สำนักหอสมุดกลาง พระจอมเกล้าลาดกระบัง

เครื่องควบคุมไฟเวที

STAGE LIGHTING CONTROL SYSTEM

โดย

นายมโน หาญดำรงค์รักษ์

นายสุปรีชา พรหมรัตน์

เลขหมู่.....
เลขทะเบียน 42664
วัน, เดือน, ปี 6 มิ.ย. 2545

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

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาอุตสาหกรรมศาสตรบัณฑิต

สาขาวิชาเทคโนโลยีอิเล็กทรอนิกส์

คณะวิศวกรรมศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2544

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หัวข้อปริญญานิพนธ์ เครื่องควบคุมไฟเวที (STAGE LIGHTING CONTROL SYSTEM)

โดย นายมโน หาญดำรงค์รักษ์ เลขประจำตัว 42015521

นายสุปรีชา พรหมรัตน์ เลขประจำตัว 42015536

อาจารย์ที่ปรึกษา ผู้ช่วยศาสตราจารย์ อุทัย ศรีธีระวิโรจน์

ภาควิชา เทคนิคอุตสาหกรรม

ปีการศึกษา 2544

คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบังอนุมัติให้นับปริญญานิพนธ์ฉบับนี้ เป็นส่วนหนึ่งของการศึกษา ตามหลักสูตรปริญญา**อุตสาหกรรมศาสตร์บัณฑิต**

คณะกรรมการสอบปริญญานิพนธ์

.....ประธานกรรมการ

(.....)

.....กรรมการ

(.....)

.....กรรมการ

(.....)

.....กรรมการ

(.....)

.....กรรมการ

(.....)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หัวข้อปริญญานิพนธ์ เครื่องควบคุมไฟเวที

โดย นายมโน หาญดำรงค์รักษ์
นายสุปรีชา พรหมรัตน์

อาจารย์ที่ปรึกษา ผู้ช่วยศาสตราจารย์ อุทัย ศรีธีระวิโรจน์
ปีการศึกษา 2544

บทคัดย่อ

ปริญญานิพนธ์ฉบับนี้นำเสนอ เครื่องควบคุมไฟเวที โดยใช้คอมพิวเตอร์เข้าช่วยในการควบคุมร่วมกับการใช้โปรโตคอลไอสแควซีบัส และการส่งข้อมูลแบบอนุกรมตามมาตรฐานอาร์เอส422 (RS-422) เพื่อนำไปใช้ในการควบคุมดิมเมอร์แพก ในส่วนของดิมเมอร์แพกนี้สามารถเพิ่มกำลังไฟฟ้าที่เอ้าท์พุทได้โดยการเปลี่ยนเอสซีอาร์ และยังมีส่วนของวงจรตรวจจับกระแสเกิน เพื่อป้องกันความเสียหายที่อาจเกิดขึ้นได้กับเอสซีอาร์ การควบคุมความสว่างสามารถทำได้โดยการควบคุมผ่านโปรแกรมใช้งานที่เขียนขึ้นโดยโปรแกรมวิชวล-เบสิก 6

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

PROJECT STAGE LIGHTING CONTROL SYSTEM

BY **Mr.Mano Harndimrongruk**

Mr.Supreecha Promrattana

ADVISOR **Asst.Prof.Uthai Sritheeravirojana**

ACADAMIC **2001**

ABSTRACT

In this thesis is presented the project of the stage lighting control system. It is controlled by computer including Protocol (I^2C BUS). The data is transferred to dimmer pack by serial port (standard RS422). Dimmer pack is controlled by RS422. In addition, dimmer pack can be increased power by using higher power SCR. Furthermore, its has the over current line detector to protect SCR. This system is controlled by Visual Basic 6.0 program.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กิตติกรรมประกาศ

ปริญญานิพนธ์ฉบับนี้ เป็นปริญญานิพนธ์ ฉบับที่ได้ร่วมมือกันทำอย่างเต็มความสามารถเพื่อให้งานออกมามีคุณภาพมากที่สุด ใช้เวลาในการค้นคว้าหาข้อมูล และรวบรวมข้อมูลทำการศึกษาเกี่ยวกับข้อมูลรวมถึงหลักการต่างๆที่เกี่ยวข้องกับปริญญานิพนธ์นี้ ซึ่งทำให้ปริญญานิพนธ์ฉบับนี้สำเร็จลุล่วงลงได้ด้วยดี ซึ่งทั้งหมดนี้ก็ด้วย คำแนะนำ ความช่วยเหลือต่างๆ และคำปรึกษาจากบุคคลหลายท่านด้วยกัน จึงขอขอบพระคุณมา ณ โอกาสนี้

ผู้ช่วยศาสตราจารย์ อุทัย ศรีธีรวิโรจน์ ที่ให้คำปรึกษา คำแนะนำ รวมถึงความช่วยเหลือต่างๆอย่างดียิ่ง

และคณาจารย์ทุกท่านที่ได้ให้การอบรมสั่งสอน แก่พวกเราจนมีความรู้ความสามารถในด้านต่างๆ และขอขอบคุณ บิดา มารดา และเพื่อนๆ ที่ช่วยเป็นกำลังใจ และให้ความช่วยเหลือในการจัดทำปริญญานิพนธ์ฉบับนี้

นาย มโน หาญดำรงค์รักษ์

นาย สุปรีชา พรหมรัตน์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ

	หน้า
บทคัดย่อ	I
ABSTRACT	II
กิตติกรรมประกาศ	III
สารบัญภาพ	IV
สารบัญตาราง	VII
บทที่ 1 บทนำ	1
บทที่ 2 ทฤษฎีที่เกี่ยวข้อง	3
2.1 ไมโครคอนโทรลเลอร์เบอร์ เอที89เอส53	3
2.1.1 ลักษณะโครงสร้างของเอที89เอส53	3
2.1.2 การใช้งานการขัดจังหวะ	5
2.1.3 การควบคุมการขัดจังหวะ	7
2.1.4 การจัดลำดับความสำคัญของการขัดจังหวะ	8
2.1.5 การใช้งานสัญญาณขอขัดจังหวะจากภายนอก	9
2.1.6 การใช้งานการขอขัดจังหวะจากตัวตั้งเวลาหมายเลขศูนย์	9
2.1.7 การใช้งานการขอขัดจังหวะจากพอร์ตอนุกรม และตัวตั้งเวลาหมายเลขหนึ่ง	10
2.1.8 การใช้งานการขอขัดจังหวะจากตัวตั้งเวลาหมายเลขสอง	10
2.2 การใช้งานตัวตั้งเวลาหรือตัวนับหมายเลขสอง	10
2.2.1 การทำงานแบบแคปเจอร์	12
2.2.2 การทำงานแบบกำหนดค่าใหม่อัตโนมัติ	13
2.2.3 การทำงานเป็นตัวสร้างอัตราการรับส่งข้อมูล	14
2.3 การใช้งานพอร์ตอนุกรม	14
2.3.1 รีจิสเตอร์ควบคุมการทำงานของพอร์ตอนุกรม	15
2.3.2 การทำงานของพอร์ตอนุกรมในรูปแบบต่างๆ	16

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หน้า

2.3.3 การกำหนดอัตรารับส่งของพอร์ตอนุกรม 17

2.3.4 การใช้ตัวตั้งเวลาหรือตัวนับหมายเลขหนึ่ง สร้างอัตราการรับส่งข้อมูล 17

2.3.5 การใช้ตัวตั้งเวลาหรือตัวนับหมายเลขสอง สร้างอัตราการรับส่งข้อมูล 18

2.4พอร์ตอนุกรมของคอมพิวเตอร์ 19

2.4.1 มาตราฐานพอร์ตอนุกรมแบบ RS – 232 19

2.4.2 ขั้วต่อสำหรับพอร์ต RS – 232และการเชื่อมต่อ 20

2.4.3 การรับส่งข้อมูลแบบไม่เข้าจังหวะ 22

2.4.4 ชนิดของยูเออาร์ที 23

2.4.5 วงจรภายในและรีจิสเตอร์ของพอร์ตอนุกรม RS – 232 23

2.4.6 ลักษณะสัญญาณอินพุตและเอาต์พุตของพอร์ต RS – 232 25

2.4.7 หมายเลขตำแหน่งของพอร์ตอนุกรม 25

2.5 ระบบเชื่อมต่อแบบ RS – 422 26

2.6 การเขียนโปแกรมเพื่อใช้งานพอร์ตอนุกรมของคอมพิวเตอร์ 27

2.6.1 การหาค่าตำแหน่งแอดเดรสของพอร์ตอนุกรม 27

2.6.2 การกำหนดค่าเริ่มต้นให้กับพอร์ตอนุกรม 30

2.6.3 การรับและส่งข้อมูลแบบอนุกรม 31

2.7 คอนโทรล เอ็มเอสคอมมิวนิเคชั่นคอนโทรลหรือ เอ็มเอสคอมม์ 32

2.7.1 CommPort 32

2.7.2 Setting 33

2.7.3 PortOpen 34

2.7.4 Input 34

2.7.5 InBufferCount 35

2.7.6 InBufferSize 35

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หน้า

2.7.7 InputLen 36

2.7.8InputMode 36

2.7.9Output 37

2.7.10OutBufferCont 38

2.7.11OutBufferSize 38

2.7.12ParityReplace 38

2.7.13DTREnable 39

2.7.14RTSEnable 39

2.7.15EOFEnable 39

2.7.16CTSHolding 40

2.7.17CDHolding 40

2.7.18DSRHolding 40

2.7.19Handshaking 41

2.7.20Break 41

2.8 เหตุการณ์ oncomm 42

2.8.1ค่าคงที่คุณสมบัติของคอนโทรล เอ็มเอสคอมม์ 43

2.9 การใช้ เอ็มเอสคอมม์ เพื่อการติดต่ออุปกรณ์ภายนอก 44

2.10ระบบส่งข้อมูลแบบแบบไอแสควซี (I^2C) 45

2.10.1คุณสมบัติทั่วไปของระบบเส้นทางแบบไอแสควซี 45

2.10.2หลักการของระบบไอแสควซี 45

2.10.3สภาวะที่เกิดขึ้นบนบัสไอแสควซี 47

2.10.4การทำงานบนบัส ไอแสควซี 48

2.11ขยายพอร์ตอินพุตเอาต์พุตให้แก่พอร์ตอนุกรม

ด้วยระบบบัส ไอแสควซี 50

2.11.1ข้อมูลเบื้องต้นของ PCF857/8574A 50

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หน้า

2.12 การควบคุมแรงดันไฟฟ้ากระแสสลับ 52

2.12.1 การควบคุมแบบปิด-เปิด 52

2.12.2 การควบคุมมุมเฟส 55

2.13 การออกแบบและการพันหม้อแปลงกระแส 63

บทที่ 3บล็อกไดอะแกรมและการทำงานของเครื่องควบคุมไฟเวที 65

3.1 ภาค คีย์บอร์ด เมาส์ โปแกรมยูสเซอร์อินเตอร์เฟส

และคอมพิวเตอร์ 65

3.2 ภาคการติดต่อระหว่างคอมพิวเตอร์กับชุดรับส่งข้อมูล 66

3.3 ภาคชุดรับส่งข้อมูลและปฏิบัติคำสั่ง 67

3.4 ภาคสวิตซ์กำลัง 69

3.4.1 วงจรแปลงสัญญาณดิจิตอลเป็นอานาลอก 69

3.4.2 การติดต่อกับ TDA844 70

3.5 วงจรควบคุมความสว่าง 71

3.6 การนำไอซีสำเร็จรูปมาใช้ในส่วนของวงจรควบคุม 71

3.6.1 คุณสมบัติของ TCA 785 71

3.6.2 การทำงานของ TCA-785 72

3.6.3 ลักษณะของพัลส์ด้านออกชนิดต่างๆ 74

3.6.4 หลักการทำงานของชุดควบคุมความสว่าง 75

3.6.5 การหาขั้วของหม้อแปลง 76

3.6.6 การหาค่า RG ที่ใช้ในการจำกัดกระแสเกต 77

3.7 ซอฟต์แวร์ส่วนของ Main CPU 80

3.8 ซอฟต์แวร์ส่วนของหน้าจอควบคุมที่คอมพิวเตอร์ 80

บทที่ 4 สรุปวิจารณ์ผลการทดลองและแนวทางการพัฒนา 81

4.1 ผลการทดลองวงจรควบคุมความสว่าง 81

4.2 หน้าจอควบคุมที่คอมพิวเตอร์ และการใช้งาน 84

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หน้า

4.2.1 ขั้นตอนการใช้งาน 85

4.3 สรุป 85

4.4 แนวทางการพัฒนา 85

เอกสารอ้างอิง 87

ภาคผนวก ก. วงจรใช้งานทั้งหมดของเครื่องควบคุมไฟเวท ก-1

ภาคผนวก ข. แผ่นวงจรพิมพ์ ข-1

ภาคผนวก ค. รูปถ่ายเครื่องควบคุมไฟเวที ค-1

ภาคผนวก ง. โปรแกรมในส่วนของไมโครคอนโทรลเลอร์ และ
โปรแกรมวิชวลเบสิก-6 ง-1

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญภาพ

หน้า

รูปที่ 1 บล็อกไดอะแกรมส่วนสำคัญของ เครื่องควบคุมไฟเวที 2

รูปที่ 2 ลักษณะการจัดขาแบบ พีดิพ 4

รูปที่ 3 แหล่งกำเนิดสัญญาณการขัดจังหวะของ เอที89เอส53 6

รูปที่ 4 ตำแหน่งของบิตต่างๆ ในรีจิสเตอร์ IE 7

รูปที่ 5 ตำแหน่งของบิตต่างๆ ในรีจิสเตอร์ IP 8

รูปที่ 6 ตำแหน่งบิตภายในรีจิสเตอร์ควบคุมตัวตั้งเวลาหรือตัวนับหมายเลข 2 10

รูปที่ 7 การทำงานของตัวตั้งเวลาหรือตัวนับหมายเลข 2 ในแบบแคปเจอร์ 12

รูปที่ 8 แสดงตำแหน่งบิตและการใช้งานของ รีจิสเตอร์ T2MOD 13

รูปที่ 9 การทำงานแบบนับลงแบบกำหนดค่าใหม่อัตโนมัติ
ของตัวตั้งเวลาหรือตัวนับหมายเลข 2 14

รูปที่10 การทำงานแบบนับขึ้นแบบกำหนดค่าใหม่อัตโนมัติ
ของตัวตั้งเวลาหรือตัวนับหมายเลข 2 14

รูปที่11 ตำแหน่งบิตต่างๆ ภายในรีจิสเตอร์ควบคุมการทำงานของพอร์ตอนุกรม 14

รูปที่12 การทำงานของวงจรกำเนิดอัตราการรับส่งข้อมูล
ของตัวตั้งเวลาหรือตัวนับหมายเลข 2 19

รูปที่13 ลักษณะตำแหน่งการวางขาของขั้วต่อแบบ DB-9 และ DB-25 20

รูปที่14 แสดงการเชื่อมต่ออุปกรณ์ภายนอกกับคอมพิวเตอร์ 22

รูปที่15 แผนผังการทำงานภายในของพอร์ตอนุกรมของเครื่องคอมพิวเตอร์ 24

รูปที่16 ผังแสดงโครงสร้างทางวงจรของพอร์ตอนุกรม 25

รูปที่17 การใช้โปแกรม ดีบักเพื่อหาค่าตำแหน่งของ พอร์ตอนุกรม 27

รูปที่18 แสดงตำแหน่งของพอร์ตอนุกรมบนระบบปฏิบัติการ วิน โดวส์ 98 28

รูปที่19 แสดงรายละเอียดของพอร์ตอนุกรม COM2 บนระบบปฏิบัติการ วิน โดวส์ 98 29

รูปที่20 โครงสร้างของวงจรเอาต์พุตของอุปกรณ์ที่ใช้การเชื่อมต่อบนระบบบัสไอแสควซี 46

รูปที่21 แสดงการเชื่อมต่ออุปกรณ์บนระบบบัสไอแสควซี ที่ใช้ไฟเลี้ยงไม่เท่ากัน 46

รูปที่22 การต่อตัวต้านทานเพื่อป้องกันแรงดันกระชากที่อาจปะปนเข้ามา
ในสายสัญญาณข้อมูลอนุกรมและสายสัญญาณนาฬิกา ของระบบบัสแบบไอแสควซี 47

รูปที่23 แสดงสถานะต่างๆ บนระบบบัสไอแสควซี 48

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่24 แสดงรูปแบบของข้อมูลอนุกรมที่เกิดขึ้นในการอ้างถึงแบบ 7 บิต ของระบบบัสแบบ-ไอแสควซี 49
รูปที่25 แสดงรูปแบบของของข้อมูลข้อมูลอนุกรมที่เกิดขึ้นในการอ้างถึงแบบ 10 บิตบนบัสแบบไอแสควซี 49
รูปที่26 การจัดขาของ PCF8574/8574A และหน้าที่การทำงานของแต่ละขา 51
รูปที่27 วงจรภายในของขาพอร์ตของวงจรรวม PCF8574/8574A 52
รูปที่28 วงจรควบคุมแรงดันไฟสลับโดยวิธีควบคุมแบบ ปิด-เปิด 53
รูปที่29 วงจรควบคุมไฟสลับที่ใช้หลักการควบคุมมุมเฟส ทำการควบคุมทางเดียว และโหลดเป็นตัว ต้านทานที่มุมจุดชนวน (α) 56
รูปที่30 วงจรควบคุมแรงดันไฟสลับ 1 เฟส ควบคุมสองทางด้วยวิธีควบคุมเฟส เมื่อจุดชนวนเกตของ T_1ที่มุมαและ T_2ที่มุม $\pi+\alpha$มีโหลดเป็นตัวต้านทานเพียงเดียว 56
รูปที่31 แสดงวงจรควบคุมไฟสลับแบบอื่นๆ 58
รูปที่32 วงจรควบคุมไฟสลับ 1 เฟส เมื่อโหลดเป็นตัวต้านทานและตัวเหนี่ยวนำและจุดชนวนเกตที่มุม $\alpha > \Phi$ ด้วยพัลส์เดี่ยว และพัลส์แบบอื่น ๆ 60
รูปที่33 แสดงกราฟของ I_Nและ I_{RN} ที่ใช้คำนวณค่ากระแสไทรสเตอร์ 62
รูปที่34 แสดงบล็อกไดอะแกรมหลักของเครื่องควบคุมไฟเวที 65
รูปที่35 การแปลงสัญญาณ RS-232 เป็น RS-422 66
รูปที่36 การเชื่อมต่อภายในของบล็อกไดอะแกรมในชุดลิงค์การ์ดและปฏิบัติคำสั่ง 68
รูปที่37 การเชื่อมต่อภายในของบล็อกไดอะแกรมในชุดรับส่งข้อมูลและปฏิบัติคำสั่ง 68
รูปที่38 บล็อกไดอะแกรมภายใน TDA8444 70
รูปที่39 รูปแบบข้อมูลที่ใช้ติดต่อกับ TDA8444 70
รูปที่40 บล็อกไดอะแกรมการเชื่อมต่อภายในของชุดเพาเวอร์แพก 72
รูปที่41 แสดงโครงสร้างภายในของวงจรรวมเบอร์ TCA785 73
รูปที่42 รูปคลื่นสัญญาณด้านเข้าและด้านออกที่สำคัญของวงจรรวมเบอร์ TCA-785 75
รูปที่43 แสดงวิธีการตรวจหาขั้วของหม้อแปลง 77
รูปที่ 44 แสดงการคำนวณหาค่า RG 77
รูปที่45 วงจรใช้งานของส่วนควบคุมกำลัง 78
รูปที่46 วงจรใช้งานในส่วนของสวิตช์กำลัง 79
รูปที่47 รูปคลื่นแรงรันตกคร่อมโหลดเมื่อกระตุ้นเอสซีอาร์ที่มุมประมาณ 0 องศา 81
รูปที่48 รูปคลื่นแรงรันตกคร่อมโหลดเมื่อกระตุ้นเอสซีอาร์ที่มุมประมาณ 45 องศา 82

