สำนักหอสมุดกลาง พระจอมเกล้าลาดกระบัง

ตัวส่งสัญญาณอนาลอกขนาดจิ๋ว

นางสาววันเพ็ญ สกุลกิจไพบูลย์
นายศุภกัลย์ วัฒนการุณ

TK
R2357
2538
เลขหมู่.....
เลขทะเบียน.....
วัน,เดือน,ปี.....
b1355196

โครงงานพิเศษนี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรวิทยาศาสตรบัณฑิต
ภาควิชาฟิสิกส์ประยุกต์
คณะวิทยาศาสตร์
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
ปีการศึกษา 2538

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Miniature Analog Signal Sender

Miss.Wanpen Sakulkitpaibool

Mr.Suppakun Wattanakaroon

A Special Project Submitted in Partial Fulfillment of the

Requirement for the Degree of Bachelor of Science

Department of Applied Physics

Faculty of Science

King Mongkut's Institute of Technology Ladkrabang

1995

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หัวข้อโครงงานพิเศษ	ตัวส่งสัญญาณอนาลอกขนาดจิ๋ว
โดย	นางสาววันเพ็ญ สกุลกิจไพบูลย์
	นายศุภกัลย์ วัฒนการุณ
ภาควิชา	ฟิสิกส์ประยุกต์
อาจารย์ที่ปรึกษา	รศ.สุรพล รักวิจัย
	ผศ.วิชิต ศิริโชติ

ภาควิชาฟิสิกส์ประยุกต์ คณะวิทยาศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง อนุมัติให้นับโครงงานพิเศษนี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรวิทยาศาสตรบัณฑิต

_____ หัวหน้าภาควิชาฟิสิกส์ประยุกต์

(รศ.สุรพล รักวิจัย)

คณะกรรมการสอบโครงงานพิเศษ

_____ ประธานกรรมการ

(ผศ.วิชิต ศิริโชติ)

_____ กรรมการ

(รศ.สุรพล รักวิจัย)

_____ กรรมการ

(ผศ.ดร.วราวุฒิ เถาลัดดา)

_____ กรรมการ

(อ.ธีรวัฒน์ ประกอบผล)

ลิขสิทธิ์ของภาควิชาฟิสิกส์ประยุกต์ คณะวิทยาศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หัวข้อโครงงานพิเศษ	ตัวส่งสัญญาณอนาลอกขนาดจิ๋ว
โดย	นางสาววันเพ็ญ สกุลกิจไพบูลย์
	นายศุภกัลย์ วัฒนการุณ
อาจารย์ที่ปรึกษา	รศ.สุรพล รักวิจัย
	ผศ.วิชิต ศิริโชติ
ภาควิชา	ฟิสิกส์ประยุกต์
ปีการศึกษา	2538

บทคัดย่อ

ตัวส่งสัญญาณอนาลอกขนาดจิ๋วได้พัฒนาขึ้นเพื่อประยุกต์ใช้ในการวัดค่าสัญญาณอนาลอกโดยที่สามารถวัดค่าได้ในช่วง -99 มิลลิโวลต์ ถึง +999 มิลลิโวลต์ ด้วยค่าความละเอียด 1 มิลลิโวลต์ โดยใช้ตัวแปลงสัญญาณอนาลอกเป็นดิจิตอลซนิดอินทิเกรตแบบสโลปคู่ สำหรับการวัดสัญญาณอนาลอกที่เปลี่ยนแปลงระดับสัญญาณอย่างช้า ๆ ด้วยอัตราการสุ่ม 96 ครั้งต่อวินาที และจะใช้ชิปไมโครคอนโทรลเลอร์เบอร์ AT89C2051 ขนาด 20 ขา ในการควบคุมตัวแปลงสัญญาณและส่งข้อมูลทางดิจิตอลให้แก่คอมพิวเตอร์ศูนย์กลาง โดยใช้ช่องของการสื่อสารดังนี้ 1.) EIA RS232C สำหรับการเชื่อมต่อระยะใกล้ 2.) ลูปกระแส 20 มิลลิแอมป์ สำหรับการเชื่อมต่อระยะไกล และ 3.) การเชื่อมต่อด้วยอินฟราเรด โดยที่การเชื่อมต่อในแบบที่ 1 และ 2 จะเป็นการใช้สายทองแดงในการเชื่อมต่อ ในขณะที่แบบสุดท้ายจะเป็นการเชื่อมต่อแบบไร้สาย ซึ่งออกแบบมาสำหรับงานที่ไม่สามารถใช้สายเข้าถึงได้ ลักษณะของตัวส่งสัญญาณอนาลอกนี้จะมีขนาดเล็กซึ่งจะทำให้ง่ายในการเคลื่อนย้ายและเหมาะสมที่จะนำไปใช้ในงานต่าง ๆ ได้มากมาย อีกทั้งในโครงงานนี้ยังได้พัฒนาตัวอ่านค่าสัญญาณอัตโนมัติเพื่อใช้สำหรับการส่งคำสั่งและจัดเก็บข้อมูลทางดิจิตอลที่ถูกส่งออกมาจากตัวส่งสัญญาณอนาลอกขนาดจิ๋วด้วยช่วงเวลาที่คงที่ โดยจะสามารถโปรแกรมใช้งานได้อย่างง่ายดาย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Special Project Title	Miniature Analog Signal Sender
Name	Miss.Wanpen Sakulkitpaibool
	Mr.Suppakun Wattanakaroon
Special Project Advisor	Assoc.Prof.Surapol Rakvijai
	Asst.Prof.Wichit Sirichote
Department	Applied Physics
Acadamic Year	1995

Abstract

A miniature analog signal sender has been developed for using in the application of measuring analog signal remotely. The sender is capable of measuring the analog signal ranging from -99 mV to +999 mV with a resolution of 1 mV. The analog-to-digital converter used in the sender is a dual slope integrating type. The converter is choose for best performance of measuring a slowly varying analog signal with a sampling rate of 96 Hz. The 20 pins microcontroller chip, AT89C2051 controls the converter and signaling transmitted digital data to the host computer via three communication channels; i.e. 1.) EIA RS232C for short range link, 2.) a 20 mA current loop for long range link and 3.) Infrared link. The first and second links use copper wire for connection while the latter is wireless link designed for hard-to-reach by wire applications. The sender is very small and compact in size which can easily move and suitable for many applications . In addition, we also have developed an automatic scanner using for commanding and recording digital data sent by the senders with the sampling period can be programmed easily.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กิตติกรรมประกาศ

โครงงานพิเศษนี้สามารถเสร็จสมบูรณ์ได้ด้วยความช่วยเหลือจากบุคคลต่าง ๆ ดังนี้

บุพการี ที่ให้ความอุปการะในทุก ๆ ด้านจนสำเร็จการศึกษา

รศ.สุรพล รักวิจัย และ
ผศ.วิชิต ศิริโชติ ที่ให้คำแนะนำต่าง ๆ ในการทำโครงงานพิเศษนี้

คุณศิระ ริกุลสุรกาน ที่เอื้อเฟื้อคอมพิวเตอร์และให้คำแนะนำต่าง ๆ

องค์การโทรศัพท์แห่งประเทศไทย ที่เอื้อเฟื้อสายสื่อสารในการทดสอบ

เพื่อน ๆ พี่ ๆ น้อง ๆ ทุกคน ที่คอยให้กำลังใจและให้ความช่วยเหลือต่าง ๆ

คณะกรรมการทุกท่าน ที่ตรวจทานรายงานฉบับนี้

ภาควิชาฟิสิกส์ประยุกต์ ที่ให้ความอนุเคราะห์เครื่องมือและอุปกรณ์ต่าง ๆ

ขอขอบคุณอย่างจริงใจ

วันเพ็ญ สกุลกิจไพบูลย์
ศุภกัลย์ วัฒนการุณ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ

	หน้า
บทคัดย่อภาษาไทย	ก
บทคัดย่อภาษาอังกฤษ	ข
กิตติกรรมประกาศ	ค
สารบัญรูป	ฉ
สารบัญตาราง	ซ
บทที่ 1 บทนำ	1
บทที่ 2 ทฤษฎีที่เกี่ยวข้อง	3
2.1 ระบบสื่อสารข้อมูลเบื้องต้น	3
2.1.1 การสื่อสารแบบซิงโครนัส	3
2.1.2 การสื่อสารแบบอะซิงโครนัส	4
2.2 พื้นฐานของตัวแปลงสัญญาณอนาลอกเป็นดิจิตอล	5
2.3 ตัวแปลงสัญญาณอนาลอกเป็นดิจิตอลแบบสโลปคู่	6
2.4 ชนิดของการสื่อสารที่เลือกใช้	9
2.4.1 RS232	9
2.4.2 ลูปกระแส	11
2.4.3 อินฟราเรด	13
2.5 การสร้างฐานเวลาให้แก่ระบบ	14
2.6 วิธีการสร้างฐานเวลาจริงให้แก่ไมโครคอนโทรลเลอร์ตระกูล 51	14
2.7 ไมโครคอนโทรลเลอร์ AT89C2051	18
บทที่ 3 การดำเนินการวิจัย	21
3.1 ตัวส่งสัญญาณอนาลอกขนาดจิ๋ว	21
3.1.1 การเชื่อมต่อทางการวัดสัญญาณอนาลอก	21
3.1.2 การทำงานของโปรแกรม	24
3.1.3 วงจรแหล่งจ่ายไฟ	24
3.1.4 ช่องสัญญาณการสื่อสารแบบต่าง ๆ ที่เลือกใช้	27

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หน้า

3.2 ตัวรับและแปลงสัญญาณ 29

3.2.1 วงจรแปลงการสื่อสารแบบลูปกระแสเป็น RS232 30

3.2.2 วงจรแปลงการสื่อสารแบบอินฟราเรดเป็น RS232 31

3.2.3 การทำงานของโปรแกรม 33

3.2.4 การเชื่อมต่อตัวรับและแปลงสัญญาณกับตัวส่งสัญญาณอนาลอกขนาดจิ๋ว 34

3.3 ตัวอ่านค่าสัญญาณและจัดเก็บข้อมูล 35

3.3.1 การเชื่อมต่อกับซิปกำเนิดเวลาจริง 36

3.3.2 การเชื่อมต่อกับหน่วยความจำภายนอก 43

3.3.3 การทำงานของโปรแกรม 44

บทที่ 4 การทดลองและผลการทดลอง 49

4.1 การทดสอบความถูกต้องของการแปลงข้อมูล 49

4.2 ความสามารถในการสื่อสารแบบลูปกระแส 52

4.3 ความสามารถในการสื่อสารอินฟราเรด 54

บทที่ 5 บทสรุป 55

5.1 สรุปและการประยุกต์ใช้งาน 55

5.2 แนวทางในการพัฒนาโครงงานต่อไป 55

ภาคผนวก ก โปรแกรมการทำงาน

ภาคผนวก ข วงจร.

ภาคผนวก ค ข้อมูลอุปกรณ์

เอกสารอ้างอิง

ประวัติผู้เขียน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูป

		หน้า
รูป 1.1	แผนผังแสดงโครงสร้างของตัวส่งสัญญาณอนาลอกขนาดจิ๋ว	2
รูป 2.1	การให้สัญญาณนาฬิกาเพื่อการรับส่งแบบซิงโครนัส	4
รูป 2.2	การส่งข้อมูลแบบอะซิงโครนัส	4
รูป 2.3	A/D คอนเวอร์ทเตอร์แบบสโลปคู่	8
รูป 2.4	ลักษณะการเชื่อมต่อสัญญาณแบบ RS232	10
รูป 2.5	ลักษณะการเชื่อมต่อสัญญาณแบบลูปกระแส	11
รูป 2.6	ลักษณะวงจรสำหรับส่งข้อมูล	12
รูป 2.7	ลักษณะวงจรสำหรับรับข้อมูล	12
รูป 2.8	ลักษณะการมอดดูเลทสัญญาณ	13
รูป 2.9	ลักษณะโครงสร้างภายในของชิปกำเนิดเวลาจริง DS1202	17
รูป 3.1	ลักษณะตัวส่งสัญญาณอนาลอกขนาดจิ๋ว	21
รูป 3.2	วงจรการเชื่อมต่อทางการวัดสัญญาณอนาลอก	22
รูป 3.3	ไทม์มิงไดอะแกรมของ CA3162	22
รูป 3.4	แผนผังแสดงลำดับการทำงานของตัวส่งสัญญาณอนาลอกขนาดจิ๋ว	25
รูป 3.5	แผนผังแสดงลำดับการทำงานของการอ่านค่าจาก A/D แปลงสัญญาณอนาลอกเป็นสัญญาณดิจิตอลทั้ง 8 ช่องสัญญาณ	26
รูป 3.6	วงจรแหล่งจ่ายไฟ	27
รูป 3.7	วงจรการสื่อสารแบบ RS232	27
รูป 3.8	วงจรการสื่อสารแบบลูปกระแส	28
รูป 3.9	วงจรการสื่อสารอินฟราเรด	28
รูป 3.10	ไดอะแกรมและลักษณะสัญญาณของตัวรับอินฟราเรด	29
รูป 3.11	ไดอะแกรมของตัวรับและแปลงสัญญาณ	30
รูป 3.12	วงจรแปลงการสื่อสารแบบลูปกระแสเป็น RS232	30
รูป 3.13	วงจรแปลงการสื่อสารแบบอินฟราเรดเป็น RS232	31
รูป 3.14	แผนผังแสดงลำดับการทำงานของตัวรับและแปลงสัญญาณ	33
รูป 3.15	การเชื่อมต่อกันด้วยการสื่อสารแบบลูปกระแส	34

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หน้า

รูป 3.16 การเชื่อมต่อกันด้วยอินฟราเรด 34
รูป 3.17 ลักษณะของตัวอ่านค่าสัญญาณและจัดเก็บข้อมูล 35
รูป 3.18 ลักษณะการเชื่อมต่อกับชิปกำเนิดเวลาจริง 36
รูป 3.19 ลักษณะการส่งไบต์คำสั่ง 37
รูป 3.20 แผนผังแสดงลำดับการทำงานการอ่าน / เขียนข้อมูลเวลาครั้งละ 1 ไบต์ ของชิปกำเนิดเวลาจริง DS1202 42
รูป 3.21 ลักษณะการเชื่อมต่อกับหน่วยความจำภายนอก 43
รูป 3.22 แผนผังแสดงลำดับการทำงานของตัวอ่านค่าสัญญาณและจัดเก็บข้อมูล 48
รูป 4.1 ไดอะแกรมของการทดลองความถูกต้องของการแปลงข้อมูล 49
รูป 4.2 กราฟการทดสอบความถูกต้องของการแปลงข้อมูล 51
รูป 4.3 กราฟการทดสอบความสามารถในการสื่อสารแบบลูปกระแส 52
รูป 4.4 การทดสอบความสามารถในการสื่อสารแบบลูปกระแส 53
รูป 4.5 การทดสอบความสามารถในการสื่อสารแบบอินฟราเรด 54

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญตาราง

		หน้า
ตาราง 2.1	การจัดตำแหน่งสัญญาณ RS232	10
ตาราง 2.2	ฟังก์ชันพิเศษ	20
ตาราง 3.1	จัมเปอร์ของตัวส่งสัญญาณอนาลอกขนาดจิ๋ว	23
ตาราง 3.2	จัมเปอร์ของตัวรับและแปลงสัญญาณ	32
ตาราง 3.3	การใช้เบิซทโหมด	37
ตาราง 3.4	ข้อมูลการทำงานของรีจิสเตอร์และหน่วยความจำใน DS1202	38
ตาราง 3.5	จัมเปอร์ของตัวอ่านค่าสัญญาณและจัดเก็บข้อมูล	48
ตาราง 4.1	ผลการทดลองความถูกต้องของการแปลงข้อมูล	50
ตาราง 4.2	ผลการทดลองความสามารถในการสื่อสารแบบลูปกระแส	52

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 1
บทนำ

ในปัจจุบันในงานวิจัยทางวิทยาศาสตร์บางครั้งจะต้องเข้าไปเกี่ยวข้องกับสารเคมีที่เป็นพิษหรือสภาพแวดล้อมที่เป็นอันตรายต่อผู้ทำการวิจัย สิ่งเหล่านี้จะเป็นอุปสรรคในการดำเนินงานวิจัย ทำให้งานวิจัยเกิดความล่าช้า และเกิดอันตรายต่อบุคลากรซึ่งถือว่าเป็นทรัพยากรที่มีความสำคัญมากที่สุด ดังนั้นเราจึงต้องทำการลดความเสี่ยงเหล่านี้โดยที่จะต้องทำให้บุคลากรที่ทำการวิจัยกับงานวิจัยแยกออกจากกัน ซึ่งจุดสำคัญที่สุดก็คือการวัดค่าสัญญาณอนาลอกต่าง ๆ ซึ่งอาจจะมาจากตัวตรวจวัดต่าง ๆ หรือจากอุปกรณ์ที่เราทำการทดสอบ โดยที่ค่าต่าง ๆ เหล่านี้จะต้องมีการตรวจวัดและส่งค่ามาให้กับบุคลากรที่ทำการวิจัยเพื่อใช้ในการวิเคราะห์งานวิจัยเหล่านั้นต่อไป จากแนวความคิดเหล่านี้ทำให้ได้เกิดโครงงานนี้ขึ้นมา โดยที่โครงงานนี้จะเป็นการสร้างตัวส่งสัญญาณอนาลอกขึ้นโดยที่จะทำการแปลงค่าศักดาไฟฟ้าและส่งค่าที่แปลงได้ขึ้นบนคอมพิวเตอร์เพื่อจะแสดงผลให้บุคลากรผู้ทำการวิจัยได้ทราบค่าโดยที่ไม่ต้องไปทำการวัดด้วยตนเอง อีกทั้งยังอำนวยความสะดวกแก่บุคลากรผู้ทำการวิจัยโดยที่จะสามารถวัดค่าสัญญาณอนาลอกจากระยะไกลประมาณเป็นกิโลเมตรได้ เพื่อให้เกิดความรวดเร็วในงานวิจัย โดยที่ไม่ต้องเดินทางไปทำการวัดในบริเวณนั้น และอีกทั้งในงานวิจัยบางครั้งจะต้องหลีกเลี่ยงการใช้สายในการทดลองเนื่องจากสภาพแวดล้อมจะทำให้เกิดความเสียหายแก่สายได้จึงต้องมีการเพิ่มลักษณะการส่งข้อมูลโดยที่ไม่ใช้สาย อีกทั้งโครงงานนี้ก็จะสร้างสิ่งอำนวยความสะดวกเพื่อที่จะเพิ่มประสิทธิภาพของการวัดสัญญาณอนาลอกให้มากยิ่งขึ้น

จากที่กล่าวมาทั้งหมดทำให้ได้เป็นโครงงานตัวส่งสัญญาณอนาลอกขนาดจิ๋วขึ้น โดยที่โครงงานนี้จะมีวัตถุประสงค์ คือ

1) สร้างระบบตรวจวัดสัญญาณอนาลอกระยะใกล้จากบริเวณต่าง ๆ หลาย ๆ จุดมาแสดงให้เห็นที่สถานีศูนย์กลางรวมเดียวกัน

2) สร้างระบบตรวจวัดสัญญาณอนาลอกระยะใกล้ไร้สายจากบริเวณต่าง ๆ หลาย ๆ จุดมาแสดงที่สถานีศูนย์กลางรวมเดียวกัน

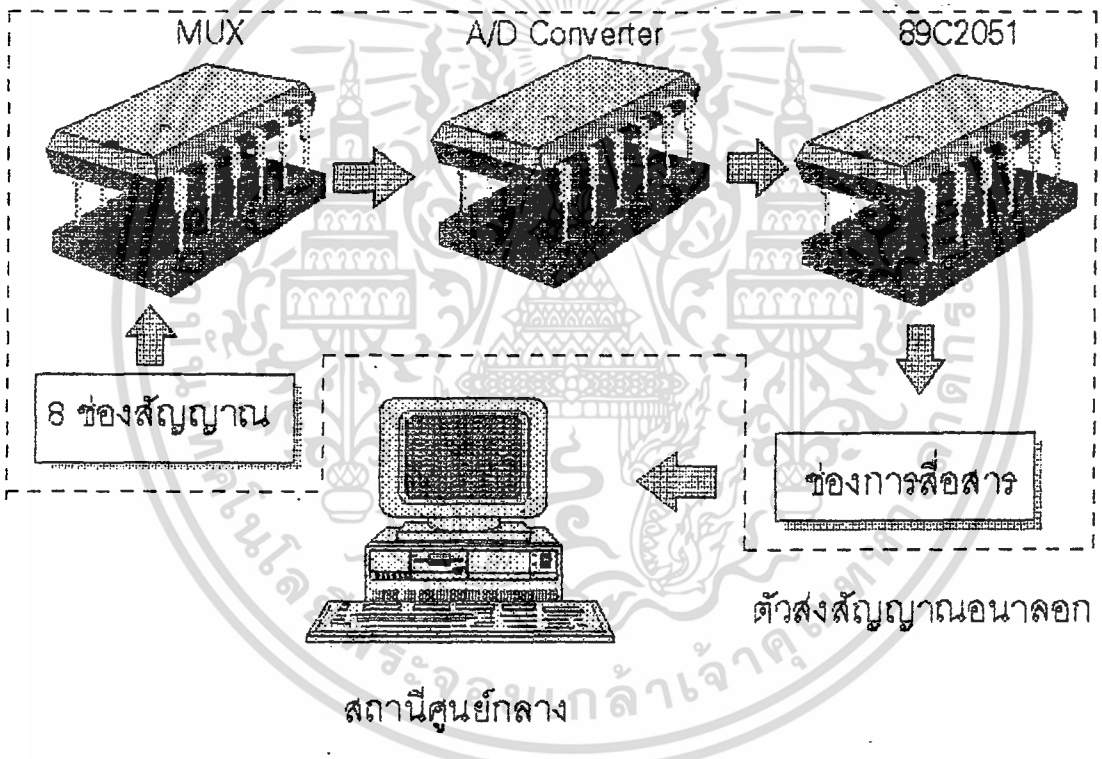
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3) สร้างระบบตรวจวัดสัญญาณอนาลอกระยะไกลจากบริเวณต่าง ๆ หลาย ๆ จุดมาแสดงที่สถานีศูนย์กลางรวมเดียวกัน

4) สร้างระบบตรวจเก็บข้อมูลเพื่อประสิทธิภาพสูงสุดของระบบตรวจวัดสัญญาณอนาลอก

5) เพื่อเป็นโครงการนำร่องในการสร้างศักยภาพทางเทคโนโลยีด้านการสื่อสารข้อมูลทางดิจิตอลระยะไกล (Digital Telemetry) สำหรับเพื่อนำไปประยุกต์ใช้งานในด้านการตรวจวัดสัญญาณต่าง ๆ ในอนาคตต่อไป

ลักษณะโครงสร้างของตัวส่งสัญญาณอนาลอกขนาดจิ๋วได้แสดงดังรูป 1.1



รูป 1.1 แผนผังแสดงโครงสร้างของตัวส่งสัญญาณอนาลอกขนาดจิ๋ว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 2
ทฤษฎีที่เกี่ยวข้อง

2.1 ระบบการสื่อสารข้อมูลเบื้องต้น

ในชีวิตประจำวันของมนุษย์เรานั้น แทบจะกล่าวได้ว่าต้องเกี่ยวข้องกับการสื่อสารในรูปแบบใดรูปแบบหนึ่งตลอดเวลาไม่ว่าจะเป็นการสนทนากัน การอ่านหนังสือ การรับส่งจดหมาย การพูดโทรศัพท์ การดูภาพยนต์หรือการนั่งฟังการบรรยายล้วนเป็นการสื่อสารซึ่งมีรูปแบบของการสื่อสารที่แตกต่างกันออกไปตามคุณสมบัติเฉพาะของแต่ละชนิดซึ่งล้วนแล้วแต่มีจุดมุ่งหมายที่จะส่งข่าวสารจากจุดหนึ่งไปยังอีกจุดหนึ่งอย่างถูกต้องทั้งสิ้น

การรับส่งเป็นรูปแบบหนึ่งของการสื่อสารซึ่งข้อมูลที่รับส่งกันนั้นเป็นสัญญาณในรูปเลขฐานสอง 0 หรือ 1 ถ้าพิจารณารูปแบบของการส่งข้อมูลจะแยกได้เป็น 2 ชนิด คือ

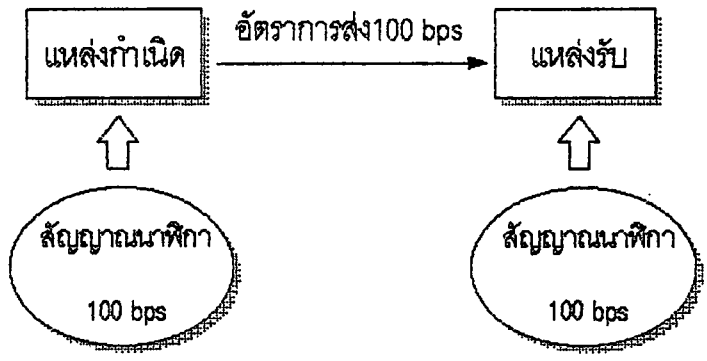
1. แบบซิงโครนัส (synchronous transmission)
2. แบบอะซิงโครนัส (asynchronous transmission)

2.1.1 การสื่อสารแบบซิงโครนัส

เป็นการรับส่งที่ด้านรับต้องรู้ว่าจะรับบิตแรกจากสายส่งนั้นเมื่อใด และหลังจากรับมาแล้วจะรับบิตที่ 2,3,. เมื่อใด ซึ่งสามารถกระทำโดยเพิ่มสัญญาณนาฬิกาเข้าไปที่จุดปลายทางของระบบทั้งสองด้านดังรูป 2.1

ซึ่งในบางระบบอาจใช้วิธีการส่งสัญญาณนาฬิกาจากด้านส่งไปด้านรับด้วย ซึ่งทางด้านรับจะนำสัญญาณที่ได้รับมาเป็นตัวขับสัญญาณนาฬิกาของตัวมันเองเพื่อให้สัญญาณนาฬิกาทางด้านรับและด้านส่งเท่ากันพอดี

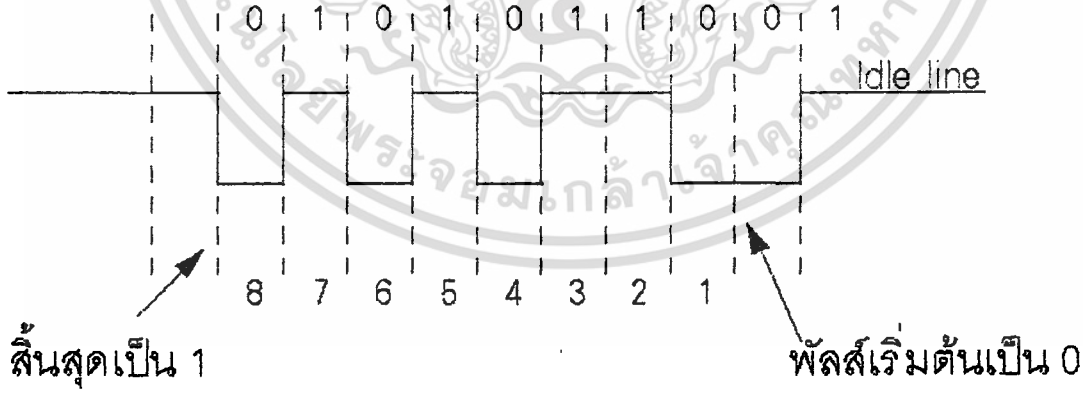
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูป 2.1 การให้สัญญาณนาฬิกาเพื่อการรับส่งแบบซิงโครนัส

2.1.2 การสื่อสารแบบอะซิงโครนัส

เมื่อจะทำการส่งแบบอะซิงโครนัสตัวอักขระจะถูกส่งออกไปในเวลาใด ๆ ก็ได้ โดยไม่จำเป็นต้องมีความสัมพันธ์ระหว่างตัวอักขระโดยจะต้องมีเวลาที่แน่นอนอย่างไร แล้วแต่ตัวอักขระแต่ละตัว อาจจะทิ้งช่วงห่างกันเท่าใดก็ได้เพราะอักขระแต่ละตัวจะมีบิตที่คอยบอกการเริ่มต้นและการสิ้นสุดของตัวอักขระตัวนั้น (start bit , stop bit) ซึ่งในทางปฏิบัติทางด้านรับจะทราบได้ว่ามีบิตเริ่มต้นเข้ามาแล้วเริ่มโดยการเปลี่ยนสถานะจาก 1 เป็น 0 หรือ จาก 0 เป็น 1 ก็ได้ (แต่ส่วนใหญ่มักให้เริ่มเปลี่ยนจาก 1 เป็น 0) ดังแสดงในรูป 2.2



รูป 2.2 การส่งข้อมูลแบบอะซิงโครนัส

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ทางด้านส่งจะเริ่มส่งบิตเริ่มต้นโดยการเปลี่ยนสถานะของค่าปกติที่เป็น 1 ไปเป็น 0 หลังจากนั้นจะส่งบิตของข่าวสารสำหรับอักขระหนึ่งตัวออกไปในขณะที่ด้านรับทราบถึงการเปลี่ยนสถานะจาก 1 เป็น 0 เป็นการบอกจุดเริ่มต้นของการทำงานว่า จะเริ่มภายหลังเวลาหรือเท่ากับครึ่งเวลาของความกว้าง 1 บิต หลังจากนั้นทางด้านรับก็จะสุ่มตัวอย่างสถานะของสายทุก ๆ ช่วงเวลา 1 บิต เพื่อหาค่ารหัสของตัวอักขระที่ส่งมาจนกว่าจะครบ 8 บิต (ถ้าเป็นรหัสแอสกี) และบิตถัดไปจะเป็นบิตสิ้นสุด หลังจากนั้นจะเว้นช่วงเวลาไปอีกนานเท่าใดก็ได้ในการเริ่มต้นอักขระตัวใหม่เพราะบิตเริ่มต้นจะคอยบอกถึงการเริ่มต้น

ด้วยวิธีการส่งแบบมีบิตเริ่มต้นและจบด้วยบิตสิ้นสุดนี้บางครั้งจึงเรียกการส่งแบบนี้ว่าการส่งแบบมีเริ่มต้นและสิ้นสุด (start - stop transmission) บิตสิ้นสุดนั้นจะแตกต่างกันไปในแต่ละระบบ เช่น การส่งรหัสแอสกี บิตสิ้นสุดจะมีความกว้างเท่ากับความกว้างของข้อมูล 1 หรือ 2 บิต ซึ่งจะเป็นช่วงเวลาที่ทางด้านรับทำงานหลังจากรับข้อมูลมาครบหนึ่งตัวอักขระ เช่น การพิมพ์ในเครื่องพิมพ์ การเจาะรูบนเทปของเครื่องเจาะเทป เป็นต้น ช่วงเวลาของบิตสิ้นสุดนี้ขึ้นอยู่กับการทำงานของเครื่องจักรกลซึ่งมีความเร็วช้าโดยอาจต้องมีเวลาของบิตสิ้นสุดเท่ากับ 2 บิต แต่ในอุปกรณ์ประเภทเครื่องจักรไฟฟ้านั้นจะมีความเร็วในการทำงานมากซึ่งอาจจะมีความกว้างของบิตดังกล่าวเพียง 1 บิตเท่านั้น

ในการส่งแบบอะซิงโครนัสนี้เนื่องจากอักขระทุกตัวที่ส่งออกมาต่างเป็นอิสระต่อกัน และทุก ๆ อักขระจะมีบิตบอกการเริ่มต้นและสิ้นสุดดังนั้นหากเกิดสัญญาณรบกวนจะก่อให้เกิดความผิดพลาดจนทำลายความถูกต้องของตัวอักขระ ความผิดพลาดนี้ถ้าเกิดขึ้นในการส่งแบบซิงโครนัสอาจจะทำลายข่าวสารหมดทั้งบล็อกก็ได้ เพราะข่าวสารทั้งบล็อก (อักขระหลายตัว) จะมีความสัมพันธ์เพียงครั้งเดียว ดังนั้นลักษณะการสื่อสารที่โครงงานตัวส่งสัญญาณอนาลอกขนาดจิ๋วเลือกใช้จะเป็นแบบอะซิงโครนัส

2.2 <u>พื้นฐานของตัวแปลงสัญญาณอนาลอกเป็นดิจิตอล</u>

ในโลกที่เราอาศัยกันอยู่ทุกวันนี้สัญญาณต่าง ๆ ที่เราคุ้นเคยกันมักเป็นสัญญาณที่มีการเปลี่ยนแปลงอย่างต่อเนื่อง หรือที่เรียกว่า สัญญาณอนาลอก นั้นเป็นสัญญาณที่เราเห็นได้ในชีวิตประจำวัน เช่น แสงสี , เสียง , อุณหภูมิ ตลอดจนแรงเคลื่อนไฟฟ้าหรือกระแสไฟฟ้า โดยมากแล้วพารามิเตอร์ของปริมาณทางฟิสิกส์และไฟฟ้านั้นจะเห็นและเข้าใจในรูปแบบของฟังก์ชันที่มีการเปลี่ยนแปลงอย่างต่อเนื่องที่ไม่มีสิ้นสุดจึงเป็นการยากที่จะควบคุมสัญญาณ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

อนาลอกให้ทำงานตามที่เราต้องการแต่ในทางตรงกันข้ามเราจะพบว่าสัญญาณอีกรูปแบบหนึ่งซึ่งไม่เป็นฟังก์ชันกับเวลาหรือที่เราเรียกว่าสัญญาณดิจิตอลนั้นสามารถควบคุมและทำความเข้าใจได้ง่ายกว่าสัญญาณอนาลอกซึ่งเราอาจจะใช้วงจรง่าย ๆ หรือไมโครโปรเซสเซอร์ควบคุมก็ได้ซึ่งถ้าเป็นสัญญาณอนาลอกนั้นเราไม่สามารถใช้ไมโครโปรเซสเซอร์ไปควบคุมได้โดยตรง

แต่เมื่อวงจรดิจิตอลต้องการรับสัญญาณอินพุทที่เป็นสัญญาณอนาลอกหรือต้องการส่งสัญญาณเอาท์พุทออกมาเป็นสัญญาณอนาลอกในทางปฏิบัติเราจะนำสัญญาณหรือวงจรทั้งสองมาเชื่อมต่อโดยตรงไม่ได้ จำเป็นต้องนำเอาสัญญาณอนาลอกนั้นมาผ่านอุปกรณ์เชื่อมต่อที่เรียกว่า ตัวแปลงสัญญาณอนาลอกเป็นสัญญาณดิจิตอล (A/D converter) โดยที่วิธีในการแปลงสัญญาณต่าง ๆ ได้ถูกพัฒนาขึ้นโดยวิธีที่ใช้อยู่ในทุก ๆ วันนี้มี 6 วิธีด้วยกัน คือ

1) เทคนิคแบบแฟลซ (Flash Techniques)
2) เทคนิคแบบสโลปเดียว (Single Slope Techniques)
3) เทคนิคแบบสโลปคู่ (Double Slope Techniques)
4) เทคนิคแบบเคาน์เตอร์เดียว (Single Counter Techniques)
5) เทคนิคแบบแทร็กกิงเคาน์เตอร์ (Tracking Counter Techniques)
6) เทคนิคแบบการประมาณค่าหลายครั้ง (Successive Approximation Techniques)

ในโครงงานตัวส่งสัญญาณอนาลอกขนาดจิ๋วจะใช้เทคนิคแบบสโลปคู่

2.3 ตัวแปลงสัญญาณอนาลอกเป็นสัญญาณดิจิตอลแบบสโลปคู่

เทคนิคการแปลงสัญญาณอนาลอกไปเป็นสัญญาณดิจิตอลแบบสโลปคู่เป็นเทคนิคที่ให้ข้อดีในด้านเสถียรภาพของการแปลงสัญญาณเมื่อสัญญาณอินพุทมีการเปลี่ยนแปลงอย่างรวดเร็ววงจรผลิตสัญญาณแรมป์อ้างอิงได้ปรับปรุงขึ้นโดยตัดเอาผลกระทบของการเลื่อนไหลเมื่อใช้วงจรไปนาน ๆ

สัญญาณอินพุทของตัวแปลงสัญญาณอนาลอกเป็นสัญญาณดิจิตอลแบบสโลปคู่จะป้อนให้กับวงจรอินทิเกรตเตอร์ (integrator) เมื่อสัญญาณอินพุทที่เป็นบวกถูกป้อนเข้ามายังตัวแปลงสัญญาณอนาลอกเป็นสัญญาณดิจิตอลความชันของแรงดันแรมป์ทางด้านเอาท์พุทของวงจรอินทิเกรตเตอร์จะมีทิศทางเป็นลบ (เพราะป้อนอินพุทเข้าขา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

