สำนักหอสมุดกลาง พระจอมเกล้าลาดกระบัง

การพัฒนาเครื่องวัดปริมาณน้ำฝน

RAIN GAUGE & DEVELOPMENT

โดย

นายเจริญจิตต์ บัวชุม

นายพรชัย ตันตโรภาส

นายสมเกียรติ เกียรติศิริกุล

เลขหมู่.....
เลขทะเบียน 42250
วัน, เดือน, ปี 16 พ.ค. 2545

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

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาอุตสาหกรรมศาสตรบัณฑิต

สาขาเทคโนโลยีอิเล็กทรอนิกส์ ภาควิชาเทคนิคอุตสาหกรรม

คณะวิศวกรรมศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2544

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หัวข้อปริญญานิพนธ์ การพัฒนาเครื่องวัดปริมาณน้ำฝน
Rain Gauge & Development

นักศึกษา นายเจริญจิตต์ บัวชุม รหัสประจำตัว 42015501
นายพรชัย ตันตโรภาส รหัสประจำตัว 42015516
นายสมเกียรติ เกียรติศิริกุล รหัสประจำตัว 42015530

อาจารย์ที่ปรึกษา ผู้ช่วยศาสตราจารย์อุทัย ศรีธีระวิโรจน์
ผู้ช่วยศาสตราจารย์อรลาภ แสงอรุณ

ภาควิชา เทคนิคอุตสาหกรรม

คณะ วิศวกรรมศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2544

คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง อนุมัติให้นับปริญญานิพนธ์ฉบับนี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรอุตสาหกรรมศาสตรบัณฑิต

คณะกรรมการสอบปริญญานิพนธ์

.....อาจารย์ที่ปรึกษา
(ผู้ช่วยศาสตราจารย์อุทัย ศรีธีระวิโรจน์)

.....อาจารย์ที่ปรึกษา
(ผู้ช่วยศาสตราจารย์อรลาภ แสงอรุณ)

.....กรรมการ
( )

.....กรรมการ
( )

.....กรรมการ
( )

ลิขสิทธิ์ของคณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หัวข้อปริญญานิพนธ์	การพัฒนาเครื่องวัดปริมาณน้ำฝน Rain Gauge & Development	
นักศึกษา	นายเจริญจิตต์ บัวชุม	รหัสประจำตัว 42015501
	นายพรชัย ตันตโรภาส	รหัสประจำตัว 42015516
	นายสมเกียรติ เกียรติศิริกุล	รหัสประจำตัว 42015530
อาจารย์ที่ปรึกษา	ผู้ช่วยศาสตราจารย์อุทัย ศรีธีระวิโรจน์	
	ผู้ช่วยศาสตราจารย์อรลาภ แสงอรุณ	
ปีการศึกษา	2544	

บทคัดย่อ

การตรวจสอบการลดทอนของสัญญาณดาวเทียม เนื่องจากฝนเป็นปัญหาที่สำคัญ และทำได้ยาก เนื่องจากลักษณะการเกิดของฝนไม่แน่นอน เครื่องวัดปริมาณน้ำฝนที่ใช้อยู่ทั่วไปเป็นชนิดถังเอียงเทน้ำเมื่อน้ำเต็ม เมื่อวัดปริมาณน้ำฝนมีโอกาสคลาดเคลื่อนได้สูง เมื่อฝนตกหนักไม่สามารถนำค่าขณะนั้นมาใช้ได้ โครงงานนี้ได้สร้างและพัฒนาเครื่องวัดปริมาณน้ำฝนชนิดถังเอียงเทน้ำแบบใหม่ขึ้น โดยมีระบบการนับจำนวนครั้ง และเวลาการเทน้ำแบบอัตโนมัติ แล้วนำผลที่ได้ไปเก็บ และประมวลผลด้วยไมโครคอมพิวเตอร์ และเก็บค่าไว้ในฐานข้อมูล สามารถเรียกดูข้อมูลระยะไกลได้ โดยติดต่อผ่านสายโทรศัพท์สาธารณะ

จากผลการทดลองวัดสามารถเก็บ และประมวลผลปริมาณน้ำฝนได้ตามต้องการ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Thesis Title	Rain Gauge & Development	
Student	Mr.Charoenchit Bouchum	ID. 42015501
	Mr.Pornchai Tantaropast	ID. 42015516
	Mr.Somkiat Kiatsirikul	ID. 42015530
Adviser	Assist.Prof.Uthai Sritheeravirojana	
	Assist.Prof.Ornlarp Sangaroon	
Academic Year	2001	

ABSTRACT

The investigations of radio waves signal attenuation due to rainfall is a problem and slowly grow, because the features of the rainfall are not clear. Conventional rain gauges such as tipping bucket type take with large error detection for heavy rains. Furthermore, the instantaneous values are not available. This project are fabricated and developed the tipping bucket rain gauge by using computer's series interface port RS-232C. In order to get rid of data, computer process the signal and stores information in memory for later uploading to analysis, automatic recorder for time setting and connect to server computer by telephone line. By the use of rain gauge, the analysis instantaneous intensity of rainfall has become possible.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กิตติกรรมประกาศ

ปริญญานิพนธ์ฉบับนี้สำเร็จลุล่วงได้ดี ขอขอบพระคุณ ผศ.อุทัย ศรีธีระวิโรจน์ และ ผศ.อรลาภ แสงอรุณ ที่ให้คำแนะนำและความช่วยเหลือเสมอมาตลอดจนชี้แนวทางในการแก้ไขปัญหา

ขอขอบพระคุณ คุณอำรุงฤทธิ์ นิ่มเสมอ หัวหน้างานสอบเทียบเครื่องมือ, พี่มานพ และ พี่ๆ ในกองช่างเครื่องมือ กรมอุตุนิยมวิทยา , คุณจตุพร สุขุมจิตพิทโยทัย ที่ให้ความช่วยเหลือในทุกๆด้าน

สุดท้ายขอขอบพระคุณ คุณพ่อ คุณแม่ ที่ให้การสนับสนุนด้านการศึกษาเสมอมา และ เพื่อนๆ ที่คอยให้กำลังใจรวมทั้งฝ่ายเสบียงที่คอยส่งข้าวส่งน้ำ

นายเจริญจิตต์ บัวชุม
นายพรชัย ตันตโรภาส
นายสมเกียรติ เกียรติศิริกุล
ผู้จัดทำ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ

	หน้า
บทคัดย่อภาษาไทย	I
บทคัดย่อภาษาอังกฤษ	II
กิตติกรรมประกาศ	III
บทที่ 1 บทนำ	1
1.1 วัตถุประสงค์	1
1.2 แนวความคิดและที่มา	1
1.3 ขอบเขตของโครงงาน	2
1.4 ความต้องการของระบบ	2
บทที่ 2 การออกแบบ Rain Gauge	3
2.1 ข้อกำหนดโดยทั่วไป	3
2.2 หน่วยที่ใช้ในการวัด	3
2.3 การวัดจำนวนน้ำฝน	3
2.4 หลักการทำงานทั่วไป	6
2.5 การออกแบบ Sensor คานกระดก	7
2.6 หลักการทำงานของ Sensor คานกระดก	9
บทที่ 3 วงจรควบคุม	12
3.1 คุณสมบัติของ AT89C52	12
3.2 การรับส่งข้อมูลอนุกรม (UART)	12
3.3 การเชื่อมต่อแบบมาตรฐาน RS-232C	14
3.4 การใช้งานไอซีสร้างฐานเวลาจริง (RTC)	15
3.5 การเชื่อมต่อกับหน่วยความจำข้อมูลภายนอก	17
บทที่ 4 การออกแบบโปรแกรม	20
4.1 การออกแบบโปรแกรมด้วย Visual Basic	20
4.2 การออกแบบฐานข้อมูลด้วย Microsoft Access	21
บทที่ 5 ผลการทดลอง	29
5.1 ผลการทดลอง	29
5.2 ปัญหาที่เกิดขึ้นและแนวทางแก้ไข	30

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5.3 สรุปผล 30
บรรณานุกรม 32
ภาคผนวก ก. รายละเอียดวงจร
ภาคผนวก ข. โปรแกรมการใช้งาน
ภาคผนวก ค. ติดตั้งการติดต่อฐานข้อมูลระยะไกล
ภาคผนวก ง. เอกสารประกอบการใช้งาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูป

รูปที่	หน้า
1.1 รูปแบบโดยรวมของโครงงาน	2
2.1 การคำนวณหาปริมาตรที่ต้องการวัด	8
2.2 ขนาดของคานกระดก	8
2.3 คำนวณปริมาตรรวมของคานกระดก	9
2.4 ขนาดของถังรองรับน้ำฝน	10
2.5 ส่วนประกอบของคานกระดก	11
3.1 รูปแบบของสัญญาณข้อมูลแบบอนุกรมในโหมด 1	13
3.2 รีจิสเตอร์ควบคุมการทำงานและบอกสถานะการสื่อสารข้อมูลอนุกรม	14
3.3 แนวทางการเปลี่ยนระดับสัญญาณ TTL ไปเป็นระดับสัญญาณ RS-232C	15
3.4 ผังเวลาแสดงถึงลำดับของสัญญาณเมื่อมีการอ่านข้อมูลจากหน่วยความจำ	17
3.5 การเชื่อมต่อหน่วยความจำข้อมูลภายนอกกับ AT89C552	19
4.1 โปรแกรม Rain Gauge 2	21
4.2 ตารางฐานข้อมูลการเก็บน้ำฝน	22
4.3 การเรียกใช้โปรแกรม Microsoft Access	23
4.4 การสร้างฐานข้อมูลเปล่า	23
4.5 การบันทึกฐานข้อมูล และการตั้งชื่อไฟล์	24
4.6 การเลือกเข้าไปออกแบบฐานข้อมูลเปล่า	24
4.7 การบันทึกฐานข้อมูล และการตั้งชื่อไฟล์	25
4.8 การเลือกเข้าไปออกแบบฐานข้อมูลเปล่า	25
4.9 การเลือกทำตารางใหม่	26
4.10 หน้าต่างก่อนการออกแบบตาราง	26
4.11 การออกแบบตาราง	27
4.12 การบันทึกตารางฐานข้อมูล	27
4.13 หน้าต่างฐานข้อมูลที่สร้างเสร็จแล้ว	28
5.1 ผลการทดลองจากการวัดปริมาณน้ำฝน	29

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 1
บทนำ

1.1 วัตถุประสงค์

1. สามารถวิเคราะห์ปริมาณน้ำฝนได้
2. ศึกษาการทำงานของวงจรไมโครคอนโทรลเลอร์
3. ศึกษาการส่งข้อมูลผ่านทางพอร์ตอนุกรม
4. ศึกษาการเขียนโปรแกรมโดยใช้ Visual Basic
5. ศึกษาการรับส่งข้อมูลระยะไกลผ่านสายโทรศัพท์ (Remote Access)

1.2 แนวความคิดและที่มา

ในอดีต คอมพิวเตอร์ที่ทำงานโดยเก็บข้อมูลขนาดใหญ่ๆ อยู่ในตัวซึ่งเป็นการเก็บอย่างมีแบบแผน ค้นหาง่าย ต่อมา คอมพิวเตอร์ ได้มีการพัฒนา Software และ Hardware ที่มีความสามารถขึ้นเรื่อยๆ และด้วยเหตุนี้เองทำให้ คอมพิวเตอร์ ได้เข้ามามีบทบาทในชีวิตประจำวันมากขึ้นและยิ่งไปกว่านั้นทุกวันนี้การติดต่อได้แคบลงจากเทคโนโลยี อินเตอร์เน็ต จึงได้นำแนวคิดนี้มาประยุกต์เข้ากับเครื่องวัดปริมาณน้ำฝน (Rain Gauge) เพื่อให้สามารถบันทึกข้อมูล และสามารถนำข้อมูลมาวิเคราะห์ได้

การติดต่อนั้นจะใช้การติดต่อระยะไกล เพื่อที่สามารถนำไปใช้กับพื้นที่ต่างๆ ได้ โดยใช้ความสามารถของระบบปฏิบัติการ Windows 98 จัดทำให้เป็น Server เพื่อให้เครื่องลูกข่าย (Client) เข้าไปทำงานได้ โดยใช้การติดต่อผ่านทางสายโทรศัพท์ เหมือนกับการติดต่อกับศูนย์ให้บริการอินเตอร์เน็ต

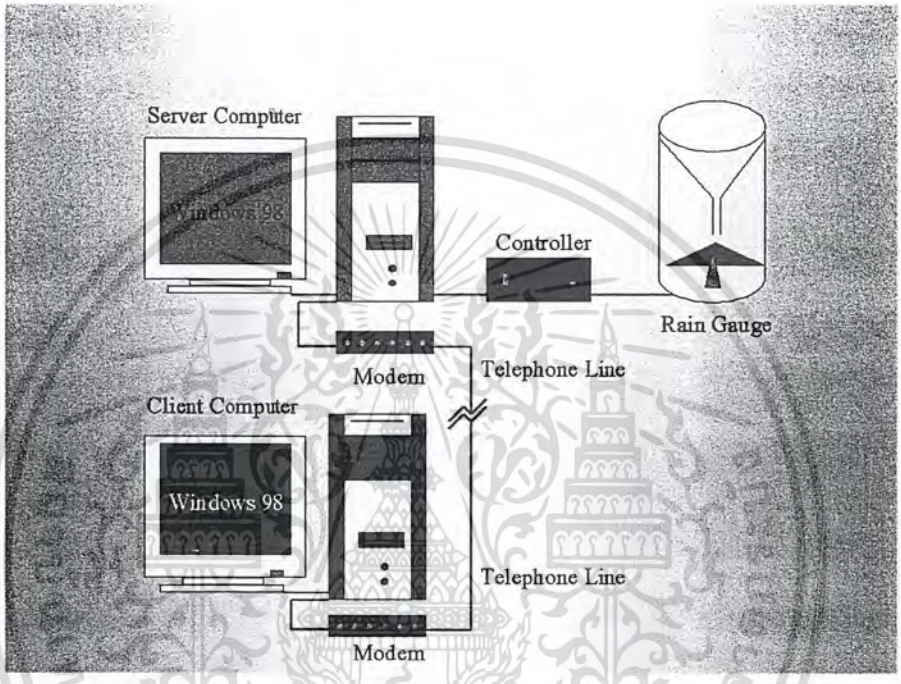
เครื่องวัดปริมาณน้ำฝนนี้จะมีพื้นที่รับน้ำฝนปากกว้าง 8 นิ้ว ตามมาตรฐานสากล ใช้หลักการของคานกระดก โดยนำ Lead Switch ไปติดไว้เพื่อสามารถแสดงเป็น Counter ได้ แล้วนำข้อมูลส่งผ่านไปยังชุดควบคุมไมโครคอนโทรลเลอร์

ข้อมูลที่เก็บได้จะจัดเก็บอยู่ในฐานข้อมูลซึ่งสามารถนำมาวิเคราะห์ปริมาณน้ำฝนได้ โดยผ่านโปรแกรมที่จัดทำขึ้นจาก Visual Basic

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1.3 ขอบเขตของโครงงาน

สามารถนำปริมาณน้ำฝนที่รับค่าเข้ามา เก็บข้อมูลผ่านทางวงจรไมโครคอนโทรลเลอร์ และสามารถแสดงผลในทางสถิติได้ และสามารถติดต่อจากเครื่องคอมพิวเตอร์ระยะไกลเพื่อสามารถนำข้อมูลที่ต้องการมาวิเคราะห์ได้ แม้จะอยู่ในสถานที่ที่ห่างไกลกัน



รูปที่ 1.1 รูปแบบโดยรวมของโครงงาน

1.4 ความต้องการของระบบ

1. เครื่องคอมพิวเตอร์ Server ติดตั้งระบบปฏิบัติการ Windows NT หรือ Windows 98 และลงโปรแกรม Visual Basic และติดตั้งโมเด็มแบบ External
2. เครื่องคอมพิวเตอร์ Client ติดตั้งระบบปฏิบัติการ Windows 98 ที่ติดตั้งโมเด็ม
3. โปรแกรม Rain Gauge สำหรับแสดงผล
4. คู่สายโทรศัพท์จำนวน 2 คู่สาย
5. เครื่องรับปริมาณน้ำฝน
6. ชุดควบคุมวงจรไมโครคอนโทรลเลอร์ที่เขียนโปรแกรมควบคุมไว้แล้ว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 2
การออกแบบ Rain Gauge

2.1 ข้อกำหนดโดยทั่วไป

จำนวนทั้งหมดของหยาดน้ำฝนซึ่งตกลงมาจากบรรยากาศบรรลุสู่ผิวพื้นโลกนั้น ใช้วัดเป็นความลึก(Depth) หรือความหนาของหยาดน้ำฟ้าบนพื้นระดับเรียบ โดยสมมติเอาว่าหยาดน้ำฟ้านั้นไม่มีการระเหยหรือซึมออกไป และถ้าเป็นจำนวนหิมะ หรือน้ำแข็งจะต้องไม่ละลายเสียก่อน และการวัดหิมะก็ต้องวัดเป็นความลึกของหิมะที่เพิ่งตกใหม่ๆบนพื้นระดับเรียบ สำหรับประเทศไทยยังไม่ปรากฏว่ามีหิมะตกลงมาเลย ส่วนลูกเห็บอาจมีบ้างเป็นครั้งคราว โดยมากจะเกิดเมื่อมีพายุฝนฟ้าคะนองอย่างแรง ดังนั้นเราจึงต้องการวัดแต่จำนวนน้ำฝนอย่างเดียว และโดยที่ประเทศไทยเป็นประเทศกสิกรรมจำนวนน้ำฝนย่อมมีความสำคัญมาก กรมอุตุนิยมวิทยาจึงมีโครงการที่จะขยายข่ายของสถานีวัดปริมาณน้ำฝนให้หนาแน่นยิ่งขึ้นเท่าที่จะทำได้ตามกำลังทรัพย์ที่จะอำนวย

2.2 หน่วยที่ใช้ในการวัด (Units of measurement)

ดังที่กล่าวมาแล้วว่า การวัดจำนวนน้ำฝนนั้นใช้วัดเป็นความสูงของน้ำฝน ที่ตกลงมาในเนื้อที่จำกัดอันหนึ่ง โดยคิดเสียว่าน้ำฝนไม่มีการระเหยหรือรั่วซึมหนีออกไปไหน โดยที่เรารู้เนื้อที่ของปากถัง ดังนั้นเมื่อเราต้องการวัดความสูง (หรือความลึก) ของน้ำฝน เราก็สามารถใช้ไม้บรรทัดหยั่งวัดความสูงได้ หรือจะทำเป็นแท่งแก้วสำหรับตวงน้ำฝนต่างหากก็ได้

หน่วยที่ใช้วัดจำนวนน้ำฝนที่ใช้วัดกันอยู่ทุกวันนี้ใช้วัดเป็น " มิลลิเมตร " การอ่านค่าของจำนวนน้ำฝนนั้นต้องวัดให้ได้ใกล้เคียงถึง 0.2 มิลลิเมตร สำหรับจำนวนน้ำฝน 10 มิลลิเมตร หรือน้อยกว่า แต่ถ้าจำนวนฝนมากกว่านี้ ย่อมให้ได้ค่าใกล้เคียง 2% จองจำนวนฝนทั้งหมด ในบางประเทศก็ใช้หน่วยที่วัดเป็นนิ้ว ซึ่งเทียบค่าได้ 1 นิ้ว = 25.4 มิลลิเมตร ก็นับว่าถูกต้องดี

2.3 การวัดจำนวนน้ำฝน (Rainfall measurement)

2.3.1 เครื่องวัดฝน (Rain gauge)

เริ่มแรกในการวัดจำนวนน้ำฝนนั้น Casteli ได้เป็นผู้คิดสร้างเครื่องวัดขึ้นในประเทศอิตาลี เมื่อปี ค.ศ. 1639 เครื่องประกอบด้วยแก้วทรงกระบอกมีเส้นผ่านศูนย์กลางปากกระบอกประมาณ 5 นิ้ว และลึก 9 นิ้ว ต่อมากจึงวิวัฒนาการเรื่อยมาจนถึงขนาดให้จดรายงานได้ด้วยตนเอง เครื่องวัดฝนธรรมดาที่ใช้เป็นประจำวัน นั้นเป็นรูปทรงกระบอก และมีกรวยต่อลงไปยังที่รองรับภายใน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ขนาดของถุงรับน้ำฝนภายนอกนั้นไม่สำคัญ แต่เนื้อที่ของบริเวณที่รับน้ำนั้นควรจะอยู่ระหว่าง 200 ถึง 500 ตร.ซม. เป็นเหมาะสมที่สุด เส้นผ่านศูนย์กลางของที่รองรับน้ำฝนภายใน ควรเท่ากับ 1/10 ของเส้นผ่านศูนย์กลางของถึงภายนอก โดยทั่วไปที่ใช้กันอยู่มีเส้นผ่านศูนย์กลางของปากถึง 8 นิ้ว , 6 นิ้ว และ 5 นิ้ว อย่างไรก็ดี ไม่ว่าจะใช้ถังขนาดไหนก็ตาม เครื่องวัดฝนที่ใช้เป็นประจำวันนั้นจะต้องเป็นไปตามหลักเกณฑ์ที่สำคัญดังต่อไปนี้

1. ขอบของปากถังด้านนอกต้องคม โดยทำให้ด้านในเป็นแนวดิ่งตั้งตรง ส่วนด้านนอกลาดเอียงลงเป็นแนวชันมาก
2. ต้องรู้เนื้อที่ของช่องปากถึงให้ใกล้เคียง 0.5% และเนื้อที่นี้จะต้องคงที่อยู่ตลอดไป
3. ถังภายนอกต้องออกแบบให้กันการกระเซ็นของน้ำฝนทั้งเข้าและออกในการนี้ ทำได้โดยทำให้ตัวถังเป็นแนวดิ่งลึกลงไปพอสมควร และแนวลาดของกรวยต้องมีความลาดอย่างน้อย 45 องศา
4. ถึงรองรับภายในต้องทำให้เป็นคอแคบๆ เพื่อป้องกันไม่ให้น้ำฝน ระเหยออกไปเนื่องจากราดิเอชั่น

2.3.2 การวัดฝน (Rain measure)

การวัดฝนโดยการวัดวิธีธรรมดาโดยทั่วไปนั้นมี 2 วิธี คือใช้แก้วตวงอย่างหนึ่ง และใช้แบบไม่บรรทัดหยั่งวัด แก้วที่ใช้ตวงต้องเป็นแก้วใส และมีสัมประสิทธิ์ของการขยายตัวต่ำ และต้องบอกให้ชัดเจนด้วยว่า ใช้กับเครื่องวัดน้ำฝนขนาดใด ขนาดเส้นผ่านศูนย์กลางของแก้วตวงต้องไม่โตกว่าประมาณ 1/3 ของปากถึงของเครื่องวัดน้ำฝน และถ้าจะให้ดีควรเล็กกว่าขนาดที่กล่าวนี้

สเกลที่แบ่งไว้บนแก้วตวงนั้นต้องขีดให้ชัดเจนดี โดยทั่วไปมักขีดทุกๆ 0.2 มิลลิเมตร และเส้นจำนวนเต็มของสเกลต้องเขียนเลขกำกับไว้ด้วย ถ้าจะให้ดีแล้วควรขีดสเกลให้อ่านได้ถึง 0.1 มิลลิเมตร ในการวัดเพื่อให้ได้ค่าใกล้เคียงที่สุดควรจะอ่านอย่าให้อัตราผิดเกินกว่า บวกลบ 0.05 มิลลิเมตร เมื่อจำนวนฝนเกินกว่า 2 มิลลิเมตร และควรอ่านอย่าให้ผิดเกินกว่า บวกลบ 0.02 มิลลิเมตร เมื่อจำนวนฝนน้อยกว่า 2 มิลลิเมตร

เพื่อที่จะให้ได้ค่าที่ถูกต้องแม่นยำจริงๆ เมื่อมีจำนวนฝนน้อย เขาทำกระบอกแก้วตวงตอนก้นสุดให้เรียวเล็กลง ในการตรวจทุกครั้งต้องถือแก้วตวงให้ได้แนวดิ่งจริงๆ เพื่อว่าระดับน้ำในแก้วตวงจะได้อยู่ในแนวนอนตามขีดสเกลทั้งเพื่อเป็นการป้องกันอัตราผิดเนื่องจากการเหลื่อม (Parallax errors) เพื่อช่วยในการนี้ควรจะขีดสเกลเฉพาะเลขหลักใหญ่ ซ้ำอีกทางด้านตรงข้ามกัน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ไม้บรรทัดหยั่งวัดนั้นควรทำด้วยไม้สีดา (Cedar wood) หรือวัตถุที่เหมาะสมอื่น เช่น Fiber-glass ซึ่งมีคุณสมบัติไม่ดูดน้ำ ขีดสเกลนั้นควรทำไว้อย่างน้อยทุกๆ 10 มิลลิเมตร อัตราผิดสูงสุดของขีดสเกลต้องไม่เกิน 0.5 มิลลิเมตร

ในการวัดด้วยไม้บรรทัดหยั่งวัดนี้ถ้าสามารถทำให้เทียบกับการวัดด้วยกระบอกแก้วตวงด้วยก็จะดี

2.3.3 การติดตั้งเครื่องมือ (Exposure)

ปรากฏว่าจำนวนน้ำฝนซึ่งจะเข้าไปในถังนั้นขึ้นอยู่กับลมประการหนึ่ง และความสูงของบรรดาสิ่งที่อยู่แวดล้อมอีกประการหนึ่ง ค่าที่เปลี่ยนแปลงเหล่านี้เป็นเหตุมาจากลม และส่วนใหญ่เป็นพวกลมหวนรอบๆ ถังวัดน้ำฝน กระแสอากาศไหลขึ้นทำให้ได้จำนวนฝนมากขึ้นกว่าความจริง ถ้ามีลมหวนอย่างแรง และบ่อยๆ จำนวนฝนที่ได้ก็ยิ่งน้อยกว่าความเป็นจริงมากขึ้น ความแรงและความถี่ ของลมหวนนี้ขึ้นอยู่กับความแรงของลมอย่างหนึ่ง ลักษณะโดยทั่วไปของพื้นดินรอบๆ เครื่องวัดอีกอย่างหนึ่ง ความห่างและความสูงของสิ่งที่อยู่แวดล้อมเครื่องวัดอย่างหนึ่ง และความสูงของเครื่องซึ่งต้องอยู่เหนือระดับพื้นดินอีกอย่างหนึ่ง ดังนั้นจึงจำเป็นต้องทำการเปรียบเทียบผลการตรวจของสถานีต่างๆด้วยว่าเป็นอย่างไร ในเมื่อสถานีต่างๆเหล่านั้นได้ทำการติดตั้งเครื่องมือเป็นแบบเดียวกัน

ตามมติคณะกรรมาธิการว่าด้วยเครื่องมือตรวจอากาศ และวิธีการตรวจ ในการประชุมสมัยที่ 3 ณ กรุงนิวเดลี เมื่อปี ค.ศ. 1962 ได้กำหนดให้ติดตั้งเครื่องวัดฝนไว้บนพื้นดินที่มีระดับเรียบ และสูงจากพื้นดิน 1 เมตร เป็นระดับมาตรฐานเดียวกันทั่วโลก ห้ามไม่ให้ตั้งบนพื้นที่ลาดชัน หรือพื้นที่ยกขึ้นมาอยู่ระดับสูง บนกำแพง หรือบนหลังคา และต้องไม่ติดตั้งในที่ๆลาดชันมากๆ ทางด้านซึ่งมีลมพัดอยู่ประจำ ระยะห่างของเครื่องต้องอยู่ห่างจากสิ่งกีดขวางแวดล้อม 4 เท่าของความสูงของสิ่งกีดขวางนั้นๆ

ฐานรองรับเครื่องอาจหล่อด้วยปูนซีเมนต์ ทำเป็นโครงรับให้พอดีกับเครื่อง และให้มีรูสำหรับระบายน้ำออก ฐานแท่นให้ติดแน่นอยู่กับพื้นดิน เพื่อว่ามีลมแรงๆเครื่องจะได้ไม่เอนเอียงหรืออาจจะถูกลมพัดให้ล้มได้ พื้นดินต้องเป็นหญ้าที่ตัดสั้น ถ้าไม่สามารถที่จะติดตั้งในพื้นหญ้าสั้นๆได้ อาจตั้งไว้บนพื้นราบแข็งๆ เช่น ลาดคอนกรีต เพราะจะทำให้น้ำฝนที่ตกลงมากระเซ็นขึ้นไปเข้าเครื่องรับ ทำให้จำนวนฝนมากกว่าความเป็นจริง

ปากถึงจะต้องให้ได้ระดับนอนจริงๆไม่เอนเอียง โดยจับด้วยระดับน้ำ ในทีลมแรงมากๆ เข้ามักใช้ขวดพลาสติกรองรับน้ำฝนภายใน เพราะสะดวกแก่การตวงวัดจำนวนน้ำฝนขณะที่มีลมแรงๆ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ควรหมั่นตรวจดูถังน้ำภายนอก และภายในเป็นระยะๆ เพราะถ้าหากรั่ว หรือชำรุดจะทำให้ผลการตรวจผิดไป แก้วตวงก็เช่นเดียวกัน เมื่อเห็นสิ่งใดชำรุดให้รีบจัดการแก้ไขเสียโดยด่วน หรือถ้าแก้ไขไม่ได้ก็ให้จัดการเบิกเปลี่ยนใหม่

ในประเทศที่มีตึกรามบ้านช่องหนาแน่น เช่น ในยุโรป หรือสหรัฐอเมริกา เขามักนำเครื่องไปติดตั้งไว้บนหลังคาที่เป็นดาดฟ้า แต่การติดตั้งไว้เช่นนี้ จำนวนน้ำฝนที่ตรวจได้ย่อมผิดกับการตรวจที่พื้นดินประมาณ 5-10% สำหรับประเทศไทยเรายังไม่มีความจำเป็นที่ต้องทำเช่นนั้น จึงกำหนดให้ติดตั้งไว้บนพื้นดินทั้งสิ้น

2.4 หลักการทำงานทั่วไป

หลักการทำงานของเครื่องวัดปริมาณน้ำฝน สามารถอธิบายได้ดังนี้คือ เมื่อมีการติดตั้งเครื่องวัดปริมาณน้ำฝน โดยนำถังรองน้ำฝนที่มีอุปกรณ์ที่ใช้วัดปริมาณน้ำฝน คอยตรวจวัดปริมาณน้ำฝนติดอยู่ไปติดตั้งไว้ในบริเวณที่ต้องการวัดปริมาณน้ำฝน จากนั้นต่อสายนำสัญญาณระหว่างส่วนรองรับปริมาณน้ำฝนเข้ากับส่วนควบคุมและต่อสายนำสัญญาณจากส่วนควบคุมเข้ากับเครื่องคอมพิวเตอร์ โดยเชื่อมต่อเข้าทางพอร์ตอนุกรมพอร์ตที่ 1 ของเครื่องคอมพิวเตอร์ จากนั้นต่อโมเด็มชนิดติดตั้งภายนอก (External modem) เข้ากับเครื่องคอมพิวเตอร์ โดยต่อเข้าทางพอร์ตอนุกรมพอร์ตที่ 2 เมื่อติดตั้งส่วนต่างๆพร้อมแล้วก็เปิดสวิทช์ส่วนควบคุม จากนั้นเปิดเครื่องคอมพิวเตอร์และรันโปรแกรม Rain gauge ทำการตั้งเวลาให้กับ Real time clock ในส่วนควบคุม เมื่อตั้งเวลาเรียบร้อยแล้วก็เป็นอันว่าเครื่องวัดปริมาณน้ำฝนพร้อมที่จะทำงานแล้ว และในกรณีกระแสไฟฟ้าขัดข้องเครื่องวัดปริมาณน้ำฝนจะยังคงทำงานต่อไปได้ด้วยพลังงานสำรองจากแบตเตอรี่ (Batterie)

เมื่อมีฝนตกอุปกรณ์รองรับน้ำฝนจะมีชุดคานกระดกเพื่อคอยรับน้ำ โดยเมื่อมีการกระดกของคานกระดกแต่ละครั้งจะมีการส่งสัญญาณไฟฟ้าไปให้กับไมโครคอนโทรลเลอร์ในส่วนควบคุมเพื่อทำการนับค่าการกระดกของคานรับน้ำฝน โดยคอนโทรลเลอร์จะทำการนับค่าการกระดกเก็บไว้ในหน่วยความจำเป็นช่วงเวลา โดยจะทำการแบ่งเวลาในแต่ละวันออกเป็น 24 ชั่วโมง เริ่มจากช่วงเวลา 07.00-08.00 เป็นช่วงเวลาที่ 1 ช่วงเวลา08.00-09.00 เป็นช่วงเวลาที่ 2 ช่วงเวลา 09.00-10.00 เป็นช่วงเวลาที่ 3 แบ่งอย่างนี้ไปเรื่อยๆ จนถึง 06.00 ถึง 07.00 เป็นช่วงเวลาที่ 24 เมื่อเครื่องคอนโทรลเลอร์ทำการเก็บค่าการกระดกครบถึง 24 ช่วงเวลาแล้ว (ซึ่งก็คือช่วงเวลาวนมาถึง 07.00 แล้ว) คอนโทรลเลอร์ซึ่งอยู่ในส่วนควบคุมจะทำการส่งสัญญาณไปให้กระตุ้นให้คอมพิวเตอร์ที่ต่ออยู่กับส่วนควบคุมทำการเปิดเครื่องคอมพิวเตอร์ พร้อมกับบูทเข้าระบบปฏิบัติการ เมื่อเข้าสู่ระบบ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

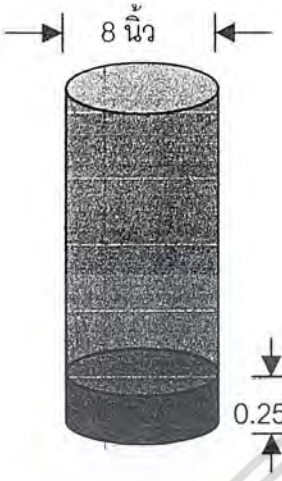
ปฏิบัติการแล้วจะทำการรันโปรแกรมRain gauge โดยอัตโนมัติ เมื่อโปรแกรมRain gauge ถูกรันขึ้นมากจะทำการสื่อสารกับส่วนควบคุมเพื่อทำการถ่ายโอนข้อมูลการกระดกจากไมโครคอนโทรลเลอร์เข้ามาเก็บไว้ในฐานข้อมูลโดยผ่านการทำงานของโปรแกรมRain gauge ซึ่งพัฒนาขึ้นมาโดยโปรแกรม Visual basic 6.0 และโปรแกรม Rain gauge จะทำการแปลงค่าข้อมูลการกระดกในแต่ละช่วงเวลาออกมาเป็นปริมาณน้ำฝนที่ตกในแต่ละช่วงเวลา (คานกระดกแต่ละครั้งมีค่าเท่ากับปริมาณน้ำฝน 0.25 มิลลิเมตร ที่ปากถังรับน้ำฝนกว้าง 8 นิ้ว) ก่อนเก็บเข้าไปในฐานข้อมูล เมื่อโปรแกรม Rain gauge ทำการรับค่าข้อมูลการกระดกจากส่วนควบคุมแล้วจะทำการสั่งให้เครื่องคอมพิวเตอร์ทำการปิดเครื่อง (Shut Down) ไมโครคอนโทรลเลอร์จะทำการเคลียร์ค่าในหน่วยความจำที่ใช้ในการเก็บค่าการกระดกในแต่ละช่วงเวลาให้เป็น 0 ทั้งหมด เพื่อรอรับการเก็บค่าการกระดกในรอบเวลาใหม่ต่อไป

เมื่อผู้ใช้งานมีความต้องการที่จะใช้ข้อมูลในฐานข้อมูลที่เครื่องคอมพิวเตอร์ที่ต่ออยู่กับชุดวัดปริมาณน้ำฝน แต่ว่าผู้ใช้งานอยู่ห่างไกลจากเครื่องวัดปริมาณน้ำฝนเป็นระยะทางไกลไม่สะดวกในการเดินทางเพื่อมาใช้งานฐานข้อมูล ก็สามารถทำการเข้าถึงข้อมูลจากระยะไกลได้โดยผู้ใช้งานจะต้องมีเครื่องคอมพิวเตอร์ พร้อมโมเด็มอีกชุดหนึ่งซึ่งมีการติดตั้งโปรแกรม Rain gauge ในส่วนของ Client ไว้พร้อมแล้ว จากนั้นจึงทำการ Dial-Up ไปสู่เครื่องคอมพิวเตอร์ที่ต่ออยู่กับชุดวัดปริมาณน้ำฝน ซึ่งในขณะนี้เครื่องคอมพิวเตอร์ที่ต่ออยู่กับชุดวัดปริมาณน้ำฝนจะทำหน้าที่เป็น Server เมื่อเครื่องคอมพิวเตอร์ Client ทำการ Dial-Up ไปที่เครื่อง Server จะมีการส่งสัญญาณโทรศัพท์ไปที่โมเด็มที่ต่ออยู่กับเครื่องคอมพิวเตอร์ Server เมื่อโมเด็มได้รับสัญญาณโทรศัพท์จะทำการส่งสัญญาณไปเปิดเครื่องคอมพิวเตอร์ Server เมื่อเครื่องคอมพิวเตอร์ Server ทำการ Boot ระบบแล้ว พร้อมกับรันโปรแกรม Raingauge โดยอัตโนมัติแล้วการเชื่อมต่อระหว่างเครื่องClient และ Server ก็จะสมบูรณ์จากนั้นเครื่องClient ก็สามารถใช้งานฐานข้อมูลในเครื่องคอมพิวเตอร์ Server จากระยะไกลได้

2.5 การออกแบบ Sensor คานกระดก

ในการออกแบบ Sensor คานกระดกนั้น เราต้องทราบค่าความละเอียดที่จะใช้ในการวัดก่อน โดยในโครงงานนี้ได้มีการใช้ค่าความละเอียดในการวัดคือ 0.25 มิลลิเมตร ที่ปากถังรับน้ำฝนขนาด 8 นิ้ว โดยในการออกแบบต้องคำนวณค่าปริมาตรน้ำฝนที่ขนาดความสูง 0.25 มิลลิเมตร ที่ปากถังขนาด 8 นิ้ว โดยเราสามารถคำนวณได้ดังต่อไปนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



$r = 4$ นิ้ว = 10.16 เซนติเมตร

$h = 0.25$ มิลลิเมตร = 0.025 เซนติเมตร

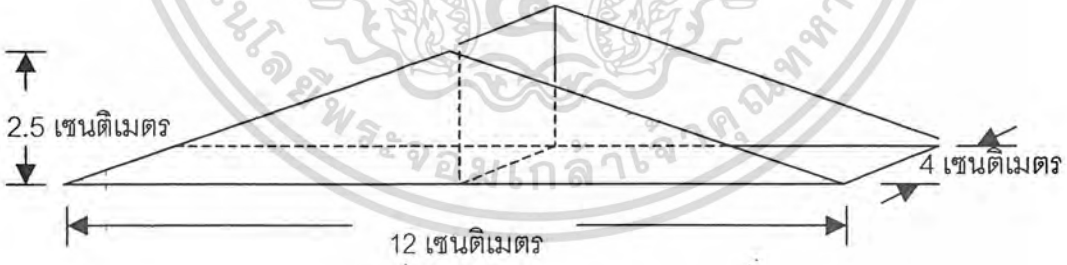
ปริมาตรทรงกระบอก $= \pi r^2 h$

$= \pi * (10.16)^2 * 0.025$

$= 8.107$ เซนติเมตร^3

รูปที่ 2.1 การคำนวณหาปริมาตรที่ต้องการวัด

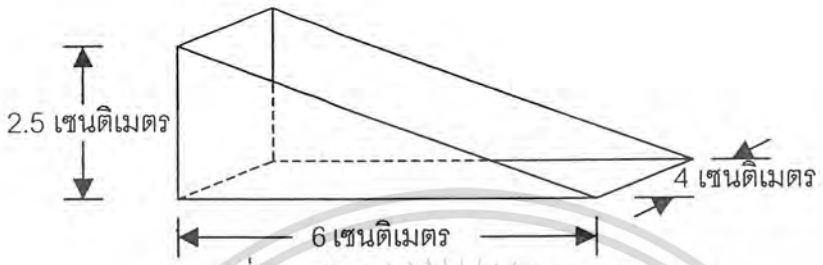
เพราะฉะนั้นปริมาตรน้ำในถังจะเท่ากับ 8.107 มิลลิลิตรดังนั้นในการออกแบบคานกระดกจะต้องออกแบบให้มีปริมาตรในการรับน้ำมากกว่า 8.107 มิลลิลิตร ในการคำนวณปริมาตรในการรับน้ำของคานกระดกนั้นจะคำนวณเพียงแค่ด้านเดียว เนื่องจากคานกระดกนั้นสมมาตรกันทั้งสองด้าน โดยในโครงงานนี้ได้มีการออกแบบคานกระดกดังต่อไปนี้



รูปที่ 2. 2 ขนาดของคานกระดก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การคำนวณหาปริมาตรในการรับน้ำฝนของคานกระดก



