฝนนั้น กระแสอากาศไหลขึ้นทำให้ได้จำนวนฝนน้อยลง และกระแสอากาศไหลลงทำให้ได้จำนวนฝนมากกว่าความเป็นจริง ถ้ายังมีลมหวนอย่างแรงและบ่อยๆ

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

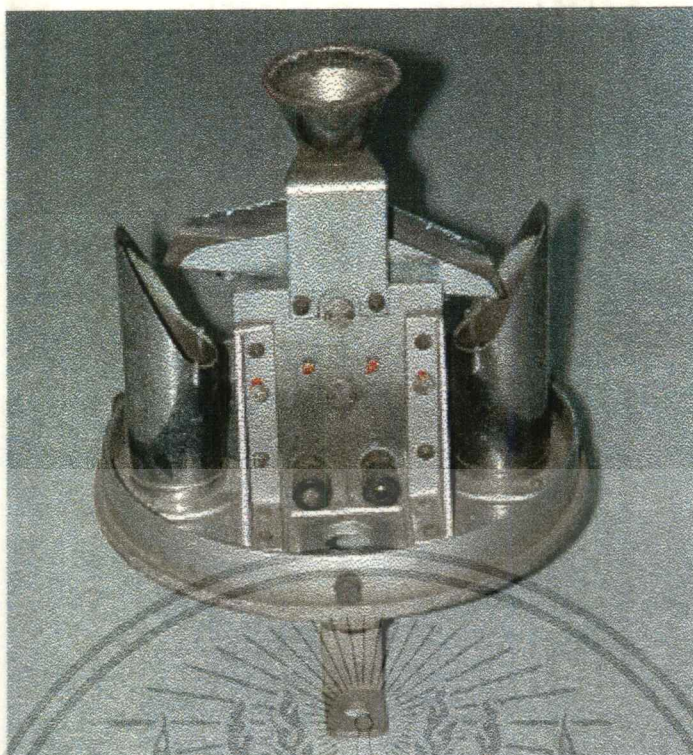
บ่อยๆ จำนวนฝนที่ได้ก็ยิ่งน้อยกว่าความเป็นจริงยิ่งขึ้น ความแรงและความถี่ของลมพายุนี้ขึ้นอยู่กับความแรงของลมอย่างหนึ่ง ลักษณะโดยทั่วไปของพื้นดินรอบๆ เครื่องวัดอย่างหนึ่ง ความห่างและความสูงของสิ่งแวดล้อม เครื่องวัดอย่างหนึ่งและความสูงของเครื่องซึ่งตั้งอยู่เหนือระดับพื้นดินอีกอย่างหนึ่ง ดังนั้น จึงจำเป็นต้องทำการเปรียบเทียบผลการตรวจของสถานีต่างๆ ด้วยว่าเป็นอย่างไร ในเมื่อสถานีต่างๆ เหล่านั้นได้ทำการติดตั้งเครื่องมือเป็นแบบเดียวกัน สำหรับประเทศไทยเรากำหนดให้ติดตั้งไว้บนพื้นดิน

2.1.5 เครื่องวัดปริมาณน้ำฝนแบบคานกระดก (Tilting Bucket)

เครื่องวัดปริมาณน้ำฝนแบบคานกระดกนี้ใช้เพื่อวัดปริมาณของน้ำฝนที่ตกลงมาซึ่งสามารถติดตั้งไว้ภายนอกอาคารได้ และทำการต่อด้วยสายเคเบิลเพื่อเข้าไปยังเครื่องบันทึกสัญญาณที่อยู่ภายในอาคารซึ่งวิธีนี้จะไม่ทำให้กระดกบันทึกสัญญาณเปียกได้



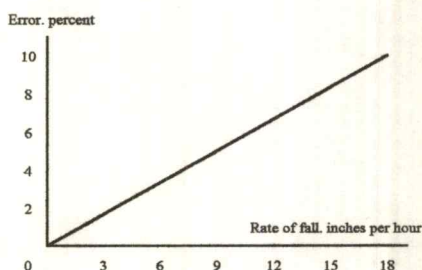
รูปที่ 2.1 แสดงเครื่องวัดน้ำฝนแบบคานกระดก



รูปที่ 2.2 กลไกของเครื่องวัดน้ำฝนแบบคานกระดก

หลักการที่นำมาใช้ของเครื่องวัดปริมาณน้ำฝนแบบคานกระดกนี้ใช้หลักการง่ายๆ คือจะมีอุปกรณ์ที่รองรับน้ำฝน ทำให้สมดุลและมีการขึ้นลงโดยสม่ำเสมอบนแกน มันจะถูกแบ่งออกเป็น 2 ส่วน เมื่อมีปริมาณน้ำฝนเข้าไปในปากถัง จะทำให้ที่รองรับด้านหนึ่งปริมาณน้ำเต็ม และเมื่อได้ปริมาณตามความจุจะทำให้คานกระดกไปยังด้านตรงข้าม เมื่อรับน้ำฝนจนเต็มจะกระดกกลับมายังด้านเดิมสลับกันไป โดยที่รองรับถูกออกแบบให้รับน้ำที่ตกลงมา 0.01 นิ้วซึ่งจะมีสกรู เอาไว้ใช้ในการปรับให้ระยะเวลาในการกระดกต่างๆ กัน อีกทั้งใช้สวิทช์ปรอทในการออกสัญญาณแล้วต่อกับเครื่องบันทึกเวลาในที่ที่ติดตั้งในอาคารซึ่งจะแก้ปัญหากระดกบันทึกเปิดได้ เครื่องวัดแบบนี้จะไม่ใช้ในบริเวณที่ฝนตกประปราย

การกระดกของที่รองรับน้ำฝนจะมีเวลาในการทำงานที่แน่นอนประมาณ 0.2 วินาที เมื่อน้ำเต็มข้างหนึ่งคานก็จะกระดกแล้วเทน้ำลงไป ขณะเดียวกันสวิทช์ปรอทก็จะทำการที่ปสวิทช์ทำให้เกิดสัญญาณพัลส์ขึ้นมาและจะเป็นฟังก์ชันเชิงเส้นกับอัตราการตกของฝนดังรูป ซึ่งอัตราการผิดพลาดนี้ไม่ค่อยจะมีผลอะไรมากนักยกเว้นในกรณีเมื่อมีฝนตกหนักกว่า 2 นิ้วต่อชั่วโมงหรือตกหนักมากๆ



รูปที่ 2.3 แสดงอัตราการผิดพลาดของเครื่องวัดปริมาณน้ำฝนแบบคานกระดก

เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์สำหรับการใช้ในเชิงวิชาการเท่านั้น เมื่อผู้ยืมได้เห็นว่าเอกสารนี้ไม่เกี่ยวข้องกับการศึกษา หรือการวิจัยใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.2 บอร์ดไมโครคอนโทรลเลอร์ รุ่น แอนท์ - 32 (ANT - 32)

2.2.1 คุณสมบัติของบอร์ด แอนท์ - 32

2.2.1.1 เป็นบอร์ดคอนโทรลใช้กับไมโครคอนโทรลเลอร์ตระกูล MCS - 51 (8031/8032) ใช้ CPU เบอร์ 80C32 ทำงานที่ความถี่สัญญาณนาฬิกา 11.0592 เมกะเฮิรตซ์

2.2.1.2 ใช้งานหน่วยความจำบนบอร์ดได้ 3 ตำแหน่งด้วยกัน คือ

เป็นหน่วยความจำโปรแกรม (program memory) ใช้กับ EPROM ขนาด 8 - 32

กิโลไบต์ เบอร์ 2764, 27128 หรือ 27256

เป็นหน่วยความจำข้อมูล (data memory) ใช้กับ RAM ขนาด 8 กิโลไบต์ เบอร์ 6264 หรือ 32 กิโลไบต์ เบอร์ 62256 สามารถแบกข้อมูลได้โดยใช้แบตเตอรี่ลิเทียม

เป็นความจำโปรแกรมและข้อมูล (program and data memory) ใช้กับ EPROM, RAM หรือ EEPROM ขนาด 8-32 กิโลไบต์ โดยใช้ EPROM เบอร์ 2764, 27256 ใช้ RAM เบอร์ 6264, 62256 หรือ EEPROM เบอร์ 2864(A), 28256(A)

2.2.1.3 มีพอร์ตอินพุต / เอาท์พุต เบอร์ 8255 จำนวน 2 ตัว (48 บิต) สำหรับต่อไปใช้งานภายนอก

2.2.1.4 มีพอร์ตแอลซีดี สำหรับการต่อการใช้งานกับแอลซีดีแบบดอทเมตริก (dotmatic)

2.2.1.5 มีวงจรการเชื่อมต่อแบบอนุกรมกับ RS232 (serial interface driver RS232) ด้วยชิพเบอร์ MAX232 สำหรับการต่อเข้ากับเครื่องไมโครคอมพิวเตอร์

2.2.1.6 มีวงจร วอตช์ด็อก ไทม์เมอร์ (watchdog time), เพาเวอร์อัพ / ดาวน์รีเซ็ต (power up/down reset) ด้วยชิพเบอร์ MAX691

2.2.1.7 มีวงจรนาฬิกาเวลาจริง ใช้ชิพเบอร์ DS1202

2.2.1.8 มีคอนเนคเตอร์สำหรับ พอร์ต 1 ของไมโครคอนโทรลเลอร์โดยเฉพาะ

2.2.1.9 มีคอนเนคเตอร์สำหรับระบบบัส

2.2.1.10 สามารถเลือกเบอร์ และชนิดหน่วยความจำ หรือกำหนดคุณสมบัติต่างๆของบอร์ดด้วยจัมเปอร์

2.2.2 การอ้างอิงตำแหน่งพอร์ต 8255 ของบอร์ดไมโครคอนโทรลเลอร์

เป็นชิพพอร์ตแบบขนานที่เป็นที่นิยมใช้กันมากมาย สำหรับบอร์ดนี้ใช้พอร์ต 8255 จำนวน 2 ตัว ทำหน้าที่เป็นพอร์ต ทำให้มีพอร์ตอินพุต/เอาต์พุตถึง $24 \times 2 = 48$ บิต โดยแบ่งเป็นพอร์ตใช้งาน 1 และ 2 โดยมีตำแหน่งแอดเดรสดังนี้

พอร์ตใช้งาน 1

พอร์ต A ตำแหน่งแอดเดรส $F800H + 00H = F800H$

พอร์ต B ตำแหน่งแอดเดรส $F800H + 01H = F801H$

พอร์ต C ตำแหน่งแอดเดรส $F800H + 02H = F802H$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

คอนโทรลพอร์ต ตำแหน่งแอดเดรส F800H + 03H = F803H

พอร์ตใช้งาน2

พอร์ต A ตำแหน่งแอดเดรส FC00H + 00H = FC00H

พอร์ต B ตำแหน่งแอดเดรส FC00H + 01H = FC01H

พอร์ต C ตำแหน่งแอดเดรส FC00H + 02H = FC02H

คอนโทรลพอร์ต ตำแหน่งแอดเดรส FC00H + 03H = FC03H

ก่อนที่จะใช้งานพอร์ต 8255 จะต้องทำการกำหนดโหมดการทำงานของพอร์ต A,B และ C ให้เป็นอินพุตหรือเอาต์พุต โดยทำการเขียนค่า คอนโทรลโค้ด ไปที่คอนโทรลพอร์ต

2.2.3 การอ้างอิง ตำแหน่งพอร์ต แอลซีดีโมดูล ของบอร์ดไมโครคอนโทรลเลอร์

บอร์ดไมโครคอนโทรลเลอร์ จะมีแอลซีดีพอร์ต ให้สำหรับการต่อใช้งานโดยสามารถต่อเข้ากับแอลซีดีโมดูลแบบคอตเมทริกได้ทันที ซึ่งจะใช้ขาสัญญาณทั้งหมด 14 ขา และสำหรับการใช้งานแอลซีดีพอร์ตเท่านั้น จะมีการจัดวงจรในแบบ เมโมรีแม็พ ซึ่งจะช่วยให้การเขียนโปรแกรมทำได้ง่าย โดยจะมองเห็นตำแหน่งต่าง ๆ ที่สรุปดังนี้

แอดเดรส	ลักษณะของพอร์ตที่ติดต่อ
FA00H	สำหรับเขียนคำสั่ง (RS = 0,R/W = 0)
FA01H	สำหรับอ่านค่าไม่ว่าง(RS = 0,R/W = 1)
FA02H	สำหรับเขียนข้อมูล(RS = 1,R/W = 0)
FA03H	สำหรับอ่านข้อมูล(RS = 1,R/W = 1)

การอ่านค่าแอลซีดี แบบ คอตเมทริก จะสามารถเลือกุ่นใด ๆ ก็ได้ โดยมีจำนวนตัวอักษรต่อบรรทัด และจำนวนบรรทัดตามที่ต้องการ เพราะสายสัญญาณที่จะใช้จะใช้แบบเดียวกันหมด จะแตกต่างกันที่โปรแกรมเท่านั้น การนับหมายเลขขั้วต่อของแอลซีดี พอร์ต จะไม่เหมือนการนับทั่ว ๆ ไป

2.2.4 นาฬิกาเวลาจริง และปฏิทิน (OPTIONAL REAL TIME CLOCK & CALENDAR)

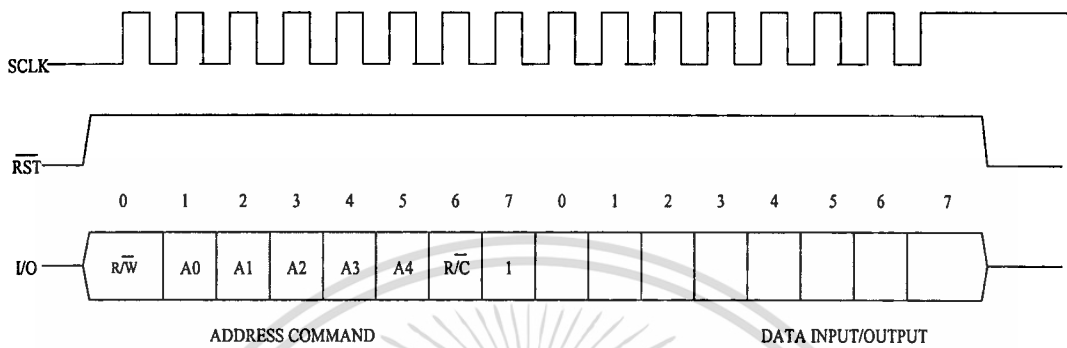
สำหรับการใช้งานระบบไมโครคอนโทรลเลอร์ที่มีเวลามาเกี่ยวข้องด้วย จำเป็นต้องมิงจรในส่วนที่ทำหน้าที่เป็นนาฬิกาเวลาจริง ซึ่งบอร์ดแอนท์ 32 ใช้ชิพเบอร์ DS1202 โดยต่อรวมกับอุปกรณ์ภายนอกเพียงเล็กน้อย และที่สำคัญคือ ไม่ต้องทำการปรับแต่ง ซึ่งเมื่อจะใช้นาฬิกาเวลาจริงนี้ ต้องมีชิพ DS1202 และ MAX691 รวมทั้ง คริสตอล 32.768 กิโลเฮิร์ซ และแบตเตอรี่ลิเทียมบนบอร์ดแอนท์-32

DS1202 ประกอบไปด้วยนาฬิกาเวลาจริง / ปฏิทิน และสแตตติสติก แรม ขนาด 24 ไบท์ ทำการอินเตอร์เฟสกับซีพียูในแบบอนุกรม โดยใช้สายเพียง 3 เส้นคือ ขารีสต ,ขาอินพุท/ เอาต์พุท (สำหรับข้อมูล) และ ขาคลิกอนุกรม (serial clock)

ขาสัญญาณทั้งสามนี้จะต่อเข้ากับขา P1.6,P1.4 และ P1.5 ของซีพียูตามลำดับ เมื่อต้องการทราบค่าเวลา จะต้องอ่านเวลาจากชิพ เพราะว่าชิพนี้ไม่มีขาสำหรับไปอินเตอร์รัปต์ ซีพียู ซีพียูสามารถเขียนหรืออ่านข้อมูลของนาฬิกาได้ 2 วิธี คือ ไบท์เดี่ยว (single byte) และหลายไบท์ (multiple byte) โดยทั้งสองวิธีซีพียู ต้องส่งไบท์คำสั่ง (8 บิต) ให้ DS1202 เพื่อบอกทราบว่าต้องการเขียนหรืออ่านนาฬิกา พร้อมตำแหน่ง และตามด้วยข้อมูลในขณะ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปเผยแพร่โดยไม่ได้รับอนุญาต
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ที่กำลังติดต่อกับ DS1202 สัญญาณที่ขารีเซตต้องเป็นลอจิก 1 ขา คล็อกอนุกรมจะเป็นสัญญาณ เพื่อทำการเขียนหรืออ่านข้อมูลโดยจะใช้สัญญาณ คล็อก 1 ลูกสำหรับข้อมูล 1 บิต ส่วนขา อินพุต/เอาต์พุต เป็นข้อมูลอนุกรม โดยจะเป็นอินพุตเมื่อทำการเขียนและเป็นเอาต์พุตเมื่อทำการอ่าน โดยข้อมูลที่จะเขียนหรืออ่านนี้จะเริ่มจากบิต 0 และจบด้วยบิต 7 ค่าของไบท์คำสั่ง



รูปที่ 2.4 รูปแสดงไคอะแกรมเวลาของการส่งข้อมูลเพื่อติดต่อกับ DS 1202 แบบไบท์เดี่ยว



COMMAND BYTES/DEFINITION													
REGISTER	FUNCTION	COMMAND	WRITE=W	RANGE	REGISTER DEFINITION								
		ADDRESS (HEX)	READ=R	DATA (BCD)	7	6	5	4	3	2	1	0	
0	SECONDS	80	W	00-59	CH	10 SEC				SEC			
		81	R										
1	MINUTES	82	W	00-59	0	10 MIN				MIN			
		83	R										
2	12 HRS	84	W	01-12	12\	0	AP	HR	HOUR				
	24 HRS	85	R	00-23	24	0	10	HR					
3	DATE	86	W	01-31	0	0	10 DATE			DATE			
		87	R										
4	MONTH	88	W	01-12	0	0	0	10M	MONTH				
		89	R										
5	DAY	8A	W	01-07	0	0	0	0	DAY				
		8B	R										
6	YEAR	8C	W	00-99	10 YEAR				YEAR				
		8D	R										
7	WRITE	8E	W	00-80	WP	ALWAYS ZERO							
	PROTECT	8F	R										

ตารางที่ 2.1 ตารางแสดงไบต์คำสั่งในการเขียนและอ่านค่านาฬิกา

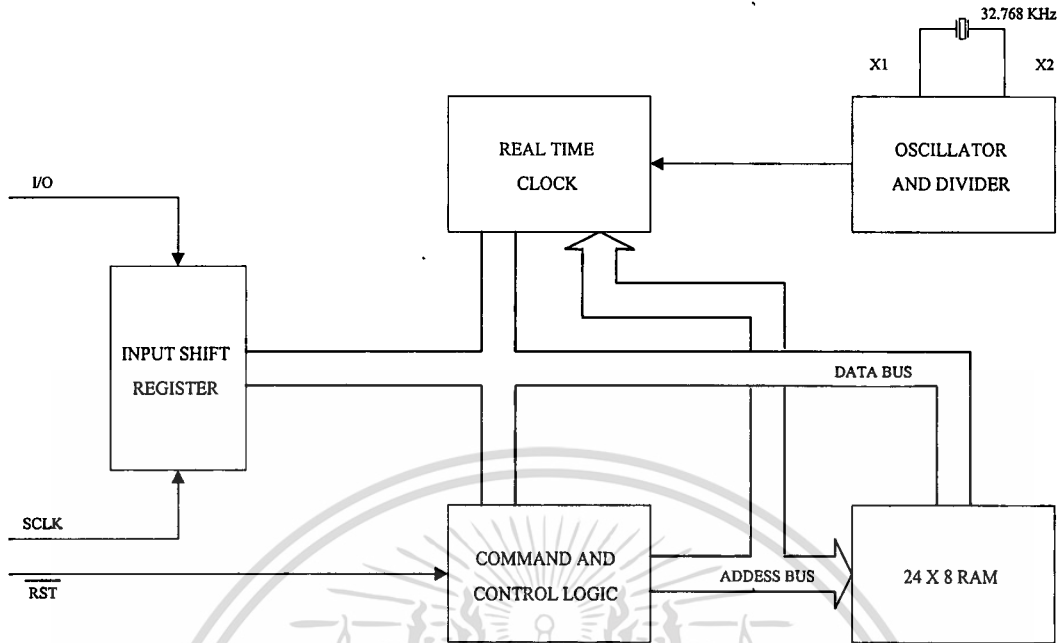
การทำงาน

จะมีการทำงานตามบล็อกไดอะแกรม ดังรูป โดยมี ชิพรีจิสเตอร์, คอนโทรลลอจิก,ออสซิลเลเตอร์ ,นาฬิกาเวลาจริง และหน่วยความจำสำรอง การเริ่มต้นต้นของการส่งผ่านข้อมูลรีเซตจะทำให้เป็นลอจิก 1 และ 8 บิต ถูกนำเข้าไปในชิพรีจิสเตอร์ซึ่งจะทำการแปลงทั้งแอดเดรสและคำสั่ง โดยที่ข้อมูลที่เป็นอินพุตจะเป็นแบบอนุกรม ใช้ 8 บิตแรกของ 32 ไบต์เป็นทางเข้า โดยไม่คำนึงว่ารอบการอ่าน,เขียนจะเกิดขึ้นหรือไม่หลังจากนั้น 8 คล็อกแรก จะเกิดขึ้นและจะนำคำสั่งไปยังชิพรีจิสเตอร์ โดยจะใช้กับข้อมูลออก เมื่อมีการอ่านหรือข้อมูลเข้า เมื่อมีการเขียน

2.3 การอินเตอร์รัปต์

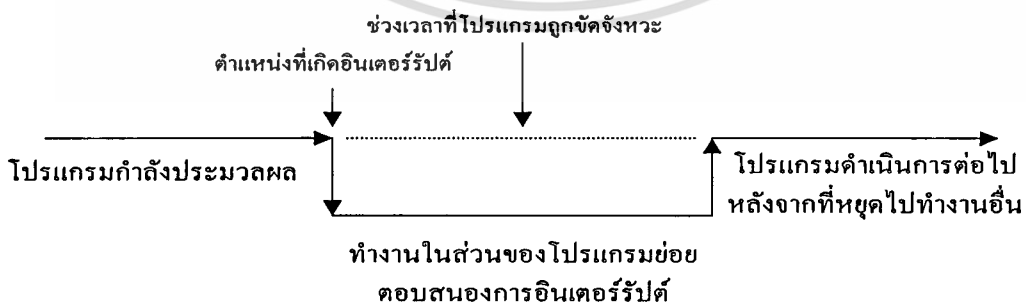
โดยปกติแล้วในการให้บริการหรือควบคุมอุปกรณ์อินพุตและเอาต์พุตต่าง ๆ ของระบบไมโครคอนโทรลเลอร์ มักจะใช้การวนรอบในช่วงระยะเวลาที่กำหนดเพื่อตรวจสอบสภาพของการติดต่อ แม้ว่าอุปกรณ์นั้น ๆ จะไม่ต้องการการติดต่อใด ๆ ก็ตามซึ่งทำให้เสียเวลาในการประมวลผลกลางไปโดยเปล่าประโยชน์ การนำเทคนิคของการอินเตอร์รัปต์เข้ามาช่วยในการดำเนินการนั้นนอกจากจะทำให้ระบบมีการตอบสนองต่อเหตุการณ์ได้รวดเร็วยิ่งขึ้นแล้ว ยังมีผลทำให้ลดความซับซ้อนของโปรแกรมลงได้อย่างมากด้วย เนื่องจากไม่จำเป็นต้องมีส่วนของโปรแกรมที่ทำการวนตรวจนี้อีกต่อไป

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.5 บล็อกไดอะแกรม DS 1202

เนื่องจากลักษณะการร้องขอการอินเตอร์รัปต์ของอุปกรณ์ต่าง ๆ เข้ามายังไมโครคอนโทรลเลอร์นั้น สามารถเกิดขึ้นได้ตลอดเวลา ไม่ว่ากำลังประมวลผลโปรแกรมอยู่ในส่วนใดของโปรแกรมก็ตาม ดังนั้นเมื่อโปรแกรมถูกอินเตอร์รัปต์ ก็จะดำเนินการคำสั่งที่กำลังทำอยู่ให้เสร็จสิ้น หลังจากนั้นไมโครคอนโทรลเลอร์จึงตอบรับการอินเตอร์รัปต์นั้น และควบคุมให้ไปดำเนินการประมวลผลโปรแกรมยังตำแหน่งหนึ่ง เพื่อให้บริการต่อการอินเตอร์รัปต์นั้น ซึ่งยังจะมีผลทำให้สถานะการรับรู้ว่ามีกรอินเตอร์รัปต์ทางด้านฮาร์ดแวร์ของไมโครคอนโทรลเลอร์เปลี่ยนแปลงและกลับไปเป็นสู่สถานะเช่นปกติ



รูปที่ 2.6 แสดงให้เห็นถึงแผนภาพการทำงานของโปรแกรม

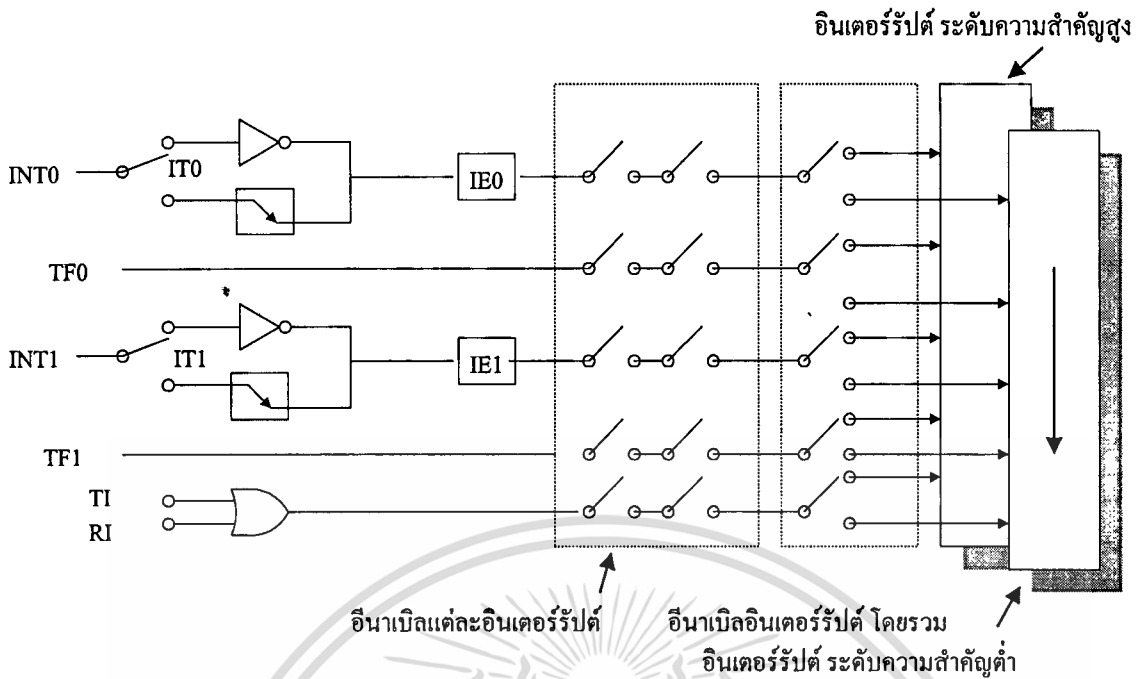
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากรูปซึ่งโดยปกติจะทำงานอย่างใดอย่างหนึ่งอยู่ เมื่อมีการอินเทอร์รัปต์เกิดขึ้นส่วนนี้จะหยุดค้าง เนื่องจากโปรเซสเซอร์จะทำการกระโดดไปทำงานในส่วนของโปรแกรมย่อยบริการอินเทอร์รัปต์ เมื่อเสร็จสิ้นงานในโปรแกรมย่อยนี้แล้วจึงจะได้กลับไปทำงานเก่าที่ค้างอยู่ต่อไป

การจำแนกประเภทของสัญญาณที่ทำให้เกิดการอินเทอร์รัปต์กับไมโครคอนโทรลเลอร์ 8051 ใช้การพิจารณาจากแหล่งที่มาของสัญญาณอินเทอร์รัปต์นั้น ได้แก่ สัญญาณอินเทอร์รัปต์ภายนอก ซึ่งมีที่มาจากวงจรหรืออุปกรณ์ภายนอกโดยสัญญาณจะผ่านเข้ามาทางขาสัญญาณของไมโครคอนโทรลเลอร์ สำหรับสัญญาณอินเทอร์รัปต์ภายในนั้น จะเป็นการสร้างสัญญาณจากวงจรภายในของไมโครคอนโทรลเลอร์เอง

2.3.1 โครงสร้างการอินเทอร์รัปต์ของ 8051

จากแผนภาพโครงสร้างระบบอินเทอร์รัปต์ของ 8051 ในรูป แสดงให้เห็นว่าสัญญาณที่เข้ามาทำการอินเทอร์รัปต์ 8051 นั้นเกิดขึ้นได้ห้าลักษณะตามตารางข้อมูล โดยจะเห็นได้ว่าสามารถที่จะกำหนดเลือกเพื่อยินยอม (หรือ อินาเบิล) และห้าม (หรือ ดิสเอนเบิล) ไม่ให้มีการอินเทอร์รัปต์แต่ละประเภทได้ โดยการกำหนดบิตของข้อมูลที่เกี่ยวข้องซึ่งมักจะอยู่ภายในรีจิสเตอร์ TCON และ SCON นอกจากนี้ยังมีตำแหน่งบิตภายในรีจิสเตอร์ IE (interrupt enable register) ซึ่งทำหน้าที่เสมือนกับเป็นสวิตช์ หลักที่เกี่ยวข้องกับสัญญาณอินเทอร์รัปต์ทั้งหมด โดยหากว่ากำหนดให้ห้ามการเกิดอินเทอร์รัปต์แล้ว การกำหนดบิตเพื่อห้ามหรือยินยอมของแต่ละอินเทอร์รัปต์ก็จะมีผลใด ๆ เกิดขึ้น นอกจากนี้แผนภาพในรูปยังแสดงให้เห็นด้วยว่าสัญญาณอินเทอร์รัปต์แต่ละประเภทยังสามารถกำหนด ระดับความสำคัญ ของการอินเทอร์รัปต์ได้สองลักษณะ คือ ระดับความสำคัญสูงหรือต่ำ กล่าวคือ ขณะที่กำลังประมวลผลอยู่ภายในส่วนของโปรแกรมย่อยบริการอินเทอร์รัปต์ของสัญญาณที่มีระดับความสำคัญต่ำอยู่ ก็อาจจะถูกขัดจังหวะให้ไปประมวลผลของสัญญาณอินเทอร์รัปต์ ที่มีระดับความสำคัญสูงกว่าได้ แต่หากว่าเป็นสัญญาณอินเทอร์รัปต์ที่ระดับความสำคัญต่ำเช่นเดียวกันแล้ว ก็จะต้องรอให้เสร็จสิ้นการประมวลผลที่ดำเนินการอยู่ก่อน



รูปที่ 2.7 แผนภาพแสดงโครงสร้างระบบอินเทอร์รัปต์ของ 8051

สัญญาณ	ความหมาย
INT0	สัญญาณอินเทอร์รัปต์ภายนอกเข้ามาหาสัญญาณ P3.2
INT1	สัญญาณอินเทอร์รัปต์ภายนอกเข้ามาหาสัญญาณ P3.3
Timer 0	สัญญาณอินเทอร์รัปต์ภายใน ที่เกิดจากวงจรนับ/จับเวลา Timer 0
Timer 1	สัญญาณอินเทอร์รัปต์ภายใน ที่เกิดจากวงจรนับ/จับเวลา Timer 1
พอร์ตอนุกรม	สัญญาณอินเทอร์รัปต์ภายใน และภายนอก ที่เกี่ยวข้องกับการส่งข้อมูล อนุกรมจากวงจรภายใน และรับข้อมูลอนุกรมจากภายนอกตามลำดับ

ตารางที่ 2.2 สัญญาณอินเทอร์รัปต์ของ 8051 ทั้ง 5 ประเภท

2.3.2 การควบคุมการอินเทอร์รัปต์

ตามโครงสร้างด้านการจัดการอินเทอร์รัปต์ของ 8051 สามารถกำหนดเลือกเพื่อยินยอมหรือไม่ยินยอม ให้มีการอินเทอร์รัปต์ของแต่ละสัญญาณได้ โดยใช้การกำหนดค่าของบิตภายใน รีจิสเตอร์ IE ซึ่งจะมีทั้งแบบที่ระบุถึงอินเทอร์รัปต์โดยรวมทั้งหมด (บิตที่ 7) และอินเทอร์รัปต์แต่ละประเภทได้ ในกรณีที่กำหนดค่าข้อมูลเป็น 1 ให้กับจะมีความหมายถึงการยินยอมให้มีการอินเทอร์รัปต์เกิดขึ้นได้ โดยจะต้องทำการกำหนดให้ยินยอมให้มีการอินเทอร์รัปต์ทั้งหมดให้เกิดขึ้นก่อน จึงจะมีผลทำให้การกำหนดบิตเพื่อยินยอมของแต่ละอินเทอร์รัปต์มีผลขึ้นได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

EA		ET2	ES	ET1	EX1	ET0	EX0
----	--	-----	----	-----	-----	-----	-----

ชื่อบิต	ตำแหน่ง	ความหมาย
EA	IE.7	อินาเบิล/คิสเอเบิล การเกิดอินเตอร์รัปต์ โดยรวม
	IE.6	
ET2	IE.5	อินาเบิล/คิสเอเบิล การเกิดอินเตอร์รัปต์ Timer2
ES	IE.4	อินาเบิล/คิสเอเบิล การเกิดอินเตอร์รัปต์ พอร์ตอนุกรม
ET1	IE.3	อินาเบิล/คิสเอเบิล การเกิดอินเตอร์รัปต์ Timer1
EX1	IE.2	อินาเบิล/คิสเอเบิล การเกิดอินเตอร์รัปต์ INT1
ET0	IE.1	อินาเบิล/คิสเอเบิล การเกิดอินเตอร์รัปต์ Timer0
EX0	IE.0	อินาเบิล/คิสเอเบิล การเกิดอินเตอร์รัปต์ INT0

ตารางที่ 2.3 แสดงบิตต่างๆ ภายในรีจิสเตอร์ IE

2.3.3 การจัดการอินเตอร์รัปต์

เมื่อมีการอินเตอร์รัปต์เกิดขึ้น ไมโครคอนโทรลเลอร์จะทำคำสั่งที่กำลังดำเนินการอยู่ให้แล้วเสร็จ จากนั้นจึงจะทำการเก็บค่าตำแหน่งแอดเดรสของคำสั่งที่จะทำงานต่อไปไว้ยังบริเวณของหน่วยความจำที่ถูกกำหนดไว้ให้เป็นสแต็ก และกระโดดไปยังตำแหน่งแอดเดรสที่ได้มีการกำหนดไว้แน่นอนตำแหน่งหนึ่งโดยอัตโนมัติ ตำแหน่งนี้เรียกว่า แอดเดรสของอินเตอร์รัปต์ (interrupt vector address) ซึ่งจะต้องทำการเขียนโปรแกรมย่อย (subroutine) ที่ตำแหน่งแอดเดรสเหล่านี้ไว้ ซึ่งเรียกว่า โปรแกรมย่อยบริการอินเตอร์รัปต์

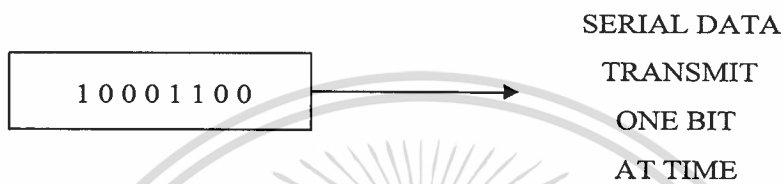
แหล่งกำเนิดสัญญาณอินเตอร์รัปต์	ตำแหน่งแอดเดรส (ฐานสิบหก)
อินเตอร์รัปต์ภายนอก 0	0003
วงจรรนับ/จับเวลา 0	0008
อินเตอร์รัปต์ภายนอก 1	0013
วงจรรนับ/จับเวลา1	001B
วงจรรับส่งข้อมูลอนุกรม	0023

ตารางที่ 2.4 แสดงตำแหน่งอินเตอร์รัปต์เวกเตอร์ของสัญญาณอินเตอร์ต่างๆ

2.4 การรับส่งข้อมูลแบบอนุกรม

การรับส่งข้อมูลโดยทั่วไปนั้น ข้อมูลทุกๆ บิตจะถูกรับหรือส่งออกไปในเวลาเดียวกัน เช่นการอ่านหรือการเขียน ข้อมูลลงในหน่วยความจำ ซึ่งเราเรียกการรับส่งข้อมูลในลักษณะนี้ว่า “การรับส่งข้อมูลแบบขนาน” (parallel communication)