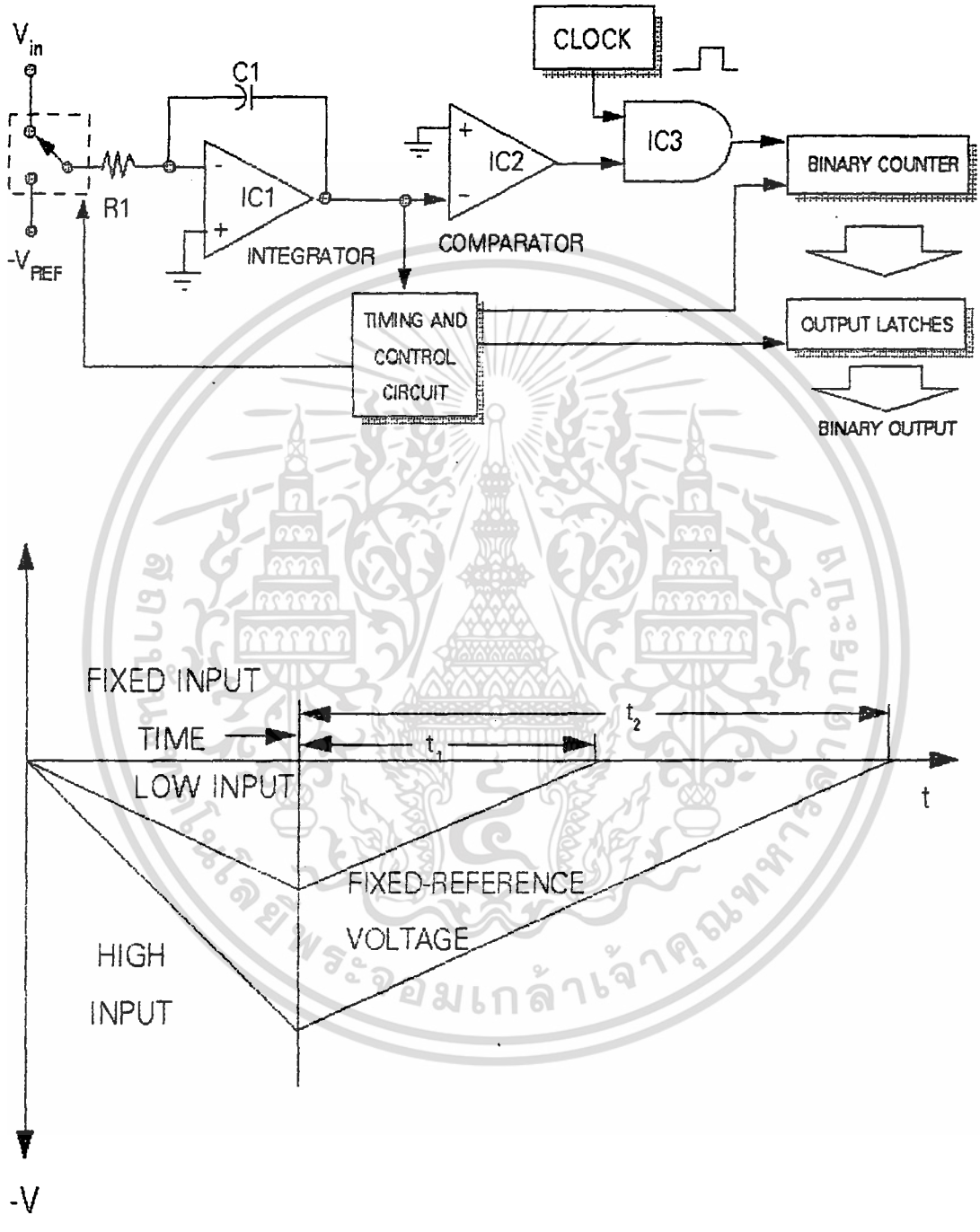
อินเวอร์ทเตอร์ของออปแอมป์) ด้วยแรงดันลบที่ได้นี้ทำให้เอาท์พุทของวงจรเปรียบเทียบเป็น 1 ด้วยเหตุนี้จึงเป็นการกระตุ้นให้เกิดสัญญาณนาฬิกาป้อนเข้าไปยังขั้วอินพุทของวงจรนับ ซึ่งจะเป็นการเริ่มต้นนับขึ้นไปเรื่อย ๆ วงจรอินทิเกรตเตอร์จะให้สัญญาณแรมป์เพียงคาบเวลาที่คงที่ขณะหนึ่งเท่านั้น หลังจากช่วงเวลานี้แล้ววงจรควบคุมจะทำการเคลียร์วงจรนับ และทำการเปลี่ยนอินพุทของวงจรอินทิเกรตเตอร์ไปต่อกับแรงดันอ้างอิงที่มีค่าเป็นลบ ($-V_{Ref}$) ดังนั้นในขณะนี้แรงดันลบถูกป้อนให้กับวงจรอินทิเกรตเตอร์ ความชันของสัญญาณแรมป์ทางเอาท์พุทกลับมามีทิศทางเป็นบวกวงจรนับจะเริ่มต้นนับใหม่จนกระทั่งเอาท์พุทของวงจรอินทิเกรตเตอร์ตกลงเป็นศูนย์ที่จุดนี้ เอาท์พุทของวงจรเปรียบเทียบกลายเป็น 0 ซึ่งทำให้สัญญาณนาฬิกาที่ป้อนให้วงจรนับหยุดลง วงจรควบคุมจะทำการตรวจสอบซึ่งเปลี่ยนและคงค่า (latch) การนับที่เอาท์พุทไว้แล้วทำการเคลียร์วงจรนับอีกครั้ง ตัวแปลงสัญญาณอนาลอกเป็นสัญญาณดิจิตอลแบบสโลปคู่นี้สัญญาณดิจิตอลที่นับได้ครั้งสุดท้ายนี้จะแทนแรงดันอินพุทอนาลอกที่เข้ามา ดังรูป 2.3

อัตราของการอินทิเกรตขึ้นอยู่กับขนาดของแรงดันอินพุทเช่นเดียวกับค่าของ R_1 และ C_1 ดังนั้นแรงดันอินพุทที่ต่ำ ๆ จะลดเอาท์พุทของวงจรอินทิเกรตเตอร์ให้น้อยกว่าแรงดันอินพุทที่มีค่าสูง ๆ ในช่วงคาบเวลาอินพุทที่แน่นอนของวัฏจักรการแปลงผัน

เมื่อแรงดันลบอ้างอิงที่มีค่าคงที่ถูกป้อนเข้ามา (ค่าของ R_1 และ C_1 ยังคงเดิม) เวลาที่ต้องการสำหรับเอาท์พุทของวงจรอินทิเกรตเตอร์ย่างเข้าสู่ศูนย์ เป็นอัตราแปรผันโดยตรงต่อขนาดดั้งเดิมของแรงดันอินพุททุก ๆ การเปลี่ยนแปลงเพราะฉนั้น ในวงจรอินทิเกรตเตอร์เวลาหรืออุณหภูมิที่มีผลต่อการทำงานของวงจรจะถูกตัดออกโดยอัตโนมัติดังนั้นตัวแปลงสัญญาณอนาลอกเป็นสัญญาณดิจิตอลแบบสโลปคู่จึงมีเสถียรภาพเหมาะสำหรับการประยุกต์ใช้งานที่มีความแม่นยำสูง

ตัวแปลงสัญญาณอนาลอกเป็นสัญญาณดิจิตอลแบบสโลปคู่มีสิ่งที่คล้ายคลึงกับตัวแปลงสัญญาณอนาลอกเป็นสัญญาณดิจิตอลแบบสโลปเดียว คือ สัญญาณอินพุทสามารถถูกเปลี่ยนให้อยู่ในรูปรหัสฐานสิบ (BCD) หรืออยู่ในรูปอื่น ๆ ได้โดยตรง เช่น รหัสไบนารี ในเครื่องวัดศักดาไฟฟ้าแบบดิจิตอล (digital voltmeter) โดยมากแล้วจะใช้เทคนิคการแปลงสัญญาณแบบสโลปคู่ในการแปลงสัญญาณอินพุทให้อยู่ในรูปรหัสฐานสิบ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูป 2.3 A/D คอนเวอร์ทเตอร์แบบสโลปคู่

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.4 ชนิดของการสื่อสารที่เลือกใช้

ลักษณะการสื่อสารข้อมูลของระบบตรวจวัดสัญญาณอนาลอกกับสถานีศูนย์กลางรวมจะเป็นทั้งการส่งในระยะใกล้โดยจะใช้การสื่อสาร RS232 , ระยะไกลโดยจะใช้การสื่อสารลูปกระแส (current loop) , และไร้สายโดยจะใช้การสื่อสารอินฟราเรด โดยที่การสื่อสารข้อมูลทั้ง 3 แบบนี้จะถูกเลือกใช้ในงานที่เหมาะสมและสภาพแวดล้อมที่เหมาะสม โดยที่ลักษณะของการสื่อสารทั้ง 3 แบบ จะถูกอธิบายดังต่อไปนี้

2.4.1 RS232

RS232 เป็นการสื่อสารในระยะใกล้ ผู้ที่กำหนดมาตรฐาน RS232 ขึ้นเมื่อกว่า 20 ปีมาแล้วก็คือ สถาบัน Electronics Industrials Association หรือ EIA โดยเป็นมาตรฐานสำหรับการเชื่อมต่อคอมพิวเตอร์ซึ่งมาตรฐานนี้เป็นการส่งข้อมูลแบบอะซิงโครนัสสัญญาณจะตรงกันข้ามกับความเป็นจริง คือ ลอจิกสูงมีระดับแรงดัน -3 ถึง -25 โวลต์ แต่ส่วนใหญ่จะใช้ -12 โวลต์และลอจิกต่ำมีระดับแรงดันตั้งแต่ +3 ถึง +25 โวลต์ ส่วนใหญ่ใช้ +12 โวลต์

การที่ได้กำหนด RS232 เป็นมาตรฐานการสื่อสารข้อมูลระหว่างคอมพิวเตอร์ทำให้มีการติดต่อสื่อสารข้อมูลกันอย่างแพร่หลาย ระบบ RS232 นี้มีความเร็วในการส่งข้อมูล 20,000 บิตต่อวินาที (20 kbps) และระยะส่งไม่เกิน 50 ฟุต ตามคุณสมบัติอย่างไรก็ตามสามารถยอมให้ส่งไกลกว่านี้ก็ได้โดยจะต้องคำนึงถึงค่าอิมพีแดนซ์ในสาย (X_C) โดยให้มีค่าน้อยที่สุดและค่าความจุไฟฟ้า 2,500 พิโคฟารัด

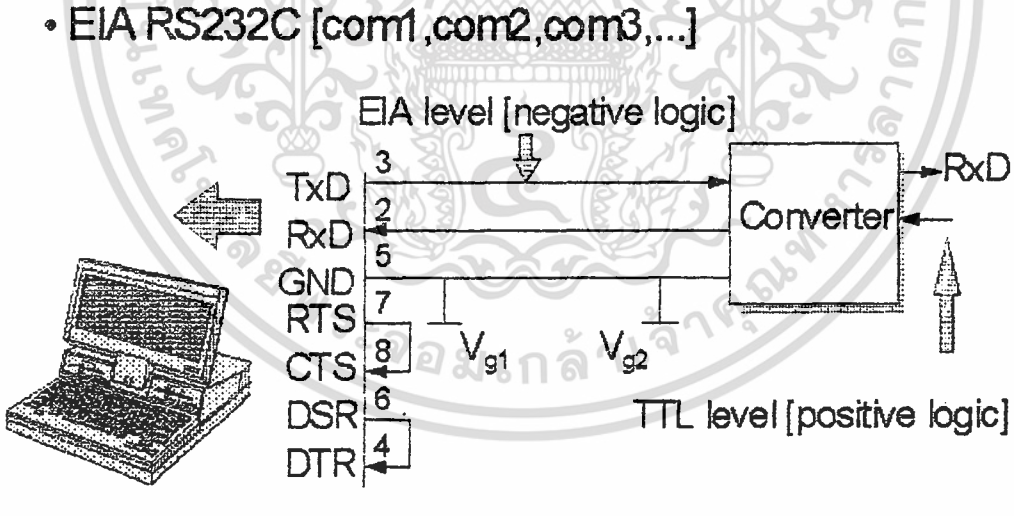
การส่งข้อมูลแบบ RS232 ค่อนข้างจะมีข้อจำกัดคือส่งได้แค่ 20,000 บิตต่อวินาทีที่ 50 ฟุต ถ้าระยะส่งไกลมากกว่านี้ความเร็วในการส่งจะลดลงและมีสัญญาณรบกวนมากขึ้นเนื่องจากสายส่งสัญญาณเป็นแบบอ้างอิงกับกราวด์

ลักษณะการเชื่อมต่อสัญญาณและการจัดตำแหน่งสัญญาณ แสดงดังรูป 2.4 และตาราง 2.1 ตามลำดับ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ขาที่	สัญญาณ
1	Chassis Ground
2	TX (Transmit Data)
3	RX (Receive Data)
4	RTS (Request To Send)
5	CTS (Clear To Send)
6	DSR (Data Set Ready)
7	GND (Signal Ground)
8	DCD (Data Carrier Detect)
9	NC

ตาราง 2.1 การจัดตำแหน่งสัญญาณ RS232

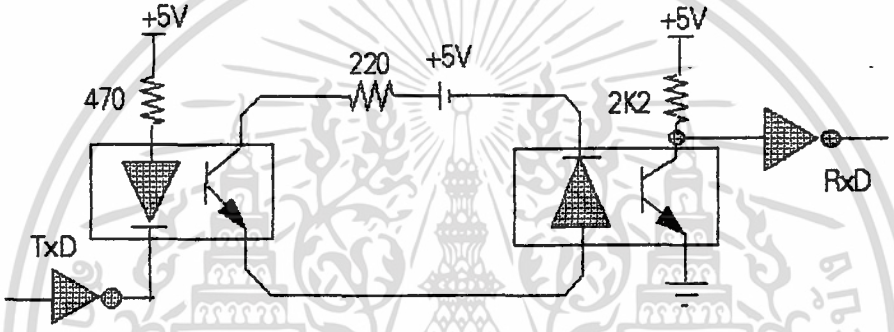


รูป 2.4 ลักษณะการเชื่อมต่อสัญญาณแบบ RS232

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.4.2 ลูปกระแส (Current loop)

การสื่อสารแบบลูปกระแสเป็นการสื่อสารระยะไกล โดยจะเป็นการสื่อสารด้วยการส่งผ่านของกระแส 20 มิลลิแอมป์ แทนที่จะส่งเป็นศักดาไฟฟ้าทำให้สามารถส่งได้ในระยะที่ไกลมากขึ้น โดยที่จะมีการแยกกราวด์ออกจากกันด้วยออปโตไอโซเลเตอร์ (optoisolator) ซึ่งจะไม่เหมือนกับ RS232 วิธีการสื่อสารแบบลูปกระแสนี้สามารถส่งได้ระยะไกลสูงสุดได้หลายกิโลเมตร ด้วยความเร็วต่ำและสามารถส่งด้วยความเร็วสูงสุดถึง 19,200 บิตต่อวินาทีที่ระยะทางสั้น ๆ ลักษณะวงจรเป็นดังรูป 2.5

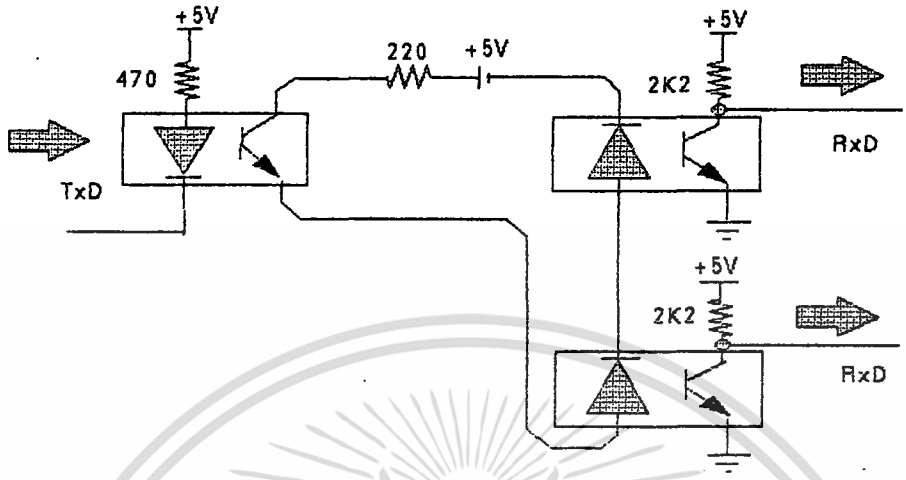


รูป 2.5 ลักษณะการเชื่อมต่อสัญญาณการสื่อสารแบบลูปกระแส

ลักษณะการสื่อสารข้อมูลแบบลูปกระแสจะสามารถทำการต่อพ่วงได้หลาย ๆ จุดตลอดระยะทางความยาวทั้งหมด โดยจะแสดงลักษณะการต่อพ่วงของการส่งข้อมูล และรับข้อมูลดังรูป 2.6 และ 2.7 ตามลำดับ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

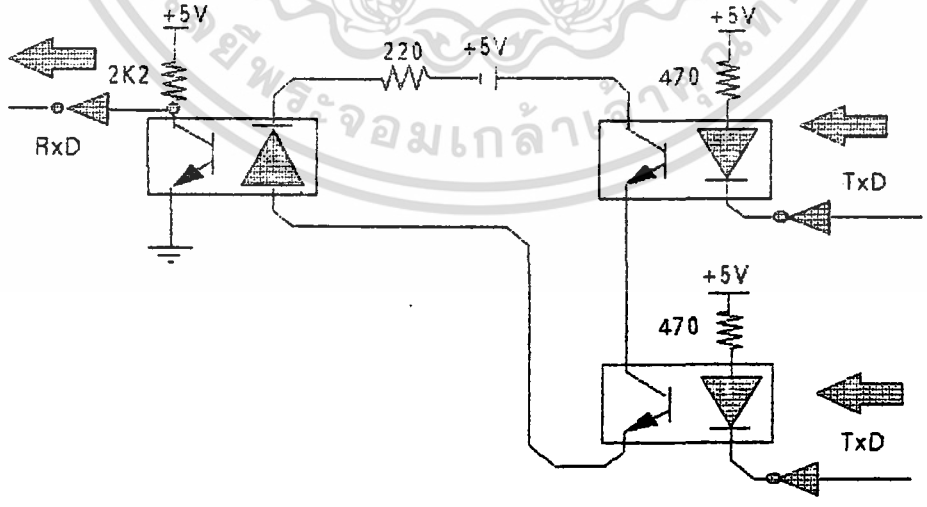
สำหรับการส่งข้อมูล



รูป 2.6 แสดงลักษณะวงจรสำหรับส่งข้อมูล

เมื่อต้องการจะส่งข้อมูลจะมีกระแสไหลในวงจรด้วยปริมาณ 20 มิลลิแอมป์ ทำให้ด้านรับทุกตัวได้รับข้อมูลด้วยลอจิก 0 แต่ถ้าไม่มีกระแสไหลในวงจร ทางด้านรับทุกตัวก็จะได้รับข้อมูลลอจิก 1

สำหรับการรับข้อมูล



รูปที่ 2.7 แสดงลักษณะวงจรสำหรับรับข้อมูล

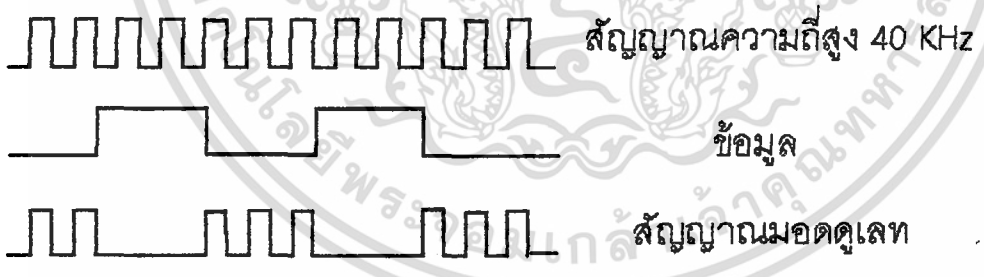
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เมื่อทางด้านตัวส่งทุกตัวไม่ทำการส่งข้อมูลก็จะทำให้เกิดกระแส 20 มิลลิแอมป์ ไหลอยู่ภายในวงจรจะทำให้ตัวรับได้รับข้อมูลเป็นลอจิก 1 แต่เมื่อตัวใดตัวหนึ่งส่งข้อมูลก็จะทำการตัดทางเดินของกระแส ทำให้ตัวรับได้รับข้อมูลเป็นลอจิก 0 โดยที่ตัวส่งทุกตัวจะต้องทำการส่งข้อมูลในเวลาที่ไม่ตรงกัน

ในแต่ละคู่ของเส้นสัญญาณ สายทั้งสองจะทำการตีเกลียวไปด้วยกันตลอดระยะทางตามความยาวของสาย ซึ่งถ้าสายมีอิมพีแดนซ์เท่ากัน และกระแสที่ไหลในสายมีขนาดเท่ากันแต่ทิศทางตรงกันข้าม ก็จะทำให้สัญญาณรบกวนต่าง ๆ หักล้างกันหมดไป

2.4.3 <u>อินฟราเรด (Infrared)</u>

การสื่อสารแบบอินฟราเรดเป็นการสื่อสารไร้สาย โดยที่จะทำการสื่อสารได้ในระยะทาง 5 ถึง 10 เมตรด้วยความเร็ว 1,200 บิตต่อวินาที โดยจะใช้การสื่อสารด้วยคลื่นแสงย่านอินฟราเรด ทำให้อุณหภูมิของสภาพแวดล้อมเป็นข้อจำกัดต่อการสื่อสารข้อมูล ลักษณะการส่งข้อมูลด้วยคลื่นแสงย่านอินฟราเรดนี้จะทำการมอดดูเลทกับสัญญาณความถี่สูงซึ่งจะมีค่าประมาณ 40 KHz ที่ต้องทำการมอดดูเลทสัญญาณความถี่สูงเนื่องจากอินฟราเรดไม่เหมาะสมที่จะใช้งานในลักษณะของสัญญาณไฟตรง ลักษณะการมอดดูเลทสัญญาณเป็นดังรูป 2.8



รูป 2.8 แสดงลักษณะการมอดดูเลทสัญญาณ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.5 การสร้างฐานเวลาให้แก่ระบบ

ในระบบที่ใช้ไมโครคอนโทรลเลอร์เป็นตัวควบคุม ส่วนใหญ่ต้องใช้เวลาเป็นตัวกำหนดการทำงาน ดังนั้นหากในระบบควบคุมที่ใช้ไมโครคอนโทรลเลอร์มีวงจรกำเนิดฐานเวลารวมอยู่ด้วยจะทำให้การออกแบบระบบและเขียนโปรแกรมได้สะดวกมากขึ้น

ในไมโครคอนโทรลเลอร์ตระกูล 51 (MCS 51) มีรีจิสเตอร์ทำหน้าที่เป็นได้ทั้งไทม์เมอร์และเคาน์เตอร์ (timer and counter) ซึ่งผู้ใช้สามารถเขียนโปรแกรมให้ทำงานเป็นตัวสร้างฐานเวลาแก่ซีพียูได้ แต่เนื่องจากความไม่สะดวกในการใช้งานจริงที่ต้องการทราบเวลาในปัจจุบัน หรือในงานที่ต้องการสัญญาณอินเตอร์รัปต์ในช่วงเวลาที่มากหรือน้อยกว่าที่ไทม์เมอร์สามารถสร้างได้ ทั้งนี้เนื่องจากต้องคำนวณคาบเวลาให้ถูกต้องและต้องมีการตั้งค่าใหม่ทุกครั้งที่เริ่มทำงานหรือหากการทำงานเกิดการผิดพลาดเวลาก็จะผิดตามไปด้วย ทั้งนี้เพราะใช้รีจิสเตอร์ภายในไมโครคอนโทรลเลอร์ตระกูล 51 เป็นตัวกำหนดฐานเวลา ดังนั้นบางครั้งในการออกแบบระบบควบคุมจำเป็นจะต้องมีวงจรซึ่งทำหน้าที่สร้างฐานเวลาโดยจำเป็นที่จะต้องทำงานอยู่ตลอดเวลาแม้ขณะไม่มีพลังงานด้วยการใช้แบตเตอรี่สำรอง วิธีนี้จะใช้ชิปที่ทำหน้าที่สร้างฐานเวลาจริงให้แก่ระบบคือชิปกำเนิดเวลาจริง (Real Time Clock , RTC) ซึ่งในปัจจุบันนี้ได้พัฒนาขึ้นจนมีขนาดเล็กลงเป็นไอซีที่ใช้พลังงานน้อยมากทำให้การใช้แบตเตอรี่สำรองสะดวกมากขึ้น ระบบควบคุมที่ใช้ชิปกำเนิดเวลาจริงเป็นตัวกำหนดเวลารวมนั้น นอกจากจะทำให้มีความเที่ยงตรงในการทำงานมากขึ้นแล้วประโยชน์อีกอย่างหนึ่งนั่นคือทำให้มีไทม์เมอร์หรือเคาน์เตอร์เหลือไว้ใช้งานอย่างอื่นด้วย

2.6 วิธีการสร้างฐานเวลาจริงให้แก่ไมโครคอนโทรลเลอร์ตระกูล 51

วิธีการสร้างฐานเวลาจริงนี้จะใช้ชิปที่ทำหน้าที่เป็นนาฬิกาที่สามารถส่งข้อมูลเวลาในขณะใด ๆ ให้แก่ไมโครคอนโทรลเลอร์ได้ เวลาที่ได้จากวิธีนี้จะเป็นเวลาจริง ๆ ที่เดินอยู่ตลอดเวลาอย่างเที่ยงตรงชิปที่ทำหน้าที่ดังกล่าวในปัจจุบันมีอยู่ด้วยกันหลายเบอร์ แต่เบอร์ที่ใช้งานได้สะดวกมากที่สุดคือชิปกำเนิดเวลาจริงของบริษัท Dallas Semiconductor เบอร์ DS1202 ซึ่งจะได้กล่าวต่อไปในบทนี้

ชิปกำเนิดเวลาจริงเบอร์ DS1202 ที่จะกล่าวถึงนี้มีความเที่ยงตรงในการทำงานสูงมาก สามารถนำมาต่อร่วมกับระบบเพื่อบอกเวลาให้แก่ไมโครคอนโทรลเลอร์ได้สะดวก เพราะใช้จำนวนสายในการติดต่อระหว่างตัวชิปเองกับไมโครคอนโทรลเลอร์เพียง 3 เส้นเท่านั้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เนื่องจากชิปกำเนิดเวลาจริงเบอร์นี้ใช้การติดต่อรับส่งข้อมูลแบบอนุกรมคุณสมบัติของชิปกำเนิดเวลาจริงเบอร์นี้มีดังนี้

- ทำหน้าที่นับวินาที นาที ชั่วโมง วันที่ของเดือน เดือน ปี รวมทั้งคำนวณปีอธิกสุรทินให้เองโดยอัตโนมัติ

- มีหน่วยความจำขนาด 20 ไบต์ สำหรับเก็บข้อมูลทั่ว ๆ ไป ส่วนใหญ่ใช้เก็บข้อมูลที่ต้องการสำรองในขณะที่ไม่มีพลังงานจ่ายให้แก่ระบบ เช่น รหัสผ่านที่เปลี่ยนค่าได้ เวลาที่ต้องการให้เครื่องจักรทำงานทำให้ไม่จำเป็นเลยที่จะต้องสำรองหน่วยความจำทั้งระบบนั่นเอง

- ใช้การติดต่อแบบอนุกรมจึงใช้จำนวนสายในการเชื่อมต่อกับระบบเพียง 3 เส้นเท่านั้น

- ใช้แรงดันไฟฟ้าเพียง 2.0 ถึง 5.5 โวลต์ และใช้กระแสเพียง 300 นาโนแอมแปร์ ที่ระดับแรงดันไฟฟ้า 2.0 โวลต์

- การโอนย้ายข้อมูลสามารถกระทำได้ทั้งในแบบครั้งละ 1 ไบต์ หรือครั้งละหลาย ๆ ไบต์ ไม่ว่าจะเป็นการเขียนหรือการอ่านข้อมูล

- ระดับสัญญาณเป็น TTL (V_{CC} = 5 V)

- ช่วงอุณหภูมิในการใช้งานกว้างมากระหว่าง -40 องศาเซลเซียส ± 88 องศาเซลเซียส

<u>รายละเอียด</u>

ชิปกำเนิดเวลาจริงเบอร์ DS1202 มีค่าเวลาจริง / ปฏิทิน (Real Time Clock / Calender) และหน่วยความจำขนาด 24 ไบต์ ใช้สายเพียง 3 เส้นในการเชื่อมต่อกับไมโครคอนโทรลเลอร์เพื่อรับส่งข้อมูลเกี่ยวกับเวลา ข้อมูลที่ชิปกำเนิดเวลาจริงเบอร์ DS1202 มีให้ประกอบด้วย

- วินาที
- นาที
- ชั่วโมง
- วันที่
- วัน
- เดือน
- ปี

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

วันที่ในวันสุดท้ายของเดือนจะถูกปรับโดยอัตโนมัติสำหรับเดือนที่มีจำนวนวันน้อยกว่า 31 วัน และมีการคำนวณจำนวนวันของเดือนกุมภาพันธ์ในปีอธิกสุรทินให้เอง ข้อมูลที่ส่งให้แก่ไมโครคอนโทรลเลอร์สามารถเลือกรูปแบบได้ทั้งแบบ 24 ชั่วโมง (0.00 - 23.59 นาฬิกา) หรือแบบ 12 ชั่วโมง (0.00 - 12.00 นาฬิกา) โดยมีข้อมูลเพิ่มเพื่อบอกให้ทราบว่าเป็นเวลาในช่วงกลางวันหรือกลางคืน

การเชื่อมต่อชิปกำเนิดเวลาจริงเบอร์ DS1202 เข้ากับระบบมีความสะดวกมาก เนื่องจากใช้จำนวนสายเพียง 3 เส้นเท่านั้น เพราะใช้การติดต่อแบบอนุกรมซนิดซิงโครนัส ขาที่ต้องใช้ในการรับส่งข้อมูลทั้งสาม คือ

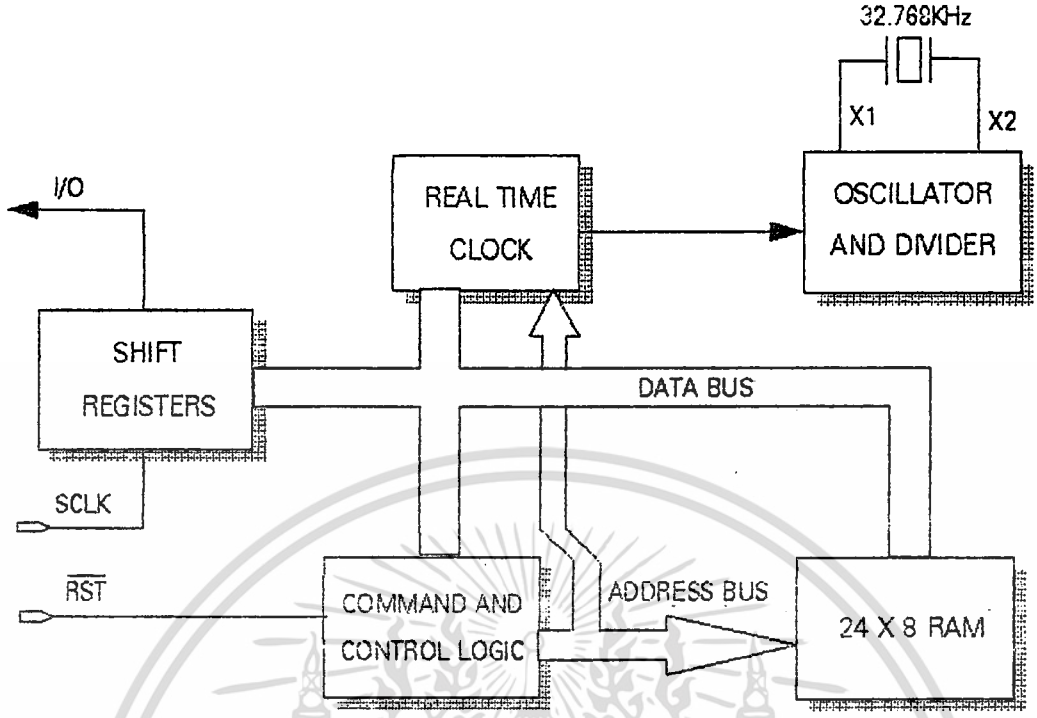
- RST (reset)
- I/O (data line)
- SCLK (serial clock)

เนื่องจากในชิปกำเนิดเวลาจริงเบอร์ DS1202 มีนาฬิกาหรือเวลาที่เดินอยู่ตลอดเวลารวมทั้งมีหน่วยความจำจำนวนหนึ่งดังนั้นในการติดต่อกับชิปกำเนิดเวลาจริงเบอร์ DS1202 ผู้ใช้สามารถเลือกได้ว่าต้องการข้อมูลจากนาฬิกาหรือจากหน่วยความจำภายในชิป การรับส่งข้อมูลสามารถกระทำได้ทั้งแบบทีละไบต์หรือการรับส่งกันคราวละหลายไบต์ ดังจะได้กล่าวต่อไป นอกจากนี้ชิปกำเนิดเวลาจริงเบอร์ DS1202 ยังถูกออกแบบให้ใช้พลังงานน้อยมาก และสิ้นเปลืองพลังงานจากแบตเตอรี่น้อยที่สุดเพื่อความสะดวกในการสำรองพลังงาน โดยชิปตัวนี้สามารถเก็บรักษาข้อมูลในหน่วยความจำและเวลาที่เดินอยู่ตลอดเวลาได้ที่กำลังไฟฟ้าน้อยกว่า 1 ไมโครวัตต์

โครงสร้างของชิปกำเนิดเวลาจริงเบอร์ DS1202 โครงสร้างของชิปเบอร์นี้มีดังแสดงในรูป 2.9

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สำนักหอสมุดกลาง พระจอมเกล้าลาดกระบัง



รูป 2.9 แสดงลักษณะโครงสร้างภายในของชิปกำเนิดเวลาจริง DS1202

โครงสร้างภายในของชิปกำเนิดเวลาจริง DS1202 ประกอบไปด้วยส่วนที่สำคัญ ๆ ดังแสดงในรูป 2.9 จะเห็นว่าชิปกำเนิดเวลาจริง เบอร์นี้ประกอบไปด้วยส่วนที่สำคัญ ๆ ดังนี้ คือ

- รีจิสเตอร์เลื่อนข้อมูล (shift register)
- ส่วนควบคุมลอจิก (control logic)
- ออสซิลเลเตอร์ (oscillator)
- เวลาจริง (real time clock)
- หน่วยความจำภายใน

ในการรับหรือส่งข้อมูลใด ๆ ให้แก่ชิปกำเนิดเวลาจริงเบอร์ DS1202 ไมโครคอนโทรลเลอร์หรือไมโครโปรเซสเซอร์ที่ต้องการติดต่อด้วยจะต้องส่งข้อมูลที่เป็นคำสั่งควบคุมการติดต่อซึ่งมีขนาด 8 บิตเสียก่อนโดยเริ่มต้นด้วยการให้ขา RST มีสถานะเป็น 1 (อยู่ในช่วงการติดต่อ) จากนั้นส่งข้อมูลจำนวน 8 บิตเข้าไปไว้ในรีจิสเตอร์เลื่อนข้อมูลของชิปกำเนิดเวลาจริง ข้อมูลขนาด 8 บิตจะประกอบด้วยคำสั่งในการควบคุมชิปกำเนิดเวลาจริงและตำแหน่งของหน่วยความจำที่ต้องการติดต่อ (address / command byte) ในแต่ละครั้ง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การรับข้อมูลจากไมโครโปรเซสเซอร์แต่ละบิตจะกระทำที่ช่วงขอบขาขึ้นของสัญญาณที่ขา SCLK

ภายในชิปกำเนิดเวลาจริงเบอร์ DS1202 ประกอบไปด้วยหน่วยความจำขนาด 24 ไบต์ และรีจิสเตอร์ที่ทำหน้าที่เก็บเวลาของชิปในขณะปัจจุบันจำนวน 8 ตัว รีจิสเตอร์ทั้งแปดตัวนี้สามารถเข้าถึงได้เสมือนเป็นหน่วยความจำตำแหน่งหนึ่ง ดังนั้นต่อไปเราจะมองว่าชิปกำเนิดเวลาจริงเบอร์นี้มีหน่วยความจำรวมทั้งสิ้น 32 ตำแหน่ง โดยประกอบขึ้นจากรีจิสเตอร์ 8 ตำแหน่ง และหน่วยความจำ 24 ตำแหน่ง

2.7 ไมโครคอนโทรลเลอร์ AT89C2051

ลักษณะเด่น

- สามารถใช้แทนไมโครคอนโทรลเลอร์ตระกูล 51
- หน่วยความจำแบบแฟลช (Flash memory) ขนาด 2 kB โปรแกรมใหม่ได้ 1000 ครั้ง รักษาข้อมูลได้ 10 ปี
- ใช้ไฟเลี้ยงตั้งแต่ 2.7 - 6 โวลต์ , ออสซิลเลเตอร์ป้อนได้ตั้งแต่ 0 Hz - 24 MHz
- สามารถล็อคโปรแกรมได้ 2 ระดับ
- หน่วยความจำข้อมูลบนชิป 128 ไบต์
- มีอินพุท / เอาท์พุทพอร์ต 15 บิต
- มีตัวนับและตั้งเวลา 16 บิต 2 ตัว
- แหล่งอินเตอร์รัปต์มี 5 แหล่ง
- พอร์ตอนุกรมแบบ UART โปรแกรมความเร็วในการส่งข้อมูลได้
- เอาท์พุทของพอร์ตสามารถขับ LED ได้โดยตรง ด้วยกระแสซิงค์ 20 มิลลิแอมป์
- มีอนาลอกคอมพาราเตอร์บนชิป
- มีโหมด Idle และ Power down เพื่อการประหยัดพลังงาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

<u>รายละเอียด</u>

AT89C2051 เป็นไมโครคอมพิวเตอร์ 8 บิต CMOS มีหน่วยความจำโปรแกรม 2kB ที่เป็นแบบแฟลชหรือเรียกว่า Programable and Erasable Read Only Memory (PEROM) ชิป 89C2051 ได้ถูกสร้างขึ้นด้วยเทคโนโลยี Atmel's high density nonvolatile memory technology หน่วยความจำโปรแกรมภายใน 2 kB , หน่วยความจำข้อมูลภายใน 128 ไบต์ , อินพุท / เอาท์พุท 15 บิต , ตัวนับและตั้งเวลา 16 บิต 2 ตัว , แหล่งอินเตอร์รัปต์ 5 แหล่ง 2 ระดับ , การสื่อสารอนุกรมแบบฟูลดูเพลก (full duplex) , อนาลอกคอมพาราเตอร์ (analog comparator) , และวงจรนาฬิกาบนชิป