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่49 รูปคลื่นแรงรันตกคร่อมโหลดเมื่อกระตุ้นเอสซีอาร์ที่มุมประมาณ 90 องศา 82
รูปที่50รูปคลื่นแรงรันตกคร่อมโหลดเมื่อกระตุ้นเอสซีอาร์ที่มุมประมาณ 135 องศา 83
รูปที่51รูปคลื่นแรงรันตกคร่อมโหลดเมื่อกระตุ้นเอสซีอาร์ที่มุมประมาณ 180 องศา 83
รูปที่52แสดงภาพหน้าจอควบคุมที่คอมพิวเตอร์ 84
รูปที่53วงจรแปลงสัญญาณอนุกรมเป็นสัญญาณแบบไอแสควซีบัส ก-1
รูปที่54วงจรแปลงสัญญาณอนุกรมRS-232เป็นRS-422 ก-2
รูปที่55วงจรหลักของชุดควบคุมเพาเวอร์แพก ก-3
รูปที่56วงจรชุดรับข้อมูลควบคุมไฟและตรวจสอบกระแส ก-4
รูปที่57ชุดจ่ายไฟสำหรับภาคควบคุมกำลังไฟ ก-5
รูปที่58วงจรตรวจจับกระแสเกิน ก-6
รูปที่59แผ่นวงจรพิมพ์ชุดควบคุมหลัก ข-1
รูปที่60แผ่นวงจรพิมพ์แปลงสัญญาณอนุกรมRS-232เป็นRS-422 ข-2
รูปที่61แผ่นวงจรพิมพ์แปลงสัญญาณอนุกรมเป็นสัญญาณแบบไอแสควซีบัส ข-3
รูปที่62แผ่นวงจรพิมพ์ชุดควบคุมการสวิตช์และสวิตช์ ข-4
รูปที่63แผ่นวงจรพิมพ์ชุดแปลงสัญญาณดิจิตอลเป็นอนาลอก ข-5
รูปที่64รูปถ่ายด้านหน้าของชุดดิมเมอร์แพก ค-1
รูปที่65รูปถ่ายด้านหลังของชุดดิมเมอร์แพก ค-2
รูปที่66รูปถ่ายด้านในของชุดดิมเมอร์แพก ค-3
รูปที่67รูปถ่ายด้านหน้าของชุดคอนโทรลยูนิต ค-4
รูปที่68รูปถ่ายด้านหลังของชุดคอนโทรลยูนิต ค-5
รูปที่69รูปถ่ายด้านในของชุดคอนโทรลยูนิต ค-6
รูปที่70รูปถ่ายด้านหน้าของชุดคอนเวอร์เตอร์ ค-7
รูปที่71รูปถ่ายด้านหลังของชุดคอนเวอร์เตอร์ ค-8
รูปที่72รูปถ่ายด้านในของชุดคอนโทรลยูนิต ค-9

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญตาราง

ตารางที่ 1. รูปแบบของการรับส่งข้อมูลของพอร์ตอนุกรม กำหนดโดย
บิต SM0 และ SM1 15
ตารางที่ 2 รายละเอียดของขาต่างๆที่มีใช้อยู่ในปัจจุบัน 21
ตารางที่ 3 แสดงข้อมูลในตำแหน่ง 0000:0411h ที่ใช้แจ้งจำนวนพอร์ตอนุกรม 26
ตารางที่ 4 ค่าคงที่สำหรับคุณสมบัติ Handshake 43
ตารางที่ 5 ค่าคงที่สำหรับคุณสมบัติ OnComm 43
ตารางที่ 6 ค่าคงที่สำหรับคุณสมบัติ Error 44
ตารางที่ 7 ค่าคงที่สำหรับคุณสมบัติ InputMode 44
ตารางที่ 8 แสดงความสัมพันธ์ของความกว้างของพัลส์ด้านออกกับค่า C_{12} 74

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 1

บทนำ

ในปัจจุบันความบันเทิงมีส่วนช่วยให้ผู้คนได้ผ่อนคลายความเครียดลงได้มาก ความบันเทิงที่เป็นที่นิยมในปัจจุบันก็คือ ดนตรี ซึ่งสังเกตได้จาก จำนวนผลงานที่ออกมาบ่อยขึ้น และจำนวนศิลปินที่เพิ่มขึ้นด้วย เมื่อมีศิลปิน มีดนตรี สิ่งที่ต้องมีตามมาก็คือ คอนเสิร์ต ไม่ว่าจะเป็นคอนเสิร์ตเล็กหรือใหญ่ ก็จะต้องใช้เวทีแสดง และที่ขาดไม่ได้เช่นกันก็คือ ระบบไฟแสงสี เพื่อเพิ่มสีสันให้น่าสนใจ

ปัจจุบันระบบแสงสีส่วนใหญ่จะต้องมีเครื่องควบคุม ซึ่งควบคุมเหล่านี้จะมีส่วนประกอบหลัก 2 ส่วนคือ คอนโซลสำหรับควบคุม และ ชุดขับกำลัง หรือ เพาเวอร์แพก (Power pack)ที่อยู่ด้านหน้าเวที อุปกรณ์ทั้ง 2ส่วนนี้ส่วนใหญ่ถ้าเป็นระบบแสงสีของเวทีคอนเสิร์ต ระดับกลาง ถึงระดับสูง ในประเทศไทย นิยมใช้ชุดควบระบบไฟแสงสีที่สั่งซื้อมาจากต่างประเทศ ซึ่งมีราคาค่อนข้างสูงมาก หรือหากเป็นงานเวทีระดับล่าง ก็มักจะใช้ชุดควบคุมอัตโนมัติที่มีจำหน่ายทั่วไป แต่มีข้อจำกัดในเรื่องของรูปแบบของแสงสีที่ได้ ที่ไม่สามารถปรับแต่งเพิ่มเติมได้มากนัก ซึ่งจะทำให้เกิดความจำเจ

ดังนั้นหากเราใช้คอมพิวเตอร์ เข้ามาจัดการตรงส่วนนี้ได้ จะทำให้ชุดควบคุมระบบไฟแสงสีนี้มีราคาถูกลงมาก เมื่อเทียบกับของต่างประเทศ ดังนั้นจึงได้คิดประดิษฐ์ ชุดควบคุมไฟเวทีคอนเสิร์ตโครงงานนี้จึงทำขึ้นเพื่อแก้ปัญหาดังกล่าว ซึ่งจะมีส่วนประกอบ 2 ส่วนเช่นเดียวกัน คือ ประกอบด้วย 1.) คอนโซลซึ่งเป็นคอมพิวเตอร์พีซีทั่วไปที่ติดตั้ง โปรแกรมยูสเซอร์อินเตอร์เฟส 2.) ชุดเพาเวอร์แพกที่ทำหน้าที่เป็นสวิตว์กำลังจำกัดแรงดันและอัตราการไหลของกระแสสู่โหลด

1.1. รายละเอียดโดยรวมของส่วนประกอบทั้ง 2 ส่วนมีดังนี้

1.1.1 หน้าจอควบคุม หรือ คอนโซล ใช้คอมพิวเตอร์พีซีที่มีระบบปฏิบัติการแบบ 32 บิต วินโดว์ 95 หรือสูงกว่าและมีโปรแกรม ยูสเซอร์อินเตอร์เฟส (User interface) ซึ่งเขียนด้วยโปรแกรม วิชวลเบสิก 6.0 ตัวโปรแกรม จะทำหน้าที่รับคำสั่งจากผู้ใช้งานที่สั่งผ่านคีย์บอร์ด หรือ เมาซ์ ของคอมพิวเตอร์

1.1.2 ชุดเพาเวอร์แพก ประกอบด้วยส่วนย่อยหลักอีก 2 ส่วนคือ

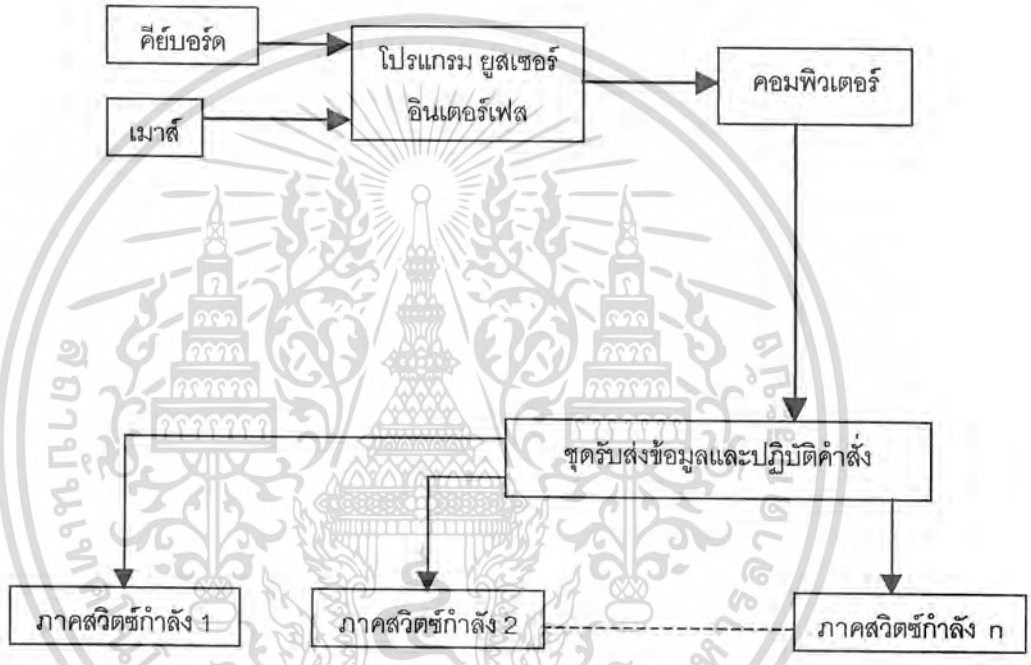
1.) ภาครับส่งข้อมูลกับคอมพิวเตอร์และปฏิบัติคำสั่งเพื่อส่งให้กับภาคสวิตซ์กำลัง อุปกรณ์ที่ทำหน้าที่หลักในภาคนี้คือ ไมโครคอนโทรลเลอร์ เบอร์ เอที89เอส53 (AT89S53)

2.) ภาคสวิตซ์กำลังทำหน้าที่รับคำสั่งจากภาครับส่งข้อมูลมาแปลงเป็นระดับแรงดันไฟ 0 ถึง 10 โวลต์ด้วยวงจร แอนาล็อกทูดิจิตอล(Analog to Digital)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

แล้วนำไปควบคุมการนำกระแสของ เอสซีอาร์ (SCR)

โพรโตคอล (Protocol) ที่ใช้ในการรับส่งข้อมูลระหว่างภาครับส่งข้อมูลกับคอมพิวเตอร์และประมวลผลคำสั่ง กับภาคสวิตช์กำลัง จะใช้โพรโตคอลที่เรียกว่า ไอแสควซีบัส (I^2C Bus) การส่งข้อมูลระหว่างคอมพิวเตอร์กับ ชุดเพาเวอร์แพก จะใช้การติดต่อแบบ อนุกรม ตามมาตรฐาน อาร์เอส-422 (RS-422) ซึ่งมีข้อดีคือรับส่งได้ไกลและมีอัตราการส่งข้อมูลสูงถึง 10 ล้านบิตต่อวินาที ผังการเชื่อมต่อของชุดควบคุมไฟเวทีแสดงดังรูปที่ 1



รูปที่ 1 บล็อกไดอะแกรมส่วนสำคัญของ เครื่องควบคุมไฟเวที

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 2
ทฤษฎีที่เกี่ยวข้อง

2.1 .ไมโครคอนโทรลเลอร์ เบอร์ เอที89เอส53

เอที89เอส53 มีสถาปัตยกรรมภายในพื้นฐานที่เหมือนกับ ไมโครคอนโทรลเลอร์เบอร์ 8051 แต่ มีข้อดีมากกว่า ดังคุณสมบัติดังต่อไปนี้

- 12 กิโลไบต์แฟลชเมมโมรี่ที่สามารถ โปรแกรมได้ด้วยวิธี เอสพีไอ(SPI) สามารถลบและเขียนใหม่ได้มากถึงประมาณ 1000 ครั้ง
- มี อีอีพีรอม (EEPROM) ขนาด 2 กิโลไบต์ในตัว สามารถลบและเขียนได้ประมาณ 1,000,000 ครั้ง
- ย่านแรงดันไฟเลี้ยง 4.0 ถึง 6.0 โวลต์
- ทำงานได้ที่สัญญาณนาฬิกาตั้งแต่ 0 ถึง 24 เมกะเฮิรตซ์ (0-24 MHz)
- สามารถล็อก แฟลชโปรแกรมเมมโมรี่เพื่อป้องกันการอ่านหรือแก้ไขโปรแกรมได้
- มีหน่วยความจำ 8 บิต ขนาด 256 กิโลไบต์
- มีตัวตั้งเวลาและตัวนับขนาด 16 บิต 3 ตัว
- มีแหล่งสร้างสัญญาณการขัดจังหวะได้ ถึง 9 แหล่ง
- มีช่องสื่อสารแบบ อะซิงโครนัสแบบ 2 ทิศทาง 1ช่อง
- สามารถใช้การอินเตอร์เฟสแบบ เอสพีไอได้มีระบบตรวจจับการทำงานของไมโครคอนโทรลเลอร์ (Watch Dog) แบบโปรแกรมได้ เพื่อป้องกันการหยุดชะงักของไมโครคอนโทรลเลอร์
- มีตัวชี้ตำแหน่งข้อมูล 2 ชุด
- สามารถสั่ง ปิดการทำงานของไมโครคอนโทรลเลอร์ ได้จาก บิตของรีจิสเตอร์พิเศษ

2.1.1 ลักษณะโครงสร้างของ เอที89เอส53

เอที89เอส53 มีรูปแบบตัวถังหลายแบบสำหรับในโครงงานนี้เลือกตัวถังแบบ พีดิพ40 (PDIP 40) ซึ่งมีขาต่อภายนอกทั้งหมด 40 ขา แบบดิพ (DIP) ดังรูปที่ 2 สำหรับหน้าที่ของขาต่าง มีดังต่อไปนี้

Vcc ขา 40 ป้อนไฟเลี้ยง +5 โวลต์

GND ขา 20 ต่อกับกราวด์ของแหล่งจ่ายไฟเลี้ยง

P0.0 ถึง P0.7 พอร์ต 0 เป็นพอร์ตขนาด 8 บิต อยู่ที่ขา 39 ถึง 32 เริ่มจาก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

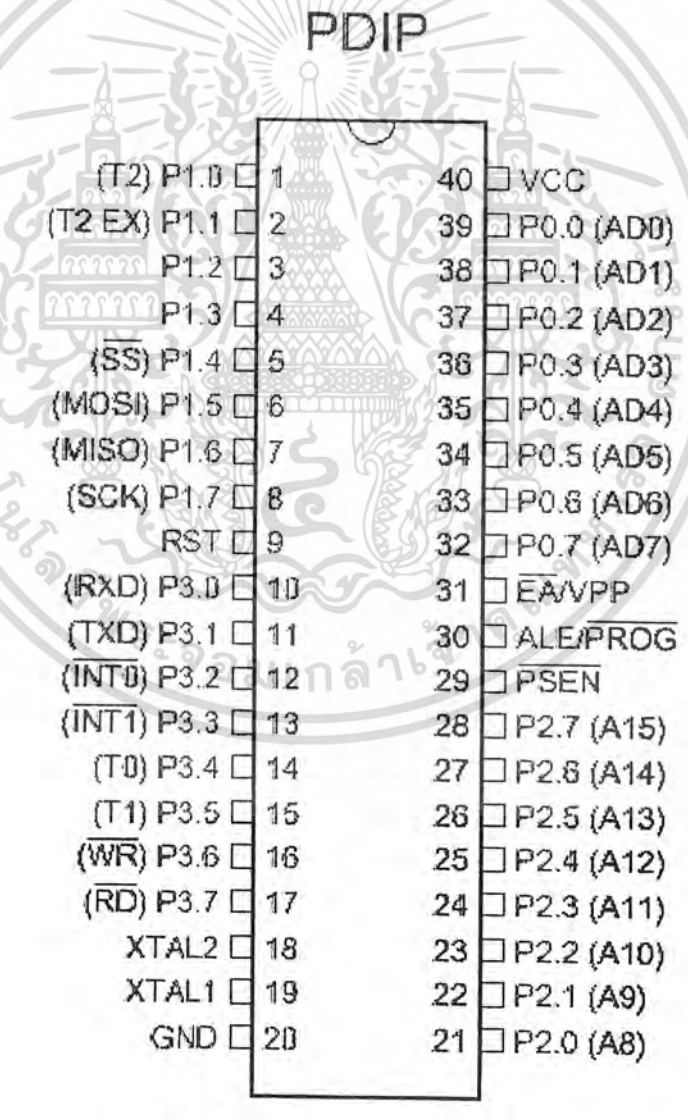
บิต 0 ถึง บิต 7 ตามลำดับ พอร์ตนี้ไม่มีการพูลอัพ (Pull Up) ภายในจึงต้องต่อตัวต้านทานพูลอัพให้เวลาใช้งาน พอร์ตนี้ใช้รับส่งข้อมูล และยังใช้อ้างตำแหน่งหน่วยความจำด้าน 8 บิตล่าง

P1.0 ถึง P1.7 พอร์ต 1 เป็นพอร์ตขนาด 8 บิต อยู่ที่ขา 1 ถึง 8 เริ่มจากบิต 0 ถึง บิต 7 ตามลำดับใช้รับส่งข้อมูลในวิธีปกติ นอกจากนี้ยังมีหน้าที่พิเศษดังต่อไปนี้คือ

P1.0 (T2) ใช้เป็นสัญญาณอินพุตของตัวนับหมายเลข 2

P1.1 (T2EX) เป็นขาควบคุมการกระตุ้นจากภายนอกให้ ตัวนับหรือตัวตั้งเวลาหมายเลข 2 เกิดการ เก็บค่าการนับ หรือตั้งเวลาใหม่

P1.4 ถึง P1.7 เป็นขาอินพุต/เอาต์พุต สำหรับการสื่อสารแบบ เอสพีไอ



รูปที่ 2 ลักษณะการจัดขาแบบ พีดิพ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

P2.0 ถึง P2.7 เป็นพอร์ตขนาด 8 บิต อยู่ที่ขา 21 ถึง 28 เริ่มตั้งแต่ P2.0 ถึง P2.7 ตามลำดับ มีหน้าที่ 2 อย่างคือ ใช้รับส่งข้อมูลแบบปกติ และใช้เป็นขาอ้างตำแหน่ง 8 บิตบน ของหน่วยความจำภายนอก

P3.0 ถึง P3.7 เป็นพอร์ตขนาด 8 บิต อยู่ที่ขา 10 ถึง 17 เริ่มตั้งแต่ P3.0 ถึง P3.7 ตามลำดับ oนอกจากใช้รับส่งข้อมูลแบบปกติแล้ว ยังมีหน้าที่พิเศษดังนี้