สำหรับในที่นี้จะกล่าวถึงการรับส่งข้อมูลอีกรูปแบบหนึ่ง ซึ่งเป็นการรับส่งข้อมูลที่ละบิตแทนที่จะทำการส่งข้อมูลพร้อมกันทุกบิตในเวลาเดียวกัน การรับส่งข้อมูลแบบนี้มีชื่อว่า “การรับส่งข้อมูลแบบอนุกรม” (serial communication)



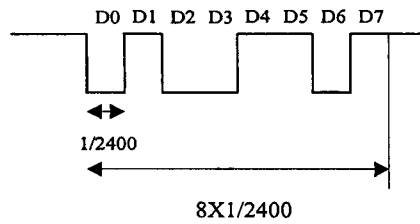
รูปที่ 2.8 แสดงการส่งข้อมูลแบบอนุกรม ซึ่งจะทำการส่งข้อมูลที่ละบิต

สำหรับการรับส่งข้อมูลแบบขนานนั้น ถึงแม้ว่าจะมีความเร็วสูงกว่าอยู่มากก็ตาม แต่ต้องใช้จำนวนสายในการส่งผ่านข้อมูลเป็นจำนวนมากกว่าแบบอนุกรมทำให้สิ้นเปลืองค่าใช้จ่ายในการวางสายโดยไม่จำเป็น และยังมีการรบกวนของสัญญาณมากกว่าแบบอนุกรมอีกด้วย ทำให้เกิดความผิดพลาดในการรับส่งผ่านข้อมูลขึ้นได้ง่าย ดังนั้นในการส่งผ่านข้อมูลในระยะทางไกลๆ เรามักจะเลือกใช้การส่งข้อมูลแบบอนุกรม เพื่อลดจำนวนของสายส่งซึ่งจะช่วยในการลดค่าใช้จ่ายในการวางสายลงได้มาก ถึงแม้การรับส่งข้อมูลแบบนี้จะมีความยุ่งยากและช้ากว่าการรับส่งข้อมูลแบบขนานอยู่บ้างก็ตาม

2.4.1 อัตราความเร็ว (BAUD RATE)

สิ่งที่สำคัญมากสิ่งหนึ่งในการรับส่งข้อมูลแบบอนุกรมนี้คือ ความถี่ที่ใช้ในการส่งข้อมูลซึ่งจะต้องสัมพันธ์กันระหว่างอุปกรณ์ที่ทำกรรับและส่งข้อมูล ความถี่ที่ใช้ในนี้มีชื่อเรียกว่า “อัตราความเร็ว” ซึ่งมีความหมายถึง “อัตราการรับส่งข้อมูลเป็นจำนวนบิตใน 1 วินาที” ถ้าหากว่าเครื่องส่งใช้ อัตราความเร็วที่ไม่สัมพันธ์กับเครื่องแล้ว ก็จะทำให้การรับส่งข้อมูลเกิดผิดพลาดขึ้นได้

โดยทั่วไปค่าของอัตราความเร็วนั้นจะใช้ค่าต่างๆ ดังต่อไปนี้คือ 110, 150, 300, 600, 1200, 2400, 4800 และ 9600 บิตต่อวินาที สำหรับในที่นี้จะสมมุติว่าต้องการที่จะส่งข้อมูลแบบอนุกรมด้วยอัตรา 2400 baud (2400 บิตต่อวินาที) และข้อมูลที่ต้องการจะส่งก็คือ 0B2H หรือ 10110010B ซึ่งเราสามารถที่จะแสดงได้ในรูปของสัญญาณดังรูป



รูปที่ 2.9 แสดงสัญญาณของข้อมูลที่ถูกส่งไปตามสายส่งสัญญาณ

จากรูปเห็นได้ว่าความกว้างของสัญญาณของแต่ละบิต จะมีค่าเท่ากับ $1/\text{baud rate}$ วินาที ซึ่งจากอัตราความเร็วที่ต้องการใช้คือ 2400 บิตต่อวินาทีนั้น จะทำให้ความกว้างของแต่ละบิตที่จะส่งไปตามสายส่งนี้ทำให้เราสามารถที่จะคำนวณเวลาที่จะต้องใช้ในการรับ ส่งข้อมูลแต่ละไบต์ (8 บิต) ได้เท่ากับ 8×416 ไมโครวินาที หรือ 3328 ไมโครวินาที อย่างไรก็ตามเพื่อป้องกันความผิดพลาดที่อาจเกิดขึ้นได้ จึงมีการเพิ่มบิตต่าง ๆ ลงไปในไบต์ของข้อมูล เพื่อช่วยในการตรวจสอบความถูกต้องของข้อมูลที่เครื่องรับได้รับมา (แต่ไม่ได้หมายความว่าเมื่อเพิ่มบิตต่างๆ เหล่านี้เข้าไปแล้ว จะทำให้การส่งผ่านข้อมูลมีความถูกต้อง 100%) สำหรับบิตต่างๆ ที่เพิ่มเข้ามานี้ก็คือ บิตเริ่มต้น (start bit) บิตหยุด (stop bit) และพริตตี้บิต (parity bit) ซึ่งจะทำให้ข้อมูลในแต่ละไบต์ที่ส่งออกไปนี้มีมากกว่า 8 บิต และเวลาที่ใช้ในการรับส่งข้อมูลก็จะมากขึ้นตามไปด้วย

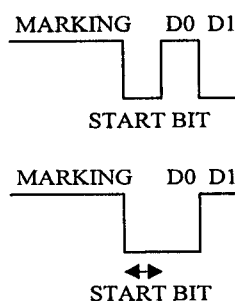
2.4.2 บิตเริ่มต้น

ในการส่งผ่านข้อมูลแบบอนุกรมนี้ เราจำเป็นต้องทำให้อุปกรณ์ที่จะรับข้อมูลทราบว่าข้อมูลที่ส่งมานั้นเริ่มต้นที่จุดใด ดังนั้นเราจึงจำเป็นต้องเพิ่มข้อมูล 1 บิต ลงไปก่อนหน้าข้อมูลจริง (actual data) ที่จะทำการส่ง (การส่งอนุกรมจะส่งบิต D0 เป็นบิตแรก และบิต D7 เป็นบิตสุดท้าย) นั่นก็คือทำการเพิ่มบิตนี้ลงไปข้างหน้าบิต D0 นั่นเอง และเรียกบิตนี้ว่า “บิตเริ่มต้น”

หน้าที่ของบิตเริ่มต้นนั้นนอกจากจะใช้ในการบอกว่าข้อมูลนั้นเริ่มต้นที่ใดแล้ว ยังทำงานร่วมกับบิตหยุด (ซึ่งจะกล่าวถึงต่อไป) เพื่อช่วยในการแยกข้อมูลแต่ละชุดออกจากกันและความกว้างของบิตนี้จะเท่ากับ ความกว้างของบิตอื่นๆ ในข้อมูลที่จะส่ง (D0 - D7)

เมื่ออุปกรณ์ที่จะส่งข้อมูลไปยังไม่ได้ทำการส่งข้อมูลใดๆ ออกมานั้น สายส่งจะอยู่ในสถานะที่เรียกว่า “มาร์คกิ้ง (MARKING)” ซึ่งเป็นสถานะที่ไม่มีมีการรับส่งข้อมูลใดๆ เกิดขึ้น ในที่นี้เราจะสมมุติให้มาร์คกิ้งของสายส่งเป็นลอจิก “1” บิตเริ่มต้นที่จะเพิ่มเข้าไปนี้จะมีลอจิกที่ตรงข้ามกับลอจิกของมาร์คกิ้ง ดังนั้นในกรณีนี้ บิตเริ่มต้นจะมีลอจิกเป็น “0”

สำหรับบิตเริ่มต้นนี้จะมีค่าความกว้างเท่ากับ 1 บิตของข้อมูล เช่น ใน 1 บิตของข้อมูลมีความยาวเท่ากับ 413 ไมโครวินาที บิตเริ่มต้นก็จะมีค่าความกว้างของสัญญาณเท่ากับ 413 ไมโครวินาทีด้วย ในรูปที่จะแสดงให้เห็นถึงบิตเริ่มต้นที่เพิ่มเข้าไปก่อนหน้าข้อมูล (ก่อนหน้า D0)



รูปที่ 2.10 แสดงการเพิ่มบิตเริ่มต้นเข้าไปก่อนหน้าบิต D0 ในกรณีที่บิต D0 เป็น “1” และ “0” ตามลำดับ

2.4.3 พาริตีบิต

สำหรับบิตนี้จะทำหน้าที่ในการบอกให้ส่วนรับข้อมูลทราบว่า ข้อมูลที่ได้รับเข้ามานั้นมีความถูกต้องเหมือนกับข้อมูลที่ถูกต้องออกมาหรือไม่ (ถึงแม้ว่าการตรวจสอบบิตนี้จะไม่พบความผิดพลาด แต่ก็ไม่ได้หมายความว่าข้อมูลที่รับเข้ามานี้จะถูกต้อง 100%) โดยที่บิตนี้จะทำหน้าที่ในการบอกให้ส่วนรับข้อมูลทราบว่า ข้อมูลที่ส่งออกมาในแต่ละไบนารีนั้นมีจำนวนบิตที่เป็น “1” อยู่เป็นจำนวนกี่ เป็นต้น สำหรับบิตที่ใช้ในการตรวจสอบนี้เรียกว่า “พาริตีบิต”

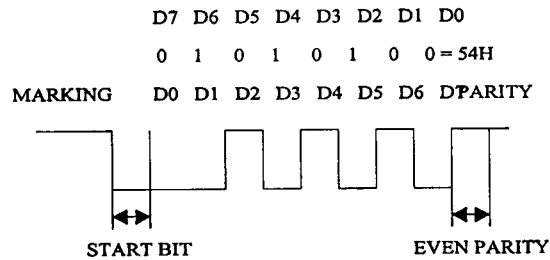
พาริตีบิตนี้จะถูกส่งออกมาโดยอุปกรณ์ส่งข้อมูล ซึ่งบิตนี้จะ เป็น “1” หรือ “0” นั้นขึ้นอยู่กับข้อมูลที่ส่งออกมา (D0 - D7) ว่ามีจำนวนบิตที่เป็น “1” เป็นจำนวนคู่หรือคี่ และยังขึ้นอยู่กับอุปกรณ์รับส่งข้อมูลด้วยว่าถูกออกแบบ (โปรแกรม) ไว้ให้รับส่งพาริตีในลักษณะของพาริตีคู่ หรือ พาริตีคี่ อีกด้วย

ในกรณีที่อุปกรณ์รับส่งข้อมูลถูกออกแบบไว้ให้เป็นพาริตีคู่ อุปกรณ์ส่งข้อมูลจะทำการส่งพาริตีบิตเป็นลอจิก “1” ออกไป เมื่อจำนวนบิตที่เป็น “1” ของข้อมูล (D0 - D7) เป็นจำนวนคี่ และจะทำการส่งพาริตีบิตเป็นลอจิก “0” เมื่อจำนวนบิตที่เป็น “1” ของข้อมูลเป็นจำนวนคู่ (คือจะทำให้จำนวนบิตที่เป็น “1” ของข้อมูล (D0 - D7) รวมกับพาริตีบิตแล้วเป็นจำนวนคู่นั่นเอง) สำหรับพาริตีคี่ก็เช่นกัน คือ พาริตีบิตจะเป็น “1” ในกรณีที่จำนวนบิตที่เป็น “1” ของข้อมูลเป็นจำนวนคู่ และจะเป็น “0” ในกรณีที่จำนวนบิตที่เป็น “1” ของข้อมูลเป็นจำนวนคี่ ในที่นี้จะสมมุติว่าอุปกรณ์ถูกออกแบบไว้สำหรับพาริตีคู่และเราต้องการที่จะส่งข้อมูลออกไปให้กับส่วนรับข้อมูลเป็นจำนวน 2 ไบนารี คือ 54H และ 55H

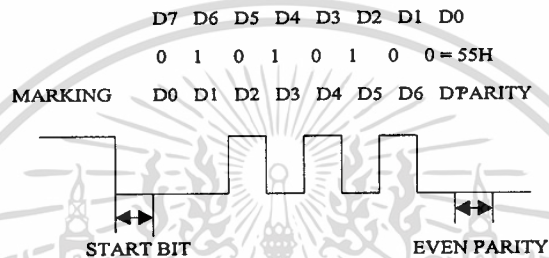
เมื่อเราส่งข้อมูล 54H ออกไป ซึ่งมีจำนวนบิตที่เป็น “1” ของข้อมูลเป็นจำนวนคู่ และจำเป็น “0” ในกรณีที่จำนวนบิตที่เป็น “1” ของข้อมูลเป็นจำนวนคี่ ในที่นี้จะสมมุติว่าอุปกรณ์ถูกออกแบบไว้สำหรับพาริตีคู่และเราต้องการที่จะส่งข้อมูลออกไปให้กับส่วนรับข้อมูลเป็นจำนวน 2 ไบนารี คือ 54H และ 55H

เมื่อเราส่งข้อมูล 54H ออกไป ซึ่งจำนวนบิตที่เป็น “1” เป็นจำนวนคี่ ดังนั้นในกรณีนี้อุปกรณ์ส่งข้อมูลก็จะทำการส่งพาริตีบิตเป็นลอจิก “1” ตามออกมาด้วย เพื่อจะให้บิตที่เป็น “1” ของข้อมูล 54H รวมกับพาริตีบิตแล้วได้เป็นจำนวนคู่ ส่วนข้อมูล 55H นั้นจำนวนบิตที่เป็น “1” นั้นเป็นจำนวนคู่อยู่แล้วดังนั้นอุปกรณ์ส่งข้อมูลก็จะส่งพาริตีบิตเป็น “0” ให้กับส่วนรับข้อมูล ดังในรูปที่ 2.12 และรูปที่ 2.13 สำหรับส่วนรับข้อมูลนั้น เมื่อทำการรับข้อมูลเข้ามาแล้วก็จะตรวจสอบสัญญาณว่า จำนวนบิตที่เป็น “1” ของข้อมูล รวมกับพาริตีบิตนั้นเป็นจำนวนคู่หรือคี่ ถ้าหากว่าเป็นจำนวนคี่ก็แสดงว่า ข้อมูลที่ได้รับเข้ามามีความผิดพลาดเกิดขึ้น (แต่ไม่ได้หมายความว่า ถ้าเป็นจำนวนคู่แล้ว ข้อมูลที่รับเข้ามาจะถูกต้องเสมอไป)

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.11 แสดงการเพิ่มพาริตี เป็น “1” ลงไปในข้อมูลแต่ละไบต์

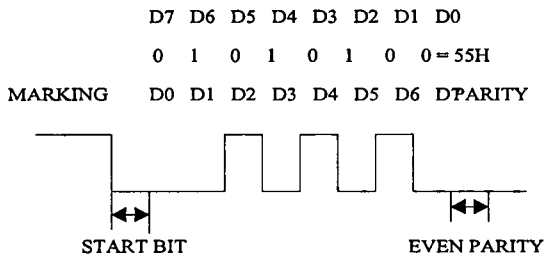


รูปที่ 2.12 แสดงการเพิ่มพาริตีที่เป็น “0” ลงไปในข้อมูลแต่ละไบต์

สิ่งสำคัญอีกสิ่งหนึ่งก็คือ ถ้าอุปกรณ์ส่งข้อมูลทำการส่งในลักษณะพาริตีคู่ หรือคี่ก็ตาม ส่วนรับข้อมูลก็ต้องทำการรับในลักษณะเดียวกับอุปกรณ์ส่งข้อมูลด้วย เช่นในกรณีที่อุปกรณ์ส่งข้อมูลทำการส่งข้อมูลในลักษณะของพาริตีคู่ อุปกรณ์รับข้อมูลก็ต้องทำการรับข้อมูลในลักษณะของพาริตีคู่ด้วย เป็นต้น

2.4.4 บิทหยุด

สำหรับบิทสุดท้ายที่เพิ่มเข้าไปนี้ จะใช้ในการตรวจสอบจุดสิ้นสุดของข้อมูล บิทนี้จะถูกเพิ่มเข้าไปหลังพาริตีบิท ถ้าอุปกรณ์รับข้อมูลบิตตรวจไม่พบบิทนี้ก็แสดงว่า ข้อมูลที่ได้รับเข้ามานั้นมีความผิดพลาดเกิดขึ้น สำหรับบิทหยุดนี้อาจมีจำนวนบิทเป็น 1, 1.5, 2 บิทก็ได้ รูปที่ 2.14 จะแสดงข้อมูลทั้ง 8 บิทที่ส่งออกมารวมทั้ง บิทเริ่มต้น บิทหยุด และ พาริตีบิท ด้วย ซึ่งจะเห็นว่า สิ่งที่ส่งออกมาในแต่ละไบต์นั้นไม่ได้มีเพียงข้อมูล 8 บิทเท่านั้น แต่อาจจะได้ถึง 12 บิท (กรณีที่ส่งบิทหยุดออกมา 2 บิท) ดังนั้น ถ้าเราทำการส่งด้วยอัตรา 2400 BAUD เราจะต้องใช้เวลาในการส่งทั้งหมดเท่ากับ 12×416 ไมโครวินาที หรือ 4099 มิลลิวินาที ไม่ใช่ 3328 ไมโครวินาที ดังที่คำนวณไว้ในตอนต้น



รูปที่ 2.13 แสดงการเพิ่มบิตต่างๆ ที่ใช้สำหรับป้องกันการผิดพลาด
ที่จะเกิดขึ้นในการรับส่งข้อมูลแบบอนุกรม

2.4.5 การรับและส่งข้อมูลแบบอนุกรมผ่าน 8032

สำหรับไมโครคอนโทรลเลอร์ 8032 สามารถรับและส่งข้อมูลแบบ UNIVERSAL ASYNCHRONOUS RECEIVE TRANSMITTER (UART) แบบฟูลดูเพล็กซ์ ที่สามารถเลือกรูปแบบในการรับส่งได้
ถึง 4 แบบ

สำหรับในโครงการได้เลือกใช้การรับและส่งข้อมูลในโหมด 2 โดยข้อมูลที่รับส่ง 1 ชุด มี ขนาด
10 บิต

บิตเริ่มต้น 1 บิต (ลอจิก 0) , บิตข้อมูล 8 บิต , บิตหยุด 1 บิต (ลอจิก 1)

และอัตราความเร็วในการส่งข้อมูลมีค่าเป็น 9600 บิตต่อวินาที โดยอัตราความเร็วในการรับส่งข้อมูล
คำนวณได้จากสมการ

$$BAUDRATE = \frac{2^{SMOD}}{32} \times \frac{Oscillator\ Frequency}{12 \times [256 - (TH1)]}$$

ตารางข้างล่างเป็นค่าอัตราความเร็วต่างๆที่ใช้กันมากและบอกว่าสามารถใช้ได้จากไทม์เมอร์ 1 อย่างไร



BAUD RATE	ความถี่ของสัญญาณ		ไทม์เมอร์ 1		
	คริสตอล	SMOD	C/T	โหมด	ค่าที่ใช้โหลด
MODE 0 MAX: 1 MHZ	12MHZ	X	X	X	X
MODE 2 MAX:375K	12MHZ	1	X	X	X
MODES 1,3:62.5K	12MHZ	1	0	2	FFH
19.2K	11.059MHZ	1	0	2	FDH
9.6K	11.059MHZ	0	0	2	FDH
4.8K	11.059MHZ	0	0	2	FAH
2.4K	11.059MHZ	0	0	2	FAH
1.2K	11.059MHZ	0	0	2	E8H
137.5K	11.059MHZ	0	0	2	1DH
110K	11.059MHZ	0	0	2	72H
110K	11.059MHZ	0	0	1	FEEDH

ตารางที่ 2.5 ค่าที่ต้องนำไปไว้ในรีจิสเตอร์ไทม์เมอร์ 1 เมื่อใช้อัตราความเร็วมาตรฐานต่าง ๆ

จากตารางจะเห็นว่าในการทำงานของพอร์ตสื่อสารข้อมูลแบบอนุกรมโหมด 0 จะมีความเร็วในการส่งมากที่สุด เมื่อเปรียบเทียบกับโหมดอื่นที่ความถี่คริสตอลค่าเดียวกัน และจะเห็นว่าหากเลือกใช้คริสตอลความถี่ 11.059 เมกะเฮิร์ตซ์ จะสามารถตั้งค่าอัตราความเร็ว ในโหมด 1 และ โหมด 3 เป็นค่ามาตรฐานที่ใช้กันทั่วไปได้ เช่น 1200, 2400, 4800, 9600, 19,200 จึงเป็นเหตุผลสำคัญที่ในระบบควบคุมส่วนใหญ่เลือกใช้ความถี่ 11.059 เมกะเฮิร์ตซ์ มากกว่า 12 เมกะเฮิร์ตซ์

จากตาราง นอกจากจะแสดงค่าอัตราความเร็วค่าต่างๆ เปรียบเทียบให้เห็นแล้ว ตารางนี้ยังแสดงค่าที่ต้องโหลดไปไว้ในรีจิสเตอร์ใช้งานเฉพาะ TH1 ที่ค่าอัตราความเร็วมาตรฐานต่างๆ ให้ทราบอีกด้วยผู้เขียนโปรแกรมสามารถนำค่านี้ไปใช้ได้เลย

จากโครงงานได้กำหนดใช้ไทม์เมอร์ 1 ทำงานในโหมด 2 และความถี่ระบบเท่ากับ 11.059 เมกะเฮิร์ตซ์ บิต SMOD=0 และรีจิสเตอร์ TH1 = FDH (253)D

$$\text{BAUDRATE} = \frac{2^0}{32} \times \frac{11.059 \times 10^6}{12 \times [256 - 253]}$$

$$\text{BAUD RATE} = 9600$$

การทำงานในการรับและส่งข้อมูลจะต้องใช้งาน SFR ต่างๆ ดังนี้

SCON (SERIAL PORT CONTROL REGISTER) เป็นรีจิสเตอร์ที่ควบคุมการรับและส่งข้อมูลแบบพอร์ตอนุกรม ในที่นี้ได้กำหนดให้การรับส่งข้อมูลทำงานในโหมด 2 โดยข้อมูล 1 ชุด มีขนาด 10 บิต ซึ่งประกอบด้วย บิตเริ่มต้น 1 บิต (ลอจิก 0), บิตข้อมูล 8 บิต และ บิตหยุด 1 บิต (ลอจิก 1 บิต) และ เป็นรีจิสเตอร์เพื่อให้มีการรับข้อมูลเข้ามาทาง RXD

TMOD (TIMER/COUNTER MODE REGISTER) เป็นรีจิสเตอร์ที่ทำหน้าที่ควบคุมการทำงานของตัวจับเวลาหรือตัวนับ ในการใช้งานนี้ได้กำหนดใช้งานไทม์เมอร์ 1 และเลือกการทำงานของไทม์เมอร์ 1 ในโหมด 2 โดยทำงานแบบ “รีโหลด (RELOAD)” และเป็นรีจิสเตอร์ที่กำหนดให้วงจรในการรับส่งข้อมูลทำงาน

TCON (TIMER CONTROL REGISTER) เป็นรีจิสเตอร์เพื่อใช้ตรวจสอบสถานะการทำงานของตัวจับเวลา ในที่นี้ได้ใช้รีจิสเตอร์นี้ในการตรวจสอบว่ามีข้อมูลเข้ามาทาง RXD หรือไม่ และใช้ตรวจสอบในการส่งข้อมูล

IE (INERRUPT ENABLE REGISTER) เป็นรีจิสเตอร์ที่ใช้ควบคุมในการขัดจังหวะ (INTERRUPT)

PCON (POWER CONTROL REGISTER) เป็นรีจิสเตอร์ที่ควบคุมการใช้งานในรูปของการลดกำลังงาน และอัตราเร็วในการส่งข้อมูล

รีจิสเตอร์ทั้งหมดที่กล่าวมานี้จะต้องทำงานที่สอดคล้องกันหมด จึงจะทำให้การรับส่งข้อมูลแบบ UART ทำงานในอัตราความเร็วที่กำหนด

25 มาตรฐาน RS 232C

พอร์ต RS 232 ทำหน้าที่รับและส่งข้อมูลในแบบอนุกรมเรียกว่า Universal Asynchronous Adapter เหตุที่มีชื่อเรียกว่า RS232C ก็เนื่องจากสมาคมผู้ผลิตอุปกรณ์อิเล็กทรอนิกส์ของอเมริกาหรือ EIA ได้กำหนดมาตรฐานของอุปกรณ์สื่อสารแบบอนุกรมเอาไว้ภายใต้ชื่อว่า RS232C ความจริงมาตรฐานของการส่งข้อมูลแบบอนุกรมมีหลายมาตรฐาน แต่ที่นิยมกันมากที่สุดสำหรับไมโครคอมพิวเตอร์ คือ RS232C

หน้าที่ของการสื่อสารแบบอะซิงโครนัสคือ

ก) รับสัญญาณ

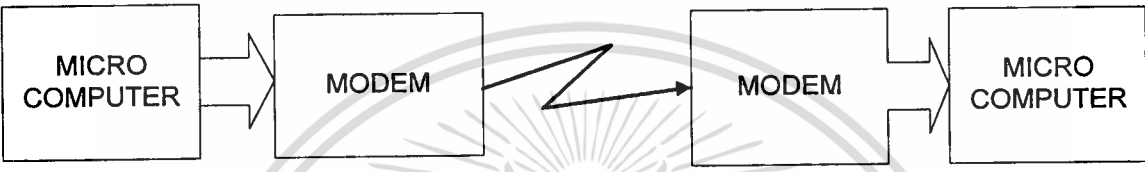
- 1.เปลี่ยนสัญญาณที่เข้ามาแบบอนุกรมมาให้เป็นแบบขนาน
- 2.ตรวจสอบความผิดพลาดของสัญญาณที่ได้รับ
- 3.ตัด Stop bit และ Parity bit ออก
- 4.ส่งสัญญาณให้ซีพียูรู้ว่ารับสัญญาณไว้ได้แล้ว

ข) ส่งสัญญาณ

- 1.เปลี่ยนสัญญาณแบบขนานจากซีพียูค่อยทยอยส่งออกแบบอนุกรม
- 2.เพิ่ม Stop bit และ Parity bit
- 3.เพิ่มสัญญาณควบคุมโมเด็มที่ต่อเนื่อง(ถ้ามี)

เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

มาตรฐาน RS 232C ได้จัดพิมพ์ขึ้นเมื่อปี ค.ศ. 1969 โดยสมาคมผู้ผลิตอุปกรณ์อิเล็กทรอนิกส์แห่งสหรัฐอเมริกา RS ย่อมาจาก Recommened Standard ส่วน 232 เป็นหมายเลขบอกของมาตรฐานตัวนี้ C เป็นหมายเลขของฉบับสุดท้ายของมาตรฐานตัวนี้ จุดประสงค์ของมาตรฐานตัวนี้ก็เพื่อบรรยายคุณลักษณะของการเชื่อมต่ออุปกรณ์รับส่งข้อมูลปลายทาง (Data Terminal Equipment DTE) กับอุปกรณ์สื่อสารข้อมูล (Data Communication Equipment DCE) สำหรับผู้ใช้ไมโครคอมพิวเตอร์ DTE ก็หมายถึงตัวไมโครคอมพิวเตอร์และ DCE ก็หมายถึงโมเด็ม อุปกรณ์อื่น ๆ เช่น เครื่องพิมพ์ที่ได้รับสัญญาณแบบอนุกรมอาจจะใช้ได้ทั้ง DTE และ DCE ขึ้นอยู่กับผู้ผลิต ข้อแตกต่างของ DTE และ DCE จะเห็นได้จาก รูปที่ 8 จากรูปนี้จะเห็นว่า RS 232C มีส่วนสำคัญอย่างยิ่งใหญ่หลวง สำหรับการสื่อสารข้อมูลระหว่างไมโครคอมพิวเตอร์



รูปที่ 2.14 แสดงการใช้ RS232C เชื่อมต่ออุปกรณ์

ความเร็วและระยะทางในการเชื่อมต่อ RS 232C สามารถเชื่อมต่อการถ่ายโอนข้อมูลได้จาก 0-20,000 บิตต่อวินาที ซึ่งเพียงพอสำหรับไมโครคอมพิวเตอร์ที่มีขนาดอัตราบอด 110 ถึง 9600 บอด ความยาวของสายเชื่อมต่อโดยสัญญาณตามมาตรฐานของ RS 232C จำกัดอยู่ที่ 50 ฟุต ซึ่งเพียงพอสำหรับการสื่อสารไมโครคอมพิวเตอร์กับอุปกรณ์รอบนอก

2.5.1 ลักษณะของสัญญาณ RS 232C

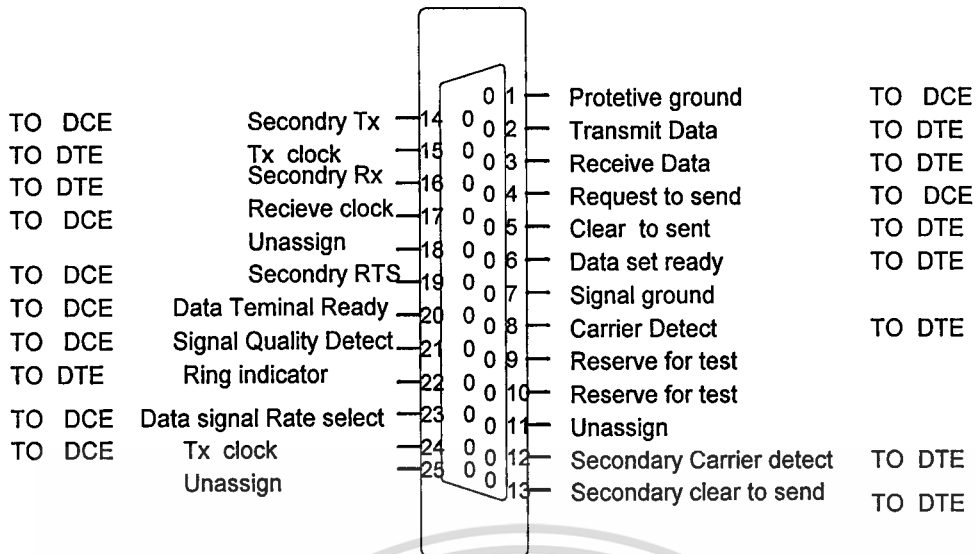
เพื่อเป็นหลักประกันว่าข้อมูลถูกส่งออกไปอย่างถูกต้อง และอุปกรณ์ถูกควบคุมอย่างถูกต้อง จำเป็นจะต้องมีข้อตกลงกันในเรื่องของสัญญาณที่ใช้ มาตรฐาน RS 232C กำหนดย่านของแรงดันไฟฟ้าในสัญญาณเพื่อสนองจุดประสงค์ข้างบน ดังแสดงในตาราง

ข้อตกลงของแรงดันไฟฟ้าในสัญญาณ			
แรงดันไฟฟ้า	สถานภาพลอจิก	สถานภาพของสัญญาณ	ฟังก์ชันในการควบคุม
บวก	0	SPACE	ON
ลบ	1	MARK	OFF

ตารางที่ 2.6 ย่านของแรงดันไฟฟ้าที่ใช้ในสัญญาณ RS 232C

2.5.2. การกำหนดจุดต่อของ RS 232C

มาตรฐานของ RS 232C กำหนดข้อต่อแบบ DB-25 แต่ละขาของข้อต่อกำหนดไว้ดังในรูป



TO DTE = Data terminal Equipment

TO DCE = Data Communication Equipment(modem)

รูปที่ 2.15 แสดงพอร์ตและขาที่ใช้งาน

สัญญาณต่าง ๆ ถูกมอบหมายให้ทำหน้าที่ดังนี้

Transmit Data (TD ขาที่ 2)

เป็นสัญญาณที่ส่งออกจาก DTE (หรือตัวไมโครคอมพิวเตอร์) ไปยังโมเด็มหรือต่อเข้าโดยตรงกับไมโครคอมพิวเตอร์ตัวอื่น หรือเครื่องพิมพ์ เมื่อไม่มีสัญญาณส่งออกสถานะภาพของลอจิกที่ขานี้จะมามีค่าเท่ากับ “1” หรือเทียบเท่ากับ Stop bit

Receive Data (RD ขาที่ 3)

เป็นทางของสัญญาณเข้าไปยัง DTE หรือไมโครคอมพิวเตอร์เมื่อไม่มีสัญญาณรับเข้ามาขานี้จะมามีสถานะภาพทางลอจิกเป็น “1”

Request To Send (RTS ขาที่ 4)

ใช้สำหรับส่งสัญญาณไปยังโมเด็มหรือเครื่องพิมพ์ที่เป็นการเรียกร้องที่จะส่งสัญญาณมายังขาที่ 2 สัญญาณนี้ใช้คู่กับ CTS หรือ Clear To Send อุปกรณ์รับหากได้สัญญาณ RTS จะตรวจสอบตัวเองว่าพร้อมจะรับสัญญาณได้หรือยัง หากพร้อมที่จะรับก็ส่งสัญญาณออกไปที่สาย CTS

Clear To Send(CTS ขาที่ 5)

ดังอธิบายไว้ใน RTS เมื่อสัญญาณอยู่ในสถานะ OFF (Negative Voltage หรือ ลอจิก “1”) หมายความว่า อุปกรณ์รับกำลังบอกว่าพร้อมที่จะรับข้อมูลแล้ว

Data Set Ready (DSR ขาที่ 6)

สัญญาณที่อยู่ในสถานะ ON (ลอจิก “0”) ไปบอกไมโครคอมพิวเตอร์ เมื่อได้รับสัญญาณ จาก modem อีกฝ่ายหนึ่ง เมื่อสัญญาณสายนี้อยู่ในสถานะ ON (หรือลอจิก “0”) เป็นการบอกไมโครคอมพิวเตอร์หรือฝ่ายส่ง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

modem ต่อเข้ากับสายโทรศัพท์เรียบร้อยแล้วและพร้อมที่จะส่งได้แล้ว modem ที่มีการหมุนหมายเลขอัตโนมัติจะส่งสัญญาณสายนี้ไปบอกให้คอมพิวเตอร์รู้ว่าต่อโทรศัพท์ได้แล้ว

Signal Ground (SG ขาที่ 7)

SG ทำหน้าที่เป็นระดับแรงดันอ้างอิงสำหรับทุก ๆ สายของสัญญาณ จะมีแรงดันเป็น “0” เมื่อเทียบกับสัญญาณตัวอื่น

Carrier Detect (CD ขาที่ 8)

โมเด็มจะส่งสัญญาณนี้จะนำไปจุด LED บอกว่าได้รับสัญญาณจากโมเด็มอีกฝ่ายหนึ่งแล้ว ไฟ LED จะอยู่บนหน้าปัดของโมเด็มเอง

Data Terminal Ready(DTR ขาที่ 20)

คอมพิวเตอร์เปิดสัญญาณนี้ให้ ON (ลอจิก “0”)เมื่อพร้อมที่จะติดต่อกับโมเด็ม โมเด็มส่วนมากจะไม่รายงานสถานะภาพของตัวเอง (CD,USR, และ CTS) ให้คอมพิวเตอร์รู้ถ้าหากคอมพิวเตอร์ไม่เปิดสัญญาณ DTR

Ring Indicator (RI ขาที่ 22)

สัญญาณนี้ใช้โมเด็มเป็นระบบอัตโนมัติ (Auto-Answer) สัญญาณนี้จะ ON เมื่อมีสัญญาณกระดิ่งมา และ ระหว่างเสียงดังของกระดิ่ง

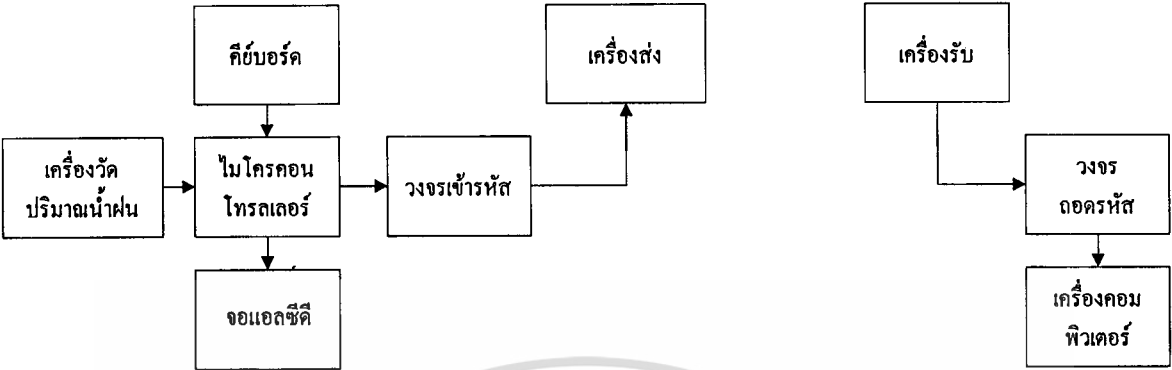
output logic levels with	15V>0h>5V
3k to 7k load	-5>0I>15 V
Driver output voltage when open circuit	V0<25V
Driver output impedance with power off	R0>300 Ohms
Output short circuit current	I0<0.5A
Driver slew rate	dv/dt<30 v/s
Receiver input impedance	7k>Rin>3k
Receiver input voltage	+15 compatible with driver
Receiver output with open circuit input	MARK
Receiver output with +3V input	SPACE
Receiver output with open -3 input	MARK
+15	LOGIC 0 = SPACE
+5	CONTROL ON
+5	Noise Margin
+3	
+3	Transition Region
-3	
-3	Noise Margin
-5	
-5	LOGIC 1 = MARK
-15	CONTROL OFF