ชิป 89C2051 สามารถเขียนโปรแกรมควบคุมการใช้พลังงานได้ 2 แบบ คือ โหมด Idle จะหยุดการทำงานของซีพียู โดยที่ตัวนับและตั้งเวลา , พอร์ตสื่อสารอนุกรม และอินเตอร์รัปต์จะทำให้ซีพียูทำงานต่อไป โดยที่ข้อมูลในหน่วยความจำภายในยังคงอยู่ และโหมด Power down จะคงค่าในหน่วยความจำภายในแต่จะหยุดป้อนสัญญาณนาฬิกาให้กับทุก ๆ ฟังก์ชันบนชิปจนกระทั่งทำการรีเซ็ตที่ฮาร์ดแวร์

พอร์ต 1 เป็นอินพุท / เอาท์พุทพอร์ต 8 บิต สองทิศทาง P1.2 ถึง P1.7 จะมีพูลอัพ (pull up) ภายใน ส่วน P1.0 และ P1.1 ต้องการพูลอัพภายนอกอีกทั้งจะเป็นอินพุทบวกและอินพุทลบของคอมพาราเตอร์ตามลำดับ อีกทั้งพอร์ต 1 สามารถรับกระแสซิงค์ 20 มิลลิแอมป์ ซึ่งจะสามารถขับ LED ได้โดยตรง

พอร์ต 3 เป็นอินพุท / เอาท์พุทพอร์ต 7 บิต สองทิศทาง มีพูลอัพภายใน โดยที่ P3.6 จะเป็นเอาท์พุทของคอมพาราเตอร์ อีกทั้งพอร์ต 3 สามารถขับ LED ได้โดยตรงด้วยและเป็นฟังก์ชันพิเศษดังตาราง 2.2

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ขาพอร์ต	ฟังก์ชัน
P3.0	RxD (Serial Input Port)
P3.1	TxD (Serial Output Port)
P3.2	INT0 (External Interrupt 0)
P3.3	INT1 (External Interrupt 1)
P3.4	T0 (Timer 0 External Input)
P3.5	T1 (Timer 1 External Input)

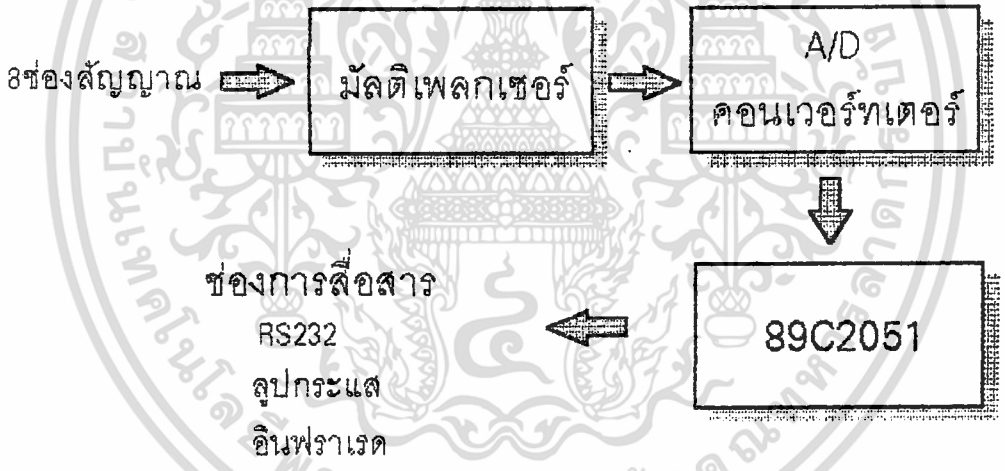
ตาราง 2.2 ฟังก์ชันพิเศษ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 3
การดำเนินการวิจัย

3.1 ตัวส่งสัญญาณอนาลอกขนาดจิ๋ว

ตัวตรวจวัดสัญญาณอนาลอกจะเป็นการนำสัญญาณอนาลอกมาแปลงเป็นสัญญาณดิจิตอลด้วยตัวแปลงสัญญาณอนาลอกเป็นสัญญาณดิจิตอลโดยในโครงงานนี้จะใช้ไอซีเบอร์ CA3162 อีกทั้งเราจะทำการเพิ่มช่องในการตรวจวัดสัญญาณอนาลอกโดยการใช้มัลติเพลกเซอร์เบอร์ 4051 8 ช่องสัญญาณ และจะนำค่าสัญญาณดิจิตอลที่ได้มาทำการประมวลผลเบื้องต้นด้วยไมโครคอนโทรลเลอร์เบอร์ 89C2051 และทำหน้าที่รอรับคำสั่งจากสถานีศูนย์กลาง และทำการสื่อสารข้อมูลตามลักษณะการสื่อสารต่าง ๆ ที่เหมาะสมดังรูป 3.1



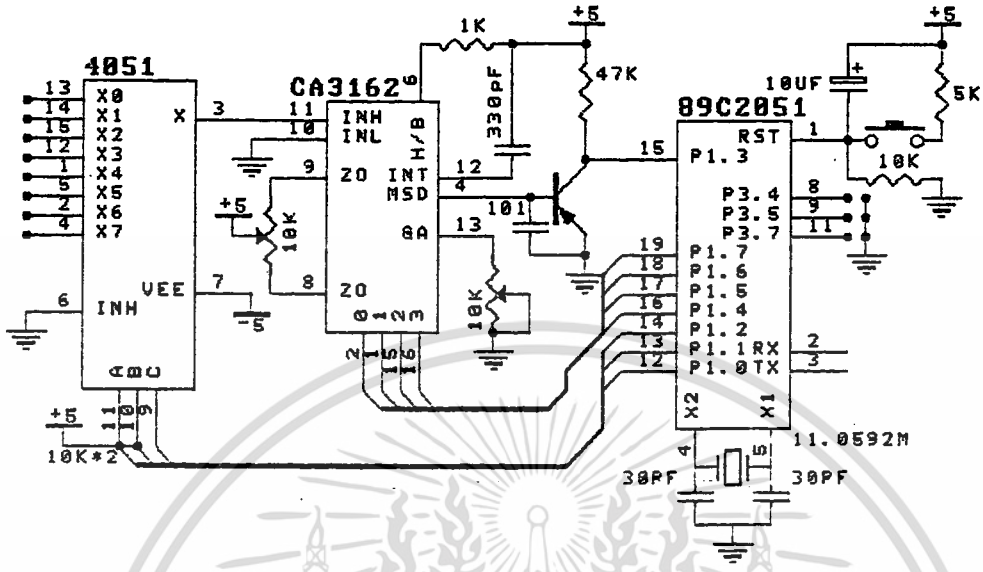
รูป 3.1 ลักษณะตัวส่งสัญญาณอนาลอกขนาดจิ๋ว

3.1.1 การเชื่อมต่อทางการวัดสัญญาณอนาลอก

CA3162 เป็นตัวแปลงสัญญาณอนาลอกเป็นสัญญาณดิจิตอลชนิดสโลปคู่ที่ไม่ต้องการสัญญาณนาฬิกาจากภายนอกและสามารถแปลงค่าได้ในช่วง -99 mV ถึง +999 mV ค่าที่เกินกว่านี้จะให้ผลเป็น — และ EEE ตามลำดับ โดยที่จะให้ผลคำตอบในรูปของรหัสฐานสิบ เมื่อทำการแปลงข้อมูลเสร็จแล้วจะส่งสัญญาณกระตุ้นที่ขา MSD (ขา 4) , NSD (ขา 3) , และ

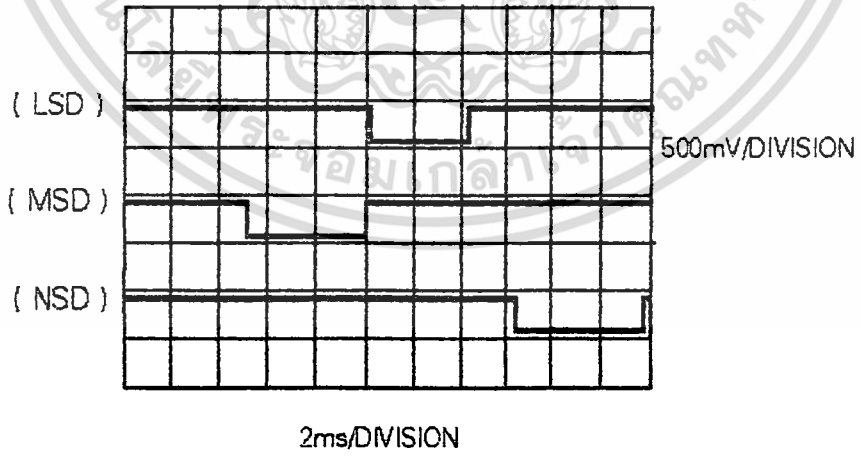
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

LSD (ขา 5) โดยเราจะทำการอ่านข้อมูลในแต่ละหลักออกมาได้ ลักษณะของวงจรจะเป็นดังรูป 3.2



รูป 3.2 วงจรการเชื่อมต่อทางการวัดสัญญาณอนาลอก

เราสามารถจะทำการเปลี่ยนช่องสัญญาณอนาลอกได้จากการเปลี่ยนแอดเดรสที่ P1.0 , P1.1 และ P1.2 และจะรอสัญญาณกระตุ้นเพื่ออ่านค่าสัญญาณดิจิตอลโดยการหน่วงเวลาตามไดอะแกรมของเวลาดังรูป 3.3



รูป 3.3 ไทม์มิงไดอะแกรมของ CA3162

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เมื่อเริ่มต้นการทำงานจะต้องมีการเลือกความเร็วในการส่งข้อมูลอนุกรม ลักษณะการส่งข้อมูล และชนิดของการสื่อสารข้อมูล ซึ่งจะทำการเลือกเฉพาะตอนเริ่มต้นการทำงานเท่านั้น โดยที่จะทำการแสดงลักษณะการเลือกจัมเปอร์ (jumper) ต่าง ๆ ดังตาราง 3.1

จัมเปอร์	ฟังก์ชัน
	ความเร็ว 1200 บิตต่อวินาที
	ความเร็ว 4800 บิตต่อวินาที
	ความเร็ว 9600 บิตต่อวินาที
	ความเร็ว 19200 บิตต่อวินาที

จัมเปอร์เลือกความเร็วในการส่งข้อมูลอนุกรม (P3.4 , P3.5)

จัมเปอร์	ฟังก์ชัน
	รอรับคำสั่ง
	ส่งข้อมูลเมื่อทำการแปลงเสร็จ

จัมเปอร์เลือกลักษณะการส่งข้อมูล (P3.7)

จัมเปอร์	ฟังก์ชัน
	RS232
	อินฟราเรด
	ลูปกระแส

จัมเปอร์เลือกชนิดของการส่งข้อมูล

ตาราง 3.1 จัมเปอร์ของตัวส่งสัญญาณอนาลอกขนาดจิ๋ว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.1.2 <u>การทำงานของโปรแกรม</u>

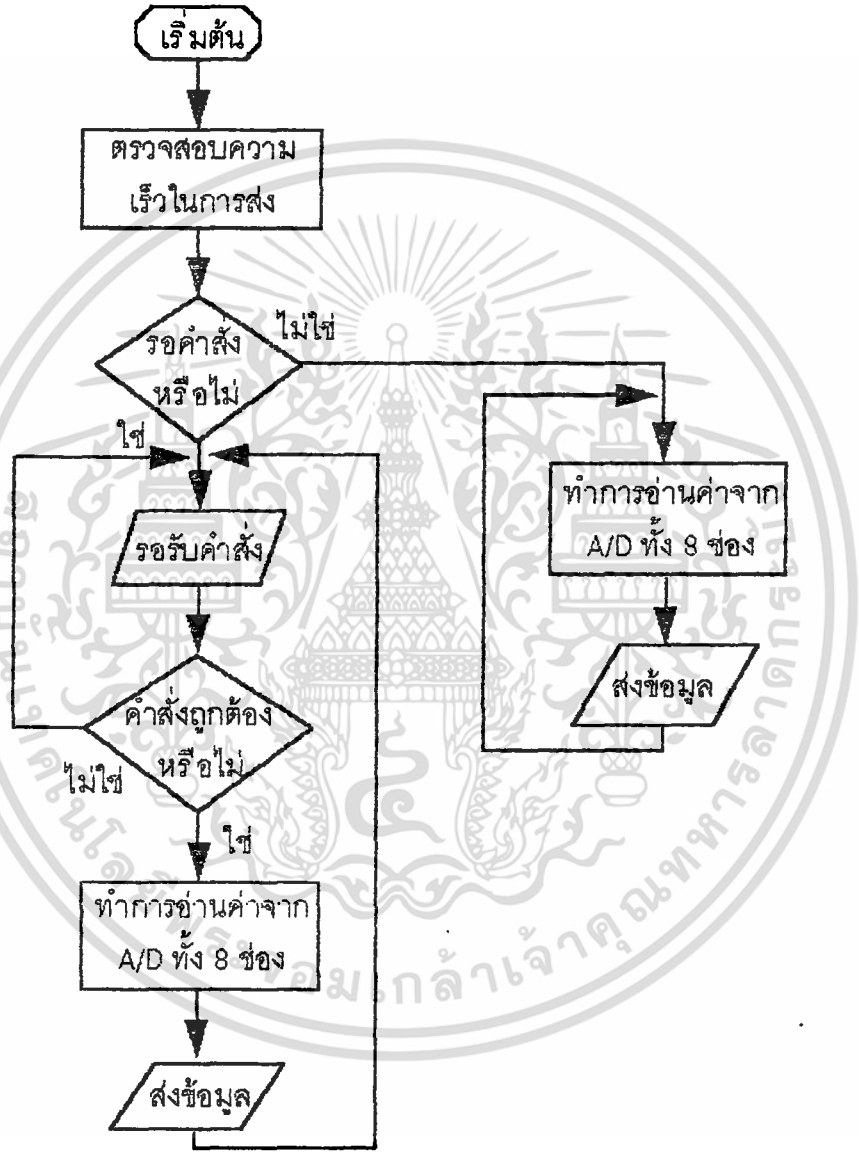
ลักษณะการทำงานเป็นดังรูป 3.4 เมื่อมีการรอรับคำสั่งจะทำการรอรับคำสั่งโดยจะมีรูปแบบเป็น " T□□/ΔΔ " โดยที่ □□ เป็นแอดเดรสของตัวส่งสัญญาณอนาลอกขนาดจิ๋ว และ ΔΔ เป็นแอดเดรสของสถานีศูนย์กลางที่ต้องการข้อมูล เมื่อคำสั่งทั้งหมดถูกต้องก็จะทำการอ่านค่าจากช่องสัญญาณทั้ง 8 ช่องของมัลติเพลกเซอร์ และทำการส่งข้อมูลโดยข้อมูลจะมีลักษณะเป็น " SOHΔΔ/□□ , ⊕ , ⊕ , – , ⊕ ,©EOT " โดยที่ SOH = 01H , ΔΔ เป็นแอดเดรสของสถานีศูนย์กลาง , □□ เป็นแอดเดรสของตัวส่งสัญญาณอนาลอกขนาดจิ๋วที่ส่งข้อมูล , ⊕ จะเป็นข้อมูล 3 ไบต์ค่าของแต่ละช่องสัญญาณ , © เป็นข้อมูลตรวจสอบความถูกต้อง (checksum) และ EOT = 04H โดยการอ่านค่าจาก A/D แปลงสัญญาณอนาลอกเป็นสัญญาณดิจิตอลทั้ง 8 ช่องสัญญาณจะมีลักษณะการทำงานดังรูป 3.5

ลักษณะการส่งสัญญาณอนาลอกขนาดจิ๋วนี้จะเพิ่มความคล่องตัวได้โดยการใช้พลังงานจากแบตเตอรี่ ดังนั้นจึงต้องมีการประหยัดพลังงานของไมโครคอนโทรลเลอร์โดยการเข้าสู่โหมดประหยัดพลังงานในขณะที่ทำการรอคำสั่งจากสถานีศูนย์กลาง

3.1.3 <u>วงจรแหล่งจ่ายไฟ</u>

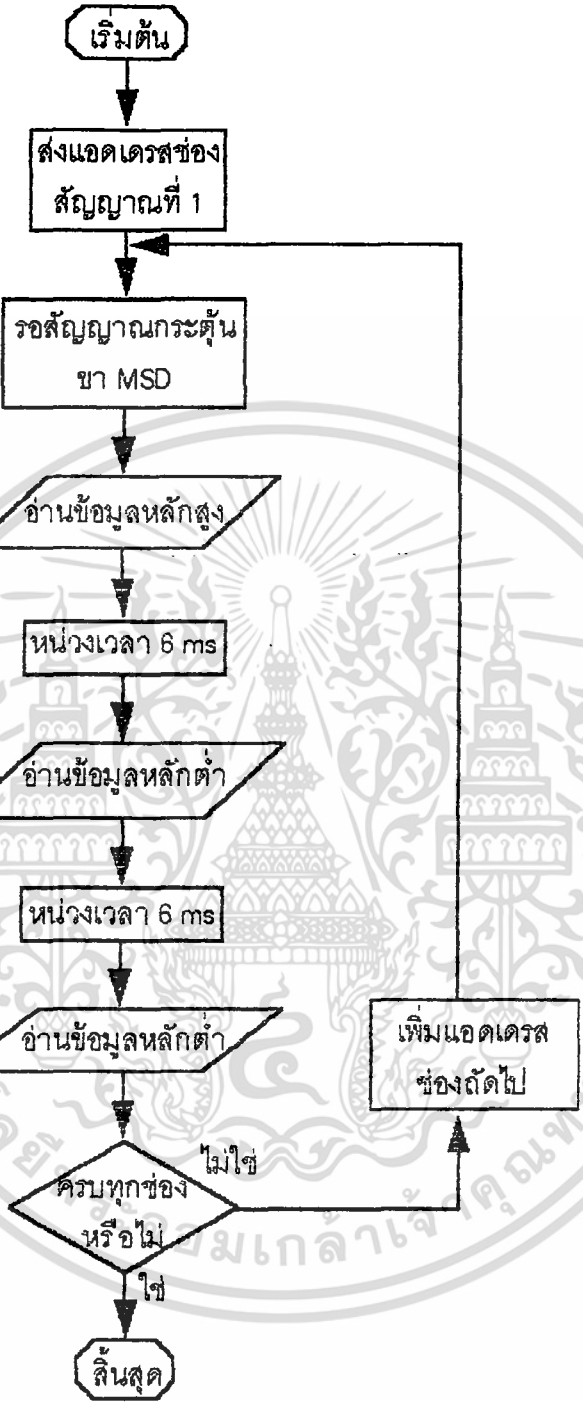
แหล่งจ่ายไฟตัวส่งสัญญาณอนาลอกขนาดจิ๋ว จะเป็นทั้งไฟบวก 5 โวลต์ และไฟลบ 5 โวลต์ โดยที่ไฟบวกจะได้รับจากตัวแปลงสัญญาณไฟ 220 โวลต์ ไฟสลับไปเป็น 9 โวลต์ ไฟตรง และจะทำการแปลงไฟที่ได้เป็น 5 โวลต์ต่อไป และไฟลบจะทำการสร้างโดยใช้ไอซี ICL 7660 โดยการป้อนไฟบวกซึ่งก็จะให้ผลเป็นไฟลบออกมา ลักษณะของวงจรแหล่งจ่ายไฟจะเป็นดังรูป 3.6

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



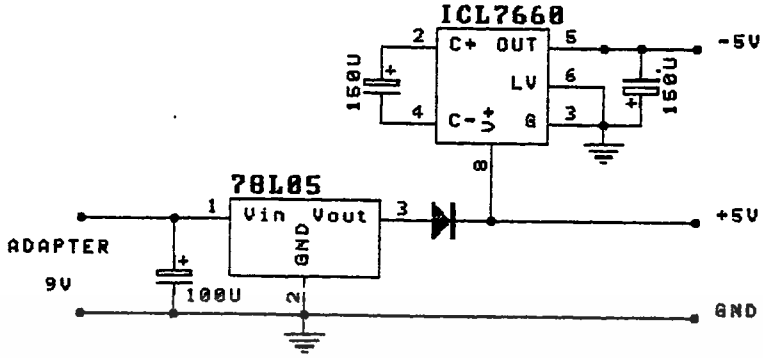
รูป 3.4 แผนผังแสดงลำดับการทำงานของตัวส่งสัญญาณอนาลอกขนาดจิ๋ว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูป 3.5 แผนผังแสดงลำดับการทำงานของการอ่านค่าจาก A/D แปลงสัญญาณอนาลอกเป็นสัญญาณดิจิตอลทั้ง 8 ช่องสัญญาณ

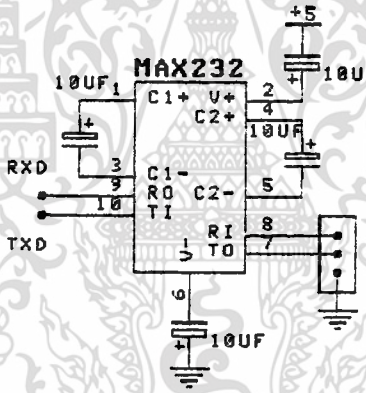
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูป 3.6 วงจรแหล่งจ่ายไฟ

3.1.4 ช่องสัญญาณสื่อสารแบบต่าง ๆ ที่เลือกใช้

1) RS232



รูป 3.7 วงจรการสื่อสารแบบ RS232

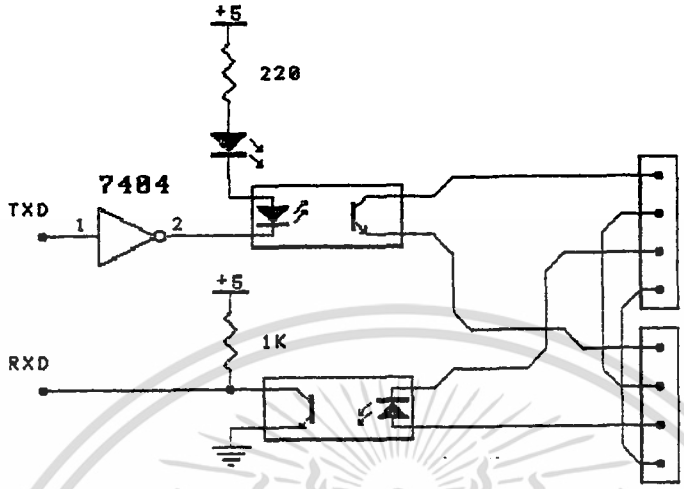
ลักษณะการสื่อสารนี้สามารถต่อได้โดยตรงกับสถานีศูนย์กลางโดยใช้ตัวส่งสัญญาณอนาลอกขนาดจิ๋วเพียงตัวเดียว ลักษณะวงจรการสื่อสารแบบ RS232 แสดงดังรูป 3.7

2) ลูปกระแส

หัวต่อตัวบนจะทำการต่อกับตัวรับ และแปลงสัญญาณเพื่อแปลงสัญญาณการสื่อสารเป็น RS232 ในการติดต่อกับสถานีศูนย์กลางรวมส่วนตัวล่างจะเป็นหัวต่อสำหรับตัวส่งสัญญาณอนาลอกขนาดจิ๋วตัวอื่น ๆ ที่จะทำการพ่วง แต่ถ้าไม่มีตัวพ่วงอีกก็จะทำการ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

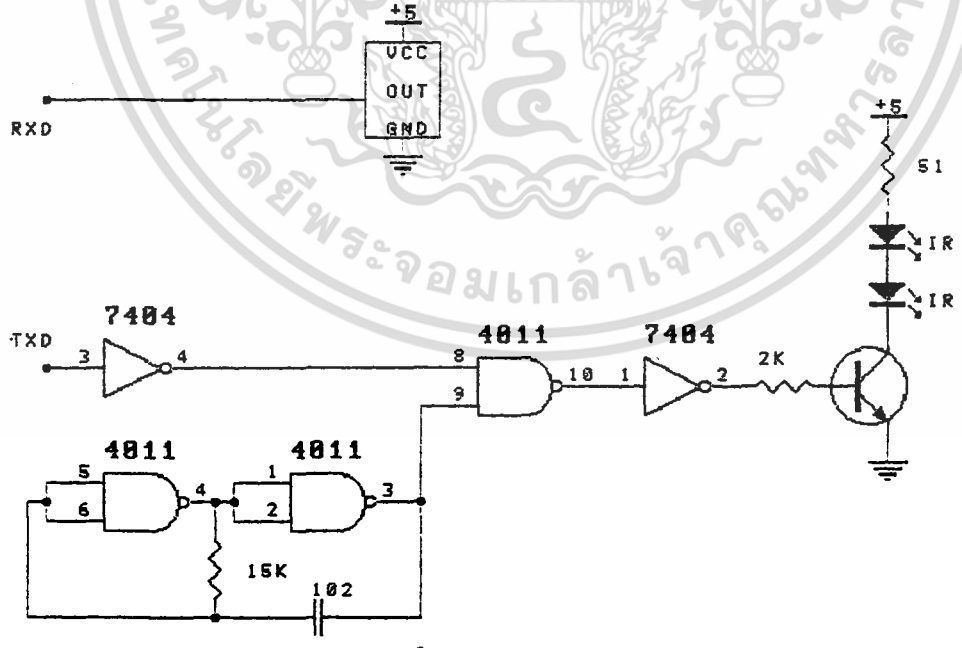
ลัดวงจรระหว่างขา 1 และ 2 , ขา 3 และ 4 ลักษณะวงจรการสื่อสารแบบลูปกระแสแสดงดังรูป3.8



รูป 3.8 วงจรการสื่อสารแบบลูปกระแส

3) อินฟราเรด

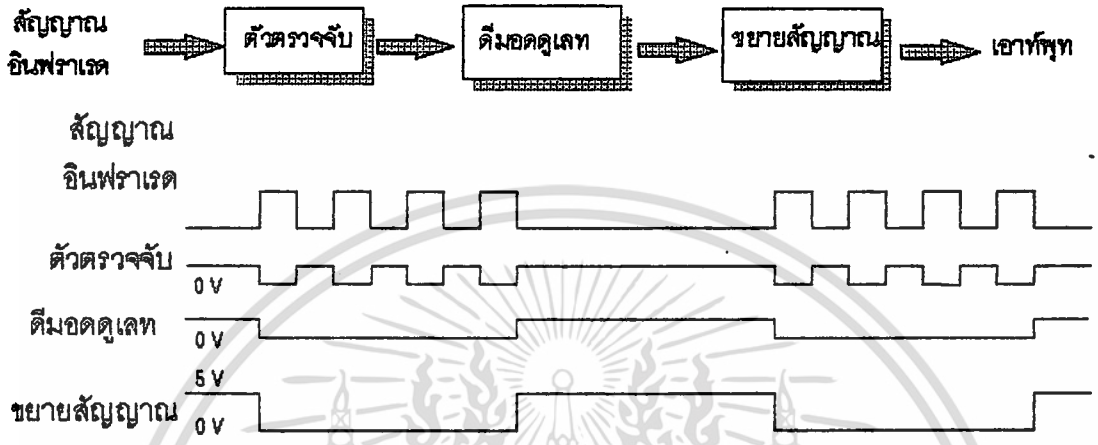
ลักษณะวงจรการสื่อสารอินฟราเรดแสดงดังรูป 3.9



รูป 3.9 วงจรการสื่อสารอินฟราเรด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ลักษณะของตัวรับอินฟราเรดจะเป็นตัวตรวจจับสัญญาณอินฟราเรด ซึ่งจะมีการดีมอดดูเลทและจะทำการขยายสัญญาณให้เป็น 5 โวลต์ เมื่อไม่มีสัญญาณอินฟราเรดและเมื่อมีสัญญาณอินฟราเรดจะเป็น 0 โวลต์ ซึ่งเป็นสัญญาณ TTL ที่ถูกต้องดังรูป 3.10



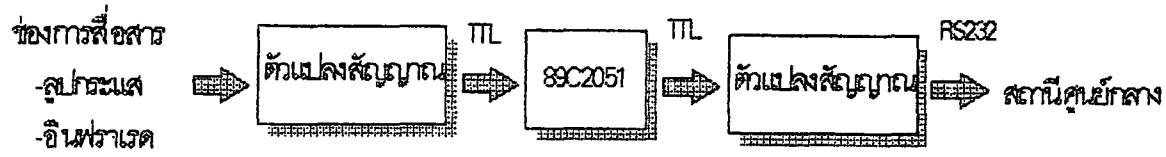
รูป 3.10 ไดอะแกรมและลักษณะสัญญาณของตัวรับอินฟราเรด

ลักษณะการส่งจะเป็นการส่งสัญญาณ TTL และทำการมอดดูเลทโดยใช้วงจรอะสเตเบิลมัลติไวเบรเตอร์ ที่จะสร้างสัญญาณสี่เหลี่ยมความถี่ประมาณ 40 kHz เพื่อไปทำการขับทรานซิสเตอร์ให้เปิด - ปิดตัวส่งสัญญาณอินฟราเรด

3.2 ตัวรับและแปลงสัญญาณ

ตัวรับและแปลงสัญญาณจะทำการรับสัญญาณจากสถานีศูนย์กลาง ซึ่งจะเป็นการติดต่อแบบ RS232 แล้วทำการแปลงสัญญาณไปเป็นการสื่อสารแบบลูปกระแสหรืออินฟราเรด เพื่อติดต่อกับตัวส่งสัญญาณอนาลอกขนาดจิ๋ว อีกทั้งยังทำการรับสัญญาณจากตัวส่งสัญญาณอนาลอกขนาดจิ๋ว ซึ่งอาจจะเป็นการสื่อสารแบบลูปกระแสหรือเป็นอินฟราเรดแล้วแปลงเป็น RS232 เพื่อส่งให้กับสถานีศูนย์กลาง โดยที่จะมีไมโครคอนโทรลเลอร์ทำการประมวลผลเบื้องต้นคือการตัดสัญญาณรบกวน หรือทำการตรวจสอบความถูกต้องของสัญญาณ (checksum) ซึ่งจะแสดงลักษณะของไดอะแกรมดังรูป 3.11

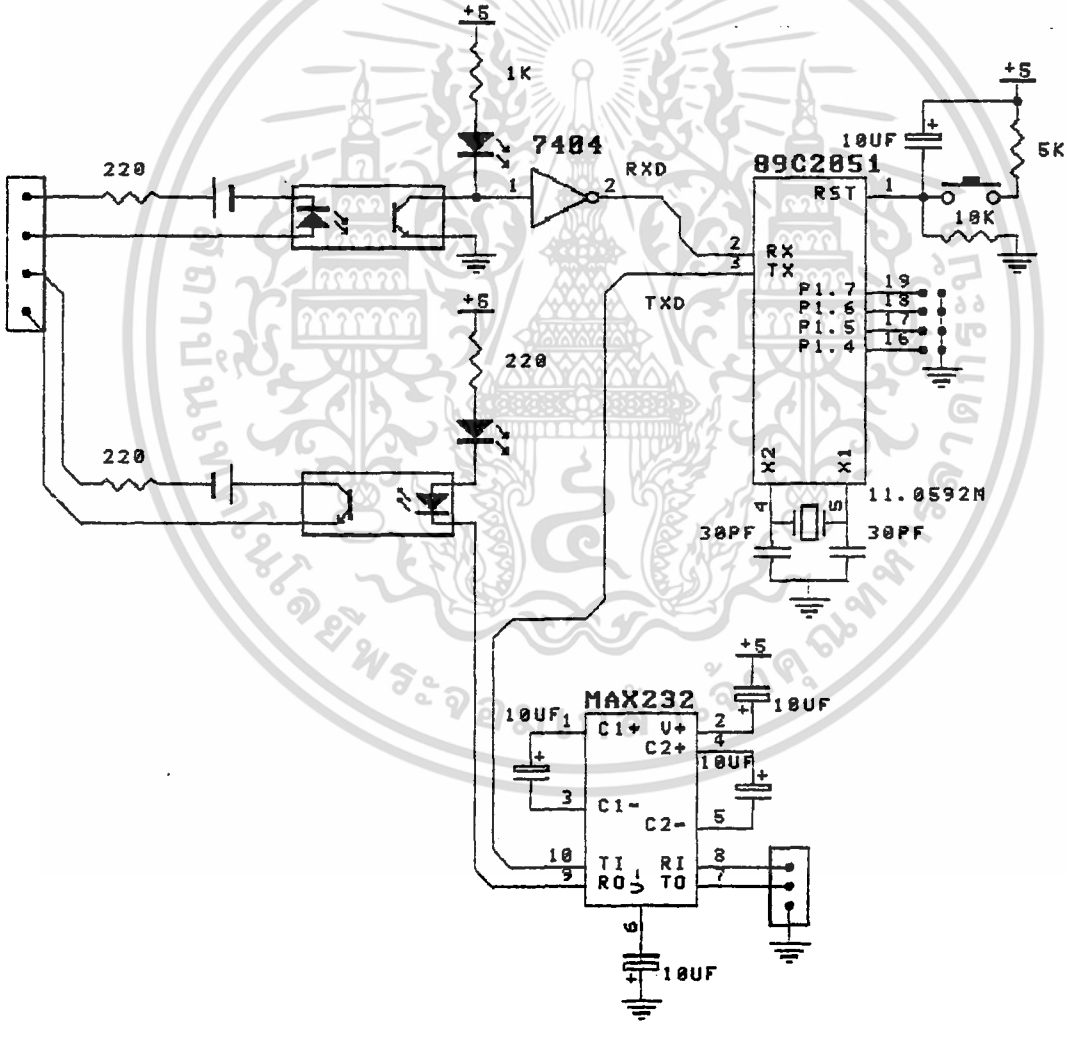
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูป 3.11 ไดอะแกรมของตัวรับและแปลงสัญญาณ

3.2.1 วงจรแปลงการสื่อสารแบบลูปกระแส ↔ RS232

วงจรแปลงการสื่อสารแบบลูปกระแสเป็น RS232 แสดงดังรูป 3.12

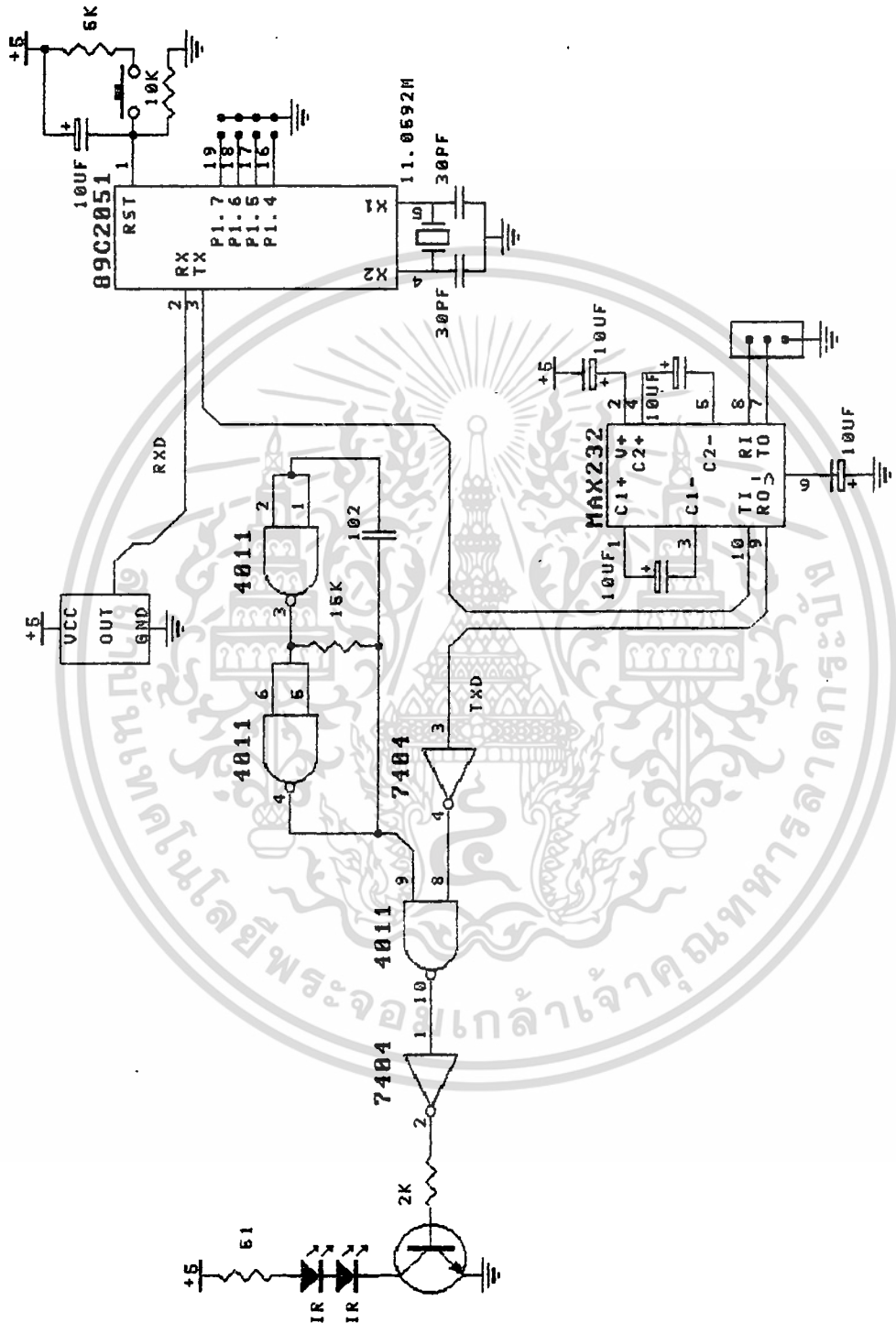


รูป 3.12 วงจรแปลงการสื่อสารแบบลูปกระแสเป็น RS232

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.2.2 วงจรแปลงการสื่อสารแบบอินฟราเรด ↔ RS232