P3.0 /RXD เป็นขาที่ใช้รับข้อมูลแบบอนุกรม

P3.1/TXD เป็นขาที่ใช้ส่งข้อมูลแบบอนุกรม

P3.2/$\overline{\text{INT0}}$ เป็นขาใช้รับสัญญาณขัดจังหวะหมายเลข 0 จากภายนอก

P3.3/$\overline{\text{INT1}}$ เป็นขาใช้รับสัญญาณขัดจังหวะหมายเลข 1 จากภายนอก

P3.4/T0 เป็นขารับสัญญาณเข้าเข้าไปยังวงจรที่ทำหน้าที่นับสัญญาณหมายเลข 0 จะใช้นับจำนวนครั้งของสัญญาณ หรือนับสัญญาณนาฬิกา

P3.5/T1 เป็นขารับสัญญาณเข้าเข้าไปยังวงจรที่ทำหน้าที่นับสัญญาณหมายเลข 1 จะใช้นับจำนวนครั้งของสัญญาณ หรือนับสัญญาณนาฬิกา

P3.6 / $\overline{\text{WR}}$ เป็นขาควบคุมการเขียนข้อมูลลงสู่หน่วยความจำข้อมูลภายนอกตัวไมโครคอนโทรลเลอร์

P3. 7/ $\overline{\text{RD}}$ เป็นขาควบคุมการอ่านข้อมูลจากหน่วยความจำข้อมูลภายนอกตัว ไมโครคอนโทรลเลอร์

RESET ขา 9 เป็นขาที่ใช้ทำการรีเซ็ตการทำงานของ ไมโครคอนโทรลเลอร์

ALE ขา 30 ใช้ส่งสัญญาณเพื่อบอกให้อุปกรณ์ภายนอกทำการ เก็บค่าของตำแหน่งหน่วยความจำด้าน 8 บิตล่าง

$\overline{\text{PSEN}}$ ขา 29 ใช้เมื่อต้องการอ่านคำสั่งจากหน่วยความจำโปแกรมภายนอก เพื่อจะนำไปทำงาน

$\overline{\text{EA}}$ ขา 31 เป็นขาอินพุตที่ต่อเข้าไปเพื่อควบคุมการสร้างสัญญาณควบคุม PSEN

XTAL1 ขา 19 ใช้เป็นขาอินพุตของสัญญาณนาฬิกาเข้าสู่วงจรออสซิลเลเตอร์ขยายแบบกลับข้าง

XTAL2 ขา 18 ใช้เป็นขาเอาต์พุตของสัญญาณนาฬิกาจากวงจรออสซิลเลเตอร์ขยายแบบกลับข้าง

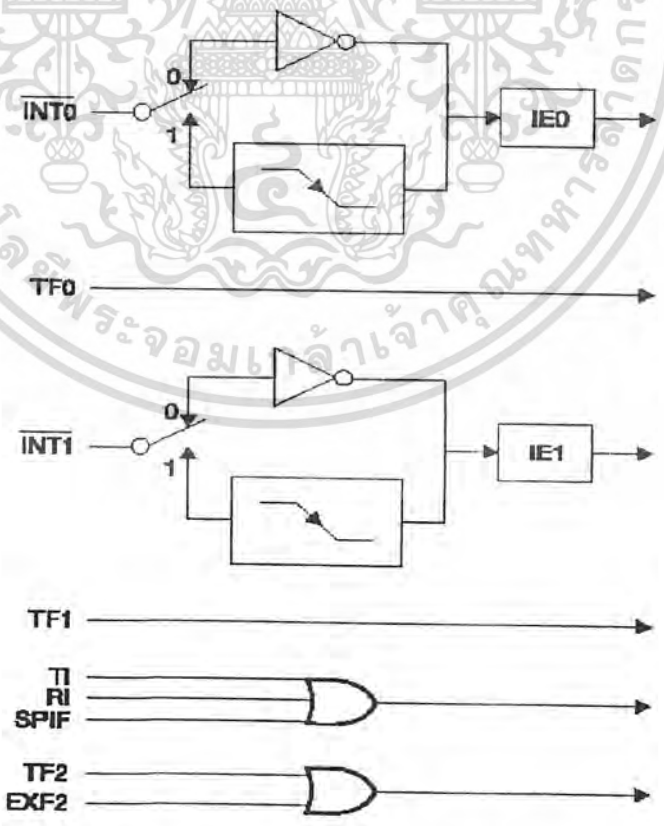
2.1.2 การใช้งานการขัดจังหวะ (Interrupt)

การติดต่อระหว่างไมโครคอนโทรลเลอร์กับอุปกรณ์ภายนอกสามารถทำได้ 2 ลักษณะคือ ใช้วิธีการโพลลิ่ง (Polling) ซึ่งไมโครคอนโทรลเลอร์ต้องคอยตรวจสอบอุปกรณ์ภาย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

นอกอยู่ตลอดเวลาว่ามีข้อมูลที่ต้องการติดต่อกับ ไมโครคอนโทรลเลอร์ หรือไม่ ทำให้ไมโครคอนโทรลเลอร์ต้องเสียเวลาไปกับการการตรวจสอบโดยไร้ประโยชน์ หากมีอุปกรณ์จำนวนมากต่ออยู่กับไมโครคอนโทรลเลอร์ การติดต่อแบบที่ สองเป็นการใช้การการขัดจังหวะ ซึ่งเป็นวิธีการที่ไมโครคอนโทรลเลอรไม่ต้องคอยตรวจสอบอุปกรณ์ภายนอกอยู่ตลอดแต่จะให้อุปกรณ์ภายนอกส่งสัญญาณการขัดจังหวะเข้ามาให้ไมโครคอนโทรลเลอร์ เมื่อต้องการที่จะติดต่อกับไมโครคอนโทรลเลอร์ เมื่อไมโครคอนโทรลเลอร์ได้รับสัญญาณการขัดจังหวะแล้วจึงจะทำการรับส่งข้อมูลกับอุปกรณ์ภายนอกนั้นๆ ต่อไป

แหล่งกำเนิดสัญญาณการขัดจังหวะของไมโครคอนโทรลเลอร์เบอร์ เอที89เอส 53 มีทั้งสิ้น 9 แหล่งดังแสดงในรูปที่ 3 ประกอบด้วย อินเตอร์รัปต์เวกเตอร์ (Interrupt Vector) 6 ชุด ได้แก่ การขัดจังหวะภายนอก 2 แหล่ง (INT0 และ INT1) การขัดจังหวะจากตัวตั้งเวลา 3 ชุด (TF0 ,TF1,ไทมเมอร์2) สำหรับไทมเมอร์2 มีแหล่งกำเนิดอยู่ 2 แหล่ง คือ TF2 และ EXF2 การขัดจังหวะจาก พอร์ตอนุกรม ซึ่งมีแหล่งขัดจังหวะ 3 แหล่ง คือ TI เป็นสัญญาณการขัดจังหวะเมื่อมีการส่งข้อมูลออก และ RI เป็นสัญญาณการขัดจังหวะเมื่อมีการรับข้อมูลเข้ามา และอีกหนึ่งแหล่งคือสัญญาณการขัดจังหวะจาก การสื่อสารแบบ เอสพีไอ

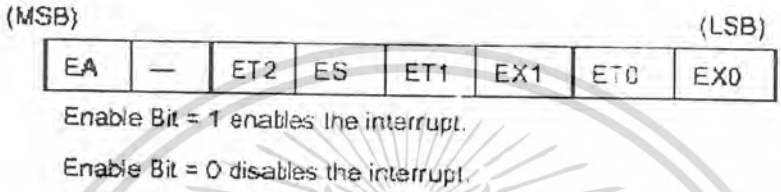


รูปที่ 3 แหล่งกำเนิดสัญญาณการขัดจังหวะของ เอที89เอส53

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.1.3 การควบคุมการขัดจังหวะ

เราสามารถควบคุมการการขัดจังหวะหรือไม่ก็ได้ โดยการเซตหรือเคลียร์บิตต่างๆที่อยู่ในรีจิสเตอร์ IE (Interrupt register) ภายในรีจิสเตอร์ IEจะมีบิต EA ที่ทำหน้าที่ควบคุมให้ ไมโครคอนโทรลเลอร์ตอบรับการขัดจังหวะหรือไม่ตอบรับทั้งหมด หากกำหนดให้บิต EA เป็น 0 ไมโครคอนโทรลเลอร์จะไม่ตอบรับการร้องขอการขัดจังหวะ ตำแหน่งของบิตต่างๆ ในรีจิสเตอร์ IE เป็นดังรูปที่ 4



รูปที่ 4 ตำแหน่งของบิตต่างๆ ในรีจิสเตอร์ IE

หน้าที่ของบิตต่างๆ ในรีจิสเตอร์ IE มีดังนี้คือ

EA หรือ **IE.7** หากกำหนดให้บิตนี้เป็น 0 จะไม่มีการตอบรับการขัดจังหวะทั้งหมด หากต้องการให้มีการตอบรับการขัดจังหวะ จะต้องเซตให้บิต EA เป็น 1 และเซตบิตควบคุมแต่ละแหล่งการขัดจังหวะที่ต้องการใช้การขัดจังหวะให้เป็น 1 ด้วย โดยใช้คำสั่ง เซตบิต หรือ เคลียร์บิต

ET2 หรือ **IE.5** ควบคุมให้ตัวตั้งเวลาหมายเลข 2 หรือ แคปเจอร์(Capture) ส่งสัญญาณการขัดจังหวะได้หรือไม่ ถ้าเซตบิตนี้เป็น 1 จะสามารถส่งสัญญาณการการขัดจังหวะได้

ES หรือ **IE.4** ควบคุมให้ พอร์ตอนุกรม ส่งสัญญาณการขัดจังหวะได้หรือไม่ ถ้าเซตบิตนี้เป็น 1 จะสามารถส่งสัญญาณการการขัดจังหวะได้

ET1 หรือ **IE.3**ควบคุมให้ตัวตั้งเวลาหมายเลข 1 ส่งสัญญาณการขัดจังหวะเมื่อถึงค่าที่ตั้งไว้ได้หรือไม่ ถ้าเซตบิตนี้เป็น 1 จะสามารถส่งสัญญาณการการขัดจังหวะได้

EX1 หรือ **IE.2**ควบคุมให้ตอบรับสัญญาณการขัดจังหวะจากภายนอกที่ขาINT 0 หรือไม่ ถ้าเซตบิตนี้เป็น 1 จะสามารถส่งสัญญาณการการขัดจังหวะได้

ET0 หรือ **IE.1**ควบคุมให้ตัวตั้งเวลาหมายเลข 0 ส่งสัญญาณการขัดจังหวะเมื่อถึงค่าที่ตั้งไว้หรือไม่ ถ้าเซตบิตนี้เป็น 1 จะสามารถส่งสัญญาณการการขัดจังหวะได้

EX0 หรือ **IE.0** ควบคุมให้ตอบรับสัญญาณการขัดจังหวะจากภายนอกที่ขาINT 1 หรือไม่ ถ้าเซตบิตนี้เป็น 1 จะสามารถส่งสัญญาณการการขัดจังหวะได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หากต้องการใช้สัญญาณการขัดจังหวะจากแหล่งใด จะต้องเซตให้บิตควบคุมเป็น 1 และต้องเซตให้บิต EA เป็น 1 ด้วย

2.1.4 การจัดลำดับความสำคัญของการขัดจังหวะ

การจัดลำดับความสำคัญของการขัดจังหวะสามารถจัดได้ 2 ระดับแตกต่างกัน โดยการเซตค่าในรีจิสเตอร์ที่ทำหน้าที่ควบคุมการ จัดลำดับความสำคัญของการขัดจังหวะ ชื่อว่า IP (Interrupt Priority Register) ในกรณีที่ไม่มีการจัดลำดับความสำคัญของการอินเตอร์รัปต์หรือจัดให้มีความสำคัญในระดับเดียวกัน ไมโครคอนโทรลเลอร์จะจัดลำดับความสำคัญของการขัดจังหวะให้อยู่ในระดับความสำคัญจากสูงไปต่ำดังนี้

IE0

TF0

IE1

TF1

RI หรือ TI

TF2 หรือ EXF2

ตำแหน่งของบิตต่างๆ ในรีจิสเตอร์ IP เป็นดังรูปที่ 5

—	—	PT 2	PS	PT1	PX1	PT0	PX0
IP.7	IP.6	IP.5	IP.4	IP.3	IP.2	IP.1	IP.0

รูปที่ 5 ตำแหน่งของบิตต่างๆ ในรีจิสเตอร์ IP

หน้าที่ของบิตต่างๆ ในรีจิสเตอร์ IE มีดังนี้คือ

IP.7 และ IP.6 ไม่ใช้

PT2 IP.5 กำหนดลำดับความสำคัญของตัวตั้งเวลาหมายเลข 2

PS IP.4 กำหนดลำดับความสำคัญของพอร์ตอนุกรม

PT1 IP.3 กำหนดลำดับความสำคัญของตัวตั้งเวลาหมายเลข 1

PX1 IP.2 กำหนดลำดับความสำคัญของสัญญาณการขัดจังหวะ INT0

PT0 IP.1 กำหนดลำดับความสำคัญของตัวตั้งเวลาหมายเลข 0

PX0 IP.0 กำหนดลำดับความสำคัญของสัญญาณการขัดจังหวะ INT 1

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หากบิตใดเป็น 1 จะมีลำดับความสำคัญสูงสุด ไมโครคอนโทรลเลอร์จะตอบรับการร้องขอขัดจังหวะที่มีระดับความสำคัญสูงกว่า สัญญาณการร้องขอขัดจังหวะที่มีระดับความสำคัญสูงกว่าสามารถร้องขอขัดจังหวะซ้อนขึ้นมาได้ในขณะที่ไมโครคอนโทรลเลอร์กำลังทำคำสั่งตอบสนองการขัดจังหวะของสัญญาณร้องขอขัดจังหวะที่มีความสำคัญต่ำกว่าได้ ในขณะที่ไมโครคอนโทรลเลอร์กำลังทำโปรแกรมตอบสนองการขัดจังหวะอยู่ จะไม่สามารถตอบรับการร้องขอขัดจังหวะจากแหล่งกำเนิดที่มีความสำคัญระดับเดียวกันหรือต่ำกว่าได้อีก จนกว่าจะจบโปรแกรมตอบสนองการขัดจังหวะก่อน

2.1.5 การใช้งานสัญญาณขอขัดจังหวะจากภายนอก

แหล่งกำเนิดสัญญาณการขัดจังหวะจากภายนอกมี 2 แหล่งคือ INT0 และ INT1 สามารถเลือกสัญญาณการร้องขอขัดจังหวะได้สองแบบคือ เลือกใช้สัญญาณกระตุ้นที่ระดับลอจิก 0 หรือใช้สัญญาณการกระตุ้นที่การเปลี่ยนแปลงของสัญญาณจาก 1 ไป 0 ซึ่งสามารถเลือกโดยกำหนดค่าบิตของ IT0 และ IT1 ที่อยู่ในรีจิสเตอร์ควบคุมตัวตั้งเวลาที่ชื่อว่า TCON หากรีเซตให้ค่าของบิตเป็น 0 จะเป็นการเลือกสัญญาณการกระตุ้นที่ระดับลอจิก 0 การตรวจสอบการร้องขอขัดจังหวะจะกระทำทุกๆ แมชชีนไซเกิล เมื่อไมโครคอนโทรลเลอร์เรียกโปรแกรมการตอบสนองการขัดจังหวะมาทำงานแล้ว ต้องยกเลิกการส่งสัญญาณการขอขัดจังหวะก่อนที่โปรแกรมการตอบสนองจะจบลง หรือก่อนที่การร้องขอขัดจังหวะจากแหล่งกำเนิดอื่นจะเข้ามา เพื่อป้องกันการเกิดการขอขัดจังหวะซ้ำ สำหรับตำแหน่งเริ่มต้นของโปรแกรมการร้องขอขัดจังหวะของ INT0 จะอยู่ที่ตำแหน่ง 0003_H และตำแหน่งเริ่มต้นของโปรแกรมการร้องขอขัดจังหวะของ INT1 จะอยู่ที่ตำแหน่ง $000B_H$ ทำให้ความยาวของโปรแกรมตอบสนองการขัดจังหวะของ INT0 ถูกจำกัดอยู่ในช่วง 0003_H - $000A_H$ ซึ่งจะไม่สามารถเก็บโปรแกรมที่มีความยาวมากกว่า 8 ไบต์ได้

2.1.6 การใช้งานการขอขัดจังหวะจาก ตัวตั้งเวลาหมายเลข 0 และ ตัวตั้งเวลาหมายเลข 1

แหล่งกำเนิดสัญญาณการขัดจังหวะ TF0 และ TF1 เป็นแหล่งกำเนิดที่มาจากตัวตั้งเวลา/ตัวนับ หมายเลข 0 และ หมายเลข 1 ตามลำดับ TF0 และ TF1 จะเกิดขึ้นเมื่อค่าเวลาของตัวตั้งเวลาหมายเลข 0 หรือ 1 ที่ตั้งไว้ครบกำหนดแล้ว หรือนับได้ครบตามจำนวนที่ตั้งไว้ในกรณีที่เป็นตัวนับ โดย TF0 และ TF1 จะถูกเซ็ตโดยตัวไมโครคอนโทรลเลอร์คอนเองและค่าของสถานะ TF0 และ TF1 จะถูกเคลียร์โดยอัตโนมัติ เมื่อไมโครคอนโทรลเลอร์ตอบรับการร้องขอขัดจังหวะโดยการเรียกโปรแกรมย่อยที่อยู่ในตำแหน่งเริ่มต้นที่ $000B_H$ สำหรับตัวตั้งเวลาหมายเลข 0 และ $001B_H$ สำหรับตัวตั้งเวลาหมายเลข 1 มาทำงาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.1.7 การใช้งานการขอขัดจังหวะจากพอร์ตอนุกรม

แหล่งกำเนิดสัญญาณการขัดจังหวะจากพอร์ตอนุกรมมีสัญญาณการขัดจังหวะเกิดจาก 2 แหล่งคือ จาก ด้านรับ และ ด้านส่ง เป็นแหล่งกำเนิดที่มาจากหน่วยเก็บข้อมูลชั่วคราวทางด้านรับและด้านส่งของพอร์ตอนุกรม สัญญาณทั้ง 2 จะถูกนำมา ออร์ (OR) กัน เพื่อรวมเป็นสัญญาณเดียวกันทำให้มีสัญญาณการร้องขอขัดจังหวะเกิดขึ้นได้ทั้งในกรณีของการรับและการส่งข้อมูลของพอร์ตอนุกรมโปรแกรมตอบสนองการขัดจังหวะของพอร์ตอนุกรมอยู่ที่ตำแหน่ง 0023_H ตำแหน่งเดียว โปรแกรมตอบสนองการขัดจังหวะของพอร์ตอนุกรมจึงต้องมีการตรวจสอบก่อนว่าการขอขัดจังหวะเกิดขึ้นจาก ด้านรับ หรือ ด้านส่ง โดยใช้คำสั่งการตรวจสอบบิต RI หรือ TI ที่อยู่ในรีจิสเตอร์ SCON เมื่อทราบที่มาของการร้องขอการขัดจังหวะแล้วจึงไปทำโปรแกรมสำหรับการรับหรือส่งข้อมูลต่อไป หลังจากตรวจสอบแหล่งกำเนิดสัญญาณการร้องขอขัดจังหวะแล้วจะต้องทำการเคลียร์บิต RI หรือ TI ด้วยโปรแกรมก่อนที่จะจบโปรแกรมตอบสนองการขัดจังหวะ