ตาราง 2.7 คุณสมบัติโดยย่อของสัญญาณ RS 232C

จากมาตรฐานของ RS 232C ซึ่งจะติดต่อสื่อสารโดยใช้ระดับแรงดันคือ ± 15 V แต่ในวงจรการทำงานของเราซึ่งทำงานเป็นดิจิทัลจะใช้ระดับสัญญาณอยู่ในช่วง 0-5 V ดังนั้นจึงต้องแปลงสัญญาณดังกล่าวให้อยู่ในรูปของสัญญาณดิจิทัลมาตรฐานเสียก่อน

ไอซีที่ทำหน้าแปลงระดับแรงดันที่ใช้ในการติดต่อกันตามมาตรฐาน RS232(± 15 V)ให้เป็นระดับแรงดันดิจิทัลหรือที่เรียกว่า Line Receiver ในที่นี้ใช้ไอซีเบอร์ MAX 232

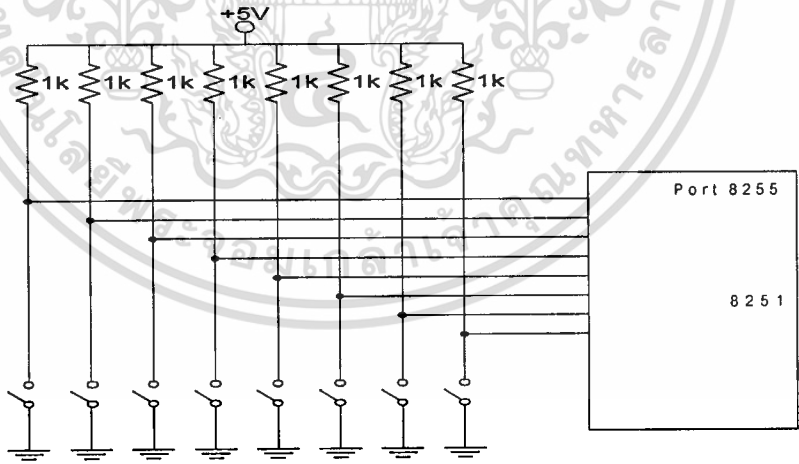
บทที่ 3
การออกแบบและการสร้าง



รูปที่ 3.1 แสดงบล็อกไดอะแกรมของระบบเครื่องบอกปริมาณน้ำฝนผ่านคลื่นวิทยุ

วงจรของเครื่องบันทึก แสดงดังรูป โดยเมื่อเริ่มต้นการทำงาน จะมีการรับข้อมูลจากคีย์บอร์ดเพื่อนำไปใช้ตั้งเวลา และจะบันทึกปริมาณน้ำฝน เมื่อมีการส่งสัญญาณอินเทอร์รัปต์จากเครื่องวัดปริมาณน้ำฝน หลังจากนั้นจะมีการนำข้อมูลที่เก็บได้ ทำการส่งขึ้นไปยังบนคอมพิวเตอร์ แล้วทำการวิเคราะห์ข้อมูลเพื่อแสดงออกมาเป็นกราฟ

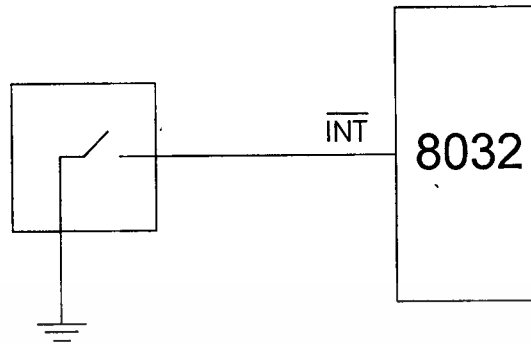
3.1 การเชื่อมต่อของคีย์บอร์ดกับบอร์ดไมโครคอนโทรลเลอร์



รูปที่ 3.2 วงจรการเชื่อมต่อคีย์บอร์ดกับ 8255

ในการเชื่อมต่อนี้ทำการเชื่อมต่อสวิทช์ 8 ตัว โดยสวิทช์ 1 ตัวจะเชื่อมต่อกับพอร์ตของ 8255 1 ขา ดังนั้น สวิทช์ 8 ตัว ก็เชื่อมต่อกับขาของ 8255 ทั้งหมด 8 ขา นั่นคือใช้พอร์ต 8255 จำนวน 1 พอร์ต

3.2 การเชื่อมต่อของเครื่องวัดปริมาณน้ำฝนกับ บอร์ดไมโครคอนโทรลเลอร์



รูปที่ 3.3 แสดงการเชื่อมต่อของเครื่องวัดปริมาณน้ำฝนกับไมโครคอนโทรลเลอร์

ในการเชื่อมต่อเครื่องวัดปริมาณน้ำฝน กับบอร์ดไมโครคอนโทรลเลอร์ นั้น เครื่องวัดปริมาณน้ำฝนแบบคานกระดกนั้น เป็นเหมือนกับสวิตช์ 1 ตัว โดยขาอินเตอร์รัปต์ของ 8032 จะแอกทีฟเมื่อลอจิกเปลี่ยนเป็นจาก “1” เป็น “0” ดังนั้นเมื่อข้างหนึ่งของสวิตช์ต่อลงกราวด์ อีกข้างหนึ่งต่อกับขาอินเตอร์รัปต์ของ 8032 เมื่อคานกระดก ก็จะเป็นเหมือนกับกดสวิตช์ ขาอินเตอร์รัปต์ก็จะต่อลงกราวด์ ทำให้เกิดอินเตอร์รัปต์ขึ้น

3.3 การเก็บข้อมูล และพื้นที่ในการเก็บข้อมูล

3.3.1 รหัสที่ใช้ในการเก็บข้อมูลและลักษณะของข้อมูลที่เก็บ

FO เป็นรหัสแสดงการเริ่มต้นการเก็บข้อมูล หลังจากรหัสนี้แล้วจะมีข้อมูลตามมาก็คือข้อมูลของ เดือน , วัน , ปี , ชั่วโมง , นาที

FC เป็นรหัสแสดงการเปลี่ยนชั่วโมง หลังจากรหัสนี้แล้วจะมีข้อมูลที่ตามมาก็คือเดือน , วัน , ปี , ชั่วโมง , นาที

FA เป็นรหัสแสดงสถานะว่าหน่วยความจำเต็มแล้ว

3.3.2 การเก็บข้อมูลของปริมาณน้ำฝน

ข้อมูลของปริมาณน้ำฝน จะมีบัพเฟอร์คอยนับจำนวนพัลส์ ในการเกาะของกระดิ่งของเครื่องวัดปริมาณน้ำฝนแบบคานกระดก เมื่อครบหนึ่งนาบี่ก็จะมีการเก็บค่าจำนวนพัลส์ในบัพเฟอร์นี้ลงหน่วยความจำต่อไป

3.3.3 ความสามารถในการเก็บข้อมูล

พื้นที่ในการเก็บข้อมูลทั้งหมดมีจำนวน 32,768 ไบต์

การคำนวณปริมาณน้ำฝนและอุณหภูมิ

ปริมาณน้ำฝน ใน 1 ชั่วโมงเก็บ รหส , วัน , เดือน , ปี , ชั่วโมง , นาที โดยมีจำนวน

ทั้งหมด 6 ไบต์และเก็บค่าของข้อมูลจำนวน 60 ไบต์

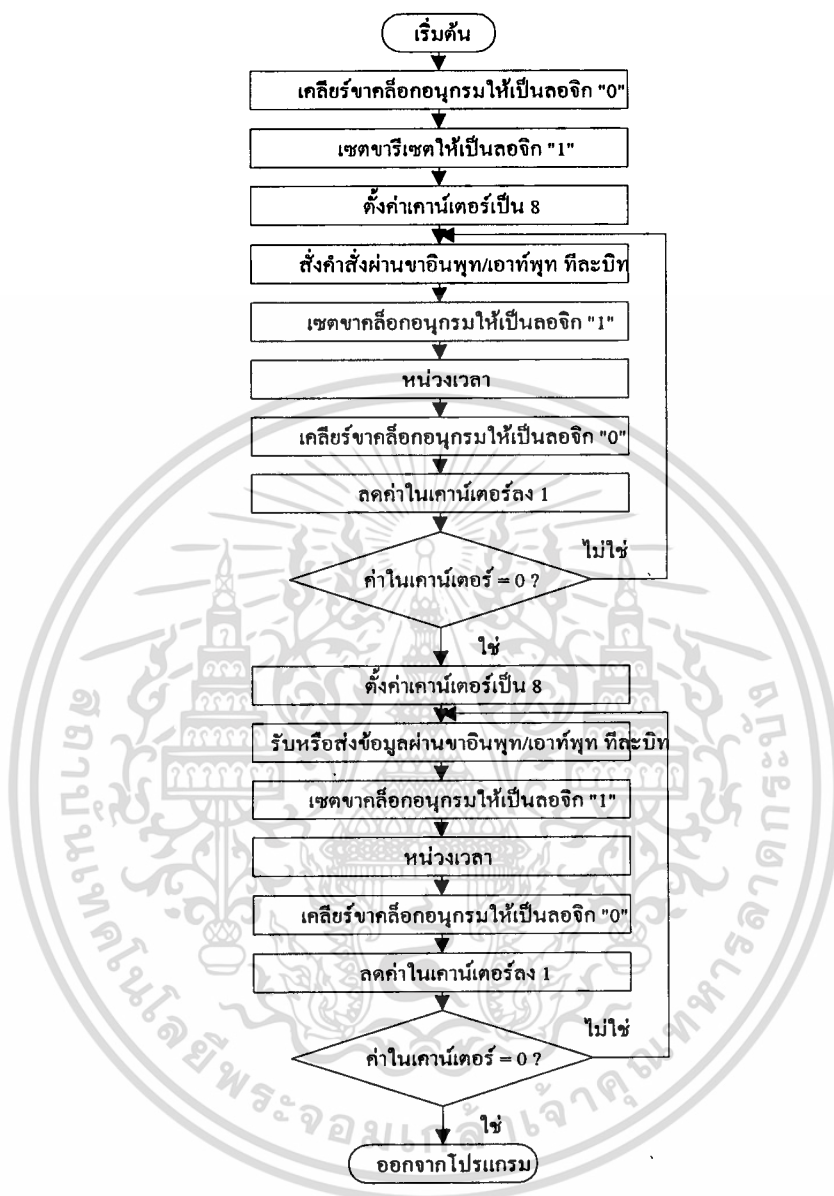
ดังนั้น	ใน 1 ชั่วโมงจะใช้พื้นที่ในการเก็บทั้งหมด	66	ไบต์
เพราะฉะนั้นใน	ใน 1 วัน จะใช้พื้นที่ในการเก็บทั้งหมด $24 * 66 =$	1,584	ไบต์
ในระยะเวลา 20 วัน	ใช้พื้นที่ในการเก็บทั้งหมด $1584 * 20 =$	31,680	ไบต์
รหัสบอกสถานะหน่วยความจำเต็มใช้พื้นที่		1	ไบต์
เก็บแอดเดรส ตำแหน่งที่เก็บข้อมูลของปริมาณน้ำฝน		2	ไบต์
ดังนั้นพื้นที่ที่เหลือในการเก็บข้อมูล	$32,768 - (2 + 1 + 31,680) =$	1085	ไบต์
ทำการเก็บข้อมูลในเวลา 16 ชั่วโมง	ใช้พื้นที่ไปทั้งหมด $=$	1056	ไบต์
จะเหลือพื้นที่อีก	$1,085 - 1,056 =$	29	ไบต์
เก็บรหส เดือน , วัน , ปี , ชั่วโมง , นาที		6	ไบต์
เพราะฉะนั้น จะเก็บข้อมูลได้อีกประมาณ	$29 - 6 =$	23	ไบต์ ๖๖๖

(23 นาที)

เพราะฉะนั้นจะจัดเก็บข้อมูลปริมาณน้ำฝนได้ทั้งหมด 20 วัน 16 ชั่วโมง 23 นาที

3.4.3 โฟลว์ชาร์ตของโปรแกรมการติดต่อกับ DS1202

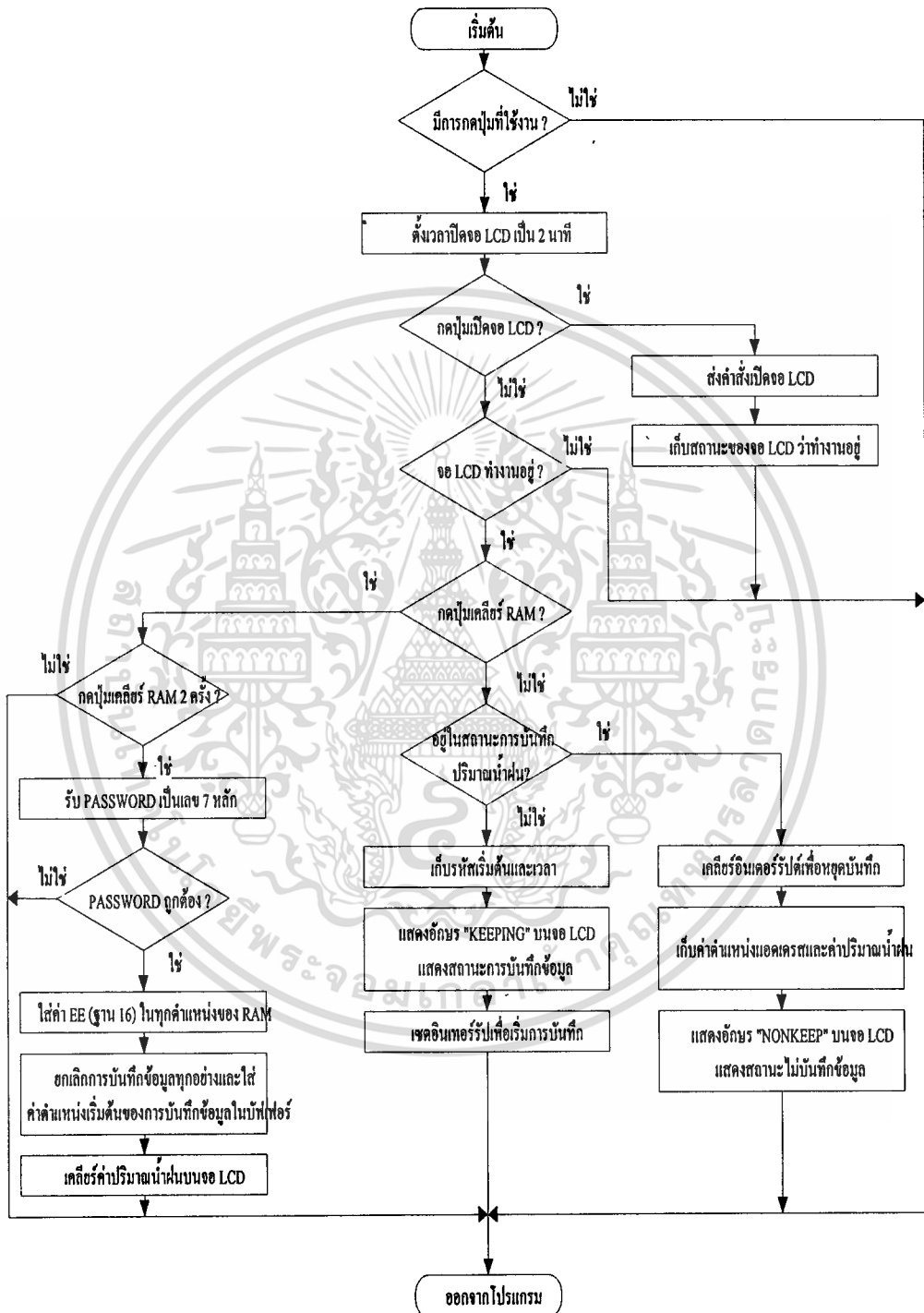
เป็นโปรแกรมย่อย ทำหน้าที่ในการติดต่อกับชิพนาฬิกา ทั้งอ่านข้อมูลและเขียนข้อมูล



รูปที่ 3.6 แสดงโฟลว์ชาร์ตของการติดต่อกับ DS1202

3.4.4 โฟลว์ชาร์ตของโปรแกรมการรับคำสั่งจากคีย์บอร์ดควบคุมการทำงาน

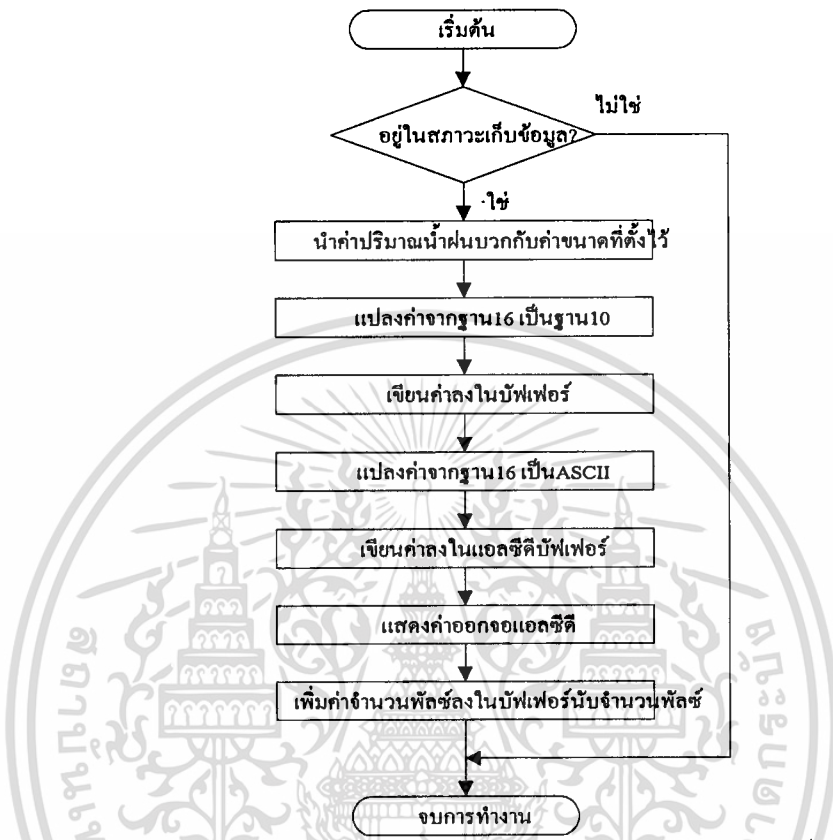
เป็นโปรแกรมย่อย ทำหน้าที่ในการรับคำสั่งจากคีย์บอร์ดที่ใช้ในการควบคุมการทำงานของจอ LCD การเก็บข้อมูลปริมาณน้ำฝน และการเคลียร์ข้อมูลในเมมโมรี่



รูปที่ 3.7 แสดงโฟลว์ชาร์ตของการรับคำสั่งจากคีย์บอร์ดควบคุมการทำงาน

3.4.5 โฟลว์ชาร์ตของโปรแกรมการรับสัญญาณจากเครื่องวัดปริมาณน้ำฝน

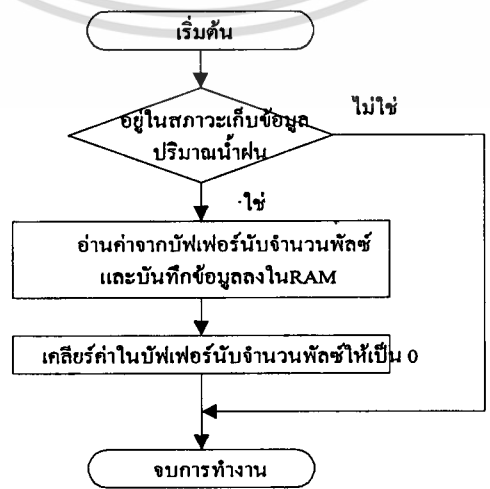
เป็นโปรแกรมย่อย ทำหน้าที่ในการรับสัญญาณจากเครื่องวัดปริมาณน้ำฝน ซึ่งจะเป็นสัญญาณที่เข้ามาทำให้เกิดการอินเทอร์รัปต์ และทำการบันทึกข้อมูลของสัญญาณ



รูปที่ 3.8 แสดงโฟลว์ชาร์ตของการรับสัญญาณจากเครื่องวัดปริมาณน้ำฝน

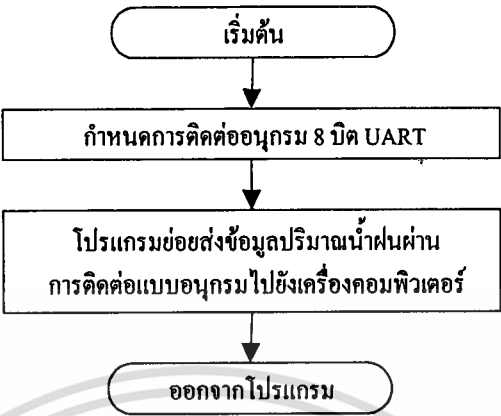
3.4.6 โฟลว์ชาร์ตของโปรแกรมการบันทึกข้อมูลปริมาณน้ำฝน

เป็นโปรแกรมย่อย ทำหน้าที่ในการบันทึกข้อมูลปริมาณน้ำฝนลงใน RAM



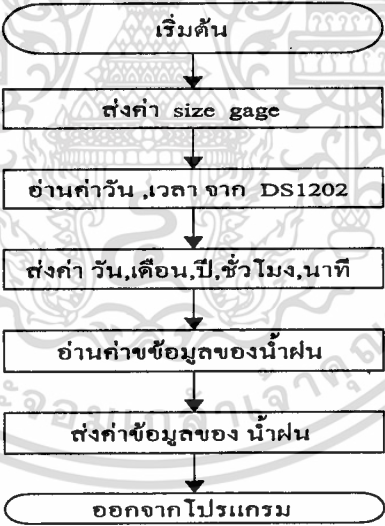
รูปที่ 3.9 แสดงโฟลว์ชาร์ตของการบันทึกข้อมูลปริมาณน้ำฝน

3.4.7 โฟลว์ชาร์ตของโปรแกรมการส่งข้อมูลจาก RAM ไปยังเครื่องคอมพิวเตอร์
เป็นโปรแกรมย่อย ทำหน้าที่ในการส่งข้อมูลจาก RAM ไปยังเครื่องคอมพิวเตอร์



รูปที่ 3.10 แสดงโฟลว์ชาร์ตของการส่งข้อมูลจาก RAM ไปยังเครื่องคอมพิวเตอร์

3.4.8 โฟลว์ชาร์ตของโปรแกรมย่อยการส่งค่าออกพอร์ตอนุกรมทุกๆ 1 นาที

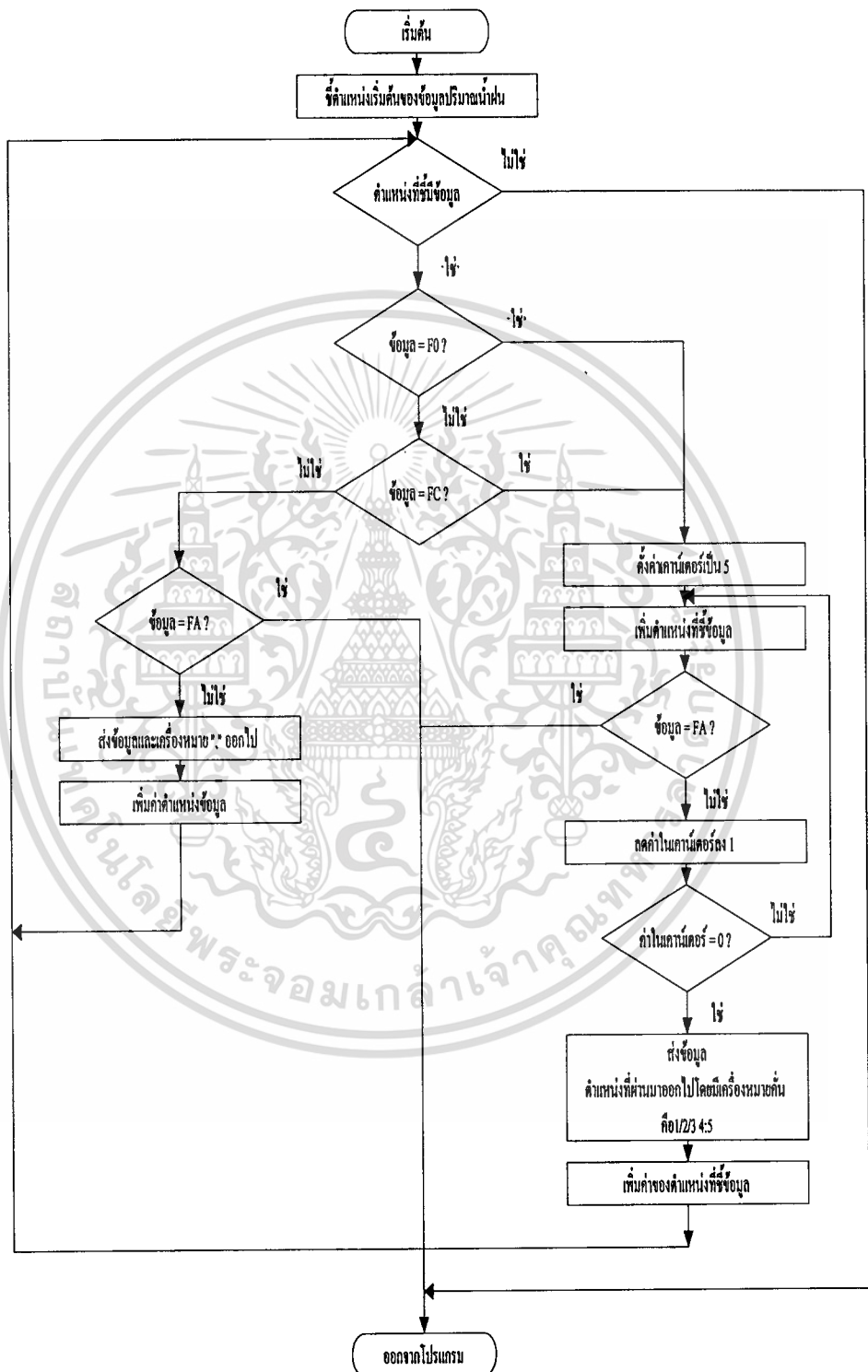


รูปที่ 3.11 แสดงโฟลว์ชาร์ตของโปรแกรมย่อยการส่งค่าออกพอร์ตอนุกรมทุกๆหนึ่งนาที

3.4.9 โฟลว์ชาร์ตของโปรแกรมการส่งข้อมูลปริมาณน้ำฝนไปยังเครื่องคอมพิวเตอร์

เป็นโปรแกรมย่อย ทำหน้าที่ในการวิเคราะห์และส่งข้อมูลปริมาณน้ำฝนไปยังเครื่อง

คอมพิวเตอร์

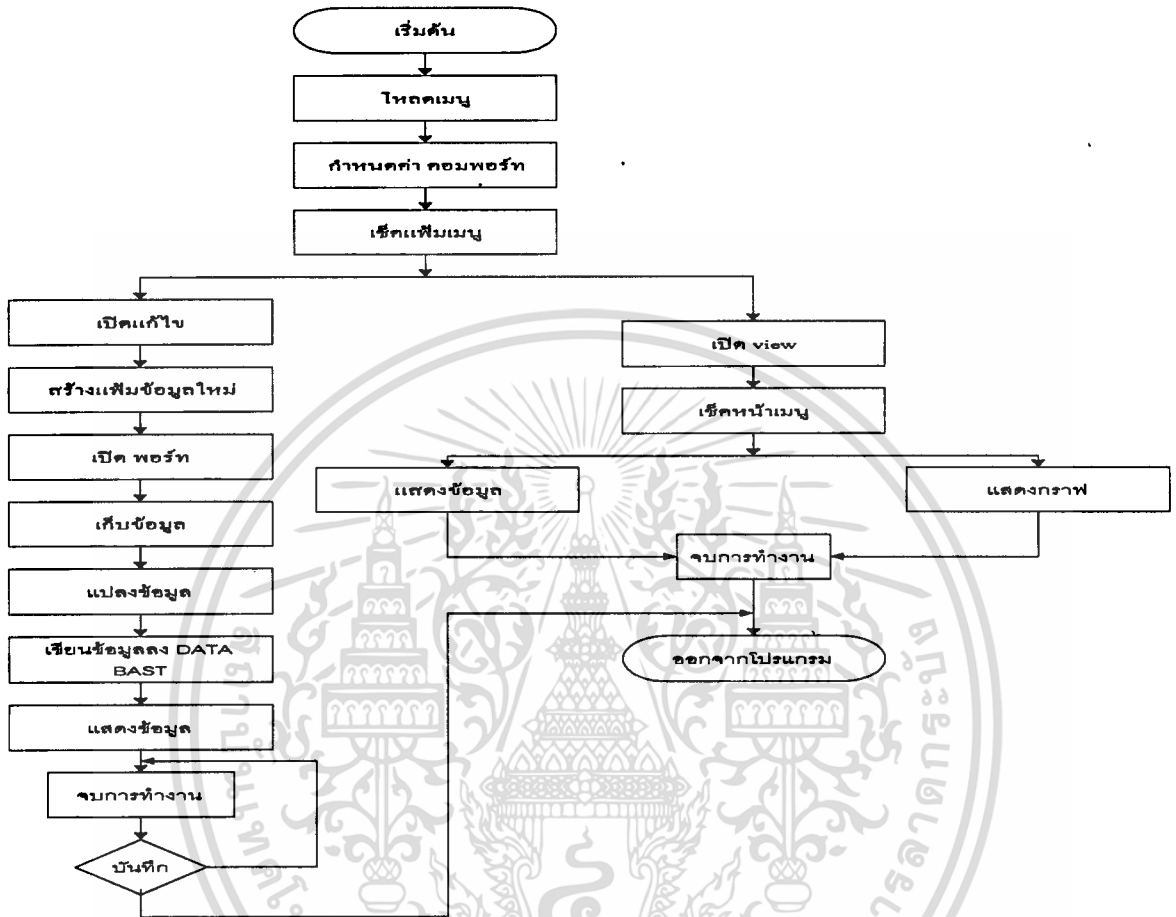


รูปที่ 3.12 แสดงโฟลว์ชาร์ตของการส่งข้อมูลปริมาณน้ำฝนไปยังเครื่องคอมพิวเตอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้拿去ใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.4.10 โฟลว์ชาร์ตของโปรแกรมหลัก รับข้อมูลจากบอร์ดไมโครคอนโทรลเลอร์มาแสดงและจัดเก็บ

เป็นโฟลว์ชาร์ตแสดงการทำงานของโปรแกรมหลัก โดยจะมีการเซตพอร์ตสื่อสารให้พร้อมจะทำงานและทำการรับข้อมูลออกมาแสดงบนหน้าจอ มีการสร้างและ จัดเก็บไฟล์ข้อมูลที่ได้รับมาจากบอร์ดไมโครคอนโทรลเลอร์ลงในฮาร์ดดิสก์



รูปที่ 3.13 แสดงโฟลว์ชาร์ตการทำงานของโปรแกรมหลักในการรับข้อมูลที่คอมพิวเตอร์

3.5 ภาค FSK Modulation

ในการออกแบบ FSK Modulation ได้นำเอาไอซีเบอร์ XR2206 ซึ่งเป็นโมโนลิธิคฟังก์ชันเบสเทอร์เรเตอร์ สามารถกำหนดลูกคลื่นได้ทั้งคลื่นไซน์ คลื่นสามเหลี่ยม คลื่นสี่เหลี่ยม หรือแรมป์(Ramp) โดยมีย่านความถี่ที่สามารถผลิตได้คือ 0.01 Hz ถึง 1.0 Mhz

ในกรณีที่เรารู้ XR2206 เป็นตัวกำเนิดคลื่นรูปไซน์ในลักษณะ FSK Generator โดยใช้โหม่งรีซิสเตอร์ R1 และ R2 ที่ต่อระหว่างขา 7 และขา 8 กับกราวด์ตามลำดับ โดยใช้สัญญาณดิจิตอลจะถูกป้อนเข้ามายังขา 9 ของไอซี ซึ่งจะเพื่อกำหนดความถี่ทางเอาต์พุตที่ขา 2 โดยถ้าหากขา 9 อยู่ในสถานะเปิดวงจรหรือมี $V_{in} \geq 2\text{ v}$ แล้ว R1 จะเป็นตัวกำหนดโหม่งร่วมกับตัวเก็บประจุที่ต่อคร่อมระหว่างขา 5 กับขา 6 และในทำนองกลับกัน ถ้า อินพุต (ขา 9) มี $V_{in} < 1\text{ v}$ แล้ว R₂ จะเป็นตัวกำหนดโหม่งร่วมกับตัวเก็บประจุระหว่างขา 5 กับขา 6 เช่นเดียวกัน จึงทำให้ความถี่ที่ได้ออกมาทางเอาต์พุตจะอยู่ในช่วงระหว่าง F1 กับ F2 โดยทั้ง F1 และ F2 จะอิสระต่อกันและเราสามารถที่จะเปลี่ยนแปลงความถี่ได้โดยการเลือกค่า R1 และ R2 ตามสมการข้างล่าง

$$F_1 = 1 / R_1 C$$

$$F_2 = 1 / R_2 C$$

ตัวเก็บประจุระหว่างขา 5 กับ ขา 6 ควรอยู่ในช่วง 1,000 pF - 100 uF และตัวต้านทาน R₁ , R₂ ควรจะมีค่าอยู่ในช่วง 4 kΩ - 200 kΩ

จากสมการเพื่อให้่ายในทางปฏิบัติจึงกำหนดให้ C = 0.001 uF และสามารถที่จะหาค่า R₁ และ R₂ ตามค่าความถี่ที่กำหนดได้คือ

ที่ $F_1 = 60\text{ kHz}$ สำหรับแทนระดับลอจิก "1" จะได้ว่า

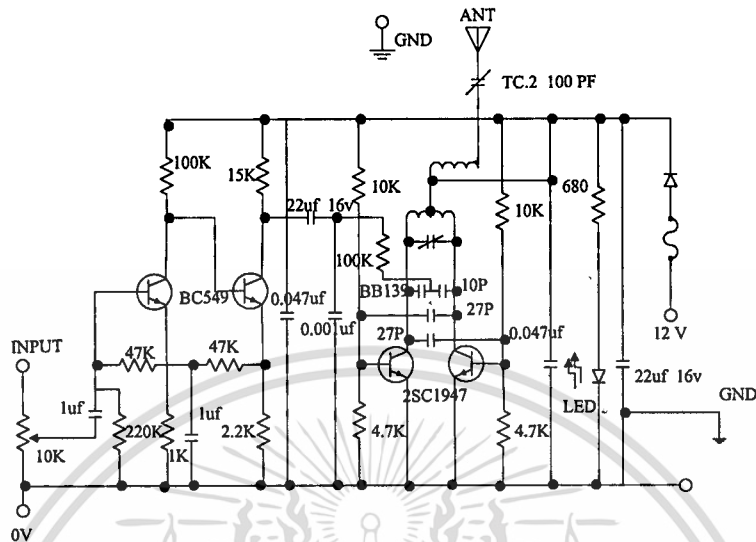
$$\begin{aligned} R_1 &= 1 / F_1 C \\ &= 1 / (60 * 10^3)(0.001 * 10^{-6}) \\ &= 16.667\text{ k}\Omega \end{aligned}$$

เนื่องจากเราต้องสามารถปรับความถี่ให้มีค่าตามที่กำหนดจึงเลือกใช้โปเทนมิเตอร์ 20 kΩ ต่ออนุกรมกับ VR 10 K และต่อระหว่างขา 7 กับกราวด์

และที่ $F_2 = 65\text{ kHz}$ สำหรับแทนระดับลอจิก "0" จะได้ว่า

$$\begin{aligned} R_2 &= 1 / F_2 C \\ &= 1 / (65 * 10^3)(0.001 * 10^{-6}) \\ &= 15.384\text{ k}\Omega \end{aligned}$$

เนื่องจากเราต้องสามารถปรับความถี่ให้มีค่าตามที่กำหนดจึงเลือกใช้โปเทนมิเตอร์ 20 kΩ ต่ออนุกรมกับ R 10 K และต่อระหว่างขา 8 กับกราวด์