วงจรแปลงการสื่อสารแบบอินฟราเรดเป็น RS232 แสดงดังรูป 3.13



รูป 3.13 วงจรแปลงการสื่อสารแบบอินฟราเรดเป็น RS232

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เมื่อเริ่มต้นการทำงานจะต้องมีการเลือกความเร็วในการส่งข้อมูลอนุกรม ลักษณะการประมวลผลข้อมูล , และเลือกชนิดของการสื่อสารข้อมูล ให้ถูกต้องกับลักษณะสัญญาณที่จะได้รับจากตัวส่งสัญญาณอนาลอกขนาดจิ๋ว และสถานีศูนย์กลางโดยจะแสดงลักษณะการเลือกจัมเปอร์ต่าง ๆ ดังตาราง 3.2

จัมเปอร์	ฟังก์ชัน
	ความเร็ว 1200 บิตต่อวินาที
	ความเร็ว 4800 บิตต่อวินาที
	ความเร็ว 9600 บิตต่อวินาที
	ความเร็ว 19200 บิตต่อวินาที

จัมเปอร์เลือกความเร็วในการส่งข้อมูลอนุกรม (P1.6 , P1.7)

จัมเปอร์	ฟังก์ชัน
	ไม่กรองสัญญาณรบกวน
	กรองสัญญาณรบกวน
	ตรวจสอบความถูกต้อง (checksum)

จัมเปอร์เลือกลักษณะการประมวลผลข้อมูล (P1.4 , P1.5)

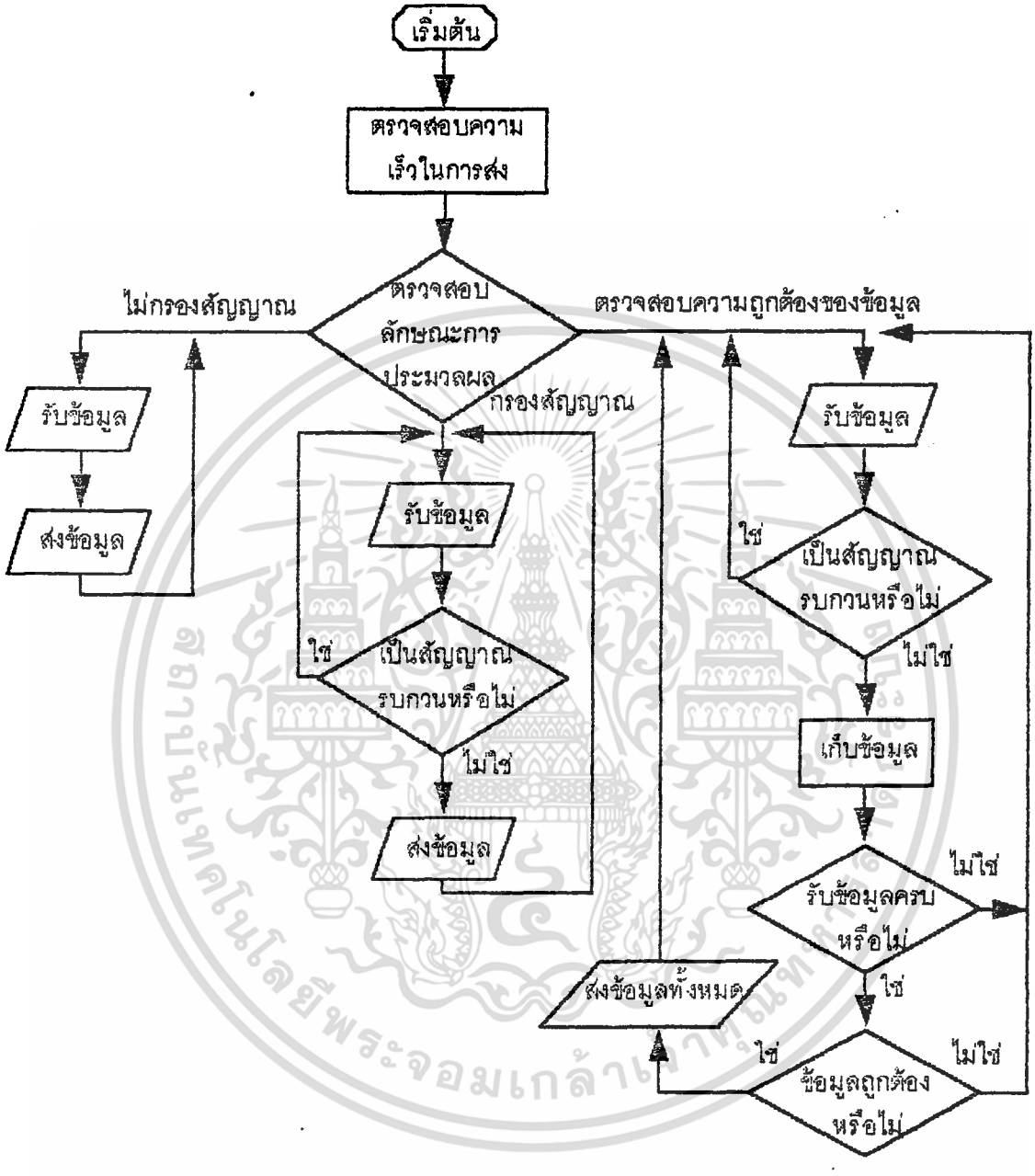
จัมเปอร์	ฟังก์ชัน
	อินฟราเรด
	ลูปกระแส

จัมเปอร์เลือกชนิดของการส่งข้อมูล

ตาราง 3.2 จัมเปอร์ของตัวรับและแปลงส่งสัญญาณ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.2.3 การทำงานของโปรแกรม



รูป 3.14 แผนผังแสดงลำดับการทำงานของตัวรับและแปลงสัญญาณ

ลักษณะการทำงานของโปรแกรมจะแสดงดังรูป 3.14 ซึ่งจะตรวจสอบความเร็วในการส่งข้อมูลอนุกรมและตรวจสอบลักษณะการประมวลผล โดยที่ถ้าเป็นการประมวลผลที่ไม่ต้องกรองสัญญาณรบกวนจะทำการรอรับข้อมูล 1 ไบต์ แล้วทำการส่งข้อมูลออกไปทันที แต่ถ้าลักษณะการประมวลผลเป็นแบบการกรองสัญญาณรบกวนในทางดิจิตอลของข้อมูล

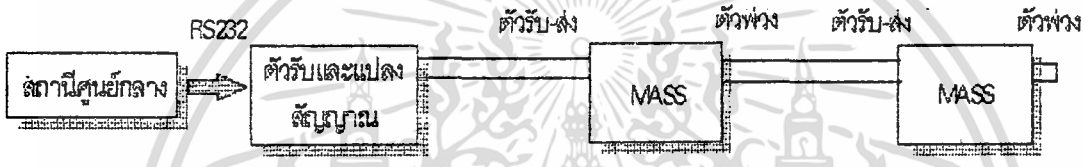
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จะตรวจสอบว่าข้อมูลเป็น 0-9 , A-F หรือไม่ อีกทั้งยังตรวจสอบข้อมูลว่าเป็นตัวควบคุมหรือไม่ เช่น SOH ,EOT เป็นต้น ถ้าเป็นข้อมูลเหล่านี้ก็จะทำการส่งข้อมูลนี้ออกไปทุก ๆ 1 ไบต์ และถ้าลักษณะการประมวลผลของข้อมูลเป็นการตรวจสอบความถูกต้อง (checksum) จะทำการรับข้อมูลจนครบแพคเกจ (package) และตรวจสอบผลข้อมูลว่าถูกต้องหรือไม่ ถ้าถูกต้องก็จะส่งข้อมูลทั้งหมดออกมา

3.2.4 การเชื่อมต่อตัวรับและแปลงสัญญาณกับตัวส่งสัญญาณอนาลอกขนาดจิ๋ว

-การเชื่อมต่อกันด้วยการสื่อสารแบบลูปกระแส

การเชื่อมต่อกันด้วยการสื่อสารแบบลูปกระแส แสดงดังรูป 3.15

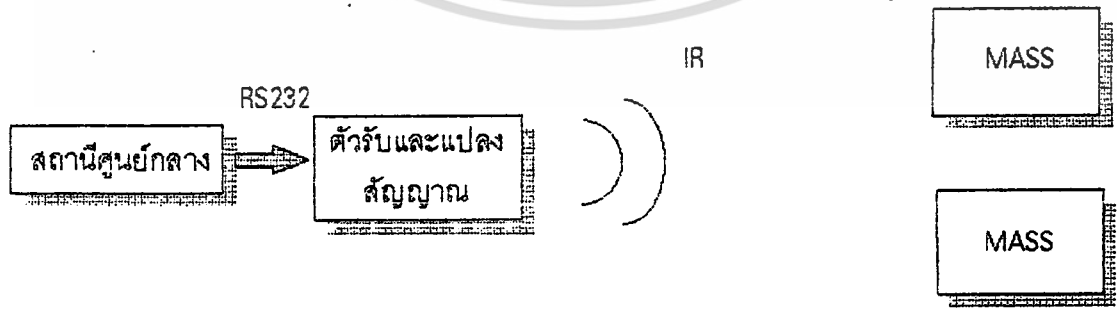


รูป 3.15 การเชื่อมต่อกันด้วยการสื่อสารแบบลูปกระแส

ลักษณะสายที่ใช้จะทำการตีเกลียวทั้งในคู่ของสายสัญญาณรับและในคู่ของสายสัญญาณส่ง โดยที่ตัวส่งสัญญาณอนาลอกขนาดจิ๋วที่พ่วงสุดท้ายจะต้องลัดวงจรหัวต่อพ่วงด้วยทั้งในคู่รับและคู่ส่ง

-การเชื่อมต่อกันด้วยอินฟราเรด

การเชื่อมต่อกันด้วยอินฟราเรด แสดงดังรูป 3.16



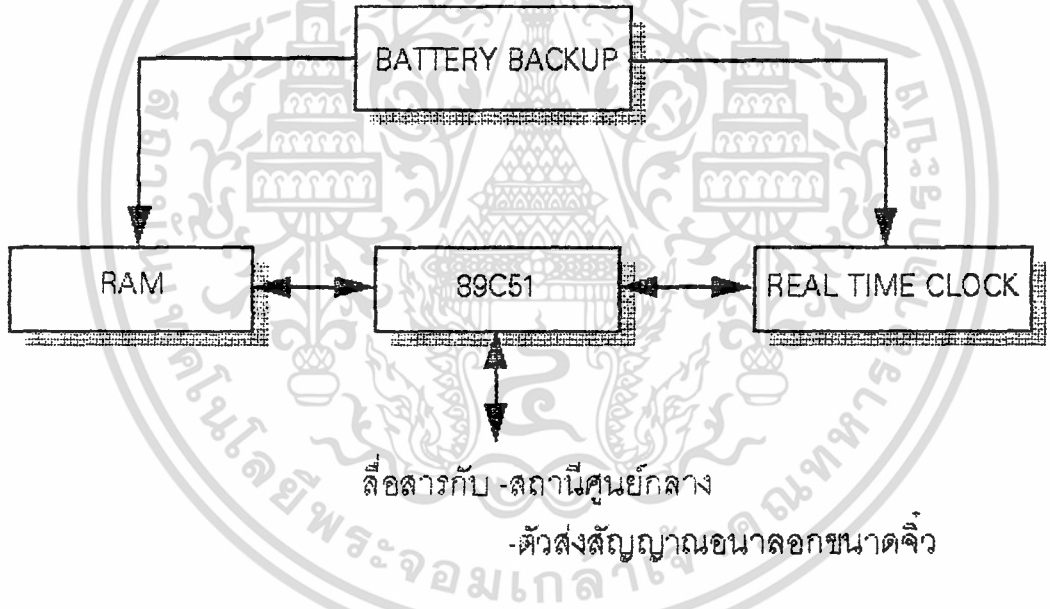
รูป 3.16 การเชื่อมต่อกันด้วยอินฟราเรด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ลักษณะการเชื่อมต่อนี้ตัวส่งสัญญาณอนาลอกขนาดจิ๋วจะต้องอยู่ในรัศมีเบี่ยงเบนได้ของสัญญาณอินฟราเรดของตัวรับและแปลงสัญญาณจึงจะสามารถติดต่อสื่อสารกันได้

3.3 ตัวอ่านค่าสัญญาณและจัดเก็บข้อมูล

ตัวอ่านค่าสัญญาณและจัดเก็บข้อมูลจะทำหน้าที่อ่านค่าสัญญาณตามแต่เวลาที่เราตั้งไว้อย่างต่อเนื่องโดยที่จะทำการอ่านค่าจากตัวส่งสัญญาณอนาลอกขนาดจิ๋วตามแต่แอดเดรสที่เราตั้งไว้ด้วยและจะทำการจัดเก็บข้อมูลเอาไว้เพื่อรอการนำข้อมูลมาแสดงผลเมื่อสถานีศูนย์กลางพร้อมที่จะทำงานได้โดยที่ลักษณะของตัวอ่านค่าสัญญาณและจัดเก็บข้อมูลจะมีลักษณะตามไดอะแกรมดังรูป 3.17

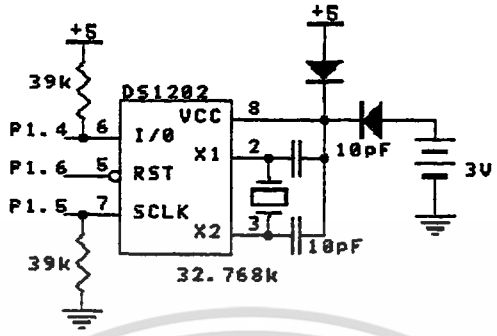


รูป 3.17 แสดงลักษณะของตัวอ่านค่าสัญญาณและจัดเก็บข้อมูล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.3.1 การเชื่อมต่อกับชิปกำเนิดเวลาจริง

ลักษณะการเชื่อมต่อกับชิปกำเนิดเวลาจริงจะเป็นดังรูป 3.18



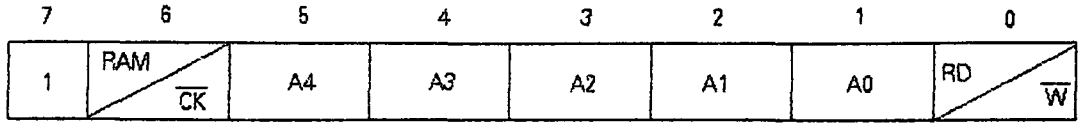
รูป 3.18 แสดงลักษณะการเชื่อมต่อกับชิปกำเนิดเวลาจริง

โดยที่ในการรับหรือส่งข้อมูลระหว่างชิปกำเนิดเวลาจริงเบอร์ DS1202 ในตอนเริ่มต้น ไมโครคอนโทรลเลอร์จะต้องส่งข้อมูลเพื่อกำหนดการทำงานให้แก่ชิปกำเนิดเวลาจริงเสียก่อน ข้อมูลที่ชิปกำเนิดเวลาจริงได้รับในตอนเริ่มต้นจะมีขนาด 1 ไบต์ซึ่งเรียกว่าไบต์คำสั่ง (command byte) และเนื่องจากข้อมูลในไบต์นี้จะเป็นตัวกำหนดการทำงานของตัวกำเนิดเวลาจริง ดังนั้นแต่ละบิตในไบต์ จะมีความหมายแตกต่างกันไปดังนี้

- MSB (บิต 7) ต้องเป็น 1 เสมอ ถ้าเป็น 0 การทำงานต่อจากนี้จะถูกหยุด
- บิต 6 ถ้าเป็น 0 จะระบุว่าต้องการติดต่อกับรีจิสเตอร์สำหรับเก็บเวลา (clock / calender register) ดังนั้นข้อมูลที่รับส่งกันจะเป็นเวลา หากบิตนี้มีค่าเป็น 1 จะระบุว่าเป็นการติดต่อกับหน่วยความจำ
- บิต 1 ถึง 5 เป็นตัวระบุตำแหน่งหน่วยความจำ (ทั้งหน่วยความจำที่เป็นรีจิสเตอร์สำหรับเก็บเวลาและหน่วยความจำทั่วไป) ที่ต้องการเข้าถึงไม่ว่าจะเป็นการอ่านหรือเขียนข้อมูลซึ่งควบคุมด้วยบิต 0 ดังจะได้กล่าวต่อไป
- บิต 0 จะระบุว่าเป็นการเขียนหรืออ่านข้อมูล ถ้าเป็น 0 หมายถึงการเขียนข้อมูลลงไปในชิป หากเป็น 1 หมายถึงการอ่านข้อมูลจากชิป

ในการส่งไบต์คำสั่งไปยังชิปกำเนิดเวลาจริงจะเริ่มต้นด้วยบิต 0 ก่อนเสมอ ซึ่งจะแสดงดังรูป 3.19

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูป 3.19 แสดงลักษณะการส่งไบต์คำสั่ง

<u>รายละเอียดของการควบคุมชิปกำเนิดเวลาจริง</u>

-**<u>เบิซทโหมด (Burse Mode)</u>** หมายถึงการรับหรือส่งข้อมูลครั้งละหลายไบต์ในการติดต่อแต่ละครั้งโดยสามารถกำหนดได้ว่าข้อมูลที่ต้องการติดต่อเป็นรีจิสเตอร์สำหรับเก็บเวลาหรือหน่วยความจำที่ใช้เก็บข้อมูลหากเป็นการรับส่งข้อมูลกับรีจิสเตอร์สำหรับเก็บเวลาก็จะรับส่งกันครั้งละ 8 ไบต์ หากเป็นการรับส่งข้อมูลกับหน่วยความจำจะรับส่งกันครั้งละ 24 ไบต์ ดังตาราง 3.3

BURSE FUNCTION	DATA (BYTES)	SCLK
CLOCK	8	72
RAM	24	200

ตาราง 3.3 แสดงการใช้เบิซท โหมด

ในการรับหรือส่งในเบิซทโหมด จะเริ่มต้นที่บิต 0 ของหน่วยความจำตำแหน่ง 0 ก่อนเสมอ ไม่ว่าจะเป็นรีจิสเตอร์สำหรับเก็บเวลาหรือหน่วยความจำ สำหรับเก็บข้อมูลทั่วไป

-**<u>คำสั่งป้องกันการเขียน</u>** เวลาที่เดินอยู่ภายในรีจิสเตอร์ทำหน้าที่เก็บเวลาและข้อมูลในหน่วยความจำทั้ง 24 ตำแหน่ง ป้องกันการเขียนข้อมูลใด ๆ ลงไป โดยที่จะควบคุมจากรีจิสเตอร์ป้องกันการเขียน ซึ่งเป็นรีจิสเตอร์ตำแหน่งที่ 7 ของรีจิสเตอร์ที่ทำหน้าที่เก็บเวลาดังตาราง 3.4 ที่โดยมีบิตป้องกันการเขียนซึ่งเป็นบิตที่ 7 ของรีจิสเตอร์ตัวนี้เป็นตัวกำหนดการทำงาน หากบิตป้องกันการเขียนเป็น 0 หมายถึงสามารถเขียนข้อมูลใด ๆ ลงไปบนรีจิสเตอร์ที่เก็บเวลาหรือหน่วยความจำได้ หากบิตนี้เป็น 1 หมายถึง ชิปกำเนิดเวลาจริงอยู่ในสถานะป้องกันการเขียนข้อมูล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รีจิสเตอร์	ฟังก์ชัน	command address(hex)	เขียน = W อ่าน = R	range data	รีจิสเตอร์ที่กำหนด 7	6	5	4	3	2	1	0
0	วินาที	80 81	W R	00-59	CH	10 วินาที			วินาที			
1	นาที	82 83	W R	00-59	0	10 นาที			นาที			
2	12 ชั่วโมง 24 ชั่วโมง	84 85	W R	01-12	12 24	0 0	AP 10		ชั่วโมง			
3	วัน	86 87	W R	00-23 01-31	0	0	10 วัน		วัน			
4	เดือน	88 89	W R	01-12	0	0	0	10	เดือน			
5	วัน	8A 8B	W R	01-07	0	0	0	0	วัน			
6	ปี	8C 8D	W R	00-99	10 ปี				ปี			
7	WRITE PROTECT	8E 8F	W R	00-80	WP	เป็น 0 ทั้งหมด						

ตาราง 3.4 แสดงข้อมูลการทำงานของรีจิสเตอร์และหน่วยความจำใน DS1202

-<u>รีเซ็ต และสัญญาณนาฬิกาควบคุม</u> (Reset and Clock Control) การรับหรือส่งข้อมูลทั้งหมดจะต้องเริ่มโดยให้ขา RST มีสถานะเป็น 1 ก่อนเสมอ โดย RST มีหน้าที่หลักอยู่ 2 ประการดังนี้

1. RST ใช้ควบคุมการเขียนหรืออ่านข้อมูลในรีจิสเตอร์เลื่อนข้อมูล

2. RST ใช้เป็นสัญญาณในการหยุดการทำงานใด ๆ กับชิปกำเนิดเวลาจริงเบอร์ DS1202 โดยปกติการเขียนข้อมูลเข้าไปในชิปกำเนิดเวลาจริงเบอร์ DS1202 จะเกิดขึ้นในช่วงขาขึ้นของสัญญาณที่ขา SCLK ส่วนการอ่านข้อมูลจากชิปกำเนิดเวลาจริงเบอร์ DS1202 จะเกิดขึ้นในช่วง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ขอบขาลงของสัญญาณที่ขา SCLK โดยในการติดต่อขา RST ต้องเป็น 1 ตลอดเวลา หากขา RST มีสถานะเป็น 0 หมายถึงยกเลิกการติดต่อหรือสิ้นสุดการติดต่อ

-**ข้อมูลอินพุท** (Data Input) ในตอนเริ่มต้นติดต่อระหว่างชิป ชิปกำเนิดเวลาจริงเบอร์ DS1202 กับไมโครคอนโทรลเลอร์ไบต์แรกจะต้องเป็นไบต์คำสั่งเสมอ หากในไบต์คำสั่งระบุว่าเป็นการเขียนข้อมูลไปในชิป ข้อมูลจะถูกรับเข้ามาในช่วงขอบขาขึ้นของ SCLK เท่านั้น โดยเริ่มต้นด้วยบิต 0 ก่อนเสมอ และหากเป็นคำสั่งให้รับส่งครั้งละ 1 ไบต์ เมื่อข้อมูลได้รับเข้ามาครบแล้ว สัญญาณ SCLK ที่รับได้เกินจะถูกละเลยไป หากเป็นคำสั่งให้รับส่งแบบเบิซทโหมด ซึ่งรับส่งครั้งละ 24 ไบต์ ก็มีลักษณะเช่นเดียวกันคือ เมื่อรับข้อมูลครบ 24 ไบต์แล้ว สัญญาณ SCLK ที่รับได้เกินจะถูกละเลยไปเช่นเดียวกัน

-**ข้อมูลเอาท์พุท** (Data Output) หลังจากรับไบต์คำสั่งแล้ว หากมีการระบุว่าเป็นการอ่านข้อมูลจากชิปกำเนิดเวลาจริงเบอร์ DS1202 ข้อมูลจะถูกส่งออกจากชิปสู่ภายนอกในขณะช่วงขอบขาลงของ SCLK หลังจากมีการรับไบต์คำสั่งเรียบร้อยแล้ว นั่นคือ บิตแรกจะถูกส่งออกจากชิปกำเนิดเวลาจริงซึ่งจะเกิดขึ้นในช่วงขอบขาลงของสัญญาณนาฬิกาลูกที่ 9 นั่นเอง

-**แฟลกหยุดเวลา** (Clock Halt Flag) บิต 7 ของรีจิสเตอร์ที่เก็บค่าวินาทีจะเป็นตัวบอกให้ชิปกำเนิดเวลาจริงเบอร์ DS1202 หยุดการทำงานของวงจรในส่วนออสซิลเลเตอร์เมื่อบิตนี้มีค่าเป็น 1 ซึ่งเป็นผลให้นาฬิกาภายในชิปหยุดทำงานไปด้วย และจะบังคับให้ชิปอยู่ในสถานะกินไฟน้อยโดยใช้กระแสไม่เกิน 100 นาโนแอมแปร์ และเมื่อบิตนี้เป็น 0 อีกครั้งวงจรออสซิลเลเตอร์จะเริ่มทำงานต่อทันที

-**โหมด AM-PM/12-24** บิต 7 ของรีจิสเตอร์ที่ทำหน้าที่ เก็บชั่วโมงถูกกำหนดให้เป็นตัวเลือกว่าจะเก็บข้อมูลเป็นแบบ 12 ชั่วโมงหรือ 24 ชั่วโมง โดย

- บิตที่ 7เป็น 1 จะเป็นการเลือกให้เก็บค่าแบบ 12 ชั่วโมง โดยมีบิต 5 เป็นตัวบอกว่าเป็นช่วงกลางวันหรือกลางคืน โดย 1 จะหมายถึง PM และ 0 จะหมายถึง AM

- บิตที่ 7 เป็น 0 จะเป็นการเลือกให้เก็บค่าแบบ 24 ชั่วโมงและบิต 5 จะเป็นบิตที่แสดงหลักสิบตัวที่ 2 ของชั่วโมง (20-23)

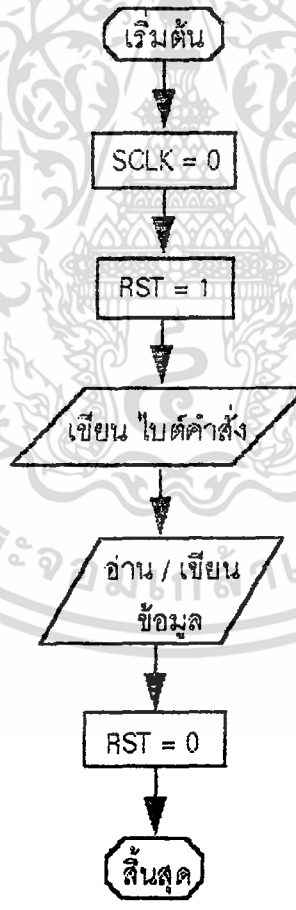
-**บิตป้องกันการเขียน** บิต 7 ของรีจิสเตอร์ป้องกันการเขียนจะเป็นบิตป้องกันการเขียนโดย 7 บิตแรก ถูกบังคับให้เป็น 0 หมด ทำให้อ่านค่าได้เป็น 0 เสมอ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

-**<u>เวลา / ปฏิทิน เบิซทโหมด</u>** (Clock / Calender Burst Mode) ไบต์คำสั่งที่มีค่า BEH จะเป็นการระบุว่าให้มีการเขียนข้อมูลในรีจิสเตอร์ที่เก็บเวลาในแบบเบิซทโหมด หากไบต์คำสั่งมีค่า BFH จะระบุว่าให้มีการอ่านข้อมูลในรีจิสเตอร์ในแบบเบิซทโหมด เช่นกัน ซึ่งในเวลา / ปฏิทิน เบิซทโหมด นี้จะมีการรับหรือส่งข้อมูลคราวละ 8 ไบต์ติดต่อกัน

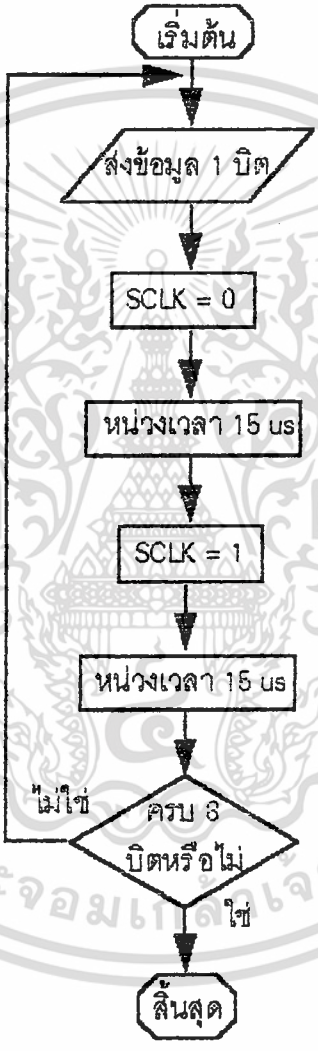
-**<u>หน่วยความจำภายในเบิซทโหมด</u>** (RAM Burst Mode) ไบต์คำสั่งที่มีค่า FEH จะเป็นการระบุว่าให้มีการเขียนข้อมูลในหน่วยความจำแบบเบิซทโหมด หากไบต์คำสั่งมีค่า FFH จะระบุว่าให้มีการอ่านข้อมูลในหน่วยความจำแบบเบิซทโหมด ทำนองเดียวกับเวลา / ปฏิทิน เบิซทโหมด โดยที่ข้อมูลจะมี 24 ไบต์

แผนผังแสดงลำดับการทำงาน ในการอ่าน / เขียนข้อมูลเวลาลงชิปกำเนิดเวลาจริงเบอร์ DS1202 จะเป็นดังรูป 3.20



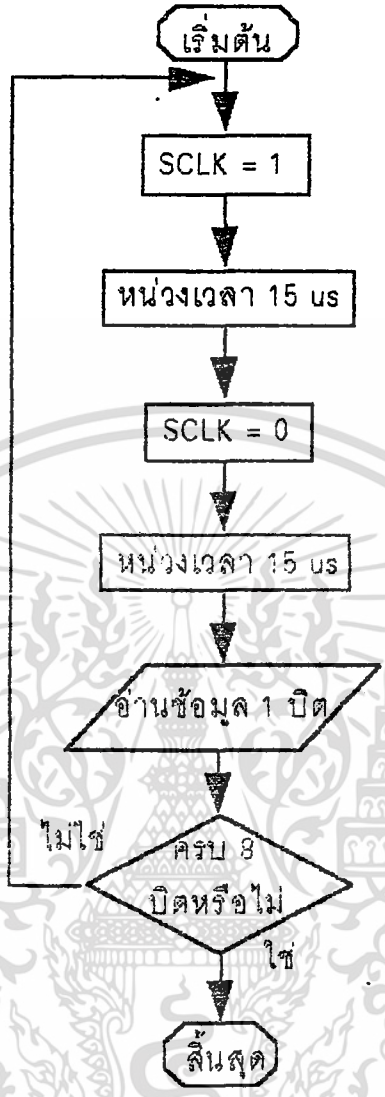
(ก) แผนผังแสดงลำดับการทำงานการอ่าน / เขียนข้อมูลเวลาลงชิปกำเนิดเวลาจริง DS1202

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



(ข) แผนผังแสดงลำดับการทำงานการเขียนข้อมูล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



(ค) แผนผังแสดงลำดับการทำงานการอ่านข้อมูล

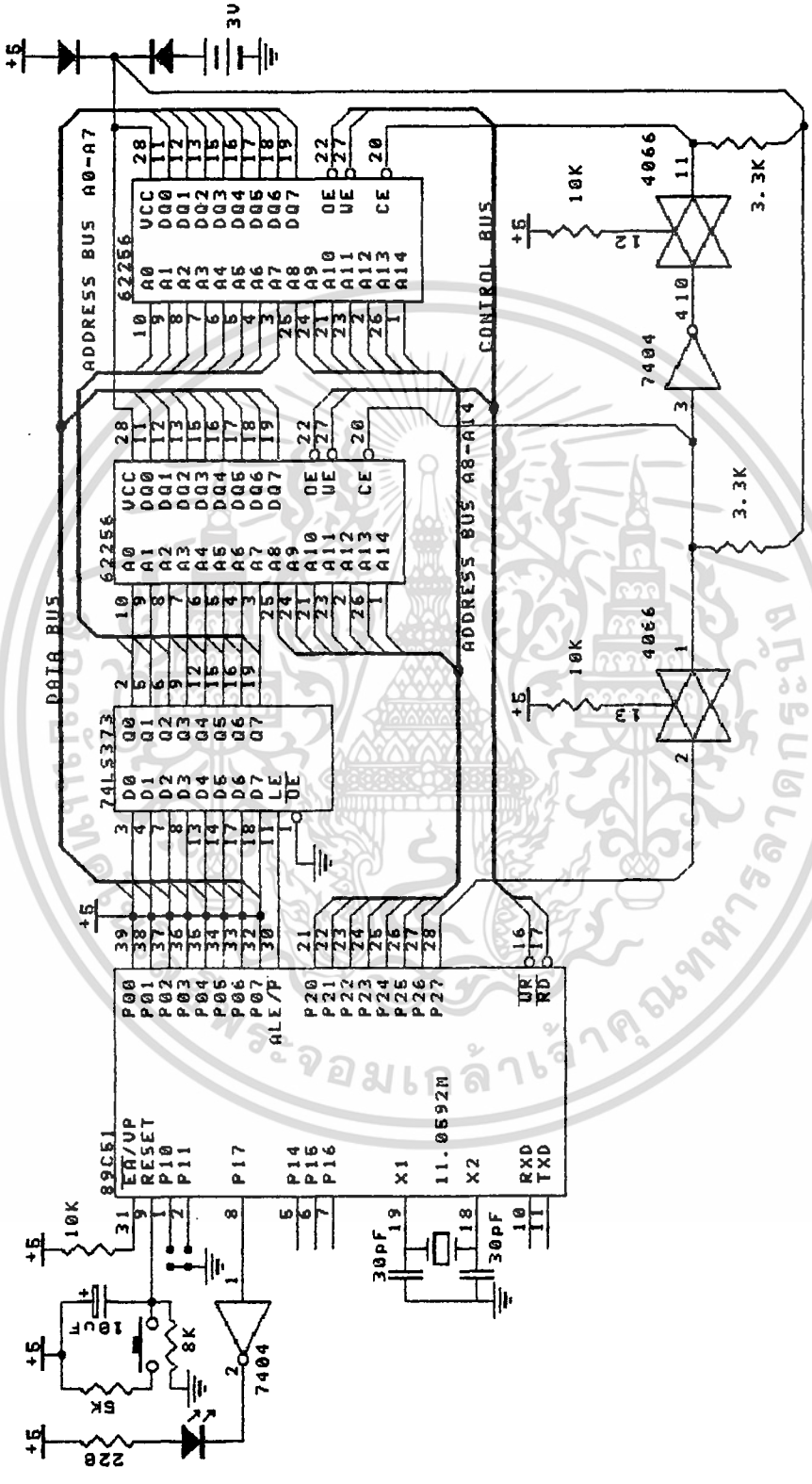
รูป 3.20 แผนผังแสดงลำดับการทำงานการอ่าน / เขียนข้อมูลเวลาครั้งละ 1 ไบต์ของซิปกำเนิดเวลาจริง DS1202

การส่งไบต์คำสั่งจะส่ง 1 ไบต์ แต่ในการอ่านหรือเขียนข้อมูลขึ้นอยู่กับว่าเราได้ทำการเลือกในโหมดใด ถ้าเป็นเบิซทโหมดสำหรับเวลา / ปฏิทิน ก็จะมีข้อมูลทั้งหมด 8 ไบต์ติดต่อกันนั่นเอง แต่ถ้าเราเลือกเป็นแบบส่งข้อมูลทีละไบต์เราจะต้องทำการส่งไบต์คำสั่งทุก ๆ การเขียน / อ่าน 1 ไบต์ด้วยเช่นกัน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.3.2 การเชื่อมต่อกับหน่วยความจำภายนอก

ลักษณะการเชื่อมต่อกับหน่วยความจำภายนอกจะเป็นดังรูป 3.21



รูป 3.21 แสดงลักษณะการเชื่อมต่อกับหน่วยความจำภายนอก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ลักษณะการเชื่อมต่อจะทำการแบคอัพ (Backup) เพื่อให้ข้อมูลยังคงอยู่ในขณะที่ไม่มีไฟเลี้ยงโดยที่จะใช้แบตเตอรี่ 3 โวลต์ และอนาลอกสวิตช์ (analog switch 4066) ร่วมกันในการเชื่อมต่อ ซึ่งเมื่อไม่มีไฟเลี้ยงจ่ายให้วงจรจะทำให้อนาลอกสวิตช์ตัดขาดการเชื่อมต่อทำให้ขาสัญญาณ CE (chip enable) เป็น 1 จึงตัดการเชื่อมต่อกับไมโครคอนโทรลเลอร์ อีกทั้งไฟเลี้ยงจากแบตเตอรี่แบคอัพ (battery backup) จะทำการจ่ายให้แก่หน่วยความจำภายนอก อีกทั้งจ่ายให้กับรีจิสเตอร์กำเนิดเวลาจริง อีกด้วย เพื่อให้เวลายังคงเดินต่อไป

3.3.3 <u>การทำงานของโปรแกรมตัวอ่านค่าสัญญาณและจัดเก็บข้อมูล</u>

ลักษณะการทำงานของตัวอ่านค่าสัญญาณและจัดเก็บข้อมูลจะทำการแสกนอ่านข้อมูลจากตัวส่งสัญญาณอนาลอกขนาดจิ๋วโดยต้องมีการตั้งค่าเริ่มต้น ดังนี้