2.1.8 การใช้งานการขอขัดจังหวะจาก ตัวตั้งเวลาหมายเลข 2

แหล่งกำเนิดสัญญาณการขัดจังหวะของ ตัวตั้งเวลาหมายเลข 2 มีอยู่ 2 แหล่งคือ TF2 และ EXF2 สัญญาณทั้ง 2 จะถูกนำมา ออร์กัน เมื่อไมโครคอนโทรลเลอร์มีการตอบรับขัดจังหวะ จะเรียกโปรแกรมย่อยที่ตำแหน่ง $002B_H$ มาทำงาน ในโปรแกรมตอบสนองการขัดจังหวะต้องมีการตรวจสอบว่าแหล่งกำเนิดมาจาก TF2 หรือ EXF2 เพื่อที่จะได้เลือกทำโปรแกรมตอบสนองได้ถูกต้องแล้วจึงทำคำสั่งเคลียร์บิต TF2 หรือ EXF2 ก่อนที่จะจบโปรแกรมตอบสนอง

2.2 การใช้งานตัวตั้งเวลาหรือ ตัวนับ หมายเลข 2

ตัวตั้งเวลาหรือตัวนับหมายเลข 2 เป็นตัวตั้งเวลาหรือตัวนับ ขนาด 16 บิต เราสามารถใช้เป็นตัวตั้งเวลา หรือ ตัวนับ ก็ได้ ภายในตัวตั้งเวลาหรือตัวนับหมายเลข 2 จะมีรีจิสเตอร์ขนาด 8 บิตเพิ่มมาอีก 2 ตัว คือ รีจิสเตอร์ อาร์แคปทูโลว์ (RCAP2L) และ รีจิสเตอร์ อาร์แคปทูไฮ (RCAP2H) การกำหนดการทำงานของ ตัวตั้งเวลาหรือตัวนับหมายเลข 2 จะกำหนดได้จากรีจิสเตอร์ควบคุมตัวตั้งเวลาหรือตัวนับหมายเลข 2ที่ชื่อ T2CON ซึ่งอยู่ที่ตำแหน่ง $0C8_H$ ซึ่งตำแหน่งของบิตต่างๆ เป็นดังรูปที่ 6

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

Bit Addressable

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

รูปที่ 6 ตำแหน่งบิตภายในรีจิสเตอร์ควบคุมตัวตั้งเวลาหรือตัวนับหมายเลข 2

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หน้าที่ของบิตต่างๆ ในรีจิสเตอร์ควบคุมตัวตั้งเวลา/ตัวนับหมายเลข 2 มีดังต่อไปนี้

TF2 บิตนี้จะถูกเซ็ตด้วยไมโครคอนโทรลเลอร์เมื่อตัวตั้งเวลาหรือตัวนับหมายเลข 2 มีการนับครบถึงค่าที่กำหนดไว้ หรือเรียกว่าการเกิดโอเวอร์โฟล (Over flow) และจะต้องสั่งเคลียร์ด้วยโปรแกรม บิตนี้จะไม่ถูกเซตหากบิต RCLK หรือ TCLK เป็น 1

EXF2 ถ้าให้ตัวตั้งเวลาหรือตัวนับหมายเลข 2 ทำงานในลักษณะ แคปเจอร์ หรือ ตั้งค่าใหม่อัตโนมัติบิตนี้จะถูกเซตเมื่อบิต EXEN2 เป็น 1 และสัญญาณที่ขากระตุ้นภายนอกของตัวตั้งเวลาหมายเลข2 (T2EX หรือ P1.1) มีการเปลี่ยนจาก 1 ไป 0 หากกำหนดให้ตัวตั้งเวลาหรือตัวนับหมายเลข 2 สามารถร้องขอการ ขัดจังหวะได้ ตัวตั้งเวลาหรือตัวนับหมายเลข 2 จะส่งสัญญาณร้องขอขัดจังหวะเมื่อบิต EXF2 เป็น 1 หลังจากที่ทำโปรแกรมตอบสนองการขัดจังหวะแล้วต้องเคลียร์บิต EXF2 ด้วยโปรแกรม ก่อนที่จะออกจากการตอบสนองการขัดจังหวะ

RCLK เป็นบิตสำหรับควบคุมสัญญาณนาฬิกาในการรับข้อมูลของพอร์ตอนุกรมซึ่งสามารถเซตหรือ เคลียร์ด้วยโปรแกรมโดยที่ หาก บิตนี้เป็น 1 จะเป็นการนำพัลส์ที่ได้จากการ เกิดโอเวอร์โฟล ของตัวตั้งเวลาหรือตัวนับหมายเลข 2 เพื่อนำไปเป็นสัญญาณนาฬิกา ของการรับข้อมูลของพอร์ตอนุกรมในแบบที่ 1 และแบบที่ 3 ส่วนถ้าหากบิตนี้เป็น 0 จะเป็นการเลือกใช้สัญญาณนาฬิกาของการรับข้อ มูลของพอร์ตอนุกรมของ ตัวตั้งเวลาหรือตัวนับหมายเลข 1

TCLK เป็นบิตสำหรับควบคุมสัญญาณนาฬิกาในการส่งข้อมูลของพอร์ตอนุกรมซึ่งสามารถเซตหรือ เคลียร์ด้วยโปรแกรมโดยที่ หาก บิตนี้เป็น 1 จะเป็นการนำพัลส์ที่ได้จากการเกิดโอเวอร์โฟล ของตัวตั้งเวลาหรือตัวนับหมายเลข 2 เพื่อนำไปเป็นสัญญาณนาฬิกาของการส่งข้อมูลของพอร์ตอนุกรมในแบบที่ 1 และแบบที่ 3 ส่วนถ้าหากบิตนี้เป็น 0 จะเป็นการเลือกใช้สัญญาณนาฬิกาของการส่งข้อ มูลของพอร์ตอนุกรมของ ตัวตั้งเวลาหรือตัวนับหมายเลข 1

EXEN2 หากกำหนดให้บิตนี้เป็น 1 จะทำให้การตั้งเก็บค่าใหม่หรือแคปเจอร์เกิดขึ้น เมื่อมีการเปลี่ยนแปลงสัญญาณที่ขากระตุ้นภายนอกของตัวตั้งเวลาหมายเลข2 จาก 1 ไป 0 และไม่มีการใช้ตัวตั้งเวลาหรือตัวนับหมายเลข 2 เป็นตัวกำเนิดสัญญาณนาฬิกาของการรับส่งข้อมูลของพอร์ตอนุกรมและหากกำหนดให้บิต EXEN2 เป็น 0 จะมีการทำงานคล้ายกับการกำหนดบิต EXEN2 เป็น 1 แต่จะไม่มีการใช้งานที่ขากระตุ้นภายนอกของตัวตั้งเวลาหมายเลข2

TR2 เป็นบิตควบคุมการทำงานหรือหยุดการทำงานของ ตัวตั้งเวลา/ตัวนับหมายเลข 2 หากบิตนี้เป็น 1 จะกำหนดให้ ตัวตั้งเวลาหรือตัวนับหมายเลข 2 ทำงาน และหากเป็น 0 จะกำหนดให้ตัวตั้งเวลา/ตัวนับหมายเลข 2 ไม่ทำงาน

C/ $\overline{\text{T2}}$ เป็นบิตเลือกวิธีการทำงานของ ตัวตั้งเวลา/ตัวนับหมายเลข 2

C/$\overline{T2}$ = ทำงานในรูปแบบของตัวจับเวลา (ความถี่ของระบบ / 12)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

$C/\overline{T2}$ =ทำงานในรูปแบบของการนับ ทุกๆขอบขาลงของสัญญาณที่จะทำการนับ

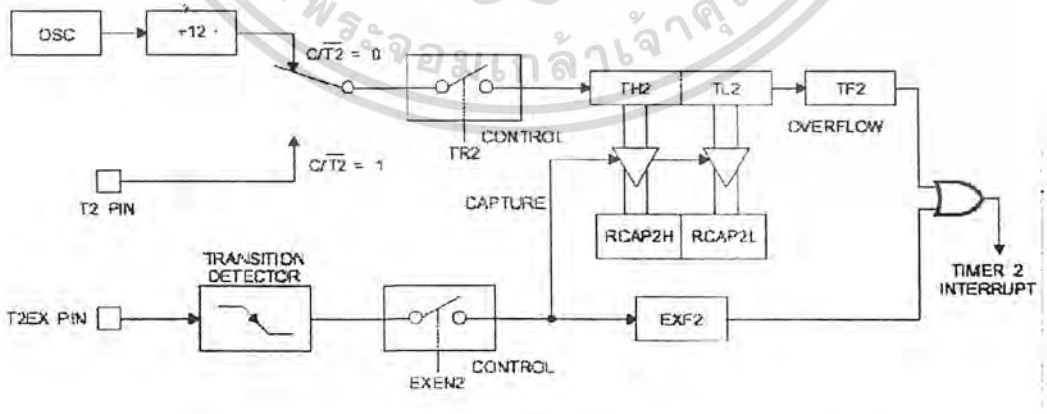
$CP/\overline{RL2}$ เป็นตัวเลือกรูปแบบการทำงานโดยสามารถกำหนดได้จากโปรแกรม

$CP/\overline{RL2}$ = 1 ทำงานในรูปแบบแคปเจอร์ ซึ่งการแคปเจอร์จะเกิดเมื่อบิต EXEN2 เป็น 1 และสัญญาณที่ขากระตุ้นภายนอกของตัวตั้งเวลาหมายเลข2 เปลี่ยนจาก 1 เป็น 0

$CP/\overline{RL2}$ = 0 ทำงานในรูปแบบการเก็บค่าใหม่ จะเกิดเมื่อตัวตั้งเวลาหรือตัวนับหมายเลข 2 เกิดโอเวอร์โฟล หรือ เมื่อบิต EXEN2 เป็น 1 แล้วมีสัญญาณ เข้ามาที่ขากระตุ้นภายนอกของตัวตั้งเวลาหมายเลข2 เปลี่ยนจาก 1 เป็น 0 ถ้ามีการเซตให้บิต RCLKหรือ TCLK เป็น 1 บิตนี้จะไม่ใช้งาน และการเก็บค่าใหม่ของตัวตั้งเวลาหรือตัวนับหมายเลข 2 จะเกิดการโอเวอร์โฟลโดยอัตโนมัติ การทำงานของตัวตั้งเวลาหรือตัวนับหมายเลข 2 มีอยู่ 3 รูปแบบคือ

2.2.1 การทำงานแบบแคปเจอร์

การทำงานแบบแคปเจอร์ สามารถเลือกทำงานได้ 2 ลักษณะจากการกำหนดค่าบิต EXEN2 ที่อยู่ในรีจิสเตอร์ควบคุมตัวตั้งเวลาหรือตัวนับหมายเลข 2โดยหากกำหนดให้บิต EXEN2 เป็น 0 ตัวตั้งเวลาหรือตัวนับหมายเลข 2 ก็จะทำงานเป็นตัวตั้งเวลาหรือตัวนับแบบ 16 บิต และหากกำหนดบิต EXEN2 เป็น 1 การทำงานของตัวตั้งเวลาหรือตัวนับหมายเลข 2 จะคล้ายกัน แต่จะมีส่วนเพิ่มเติมคือเมื่อสัญญาณจากภายนอกที่ขา T2EX (P1.1) เปลี่ยนจาก 1 ไป 0 จะทำให้ค่าที่อยู่ในรีจิสเตอร์ของการนับ TL2 และ TH2 ถูกเก็บค่าเข้าไปในรีจิสเตอร์ อาร์แคปทูโลว์และอาร์แคปทูไฮ ตามลำดับ และทำให้บิต EXF2 ในรีจิสเตอร์ควบคุมตัวตั้งเวลาหรือตัวนับหมายเลข 2 ถูกเซตด้วย จึงมีการร้องขอขัดจังหวะได้เหมือนกับบิต TF2 แต่บิต EXF2 ต้องสั่งเคลียร์ด้วยโปรแกรม การทำงานของตัวตั้งเวลา/ตัวนับหมายเลข 2 ในแบบแคปเจอร์แสดงดังรูปที่ 7



รูปที่ 7 การทำงานของตัวตั้งเวลาหรือตัวนับหมายเลข 2 ในแบบแคปเจอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.2.2 การทำงานแบบกำหนดค่าใหม่อัตโนมัติ

การทำงานแบบนี้สามารถเลือกการทำงานได้ 2 ลักษณะ โดยการกำหนดค่าของบิต EXEN2 ที่อยู่ในรีจิสเตอร์ควบคุมตัวตั้งเวลา/ตัวนับหมายเลข 2 โดย

1) หากกำหนดให้บิต EXEN2 เป็น 0 เมื่อค่าของการนับหรือการตั้งเวลาเกิดโอเวอร์โฟลขึ้นนอกจากจะเซตบิต TF2 แล้ว จะทำให้ค่าในรีจิสเตอร์ อาร์แคปทูโลว์ และ อาร์แคปทูไฮ ถูกนำไปกำหนดค่าให้กับรีจิสเตอร์ TL2 และ TH2 ตามลำดับ เพื่อเป็นค่าเริ่มต้นของการนับครั้งต่อไป

2) หากกำหนดให้บิตEXEN2 เป็น 1 เมื่อค่าของการนับหรือการตั้งเวลาเกิดโอเวอร์โฟลขึ้นนอกจากจะเซตบิต TF2 และ นำค่าในรีจิสเตอร์ อาร์แคปทูโลว์ และ อาร์แคปทูไฮ ไปกำหนดค่าให้กับรีจิสเตอร์ TL2 และ TH2 ตามลำดับ แล้วการ นำค่าในรีจิสเตอร์ อาร์แคปทูโลว์ และ อาร์แคปทูไฮ ไปกำหนดค่าให้กับรีจิสเตอร์ TL2 และ TH2 ยังสามารถควบคุมได้จากสัญญาณที่ขากระตุ้นภายนอกของตัวตั้งเวลาหมายเลข2โดยการกำหนดค่าใหม่จะกระทำเมื่อสัญญาณที่ขากระตุ้นภายนอกของตัวตั้งเวลาหมายเลข2 เปลี่ยนจาก 1 ไป 0 และจะทำให้บิต EXF2 ถูกเซตด้วย

นอกจากนี้ไมโครคอนโทรลเลอร์เบอร์ เอที89เอส53 ยังมีคุณสมบัติพิเศษคือสามารถกำหนดได้ว่าจะให้ตัวตั้งเวลาหรือตัวนับหมายเลข 2 นับแบบขึ้นหรือลงโดยกำหนดได้จากบิตควบคุมการนับขึ้นหรือลง หรือ DCEN (Down Counter Enable) ซึ่งบิตนี้จะอยู่ในรีจิสเตอร์พิเศษชื่อ T2MOD ซึ่งอยู่ที่ตำแหน่ง $0C9_H$ ซึ่งแสดงดังรูปที่ 8 การทำงานแบบนับลงในแบบกำหนดค่าใหม่อัตโนมัติ แสดงดังรูปที่ 9 และการทำงานนับขึ้นแบบกำหนดค่าใหม่อัตโนมัติ แสดงดังรูปที่ 10

T2MOD Address = 0C9H — Reset Value = XXXX XX00B

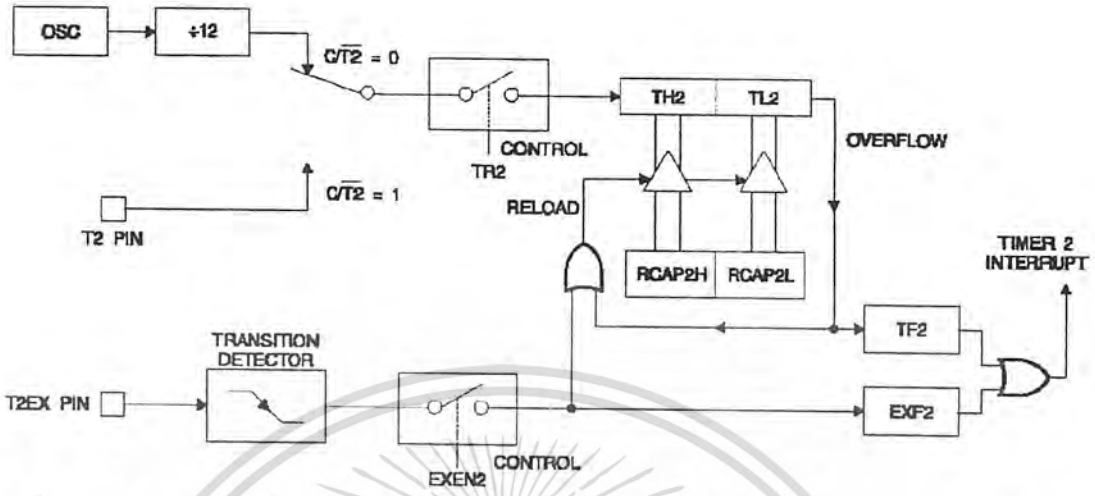
Not Bit Addressable

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

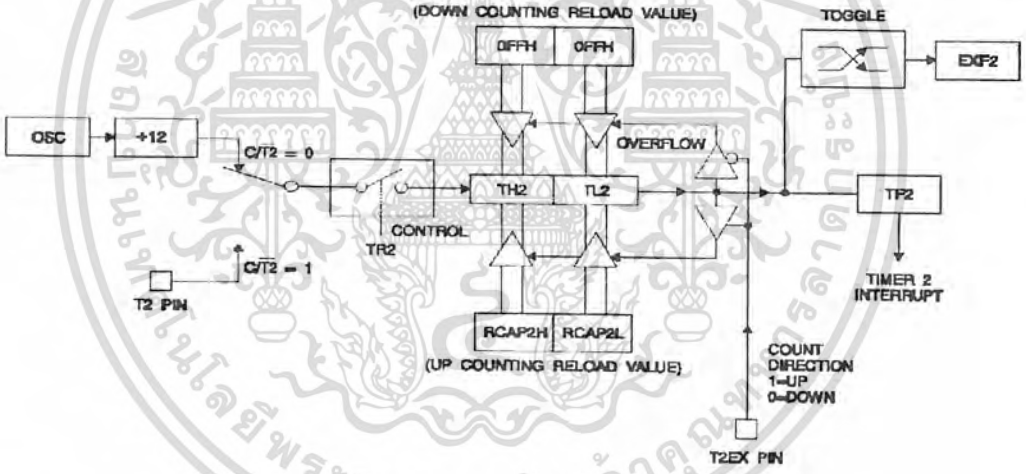
Symbol	Function
—	Not implemented, reserved for future use.
T2OE	Timer 2 Output Enable bit.
DCEN	When set, this bit allows Timer 2 to be configured as an up/down counter.