รูปที่ 2.3 คำนวณปริมาตรรวมของคานกระดก

ฐาน = 6 เซนติเมตร , สูง = 2.5 เซนติเมตร ,
หนา = 4 เซนติเมตร

ปริมาตรทรงสามเหลี่ยมมุมฉาก = 1/2 * ฐาน * สูง * หนา

$$= \frac{1/2 * 6 * 2.5 * 4}{3}$$

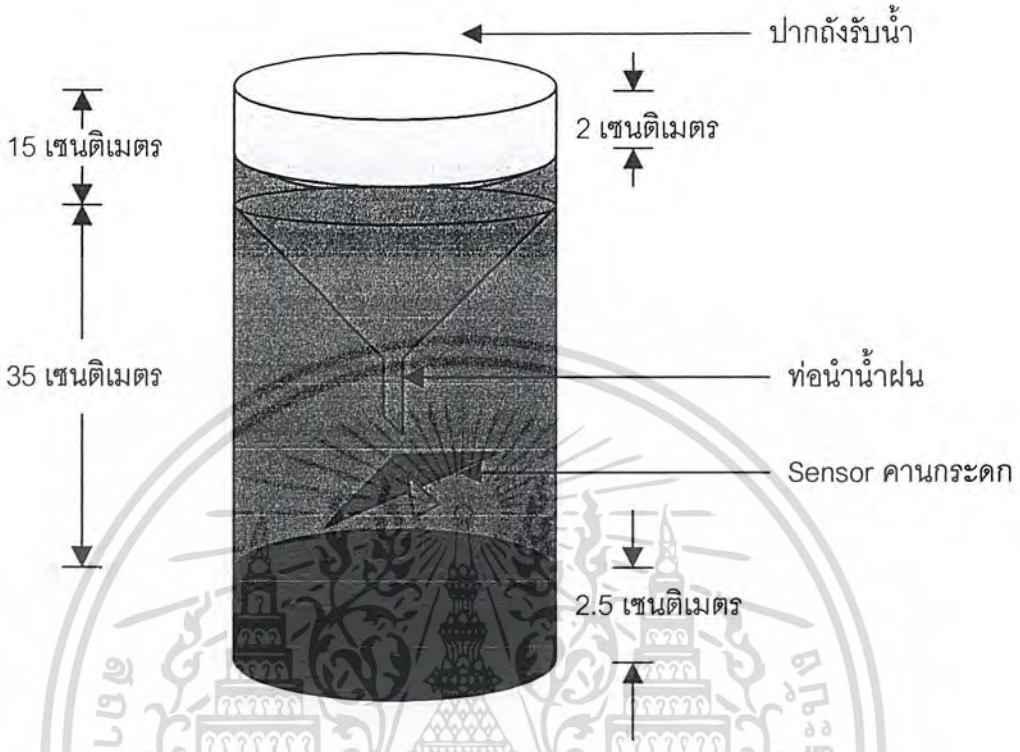
= 30 เซนติเมตร

เพราะฉะนั้นคานกระดกสามารถรับน้ำได้สูงสุด 30 มิลลิลิตร ซึ่งมากพอที่จะรับน้ำฝนที่ความสูง 0.25 มิลลิเมตร ที่ปากถังรับน้ำฝนกว้าง 8 นิ้วได้

2.6 หลักการทำงานของSensor คานกระดก

การทำงานของ Sensor คานกระดก เริ่มจากมีฝนตกน้ำฝนจะตกลงสูปากถังรับน้ำจากนั้นน้ำฝนจะไหลลงสู่ท่อนำน้ำฝนลงไปสูตัวSensor คานกระดก

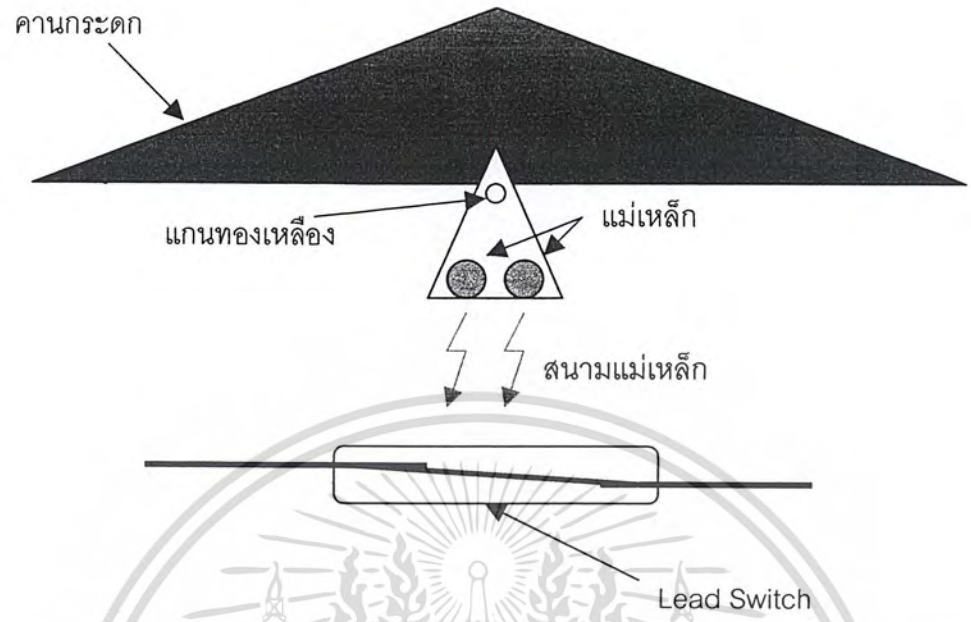
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2. 4 ขนาดของถังรองรับน้ำฝน

เมื่อSensor คานกระดกได้รับน้ำฝนจนเต็มถึงค่าที่ปรับตั้งไว้แล้ว คานกระดกจะกระดกเทน้ำทิ้งไปทำให้ แม่เหล็กซึ่งยึดติดอยู่กับตัวคานกระดกเคลื่อนที่ผ่านหน้า Lead Switch ซึ่งจะทำให้ Lead Switch เกิดสภาวะ On ขึ้นมา และเมื่อแม่เหล็กเคลื่อนที่ผ่าน Lead Switch ไปแล้วก็จะทำให้ Lead Switch กลับมา Off อีกครั้งหนึ่ง ซึ่งการเปลี่ยนสถานะของ Lead Switch นี้จะถูกส่งไปให้ยังไมโครคอนโทรลเลอร์เป็นตัวเก็บค่าการกระดกนี้ไว้ เพื่อทำการรวบรวมค่า และส่งผ่านไปเก็บในฐานข้อมูลในเครื่องคอมพิวเตอร์ต่อไป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.5 ส่วนประกอบของคานกระดก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ใบรับรอง

หนังสือฉบับนี้ขอรับรองว่า โครงงานเรื่อง เครื่อง วัดปริมาณน้ำฝน ที่นักศึกษา สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง คณะ วิศวกรรมศาสตร์ ได้มาทำการ ศึกษา สร้างและเก็บข้อมูลเกี่ยวกับตัว Sensor ที่ใช้ในการวัดปริมาณน้ำฝน มีนักศึกษาจำนวน 3 คน ซึ่งมีรายนามดังต่อไปนี้

1. นายเจริญจิตต์ บัวชุม
2. นายพรชัย ตันตโรภาส
3. นายสมเกียรติ เกียรติศิริกุล

โดยมีเจ้าหน้าที่ของกรมอุตุนิยมวิทยา กองช่างเครื่องมือ ฝ่ายมาตรฐานเครื่องมือ เป็นที่ปรึกษา ให้คำแนะนำ ควบคุม ดูแล

ปรากฏว่า เครื่องวัดปริมาณน้ำฝนเครื่องนี้ได้ถูกสร้างขึ้นตามทฤษฎีและหลักวิชาการ พร้อมทำการสอบเทียบผลการตรวจวัดซึ่งได้ค่าที่ถูกต้องอยู่ในเกณฑ์ที่ยอมรับได้ จึงสามารถนำไปใช้งานได้จริง

ลงชื่อ.....

(นายอำรุงฤทธิ์ นิ่มเสมอ)

ตำแหน่ง หัวหน้างานสอบเทียบเครื่องมือ

4 ตุลาคม 2544

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 3
วงจรควบคุม

3.1 คุณสมบัติของ AT89C52

1. เป็นไมโครคอนโทรลเลอร์ที่ใช้ซีพียูขนาด 8 บิต
2. ภายในมีหน่วยความจำโปรแกรมเป็นแบบแฟลชสามารถลบและเขียนใหม่ได้ประมาณ1000 ครั้ง
3. มีหน่วยความจำชั่วคราว (RAM) 256 ไบต์ 8 บิต
4. ขาพอร์ตเป็นแบบสองทิศทางสามารถใช้งานเป็นได้ทั้งอินพุตและเอาต์พุต
5. มีการสื่อสารอนุกรมแบบฟูลดูเพล็กซ์
6. มีวงจรกำเนิดสัญญาณนาฬิกาอยู่ภายในชิป

3.2 การรับส่งข้อมูลอนุกรม(UART)

พอร์ตสื่อสารอนุกรมของ mcs-51 มีโครงสร้างในการทำงานในแบบฟูลดูเพล็กซ์(Full Duplex)ซึ่งจะสามารถรับและส่งข้อมูลอนุกรมได้ในเวลาเดียวกันที่ด้านส่งใช้ขา TxD (พอร์ต 3.1) ที่ด้านรับใช้ขา RxD (พอร์ต 3.0) และใช้ SBUF เป็นบัฟเฟอร์สำหรับรับและส่งข้อมูลอนุกรม

พอร์ตสื่อสารอนุกรมของ mcs-51 มีโหมดในการทำงานอยู่ด้วยกันหลายโหมด โดยการกำหนดสถานะของบิต SM0 และ SM1 ซึ่งอยู่ในรีจิสเตอร์ควบคุม SCON ในที่นี้กำหนดให้ทำงานในโหมด 1 ด้วยอัตราบอดเรต 9,600 ใช้ความถี่คริสตอลค่า 11,059 เมกะเฮิร์ต และใช้วงจร Timer 1 ทำงานในโหมด 1 ด้วยค่า FDH อัตราบอดเรตในโหมดนี้คำนวณได้จากสูตร

$$\text{Baud Rate Mode 1} = \frac{2^{SMOD} \times \text{Oscillator Freq.}}{32 \times 12\left[256-(TH1)\right]}$$

แทนค่า

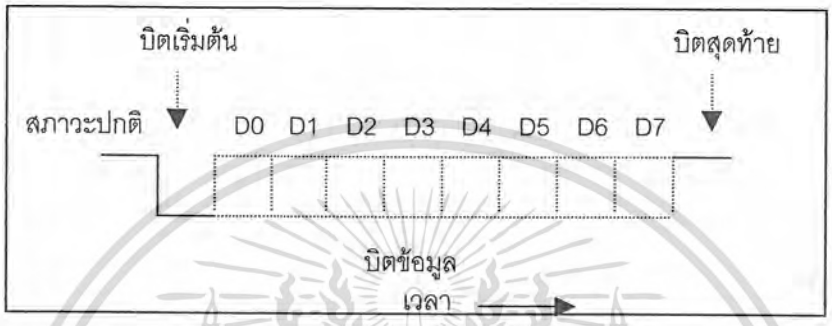
FDH=253 ที่ TH1

Baud Rate Mode 1 = $2^0(11.059*10^6)/(32*12*(256-253))$

= 9,600

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การเลือกโหมดการสื่อสารพอร์ตอนุกรมสามารถกำหนดได้จากรีจิสเตอร์ SCON ดังรูปที่ 3.1 และในที่นี้เลือกโหมดการสื่อสารอนุกรมโหมด 1 คือการสื่อสารข้อมูลอนุกรมจำนวน 10 บิต ประกอบไปด้วยบิตเริ่มต้นจำนวน 1 บิต บิตข้อมูลจำนวน 8 บิต และบิตหยุดจำนวน 1 บิต ดังแสดงดังรูปที่ 3.1 ข้อมูลที่ถูกส่งออกไปเป็นบิตแรกคือบิตที่มีนัยสำคัญต่ำสุด



รูปที่ 3.1 รูปแบบของสัญญาณข้อมูลอนุกรมในโหมด 1 ที่ใช้ข้อมูล 8 บิต บิตเริ่มต้นและบิตสุดท้ายอย่างละ 1 บิต

เมื่อได้ทำการเขียนหรือโอนย้ายข้อมูลเพื่อส่งออกไปทางพอร์ตสื่อสารอนุกรม เราจะต้องเขียนหรือโอนย้ายข้อมูลไปให้กับรีจิสเตอร์ SBUF โดยจะต้องทำการตรวจสอบค่าของแฟล็กสถานะ TI ภายในรีจิสเตอร์ SCON ซึ่งจะบอกว่าได้ทำการส่งข้อมูลเสร็จสิ้นหรือยัง และเมื่อต้องการจะรับข้อมูลก็จะต้องทำการตรวจสอบค่าของแฟล็กสถานะ RI ภายในรีจิสเตอร์ SCON ซึ่งจะบอกว่าได้ทำการรับข้อมูลเสร็จสิ้นหรือยัง

ถ้าหากมีสถานะดังต่อไปนี้จะเกิดการย้ายข้อมูลไปเก็บยังรีจิสเตอร์ SBUF

1. แฟล็ก TI เป็น 1 (แสดงว่าได้มีการส่งข้อมูลครบ 1 ไบต์แล้ว) พร้อมรับข้อมูลไบต์ถัดไป
2. แฟล็ก TI เป็น 0 (แสดงว่ายังทำการส่งข้อมูลยังไม่เสร็จสิ้น) ยังไม่พร้อมรับข้อมูลไบต์ถัดไป ถ้าหากทำการส่งข้อมูลไบต์ถัดไปในสภาวะนี้จะทำให้ข้อมูลที่ส่งออกไปเกิดการผิดพลาดขึ้น

ถ้าหากมีสถานะดังต่อไปนี้จะเกิดการย้ายข้อมูลจากรีจิสเตอร์ SBUF ออกมา

1. แฟล็ก RI เป็น 1 (แสดงว่าได้มีการรับข้อมูลครบ 1 ไบต์แล้ว) พร้อมรับข้อมูลไบต์ถัดไป
2. แฟล็ก RI เป็น 0 (แสดงว่ายังทำการรับข้อมูลไม่เสร็จสิ้น) ยังไม่พร้อมรับข้อมูลไบต์ถัดไป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ชื่อบิต: SCON ตำแหน่ง: 89H ค่าบิตเริ่มต้น: 0000 0000

SM0	SM1	SM3	REN	TB8	RB8	TI	RI

ชื่อบิต	ตำแหน่ง	ความหมาย
SM0	SCON.7	บิตเลือกโหมดการทำงาน
SM1	SCON.6	บิตเลือกโหมดการทำงาน
SM2	SCON.5	แฟล็กกำหนดการทำงานแบบมัลติโปรเซสเซอร์
REN	SCON.4	แฟล็กยอมรับให้มีการรับข้อมูล
TB8	SCON.3	ค่าของบิตที่ 9 สำหรับการส่งข้อมูลออก
RB8	SCON.2	ค่าของบิตที่ 9 ที่รับข้อมูลเข้า
TI	SCON.1	แฟล็กแสดงการอินเตอร์รัปต์ภายหลังจากการส่งข้อมูล
RI	SCON.0	แฟล็กแสดงการอินเตอร์รัปต์เมื่อมีการรับข้อมูลเข้า

รูปที่ 3.2 รีจิสเตอร์ควบคุมการทำงานและบอกสถานะการสื่อสารข้อมูลอนุกรม SCON

ในโปรแกรมยอมให้มีการอินเตอร์รัพเกิดขึ้นได้จากพอร์ตอนุกรม โดยการกำหนดค่าให้กับรีจิสเตอร์ IE ให้เท่ากับ 90H

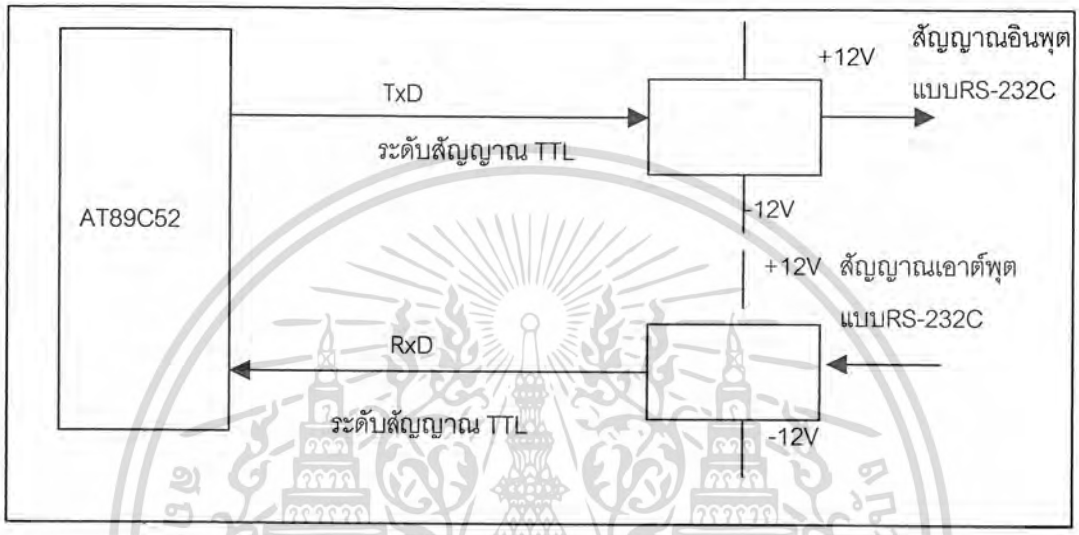
3.3 การเชื่อมต่อแบบมาตรฐาน RS-232C

ในการเชื่อมต่อแบบอนุกรมเข้ากับอุปกรณ์คอมพิวเตอร์ต่างๆ เช่น คอมพิวเตอร์ส่วนบุคคล เทเล็กซ์ หรือ โทรพิมพ์ เป็นต้น มักจะกำหนดใช้การเชื่อมต่อตามมาตรฐาน RS-232C เช่นเดียวกัน เครื่องวัดปริมาณน้ำฝนนี้ก็ใช้มาตรฐาน RS-232C ในการเชื่อมต่อกับคอมพิวเตอร์ส่วนบุคคล เพื่อเก็บข้อมูลและแสดงผลสถิติน้ำฝน

มาตรฐาน RS-232C มีเพื่อลดปัญหาการเข้ากันไม่ได้ระหว่างสัญญาณของอุปกรณ์ที่มาเชื่อมต่อกันทั้งสองด้านให้น้อยลง เนื่องจากระดับโวลเตจที่ใช้และการแทนความหมายของระดับลอจิกตามมาตรฐานนี้แตกต่างไปจากที่ใช้งานกันในระบบดิจิตอลทั่วไป โดยระดับสัญญาณของ RS-232C เป็นแบบไบโพลาร์ (Bipolar)ระดับโวลท์เตจทางด้านลบช่วง -3V ถึง -20V แทนค่าลอจิก 1 และโวลท์เตจทางด้านบวกช่วง +3V ถึง +20V แทนค่าลอจิก 0 ดังนั้นจะเห็นได้ว่ามีความจำเป็นจะต้องเพิ่มอุปกรณ์หรือวงจรพิเศษเข้าไป เพื่อเปลี่ยนระดับโวลท์เตจจากระบบ 0V

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ถึง +5V จากขาสัญญาณของ AT89C52 เป็นระดับโวลท์เตจที่สูงกว่าค่า +3V หรือต่ำกว่า -3V ดังในรูปที่ 3.3 ซึ่งแสดงให้เห็นว่าระดับสัญญาณแบบ TTL จากขาสัญญาณ TxD และ RxD ของ AT89C52 จะต้องถูกปรับเปลี่ยนไปเป็นระดับสัญญาณ RS-232C ก่อนที่จะทำการส่งออกไปในสายนำสัญญาณต่อไป



รูปที่ 3.3 แนวทางการเปลี่ยนแปลงระดับสัญญาณTTL ไมโครคอนโทรลเลอร์ AT89C52 ไปเป็นระดับสัญญาณแบบ RS-232C และการแปลงระดับสัญญาณอินพุตแบบ RS-232C ไปเป็นระดับสัญญาณแบบ TTLก่อนที่จะได้เชื่อมต่อกับขาสัญญาณของไมโครคอนโทรลเลอร์ AT89C52

ซึ่งในวงจรเครื่องวัดปริมาณน้ำฝนนี้ใช้ไอซีเบอร์ MAX232 ในการแปลงระดับสัญญาณดังกล่าว

3.4 การใช้งานไอซีสร้างฐานเวลาจริงหรือรีลไทม์คล็อก (RTC)

ไอซีสร้างฐานเวลาจริงให้แก่ระบบไมโครคอนโทรลเลอร์เราเลือกใช้เบอร์ DS1307 มันสามารถให้ข้อมูลเกี่ยวกับเวลาทั้งหมด ไม่ว่าจะเป็น วินาที, นาที, ชั่วโมง, วันที่(date), วันในสัปดาห์(day), เดือน และปี โดยสามารถปรับวันเดือนปีให้ตรงตามปฏิทินได้อย่างถูกต้อง ซึ่งในการทำงานของเครื่องวัดปริมาณน้ำฝนนี้มีความจำเป็นอย่างยิ่ง เพราะว่าในการบันทึกค่าปริมาณน้ำฝนที่วัดได้จะนำเวลามาเป็นตัวอ้างอิง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.4.1 คุณสมบัติของไอซี RTC เบอร์ DS 1307

1. เป็นไอซีรีลไทม์คล็อกให้ข้อมูลตั้งแต่วินาที นาที ชั่วโมง วันที่ของเดือน เดือน ปี รวมทั้งการปรับวันในปีอธิกสุรทินด้วยสามารถให้ข้อมูลเวลาได้อย่างเที่ยงตรงถึงปีคริสตศักราช 2100
2. มีหน่วยความจำนอนโวลาไทม์แรมจำนวน 56 ไบต์อยู่ภายใน สามารถใช้เก็บข้อมูลทั่วไปได้
3. ใช้การเชื่อมต่อแบบระบบบัส I^2C
4. มีวงจรตรวจจับไฟเลี้ยงต่ำหรือหายไปอย่างอัตโนมัติ และสามารถรักษาข้อมูลเวลาไว้ได้แม้ไม่มีไฟเลี้ยงไอซี

3.4.2 การทำงานของ DS 1307

ไอซี DS 1307 จัดการเชื่อมต่อในแบบบัส I^2C โดยจะทำงานเป็นอุปกรณ์สเลฟเสมอ เมื่อจะติดต่อจะต้องกำหนดรูปแบบตามที่กำหนดไว้ในการติดต่อแบบ I^2C ซึ่งวงจรออสซิลเลเตอร์ถือเป็นหัวใจหลักของไอซีเพราะเป็นจุดเริ่มต้นในการสร้างข้อมูลฐานเวลาจริงในขณะที่DS1307 ทำงานที่ขา SQW/OUT จะมีสัญญาณพัลส์สี่เหลี่ยมส่งออกมาตลอดเวลาถ้ามีการอีนาเบิลวงจรกำเนิดสัญญาณพัลส์ที่รีจิสเตอร์ควบคุมค่าความถี่นี้สามารถเลือกได้ 4 ค่าคือ 1 HZ, 4.096 kHZ, 8.192 kHZ, และ 32 kHZ

นอกจากนี้ยังมีวงจรควบคุมพลังงานไฟฟ้าซึ่งจะคอยตรวจสอบสภาวะไฟเลี้ยงของไอซี หากไฟเลี้ยงต่ำกว่า $1.25 \times V_{BAT}$ ก็จะควบคุมให้ DS 1307 หยุดการทำงานรีเซตค่าตัวนับแอดเดรสภายในทำให้ไม่สามารถติดต่อกับ DS 1307ได้ดังนั้นในการใช้งานจะต้องระมัดระวังอย่าให้ไฟเลี้ยงต่ำกว่า $1.25 \times V_{BAT}$ หรือประมาณ 3.75V ในกรณีที่ใช้ V_{BAT} เท่ากับ 3V ถ้าหากไฟเลี้ยงมีค่าต่ำกว่า V_{BAT} ไอซี DS 1307 จะเข้าสู่โหมดสำรองข้อมูลกระแสต่ำทันที จะไม่มีการส่งสัญญาณพัลส์ออกมาที่ขา SQW/OUT แต่วงจรสร้างฐานเวลายังคงทำงานอยู่เพื่อให้ค่าของเวลาเดินไปอย่างไม่ผิดพลาด

3.4.3 การต่อร่วมกับ AT89C52

ขา 5 – เป็นขา I/O ต่อกับขา P1.0 ใช้ในการส่ง Address Command และรับ-ส่ง Data เป็นการรับ-ส่งแบบอนุกรม

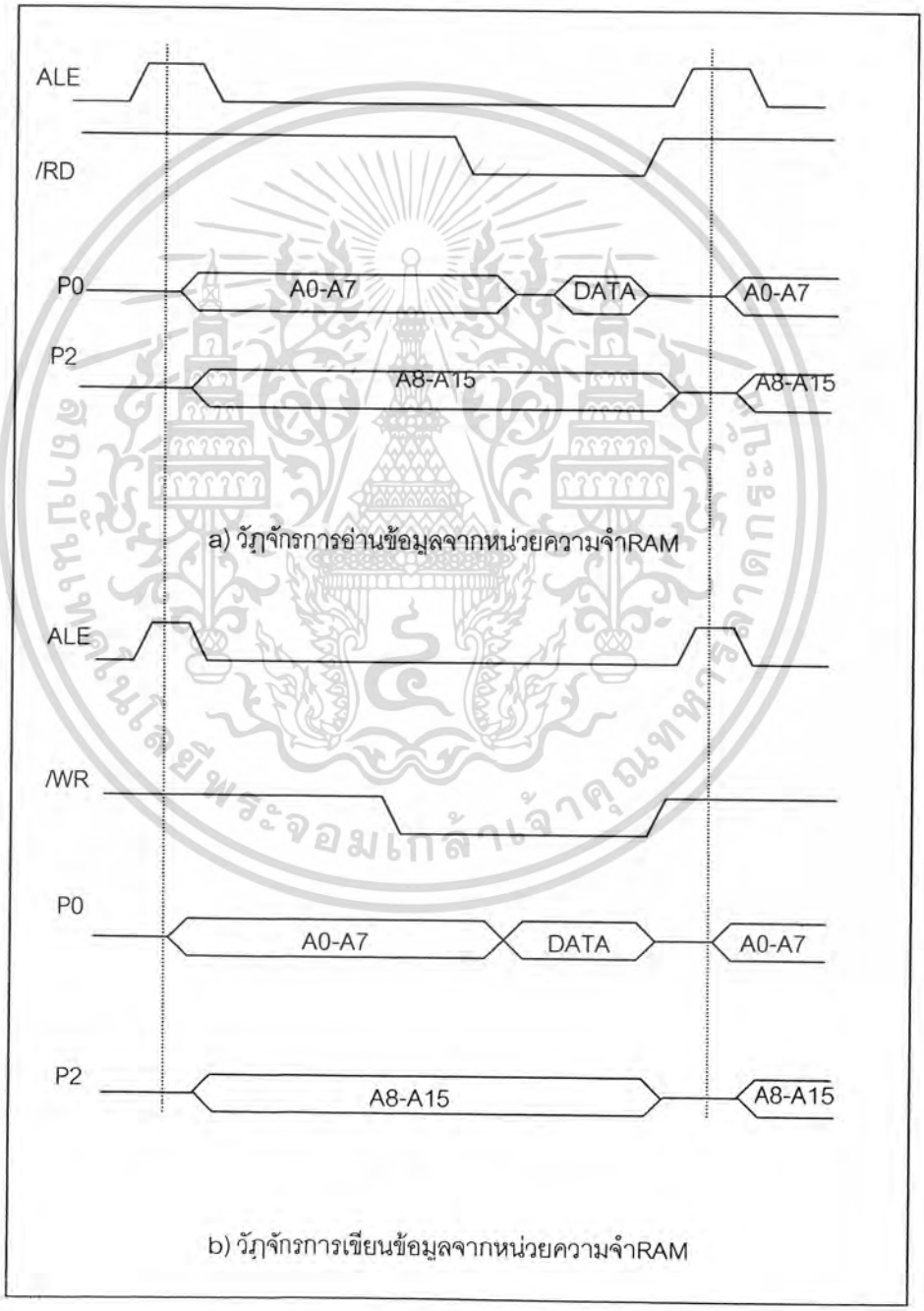
ขา 6 – เป็นขา CLK (Clock) ต่อกับขา P1.1 ใช้ในการกำหนดจังหวะในการรับส่งข้อมูลกับ DS 1307 ซึ่งจะกำหนดได้จากโปรแกรม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สำนักหอสมุดกลาง พระจอมเกล้าลาดกระบัง

3.5 การเชื่อมต่อกับหน่วยความจำข้อมูลภายนอก

เราใช้หน่วยความจำข้อมูลชั่วคราวสำหรับเก็บข้อมูลปริมาณน้ำฝน ไว้ในกรณีที่ยังไม่มีการติดต่อกับคอมพิวเตอร์ ในที่นี้จะใช้หน่วยความจำชั่วคราวภายนอก (RAM) ความจุ 32 Kbyte ข้อมูลที่เก็บจะไม่สูญหายขณะไม่มีไฟจ่ายให้ระบบเนื่องจากได้มีการใช้แบตเตอรี่สำรองช่วยจ่ายไฟให้กับระบบแทน



a) วัฏจักรการอ่านข้อมูลจากหน่วยความจำRAM

b) วัฏจักรการเขียนข้อมูลจากหน่วยความจำRAM

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

42250

รูปที่ 3.4 ผังเวลาแสดงถึงลำดับของสัญญาณเมื่อมีการอ่านข้อมูลจากหน่วยความจำชั่วคราว (RAM) ดังนี้

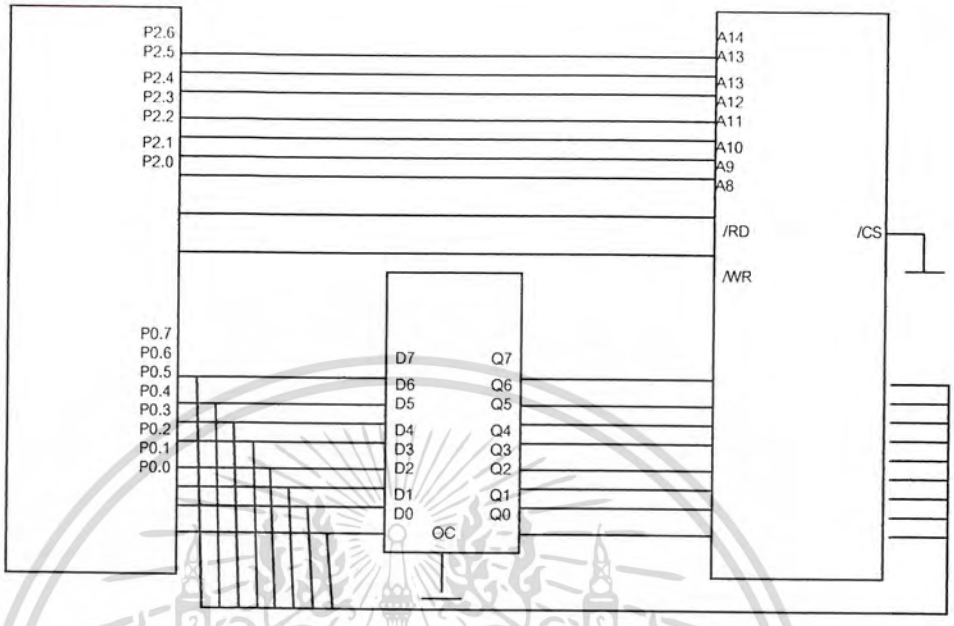
1. ไมโครคอนโทรลเลอร์จะเริ่มจาก การส่งค่าแอดเดรสของหน่วยความจำที่ต้องการออกมาทางบัสแอดเดรส
2. สัญญาณ OE จะต้องถูกเปลี่ยนให้เป็นระดับลอจิกต่ำ เพื่อระบุว่าต้องการอ่านข้อมูลจากหน่วยความจำ
3. หลังจากนั้นไมโครคอนโทรลเลอร์จะต้องหยุดรอในช่วงระยะเวลาหนึ่งเรียกว่าRead Access Time เพื่อให้วงจรภายในหน่วยความจำชั่วคราว(RAM)ทำการถอดรหัสแอดเดรสและอ่านข้อมูล
4. หลังจากสิ้นสุดเวลาในข้อ 3 แล้ว ข้อมูลจะถูกส่งออกมาทางขาสัญญาณของบัสข้อมูล และไมโครคอลโทลเลอร์สามารถอ่านข้อมูลนี้ไปประมวลผลต่อไป

จากรูปที่3.4 (b) เป็นผังเวลาแสดงถึงลำดับของสัญญาณเมื่อมีการเขียนข้อมูลลงในหน่วยความจำชั่วคราว (RAM)

1. ไมโครคอนโทรลเลอร์จะต้องเริ่มต้นด้วยการส่งค่าแอดเดรสของหน่วยความจำที่ต้องการออกมาทางบัสแอดเดรส
2. ลำดับถัดไปในบัสข้อมูลจะเป็นข้อมูลที่ต้องการจะเขียนลงในหน่วยความจำ
3. หลังจากนั้นไมโครคอนโทรลเลอร์จะต้องหยุดรอในช่วงระยะเวลาหนึ่ง เรียกว่า Write Access Timeเพื่อให้หน่วยความจำชั่วคราว (RAM) ทำการถอดรหัสแอดเดรสและเตรียมการเขียนข้อมูล
4. เมื่อสิ้นสุดเวลาที่หยุดรอไมโครคอนโทรลเลอร์จะต้องขับสัญญาณWR ให้เป็นระดับลอจิกต่ำเพื่อให้ข้อมูลภายในบัสข้อมูลถูกนำไปยังตำแหน่งที่ต้องการ ภายในหน่วยความจำชั่วคราว (RAM) ต่อไป

จากรูปที่ 3.5 เราใช้ไอซีเบอร์ 74LS737 ทำหน้าที่ในการแลตซ์ (Latch) หรือค้าง ค่าแอดเดรสด้านต่ำให้กับอินพุทของหน่วยความจำชั่วคราว (RAM) ซึ่งมีอยู่เพียงตัวเดียวสำหรับขาสัญญาณ OE (หรือ RD) จะต่อเข้ากับขาสัญญาณ RD (P3.7) และขาสัญญาณ WR (P3.6) ของ AT89C52 และเนื่องจากว่าในวงจรนี้มีไอซีหน่วยความจำเพียงตัวเดียว จึงได้ทำการต่อสัญญาณ CS เข้ากับกราวด์โดยตรงเพื่อเป็นการเลือกใช้ให้ไอซีทำงานตลอด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.5 การเชื่อมต่อหน่วยความจำข้อมูลภายนอกกับ AT89C52

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 4
การออกแบบโปรแกรม

4.1 การออกแบบโปรแกรมด้วยVisual Basic 6

Visual Basic เป็นภาษาคอมพิวเตอร์ที่ได้รับความนิยมนำมาใช้ในการพัฒนาโปรแกรมบน Windows เนื่องจาก เป็นภาษาคอมพิวเตอร์ที่ใช้เทคโนโลยีในลักษณะ Visualize ซึ่งเพียงแต่เลือก Control ที่เหมาะสม แล้ววาดลงบน Form ก็สามารถสร้างจอภาพสำหรับติดต่อกับผู้ใช้ รวมทั้งการใช้เทคนิคการเขียนโปรแกรมแบบ Event-driven ซึ่งเป็นการเขียนโปรแกรมเพื่อกำหนดขั้นตอนการทำงานให้กับ Control ต่างๆ ที่สร้างขึ้งตามเหตุการณ์ (Event) ต่างๆที่เกิดขึ้น เช่น การเลื่อนเมาส์หรือการรับข้อมูลจากคีย์บอร์ดฯลฯเป็นต้น ประกอบกับภาษาที่ใช้เขียนโปรแกรมเป็นภาษา BASIC ซึ่งเป็นภาษาคอมพิวเตอร์ที่ผู้ใช้บนคอมพิวเตอร์ส่วนบุคคลส่วนใหญ่คุ้นเคยทำให้การพัฒนาโปรแกรมบน Windows ด้วย Visual Basic มีขั้นตอนน้อย กระทำได้ง่าย และสะดวกต่อการใช้งาน จึงทำให้การพัฒนาโปรแกรมสามารถทำได้อย่างรวดเร็ว

ในโครงงานนี้ได้แบ่งโปรแกรมที่พัฒนาขึ้นด้วย Visual Basic 6 ออกเป็น 2 ส่วนคือ

1. โปรแกรมส่วนที่ติดตั้งกับ Server Computer
2. โปรแกรมส่วนที่ติดตั้งกับ Client Computer

4.1.1 โปรแกรมส่วนที่ติดตั้งกับ Server Computer

โปรแกรมส่วนนี้จะมีการติดต่อกับไมโครคอนโทรลเลอร์ และฐานข้อมูลภายใน Server Computer โดยการทำงานจะเริ่มจากเมื่อไมโครคอนโทรลเลอร์เช็คค่าจากReal Time Clock แล้วพบว่าเวลาอยู่ในช่วงเวลาระหว่าง 7.00 – 7.20 น. ไมโครคอนโทรลเลอร์จะทำการส่งสัญญาณไปกระตุ้นให้เครื่อง Server คอมพิวเตอร์ทำการบูทระบบขึ้นมา โดยเมื่อเครื่องคอมพิวเตอร์บูทระบบขึ้นมาแล้วจะทำการเรียกโปรแกรมในส่วนที่ติดตั้งไว้ใน Server Computer ขึ้นมา เมื่อโปรแกรมในส่วนของServerทำงานจะทำการเช็คค่าจากไมโครคอนโทรลเลอร์เพื่อยืนยันเวลาอีกครั้งว่าเป็นเวลาอยู่ในช่วง 7.00 – 7.20 น. จากนั้นจะมีการถ่ายโอนข้อมูลปริมาณน้ำฝนที่เก็บไว้จากไมโครคอนโทรลเลอร์ ส่งไปเก็บยังฐานข้อมูลในเครื่องคอมพิวเตอร์ โดยมีโปรแกรมในส่วนของServer Computer เป็นตัวจัดการ โดยในการทำงานระหว่างโปรแกรมใน Server Computer และไมโครคอนโทรลเลอร์ จะทำงานร่วมกันโดยอัตโนมัติ โดยไม่จำเป็นต้องการรับอินพุตใดๆจากผู้ใช้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

และเมื่อทำการถ่ายโอนข้อมูลเสร็จแล้ว เครื่องคอมพิวเตอร์Server จะทำการปิดตัวเองลงหลังจากเริ่มการทำงานแล้ว 20 นาที

4.1.2 โปรแกรมส่วนที่ติดตั้งกับ Client Computer

โปรแกรมส่วนนี้จะมีการทำงานร่วมกับฐานข้อมูล โดยจะมีการนำข้อมูลจากฐานข้อมูลมาแสดงผลในรูปแบบต่างๆทั้งที่เป็นกราฟ และเป็นตารางแสดงฐานข้อมูล โดยผู้ใช้สามารถเลือกใช้งานได้ตามต้องการ โดยจะได้กล่าวถึงการใช้งานโปรแกรมโดยละเอียดในภาคผนวกอีกครั้งหนึ่ง

รูปที่ 4.1 โปรแกรม Rain Gauge 2

4.2 การออกแบบฐานข้อมูลด้วย Microsoft Access

ในส่วนของฐานข้อมูลของโปรแกรมนั้นจะใช้ตารางฐานข้อมูลที่สร้างด้วยโปรแกรม Microsoft Access เพื่อใช้ทำงานร่วมกับโปรแกรม Visual Basic โดยในการออกแบบตาราง (Table) จะแบ่งการเก็บข้อมูลออกเป็นRecord โดยในแต่ละRecord นั้นจะประกอบไปด้วย Field ต่างๆ โดย Field แต่ละ Field คือ ข้อมูลประเภทเดียวกัน เช่น ในตาราง (Table) ชื่อ RainDatabase ซึ่งใช้เป็นตารางบันทึกฐานข้อมูลปริมาณน้ำฝน ซึ่งประกอบไปด้วย field ID ทำหน้าที่เก็บลำดับของข้อมูลทั้งหมด เพื่อใช้ในการตรวจสอบการอ่านข้อมูลในตารางว่าอ่านไปถึงRecordใดแล้ว,Field Dayใช้สำหรับเก็บวันที่ที่ทำการบันทึกข้อมูลที่ได้รับจากไมโครคอนโทรลเลอร์ ซึ่งไมโครคอนโทรลเลอร์จะส่งข้อมูลมาบันทึกในทุกๆวัน , Field Month ใช้สำหรับเก็บข้อมูลเดือน ,

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Field Year ใช้สำหรับเก็บข้อมูลปี , Field Time ใช้สำหรับเก็บช่วงเวลาในแต่ละวันโดยมีการแบ่งช่วงเวลาในแต่ละวันออกเป็น24ช่วงเวลาดังนี้ 07.00-08.00น.,08.00-09.00น.,09.00-10.00 น.,...,06.00-07.00 น. , Field RainAmount ใช้สำหรับเก็บข้อมูลปริมาณน้ำฝน

RainDatabase : ตาราง

ID	Day	Month	Year	Time	RainAmount
35	22	8	2001	07.00-08.00	6
36	22	8	2001	08.00-09.00	1
37	22	8	2001	09.00-10.00	2
38	22	8	2001	10.00-11.00	4
43	22	8	2001	11.00-12.00	1
44	22	8	2001	12.00-13.00	0
45	22	8	2001	13.00-14.00	0
46	22	8	2001	14.00-15.00	0
47	22	8	2001	15.00-16.00	0
48	22	8	2001	16.00-17.00	125
49	22	8	2001	17.00-18.00	64
51	22	8	2001	18.00-19.00	31
52	22	8	2001	19.00-20.00	12
53	22	8	2001	20.00-21.00	6

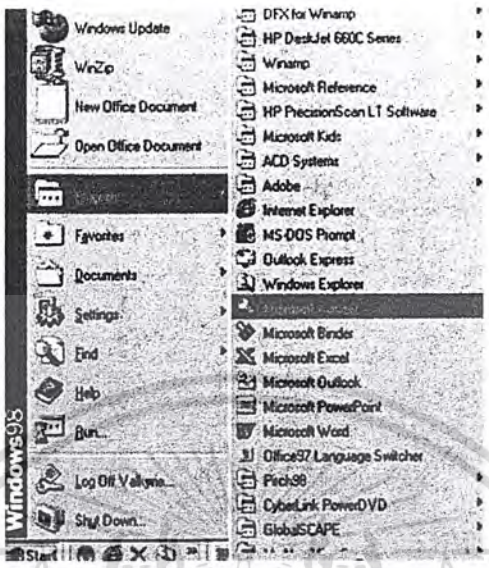
ระเบียน: 1 จาก 71

รูปที่ 4.2 ตารางฐานข้อมูลการเก็บน้ำฝน

4.2.1 วิธีการสร้างตาราง(Table)