R คือทำการอ่านค่าเวลาที่เดินของตัวอ่านค่าสัญญาณและจัดเก็บข้อมูล

W คือการเขียนเวลาใหม่ให้กับตัวอ่านค่าสัญญาณและจัดเก็บข้อมูล

โดยมีรูปแบบดังนี้

W □□ / □□ / □□ □□ : □□ : □□

วัน เดือน ปี ชั่วโมง นาที วินาที

A คือการตั้งว่าจะทำการอ่านค่าตัวส่งสัญญาณอนาลอกขนาดจิ๋วที่ตำแหน่งแอดเดรสใด

โดยมีรูปแบบดังนี้

A □□ , □□ , □□

S คือการตั้งว่าจะทำการอ่านค่าห่างกันเท่าใด

โดยมีรูปแบบดังนี้

S □□ : □□ : □□

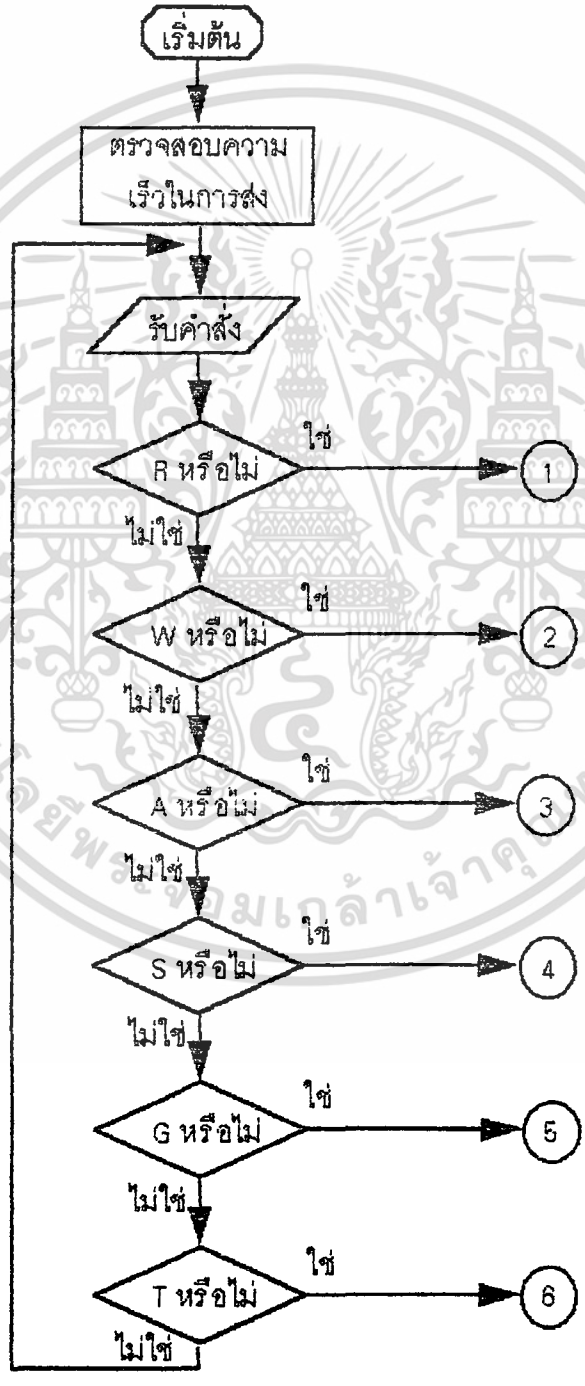
ชั่วโมง นาที วินาที

G คือการสั่งให้ตัวอ่านค่าสัญญาณและจัดเก็บข้อมูลเริ่มทำการแสกนข้อมูลจากตัวส่งสัญญาณอนาลอกขนาดจิ๋ว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

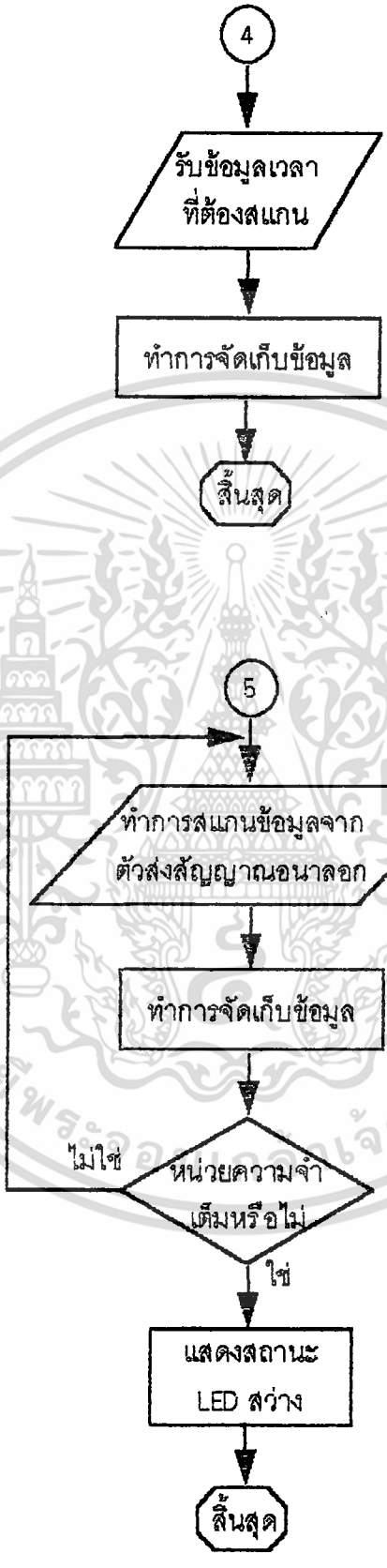
T คือการสั่งให้ตัวอ่านค่าสัญญาณและจัดเก็บข้อมูลทำการถ่ายข้อมูลที่เก็บไว้แสดงผลบนสถานีศูนย์กลาง

เมื่อทำการแสกนข้อมูลจากตัวส่งสัญญาณอนาลอกขนาดจิ๋วถ้าข้อมูลเต็มหน่วยความจำจะทำให้ LED แสดงสถานะติด อีกทั้งเมื่อทำการถ่ายข้อมูลครบแล้วก็จะทำให้ LED แสดงสถานะติดเช่นเดียวกัน

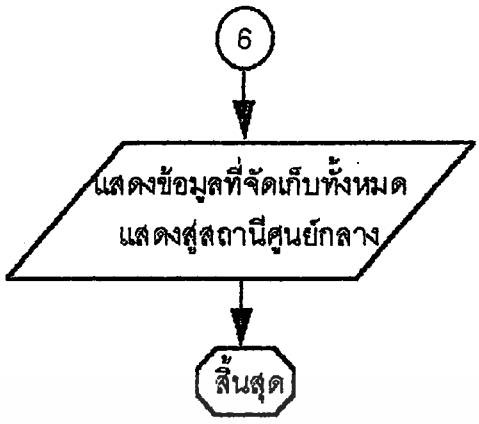


เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูป 3.22 แผนผังแสดงลำดับการทำงานของตัวอ่านค่าสัญญาณและจัดเก็บข้อมูล

จัมเปอร์	ฟังก์ชัน
	ความเร็ว 1200 บิตต่อวินาที
	ความเร็ว 4800 บิตต่อวินาที
	ความเร็ว 9600 บิตต่อวินาที
	ความเร็ว 19200 บิตต่อวินาที

จัมเปอร์เลือกความเร็วในการส่งข้อมูลอนุกรม (P1.0 , P1.1)

ตาราง 3.5 จัมเปอร์ของตัวอ่านค่าสัญญาณและจัดเก็บข้อมูล

แผนผังแสดงลำดับการทำงานของตัวอ่านค่าสัญญาณและจัดเก็บข้อมูลแสดงดัง รูป 3.22 และลักษณะจัมเปอร์เลือกความเร็วในการสื่อสารข้อมูลจะแสดงดังตาราง 3.5

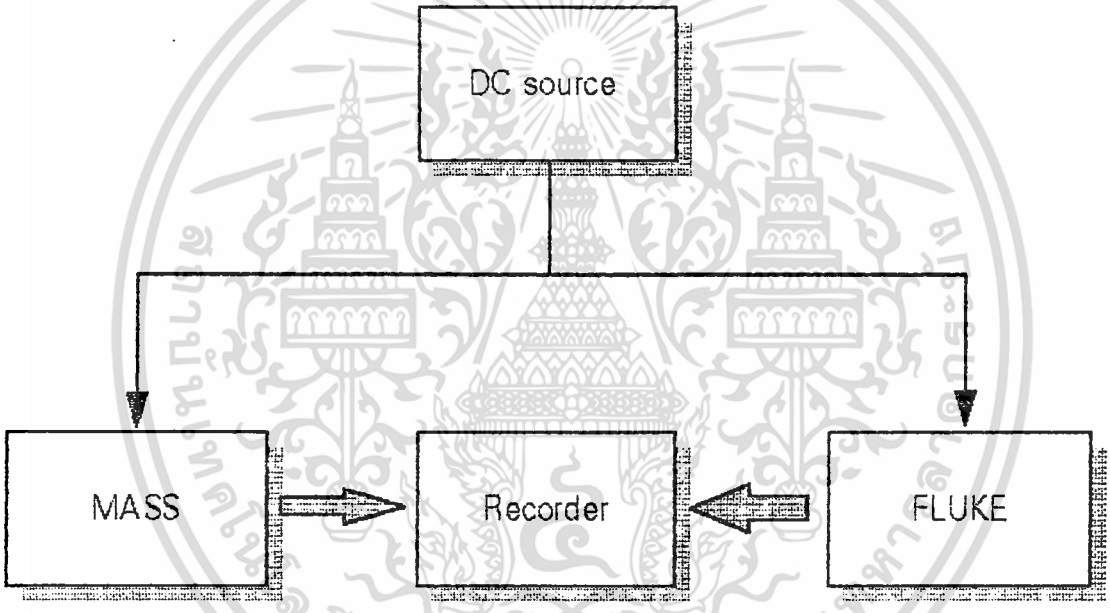
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 4

การทดลองและผลการทดลอง

4.1 การทดสอบความถูกต้องของการแปลงข้อมูล

จุดประสงค์การทดลองเพื่อทดสอบความถูกต้องของการแปลงข้อมูลโดยการเปรียบเทียบผลลัพธ์ในการแปลงข้อมูลของตัวส่งสัญญาณอนาลอกขนาดจิ๋วเทียบกับเครื่องวัดศักดาไฟฟ้าของ FLUKE ซึ่งลักษณะไดอะแกรมของการทดลองจะเป็นดังรูป 4.1



รูป 4.1 ไดอะแกรมของการทดลองความถูกต้องของการแปลงข้อมูล

โดยที่ผลจากการทดลองเป็นดังตาราง 4.1

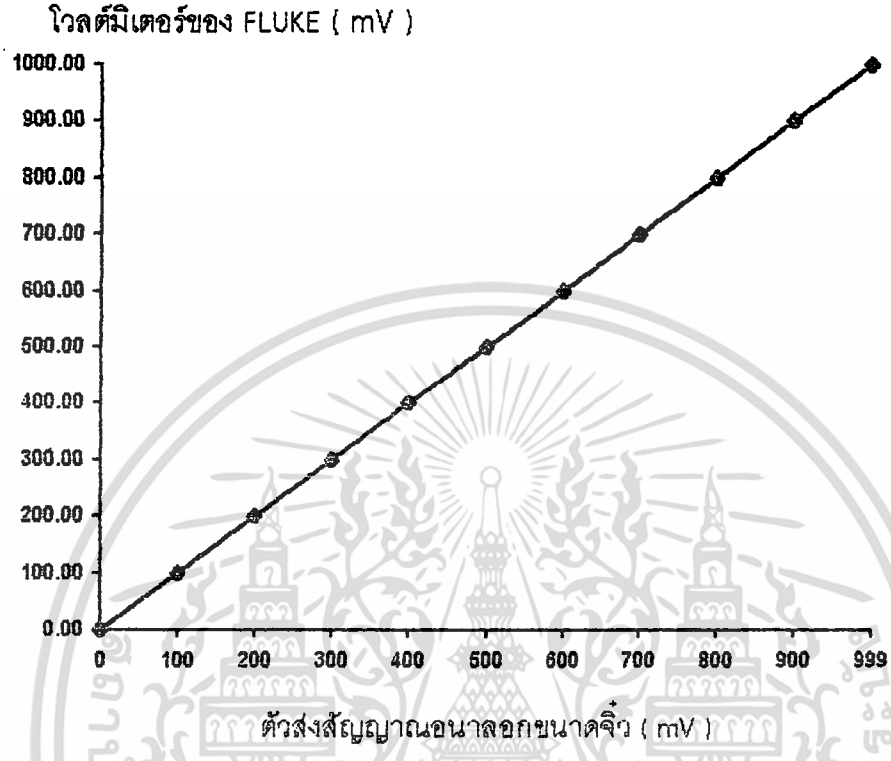
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตัวส่งสัญญาณอนาลอก (mV)	FLUKE (mV)
0	0.00
50	50.11
100	100.13
150	150.12
200	200.13
250	250.11
300	300.11
350	350.10
400	400.11
450	450.12
500	500.11
550	550.10
600	600.12
650	650.13
700	700.12
750	750.12
800	800.11
850	850.10
900	900.10
950	950.10
999	1000.00

ตาราง 4.1 ผลการทดลองความถูกต้องของการแปลงข้อมูล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากผลของการทดลองความถูกต้องของการแปลงข้อมูลสามารถแสดงเป็นกราฟได้ดังรูป 4.2



รูป 4 2 กราฟการทดสอบความถูกต้องของการแปลงข้อมูล

ซึ่งผลการทดลองจะได้ว่าความชันของการแปลงสัญญาณจะเป็นเส้นตรง จากสมการเส้นตรง

$$Y = mX + b$$

b = 0 และ m = 1000.00/999 จะได้

$$Y = 1.0010\,X$$

ซึ่งเปอร์เซ็นต์ความผิดพลาดจะเป็น

$$\frac{1.0010 - 1.0000}{1.0000} \times 100 = 0.1\%$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

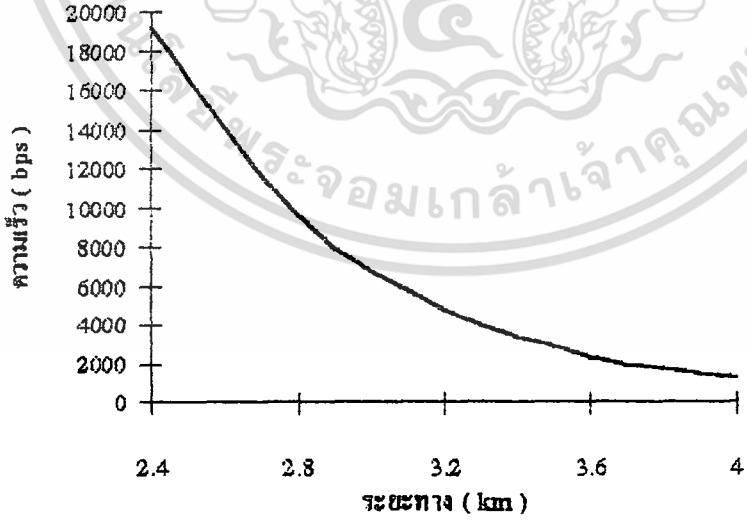
4.2 ความสามารถในการสื่อสารแบบลูปกระแส

จุดประสงค์การทดลองเพื่อทดสอบความสามารถในการสื่อสารแบบลูปกระแส โดยที่จะทำการเปลี่ยนค่าความเร็วในการสื่อสารข้อมูลอนุกรมเป็นค่าความเร็วต่าง ๆ แล้วหาระยะทางไกลที่สุดที่สามารถสื่อสารข้อมูลได้อย่างถูกต้องโดยการตรวจสอบความถูกต้องของข้อมูล (checksum) โดยที่ผลของการทดลองจะแสดงดังตาราง 4.2

ความเร็วในการสื่อสาร (bps)	ระยะทาง (km)
1200	4.0
2400	3.6
4800	3.2
9600	2.8
19200	2.4

ตาราง 4.2 ผลการทดลองความสามารถในการสื่อสารแบบลูปกระแส

จากผลของการทดลองความสามารถในการสื่อสารแบบลูปกระแสสามารถแสดงเป็นกราฟได้ดังรูป 4.3



รูป 4.3 กราฟการทดสอบความสามารถในการสื่อสารแบบลูปกระแส

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การทดสอบความสามารถในการสื่อสารแบบลูปกระแสแสดงดังรูป 4.4

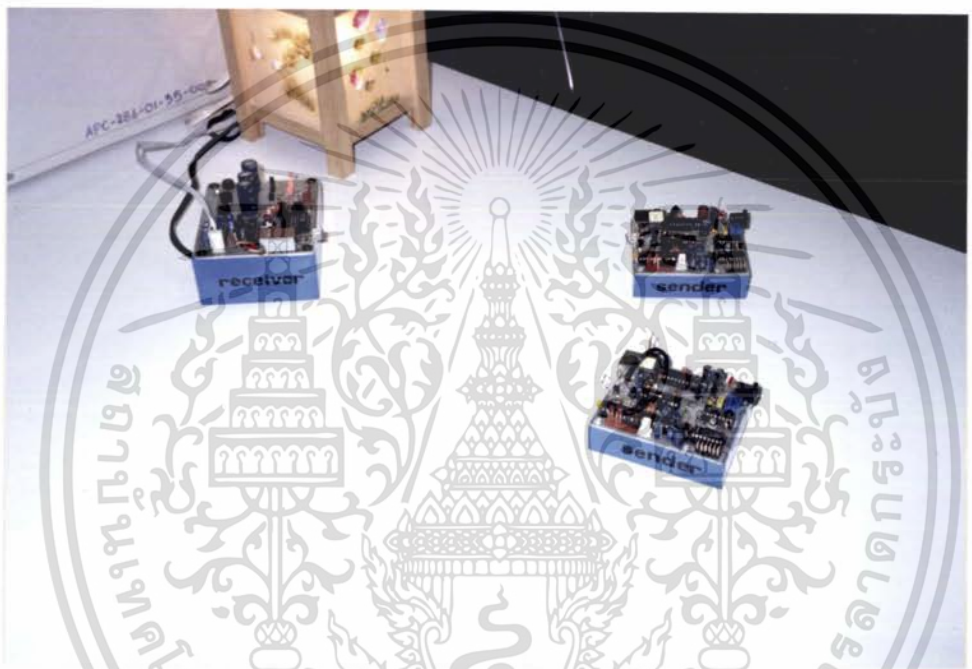
รูป 4.4 การทดสอบความสามารถในการสื่อสารแบบลูปกระแส

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.3 ความสามารถในการสื่อสารอินฟราเรด

จุดประสงค์การทดลองเพื่อทดสอบความสามารถในการสื่อสารแบบอินฟราเรด โดยที่จะใช้ค่าความเร็วในการสื่อสารข้อมูลอนุกรมเท่ากับ 1200 บิตต่อวินาที แล้วหาระยะทางไกลที่สุดและมุมที่สามารถทำการเบี่ยงเบนได้มากที่สุดที่ยังสามารถสื่อสารข้อมูลได้อย่างถูกต้อง โดยการตรวจสอบความถูกต้องของข้อมูล (checksum) โดยที่ผลของการทดลองพบว่าสามารถส่งข้อมูลได้ภายในระยะทาง 6 เมตร และมุมเบี่ยงเบนในช่วง ± 45 องศา

การทดสอบความสามารถในการสื่อสารอินฟราเรดแสดงดังรูป 4.5



รูป 4.5 การทดสอบความสามารถในการสื่อสารอินฟราเรด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 5
บทสรุป

5.1 สรุปและการประยุกต์ใช้งาน

โครงงานพิเศษตัวส่งสัญญาณอนาลอกขนาดจิ๋วเป็นโครงงานที่จัดทำขึ้นเพื่อใช้ในการวัดค่าสัญญาณอนาลอกและจะทำการส่งข้อมูลที่วัดได้นี้กลับมาเพื่อทำการประมวลผลหรือวิเคราะห์ผลจากงานวิจัยต่าง ๆ โดยจะสามารถเลือกใช้ช่องของการสื่อสารต่าง ๆ ตามความเหมาะสมของงานนั้น ๆ

ลักษณะการประยุกต์ใช้โครงงานตัวส่งสัญญาณอนาลอกขนาดจิ๋วสามารถนำไปใช้ในโรงงานอุตสาหกรรมซึ่งจะสามารถทำการวัดค่าสัญญาณอนาลอกต่าง ๆ ดังเช่น วัดค่าอุณหภูมิของเตาความร้อน , วัดค่า pH ของน้ำเสียในบ่อบำบัดน้ำเสียเป็นต้น ซึ่งจะทำให้เกิดความรวดเร็วในการวิเคราะห์ข้อมูลและความปลอดภัยแก่ผู้ทำการวัดโดยที่ไม่ต้องไปทำการวัดในบริเวณนั้นด้วยตนเองอีกทั้งยังทำให้ลักษณะการจัดเก็บข้อมูลเป็นไปได้อย่างเป็นระเบียบและง่ายดายขึ้นอีกด้วย ส่วนการประยุกต์ใช้โครงงานตัวส่งสัญญาณอนาลอกขนาดจิ๋วในด้านงานวิจัยทางฟิสิกส์ก็ยังสามารถทำได้ ดังเช่น การวัดค่าการนำความร้อน (Thermal Conductivity) ของโลหะต่าง ๆ โดยจะใช้ตัวส่งสัญญาณอนาลอกขนาดจิ๋วในการวัดค่าอุณหภูมิของโลหะนั้น ๆ ที่ตำแหน่งต่าง ๆ และส่งค่าที่วัดได้มาทำการวิเคราะห์เพื่อหาค่าการนำความร้อนต่อไป เป็นต้น

5.2 แนวทางในการพัฒนาโครงงานต่อไป

1 เพิ่มประสิทธิภาพในการสื่อสารอินฟราเรดให้สื่อสารได้ไกลมากขึ้นและด้วยความเร็วที่สูงขึ้น

2 เพิ่มประสิทธิภาพในการแปลงข้อมูลของตัวส่งสัญญาณอนาลอกขนาดจิ๋วโดยการเพิ่มดิจิตอลฟิลเตอร์

3 ลดขนาดของตัวส่งสัญญาณอนาลอกขนาดจิ๋วเพื่อให้เกิดความคล่องตัวในการใช้งานโดยการใช้เทคโนโลยี Surface Mounted.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก ก

โปรแกรมการทำงาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โปรแกรมการทำงาน

ตัวส่งสัญญาณอนาลอกขนาดจิ๋ว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
;                          SENDER ADDRESS " 01 "

F_ADD       EQU     30H          ; Fisrt asoii address of SENDER " 0 "
S_ADD       EQU     31H          ; Seoond asoii address of SENDER " 1 "
CHE_SUM     EQU     4BH          ; Byte oheok sum
F_CEN       EQU     50H          ; First byte address of CENTER
S_CEN       EQU     51H          ; Seoond byte address of CENTER
SOH         EQU     01H          ; Asoii " SOH "
EOT         EQU     04H          ; Asoii " EOT "
MIN         EQU     2FH          ; Asoii " / "
CR          EQU     0DH          ; Asoii Return
LF          EQU     0AH          ; Asoii Line Feed
COMMA       EQU     2CH          ; Asoii Comma " , "
NEGA        EQU     2DH          ; Asoii " - " use in negative
OVER        EQU     45H          ; Asoii " E " use in overflow
BAUD_12     EQU     0E8H         ; Baud rate 1200 bps
BAUD_24     EQU     0F4H         ; Baud rate 2400 bps
BAUD_96     EQU     0FDH         ; Baud rate 9600 bps
STRACK      EQU     60H          ; Address STRACK POINTER
BUFF1       EQU     30H          ; First Buffer read data
BUFF2       EQU     31H          ; Seoond Buffer read data
BUFF3       EQU     32H          ; Third Buffer read data
BUFFP       EQU     33H          ; Buffer PCON
ASI_T       EQU     't'          ; Asoii " t "

            ORG     0000H
            LJMP    MAIN

            ORG     0030H
;                          Initial Serial port
;           initial serial port boud rate 1200,4800,9600,19200 bps
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
INTI:       MOV     TMOD,#00100101B     ; Initial serial port
            MOV     SCON,#01010010B
            MOV     PCON,#00000000B
            MOV     TH1,#BAUD_12        ; 1200 bps
            JB      P3.4,INT_1
            MOV     TH1,#BAUD_24
            MOV     PCON,#10000000B     ; 4800 bps
            SJMP    INT_3
INT_1:      JB      P3.5,INT_2
            MOV     TH1,#BAUD_96        ; 9600 bps
            SJMP    INT_2
INT_3:      JB      P3.5,INT_2
            MOV     TH1,#BAUD_96        ; 19200 bps
            MOV     PCON,#10000000B
INT_2:      SETB    TR1
            RET
;                       Check Key REAL TIME
;           Check key (P3.7) : HI = transmit
;           LO = Real Time

CHECK_KEY:  MOV     S_CEN,#00H
            MOV     F_CEN,#00H
            JNB     P3.7,CHK_1
            CLR     F0
            SJMP    RE_T
CHK_1:      SETB    F0
RE_T:       RET
;                       Send data of Serial port
;           input register A  data to send

SEND:       PUSH  07H
            JNB   TI,$                  ; Send data to serial port
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        CLR     TI
        MOV     SBUF,A
        MOV     R7,#10H
        DJNZ    R7,$
        POP     07H
        RET
;                       Receive data of Serial port
;       output register A data to receive

RECIVE: JNB     RI,$                    ; Receive data of serial port
        CLR     RI
        MOV     A,SBUF
        ACALL   FILLTER
        RET
;                       Fillter Data

FILLTER: CJNE   A,#ASI_T,FILL_1
        RET
FILL_1: CJNE    A,#MIN,FILL_2
        RET
FILL_2: CJNE    A,#CR,FILL_3
        RET
FILL_3: CLR     C
        CJNE    A,#30H,FILL_4
FILL_4: JNC     FILL_5
        POP     5FH
        POP     5FH
        SJMP    RECIVE
FILL_5: CLR     C
        CJNE    A,#3AH,FILL_6
FILL_6: JC      FILL_7
        POP     5FH
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                POP     5FH
                SJMP    RECIVE
FILL_7:         RET
;                                   Send COMMA

SD_COM:         MOV     A,#COMMA        ; Send " , "
                ACALL   SEND
                ADD     A,CHE_SUM       ; Check Sum
                MOV     CHE_SUM,A
                RET
;                                   send CR & LF

SD_CR_LF:       MOV     A,#CR           ; Send Return
                ACALL   SEND
                MOV     A,#LF           ; Send Line Feed
                ACALL   SEND
                RET
;                                   Binary to Ascii
;               Convert Binary to Aseii code
;               entry  A
;               exit   A

BIN_ASCII:      PUSH    07H
                ANL     A,#0FH
                MOV     R7,A
                CLR     C
                SUBB    A,#0AH
                JNC     ASCII_AF2
                MOV     A,R7
                ADD     A,#30H
                POP     07H
                RET
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
ASCII_AF2:  MOV     A,R7
            ADD     A,#37H
            POP     07H
            RET
;                           Byte (2 digit) to Ascii
;           Convert A to asoii and SEND

BYTE_ASCII: MOV     B,A
            ACALL   BIN_ASCII
            ACALL   SEND
            MOV     A,B
            SWAP    A
            ACALL   BIN_ASCII
            ACALL   SEND
            RET
;                           Send bottom of package
;           send  Cheok Sum & send EOT
;           Cheok sum send by binary type

SD_BOTTOM:  MOV     A,CHE_SUM       ; Send oheok sum
            ACALL   BYTE_ASCII
            MOV     A,#EOT          ; " EOT "
            ACALL   SEND
            RET
;                       Check OVERFLOW & NEGATIVE
;           move overflow data to 'E' & move negative data to '-'
;           input register A      output register A

CH_READ:    CJNE    A,#3AH,RE1      ; Check OVERFLOW & NEGATIVE data
            MOV     A,#NEGA         ; " - "
            SJMP    RE2
RE1:        CJNE    A,#3BH,RE2
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                MOV     A,#OVER             ; " E "
RE2:            RET
;                               Send Ascii of data
;               data to convert is LOW byte & HIGH byte must 0000____H
;               input data to send in register A
;               & check OVERFLOW and NEGATIVE

SEND_ASC:       ADD     A,#30H
                ACALL   CH_READ             ; Check overflow & negative
                ACALL   SEND
                RET
;                               Send data 8 Channels
;               Send data of RAM 33H - 4AH

SEDATA:         PUSH    01H
                MOV     R0,#32H             ; Initial counter to out data
                MOV     R1,#00H
seda_1:         INC     R0
                MOV     A,@R0
                ACALL   SEND_ASC
                ADD     A,CHE_SUM           ; Check sum
                MOV     CHE_SUM,A
                INC     R1
                CJNE    R1,#03H,SEDA_2
                MOV     R1,#00H
                ACALL   SD_COM
SEDA_2:         CJNE    R0,#4AH,SEDA_1      ; Check Final data
                POP     01H
                RET
;                               Send Header of data package
;  Format SOH F_CEN S_CEN < F_ADD S_ADD , Data 8 channels , Check sum  EOT
;          Check sum start at F_CEN end at comma " , " before Data 8 channels
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
SD_HEAD:    MOV     A,#SOH          ; " SOH "
            ACALL   SEND
            MOV     A,F_CEN         ; First address of CENTER
            ACALL   SEND
            MOV     CHE_SUM,A       ; Initial Check sum
            MOV     A,S_CEN         ; Second address of CENTER
            ACALL   SEND
            ADD     A,CHE_SUM       ; Check sum
            MOV     CHE_SUM,A
            MOV     A,#MIN          ; " / "
            ACALL   SEND
            ADD     A,CHE_SUM       ; Check sum
            MOV     CHE_SUM,A
            MOV     A,#F_ADD        ; First address of SENDER
            ACALL   SEND
            ADD     A,CHE_SUM       ; Check sum
            MOV     CHE_SUM,A
            MOV     A,#S_ADD        ; Second address of SENDER
            ACALL   SEND
            ADD     A,CHE_SUM       ; Check sum
            MOV     CHE_SUM,A
            RET
;                       Send data to Center

SEND_DATA:  ACALL   SD_HEAD         ; Send Header of package
            ACALL   SD_COM          ; Send Asoii COMMA " , "
            ACALL   SEDATA          ; Send data 8 channels
            ACALL   SD_BOTTOM       ; Send Bottom of package
            ACALL   SD_CR_LF        ; Send CR & Send LF
            RET
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
;                          Delay 4 ms

DELAY:  PUSH    05H
        PUSH    06H
        MOV     R6,#07H
LOOP:   MOV     R5,#0F0H
        DJNZ    R5,$
        DJNZ    R6,LOOP
        POP     06H
        POP     05H
        RET
;                       Read data to get to Buffer
;       read data A/D receive in 30H , 31H , 32H

READ:   JNB     P1.3,$          ; Check bit DRIVER of A/D
        JB      P1.3,$
        MOV     R6,#0AH         ; Delay to read data first data
        DJNZ    R6,$
        MOV     A,P1            ; Read first data
        SWAP    A
        ANL     A,#0FH
        MOV     BUFF1,A         ; BUFFER 1
        ACALL   DELAY           ; Delay 4 ms
        MOV     A,P1            ; Read second data
        SWAP    A
        ANL     A,#0FH
        MOV     BUFF3,A         ; BUFFER 2
        ACALL   DELAY           ; Delay 4 ms
        MOV     A,P1            ; Read third data
        SWAP    A
        ANL     A,#0FH
        MOV     BUFF2,A         ; BUFFER 3
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        RET
;                   Read data & Get data of A/D to RAM
;       input counter by register R0
;       Get data 8 chennal in 33H - 4AH

DE_READ: PUSH   07H
         MOV    R7,#03H          ; Delay to read data of A/D 12 ms
CH:      ACALL  DELAY            ; Delay 4 ms
         DJNZ   R7,CH
         ACALL  READ             ; Read data Multiplexer
         INC    R0               ; Inerese buffer to receive
         MOV    @R0,BUFF1        ; Get data first byte
         INC    R0
         MOV    @R0,BUFF2        ; Get data second byte
         INC    R0
         MOV    @R0,BUFF3        ; Get data third byte
         POP    07H
         RET
;                   Scan Multiplexer
;       Scan & Initial counter to get data in RAM 33H - 4AH

SCAN:    MOV    R0,#32H          ; Initial buffer to receive 32H to 4AH
         MOV    P1,#0F8H         ; Channel 1
         ACALL  DE_READ
         MOV    P1,#0F9H         ; Channel 2
         ACALL  DE_READ
         MOV    P1,#0FAH         ; Channel 3
         ACALL  DE_READ
         MOV    P1,#0FBH         ; Channel 4
         ACALL  DE_READ
         MOV    P1,#0FCH         ; Channel 5
         ACALL  DE_READ
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                MOV     P1,#0FDH            ; Channel 6
                ACALL   DE_READ
                MOV     P1,#0FEH            ; Channel 7
                ACALL   DE_READ
                MOV     P1,#0FFH            ; Channel 8
                ACALL   DE_READ
                MOV     P1,#0F8H            ; Channel 1
                RET
;                       Check data receive of CENTER
;               Format: SOH F_ADD S_ADD < F_CEN S_CEN
;               no check sum

CHECK:          ACALL   RECIVE
                CJNE    A,#ASI_T,CHECK      ; " t "
                MOV     IE,#00H             ; Clear Interrupt Serial Port
                ACALL   RECIVE
                CJNE    A,#F_ADD,CHECK      ; Check first address of sender
                ACALL   RECIVE
                CJNE    A,#S_ADD,CHECK      ; Check second address of sender
                ACALL   RECIVE
                CJNE    A,#MIN,CHECK        ; " / "
                ACALL   RECIVE
                MOV     F_CEN,A             ; Get first address of CENTER
                ACALL   RECIVE
                MOV     S_CEN,A             ; Get second address of CENTER
                MOV     A,F_CEN             ; Check first address of SENDER will-
                CJNE    A,#F_ADD,EXIT       ; not equa first address of CENTER
                MOV     A,S_CEN             ; Check second address of SENDER will-
                CJNE    A,#S_ADD,EXIT       ; not equa second address of CENTER
                SJMP    CHECK
EXIT:           ACALL RECIVE                ; " Return "
                CJNE    A,#CR,CHECK
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                RET
;                                   Main Program

MAIN:           MOV     R0,#00H             ; Power up delay
                DJNZ    R0,$
                MOV     SP,#STRACK          ; Set strack pointer
                ACALL   CHECK_KEY           ; Check key Real Time
                ACALL   INTI                ; Initial Serial port
SUMAIN:         JB      F0,SUB_1            ; Check bit to REAL TIME
                ACALL   CHECK               ; Check data of Center
SUB_1:          ACALL   SCAN                ; Scan data of Multiplexer
                ACALL   SEND_DATA           ; Send data to Center
                SJMP    SUMAIN
                END
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โปรแกรมการทำงาน
ตัวรับและแปลงสัญญาณ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
;                                   RECEIVER

SOH         EQU     01H             ; Asoii " SOH "
EOT         EQU     04H             ; Asoii " EOT "
MIN         EQU     2FH             ; Asoii " / "
CR          EQU     0DH             ; Asoii Return
LF          EQU     0AH             ; Asoii Line Feed
COMMA       EQU     2CH             ; Asoii Comma " , "
NEGA        EQU     2DH             ; Asoii " - " use in negative
OVER        EQU     45H             ; Asoii " E " use in overflow
BAUD_12     EQU     0E8H            ; Boud rate 1200 bps
BAUD_24     EQU     0F4H            ; Baud rate 2400 bps
BAUD_96     EQU     0FDH            ; Baud rate 9600 bps
STRACK      EQU     60H             ; Address STRACK POINTER
CHECKSUM    EQU     50H
BUFFSUM     EQU     51H

            ORG     0000H
            LJMP    MAIN

            ORG     0030H
;                                   Initial Serial port
;           initial serial port boud rate 1200,9600,19200 bps

INTI:       MOV     TMOD,#00100010B ; Initial serial port
            MOV     SCON,#01010010B
            MOV     TH1,#BAUD_12    ; 1200 bps
            JB      P1.7,INT_1
            MOV     TH1,#BAUD_24
            MOV     PCON,#10000000B ; 4800 bps
            SJMP    INT_3
INT_1:      JB      P1.6,INT_2
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                MOV     TH1,#BAUD_96            ; 9600 bps
                SJMP    INT_2
INT_3:          JB      P1.6,INT_2
                MOV     TH1,#BAUD_96            ; 19200 bps
                MOV     PCON,#10000000B
INT_2:          SETB    TR1
                RET
;                               Send data of Serial port
;               input register A  data to send

SEND:           JNB     TI,$                    ; Send data to serial port
                CLR     TI
                MOV     SBUF,A
                RET
;                               Receive data of Serial port
;               output register A data to receive

RECIVE:         JNB     RI,$                    ; Receive data of serial port
                CLR     RI
                MOV     A,SBUF
                JNB     F0,NO_FILL
                ACALL   FILLTER
                XCH     A,CHECKSUM
                ADD     A,CHECKSUM
                XCH     A,CHECKSUM
NO_FILL:        RET
FILLTER:        CJNE    A,#00H,FILL
                RET
FILL:           CJNE    A,#LF,FILL_1
                RET
FILL_1:         CJNE    A,#MIN,FILL_2
                RET
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
FILL_2:     CJNE    A,#CR,FILL_3
            RET
FILL_3:     CJNE    A,#COMMA,FILL_4
            RET
FILL_4:     CJNE    A,#SOH,FILL_5
            RET
FILL_5:     CJNE    A,#EOT,FILL_6
            RET
FILL_6:     CJNE    A,#NEGA,FILL_7
            RET
FILL_7:     CJNE    A,#OVER,FILL_8
FILL_8:     CLR     C
            CJNE    A,#30H,FILL_9
FILL_9:     JNC     FILL_10
            POP     5FH
            POP     5FH
            SJMP    RECIVE
FILL_10:    CLR     C
            CJNE    A,#3AH,FILL_11
FILL_11:    JC      FILL_12
            CJNE    A,#47H,FILL_13
FILL_13:    JC      FILL_14
            POP     5FH
            POP     5FH
            SJMP    RECIVE
FILL_14:    CJNE    A,#41H,FILL_15
FILL_15:    JNC     FILL_12
            POP     5FH
            POP     5FH
            SJMP    RECIVE
FILL_12:    RET
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
;                    Send 2 Byte & Recive 2 Byte
;          Data in Register A to Serial port (2 byte)
;          Serial port to Register A (2 byte)