รูปที่ 8 แสดงตำแหน่งบิตและการใช้งานของ รีจิสเตอร์ T2MOD

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 9 การทำงานแบบนับลงแบบกำหนดค่าใหม่อัตโนมัติของตัวตั้งเวลาหรือตัวนับหมายเลข 2



รูปที่ 10 การทำงานแบบนับขึ้นแบบกำหนดค่าใหม่อัตโนมัติของตัวตั้งเวลาหรือตัวนับหมายเลข 2

2.2.3 การทำงานเป็นตัวสร้างอัตราการรับส่งข้อมูลอนุกรม

การทำงานแบบนี้ จะกำหนดโดยเซตบิต RCLK และหรือ TCLK ที่อยู่ในรีจิสเตอร์ควบคุมตัวตั้งเวลาหรือตัวนับหมายเลข 2 ให้เป็น 1

2.3 การใช้งานพอร์ตอนุกรม

การส่งข้อมูลแบบอนุกรมมีโครงสร้างที่สำคัญหลายอย่างเช่นความเร็วในการส่งข้อมูลซึ่งเรียกว่า บอดเรต (Baud Rate) มีหน่วยเป็น บิตต่อวินาที เอที86เอส53 มีพอร์ตอนุกรมที่สามารถรับ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ส่งข้อมูลพร้อมกันได้ในเวลาเดียวกัน การควบคุมการทำงานของพอร์ตอนุกรมสามารถควบคุมได้จากรีจิสเตอร์ควบคุมการทำงานของพอร์ตอนุกรมที่ชื่อว่า SCON (Serial Port Control Register)

2.3.1 รีจิสเตอร์ควบคุมการทำงานของพอร์ตอนุกรม

รีจิสเตอร์ควบคุมการทำงานของพอร์ตอนุกรมเป็นรีจิสเตอร์ที่ทำหน้าที่ควบคุมรูปแบบการทำงานของพอร์ตอนุกรม และเป็นที่เก็บข้อมูลบิตที่ 9 ของการรับส่งข้อมูลและมีบิตของการร้องขอขัดจังหวะของพอร์ตอนุกรม การควบคุมรูปแบบการทำงาน สามารถทำได้โดยกำหนดค่าให้กับบิตต่างๆ ในรีจิสเตอร์ควบคุมการทำงานของพอร์ตอนุกรมซึ่งตำแหน่งบิตต่างๆ แสดงดังรูปที่ 11

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
SMO	SM 1	SM 2	REN	TB 8	RB 8	TI	RI

รูปที่ 11 ตำแหน่งบิตต่างๆ ภายในรีจิสเตอร์ควบคุมการทำงานของพอร์ตอนุกรม

ความหมายของบิตต่างๆภายในรีจิสเตอร์ควบคุมการทำงานของพอร์ตอนุกรมมีดังต่อไปนี้

SM0 , SM1 เป็นบิตกำหนดรูปแบบการทำงานของพอร์ตอนุกรมซึ่งมีอยู่ 4 รูปแบบดังแสดงในตารางที่ 1

SM 0	SM 1	รูปแบบ	การทำงาน	อัตราการรับส่ง
0	0	0	แบบเลื่อนข้อมูลออกธรรมดา	ความถี่ระบบ / 12
0	1	1	รับส่งข้อมูลแบบ 8 บิต	กำหนดเองได้
1	0	2	รับส่งข้อมูลแบบ 9 บิต	ความถี่ระบบ / 32 หรือ ความถี่ระบบ / 64
1	1	3	รับส่งข้อมูลแบบ 9 บิต	กำหนดเองได้

ตารางที่ 1. รูปแบบของการรับส่งข้อมูลของพอร์ตอนุกรม กำหนดโดย บิต SM0 และ SM1

SM2 เป็นการทำงานในลักษณะที่ใช้ไมโครคอนโทรลเลอร์หลายๆ ตัวต่อเข้าด้วยกัน จะใช้ในรูปแบบการติดต่อในรูปแบบที่ 2 หรือ 3 ซึ่งสามารถควบคุมได้โดย หากกำหนดบิต SM2 เป็น 0 จะทำให้บิตของการขัดจังหวะทางด้านรับ ไม่ถูกเซตเมื่อรับข้อมูลเข้ามาแล้วมีค่าบิตที่ 9 เป็น 0 สำหรับการทำงานในรูปแบบที่ 1 ถ้าเซต SM2 เป็น 1 บิตของการขัดจังหวะทางด้านรับ จะไม่ถูกเซตหากข้อมูลที่รับเข้ามาไม่มี บิตหยุดการทำงานในรูปแบบ 0 จะต้องกำหนดให้ SM2 เป็น 0

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

REN เซตหรือเคลียร์ด้วยโปรแกรมใช้ควบคุมการทำงานของพอร์ตอนุกรม
REN เป็น 1 ให้มีการรับข้อมูลทางพอร์ตอนุกรม
REN เป็น 0 ไม่ให้มีการรับข้อมูลทางพอร์ตอนุกรม

TB8 เป็นบิตที่ 9 ที่ต้องการส่งในรูปแบบที่ 2 หรือ 3 สามารถเซตหรือเคลียร์ด้วยโปรแกรมได้

RB8 เป็นบิตเก็บข้อมูลที่รับเข้ามาบิตที่ 9 ในรูปแบบที่ 2 หรือ 3 สำหรับการทำงานในรูปแบบ 1 หากกำหนดให้บิต SM2 เป็น 0 บิต RB8 จะเป็นค่าของบิตหยุดที่รับเข้ามา ส่วนการรับในรูปแบบ 0 ไม่มีการใช้ RB8

T I บิตของการขอขัดจังหวะด้านส่งข้อมูล บิตนี้จะถูกเซตโดยไมโครคอนโทรลเลอร์เมื่อจบการส่งข้อมูลบิตที่ 8 ในรูปแบบ 0 หรือเมื่อเริ่มส่งบิตปิดท้ายในการส่งรูปแบบ 1,2 หรือ 3 เมื่อจบโปรแกรมตอบสนองการขัดจังหวะของการส่งข้อมูลจะต้องสั่งเคลียร์บิตนี้ด้วยโปรแกรมเสมอ

RI บิตของการขอขัดจังหวะด้านรับข้อมูล บิตนี้จะถูกเซตโดยไมโครคอนโทรลเลอร์เมื่อข้อมูลบิตที่ 8 ในรูปแบบ 0 ถูกรับเข้ามา หรือเมื่อบิตปิดท้ายถูกรับเข้ามาในครึ่งเวลาแรก ในการรับรูปแบบ 1,2 หรือ 3 เมื่อจบโปรแกรมตอบสนองการขัดจังหวะของการรับข้อมูลจะต้องสั่งเคลียร์บิตนี้ด้วยโปรแกรมเสมอ

การรับส่งข้อมูลของพอร์ตอนุกรมจะมีรีจิสเตอร์ที่ทำหน้าที่เก็บข้อมูลของการรับส่งอยู่ 1 ตัว คือรีจิสเตอร์ SBUF การส่งข้อมูลออกไปยังพอร์ตอนุกรมของไมโครคอนโทรลเลอร์ทำได้โดยการใส่ข้อมูลลงไปในรีจิสเตอร์ตัวนี้ ส่วนการอ่านข้อมูลที่รับเข้ามาจากพอร์ตอนุกรมจะอ่านจากรีจิสเตอร์ตัวนี้เช่นกันวงจรด้านรับจะมีหน่วยพักข้อมูลขนาด 1 ไบต์ ที่ทำหน้าที่เก็บข้อมูลที่รับเข้ามา เพื่อทำให้ด้านรับสามารถรับข้อมูลไบต์ที่ 2 เข้ามาได้ทันทีหลังจากที่ข้อมูลไบต์แรกเข้ามาแล้ว แม้จะยังไม่ได้ถูกอ่านออกไป แต่ถ้าข้อมูลไบต์แรกยังไม่ถูกอ่านก่อนที่ข้อมูลไบต์ที่ 2 จะเข้ามาครบ ข้อมูลในไบต์ที่ 2 จะถูกยกเลิกทันที

2.3.2 การทำงานของพอร์ตอนุกรมในรูปแบบต่างๆ

การรับส่งข้อมูลจะรับข้อมูลที่ขา RXD(P3.0) และส่งข้อมูลออกทางขา TX(P1.1) การรับส่งข้อมูลจะเริ่มจากบิตด้านต่ำก่อน

1) รูปแบบ 0 ใช้ในการรับส่งข้อมูลขนาด 8 บิตแบบอนุกรม อัตราการรับส่งข้อมูลจะเป็น 1/12 เท่าของสัญญาณนาฬิกา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สำนักหอสมุดกลาง พระจอมเกล้าลาดกระบัง

2) รูปแบบ 1 ใช้ในการรับส่งข้อมูลอนุกรม 10 บิต โดยจะประกอบด้วย บิตเริ่มต้น ซึ่งมีค่าเป็น 0 จำนวน 1 บิต บิตข้อมูล 8 บิต และ บิตหยุด ซึ่งมีค่าเป็น 1 จำนวน 1 บิต ด้านรับข้อมูลจะค่าบิตหยุดไปเก็บในบิต RB8 ที่อยู่ในรีจิสเตอร์ควบคุมการทำงานของพอร์ตอนุกรม สำหรับอัตราการรับส่งข้อมูลสามารถกำหนดได้ตามต้องการ

3) รูปแบบ 2 ใช้ในการรับส่งข้อมูลอนุกรม 11 บิต โดยจะประกอบด้วย บิตเริ่มต้น ซึ่งมีค่าเป็น 0 จำนวน 1 บิต บิตข้อมูล 8 บิต บิตข้อมูลที่สามารถ กำหนดได้จำนวน 1 บิต และบิตหยุด ซึ่งมีค่าเป็น 1 จำ นวน 1 บิต สำหรับด้านส่ง ข้อมูลบิตที่ 9 จะถูกเก็บไว้ที่บิต TB8 ในรีจิสเตอร์ควบคุมการทำงานของพอร์ตอนุกรม ซึ่งสามารถกำหนดให้เป็น 1 หรือ 0 ก็ได้ ในการส่งข้อมูลเราสามารถนำค่าใน พาริตี้บิต (Parity Bit) มากำหนดให้กับบิต TB8 ในรีจิสเตอร์ควบคุมการทำงานของพอร์ตอนุกรม เพื่อใช้ในการส่งข้อมูลแบบมีพาริตี้บิต ส่วนการรับข้อมูลจะนำบิตที่ 9 ที่รับเข้ามาไปเก็บในบิต RB8 ที่อยู่ในรีจิสเตอร์ควบคุมการทำงานของพอร์ตอนุกรม อัตราการรับส่งข้อมูลในรูปแบบนี้สามารถเลือกได้ 2 อัตรา คือ 1/32 เท่าของสัญญาณนาฬิกาของระบบ และ1/64 เท่าของสัญญาณนาฬิกาของระบบ

4) รูปแบบ 3 การทำงานของรูปแบบนี้จะมีลักษณะคล้ายกับรูปแบบ 2 แต่รูปแบบ 3 สามารถที่จะกำหนดอัตราการรับส่งข้อมูลได้ตามต้องการ

2.3.3 การกำหนดอัตรารับส่งของพอร์ตอนุกรม

อัตราการรับส่งในรูปแบบต่างๆ มีรายละเอียดดังต่อไปนี้คือ

1) อัตราการรับส่งในรูปแบบ 0 = ความถี่สัญญาณนาฬิกาของระบบ / 12

2) อัตราการรับส่งในรูปแบบ 2 ถ้าบิต SMOD=0 จะมีค่าเท่ากับ 1/64 เท่าของความถี่สัญญาณนาฬิกาของระบบ

3) อัตราการรับส่งในรูปแบบ 2 ถ้าบิต SMOD=1 จะมีค่าเท่ากับ 1/32 เท่าของความถี่สัญญาณนาฬิกาของระบบ

สำหรับอัตราการรับส่งในรูปแบบ 1 และ 3 จะกำหนดได้ตามต้องการ โดยการใช้ ตัวตั้งเวลาหรือตัวนับ หมายเลข 1 หรือ 2 เป็นตัวกำหนด นอกจากนี้อัตราการรับส่งข้อมูลสามารถกำหนดได้จากตัวตั้งเวลาหรือตัวนับทั้ง 2 ตัว เพื่อใช้ในกรณีที่อัตราการรับ-ส่งข้อมูลไม่เท่ากันได้

2.3.4 การใช้ ตัวตั้งเวลาหรือตัวนับหมายเลข 1 สร้างอัตราการรับส่งข้อมูล

การใช้ตัวตั้งเวลา/ตัวนับหมายเลข 1เป็นตัวสร้างอัตราการรับส่งข้อมูลของพอร์ตอนุกรมในรูปแบบ 1 หรือ 3 ต้องควบคุมไม่ให้ ตัวตั้งเวลาหรือตัวนับหมายเลข 1 ส่งสัญญาณขอขัดจังหวะโดยการควบคุมที่รีจิสเตอร์ IE เราสามารถใช้ตัวตั้งเวลาหรือตัวนับหมายเลข 1 ทำงานในลักษณะของตัวนับหรือตัวตรวจจับการตั้งเวลาได้อย่างใดอย่างหนึ่งทั้ง 3 รูปแบบสำหรับ อัตรา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

42664

การรับส่งข้อมูลมาจากอัตราการเกิดโอเวอร์โฟลของตัวตั้งเวลาหรือตัวนับหมายเลข 1 และค่าของบิต SMOD ที่อยู่ในรีจิสเตอร์ PCON การใช้งานทั่วไปจะกำหนดให้ตัวตั้งเวลาหรือตัวนับหมายเลข 1 ทำงานในลักษณะของตัวจับเวลาแบบกำหนดค่าใหม่อัตโนมัติ โดยมีค่าการควบคุมในรีจิสเตอร์ TMOD = 0010xxxx ซึ่งจะได้อัตราการรับส่งข้อมูลในรูปแบบ 1 และ 3 เป็นดังนี้

$$\text{อัตราการรับส่งข้อมูล} = \frac{2^{SMOD}}{32} * \frac{\textit{ความถี่นาฬิกาของระบบ}}{12 * [256 - (TH\,1)]}$$

หากต้องการให้ตัวตั้งเวลา/ตัวนับหมายเลข 1 มีอัตราการรับส่งต่ำมากๆ สามารถทำได้โดยใช้ ตัวตั้งเวลาหรือตัวนับหมายเลข 1 ทำงานในลักษณะตัวตรวจจับเวลาแบบ 16 บิต และให้มีการขอขัดจังหวะจากตัวตั้งเวลาหรือตัวนับหมายเลข 1 แล้วให้โปรแกรมตอบสนองการขัดจังหวะของตัวตั้งเวลาหรือตัวนับหมายเลข 1 ทำการกำหนดค่าใหม่ให้กับตัวตั้งเวลาหรือตัวนับหมายเลข 1

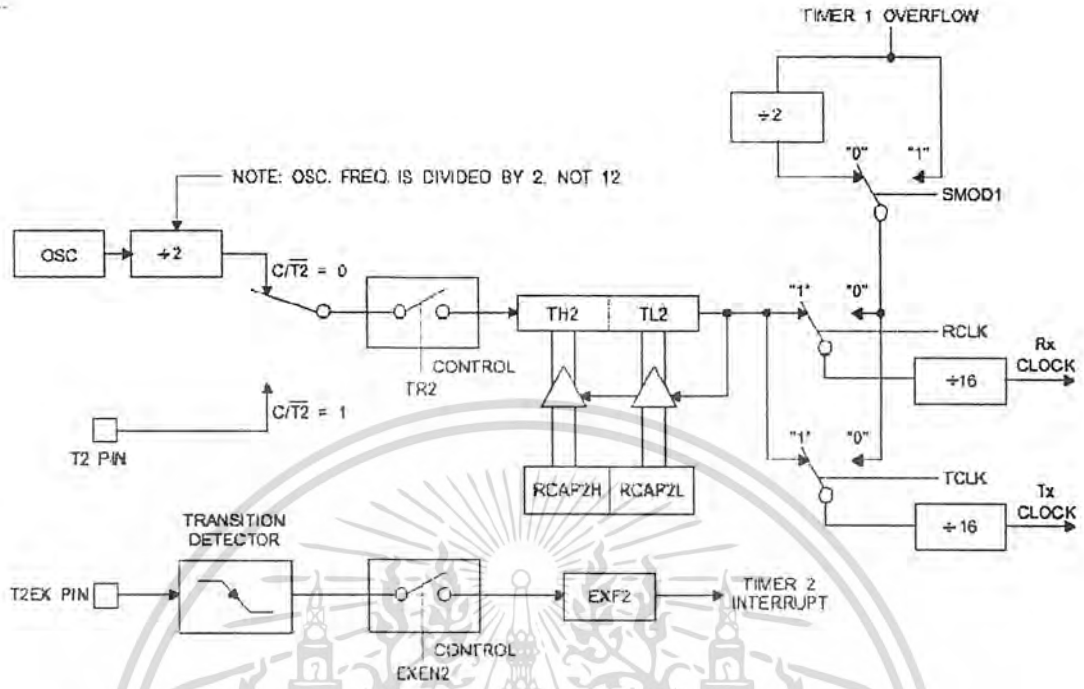
2.3.5 การใช้ ตัวตั้งเวลาหรือตัวนับหมายเลข 2 สร้างอัตราการรับส่งข้อมูล

การเลือกตัวตั้งเวลาหรือตัวนับหมายเลข 2เพื่อกำหนดอัตราการรับส่งข้อมูลของพอร์ตอนุกรม จะทำที่บิต TCLK สำหรับการส่งข้อมูล และบิต RCLK หรับการรับข้อมูล ซึ่งบิตทั้งสองอยู่ในรีจิสเตอร์ที่ T2CON การทำงานของวงจรเลือกตัวกำเนิดอัตราการรับส่งข้อมูลของตัวตั้งเวลาหรือตัวนับหมายเลข 2 แสดงดังรูปที่ 12 ในการใช้ตัวตั้งเวลาหรือตัวนับหมายเลข 2 สร้างอัตราการรับส่งข้อมูลจะตั้งให้ตัวตั้งเวลาหรือตัวนับหมายเลข 2 ทำงานในรูปแบบตั้งเวลาหรือตัวนับก็ได้ การใช้งานทั่วไปจะกำหนดให้ตัวตั้งเวลาหรือตัวนับหมายเลข 2 ทำงานเป็นตัวตั้งเวลา การทำงานเพื่อสร้างอัตรารับส่งข้อมูลจะแตกต่างจากการตั้งเวลาคือ การทำงานแบบตั้งเวลาค่าของเวลาจะเพิ่มทุกๆ แมชชีนไซเกิล (1/12 เท่าของสัญญาณนาฬิกาของระบบ) แต่การทำงานในแบบสร้างอัตราการรับส่งข้อมูลจะเพิ่มขึ้นทุก ๆ ½ เท่าของสัญญาณนาฬิกา ดังนั้นสามารถคำนวณอัตราการรับส่งเมื่อใช้ตัวตั้งเวลาหรือตัวนับหมายเลข 2 ได้จาก

$$\text{อัตราการรับส่งในรูปแบบ 1 หรือ 3} = \frac{\textit{ความถี่ของสัญญาณนาฬิกาของระบบ}}{32 * [65536 - (RCAP2H, RCAP2L)}$$

(RCAP2H , RCAP2L) คือค่าของ RCAP2H และ RCAP2L ที่ต่อกันเป็นข้อมูลขนาด 16 บิต

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 12 การทำงานของวงจรกำเนิดอัตราการรับส่งข้อมูลของตัวตั้งเวลาหรือตัวนับหมายเลข 2