รูปที่ 3.15 FM Transmitter Circuit

ในส่วนนี้ก็จะทำหน้าที่รับสัญญาณวิทยุที่ส่งมาจากภาคส่ง แล้วทำการตีความถอดเพื่อเอาสัญญาณข้อมูลที่เป็นความถี่ซึ่งแทนระดับของลอจิก “1” และ “0” คือ $F = 60 \text{ kHz}$ และ $F = 65 \text{ kHz}$ ออกจากคลื่นพาหะก่อนจะส่งไปยังภาค FM Demodulator เพื่อเปลี่ยนจากระดับความถี่ให้กลับมาเป็นระดับลอจิกเหมือนกับทางภาคส่ง

ภาครับ ใช้ ไอซี TA 8122 AN เป็นของบริษัทโตชิบาซึ่งเป็นไอซีที่สามารถใช้งานเป็นเครื่องรับวิทยุ AM/FM และนอกจากนี้ยังใช้แรงดันไฟเลี้ยงในย่านแรงดันต่ำ ตลอดจนได้นำเอาวงจรที่จำเป็นต่าง ๆ รวมไว้ภายในไอซีตัวเดียว และไอซีตัวนี้มีคุณสมบัติที่เด่น ๆ ดังนี้คือ

- ภายในไอซีประกอบด้วยภาคการทำงานต่าง ๆ เช่นภาค AM/FM IF , ภาค FM Multiplex , ภาค FM Oscillator Output
- ไม่ต้องการตัวเก็บประจุในการคัปปลิงในส่วนของ AM Detector Coil , IF Transformer และภาคขยาย IF
- ไม่มีการปรับแต่งเมื่อมีการใช้งานจริง ในส่วนของภาค FM Quaddetector , และส่วนของ VCO Multiplex
- ใช้งานได้ ในย่านแรงไฟแรงดันต่ำตั้งแต่ 1.8 v ถึง 8v

สำหรับรายละเอียดทางเทคนิคต่าง ๆ มีดังนี้

ข้อมูล	สัญลักษณ์	ค่า	หน่วย
แรงดันซัพพลาย	Vcc	1.8-8	V
กระแสซัพพลาย	Icc	FM=18.5 AM=8.3	mA
กระแสหลอด LED	I _{led}	10	mA
อัตราการสูญเสียกำลังงาน	p _d	สูงสุด1,200	mW
อุณหภูมิทำงานปกติ	T _{opr}	-25 ถึง 75	°C

ตารางที่ 3.1 ข้อมูลโดยรวมของไอซี TA8122AN

ข้อมูล	สัญลักษณ์	ค่า	หน่วย
อัตราขยายแรงดัน	G _v	40	mV _{rms}
แรงดันเอาต์พุต	V _{OD}	60	mV _{rms}
อัตราส่วนสัญญาณต่อสัญญาณรบกวน	S/N	44	db
ความเพี้ยนรวม	THD	1.0	%
ความไวของสัญญาณไฟจูนนิ่ง	V _L	24	°C
ความต้านทานที่ขา 19	R ₁₉	FM=0.75 AM=12.5	kΩ

ตารางที่ 3.2 ข้อมูลในการรับในแบบเอเอ็ม

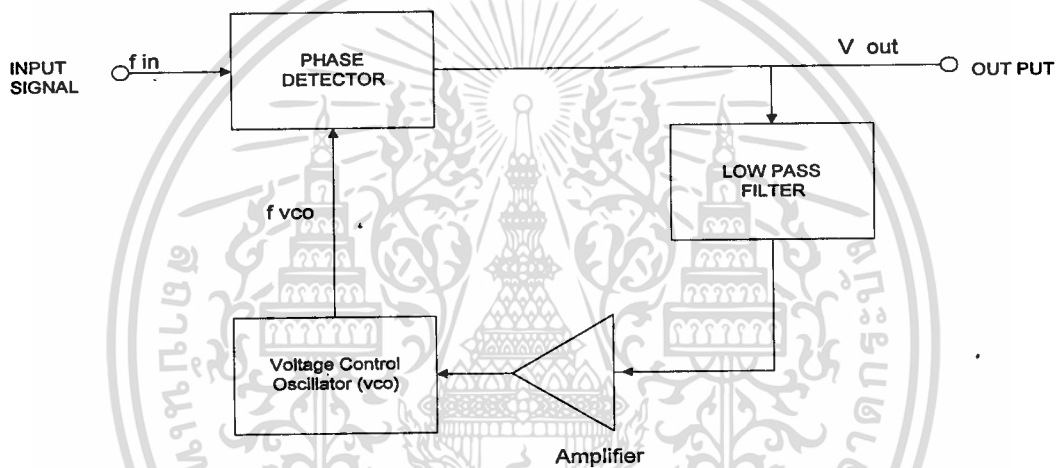


3.8 ภาค FSK Demodulation

ในการออกแบบ FSK Demodulation ได้นำเอาไอซี XR 2211 ซึ่งเป็นโมโนลิธิคเฟสล็อก ลูป ที่มีการใช้งานเช่น

- AM และ FM Demodulator
- Frequency Shift Keying(FSK) Decoders
- Frequency Synthesizers
- Television Synthesizers
- Motor Speed Controls

3.8.1. พื้นฐานของ เฟสล็อก ลูปแสดงได้ดังรูป



รูปที่ 3.17 บล็อกไดอะแกรมการทำงานของเฟสล็อก ลูป

ตามรูปเฟสพื้นฐานประกอบด้วย

- 1.ภาคเปรียบเทียบเฟส (Phase Detector หรือ Phase Comparison)
- 2.ถากรองความถี่ต่ำผ่าน(lowpass Filter)
- 3.ภาคขยาย (Amplifier)
- 4.ภาคกำเนิดความถี่ที่ควบคุมได้ด้วยแรงดัน (Voltage Controlled Oscillator , VCO)

วงจรการทำงานนี้ เป็นแบบหนึ่งอินพุต และหนึ่งเอาต์พุต และมีการนำเอาส่วนของเอาต์พุตป้อนกลับมายังภาค phase Detector จุดประสงค์ของเฟสล็อก ลูปคือ ทำให้แรงดันเอาต์พุต (V_{OUT}) มีการเปลี่ยนแปลงเป็นสัดส่วนโดยตรงกับความแตกต่างทางเฟสระหว่าง VOC (f_{vco}) กับสัญญาณทางอินพุต (Signal) และจะให้ f_{vco} มีค่าเท่ากับ f_{in}

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หน้าที่การทำงานของภาคต่าง ๆ

1. phase Detector ทำหน้าที่เปรียบเทียบเฟสระหว่างสัญญาณอินพุต กับสัญญาณที่ได้จาก VOC และ เอาท์พุท ที่ได้ออกมาเป็นแรงดัน DC ที่เป็นสัดส่วนที่สอดคล้องกับความต่างเฟสระหว่างสองสัญญาณ
2. Lowpass Filter กรองเอาความถี่ที่เกินกว่าช่วงความถี่ (Frequency range) ออกและอีกสิ่งหนึ่งคือ ทำให้ผลตอบสนองต่อการเปลี่ยนแปลงทางด้านความถี่ทางอินพุตดีขึ้น
3. Amplifier ขยายแรงดัน DC เพื่อเพิ่มความไวของเฟสล็อก
4. VCO (Voltage Controlled Oscillator) เป็น Oscillator ซึ่งความถี่ทางด้านเอาท์พุทสามารถควบคุมทางด้านของแรงดัน DC

3.8.2. สถานะของการทำงานของเฟสล็อก

1. Free -running

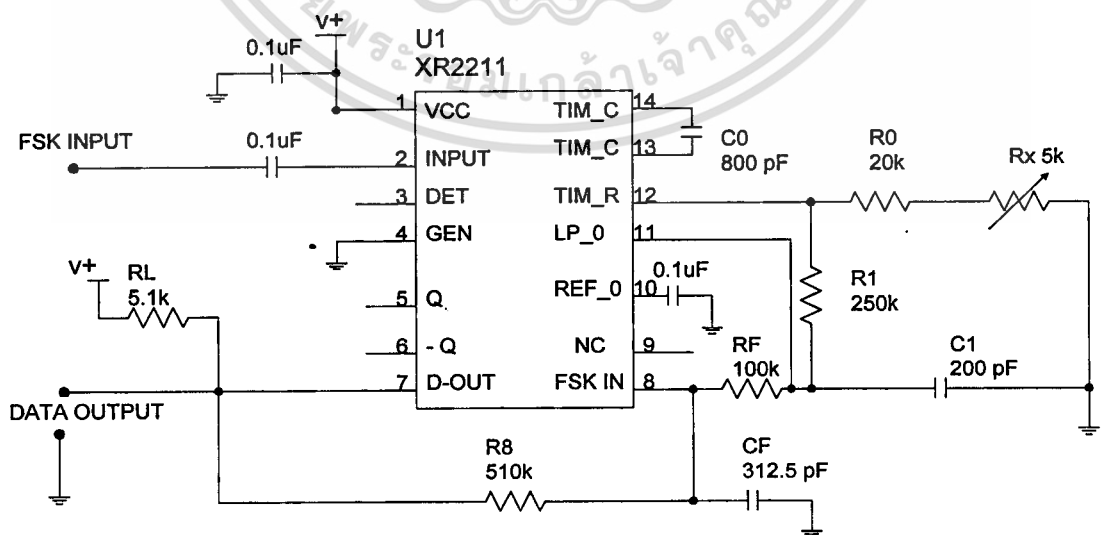
สถานะนี้เกิดขึ้นเมื่อสัญญาณที่รับเข้ามาแตกต่างไปต่างไปจากสัญญาณที่ได้จาก VCO โดย Phase Detector จะให้ค่าแรงดัน DC ออกมา ซึ่งเรียกว่า Error Voltage ป้อนกลับให้ VCO และเป็นสาเหตุให้ความถี่ของ VCO เปลี่ยนแปลง

2. Capture

สถานะนี้เริ่มต้นเมื่อ VCO เริ่มเปลี่ยนแปลงความถี่ เพื่อลดความแตกต่างระหว่างความถี่ที่สร้างขึ้นโดย VCO กับความถี่ที่รับเข้ามา

3. Phase-Lock

เมื่อความถี่ของ VCO เหมือนกับความถี่ทางอินพุตจะเรียกว่า Phase Locked และสำหรับโครงสร้างภายในไอซีเบอร์ XR 2211 สามารถประยุกต์วงจรเป็น FSK Demodulator ได้ดังรูป



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับใช้ภายในงานจริง รูปที่ 3.18 วงจรใช้งานจริง XR 2211 FSK Decoder
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.8.3. การออกแบบ FSK Demodulator โดยใช้ XR2211

ข้อ กำหนดในการออกแบบคือ

$$F_1 = 60 \text{ kHz}$$

$$F_2 = 65 \text{ kHz}$$

$$\text{Baud Rate} = 9,600$$

1. คำนวณหาความถี่กลาง (Center Frequency), f_0

$$f_0 = (f_1 + f_2) / 2$$

$$= (60 + 65) / 2$$

$$= 62.5 \text{ kHz}$$

2. ค่าที่ควรจะใช้สำหรับโทมมิ่งรีซิสเตอร์ (R_0) ควรจะอยู่ในช่วง 10 kHz ถึง 100 kHz ในกรณีนี้
เลือกใช้ $R_0 = 20 \text{ kHz}$

3. คำนวณค่าของ C_0 จาก

$$C_0 = 1 / R_0 f_0$$

$$= 1 / (20 \times 10^3)(62.5 \times 10^3)$$

$$= 800 \text{ Pf}$$

4. คำนวณค่า R_1 โดย

$$R_1 = R_0 [f_0 / (f_1 - f_2)]$$

$$= 250 \text{ k}\Omega$$

5. คำนวณค่า C_1 โดยทั่ว ๆ ไปกำหนดให้ Damping factor (ζ) = 1 / 2

$$C_1 = C_0 / 4 \text{ สำหรับ } \zeta = 1 / 2$$

$$= 800 / 4$$

$$= 200 \text{ pF}$$

6. คำนวณค่า Filter Capacitance, C_f

$$\text{สำหรับ } R_F = 100 \text{ k}\Omega, R_B = 510 \text{ k}\Omega$$

$$C_F = 3 / \text{Baud Rate } \mu\text{F}$$

$$= 3 / 9,600$$

$$= 312.5 \text{ pF}$$

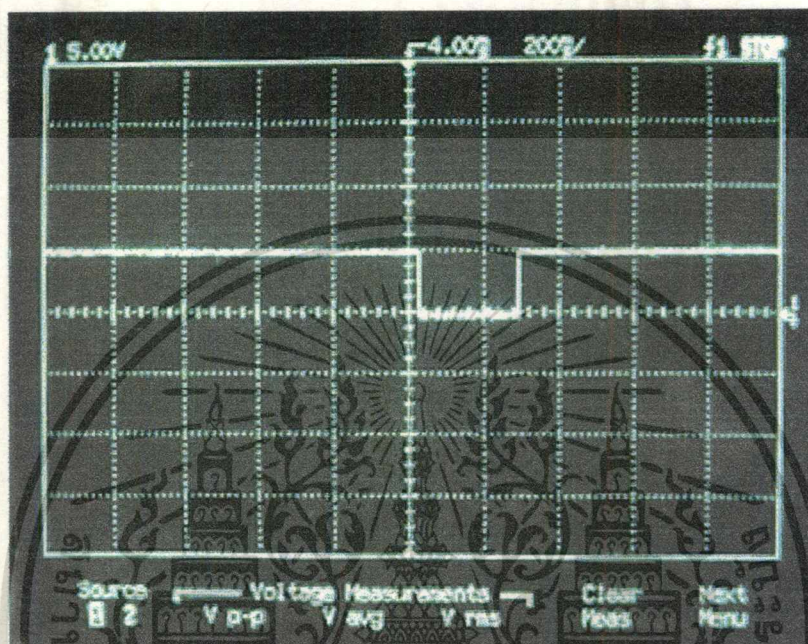
สัญญาณเอาต์พุตของไอซี XR2211 นั้นจะได้ออกมาเป็นระดับลอจิก “0” หรือ “1” ตามที่ทางเครื่องส่งได้ส่งมาให้ และระดับของสัญญาณนี้ จะมีค่าอยู่ในระดับสัญญาณ ดิจิตอลมาตรฐานแล้ว นั่นคือ 0V สำหรับลอจิก “0” และ 5V สำหรับลอจิก “1”

บทที่ 4

การทดลองและผลการทดลอง

4.1 ทดลองการเชื่อมต่อของคีย์บอร์ด ไมโครคอนโทรลเลอร์

4.1.1 ทำการต่อวงจรเชื่อมต่อสวิตช์ คีย์บอร์ดและสังเกตสัญญาณที่เข้าขาของพอร์ต 8255 เมื่อกดคีย์บวกและเทียบกับหน้าจอ LCD จะเห็นว่าค่าตัวเลขเพิ่มขึ้นดังรูปที่ 4.1 และรูปที่ 4.2 ตามลำดับ



รูปที่ 4.1 สัญญาณที่เข้าไปยังขาของพอร์ต 8255 เมื่อทำการกดสวิตช์



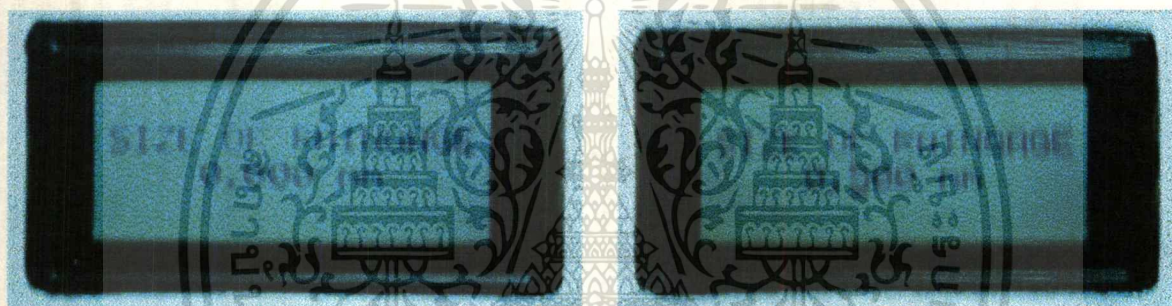
รูปที่ 4.2 แสดงหน้าจอ LCD เทียบกับรูปสัญญาณเมื่อกดคีย์บวก

4.2 ทดลองป้อนโปรแกรมลงบอร์ดไมโครคอนโทรลเลอร์ และเชื่อมต่อบอร์ดไมโครคอนโทรลเลอร์กับจอ LCD และคีย์บอร์ด

4.2.1 ทำการป้อนโปรแกรมลงบอร์ดไมโครคอนโทรลเลอร์ และสังเกตผลที่หน้าจอ LCD



รูปที่ 4.3 แสดง DEMO หน้าจอเมื่อป้อนโปรแกรมลงบอร์ดไมโครคอนโทรลเลอร์



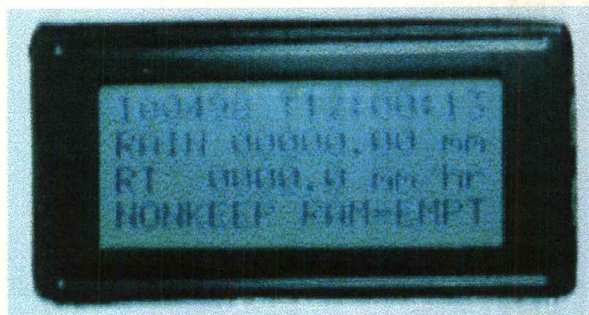
รูปที่ 4.4 แสดงหน้าจอก่อนการป้อนค่าขนาด
ของอุปกรณ์ที่ใช้รับปริมาณน้ำฝน

รูปที่ 4.5 แสดงหน้าจอการป้อนค่าขนาด
ของอุปกรณ์ที่ใช้รับปริมาณน้ำฝน



รูปที่ 4.6 แสดงหน้าจอก่อนป้อนค่าเริ่มต้นของ วัน,เดือน,ปี และเวลา

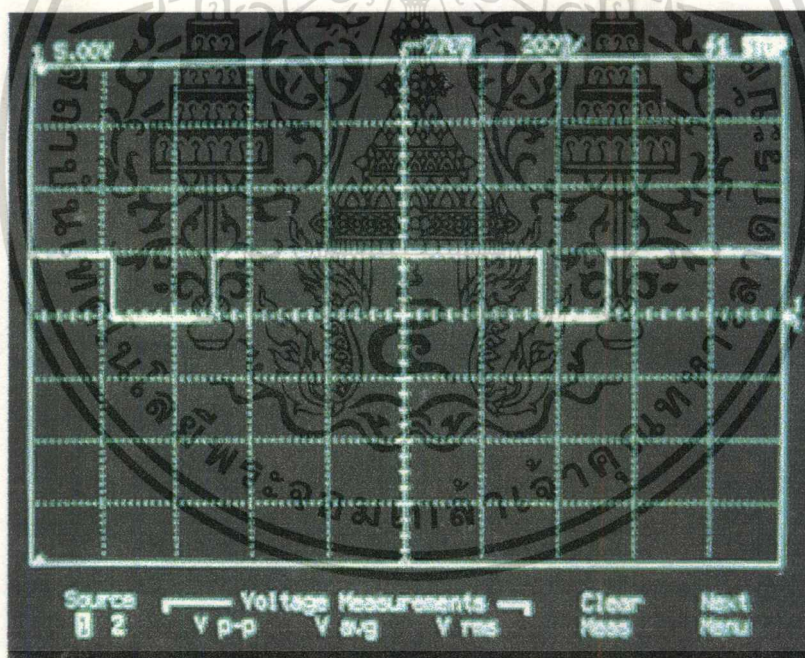
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.7 แสดงหน้าจอเมื่อป้อนค่า วัน,เดือน,ปี และ เวลา เริ่มต้นเรียบร้อยแล้ว

4.3 ทดลองการเชื่อมต่อของเครื่องวัดปริมาณน้ำฝนกับ บอร์ดไมโครคอนโทรลเลอร์

4.3.1 ทำการต่อเครื่องวัดปริมาณน้ำฝน โดยข้างหนึ่งต่อกับขาอินเทอร์รัปต์ 0 ของ 8032 ในบอร์ดไมโครคอนโทรลเลอร์ อีกข้างต่อลงกราวด์ และทำการสังเกตสัญญาณที่เปลี่ยนแปลงที่ขาอินเทอร์รัปต์ของ 8032 และสังเกตหน้าจอ LCD เทียบกับการเปลี่ยนแปลงที่ขาอินเทอร์รัปต์ จะได้ผลดังรูปที่ 4.8 และรูปที่ 4.9 ตามลำดับ



รูปที่ 4.8 แสดงสัญญาณที่ขาอินเทอร์รัปต์ขณะที่คานกระดกของเครื่องรับ มีการเคาะหลายครั้ง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

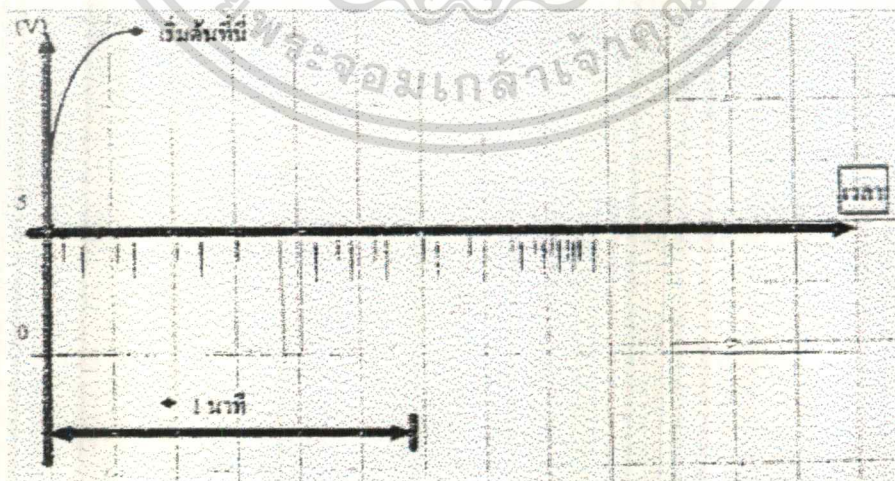


รูปที่ 4.9 แสดงการเปลี่ยนแปลงหน้าจอ LCD เมื่อมีสัญญาณเข้ามาที่ขาอินเทอร์รัปต์

4.3.2 ทำการต่อเครื่องวัดปริมาณน้ำฝน กับบอร์ดไมโครคอนโทรลเลอร์และต่อเครื่องบันทึกสัญญาณแบบกระดาษด้วย และทดสอบการทำงานของบอร์ดไมโครคอนโทรลเลอร์ โดยการหยดน้ำให้กับเครื่องวัดปริมาณน้ำฝน และสังเกตรูปสัญญาณที่เปลี่ยนแปลงบนกระดาษกราฟและทำการเปรียบเทียบกับบอร์ดไมโครคอนโทรลเลอร์ว่าสามารถแสดงค่าบนหน้าจอ LCD



รูปที่ 4.10 แสดงการเปลี่ยนแปลงข้อมูลบนจอ LCD

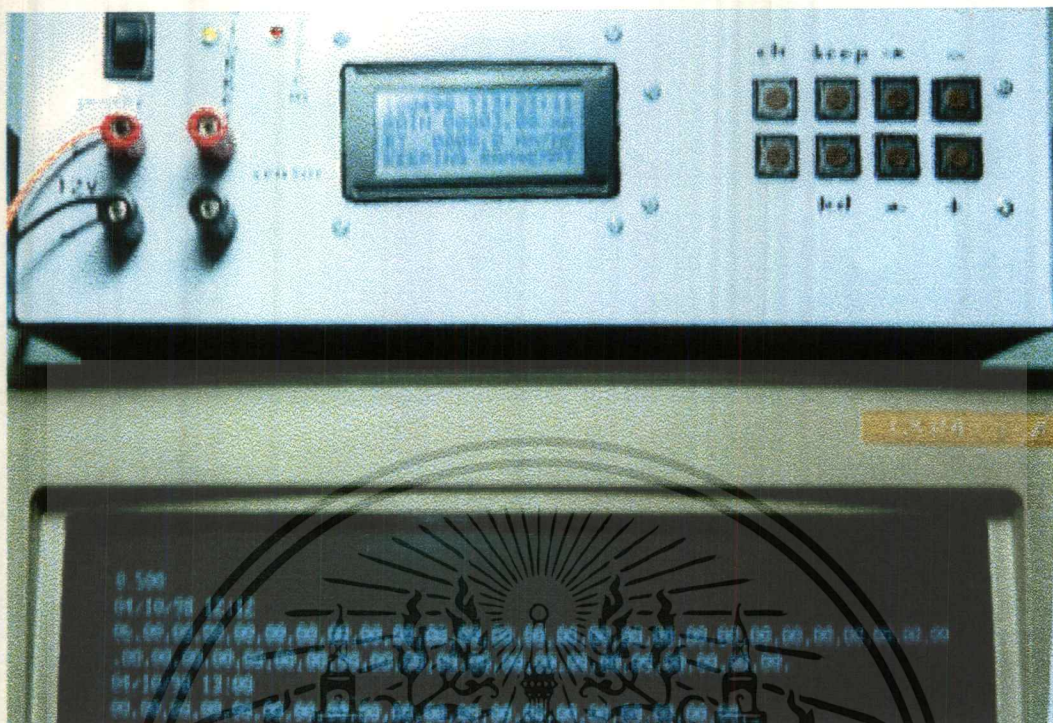


ความเร็วของกระดาษบันทึก 6 เซนติเมตร / นาที

รูปที่ 4.11 แสดงรูปสัญญาณที่บันทึกได้บนกระดาษกราฟ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

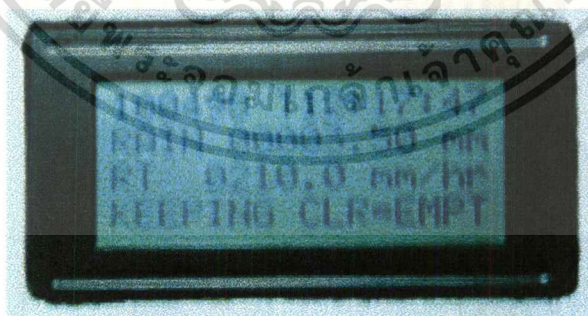
4.3.3 ทดลองทำการถ่ายข้อมูลที่เก็บในเมโมรี่ออกมาแสดงหน้าจอ โดยใช้โปรแกรมเอ็กซ์ทอลค์



รูปที่ 4.12 แสดงข้อมูลที่เครื่องไมโครคอนโทรลเลอร์บันทึกได้ โดยทำการส่งข้อมูลขึ้นบนคอมพิวเตอร์

4.4 ทดลองสวิตช์ควบคุมการทำงานของบอร์ดไมโครคอนโทรลเลอร์

4.4.1 ทดลองการทำงานของสวิตช์ควบคุมการลบข้อมูลในเมโมรี่ดังแสดงในรูปที่ 4.12 และแสดงการตรวจสอบรหัสผ่านดังรูปที่ 4.13 4.14 และ 4.15 ตามลำดับ

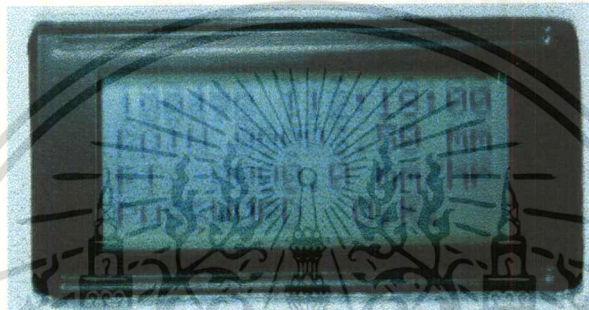


รูปที่ 4.13 แสดงหน้าจอ LCD เมื่อมีการกดคีย์ลบข้อมูลในเมโมรี่

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.14 แสดงการทำงานของโปรแกรม การลบข้อมูลในเมโมรี่ โดยจะมีการตรวจสอบรหัสผ่าน



รูปที่ 4.15 แสดงการทำงานการลบข้อมูลในเมโมรี่ โดยเป็นสภาวะหลังจากการมีการตรวจสอบรหัสผ่านว่าถูกต้อง



รูปที่ 4.16 แสดงการทำงานการลบข้อมูลในเมโมรี่ โดยเป็นสภาวะหลังจากการมีการตรวจสอบรหัสผ่านว่าไม่ถูกต้อง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.4.2 ทดสอบการทำงานของสวิตช์ควบคุมการบันทึกข้อมูลของปริมาณน้ำฝน เมื่อทำการกดแล้วจะขึ้นคำว่า KEEPING แสดงว่า อยู่ในสถานะกำลังเก็บข้อมูล



รูปที่ 4.17 แสดงหน้าจอ LCD หลังจากมีการกดปุ่มควบคุมการบันทึกข้อมูลของปริมาณน้ำฝน

4.5 ทดลอง โปรแกรมอิเล็กทรอนิกส์ที่ใช้แสดงรูปแบบการส่งข้อมูลออกทางพอร์ตอนุกรม โดยรวมผลการบันทึกข้อมูลของปริมาณน้ำฝนด้วย



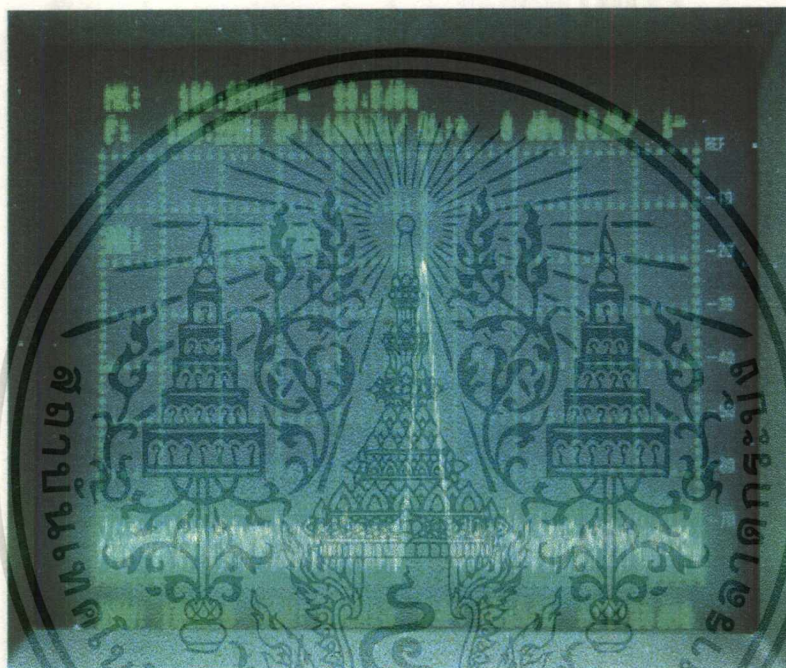
รูปที่ 4.18 แสดงตัวอย่าง โปรแกรมอิเล็กทรอนิกส์ที่ใช้แสดงรูปแบบข้อมูลที่ส่งออกทางพอร์ตอนุกรมแต่ละชุด

4.6 การทดลองชุดรับส่งข้อมูลแบบดิจิตอล

ชุดรับส่งข้อมูลดิจิตอลประกอบด้วย

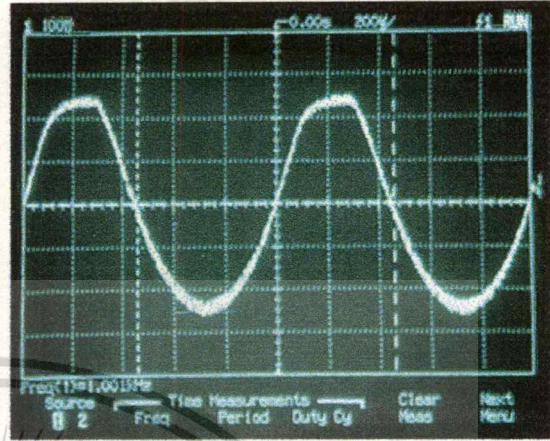
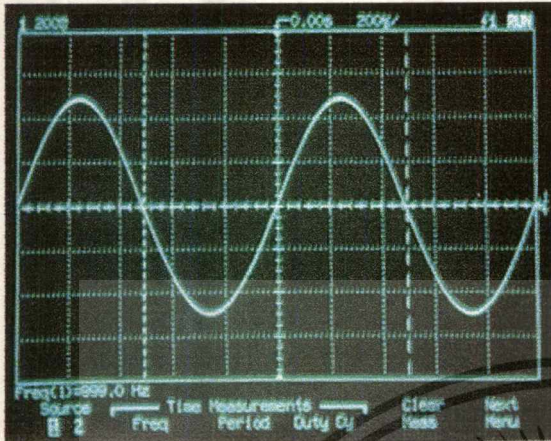
- FSK Modulator
- FM Transmitter
- FM Receiver
- FSK Demodulator

ในส่วนของการส่งทำการวัดสเปกตรัมของสัญญาณที่ส่งออกมาโดยใช้เครื่องสเปกตรัมมอานาไลซ์เซอร์ วัดสัญญาณที่เรดิโอออกมายังสายโพรบได้ดังนี้



รูปที่ 4.19 แสดงความถี่ของสัญญาณที่ส่งออกจากเครื่องส่ง

ทำการวัดสัญญาณที่เข้ามาทางเครื่องรับ ส่วนที่เป็น OUTPUT ที่ขา 18 ของ IC TA8122AN เปรียบเทียบกับสัญญาณอินพุตทางเครื่องส่ง

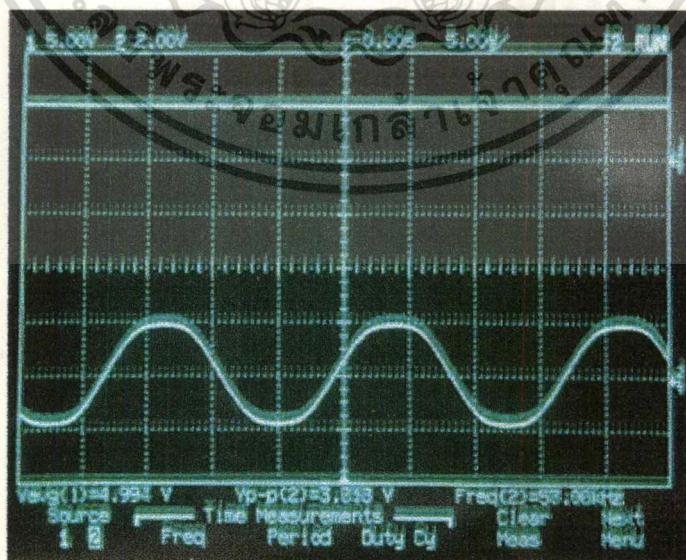


รูปที่ 4.20 แสดงสัญญาณอินพุตที่ป้อนให้เครื่องส่ง

รูปที่ 4.21 แสดงความถี่ OUTPUT ที่ IC TA8122AN

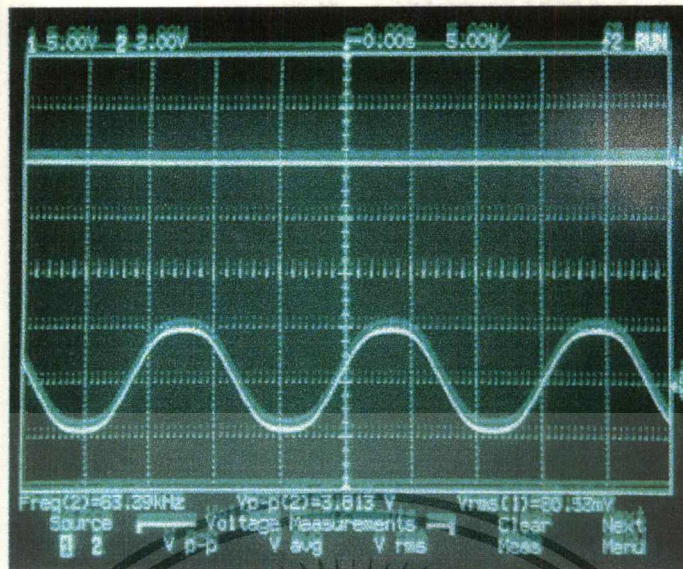
วงจรFSK Modulation

ทดลองโดยการทำการป้อนลอจิก “1” (ระดับแรงดัน 5V) เข้าที่อินพุตแล้ววัดระดับแรงดันและความถี่ที่เอาต์พุตของวงจร จะได้รูปสัญญาณที่วัดเทียบกับระหว่างอินพุตและเอาต์พุตดังรูปที่ 4.22 (โดยที่แชนแนลที่ 1 เป็นสัญญาณของอินพุต และแชนแนลที่ 2 เป็นสัญญาณของเอาต์พุต) จากนั้นทำการป้อนลอจิก “0” (ระดับแรงดัน 0 V) เข้าที่อินพุตแล้ววัดระดับแรงดันและความถี่ที่เอาต์พุตดังรูปที่ 4.23 (โดยที่แชนแนลที่ 1 เป็นสัญญาณของอินพุต และแชนแนลที่ 2 เป็นสัญญาณเอาต์พุต)



รูปที่ 4.22 แสดงสัญญาณเอาต์พุตของวงจร FSK Modulation เทียบกับอินพุตเมื่อป้อนลอจิก “1”