;                         Send Byte
;          SEND 2 BYTE HEX FROM A
;          IN  = A
;          REG = A,R2,R3

SBYTE:     LCALL    HTOA
           MOV      A,R2
           LCALL    SEND
           MOV      A,R3
           LCALL    SEND
           RET

;                         Recive Byte
;          READ HEX-ASCII FROM SERIAL (2 BYTE)
;          OUT = A
;          REG = A,R2,R3

RBYTE:     LCALL    RECIVE
           MOV      R2,A
           LCALL    RECIVE
           MOV      R3,A
           LCALL    ATOH
           RET

;                         Htoa Sub
;          CONVERT HEX TO ASCII
;          IN  = A
;          OUT = R2,R3
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
;           REG = A,R2,R3

HTOA:       PUSH    ACC
            SWAP    A
            LCALL   HTOAS
            MOV     R2,A
            POP     ACC
            LCALL   HTOAS
            MOV     R3,A
            RET
HTOAS:      ANL     A,#0FH
            CJNE    A,#0AH,$+3
            JNC     HTOAS1
            ORL     A,#30H
            RET
HTOAS1:     SUBB    A,#9
            ORL     A,#40H
            RET

;                       Atoh Sub
;           ASCII TO HEX CONVERT
;           IN  = R2,R3
;           OUT = A
;           REG = A,R2

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

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

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

;                                   Send Data

SEN_DATA:       MOV     R0,#30H
                MOV     A,#SOH
                ACALL   SEND
                MOV     A,@R0
                INC     R0
                ACALL   SEND
                MOV     A,@R0
                INC     R0
                ACALL   SEND
                MOV     A,#MIN
                ACALL   SEND
                MOV     A,@R0
                INC     R0
                ACALL   SEND
                MOV     A,@R0
                INC     R0
                ACALL   SEND
                MOV     A,#COMMA
                ACALL   SEND
                MOV     R1,#08H
LOOP_SEN:       MOV     A,@R0
                INC     R0
                ACALL   SEND
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        MOV     A,@R0
        INC     R0
        ACALL   SEND
        MOV     A,@R0
        INC     R0
        ACALL   SEND
        MOV     A,#COMMA
        ACALL   SEND
        DJNZ    R1,LOOP_SEN
        MOV     A,@R0
        INC     R0
        ACALL   SBYTE
        MOV     A,#EOT
        ACALL   SEND
        MOV     A,#CR
        ACALL   SEND
        MOV     A,#LF
        ACALL   SEND
        RET

;                       Receive Data

CHECK:  MOV     R0,#30H
        ACALL   RECIVE
        CJNE    A,#SOH,ERROR
        MOV     CHECKSUM,#00H
        ACALL   RECIVE
        MOV     @R0,A
        INC     R0
        ACALL   RECIVE
        MOV     @R0,A
        INC     R0
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        ACALL   RECIVE
        CJNE    A,#MIN,ERROR
        ACALL   RECIVE
        MOV     @R0,A
        INC     R0
        ACALL   RECIVE
        MOV     @R0,A
        INC     R0
        ACALL   RECIVE
        CJNE    A,#COMMA,ERROR
        MOV     R1,#08H
LOOP_CH: ACALL  RECIVE
        MOV     @R0,A
        INC     R0
        ACALL   RECIVE
        MOV     @R0,A
        INC     R0
        ACALL   RECIVE
        MOV     @R0,A
        INC     R0
        ACALL   RECIVE
        CJNE    A,#COMMA,ERROR
        DJNZ    R1,LOOP_CH
        MOV     A,CHECKSUM
        SWAP    A
        MOV     BUFFSUM,A
        ACALL   RBYTE
        CJNE    A,BUFFSUM,ERROR
        MOV     @R0,A
        ACALL   RECIVE
        CJNE    A,#EOT,ERROR
        ACALL   SEN_DATA
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
ERROR:      RET

;                              Main Program

MAIN:       MOV     R0,#00H          ; Power up delay
            DJNZ    R0,$
            MOV     SP,#STRACK       ; Set strack pointer
            ACALL   INTI             ; Initial Serial port
            JB      P1.5,IN_OUT
            JB      P1.4,SUMAIN
            SETB    F0
CHE:        ACALL   CHECK
            SJMP    CHE
SUMAIN:     SETB    F0
SUMAIN_1:   ACALL   RECIVE
            ACALL   SEND
            SJMP    SUMAIN_1
IN_OUT:     CLR     F0
IN_OUT_1:   ACALL   RECIVE
            ACALL   SEND
            SJMP    IN_OUT_1
            END
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โปรแกรมการทำงาน
ตัวอ่านค่าสัญญาณและจัดเก็บข้อมูล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
;                              SCANNER " 99 "

STRACK      EQU     60H             ; Address STRACK POINTER
BAUD_12     EQU     0E8H            ; Baud Rate 1200 bps
BAUD_24     EQU     0F4H            ; Baud Rate 2400 bps
BAUD_96     EQU     0FDH            ; Baud Rate 9600 bps

I_O         EQU     P1.4            ; I/O  of RTC
SCLK        EQU     P1.5            ; SCLK of RTC
RST         EQU     P1.6            ; RST  of RTC

R           EQU     'r'             ; Asoii "r"
W           EQU     'w'             ; Asoii "w"
S           EQU     's'             ; Asoii "s"
G           EQU     'g'             ; Asoii "g"
T           EQU     't'             ; Asoii "t"
MIN         EQU     '/'             ; Asoii "/"
COLON       EQU     ':'             ; Asoii ":"
SPACE       EQU     ' '             ; Asoii " "
COMMA       EQU     ','             ; Asoii ","
NEG         EQU     '-'             ; Asoii "-"
CR          EQU     0DH             ; Asoii RETURN
LF          EQU     0AH             ; Asoii LINE FEED
SOH         EQU     01H
EOT         EQU     04H

A_DATE      EQU     30H             ; Address of DATE
A_MONTH     EQU     31H             ; Address of MONTH
A_YEAR      EQU     32H             ; Address of YEAR
A_HOUR      EQU     33H             ; Address of HOUR
A_MINUTE    EQU     34H             ; Address of MINUTE
A_SECOND    EQU     35H             ; Address of SECOND
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
AS_HOUR     EQU     36H             ; Address of Scan HOUR
AS_MINUTE   EQU     37H             ; Address of Scan MINUTE
AS_SECOND   EQU     38H             ; Address of Scan SECOND

B_HOUR      EQU     39H             ; Address of Buffer HOUR
B_MINUTE    EQU     3AH             ; Address of Buffer MINUTE
B_SECOND    EQU     3BH             ; Address of Buffer SECOND

NUMBER_ADD EQU      3CH             ; Address of number of address sender
START_ADD   EQU     3DH

ADD_SCAN    EQU     99H             ; ADDRESS OF SCANNER

R_DATE      EQU     87H             ; Command Read DATE
R_MONTH     EQU     89H             ; Command Read MONTH
R_YEAR      EQU     8DH             ; Command Read YEAR
R_HOUR      EQU     85H             ; Command Read HOUR
R_MINUTE    EQU     83H             ; Command Read MINUTE
R_SECOND    EQU     81H             ; Command Read SECOND

W_DATE      EQU     86H             ; Command Write DATE
W_MONTH     EQU     88H             ; Command Write MONTH
W_YEAR      EQU     8CH             ; Command Write YEAR
W_HOUR      EQU     84H             ; Command Write DATE
W_MINUTE    EQU     82H             ; Command Write MONTH
W_SECOND    EQU     80H             ; Command Write YEAR
W_PROT      EQU     8EH             ; Command Write Protect
BUF_ADD     EQU     20H
CHECKSUM    EQU     21H             ; Address CHECK SUM
CHECK_S     EQU     22H             ; Address to compare
FLAG_RAM    EQU     30H             ; Bit address of RAM HIGH
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
FLAG_STA   EQU     31H                 ; Bit address status of P1.7

           ORG     0000H
           LJMP    MAIN

           ORG     0030H
;                          Initial Serial port
;          initial serial port boud rate 1200,4800,9600,19200 bps

INTI:      MOV     TMOD,#00100010B     ; Initial serial port
           MOV     SCON,#01010010B
           MOV     TH1,#BAUD_12        ; 1200 bps
           JB      P1.0,INT_1
           MOV     TH1,#BAUD_24
           MOV     PCON,#10000000B     ; 4800 bps
           SJMP    INT_3
INT_1:     JB      P1.1,INT_2
           MOV     TH1,#BAUD_96        ; 9600 bps
           SJMP    INT_2
INT_3:     JB      P1.1,INT_2
           MOV     TH1,#BAUD_96        ; 19200 bps
           MOV     PCON,#10000000B
INT_2:     SETB    TR1
           RET
;                          Send data of Serial port
;          input register A  data to send

SEND:      JNB     TI,$                ; Send data to serial port
           CLR     TI
           MOV     SBUF,A
           XCH     A,CHECKSUM
           ADD     A,CHECKSUM
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                XCH     A,CHECKSUM
                RET
;                           Receive data of Serial port
;               output register A data to receive

RECIVE:         JNB     RI,$                ; Receive data of serial port
                CLR     RI
                MOV     A,SBUF
                XCH     A,CHECKSUM
                ADD     A,CHECKSUM
                XCH     A,CHECKSUM
                RET
;                           Receive with TIME OUT

RECIVE_T:       SETB    RS0
                SETB    RS1
                MOV     R7,#00H
AGAIN:          MOV     R6,#00H
GET_RI:         JB      RI,READY            ; Receive data of serial port
                DJNZ    R6,GET_RI
                DJNZ    R7,AGAIN
                MOV     A,#0FFH
                CLR     RS0
                RET
READY:          CLR     RI
                MOV     A,SBUF
                CLR     RS0
                CLR     RS1
                RET
;                           Delay REAL TIME CLOCK

DELAY_RTC:      PUSH    00H
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                MOV     R0,#05H
                DJNZ    R0,$
                POP     00H
                RET
;                               Clock to Write Data
;               clock in pin SCLK

SCLKW:          CLR     SCLK
                ACALL   DELAY_RTC
                SETB    SCLK
                ACALL   DELAY_RTC
                RET
;                               Clock to Read Data
;               clock in pin SCLK

SCLKR:          SETB    SCLK
                ACALL   DELAY_RTC
                CLR     SCLK
                ACALL   DELAY_RTC
                RET
;                               Read REAL TIME CLOCK
;               read data from REAL TIME CLOCK 6 byte dd/mm/yy hh:mm:ss
;               R6  keep Command word
;               R7  keep Data

READ_RTC:       CLR     RST
                CLR     SCLK
                SETB    RST
                ACALL   DELAY_RTC
                MOV     B,#08H
                CLR     C
BYTERD1:        MOV     A,R6
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
            RRC     A
            MOV     R6,A
            MOV     I_O,C
            ACALL   SCLKW
            DJNZ    B,BYTERD1
            MOV     B,#08H
            MOV     R7,#00H
BYTERD2:    ACALL   SCLKR
            MOV     A,R7
            MOV     C,I_O
            RRC     A
            MOV     R7,A
            DJNZ    B,BYTERD2
            CLR     RST
            ACALL   DELAY_RTC
            RET
;                   Write REAL TIME CLOCK
;           write data to REAL TIME CLOCK 6 byte  dd/mm/yy hh:mm:ss
;           R6  keep Command word
;           R7  keep Data

WRITE_RTC:  CLR     RST
            CLR     SCLK
            SETB    RST
            ACALL   DELAY_RTC
            MOV     B,#08H
            CLR     C
BYTEWR1:    MOV     A,R6
            RRC     A
            MOV     R6,A
            MOV     I_O,C
            ACALL   SCLKW
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
            DJNZ    B,BYTEWR1
            MOV     B,#08H
            CLR     C
BYTEWR2:    MOV     A,R7
            RRC     A
            MOV     R7,A
            MOV     I_O,C
            ACALL   SCLKW
            DJNZ    B,BYTEWR2
            CLR     RST
            ACALL   DELAY_RTC
            RET
;                       Send CR & LF

SEND_CR_LF: MOV     A,#CR
            ACALL   SEND
            MOV     A,#LF
            ACALL   SEND
            RET
;                       Send  CR

SEND_CR:    MOV     A,#CR
            ACALL   SEND
            RET
;                       Send COMMA

SEND_COMMA: MOV     A,#COMMA
            ACALL   SEND
            RET
;                       Send COLON

SEND_COLON: MOV     A,#COLON
```

เอกสารนี้สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                ACALL   SEND
                RET
;                                   Send SPACE

SEND_SPACE: MOV     A,#SPACE
                ACALL SEND
                RET
;                                   Send MIN "/"

SEND_MIN:   MOV     A,#MIN
                ACALL   SEND
                RET
;                                   Send " t "

SEND_T:     MOV     A,#'t'
                ACALL   SEND
                RET
;                                   Read Time
;               Read Data from RTC

READ_TIME:  MOV     R6,#R_DATE          ; Command read DATE
                ACALL   READ_RTC
                MOV     A_DATE,R7           ; Keep data in Address of DATE
                MOV     R6,#R_MONTH         ; Command read MONTH
                ACALL   READ_RTC
                MOV     A_MONTH,R7          ; Keep data in Address of MONTH
                MOV     R6,#R_YEAR          ; Command read YEAR
                ACALL   READ_RTC
                MOV     A_YEAR,R7           ; Keep data in Address of YEAR
                MOV     R6,#R_HOUR          ; Command read HOUR
                ACALL   READ_RTC
                MOV     A_HOUR,R7           ; Keep data in Address of HOUR
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        MOV     R6,#R_MINUTE        ; Command read MINUTE
        ACALL   READ_RTC
        MOV     A_MINUTE,R7         ; Keep data in Address of MINUTE
        MOV     R6,#R_SECOND        ; Command read SECOND
        ACALL   READ_RTC
        MOV     A_SECOND,R7         ; Keep data in Address of SECOND
        RET
;                       Send 2 Byte & Recive 2 Byte
;       Data in Register A to Serial port (2 byte)
;       Serial port to Register A (2 byte)

;                           Send Byte
;       SEND 2 BYTE HEX FROM A
;       IN  = A
;       REG = A,R2,R3

SBYTE:  LCALL   HTOA
        MOV     A,R2
        LCALL   SEND
        MOV     A,R3
        LCALL   SEND
        RET

;                           Recive Byte
;       READ HEX-ASCII FROM SERIAL (2 BYTE)
;       OUT = A
;       REG = A,R2,R3

RBYTE:  LCALL   RECIVE
        MOV     R2,A
        LCALL   RECIVE
        MOV     R3,A
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                LCALL   ATOH
                RET

;                               Htoa Sub
;               CONVERT HEX TO ASCII
;               IN  = A
;               OUT = R2,R3
;               REG = A,R2,R3

HTOA:           PUSH    ACC
                SWAP    A
                LCALL   HTOAS
                MOV     R2,A
                POP     ACC
                LCALL   HTOAS
                MOV     R3,A
                RET
HTOAS:          ANL     A,#0FH
                CJNE    A,#0AH,$+3
                INC     HTOAS1
                ORL     A,#30H
                RET
HTOAS1:         SUBB    A,#9
                ORL     A,#40H
                RET

;                               Atoh Sub
;               ASCII TO HEX CONVERT
;               IN  = R2,R3
;               OUT = A
;               REG = A,R2
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
ATOH:       MOV     A,R2
            LCALL   ATOHS
            SWAP    A
            MOV     R2,A
            MOV     A,R3
            LCALL   ATOHS
            ORL     A,R2
            RET
ATOHS:      CJNE    A,#'A',$+3
            JC      ATOHS1
            ADD     A,#9
ATOHS1:     ANL     A,#0FH
            RET
;                       Send Time
;           Send time to Central Station

SEND_TIME:  ACALL   SEND_CR_LF
            MOV     A,A_DATE
            ACALL   SBYTE
            ACALL   SEND_MIN
            MOV     A,A_MONTH
            ACALL   SBYTE
            ACALL   SEND_MIN
            MOV     A,A_YEAR
            ACALL   SBYTE
            ACALL   SEND_SPACE
            MOV     A,A_HOUR
            ACALL   SBYTE
            ACALL   SEND_COLON
            MOV     A,A_MINUTE
            ACALL   SBYTE
            ACALL   SEND_COLON
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                MOV     A,A_SECOND
                ACALL   SBYTE
                RET
;                                   Write time

WRITE_TIME: MOV     R6,#W_PROT
                MOV     R7,#00H
                ACALL   WRITE_RTC
                MOV     R6,#W_DATE
                MOV     R7,A_DATE
                ACALL   WRITE_RTC
                MOV     R6,#W_MONTH
                MOV     R7,A_MONTH
                ACALL   WRITE_RTC
                MOV     R6,#W_YEAR
                MOV     R7,A_YEAR
                ACALL   WRITE_RTC
                MOV     R6,#W_HOUR
                MOV     R7,A_HOUR
                ACALL   WRITE_RTC
                MOV     R6,#W_MINUTE
                MOV     R7,A_MINUTE
                ACALL   WRITE_RTC
                MOV     R6,#W_SECOND
                MOV     R7,A_SECOND
                ACALL   WRITE_RTC
                MOV     R6,#W_PROT
                MOV     R7,#80H
                ACALL   WRITE_RTC
                RET
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
;                          Recive time
;          DD/MM/YY HH:MM:SS

REC_TI_DATE: ACALL  RECIVE
             CJNE   A,#SPACE,RE_WRITE_1
             ACALL  RBYTE
             MOV    A_DATE,A
             ACALL  RECIVE
             CJNE   A,#MIN,RE_WRITE_1
             ACALL  RBYTE
             MOV    A_MONTH,A
             ACALL  RECIVE
             CJNE   A,#MIN,RE_WRITE_1
             ACALL  RBYTE
             MOV    A_YEAR,A
RE_WRITE_1: RET
REC_TI_HOUR: ACALL  RECIVE
             CJNE   A,#SPACE,RE_WRITE_2
             ACALL  RBYTE
             MOV    A_HOUR,A
             ACALL  RECIVE
             CJNE   A,#COLON,RE_WRITE_2
             ACALL  RBYTE
             MOV    A_MINUTE,A
             ACALL  RECIVE
             CJNE   A,#COLON,RE_WRITE_2
             ACALL  RBYTE
             MOV    A_SECOND,A
             ACALL  RECIVE
             CJNE   A,#CR,RE_WRITE_2
RE_WRITE_2: RET
RECIVE_TIME: ACALL  REC_TI_DATE
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
            ACALL   REC_TI_HOUR
            ACALL   WRITE_TIME
            RET
;                           Recive SCAN TIME
;           Recive Time to Scan   hh:mm:ss

R_SCAN_TIME: ACALL  REC_TI_HOUR
            MOV     AS_HOUR,A_HOUR
            MOV     A,A_HOUR
            MOV     DPTR,#0004H
            MOVX    @DPTR,A
            MOV     AS_MINUTE,A_MINUTE
            MOV     A,A_MINUTE
            MOV     DPTR,#0005H
            MOVX    @DPTR,A
            MOV     AS_SECOND,A_SECOND
            MOV     A,A_SECOND
            MOV     DPTR,#0006H
            MOVX    @DPTR,A
            RET
;                           Check Data in RAM

CHECK_RAM: MOV      A,DPH
            CJNE    A,#00H,RE_CH_RAM
            MOV     A,DPL
            CJNE    A,#00H,RE_CH_RAM
            SETB    P1.7
            SJMP    $
RE_CH_RAM: RET
;                           Receive data in RAM
;           DPTR = R3,R2 = DPH,DPL   Register blank 1
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
TIME_IN_RAM: SETB   RS0
             MOV    DPH,R3
             MOV    DPL,R2
             MOV    A,A_DATE
             MOVX   @DPTR,A
             INC    DPTR
             ACALL  CHECK_RAM
             MOV    A,A_MONTH
             MOVX   @DPTR,A
             INC    DPTR
             ACALL  CHECK_RAM
             MOV    A,A_YEAR
             MOVX   @DPTR,A
             INC    DPTR
             ACALL  CHECK_RAM
             MOV    A,A_HOUR
             MOVX   @DPTR,A
             INC    DPTR
             ACALL  CHECK_RAM
             MOV    A,A_MINUTE
             MOVX   @DPTR,A
             INC    DPTR
             ACALL  CHECK_RAM
             MOV    A,A_SECOND
             MOVX   @DPTR,A
             INC    DPTR
             ACALL  CHECK_RAM
             MOV    R3,DPH
             MOV    R2,DPL
             MOV    A,R3
             MOV    DPTR,#0000H
             MOVX   @DPTR,A
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
            MOV     A,R2
            MOV     DPTR,#0001H
            MOVX    @DPTR,A
            CLR     RS0
            RET
;                                   Check Data

DATA_IN_RAM: CLR    F0
            ACALL   RECIVE_T
            CJNE    A,#0FFH,FIRST_DATA
            SJMP    ERROR
FIRST_DATA: CJNE    A,#SOH,CHECK_POINT
            SJMP    OK_DATA
CHECK_POINT: ACALL  RECIVE_T
            CJNE    A,#0FFH,FIRST_DATA
            SJMP    ERROR
            CJNE    A,#SOH,OK_DATA
            SJMP    ERROR
OK_DATA:    MOV     CHECKSUM,#00H
            ACALL   RBYTE
            CJNE    A,#ADD_SCAN,ERROR
            ACALL   RECIVE
            CJNE    A,#MIN,ERROR
            ACALL   RBYTE
            MOV     BUF_ADD,A
            SETB    RS0
            SETB    RS1
            MOV     A,@R0
            CLR     RS0
            CLR     RS1
            CJNE    A,BUF_ADD,ERROR
            ACALL   RECIVE
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                CJNE    A,#COMMA,ERROR
                SETB    RS0
                MOV     DPH,R3
                MOV     DPL,R2
                MOV     R4,#08H
LOOP_DATA:      ACALL   RECIVE
                ACALL   ATOHS
                MOVX    @DPTR,A
                INC     DPTR
                ACALL   CHECK_RAM
                CLR     RS0
                ACALL   RBYTE
                SETB    RS0
                MOVX    @DPTR,A
                INC     DPTR
                ACALL   CHECK_RAM
                ACALL   RECIVE
                CJNE    A,#COMMA,ERROR
                DJNZ    R4,LOOP_DATA
                MOV     CHECK_S,CHECKSUM
                ACALL   RBYTE
                SWAP    A
                CJNE    A,CHECK_S,ERROR
                ACALL   RECIVE
                CJNE    A,#EOT,ERROR
                MOV     R2,DPL
                MOV     R3,DPH
                MOV     A,R3
                MOV     DPTR,#0000H
                MOVX    @DPTR,A
                MOV     A,R2
                MOV     DPTR,#0001H
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
            MOVX    @DPTR,A
            SJMP    RET_DATA
ERROR:      SETB    F0
RET_DATA:   CLR     RS0
            RET
;                           Send Command

SE_COMMAND: ACALL   SEND_T
            SETB    RS0
            SETB    RS1
            MOV     A,@R0
            CLR     RS0
            CLR     RS1
            ACALL   SBYTE
            ACALL   SEND_MIN
            MOV     A,#ADD_SCAN
            ACALL   SBYTE
            ACALL   SEND_CR
            ACALL   DATA_IN_RAM
            JB      F0,SE_COMMAND
            RET
;                           Run SCANNER
RUN_SCAN:   CLR     FLAG_RAM            ; Initial FLAG of RAM
            MOV     A,#00H
            MOV     DPTR,#0002H
            MOVX    @DPTR,A
            SETB    RS0
            MOV     R2,#07H             ; Initial DPTR
            MOV     A,NUMBER_ADD
            ADD     A,R2
            MOV     R2,A
            MOV     R3,#00H             ; Initial DPTR
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                CLR     RS0
RUN_SCAN_1: ACALL   CHECK_TIME
                MOV     R1,NUMBER_ADD
                SETB    RS0
                SETB    RS1
                MOV     R0,#START_ADD
                CLR     RS0
                CLR     RS1
LOOP_SCAN: ACALL    TIME_IN_RAM
ANOTHER_A: ACALL    SE_COMMAND
                SETB    RS0
                SETB    RS1
                INC     R0
                CLR     RS0
                CLR     RS1
                DJNZ    R1,ANOTHER_A
                SJMP    RUN_SCAN_1
                RET
;                       Check time to Scan

CHECK_TIME: ACALL   READ_TIME
                ACALL   ADD_SECOND
                ACALL   ADD_MIN
                ACALL   ADD_HOUR
                MOV     B_HOUR,A_HOUR
                MOV     B_MINUTE,A_MINUTE
                MOV     B_SECOND,A_SECOND
NOT_TIME:   ACALL   READ_TIME
                MOV     A,A_SECOND
                CJNE    A,B_SECOND,NOT_TIME
                MOV     A,A_MINUTE
                CJNE    A,B_MINUTE,NOT_TIME
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                MOV     A,A_HOUR
                CJNE    A,B_HOUR,NOT_TIME
                RET

ADD_SECOND:     MOV     A,A_SECOND
                ADD     A,AS_SECOND
                DA      A
                JC      OVER_SEC
                CJNE    A,#60H,$+3
                JNC     OVER_SEC_1
                MOV     A_SECOND,A
                CLR     C
                RET

OVER_SEC_1:     CLR     ACC.6
                CPL     ACC.5
                ANL     A,#7FH
                MOV     A_SECOND,A
                SJMP    SET_MIN
OVER_SEC:       ADD     A,#40H
                DA      A
                MOV     A_SECOND,A
SET_MIN:        SETB    C
                RET

ADD_MIN:        MOV     A,A_MINUTE
                ADDC    A,AS_MINUTE
                DA      A
                JC      OVER_MIN
                CJNE    A,#60H,$+3
                JNC     OVER_MIN_1
                MOV     A_MINUTE,A
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                CLR     C
                RET

OVER_MIN_1:     CLR     ACC.6
                CPL     ACC.5
                ANL     A,#7FH
                MOV     A_MINUTE,A
                SJMP    SET_HOUR
OVER_MIN:       ADD     A,#40H
                DA      A
                MOV     A_MINUTE,A
SET_HOUR:       SETB    C
                RET

ADD_HOUR:       MOV     A,A_HOUR
                ADDC    A,AS_HOUR
                DA      A
                CJNE    A,#24H,$+3
                JNC     OVER_HOUR_1
                MOV     A_HOUR,A
                CLR     C
                RET

OVER_HOUR_1:    CJNE    A,#44H,$+3
                JC      OVER_HOUR_2
                CPL     ACC.6
                CPL     ACC.5
                CLR     ACC.2
                SJMP    OVER_OUT
OVER_HOUR_2:    CJNE    A,#42H,$+3
                JC      OVER_HOUR_3
                CPL     ACC.6
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                CPL     ACC.4
                CPL     ACC.3
                CPL     ACC.1
                SJMP    OVER_OUT
OVER_HOUR_3:    CJNE    A,#40H,$+3
                JC      OVER_HOUR_4
                CPL     ACC.6
                CPL     ACC.4
                CPL     ACC.2
                CPL     ACC.1
                SJMP    OVER_OUT
OVER_HOUR_4:    CJNE    A,#34H,$+3
                JC      OVER_HOUR_5
                CPL     ACC.5
                CLR     ACC.3
                CPL     ACC.2
                SJMP    OVER_OUT
OVER_HOUR_5:    CJNE    A,#32H,$+3
                JC      OVER_HOUR_6
                CPL     ACC.5
                CPL     ACC.4
                CPL     ACC.3
                CPL     ACC.1
                SJMP    OVER_OUT
OVER_HOUR_6:    CJNE    A,#30H,$+3
                JC      OVER_HOUR_7
                CPL     ACC.5
                CPL     ACC.4
                CPL     ACC.2
                CPL     ACC.1
                SJMP    OVER_OUT
OVER_HOUR_7:    CPL     ACC.5
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
            CLR     ACC.3
            CPL     ACC.2
OVER_OUT:   MOV     A_HOUR,A
            RET
;                   Recive Address of SENDER

RECIVE_ADD: SETB    RS1
            ACALL   RECIVE
            CJNE    A,#SPACE,RE_RETURN
            MOV     DPTR,#0007H
            MOV     R1,#START_ADD       ; Address to start
            MOV     R0,#00H             ; Number of Address
LOOP_ADD:   CLR     RS1
            ACALL   RBYTE
            SETB    RS1
            MOV     @R1,A
            MOVX    @DPTR,A
            INC     DPTR
            INC     R0
            INC     R1
            ACALL   RECIVE
            CJNE    A,#COMMA,RE_R_ADD
            SJMP    LOOP_ADD
RE_R_ADD:   MOV     NUMBER_ADD,R0
            MOV     A,R0
            MOV     DPTR,#0003H
            MOVX    @DPTR,A
RE_RETURN:  CLR     RS1
            RET

CHECK_OUT:  PUSH    ACC
            MOV     A,DPH
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
            CJNE    A,0BH,CHECK_BOT    ; 0BH = R3
            MOV     A,DPL
            CJNE    A,0AH,CHECK_BOT    ; 0AH = R2
            POP     ACC
            POP     ACC
            POP     ACC
            MOV     A,R1
            CJNE    A,NUMBER_ADD,NO_COMPLET
            SETB    P1.7
            SJMP    $
NO_COMPLET: INC     R1
            AJMP    START_TRANS
CHECK_BOT: MOV      A,DPH
            CJNE    A,#00H,RE_CH_OUT
            MOV     A,DPL
            CJNE    A,#00H,RE_CH_OUT
            SETB    P1.7
            SJMP    $
RE_CH_OUT: POP      ACC
            RET

CHECK_OUT_1: PUSH   ACC
            MOV     A,DPH
            CJNE    A,0BH,CHECK_BOT_1 ; 0BH = R3
            MOV     A,DPL
            CJNE    A,0AH,CHECK_BOT_1 ; 0AH = R2
            POP     ACC
            POP     ACC
            POP     ACC
            POP     ACC
            POP     ACC
            MOV     A,R1
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                CJNE    A,NUMBER_ADD,NO_COMPLET_1
                SETB    P1.7
                SJMP    $
NO_COMPLET_1: INC   R1
                SJMP    START_TRANS
CHECK_BOT_1: MOV    A,DPH
                CJNE    A,#00H,RE_CH_OUT_1
                MOV     A,DPL
                CJNE    A,#00H,RE_CH_OUT_1
                SETB    P1.7
                SJMP    $
RE_CH_OUT_1: POP    ACC
                RET

CHECK_INC:  MOV     A,DPH
                CJNE    A,#00H,CH_INC_OUT
                MOV     A,DPL
                CJNE    A,#00H,CH_INC_OUT
                MOV     R2,#0FFH
                MOV     R3,#0FFH
CH_INC_OUT: RET

COUNTER_1:  MOV     A,R1
                DEC     A
                MOV     B,#16D
                MUL     AB
                MOV     5FH,A
                JZ      RET_COUNT_1
LOOP_CO_1:  INC     DPTR
                ACALL   CHECK_OUT_1
                DJNZ    5FH,LOOP_CO_1
RET_COUNT_1: RET
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
COUNTER_2: MOV     A,R1
           CLR     C
           MOV     5FH,A
           MOV     A,R6
           SUBB    A,5FH
           MOV     B,#16D
           MUL     AB
           MOV     5FH,A
           JZ      RET_COUNT_2
LOOP_CO_2: INC     DPTR
           ACALL   CHECK_OUT_1
           DJNZ    5FH,LOOP_CO_2
RET_COUNT_2: RET

S_SENDER:  MOV     A,#'S'
           ACALL   SEND
           MOV     A,#'e'
           ACALL   SEND
           MOV     A,#'n'
           ACALL   SEND
           MOV     A,#'d'
           ACALL   SEND
           MOV     A,#'e'
           ACALL   SEND
           MOV     A,#'r'
           ACALL   SEND
           ACALL   SEND_SPACE
           MOV     A,#'@'
           ACALL   SEND
           ACALL   SEND_SPACE
           RET
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
;                    Transmit data in RAM to Central
;          DPTR = #00,R0 = DPH,DPL  register blank 1
;
TRANSMIT:  SETB     RS0
           MOV      DPH,R3
           MOV      DPL,R2
           INC      DPTR
           MOV      R3,DPH
           MOV      R2,DPL
           ACALL    CHECK_INC
           MOV      R6,NUMBER_ADD
           MOV      R1,#01H              ; Status Data Point

START_TRANS: ACALL  SEND_CR_LF
           ACALL    SEND_SPACE
           ACALL    S_SENDER
           MOV      A,R1
           DEC      A
           SETB     RS1
           MOV      R0,#START_ADD
           ADD      A,R0
           MOV      R0,A
           MOV      A,@R0
           CLR      RS1
           CLR      RS0
           ACALL    SBYTE                ; Send Address SENDER
           SETB     RS0
           ACALL    SEND_CR_LF

           MOV      R0,#07H
           MOV      A,NUMBER_ADD
           ADD      A,R0
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                MOV     R0,A                    ; INITIAL OF DATA DPTR
                MOV     DPH,#00H
                MOV     DPL,R0

LOOP_OUT_TI:    MOV     CHECKSUM,#00H
                MOVX    A,@DPTR
                INC     DPTR
                ACALL   CHECK_OUT
                CLR     RS0
                ACALL   SBYTE                   ; Send DAY
                SETB    RS0
                ACALL   SEND_MIN
                MOVX    A,@DPTR
                INC     DPTR
                ACALL   CHECK_OUT
                CLR     RS0
                ACALL   SBYTE                   ; Send MONTH
                SETB    RS0
                ACALL   SEND_MIN
                MOVX    A,@DPTR
                INC     DPTR
                ACALL   CHECK_OUT
                CLR     RS0
                ACALL   SBYTE                   ; Send YEAR
                SETB    RS0
                ACALL   SEND_SPACE
                MOVX    A,@DPTR
                INC     DPTR
                ACALL   CHECK_OUT
                CLR     RS0
                ACALL   SBYTE                   ; Send HOUR
                SETB    RS0
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
            ACALL   SEND_COLON
            MOVX    A,@DPTR
            INC     DPTR
            ACALL   CHECK_OUT
            CLR     RS0
            ACALL   SBYTE               ; Send MINUTE
            SETB    RS0
            ACALL   SEND_COLON
            MOVX    A,@DPTR
            INC     DPTR
            ACALL   CHECK_OUT
            CLR     RS0
            ACALL   SBYTE               ; Send SECOND
            SETB    RS0
            ACALL   SEND_SPACE
            ACALL   SEND_COMMA
            MOV     R7,#08H
            ACALL   COUNTER_1
LOOP_OUT_DA: MOVX   A,@DPTR
            INC     DPTR
            ACALL   CHECK_OUT
            CJNE    A,#0DH,OUT__
            MOV     A,#NEG
            SJMP    LOOP_OUT_1
OUT__:      ACALL   HTOAS
LOOP_OUT_1: ACALL   SEND
            MOVX    A,@DPTR
            INC     DPTR
            ACALL   CHECK_OUT
            CJNE    A,#0DDH,OUT__1
            MOV     A,#NEG
            ACALL   SEND
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                ACALL   SEND
                SJMP    LOOP_OUT_2
OUT__1:         CLR     RS0
                ACALL   SBYTE
                SETB    RS0
LOOP_OUT_2: ACALL   SEND_COMMA
                DJNZ    R7,LOOP_OUT_DA
                MOV     A,CHECKSUM
                CLR     RS0
                ACALL   SBYTE
                SETB    RS0
                ACALL   SEND_CR_LF
                ACALL   COUNTER_2
                AJMP    LOOP_OUT_TI
                CLR     RS0
                RET
;                       Check Telecommand
;               R  Read Data from RTC
;               W  Write Data to RTC
;               S  Write Data to Scan
;               G  Run Scanner
;               T  Transmit Data from RAM to Central Station

CHECK_COM: ACALL    RECIVE
                CJNE    A,#R,CHECK_1        ; Check command "r"
                ACALL   RECIVE
                CJNE    A,#CR,RE_CHECK
                ACALL   READ_TIME           ; Read time
                ACALL   SEND_TIME           ; Send time to Central Station
                SJMP    RE_CHECK
CHECK_1:        CJNE    A,#W,CHECK_2        ; Check command "w"
                ACALL   RECIVE_TIME         ; Write time
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                SJMP    RE_CHECK
CHECK_2:        CJNE    A,#S,CHECK_3        ; Check command "s"
                ACALL   R_SCAN_TIME         ; Recive SCAN TIME
                SJMP    RE_CHECK
CHECK_3:        CJNE    A,#G,CHECK_4        ; Check command "g"
                ACALL   RECIVE
                CJNE    A,#CR,RE_CHECK
                ACALL   RUN_SCAN            ; Run SCANNER
                SJMP    RE_CHECK
CHECK_4:        CJNE    A,#'a',CHECK_5      ; Check command "a"
                ACALL   RECIVE_ADD          ; Read address of Sender
                SJMP    RE_CHECK
CHECK_5:        CJNE    A,#'t',RE_CHECK     ; Check command "t"
                ACALL   RECIVE
                CJNE    A,#CR,RE_CHECK
                ACALL   TRANSMIT            ; Send data in RAM
RE_CHECK:       RET
;                           Read Data from RAM

READ_RAM:       MOV     DPTR,#0003H
                MOVX    A,@DPTR
                MOV     NUMBER_ADD,A
                MOV     R0,A
                MOV     R1,START_ADD
                MOV     DPTR,#0007H
LOOP_READ_RAM:  MOVX    A,@DPTR
                MOV     @R1,A
                INC     R1
                INC     DPTR
                DJNZ    R0,LOOP_READ_RAM
                SETB    RS0
                MOV     DPTR,#0000H
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

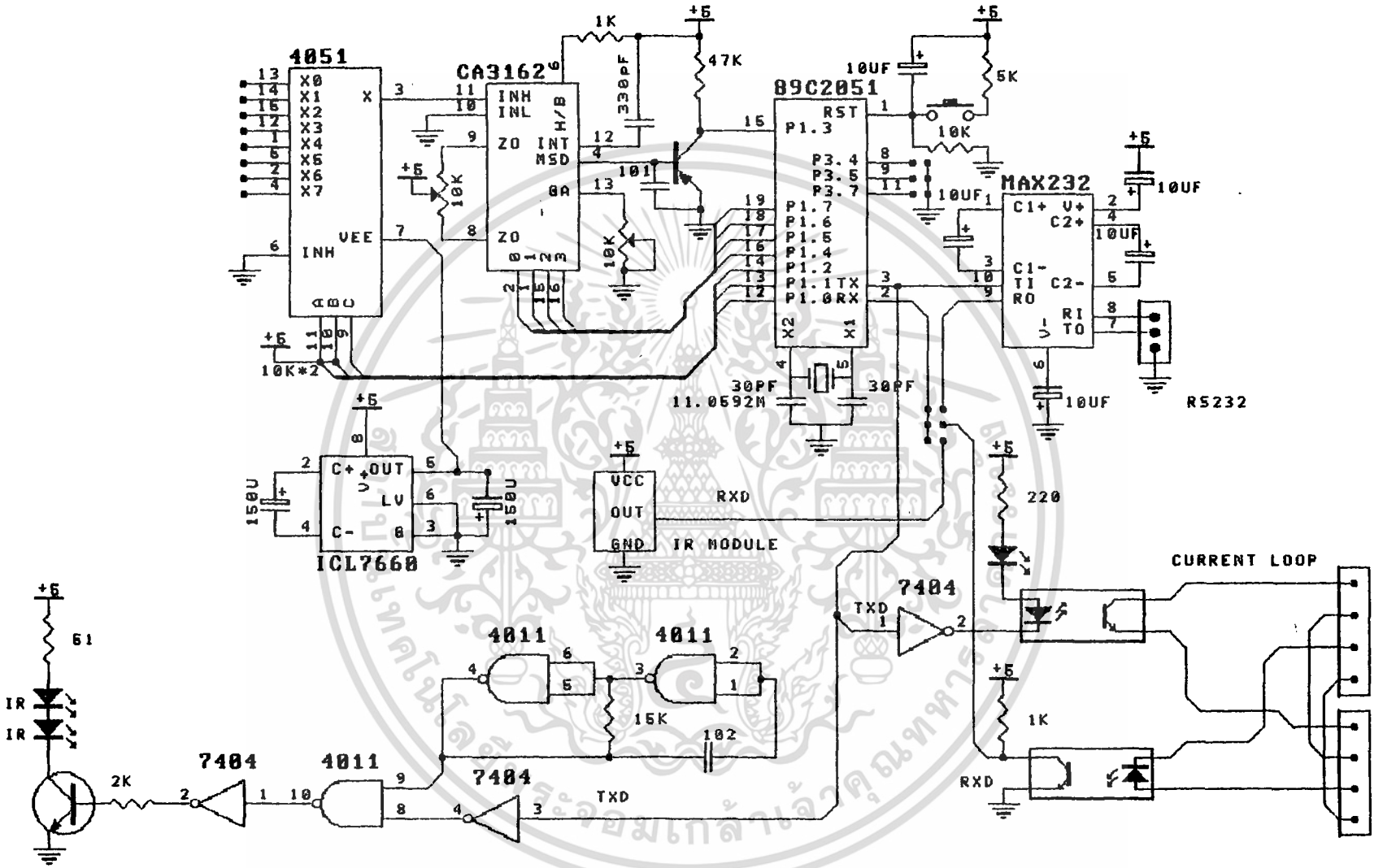
```
        MOVX    A,@DPTR
        MOV     R3,A
        MOV     DPTR,#0001H
        MOVX    A,@DPTR
        MOV     R2,A
        CLR     RS0
        MOV     DPTR,#0004H
        MOVX    A,@DPTR
        MOV     AS_HOUR,A
        MOV     DPTR,#0005H
        MOVX    A,@DPTR
        MOV     AS_MINUTE,A
        MOV     DPTR,#0006H
        MOVX    A,@DPTR
        MOV     AS_SECOND,A
        RET
;                       Main Program
MAIN:   MOV     R0,#00H             ; Power up delay
        DJNZ    R0,$
        MOV     SP,#STRACK          ; Set strack pointer
        CLR     P1.7
        ACALL   READ_RAM
        ACALL   INTI                ; Initial Serial port
SUMAIN: ACALL   CHECK_COM           ; Check Telecommand
        SJMP    SUMAIN
        END
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