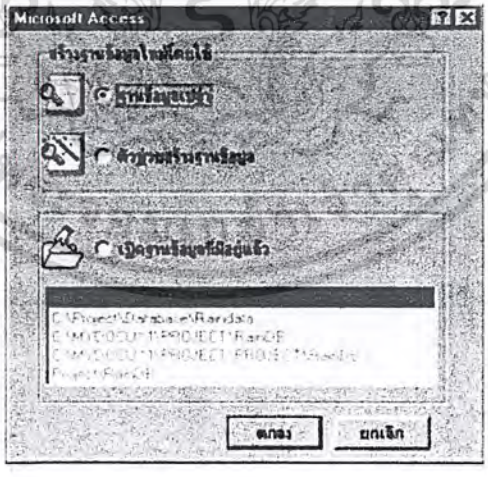
1. เปิดโปรแกรม Microsoft Access

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.3 การเรียกใช้โปรแกรม Access

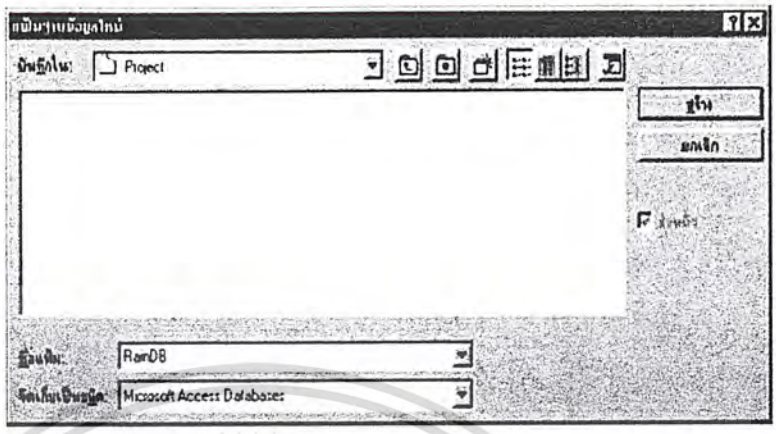
2. สร้างฐานข้อมูลใหม่ โดยเลือกเป็น Blank Database ซึ่งจะเป็นฐานข้อมูลเปล่าไม่มี Table



รูปที่ 4.4 การสร้างฐานข้อมูลเปล่า

ใส่ชื่อฐานข้อมูลลงไปโดยตั้งชื่อให้มีนามสกุลเป็น.mdb

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



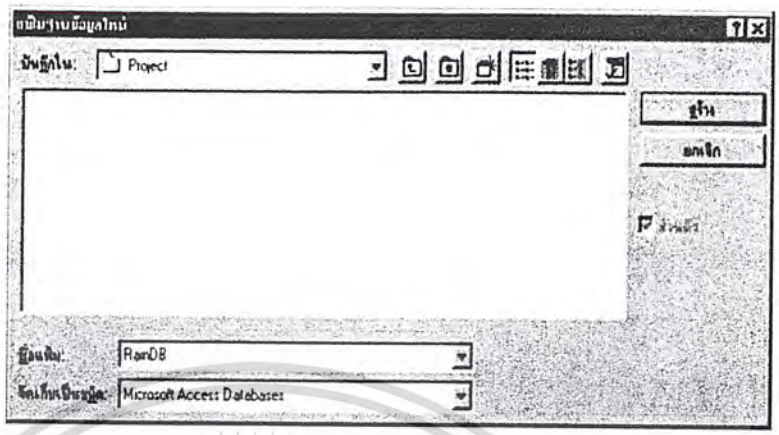
รูปที่ 4.5 การบันทึก ฐานข้อมูล และการตั้งชื่อไฟล์

3. เมื่อได้ส่วนประกอบบนหน้าจอดังรูปให้คลิกที่แถบด้านบนที่ชื่อ ตาราง แล้วคลิกสร้างใหม่

รูปที่ 4.6 การเลือกเข้าไปออกแบบฐานข้อมูลเปล่า

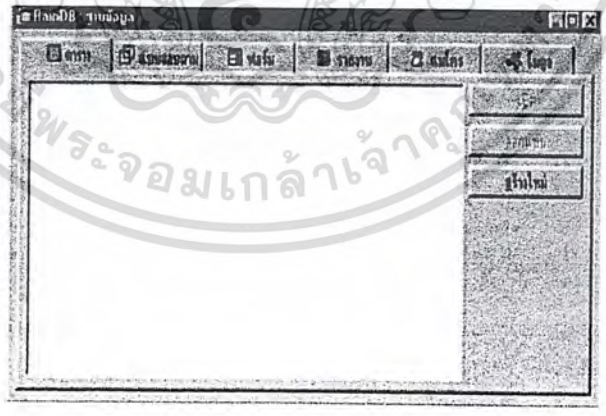
จะปรากฏดังรูป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.7 การบันทึก ฐานข้อมูล และการตั้งชื่อไฟล์

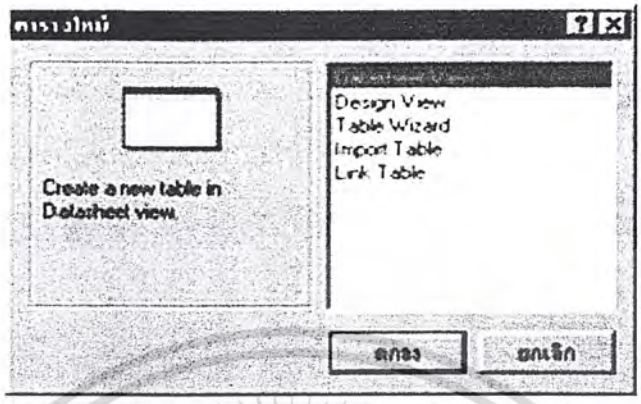
4. เมื่อได้ส่วนประกอบบนหน้าจอดังรูปให้คลิกที่แถบด้านบนที่ชื่อ ตาราง แล้วคลิกสร้างใหม่



รูปที่ 4.8 การเลือกเข้าไปออกแบบฐานข้อมูลเปล่า

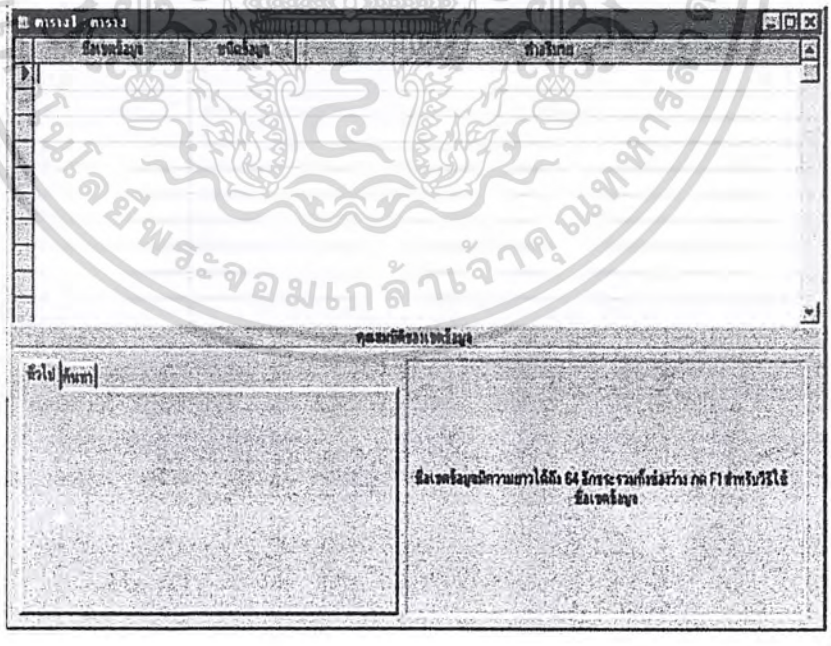
จะปรากฏดังรูป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.9 การเลือกทำตารางใหม่

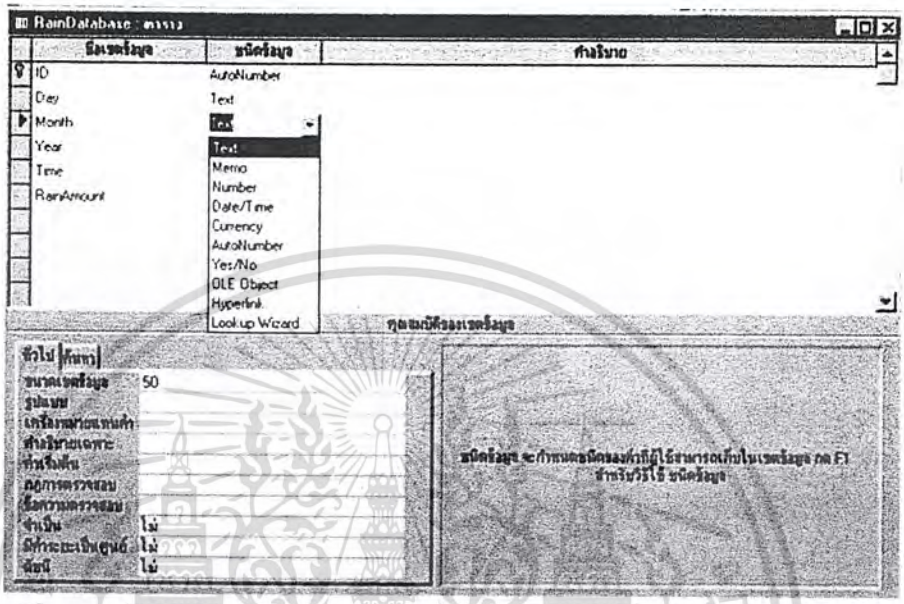
5. เลือกที่ Design View แล้วกด ตกลง จะเข้าสู่หน้าต่างการออกแบบตารางดังรูปด้านล่าง



รูปที่ 4.10 หน้าต่างก่อนการออกแบบตาราง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

6. พิมพ์ชื่อเขตข้อมูล (Field Name) ลงไปในคอลัมน์ชื่อเขตข้อมูล ดังรูปด้านล่างนี้ แล้วจากนั้นก็กำหนดชนิดของข้อมูลให้สอดคล้องกับข้อมูลที่เราต้องการเก็บไว้ใน Field นั้นๆ



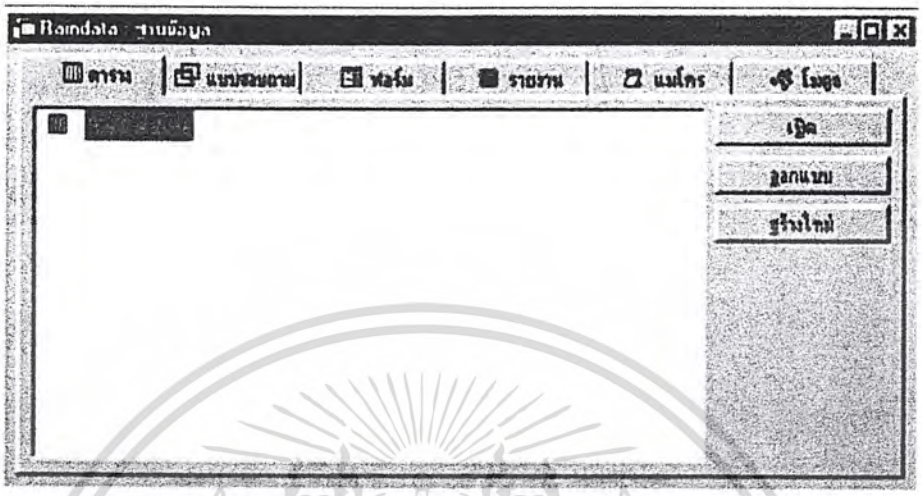
รูปที่ 4.11 การออกแบบตาราง

7. ปิดหน้าต่าง Design แล้วจะปรากฏ ไดอะล็อกบ๊อก ดังรูปด้านล่างนี้ซึ่งจะถามว่าจะทำการบันทึกตารางที่ทำไว้ในชื่อใด ให้ป้อนชื่อที่ต้องการสร้างแล้วตอบตกลง

รูปที่ 4.12 การบันทึก ตารางฐานข้อมูล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

8. จากนั้นจะได้ตารางที่ใช้งานดังรูป



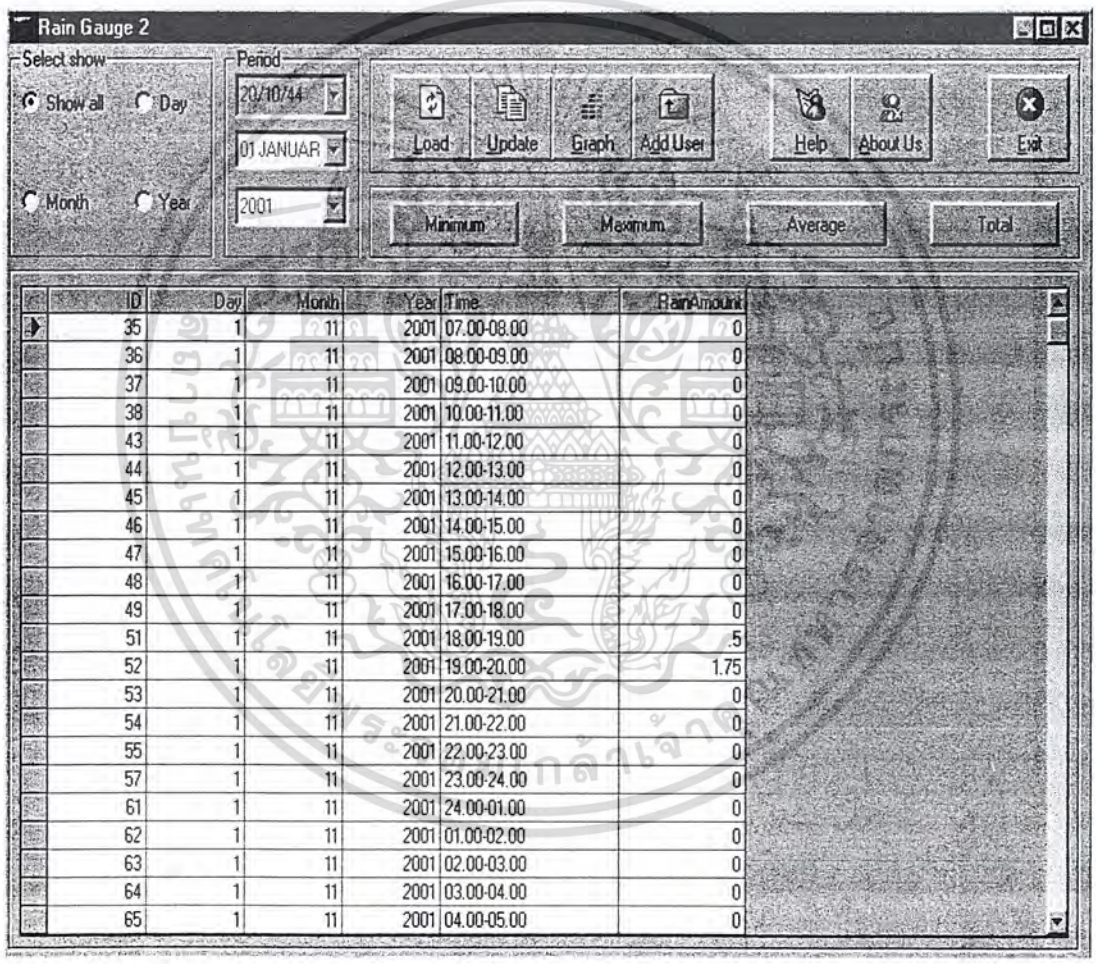
รูปที่ 4.13 หน้าต่างฐานข้อมูลที่สร้างเสร็จแล้ว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 5
ผลการทดลอง

5.1 ผลการทดลอง

ผลการทดลองเมื่อนำเครื่องวัดปริมาณน้ำฝนออกไปตั้งไว้กลางแจ้งเป็นระยะเวลา 1 วัน และเมื่อครบเวลา 1 วัน จึงได้ทำการเปิดฐานข้อมูล ผลที่ได้เป็นดังต่อไปนี้



ID	Day	Month	Year	Time	RainAmount
35	1	11	2001	07.00-08.00	0
36	1	11	2001	08.00-09.00	0
37	1	11	2001	09.00-10.00	0
38	1	11	2001	10.00-11.00	0
43	1	11	2001	11.00-12.00	0
44	1	11	2001	12.00-13.00	0
45	1	11	2001	13.00-14.00	0
46	1	11	2001	14.00-15.00	0
47	1	11	2001	15.00-16.00	0
48	1	11	2001	16.00-17.00	0
49	1	11	2001	17.00-18.00	0
51	1	11	2001	18.00-19.00	.5
52	1	11	2001	19.00-20.00	1.75
53	1	11	2001	20.00-21.00	0
54	1	11	2001	21.00-22.00	0
55	1	11	2001	22.00-23.00	0
57	1	11	2001	23.00-24.00	0
61	1	11	2001	24.00-01.00	0
62	1	11	2001	01.00-02.00	0
63	1	11	2001	02.00-03.00	0
64	1	11	2001	03.00-04.00	0
65	1	11	2001	04.00-05.00	0

รูปที่ 5.1 ผลการทดลองจากการวัดปริมาณน้ำฝน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5.2 ปัญหาที่เกิดขึ้น และแนวทางการแก้ไข

1. จากการทดลองสังเกตพบว่าเมื่อเวลาคานกระดก กระดกน้ำทิ้งไปแล้วจะไม่สามารถเทน้ำทิ้งได้หมดจด ยังคงมีน้ำบางส่วนติดค้างเกาะอยู่ที่ผิวของคานกระดก ซึ่งเป็นสาเหตุหนึ่งที่ทำให้เกิดค่าผิดพลาดในการวัดขึ้น ซึ่งแนวทางการแก้ไขสามารถทำได้โดยการขัดผิวของคานกระดกให้มีความมัน เพื่อลดแรงเสียดทานที่ผิวของคานกระดก
2. จากการทดลองสังเกตพบว่าเมื่อบางครั้งมีใบไม้หล่นจากต้นไม้เข้ามาในปากถังรับน้ำซึ่งเป็นสาเหตุให้ไปอุดตันทางเดินของน้ำทำให้เครื่องวัดปริมาณน้ำฝนไม่สามารถทำงานได้ แนวทางแก้ไขคือ ต้องหาที่โล่งแจ้งเพื่อติดตั้งเครื่องวัดปริมาณน้ำฝน เพื่อหลีกเลี่ยงปัญหาดังกล่าว

5.3 สรุปผล

จากการพัฒนาการโครงงานเครื่องวัดปริมาณน้ำฝน ได้ทำการสร้างตัวเครื่องวัดปริมาณน้ำฝนขึ้นมาใหม่อีก 1 ชุด โดยยังคงขนาดปากถังเดิมเอาไว้ที่เส้นผ่านศูนย์กลาง 8 นิ้ว เนื่องจากเป็นขนาดมาตรฐานที่เหมาะสมในการใช้วัดปริมาณน้ำฝน และได้มีการปรับปรุงรูปแบบของเครื่องวัดให้เหมาะสมต่อการใช้งานมากยิ่งขึ้นโดยลดขั้นตอนในการสร้างที่ยุ่งยากลงให้เหลือตัวถังเดี่ยว ไม่ต้องมีการสร้างถึงสองส่วนแล้วนำมาประกอบกันแบบเดิม และได้มีการใช้ฐานอลูมิเนี่ยมทึบซึ่งมีผลดีต่อการใช้งานคือป้องกันไม่ให้แมลง หรือสัตว์เล็กใดๆเข้าไปภายในเครื่องได้ ได้มีการเปลี่ยนมาใช้สแตนเลสมาทำเป็นตัวเครื่องเพื่อความแข็งแรงและความคงทน ในส่วนของตัววัดปริมาณน้ำฝนจากเดิมใช้หลักการไซฟอน ซึ่งการสร้างให้ทำงานได้สมบูรณ์ทำได้ยาก จึงได้เปลี่ยนมาใช้หลักการของคานกระดก ซึ่งมีข้อดีคือ

1. ง่ายต่อการออกแบบ
2. ราคาถูก
3. การซ่อมบำรุงทำได้ง่าย
4. มีความแม่นยำในการวัดที่ดีกว่าเมื่อเทียบกับของเดิม (ของเดิมค่าผิดพลาดประมาณ 10 มิลลิเมตร แต่สำหรับคานกระดกมีค่าผิดพลาดไม่เกิน 0.25 มิลลิเมตร)
5. การติดตั้งทำได้สมบูรณ์กว่าเนื่องจากมีน็อตปรับระนาบที่ฐานให้ได้ระนาบ
6. มีความทนทานกว่าเดิมเนื่องจากใช้สแตนเลสนำมาทำตัวถัง
7. ได้รับการรับรองการปรับแต่งแล้วจากงานสอบเทียบเครื่องมือ กรมอุตุนิยมวิทยา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ในส่วนโปรแกรมที่ใช้งานได้มีการเพิ่มการทำงานในส่วนของการเข้าถึงข้อมูลจากระยะไกล เพื่อให้ผู้ใช้สามารถใช้งานเครื่องวัดปริมาณน้ำฝนได้สะดวกขึ้น และมีการประหยัดพลังงานมากขึ้นโดยมีการเขียนโปรแกรมให้เครื่องคอมพิวเตอร์สามารถ Shut Down ตัวเองเมื่อไม่มีการใช้งานใดๆ หรือเมื่อเครื่องคอมพิวเตอร์ทำงานเสร็จตามหน้าที่ประจำวันแล้ว และจะมีการเปิดเครื่องคอมพิวเตอร์ที่ต่ออยู่กับเครื่องวัดปริมาณน้ำฝน เพียงแค่ช่วงที่มาการถ่ายโอนข้อมูลจากส่วนควบคุมมาเก็บไว้ในคอมพิวเตอร์เมื่อครบวงรอบของช่วงเวลา 1 วัน และเมื่อมีการใช้งานจากการเข้าถึงข้อมูลจากระยะไกลเท่านั้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บรรณานุกรม

1.รัชนัย,ไตรภพ อินทุใส, "ไมโครคอนโทรลเลอร์ 8051" กรุงเทพ, สำนักพิมพ์ฟิสิกส์เซ็นเตอร์ ,2543.

2. รศ.สมยศ จุณณะปิยะ ,"การประยุกต์ใช้งานไมโครคอนโทรลเลอร์" ,กรุงเทพ , วิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง, 2543

3. ธาริน สิทธิธรรมชารี, "คู่มือการเขียนโปรแกรม Visual Basic 6.0 ฉบับเพื่อการใช้งาน" กรุงเทพ, บริษัท ซัคเซส มีเดีย จำกัด.

4. กิตติ ภักดีวัฒนะกุล, จำลองครูอุตสาหะ "Visual Basic 6 ฉบับโปรแกรมเมอร์" กรุงเทพ, บริษัท ดวงกมลสมัย จำกัด, 2543.

5. ธรพล ฉันจรัสวิชัย, "การออกแบบและสร้างฐานข้อมูลด้วย Visual Basic 6.0" กรุงเทพ, บริษัท ซีเอ็ดยูเคชั่น จำกัด(มหาชน), 2543.

6. John Clark Craig, Jeff Webb, "Microsoft Visual Basic 6.0 Developer's Workshop" , USA ,Microsoft Press.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก ก.

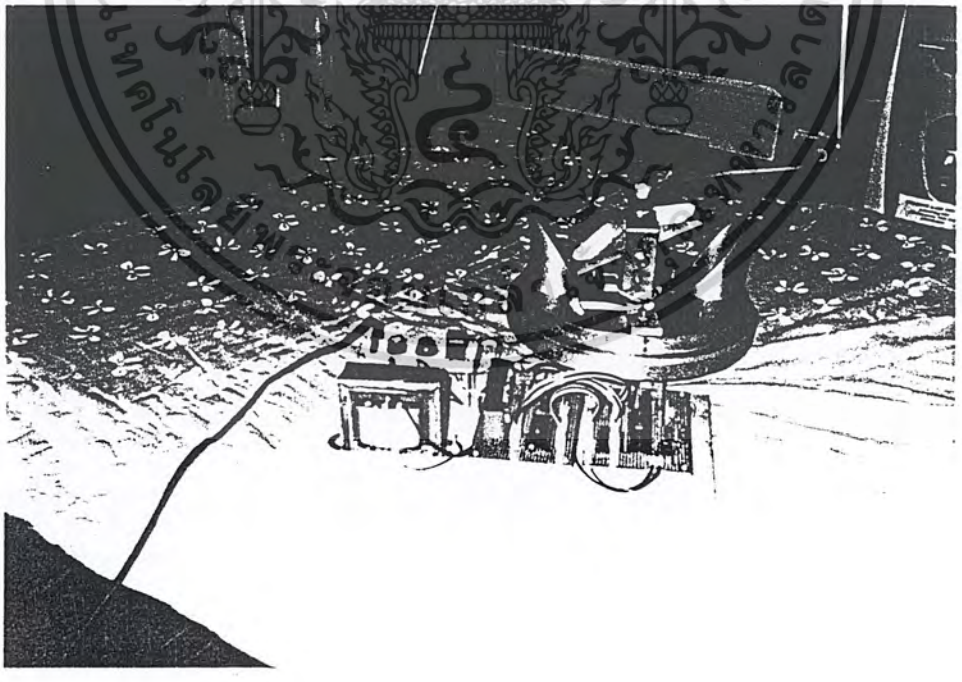
รายละเอียดวงจร

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปแสดงเครื่องวัดปริมาณน้ำฝน

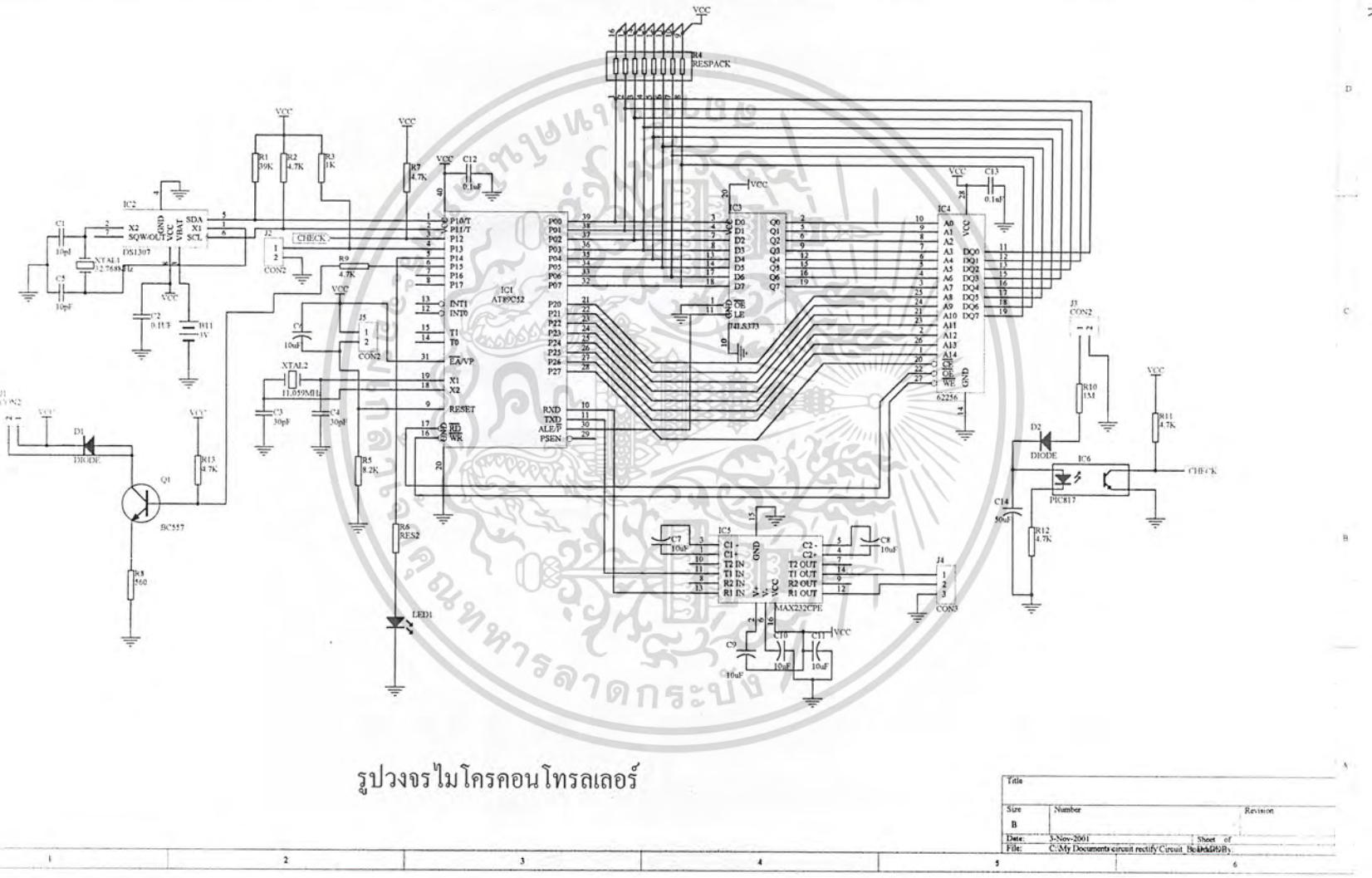
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปแสดงชุดกลไกภายในเครื่องวัดปริมาณน้ำฝน



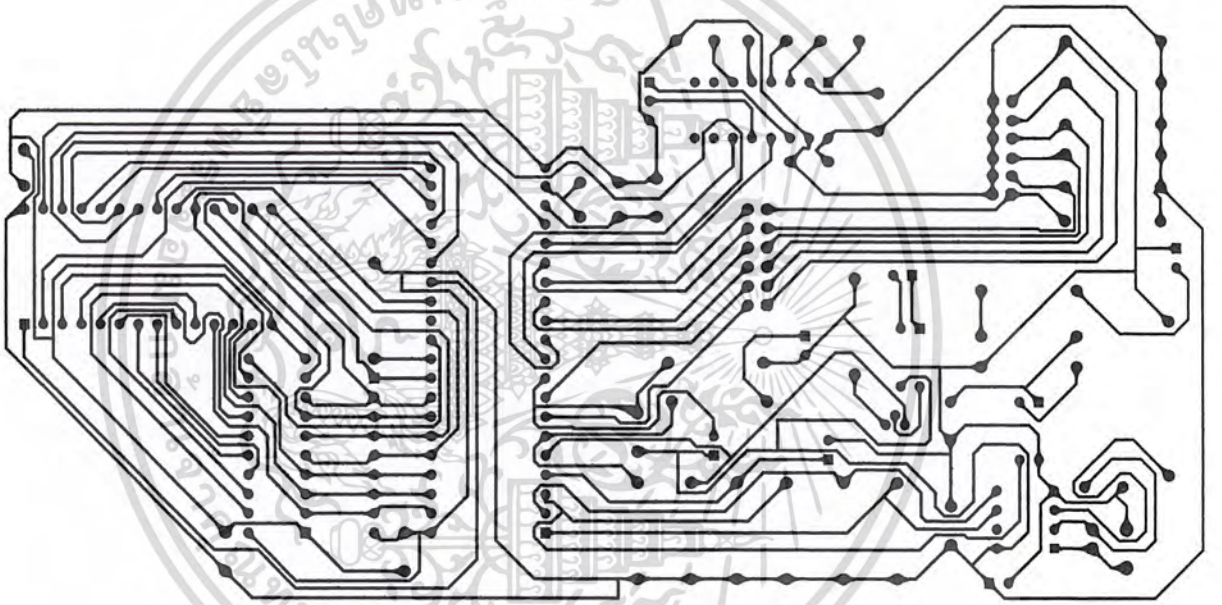
รูปแสดงชุดทดลองแผงวงจรไมโครคอนโทรลเลอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



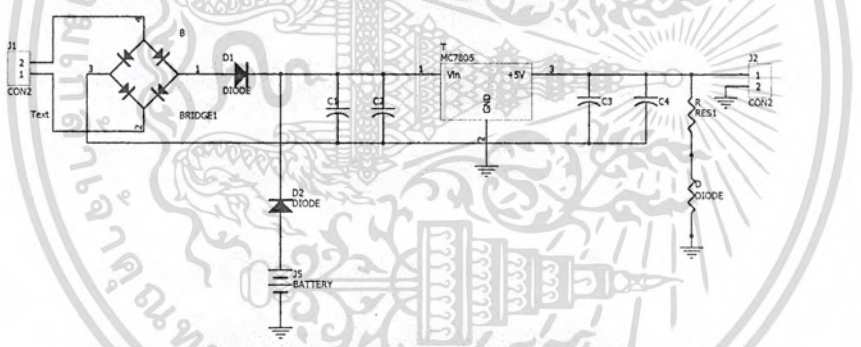
รูปวงจรไมโครคอนโทรลเลอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปลายวงจรไมโครคอนโทรลเลอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปวงจรภาคจ่ายไฟ

Title			
Size B	Number		Revision
Date:	3-Nov-2001	Sheet of	
File:	C:\MyDesign9.ddb	Drawn By:	

ลิขสิทธิ์ของมหาวิทยาลัยราชภัฏราชนครินทร์ ขอสงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น

ที่มา: ผลงานนักศึกษาในโครงการ

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก ข.

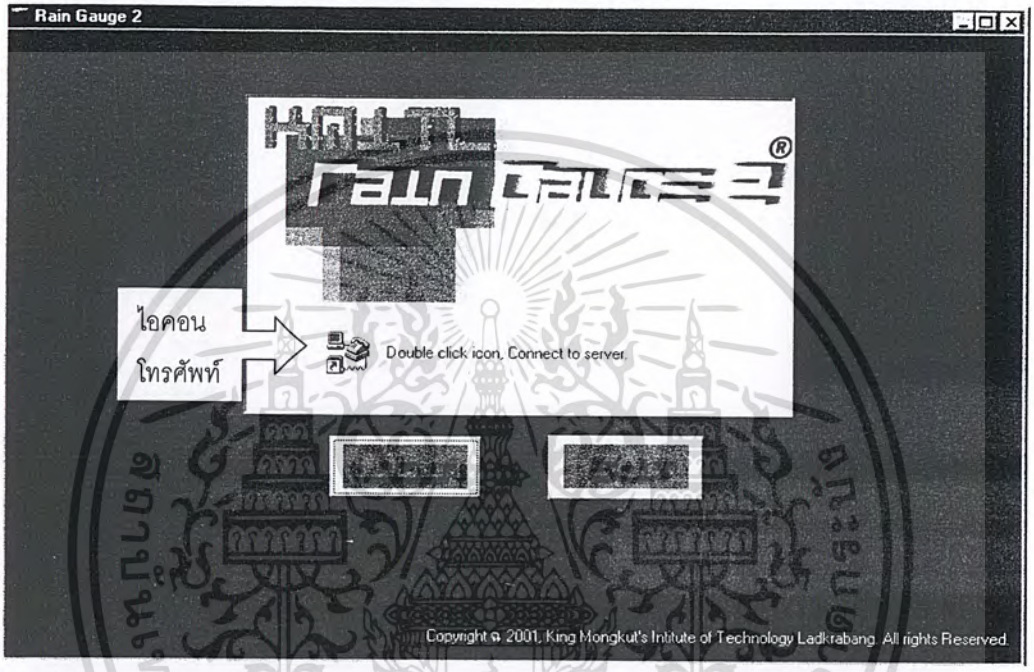
โปรแกรมการใช้งาน

- คู่มือการใช้งานโปรแกรม
- Source code โปรแกรม

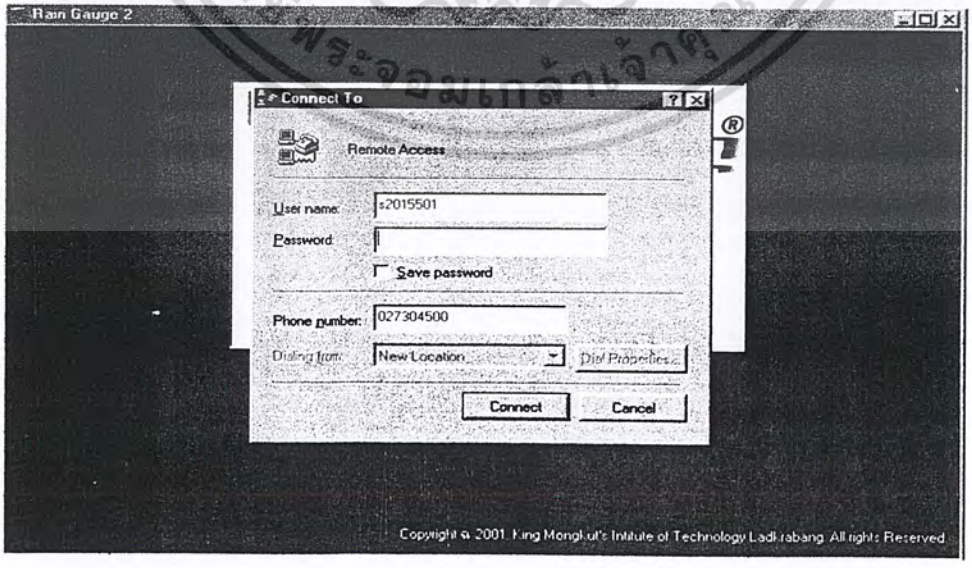
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

คู่มือการใช้งานโปรแกรม

1. เมื่อทำการติดตั้งโปรแกรมเป็นที่เรียบร้อยแล้ว เมื่อดับเบิ้ลคลิกที่ไอคอนโปรแกรม Rain Gauge 2 แล้วจะมีหน้าต่างเริ่มโปรแกรมขึ้นมา ให้ดับเบิ้ลคลิกที่ไอคอน รูปโทรศัพท์ที่อยู่ กลางหน้าจอ ดังรูปด้านล่างนี้

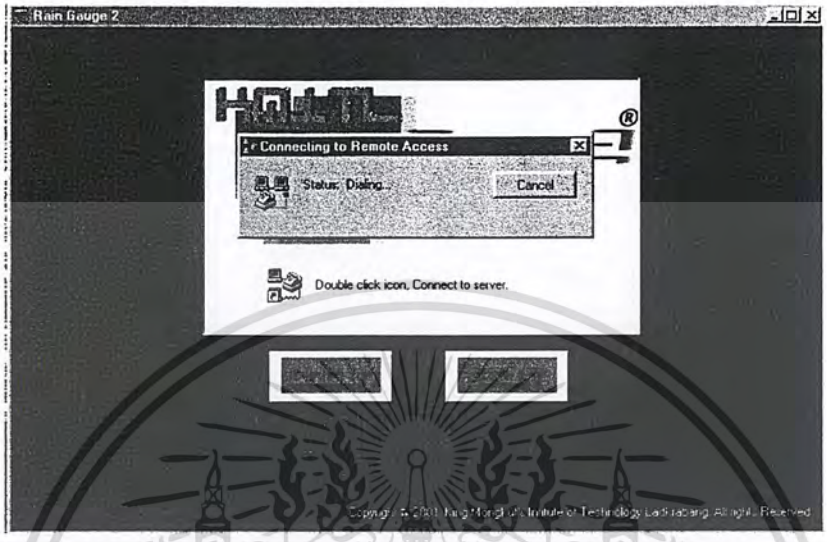


เมื่อดับเบิ้ลคลิกแล้วจะปรากฏไดอะลอกบ๊อกให้ทำการConnect ไปยังเครื่อง Server

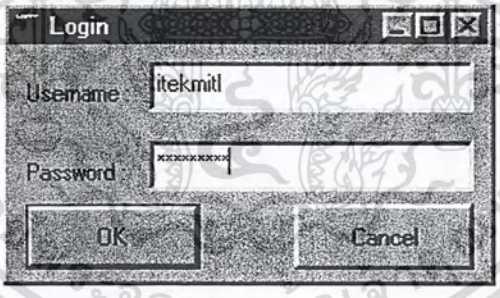


เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

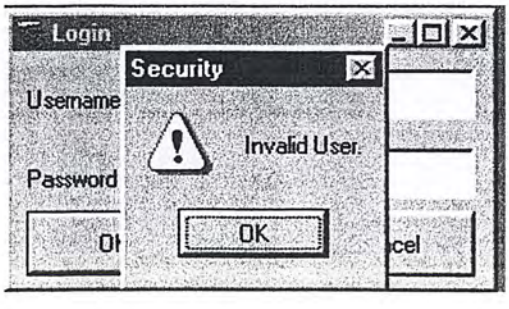
ให้ทำการป้อน Password และคลิกที่Connect ไปได้เลย โปรแกรมจะทำการหมุดโทรศัพท์ไปยังเครื่อง Server และทำการเชื่อมต่อกัน ดังรูปด้านล่างนี้



2. เมื่อเครื่องClient และเครื่องServer เชื่อมต่อกันแล้วให้คลิกที่ Login เพื่อเปิดหน้าต่าง Login ดังรูปด้านล่างนี้

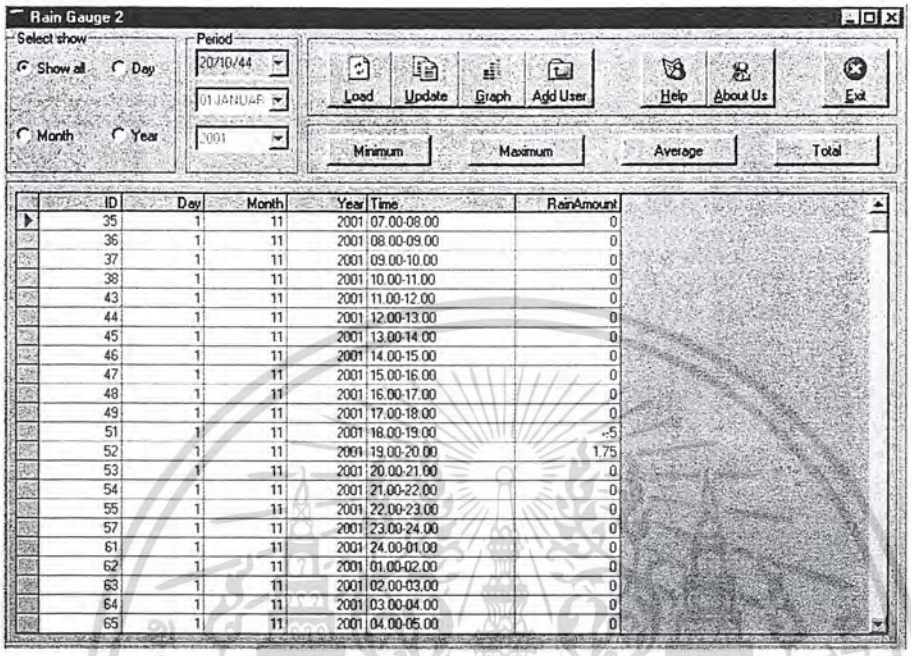


จากนั้นให้ป้อนUsername และ Password ลงไปจากนั้นคลิกที่ปุ่ม OK ถ้าUsername และ Password ผิดพลาดจะมีการแจ้งเตือนว่า invalid User ดังรูปด้านล่างนี้