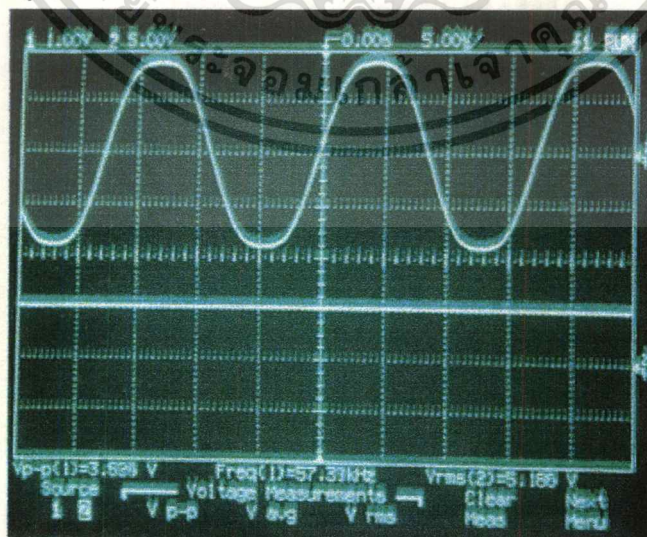
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.23 แสดงสัญญาณเอาต์พุตของวงจร FSK Modulation เทียบกับอินพุตเมื่อป้อนลอจิก “0”

วงจร FSK Demodulation

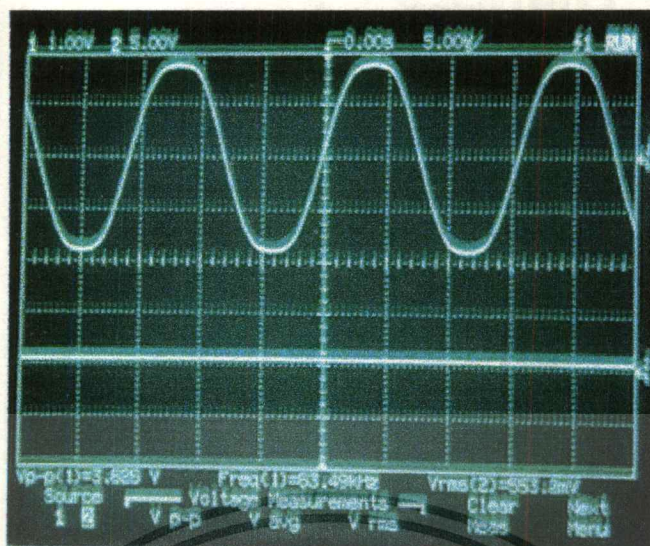
ทดลองโดยนำสัญญาณเอาต์พุตที่ออกจาก FSK Modulation ป้อนเข้าทางอินพุตของวงจร FSK Demodulation โดยป้อนเอาต์พุตของ FSK Modulation ที่ได้จากลอจิก “1” แล้ววัดเอาต์พุตของวงจร FSK Demodulation จะเห็นว่าได้รูปสัญญาณซึ่งวัดเทียบกับระหว่างอินพุตและเอาต์พุต ดังรูปที่ 4.24 (โดยเซนแนลที่ 1 เป็นรูปสัญญาณอินพุตและเซนแนลที่ 2 เป็นรูปสัญญาณเอาต์พุต) จากนั้นทำการป้อนเอาต์พุตของวงจร FSK Modulation ที่ได้จากลอจิก “0” แล้ววัดเอาต์พุตของวงจร FSK Demodulation จะเห็นว่าได้รูปสัญญาณซึ่งวัดเทียบกับระหว่างอินพุตและเอาต์พุต ดังรูปที่ 4.25 (โดยเซนแนลที่ 1 เป็นรูปสัญญาณอินพุต และเซนแนลที่ 2 เป็นรูปสัญญาณเอาต์พุต)



รูปที่ 4.24 แสดงสัญญาณเอาต์พุตของวงจร FSK Demodulation

เทียบกับเอาต์พุตของ FSK Modulation เมื่อป้อนลอจิก “1”

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น เมื่อนำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



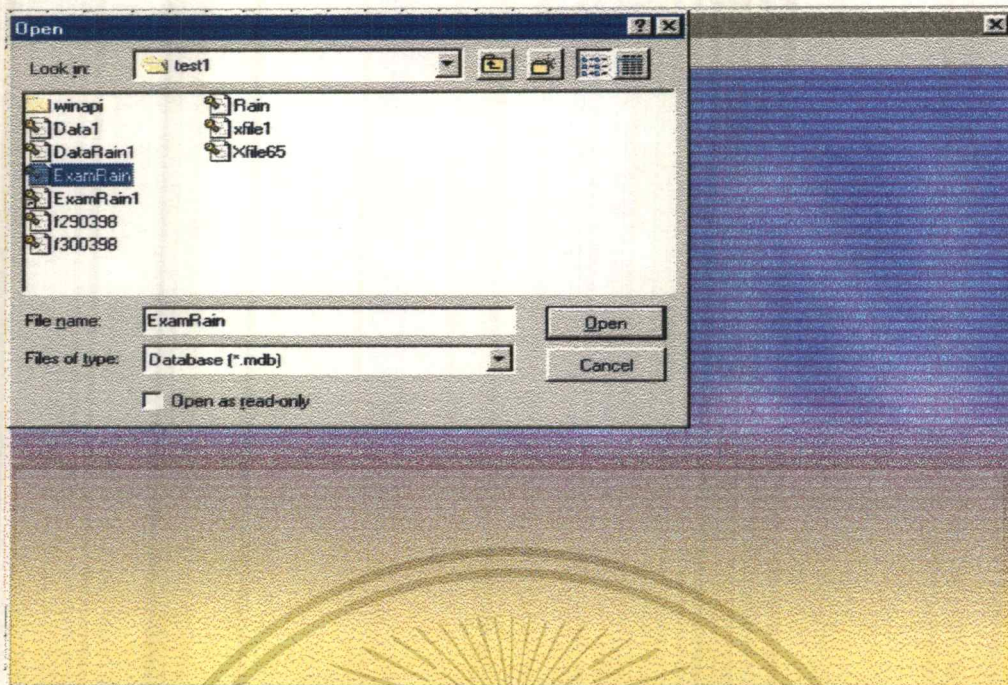
รูปที่ 4.25 แสดงสัญญาณเอาต์พุตของวงจร FSK Demodulation
เทียบกับเอาต์พุตของ FSK Modulation เมื่อป้อนลอจิก “0”

4.7 การทดลองโปรแกรมคอมพิวเตอร์ของการรับข้อมูลและแสดงกราฟข้อมูลปริมาณน้ำฝน

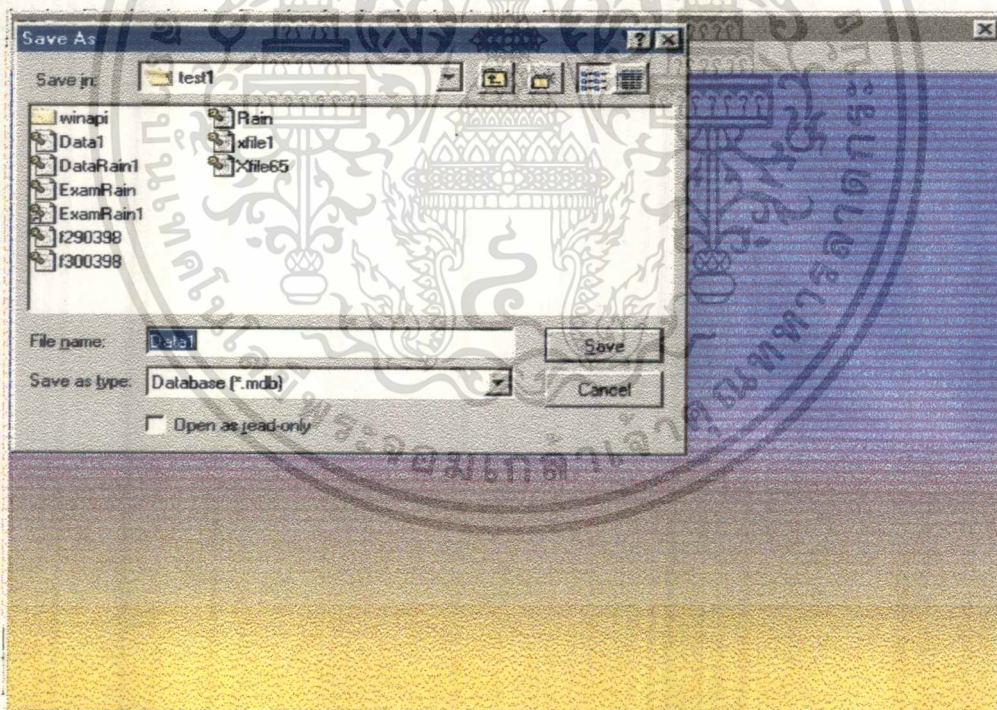


รูปที่ 4.26 ผลที่หน้าจอคอมพิวเตอร์เมื่อเปิดโปรแกรมการรับข้อมูล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

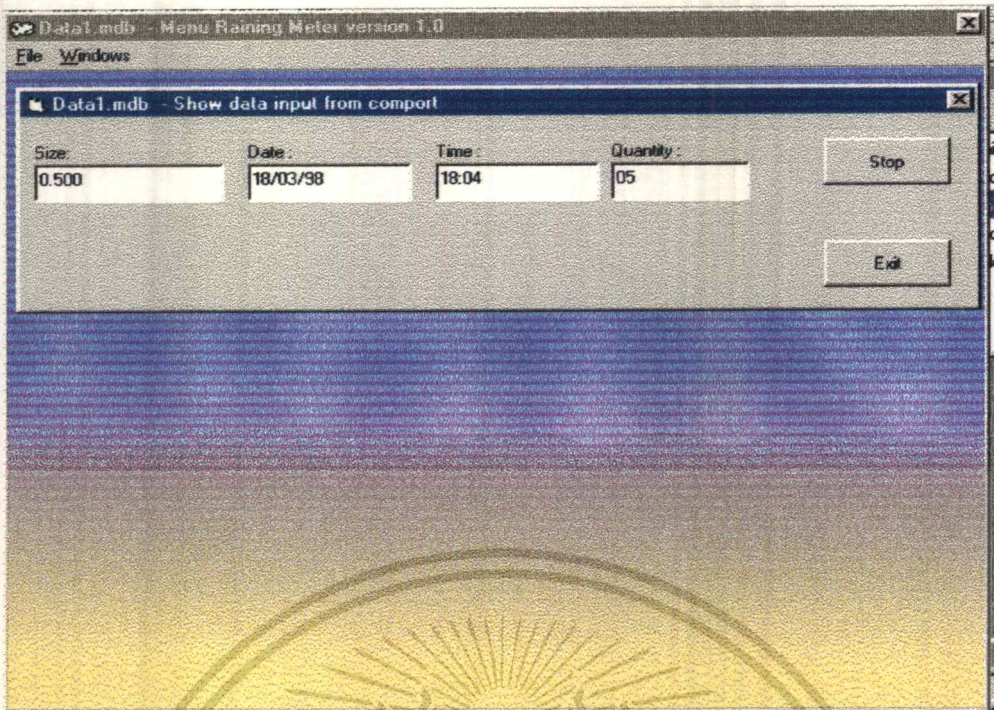


รูปที่ 4.27 ผลที่หน้าจอคอมพิวเตอร์เมื่อเปิด ไฟล์เพื่อแก้ไข เก็บข้อมูล และแสดงกราฟ

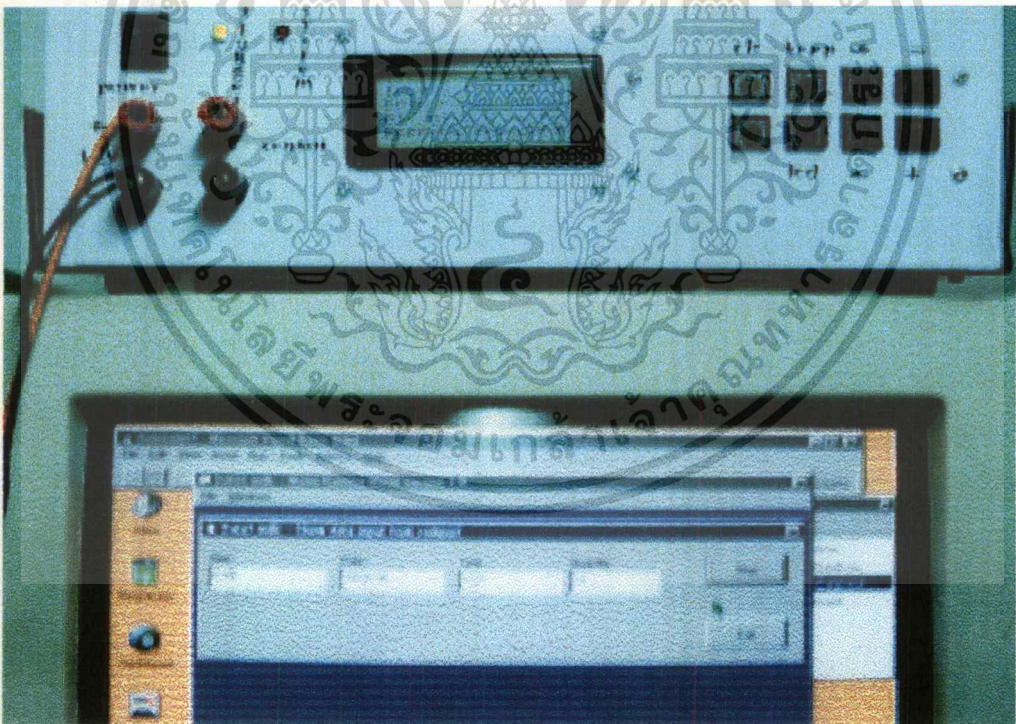


รูปที่ 4.28 ผลที่หน้าจอคอมพิวเตอร์เมื่อเก็บไฟล์ ซึ่งอยู่ในรูปแบบค่าตัวเลข

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

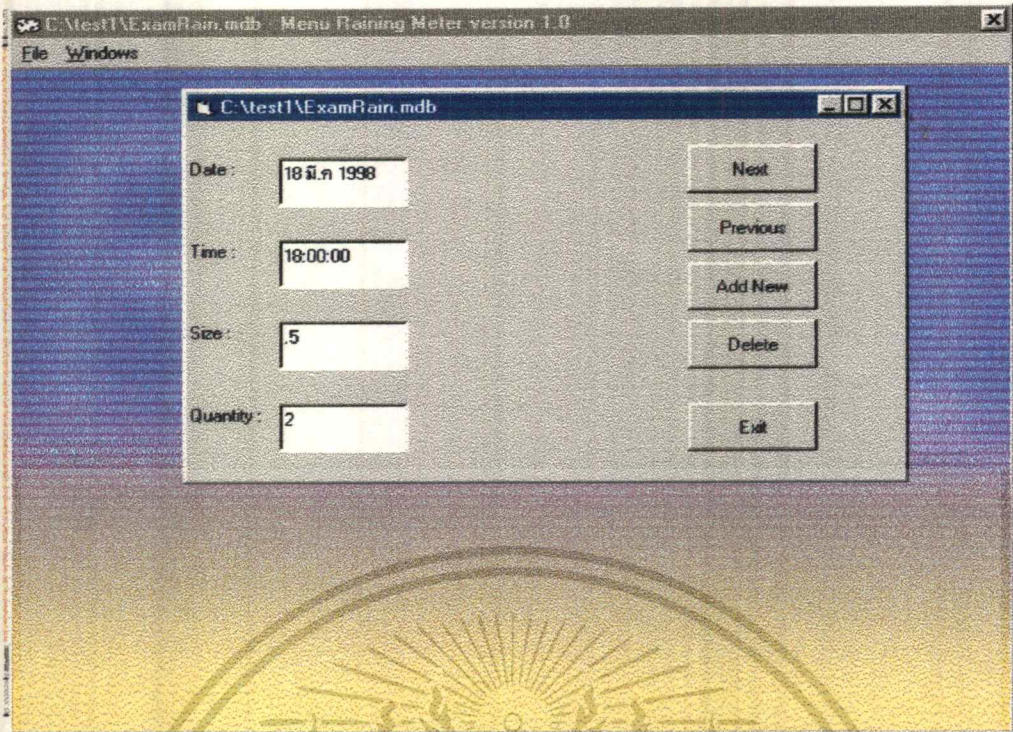


รูปที่ 4.29 ผลที่หน้าจอคอมพิวเตอร์เมื่อรับข้อมูลจากพอร์ตอนุกรม

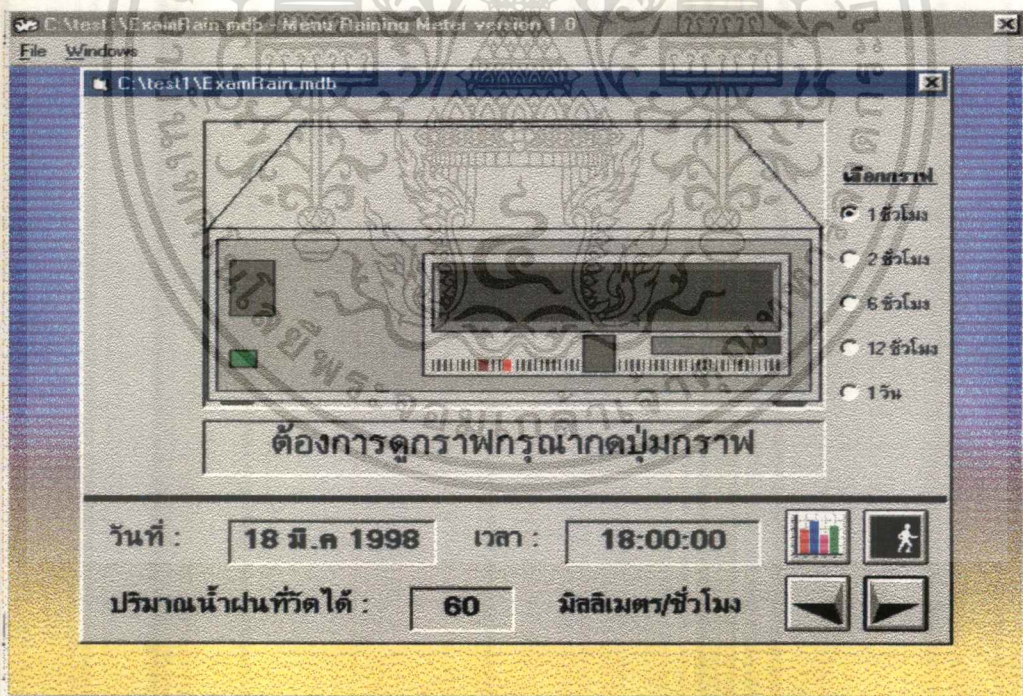


รูปที่ 4.30 แสดงการเปรียบเทียบการรับข้อมูลจากเครื่องกับที่หน้าจอคอมพิวเตอร์ ว่ามีการส่งข้อมูลมายังเครื่องรับ แล้วส่งมายังคอมพิวเตอร์อีกที เพื่อเก็บข้อมูลลงค่าเบสและแสดงกราฟต่อไป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

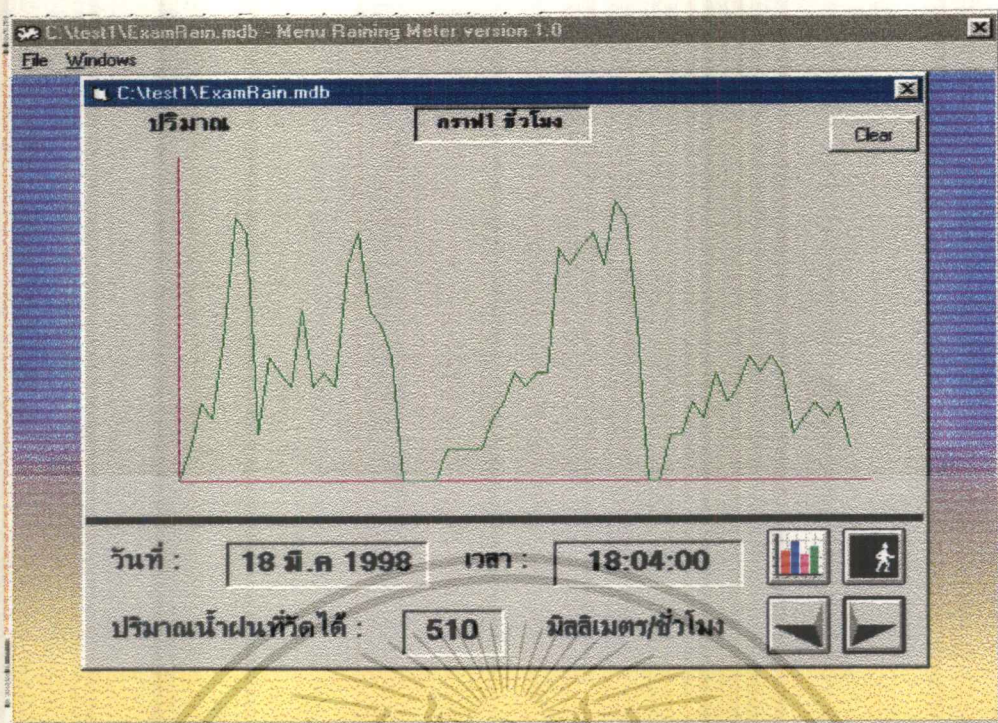


รูปที่ 4.31 ผลที่หน้าจอคอมพิวเตอร์เมื่อแสดงตารางค่าเบส เพื่อขอข้อมูลและแก้ไข



รูปที่ 4.32 ผลที่หน้าจอคอมพิวเตอร์เมื่อแสดงการเลือกกราฟที่จะดู

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.33 ผลที่หน้าจอคอมพิวเตอร์เมื่อแสดงผลกราฟปริมาณน้ำฝนในเวลา 1 ชั่วโมง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 5

บทวิจารณ์และบทสรุป

การทำงานของเครื่องบอกปริมาณน้ำฝนระยะไกลด้วยคลื่นวิทยุ นั้น เครื่องจะทำงานเทียบกับเวลาจริง ดังนั้นก่อนทำงาน จึงต้องป้อนค่าเวลา เพื่อให้เวลาที่ใช้นับทิกข้อมูลเป็นเวลาจริงอยู่เสมอ และในการป้อนจะใช้ปุ่มในการป้อนข้อมูล เมื่อป้อนเวลาจนครบก็จะนำเวลาป้อนให้กับ DS 1202 ที่ทำหน้าที่เป็นนาฬิกา

การเก็บข้อมูลเราจะตั้งค่าขนาดของอุปกรณ์วัดปริมาณน้ำฝนแบบคานกระดกว่ามีขนาดกี่มิลลิเมตร เพื่อให้ข้อมูลที่ได้นับตรงกับค่าของขนาดอุปกรณ์ที่ใช้ เมื่อเครื่องทำการนับทิกข้อมูลจนเต็มแล้ว จะไม่ทำการนับทิกข้อมูลที่ส่งมาได้อีก แต่สัญญาณนาฬิกาจะยังคงเดินต่อไปจนกว่าจะมีการถ่ายข้อมูลออกจากหน่วยความจำจึงจะสามารถเก็บข้อมูลได้อีก

จากการทดลองต่อเครื่องวัดปริมาณน้ำฝนกับบอร์ดไมโครคอนโทรลเลอร์ได้ผลคือ เมื่อทำการเทน้ำใส่ถังของเครื่องวัดปริมาณน้ำฝน จะมีสัญญาณเข้ามาที่ขาอินเทอร์รัปต์ 0 ของ 8032 เมื่อข้อมูลที่เข้ามาแล้วสามารถนับทิกข้อมูลและแสดงผลหน้าจอแอลซีดีได้ จากผลที่ได้จะเห็นว่าสามารถทำงานได้ใกล้เคียงกับเครื่องนับทิกข้อมูลปริมาณน้ำฝน

จากนั้นจะทำการส่งข้อมูลที่เก็บไว้ในหน่วยความจำ ไปยังวงจรเข้ารหัสเพื่อส่งไปยังเครื่องส่งวิทยุ และสร้างเครื่องรับเพื่อรับข้อมูลที่ส่งมาและส่งต่อไปยังวงจรถอดรหัส แล้วต่อไปยังคอมพิวเตอร์เพื่อนำไปวิเคราะห์แสดงผลทางหน้าจอ

การพิจารณาเลือกความถี่ของ FSK Modulator จากการทดลองใช้ไอซีเบอร์ XR 2206 เป็น FSK Modulator และใช้ไอซีเบอร์ XR 2211 เป็น FSK Demodulator

จากการทดลองใช้ความถี่ของ FSK Modulator ดังนี้

$$\text{"0"} = 9,600 \text{ Hz}$$

$$\text{"1"} = 19,200 \text{ Hz}$$

แต่เมื่อทำการทดลองจะพบว่าไม่สามารถที่จะทำการรับส่งข้อมูลที่อัตราเร็ว 9,600 บิตต่อวินาทีได้ จึงได้ทำการทดลองต่อไป โดยการเพิ่มความถี่ที่ใช้ในการรับส่งข้อมูลขึ้นอีกและผลที่ได้รับคือ ควรใช้ค่าความถี่ที่ใช้ในการรับส่งข้อมูลที่มีค่าเป็น 6 เท่าของอัตราเร็ว นั่นคือ ที่ 9,600 บิตต่อวินาที

$$\text{"1"} = 60 \text{ kHz}$$

$$\text{"0"} = 65 \text{ kHz}$$

ซึ่งจากการทดลองรับและส่งข้อมูลที่อัตราเร็ว 9,600 บิตต่อวินาที โดยใช้ค่าความถี่ดังกล่าวสามารถที่จะรับส่งข้อมูลได้อย่างถูกต้อง

การเลือกความถี่ของภาคส่งใช้คลื่นวิทยุซึ่งมีการมอดูเลททางความถี่ โดยใช้ความถี่ที่ 108.25 MHz เนื่องจากความถี่นี้อยู่ห่างจากสถานีส่ง FM กระจายเสียง แต่จะมีปัญหาการถูกกวนเล็กน้อยจากสถานี 107.5 MHz วงจรที่ใช้เกิดปัญหาคือมีสัญญาณรบกวนมาก ดังนั้นจึงใส่ภาคปริแอมป์เพื่อขยายสัญญาณที่จะส่งให้มีขนาดมากกว่าสัญญาณรบกวน ค่าตั้งที่ใช้กำลังต่ำทำให้ความเข้มของสัญญาณไม่มากพอที่ส่งไปได้ไกล จำเป็นต้องเพิ่มวงจรบูสเตอร์เข้าไป

ทางด้าน ภาครับใช้ IC TA8122AN ซึ่ง เป็นจูนเนอร์ที่เป็นมาตรฐานที่สามารถถอดสัญญาณความถี่ประมาณ 60-70 KHz ซึ่งเป็นสัญญาณที่ส่งมา และมีการใส่ภาคขยายเข้าไปเพื่อขยายให้พอกับวงจรถอดรหัส ปัญหาที่เกิดขึ้นคือสัญญาณที่มีความถี่ต่ำกว่า 60 KHz เข้ามาด้วยทำให้บางครั้งมีผลต่อวงจรถอดรหัส อาจจะต้องใส่วงจรแบนด์พาสฟิลเตอร์เพื่อที่จะกรองเอาความถี่ที่ต้องการเข้ามาเท่านั้น



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

; FILENAME      RAIN.ASM
; DESCRIPTION   GET START TIME FROM KEYBOARD (UP-DOWN KEYBOARD)
;              & REAL TIME TO LCD MODULE
;              & GET INTERRUPT SIGNAL TO SUB ROUTINE
;              & VALUE OF RAIN TO LCD MODULE
;              & WRITE DATA TO RAM
;              & UPLOAD DATA TO PC
;              & PASSWORD FOR CLEAR RAM
;              & SEND DATA OF RAIN TO COMPUTER EVERY MINUTE
; HARDWARE      - ANT-32 V3.0 FULL OPTION
;              - DS1202
;              - LCD164 DMC
; ASSEMBLER     SXA51
;
; ***** VARIABLE *****
; ***** PORT *****

DATABI EQU P1.4
CLKBIT EQU P1.5
RSTBIT EQU P1.6

; ***** PARAMETER *****
LCDWRC EQU 0FA00H ;LCD WRITE INSTRUCTION
LCDRDC EQU 0FA01H ;LCD READ INSTRUCTION (CHECK BUSY FLAG)
LCDWRD EQU 0FA02H ;LCD WRITE DATA
LCDRDD EQU 0FA03H ;LCD READ DATA
P1A EQU 0F800H ;USER PORT 1 PORT A
P1B EQU 0F801H ;USER PORT 1 PORT B
P1C EQU 0F802H ;USER PORT 1 PORT C
P1P EQU 0F803H ;USER PORT 1 CONTROL PORT
P2A EQU 0FC00H ;USER PORT 2 PORT A
P2B EQU 0FC01H ;USER PORT 2 PORT B
P2C EQU 0FC02H ;USER PORT 2 PORT C
P2P EQU 0FC03H ;USER PORT 2 CONTROL PORT
;BRAT12 EQU 0E8H ;1200 BAUD RATE VAR
;BRAT24 EQU 0F4H ;2400
;BRAT48 EQU 0FAH ;4800
BRAT96 EQU 0FDH ;9600
ADLINE1 EQU 80H ;ADDRESS LINE1
ADLINE2 EQU 0C0H ;ADDRESS LINE2
ADLINE3 EQU 90H ;ADDRESS LINE3
ADLINE4 EQU 0D0H ;ADDRESS LINE4
;
; ***** INTERNAL RAM *****
;
; ORG 00H
;
; DS 8 ;REGISTOR BANK-0
DISBUF: DS 64 ;DISPLAY BUFFER (SEGMENT CODE)
SECBUF: DS 1 ;SECOND BUFFER
MINBUF: DS 1 ;MINUTE BUFFER
HOURBUF: DS 1 ;HOUR BUFFER
TIMLCD: DS 1 ;TIME FOR TURN OFF LCD
RAIN: DS 5 ;VALUE RAIN
PULSE: DS 1 ;PULSE OF RAIN IN EACH MINUTE
STATUS: DS 1 ;STATUS BUFFER
RHADRAM: DS 1 ;RAM ADDRESS (RAIN)
RLADRAM: DS 1 ;RAM ADDRESS
DISBUFS: DS 16 ;DISPLAY BUFFER
CLER: DS 1 ;CLEAR BUFFER
SIZEGAGE: DS 2 ;RAIN GAGE SIZE
ADDRL: DS 1 ;KEEP ADDRESS LOW FOR SEND
ADDRH: DS 1 ;KEEP ADDRESS HIGH FOR SEND
SYSSTK: DS 48
INTEND:
;
; ***** RESET *****

```

เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์หรือการใชงานเพื่อการศึกษาเท่านั้น ไมอนุญาตให้นำไปใชประโยชน์ดานการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

ORG    0000H
SJMP   MAIN

;

ORG    0003H
LJMP   INTER0

;

ORG    000BH
RETI

;

ORG    0013H
LJMP   INTER1

;

ORG    001BH
RETI

;

ORG    0023H
RETI

```

```

; ***** MAIN *****
;
MAIN:  MOV    R2,#40H                ;POWER UP DELAY
RES1:  MOV    R3,#0
        DJNZ  R3,$
        DJNZ  R2,RES1
        MOV   SP,#SYSSTK
        MOV   R0,#08H                ;CLEAR INT-RAM (ALL)
        MOV   R2,#INTEND-08H
RES2:  MOV   @R0,#0
        INC   R0
        DJNZ  R2,RES2
        LCALL LCDINI                ;LCD INITIALIZE
        MOV   R2,#20                ;DELAY
        LCALL DTSEC
        LCALL CURSOF
        LCALL CLRDIS                ;CLEAR DISPLAY
        MOV   DPTR,#7FFEH           ;OLD ADDRESS BEFORE TURN OFF
        MOVX  A,@DPTR
        MOV   RHADRAM,A
        INC   DPTR
        MOVX  A,@DPTR
        MOV   RLADRAM,A
        MOV   TIMLCD,#02H           ;SET START TIME FOR LCD
        MOV   CLER,#02H             ;SET 2 PULSE FOR CLEAR
        MOV   STATUS,#00H           ;CLEAR STATUS
        LCALL DEMO
        LCALL GAGESIZE
        LCALL SCHAR_LCD             ;SET CHARACTER ON LCD DISPLAY
        LCALL KEYBOARD              ;GET START TIME FROM KEYBOARD
        LCALL WR_START TIME
        MOV   IE,#10000000B
        SETB  IE.2
        MOV   A,STATUS              ;KEEP STATUS LCD FOR CHECKING
        SETB  ACC.7
        MOV   STATUS,A
        MOV   DISBUF+44,#00H        ;USE FOR RATE SUB
        MOV   DISBUF+45,#00H
        MOV   DISBUF+46,#00H
;
; ***** LOOP *****
;
RTC:   MOV    DPTR,#P1A              ;GET DATA FROM KEYBOARD
        MOVX  A,@DPTR
        CJNE  A,#00H,KL0
        SJMP  RTC0
KL0:   CJNE  A,#11111111B,KL1        ;CHECK KEYBOARD
RTC0:  MOV    R2,#81H               ;READ TIME LOOP

```

เอกสารนี้เป็นเอกสารที่จัดทำขึ้นเพื่อการศึกษาค้นคว้าเท่านั้น ไม่ควรนำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

LCALL  RTCRD
MOV    A,R3                                ;SECOND
CJNE   A,SECBUF,RTC1
SJMP   RTC
KL1:   LJMP  KEYLOOP
RTC1:  MOV   SECBUF,A
LCALL  HTOA
MOV    DISBUF+14,R2
MOV    DISBUF+15,R3
MOV    R2,#83H                            ;MINUTE
LCALL  RTCRD
MOV    A,R3
CJNE   A,MINBUF,RTC2
LJMP   RTC4
RTC2:  MOV   MINBUF,A
LCALL  HTOA
MOV    DISBUF+11,R2
MOV    DISBUF+12,R3
JNB    IE.0,RTC3                          ;ABOUT KEEPING RAIN DATA
PUSH   DPH
PUSH   DPL
MOV    DPH,RHADRAM
MOV    DPL,RLADRAM
MOV    A,PULSE
LCALL  HEXTOBCD
LCALL  CH_RAM
LCALL  SENDDATA
MOV    RHADRAM,DPH
MOV    RLADRAM,DPL
POP    DPL
POP    DPH
MOV    DISBUF+44,#'m'
MOV    DISBUF+45,#'/'
MOV    DISBUF+46,#'h'
LCALL  LCDSL3
MOV    DISBUF+44,#00H                      ;USE FOR RATE SUB
MOV    DISBUF+45,#00H
MOV    DISBUF+46,#00H
MOV    DISBUF+36,#'0'
MOV    DISBUF+37,#'0'
MOV    DISBUF+38,#'0'
MOV    DISBUF+39,#'0'
MOV    DISBUF+41,#'0'
MOV    PULSE,#00H
RTC3:  DJNZ  TIMLCD,RTC4                  ;ABOUT TURN OFF LCD
LCALL  DISPOF
MOV    A,STATUS
CLR    ACC.7
MOV    STATUS,A
MOV    CLER,#02H                          ;SET 2 PULSE FOR CLEAR
MOV    DISBUF+56,#'R'
MOV    DISBUF+57,#'A'
MOV    DISBUF+58,#'M'
LCALL  LCDSL4
RTC4:  MOV   R2,#85H                      ;HOUR
LCALL  RTCRD
MOV    A,R3
CJNE   A,HOURLBUF,RTC5
SJMP   RTC6
RTC5:  MOV   HOURLBUF,A
LCALL  HTOA
MOV    DISBUF+8,R2
MOV    DISBUF+9,R3
JNB    IE.0,RTC6
RTC6:  LCALL  KEEPFC
MOV    R2,#87H                            ;DATE

```

```

        LCALL  RTCRD
        MOV    A,R3
        LCALL  HTOA
        MOV    DISBUF,R2
        MOV    DISBUF+1,R3
RTC7:   MOV    R2,#89H                ;MONTH
        LCALL  RTCRD
        MOV    A,R3
        LCALL  HTOA
        MOV    DISBUF+2,R2
        MOV    DISBUF+3,R3
        MOV    R2,#8DH                ;YEAR
        LCALL  RTCRD
        MOV    A,R3
        LCALL  HTOA
        MOV    DISBUF+4,R2
        MOV    DISBUF+5,R3
RTC8:   MOV    DISBUF+6,#' '
        MOV    DISBUF+7,#'T'
        MOV    DISBUF+10,#':'
        MOV    DISBUF+13,#':'
        MOV    DPTR,#7FFEH

```

```

;
;      ; **** OLD ADDRESS BEFORE TURN OFF ****
;

```

```

        MOV    A,RHADRAM
        MOVX   @DPTR,A
        INC    DPTR
        MOV    A,RLADRAM
        MOVX   @DPTR,A
        LCALL  LCDSL1
        LCALL  LCDSL2
        LCALL  LCDSL4
        LJMP   RTC

```