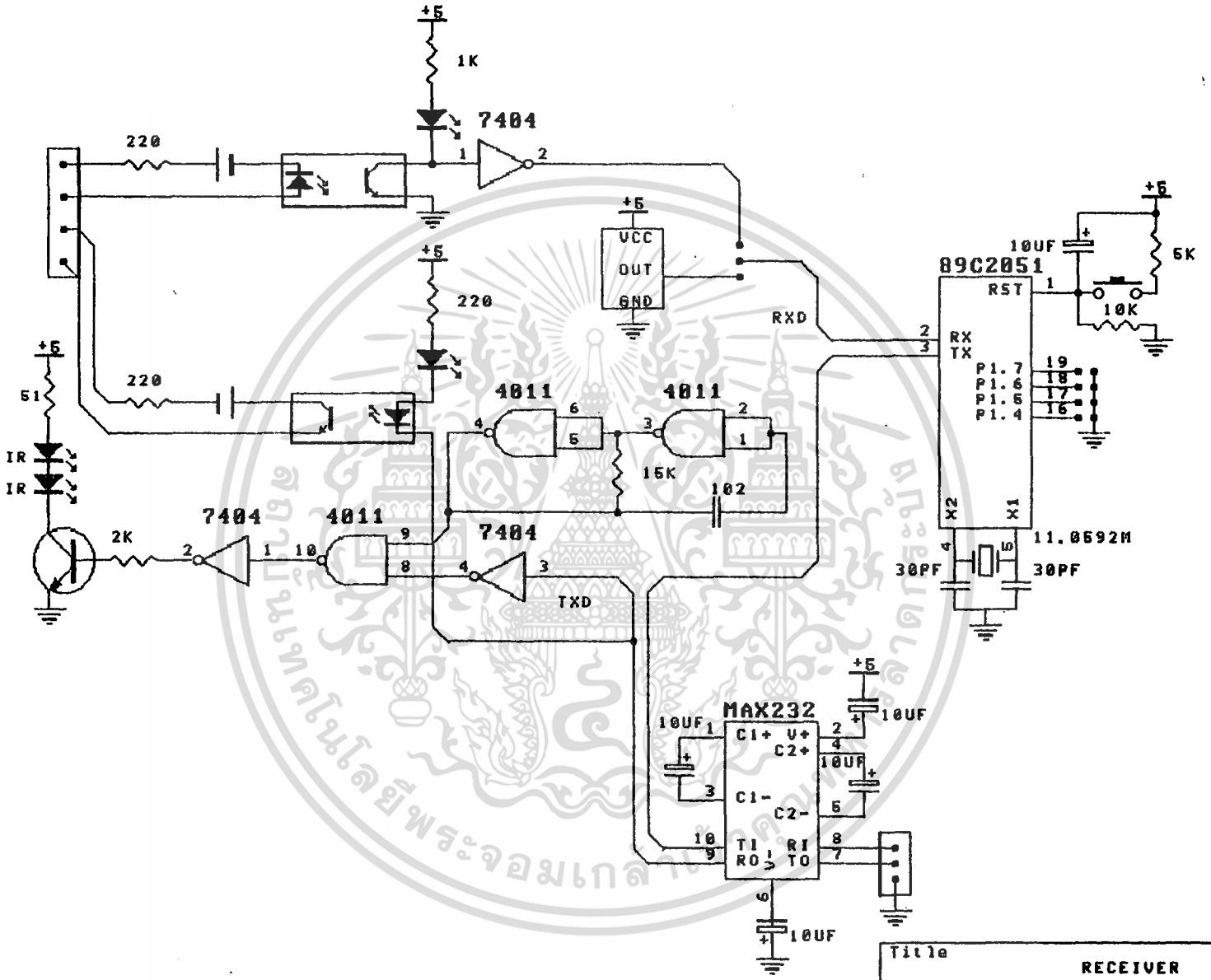
ภาคผนวก ข

วงจร

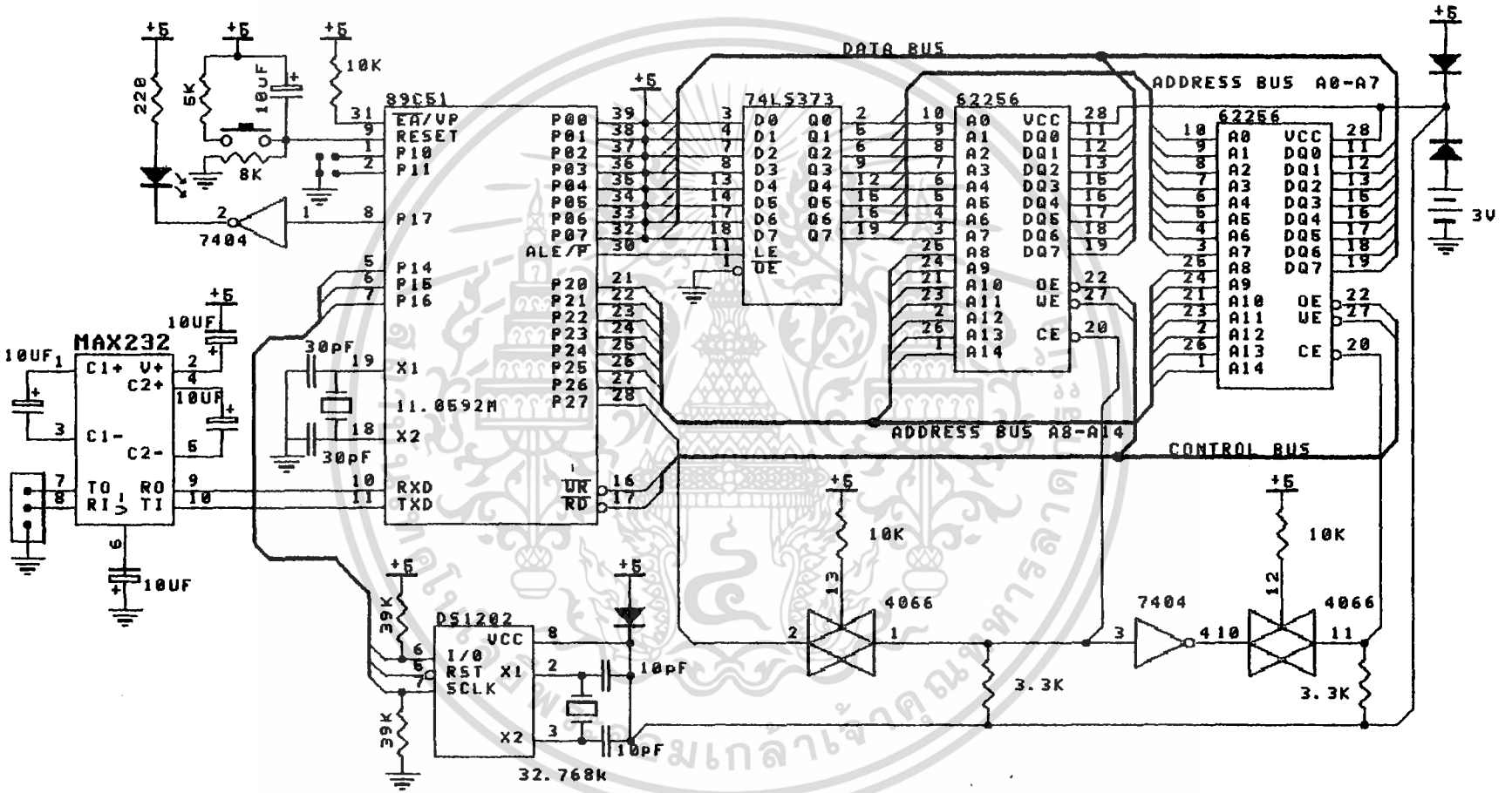
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



Title: MINIATURE ANALOG SIGNAL SENDER (MASS)		
Size: A4	Number	Revision
Date: 29-FEB 1996		Sheet 1 of 3
File: PRO1/3		Drawn By:



Title	RECEIVER	
Size A4	Number	Revision
Date: 29-FEB 1996		Sheet 2 of 3
File: PRO9/3		Drawn By:



Title: SCANNER		
Size: A4	Number	Revision
Data: 29-FEB 1996		Sheet 3 of 3
File: PR010/3		Drawn By:

ภาคผนวก ค
ข้อมูลอุปกรณ์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Features

- Compatible with MCS-51™ Products
- 2 Kbytes of Reprogrammable Flash Memory
  - Endurance: 1,000 Write/Erase Cycles
  - Data Retention: 10 Years
- 2.7 V to 6 V Operating Range
- Fully Static Operation: 0 Hz to 24 MHz
- Two-Level Program Memory Lock
- 128 x 8-Bit Internal RAM
- 15 Programmable I/O Lines
- Two 16-Bit Timer/Counters
- Five Interrupt Sources
- Programmable Serial UART Channel
- Direct LED Drive Outputs
- On-Chip Analog Comparator
- Low Power Idle and Power Down Modes

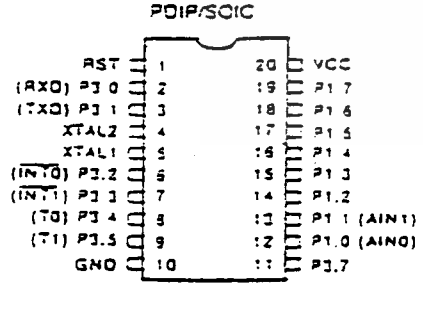
Description

The AT89C2051 is a low-voltage, high-performance CMOS 8-bit microcomputer with 2 Kbytes 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 MCS-51™ instruction set and pinout. By combining a versatile 8-bit CPU with Flash on a monolithic chip, the Atmel AT89C2051 is a powerful microcomputer which provides a highly flexible and cost effective solution to many embedded control applications.

The AT89C2051 provides the following standard features: 2 Kbytes of Flash, 128 bytes of RAM, 15 I/O lines, two 16-bit timer/counters, a five source two-level interrupt architecture, a full duplex serial port, a precision analog comparator, on-chip oscillator and clock circuitry. In addition, the AT89C2051 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 Configuration

PDIP/SOIC



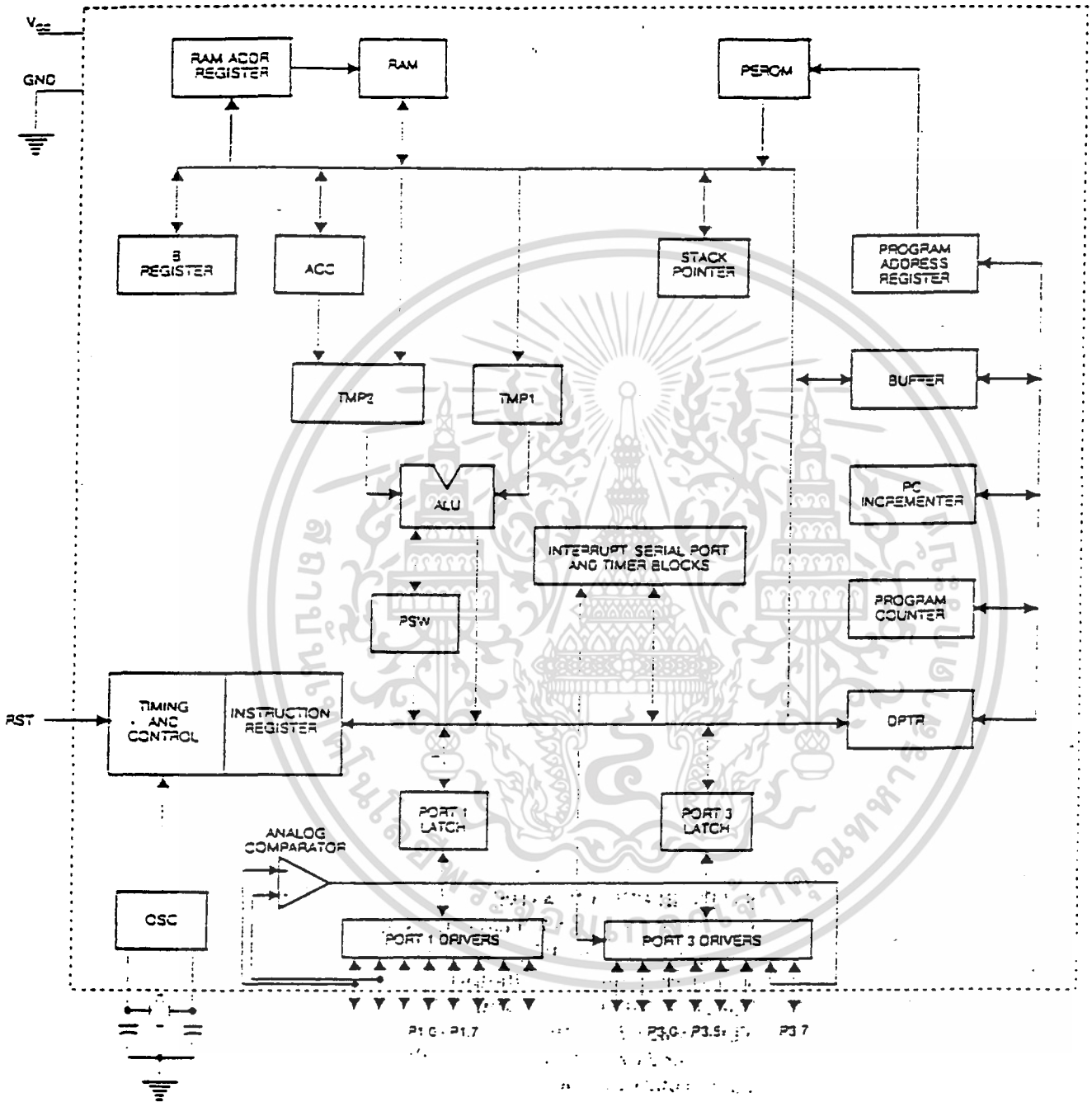
SALES REPRESENTATIVE
SEMIKON COMPANY LIMITED (SMK)
4 Fl., Room 406 Phansak Bldg.,
138/1 Phetburi Rd., Rajthevee
Bangkok 10400, Thailand
TEL: (662) 215-9496-7
(662) 215-0760-74 Ext. 406
Fax: (662) 215-6857

8-Bit Microcontroller with 2 Kbytes Flash

AT89C2051

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Block Diagram



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Pin Description

Vcc

Supply voltage.

GND

Ground.

Port 1

Port 1 is an 8-bit bidirectional I/O port. Port pins P1.2 to P1.7 provide internal pullups. P1.0 and P1.1 require external pullups. P1.0 and P1.1 also serve as the positive input (AIN0) and the negative input (AIN1), respectively, of the on-chip precision analog comparator. The Port 1 output buffers can sink 20 mA and can drive LED displays directly. When 1s are written to Port 1 pins, they can be used as inputs. When pins P1.2 to P1.7 are used as inputs and are externally pulled low, they will source current (I_{IL}) because of the internal pullups.

Port 1 also receives code data during Flash programming and program verification.

Port 3

Port 3 pins P3.0 to P3.5, P3.7 are seven bidirectional I/O pins with internal pullups. P3.6 is hard-wired as an input to the output of the on-chip comparator and is not accessible as a general purpose I/O pin. The Port 3 output buffers can sink 20 mA. 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 AT89C2051 as listed below:

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)

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

RST

Reset input. All I/O pins are reset to 1s as soon as RST goes high. Holding the RST pin high for two machine cycles while the oscillator is running resets the device.

XTAL1

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

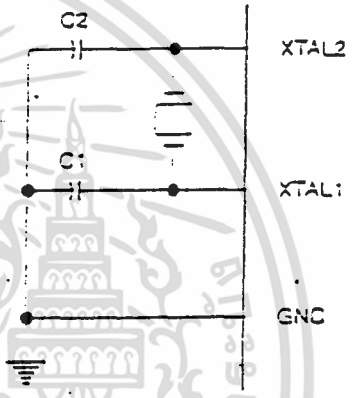
XTAL2

Output from the inverting oscillator amplifier.

Oscillator Characteristics

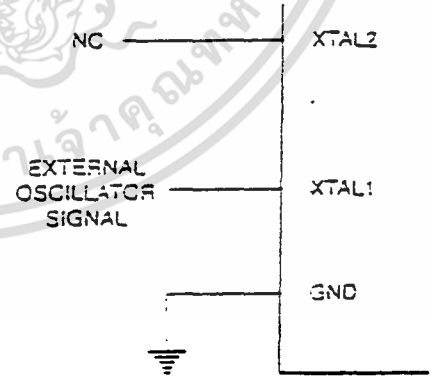
XTAL1 and XTAL2 are the input and output, respectively, of an inverting amplifier which can be configured for use as an on-chip oscillator, as shown in Figure 1. Either a quartz crystal or ceramic resonator may be used. To drive the device from an external clock source, XTAL2 should be left unconnected while XTAL1 is driven as shown in Figure 2. 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.

Figure 1. Oscillator Connections



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

Figure 2. External Clock Drive Configuration



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลง เผยแพร่ หรืออ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Special Function Registers

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

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.

Table 1. AT89C2051 SFR Map and Reset Values

0F8H									0FFH
0F0H	B 00000000								0F7H
0E8H									0EFH
0E0H	ACC 00000000								0E7H
0D8H									0DFH
0D0H	PSW 00000000								0D7H
0C8H									0CFH
0C0H									0C7H
0B8H	IP X0000000								0BFH
0B0H	P3 11111111								0B7H
0A8H	IE 00000000								0AFH
0A0H									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		SP 00000111	DPL 00000000	DPH 00000000				PCON 00000000	87H

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Program Memory Lock Bits

On the chip are two lock bits which can be left unprogrammed (U) or can be programmed (P) to obtain the additional features listed in the table below:

Lock Bit Protection Modes[1]

Program Lock Bits			
	LB1	LB2	Protection Type
1	U	U	No program lock features.
2	P	U	Further programming of the Flash is disabled.
3	P	P	Same as mode 2, also verify is disabled.

Note: 1. The Lock Bits can only be erased with the Chip Erase operation

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.

P1.0 and P1.1 should be set to '0' if no external pullups are used, or set to '1' if external pullups are used.

It should be noted that when idle 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 is terminated by reset, the instruction following the one that invokes Idle should not be one that writes 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.

P1.0 and P1.1 should be set to '0' if no external pullups are used, or set to '1' if external pullups are used.

Programming The Flash

The AT89C2051 is shipped with the 2 Kbytes of on-chip PEROM code memory array in the erased state (i.e., contents = FFH) and ready to be programmed. The code memory array is programmed one byte at a time. *Once the array is programmed, to re-program any non-blank byte, the entire memory array needs to be erased electrically.*

Internal Address Counter: The AT89C2051 contains an internal PEROM address counter which is always reset to 000H on the rising edge of RST and is advanced by applying a positive going pulse to pin XTAL1.

Programming Algorithm: To program the AT89C2051, the following sequence is recommended.

1. Power-up sequence:
   Apply power between Vcc and GND pins
   Set RST and XTAL1 to GND
   With all other pins floating, wait for greater than 10 milliseconds
2. Set pin RST to 'H'
   Set pin P3.2 to 'H'
3. Apply the appropriate combination of 'H' or 'L' logic levels to pins P3.3, P3.4, P3.5, P3.7 to select one of the programming operations shown in the PEROM Programming Modes table.

To Program and Verify the Array:

4. Apply data for Code byte at location 000H to P1.0 to P1.7.
5. Raise RST to 12V to enable programming.
6. Pulse P3.2 once to program a byte in the PEROM array or the lock bits. The byte-write cycle is self-timed and typically takes 1.2 ms.
7. To verify the programmed data, lower RST from 12V to logic 'H' level and set pins P3.3 to P3.7 to the appropiate levels. Output data can be read at the port P1 pins.
8. To program a byte at the next address location, pulse XTAL1 pin once to advance the internal address counter. Apply new data to the port P1 pins.
9. Repeat steps 5 through 8, changing data and advancing the address counter for the entire 2 Kbytes array or until the end of the object file is reached.
10. Power-off sequence:
    set XTAL1 to 'L'
    set RST to 'L'
    Float all other I/O pins
    Turn Vcc power off

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

$\overline{\text{Data}}$ Polling: The AT89C2051 features $\overline{\text{Data}}$ 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 P1.7. Once the write cycle has been completed, true data is valid on all outputs, and the next cycle may begin. $\overline{\text{Data}}$ Polling may begin any time after a write cycle has been initiated.

Ready/$\overline{\text{Busy}}$: The Progress of byte programming can also be monitored by the RDY/$\overline{\text{BSY}}$ output signal. Pin P3.1 is pulled low after P3.2 goes High during programming to indicate BUSY. P3.1 is pulled High again when programming is done to indicate READY.

Program Verify: If lock bits LB1 and LB2 have not been programmed code data can be read back via the data lines for verification:

1. Reset the internal address counter to 000H by bringing RST from 'L' to 'H'.
2. Apply the appropriate control signals for Read Code data and read the output data at the port P1 pins.
3. Pulse pin XTAL1 once to advance the internal address counter.
4. Read the next code data byte at the port P1 pins
5. Repeat steps 3 and 4 until the entire array is read.

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 PEROM array (2 Kbytes) and the two Lock Bits are erased electrically by using the proper combination of control signals and by holding P3.2 low for 10 ms. The code array is written with all "1"s in the Chip Erase operation and must be executed before any non-blank memory byte can be re-programmed.

Reading the Signature Bytes: The signature bytes are read by the same procedure as a normal verification of locations 000H, 001H, and 002H, except that P3.5 and P3.7 must be pulled to a logic low. The values returned are as follows.

(000H) = 1EH indicates manufactured by Atmel
(001H) = 21H indicates 89C2051
(002H) = FFH indicates 12 V programming

Flash Programming Modes

Mode	RST	P3.2/$\overline{\text{PROG}}$	P3.3	P3.4	P3.5	P3.7
Write Code Data[1,3]	12V	⎍	L	H	H	H
Read Code Data[1]	H	H	L	L	H	H
Write Lock Bit - 1	12V	⎍	H	H	H	H
Bit - 2	12V	⎍	H	H	L	L
Chip Erase	12V	⎍[2]	H	L	L	L
Read Signature Byte	H	H	L	L	L	L

Notes: 1. The internal PEROM address counter is reset to 000H on the rising edge of RST and is advanced by a positive pulse at XTAL1 pin.
2. Chip Erase requires a 10 ms $\overline{\text{PROG}}$ pulse.
3. P3.1 is pulled Low during programming to indicate RDY/$\overline{\text{BSY}}$.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Figure 3. Programming the Flash Memory

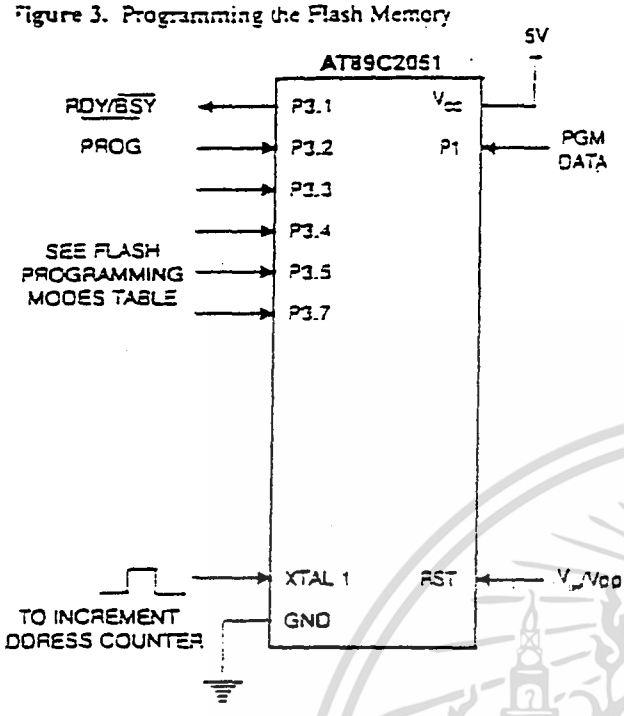
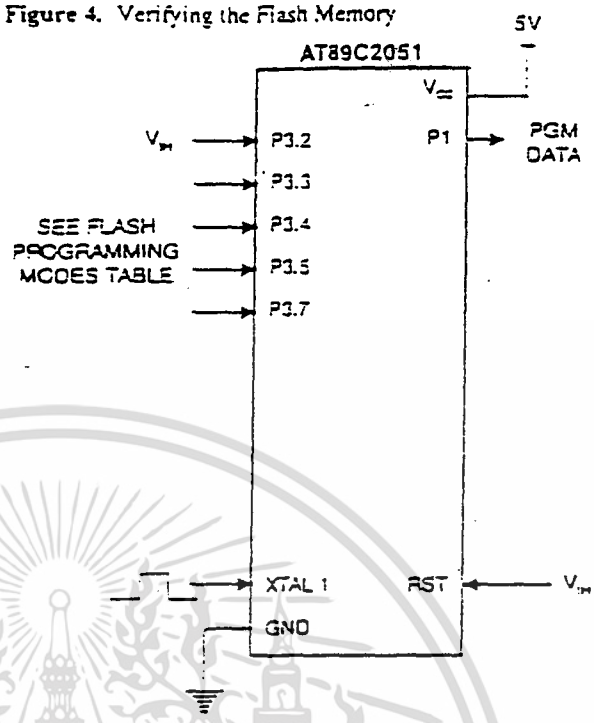


Figure 4. Verifying the Flash Memory



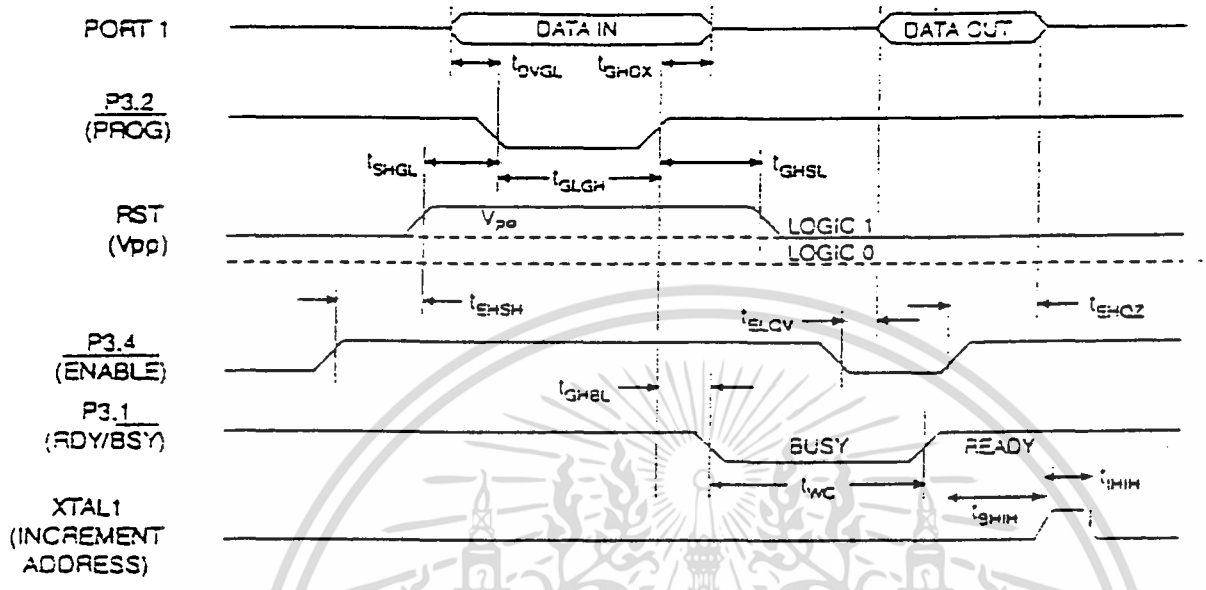
Flash Programming and Verification Characteristics

T_A = 21°C to 27°C, V_{CC} = 5.0 ± 10%

Symbol	Parameter	Min	Max	Units
V_{PP}	Programming Enable Voltage	11.5	12.5	V
I_{PP}	Programming Enable Current		250	µs
t_{DVGL}	Data Setup to $\overline{PROG}$ Low	1.0		µs
t_{GHDX}	Data Hold After $\overline{PROG}$	1.0		µs
t_{EHSH}	P3.4 ($\overline{ENABLE}$) High to V_{PP}	1.0		µs
t_{SHGL}	V_{PP} Setup to $\overline{PROG}$ Low	10		µs
t_{GHSL}	V_{PP} Hold After $\overline{PROG}$	10		µs
t_{GLGH}	$\overline{PROG}$ Width	1	110	µs
t_{ELQV}	$\overline{ENABLE}$ Low to Data Valid		1.0	µs
t_{EHQZ}	Data Float After $\overline{ENABLE}$	0	1.0	µs
t_{GHBL}	$\overline{PROG}$ High to $\overline{BUSY}$ Low		50	ns
t_{WC}	Byte Write Cycle Time		2.0	ms
t_{BHIH}	RDY/$\overline{BSY}$ to Increment Clock Delay	1.0		µs
t_{IHIL}	Increment Clock High	200		ns

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Flash Programming and Verification Waveforms



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	-1.0 V to +7.0 V
Maximum Operating Voltage	6.6 V
DC Output Current	25.0 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.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

D.C. Characteristics

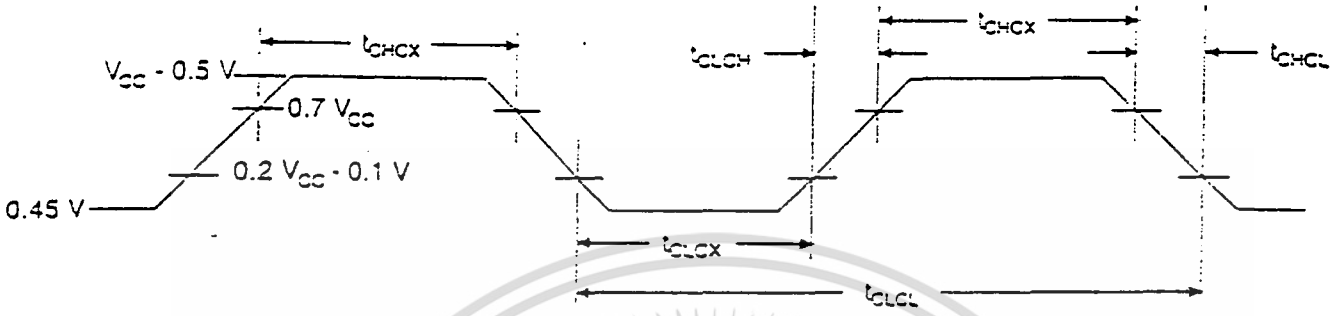
T_A = -40°C to 85°C, V_{CC} = 2.7 V to 6.0 V (unless otherwise noted)

Symbol	Parameter	Condition	Min	Max	Units
V_{IL}	Input Low Voltage		-0.5	0.2 V_{CC}-0.1	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, 3)	I_{OL} = 20 mA, V_{CC} = 5 V I_{OL} = 10 mA, V_{CC} = 2.7 V		0.45	V
V_{OH}	Output High Voltage (Ports 1, 3)	I_{CH} = -80 µA, V_{CC} = 5 V ± 10%	2.4		V
		I_{OH} = -30 µA	0.75 V_{CC}		V
		I_{OH} = -12 µA	0.9 V_{CC}		V
I_{IL}	Logical 0 Input Current (Ports 1, 2, 3)	V_{IN} = 0.45 V		-50	µA
I_{TL}	Logical 1 to 0 Transition Current (Ports 1, 2, 3)	V_{IN} = 2 V		-750	µA
I_{LI}	Input Leakage Current (Port P1.0, P1.1)	0 < V_{IN} < V_{CC}		±10	µA
V_{OS}	Comparator Input Offset Voltage	V_{CC} = 5 V		20	mV
V_{CM}	Comparator Input Common Mode Voltage		0	V_{CC}	V
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, V_{CC} = 6 V/3 V		20/5.5	mA
		Idle Mode, 12 MHz, V_{CC} = 6 V/3 V P1.0 & P1.1 = 0V or V_{CC}		5/1	mA
	Power Down Mode[2]	V_{CC} = 6 V P1.0 & P1.1 = 0V or V_{CC}		100	µA
		V_{CC} = 3 V P1.0 & P1.1 = 0V or V_{CC}		20	µA

Notes: 1. Under steady state (non-transient) conditions, I_{OL} must be externally limited as follows:
Maximum I_{OL} per port pin: 20 mA
Maximum total IOL for all output pins: 30 mA
If IOL exceeds the test condition, VOL 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 2 V

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

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

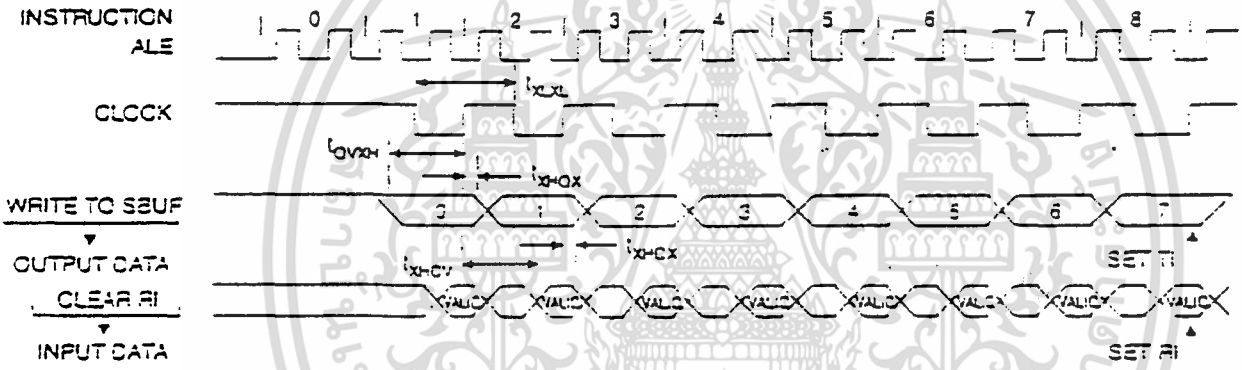
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น รวมทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Serial Port Timing: Shift Register Mode Test Conditions