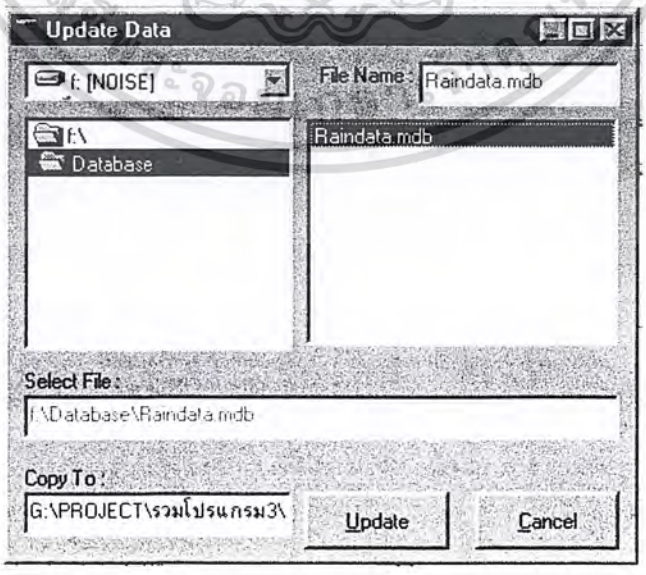
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

แต่หากป้อนUsername และ Password ถูกต้องโปรแกรมก็จเข้าสู่หน้าต่างหลักของโปรแกรม ดังรูปด้านล่างนี้



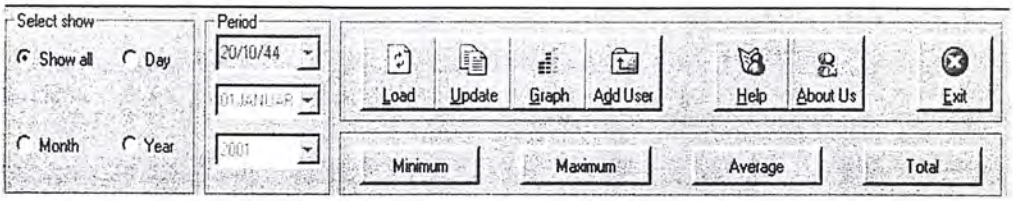
ID	Day	Month	Year	Time	RainAmount
35	1	11	2001	07.00-08.00	0
36	1	11	2001	08.00-09.00	0
37	1	11	2001	09.00-10.00	0
38	1	11	2001	10.00-11.00	0
43	1	11	2001	11.00-12.00	0
44	1	11	2001	12.00-13.00	0
45	1	11	2001	13.00-14.00	0
46	1	11	2001	14.00-15.00	0
47	1	11	2001	15.00-16.00	0
48	1	11	2001	16.00-17.00	0
49	1	11	2001	17.00-18.00	0
51	1	11	2001	18.00-19.00	5
52	1	11	2001	19.00-20.00	1.75
53	1	11	2001	20.00-21.00	0
54	1	11	2001	21.00-22.00	0
55	1	11	2001	22.00-23.00	0
57	1	11	2001	23.00-24.00	0
61	1	11	2001	24.00-01.00	0
62	1	11	2001	01.00-02.00	0
63	1	11	2001	02.00-03.00	0
64	1	11	2001	03.00-04.00	0
65	1	11	2001	04.00-05.00	0

3. ให้คลิกที่ปุ่ม Update เพื่อเข้าสู่หน้าต่าง Update จากนั้นให้ไปหาไฟล์ชื่อ Raindata.mdb ใน Harddisk ของเครื่อง Server ดังรูปด้านล่างนี้ เมื่อหาพบแล้วให้คลิก Update เพื่อทำการUpdate ฐานข้อมูลในเครื่องClient ให้เหมือนกับในเครื่องServer เมื่อเสร็จแล้วจะกลับสู่หน้าต่างหลัก



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4. ในการดูฐานข้อมูลแบบตารางที่หน้าจอหลักให้ทำการเลือกรูปแบบในการดูก่อน

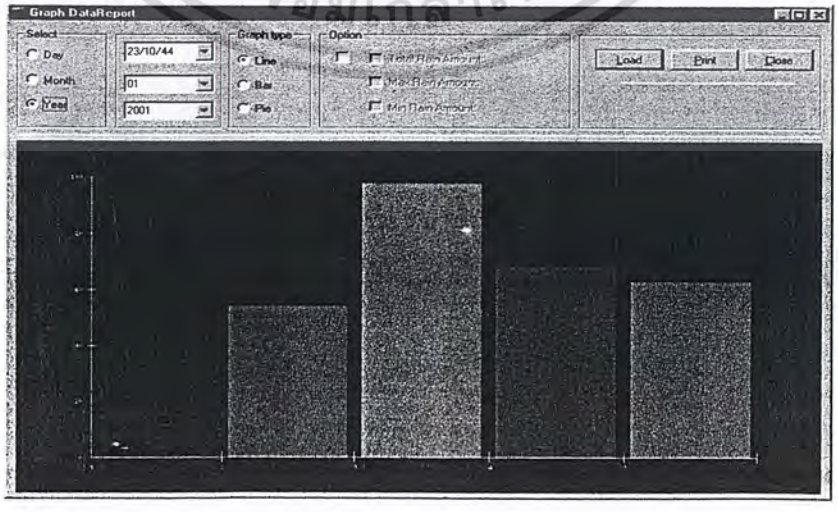


โดยให้เลือกรูปแบบการดูที่ส่วนของ Select Show ก่อนโดยเลือกถ้าเลือก

- Show all จะเป็นการเลือกดูข้อมูลทั้งหมด
- Day จะเป็นการเลือกดูข้อมูลรายวัน เมื่อเลือกที่รูปแบบนี้แล้วจะต้องไปเลือกวันที่ในการดูในส่วนของ Period อีกครั้งหนึ่ง
- Month จะเป็นการเลือกดูข้อมูลรายเดือนจะต้องไปเลือกเดือน และปีที่ต้องการดูในส่วนของPeriod อีกครั้งหนึ่ง
- Year จะเป็นการเลือกดูข้อมูลรายปี จะต้องไปทำการเลือกปีที่ต้องการดูในส่วนของPeriod อีกครั้งหนึ่ง

เมื่อเลือกเสร็จแล้วให้คลิกที่ปุ่ม Load เพื่อทำการแสดงผลฐานข้อมูลตามรูปแบบที่กำหนด และเมื่อต้องการทราบค่าสูงสุด ,ค่าต่ำสุด , ค่าเฉลี่ย และค่าผลรวมทั้งหมดของ ปริมาณน้ำฝนที่แสดงอยู่ในฐานข้อมูลที่เราเลือกขึ้นมา สามารถทำได้โดยคลิกที่ปุ่ม Maximum , Minimum , Average และ Total ตามลำดับ

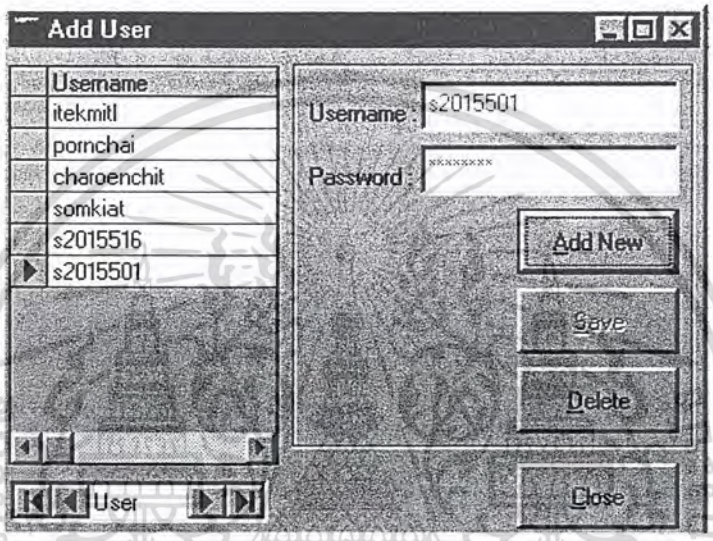
5. ในการดูข้อมูลแบบGraph ให้คลิกที่ปุ่ม Graph จะเข้าสู่หน้าจอแสดงผลแบบ Graph



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

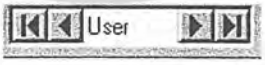
โดยในการใช้งาน Graph จะมีขั้นตอนการทำงานเหมือนกับการใช้งานฐานข้อมูลแบบตาราง คือ ต้องมีการเลือกช่วงเวลาก่อน และจะเพิ่มมาในส่วนของการเลือกรูปแบบกราฟ เมื่อเลือกได้แล้วก็ให้คลิกปุ่ม Load ก็จะทำกาแสดงผลกราฟตามที่ต้องการ

6. การเพิ่มผู้ใช้งาน ทำได้โดยการคลิกที่ปุ่ม Add User จะปรากฏหน้าต่างในการAdd User ขึ้นมา ดังรูปด้านล่างนี้



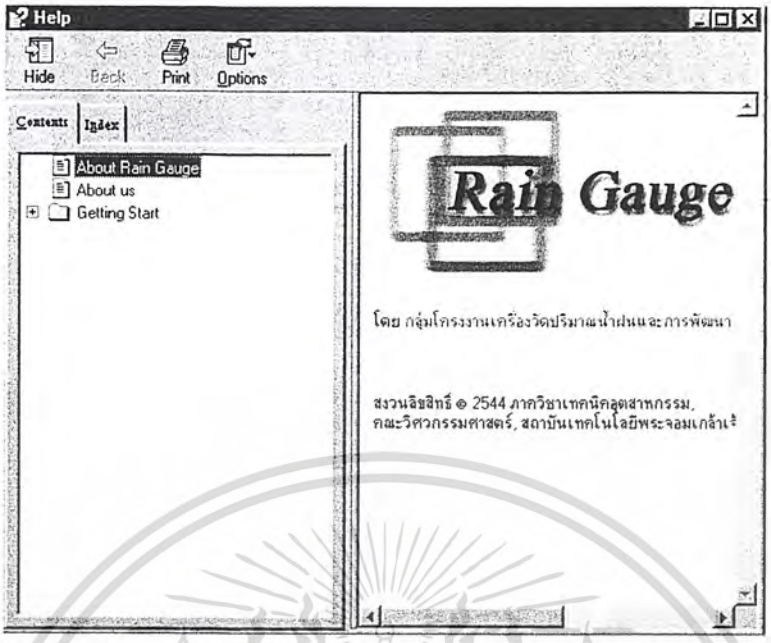
เมื่อปรากฏหน้าต่าง Add User ขึ้นมาแล้วก็สามารถทำการเพิ่มผู้ใช้ได้โดยคลิกที่ปุ่ม Add New แล้วป้อน Username และ Password ใหม่ที่เราต้องการลงไปจากนั้นคลิกที่ Save ก็จะเป็นการเพิ่มผู้ใช้

แต่หากต้องการลบผู้ใช้ออกให้คลิกที่ลูกศรด้านล่าง ดังแสดงในรูปด้านล่างนี้เพื่อใช้ในการเลือกผู้ใช้ที่ต้องการลบ เมื่อเลือกได้แล้วให้คลิกที่ปุ่ม Delete เพื่อลบชื่อผู้ใช้ที่ไม่ต้องการออกไป



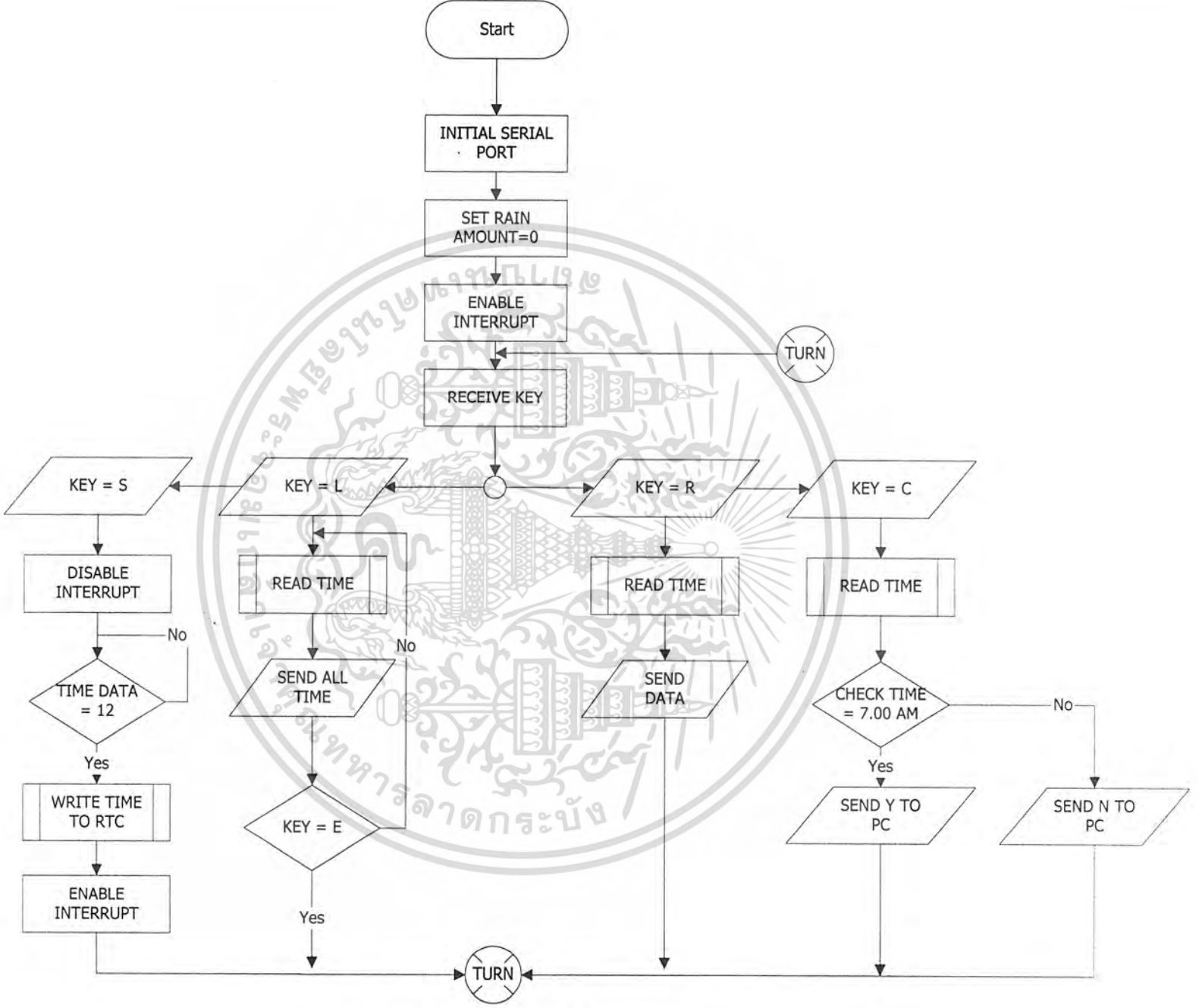
6. เมื่อต้องการความช่วยเหลือ หรือวิธีการใช้งานโปรแกรมสามารถคลิกที่ปุ่ม Help เพื่อเรียกหน้าต่างHelp ขึ้นมา ดังแสดงในรูปหน้าถัดไป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



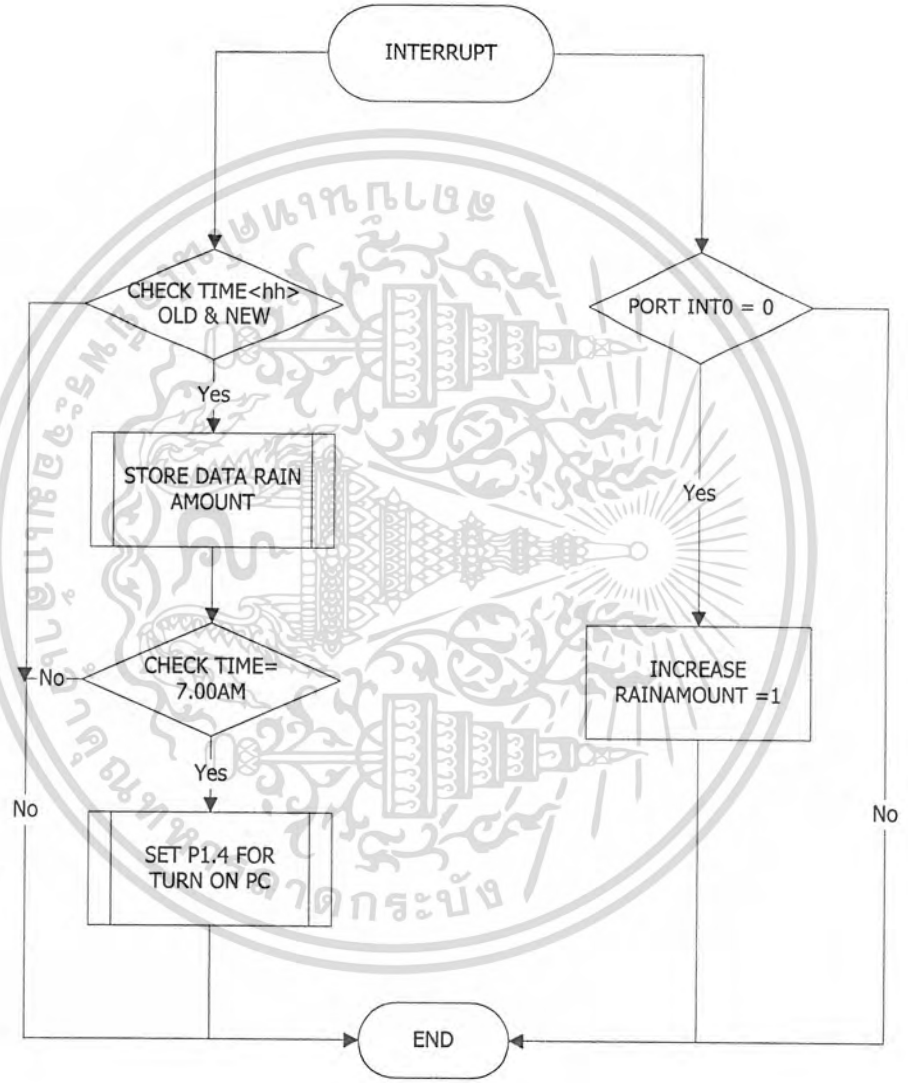
จากวิธีการใช้งานต่างๆดังที่แสดงมาแล้วนั้น ยังสามารถใช้ Help ช่วยในการสอนวิธีการใช้โปรแกรมได้อีกทางหนึ่ง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



แผนผังการทำงานของโปรแกรมใน MCS-51

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



แผนผังโหมด Interrupt

****** Source Code MCS51 ********

```
#include <8051io.h>
#include <8051reg.h>
#include <8051int.h>

register unsigned char year,month,date,hrs,min,sec,menu_choice,y,a;
register unsigned char oldhrs,oldsec;
register int rain[24],raingauge,i,num;

ADDR_RTC() asm{
                MOV   R0,#-5            ;have  1 param
                LCALL   ?auto0
?WRITE_BYTE     CLR    P1.1             ;start bit
                CLR    P1.0
                MOV    A,#$D0           ;send control byte
                LCALL  WRITE
                SETB   P1.1
                SETB   P1.0
                JB     P1.1,?WRITE_BYTE    ;loop until busy
                CLR    P1.0
                MOV    A,[R0]           ;send address low
                LCALL  WRITE
                SETB   P1.1
                SETB   P1.0
                JB     P1.1,?WRITE_BYTE    ;loop until busy
                CLR    P1.0
}
WRITE_RTC() asm{
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                MOV   R0,#-5              ;have  1 param
                LCALL  ?auto0
                MOV    A,[R0]             ;send data
                LCALL  WRITE
                SETB   P1.1
                SETB   P1.0
                JB     P1.1,?WRITE_BYTE    ;loop until busy
                CLR    P1.1
                CLR    P1.0
                SETB   P1.0               ;stop bit
                SETB   P1.1
                MOV    B,#$00
}
int READ() asm{
?READ_BYTE      MOV   R0,#-5              ;have  1 param
                LCALL  ?auto0
                CLR    P1.1               ;start bit
                CLR    P1.0
                MOV    A,#$D0             ;send control byte
                LCALL  WRITE
                SETB   P1.1
                SETB   P1.0
                JB     P1.1,?READ_BYTE     ;loop until busy
                CLR    P1.0
                MOV    A,[R0]             ;send address low
                LCALL  WRITE
                SETB   P1.1
                SETB   P1.0
                JB     P1.1,?READ_BYTE     ;loop until busy
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                CLR    P1.0

                SETB   P1.0
                SETB   P1.1
                CLR    P1.1            ;start bit
                CLR    P1.0
                MOV    A,#$D1          ;send control byte
                LCALL  WRITE
                SETB   P1.1
                SETB   P1.0
                JB     P1.1,?READ_BYTE ;loop until busy
                CLR    P1.0
                LCALL  READ_1
                SETB   P1.1
                SETB   P1.0
                CLR    P1.0

                SETB   P1.0            ;stop bit
                SETB   P1.1
                MOV    B,#$00
}
WRITE() asm{
                PUSH   $02
                MOV    R2,#$08
?LOOP_SEND      RLC    A
                MOV    P1.1,C
                SETB   P1.0
                CLR    P1.0
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                    DJNZ   R2,?LOOP_SEND
                    POP    $02
 }
 READ_1() asm{
                    PUSH   $02
                    MOV    R2,#$08
?LOOP_READ1              SETB   P1.0
                    MOV    C,P1.1
                    CLR    P1.0
                    RLC    A
                    DJNZ   R2,?LOOP_READ1
                    POP    $02
}
INTERRUPT(_IE0_)int0active()
{
    raingauge++;
    P1 |= 0x20;
    for(i=0;i<3000;i++);
    P1 &= 0xDF;
    for(i=0;i<5000;i++);
}

INTERRUPT(_TF0_)myfunc()
{
 TH0 = 0x04;
 TL0 = 0xFF;
 hrs  = READ(2);
 min  = READ(1);
 sec  = READ(0);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
            if(hrs!=oldhrs){
               a = ((hrs&0xF0)>>4);
               a =(hrs&0x0F)+(a*10);
               if(a<8)num = 16+a;
               else num = a-8;
               rain[num] = raingauge;
               } oldhrs = hrs;
               if(a == 7 && sec <2){
                if(min == 0){
                P1 |= 0x10;
               }}else  P1 &= 0xEF;
}
unsigned char set_time()
{
unsigned char w,y;
      w = getch();
      w = (((w-0x30)<<4)&0xF0);
      y = getch();
      y = (((y-0x30)&0x0F)|w);
      return(y);
}

show_time(data)
unsigned char data;
{  unsigned char a;
     a = data;
     a = ((data&0xF0)>>4);
     printf("%d",a);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        data =(data&0x0F);
        printf("%d",data);
}
unsigned char time(data)
unsigned char data;
{  unsigned char a;
        a = data;
        a = ((data&0xF0)>>4);
        a =(data&0x0F)+(a*10);
        return(a);
}

main( )
{
    oldsec = 0; raingauge = 0; num = 0;
    serinit(9600);
    TMOD = 0x21;
    TCON = 0x50;
    TH0 = 0x00;
    TL0 = 0x00;
    oldhrs = READ(2);
    oldhrs = time(oldhrs);
    P1 &= 0xDF;
    IE  = 0x83;
    rain[0] = 0;   rain[1] = 0;  rain[2] = 0;  rain[3] = 0;   rain[4] = 0;
    rain[5] = 0;   rain[6] = 0;  rain[7] = 0;  rain[8] = 0;   rain[9] = 0;
    rain[10]= 0;   rain[11]= 0;  rain[12]= 0;  rain[13]= 0;   rain[14]= 0;
    rain[15]= 0;   rain[16]= 0;  rain[17]= 0;  rain[18]= 0;   rain[19]= 0;
    rain[20]= 0;   rain[21]= 0;  rain[22]= 0;  rain[23]= 0;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
while(1){
y = 0;
menu_choice = getch();
switch(menu_choice){
case 's': IE = 0x03;
        year  = set_time();
        month = set_time();
        date  = set_time();
        hrs   = set_time();
        min   = set_time();
        sec   = set_time();
        asm{    SETB   P1.1

                SETB   P1.0
        }
        ADDR_RTC(0);
        WRITE_RTC(sec);
        ADDR_RTC(1);
        WRITE_RTC(min);
        ADDR_RTC(2);
        WRITE_RTC(hrs);
        ADDR_RTC(4);
        WRITE_RTC(date);
        ADDR_RTC(5);
        WRITE_RTC(month);
        ADDR_RTC(6);
        WRITE_RTC(year);
        IE = 0x83;
        break;
```

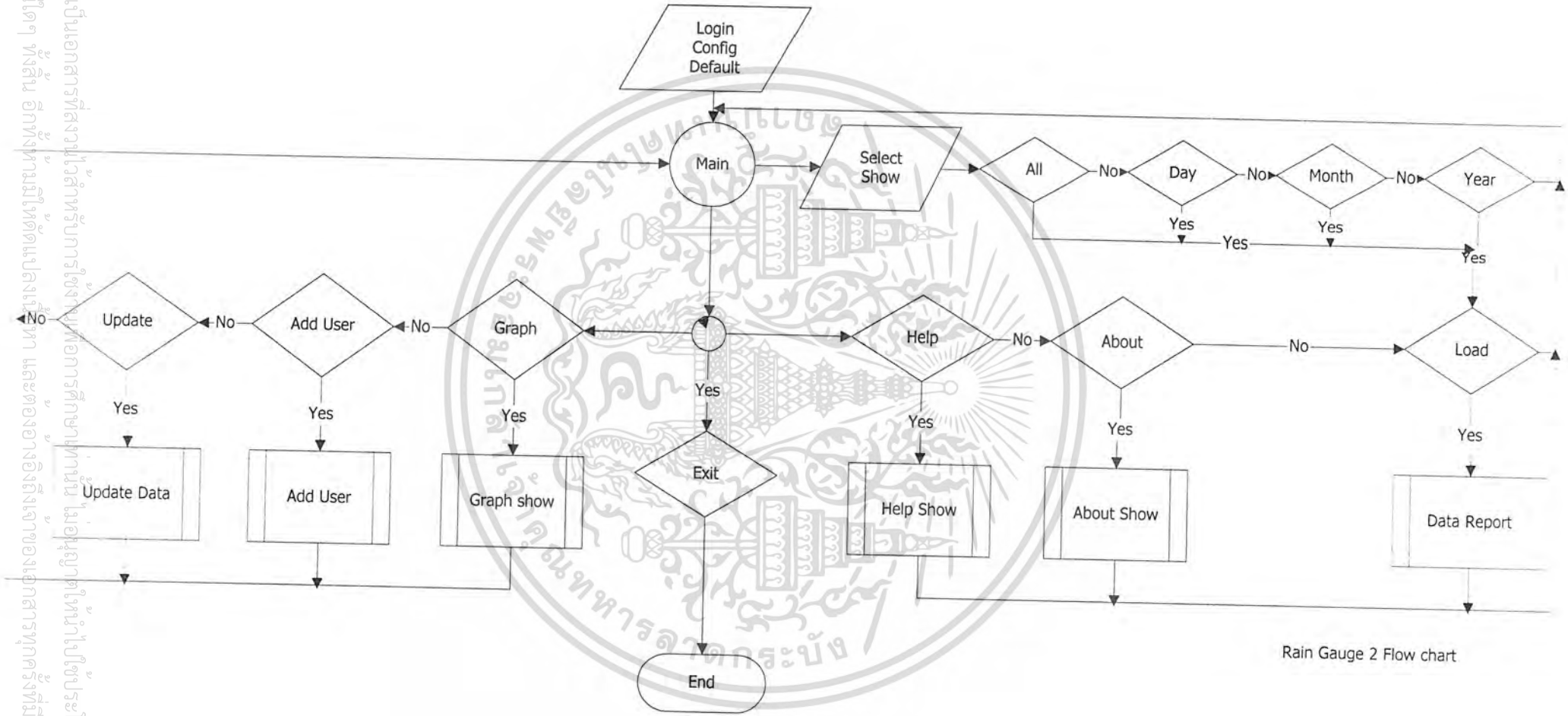
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
 case 'c':
        hrs  = READ(2);
        min  = READ(1);
        hrs  = time(hrs);
        min  = time(min);
        if(hrs==7){
        if(min<20) printf("y");
        }else printf("n");
        break;
 case 'r':
        year = READ(6);
        month= READ(5);
        date = READ(4);
        for(i=0;i<24;i++)
        printf("%d,",rain[i]);
        show_time(date);
        printf(",");
        show_time(month);
        printf(",20");
        show_time(year);
        printf(",");
        break;
 case '|': while(y!='e'){
            year = READ(6);
            month= READ(5);
            date = READ(4);
            hrs  = READ(2);
            min  = READ(1);
            sec  = READ(0);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                show_time(sec);
                printf(",");
                show_time(min);
                printf(",");
                show_time(hrs);
                printf(",");
                show_time(date);
                printf(",");
                show_time(month);
                printf(",");
                printf("20");
                show_time(year);
                printf(",");
                y = getch();
                }break;
        }
    }
}
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



แผนผังการทำงานของโปรแกรม Client

Source Code Rain Gauge 2

```
***** frmIntro.frm *****
************************
Option Explicit
Private Sub cmdCancel_Click()
Beep
Unload Me
End Sub

Private Sub cmdLogin_Click()
frmLogin.Show
Unload Me
End Sub

Private Sub Form_Paint()
Dim lngY As Long
Dim lngScaleHeight As Long
Dim lngScaleWidth As Long
ScaleMode = vbPixels
lngScaleHeight = ScaleHeight
lngScaleWidth = ScaleWidth
DrawStyle = vbInvisible
FillStyle = vbFSSolid
For lngY = 0 To lngScaleHeight
   FillColor = RGB(0, 0, 255 - (lngY * 255) \ lngScaleHeight)
   Line (-1, lngY - 1)-(lngScaleWidth, lngY + 1), , B
Next lngY
End Sub
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
***** frmLogin.frm *****
**************************
Private Sub cmdCancel_Click()
End
End Sub

Private Sub cmdOK_Click()
Data1.Recordset.FindFirst "Username = '" & txtUsername & "'"
If txtUsername.Text = txtUser.Text And _
  txtPassword.Text = txtPass.Text Then
  frmMain.Show
  Unload Me
Else
   If MsgBox("Invalid User.", 48, "Security") = vbOK Then
   txtUsername = ""
   txtPassword = ""
   txtUsername.SetFocus
   End If
End If
End Sub

Private Sub Form_Activate()
txtUsername.SetFocus
End Sub

Private Sub Form_Load()
Data1.DatabaseName = App.Path & "\User.mdb"
End Sub
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
***** frmLogin.frm *****
**************************
Private Sub cmdCancel_Click()
End
End Sub

Private Sub cmdOK_Click()
Data1.Recordset.FindFirst "Username = '" & txtUsername & "'"
If txtUsername.Text = txtUser.Text And _
  txtPassword.Text = txtPass.Text Then
  frmMain.Show
  Unload Me
Else
   If MsgBox("Invalid User.", 48, "Security") = vbOK Then
   txtUsername = ""
   txtPassword = ""
   txtUsername.SetFocus
   End If
End If
End Sub

Private Sub Form_Activate()
txtUsername.SetFocus
End Sub

Private Sub Form_Load()
Data1.DatabaseName = App.Path & "\User.mdb"
End Sub
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
***** frmMain.frm *****
************************
Option Explicit
Dim sqlcmd As String

Private Sub cmdAbout_Click()
frmAbout.Show
End Sub

Private Sub cmdAddUser_Click()
frmAddUser.Show
End Sub

Private Sub cmdAvr_Click()
If optShowall.Value = True Then
sqlcmd = "select avg(RainAmount) as Average_Value from RainDatabase"
End If

If optDay.Value = True Then
sqlcmd = "select avg(RainAmount) as Average_Value from RainDatabase"
sqlcmd = sqlcmd & " where Day = " & DTPicker1.Day
sqlcmd = sqlcmd & " and Month = " & DTPicker1.Month
sqlcmd = sqlcmd & " and Year = " & DTPicker1.Year
End If

If optMonth.Value = True Then
sqlcmd = "select  avg(RainAmount) as Average_Value from RainDatabase"
sqlcmd = sqlcmd & " where Month = " & Val(cmbMonth.Text)
sqlcmd = sqlcmd & " and Year = " & cmbYear.Text
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
End If

If optYear.Value = True Then
sqlcmd = "select avg(RainAmount) as Average_Value from RainDatabase"
sqlcmd = sqlcmd & " where Year = " & cmbYear.Text
End If
Text5.Text = sqlcmd
Data1.RecordSource = sqlcmd
Data1.Refresh
End Sub

Private Sub cmdExit_Click()
Beep
Unload Me
End Sub

Private Sub cmdGraph_Click()
frmGraph.Show
Unload Me
End Sub

Private Sub cmdHelp_Click()
cmdHelp.HelpContextID = 1001
SendKeys "{F1}"
End Sub

Private Sub cmdLoad_Click()

If optShowall.Value = True Then
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
sqlcmd = "select Day,Month,Year,Time,RainAmount from RainDatabase"
End If

If optDay.Value = True Then
sqlcmd = "select Day,Month,Year,Time,RainAmount from RainDatabase"
sqlcmd = sqlcmd & " where Day = " & DTPicker1.Day
sqlcmd = sqlcmd & " and Month = " & DTPicker1.Month
sqlcmd = sqlcmd & " and Year = " & DTPicker1.Year
End If

If optMonth.Value = True Then
sqlcmd = "select Day, Month, Year, Time, RainAmount from RainDatabase"
sqlcmd = sqlcmd & " where Month = " & Val(cmbMonth.Text)
sqlcmd = sqlcmd & " and Year = " & cmbYear.Text
End If

If optYear.Value = True Then
sqlcmd = "select Day, Month, Year, Time, RainAmount from RainDatabase"
sqlcmd = sqlcmd & " where Year = " & cmbYear.Text
End If
Text5.Text = sqlcmd
Data1.RecordSource = sqlcmd
Data1.Refresh
End Sub

Private Sub cmdMax_Click()

If optShowall.Value = True Then
sqlcmd = "select max(RainAmount) as Maximum_Value from RainDatabase"
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
End If

If optDay.Value = True Then
sqlcmd = "select max(RainAmount) as Maximum_Value from RainDatabase"
sqlcmd = sqlcmd & " where Day = " & DTPicker1.Day
sqlcmd = sqlcmd & " and Month = " & DTPicker1.Month
sqlcmd = sqlcmd & " and Year = " & DTPicker1.Year
End If

If optMonth.Value = True Then
sqlcmd = "select  max(RainAmount) as Maximum_Value from RainDatabase"
sqlcmd = sqlcmd & " where Month = " & Val(cmbMonth.Text)
sqlcmd = sqlcmd & " and Year = " & cmbYear.Text
End If

If optYear.Value = True Then
sqlcmd = "select max(RainAmount) as Maximum_Value from RainDatabase"
sqlcmd = sqlcmd & " where Year = " & cmbYear.Text
End If


Text5.Text = sqlcmd
Data1.RecordSource = sqlcmd
Data1.Refresh
End Sub

Private Sub cmdMin_Click()

If optShowall.Value = True Then
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
sqlcmd = "select min(RainAmount) as Minimum_Value from RainDatabase"
End If

If optDay.Value = True Then
sqlcmd = "select min(RainAmount) as Minimum_Value from RainDatabase"
sqlcmd = sqlcmd & " where Day = " & DTPicker1.Day
sqlcmd = sqlcmd & " and Month = " & DTPicker1.Month
sqlcmd = sqlcmd & " and Year = " & DTPicker1.Year
End If

If optMonth.Value = True Then
sqlcmd = "select  min(RainAmount) as Minimum_Value from RainDatabase"
sqlcmd = sqlcmd & " where Month = " & Val(cmbMonth.Text)
sqlcmd = sqlcmd & " and Year = " & cmbYear.Text
End If

If optYear.Value = True Then
sqlcmd = "select min(RainAmount) as Minimum_Value from RainDatabase"
sqlcmd = sqlcmd & " where Year = " & cmbYear.Text
End If
Text5.Text = sqlcmd
Data1.RecordSource = sqlcmd
Data1.Refresh
End Sub

Private Sub cmdTotal_Click()
If optShowall.Value = True Then
sqlcmd = "select sum(RainAmount) as Total_Value from RainDatabase"
End If
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
If optDay.Value = True Then
sqlcmd = "select sum(RainAmount) as Total_Value from RainDatabase"
sqlcmd = sqlcmd & " where Day = " & DTPicker1.Day
sqlcmd = sqlcmd & " and Month = " & DTPicker1.Month
sqlcmd = sqlcmd & " and Year = " & DTPicker1.Year
End If

If optMonth.Value = True Then
sqlcmd = "select  sum(RainAmount) as Total_Value from RainDatabase"
sqlcmd = sqlcmd & " where Month = " & Val(cmbMonth.Text)
sqlcmd = sqlcmd & " and Year = " & cmbYear.Text
End If

If optYear.Value = True Then
sqlcmd = "select sum(RainAmount) as Total_Value from RainDatabase"
sqlcmd = sqlcmd & " where Year = " & cmbYear.Text
End If
Text5.Text = sqlcmd
Data1.RecordSource = sqlcmd
Data1.Refresh
End Sub

Private Sub cmdUpdate_Click()
frmUpdate.Show
Unload Me
End Sub
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
Private Sub Form_Load()
Data1.DatabaseName = App.Path & "\Raindata.mdb"
App.HelpFile = "RainGauge.chm"
End Sub


Private Sub optDay_Click()
If optDay.Value = True Then
DTPicker1.Enabled = True
cmbMonth.Enabled = False
cmbYear.Enabled = False
End If
End Sub

Private Sub optMonth_Click()
If optMonth.Value = True Then
DTPicker1.Enabled = False
cmbMonth.Enabled = True
cmbYear.Enabled = True
End If
End Sub

Private Sub optShowall_Click()
If optShowall.Value = True Then
DTPicker1.Enabled = False
cmbMonth.Enabled = False
cmbYear.Enabled = False
End If
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
sqlcmd = "select Day,Month,Year,Time,RainAmount from RainDatabase"
Text5.Text = sqlcmd
Data1.RecordSource = sqlcmd
Data1.Refresh
End Sub
Private Sub optYear_Click()
If optYear.Value = True Then
DTPicker1.Enabled = False
cmbMonth.Enabled = False
cmbYear.Enabled = True
End If
End Sub


***** frmUpdate.frm *****
**************************
Private Sub cmdCancel_Click()
frmMain.Show
Unload Me
End Sub

Private Sub cmdCopy_Click()
FileCopy txtText2.Text, txtText3.Text
frmMain.Show
Unload Me
End Sub

Private Sub Dir1_Change()
File1.Path = Dir1.Path
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
sqlcmd = "select Day,Month,Year,Time,RainAmount from RainDatabase"
Text5.Text = sqlcmd
Data1.RecordSource = sqlcmd
Data1.Refresh
End Sub
Private Sub optYear_Click()
If optYear.Value = True Then
DTPicker1.Enabled = False
cmbMonth.Enabled = False
cmbYear.Enabled = True
End If
End Sub


***** frmUpdate.frm *****
**************************
Private Sub cmdCancel_Click()
frmMain.Show
Unload Me
End Sub

Private Sub cmdCopy_Click()
FileCopy txtText2.Text, txtText3.Text
frmMain.Show
Unload Me
End Sub

Private Sub Dir1_Change()
File1.Path = Dir1.Path
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
End Sub

Private Sub Drive1_Change()
Dir1.Path = Drive1.Drive
End Sub

Private Sub File1_Click()
txtText2.Text = Dir1.Path & "\" & File1.FileName
If txtText2.Text = txtText3.Text Then
cmdCopy.Enabled = False
End If
End Sub

Private Sub Form_Activate()
txtText3.Text = App.Path & "\Raindata.mdb"
End Sub
Private Sub txtText2_Change()
cmdCopy.Enabled = True
End Sub
```