2.4 พอร์ตอนุกรมของคอมพิวเตอร์

2.4.1 มาตรฐานพอร์ตอนุกรมแบบ RS-232

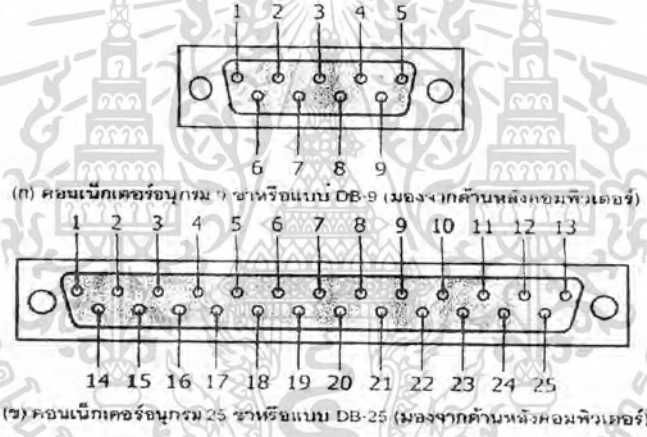
มาตรฐานการเชื่อมต่ออนุกรมแบบ RS-232 เป็นมาตรฐานอุตสาหกรรมที่ออกแบบมาเพื่อการส่งข้อมูลแบบ ไม่เข้าจังหวะกัน 2 ทิศทาง โดยมาตรฐานแบบ RS-232 ในอดีตถูกออกแบบมาเพื่อการส่งข้อมูลจากคอมพิวเตอร์ไปยังโมเด็มเพียงอย่างเดียวเพื่อที่จะนำข้อมูลจากโมเด็มนี้ผ่านสายโทรศัพท์ไปยังคอมพิวเตอร์ที่อยู่ห่างไกลกันโดยคณะกรรมการที่เรียกว่า สมาคมอุตสาหกรรมอิเล็กทรอนิกส์ (Electronic Industries Association : EIA) ได้วางมาตรฐานที่มีชื่อเรียกว่า EIA RS-232 มาตรฐานนี้ในช่วงแรกใช้จะใช้ขั้วต่อแบบ DB-25 โดยกำหนดความยาวของสายสัญญาณสูงสุดไว้ที่ 50 ฟุต มีระดับสัญญาณตั้งแต่ -3 โวลต์ ถึง -12 โวลต์ แสดงว่ามีข้อมูล และ +3 โวลต์ ถึง +12 โวลต์ แสดงว่าเป็นช่องว่าง มาตรฐานแบบ RS-232 ได้กำหนดรูปแบบของอุปกรณ์เชื่อมต่อข้อมูล (Data Terminal Equipment : DTE) กับวงจรข้อมูลปลายทาง (Data Communication Equipment : DCE) ไว้ว่า อุปกรณ์เชื่อมต่อข้อมูล จะต้องเป็นอุปกรณ์ที่มีการประมวลผลในตัวเช่น ไมโครคอนโทรลเลอร์หรือ ไมโครคอมพิวเตอร์ซึ่งมีความสามารถในการสร้างบิตอนุกรมได้ ส่วนวงจรข้อมูลปลายทาง จะทำหน้าที่เพียงตัวรับข้อมูลที่ส่งมาจาก อุปกรณ์เชื่อมต่อข้อมูล เท่านั้นโดยการรับส่งข้อมูลทั้งสองจะเป็นไปตามมาตรฐานแบบ RS-232 ข้อแตก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ต่างระหว่าง อุปกรณ์เชื่อมต่อข้อมูล และ วงจรข้อมูลปลายทางที่เห็นได้ชัดคือ ขั้วต่อของ อุปกรณ์เชื่อมต่อข้อมูล จะเป็นตัวผู้ ส่วนขั้วต่อของ วงจรข้อมูลปลายทางจะเป็นตัวเมีย ซึ่งพอร์ตอนุกรมของคอมพิวเตอร์ที่ใช้กันอยู่ทั่วไปจะเป็น อุปกรณ์เชื่อมต่อข้อมูล ส่วนขั้วต่อที่อยู่ที่โมเด็มจะเป็นวงจรข้อมูลปลายทาง

2.4.2 ขั้วต่อสำหรับพอร์ต RS-232 และการเชื่อมต่อ

มาตรฐานการเชื่อมต่อแบบ RS-232 จะใช้ขั้วต่อแบบ DB-25 ตัวผู้หรือ DB-9 ตัวผู้ ซึ่งขั้วต่อแบบ DB-25 จะมีขาต่อใช้งานเพียง 9 เส้น เช่นเดียวกับขั้วต่อแบบ DB-9 เนื่องจากขาอื่นๆ ที่เคยใช้งานในอดีต ปัจจุบันมีการใช้งานไม่มากนักจึงถูกยกเลิกไป รายละเอียดตำแหน่งของขาต่างๆของคอนเน็กเตอร์แสดงดังรูปที่ 13 และ ตารางที่ 2 แสดงรายละเอียดของขาต่างๆ ที่ใช้งานในปัจจุบัน



รูปที่ 13 ลักษณะตำแหน่งการวางขาของขั้วต่อแบบ DB-9 และ DB-25

การเชื่อมต่อคอมพิวเตอร์กับอุปกรณ์ภายนอกแสดงดังรูปที่ 14 ลูกศรในรูปแสดงถึงทิศทางของข้อมูล ในรูปที่ 14 (ก) เป็นการเชื่อมต่อโดยตรงไม่ต้องผ่านโมเด็ม โดยมีการตรวจสอบหรือแฮนด์เช็กเต็มรูปแบบ ส่วนรูปที่ 14 (ข) เป็นการเชื่อมต่อในลักษณะที่ใช้สายสัญญาณเพียง 3 เส้น โดยเส้นหนึ่งสำหรับส่งข้อมูล อีกเส้นหนึ่งสำหรับรับข้อมูล และอีกเส้นหนึ่งเป็นกราวด์ รายละเอียดและหน้าที่การทำงานของแต่ละขาเป็นดังนี้

Data Carrier Detect : DCD หรืออาจเรียกว่าการตรวจสอบสัญญาณพาห์ (Carrier Detect : CD) ขานี้จะกระตุ้นเมื่อมีการส่งสัญญาณพาห์จากอุปกรณ์สื่อสารข้อมูล เช่น โมเด็ม

Receive Data : RD หรือ **RxD** ขานี้ใช้เพื่อรับสัญญาณอนุกรมจากภายนอกเข้ามายังคอมพิวเตอร์โดยนำข้อมูลที่ได้ไปเก็บไว้ในรีจิสเตอร์พักข้อมูล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ขั้วต่อ DB-9	ขั้วต่อ DB-25	ชื่อสายสัญญาณ	ทิศทางของข้อมูล
1	8	Data Carrier Detect : DCD	อินพุต
2	3	Received Data : RxD	อินพุต
3	2	Transmitted Data : TxD	เอาต์พุต
4	20	Data Terminal Ready : DTR	เอาต์พุต
5	7	Signal Ground : GND	-
6	6	Data Set Ready : DSR	อินพุต
7	4	Request to Send : RTS	เอาต์พุต
8	5	Clear to Send : CTS	อินพุต
9	22	Ring Indicator : RI	อินพุต

ตารางที่ 2 รายละเอียดของขาต่างๆที่มีใช้อยู่ในปัจจุบัน

Transmitted Data : TD หรือ **TxD** ขานี้ใช้เพื่อส่งข้อมูลออกจากคอมพิวเตอร์ โดยนำข้อมูลที่เก็บอยู่ในบัฟเฟอร์สำหรับส่งข้อมูลส่งออกไป

Data Terminal Ready : DTR เป็นขาสัญญาณที่ส่งออกจากคอมพิวเตอร์เพื่อให้อุปกรณ์ปลายทางรับรู้ว่าต้องการจะติดต่อด้วย โดยขา DTR ต้องเชื่อมต่อกับขา DSR ของอุปกรณ์ปลายทาง และขา DTR ของอุปกรณ์ปลายทางต้องเชื่อมต่อกับขา DSR ของคอมพิวเตอร์ ถ้าใช้การเชื่อมต่อแบบ 3 สาย จะต้องต่อขา DTR และ DSR ของตัวมันเองเข้าด้วยกัน และต้องต่อขา DCD ด้วย ในกรณีที่โปรแกรมสื่อสารมีการตรวจจับคลื่นพาห์

Signal Ground : GND ขากราวด์ของระบบ

Data Set Ready : DSR ขานี้จะใช้คู่กับขา DTR เพื่อตรวจสอบการเชื่อมต่อกันระหว่างคอมพิวเตอร์กับอุปกรณ์ปลายทาง จะเป็นขาสำหรับรับข้อมูลจากภายนอกที่ส่งมาจากขา DTR

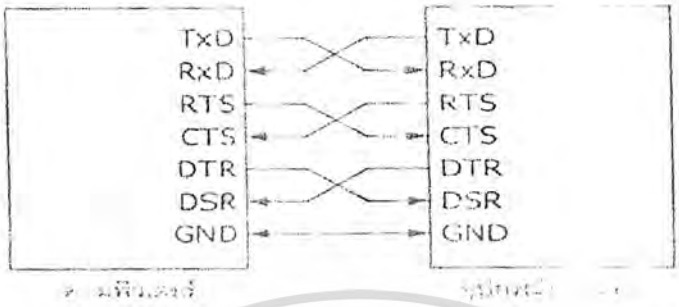
Request To Send : RTS เป็นขาสำหรับส่งสัญญารร้องขอให้อุปกรณ์ปลาบทางส่งข้อมูลกลับมายังคอมพิวเตอร์ โดยขาที่รับสัญญาณ RTS ก็คือขา CTS ในกรณีที่ใช้การเชื่อมต่อแบบ 3 สาย จะต้องเชื่อต่อขา RTS และ CTS ของตัวมันเองเข้าด้วยกัน เพื่อให้การรับและส่งข้อมูลเกิดขึ้นได้พร้อม ๆ ตลอดเวลา

Clear To Send : CTS ขานี้จะคอยรับสัญญาณจากขา RTS เมื่อรับสัญญาณได้ ข้อมูลที่ขา TxD จะส่งออกไป ขานี้จึงถูกใช้เพื่อตรวจสอบอุปกรณ์ต่อพ่วงว่าพร้อมจะรับข้อมูลหรือไม่

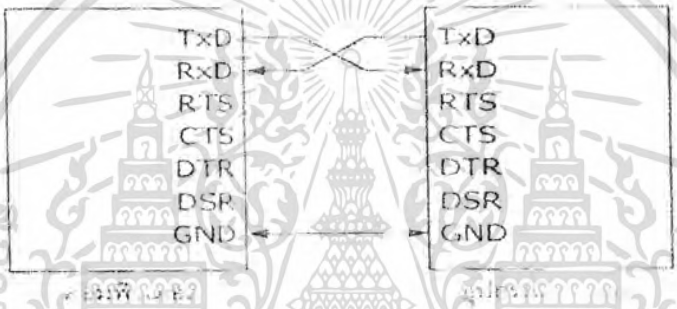
Ring Indicator : RI ใช้แสดงสถานะสัญญาณเรียกจากสายโทรศัพท์ ปกติในการสื่อสารโดยทั่วไปสายนี้จะไม่ถูกใช้งาน จะใช้งานก็ต่อเมื่อมีการเชื่อมต่อกับโมเด็มและโปแกรมมีการ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตรวจสอบสัญญาณนี้เท่านั้น



(ก) การต่ออุปกรณ์ภายนอกเข้ากับคอมพิวเตอร์
แบบ Null modem



(ข) การต่ออุปกรณ์ภายนอกเข้ากับคอมพิวเตอร์
แบบ RS-232 โดยใช้สายสัญญาณเพียง 3 เส้น

รูปที่ 14 แสดงการเชื่อมต่ออุปกรณ์ภายนอกกับคอมพิวเตอร์

2.4.3 การรับส่งข้อมูลแบบไม่เข้าจังหวะ

UART ย่อมาจากคำว่า Asynchronous Receiver Transmitter ซึ่งหมายถึงอุปกรณ์ที่ทำหน้าที่รับและส่งข้อมูลแบบไม่เข้าจังหวะกัน หน้าที่หลักของ ยูเออาร์ที คือทำหน้าที่แปลงข้อมูลแบบขนานของคอมพิวเตอร์ให้เป็นรูปแบบอนุกรมแบบไม่เข้าจังหวะแล้วส่งออกไป และทำหน้าที่แปลงข้อมูลแบบอนุกรมที่รับเข้ามาให้เป็นแบบขนานก่อนส่งให้คอมพิวเตอร์ นอกจากนี้ยังแจ้งข้อมูลอื่นๆ ให้คอมพิวเตอร์ทราบด้วย เช่น ความเร็วในการรับส่ง ภายใน ยูเออาร์ที จะมีส่วนของวงจรสร้าง บอดเรตแบบโปรแกรมได้ โดยการกำหนดค่าตัวหารให้กับสัญญาณนาฬิกาของ ยูเออาร์ที โดยตัวหารนี้มีขนาด 16 บิต ดังนั้นจึงสามารถกำหนดตัวหารอยู่ในช่วง 1- 65535 ยูเออาร์ที สามารถรับส่งข้อมูลได้ทั้งแบบฟูลดูเพล็กซ์ (Full Duplex) ซึ่งเป็นการรับและส่งข้อมูลพร้อมกัน และฮาล์ฟดูเพล็กซ์ (Half Duplex)เป็นการรับและส่งข้อมูลแบบทิศทางเดียวไม่พร้อมกัน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.4.4 ชนิดของ ยูเออาร์ที

ในเครื่องคอมพิวเตอร์ที่ใช้กันอยู่ทั่วไปมี ยูเออาร์ที ที่ใช้กันอยู่ 2 เบอร์คือ 8250 ซึ่งเป็น ยูเออาร์ที มาตรฐานที่ใช้กันมานานแล้ว ยูเออาร์ทีเบอร์นี้จะมีบัฟเฟอร์สำหรับรับและส่งข้อมูลตำแหน่งเดียวกัน ทำให้การรับและส่งข้อมูลถูกจำกัดความเร็วอยู่ที่ 57.6 กิโลบิตเท่านั้น และยูเออาร์ทีเบอร์นี้ถือว่าเป็นต้นแบบของยูเออาร์ที ที่ใช้ในคอมพิวเตอร์ โดยคอมพิวเตอร์ทุกรุ่นต้องสนับสนุนการทำงานตามรูปแบบของ ยูเออาร์ที เบอร์นี้

ยูเออาร์ที อีกเบอร์หนึ่งคือ 16450 มีความสามารถรับส่งข้อมูลได้ที่ความเร็ว 115,200 บิตต่อวินาที และเพิ่มรีจิสเตอร์สำหรับพักข้อมูลให้ ยูเออาร์ที นอกจากนี้ยังเพิ่มส่วนของชิฟต์รีจิสเตอร์แบบ เข้าก่อนออกก่อน (First In First Out : FIFO) ขนาด 16 ไบต์เข้าไปทำให้สนับสนุนความเร็วในการรับส่งข้อมูลที่ 256 กิโลบิต ต่อวินาที ปัจจุบันคอมพิวเตอร์ส่วนใหญ่ใช้ยูเออาร์ที เบอร์นี้ หรือ ใหม่กว่า เช่น เบอร์ TL16C750 ซึ่งมีรีจิสเตอร์แบบ เข้าก่อนออกก่อน ขนาด 64 ไบต์ ทำงานได้ที่ระดับแรงดัน +5โวลต์ และ - 3 โวลต์ มีโหมดประหยัดพลังงานสามารถรับส่งข้อมูลได้ที่ความเร็ว 1 เมกะบิตต่อวินาทีเมื่อใช้สัญญาณนาฬิกา 16 เมกะเฮิรตซ์ แต่ความเร็วในการส่งข้อมูลที่มากมายของ ยูเออาร์ที เบอร์ใหม่ๆ ก็ไม่ได้ช่วยให้การรับส่งข้อมูลของคอมพิวเตอร์เร็วขึ้น เพราะคอมพิวเตอร์ยังคงใช้ความถี่ของสัญญาณนาฬิกาในการแปลงข้อมูลเพียง 1.8432 เมกะเฮิรตซ์ เท่านั้น

2.4.5 วงจรภายในและรีจิสเตอร์ของพอร์ตอนุกรม RS-232

เครื่องคอมพิวเตอร์โดยทั่วไปสามารถต่อพอร์ตอนุกรมได้สูงสุด 4 พอร์ต มีชื่อเรียกคือ COM1 ,COM2 , COM3 ,COM4 จากรูปที่ 15 แสดงแผนผังภายในของพอร์ตอนุกรมซึ่งประกอบไปด้วยรีจิสเตอร์ขนาด 8 บิต 8 ตัว ที่ใช้งานร่วมกับ ยูเออาร์ที ตำแหน่งที่อยู่ของรีจิสเตอร์ภายในพอร์ตอนุกรมสามารถคำนวณได้จากค่ารีจิสเตอร์พื้นฐานของพอร์ตอนุกรม เช่น พอร์ตอนุกรม COM1 อยู่ที่ตำแหน่ง $3F8_H$ ตำแหน่งของรีจิสเตอร์ต่างๆ จะเป็นตำแหน่งที่บวกเข้าไปกับค่า $3F8_H$ โดยรีจิสเตอร์ที่ใช้กับพอร์ตอนุกรมมีดังนี้

00_H เป็นรีจิสเตอร์บัฟเฟอร์สำหรับรับข้อมูลที่รับเข้ามาหรือเตรียมข้อมูลที่จะส่งออกไป

01_H รีจิสเตอร์ควบคุมการทำงานของการขัดจังหวะ ใช้ในการเซตรูปแบบของการขัดจังหวะ

02_H รีจิสเตอร์แสดงรูปแบบการขัดจังหวะใช้ตรวจสอบรูปแบบของการขัดจังหวะเมื่อมีการขัดจังหวะ

03_H รีจิสเตอร์กำหนดรูปแบบของข้อมูล

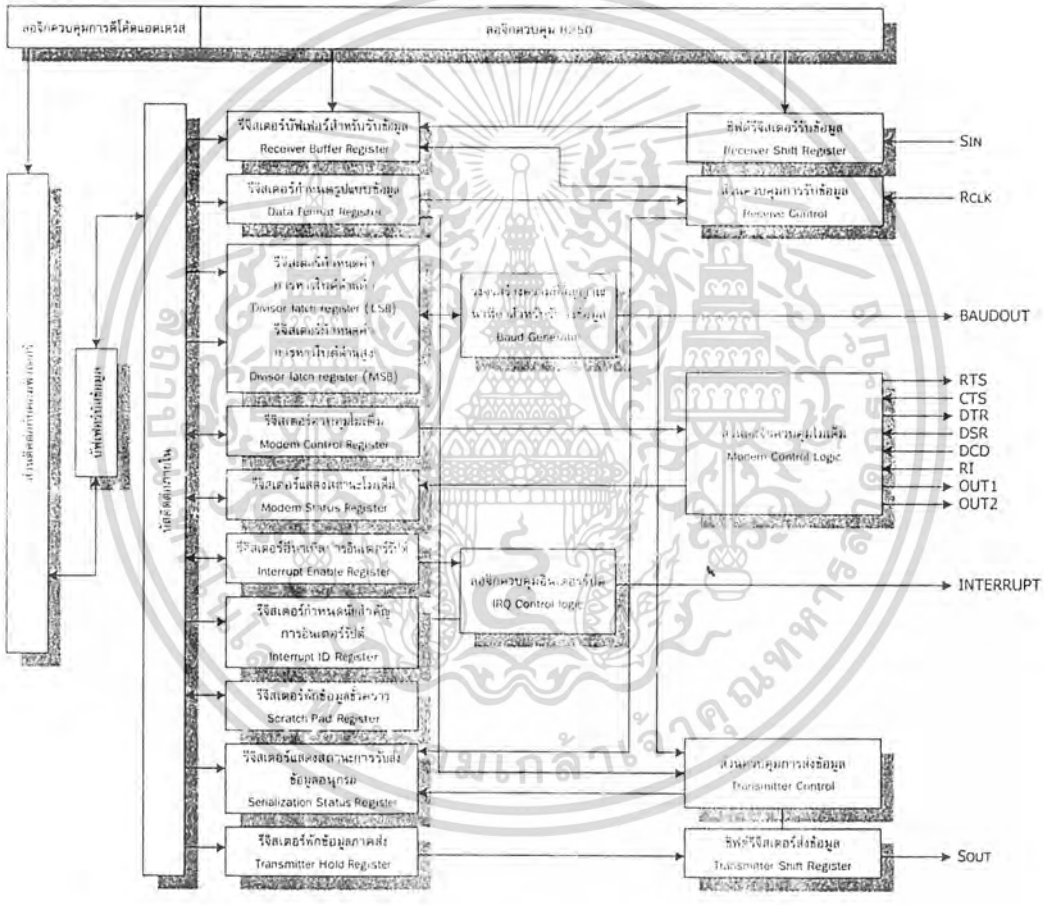
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