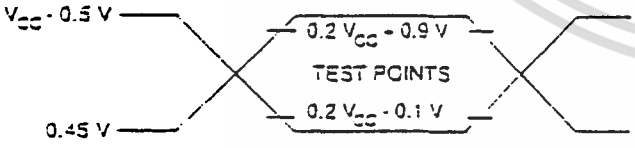
(V_{CC} = 5.0 V ± 20%; Load Capacitance = 80 pF)

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		$2t_{CLCL}$-33		ns
t_{XHDX}	Input Data Hold After Clock Rising Edge	0		0		ns
t_{XHDV}	Clock Rising Edge to Input Data Valid		700		10tCLCL-133	ns

Shift Register Mode Timing Waveforms



AC Testing Input/Output Waveforms[1]



Note: 1. AC inputs during testing are driven at V_{CC} - 0.5 V for a logic 1 and 0.45 V 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

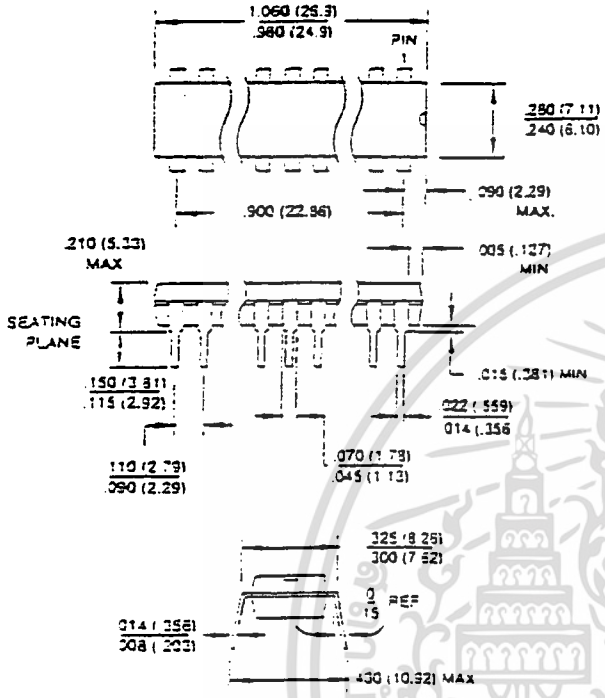
Speed (MHz)	Power Supply	Ordering Code	Package	Operation Range
12	2.7 V to 6.0 V	AT89C2051-12PC AT89C2051-12SC	20P3 20S	Commercial (0°C to 70°C)
		AT89C2051-12PI AT89C2051-12SI	20P3 20S	Industrial (-40°C to 85°C)
16	3.0 V to 6.0 V	AT89C2051-16PC AT89C2051-16SC	20P3 20S	Commercial (0°C to 70°C)
		AT89C2051-16PI AT89C2051-16SI	20P3 20S	Industrial (-40°C to 85°C)
20	3.3 V to 6.0 V	AT89C2051-20PC AT89C2051-20SC	20P3 20S	Commercial (0°C to 70°C)
		AT89C2051-20PI AT89C2051-20SI	20P3 20S	Industrial (-40°C to 85°C)
24	4.0 V to 6.0 V	AT89C2051-24PC AT89C2051-24SC	20P3 20S	Commercial (0°C to 70°C)
		AT89C2051-24PI AT89C2051-24SI	20P3 20S	Industrial (-40°C to 85°C)
16	4.0 V to 6.0 V	AT89C2051-16PA AT89C2051-16SA	20P3 20S	Automotive (-40°C to 125°C)

Package Type	
20P3	20 Lead, 0.300" Wide, Plastic Dual In-line Package (PDIP)
20S	20 Lead, 0.300" Wide, Plastic Gull Wing Small Outline (SOIC)

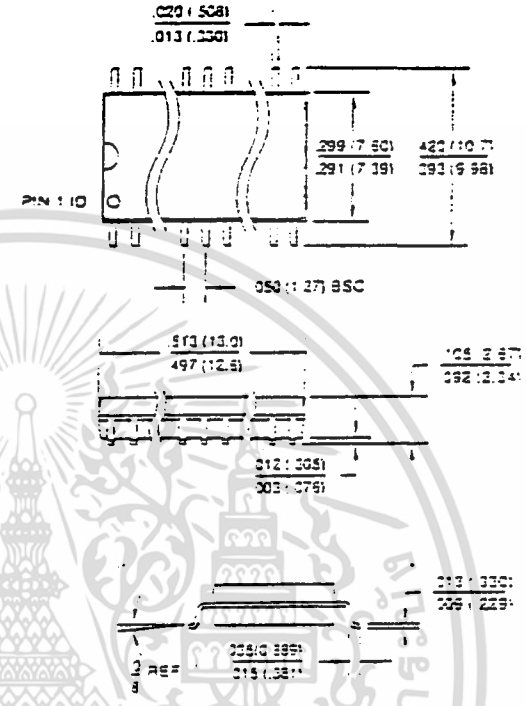
เอกสารนี้เป็น[illegible] ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Packaging Information

20P3, 20 Lead, 0.300" Wide,
Plastic Dual Inline Package (PDIP)
Dimensions in Inches and (Millimeters)



20S, 20 Lead, 0.300" Wide, Plastic Gull Wing Small
Outline (SOIC)
Dimensions in Inches and (Millimeters)



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

HARRIS SEMICONDUCTOR

CA3162

December 1993

A/D Converter for 3-Digit Display

Features

- Dual Slope A/D Conversion
- Multiplexed BCD Display
- Ultra Stable Internal Band Gap Voltage Reference
- Capable of Reading 99mV Below Ground with Single Supply
- Differential Input
- Internal Timing - No External Clock Required
- Choice of Low Speed (4Hz) or High Speed (96Hz) Conversion Rate
- "Hold" Inhibits Conversion but Maintains Delay
- Overrange Indication
  - "EEE" for Reading Greater than +999mV, "-" for Reading More Negative than -99mV When Used With CA3161E
- BCD-to-Seven-Segment Decoder/Driver
- Extended Temperature Range Version Available

Description

The CA3162E and CA3162AE are I^2L monolithic A/D converters that provide a 3 digit multiplexed BCD output. They are used with the CA3161E BCD-to-Seven-Segment Decoder/Driver* and a minimum of external parts to implement a complete 3 digit display. The CA3162AE is identical to the CA3162E except for an extended operating temperature range.

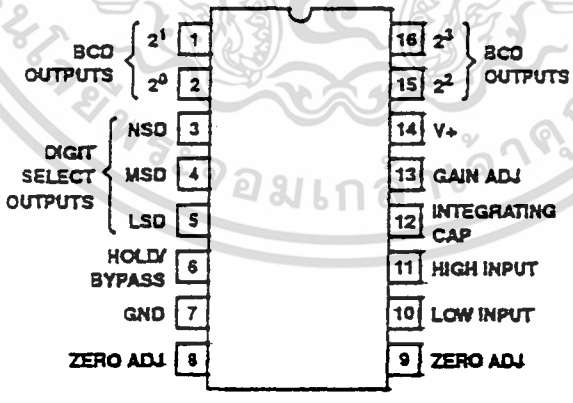
* The CA3161E is described in Display Drivers section of this data book.

Ordering Information

PART NUMBER	TEMPERATURE RANGE	PACKAGE
CA3162E	0°C to +70°C	16 Lead Plastic DIP
CA3162AE	-40°C to +85°C	16 Lead Plastic DIP

Pinout

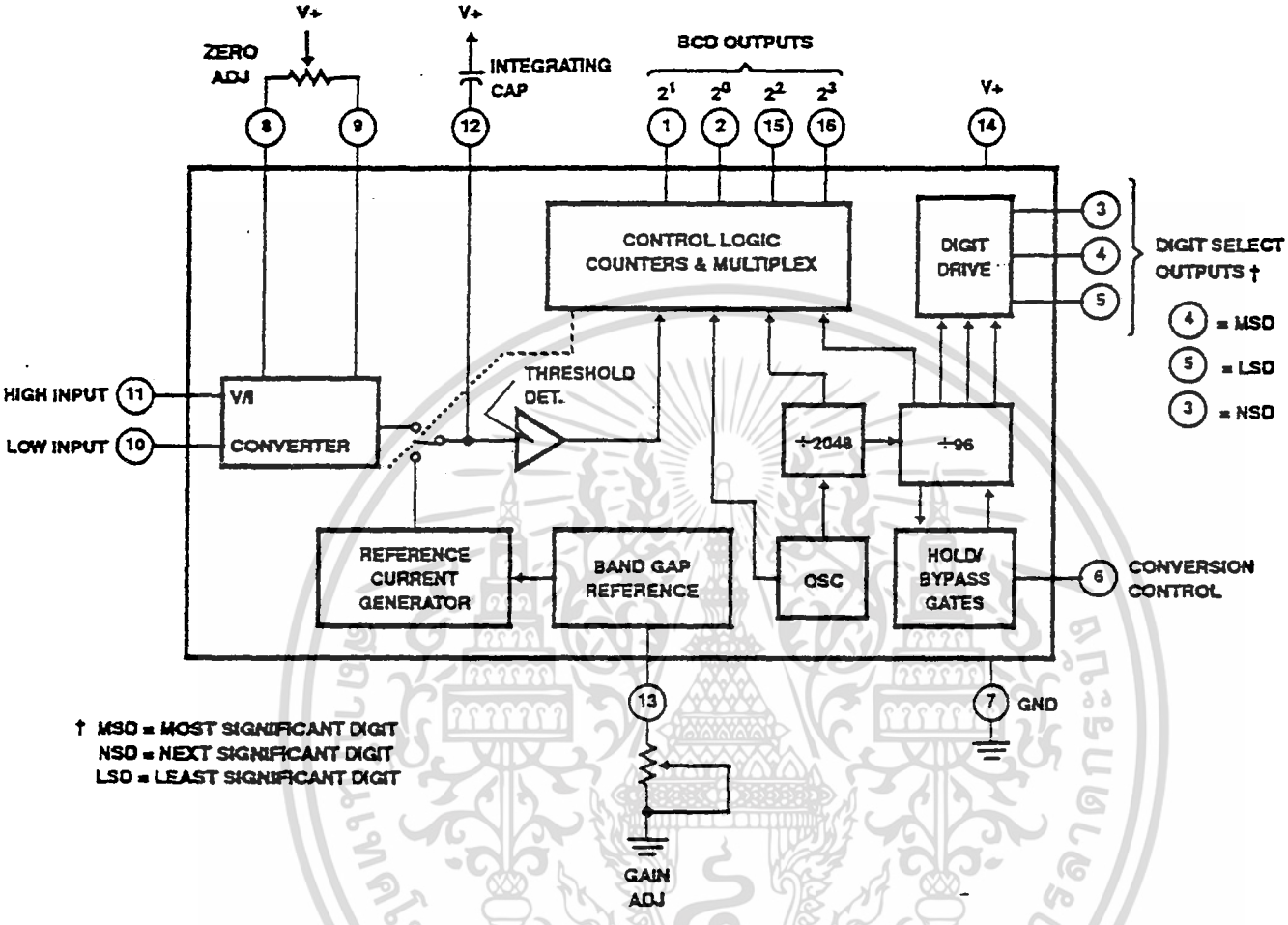
CA3162 (PDIP)
TOP VIEW



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

CAUTION: These devices are sensitive to electrostatic discharge. Users should follow proper I.C. Handling Procedures.
Copyright © Harris Corporation 1993

Functional Block Diagram



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Absolute Maximum Ratings

DC Supply Voltage (Between Pins 7 & 14) ..... +7V
Input Voltage (Pin 10 or 11 to Ground) ..... ±15V
Storage Temperature Range ..... -65°C to +150°C
Lead Temperature (Soldering 10s) ..... +300°C

Thermal Information

Thermal Resistance θ_{JA}
Plastic DIP Package ..... 90°C/W
Operating Temperature Range
CA3162E ..... 0 to +75°C
CA3162AE ..... -40°C to +85°C
Maximum Power Dissipation
Plastic DIP Package 0.67W
Junction Temperature ..... +150°C

CAUTION: Stresses above those listed in "Absolute Maximum Ratings" may cause permanent damage to the device. This is a stress only rating and operation of the device at these or any other conditions above those indicated in the operational sections of this specification is not implied.

Electrical Specifications

T_A = +25°C, V+ = 5V, Zero Pot Centered, Gain Pot = 2.4kΩ Unless Otherwise Specified

PARAMETERS	TEST CONDITIONS	MIN	TYP	MAX	UNITS
Operating Supply Voltage Range, V+		4.5	5	5.5	V
Supply Current, I+	100kΩ to V+ on Pins 3, 4, 5	-	-	17	mA
Input Impedance, Z_I		-	100	-	MΩ
Input Bias Current, I_{IB}	Pins 10 and 11	-	-80	-	nA
Unadjusted Zero Offset	V_{11}-V_{10} = 0V, Read Decoded Output	-12	-	+12	mV
Unadjusted Gain	V_{11}-V_{10} = 900mV, Read Decoded Output	846	-	954	mV
Linearity	Notes 1 and 2	-1	-	+1	Count
Conversion Rate					
Slow Mode	Pin 6 = Open or GND	-	4	-	Hz
Fast Mode	Pin 6 = 5V	-	96	-	Hz
Conversion Control Voltage (Hold Mode) at Pin 6		0.8	1.2	1.6	V
Common Mode Input Voltage Range, V_{ICR}	Notes 3, 4	-0.2	-	+0.2	V
BCD Sink Current at Pins 1, 2, 15, 16	V_{BCD} ≥ 0.5V, at Logic Zero State	0.4	1.6	-	mA
Digit Select Sink Current at Pins 3, 4, 5	V_{DIGIT} Select = 4V at Logic Zero State	1.6	2.5	-	mA
Zero Temperature Coefficient	V_I = 0V, Zero Pot Centered	-	10	-	µV/°V
Gain Temperature Coefficient	V_I = 900mV, Gain Pot = 2.4kΩ	-	0.005	-	%/°C

NOTES:

1. Apply zero volts across V_{11} to V_{10}. Adjust zero potentiometer to give 000mV reading. Apply 900mV to input and adjust gain potentiometer to give 900mV reading.
2. Linearity is measured as a difference from a straight line drawn through zero and positive full scale. Limits do not include ±0.5 count bit digitizing error.
3. For applications where low input pin 10 is not operated at pin 7 potential, a return path of not more than 100kΩ resistance must be provided for input bias currents.
4. The common mode input voltage above ground cannot exceed +0.2V if the full input signal range of 999mV is required at pin 11. That is, pin 11 may not operate higher than 1.2V positive with respect to ground or 0.2V negative with respect to ground. If the maximum input signal is less than 999mV, the common mode input voltage may be raised accordingly.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Timing Diagram

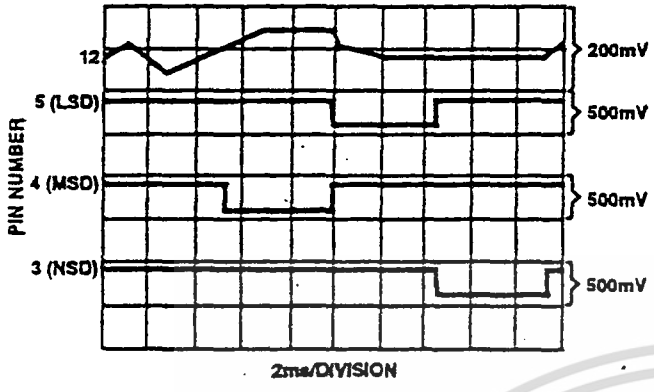


FIGURE 1. HIGH SPEED MODE

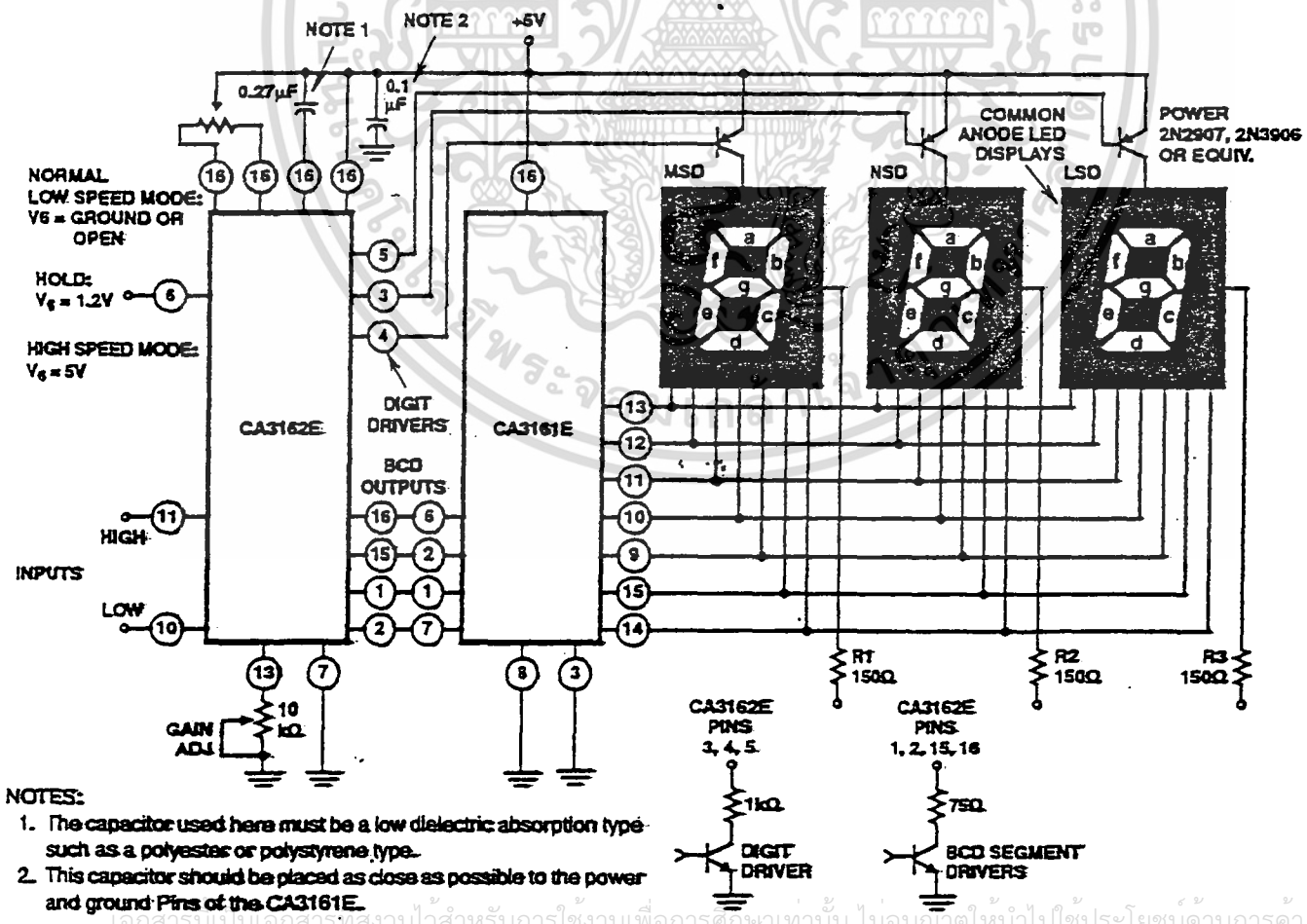
Detailed Description

The Functional Block Diagram of the CA3162E shows the V/I converter and reference current generator, which is the heart of the system. The V/I converter converts the input voltage applied between pins 10 and 11 to a current that charges the integrating capacitor on pin 12 for a predetermined time interval. At the end of the charging interval, the V/I converter is disconnected from the integrating capacitor, and a band gap reference constant current source of opposite polarity is connected. The number of clock counts that elapse before the charge is restored to its original value is a direct measure of the signal induced current. The restoration is sensed by the comparator, which in turn latches the counter. The count is then multiplexed to the BCD outputs.

The timing for the CA3162E is supplied by a 786Hz ring oscillator, and the input at pin 6 determines the sampling rate. A 5V input provides a high speed sampling rate (96Hz), and grounding or floating pin 6 provides a low speed (4Hz) sampling rate. When pin 6 is fixed at +1.2V (by placing a 12K resistor between pin 6 and the +5V supply) a "hold" feature is available. While the CA3162E is in the hold mode, sampling continues at 4Hz but the display data are latched to the last reading prior to the application of the 1.2V. Removal of the 1.2V restores continuous display changes. Note, however, that the sampling rate remains at 4Hz.

Figure 1 shows the timing of sampling and digit select pulses for the high speed mode. Note that the basic A/D conversion process requires approximately 5ms in both modes.

The "EEE" or "—" displays indicate that the range of the system has been exceeded in the positive or negative direction, respectively. Negative voltages to -99mV are displayed with the minus sign in the MSD. The BCD code is 1010 for a negative overrange (—) and 1011 for a positive overrange (EEE).



NOTES:

1. The capacitor used here must be a low dielectric absorption type such as a polyester or polystyrene type.
2. This capacitor should be placed as close as possible to the power and ground Pins of the CA3161E.

FIGURE 2. BASIC DIGITAL READOUT SYSTEM USING THE CA3162E AND THE CA3161E

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

CA3162E Liquid Crystal Display (LCD) Application

Figure 3 shows the CA3162E in a typical LCD application. LCDs may be used in favor of LED displays in applications requiring lower power dissipation, such as battery-operated equipment, or when visibility in high-ambient-light conditions is desired.

Multiplexing of LCD digits is not practical, since LCDs must be driven by an AC signal and the average voltage across each segment is zero. Three CD4056B liquid-crystal decoder/drivers are therefore used. Each CD4056B contains an input latch so that the BCD data for each digit may be latched into the decoder using the inverted digit-select outputs of the CA3162E as strobes.

The capacitors on the outputs of inverters G3 and G4 filter out the decode spikes on the MSD and NSD signals. The capacitors and pull-up resistors connected to the MSD, NSD and LSD outputs are there to shorten the digit drive signal thereby providing proper timing for the CD4056B latches.

Inverters G1 and G2 are used as an astable multivibrator to provide the AC drive to the LCD backplane. Inverters G3, G4 and G5 are the digit-select inverters and require pull-up resistors to interface the open-collector outputs of the CA3162E to CMOS logic. The BCD outputs of the CA3162E may be connected directly to the corresponding CD4056B inputs (using pull-up resistors). In this arrangement, the CD4056B decodes the negative sign (-) as an "L" and the positive overload indicator (E) as an "H".

The circuit as shown in Figure 3 using G7, G8 and G9 will decode the negative sign (-) as a negative sign (-), and the positive overload indicator (E) as "H".

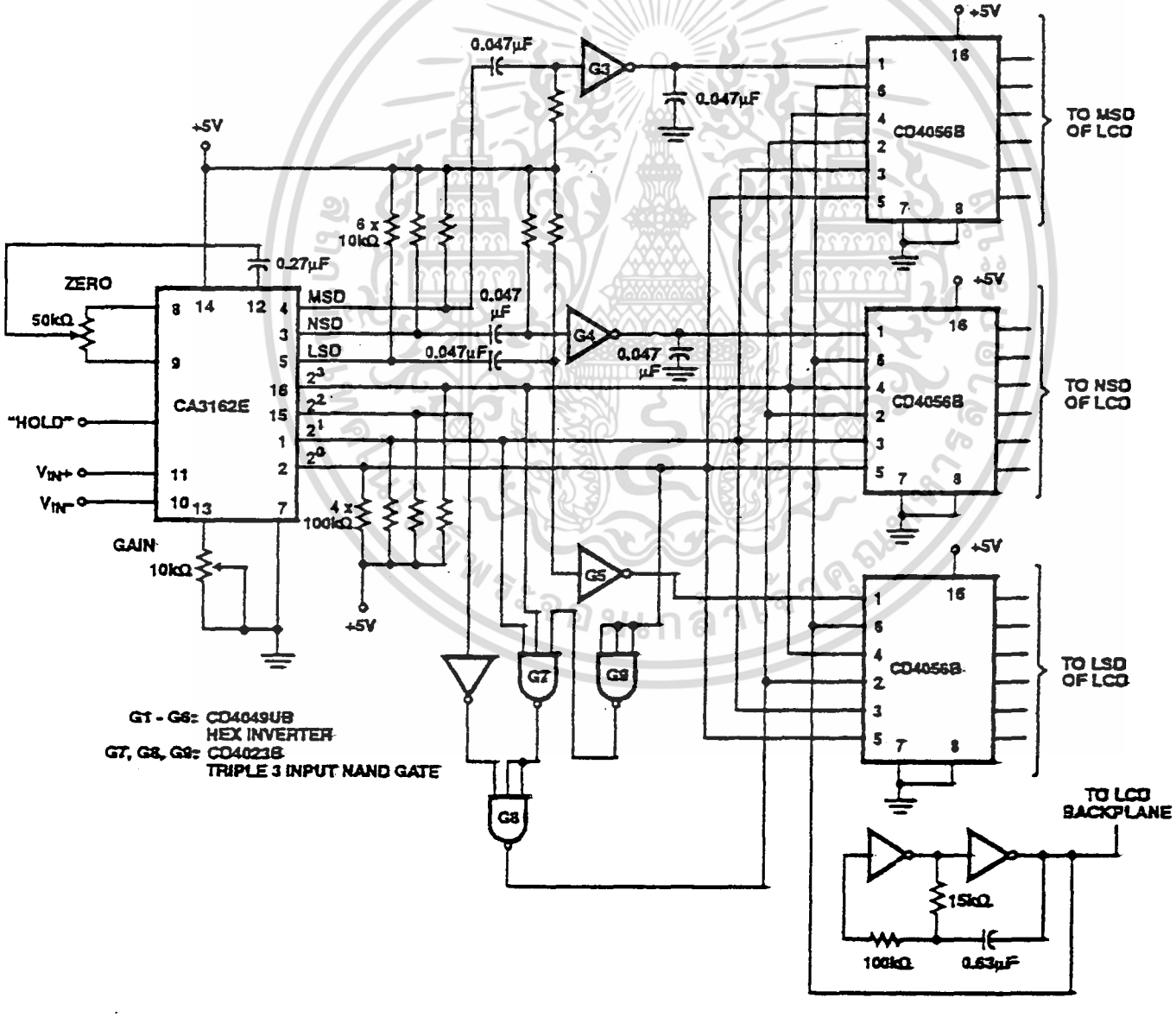


FIGURE 3. TYPICAL LCD APPLICATION

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

:162E Common-Cathode, LED Display Application

re 4 shows the CA3162E connected to a CD4511B
xde/driver to operate a common-cathode LED display.
ce the CA3161E, the CD4511B remains blank for all
) codes greater than nine. After 999mV the display
ks rather than displaying EEE, as with the CA3161E.
n displaying negative voltage, the first digit remains
k, instead of (-), and during a negative or positive over-
e the display blanks.

The additional logic shown within the dotted area of Figure 4 restores the negative sign (-), allowing the display of negative numbers as low as -99mV. Negative overrange is indicated by a negative sign (-) in the MSD position. The rest of the display is blanked. During a positive overrange, only segment b of the MSD is displayed. One inverter from the CD4049B is used to operate the decimal points. By connecting the inverter input to either the MSD or NSD line either DP1 or DP2 will be displayed.

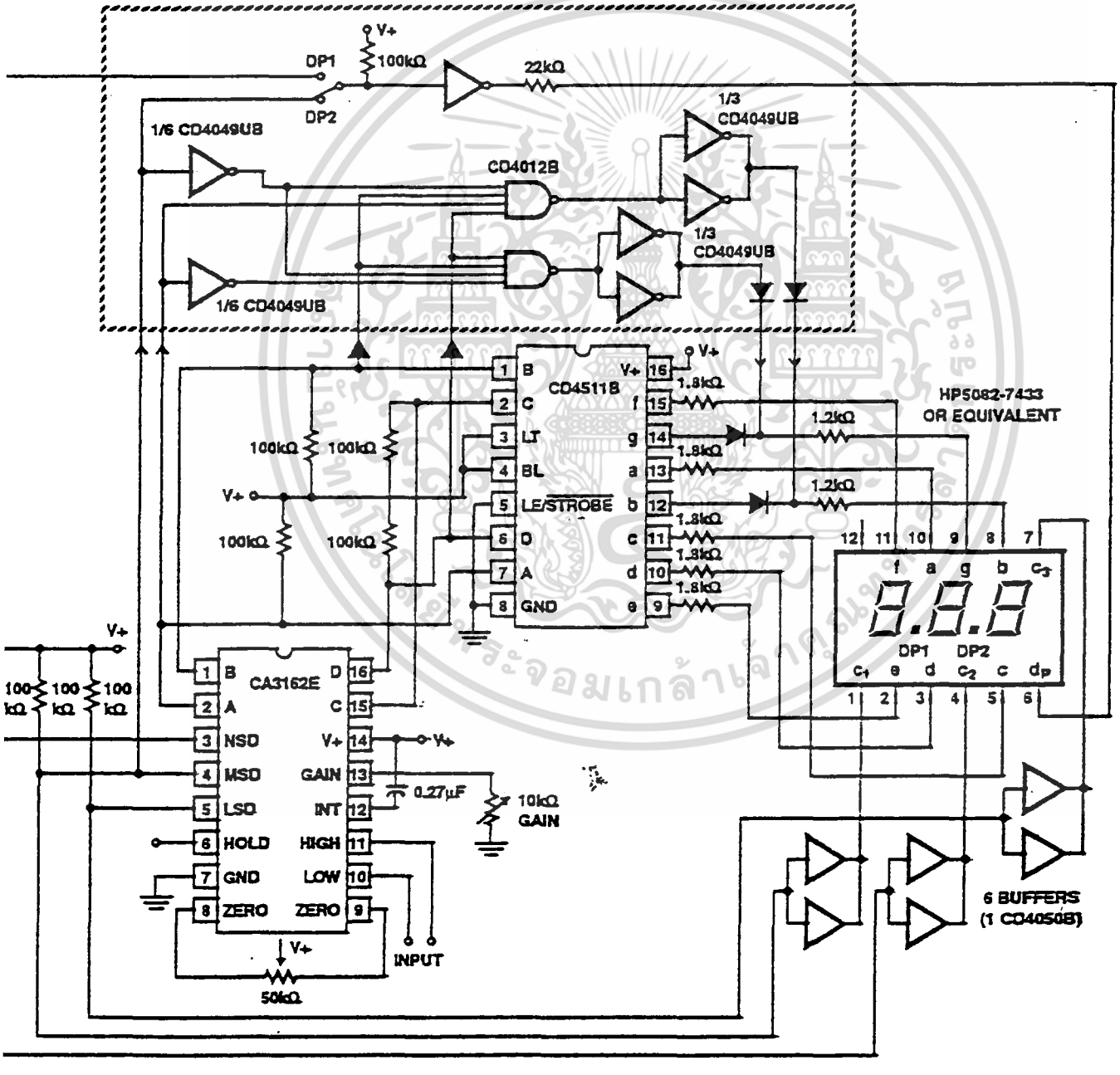


FIGURE 4. TYPICAL COMMON-CATHODE LED APPLICATION

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

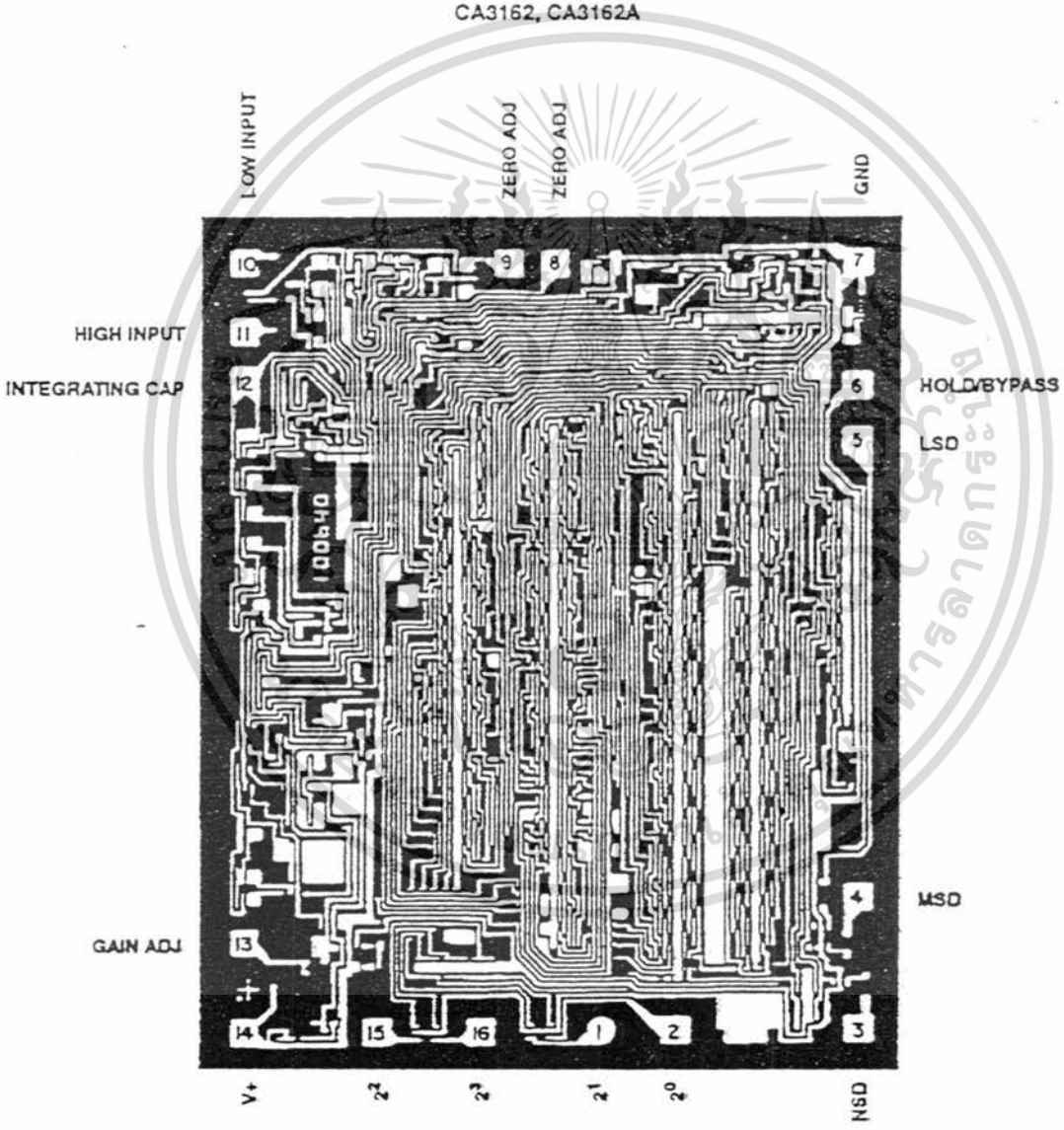
Die Characteristics

DIE DIMENSIONS:
101 x 124 x 20 ± 1mils

METALLIZATION:
Type: Al
Thickness: 17.5kÅ ± 2.5kÅ

GLASSIVATION:
Type: 3% PSG
Thickness: 13kÅ ± 2.5kÅ

Metallization Mask Layout



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เอกสารอ้างอิง

1. Fred Halsall , Data Communications , Computer Networks and Open Systems , 3rd ed. , pp. 73 - 112 , Addision - Wesley , Wokingham , 1992
2. John G. Proakis , Digital Communication , pp. 139 - 200 , McGraw - Hill , Auckland , 1987
3. Kenneth J. Ayala , The 8051 Microcontroller Architecture , Programming , and Applications , pp. 15 - 43 , West Publishing Company , ST. Paul , 1991
4. น.ต.ดร.ไพศาล สงวนหมู่ , รศ.ยืน ภู่วรวรรณ , การสื่อสารข้อมูล และไมโครคอมพิวเตอร์เนตเวอร์ค , หน้า 40 - 62 , บริษัท ซีเอ็ดยูเคชัน จำกัด , กรุงเทพมหานคร , 2529
5. วิสุทธิ์ อัศวนนทวงศ์ , " พื้นฐาน A/D และ D/A คอนเวอร์ทเตอร์ " วารสารคอมพิวเตอร์อิเล็กทรอนิคส์เวิลด์ เล่มที่ 135 : หน้า 102-115

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ประวัติผู้เขียน

นางสาววันเพ็ญ สกุลกิจไพบูลย์ เกิดวันที่ 29 มกราคม พ.ศ. 2518 สำเร็จการศึกษาระดับมัธยมศึกษาตอนต้นและระดับมัธยมศึกษาตอนปลาย จากโรงเรียนนาคประสิทธิ์ นครปฐม แล้วเข้าศึกษาต่อในระดับปริญญาตรีสาขาฟิสิกส์ประยุกต์ คณะวิทยาศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง และสำเร็จการศึกษาในปีการศึกษา 2538

นายศุภกัลย์ วัฒนการุณ เกิดวันที่ 30 สิงหาคม พ.ศ. 2517 สำเร็จการศึกษาระดับมัธยมศึกษาตอนต้นและระดับมัธยมศึกษาตอนปลาย จากโรงเรียนเตรียมอุดมศึกษา พัฒนาการ กรุงเทพ ฯ แล้วเข้าศึกษาต่อในระดับปริญญาตรีสาขาฟิสิกส์ประยุกต์ คณะวิทยาศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง และ สำเร็จการศึกษาในปีการศึกษา 2538

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้