***** frmGraph.frm *****

****

```
Option Explicit
Dim sqlcmd As String
Dim sqlstr As String
Private Sub Label3_Click()
End Sub
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
Private Sub Check4_Click()
If Check4.Value = 1 Then
Check1.Enabled = True
Check2.Enabled = True
Check3.Enabled = True
End If
If Check4.Value = 0 Then
Check1.Enabled = False
Check2.Enabled = False
Check3.Enabled = False
End If
End Sub

Private Sub cmdClose_Click()
frmMain.Show
Unload Me
End Sub

Private Sub cmdLoad_Click()
Dim rs As Object
Dim j As Integer
Dim i As Integer
Dim h As String

If Check4.Value = 1 Then
If Check1.Value = 1 Then

If optDay.Value = True Then
sqlstr = "select sum(RainAmount) as Sum_Value from RainDatabase"
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
sqlstr = sqlstr & " where Day = " & DTPicker1.Day
sqlstr = sqlstr & " and Month = " & DTPicker1.Month
sqlstr = sqlstr & " and Year = " & DTPicker1.Year
End If

If optMonth.Value = True Then
sqlstr = "select  sum(RainAmount) as Sum_Value from RainDatabase"
sqlstr = sqlstr & " where Month = " & Val(cmbMonth.Text)
sqlstr = sqlstr & " and Year = " & cmbYear.Text
End If

If optYear.Value = True Then
sqlstr = "select sum(RainAmount) as Sum_Value from RainDatabase"
sqlstr = sqlstr & " where Year = " & cmbYear.Text
End If

Data1.RecordSource = sqlstr
Data1.Refresh
Check1.Caption = "Total Rain Amount: " & DBGrid1.Text & " mm."
End If

If Check2.Value = 1 Then

If optDay.Value = True Then
sqlstr = "select max(RainAmount) as Maximum_Value from RainDatabase"
sqlstr = sqlstr & " where Day = " & DTPicker1.Day
sqlstr = sqlstr & " and Month = " & DTPicker1.Month
sqlstr = sqlstr & " and Year = " & DTPicker1.Year
End If
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
If optMonth.Value = True Then
sqlstr = "select  max(RainAmount) as Maximum_Value from RainDatabase"
sqlstr = sqlstr & " where Month = " & Val(cmbMonth.Text)
sqlstr = sqlstr & " and Year = " & cmbYear.Text
End If

If optYear.Value = True Then
sqlstr = "select max(RainAmount) as Maximum_Value from RainDatabase"
sqlstr = sqlstr & " where Year = " & cmbYear.Text
End If

Data1.RecordSource = sqlstr
Data1.Refresh
Check2.Caption = "Max Rain Amount: " & DBGrid1.Text & " mm."
End If

If Check3.Value = 1 Then

If optDay.Value = True Then
sqlstr = "select min(RainAmount) as Minimum_Value from RainDatabase"
sqlstr = sqlstr & " where Day = " & DTPicker1.Day
sqlstr = sqlstr & " and Month = " & DTPicker1.Month
sqlstr = sqlstr & " and Year = " & DTPicker1.Year
End If

If optMonth.Value = True Then
sqlstr = "select  min(RainAmount) as Minimum_Value from RainDatabase"
sqlstr = sqlstr & " where Month = " & Val(cmbMonth.Text)
sqlstr = sqlstr & " and Year = " & cmbYear.Text
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
End If

If optYear.Value = True Then
sqlstr = "select min(RainAmount) as Minimum_Value from RainDatabase"
sqlstr = sqlstr & " where Year = " & cmbYear.Text
End If
Data1.RecordSource = sqlstr
Data1.Refresh
Check3.Caption = "Min Rain Amount: " & DBGrid1.Text & " mm."
End If
End If

If optDay.Value = True Then
sqlstr = "select sum(RainAmount) as Sum_Value from RainDatabase"
sqlstr = sqlstr & " where Day = " & DTPicker1.Day
sqlstr = sqlstr & " and Month = " & DTPicker1.Month
sqlstr = sqlstr & " and Year = " & DTPicker1.Year
End If

If optMonth.Value = True Then
sqlstr = "select  sum(RainAmount) as Sum_Value from RainDatabase"
sqlstr = sqlstr & " where Month = " & Val(cmbMonth.Text)
sqlstr = sqlstr & " and Year = " & cmbYear.Text
End If

If optYear.Value = True Then
sqlstr = "select sum(RainAmount) as Sum_Value from RainDatabase"
sqlstr = sqlstr & " where Year = " & cmbYear.Text
End If
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
Data1.RecordSource = sqlstr
Data1.Refresh
Graph1.BottomTitle = "Total Rain Amount: " & DBGrid1.Text & " mm."

If optLine.Value = True Then
  h = gphLine
End If

If optBar.Value = True Then
  h = gphBar2D
End If

If optPie.Value = True Then
  h = gphPie3D
End If

Graph1.GraphType = h
Graph1.DataReset = gphAllData
Graph1.LineStats = gphMeanMinmax

If optDay.Value = True Then
  i = 24
  Graph1.GraphTitle = "Show of Rain Amount on : " & DTPicker1.Value & "."
End If

If optMonth.Value = True Then
  i = 31
  Graph1.GraphTitle = "Show of Rain Amount on : " & cmbMonth.Text & "/" &
cmbYear.Text & "."
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
End If

If optYear.Value = True Then
   i = 12
   Graph1.GraphTitle = "Show of Rain Amount on : " & cmbYear.Text & "."
End If

Graph1.NumSets = 1
Graph1.NumPoints = i
If optDay.Value = True Then
sqlcmd = "select RainDatabase.Time, "
sqlcmd = sqlcmd & " sum(RainDatabase.RainAmount) as Total_Value "
sqlcmd = sqlcmd & " from RainDatabase "
sqlcmd = sqlcmd & " where RainDatabase.Day = " & DTPicker1.Day
sqlcmd = sqlcmd & " and RainDatabase.Month = " & DTPicker1.Month
sqlcmd = sqlcmd & " and RainDatabase.Year = " & DTPicker1.Year
sqlcmd = sqlcmd & " group by RainDatabase.Time;"

End If

If optMonth.Value = True Then
sqlcmd = "select RainDatabase.Day, "
sqlcmd = sqlcmd & " sum(RainDatabase.RainAmount) as Total_Value "
sqlcmd = sqlcmd & " from RainDatabase "
sqlcmd = sqlcmd & " where RainDatabase.Month = " & Val(cmbMonth.Text)
sqlcmd = sqlcmd & " and RainDatabase.Year = " & cmbYear.Text
sqlcmd = sqlcmd & " group by RainDatabase.Day;"
End If
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
If optYear.Value = True Then
sqlcmd = "select RainDatabase.Month, "
sqlcmd = sqlcmd & " sum(RainDatabase.RainAmount) as Total_Value "
sqlcmd = sqlcmd & " from RainDatabase "
sqlcmd = sqlcmd & " where RainDatabase.Year = " & cmbYear.Text
sqlcmd = sqlcmd & " group by RainDatabase.Month;"
End If

Data1.RecordSource = sqlcmd
Data1.Refresh
Set rs = Data1.Recordset
Do
  j = j + 1
  Graph1.ThisPoint = j
  Graph1.GraphData = rs.Fields("Total_Value").Value
  Graph1.ThisPoint = j

  If optDay.Value = True Then
  Graph1.LabelText = rs.Fields("Time").Value
  End If

  If optMonth.Value = True Then
  Graph1.LabelText = rs.Fields("Day").Value
  End If

  If optYear.Value = True Then
  Graph1.LabelText = rs.Fields("Month").Value
  End If
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    rs.MoveNext
Loop Until rs.EOF
Graph1.DrawMode = gphDraw

End Sub

Private Sub cmdPrint_Click()
Graph1.DrawMode = gphPrint
End Sub

Private Sub Form_Load()
Data1.DatabaseName = App.Path & "\Raindata.mdb"
End Sub

Private Sub optDay_Click()
If optDay.Value = True Then
DTPicker1.Enabled = True
cmbMonth.Enabled = False
cmbYear.Enabled = False
End If
End Sub

Private Sub optMonth_Click()
If optMonth.Value = True Then
DTPicker1.Enabled = False
cmbMonth.Enabled = True
cmbYear.Enabled = True
End If
End Sub
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
Private Sub optOp_Click()
End Sub

Private Sub optYear_Click()
If optYear.Value = True Then
DTPicker1.Enabled = False
cmbMonth.Enabled = False
cmbYear.Enabled = True
End If
End Sub


***** frmAddUser.frm *****
******************************
Private Sub cmdAddNew_Click()
Data1.Recordset.AddNew
cmdSave.Enabled = True
cmdAddNew.Enabled = False
txtUsername.Enabled = True
txtPassword.Enabled = True
txtUsername.SetFocus
End Sub

Private Sub cmdClose_Click()
Unload Me
End Sub

Private Sub cmdSave_Click()
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
If Len(txtUsername.Text) = 0 Or Len(txtUsername.Text) = 0 Then
If MsgBox("Please fill Username and Password into TextBox.", 48, "Error") = vbOK Then
txtUsername.SetFocus
End If
End If

If Len(txtUsername.Text) <> 0 Or Len(txtUsername.Text) <> 0 Then
Data1.Recordset.AddNew
cmdAddNew.Enabled = True
cmdSave.Enabled = False
txtUsername.Enabled = False
txtPassword.Enabled = False
End If
End Sub

Private Sub cmdDelete_Click()
Data1.Recordset.Delete
Data1.Recordset.MoveLast
End Sub

Private Sub Form_Activate()
txtUsername.Enabled = False
txtPassword.Enabled = False
Data1.Recordset.MoveLast
cmdSave.Enabled = False
End Sub

Private Sub Form_Load()
Data1.DatabaseName = App.Path & "\User.mdb"
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
End Sub

Private Sub txtUsername_Change()
If txtUsername.Text = "itekmitl" _
And txtPassword.Text = "kmitl2001" Then
cmdDelete.Enabled = False
Else
cmdDelete.Enabled = True
End If
End Sub


***** frmAbout.frm *****
**************************
Option Explicit
Const READ_CONTROL = &H20000
Const KEY_QUERY_VALUE = &H1
Const KEY_SET_VALUE = &H2
Const KEY_CREATE_SUB_KEY = &H4
Const KEY_ENUMERATE_SUB_KEYS = &H8
Const KEY_NOTIFY = &H10
Const KEY_CREATE_LINK = &H20
Const KEY_ALL_ACCESS = KEY_QUERY_VALUE + KEY_SET_VALUE + _
                KEY_CREATE_SUB_KEY + KEY_ENUMERATE_SUB_KEYS + _
                KEY_NOTIFY + KEY_CREATE_LINK + READ_CONTROL


Const HKEY_LOCAL_MACHINE = &H80000002
Const ERROR_SUCCESS = 0
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
Const REG_SZ = 1                    ' Unicode nul terminated string
Const REG_DWORD = 4                 ' 32-bit number

Const gREGKEYSYSINFOLOC = "SOFTWARE\Microsoft\Shared Tools Location"
Const gREGVALSYSINFOLOC = "MSINFO"
Const gREGKEYSYSINFO = "SOFTWARE\Microsoft\Shared Tools\MSINFO"
Const gREGVALSYSINFO = "PATH"

Private Declare Function RegOpenKeyEx Lib "advapi32" Alias "RegOpenKeyExA" (ByVal
hKey As Long, ByVal lpSubKey As String, ByVal ulOptions As Long, ByVal samDesired
As Long, ByRef phkResult As Long) As Long
Private Declare Function RegQueryValueEx Lib "advapi32" Alias "RegQueryValueExA"
(ByVal hKey As Long, ByVal lpValueName As String, ByVal lpReserved As Long, ByRef
lpType As Long, ByVal lpData As String, ByRef lpcbData As Long) As Long
Private Declare Function RegCloseKey Lib "advapi32" (ByVal hKey As Long) As Long


Private Sub cmdSysInfo_Click()
  Call StartSysInfo
End Sub

Private Sub cmdOK_Click()
  Unload Me
  frmMain.Show
End Sub

Private Sub Form_Load()
    Me.Caption = "About " & App.Title
    lblVersion.Caption = "Version " & App.Major & "." & App.Minor & "." & App.Revision
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    lblTitle.Caption = App.Title
End Sub

Public Sub StartSysInfo()
    On Error GoTo SysInfoErr

    Dim rc As Long
    Dim SysInfoPath As String


    If GetKeyValue(HKEY_LOCAL_MACHINE, gREGKEYSYSINFO, gREGVALSYSINFO,
SysInfoPath) Then

    ElseIf GetKeyValue(HKEY_LOCAL_MACHINE, gREGKEYSYSINFOLOC,
gREGVALSYSINFOLOC, SysInfoPath) Then

      If (Dir(SysInfoPath & "\MSINFO32.EXE") <> "") Then
        SysInfoPath = SysInfoPath & "\MSINFO32.EXE"


      Else
        GoTo SysInfoErr
      End If

    Else
      GoTo SysInfoErr
    End If

    Call Shell(SysInfoPath, vbNormalFocus)
```

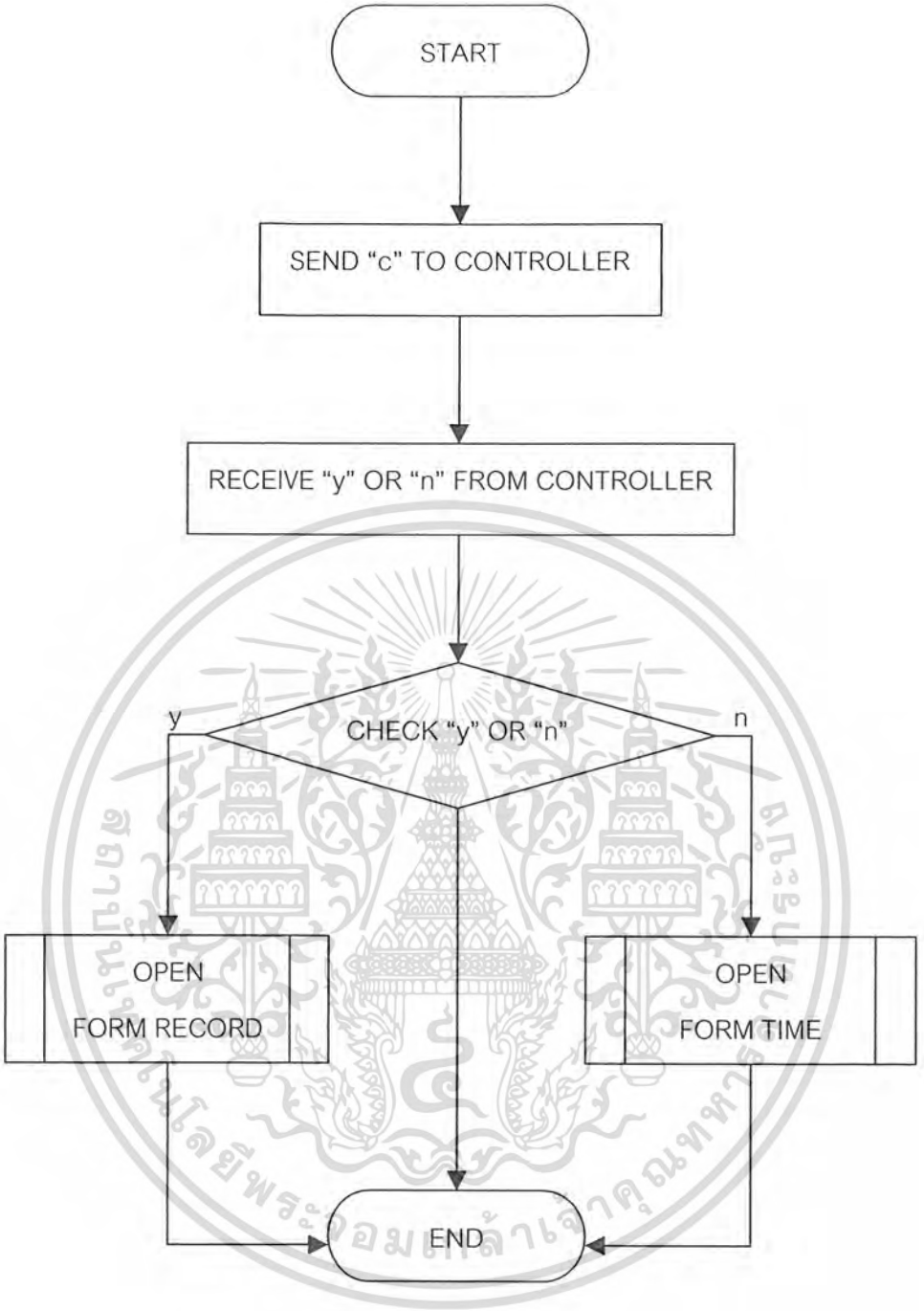
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    Exit Sub
SysInfoErr:
    MsgBox "System Information Is Unavailable At This Time", vbOKOnly
End Sub

Public Function GetKeyValue(KeyRoot As Long, KeyName As String, SubKeyRef As
String, ByRef KeyVal As String) As Boolean
    Dim i As Long
    Dim rc As Long
    Dim hKey As Long
    Dim hDepth As Long
    Dim KeyValType As Long
    Dim tmpVal As String
    Dim KeyValSize As Long

    rc = RegOpenKeyEx(KeyRoot, KeyName, 0, KEY_ALL_ACCESS, hKey)

    If (rc <> ERROR_SUCCESS) Then GoTo GetKeyError

    tmpVal = String$(1024, 0)
    KeyValSize = 1024

    rc = RegQueryValueEx(hKey, SubKeyRef, 0, _
                KeyValType, tmpVal, KeyValSize)

    If (rc <> ERROR_SUCCESS) Then GoTo GetKeyError

    If (Asc(Mid(tmpVal, KeyValSize, 1)) = 0) Then
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

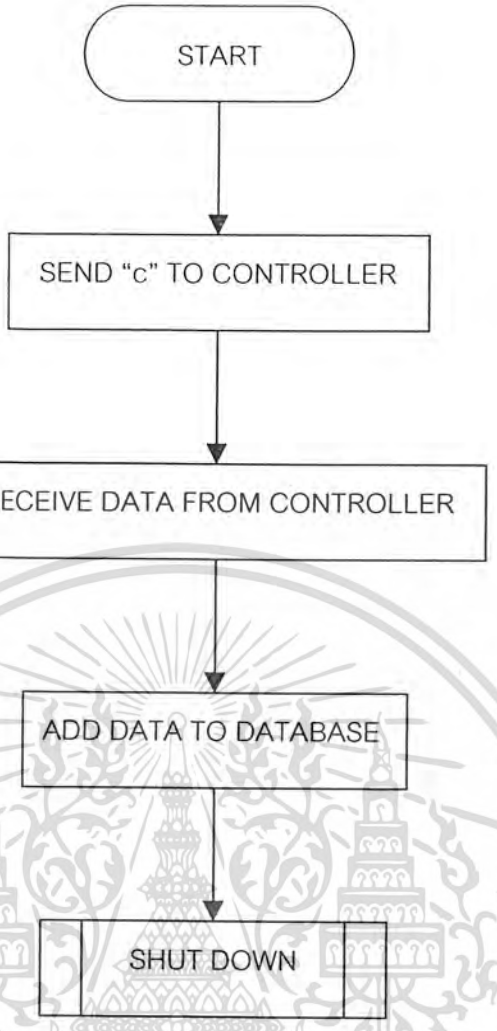
```
        tmpVal = Left(tmpVal, KeyValSize - 1)
    Else
        tmpVal = Left(tmpVal, KeyValSize)
    End If

    Select Case KeyValType
    Case REG_SZ
        KeyVal = tmpVal
    Case REG_DWORD
        For i = Len(tmpVal) To 1 Step -1
            KeyVal = KeyVal + Hex(Asc(Mid(tmpVal, i, 1)))
        Next
        KeyVal = Format$("&h" + KeyVal)
    End Select

    GetKeyValue = True
    rc = RegCloseKey(hKey)
    Exit Function

GetKeyError:
    KeyVal = ""
    GetKeyValue = False
    rc = RegCloseKey(hKey)
End Function
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



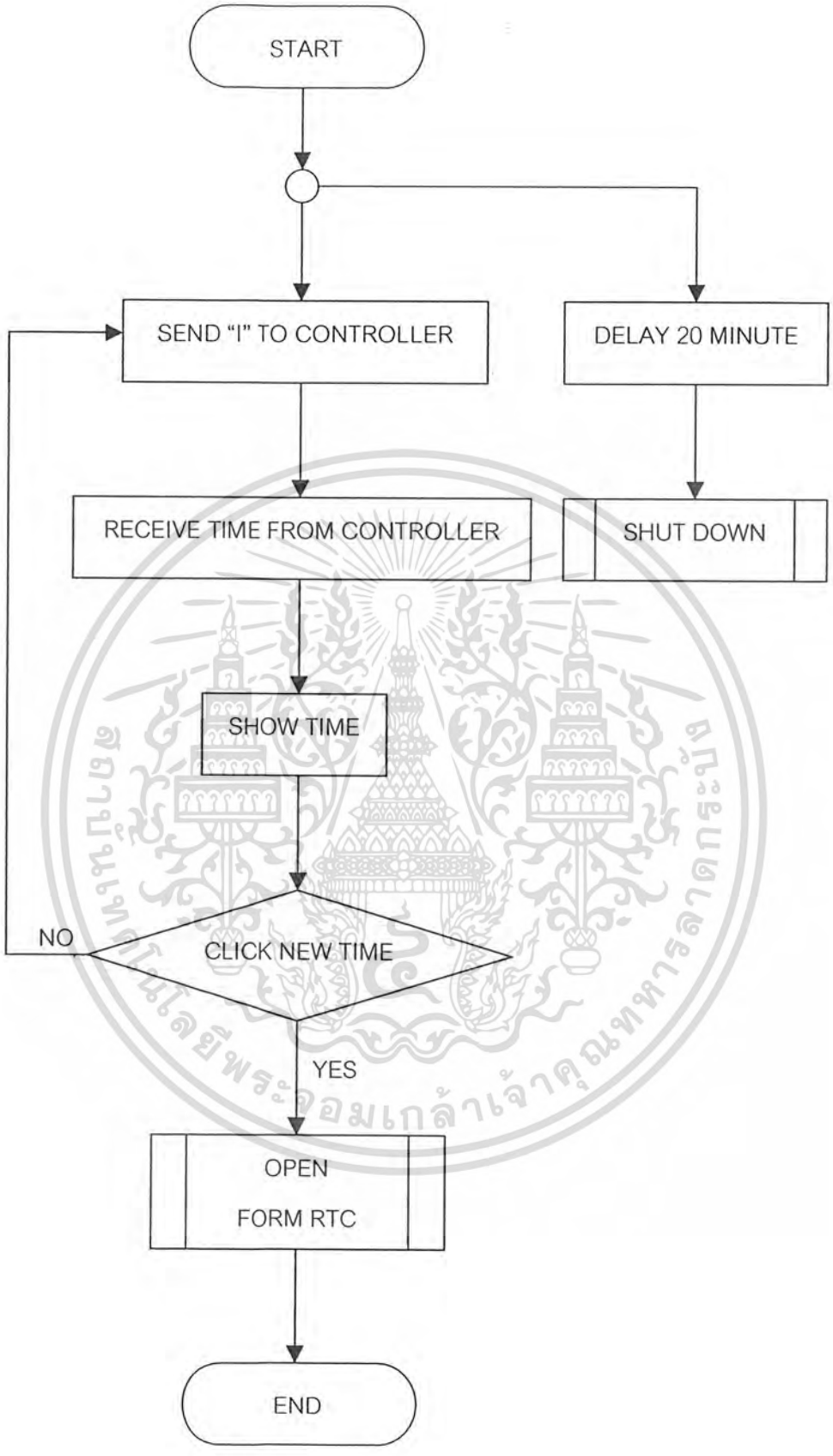
แผนผังแสดงการทำงานของโปรแกรมในส่วนของ Server Computer ที่หน้าต่าง Check

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



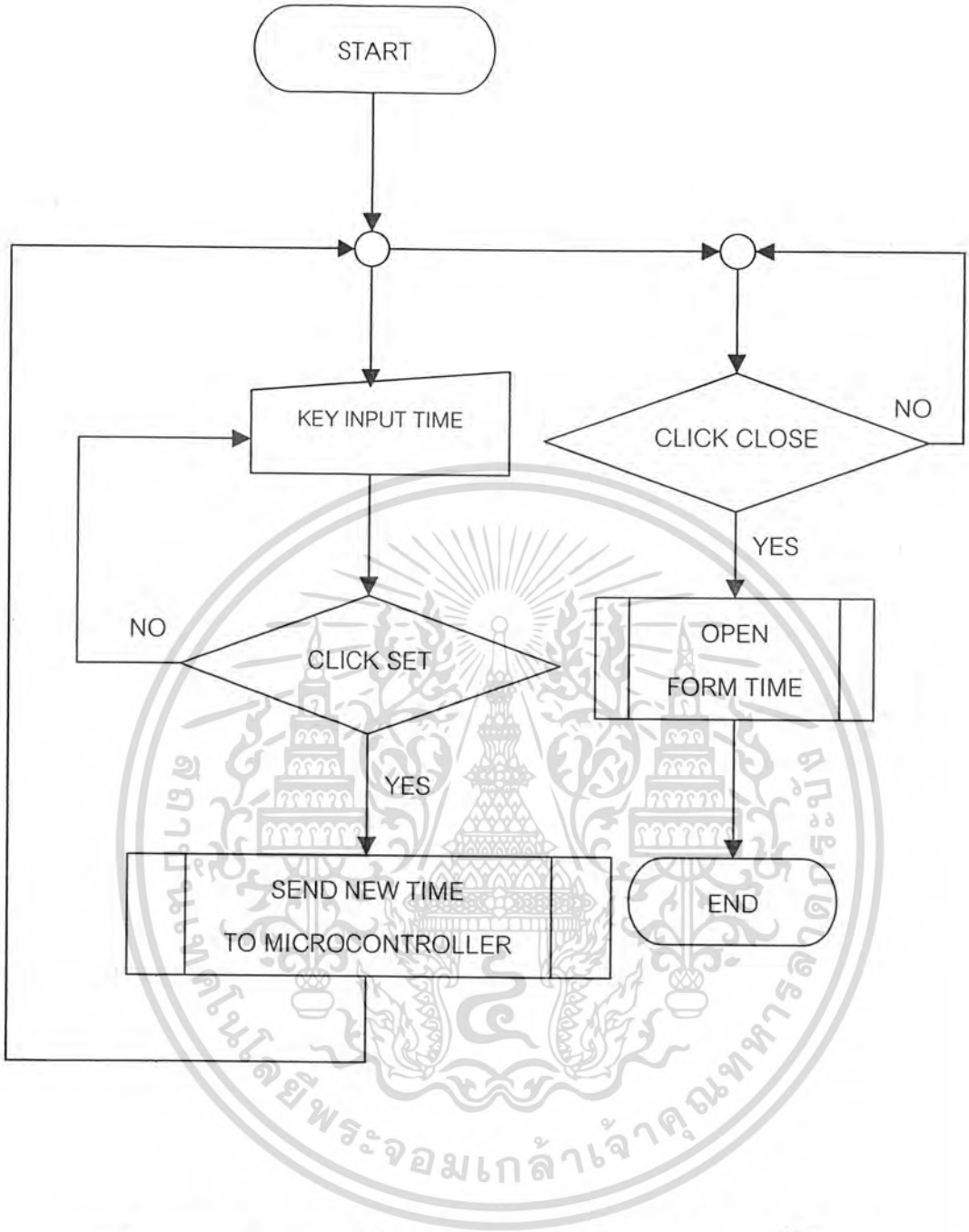
แผนผังแสดงการทำงานของโปรแกรมในส่วนของ Server Computer ที่หน้าต่าง Receive

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



แผนผังแสดงการทำงานของโปรแกรมในส่วนของ Server Computer ที่หน้าต่าง Time

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



แผนผังแสดงการทำงานของโปรแกรมในส่วนของ Server Computer ที่หน้าต่าง RTC

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

***** Module1.bas *****

****

```
Public Declare Function ExitWindowsEx Lib "user32" (ByVal uFlags As Long, ByVal
dwReserved As Long) As Long

Public Const EWX_SHUTDOWN = 1
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
***** frmCheck.frm *****
**************************
Private Sub Form_Activate()
If MSComm1.PortOpen = False Then
MSComm1.PortOpen = True
End If
MSComm1.Output = "e"
MSComm1.InBufferCount = 0
MSComm1.Output = "c"
Timer1.Enabled = True
End Sub

Private Sub Timer1_Timer()
If MSComm1.PortOpen = True Then
MSComm1.InputLen = 0
txtReceive.Text = MSComm1.Input
Timer1.Enabled = False
End If
End Sub

Private Sub txtReceive_Change()
If txtReceive.Text = "y" Then
frmRecord.Show
Unload Me
End If
If txtReceive.Text = "n" Then
frmSetTime.Show
Unload Me
End If
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
End Sub



***** frmRecord.frm *****
**************************
Private Sub Data1_Validate(Action As Integer, Save As Integer)
End Sub

Private Sub Form_Activate()
If MSComm1.PortOpen = False Then
MSComm1.PortOpen = True
End If
MSComm1.InBufferCount = 0
MSComm1.Output = "r"
Timer1.Enabled = True
End Sub

Private Sub Form_Load()
Data1.DatabaseName = App.Path & "\Raindata.mdb"
End Sub

Private Sub Timer1_Timer()

If MSComm1.PortOpen = True Then
MSComm1.InputLen = 0
txtReceive.Text = MSComm1.Input
Timer1.Enabled = False
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
End If

End Sub

Private Sub txtReceive_Change()
Dim b As Integer
Dim n As Integer
Dim i As Integer
Dim a As Integer
Dim X As Integer
Dim p(27) As Integer
b = 0
n = Len(txtReceive.Text)
For i = 1 To n
a = Asc(Mid(txtReceive.Text, i, 1))
   If a = 44 Then
   b = b + 1
   p(b) = i
   End If
Next i
If b = 27 Then
   Dim Day As Integer
   Dim Month As Integer
   Dim Year As Integer
   Dim time As Integer
   Dim RainAmount As Single
   Day = Val(Mid(txtReceive.Text, p(24) + 1, p(25) - p(24) - 1))
   Month = Val(Mid(txtReceive.Text, p(25) + 1, p(26) - p(25) - 1))
   Year = Val(Mid(txtReceive.Text, p(26) + 1, p(27) - p(26) - 1))
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
End If

End Sub

Private Sub txtReceive_Change()
Dim b As Integer
Dim n As Integer
Dim i As Integer
Dim a As Integer
Dim X As Integer
Dim p(27) As Integer
b = 0
n = Len(txtReceive.Text)
For i = 1 To n
a = Asc(Mid(txtReceive.Text, i, 1))
  If a = 44 Then
  b = b + 1
  p(b) = i
  End If
Next i
If b = 27 Then
  Dim Day As Integer
  Dim Month As Integer
  Dim Year As Integer
  Dim time As Integer
  Dim RainAmount As Single
  Day = Val(Mid(txtReceive.Text, p(24) + 1, p(25) - p(24) - 1))
  Month = Val(Mid(txtReceive.Text, p(25) + 1, p(26) - p(25) - 1))
  Year = Val(Mid(txtReceive.Text, p(26) + 1, p(27) - p(26) - 1))
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
p(0) = 0
For i = 0 To 23
  RainAmount = Val(Mid(txtReceive.Text, p(i) + 1, p(i + 1) - p(i) - 1)) * 0.25
  Data1.Recordset.AddNew
  Data1.Recordset!Day = Day
  Data1.Recordset!Month = Month
  Data1.Recordset!Year = Year

  Select Case i
    Case 0
    Data1.Recordset!time = "07.00-08.00"
    Case 1
    Data1.Recordset!time = "08.00-09.00"
    Case 2
    Data1.Recordset!time = "09.00-10.00"
    Case 3
    Data1.Recordset!time = "10.00-11.00"
    Case 4
    Data1.Recordset!time = "11.00-12.00"
    Case 5
    Data1.Recordset!time = "12.00-13.00"
    Case 6
    Data1.Recordset!time = "13.00-14.00"
    Case 7
    Data1.Recordset!time = "14.00-15.00"
    Case 8
    Data1.Recordset!time = "15.00-16.00"
    Case 9
    Data1.Recordset!time = "16.00-17.00"
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    Case 10
    Data1.Recordset!time = "17.00-18.00"
    Case 11
    Data1.Recordset!time = "18.00-19.00"
    Case 12
    Data1.Recordset!time = "19.00-20.00"
    Case 13
    Data1.Recordset!time = "20.00-21.00"
    Case 14
    Data1.Recordset!time = "21.00-22.00"
    Case 15
    Data1.Recordset!time = "22.00-23.00"
    Case 16
    Data1.Recordset!time = "23.00-24.00"
    Case 17
    Data1.Recordset!time = "24.00-01.00"
    Case 18
    Data1.Recordset!time = "01.00-02.00"
    Case 19
    Data1.Recordset!time = "02.00-03.00"
    Case 20
    Data1.Recordset!time = "03.00-04.00"
    Case 21
    Data1.Recordset!time = "04.00-05.00"
    Case 22
    Data1.Recordset!time = "05.00-06.00"
    Case 23
    Data1.Recordset!time = "06.00-07.00"
End Select
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        Data1.Recordset!RainAmount = RainAmount
        Data1.Recordset.Update
        Data1.Recordset.Bookmark = Data1.Recordset.LastModified
    Next i
    Call ExitWindowsEx(EWX_SHUTDOWN, 1)
End If
End Sub


***** FrmRTC.frm *****
************************

Private Sub cmdClose_Click()
If MSComm1.PortOpen = False Then
MSComm1.PortOpen = True
End If
MSComm1.Output = "e"
Timer1.Enabled = True
frmSetTime.Show
Unload Me
End Sub

Private Sub cmdSet_Click()

cmdSet.Enabled = False

If MSComm1.PortOpen = False Then
MSComm1.PortOpen = True
End If
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
MSComm1.Output = "s"
Timer1.Enabled = True

Dim Year As String
Dim Month As String
Dim Day As String
Dim Hour As String
Dim Minute As String
Dim Second As String

  Year = Mid(DTPicker1.Year, 3, 2)

  If Len(DTPicker1.Month) = 1 Then
  Month = "0" & DTPicker1.Month
  Else: Month = DTPicker1.Month
  End If

  If Len(DTPicker1.Day) = 1 Then
  Day = "0" & DTPicker1.Day
  Else: Day = DTPicker1.Day
  End If

  If Len(txtHour.Text) = 1 Then
  Hour = "0" & txtHour.Text
  Else: Hour = txtHour.Text
  End If

  If Len(txtMinute.Text) = 1 Then
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
Minute = "0" & txtMinute.Text
Else: Minute = txtMinute.Text
End If

If Len(txtSecond.Text) = 1 Then
Second = "0" & txtSecond.Text
Else: Second = txtSecond.Text
End If

txtSendTime.Text = Year & Month & Day & Hour & Minute & Second

MSComm1.Output = txtSendTime.Text
Timer1.Enabled = True

End Sub

Private Sub Timer1_Timer()
If MSComm1.PortOpen = True Then
MSComm1.InputLen = 0

Timer1.Enabled = False
End If
End Sub

Private Sub txtHour_Change()
If Val(txtHour.Text) > 24 Then
If MsgBox("invalid Hour", 48, "Time Error") = vbOK Then
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
txtHour.Text = ""
txtHour.SetFocus
End If
End If
End Sub

Private Sub txtMinute_Change()
If Val(txtMinute.Text) > 60 Then
If MsgBox("invalid Minute", 48, "Time Error") = vbOK Then
txtMinute.Text = ""
txtMinute.SetFocus
End If
End If
End Sub

Private Sub txtSecond_Change()
If Val(txtSecond.Text) > 60 Then
If MsgBox("invalid Second", 48, "Time Error") = vbOK Then
txtSecond.Text = ""
txtSecond.SetFocus
End If
End If
End Sub


***** frmSettime.frm *****
*************************
Private Sub cmdExit_Click()
Timer1.Enabled = False
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
If MSComm1.PortOpen = True Then
MSComm1.Output = "e"
End If
End
End Sub

Private Sub cmdNewTime_Click()

MSComm1.InBufferCount = 0
MSComm1.Output = "e"
Timer1.Enabled = True

frmRTC.Show
Unload Me
End Sub

Private Sub Form_Activate()
txtStartTime.Text = time$
Timer2.Enabled = True
If MSComm1.PortOpen = False Then
MSComm1.PortOpen = True
End If
MSComm1.Output = "e"
End Sub
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
If MSComm1.PortOpen = True Then
MSComm1.Output = "e"
End If
End
End Sub

Private Sub cmdNewTime_Click()

MSComm1.InBufferCount = 0
MSComm1.Output = "e"
Timer1.Enabled = True

frmRTC.Show
Unload Me
End Sub

Private Sub Form_Activate()
txtStartTime.Text = time$
Timer2.Enabled = True
If MSComm1.PortOpen = False Then
MSComm1.PortOpen = True
End If
MSComm1.Output = "e"
End Sub
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
Private Sub Timer1_Timer()
If MSComm1.PortOpen = True Then
MSComm1.InputLen = 0
txtReceive.Text = MSComm1.Input
Timer1.Enabled = False
End If
End Sub

Private Sub Timer2_Timer()
txtTime1.Text = time$
End Sub

Private Sub txtReceive_Change()
Dim Day As String
Dim Month As String
Dim Year As String
Dim Hour As String
Dim Minute As String
Dim Second As String
Dim c As String
Dim b As Integer
Dim n As Integer
Dim i As Integer
Dim a As Integer
Dim p(7) As Integer
b = 0
n = Len(txtReceive.Text)
For i = 1 To n
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    If Len(Minute) = 1 Then
    c = Minute
    Minute = "0" & c
    End If

    Second = Mid(txtReceive.Text, p(0) + 1, p(1) - p(0) - 1)
    If Len(Second) = 1 Then
    c = Second
    Second = "0" & c
    End If
    txtTime.Text = Hour & ":" & Minute & ":" & Second
    End If
End Sub

Private Sub txtTime1_Change()
Static m1 As Integer
Static s1 As Integer
Dim m2 As String
Dim s2 As String
Dim c As String

If cmdNewTime.Enabled = True Then
If MSComm1.PortOpen = False Then
MSComm1.PortOpen = True
End If
MSComm1.InBufferCount = 0
MSComm1.Output = "I"
Timer1.Enabled = True
End If
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
s1 = s1 + 1
If s1 > 59 Then
m1 = m1 + 1
s1 = 0
If m1 = 24 Then
m1 = 0
End If
End If

m2 = 19 - m1
If Len(m2) = 1 Then
c = m2
m2 = "0" & c
End If

s2 = 59 - s1
If Len(s2) = 1 Then
c = s2
s2 = "0" & c
End If

txtCountDown.Text = m2 & ":" & s2

   If m2 = 0 And s2 = 0 Then
   Call ExitWindowsEx(EWX_SHUTDOWN, 1)
   End If
End Sub
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก ค.