04_H รีจิสเตอร์ควบคุมโมเด็ม ใช้ตรวจสอบบิตสำหรับติดต่อกับโมเด็ม เช่น RTS , DTR

05_H รีจิสเตอร์แสดงสถานะการรับและการส่งข้อมูลแบบอนุกรม

06_H รีจิสเตอร์แสดงสถานะของโมเด็ม ซึ่งแสดงสถานะของขา DCD,RI,DSR และCTS

07_H รีจิสเตอร์สำหรับการเก็บข้อมูลชั่วคราว

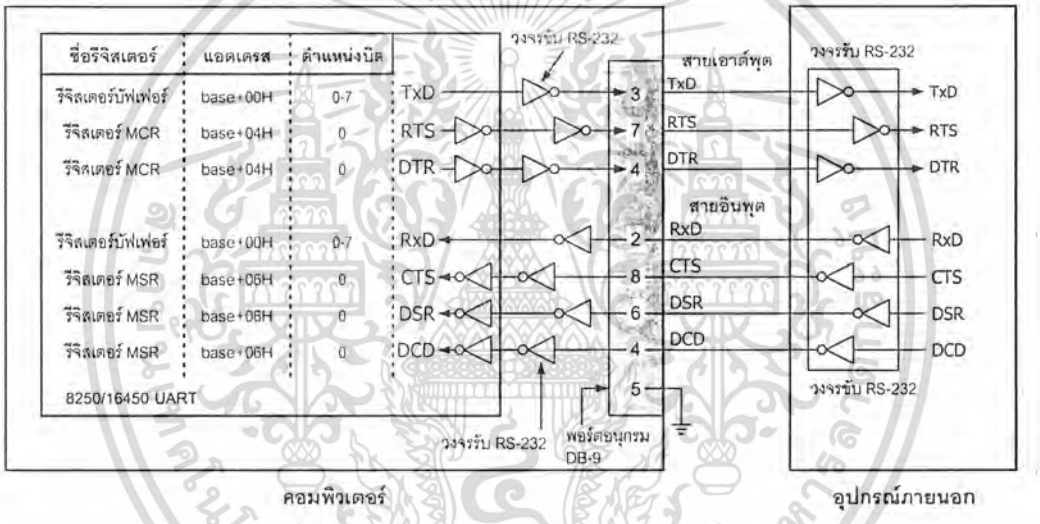


รูปที่ 15 แผนผังการทำงานภายในของพอร์ตอนุกรมของเครื่องคอมพิวเตอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.4.6 ลักษณะสัญญาณอินพุตและเอาต์พุตของพอร์ต RS-232

สัญญาณเอาต์พุตที่ใช้ควบคุมและสัญญาณแสงสถานะอินพุตของพอร์ตอนุกรม RS-232 จะถูกกลับสถานะภายในตัว ยูเออาร์ที ส่วนสัญญาณข้อมูลทั้งภาคส่งและภาครับจะไม่ถูกกลับสถานะ ยูเออาร์ที จะให้ระดับสัญญาณเอาต์พุตออกมาเป็น ทีทีแอล เท่านั้น ดังนั้นเมื่อสัญญาณถูกส่งออกมาจากยูเออาร์ทีจึงต้องส่งเข้าสู่วงจรขับเพื่อปรับระดับแรงดันให้ได้สัญญาณเป็นไปตามมาตรฐาน RS-232 ก่อนส่งออกไปจากคอมพิวเตอร์ สำหรับอุปกรณ์ต่อเชื่อมปลายทางก็จะต้องมีวงจรขับในลักษณะนี้เช่นกัน แต่วงจรขับที่ใช้ทั้งภายในคอมพิวเตอร์และอุปกรณ์ต่อเชื่อม ปลายทางนั้นจะถูกกลับสถานะ ดังแสดงในรูปที่ 16



รูปที่ 16 ผังแสดงโครงสร้างทางวงจรของพอร์ตอนุกรม

2.4.7 หมายเลขตำแหน่งของพอร์ตอนุกรม

ตำแหน่งพื้นฐานของพอร์ตอนุกรมมี 4 ตำแหน่งคือ

COM1 อยู่ที่ตำแหน่ง 3F8h

COM2 อยู่ที่ตำแหน่ง 2F8h

COM3 อยู่ที่ตำแหน่ง 3E8h

COM4 อยู่ที่ตำแหน่ง 2E8h

เมื่อเริ่มเปิดเครื่องคอมพิวเตอร์ระบบของคอมพิวเตอร์จะทำการตรวจสอบตำแหน่งของพอร์ตอนุกรมทั้งหมด และจะนำค่าหมายเลขตำแหน่งที่พบไปเก็บไว้ในหน่วยความจำขนาด 2 ไบต์ที่ตำแหน่งดังนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

COM1 0000:0400h - 0000:0401h

COM2 0000:0402h - 0000:0403h

COM3 0000:0404h - 0000:0405h

COM4 0000:0406h - 0000:0407h

ส่วนหน่วยความจำตำแหน่ง 0000:0411h ยังใช้สำหรับแสดงจำนวนของพอร์ตอนุกรมที่มีใช้อยู่ในคอมพิวเตอร์ดังแสดงในตารางที่ 3

บิต 3	บิต 2	บิต 1	จำนวนพอร์ต
0	0	0	ไม่มีพอร์ตอนุกรม
0	0	1	มี 1 พอร์ต
0	1	0	มี 2 พอร์ต
0	1	1	มี 3 พอร์ต
1	0	0	มี 4 พอร์ต

ตารางที่ 3 แสดงข้อมูลในตำแหน่ง 0000:0411h ที่ใช้แจ้งจำนวนพอร์ตอนุกรม

2.5 ระบบเชื่อมต่อแบบ RS-422

การติดต่อระหว่างคอมพิวเตอร์กับอุปกรณ์ปลายทางแบบอนุกรม ทางพอร์ตอนุกรมของคอมพิวเตอร์ โดยปกติคอมพิวเตอร์จะส่งข้อมูลของพอร์ตอนุกรมเป็นมาตรฐานสัญญาณแบบ RS-232 ซึ่งสามารถรับส่งข้อมูลอนุกรมในระยะทางได้เพียง ไม่เกิน 50 ฟุต และอัตราการรับส่งข้อมูลสูงสุดเพียง 20,000 บิตต่อวินาที อย่างไรก็ตามสามารถส่งให้ไกลกว่านี้ได้โดยจะต้องคำนึงถึงค่าความต้านทานในสายให้น้อยที่สุด

ปัญหาดังกล่าวสามารถแก้ไขได้โดย แปลงระดับสัญญาณมาตรฐาน RS-232 เป็น มาตรฐาน RS-422 ซึ่งมาตรฐาน RS-422 นี้สามารถรับส่งข้อมูลได้ ไกลถึง 4,000 ฟุต และมีอัตราการรับส่งข้อมูลสูงสุดถึง 10 ล้านบิตต่อวินาที เพราะมีการแยกสายสัญญาณออกเป็น 2 เส้น ต่อ 1 สัญญาณ โดยไม่ใช้กราวด์ร่วมกัน อีกทั้งยังใช้ระดับสัญญาณที่ +5 โวลต์ เท่านั้นจึงไม่จำเป็นต้องใช้แหล่งจ่ายไฟเพิ่มเติม ในหนึ่งเส้นสัญญาณสายทั้งสองจะตีเกลียวไปด้วยกัน ทำให้อิมพิแดนซ์เท่ากันและกระแสในสายจะเท่ากันแต่ทิศทางตรงข้ามกัน ดังนั้นสัญญาณรบกวนต่างๆ จะหักล้างกันหมด ส่วนสายกราวด์ นั้นจะใช้อ้างอิงระดับแรงดันเท่านั้นโดยไม่ได้เป็นทางผ่านสัญญาณแต่อย่างใดในการส่งสัญญาณแบบนี้ในขณะที่สายเส้นหนึ่งมีแรงดัน +5 โวลต์ สายอีกเส้นที่คู่กันจะ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

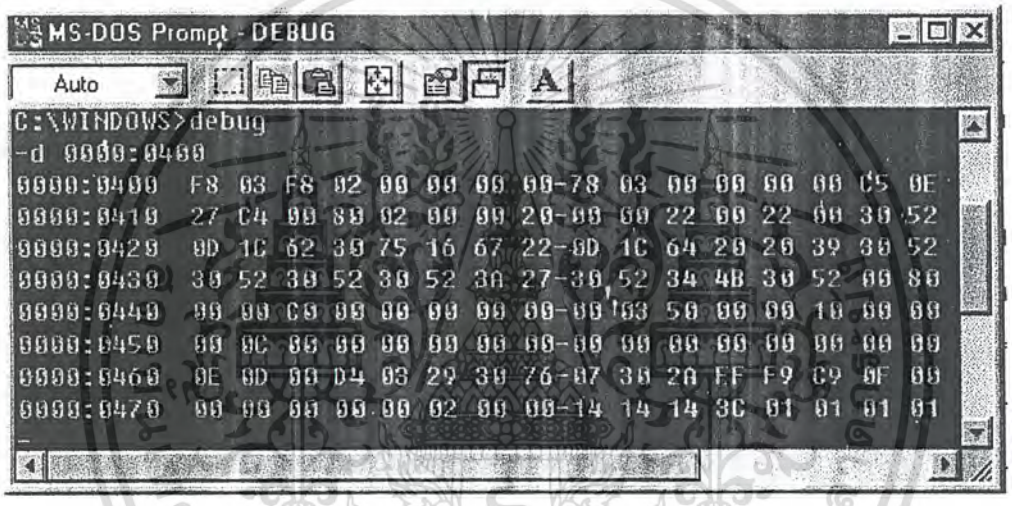
เป็น 0 โวลต์ และในทางตรงกันข้าม เมื่อสายอีกเส้นหนึ่งเป็น 0 โวลต์ อีกสายก็จะเป็น +5 โวลต์ ดังนั้นการสวิงของแรงดันที่ปรากฏระหว่าง 2 สาย จะมีค่าเป็น 10 โวลต์

2.6 การเขียนโปแกรมเพื่อใช้งานพอร์ตอนุกรมของคอมพิวเตอร์

2.6.1 การหาค่าตำแหน่งแดดเดรสของพอร์ตอนุกรม

การหาค่าตำแหน่งแอดเดรสของพอร์ตอนุกรมสามารถทำได้หลายวิธีคือ

1) โดยการใช้โปรแกรมDebugไปดูค่าตำแหน่งแอดเดรสที่ตำแหน่ง 000:0400H โดยใช้พิมพ์คำสั่งที่ดอสพร้อมพ์ดังแสดงในรูปที่ 17



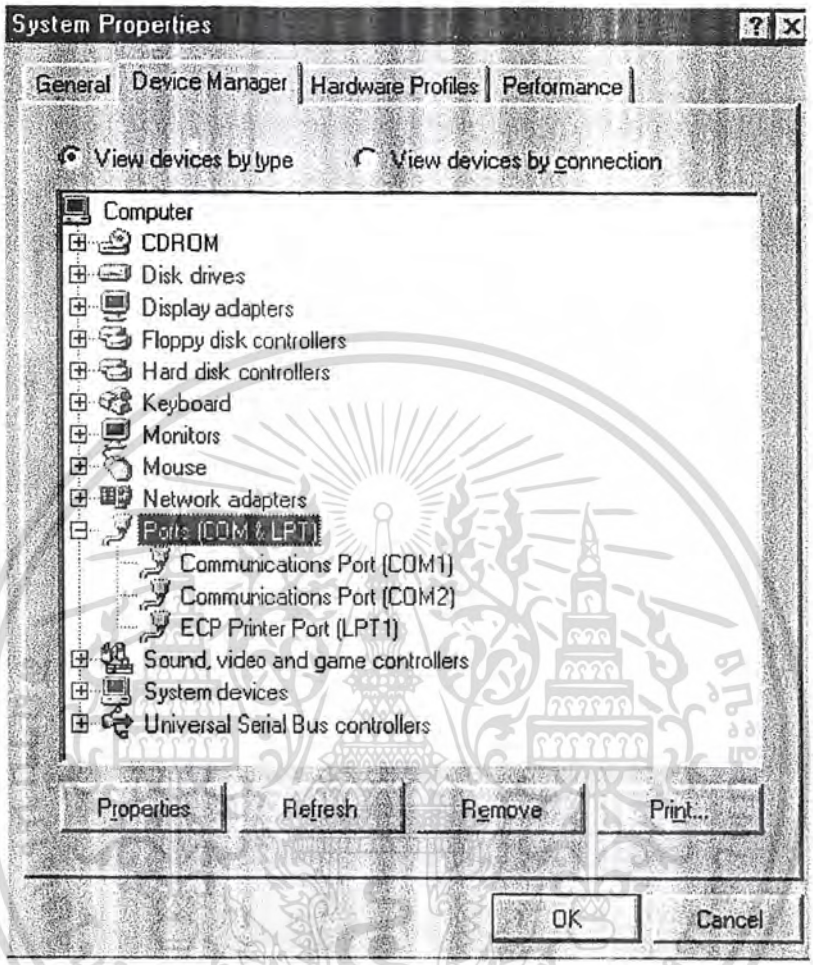
รูปที่ 17 การใช้โปแกรม ดีบักเพื่อหาค่าตำแหน่งของ พอร์ตอนุกรม

ค่าที่เห็นในรูปแสดงว่า พอร์ตอนุกรมของคอมพิวเตอร์เครื่องนี้มีถึง 4 พอร์ต มีตำแหน่งไล่เรียงกันตั้งแต่หน่วยความจำตำแหน่งที่ 0000:0400H - 0000:0407H ตัวอย่างเช่นที่ตำแหน่งหน่วยความจำ 0000:0400H - 0000 : 0401H แสดงตัวเลขเลข F8 03 ซึ่งหมายความว่าแอดเดรสของพอร์ต COM1 คือ 03F8H นั่นเอง สำหรับจำนวนของพอร์ตอนุกรมที่ระบุอยู่ที่หน่วยความจำตำแหน่ง 0000 : 0411H มีค่าเท่ากับ D8H ซึ่งเมื่อแปลงเป็นเลขฐานสองจะได้ค่าเป็น 1011000 บิตที่ 1-3 มีค่าเท่ากับ 100 หมายความว่ามีจำนวนพอร์ตอนุกรมทั้งหมด 4 พอร์ตดังที่ได้กล่าวไปแล้วตอนต้น

2) สามารถดูได้จากวินโดวส์ 95 ขึ้นไปโดยไปที่ Control Panel เรียก System => Device Manager => Ports (COM &LPT) จากนั้นเลือกพอร์ตอนุกรมที่ต้องการดู ค่าแล้วดังแสดงในรูปที่ 18 แล้ว Properties => Resources ดังแสดงในรูปที่ 19 ซึ่งในหน้าต่าง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