```

;
; ***** END MAIN *****
;

```

```

; ***** SUB PROGRAM *****
;

```

```

; ***** DEMO *****
;

```

```

DEMO:   LCALL  CREATEDEMO
        LCALL  CREATECHAR
        MOV    DISBUF+50,#01H
        MOV    DISBUF+51,#02H
        MOV    DISBUF+52,#03H
        LCALL  LCDSL41
        LCALL  KDELAY
        LCALL  KDELAY
        LCALL  DISPOF
        LCALL  KDELAY
        LCALL  KDELAY
        LCALL  DISPON
        LCALL  KDELAY
        LCALL  KDELAY
        LCALL  DISPOF
        LCALL  KDELAY
        LCALL  KDELAY
        LCALL  DISPON
        LCALL  KDELAY
        LCALL  KDELAY
        RET

```

```

; ***** KEEPFC *****
;

```

```

; KEEP CODE FC FOR CHANGING HOUR & KEEP TIME
;

```

เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์ไว้เพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

KEEPFC:  MOV    DPH,RHADRAM          ;RAM ADDRESS
          MOV    DPL,RLADRAM
          MOV    A,#0FCH              ;KEEP CHANGE HOUR CODE,DATE AND TIME
          LCALL  CH_RAM
          LCALL  K_DT                 ;KEEP DATE & TIME
          MOV    RHADRAM,DPH
          MOV    RLADRAM,DPL
          RET

;
; ***** KEYLOOP *****
; LOOP FOR CHECK DATA FROM KEYBOARD
;
KEYLOOP:  ORL    A,#01001111B
          CJNE   A,#11111111B,CH_HKEY
          LJMP   RTC0
CH_HKEY:  MOV    TIMLCD,#02H          ;SET TIME FOR TURN OF LCD
          LCALL  KDELAY
          CJNE   A,#11101111B,CH_LCD  ;CHECK DATA FROM KEYBOARD
ONLCD:   LCALL  DISPON               ;TURN ON LCD
          MOV    A,STATUS              ;KEEP STATUS LCD FOR CHECKING
          SETB   ACC.7
          MOV    STATUS,A
          LJMP   RTC0
CH_LCD:  MOV    R7,A                 ;CHECK LCD STATUS
          MOV    A,STATUS
          JB     ACC.7,CH_CRAM
          LJMP   RTC0
CH_CRAM: MOV    A,R7                 ;CHECK CLEAR RAM
          CJNE   A,#01111111B,CH_ORAI
          MOV    DISBUF+56,#'C'
          MOV    DISBUF+57,#'L'
          MOV    DISBUF+58,#'R'
          LCALL  LCDSL4
          DJNZ   CLER,CR
          LJMP   CLRRAM
CR:      LJMP   RTC0
CH_ORAI: JB     IE.0,OFFRAIN          ;CHECK KEEPING DATA STATUS
          MOV    A,STATUS              ;CHECK RAM OF RAIN
          JB     ACC.4,ONR
          MOV    DPH,RHADRAM          ;RAM ADDRESS
          MOV    DPL,RLADRAM
          MOV    A,#0F0H              ;KEEP START CODE,DATE AND TIME
          LCALL  CH_RAM
          LCALL  K_DT                 ;KEEP DATE & TIME
          MOV    RHADRAM,DPH
          MOV    RLADRAM,DPL
          SETB   IE.0
          MOV    RAIN,#00H            ;CLEAR VALUE OF RAIN ON LCD
          MOV    RAIN+1,#00H
          MOV    RAIN+2,#00H
          MOV    RAIN+3,#00H
          MOV    RAIN+4,#00H
          MOV    PULSE,#00H           ;CLEAR PULSE BUFFER
          MOV    DISBUF+22,#'0'
          MOV    DISBUF+23,#'0'
          MOV    DISBUF+24,#'0'
          MOV    DISBUF+25,#'0'
          MOV    DISBUF+27,#'0'
          MOV    DISBUF+28,#'0'
          LCALL  LCDSL2
          LCALL  KEEPING
          LCALL  LCDSL4
          LJMP   RTC0
OFFRAIN: CLR    IE.0
          MOV    DPH,RHADRAM          ;RAM ADDRESS
          MOV    DPL,RLADRAM

```

```

MOV     A,PULSE
LCALL   HEXTOBCD
LCALL   CH RAM
MOV     RHADRAM,DPH
MOV     RLADRAM,DPL
LCALL   NONKEEP
LCALL   LCDSL4
ONR:     LJMP   RTC0
;
; ***** CLRRAM SUB *****
; CLEAR RAM
; REG = A,R0-R7
;
CLRRAM: CLR     IE.0
                CLR     IE.2
                PUSH ACC
                PUSH 00H
                PUSH 01H
                PUSH 02H
                PUSH 03H
                PUSH 04H
                PUSH 05H
                PUSH 06H
                PUSH 07H
                MOV     DISBUFS,#'P' ;PASSWORD
MOV     DISBUFS+1,#'A'
MOV     DISBUFS+2,#'S'
MOV     DISBUFS+3,#'S'
MOV     DISBUFS+4,#'W'
MOV     DISBUFS+5,#'O'
MOV     DISBUFS+6,#'R'
MOV     DISBUFS+7,#'D'
MOV     DISBUFS+8,#' '
MOV     DISBUFS+9,#' '
MOV     DISBUFS+10,#' '
MOV     DISBUFS+11,#' '
MOV     DISBUFS+12,#' '
MOV     DISBUFS+13,#' '
MOV     DISBUFS+14,#' '
MOV     DISBUFS+15,#' '
LCALL   LCDSL5
MOV     DISBUFS+9,#'0'
MOV     R0,#DISBUF
MOV     R1,#DISBUFS+9
LCALL   LCDSL5
GET:     MOV     DPTR,#P1A ;GET DATA FROM KEYBOARD
MOVX    A,@DPTR
CJNE    A,#00H,GETJ
SJMP    GET
GETJ:    ORL     A,#11110100B
CJNE    A,#11111111B,GETCH
SJMP    GET ;LOOP FOR GET DATA
GETCH:   LCALL   KDELAY ;DELAY
CJNE    A,#11111110B,GETCH1
MOV     A,@R1 ;INCREMENT NUMBER
INC     A
CJNE    A,#3AH,GETS11
MOV     A,#30H
GETS11:  MOV     @R1,A
LCALL   LCDSL5 ;SHOW DATA IN LINE S
SJMP    GET
GETCH1:  CJNE    A,#11111101B,GETCH2
MOV     A,@R1 ;DECREMENT NUMBER
DEC     A
CJNE    A,#2FH,GETS21
MOV     A,#39H

```

```

GETS21:      MOV    @R1,A
             LCALL   LCDSLS
             SJMP    GET
GETCH2: CJNE   R1,#DISBUFS+15,GETS4      ;CHECK POSITION
             MOV    A,@R1
             CJNE   A,'#0',UNCORRECT
             DEC    R0
             MOV    A,@R0
             CJNE   A,'#0',UNCORRECT    ;CHECK PASSWORD
             DEC    R0
             MOV    A,@R0
             CJNE   A,'#0',UNCORRECT
             DEC    R0
             MOV    A,@R0
             CJNE   A,'#0',UNCORRECT
             DEC    R0
             MOV    A,@R0
             CJNE   A,'#0',UNCORRECT
             DEC    R0
             MOV    A,@R0
             CJNE   A,'#0',UNCORRECT
             DEC    R0
             MOV    A,@R0
             CJNE   A,'#1',UNCORRECT
             SJMP    CLRRAM1              ;PASSWORD IS CORRECT (CLEAR)
GETS4:      MOV    A,@R1                  ;SHIFT RIGHT
             MOV    @R0,A
             MOV    @R1,#0FFH
             INC    R1
             INC    R0
             MOV    @R1,'#0'
             LCALL   LCDSLS
             LJMP    GET
;
; ***** UNCORRECT *****
;
UNCORRECT:SETB IE.2
MOV    DISBUFS,'#*'
MOV    DISBUFS+1,'#*'
MOV    DISBUFS+2,'#*'
MOV    DISBUFS+3,'#U'
MOV    DISBUFS+4,'#N'
MOV    DISBUFS+5,'#C'
MOV    DISBUFS+6,'#O'
MOV    DISBUFS+7,'#R'
MOV    DISBUFS+8,'#R'
MOV    DISBUFS+9,'#E'
MOV    DISBUFS+10,'#C'
MOV    DISBUFS+11,'#T'
MOV    DISBUFS+12,'# '
MOV    DISBUFS+13,'#*'
MOV    DISBUFS+14,'#*'
MOV    DISBUFS+15,'# '
LCALL   LCDSLS
LCALL   KDELAY
LCALL   KDELAY
MOV    CLER,#02H
MOV    DISBUF+56,'#R'
MOV    DISBUF+57,'#A'
MOV    DISBUF+58,'#M'
POP    07H
POP    06H
POP    05H
POP    04H
POP    03H
POP    02H

```

เอกสารนี้เป็นเอกสารที่จัดทำขึ้นไว้สำหรับกรใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

POP    01H
POP    00H
POP    ACC
LCALL  LCDSL4
SETB   IE.2
LJMP   RTC0
; *****CLEAR*****
;
CLR RAM1: MOV    DISBUFS+9, #' '
MOV    DISBUFS+10, #'O'
MOV    DISBUFS+11, #'.'
MOV    DISBUFS+12, #'K'
MOV    DISBUFS+13, #'.'
MOV    DISBUFS+14, #' '
MOV    DISBUFS+15, #' '
LCALL  LCDSL5
LCALL  KDELAY
LCALL  KDELAY
MOV    DPTR, #0000H
MOV    A, #0EEH
CLEAR:  MOVX     @DPTR, A                ;CLEAR RAM
INC     DPTR
MOV     R6, DPH
CJNE    R6, #80H, CLEAR
MOV     A, STATUS                ;CLEAR STATUS FOR STARTING
CLR     ACC.3
CLR     ACC.4
MOV     STATUS, A
CLR     IE.0                    ; CLEAR OPERATING RAIN DATA
LCALL   NONKEEP                ;CLEAR LCD FOR STARTING
MOV     DISBUF+55, #' '
LCALL   RAMEMPTY
LCALL   LCDSL4
MOV     RAIN, #00H
MOV     RAIN+1, #00H
MOV     RAIN+2, #00H
MOV     RHADRAM, #00H            ;SET ADDRESS RAM FOR START
MOV     RLADRAM, #00H
MOV     CLER, #02H
POP     07H
POP     06H
POP     05H
POP     04H
POP     03H
POP     02H
POP     01H
POP     00H
POP     ACC
SETB    IE.2
LJMP    RTC0
;
;***** DTOH *****
;DEC TO HEX
;REG = IN ACC , OUT ACC
;
DTOH:  PUSH    06H
        PUSH    07H
        MOV     R7, A
        SWAP    A
        ANL     A, #0FH
        MOV     B, #0AH
        MUL     AB
        MOV     R6, A
        MOV     A, R7
        ANL     A, #0FH
        ADD     A, R6

```

```

POP 07H
POP 06H
RET

```

```

; ***** KEEP DATE & TIME *****
;

```

```

K_DT: PUSH 02H
      PUSH 03H
MM:   MOV     R2,#89H           ;MONTH
      LCALL   RTCRD
      MOV     A,R3
      LCALL   CH_RAM
DD:   MOV     R2,#87H           ;DATE
      LCALL   RTCRD
      MOV     A,R3
      LCALL   CH_RAM
YY:   MOV     R2,#8DH           ;YEAR
      LCALL   RTCRD
      MOV     A,R3
      LCALL   CH_RAM
HH:   MOV     R2,#85H           ;HOUR
      LCALL   RTCRD
      MOV     A,R3
      LCALL   CH_RAM
MIN:  MOV     R2,#83H           ;MINUTE
      LCALL   RTCRD
      MOV     A,R3
      LCALL   CH_RAM
      POP     03H
      POP     02H
      RET

```

```

; ***** CH_RAM *****
; CHECK RAM (RAIN)
;

```

```

CH_RAM: MOVX   @DPTR,A
        MOV     ADDRL,DPL
        MOV     ADDRH,DPH
        INC     DPTR
RAM1:   MOV     R2,DPH
        MOV     R3,DPL
        CJNE    R3,#00H,BELOW
        CJNE    R2,#10H,RAM2
        MOV     DISBUF+60,'#1'
        MOV     DISBUF+61,'#/'
        MOV     DISBUF+62,'#8'
        MOV     DISBUF+63,'# '
        LCALL   LCDSL4
        RET

```

```

BELOW:  LJMP    CH_RAM1
RAM2:   CJNE    R2,#20H,RAM3
        MOV     DISBUF+60,'#2'
        MOV     DISBUF+61,'#/'
        MOV     DISBUF+62,'#8'
        MOV     DISBUF+63,'# '
        LCALL   LCDSL4
        RET

```

```

RAM3:   CJNE    R2,#30H,RAM4
        MOV     DISBUF+60,'#3'
        MOV     DISBUF+61,'#/'
        MOV     DISBUF+62,'#8'
        MOV     DISBUF+63,'# '
        LCALL   LCDSL4
        RET

```

```

RAM4:   CJNE    R2,#40H,RAM5
        MOV     DISBUF+60,'#4'

```

```

MOV    DISBUF+61,# '/'
MOV    DISBUF+62,# '8'
MOV    DISBUF+63,# ' '
LCALL  LCDSL4
RET

RAM5:  CJNE  R2,#50H,RAM6
MOV    DISBUF+60,# '5'
MOV    DISBUF+61,# '/'
MOV    DISBUF+62,# '8'
MOV    DISBUF+63,# ' '
LCALL  LCDSL4
RET

RAM6:  CJNE  R2,#60H,RAM7
MOV    DISBUF+60,# '6'
MOV    DISBUF+61,# '/'
MOV    DISBUF+62,# '8'
MOV    DISBUF+63,# ' '
LCALL  LCDSL4
RET

RAM7:  CJNE  R2,#70H,CH_RAM1
MOV    DISBUF+60,# '7'
MOV    DISBUF+61,# '/'
MOV    DISBUF+62,# '8'
MOV    DISBUF+63,# ' '
LCALL  LCDSL4
RET

CH_RAM1: CJNE  R3,#0CDH,CH_RAM2
CJNE  R2,#79H,CH_RAM2
MOV    DISBUF+60,# '1'
MOV    DISBUF+61,# 'D'
MOV    DISBUF+62,# 'A'
MOV    DISBUF+63,# 'Y'
LCALL  LCDSL4
RET

CH_RAM2: CJNE  R3,#0BBH,CH_RAM3
CJNE  R2,#7FH,CH_RAM3
MOV    DISBUF+60,# '1'
MOV    DISBUF+61,# 'h'
MOV    DISBUF+62,# 'r'
MOV    DISBUF+63,# '.'
LCALL  LCDSL4
RET

CH_RAM3: CJNE  R3,#0FDH,CON
CJNE  R2,#7FH,CON
SJMP  FURAM

CON:   RET
; ***** FURAM *****
; FULL RAM
;
FURAM:  MOV    A,#0FAH
MOVX   @DPTR,A
LCALL  NONKEEP
MOV    DISBUF+60,# 'F'
MOV    DISBUF+61,# 'U'
MOV    DISBUF+62,# 'L'
MOV    DISBUF+63,# 'L'
MOV    A,STATUS
SETB   ACC.4
SETB   ACC.7
MOV    STATUS,A
MOV    TIMLCD,#02H
LCALL  DISPON
LCALL  LCDSL4
CLR    IE.0
LJMP   RTC4

```

เอกสารนี้เป็นเอกสารที่เผยแพร่ไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

; *** SET CHARACTER ON LCD DISPLAY SUB *****

;

SCHAR_LCD: MOV R0,#DISBUF+8

MOV R7,#40

CLRB: MOV @R0,#'0'

INC R0

DJNZ R7,CLRB

MOV DISBUF,'0'

MOV DISBUF+1,'d'

MOV DISBUF+2,'m'

MOV DISBUF+3,'m'

MOV DISBUF+4,'y'

MOV DISBUF+5,'y'

MOV DISBUF+6,' ' '

MOV DISBUF+7,'T'

MOV DISBUF+10,':'

MOV DISBUF+13,':'

MOV DISBUF+16,'R'

MOV DISBUF+17,'A'

MOV DISBUF+18,'I'

MOV DISBUF+19,'N'

MOV DISBUF+20,' ' '

MOV DISBUF+26,'.'

MOV DISBUF+29,' ' '

MOV DISBUF+30,'m'

MOV DISBUF+31,'m'

MOV DISBUF+32,'R'

MOV DISBUF+33,'T'

MOV DISBUF+34,' ' '

MOV DISBUF+35,' ' '

MOV DISBUF+36,'0'

MOV DISBUF+37,'0'

MOV DISBUF+38,'0'

MOV DISBUF+39,'0'

MOV DISBUF+40,'.'

MOV DISBUF+41,'0'

MOV DISBUF+42,' ' '

MOV DISBUF+43,'m'

MOV DISBUF+44,'m'

MOV DISBUF+45,'/'

MOV DISBUF+46,'h'

MOV DISBUF+47,'r'

LCALL NONKEEP

MOV DISBUF+55,' ' '

LCALL RAMEMPTY

MOV DPL,#00H

MOV A,RHADRAM

ANL A,#0F0H

MOV DPH,A

LCALL RAM1

RET

;

; ***** NONKEEP SUB *****

;

NONKEEP: MOV DISBUF+48,'N'

MOV DISBUF+49,'O'

MOV DISBUF+50,'N'

MOV DISBUF+51,'K'

MOV DISBUF+52,'E'

MOV DISBUF+53,'E'

MOV DISBUF+54,'P'

RET

;

; ***** KEEPING SUB *****

KEEPING: MOV DISBUF+48,'K'

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับใช้ในการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

MOV   DISBUF+49, #'E'
MOV   DISBUF+50, #'E'
MOV   DISBUF+51, #'P'
MOV   DISBUF+52, #'I'
MOV   DISBUF+53, #'N'
MOV   DISBUF+54, #'G'
RET

```

```

;
; ***** RAM EMPTY SUB *****
;

```

RAMEMPTY:

```

MOV   DISBUF+56, #'R'
MOV   DISBUF+57, #'A'
MOV   DISBUF+58, #'M'
MOV   DISBUF+59, #'='
MOV   DISBUF+60, #'E'
MOV   DISBUF+61, #'M'
MOV   DISBUF+62, #'P'
MOV   DISBUF+63, #'T'
RET

```

```

;
; ***** GAGESIZE *****
; GET SIZE OF RAINGAGE
;

```

```

GAGESIZE:MOV   R1, #64
MOV   R0, #DISBUF
CLRLOOP: MOV   @R0, #' '
INC   R0
DJNZ  R1, CLRLOOP
MOV   DISBUF+16, #'S'
MOV   DISBUF+17, #'I'
MOV   DISBUF+18, #'Z'
MOV   DISBUF+19, #'E'
MOV   DISBUF+21, #'O'
MOV   DISBUF+22, #'F'
MOV   DISBUF+24, #'R'
MOV   DISBUF+25, #'A'
MOV   DISBUF+26, #'I'
MOV   DISBUF+27, #'N'
MOV   DISBUF+28, #'G'
MOV   DISBUF+29, #'A'
MOV   DISBUF+30, #'G'
MOV   DISBUF+31, #'E'
MOV   DISBUF+36, #'0'
MOV   DISBUF+37, #'.'
MOV   DISBUF+38, #'0'
MOV   DISBUF+39, #'0'
MOV   DISBUF+40, #'0'
MOV   DISBUF+42, #'m'
MOV   DISBUF+43, #'m'

```

```

LCALL LCDSL2
LCALL LCDSL4
LCALL CURSON
LCALL CURSHM
LCALL LCDSL1
LCALL LCDSL3
MOV   R1, #DISBUF+38
MOV   R7, #0AH
LCALL SETCURS

```

```

GOT:  MOV   DPTR, #P1A

```

;GET DATA FROM KEYBOARD

```

MOVX  A, @DPTR
CJNE  A, #00H, GOTJ
SJMP  GOT

```

```

GOTJ: ORL   A, #11110100B

```

```

CJNE  A, #11111111B, GOTCH
SJMP  GOT

```

;LOOP FOR GET DATA

เอกสารนี้เป็นทรัพย์สินทางปัญญาของมหาวิทยาลัยเทคโนโลยีพระจอมเกล้าธนบุรี ไม่ควรเผยแพร่โดยไม่ได้รับอนุญาต

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

GOTCH:      LCALL      KDELAY                      ;DELAY
            CJNE A,#1111110B,GOTCH1
            MOV  A,@R1                              ;INCREMENT NUMBER
            INC  A
            CJNE A,#3AH,GOTS11
            MOV  A,#30H
GOTS11:      MOV  @R1,A
            LCALL LCDSL3                          ;SHOW DATA IN LINE 3
            LCALL SETCURS
            SJMP GOT
GOTCH1:      CJNE A,#11111101B,GOTCH2
            MOV  A,@R1                              ;DECREMENT NUMBER
            DEC  A
            CJNE A,#2FH,GOTS21
            MOV  A,#39H
GOTS21:      MOV  @R1,A
            LCALL LCDSL3
            LCALL SETCURS
            SJMP GOT
GOTCH2:      CJNE R1,#DISBUF+40,GOTS4              ;CHECK POSITION
            MOV  R2,DISBUF+36
            MOV  R3,DISBUF+38
            LCALL ATOH
            MOV  SIZEGAGE,A
            MOV  R2,DISBUF+39
            MOV  R3,DISBUF+40
            LCALL ATOH
            MOV  SIZEGAGE+1,A
            RET
GOTS4:      INC  R1                                ;SHIFT RIGHT
            DEC  R7
            LCALL CURR
            LJMP GOT
;
; ***** SET CURSOR *****
; SET CURSOR FOR GET SIZE OF RAINGAGE
;
SETCURS:     MOV  A,R7
            MOV  R6,A
SETCURS1:    LCALL CURL
            DJNZ R6,SETCURS1
            RET
;
; ***** KEYBOARD SUB *****
; GET START TIME FROM KEYBOARD
;
KEYBOARD:    LCALL LCDSL2
            LCALL LCDSL3
            LCALL LCDSL4
            LCALL LCDSL1
            LCALL CURSON
            LCALL CURSHM
            MOV  DPTR,#P1P                        ;INITIAL I/P PORT FOR
KEYBOARD
            MOV  A,#92H
            MOVX @DPTR,A
            MOV  R1,#DISBUF
            MOV  R5,#0                            ;POSITION IN LINE
            SJMP SHOL
KEY:         MOV  DPTR,#P1A                        ;GET DATA FROM KEYBOARD
            MOVX A,@DPTR
            CJNE A,#00H,KEYJ
            SJMP KEY
KEYJ:        ORL  A,#11110000B
            CJNE A,#11111111B,KEYCH
            SJMP KEY                                ;LOOP FOR GET DATA

```

```

KEYCH:  LCALL KDELAY                                ;DELAY
        CJNE A,#11111110B,KEYCH1
                SJMP KEYS1                                ;INCREMENT NUMBER
KEYCH1: CJNE      A,#111111101B,KEYCH2
                SJMP KEYS2                                ;DECREMENT NUMBER
KEYCH2:CJNE A,#11111011B,KEYS4
                SJMP KEYS3                                ;SHIFT RIGHT CURSOR
                ;SHIFT LEFT CURSOR
;
; ***** KEY DELAY *****
;
KDELAY:  PUSH 04H
        PUSH 06H
        PUSH 07H
        MOV R6,#0
        MOV R7,#0
        MOV R4,#3
KDEL:    DJNZ R6,KDEL
        DJNZ R7,KDEL
        DJNZ R4,KDEL
        POP 07H
        POP 06H
        POP 04H
        RET
;
; ***** INCREMENT NUMBER *****
;
KEYS1:    MOV A,@R1
        INC A
        CJNE A,#3AH,KEYS11
        MOV A,#30H
KEYS11:   MOV @R1,A
        SJMP SHOL ;SHOW DATA IN LINE1
;
; ***** DECREMENT NUMBER *****
;
KEYS2:    MOV A,@R1
        DEC A
        CJNE A,#2FH,KEYS21
        MOV A,#39H
KEYS21:   MOV @R1,A
        SJMP SHOL
;
; ***** SHOW LINE *****
;
SHOL:    MOV SECBUF,R5
        MOV R6,SECBUF
        LCALL LCDSL1 ;SHOW LCD LINE1
        LCALL CURSHM
        CJNE R6,#0,SHC ;SHOW CURSOR AT LCD
;
; ***** SHIFT LEFT CURSOR *****
;
KEYS3:    CJNE R1,#DISBUF,KEYS31 ;CHECK POSITION OF CURSOR AND BUFFER
        LJMP KEY
KEYS31:    LCALL CURL
        DEC R1
        DEC R5
        CJNE R5,#6,KEYS32
        SJMP KEYS31
KEYS32:    CJNE R5,#7,KEYS33
        SJMP KEYS31

```

```

KEYS33: CJNE      R5,#10,KEYS34
          LJMP      KEYS31
KEYS34: CJNE      R5,#13,KEYS35
          LJMP      KEYS31
KEYS35: LJMP      KEY
;
; ***** SHIFT RIGHT CURSOR *****
;
KEYS4:      CJNE R1,#DISBUF+15,KEYS41
          LCALL     CURSOF
          RET
KEYS41:LCALL     CURR
          INC  R1
          INC  R5
          CJNE R5,#6,KEYS42
          SJMP KEYS41
KEYS42:CJNE R5,#7,KEYS43
          SJMP KEYS41
KEYS43:CJNE R5,#10,KEYS44
          SJMP KEYS41
KEYS44:CJNE R5,#13,KEYS45
          SJMP      KEYS41
KEYS45:MOV  @R1,#'0'
          SJMP SHOL
;
; ***** WRITE TIME TO DS1202 SUB *****
; WRITE START TIME TO DS1202 ( REAL TIME CHIP )
;
WR_START_TIME:
          CLR      RSTBIT
          SETB     CLKBIT
          LCALL    RTCDL          ;DELAY
          MOV      R2,#8EH        ;WRITE PROTECT = 0
          MOV      R3,#0
          LCALL    RTCWR          ;WRITE TO DS1202
          MOV      R2,DISBUF+14   ;SECOND
          MOV      R3,DISBUF+15
          LCALL    ATOH           ;ASCII TO HEX
          MOV      R2,#80H
          MOV      R3,A
          LCALL    RTCWR
          MOV      R2,DISBUF+11   ;MINUTE
          MOV      R3,DISBUF+12
          LCALL    ATOH
          MOV      R2,#82H
          MOV      R3,A
          LCALL    RTCWR
          MOV      R2,DISBUF+8    ;HOUR
          MOV      R3,DISBUF+9
          LCALL    ATOH
          MOV      R2,#84H
          MOV      R3,A
          LCALL    RTCWR
          MOV      R2,DISBUF     ;DATE
          MOV      R3,DISBUF+1
          LCALL    ATOH
          MOV      R2,#86H
          MOV      R3,A
          LCALL    RTCWR
          MOV      R2,DISBUF+2    ;MONTH
          MOV      R3,DISBUF+3
          LCALL    ATOH
          MOV      R2,#88H
          MOV      R3,A
          LCALL    RTCWR
          MOV      R2,DISBUF+4    ;YEAR

```

เอกสารนี้เป็นเอกสารต้นฉบับสำหรับใช้ในการเรียนการสอนเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

MOV    R3,DISBUF+5
LCALL  ATOH
MOV    R2,#8CH
MOV    R3,A
LCALL  RTCWR
      MOV    R2,#8EH           ;WRITE PROTECT = 1
      MOV    R3,#80H
      LCALL  RTCWR
RET

```

```

;
; ***** HTOA SUB *****
; CONVERT HEX TO ASCII
; IN  = A
; OUT = R2,R3
; REG = A,R2,R3
;

```

```

HTOA:  PUSH  ACC
      SWAP  A
      LCALL HTOAS
      MOV   R2,A
      POP   ACC
      LCALL HTOAS
      MOV   R3,A
      RET

```

```

HTOAS:  ANL   A,#0FH
      CJNE  A,#0AH,$+3
      JNC   HTOAS1
      ORL   A,#30H
      RET
HTOAS1: SUBB  A,#9
      ORL   A,#40H
      RET

```

```

;
; ***** ATOH SUB *****
; ASCII TO HEX CONVERT
; IN  = R2,R3    30H,41H
; OUT = A        0AH
; REG = A,R2
;

```

```

ATOH:  MOV    A,R2
      LCALL  ATOHS
      SWAP  A
      MOV   R2,A
      MOV   A,R3
      LCALL  ATOHS
      ORL   A,R2
      RET

```

```

ATOHS:  CJNE  A,#'A',$+3
      JC     ATOHS1
      ADD   A,#9
ATOHS1:  ANL   A,#0FH
      RET

```

```

;
; ***** RTCWR SUB *****
; WRITE SINGLE BYTE TO DS1202
; IN  = R2 COMMAND
;      R3 DATA
; REG = A,R2,R3,R4,R5
;

```

```

RTCWR:  CLR    CLKBIT           ;CLK=0
      LCALL  RTCDL
      SETB   RSTBIT           ;RST=1
      LCALL  RTCDL
      MOV    A,R2             ;COMMAND
      LCALL  RTCWR
      MOV    A,R3             ;DATA

```

```

        LCALL      RTCWRC
        CLR        RSTBIT                      ;RST=0
        LCALL      RTCDL
        RET

RTCWRC:  MOV       R4,#8                      ;WRITE COMMAND/DATA
RTCWRC1: RRC       A
        MOV        DATABI,C
        SETB       CLKBIT                      ;RISING EDGE CLOCK
        LCALL      RTCDL
        CLR        CLKBIT
        LCALL      RTCDL
        DJNZ       R4,RTCWRC1
        RET

RTCDL:   MOV       R5,#4                      ;DELAY
        DJNZ       R5,$
        RET

;
; ***** RTCRD SUB *****
; READ SINGLE BYTE FROM DS1202
; IN  = R2 COMMAND
; OUT = R3 DATA
; REG = A,R2,R3,R4,R5
;
RTCRD:   CLR        CLKBIT                      ;CLK=0
        LCALL      RTCDL
        SETB       RSTBIT                      ;RST=1
        LCALL      RTCDL
        MOV        A,R2                      ;COMMAND
        LCALL      RTCWRC
        MOV        R4,#8                      ;READ DATA
        CLR        A
RTCRD1:  CLR        CLKBIT
        LCALL      RTCDL
        MOV        C,DATABI
        RRC        A
        SETB       CLKBIT
        LCALL      RTCDL
        DJNZ       R4,RTCRD1
        MOV        R3,A
        CLR        RSTBIT                      ;RST=0
        LCALL      RTCDL
        RET

;
; ***** LCDSL1 *****
; LCD SHOW LINE1
; REG = A,R0
;
LCDSL1:  MOV        A,#ADLINE1                  ;DISPLAY TIME ADDRESS LINE1
        MOV        R0,#DISBUF
        LCALL      LCDINTS
        RET

;
; *****LCDSL2 *****
; LCD SHOW LINE2
; REG = A,R0
;
LCDSL2:  MOV        A,#ADLINE2                  ;DISPLAY TIME ADDRESS LINE2
        MOV        R0,#DISBUF+16
        LCALL      LCDINTS
        RET

;
; ***** LCDSL3*****
; LCD SHOW LINE3
; REG = A,R0
;
LCDSL3:  MOV        A,#ADLINE3                  ;DISPLAY TIME ADDRESS LINE3

```

```

MOV          R0,#DISBUF+32
LCALL        LCDINTS
RET

;
; ***** LCDSL4 *****
; LCD SHOW LINE4
; REG = A,R0
;
LCDSL4: MOV    A,#ADLINE4          ;DISPLAY TIME ADDRESS LINE4
          MOV    R0,#DISBUF+48
          LCALL  LCDINTS
          RET

;
; ***** LCDSL41 *****
; LCD SHOW LINE4
; REG = A,R0
;
LCDSL41: MOV    A,#ADLINE4          ;DISPLAY TIME ADDRESS LINE4
          INC     A
          INC     A
          MOV    R0,#DISBUF+50
          LCALL  LCDINTS_1
          RET

; ***** LCDSLS *****
; LCD SHOW LINE S
; REG = A,R0
;
LCDSLS: PUSH    ACC
          PUSH    00H
          MOV     A,#ADLINE4          ;DISPLAY TIME ADDRESS LINE S
          MOV     R0,#DISBUFS
          LCALL  LCDINTS
          POP     00H
          POP     ACC
          RET

;
; ***** LCDINI SUB *****
; LCD INITIALIZE
; REG = A,DPTR
;
LCDINI:  LCALL    FUNSET              ;FUNCTION SET
          LCALL    CLRDIS            ;CLEAR DISPLAY
          LCALL    DISPON           ;ON DISPLAY
          RET

;
; ***** CLRDIS SUB *****
; CLEAR DISPLAY INSTRUCTION
; REG = A,DPTR
;
CLRDIS:  MOV     A,#01H
          LCALL    LCDWI
          RET

;
; ***** CURSHM SUB *****
; CURSOR AT HOME INSTRUCTION
; REG = A,DPTR
;
CURSHM:  MOV     A,#02H
          LCALL    LCDWI
          RET

;
; ***** CURSON SUB *****
; DISPLAY ON/OFF INSTRUCTION
; ON CURSOR (NO FLASH)
; REG = A,DPTR

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้.