วิธีการติดตั้งการติดต่อฐานข้อมูลจากระยะไกล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การติดตั้งเพื่อทำการติดต่อกับเครื่องคอมพิวเตอร์ระยะไกล

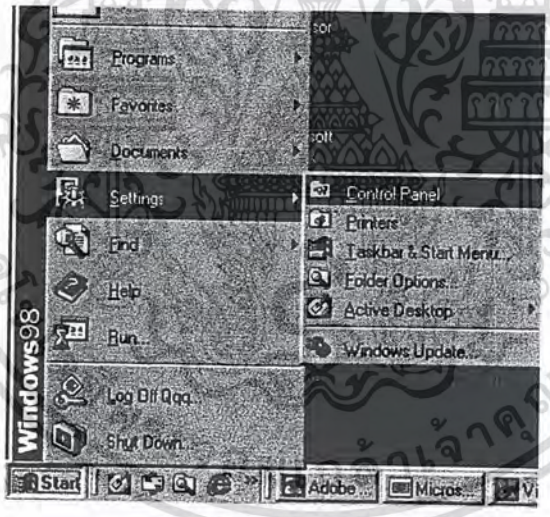
เมื่อเราติดตั้งระบบ Windows 98 แล้ว เราจะต้องเพิ่มโปรแกรม Dial –up Server เข้าไปเพื่อสามารถให้ใช้การเป็นเครื่อง Server ได้ โดยสามารถให้บุคคลภายนอกติดต่อเข้ามา โดยมีการกำหนด พาสเวิร์ด

ที่เครื่องให้บริการ (Server)

เมื่อเรากำหนดยูสเซอร์และพาสเวิร์ดของเครื่องคอมพิวเตอร์ที่มาใช้บริการแล้ว เพื่อให้ยูสเซอร์ที่เป็นเครื่องคอมพิวเตอร์ระยะไกลสามารถติดต่อเข้ามาได้จะต้องมาทำงานที่ Dial-up Server cแล้วเปิดโปรแกรมนี้ไว้ โดยที่เราต้องต่อโมเด็มไว้กับเครื่อง

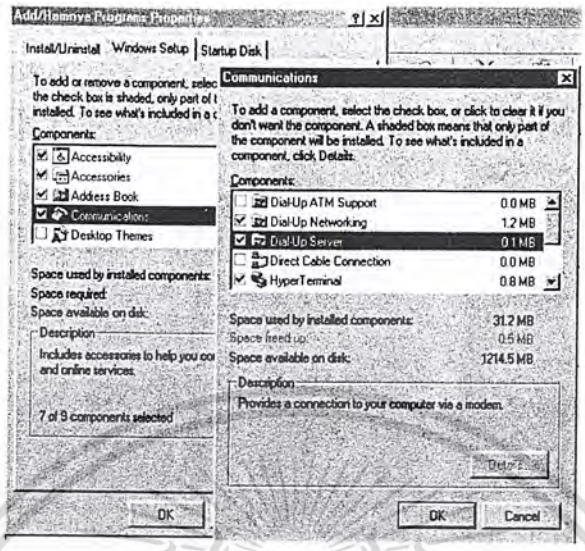
วิธีการติดตั้ง Dial-up Server

เข้าไปที่ Start menu ไปที่ Control Panel ดังรูป

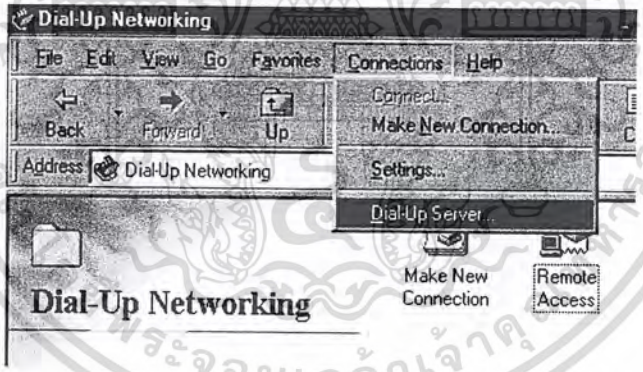


โดยให้มาที่ Control Panel และเลือกทำงานที่ Add Remove โปรแกรมในหน้าต่าง Add Remove โปรแกรม มาทำงานที่ Tab Window Setup เลือก Communication และเลือก Dial-up Server ดังแสดงในรูป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

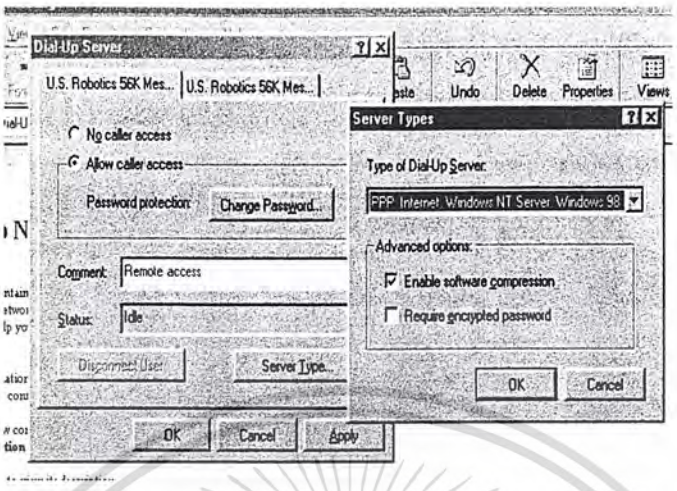


เมื่อติดตั้งเรียบร้อยแล้วให้ทำการรันโปรแกรม Dial-up Server โดยเข้าไปที่ My computer เลือก Dial-up Networking ที่เมนู Connections เลือก Dial-up Server ดังรูป



เมื่อโปรแกรมเปิดขึ้นมาให้เราคลิกที่ Allow Caller Access เพื่อให้ผู้อื่นสามารถเข้ามาที่เครื่อง Server ได้ แล้ว เซ็ตค่า Server Type ดังรูป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

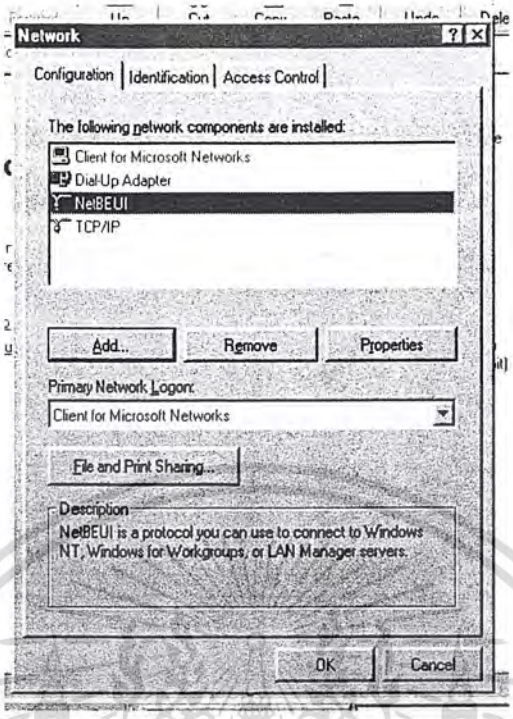


เสร็จแล้วจะมี Icon ที่ Tark bar แสดงสถานะของเครื่อง Server ดังรูป

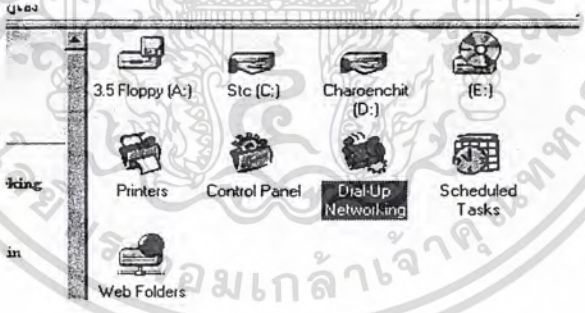
ที่เครื่องคอมพิวเตอร์ระยะไกล

ระบบปฏิบัติการจะเป็น Windows 95 หรือ Windows 98 ก็ได้ จะต้องมี โมเด็ม ติดตั้ง นอกการปรับปรุงให้มาที่ Control Panel และทำงานที่เน็ตเวิร์คโดยให้ติดตั้ง Client for Microsoft Network ติดตั้ง Dial-up Adapter และติดตั้ง Protocol NetBEUI ที่เราจะเป็นสื่อกลางสำหรับติดต่อเข้ามาที่เครื่องที่ให้บริการที่เป็น Windows 98 ดังรูป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

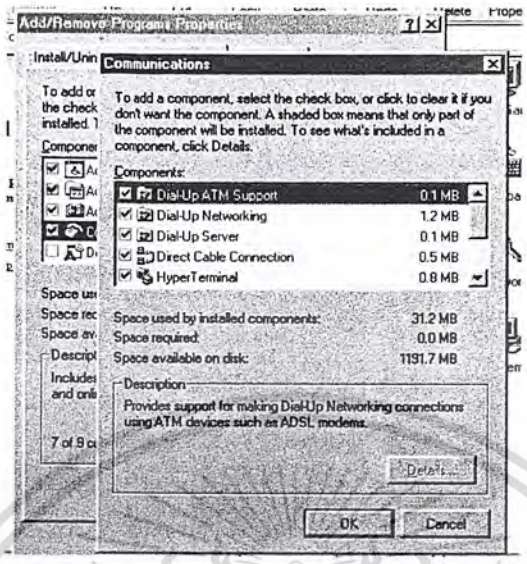


ต่อมาต้องมาที่ My Computer ที่อยู่บนหน้าจอของเรา และดูว่ามีโปรแกรม Dial-up Networking อยู่หรือไม่ดังแสดงในรูป

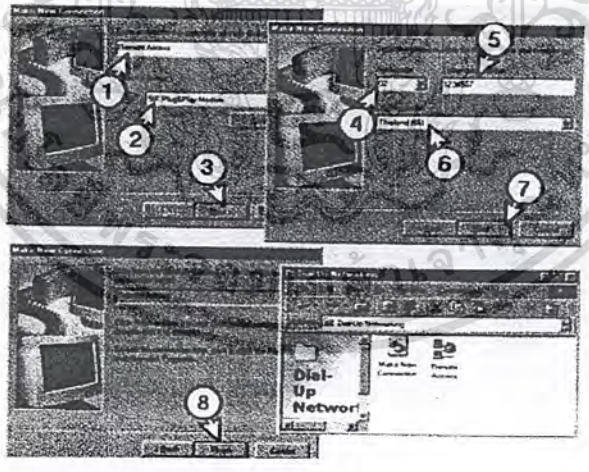


ถ้าไม่มีก็ต้องติดตั้งใหม่ โดยให้มาที่ Control Panel และเลือกทำงานที่ Add Remove โปรแกรมในหน้าต่าง Add Remove โปรแกรม มาทำงานที่ Tab Window Setup เลือก Communication และเลือก Dial-up Network ดังแสดงในรูป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

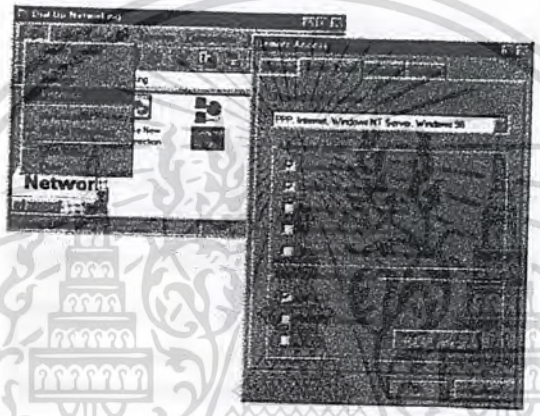


เข้าไปทำงานที่ Dial-up Network และเตรียมการสร้างการติดต่อใหม่ ซึ่งจะมี Wizard ช่วยให้เราติดตั้ง คือให้ตั้งชื่อการติดต่อ ชื่อ Remote Access บันทึกโทรศัพท์ รหัสจังหวัด และประเทศ เลือกโมเด็ม และคลิก OK ดังรูป



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จะได้การติดต่อใหม่ในหน้าต่าง Dial-up Network ที่ชื่อ Remote Access ต่อไปจะต้องมาปรับปรุง Property ของการติดต่อใหม่นี้ โดยการคลิกให้ Active และมาที่เมนูไฟล์ คลิกเลือก Property ที่หน้าต่าง Remote Access Property มาคลิกบัตทอน Server Type เพื่อปรับปรุงชนิดของ Server ที่ต้องการติดต่อ และเลือกสื่อกลางที่นำพาการติดต่อ คือให้เลือก Log on to Network และเลือก Enable Software Compress ส่วนที่ Protocol ให้เลือกตัวกลางที่เดินทางคือ NetBEUI และคลิก OK ดังแสดง ในรูป ก็พร้อมที่จะติดต่อมายังเครื่องคอมพิวเตอร์ที่ให้บริการที่เป็น Windows 98



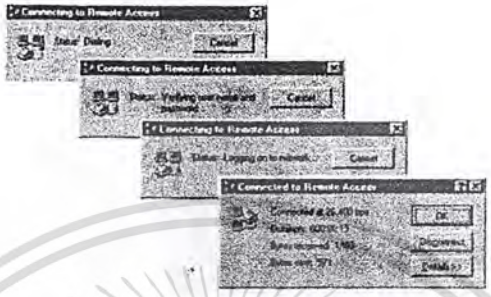
การติดต่อ

การติดต่อเข้ามาใช้บริการ ในเครื่องให้บริการด้วย Dial-up Network เมื่อเราติดตั้งทุกอย่างเรียบร้อยแล้ว เราต้องขอชื่อยูสเซอร์ และพาสเวิร์ดที่เรากำหนดมาจาก เครื่อง Server ก็พร้อมที่จะติดต่อมายังเครื่องให้บริการได้โดยมีขั้นตอนดังนี้

1. ให้เตรียมเปิดเครื่องใหม่ เมื่อมีหน้าต่างให้บันทึก Username และ Password ให้เราบันทึกตามที่ กำหนดมา และจะต้องเป็นยูสเซอร์ที่ถูก Permission แล้ว
2. เมื่อผ่านเข้าการทำงานใน Windows ได้แล้ว ให้มาที่ My Computer และเข้ามาที่หน้าต่าง Dial-up Network และมาคลิกที่ชื่อการติดต่อที่ชื่อ Remote Access ที่ได้สร้างไว้ ก็จะได้หน้าต่างสำหรับหมุนโทรศัพท์ ที่หน้าต่างนี้ให้เราบันทึกชื่อ Username และ Password อีกครั้งหนึ่ง เหมือนในตอนที่เปิดเครื่องใหม่ และควร Save ไว้ด้วย ส่วนเบอร์โทรศัพท์จะมีมาให้แล้วตอนที่เราสร้างการติดต่อใหม่ สุดท้ายก็ให้คลิก OK เพื่อหมุนโทรศัพท์เพื่อติดต่อมาที่ Remote Access ของ เครื่องที่ให้บริการ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การติดต่อจะมีรูปแบบเหมือนกับการติดต่อไปยังศูนย์บริการอินเตอร์เน็ตทุกประการ โดยจะมีการตรวจสอบ รหัสผ่าน เมื่อรหัส และ Username ที่กำหนดใน User Manager ถูกต้องก็จะมีการติดต่อสำเร็จ ดังรูป



การตรวจสอบการติดต่อระยะไกล

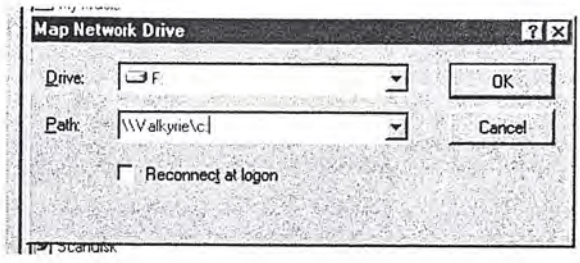
เมื่อเราติดต่อได้แล้วจะมีหน้าต่างแสดงเวลาที่เราติดต่อเหมือนกับไปยังศูนย์บริการอินเตอร์เน็ต เมื่อเรา Minimum หน้าต่างนี้ก็จะมาอยู่ที่ Tark Bar ด้านขวามือ และจะต้องตรวจสอบว่า เราสามารถติดต่อกับเครื่องที่ให้บริการได้หรือไม่ โดยให้มาคลิกขวาที่ Network Neighborhood เลือก Find Computer เครื่องจะมองเห็นเครื่อง Server โดยเราใส่ชื่อเครื่อง Server ในช่องค้นหา

การ SetUp Server

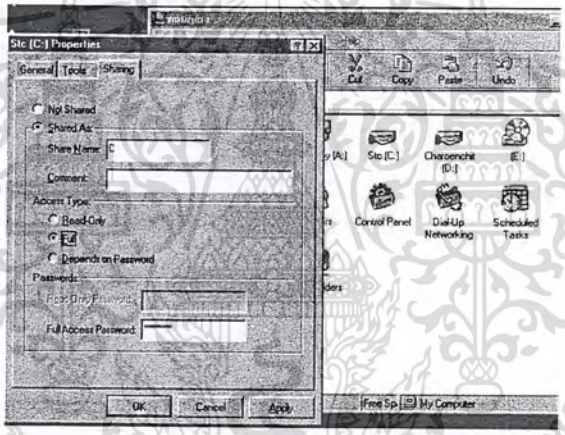
ที่เครื่องลูกข่าย หรือเครื่องที่ใช้บริการนั้น เราจะต้องทำให้เครื่องของเรารู้จัก File Server ที่อยู่ในเครื่องให้บริการก่อน โดยเราต้องลงทะเบียนที่ File ที่เป็น Server

เราจะต้อง Connect ไปยังเครื่องคอมพิวเตอร์ ที่ให้บริการโดยเรียกโปรแกรมWindow Explorer ทำงานและ Click ที่เมนู Tool , Click เมนู Connect ที่หน้าต่าง Connect ให้เลือกทำงานที่เครื่องคอมพิวเตอร์ที่ให้บริการ หรือให้บันทึกเป็น **\\ชื่อเครื่องคอมพิวเตอร์** \Drive เช่น ถ้าเครื่องที่ให้บริการมีชื่อเป็น Valkyrie ก็ให้บันทึกเป็น \\Valkyrie\C และ Click OK ดังในรูปในที่นี้เราจะให้เป็น Drive F:

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



และที่เครื่อง Server นั้นจะต้องจะต้องติดตั้ง Protocol NetBEUI ไว้ด้วย รวมทั้งการ Sharing Drive ที่จะให้เครื่อง Client นั้นสามารถดึงข้อมูลมาได้



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก ง.

เอกสารประกอบการใช้งาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Features

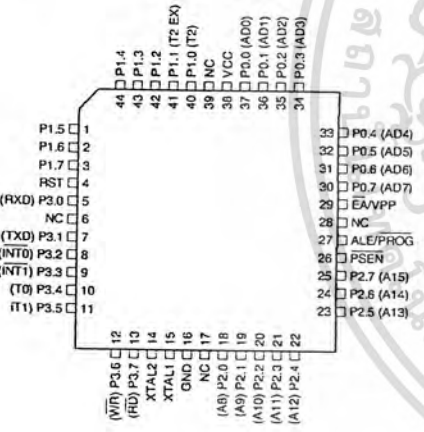
- Compatible with MCS-51™ Products
- 8K Bytes of In-System Reprogrammable Flash Memory
- Endurance: 1,000 Write/Erase Cycles
- Fully Static Operation: 0 Hz to 24 MHz
- Three-level Program Memory Lock
- 256 x 8-bit Internal RAM
- 32 Programmable I/O Lines
- Three 16-bit Timer/Counters
- Eight Interrupt Sources
- Programmable Serial Channel
- Low-power Idle and Power-down Modes

Description

The AT89C52 is a low-power, high-performance CMOS 8-bit microcomputer with 8K bytes of Flash programmable and erasable read only memory (PEROM). The device is manufactured using Atmel's high-density nonvolatile memory technology and is compatible with the industry-standard 80C51 and 80C52 instruction set and pinout. The on-chip Flash allows the program memory to be reprogrammed in-system or by a conventional nonvolatile memory programmer. By combining a versatile 8-bit CPU with Flash on a monolithic chip, the Atmel AT89C52 is a powerful microcomputer which provides a highly-flexible and cost-effective solution to many embedded control applications.

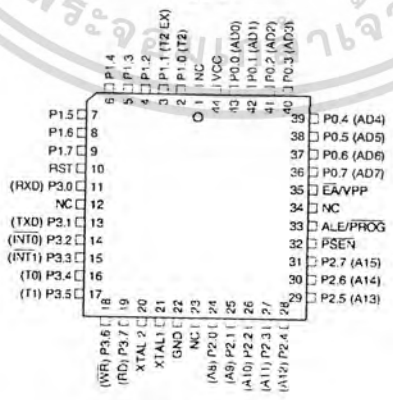
Pin Configurations

PQFP/TQFP



PDIP

PLCC



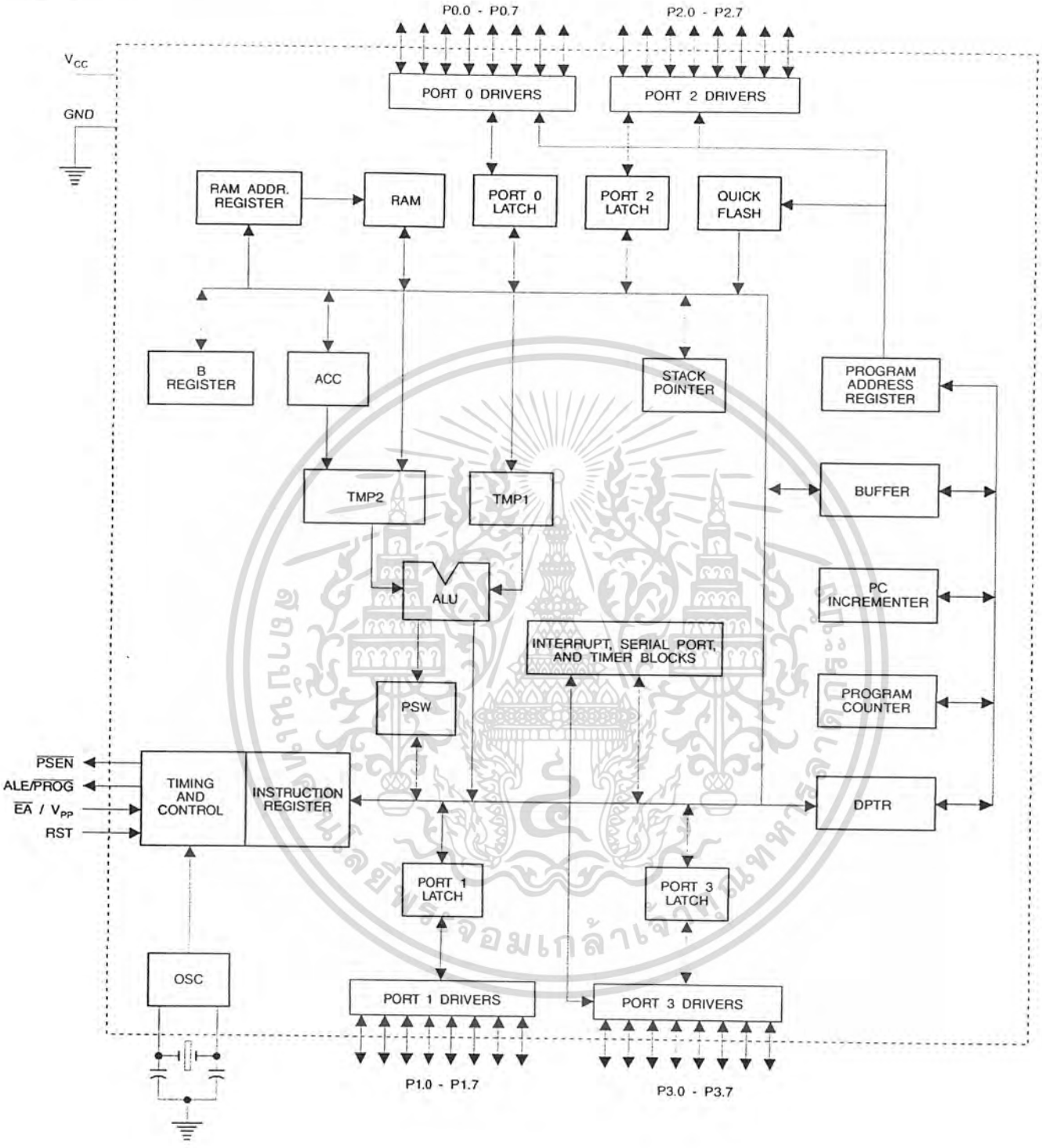
8-bit Microcontroller with 8K Bytes Flash

AT89C52

Rev. 0313H–02/00

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Block Diagram



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

The AT89C52 provides the following standard features: 8K bytes of Flash, 256 bytes of RAM, 32 I/O lines, three 16-bit timer/counters, a six-vector two-level interrupt architecture, a full-duplex serial port, on-chip oscillator, and clock circuitry. In addition, the AT89C52 is designed with static logic for operation down to zero frequency and supports two software selectable power saving modes. The Idle Mode stops the CPU while allowing the RAM, timer/counters, serial port, and interrupt system to continue functioning. The Power-down mode saves the RAM contents but freezes the oscillator, disabling all other chip functions until the next hardware reset.

Pin Description

VCC

Supply voltage.

GND

Ground.

Port 0

Port 0 is an 8-bit open drain bi-directional I/O port. As an output port, each pin can sink eight TTL inputs. When 1s are written to port 0 pins, the pins can be used as high-impedance inputs.

Port 0 can also be configured to be the multiplexed low-order address/data bus during accesses to external program and data memory. In this mode, P0 has internal pullups.

Port 0 also receives the code bytes during Flash programming and outputs the code bytes during program verification. External pullups are required during program verification.

Port 1

Port 1 is an 8-bit bi-directional I/O port with internal pullups. The Port 1 output buffers can sink/source four TTL inputs. When 1s are written to Port 1 pins, they are pulled high by the internal pullups and can be used as inputs. As inputs, Port 1 pins that are externally being pulled low will source current (I_{IL}) because of the internal pullups.

In addition, P1.0 and P1.1 can be configured to be the timer/counter 2 external count input (P1.0/T2) and the timer/counter 2 trigger input (P1.1/T2EX), respectively, as shown in the following table.

Port 1 also receives the low-order address bytes during Flash programming and verification.

Port Pin	Alternate Functions
P1.0	T2 (external count input to Timer/Counter 2), clock-out
P1.1	T2EX (Timer/Counter 2 capture/reload trigger and direction control)

Port 2

Port 2 is an 8-bit bi-directional I/O port with internal pullups. The Port 2 output buffers can sink/source four TTL inputs. When 1s are written to Port 2 pins, they are pulled high by the internal pullups and can be used as inputs. As inputs, Port 2 pins that are externally being pulled low will source current (I_{IL}) because of the internal pullups.

Port 2 emits the high-order address byte during fetches from external program memory and during accesses to external data memory that use 16-bit addresses (MOVX @ DPTR). In this application, Port 2 uses strong internal pullups when emitting 1s. During accesses to external data memory that use 8-bit addresses (MOVX @ RI), Port 2 emits the contents of the P2 Special Function Register.

Port 2 also receives the high-order address bits and some control signals during Flash programming and verification.

Port 3

Port 3 is an 8-bit bi-directional I/O port with internal pullups. The Port 3 output buffers can sink/source four TTL inputs. When 1s are written to Port 3 pins, they are pulled high by the internal pullups and can be used as inputs. As inputs, Port 3 pins that are externally being pulled low will source current (I_{IL}) because of the pullups.

Port 3 also serves the functions of various special features of the AT89C51, as shown in the following table.

Port 3 also receives some control signals for Flash programming and verification.

Port Pin	Alternate Functions
P3.0	RXD (serial input port)
P3.1	TXD (serial output port)
P3.2	$\overline{INT0}$ (external interrupt 0)
P3.3	$\overline{INT1}$ (external interrupt 1)
P3.4	T0 (timer 0 external input)
P3.5	T1 (timer 1 external input)
P3.6	$\overline{WR}$ (external data memory write strobe)
P3.7	$\overline{RD}$ (external data memory read strobe)

RST

Reset input. A high on this pin for two machine cycles while the oscillator is running resets the device.

ALE/$\overline{PROG}$

Address Latch Enable is an output pulse for latching the low byte of the address during accesses to external memory. This pin is also the program pulse input ($\overline{PROG}$) during Flash programming.

In normal operation, ALE is emitted at a constant rate of 1/6 the oscillator frequency and may be used for external

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

iming or clocking purposes. Note, however, that one ALE pulse is skipped during each access to external data memory.

f desired, ALE operation can be disabled by setting bit 0 of SFR location 8EH. With the bit set, ALE is active only during a MOVX or MOVC instruction. Otherwise, the pin is weakly pulled high. Setting the ALE-disable bit has no effect if the microcontroller is in external execution mode.

PSEN

Program Store Enable is the read strobe to external program memory.

When the AT89C52 is executing code from external program memory, $\overline{PSEN}$ is activated twice each machine cycle, except that two $\overline{PSEN}$ activations are skipped during each access to external data memory.

$\overline{EA}$/VPP

External Access Enable. $\overline{EA}$ must be strapped to GND in order to enable the device to fetch code from external program memory locations starting at 0000H up to FFFFH. Note, however, that if lock bit 1 is programmed, $\overline{EA}$ will be internally latched on reset.

$\overline{EA}$ should be strapped to V_{CC} for internal program executions.

This pin also receives the 12-volt programming enable voltage (V_{PP}) during Flash programming when 12-volt programming is selected.

XTAL1

Input to the inverting oscillator amplifier and input to the internal clock operating circuit.

XTAL2

Output from the inverting oscillator amplifier.

Table 1. AT89C52 SFR Map and Reset Values

0F8H							0FFH
0F0H	B 00000000						0F7H
0E8H							0EFH
0E0H	ACC 00000000						0E7H
0D8H							0DFH
0D0H	PSW 00000000						0D7H
0C8H	T2CON 00000000	T2MOD XXXXXX00	RCAP2L 00000000	RCAP2H 00000000	TL2 00000000	TH2 00000000	0CFH
0C0H							0C7H
0B8H	IP XX000000						0BFH
0B0H	P3 11111111						0B7H
0A8H	IE 0X000000						0AFH
0A0H	P2 11111111						0A7H
98H	SCON 00000000	SBUF XXXXXXXX					9FH
90H	P1 11111111						97H
88H	TCON 00000000	TMOD 00000000	TL0 00000000	TL1 00000000	TH0 00000000	TH1 00000000	8FH
80H	P0 11111111	SP 00000111	DPL 00000000	DPH 00000000		PCON 0XXX0000	87H

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Special Function Registers

A map of the on-chip memory area called the Special Function Register (SFR) space is shown in Table 1.

Note that not all of the addresses are occupied, and unoccupied addresses may not be implemented on the chip. Read accesses to these addresses will in general return random data, and write accesses will have an indeterminate effect.

User software should not write 1s to these unlisted locations, since they may be used in future products to invoke new features. In that case, the reset or inactive values of the new bits will always be 0.

Timer 2 Registers Control and status bits are contained in registers T2CON (shown in Table 2) and T2MOD (shown in Table 4) for Timer 2. The register pair (RCAP2H, RCAP2L) are the Capture/Reload registers for Timer 2 in 16-bit capture mode or 16-bit auto-reload mode.

Interrupt Registers The individual interrupt enable bits are in the IE register. Two priorities can be set for each of the six interrupt sources in the IP register.r

Table 2. T2CON – Timer/Counter 2 Control Registe

T2CON Address = 0C8H — Reset Value = 0000 0000B

Bit Addressable

Bit	TF2	EXF2	RCLK	TCLK	EXEN2	TR2	C/$\overline{T2}$	CP/$\overline{RL2}$
	7	6	5	4	3	2	1	0

Symbol	Function
TF2	Timer 2 overflow flag set by a Timer 2 overflow and must be cleared by software. TF2 will not be set when either RCLK = 1 or TCLK = 1.
EXF2	Timer 2 external flag set when either a capture or reload is caused by a negative transition on T2EX and EXEN2 = 1. When Timer 2 interrupt is enabled, EXF2 = 1 will cause the CPU to vector to the Timer 2 interrupt routine. EXF2 must be cleared by software. EXF2 does not cause an interrupt in up/down counter mode (DCEN = 1).
RCLK	Receive clock enable. When set, causes the serial port to use Timer 2 overflow pulses for its receive clock in serial port Modes 1 and 3. RCLK = 0 causes Timer 1 overflow to be used for the receive clock.
TCLK	Transmit clock enable. When set, causes the serial port to use Timer 2 overflow pulses for its transmit clock in serial port Modes 1 and 3. TCLK = 0 causes Timer 1 overflows to be used for the transmit clock.
EXEN2	Timer 2 external enable. When set, allows a capture or reload to occur as a result of a negative transition on T2EX if Timer 2 is not being used to clock the serial port. EXEN2 = 0 causes Timer 2 to ignore events at T2EX.
TR2	Start/Stop control for Timer 2. TR2 = 1 starts the timer.
C/$\overline{T2}$	Timer or counter select for Timer 2. C/$\overline{T2}$ = 0 for timer function. C/$\overline{T2}$ = 1 for external event counter (falling edge triggered).
CP/$\overline{RL2}$	Capture/Reload select. CP/$\overline{RL2}$ = 1 causes captures to occur on negative transitions at T2EX if EXEN2 = 1. CP/$\overline{RL2}$ = 0 causes automatic reloads to occur when Timer 2 overflows or negative transitions occur at T2EX when EXEN2 = 1. When either RCLK or TCLK = 1, this bit is ignored and the timer is forced to auto-reload on Timer 2 overflow.

Data Memory

The AT89C52 implements 256 bytes of on-chip RAM. The upper 128 bytes occupy a parallel address space to the Special Function Registers. That means the upper 128 bytes have the same addresses as the SFR space but are physically separate from SFR space.

When an instruction accesses an internal location above address 7FH, the address mode used in the instruction specifies whether the CPU accesses the upper 128 bytes of RAM or the SFR space. Instructions that use direct addressing access SFR space.

For example, the following direct addressing instruction accesses the SFR at location 0A0H (which is P2).

```
MOV 0A0H, #data
```


เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Instructions that use indirect addressing access the upper 128 bytes of RAM. For example, the following indirect addressing instruction, where R0 contains 0A0H, accesses the data byte at address 0A0H, rather than P2 (whose address is 0A0H).

```
MOV @R0, #data
```

Note that stack operations are examples of indirect addressing, so the upper 128 bytes of data RAM are available as stack space.

Timer 0 and 1

Timer 0 and Timer 1 in the AT89C52 operate the same way as Timer 0 and Timer 1 in the AT89C51.

Timer 2

Timer 2 is a 16-bit Timer/Counter that can operate as either a timer or an event counter. The type of operation is selected by bit C/$\overline{T2}$ in the SFR T2CON (shown in Table 2). Timer 2 has three operating modes: capture, auto-reload (up or down counting), and baud rate generator. The modes are selected by bits in T2CON, as shown in Table 3.

Timer 2 consists of two 8-bit registers, TH2 and TL2. In the Timer function, the TL2 register is incremented every machine cycle. Since a machine cycle consists of 12 oscillator periods, the count rate is 1/12 of the oscillator frequency.

Table 3. Timer 2 Operating Modes

RCLK +TCLK	CP/$\overline{RL2}$	TR2	MODE
0	0	1	16-bit Auto-reload
0	1	1	16-bit Capture
1	X	1	Baud Rate Generator
X	X	0	(Off)

In the Counter function, the register is incremented in response to a 1-to-0 transition at its corresponding external input pin, T2. In this function, the external input is sampled during S5P2 of every machine cycle. When the samples show a high in one cycle and a low in the next cycle, the count is incremented. The new count value appears in the register during S3P1 of the cycle following the one in which the transition was detected. Since two machine cycles (24 oscillator periods) are required to recognize a 1-to-0 transition, the maximum count rate is 1/24 of the oscillator frequency. To ensure that a given level is sampled at least once before it changes, the level should be held for at least one full machine cycle.

Capture Mode

In the capture mode, two options are selected by bit EXEN2 in T2CON. If EXEN2 = 0, Timer 2 is a 16-bit timer or counter which upon overflow sets bit TF2 in T2CON. This bit can then be used to generate an interrupt. If EXEN2 = 1, Timer 2 performs the same operation, but a 1-to-0 transition at external input T2EX also causes the current value in TH2 and TL2 to be captured into RCAP2H and RCAP2L, respectively. In addition, the transition at T2EX causes bit EXF2 in T2CON to be set. The EXF2 bit, like TF2, can generate an interrupt. The capture mode is illustrated in Figure 1.

Auto-reload (Up or Down Counter)

Timer 2 can be programmed to count up or down when configured in its 16-bit auto-reload mode. This feature is invoked by the DCEN (Down Counter Enable) bit located in the SFR T2MOD (see Table 4). Upon reset, the DCEN bit is set to 0 so that timer 2 will default to count up. When DCEN is set, Timer 2 can count up or down, depending on the value of the T2EX pin.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Figure 1. Timer in Capture Mode

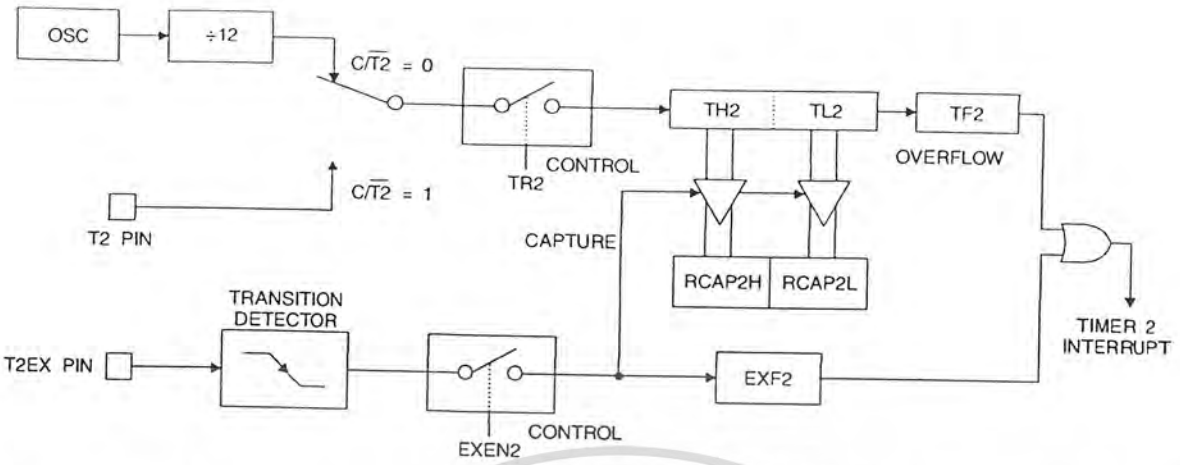


Figure 2 shows Timer 2 automatically counting up when DCEN = 0. In this mode, two options are selected by bit EXEN2 in T2CON. If EXEN2 = 0, Timer 2 counts up to 0FFFFH and then sets the TF2 bit upon overflow. The overflow also causes the timer registers to be reloaded with the 16-bit value in RCAP2H and RCAP2L. The values in Timer in Capture ModeRCAP2H and RCAP2L are preset by software. If EXEN2 = 1, a 16-bit reload can be triggered either by an overflow or by a 1-to-0 transition at external input T2EX. This transition also sets the EXF2 bit. Both the TF2 and EXF2 bits can generate an interrupt if enabled.

Setting the DCEN bit enables Timer 2 to count up or down, as shown in Figure 3. In this mode, the T2EX pin controls the direction of the count. A logic 1 at T2EX makes Timer 2 count up. The timer will overflow at 0FFFFH and set the TF2 bit. This overflow also causes the 16-bit value in RCAP2H and RCAP2L to be reloaded into the timer registers, TH2 and TL2, respectively.

A logic 0 at T2EX makes Timer 2 count down. The timer underflows when TH2 and TL2 equal the values stored in RCAP2H and RCAP2L. The underflow sets the TF2 bit and causes 0FFFFH to be reloaded into the timer registers.

The EXF2 bit toggles whenever Timer 2 overflows or underflows and can be used as a 17th bit of resolution. In this operating mode, EXF2 does not flag an interrupt.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Figure 2. Timer 2 Auto Reload Mode (DCEN = 0)

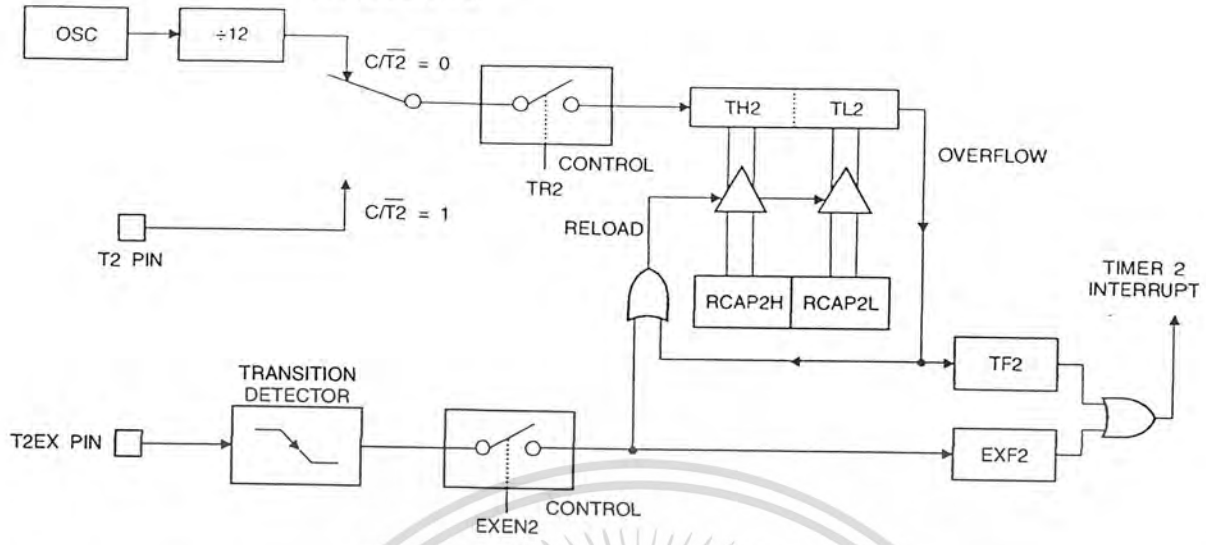


Table 4. T2MOD – Timer 2 Mode Control Register

T2MOD Address = 0C9H

Reset Value = XXXX XX00B

Not Bit Addressable

Bit	7	6	5	4	3	2	1	0
	–	–	–	–	–	–	T2OE	DCEN

Symbol	Function
–	Not implemented, reserved for future
T2OE	Timer 2 Output Enable bit.
DCEN	When set, this bit allows Timer 2 to be configured as an up/down counter.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Figure 3. Timer 2 Auto Reload Mode (DCEN = 1)

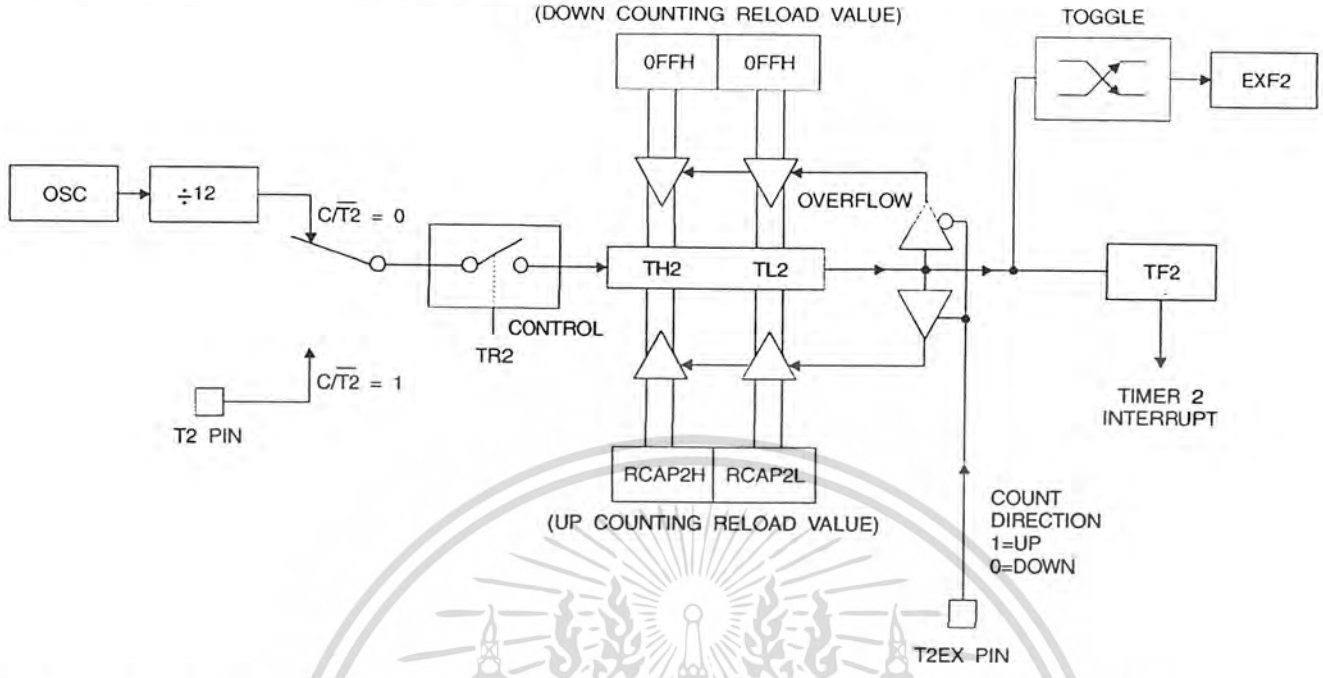
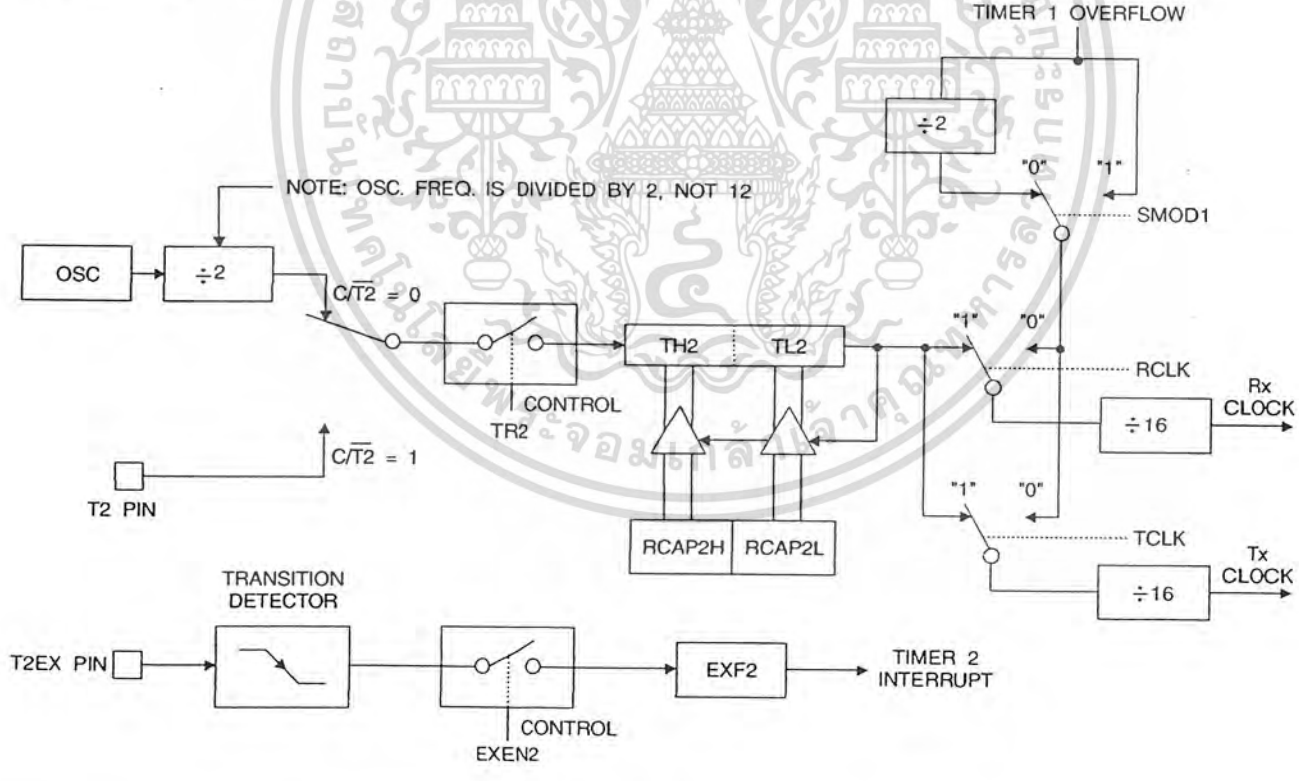


Figure 4. Timer 2 in Baud Rate Generator Mode



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Baud Rate Generator

Timer 2 is selected as the baud rate generator by setting TCLK and/or RCLK in T2CON (Table 2). Note that the baud rates for transmit and receive can be different if Timer 2 is used for the receiver or transmitter and Timer 1 is used for the other function. Setting RCLK and/or TCLK puts Timer 2 into its baud rate generator mode, as shown in Figure 4.

The baud rate generator mode is similar to the auto-reload mode, in that a rollover in TH2 causes the Timer 2 registers to be reloaded with the 16-bit value in registers RCAP2H and RCAP2L, which are preset by software.

The baud rates in Modes 1 and 3 are determined by Timer 2's overflow rate according to the following equation.

$$\text{Modes 1 and 3 Baud Rates} = \frac{\text{Timer 2 Overflow Rate}}{16}$$

The Timer can be configured for either timer or counter operation. In most applications, it is configured for timer operation ($C/\overline{T2} = 0$). The timer operation is different for Timer 2 when it is used as a baud rate generator. Normally, as a timer, it increments every machine cycle (at 1/12 the oscillator frequency). As a baud rate generator, however, it increments every state time (at 1/2 the oscillator frequency). The baud rate formula is given below.

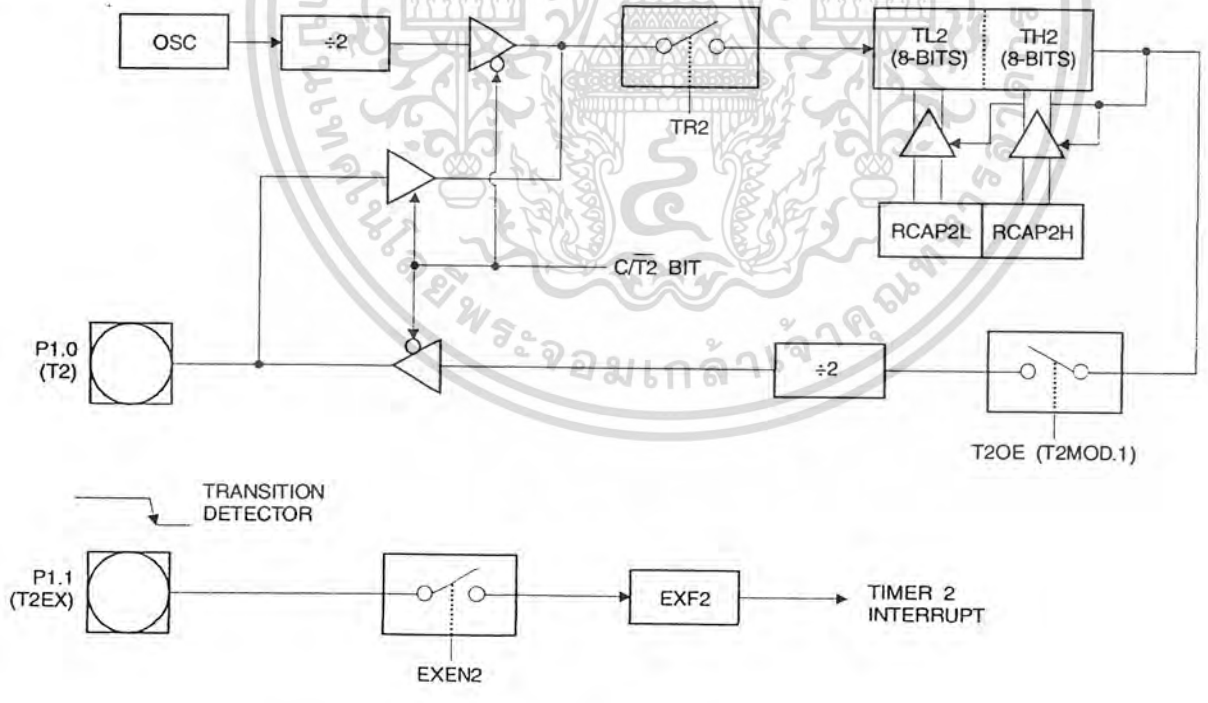
$$\frac{\text{Modes 1 and 3}}{\text{Baud Rate}} = \frac{\text{Oscillator Frequency}}{32 \times [65536 - (\text{RCAP2H,RCAP2L})]}$$

where (RCAP2H, RCAP2L) is the content of RCAP2H and RCAP2L taken as a 16-bit unsigned integer.

Timer 2 as a baud rate generator is shown in Figure 4. This figure is valid only if RCLK or TCLK = 1 in T2CON. Note that a rollover in TH2 does not set TF2 and will not generate an interrupt. Note too, that if EXEN2 is set, a 1-to-0 transition in T2EX will set EXF2 but will not cause a reload from (RCAP2H, RCAP2L) to (TH2, TL2). Thus when Timer 2 is in use as a baud rate generator, T2EX can be used as an extra external interrupt.

Note that when Timer 2 is running (TR2 = 1) as a timer in the baud rate generator mode, TH2 or TL2 should not be read from or written to. Under these conditions, the Timer is incremented every state time, and the results of a read or write may not be accurate. The RCAP2 registers may be read but should not be written to, because a write might overlap a reload and cause write and/or reload errors. The timer should be turned off (clear TR2) before accessing the Timer 2 or RCAP2 registers.

Figure 5. Timer 2 in Clock-out Mode



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Programmable Clock Out

A 50% duty cycle clock can be programmed to come out on P1.0, as shown in Figure 5. This pin, besides being a regular I/O pin, has two alternate functions. It can be programmed to input the external clock for Timer/Counter 2 or to output a 50% duty cycle clock ranging from 61 Hz to 4 MHz at a 16 MHz operating frequency.

To configure the Timer/Counter 2 as a clock generator, bit C/$\overline{T2}$ (T2CON.1) must be cleared and bit T2OE (T2MOD.1) must be set. Bit TR2 (T2CON.2) starts and stops the timer.

The clock-out frequency depends on the oscillator frequency and the reload value of Timer 2 capture registers (RCAP2H, RCAP2L), as shown in the following equation.

$$\text{Clock-Out Frequency} = \frac{\text{Oscillator Fequency}}{4 \times [65536 - (\text{RCAP2H,RCAP2L})]}$$

In the clock-out mode, Timer 2 roll-overs will not generate an interrupt. This behavior is similar to when Timer 2 is used as a baud-rate generator. It is possible to use Timer 2 as a baud-rate generator and a clock generator simultaneously. Note, however, that the baud-rate and clock-out frequencies cannot be determined independently from one another since they both use RCAP2H and RCAP2L.

UART

The UART in the AT89C52 operates the same way as the UART in the AT89C51.

Interrupts

The AT89C52 has a total of six interrupt vectors: two external interrupts ($\overline{INT0}$ and $\overline{INT1}$), three timer interrupts (Timers 0, 1, and 2), and the serial port interrupt. These interrupts are all shown in Figure 6.

Each of these interrupt sources can be individually enabled or disabled by setting or clearing a bit in Special Function Register IE. IE also contains a global disable bit, EA, which disables all interrupts at once.

Note that Table shows that bit position IE.6 is unimplemented. In the AT89C51, bit position IE.5 is also unimplemented. User software should not write 1s to these bit positions, since they may be used in future AT89 products.

Timer 2 interrupt is generated by the logical OR of bits TF2 and EXF2 in register T2CON. Neither of these flags is cleared by hardware when the service routine is vectored to. In fact, the service routine may have to determine whether it was TF2 or EXF2 that generated the interrupt, and that bit will have to be cleared in software.

The Timer 0 and Timer 1 flags, TF0 and TF1, are set at S5P2 of the cycle in which the timers overflow. The values are then polled by the circuitry in the next cycle. However, the Timer 2 flag, TF2, is set at S2P2 and is polled in the same cycle in which the timer overflows.

Table 5. Interrupt Enable (IE) Register

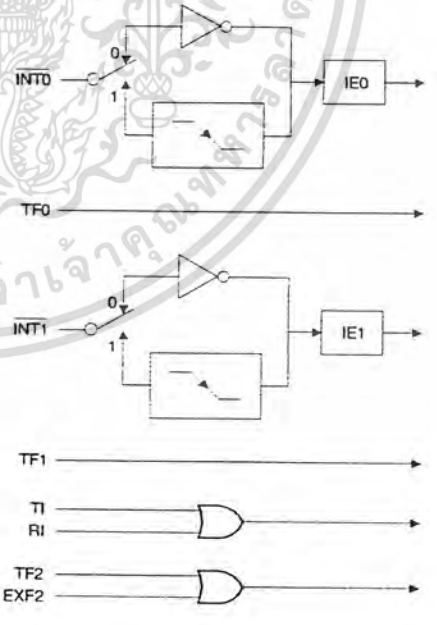
(MSB)							(LSB)
EA	–	ET2	ES	ET1	EX1	ET0	EX0

Enable Bit = 1 enables the interrupt.

Enable Bit = 0 disables the interrupt.

Symbol	Position	Function
EA	IE.7	Disables all interrupts. If EA = 0, no interrupt is acknowledged. If EA = 1, each interrupt source is individually enabled or disabled by setting or clearing its enable bit.
–	IE.6	Reserved.
ET2	IE.5	Timer 2 interrupt enable bit.
ES	IE.4	Serial Port interrupt enable bit.
ET1	IE.3	Timer 1 interrupt enable bit.
EX1	IE.2	External interrupt 1 enable bit.
ET0	IE.1	Timer 0 interrupt enable bit.
EX0	IE.0	External interrupt 0 enable bit.
User software should never write 1s to unimplemented bits, because they may be used in future AT89 products.		

Figure 6. Interrupt Sources



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Oscillator Characteristics

XTAL1 and XTAL2 are the input and output, respectively, of an inverting amplifier that can be configured for use as an on-chip oscillator, as shown in Figure 7. Either a quartz crystal or ceramic resonator may be used. To drive the device from an external clock source, XTAL2 should be left unconnected while XTAL1 is driven, as shown in Figure 8. There are no requirements on the duty cycle of the external clock signal, since the input to the internal clocking circuitry is through a divide-by-two flip-flop, but minimum and maximum voltage high and low time specifications must be observed.

Idle Mode

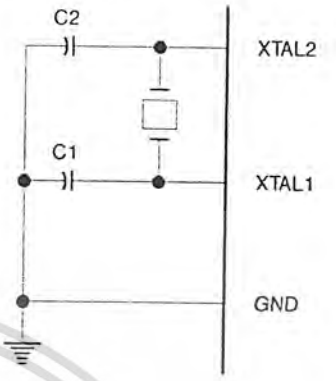
In idle mode, the CPU puts itself to sleep while all the on-chip peripherals remain active. The mode is invoked by software. The content of the on-chip RAM and all the special functions registers remain unchanged during this mode. The idle mode can be terminated by any enabled interrupt or by a hardware reset.

Note that when idle mode is terminated by a hardware reset, the device normally resumes program execution from where it left off, up to two machine cycles before the internal reset algorithm takes control. On-chip hardware inhibits access to internal RAM in this event, but access to the port pins is not inhibited. To eliminate the possibility of an unexpected write to a port pin when idle mode is terminated by a reset, the instruction following the one that invokes idle mode should not write to a port pin or to external memory.

Power-down Mode

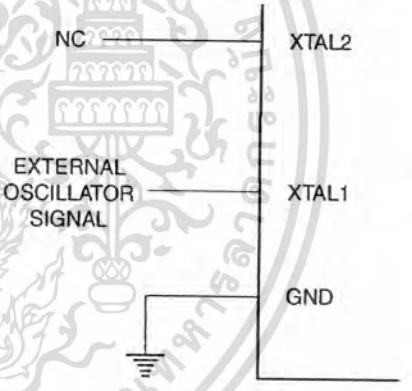
In the power-down mode, the oscillator is stopped, and the instruction that invokes power-down is the last instruction executed. The on-chip RAM and Special Function Registers retain their values until the power-down mode is terminated. The only exit from power-down is a hardware reset. Reset redefines the SFRs but does not change the on-chip RAM. The reset should not be activated before V_{CC} is restored to its normal operating level and must be held active long enough to allow the oscillator to restart and stabilize.

Figure 7. Oscillator Connections



Note: C1, C2 = 30 pF ± 10 pF for Crystals
= 40 pF ± 10 pF for Ceramic Resonators

Figure 8. External Clock Drive Configuration



Status of External Pins During Idle and Powe-down Modes

Mode	Program Memory	ALE	$\overline{PSEN}$	PORT0	PORT1	PORT2	PORT3
Idle	Internal	1	1	Data	Data	Data	Data
Idle	External	1	1	Float	Data	Address	Data
Power-down	Internal	0	0	Data	Data	Data	Data
Power-down	External	0	0	Float	Data	Data	Data

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Program Memory Lock Bits

The AT89C52 has three lock bits that can be left unprogrammed (U) or can be programmed (P) to obtain the additional features listed in the following table.

Lock Bit Protection Modes

Program Lock Bits				
	LB1	LB2	LB3	Protection Type
1	U	U	U	No program lock features.
2	P	U	U	MOVC instructions executed from external program memory are disabled from fetching code bytes from internal memory, $\overline{EA}$ is sampled and latched on reset, and further programming of the Flash memory is disabled.
3	P	P	U	Same as mode 2, but verify is also disabled.
4	P	P	P	Same as mode 3, but external execution is also disabled.

When lock bit 1 is programmed, the logic level at the $\overline{EA}$ pin is sampled and latched during reset. If the device is powered up without a reset, the latch initializes to a random value and holds that value until reset is activated. The latched value of $\overline{EA}$ must agree with the current logic level at that pin in order for the device to function properly.

Programming the Flash

The AT89C52 is normally shipped with the on-chip Flash memory array in the erased state (that is, contents = FFH) and ready to be programmed. The programming interface accepts either a high-voltage (12-volt) or a low-voltage (V_{CC}) program enable signal. The Low-voltage programming mode provides a convenient way to program the AT89C52 inside the user's system, while the high-voltage programming mode is compatible with conventional third-party Flash or EPROM programmers.

The AT89C52 is shipped with either the high-voltage or low-voltage programming mode enabled. The respective top-side marking and device signature codes are listed in the following table.

	V_{PP} = 12V	V_{PP} = 5V
Top-side Mark	AT89C52 xxxx yyww	AT89C52 xxxx - 5 yyww
Signature	(030H) = 1EH (031H) = 52H (032H) = FFH	(030H) = 1EH (031H) = 52H (032H) = 05H

The AT89C52 code memory array is programmed byte-by-byte in either programming mode. *To program any non-blank byte in the on-chip Flash Memory, the entire memory must be erased using the Chip Erase Mode.*

Programming Algorithm Before programming the AT89C52, the address, data and control signals should be set up according to the Flash programming mode table and Figure 9 and Figure 10. To program the AT89C52, take the following steps.

1. Input the desired memory location on the address lines.
2. Input the appropriate data byte on the data lines.
3. Activate the correct combination of control signals.
4. Raise $\overline{EA}/V_{PP}$ to 12V for the high-voltage programming mode.
5. Pulse ALE/$\overline{PROG}$ once to program a byte in the Flash array or the lock bits. The byte-write cycle is self-timed and typically takes no more than 1.5 ms. Repeat steps 1 through 5, changing the address and data for the entire array or until the end of the object file is reached.

$\overline{Data}$ Polling The AT89C52 features $\overline{Data}$ Polling to indicate the end of a write cycle. During a write cycle, an attempted read of the last byte written will result in the complement of the written data on PO.7. Once the write cycle has been completed, true data is valid on all outputs, and the next cycle may begin. $\overline{Data}$ Polling may begin any time after a write cycle has been initiated.

Ready/$\overline{Busy}$ The progress of byte programming can also be monitored by the RDY/$\overline{BSY}$ output signal. P3.4 is pulled low after ALE goes high during programming to indicate $\overline{BUSY}$. P3.4 is pulled high again when programming is done to indicate READY.

Program Verify If lock bits LB1 and LB2 have not been programmed, the programmed code data can be read back via the address and data lines for verification. The lock bits cannot be verified directly. Verification of the lock bits is achieved by observing that their features are enabled.

Chip Erase The entire Flash array is erased electrically by using the proper combination of control signals and by holding ALE/$\overline{PROG}$ low for 10 ms. The code array is written with all 1s. The chip erase operation must be executed before the code memory can be reprogrammed.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Reading the Signature Bytes The signature bytes are read by the same procedure as a normal verification of locations 030H, 031H, and 032H, except that P3.6 and P3.7 must be pulled to a logic low. The values returned are as follows.

(030H) = 1EH indicates manufactured by Atmel
(031H) = 52H indicates 89C52
(032H) = FFH indicates 12V programming
(032H) = 05H indicates 5V programming

Programming Interface

Every code byte in the Flash array can be written, and the entire array can be erased, by using the appropriate combination of control signals. The write operation cycle is self-timed and once initiated, will automatically time itself to completion.

All major programming vendors offer worldwide support for the Atmel microcontroller series. Please contact your local programming vendor for the appropriate software revision.

Flash Programming Modes

Mode		RST	$\overline{PSEN}$	ALE/$\overline{PROG}$	$\overline{EA}/V_{PP}$	P2.6	P2.7	P3.6	P3.7
Write Code Data		H	L		H/12V	L	H	H	H
Read Code Data		H	L	H	H	L	L	H	H
Write Lock	Bit - 1	H	L		H/12V	H	H	H	H
	Bit - 2	H	L		H/12V	H	H	L	L
	Bit - 3	H	L		H/12V	H	L	H	L
Chip Erase		H	L	(1)	H/12V	H	L	L	L
Read Signature Byte		H	L	H	H	L	L	L	L

ote: 1. Chip Erase requires a 10 ms $\overline{PROG}$ pulse.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Figure 9. Programming the Flash Memory

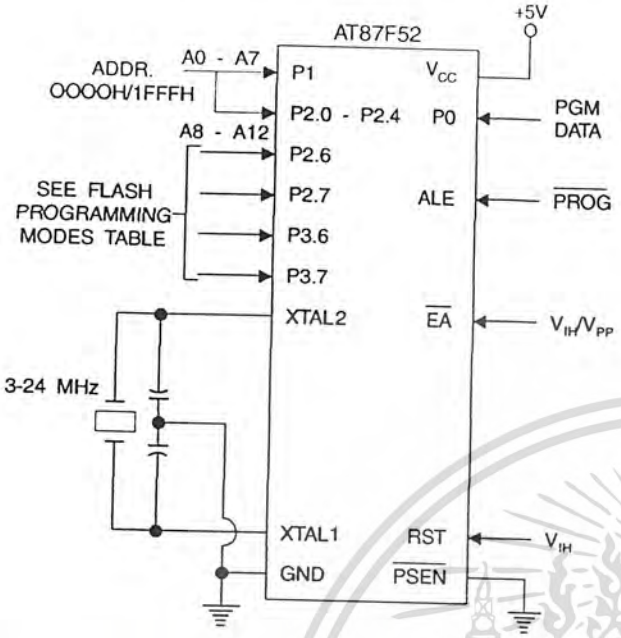
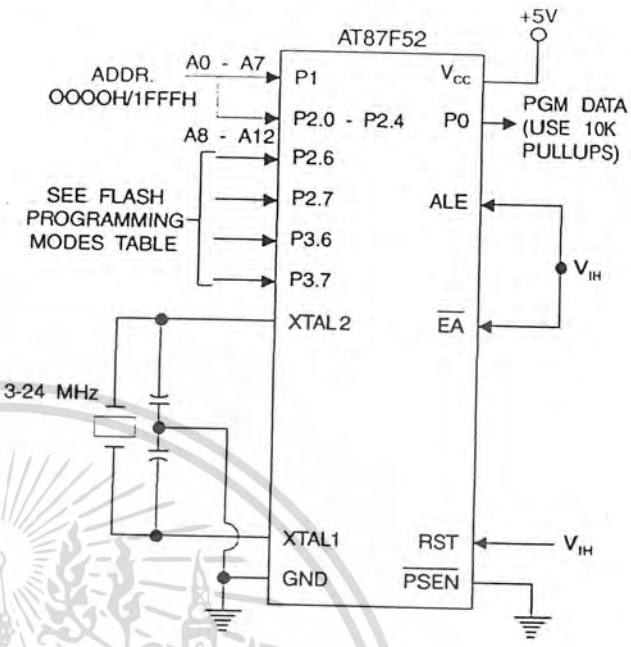


Figure 10. Verifying the Flash Memory



Flash Programming and Verification Characteristics

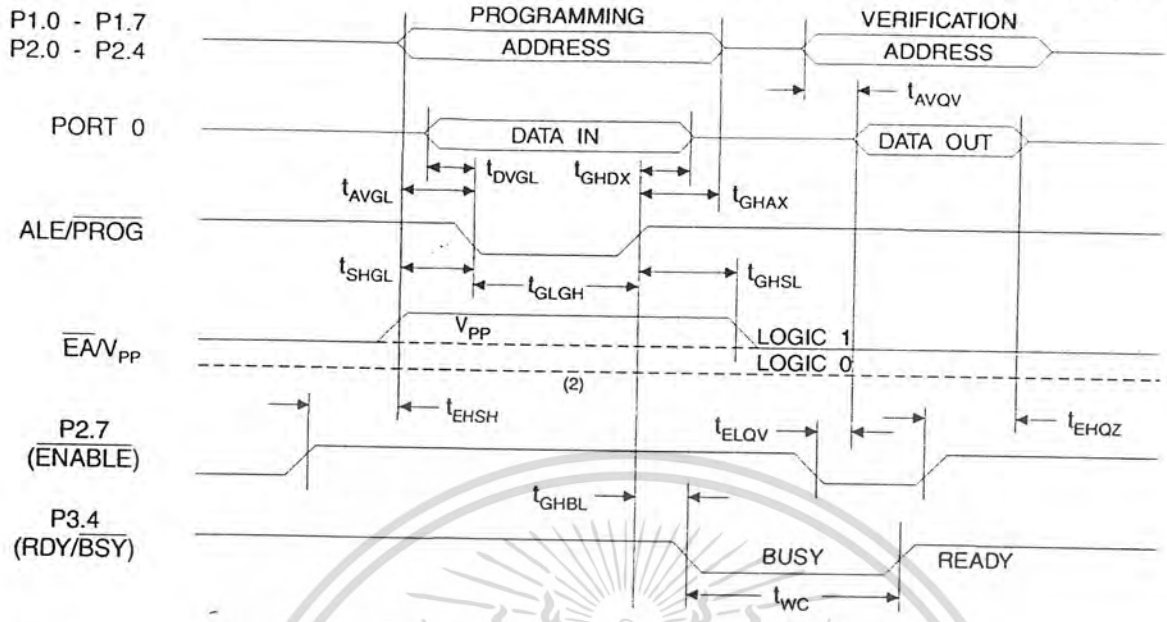
T_A = 0°C to 70°C, V_{CC} = 5.0 ± 10%

Symbol	Parameter	Min	Max	Units
V_{PP}(1)	Programming Enable Voltage	11.5	12.5	V
I_{PP}(1)	Programming Enable Current		1.0	mA
$1/t_{CLCL}$	Oscillator Frequency	3	24	MHz
t_{AVGL}	Address Setup to $\overline{PROG}$ Low	$48t_{CLCL}$		
t_{GHAX}	Address Hold after $\overline{PROG}$	$48t_{CLCL}$		
t_{DVGL}	Data Setup to $\overline{PROG}$ Low	$48t_{CLCL}$		
t_{GHDX}	Data Hold After $\overline{PROG}$	$48t_{CLCL}$		
t_{EHSH}	P2.7 ($\overline{ENABLE}$) High to V_{PP}	$48t_{CLCL}$		
t_{SHGL}	V_{PP} Setup to $\overline{PROG}$ Low	10		µs
t_{GHSL}(1)	V_{PP} Hold after $\overline{PROG}$	10		µs
t_{GLGH}	$\overline{PROG}$ Width	1	110	µs
t_{AVQV}	Address to Data Valid		$48t_{CLCL}$	
t_{ELQV}	$\overline{ENABLE}$ Low to Data Valid		$48t_{CLCL}$	
t_{EHQZ}	Data Float after $\overline{ENABLE}$	0	$48t_{CLCL}$	
t_{GHBL}	$\overline{PROG}$ High to $\overline{BUSY}$ Low		1.0	µs
t_{WC}	Byte Write Cycle Time		2.0	ms

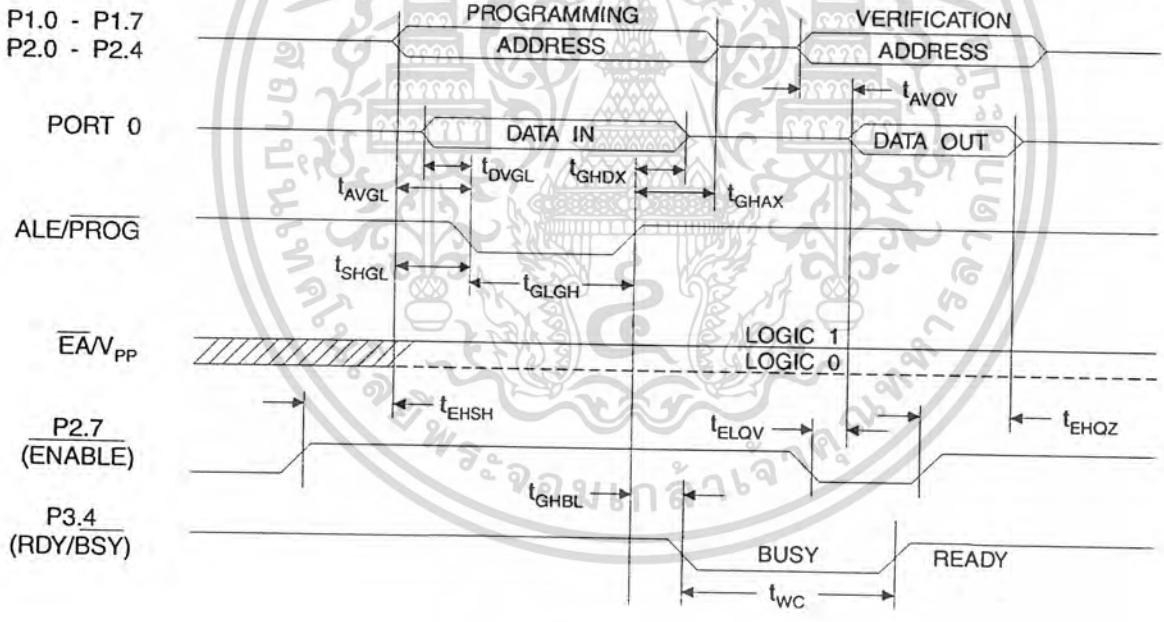
Note: 1. Only used in 12-volt programming mode.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Flash Programming and Verification Waveforms - High-voltage Mode (V_{PP}=12V)



Flash Programming and Verification Waveforms - Low-voltage Mode (V_{PP}=5V)



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Absolute Maximum Ratings*

Operating Temperature	-55°C to +125°C
Storage Temperature	-65°C to +150°C
Voltage on Any Pin with Respect to Ground	-1.0V to +7.0V
Maximum Operating Voltage	6.6V
DC Output Current	15.0 mA

*NOTICE: Stresses beyond those listed under "Absolute Maximum Ratings" may cause permanent damage to the device. This is a stress rating only and functional operation of the device at these or any other conditions beyond those indicated in the operational sections of this specification is not implied. Exposure to absolute maximum rating conditions for extended periods may affect device reliability.

DC Characteristics

The values shown in this table are valid for T_A = -40°C to 85°C and V_{CC} = 5.0V ± 20%, unless otherwise noted.

Symbol	Parameter	Condition	Min	Max	Units
V_{IL}	Input Low-voltage	(Except $\overline{EA}$)	-0.5	0.2 V_{CC}-0.1	V
V_{IL1}	Input Low-voltage ($\overline{EA}$)		-0.5	0.2 V_{CC}-0.3	V
V_{IH}	Input High-voltage	(Except XTAL1, RST)	0.2 V_{CC}+0.9	V_{CC}+0.5	V
V_{IH1}	Input High-voltage	(XTAL1, RST)	0.7 V_{CC}	V_{CC}+0.5	V
V_{OL}	Output Low-voltage[1] (Ports 1,2,3)	I_{OL} = 1.6 mA		0.45	V
V_{OL1}	Output Low-voltage[1] (Port 0, ALE, $\overline{PSEN}$)	I_{OL} = 3.2 mA		0.45	V
V_{OH}	Output High-voltage (Ports 1,2,3, ALE, $\overline{PSEN}$)	I_{OH} = -60 µA, V_{CC} = 5V ± 10%	2.4		V
		I_{OH} = -25 µA	0.75 V_{CC}		V
		I_{OH} = -10 µA	0.9 V_{CC}		V
V_{OH1}	Output High-voltage (Port 0 in External Bus Mode)	I_{OH} = -800 µA, V_{CC} = 5V ± 10%	2.4		V
		I_{OH} = -300 µA	0.75 V_{CC}		V
		I_{OH} = -80 µA	0.9 V_{CC}		V
I_{IL}	Logical 0 Input Current (Ports 1,2,3)	V_{IN} = 0.45V		-50	µA
I_{TL}	Logical 1 to 0 Transition Current (Ports 1,2,3)	V_{IN} = 2V, V_{CC} = 5V ± 10%		-650	µA
I_{LI}	Input Leakage Current (Port 0, $\overline{EA}$)	0.45 < V_{IN} < V_{CC}		±10	µA
RRST	Reset Pulldown Resistor		50	300	KΩ
C_{IO}	Pin Capacitance	Test Freq. = 1 MHz, T_A = 25°C		10	pF
I_{CC}	Power Supply Current	Active Mode, 12 MHz		25	mA
		Idle Mode, 12 MHz		6.5	mA
	Power-down Mode[1]	V_{CC} = 6V		100	µA
		V_{CC} = 3V		40	µA

Notes: 1. Under steady state (non-transient) conditions, I_{OL} must be externally limited as follows:
Maximum I_{OL} per port pin: 10 mA
Maximum I_{OL} per 8-bit port:
Port 0: 26 mA Ports 1, 2, 3: 15 mA
Maximum total I_{OL} for all output pins: 71 mA
If I_{OL} exceeds the test condition, V_{OL} may exceed the related specification. Pins are not guaranteed to sink current greater than the listed test conditions.
2. Minimum V_{CC} for Power-down is 2V.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

AC Characteristics

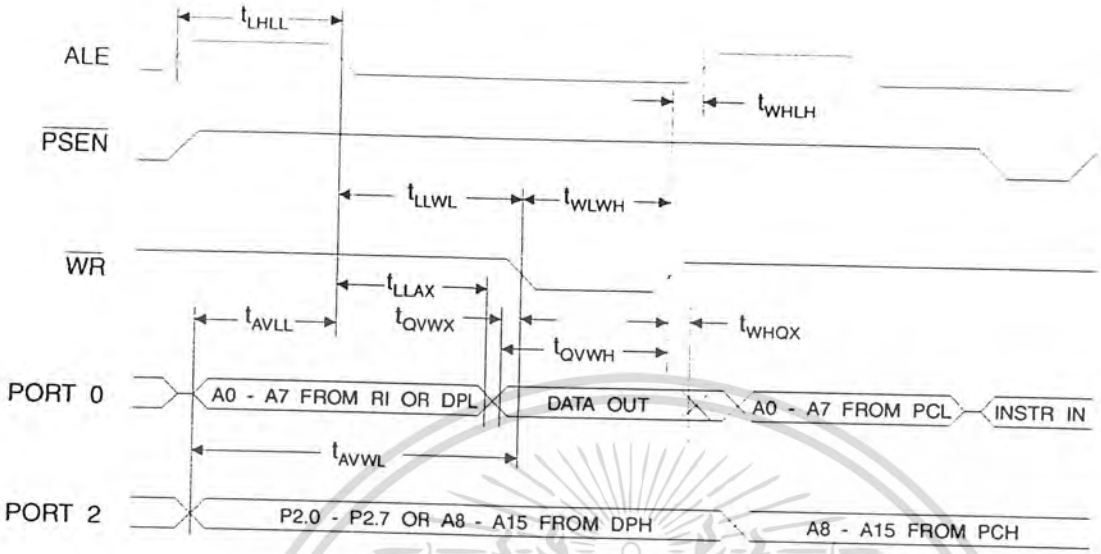
Under operating conditions, load capacitance for Port 0, ALE/$\overline{PROG}$, and $\overline{PSEN}$ = 100 pF; load capacitance for all other outputs = 80 pF.

External Program and Data Memory Characteristics

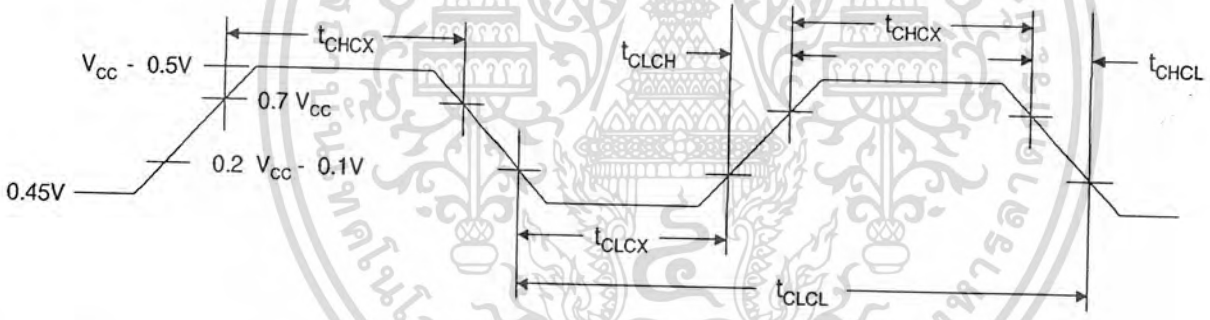
Symbol	Parameter	12 MHz Oscillator		Variable Oscillator		Units
		Min	Max	Min	Max	
$1/t_{CLCL}$	Oscillator Frequency			0	24	MHz
t_{LHLL}	ALE Pulse Width	127		$2t_{CLCL}$-40		ns
t_{AVLL}	Address Valid to ALE Low	43		t_{CLCL}-13		ns
t_{LLAX}	Address Hold After ALE Low	48		t_{CLCL}-20		ns
t_{LLIV}	ALE Low to Valid Instruction In		233		$4t_{CLCL}$-65	ns
t_{LLPL}	ALE Low to $\overline{PSEN}$ Low	43		t_{CLCL}-13		ns
t_{PLPH}	$\overline{PSEN}$ Pulse Width	205		$3t_{CLCL}$-20		ns
t_{PLIV}	$\overline{PSEN}$ Low to Valid Instruction In		145		$3t_{CLCL}$-45	ns
t_{PXIX}	Input Instruction Hold after $\overline{PSEN}$	0		0		ns
t_{PXIZ}	Input Instruction Float after $\overline{PSEN}$		59		t_{CLCL}-10	ns
t_{PXAV}	$\overline{PSEN}$ to Address Valid	75		t_{CLCL}-8		ns
t_{AVIV}	Address to Valid Instruction In		312		$5t_{CLCL}$-55	ns
t_{PLAZ}	$\overline{PSEN}$ Low to Address Float		10		10	ns
t_{RLRH}	$\overline{RD}$ Pulse Width	400		$6t_{CLCL}$-100		ns
t_{WLWH}	$\overline{WR}$ Pulse Width	400		$6t_{CLCL}$-100		ns
t_{RLDV}	$\overline{RD}$ Low to Valid Data In		252		$5t_{CLCL}$-90	ns
t_{RHDX}	Data Hold After $\overline{RD}$	0		0		ns
t_{RHDZ}	Data Float After $\overline{RD}$		97		$2t_{CLCL}$-28	ns
t_{LLDV}	ALE Low to Valid Data In		517		$8t_{CLCL}$-150	ns
t_{AVDV}	Address to Valid Data In		585		$9t_{CLCL}$-165	ns
t_{LLWL}	ALE Low to $\overline{RD}$ or $\overline{WR}$ Low	200	300	$3t_{CLCL}$-50	$3t_{CLCL}$+50	ns
t_{AVWL}	Address to $\overline{RD}$ or $\overline{WR}$ Low	203		$4t_{CLCL}$-75		ns
t_{QVWX}	Data Valid to $\overline{WR}$ Transition	23		t_{CLCL}-20		ns
t_{QVWH}	Data Valid to $\overline{WR}$ High	433		$7t_{CLCL}$-120		ns
t_{WHQX}	Data Hold After $\overline{WR}$	33		t_{CLCL}-20		ns
t_{RLAZ}	$\overline{RD}$ Low to Address Float		0		0	ns
t_{WHLH}	$\overline{RD}$ or $\overline{WR}$ High to ALE High	43	123	t_{CLCL}-20	t_{CLCL}+25	ns

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

External Data Memory Write Cycle



External Clock Drive Waveforms



External Clock Drive

Symbol	Parameter	Min	Max	Units
$1/t_{CLCL}$	Oscillator Frequency	0	24	MHz
t_{CLCL}	Clock Period	41.6		ns
t_{CHCX}	High Time	15		ns
t_{CLCX}	Low Time	15		ns
t_{CLCH}	Rise Time		20	ns
t_{CHCL}	Fall Time		20	ns

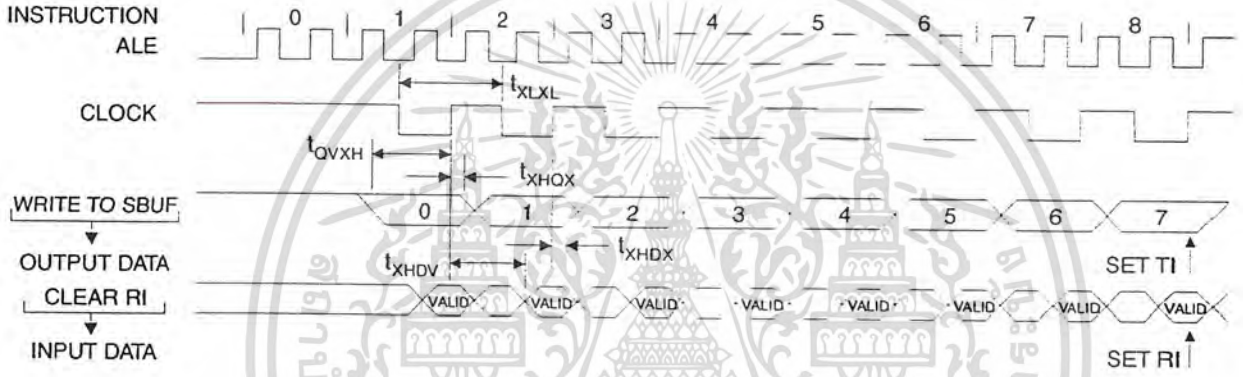
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Serial Port Timing: Shift Register Mode Test Conditions

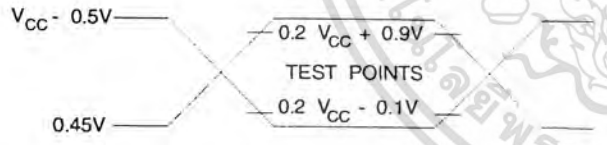
The values in this table are valid for V_{CC} = 5.0V ± 20% and Load Capacitance = 80 pF.

Symbol	Parameter	12 MHz Osc		Variable Oscillator		Units
		Min	Max	Min	Max	
t_{XLXL}	Serial Port Clock Cycle Time	1.0		$12t_{CLCL}$		µs
t_{QVXH}	Output Data Setup to Clock Rising Edge	700		$10t_{CLCL}$-133		ns
t_{XHQX}	Output Data Hold After Clock Rising Edge	50		$2t_{CLCL}$-117		ns
t_{XHDX}	Input Data Hold After Clock Rising Edge	0		0		ns
t_{XHDV}	Clock Rising Edge to Input Data Valid		700		$10t_{CLCL}$-133	ns

Shift Register Mode Timing Waveforms

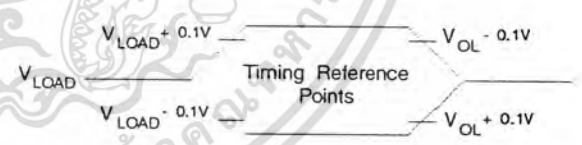


AC Testing Input/Output Waveforms[1]



Note: 1. AC Inputs during testing are driven at V_{CC} - 0.5V for a logic 1 and 0.45V for a logic 0. Timing measurements are made at V_{IH} min. for a logic 1 and V_{IL} max. for a logic 0.