```

CURSON: MOV      A,#00001110B
          LCALL   LCDWI
          RET

```

```

;
; ***** DISPON SUB *****
; ***** CURSOF SUB *****
; DISPLAY ON/OFF INSTRUCTION
; OFF CURSOR & BLANK BUT ON DISPLAY
; REG = A,DPTR
;

```

```

DISPON:
CURSOF: MOV      A,#00001100B
          LCALL   LCDWI
          RET

```

```

;
; ***** DISPOF *****
; DISPLAY OFF INSTRUCTION
;

```

```

DISPOF: MOV      A,#00001000B
          LCALL   LCDWI
          RET

```

```

;
; ***** CURL SUB *****
; DISPLAY SHIFT INSTRUCTION
; CURSOR SHIFT LEFT
; REG = A,DPTR
;

```

```

CURL: MOV      A,#00010000B
          LCALL   LCDWI
          RET

```

```

;
; ***** CURR SUB *****
; DISPLAY SHIFT INSTRUCTION
; CURSOR SHIFT RIGHT
; REG = A,DPTR
;

```

```

CURR: MOV      A,#00010100B
          LCALL   LCDWI
          RET

```

```

;
; ***** FUNSET SUB *****
; FUNCTION SET INSTRUCTION
; 8 BIT, 1/16 DUTY, 5*7 DOT
; REG = A,DPTR
;

```

```

FUNSET: MOV      A,#00111000B ;FUNCTION SET
          LCALL   LCDWI
          RET

```

```

;
; ***** LCDINT SUB *****
; LOAD INTERNAL RAM TO LCD-MODULE (DMC164)
; IN = DISBUF
; REG = A,R0,R2
;

```

```

LCDINTS: PUSH    02H
          LCALL   LCDWI ;LOAD ONE LINE
          MOV     R2,#16 ;16 CHAR.
LCDINTS1:MOV     A,@R0 ;DISPLAY BUFFER
          LCALL   LCDWD ;WRITE DATA
          INC     R0
          DJNZ    R2,LCDINTS1
          POP     02H
          RET

```

```

;
; ***** LCDINT1 SUB *****
; LOAD INTERNAL RAM TO LCD-MODULE (DMC164)

```

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

; IN = DISBUF
; REG = A,R0,R2
LCDINTS_1:PUSH 02H
          LCALL LCDWI
          MOV R2,#3
LCDINTS_2:MOV A,@R0
          LCALL LCDWD
          INC R0
          DJNZ R2,LCDINTS_2
          POP 02H
          RET
;
; ***** LCDWI SUB *****
; LCD WRITE INSTRUCTION (RS=0)
; IN = A
; REG = A
;
LCDWI: PUSH DPH
        PUSH DPL
        MOV DPTR,#LCDWRC
        MOVX @DPTR,A
        MOV DPTR,#LCDRDC
LCDWI1: MOVX A,@DPTR ;WAIT FOR BF=0
        JB ACC.7,LCDWI1
        POP DPL
        POP DPH
        RET
;
; ***** LCDWD SUB *****
; LCD WRITE DATA (RS=1)
; IN = A
; REG = A
;
LCDWD: PUSH DPH
        PUSH DPL
        MOV DPTR,#LCDWRD
        MOVX @DPTR,A
        MOV DPTR,#LCDRDC
LCDWD1: MOVX A,@DPTR ;WAIT FOR BF=0
        JB ACC.7,LCDWD1
        POP DPL
        POP DPH
        RET
;
; ***** DTSEC SUB *****
; DELAY 1/10 SECOND
; IN = R2
; REG = R2,R3,R4
;
DTSEC: MOV R3,#179
DTSEC1: MOV R4,#0
        DJNZ R4,$
        NOP
        NOP
        DJNZ R3,DTSEC1
        DJNZ R2,DTSEC
        RET
;
; ***** INTO *****
;PROGRAM INTERRUPT FROM INTO PIN
INTER0: PUSH ACC
        PUSH B
        PUSH 00H
        PUSH 02H
        PUSH 03H
        PUSH 07H

```

เอกสารนี้เป็นเอกสารที่จัดทำขึ้นเพื่อการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

PUSH    DPH
PUSH    DPL
PUSH    PSW
LCALL   KDELAY

```

```

;
; ***** ABOUT LCD *****
;

```

```

        MOV     R0, #RAIN+4
        MOV     A, @R0
        ADD     A, SIZEGAGE+1
        LCALL   HTODL                ;HEX TO DECIMAL LOW BIT
WITH CARRY
        LCALL   HTODH                ;HEX TO DECIMAL HIGH BIT
WITH CARRY
        MOV     DISBUF+28, R2        ;WRITE TO LCD BUFFER
        MOV     R0, #RAIN+3        ;ON NEXT POSITION
        MOV     A, @R0
        ADD     A, SIZEGAGE
        LCALL   HTODL
        LCALL   HTODH
        MOV     DISBUF+25, R2
        MOV     DISBUF+27, R3
        MOV     R0, #RAIN+2        ;ON NEXT POSITION
        MOV     A, @R0
        LCALL   HTODL
        LCALL   HTODH
        MOV     DISBUF+23, R2
        MOV     DISBUF+24, R3
        MOV     R0, #RAIN+1
        MOV     A, @R0
        LCALL   HTODL
        LCALL   HTODH
        MOV     DISBUF+22, R3
        MOV     DISBUF+21, R2
        LCALL   LCDSL2              ;LCD SHOW LINE2
;
; ***** RATE *****
;

```

```

        MOV     R7, #06H
RATE1:   MOV     R0, #DISBUF+46      ;ON NEXT POSITION
        MOV     A, @R0
        ADD     A, SIZEGAGE+1
        LCALL   HTODL
        LCALL   HTODH
        DJNZ    R7, RATE1
        MOV     DISBUF+41, R2
        MOV     R7, #06H
RATE2:   MOV     R0, #DISBUF+45      ;ON NEXT POSITION
        MOV     A, @R0
        ADD     A, SIZEGAGE
        LCALL   HTODL
        LCALL   HTODH
        DJNZ    R7, RATE2
        MOV     DISBUF+38, R2
        MOV     DISBUF+39, R3
        MOV     R7, #06H
RATE3:   MOV     R0, #DISBUF+44      ;ON NEXT POSITION
        MOV     A, @R0
        LCALL   HTODL
        LCALL   HTODH
        DJNZ    R7, RATE3
        MOV     DISBUF+36, R2
        MOV     DISBUF+37, R3
;

```

เอกสารนี้เป็นเอกสารที่จัดทำขึ้นเพื่อการเรียนการสอนเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
 ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

POP      PSW
POP      DPL
POP      DPH
POP      07H
POP      03H
POP      02H
POP      00H
POP      B
POP      ACC
RETI

```

```

;
; ***** HEX TO DECIMAL LOW BIT WITH CARRY *****
;

```

```

HTODL:    SWAP A                      ;CONVERT HEX TO DECIMAL WITH CARRY
          CJNE A,#9FH,IN1
          SJMP IN11
IN1:      JC IN11
          ADD A,#61H                  ;CARRY TO NEXT POSITION
IN11:     SWAP A
          RET

```

```

;
; ***** HEX TO DECIMAL HIGH BIT WITH CARRY *****
;

```

```

HTODH:    CJNE A,#9FH,IN2
          SJMP IN21
IN2:      JC IN21
          DEC R0
          MOV B,@R0                  ;CARRY TO NEXT POSITION
          INC B
          MOV @R0,B
          INC R0
          ADD A,#60H
IN21:     MOV @R0,A
          LCALL HTOA                 ;CONVERT HEX TO ASCII
          RET

```

```

;
; ***** INT1 *****
;PROGRAM INTERRUPT FROM INT1 PIN
;SERIAL RS232 FOR LOAD DATA FROM RAM TO PC
;

```

```

INTER1:   PUSH PSW
          PUSH 07H
          PUSH 06H
          PUSH 05H
          PUSH 04H
          PUSH 03H
          PUSH 02H
          PUSH 01H
          PUSH 00H
          PUSH ACC
          PUSH DPL
          PUSH DPH
          LCALL KDELAY
          MOV A,STATUS                ;ON LCD
          SETB ACC.7
          MOV STATUS,A
          MOV TIMLCD,#02H
          LCALL DISPON

```

```

;
; *****INITIAL SERIAL*****
;

```

```

CLR      TR1                          ;TR1 IS IN TCON
MOV      TMOD,#00100000B              ;SET TIMER1 MODE 2
MOV      TH1,#BRAT96                  ;SET BAUDRATE 9600 BPS
MOV      TL1,#BRAT96
SETB     TR1                          ;TIMER1 ON

```

เอกสารนี้เป็นเอกสารงานวิจัยสำหรับใช้ในการศึกษาเท่านั้น ไม่อนุญาตให้ไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

        ANL          PCON,#7FH          ;SMOD = 0
        MOV          SCON,#01010000B   ;SERIAL 8 BIT UART MODE
                                           ;MODE

1,SINGLE PROCESSOR                      ;BIT 9 = 0

; ***** SERIAL OUT *****
;
        MOV          DPTR,#0000H        ;ADDRESS IN RAM
                                           ;START BY RAIN'S DATA

        LCALL        CRRET
        LCALL        LFEED
        CLR          TI
        MOV          SBUF,#'0'
        JNB          TI,$
        CLR          TI
        MOV          SBUF,#'.'
        JNB          TI,$
        MOV          A,SIZEGAGE
        LCALL        HTOA
        CLR          TI
        MOV          SBUF,R3
        JNB          TI,$
        MOV          A,SIZEGAGE+1
        LCALL        HTOA
        CLR          TI
        MOV          SBUF,R2
        JNB          TI,$
        CLR          TI
        MOV          SBUF,R3
        JNB          TI,$

;
SERIAL:
;
; ***** CHECK DATA *****
;
CH_DA:  MOVX         A,@DPTR
        CJNE         A,#0EEH,CH_COD
        SJMP         IEND

;
CH_COD:CJNE         A,#0F0H,CH_FC
DMYOUT:LCALL        CRRET                ;RETURN TO END LINE
        LCALL        LFEED              ;FEED NEW LINE
        MOV          R7,#05H
        MOV          R3,DPH              ;KEEP ADDRESS
        MOV          R2,DPL
CHELO:  INC          DPTR                ;LOOP FOR CHECK CODE 'FA'
        MOVX         A,@DPTR
        CJNE         A,#0FAH,COUNT
        SJMP         IEND

COUNT: DJNZ         R7,CHELO
        MOV          DPL,R2              ; RETURN ADDRESS
        MOV          DPH,R3
        INC          DPTR                ; SEND DATE,MONTH, YEAR

AND TIME
        LCALL        DOUT
        LCALL        SENDST
        INC          DPTR
        LCALL        DOUT
        LCALL        SENDST
        INC          DPTR
        LCALL        DOUT
        LCALL        SENDSP
        INC          DPTR
        LCALL        DOUT
        LCALL        SENDSEMI
        INC          DPTR

```

เอกสารนี้เป็นเอกสารสงวนลิขสิทธิ์สำหรับใช้ในการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

        LCALL      DOUT
        LCALL      CRRET
        LCALL      LFEED
        INC        DPTR
        LJMP       SERIAL
CH_FC:  CJNE      A,#0FCH,CH_FA
        SJMP       DMYOUT
CH_FA:  CJNE      A,#0FAH,CH_DA1      ;*****CH_DA1 => CH_FF
        SJMP       IEND
CH_DA1: LCALL      DOUT
        LCALL      SENDJU
        INC        DPTR
        LJMP       SERIAL
IEND:   MOV        DISBUF+6,#' '
        MOV        DISBUF+7,#'T'
        MOV        DISBUF+10,#':'
        MOV        DISBUF+13,#':'
        POP        DPH
        POP        DPL
        POP        ACC
        POP        00H
        POP        01H
        POP        02H
        POP        03H
        POP        04H
        POP        05H
        POP        06H
        POP        07H
        POP        PSW
        RETI
;
; ***** DATA OUT *****
;
DOUT:   CLR        TI
        MOVX       A,@DPTR
        LCALL      HTOA      ;CONVERT DATA HEX TO
ASCII   MOV        SBUF,R2    ;MOVE DATA TO SBUF
        JNB        TI,$
        CLR        TI
        MOV        SBUF,R3
        JNB        TI,$
        RET
;
; ***** SEND SEMICOLON *****
;
SENDSEMI:CLR      TI          ;SEND ':'
        MOV        SBUF,#':'
        JNB        TI,$
        RET
;
; ***** SEND CODE , *****
;
SENDJU:  CLR        TI          ;SEND ','
        MOV        SBUF,#','
        JNB        TI,$
        RET
;
; ***** SEND CODE / *****
;
SENDST:  CLR        TI          ;SEND '/'
        MOV        SBUF,# '/'
        JNB        TI,$
        RET

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

;
CRRET:  CLR      TI
        MOV      SBUF,#0DH
        JNB      TI,$
        RET

;
; ***** LINE FEED *****
;
LFEED:  CLR      TI
        MOV      SBUF,#0AH
        JNB      TI,$
        RET

;
; ***** SEND CODE ' ' *****
;
SENDSP: CLR      TI
        MOV      SBUF,#' '
        JNB      TI,$
        RET

;
;*****
;SUB PROGRAM FOR CONVERSE HEX TO BCD
;INPUT = REG A
;OUTPUT = R3(high),R2(low)
;
HEXTOBCD:MOV     R2,#00
        MOV      R3,#00
MORE_100:CJNE    A,#100,$+3
        JC      LESS_100
        SUBB     A,#100
        INC      R3
        JMP      MORE_100
LESS_100: CJNE    A,#10,$+3
        JC      LESS_10
        SUBB     A,#10
        INC      R2
        JMP      LESS_100
LESS_10:  PUSH    ACC
        MOV      A,R2
        SWAP     A
        MOV      R2,A
        POP      ACC
        ADD      A,R2
        MOV      R2,A
        RET

;*****
;***** SEND SIZE DATE MONTH YEAR HOUR MINUTE AND DATA *****
;SEND WHEN KEEP DATA RAIN
;
SENDDATA:PUSH    DPH
        PUSH     DPL
        PUSH     02H
        PUSH     03H

;
;*****INITIAL SERIAL*****
;
        CLR      TR1
        MOV      TMOD,#00100000B
        MOV      TH1,#BRAT96
        MOV      TL1,#BRAT96
        SETB     TR1
        ANL      PCON,#7FH
        MOV      SCON,#01010000B

;TR1 IS IN TCON
;SET TIMER1 MODE 2
;SET BAUDRATE 9600 BPS
;TIMER1 ON
;SMOD = 0
;SERIAL 8 BIT UART MODE
;MODE 1,SINGLE PROCESSOR
;BIT 9=0

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

MOV          SBUF,#'0'                ;SEND SIZEGAGE
JNB          TI,$
CLR          TI
MOV          SBUF,#'.'
JNB          TI,$
MOV          A,SIZEGAGE
LCALL       HTOA
CLR          TI
MOV          SBUF,R3
JNB          TI,$
MOV          A,SIZEGAGE+1
LCALL       HTOA
CLR          TI
MOV          SBUF,R2
JNB          TI,$
CLR          TI
MOV          SBUF,R3
JNB          TI,$
CLR          TI

;
; ***** SEND DATE & TIME *****
;
SE_DD:  MOV    R2,#87H                ;DATE
        LCALL  RTCRD
        MOV    A,R3
        LCALL  DOUT1
        LCALL  SENDSLASH
SE_MM:  MOV    R2,#89H                ;MONTH
        LCALL  RTCRD
        MOV    A,R3
        LCALL  DOUT1
        LCALL  SENDSLASH
SE_YY:  MOV    R2,#8DH                ;YEAR
        LCALL  RTCRD
        MOV    A,R3
        LCALL  DOUT1
SE_HH:  MOV    R2,#85H                ;HOURL
        LCALL  RTCRD
        MOV    A,R3
        LCALL  DOUT1
        LCALL  SENDSEMICO
SE_MIN: MOV    R2,#83H                ;MINUTE
        LCALL  RTCRD
        MOV    A,R3
        LCALL  DOUT1
DATAPULSE:MOV  DPL,ADDRL
        MOV    DPH,ADDRH
SENDPULSE:LCALL DOUT2                ;SEND DATA PULSE OF RAIN
        POP    03H                    ;DECIMAL VALUE
        POP    02H
        POP    DPL
        POP    DPH
        RET

;
; ***** DATA OUT *****
;
DOUT1:  LCALL  HTOA
        MOV    SBUF,R2
        JNB    TI,$
        CLR    TI
        MOV    SBUF,R3
        JNB    TI,$
        CLR    TI
        RET

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
 ; ***** DATA OUT *****
 ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

;
DOUT2:  MOVX    A,@DPTR
        LCALL   HTOA
        MOV     SBUF,R2
        JNB     TI,$
        CLR     TI
        MOV     SBUF,R3
        JNB     TI,$
        CLR     TI
        RET

;
; ***** SEND CODE / EVERY MINUTE *****
;
SENDSLASH:
        MOV     SBUF,'#/'
        JNB     TI,$
        CLR     TI
        RET

;
; ***** SEND SEMICOLON *****
;
SENDSEMICO:
        MOV     SBUF,'#:'
        JNB     TI,$
        CLR     TI
        RET

;
; ***** DELAY 1 MSEC *****
;
; DELAY_1MS: MOV  R6,#03H
; DEL2:      MOV  R7,#97H
; DEL1:      DJNZ R7,DEL1
;            DJNZ R6,DEL2
;
; *****
;
;      CREATEDEMO BY USE TABLE
; *****
;
;      DELAY 50 US FOR BUSY LCD
BUSYDELAY: MOV  R7,#23D
BUSY:      DJNZ R7,BUSY
        RET

CREATEDEMO: MOV  DPTR,#LCDWRC      ;LOAD ADDRESS INSTRUCTION
            MOV  A,#ADLINE1      ;ADRESS LINE 1
            MOVX @DPTR,A
            ACALL BUSYDELAY
            MOV  DPTR,#DEMO_TBL1 ;LOAD DEMO TABLE LINE 1
            ACALL WRCHAR1        ;WRITE CHAR TO LCD
            MOV  DPTR,#LCDWRC
            MOV  A,#ADLINE2      ;ADRESS LINE 2
            MOVX @DPTR,A
            ACALL BUSYDELAY
            MOV  DPTR,#DEMO_TBL2 ;LOAD DEMO TABLE LINE 2
            ACALL WRCHAR1
            MOV  DPTR,#LCDWRC
            MOV  A,#ADLINE3      ;ADRESS LINE 3
            MOVX @DPTR,A
            ACALL BUSYDELAY
            MOV  DPTR,#DEMO_TBL3 ;LOAD DEMO TABLE LINE 3
            ACALL WRCHAR1
            MOV  DPTR,#LCDWRC
            MOV  A,#ADLINE4      ;ADRESS LINE 4
            MOVX @DPTR,A
            ACALL BUSYDELAY
            MOV  DPTR,#DEMO_TBL4 ;LOAD DEMO TABLE LINE 4
            ACALL WRCHAR1
            RET

```

เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์ไว้เพื่อการศึกษาเท่านั้น ไม่อนุญาตให้拿去ใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

;
CREATECHAR:  MOV     DPTR,#LCDWRC
             MOV     A,#01001000B      ;SET CGRAM ADDRESS
             MOVX    @DPTR,A
             ACALL   BUSYDELAY
             MOV     DPTR,#CHAR_TBL1   ;LOAD CHAR TABLE
             ACALL   WRCHAR             ;WRITE CHAR TO LCD
             MOV     DPTR,#CHAR_TBL2
             ACALL   WRCHAR
             MOV     DPTR,#CHAR_TBL3
             ACALL   WRCHAR
             RET

;
WRCHAR:MOV    R5,#00H
LOAD:  MOV    A,R5
             MOVX    A,@A+DPTR
             PUSH    DPL
             PUSH    DPH
             MOV     DPTR,#LCDWRD      ;LOAD LCD WRITE DATA
             MOVX    @DPTR,A
             ACALL   BUSYDELAY
             POP     DPH
             POP     DPL
             INC     R5
             CJNE    R5,#08H,LOAD
             RET

;
WRCHAR1: MOV   R4,#00H
LOAD1:  MOV    A,R4
             MOVX    A,@A+DPTR
             PUSH    DPL
             PUSH    DPH
             MOV     DPTR,#LCDWRD      ;LOAD LCD WRITE DATA
             MOVX    @DPTR,A
             ACALL   BUSYDELAY
             POP     DPH
             POP     DPL
             INC     R4
             CJNE    R4,#0FH,LOAD1
             RET

;
;***** TABLE *****
;
DEMO_TBL1:  DB      "    RAIN    "
DEMO_TBL2:  DB      "  TELEMETERING  "
DEMO_TBL3:  DB      "    SYSTEM    "
DEMO_TBL4:  DB      "    (2540)    "

;
CHAR_TBL1:  DB      00001101B
             DB      00010010B
             DB      00000101B
             DB      00001001B
             DB      00010101B
             DB      00011011B
             DB      00011001B
             DB      00000000B

;
CHAR_TBL2:  DB      00001110B
             DB      00010001B
             DB      00000001B
             DB      00001101B
             DB      00001101B
             DB      00000101B
             DB      00000011B
             DB      00000000B

```

เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์ไว้สำหรับใช้ในการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

CHAR_TBL3:

DB	00001110B
DB	00010001B
DB	00000001B
DB	00001001B
DB	00010101B
DB	00011011B
DB	00011001B
DB	00000000B
END	



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

VERSION 4.00

Begin VB.Form frmMainMenu

BorderStyle = 1 'Fixed Single
Caption = "Menu Raining Meter version 1.0"
ClientHeight = 6735
ClientLeft = 150
ClientTop = 1635
ClientWidth = 9330
Height = 7425
Icon = "menu.frx":0000
Left = 90
LinkTopic = "Form1"
MaxButton = 0 'False
MinButton = 0 'False
Picture = "menu.frx":0442
ScaleHeight = 6735
ScaleWidth = 9330
Top = 1005
Width = 9450

Begin VB.TextBox txtDataBaseName

Height = 375
Left = 240
TabIndex = 0
Top = 840
Visible = 0 'False
Width = 975

End

Begin VB.Timer Timer1

Enabled = 0 'False
Interval = 200
Left = 840
Top = 120

End

Begin MSCommLib.MSComm Commu1

Left = 240
Top = 120
_Version = 65536
ExtentX = 847

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
_ExtentY = 847
_StockProps = 0
CDTimeout = 0
CommPort = 2
CTSTimeout = 0
DSRTimeout = 0
DTREnable = -1 'True
Handshaking = 2
InBufferSize = 20
InputLen = 0
Interval = 1000
NullDiscard = 0 'False
OutBufferSize = 20
ParityReplace = "?"
RThreshold = 500
RTSEnable = 0 'False
Settings = "9600,n,8,1"
SThreshold = 500
```

End

Begin MSComDlg.CommonDialog CMDialog1

```
Left = 1440
Top = 120
_Version = 65536
_ExtentX = 847
_ExtentY = 847
_StockProps = 0
CancelError = -1 'True
```

End

Begin VB.Line Line1

```
BorderColor = &H00808080&
BorderStyle = 6 'Inside Solid
BorderWidth = 2
X1 = 0
X2 = 6600
Y1 = 0
Y2 = 0
```

End

Begin VB.Menu mnufile

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

Caption      = "&File"

Begin VB.Menu fnew
    Caption    = "&New"
    Shortcut   = ^N
End

Begin VB.Menu fopen
    Caption    = "Open"
    Begin VB.Menu mnuOpenEdit
        Caption = "Open for Edit"
    End
    Begin VB.Menu mnuOpenView
        Caption = "Open for View"
    End
End

Begin VB.Menu fsave
    Caption    = "Save"
    Enabled    = 0 'False
    Shortcut   = ^S
End

Begin VB.Menu fclose
    Caption    = "Close"
    Enabled    = 0 'False
End

Begin VB.Menu a4
    Caption    = "-"
End

Begin VB.Menu fexit
    Caption    = "&Exit"
End

Begin VB.Menu mnu2
    Caption    = "-"
End

Begin VB.Menu mnucomport
    Caption    = "&Windows"

    Begin VB.Menu Mopen
        Caption = "&Keeping Data"
        Enabled = 0 'False
    End
End

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

End
Begin VB.Menu qc
    Caption    = "-"
End
Begin VB.Menu mnuShowData
    Caption    = "Show Data Base"
End
Begin VB.Menu qb
    Caption    = "-"
End
Begin VB.Menu winshowgraph
    Caption    = "Show Graph"
    Enabled    = 0 'False
End
Begin VB.Menu qa
    Caption    = "-"
End
End
Attribute VB_Name = "frmMainMenu"
Attribute VB_Creatable = False
Attribute VB_Exposed = False

Private Sub fclose_Click()
    txtDataBaseName.Text = ""
    frmMainMenu.Caption = "Menu Raining Meter version 1.0"
    frmMainMenu.mnuOpenEdit = True
    frmMainMenu.winshowgraph = False
    frmMainMenu.fclose = False
    MsgBox "Datafile are closed", , "Error #14027"
    frmMainMenu.fnew.Enabled = True
    Status = stNothing
End Sub

Private Sub fexit_Click()
    ' ออกจากโปรแกรมใหญ่ โดย ตรวจสอบ com port ว่าเปิดอยู่หรือเปล่า
    If Status = stNothing Then

```

```

        If Commu1.PortOpen Then 'คิดว่า com port ยังเปิดค้างอยู่

```

เอกสารนี้เป็นเอกสารหลวงไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

Commu1.PortOpen = Not Commu1.PortOpen ' ปิด com port นั้นๆ
End If ' com port ปิดเรียบร้อยแล้ว
ElseIf Status = stOpenEdit Then
    MsgBox " You must Save Befor ", , "Error #14018"
    Exit Sub
ElseIf Status = stNewFile Then 'Or stOpenFile Then
    MsgBox " You must Save file befor Exit main program ", , " Error#14009"
    Exit Sub
ElseIf Status = stOpenView Then
    Unload Me
ElseIf Status = stSaveFile Then
    If Commu1.PortOpen Then 'คิดว่า com port ยังเปิดค้างอยู่
        Commu1.PortOpen = Not Commu1.PortOpen ' ปิด com port นั้นๆ
    End If ' com port ปิดเรียบร้อยแล้ว
    ElseIf NotSave = nsNot Then
        MsgBox " You must Save New file befor Exit main program ", , " Error#14019"
        Exit Sub
    End If
    Unload Me ' ปิด Main form
    End ' จบโปรแกรมหลัก
End Sub

Private Sub fnew_Click()
    frmMainMenu.MOpen.Enabled = True
    frmMainMenu.fsave.Enabled = True
    SourceFile = "DataRain1.mdb" ' Define source filename.
    DestinationFile = "Data1.mdb" ' Define target filename.
    FileCopy SourceFile, DestinationFile ' Copy source to target.
    'set Name Data base File for Open port
    txtDataBaseName.Text = DestinationFile
    frmMainMenu.Caption = DestinationFile + " - Menu Raining Meter version 1.0"
    ' set status is new
    Status = stNewFile
    NotSave = nsNot
    FileOpenDuble = fodActive
End Sub

```

```

Line1.X2 = frmMainMenu.Width
frmMainMenu.Width = 9450
frmMainMenu.Height = 7500
If WindowsState = 0 Then
    Move (Screen.Width - frmMainMenu.Width) / 2, (Screen.Height - frmMainMenu.Height) / 2
End If
' set com port ของเครื่อง computer
frmMainMenu.Commu1.CommPort = 2 ' set เป็น com 2
frmMainMenu.Commu1.Settings = "9600,N,8,1" ' set การติดต่อรับส่งข้อมูล
frmMainMenu.Commu1.Handshaking = 2 ' set ว่ามีการเช็ค handshaking เป็นแบบ RTS/CTS
End Sub

```

```

Private Sub Form_Unload(Cancel As Integer)
' ออกจากโปรแกรมใหญ่ โดย ตรวจ com port ว่าเปิดอยู่หรือเปล่า
If Status = stNothing Then
    If Commu1.PortOpen Then ' เช็ค ว่า com port ยังเปิดค้างอยู่
        Commu1.PortOpen = Not Commu1.PortOpen ' ปิด com port นั้นๆ
    End If ' com port ปิดเรียบร้อยแล้ว
ElseIf Status = stSaveFile Then
    If Commu1.PortOpen Then ' เช็ค ว่า com port ยังเปิดค้างอยู่
        Commu1.PortOpen = Not Commu1.PortOpen ' ปิด com port นั้นๆ
    End If ' com port ปิดเรียบร้อยแล้ว
ElseIf Status = stNewFile Then
    MsgBox " You must Save file befor Exit main program ", , " Error#14009"
    fsave_Click
End If
    Unload Me ' ปิด Main form
    End ' จบโปรแกรมหลัก
End Sub

```

```

Private Sub fsave_Click()
' check open file befor
If Status = stNothing Then
    MsgBox "You must New or Open from Menu file", , "Error #14007"
'กรณีเปิด New แล้วไม่ Show Database
ElseIf Status = stNewFile And NotSave = nsNot Then
    ' Set an error trap to detect the pressing of the
    Cancel key of the Save dialog box

```

เอกสารนี้เป็นทรัพย์สินทางปัญญาของสำนักงานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

On Error GoTo OpenError

' Fill the item of the File Type list box of the Save dialog

CMDialog1.Filter = "Database (*.mdb)|*.mdb| All Files (*.*)|*.*"

' Set the DEfault File Type to TExt File (*.mdb)

CMDialog1.FilterIndex = 1

' Set Filename is " Data1. mdb "

CMDialog1.filename = "Data1.mdb"

'DisplaY Save Dialog box

CMDialog1.Action = 2

SourceFile = txtDataBaseName.Text ' Define source filename.

DestinationFile = CMDialog1.filename ' Define target filename.

FileCopy SourceFile, DestinationFile ' Copy source to target.

Kill SourceFile

frmMainMenu.Caption = DestinationFile + "- Menu Raining Meter version 1.0"

txtDataBaseName.Text = DestinationFile

Status = stSaveFile

NotSave = nsNot + 1

Exit Sub

OpenError:

'The Pressed The Cancel key

MsgBox " You Cancel Save DialogBox"

Exit Sub

'กรณี New แล้วเปิด Show Database

ElseIf Status = stSaveView And NotSave = nsNot Then

On Error GoTo OpenError1

CMDialog1.Filter = "Database (*.mdb)|*.mdb| All Files (*.*)|*.*"

CMDialog1.FilterIndex = 1

CMDialog1.filename = "Data1.mdb"

CMDialog1.Action = 2

SourceFile = txtDataBaseName.Text ' Define source filename.

DestinationFile = CMDialog1.filename ' Define target filename.

FileCopy SourceFile, DestinationFile ' Copy source to target.

Kill SourceFile

frmMainMenu.Caption = DestinationFile + "- Menu Raining Meter version 1.0"

txtDataBaseName.Text = DestinationFile

Status = stSaveFile

NotSave = nsNot + 1

Exit Sub

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

OpenError1:

MsgBox " You Cancel Save DialogBox"

Exit Sub

ElseIf Status = stSaveFile Then

Status = stSaveFile

ElseIf Status = stOpenEdit Then

Status = stSaveFile

ElseIf NotSave = nsNot Then

' Set an error trap to detect the pressing of the

' Cancel key of the Save dialog box

On Error GoTo OpenError

' Fill the item of the File Type list box of the Save dialog

CMDialog1.Filter = "Database (*.mdb)|*.mdb| All Files (*.*)|*.*"

' Set the DEfault File Type to TExt File (*.mdb)

CMDialog1.FilterIndex = 1

' Set Filename is " Data1. mdb "

CMDialog1.filename = "Data1.mdb"

'Display Save Dialog box

CMDialog1.Action = 2

SourceFile = "Data1.mdb" ' Define source filename.

DestinationFile = CMDialog1.filename ' Define target filename.

FileCopy SourceFile, DestinationFile ' Copy source to target.

Kill SourceFile

frmMainMenu.Caption = DestinationFile + "- Menu Raining Meter version 1.0"

txtDataBaseName.Text = DestinationFile

Status = stSaveFile

Exit Sub

End If

End Sub

Private Sub mnuOpenEdit_Click()

frmMainMenu.fnew.Enabled = False

frmMainMenu.fsav.Enabled = True

frmMainMenu.mnuOpenView.Enabled = False

frmMainMenu.MOpen.Enabled = True

frmMainMenu.winshowgraph.Enabled = False

' Set an error trap to detect the pressing of the

' Cancel key of the Open dialog box

เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

On Error GoTo OpenError

' Fill the item of the File Type list box of the Open dialog

CMDialog1.Filter = "Database (*.mdb)|*.mdb| All Files (*.*)|*.*"

' Set the DEfault File Type to TExt File (*.mdb)

CMDialog1.FilterIndex = 1

'Display OPEN Dialog box

CMDialog1.Action = 1

' display the Name of the file that your selected

txtDataBaseName.Text = CMDialog1.filename

frmMainMenu.Caption = CMDialog1.filename + " - Menu Raining Meter version 1.0"

Status = stOpenEdit

Exit Sub

OpenError:

'The Pressed The Cancel key

MsgBox " You Cancel open DialogBox"

Exit Sub

End Sub

Private Sub mnuOpenView_Click()

frmMainMenu.fnew.Enabled = False

frmMainMenu.mnuOpenEdit = False

frmMainMenu.fclose.Enabled = True

frmMainMenu.fsave.Enabled = False

frmMainMenu.winshowgraph.Enabled = True

frmMainMenu.MOpen.Enabled = False

' Set an error trap to detect the pressing of the

' Cancel key of the OPEN dialog box

On Error GoTo OpenError

' Fill the item of the File Type list box of the Open dialog

CMDialog1.Filter = "Database (*.mdb)|*.mdb| All Files (*.*)|*.*"

' Set the DEfault File Type to TExt File (*.mdb)

CMDialog1.FilterIndex = 1

'Display OPEN Dialog box

CMDialog1.Action = 1

' display the Name of the file that your selected

txtDataBaseName.Text = CMDialog1.filename

frmMainMenu.Caption = CMDialog1.filename + " - Menu Raining Meter version 1.0"

Status = stOpenView

เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

Exit Sub

OpenError:
'The Pressed The Cancel key
MsgBox " You Cancel open DialogBox"
Exit Sub

End Sub

```

```

Private Sub mnuShowData_Click()
If Status = stNewFile Then
' แสดง form Data ที่เป็นData จาก File Data Base
frmShowDataBase.Visible = True ' show form ของฐานข้อมูล
ElseIf Status = stOpenView Then
' แสดง form Data ที่เป็นData จาก File Data Base
frmShowDataBase.Visible = True ' show form ของฐานข้อมูล
NotSave = nsNot + 1
ElseIf Status = stSaveFile Then
' แสดง form Data ที่เป็นData จาก File Data Base
frmShowDataBase.Visible = True ' show form ของฐานข้อมูล
ElseIf Status = stNonSave Then
MsgBox " File is not Save ", , " Error #14011"
Exit Sub
ElseIf Status = stNothing Then
MsgBox " File is not Open ", , " Error #14012"
Exit Sub
ElseIf Status = stOpenEdit Then
frmShowDataBase.Visible = True ' show form ของฐานข้อมูล
Exit Sub
End If
End Sub

```

```

Private Sub MOpen_Click()
' แสดง form ของ การรับข้อมูลจาก com port
If txtDataBaseName.Text = "" Then
MsgBox "You must select New or open from File Menu", , " Error #14002"
Else
If Status = stSaveFile Then
MsgBox " You must open Edit befor ", , " Error #14015"
Exit Sub

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

Else
    FrmShowDataInput.Visible = True ' นำเอา form ของการรับข้อมูลจาก com port ขึ้นมาแสดงบนหน้าจอ
End If
End If
End Sub

Private Sub Timer1_Timer()
T: Timer1.Interval = Timer1.Interval - 1
    If Timer1.Interval = 0 Then
        frmDemo.Show
        Beep
        Else
            GoTo T
        End If
    End Sub

Private Sub winshowgraph_Click()
'If frmShowGraph.DataTable2.Recordset.EOF Or frmShowGraph.DataTable2.Recordset.BOF Then ' เช็คว่าใน file
Data Base นั้น ไม่มีข้อมูลอยู่
' MsgBox "No Record in Data File ", , " Error #14005" ' แสดง dialogBox ERROR บอกว่าไม่มีข้อมูลหรือ record
ใดอยู่เลย
' Exit Sub
'Else ' เช็คว่ามีข้อมูลหรือ record อยู่ใน File Data Base
    frmShowGraph.Visible = True ' show graph form ของฐานข้อมูล
    NotSave = nsNot + 1
    frmMainMenu.Enabled = False
'End If
End Sub

```

```

Else
    FrmShowDataInput.Visible = True    ' นำเอา form ของการรับข้อมูลจาก com port ขึ้นมาแสดงบนหน้าจอ
End If
End If
End Sub

Private Sub Timer1_Timer()
    T: Timer1.Interval = Timer1.Interval - 1
    If Timer1.Interval = 0 Then
        frmDemo.Show
        Beep
        Else
            GoTo T
        End If
    End Sub

Private Sub winshowgraph_Click()
    'If frmShowGraph.DataTable2.Recordset.EOF Or frmShowGraph.DataTable2.Recordset.BOF Then    ' เช็คว่าใน file
    Data Base นั้น 'ไม่มีข้อมูลอยู่
    '    MsgBox "No Record in Data File ", , " Error #14005"    ' แสดง dialogBox ERROR บอกว่าไม่มีข้อมูลหรือ record
    '    ใดๆอยู่เลย
    '    Exit Sub
    'Else    ' เช็คว่ามีข้อมูลหรือ record อยู่ใน File Data Base
        frmShowGraph.Visible = True    ' show graph form ของฐานข้อมูล
        NotSave = nsNot + 1
        frmMainMenu.Enabled = False
    'End If
End Sub

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

VERSION 4.00

Begin VB.Form FrmShowDataInput

BorderStyle = 3 'Fixed Dialog
Caption = "Show data input from comport"
ClientHeight = 2025
ClientLeft = 1905
ClientTop = 1785
ClientWidth = 9135
Height = 2430
Left = 1845
LinkTopic = "Form2"
LockControls = -1 'True
MaxButton = 0 'False
MinButton = 0 'False
ScaleHeight = 2025
ScaleWidth = 9135
ShowInTaskbar = 0 'False
Top = 1440
Visible = 0 'False
Width = 9255

Begin VB.Data DataTable1

Caption = "Data1"
Connect = "Access"
DatabaseName = ""
Exclusive = 0 'False
Height = 315
Left = 4680
Options = 0
ReadOnly = 0 'False
RecordsetType = 1 'Dynaset
RecordSource = "DataRain"
Top = 1440
Visible = 0 'False
Width = 2295

End

Begin VB.CommandButton CmStart

Caption = "Start"
Height = 495

เอกสารนี้เป็นเอกสารสงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Left = 7680
TabIndex = 13
Top = 240
Width = 1215

End

Begin VB.CommandButton cdExit

Caption = "Exit"
Height = 495
Left = 7680
TabIndex = 12
Top = 1320
Width = 1215

End

Begin VB.TextBox Bit2Text

DataField = "Date"
DataSource = "DataTable1"
Height = 405
Left = 2160
TabIndex = 1
Top = 480
Width = 1575

End

Begin VB.Timer TimerShowData

Enabled = 0 'False
Interval = 1000
Left = 7080
Top = 0

End

Begin VB.TextBox ErrorBitText

Height = 400
Left = 3480
TabIndex = 5
Top = 1440
Visible = 0 'False
Width = 900

End

Begin VB.TextBox TotalBitText

Height = 405

เอกสารนี้เป็นเอกสารสงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
Left      = 120
TabIndex  = 4
Top       = 1440
Visible   = 0 'False
Width     = 3015
```

End

Begin VB.TextBox Bit4Text

```
DataField  = "Quantity"
DataSource  = "DataTable1"
Height     = 405
Left       = 5640
TabIndex   = 3
Top        = 480
Width      = 1335
```

End

Begin VB.TextBox Bit3Text

```
DataField  = "Time"
DataSource  = "DataTable1"
Height     = 405
Left       = 3960
TabIndex   = 2
Top        = 480
Width      = 1575
```

End

Begin VB.TextBox Bit1Text

```
DataField  = "Size"
DataSource  = "DataTable1"
Height     = 405
Left       = 120
TabIndex   = 0
Top        = 480
Width      = 1815
```

End

Begin MSComDlg.CommonDialog CMDialog2

```
Left      = 6960
Top       = 1080
_Version  = 65536
_ExtentX  = 847
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

_ExtentY = 847

_StockProps = 0

End

Begin VB.Label Label7

Caption = "Size:"

Height = 300

Left = 120

TabIndex = 11

Top = 240

Width = 795

End

Begin VB.Label Label6

Caption = "Date :"

Height = 300

Left = 2160

TabIndex = 10

Top = 240

Width = 795

End

Begin VB.Label Label5

Caption = "Time :"