Float Waveforms[1]



Note: 1. For timing purposes, a port pin is no longer floating when a 100 mV change from load voltage occurs. A port pin begins to float when a 100 mV change from the loaded V_{OH}/V_{OL} level occurs.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Ordering Information

Speed (MHz)	Power Supply	Ordering Code	Package	Operation Range
12	5V ± 20%	AT89C52-12AC	44A	Commercial (0°C to 70°C)
		AT89C52-12JC	44J	
		AT89C52-12PC	40P6	
		AT89C52-12QC	44Q	
		AT89C52-12AI	44A	Industrial (-40°C to 85°C)
		AT89C52-12JI	44J	
		AT89C52-12PI	40P6	
		AT89C52-12QI	44Q	
16	5V ± 20%	AT89C52-16AC	44A	Commercial (0°C to 70°C)
		AT89C52-16JC	44J	
		AT89C52-16PC	40P6	
		AT89C52-16QC	44Q	
		AT89C52-16AI	44A	Industrial (-40°C to 85°C)
		AT89C52-16JI	44J	
		AT89C52-16PI	40P6	
		AT89C52-16QI	44Q	
20	5V ± 20%	AT89C52-20AC	44A	Commercial (0°C to 70°C)
		AT89C52-20JC	44J	
		AT89C52-20PC	40P6	
		AT89C52-20QC	44Q	
		AT89C52-20AI	44A	Industrial (-40°C to 85°C)
		AT89C52-20JI	44J	
		AT89C52-20PI	40P6	
		AT89C52-20QI	44Q	
24	5V ± 20%	AT89C52-24AC	44A	Commercial (0°C to 70°C)
		AT89C52-24JC	44J	
		AT89C52-24PC	40P6	
		AT89C52-24QC	44Q	
		AT89C52-24AI	44A	Industrial (-40°C to 85°C)
		AT89C52-24JI	44J	
		AT89C52-24PI	40P6	
		AT89C52-24QI	44Q	

Package Type	
44A	44-lead, Thin Plastic Gull Wing Quad Flatpack (TQFP)
44J	44-lead, Plastic J-leaded Chip Carrier (PLCC)
40P6	40-lead, 0.600" Wide, Plastic Dual Inline Package (PDIP)
44Q	44-lead, Plastic Gull Wing Quad Flatpack (PQFP)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

19-4323; Rev 9; 5/00

MAXIM

+5V-Powered, Multichannel RS-232 Drivers/Receivers

General Description

The MAX220–MAX249 family of line drivers/receivers is intended for all EIA/TIA-232E and V.28/V.24 commupications interfaces, particularly applications where ±12V is *not* available.

These parts are especially useful in battery-powered systems, since their low-power shutdown mode reduces power dissipation to less than 5µW. The MAX225, MAX233, MAX235, and MAX245/MAX246/MAX247 use no external components and are recommended for applications where printed circuit board space is critical.

Applications

Portable Computers

Low-Power Modems

Interface Translation

Battery-Powered RS-232 Systems

Multidrop RS-232 Networks

Features

Superior to Bipolar

- **Operate from Single +5V Power Supply (+5V and +12V—MAX231/MAX239)**
- **Low-Power Receive Mode in Shutdown (MAX223/MAX242)**
- **Meet All EIA/TIA-232E and V.28 Specifications**
- **Multiple Drivers and Receivers**
- **3-State Driver and Receiver Outputs**
- **Open-Line Detection (MAX243)**

Ordering Information

PART	TEMP. RANGE	PIN-PACKAGE
MAX220CPE	0°C to +70°C	16 Plastic DIP
MAX220CSE	0°C to +70°C	16 Narrow SO
MAX220CWE	0°C to +70°C	16 Wide SO
MAX220C/D	0°C to +70°C	Dice*
MAX220EPE	-40°C to +85°C	16 Plastic DIP
MAX220ESE	-40°C to +85°C	16 Narrow SO
MAX220EWE	-40°C to +85°C	16 Wide SO
MAX220EJE	-40°C to +85°C	16 CERDIP
MAX220MJE	-55°C to +125°C	16 CERDIP

Ordering Information continued at end of data sheet.

**Contact factory for dice specifications.*

Selection Table

Part Number	Power Supply (V)	No. of RS-232 Drivers/Rx	No. of Ext. Caps	Nominal Cap. Value (µF)	SHDN & Three-State	Rx Active in SHDN	Data Rate (kbps)	Features
MAX220	+5	2/2	4	0.1	No	—	120	Ultra-low-power, industry-standard pinout
MAX222	+5	2/2	4	0.1	Yes	—	200	Low-power shutdown
MAX223 (MAX213)	+5	4/5	4	1.0 (0.1)	Yes	✓	120	MAX241 and receivers active in shutdown
MAX225	+5	5/5	0	—	Yes	✓	120	Available in SO
MAX230 (MAX200)	+5	5/0	4	1.0 (0.1)	Yes	—	120	5 drivers with shutdown
MAX231 (MAX201)	+5 and +7.5 to +13.2	2/2	2	1.0 (0.1)	No	—	120	Standard +5/+12V or battery supplies; same functions as MAX232
MAX232 (MAX202)	+5	2/2	4	1.0 (0.1)	No	—	120 (64)	Industry standard
MAX232A	+5	2/2	4	0.1	No	—	200	Higher slew rate, small caps
MAX233 (MAX203)	+5	2/2	0	—	No	—	120	No external caps
MAX233A	+5	2/2	0	—	No	—	200	No external caps, high slew rate
MAX234 (MAX204)	+5	4/0	4	1.0 (0.1)	No	—	120	Replaces 1488
MAX235 (MAX205)	+5	5/5	0	—	Yes	—	120	No external caps
MAX236 (MAX206)	+5	4/3	4	1.0 (0.1)	Yes	—	120	Shutdown, three state
MAX237 (MAX207)	+5	5/3	4	1.0 (0.1)	No	—	120	Complements IBM PC serial port
MAX238 (MAX208)	+5	4/4	4	1.0 (0.1)	No	—	120	Replaces 1488 and 1489
MAX239 (MAX209)	+5 and +7.5 to +13.2	3/5	2	1.0 (0.1)	No	—	120	Standard +5/+12V or battery supplies; single-package solution for IBM PC serial port
MAX240	+5	5/5	4	1.0	Yes	—	120	DIP or flatpack package
MAX241 (MAX211)	+5	4/5	4	1.0 (0.1)	Yes	—	120	Complete IBM PC serial port
MAX242	+5	2/2	4	0.1	Yes	✓	200	Separate shutdown and enable
MAX243	+5	2/2	4	0.1	No	—	200	Open-line detection simplifies cabling
MAX244	+5	8/10	4	1.0	No	—	120	High slew rate
MAX245	+5	8/10	0	—	Yes	✓	120	High slew rate, int. caps, two shutdown modes
MAX246	+5	8/10	0	—	Yes	✓	120	High slew rate, int. caps, three shutdown modes
MAX247	+5	8/9	0	—	Yes	✓	120	High slew rate, int. caps, nine operating modes
MAX248	+5	8/8	4	1.0	Yes	✓	120	High slew rate, selective half-chip enables
MAX249	+5	6/10	4	1.0	Yes	✓	120	Available in quad flatpack package

For free samples and the latest literature, visit www.maxim-ic.com or phone 1-800-998-8800. For small orders, phone 1-800-835-8769.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

+5V-Powered, Multichannel RS-232 Drivers/Receivers

ABSOLUTE MAXIMUM RATINGS—MAX220/222/232A/233A/242/243

Supply Voltage (V_{CC}) ..... -0.3V to +6V
Input Voltages
T_{IN} ..... -0.3V to (V_{CC} + 0.3V)
R_{IN} (Except MAX220) ..... ±30V
R_{IN} (MAX220) ..... ±25V
T_{OUT} (Except MAX220) (Note 1) ..... ±15V
T_{OUT} (MAX220) ..... ±13.2V
Output Voltages
T_{OUT} ..... ±15V
R_{OUT} ..... -0.3V to (V_{CC} + 0.3V)
Driver/Receiver Output Short Circuited to GND ..... Continuous
Continuous Power Dissipation (T_A = +70°C)
16-Pin Plastic DIP (derate 10.53mW/°C above +70°C) 842mW
18-Pin Plastic DIP (derate 11.11mW/°C above +70°C) 889mW
20-Pin Plastic DIP (derate 8.00mW/°C above +70°C) 440mW
16-Pin Narrow SO (derate 8.70mW/°C above +70°C) 696mW
16-Pin Wide SO (derate 9.52mW/°C above +70°C) 762mW
18-Pin Wide SO (derate 9.52mW/°C above +70°C) 762mW
20-Pin Wide SO (derate 10.00mW/°C above +70°C) 800mW
20-Pin SSOP (derate 8.00mW/°C above +70°C) 640mW
16-Pin CERDIP (derate 10.00mW/°C above +70°C) 800mW
18-Pin CERDIP (derate 10.53mW/°C above +70°C) 842mW
Operating Temperature Ranges
MAX2_ _AC_ _, MAX2_ _C_ _ ..... 0°C to +70°C
MAX2_ _AE_ _, MAX2_ _E_ _ ..... -40°C to +85°C
MAX2_ _AM_ _, MAX2_ _M_ _ ..... -55°C to +125°C
Storage Temperature Range ..... -65°C to +160°C
Lead Temperature (soldering, 10sec) ..... +300°C

Note 1: Input voltage measured with T_{OUT} in high-impedance state, $\overline{SHDN}$ or V_{CC} = 0V.
Note 2: For the MAX220, V+ and V- can have a maximum magnitude of 7V, but their absolute difference cannot exceed 13V.

Stresses beyond those listed under "Absolute Maximum Ratings" may cause permanent damage to the device. These are stress ratings only, and functional operation of the device at these or any other conditions beyond those indicated in the operational sections of the specifications is not implied. Exposure to absolute maximum rating conditions for extended periods may affect device reliability.

ELECTRICAL CHARACTERISTICS—MAX220/222/232A/233A/242/243

(V_{CC} = +5V ±10%, C1–C4 = 0.1µF, MAX220, C1 = 0.047µF, C2–C4 = 0.33µF, T_A = T_{MIN} to T_{MAX}, unless otherwise noted.)

PARAMETER	CONDITIONS		MIN	TYP	MAX	UNITS
RS-232 TRANSMITTERS						
Output Voltage Swing	All transmitter outputs loaded with 3kΩ to GND		±5	±8		V
Input Logic Threshold Low	All devices except MAX220			1.4	0.8	V
	MAX220: V_{CC} = 5.0V		2	1.4		
Input Logic Threshold High	All except MAX220, normal operation		2.4			V
	$\overline{SHDN}$ = 0V, MAX222/242, shutdown, MAX220			5	40	µA
Logic Pull-Up/Input Current	V_{CC} = 5.5V, $\overline{SHDN}$ = 0V, V_{OUT} = ±15V, MAX222/242			±0.01	±1	
Output Leakage Current	V_{CC} = $\overline{SHDN}$ = 0V, V_{OUT} = ±15V			±0.01	±10	µA
Data Rate				±0.01	±10	
Transmitter Output Resistance	V_{CC} = V+ = V- = 0V, V_{OUT} = ±2V			200	116	kb/s
Output Short-Circuit Current	V_{OUT} = 0V		300	10M		Ω
RS-232 RECEIVERS				±7	±22	mA
RS-232 Input Voltage Operating Range					±30	V
RS-232 Input Threshold Low	V_{CC} = 5V	All except MAX243 $R2_{IN}$	0.8	1.3		V
		MAX243 $R2_{IN}$ (Note 2)	-3			
RS-232 Input Threshold High	V_{CC} = 5V	All except MAX243 $R2_{IN}$		1.8	2.4	V
		MAX243 $R2_{IN}$ (Note 2)		-0.5	-0.1	
RS-232 Input Hysteresis	All except MAX243, V_{CC} = 5V, no hysteresis in shdn.		0.2	0.5	1	V
	MAX243			1		
RS-232 Input Resistance			3	5	7	kΩ
TTL/CMOS Output Voltage Low	I_{OUT} = 3.2mA			0.2	0.4	V
TTL/CMOS Output Voltage High	I_{OUT} = -1.0mA		3.5	V_{CC} - 0.2		V
TTL/CMOS Output Short-Circuit Current	Sourcing V_{OUT} = GND		-2	-10		mA
	Shrinking V_{OUT} = V_{CC}		10	30		

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ELECTRICAL CHARACTERISTICS—MAX220/222/232A/233A/242/243 (continued)

(V_{CC} = +5V ±10%, C1–C4 = 0.1µF, MAX220, C1 = 0.047µF, C2–C4 = 0.33µF, T_A = T_{MIN} to T_{MAX}, unless otherwise noted.)

PARAMETER	CONDITIONS		MIN	TYP	MAX	UNITS
TTL/CMOS Output Leakage Current	$\overline{SHDN}$ = V_{CC} or $\overline{EN}$ = V_{CC} ($\overline{SHDN}$ = 0V for MAX222), 0V ≤ V_{OUT} ≤ V_{CC}			±0.05	±10	µA
$\overline{EN}$ Input Threshold Low	MAX242			1.4	0.8	V
$\overline{EN}$ Input Threshold High	MAX242		2.0	1.4		V
Operating Supply Voltage			4.5		5.5	V
V_{CC} Supply Current ($\overline{SHDN}$ = V_{CC}), Figures 5, 6, 11, 19	No load	MAX220		0.5	2	mA
		MAX222/232A/233A/242/243		4	10	
	3kΩ load both inputs	MAX220		12		
		MAX222/232A/233A/242/243		15		
Shutdown Supply Current	MAX222/242	T_A = +25°C		0.1	10	µA
		T_A = 0°C to +70°C		2	50	
		T_A = -40°C to +85°C		2	50	
		T_A = -55°C to +125°C		35	100	
$\overline{SHDN}$ Input Leakage Current	MAX222/242				±1	µA
$\overline{SHDN}$ Threshold Low	MAX222/242			1.4	0.8	V
$\overline{SHDN}$ Threshold High	MAX222/242		2.0	1.4		V
Transition Slew Rate	C_L = 50pF to 2500pF, R_L = 3kΩ to 7kΩ, V_{CC} = 5V, T_A = +25°C, measured from +3V to -3V or -3V to +3V	MAX222/232A/233A/242/243	6	12	30	V/µs
		MAX220	1.5	3	30	
Transmitter Propagation Delay TLL to RS-232 (normal operation), Figure 1	t_{PHLT}	MAX222/232A/233A/242/243		1.3	3.5	µs
		MAX220		4	10	
	t_{PLHT}	MAX222/232A/233A/242/243		1.5	3.5	
		MAX220		5	10	
Receiver Propagation Delay RS-232 to TLL (normal operation), Figure 2	t_{PHLR}	MAX222/232A/233A/242/243		0.5	1	µs
		MAX220		0.6	3	
	t_{PLHR}	MAX222/232A/233A/242/243		0.6	1	
		MAX220		0.8	3	
Receiver Propagation Delay RS-232 to TLL (shutdown), Figure 2	t_{PHLS}	MAX242		0.5	10	µs
	t_{PLHS}	MAX242		2.5	10	
Receiver-Output Enable Time, Figure 3	t_{ER}	MAX242		125	500	ns
Receiver-Output Disable Time, Figure 3	t_{DR}	MAX242		160	500	ns
Transmitter-Output Enable Time ($\overline{SHDN}$ goes high), Figure 4	t_{ET}	MAX222/242, 0.1µF caps (includes charge-pump start-up)		250		µs
Transmitter-Output Disable Time ($\overline{SHDN}$ goes low), Figure 4	t_{DT}	MAX222/242, 0.1µF caps		600		ns
Transmitter + to - Propagation Delay Difference (normal operation)	t_{PHLT} - t_{PLHT}	MAX222/232A/233A/242/243		300		ns
		MAX220		2000		
Receiver + to - Propagation Delay Difference (normal operation)	t_{PHLR} - t_{PLHR}	MAX222/232A/233A/242/243		100		ns
		MAX220		225		

Note 3: MAX243 $R2_{OUT}$ is guaranteed to be low when $R2_{IN}$ is ≥ 0V or is floating.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ELECTRICAL CHARACTERISTICS—MAX220/222/232A/233A/242/243 (continued)

(V_{CC} = +5V ±10%, C1–C4 = 0.1µF, MAX220, C1 = 0.047µF, C2–C4 = 0.33µF, T_A = T_{MIN} to T_{MAX}, unless otherwise noted.)

PARAMETER	CONDITIONS		MIN	TYP	MAX	UNITS
TTL/CMOS Output Leakage Current	$\overline{SHDN}$ = V_{CC} or $\overline{EN}$ = V_{CC} ($\overline{SHDN}$ = 0V for MAX222), 0V ≤ V_{OUT} ≤ V_{CC}			±0.05	±10	µA
$\overline{EN}$ Input Threshold Low	MAX242			1.4	0.8	V
$\overline{EN}$ Input Threshold High	MAX242		2.0	1.4		V
Operating Supply Voltage			4.5		5.5	V
V_{CC} Supply Current ($\overline{SHDN}$ = V_{CC}), Figures 5, 6, 11, 19	No load	MAX220		0.5	2	mA
		MAX222/232A/233A/242/243		4	10	
	3kΩ load both inputs	MAX220		12		
		MAX222/232A/233A/242/243		15		
Shutdown Supply Current	MAX222/242	T_A = +25°C		0.1	10	µA
		T_A = 0°C to +70°C		2	50	
		T_A = -40°C to +85°C		2	50	
		T_A = -55°C to +125°C		35	100	
$\overline{SHDN}$ Input Leakage Current	MAX222/242				±1	µA
$\overline{SHDN}$ Threshold Low	MAX222/242			1.4	0.8	V
$\overline{SHDN}$ Threshold High	MAX222/242		2.0	1.4		V
Transition Slew Rate	C_L = 50pF to 2500pF, R_L = 3kΩ to 7kΩ, V_{CC} = 5V, T_A = +25°C, measured from +3V to -3V or -3V to +3V	MAX222/232A/233A/242/243	6	12	30	V/µs
		MAX220	1.5	3	30	
Transmitter Propagation Delay TLL to RS-232 (normal operation), Figure 1	t_{PHLT}	MAX222/232A/233A/242/243		1.3	3.5	µs
		MAX220		4	10	
	t_{PLHT}	MAX222/232A/233A/242/243		1.5	3.5	
		MAX220		5	10	
Receiver Propagation Delay RS-232 to TLL (normal operation), Figure 2	t_{PHLR}	MAX222/232A/233A/242/243		0.5	1	µs
		MAX220		0.6	3	
	t_{PLHR}	MAX222/232A/233A/242/243		0.6	1	
		MAX220		0.8	3	
Receiver Propagation Delay RS-232 to TLL (shutdown), Figure 2	t_{PHLS}	MAX242		0.5	10	µs
	t_{PLHS}	MAX242		2.5	10	
Receiver-Output Enable Time, Figure 3	t_{ER}	MAX242		125	500	ns
Receiver-Output Disable Time, Figure 3	t_{DR}	MAX242		160	500	ns
Transmitter-Output Enable Time ($\overline{SHDN}$ goes high), Figure 4	t_{ET}	MAX222/242, 0.1µF caps (includes charge-pump start-up)		250		µs
Transmitter-Output Disable Time ($\overline{SHDN}$ goes low), Figure 4	t_{DT}	MAX222/242, 0.1µF caps		600		ns
Transmitter + to - Propagation Delay Difference (normal operation)	t_{PHLT} - t_{PLHT}	MAX222/232A/233A/242/243		300		ns
		MAX220		2000		
Receiver + to - Propagation Delay Difference (normal operation)	t_{PHLR} - t_{PLHR}	MAX222/232A/233A/242/243		100		ns
		MAX220		225		

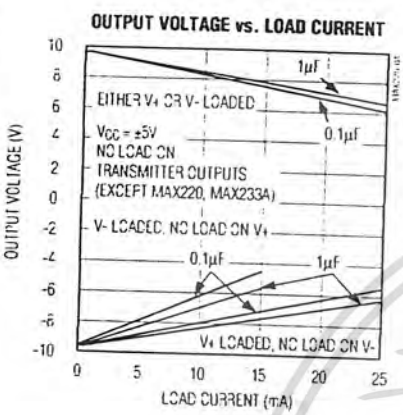
Note 3: MAX243 $R2_{OUT}$ is guaranteed to be low when $R2_{IN}$ is ≥ 0V or is floating.

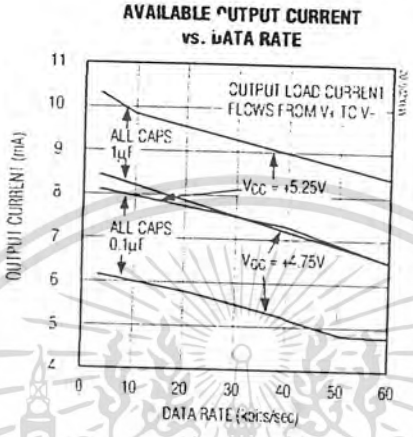
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

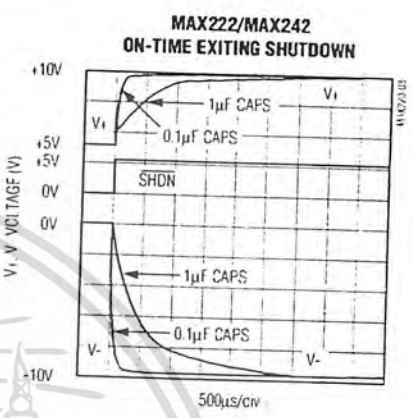
+5V-Powered, Multichannel RS-232 Drivers/Receivers

Typical Operating Characteristics

MAX220/MAX222/MAX232A/MAX233A/MAX242/MAX243







เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

DALLAS SEMICONDUCTOR

DS1307
64 X 8 Serial Real Time Clock

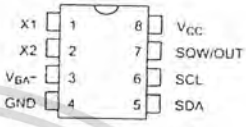
FEATURES

- Real time clock counts seconds, minutes, hours, date of the month, month, day of the week, and year with leap year compensation valid up to 2100
- 56 byte nonvolatile RAM for data storage
- 2–wire serial interface
- Programmable squarewave output signal
- Automatic power fail detect and switch circuitry
- Consumes less than 500 nA in battery backup mode at 25°C
- Optional industrial temperature range –40°C to +85°C (IND)
- Available in 8–pin DIP or SOIC

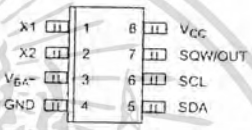
ORDERING INFORMATION

DS1307	Serial Timekeeping Chip; 8–pin DIP
DS1307Z	Serial Timekeeping Chip; 8–pin SOIC (150 mil)
DS1307N	8–pin DIP (IND)
DS1307ZN	8–pin SOIC (IND)

PIN ASSIGNMENT



DS1307 8–PIN DIP (300 MIL)



DS1307Z 8–PIN SOIC (150 MIL)

PIN DESCRIPTION

V_{CC}	– Primary Power Supply
X1, X2	– 32.768 KHz Crystal Connection
V_{BAT}	– +3 Volt Battery Input
GND	– Ground
SDA	– Serial Data
SCL	– Serial Clock
SQW/OUT	– Square wave/Output Driver

DESCRIPTION

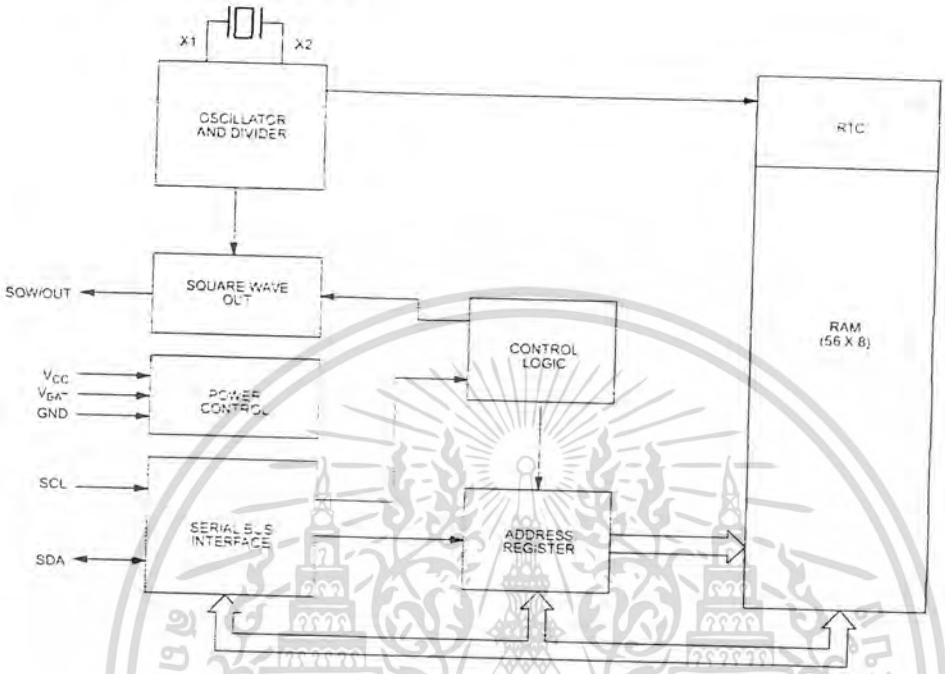
The DS1307 Serial Real Time Clock is a low power full BCD clock calendar plus 56 bytes of nonvolatile SRAM. Address and data are transferred serially via a 2–wire bi–directional bus. The clock/calendar provides seconds, minutes, hours, day, date, month, and year information. The end of the month date is automatically adjusted for months with less than 31 days, including corrections for leap year. The clock operates in either the 24–hour or 12–hour format with AM/PM indicator. The DS1307 has a built–in power sense circuit which detects power failures and automatically switches to the battery supply.

OPERATION

The DS1307 operates as a slave device on the serial bus. Access is obtained by implementing a START condition and providing a device identification code followed by a register address. Subsequent registers can be accessed sequentially until a STOP condition is executed. When V_{CC} falls below 1.25 x V_{BAT} the device terminates an access in progress and resets the device address counter. Inputs to the device will not be recognized at this time to prevent erroneous data from being written to the device from an out of tolerance system. When V_{CC} falls below V_{BAT} the device switches into a low current battery backup mode. Upon power up, the device switches from battery to V_{CC} when V_{CC} is greater than V_{BAT}+0.2V and recognizes inputs when V_{CC} is greater than 1.25 x V_{BAT}. The block diagram in Figure 1 shows the main elements of the Serial Real Time Clock. The following paragraphs describe the function of each pin.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

DS1307 BLOCK DIAGRAM Figure 1



SIGNAL DESCRIPTIONS

V_{CC}, GND – DC power is provided to the device on these pins. V_{CC} is the +5 volt input. When 5 volts are applied within normal limits, the device is fully accessible and data can be written and read. When a 3 volt battery is connected to the device and V_{CC} is below 1.25 x V_{BAT}, reads and writes are inhibited. However, the Timekeeping function continues unaffected by the lower input voltage. As V_{CC} falls below V_{BAT} the RAM and timekeeper are switched over to the external 3 volt battery.

V_{BAT} – Battery input for any standard 3 volt lithium cell or other energy source. Battery voltage must be held between 2.5 and 3.5 volts for proper operation. The nominal write protect trip point voltage at which access to the real time clock and user RAM is denied is set by the internal circuitry as 1.25 x V_{BAT} nominal. A Lithium battery with 35 mAh or greater will back up the DS1307 for more than 10 years in the absence of power.

SCL (Serial Clock Input) – SCL is used to synchronize data movement on the serial interface.

SDA (Serial Data Input/Output) – SDA is the input/output pin for the 2–wire serial interface. The SDA pin is open drain which requires an external pull–up resistor.

SQW/OUT (Square Wave/ Output Driver) – When enabled, the SQWE bit set to 1, the SQW/OUT pin outputs one of four square wave frequencies (1 Hz, 4 KHz, 8 KHz, 32 KHz). The SQW/OUT pin is open drain which requires an external pull–up resistor.

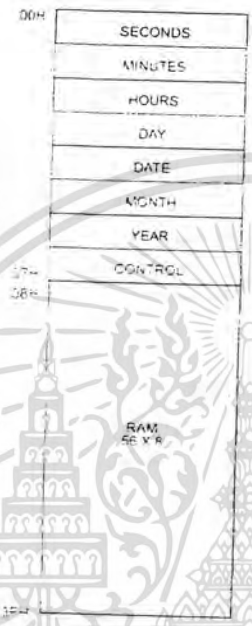
X1, X2 – Connections for a standard 32.768 KHz quartz crystal. The internal oscillator circuitry is designed for operation with a crystal having a specified load capacitance (CL) of 12.5 pF.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

RTC AND RAM ADDRESS MAP

The address map for the RTC and RAM registers of the DS1307 is shown in Figure 2. The real time clock registers are located in address locations 00h to 07h. The RAM registers are located in address locations 08h to 3Fh. During a multibyte access, when the address pointer reaches 3Fh, the end of RAM space, it wraps around to location 00h, the beginning of the clock space.

DS1307 ADDRESS MAP Figure 2



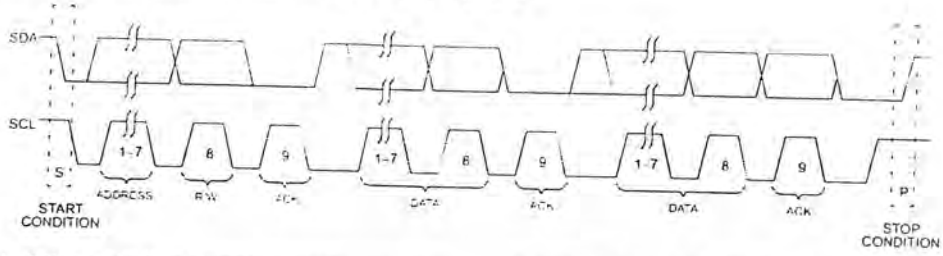
CLOCK AND CALENDAR

The time and calendar information is obtained by reading the appropriate register bytes. The real time clock registers are illustrated in Figure 3. The time and calendar are set or initialized by writing the appropriate register bytes. The contents of the time and calendar registers are in the Binary–Coded Decimal (BCD) format. Bit 7 of Register 0 is the Clock Halt (CH) bit. When this bit is set to a one, the oscillator is disabled. When cleared to a zero, the oscillator is enabled.

The DS1307 can be run in either 12–hour or 24–hour mode. Bit 6 of the hours register is defined as the 12– or 24–hour mode select bit. When high, the 12–hour mode is selected. In the 12–hour mode, bit 5 is the AM/PM bit with logic high being PM. In the 24–hour mode, bit 5 is the second 10 hour bit (20–23 hours).

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

DATA TRANSFER ON 2–WIRE SERIAL BUS Figure 5



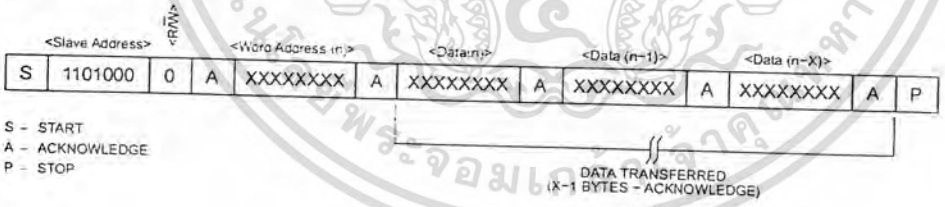
1. Data transfer from a master transmitter to a slave receiver. The first byte transmitted by the master is the slave address. Next follows a number of data bytes. The slave returns an acknowledge bit after each received byte. Data is transfered with the most significant bit (MSB) first.
2. Data transfer from a slave transmitter to a master receiver. The first byte (the slave address) is transmitted by the master. The slave then returns an acknowledge bit. This is followed by the slave transmitting a number of data bytes. The master returns an acknowledge bit after all received bytes other than the last byte. At the end of the last received byte, a 'not acknowledge' is returned.

The master device generates all of the serial clock pulses and the START and STOP conditions. A transfer is ended with a STOP condition or with a repeated START condition. Since a repeated START condition is also the beginning of the next serial transfer, the bus will not be released. Data is transfered with the most significant bit (MSB) first.

The DS1307 may operate in the following two modes:

1. Slave receiver mode (DS1307 write mode): Serial data and clock are received through SDA and SCL. After each byte is received an acknowledge bit is transmitted. START and STOP conditions are recognized as the beginning and end of a serial transfer. Address recognition is performed by hardware after reception of the slave address and direction bit (See Figure 6). The address byte is the first byte received after the start condition is generated by the master. The address byte contains the 7 bit DS1307 address, which is 1101000, followed by the direction bit (R/W) which for a write is a 0. After receiving and decoding the address byte the DS1307 outputs an acknowledge on the SDA line. After the DS1307 acknowledges the slave address + write bit, the master transmits a register address to the DS1307 This will set the register pointer on the DS1307. The master will then begin transmitting each byte of data with the DS1307 acknowledging each byte received. The master will generate a stop condition to terminate the data write.

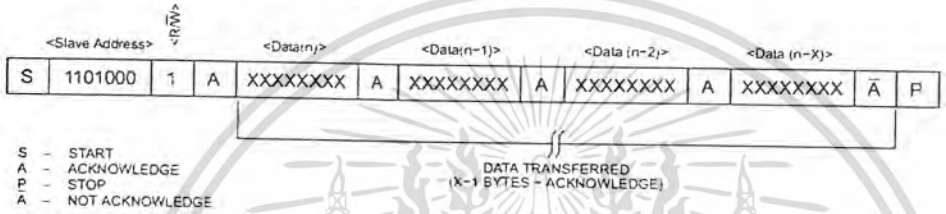
DATA WRITE – SLAVE RECEIVER MODE Figure 6



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2. Slave transmitter mode (DS1307 read mode): The first byte is received and handled as in the slave receiver mode. However, in this mode, the direction bit will indicate that the transfer direction is reversed. Serial data is transmitted on SDA by the DS1307 while the serial clock is input on SCL. START and STOP conditions are recognized as the beginning and end of a serial transfer (See Figure 7). The address byte is the first byte received after the start condition is generated by the master. The address byte contains the 7 bit DS1307 address, which is 1101000, followed by the direction bit (R/W\) which for a read is a 1. After receiving and decoding the address byte the DS1307 inputs an acknowledge on the SDA line. The DS1307 then begins to transmit data starting with the register address pointed to by the register pointer. If the register pointer is not written to before the initiation of a read mode the first address that is read is the last one stored in the register pointer. The DS1307 must receive a Not Acknowledge to end a read.

DATA READ – SLAVE TRANSMITTER MODE Figure 7



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ABSOLUTE MAXIMUM RATINGS*

Voltage on Any Pin Relative to Ground	–0.5V to +7.0V
Operating Temperature	0°C to 70°C
Storage Temperature	–55°C to +125°C
Soldering Temperature	260°C for 10 seconds

* This is a stress rating only and functional operation of the device at these or any other conditions above those indicated in the operation sections of this specification is not implied. Exposure to absolute maximum rating conditions for extended periods of time may affect reliability.

The Dallas Semiconductor DS1307 is built to the highest quality standards and manufactured for long term reliability. All Dallas Semiconductor devices are made using the same quality materials and manufacturing methods. However, standard versions of the DS1307 are not exposed to environmental stresses, such as burn–in, that some industrial applications require. Products which have successfully passed through this series of environmental stresses are marked IND or N, denoting their extended operating temperature and reliability rating. For specific reliability information on this product, please contact the factory at (972) 371–4448.

RECOMMENDED DC OPERATING CONDITIONS

(0°C to 70°C)

PARAMETER	SYMBOL	MIN	TYP	MAX	UNITS	NOTES
Supply Voltage	V_{CC}	4.5	5.0	5.5	V	1
Logic 1	V_{IH}	2.2		V_{CC}+0.3	V	1
Logic 0	V_{IL}	–0.3		+0.8	V	1
V_{BAT} Battery Voltage	V_{BAT}	2.5		3.5	V	1

DC ELECTRICAL CHARACTERISTICS

(0°C to 70°C; V_{CC}=4.5V to 5.5V)

PARAMETER	SYMBOL	MIN	TYP	MAX	UNITS	NOTES
Input Leakage	I_{LI}			1	µA	10
I/O Leakage	I_{LO}			1	µA	11
Logic 0 Output	V_{OL}			0.4	V	2
Active Supply Current	I_{CCA}			1.5	mA	9
Standby Current	I_{CCS}			200	µA	3
Battery Current (OSC ON); SQW/OUT OFF	I_{BAT1}		300	500	nA	4
Battery Current (OSC ON); SQW/OUT ON (32 KHz)	I_{BAT2}		480	800	nA	4

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

AC ELECTRICAL CHARACTERISTICS

(0°C to 70°C; V_{CC}=4.5V to 5.5V)

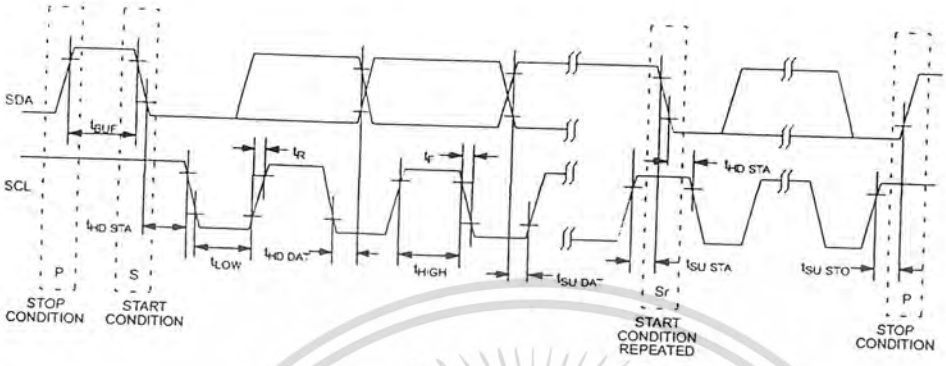
PARAMETER	SYMBOL	MIN	TYP	MAX	UNITS	NOTES
SCL Clock Frequency	f_{SCL}	0		100	KHz	
Bus Free Time Between a STOP and START Condition	t_{BUF}	4.7			µs	
Hold Time (Repeated) START Condition	$t_{HD:STA}$	4.0			µs	5
LOW Period of SCL Clock	t_{LOW}	4.7			µs	
HIGH Period of SCL Clock	t_{HIGH}	4.0			µs	
Set–up Time for a Repeated START Condition	$t_{SU:STA}$	4.7			µs	
Data Hold Time	$t_{HD:DAT}$	0			µs	6, 7
Data Set–up Time	$t_{SU:DAT}$	250			ns	
Rise Time of Both SDA and SCL Signals	t_R			1000	ns	
Fall Time of Both SDA and SCL Signals	t_F			300	ns	
Set–up Time for STOP Condition	$t_{SU:STO}$	4.7			µs	
Capacitive Load for each Bus Line	C_B			400	pF	8
I/O Capacitance	$C_{I/O}$		10		pF	
Crystal Capacitance			12.5		pF	

NOTES:

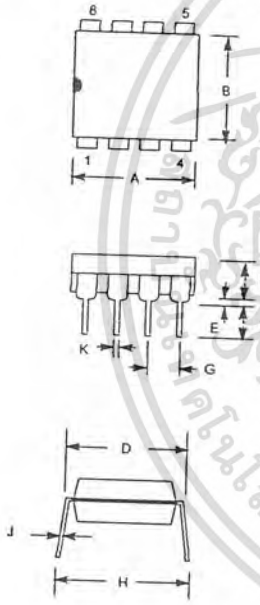
1. All voltages are referenced to ground.
2. Logic zero voltages are specified at a sink current of 5 mA at V_{CC}=4.5V, V_{OL}=GND for capacitive loads.
3. I_{CCS} specified with V_{CC}=5.0V and SDA, SCL=5.0V.
4. V_{CC}=0V, V_{BAT}=3V.
5. After this period, the first clock pulse is generated.
6. A device must internally provide a hold time of at least 300 ns for the SDA signal (referred to the V_{IHMIN} of the SCL signal) in order to bridge the undefined region of the falling edge of SCL.
7. The maximum $t_{HD:DAT}$ has only to be met if the device does not stretch the LOW period (t_{LOW}) of the SCL signal.
8. C_B – total capacitance of one bus line in pF.
9. I_{CCA} – SCL clocking at max frequency = 100 KHz.
10. SCL only.
11. SDA and SQW/OUT

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

TIMING DIAGRAM



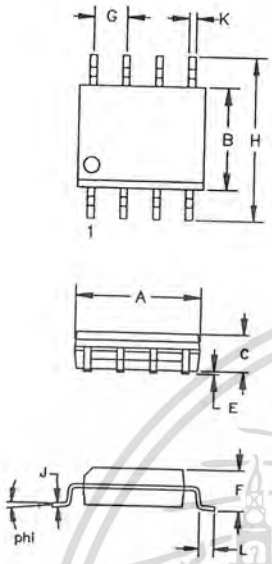
DS1307 64 X 8 SERIAL REAL TIME CLOCK 8-PIN DIP



PKG	8-PIN	
DIM	MIN	MAX
A IN. MM	0.360 9.14	0.400 10.16
B IN. MM	0.240 6.10	0.260 6.60
C IN. MM	0.120 3.05	0.140 3.56
D IN. MM	0.300 7.62	0.325 8.26
E IN. MM	0.015 0.38	0.040 1.02
F IN. MM	0.120 3.04	0.140 3.56
G IN. MM	0.090 2.29	0.110 2.79
H IN. MM	0.320 8.13	0.370 9.40
J IN. MM	0.008 0.20	0.012 0.30
K IN. MM	0.015 0.38	0.021 0.53

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

DS1307Z 64 X 8 SERIAL REAL TIME CLOCK 8–PIN SOIC (150 MIL)



PKG	8–PIN (150 MIL)	
DIM	MIN	MAX
A IN. MM	0.188 4.78	0.196 4.98
B IN. MM	0.150 3.81	0.158 4.01
C IN. MM	0.048 1.22	0.062 1.57
E IN. MM	0.004 0.10	0.010 0.25
F IN. MM	0.053 1.35	0.069 1.75
G IN. MM	0.050 BSC 1.27 BSC	
H IN. MM	0.230 5.84	0.244 6.20
J IN. MM	0.007 0.18	0.011 0.28
K IN. MM	0.012 0.30	0.020 0.51
L IN. MM	0.016 0.41	0.050 1.27
phi	0°	8°

56–G2008–001

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้