Height = 300

Left = 3960

TabIndex = 9

Top = 240

Width = 795

End

Begin VB.Label Label4

Caption = "Quantity :"

Height = 300

Left = 5640

TabIndex = 8

Top = 240

Width = 795

End

Begin VB.Label Label2

Caption = "Total Bit of Data :"

Height = 300

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
Left      = 120
TabIndex  = 7
Top       = 1200
Visible   = 0 'False
Width     = 1215
```

End

Begin VB.Label Label1

```
Caption    = "Event and Error :"  
Height     = 300  
Left       = 3480  
TabIndex   = 6  
Top        = 1080  
Visible    = 0 'False  
Width      = 1215
```

End

End

Attribute VB_Name = "FrmShowDataInput"

Attribute VB_Creatable = False

Attribute VB_Exposed = False

' กำหนดค่าตัวแปรที่จะใช้ในการเช็คว่ามีกำหนด Current record ในฐานข้อมูลไหมแล้ว

Dim FristAddNew As Integer

Private Sub cdExit_Click()

' ออกจากโปรแกรมย่อย แต่ไม่ออกจากโปรแกรมหลัก

If CmStart.Caption = "Stop" Then

Exit Sub

Else

If Status = stNewFile Then

MsgBox " You must Save New file befor Exit program ", , " Error #14008"

ElseIf Status = stOpenEdit Then

MsgBox " You must Save Open file befor Exit program ", , " Error #140016"

Status = stOpenEdit

NotSave = nsSave

Else

MsgBox " You must Save Open file befor Exit program ", , " Error #140020"

Status = stOpenEdit

NotSave = nsSave

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

End If

Unload FrmShowDataInput ' ปิด form ย่อยนี้

End If

End Sub

Private Sub CmStart_Click()

' การเริ่มต้นการรับข้อมูลจาก comport และเก็บลงใน record ของ File Data Base

' โดยเมื่อกดปุ่มครั้งแรก จะ Start ถ้ากดอีกทีจะ Stop

If frmMainMenu.Commu1.PortOpen = False Then

If CmStart.Caption = "Start" Then ' เช็คว่าโปรแกรมไม่ได้ทำงานอยู่

TimerShowData.Enabled = True ' สั่งให้โปรแกรมทำงาน รับข้อมูลจาก com port และ save ข้อมูลลงrecord

CmStart.Caption = "Stop" ' แสดงข้อความว่าถ้ากดปุ่มนี้อีกทีโปรแกรมจะหยุดทำงาน

Else ' เช็คว่าโปรแกรมกำลังทำงานอยู่

TimerShowData.Enabled = False ' สั่งให้โปรแกรมหยุดทำงาน

CmStart.Caption = "Start" ' แสดงข้อความว่าถ้ากดปุ่มนี้อีกทีโปรแกรมจะทำงานต่อ

End If

Else

Exit Sub

End If

End Sub

Private Sub Form_Load()

If Status = stOpenEdit Then

DataTable1.DatabaseName = frmMainMenu.txtDataBaseName.Text

FristAddNew = 0 ' set ค่า ตัวแปลบอกว่ายังไม่มีการ กำหนด Current record

FrmShowDataInput.Caption = DataTable1.DatabaseName + " - Show data input from comport"

ElseIf Status = stNewFile Then

DataTable1.DatabaseName = frmMainMenu.txtDataBaseName.Text

FristAddNew = 0 ' set ค่า ตัวแปลบอกว่ายังไม่มีการ กำหนด Current record

FrmShowDataInput.Caption = DataTable1.DatabaseName + " - Show data input from comport"

End If

End Sub

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Private Sub Form_MouseMove(Button As Integer, Shift As Integer, X As Single, Y As Single)

' ทำการกำหนด current record ก่อนที่จะกดปุ่ม start

If FristAddNew = 0 Then ' เช็คว่ายังไม่มีกำหนด current record

DataTable1.Recordset.AddNew ' กำหนด current record ก่อนที่จะมีการจัดการกับฐานข้อมูลใหม่

FristAddNew = 1 ' กำหนดค่าตัวแปลบอกว่ามีการ กำหนด current record แล้ว

Else ' เช็คว่ามีการกำหนด current record แล้ว

FristAddNew = 1 ' กำหนดค่าตัวแปลบอกว่ามีการ กำหนด current record แล้ว

End If

End Sub

Private Sub Form_Unload(Cancel As Integer)

' การออกจากโปรแกรมย่อย และ ตรวจสอบว่าปิด com port หรือยัง

If frmMainMenu.Commu1.PortOpen = True Then ' เช็คว่า com port ยังเปิดอยู่

frmMainMenu.Commu1.PortOpen = False ' ปิด com port นั้นซะ

AddDataAuto

TimerShowData.Enabled = False ' ออกจาก โปรแกรมรับข้อมูลและ save ข้อมูลลงใน record

Else ' เช็คว่า com port ปิดอยู่แล้ว

TimerShowData.Enabled = False ' ออกจาก โปรแกรมรับข้อมูลและ save ข้อมูลลงใน record

End If

End Sub

Private Sub TimerShowData_Timer()

Dim DataInput As String

Dim DataError As Integer

frmMainMenu.Commu1.InputLen = 0 ' clear ข้อมูล input

frmMainMenu.Commu1.PortOpen = True ' เปิด port

Do ' รับข้อมูลจาก com port

Dummy = DoEvents()

DataError = frmMainMenu.Commu1.CommEvent ' เช็คว่าและแสดง Event and Error ขณะรับ-ส่งข้อมูล

ErrorBitText.Text = DataError

Loop Until frmMainMenu.Commu1.InBufferCount >= 20 ' loop รับข้อมูลที่จะส่งมา 1 ชุด มี 20 ตัว

'----- การแสดงข้อมูลแต่ละบิต ที่รับเข้ามาทาง port Com1-----

a\$ = frmMainMenu.Commu1.Input ' นำข้อมูลทั้ง 20 บิตที่รับทาง port มาเก็บไว้ใน ตัวแปล a\$

' การแสดงข้อมูลในชุดแรกที่รับมาจาก port

DataInput = Left(a\$, 5) ' เอาข้อมูล 5 ตัวจากทางซ้าย ไปเก็บไว้ในตัวแปล DataInput

Bit1Text.Text = DataInput ' เอาข้อมูลในตัวแปลไปแสดงที่หน้าจอ จะเป็นค่า Size

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

'การแสดงผลข้อมูลในชุดที่ 2 ที่รับมาจาก port

DataInput = Left(a\$, 13) ' เอาข้อมูล 13 ตัวจากทางซ้าย ไปเก็บไว้ในตัวแปร DataInput

DataInput = Right(DataInput, 8) 'เอาข้อมูล 8 ตัวจากทางขวาใน ตัวแปร DataInput ไปเก็บในตัวแปร DataInput จะ

เป็นค่า Date

Bit2Text.Text = DataInput ' เอาข้อมูลในตัวแปร Datainput ไปแสดงที่หน้าจอ

'การแสดงผลข้อมูลในชุดที่ 3 ที่รับมาจาก port

DataInput = Left(a\$, 18) ' เอาข้อมูล 18 ตัวจากทางซ้าย ไปเก็บไว้ในตัวแปร DataInput

DataInput = Right(DataInput, 5) 'เอาข้อมูล 5 ตัวจากทางขวาใน ตัวแปร DataInput ไปเก็บในตัวแปร DataInput

Bit3Text.Text = DataInput ' เอาข้อมูลในตัวแปร Datainput ไปแสดงที่หน้าจอ เป็นค่า Time

'การแสดงผลข้อมูลในชุดที่ 4 ที่รับมาจาก port

DataInput = Left(a\$, 20) ' เอาข้อมูล 20 ตัวจากทางซ้าย ไปเก็บไว้ในตัวแปร DataInput

DataInput = Right(DataInput, 2) 'เอาข้อมูล 2 ตัวจากทางขวาใน ตัวแปร DataInput ไปเก็บในตัวแปร DataInput

Bit4Text.Text = DataInput ' เอาข้อมูลในตัวแปร Datainput ไปแสดงที่หน้าจอจะเป็นค่า Quantity

TotalBitText.Text = a\$ ' แสดงข้อมูลทั้ง 5 =6f ที่ได้รับจาก port บนหน้าจอ

If DataError=4 Then

AddDataAuto ' เก็บข้อมูลที่รับมาจาก port ลงใน record ของ ฐานข้อมูลโดยอัตโนมัติ

End If

frmMainMenu.Commu1.PortOpen = False ' ปิด com port

End Sub

Public Sub AddDataAuto()

' เก็บข้อมูลที่รับจาก com port ลงใน record โดย อัตโนมัติ

Dim XDate, XSize, XTime, XQuantity As String

XSize = Bit1Text.Text ' โอนข้อมูลของ Size ที่รับมา เก็บไว้ใน XSize

XDate = Bit2Text.Text ' โอนข้อมูลของ Date ที่รับมา เก็บไว้ใน XDate

XTime = Bit3Text.Text ' โอนข้อมูลของ Time ที่รับมา เก็บไว้ใน XTime

XQuantity = Bit4Text.Text ' โอนข้อมูลของ Quantity ที่รับมา เก็บไว้ใน XQuantity

DataTable1.Recordset.AddNew ' นำข้อมูลทั้งหมดที่รับมาจาก com port ไปเก็บลงใน record

Bit1Text.Text = XSize 'เอาข้อมูลใน Xsize ไปแสดงบนจอ

Bit2Text.Text = XDate 'เอาข้อมูลใน XDate ไปแสดงบนจอ

Bit3Text.Text = XTime 'เอาข้อมูลใน XTime ไปแสดงบนจอ

Bit4Text.Text = XQuantity 'เอาข้อมูลใน XQuantity ไปแสดงบนจอ

End Sub

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

VERSION 4.00

Begin VB.Form frmShowDataBase

Caption = "Show Data Base"

ClientHeight = 3645

ClientLeft = 2970

ClientTop = 1920

ClientWidth = 6690

Height = 4050

Left = 2910

LinkTopic = "Form2"

LockControls = -1 'True

ScaleHeight = 3645

ScaleWidth = 6690

Top = 1575

Visible = 0 'False

Width = 6810

Begin VB.CommandButton cmExit

Caption = "Exit"

Height = 495

Left = 4680

TabIndex = 12

Top = 2880

Width = 1215

End

Begin VB.CommandButton cmAddNew

Caption = "Add New"

Height = 495

Left = 4680

TabIndex = 11

Top = 1440

Width = 1215

End

Begin VB.CommandButton cmNext

Caption = "Next "

Height = 495

Left = 4680

TabIndex = 10

Top = 240

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Width = 1215

End

Begin VB.CommandButton cmPrevious

Caption = "Previous"

Height = 495

Left = 4680

TabIndex = 9

Top = 840

Width = 1215

End

Begin VB.CommandButton cmDelete

Caption = "Delete"

Height = 495

Left = 4680

TabIndex = 8

Top = 2040

Width = 1215

End

Begin VB.Data DataTable2

Caption = "Data1"

Connect = "Access"

DatabaseName = ""

Exclusive = -1 'True

Height = 315

Left = 2160

Negotiate = -1 'True

Options = 0

ReadOnly = 0 'False

RecordsetType = 1 'Dynaset

RecordSource = "DataRain"

Top = 2160

Visible = 0 'False

Width = 2415

End

Begin VB.TextBox txtTime

DataField = "Time"

DataSource = "DataTable2"

Height = 495

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Left = 840
TabIndex = 3
Top = 1200
Width = 1215

End

Begin VB.TextBox txtDate

DataField = "Date"
DataSource = "DataTable2"
Height = 495
Left = 840
TabIndex = 2
Top = 360
Width = 1215

End

Begin VB.TextBox txtSize

DataField = "Size"
DataSource = "DataTable2"
Height = 495
Left = 840
TabIndex = 1
Top = 2040
Width = 1215

End

Begin VB.TextBox txtQuantity

DataField = "Quantity"
DataSource = "DataTable2"
Height = 495
Left = 840
TabIndex = 0
Top = 2880
Width = 1215

End

Begin VB.Label Label4

Caption = "Date :"
Height = 495
Left = 0
TabIndex = 7
Top = 360

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

Width      = 975
End
Begin VB.Label Label3
Caption     = "Time :."
Height     = 495
Left       = 0
TabIndex   = 6
Top        = 1200
Width      = 975

```

End

Begin VB.Label Label2

```

Caption     = "Size :."
Height     = 495
Left       = 0
TabIndex   = 5
Top        = 2040
Width      = 975

```

End

Begin VB.Label Label1

```

Caption     = "Quantity :."
Height     = 495
Left       = 0
TabIndex   = 4
Top        = 2880
Width      = 975

```

End

End

Attribute VB_Name = "frmShowDataBase"

Attribute VB_Creatable = False

Attribute VB_Exposed = False

Private Sub CmAddNew_Click()

' ทำการเพิ่มข้อมูลใหม่ลงไปใน file Data base

DataTable2.Recordset.AddNew ' เพิ่มข้อมูลใหม่ลงไปในตาราง record

txtDate.SetFocus ' set cursor ให้อยู่ที่ ของ txtDate

End Sub

Private Sub cmDelete_Click()

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

' ทำการลบข้อมูลเก่าที่มีอยู่ ออกจาก File Data Base

If DataTable2.Recordset.EOF Then ' เช็ค record ปัจจุบันที่แสดงอยู่บนหน้าจอว่าเป็น Record สุดท้ายในของ
ตารางข้อมูล (End Of File)

MsgBox " This is last record " , , " Error #14001" 'แสดง Dialog ERROR บอกว่า ไม่มีข้อมูลอยู่ ลบไม่ได้

Else ' เช็ค record ปัจจุบันที่แสดงบนจอ ว่าไม่ใช่ record สุดท้าย

DataTable2.Recordset.Delete ' ทำการลบ ข้อมูลใน record นั้น ซะ

DataTable2.Recordset.MoveNext ' แสดงข้อมูลใน record ถัดไป

txtDate.SetFocus ' set cursor ไปแสดงที่ txtDate

End If

End Sub

Private Sub cmexit_Click()

' ออกจาก Form ย่อย แต่ยังอยู่ในโปรแกรมหลัก

Status = stSaveView

Unload frmShowDataBase ' ปิด form ที่แสดงข้อมูลที่ได้รับจาก com port

End Sub

Private Sub CmNext_Click()

' แสดงข้อมูลใน record ถัดไป บนหน้าจอ

If DataTable2.Recordset.EOF Or DataTable2.Recordset.BOF Then ' เช็คไว้ใน file Data Base นั้น ไม่มีข้อมูลอยู่

MsgBox "No Record in Data File " , , " Error #14003" ' แสดง dialogBox ERROR บอกว่าไม่มีข้อมูลหรือ record

ใดๆอยู่เลย

Else ' เช็คว่ามีข้อมูลหรือ record อยู่ใน File Data Base

DataTable2.Recordset.MoveNext ' นำ ข้อมูลที่มีอยู่ ใน record ถัดไป แสดงบนจอ

If DataTable2.Recordset.EOF Then ' เช็ค ว่า record ถัดไปนั้นเป็น record สุดท้าย

MsgBox " This is last record " , , " Error #14004" ' แสดง dialog Box ERROR บอกว่านี่คือข้อมูลสุดท้าย

DataTable2.Recordset.MovePrevious ' แสดง ข้อมูลใน record สุดท้ายบนจอ

Else ' เช็ค record ถัดไปว่าไม่ใช่ record สุดท้าย

txtDate.SetFocus ' set cursor ให้แสดงที่ txtDate

End If

End If

End Sub

Private Sub CmPrevious_Click()

' แสดงข้อมูลใน record ก่อนหน้านี้นบนหน้าจอ

If DataTable2.Recordset.EOF Or DataTable2.Recordset.BOF Then ' เช็คไว้ใน file Data Base นั้น ไม่มีข้อมูลอยู่

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

MsgBox "No Record in Data File ", , " Error #14005" ' แสดง dialogBox ERROR บอกว่าไม่มีข้อมูลหรือ record
ใดๆอยู่เลย
Else ' เช็คว่ามีข้อมูลหรือ record อยู่ใน File Data Base
    DataTable2.Recordset.MovePrevious ' นำข้อมูลใน record ก่อนหน้านี้มาแสดงบนจอ
    If DataTable2.Recordset.BOF Then ' เช็คว่า record ก่อนหน้านี้ เป็น record แรก ( Begin Of File )
        MsgBox " This is First record ", , " Error #14006" ' แสดง Dialog Box Error บอกว่า ก่อนหน้านี้ไม่มีข้อมูล นี่คื
ข้อมูลแรก
        DataTable2.Recordset.MoveNext ' แสดงข้อมูลใน record แรกบนจอ
    Else ' เช็คว่า REcord ก่อนหน้านี้ไม่ใช่ record แรก
        txtDate.SetFocus ' set cursor ให้แสดงที่ txtDate
    End If
End If
End Sub

Private Sub Form_Load()
    DataTable2.DatabaseName = frmMainMenu.txtDataBaseName.Text
    frmShowDataBase.Caption = DataTable2.DatabaseName
End Sub

```

VERSION 4.00

Begin VB.Form frmShowGraph

BorderStyle = 1 'Fixed Single

Caption = "Show Graph"

ClientHeight = 5865

ClientLeft = 1620

ClientTop = 1545

ClientWidth = 7980

Height = 6270

Left = 1560

LinkTopic = "Form2"

LockControls = -1 'True

MaxButton = 0 'False

MinButton = 0 'False

ScaleHeight = 5865

ScaleWidth = 7980

Top = 1200

Visible = 0 'False

Width = 8100

Begin VB.OptionButton op5

Caption = "1 วัน"

Height = 375

Left = 6960

TabIndex = 18

Top = 3000

Width = 975

End

Begin VB.OptionButton op4

Caption = "12 ชั่วโมง"

Height = 375

Left = 6960

TabIndex = 17

Top = 2520

Width = 975

End

Begin VB.OptionButton op3

Caption = "6 ชั่วโมง"

Height = 375

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Left = 6960
TabIndex = 16
Top = 2040
Width = 975

End

Begin VB.OptionButton op2

Caption = "2 ชั่วโมง"
Height = 375
Left = 6960
TabIndex = 15
Top = 1560
Width = 975

End

Begin VB.OptionButton op1

Caption = "1 ชั่วโมง"
Height = 375
Left = 6960
TabIndex = 14
Top = 1080
Value = -1 'True
Width = 975

End

Begin VB.CommandButton CMCLS

Caption = "Clear"
Height = 375
Left = 7080
TabIndex = 13
Top = 120
Visible = 0 'False
Width = 855

End

Begin VB.PictureBox pic1

Height = 3135
Left = 1080
Picture = "ShowGraph.frx":0000
ScaleHeight = 3075
ScaleWidth = 5715
TabIndex = 11

Top = 240
Width = 5775

End

Begin VB.Data DataTable2

Caption = "Data1"
Connect = "Access"
DatabaseName = ""
Exclusive = -1 'True
Height = 315
Left = 0
Negotiate = -1 'True
Options = 0
ReadOnly = 0 'False
RecordsetType = 1 'Dynaset
RecordSource = "DataRain"
Top = 1800
Visible = 0 'False
Width = 1140

End

Begin VB.TextBox txtSize

Alignment = 1 'Right Justify
DataField = "Size"
DataSource = "DataTable2"

BeginProperty Font

name = "MS Sans Serif"
charset = 222
weight = 700
size = 12
underline = 0 'False
italic = 0 'False
striketrough = 0 'False

EndProperty

Height = 495
Left = 0
TabIndex = 1
Top = 480
Visible = 0 'False
Width = 495

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

End

Begin VB.TextBox txtQuantity

Alignment = 1 'Right Justify

DataField = "Quantity"

DataSource = "DataTable2"

BeginProperty Font

name = "MS Sans Serif"

charset = 222

weight = 700

size = 12

underline = 0 'False

italic = 0 'False

striketrough = 0 'False

EndProperty

Height = 495

Left = 0

TabIndex = 0

Top = 1080

Visible = 0 'False

Width = 495

End

Begin VB.Label txttotal

Alignment = 2 'Center

BorderStyle = 1 'Fixed Single

BeginProperty Font

name = "MS Sans Serif"

charset = 222

weight = 700

size = 14.25

underline = 0 'False

italic = 0 'False

striketrough = 0 'False

EndProperty

Height = 495

Left = 3000

TabIndex = 23

Top = 5280

Width = 975

เอกสารนี้เป็นเอกสารสงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

End

Begin VB.Label txttime

Alignment = 2 'Center
BorderStyle = 1 'Fixed Single
DataField = "time"
DataSource = "DataTable2"

BeginProperty Font

name = "MS Sans Serif"
charset = 222
weight = 700
size = 14.25
underline = 0 'False
italic = 0 'False
striketrough = 0 'False

EndProperty

Height = 495
Left = 4440
TabIndex = 22
Top = 4560
Width = 1815

End

Begin VB.Label txtdate

Alignment = 2 'Center
BorderStyle = 1 'Fixed Single
DataField = "date"
DataSource = "DataTable2"

BeginProperty Font

name = "MS Sans Serif"
charset = 222
weight = 700
size = 14.25
underline = 0 'False
italic = 0 'False
striketrough = 0 'False

EndProperty

Height = 495
Left = 1320
TabIndex = 21

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Top = 4560

Width = 1935

End

Begin VB.Label head

Alignment = 2 'Center

Caption = "เลือกกราฟ"

BeginProperty Font

name = "MS Sans Serif"

charset = 222

weight = 700

size = 9.75

underline = -1 'True

italic = 0 'False

striketrough = 0 'False

EndProperty

Height = 375

Left = 6960

TabIndex = 20

Top = 720

Width = 975

End

Begin VB.Label topic

Alignment = 2 'Center

BorderStyle = 1 'Fixed Single

BeginProperty Font

name = "MS Sans Serif"

charset = 222

weight = 700

size = 9.75

underline = 0 'False

italic = 0 'False

striketrough = 0 'False

EndProperty

Height = 375

Left = 3120

TabIndex = 19

Top = 0

Width = 1695

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

End

Begin VB.Label lb1

Alignment = 2 'Center

BorderStyle = 1 'Fixed Single

Caption = "ต้องการดูกราฟกรณีนากดปุ่มกราฟ"

BeginProperty Font

name = "MS Sans Serif"

charset = 222

weight = 400

size = 18

underline = 0 'False

italic = 0 'False

striketrough = 0 'False

EndProperty

Height = 615

Left = 1080

TabIndex = 12

Top = 3480

Width = 5775

End

Begin VB.Label Label5

Alignment = 2 'Center

BackStyle = 0 'Transparent

Caption = " मिलिเมตร/ชั่วโมง"

BeginProperty Font

name = "MS Sans Serif"

charset = 222

weight = 400

size = 14.25

underline = 0 'False

italic = 0 'False

striketrough = 0 'False

EndProperty

Height = 495

Left = 4200

TabIndex = 10

Top = 5280

Width = 2055

เอกสารนี้เป็นเอกสารสงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

End

Begin VB.Label Label1

BackStyle = 0 'Transparent

Caption = "ปริมาณน้ำฝนที่วัดได้:"

BeginProperty Font

name = "MS Sans Serif"

charset = 222

weight = 400

size = 14.25

underline = 0 'False

italic = 0 'False

striketrough = 0 'False

EndProperty

Height = 495

Left = 240

TabIndex = 9

Top = 5280

Width = 2415

End

Begin Threed.SSCommand cmexit

Height = 600

Left = 7200

TabIndex = 8

Top = 4440

Width = 600

_Version = 65536

_ExtentX = 1058

_ExtentY = 1058

_StockProps = 78

AutoSize = 2

Picture = "ShowGraph.frx":1322

End

Begin Threed.SSCommand plotgraph

Height = 600

Left = 6480

TabIndex = 7

Top = 4440

Width = 600

เอกสารนี้เป็นเอกสารทสงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
_Version      = 65536
_ExtentX      = 1058
_ExtentY      = 1058
_StockProps   = 78
AutoSize      = 2
Picture       = "ShowGraph.frx":1774
```

End

Begin VB.Line Line1

```
BorderWidth   = 5
X1            = 0
X2            = 7920
Y1            = 4320
Y2            = 4320
```

End

Begin VB.Label Label6

```
Alignment      = 2 'Center
AutoSize       = -1 'True
BackStyle      = 0 'Transparent
Caption        = "ปริมาณ"
BeginProperty Font
    name        = "MS Sans Serif"
    charset     = 222
    weight      = 700
    size        = 12
    underline   = 0 'False
    italic      = 0 'False
    strikethrough = 0 'False
```

EndProperty

```
Height        = 300
Left          = 600
TabIndex      = 6
Top           = 0
Visible       = 0 'False
Width         = 795
```

End

Begin Threed.SSCommand cmPrevious

```
Height        = 600
Left          = 6480
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
TabIndex    = 5
Top         = 5160
Width       = 600
_Version    = 65536
_ExtentX    = 1058
_ExtentY    = 1058
_StockProps = 78
AutoSize    = 2
Picture     = "ShowGraph.frx":1BC6
```

End

Begin Threed.SSCommand cmnext

```
Height      = 600
Left        = 7200
TabIndex    = 4
Top         = 5160
Width       = 600
_Version    = 65536
_ExtentX    = 1058
_ExtentY    = 1058
_StockProps = 78
AutoSize    = 2
Picture     = "ShowGraph.frx":2018
```

End

Begin VB.Label Label4

```
BackStyle   = 0 'Transparent
Caption     = "วันที่ :"
```

BeginProperty Font

```
name        = "MS Sans Serif"
charset     = 222
weight      = 400
size        = 14.25
underline   = 0 'False
italic      = 0 'False
strikethrough = 0 'False
```

EndProperty

```
Height      = 495
Left        = 240
```

```
TabIndex    = 3
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Top = 4560

Width = 975

End

Begin VB.Label Label3

BackStyle = 0 'Transparent

Caption = "เวลา :"

BeginProperty Font

name = "MS Sans Serif"

charset = 222

weight = 400

size = 14.25

underline = 0 'False

italic = 0 'False

striketrough = 0 'False

EndProperty

Height = 495

Left = 3600

TabIndex = 2

Top = 4560

Width = 975

End

End

Attribute VB_Name = "frmShowGraph"

Attribute VB_Creatable = False

Attribute VB_Exposed = False

Dim lsize, lquantity As String

Dim QTotal, Total, Watch As Single 'Integer 'String

Dim a, b, c As Integer

Dim tsize, tquantity As String

Sub firstshow()

DataTable2.Recordset.MoveFirst 'กำหนดจุดเริ่มต้นใหม่

showtotal

End Sub

Sub optionhide()

op1.Visible = False

op2.Visible = False

op3.Visible = False

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

op4.Visible = False
op5.Visible = False
head.Visible = False
topic.Visible = True
End Sub

Sub optionshow()
op1.Visible = True
op2.Visible = True
op3.Visible = True
op4.Visible = True
op5.Visible = True
head.Visible = True
topic.Visible = False
End Sub

```

```

Sub plotline()
pic1.Visible = False
lb1.Visible = False
CMCLS.Visible = True
Label6.Visible = True
Cls
Scale (-900, -900)-(900, 900)
Line (-700, 300)-(-790, 300), QBColor(5)
Line (-700, 300)-(-700, -750), QBColor(5)
End Sub

```

```

Sub showtotal()
Isize = DataTable2.Recordset.Fields("size").Value
Iquantity = DataTable2.Recordset.Fields("quantity").Value
QTotal = Isize * Iquantity * 60
txttotal.Caption = QTotal
End Sub

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

Sub startvalue()
    topic.Visible = False
    c = 24
    topic.Caption = "กราฟ" + op1.Caption
End Sub

```

```

Private Sub CMCLS_Click()
    Cls
    Label6.Visible = False
    pic1.Visible = True
    lb1.Visible = True
    CMCLS.Visible = False
    optionshow
    firstshow
End Sub

```

```

Private Sub cmexit_Click()
    'ออกจาก Form ย่อย แต่ยังคงอยู่ในโปรแกรมหลัก
    Unload frmShowGraph 'ปิด form ที่แสดงข้อมูลที่ได้รับจาก com port
    frmMainMenu.Enabled = True
End Sub

```

```

Private Sub CmNext_Click()
    'แสดงข้อมูลใน record ถัดไป บนหน้าจอ
    If DataTable2.Recordset.EOF Or DataTable2.Recordset.BOF Then 'คิดว่าใน file Data Base นั้น ไม่มีข้อมูลอยู่
        MsgBox "No Record in Data File ", , " Error #14003" 'แสดง dialogBox ERROR บอกว่าไม่มีข้อมูลหรือ record
        ใดๆอยู่เลย
    Else 'คิดว่ามีข้อมูลหรือ record อยู่ใน File Data Base
        DataTable2.Recordset.MoveNext 'นำ ข้อมูลที่มีอยู่ ใน record ถัดไป แสดงบนจอ
        If DataTable2.Recordset.EOF Then 'คิดว่า record ถัดไปนั้นเป็น record สุดท้าย
            txttotal.Caption = ""
            MsgBox " This is last record ", , " Error #14004" 'แสดง dialog Box ERROR บอกว่านี่คือข้อมูลสุดท้าย
            DataTable2.Recordset.MovePrevious 'แสดง ข้อมูลใน record สุดท้ายบนจอ
            showtotal
        Else
            showtotal
        End If
    End If

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

MsgBox " This is last record ", , " Error #14004" ' แสดง dialog Box ERROR บอกว่านี่คือข้อมูลสุดท้าย

DataTable2.Recordset.MovePrevious 'แสดง ข้อมูลใน record สุดท้ายบนจอ

End If

End If

End Sub

Private Sub CmPrevious_Click()

' แสดงข้อมูลใน record ก่อนหน้านั้นบนหน้าจอ

If DataTable2.Recordset.EOF Or DataTable2.Recordset.BOF Then ' เช็คว่าใน file Data Base นั้น ไม่มีข้อมูลอยู่

MsgBox "No Record in Data File ", , " Error #14005" ' แสดง dialogBox ERROR บอกว่าไม่มีข้อมูลหรือ record ใดๆอยู่เลย

Else ' เช็คว่ามีข้อมูลหรือ record อยู่ใน File Data Base

DataTable2.Recordset.MovePrevious ' นำข้อมูลใน record ก่อนหน้านั้นมาแสดงบนจอ

If DataTable2.Recordset.BOF Then ' เช็คว่า record ก่อนหน้านั้น เป็น record แรก (Begin Of File)

txttotal.Caption = ""

MsgBox " This is First record ", , " Error #14006" ' แสดง Dialog Box Error บอกว่า ก่อนหน้านั้นไม่มีข้อมูล นี่คื
ข้อมูลแรก

DataTable2.Recordset.MoveNext ' แสดงข้อมูลใน record แรกบนจอ

showtotal

Else

showtotal

End If

If DataTable2.Recordset.BOF Then ' เช็คว่า record ก่อนหน้านั้น เป็น record แรก (Begin Of File)

MsgBox " This is First record ", , " Error #14006" ' แสดง Dialog Box Error บอกว่า ก่อนหน้านั้นไม่มีข้อมูล นี่คื

ข้อมูลแรก

DataTable2.Recordset.MoveNext ' แสดงข้อมูลใน record แรกบนจอ

End If

End If

End Sub

Private Sub Form_Activate()

firstshow

End Sub

Private Sub Form_Load()

If WindowsState = 0 Then

Move (Screen.Width - frmShowGraph.Width) / 2, (Screen.Height - frmShowGraph.Height) / 2

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
End If  
DataTable2.DatabaseName = frmMainMenu.txtDataBaseName.Text  
frmShowGraph.Caption = DataTable2.DatabaseName ' + " - Menu Raining Meter version 1.0"  
startvalue  
End Sub
```

```
Private Sub op1_Click()  
    c = 24  
    topic.Caption = "กราฟ" + op1.Caption  
End Sub
```

```
Private Sub op2_Click()  
    c = 12  
    topic.Caption = "กราฟ" + op2.Caption  
End Sub
```

```
Private Sub op3_Click()  
    c = 4  
    topic.Caption = "กราฟ" + op3.Caption  
End Sub
```

```
Private Sub op4_Click()  
    c = 2  
    topic.Caption = "กราฟ" + op4.Caption  
End Sub
```

```
Private Sub op5_Click()  
    c = 1  
    topic.Caption = "กราฟ" + op5.Caption  
End Sub
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
Private Sub plotgraph_Click()  
    optionhide  
    plotline  
    a = -700  
    b = 300  
    DataTable2.Recordset.MoveFirst  
    Do  
        Isize = DataTable2.Recordset.Fields("size").Value  
        Iquantity = DataTable2.Recordset.Fields("quantity").Value  
        Total = Isize * Iquantity * 100  
        Watch = c + a  
        Line (a, b)-(Watch, -(Total - 300)), QBColor(2)  
        a = Watch  
        b = 300 - Total  
        DataTable2.Recordset.MoveNext  
    Loop Until DataTable2.Recordset.EOF  
    firstshow  
End Sub
```

'สร้างแกนกราฟ

'กำหนดจุดเริ่มต้นของข้อมูล

'ดึงข้อมูลจาก Record

'ดึงข้อมูลจาก Record

'Plot กราฟ

'เลื่อนข้อมูล

'เช็คข้อมูลว่าหมดหรือยัง



Attribute VB_Name = "Module1"

DefInt A-Z

Public Const stNewFile = 1

Public Const stNothing = 0

Public Const stSaveFile = 2

Public Const stOpenFile = 3

Public Const stNonSave = 4

Public Const stLoading = 5

Public Const stOpenView = 6

Public Const stOpenEdit = 7

Public Const stSaveView = 8

Public Const nsNot = 0

Public Const nsSave = 1

Public Const fodActive = 1

Public Const fodPasstive = 0

Public Status

Public mnuhelp

Public FileOpenDuble

Public NotSave

' Public variables

Public Echo ' Echo On/Off flag.

Public Const vbMSCommEvReceive = 2

Public Const vbMSCommEvSend = 1

Public Const vbMSCommEvCTS = 3

Public Const vbMSCommEvDSR = 4

Public Const vbMSCommEvCD = 5

Public Const vbMSCommEvRing = 6

Public Const vbMSCommEvEOF = 7

Public Const vbMSCommErBreak = 1001

Public Const vbMSCommErCTSTO = 1002

Public Const vbMSCommErDSRTO = 1003

```
Public Const vbMSCommErFrame = 1004
Public Const vbMSCommErOverrun = 1005
Public Const vbMSCommErCDTO = 1006
Public Const vbMSCommErRxOver = 1007
Public Const vbMSCommErRxParity = 1008
Public Const vbMSCommErTxFull = 1009
```

```
#If Win32 Then
```

```
    Declare Sub SetWindowPos Lib "user32" (ByVal hWnd As Long, ByVal hWndInsertAfter As Long, ByVal X As Long, ByVal Y As Long, ByVal cx As Long, ByVal cy As Long, ByVal wFlags As Long)
```

```
#Else
```

```
    Declare Sub SetWindowPos Lib "USER" (ByVal hWnd%, ByVal hWndInsertAfter%, ByVal X%, ByVal Y%, ByVal cx%, ByVal cy%, ByVal wFlags%)
```

```
#End If
```



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กิตติกรรมประกาศ

ขอขอบพระคุณผู้ช่วยศาสตราจารย์ นิภา ลีลาสุทธิ เป็นอย่างสูง ที่ได้ให้การประสิทธิ์ประสาทวิชาการ แก่ผู้จัดทำ ตลอดจนช่วยแนะนำและให้คำปรึกษาในการทำปริญญานิพนธ์นี้ ขอขอบพระคุณอาจารย์และเจ้าหน้าที่ภาควิชาวิศวกรรมโทรคมนาคมที่มีส่วนช่วยเหลือ ในการทำปริญญานิพนธ์นี้ จนสำเร็จลุล่วงด้วยดี



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หนังสืออ้างอิง

โชคชัย เดชพชรรุ่ง “Visual Basic 4 สำหรับ Windows 95 Step by Step”

บริษัท ซีเอ็ดยูเคชั่น จำกัด, กรุงเทพฯ, 2539

ผศ.สมยศ จุณณะปิยะ “การใช้งานไมโครคอนโทรลเลอร์ตระกูล MCS-51”

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง, กรุงเทพฯ, 2537

ปรเมษฐ์ ประณยานันท์ - ปิยพงษ์ เผ่าวนิช “คู่มือและการประยุกต์ใช้งานไมโครคอนโทรลเลอร์ MCS-51”

บริษัท ซีเอ็ดยูเคชั่น จำกัด, กรุงเทพฯ, 2536

ราบินเดอร์ ศรีกิจจาภรณ์ “คู่มือการใช้งาน Visual Basic สำหรับวินโดวส์”

บริษัท ซีเอ็ดยูเคชั่น จำกัด, กรุงเทพฯ, 2538

“ANT-32 VERSION 3.0 EMBEDDED CONTROL BOARD “

บริษัท ศิลาเรีเสิร์ช จำกัด, กรุงเทพฯ

“DMCxxx USER'S MANUAL DOT MATRIX LCD MODULE”

บริษัท ศิลาเรีเสิร์ช จำกัด, กรุงเทพฯ