นี้จะแสดงทั้งตำแหน่งของพอร์ตอนุกรมนั้น ๆ รวมถึงตำแหน่งของการขัดจังหวะที่ใช้ด้วย

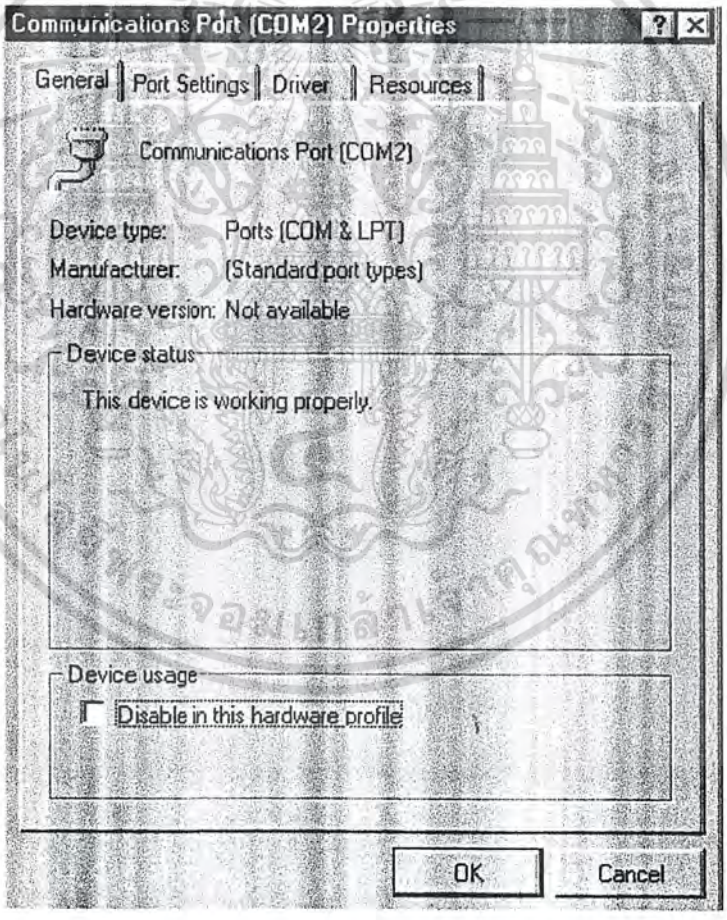


รูปที่ 18 แสดงตำแหน่งของพอร์ตอนุกรมบนระบบปฏิบัติการ วินโดวส์ 98

3) การเขียนโปแกรมเพื่ออ่านค่า โดยสามารถใช้โปรแกรมภาษาใด ๆ ก็ได้เพื่ออ่านค่าแต่ในตัวอย่างนี้จะใช้ QBASIC ในการแสดงค่าแอดเดรสของพอร์ตอนุกรม โดยจะใช้คำสั่ง PEEK ซึ่งเป็นคำสั่งที่ใช้อ่านค่าจากหน่วยความจำ นอกจากนี้ยังอ่านค่าจำนวนของพอร์ตอนุกรมออกมาโดยใช้คำสั่งเดียวกัน แต่จะใช้คำสั่ง AND เข้าช่วยเพื่อเลือกเอาเฉพาะบิตที่ต้องการอ่านค่าเท่านั้น จากนั้น ก็ทำการเลื่อนบิตไปทางขวา 1 บิต โดยใช้วิธีการหารด้วย 2 เพื่อให้บิตที่ต้องการไปอยู่ด้านขวามือสุดและแสดงค่าจำนวนพอร์ตที่แท้จริงออกมา คำสั่ง HEX$ ช่วยให้การแสดงผลตำแหน่งแอดเดรสของพอร์ตอนุกรมที่ออกมาเป็นค่าเลขฐานสิบหก ส่วนการคูณค่าด้วย 100H นั้นก็เพื่อที่จะเลื่อนบิตไปด้านหน้า 1 ไบต์ทำให้เมื่อเวลานำมาบวกแล้วจะได้ค่าเป็น 2 ไบต์พอดี มีโปรแกรมตัวอย่างแสดงดังนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
DEE SEG =0
CLS
PRINT "Address of COM1 :", HEX$ ( ( PEEK ( &H401 ) * &H100 ) + PEEK ( &H400 ))
PRINT "Address of COM2 :", HEX$ ( ( PEEK ( &H403 ) * &H100 ) + PEEK ( &H402 ))
PRINT "Address of COM3 :", HEX$ ( ( PEEK ( &H405 ) * &H100 ) + PEEK ( &H404 ))
PRINT "Address of COM4 :", HEX$ ( ( PEEK ( &H407 ) * &H100 ) + PEEK ( &H406 ))
PRINT "Number of RS – 232 Port in This Computer :", (PEEK ( &H411 ) AND
&HE) /2
END
```



รูปที่ 19 แสดงรายละเอียดของพอร์ตอนุกรม COM2 บนระบบปฏิบัติการ วินโดวส์ 98

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.6.2 การกำหนดค่าเริ่มต้นให้กับพอร์ตอนุกรม

ก่อนการใช้งานพอร์ตอนุกรมนั้นจะต้องมีการกำหนดค่าเริ่มต้นให้กับตัวมันก่อน ซึ่งก็คือการกำหนดจำนวนบิตข้อมูลที่ต้องการส่ง , จำนวนบิตปิดท้าย , ชนิดของพาริตี้ที่ใช้ และความเร็วในการส่งข้อมูล การกำหนดสามารถทำได้หลายวิธี

1) เป็นการกำหนดจากดอสพรอมพ์โดยใช้คำสั่ง MODE ซึ่งมีวิธีการใช้งานดังนี้

MODE COMm : baud=b, parity=p, data=d, stop=s, retry=r หรือ MODE COMm : b,p,d,s,r

ตัวอย่าง MODE COM1 : 9600, n,8,1 จะเป็นการกำหนดให้พอร์ตอนุกรม COM1 มีความเร็วในการรับส่งข้อมูลเท่ากับ 9600 บิตต่อวินาที ไม่มีการตรวจสอบพาริตี้ รับส่งข้อมูลแบบ 8 บิต และมีบิตปิดท้าย 1 บิต

2) เป็นการกำหนดโดยใช้การขัดจังหวะของดอส ตำแหน่งที่ 14H ซึ่งการใช้งานจะต้องกำหนดค่าต่าง ๆ ลงในรีจิสเตอร์ด้วย โดยจะต้องกำหนดให้รีจิสเตอร์ AH มีค่าเท่ากับ 0 รีจิสเตอร์ DX เก็บค่าของพอร์ตอนุกรมที่ต้องการกำหนดค่าเริ่มต้น โดย

DX = 0 จะกำหนดให้กับพอร์ตอนุกรม COM1

DX = 1 จะกำหนดให้กับพอร์ตอนุกรม COM2

DX = 2 จะกำหนดให้กับพอร์ตอนุกรม COM3

DX = 3 จะกำหนดให้กับพอร์ตอนุกรม COM4

รีจิตเตอร์ AL ซึ่งมีขนาด 8 บิตใช้เก็บค่าเริ่มต้นต่าง ๆ มีรายละเอียดดังนี้

บิต 7	บิต 6	บิต 5	บิต 4	บิต 3	บิต 2	บิต 1	บิต 0
BD2	BD1	BDO	PAR1	STOP	STOP	DA1	DAO

BD2,BD1,BDO ใช้สำหรับกำหนดค่าบอดรต

"111" บอดรตเท่ากับ 9,600 บิตต่อวินาที

"110" บอดเรตเท่ากับ 4,800 บิตต่อวินาที

"101" บอดเรตเท่ากับ 2,400 บิตต่อวินาที

"100" บอดเรตเท่ากับ 1,200 บิตต่อวินาที

"011" บอดเรตเท่ากับ 600 บิตต่อวินาที

"010" บอดเรตเท่ากับ 300 บิตต่อวินาที

"001" บอดเรตเท่ากับ 150 บิตต่อวินาที

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

"000" บอดเรตเท่ากับ 110 บิตต่อวินาที

PAR1,PARO ใช้กำหนดค่าพาริตี้โดย

"00" หรือ "10" ไม่มีการตรวจสอบพาริตี้

"01" พาริตี้คี่

"11" พาริตี้คู่

STOP ใช้กำหนดจำนวนของบิตปิดท้าย

"1" มีบิตปิดท้ายเท่ากับ 2 บิต

"0" มีบิตปิดท้ายเท่ากับ 1 บิต

DA1,DAO ใช้กำหนดความยาวของข้อมูลโดย

"10" ความยาวข้อมูลเท่ากับ 7 บิต

"11" ความยาวข้อมูลเท่ากับ 8 บิต

2.6.3.การรับและส่งข้อมูลแบบอนุกรม

มีหลากหลายวิธีในการรับและส่งข้อมูลแบบอนุกรมผ่านพอร์ตอนุกรม เช่น ใช้คำสั่งพิมพ์ออกทางเครื่องพิมพ์ เรียกอินเตอร์รัปต์ของไบออสหรือของดอสการเขียนหรืออ่านไปยังแอดดรสของพอร์ตโดยตรง วิธีสุดท้ายเป็นวิธีที่มีความยืดหยุ่นในการใช้งานที่สุด ยกตัวอย่าง ถ้าต้องการส่งข้อมูลไปยังพอร์ตอนุกรม COM1 สามารถเขียนข้อมูลโดยตรงไปที่รีจิสเตอร์พักข้อมูลหรับส่งข้อมูล (แอดเดรส 3F8H) โดยใช้คำสั่งภาษา QBASIC ง่าย ๆ ดังนี้

OUT & H3F8, X

ค่า X ในที่นี้หมายถึงข้อมูลที่ต้องการส่ง มีขนาด 8 บิต

สำหรับการอ่านข้อมูลจากพอร์ตอนุกรม จะเป็นการอ่านข้อมูลมาจากรีจิสเตอร์พักข้อมูลสำหรับรับข้อมูล (แอดเดรส 3F8H เช่นเดียวกัน) ซึ่งสามารถเขียนโปรแกรมง่าย ๆ ได้ดังนี้

Y = INP (&H3F8)

ค่า Y ในที่นี้คือค่าที่อ่านได้จากรีจิสเตอร์พักข้อมูลสำหรับรับข้อมูล โดยมีขนาด 8 บิต แต่เมื่อใช้คำสั่งนี้ในขณะที่โปรแกรมทำงานผ่านระบบปฏิบัติการวินโดวส์ จะไม่สามารพใช้งานได้เนื่องจากระบบปฏิบัติการวินโดว์ ได้เข้าฝังตัวพอร์ตอนุกรมเข้าเป็นส่วนหนึ่งของระบบปฏิบัติการแล้ว ดังนั้นการเรียกใช้งานจึงจำเป็นต้องเรียนกผ่านครื่องมือที่ติดต่อผ่านระบบปฏิบัติการวินโดว์ เช่น การใช้คอนโทรล MSCOMM32.OCX ของโปรแกรม วิชวลเบสิก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.7 คอนโทรล เอ็มเอสคอมมิวนิเคชั่นคอนโทรลหรือ เอ็มเอสคอมม์ (MSComm)

สำหรับการใช้งาน วิชวลเบสิก ตั้งแต่เวอร์ชัน 2 เป็นต้นมาจะมีคอนโทรลสำหรับการสื่อสารอนุกรมผ่านทางพอร์ตอนุกรมของคอมพิวเตอร์มาให้ โดยใน วิชวลเบสิก เวอร์ชัน 2 และเวอร์ชัน 3 จะใช้ชื่อว่า MSCOMM.VBX ส่วนเวอร์ชัน 4 ใช้ชื่อว่า MSCOMM 16.OCX สำหรับการทำงานกับระบบปฏิบัติการ 16 บิต และ MSCOMM32.OCX สำหรับการทำงานกับระบบปฏิบัติการ 32 บิต สำหรับในวิชวลเบสิก เวอร์ชัน 5 จะมีเพียง MSCOMM32.OCX เท่านั้นเพราะถูกออกแบบมาให้ใช้งานกับระบบปฏิบัติการ 32 บิต

เอ็มเอสคอมม์ จัดเตรียมทางเลือกเอาไว้ 2 ทางเพื่อความสะดวกในการสื่อสารข้อมูล ทางแรก คือ การสื่อสารข้อมูลที่กระตุ้นด้วยเหตุการณ์ (event-driven communications) เป็นรูปแบบการใช้งานที่มีประสิทธิภาพมากสำหรับการตอบสนองแบบทันทีทันใด เช่น เมื่อตัวอักษรถูกส่งมาที่พอร์ตอนุกรมหรือเกิดการเปลี่ยนแปลงที่ขาตรวจจับสัญญาณพาห์ (Data Carrier Detect : DCD) หรือขาร้องขอเพื่อส่งข้อมูล (Request To Send RTS) เหตุการณ์ ออนคอม์ (Oncomm) ของเอ็มเอสคอมม์จะสามารถตรวจจับสัญญาณนั้นได้ทันที ซึ่งจะกล่าวถึงรายละเอียดในหัวข้อคุณสมบัติ คอมมือีเวนท์ ต่อไป ส่วนทางเลือกที่สองเป็นการคอยตรวจสอบค่าเหตุการณ์และความผิดพลาดที่เกิดขึ้นด้วยการดูค่าที่เปลี่ยนแปลงภายในคุณสมบัติคอมม์อีเวนท์ (CommEvent) หลังจากให้โปรแกรมทำงานในฟังก์ชั่นต่าง ๆ ไปเรียบร้อยแล้ว ซึ่งวิธีนี้ใช้งานได้ดีในกรณีที่โปรแกรมมีขนาดเล็ก

คอนโทรลเอ็มเอสคอมม์ 1 ตัวสามารถควบคุมการทำงานของพอร์ตอนุกรมได้ 1 พอร์ต ถ้าในโปรแกรมที่ใช้งานต้องการติดต่อกับพอร์ตอนุกรมมากกว่า 1 พอร์ตจะต้องใช้คอนโทรลเอ็มเอสคอมม์มากกว่า 1 ตัวเพื่อควบคุมพอร์ตอนุกรมในแต่ละพอร์ต แอดเดรสพอร์ตอนุกรมและเอดเดรสของการเกิดอินเตอร์รัปต์สามารถเปลี่ยนแปลงได้จากการแก้ไขค่าที่คอนโทรลพาเนล (Control Panel) ถึงแม้ว่า คอนโทรล เอ็มเอสคอมม์ จะมีคุณสมบัติ มากมาย แต่สามารถทำความเข้าใจได้ไม่ยากดังนี้

2.7.1 CommPort

ใช้ในการกำหนดและอ่านค่าพอร์ตอนุกรมที่ติดต่ออยู่ (COM1,COM2,COM3, COM4) รูปแบบการใช้งานคือ

object.CommPort [= value]

โดย Value เป็นค่าของพอร์ตอนุกรม ชนิดของข้อมูลเป็นเลขจำนวนเต็ม ค่า Value สามารถกำหนดได้ในช่วง 1-16 (ค่าเริ่มต้นกำหนดไว้ที่ 1) เมื่อมีกำหนดค่าแล้วทำการเปิดพอร์ตโดยใช้คุณสมบัติ PortOpen แต่ว่าพอร์ตนั้นไม่มีอยู่ในระบบเอ็มเอสคอมม์จะสร้างสัญญาณแสดง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ข้อผิดพลาด error68 ขึ้นมา ซึ่งหมายถึง อุปกรณ์ตัวนี้ไม่มีอยู่ในระบบ ดังนั้นการเขียนโปรแกรมจึงจำเป็นต้องกำหนดตำแหน่งของพอร์ตอนุกรมก่อนที่ใช้คำสั่ง OpenPort

2.7.2 Setting

ใช้ในการกำหนดและอ่านค่าอัตราเร็วในการรับส่งข้อมูล , พาริตี้ , จำนวนของบิตข้อมูล , จำนวนของบิตปิดท้าย รูปแบบการใช้งานคือ

object.settings [= value]

ค่า Value มีชนิดข้อมูลเป็นแบบ ข้อความ มีรูปแบบเป็น "BBBB,P,D,S" โดย BBBB เป็นค่าอัตราเร็วในการรับส่งข้อมูล,P เป็นค่าพาริตี้ ,D เป็นจำนวนของบิตข้อมูล และ S เป็นจำนวนของบิตปิดท้าย ปกติแล้วค่านี้ถูกกำหนดไว้เป็น "9600,N, 8, 1" ค่าความเร็วในการรับส่งข้อมูลมาตรฐานที่ใช้กับเอ็มเอสคอมม์ มีดังนี้

110	บิต่อวินาที	
300	บิตต่อนาที	
600	บิตต่อนาที	
1200	บิตต่อนาที	
2400	บิตต่อนาที	
9600	บิตต่อนาที	(ค่าปกติ)
14400	บิตต่อนาที	
19200	บิตต่อนาที	
28800	บิตต่อนาที	
38400	บิตต่อนาที	(สวงน)
56000	บิตต่อนาที	(สวงน)
128000	บิตต่อนาที	(สวงน)

สำหรับค่ามาตรฐานในการกำหนดค่าพาริตี้มีดังนี้

สัญลักษณ์	รายละเอียด
E	พาริตี้คู่ (Evcn)
M	ลอจิก "1" (MARK)
N	ไม่ใช้ (ค่าปกติ)
O	พาริตี้คี่ (Odd)
S	ลิจิก "0" (Space)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ค่าที่ใช้ในการกำหนดจำนวนบิตมี 5 ค่าคือ 4,5,6,7 และ 8 (เป็นค่าปกติ)
ค่าที่ระบุจำนวนบิตปิดท้ายมี 3 ค่าคือ 1 (เป็นค่าปกติ) 1.5 และ 2
ตัวอย่างการใช้งานคำสั่ง Settings โดยจะเป็นการกำหนดค่าอัตราเร็วในการรับส่งข้อมูล ท่ากับ 9600 ไม่มีพาริตี้ จำนวนบิตข้อมูล 8 บิต และบิตปิดท้าย 1 บิต สามารถเขียนโปรแกรมได้ดังนี้

MSComml . Settubgs = "9600,N,8,1"

สาเหตุที่ค่าที่กำหนดจะต้องอยู่ภายในเครื่องหมายคำพูด "" เนื่องจาก ค่าที่กำหนดนี้อยู่ในรูปตัวแปรแบบข้อความ

2.7.3 PortOpen

ใช้ในการกำหนดและอ่านค่าสถานะของพอร์ตอนุกรม เพื่อเปิดและปิดพอร์ตอนุกรม รูปแบบการใช้งานคือ

object. PortOpen [= value]

ค่า Value มีชนิดข้อมูลเป็นแบบบูลีนคือ จริง(Truc) กับ เท็จ (False)โดย จริงหมายถึงการเปิดพอร์ตออนุกรมและเท็จหมายถึงการปิดพอร์ตอนุกรม สำหรับการปิดพอร์ตนั้นจะมีการเคลีย์ตัวพักข้อมูลด้านรับและตัวพักข้อมูลด้านส่งด้วย คอนโทรลเอ็มเอสคอมม์ จะปิดพอร์ตอนุกรมโดยอัตโนมัติเมื่อออกจากโปรแกรม ก่อนที่จะใช้คุณสมบัติ PortOpen ต้องตรวจสอบให้แน่ใจก่อนว่าคุณสมบัติ Commport นั้นได้ทำการกำหนดตำแหน่งของพอร์ตอนุกรมไว้ถูกต้องหรือไม่ มิเช่นนั้น เอ็มเอสคอมม์ จะแสดงข้อผิดพลาด Error68 แจ้งแก่ผู้ใช้งาน หรือถ้าพอร์ตอนุกรมนั้นถูกเปิดเอาไว้แล้ว โปรแกรมก็จะแจ้งข้อผิดพลาดออกมาเช่นเดียวกัน

ถ้าคุณสมบัติ DTREnable หรือ RTSEnable ถูกกำหนดให้เป็นจริงก่อนที่จะทำการเปิดพอร์ต ค่าคุณสมบัติของ DTREnable หรือ RTSEnable จะถูกเซตเป็น เท็จหลังจากปิดพอร์ต แต่ถ้าเซตเป็นเท็จ หลังจากปิดโปรแกรมแล้ว ค่าที่กำหนดไว้จะเป็นค่าเดิม ตัวอย่างการใช้คำสั่งเปิดพอร์ต เพื่อติดต่อสื่อสารกับพอร์ตอนุกรม COM1 และมีอัตราเร็วในการรับส่ง 9600 บิตต่อวินาที ไม่มีพาริตี้ จำนวนบิตข้อมูล 8 บิต และบิตปิดท้าย 1 บิต มีดังนี้

```
MSComm . Settings    = "9600,8,1"
MSComm . CommPort    = 1
MSComm . PortOpen    = True
```

2.7.4 Input

ใช้อ่านค่าและลบค่าขอบวนข้อมูลจากตัวพักข้อมูลภาครับ รูปแบบการใช้งานคือ

object . Input

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

คุณสมบัติ InputLen เป็นตัวกำหนดจำนวนของตัวอักษรที่จะอ่านโดยคุณสมบัติ Input กำหนดค่าให้ InputLen เท่ากับ 0 เป็นการกำหนดให้คุณสมบัติ Input ทำการอ่านค่าข้อมูลในตัวพักข้อมูลภาครับทั้งหมด คุณสมบัติ InputMode เป็นตัวกำหนดชนิดของข้อมูลที่คุณสมบัติ Input รับเข้ามาถ้า InputMode ถูกกำหนดเป็น comInputModeText คุณสมบัติ Input จะส่งค่าข้อมูลกลับมาในรูปแบบของข้อความชนิดข้อมูลเป็นแบบแปรเปลี่ยนได้ (Variant) ถ้า InputMode กำหนดเป็น comInputModeBiary คุณสมบัติ Input จะส่งข้อมูลกลับมาในรูปของไบนารี่ (Binary) และชนิดข้อมูลเป็นแบบ แปรเปลี่ยนได้ ตัวอย่างโปรแกรมแสดงให้เห็นถึงวิธีในการรับข้อมูลจากบัฟเฟอร์รับข้อมูลทั้งหมด

```
Private Sub Commandl_Click ()
Dim InString as String
MSComm . ImputLen = 0          ' Retrieve all available data.
If MSComml . UbBufferCount Then      ' Check for data.
      InString = MSComml . Input          ' Read data.
End If
End Sub
```

2.7.5 InBufferCount

ใช้ส่งค่าจำนวนของตัวอักษรที่อยู่ในตัวพักข้อมูลภาครับรูปแบบการใช้งานคำสั่งคือ

object.InBufferCount [= value]

คำสั่ง InBufferCount จะแสดงค่าจำนวนของตัวอักษร ซึ่งรับมาจากภายนอกและยังเก็บอยู่ในตัวพักข้อมูลภาครับ เพื่อให้ผู้ใช้งานอ่านค่าออกไป สำหรับการเคลียร์ค่าตัวพักข้อมูลภาครับทำได้โดยกำหนดให้ InBufferCont มีค่าเป็น 0

2.7.6 InBufferSize

ใช้กำหนดและคืนค่าขนาดตัวพักข้อมูลภาครับในหน่วยเป็นไบต์ โดยค่าเริ่มต้นถูกกำหนดไว้ที่ 1024 ไบต์ รูปแบบการใช้งานคำสั่งคือ

object . InBuffSize

การกำหนดค่าตัวพักข้อมูลภาครับขนาดใหญ่จะทำให้ หน่วยความจำที่เหลือสำหรับการใช้งานส่วนอื่น ๆ จะเหลือน้อย อย่างไรก็ตามการกำหนดค่า บัฟเฟอร์ภาครับที่น้อยเกินไปจะทำให้เกิดการโอเวอร์โฟลวหรือข้อมูลล้นตัวพักข้อมูล เว้นแต่จะมีการใช้แฮนด์เช็ก ดังนั้นค่าปานกลางที่

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เหมาะสมก็คือค่า 1024 ซึ่งเป็นค่าเริ่มต้นนั่นเอง แต่ถ้าโปรแกรมมีการเกิดข้อมูลล้นแล้วจึงปรับเพิ่ม ค่าขนาดของตัวพักข้อมูลให้มีค่ามากขึ้น

2.7.7 InputLen

ใช้กำหนดค่าและคืนค่าจำนวนของตัวอักษรที่อ่านจากตัวพักข้อมูลภาครับ มีรูปแบบการใช้งานคำสั่งคือ

object . InputLen [= value]

ค่าเริ่มต้นของคุณสมบัติ InputLen มีค่าเท่ากับ "0" การกำหนดค่าเท่ากับ "0" จะทำให้คำสั่ง Inpuut ของ เอ็มเอสคอมม์ อ่านค่าข้อมูลที่อยู่ภายในตัวพักข้อมูลภาครับทั้งหมด

ถ้าไม่มีข้อมูลอยู่ในตัวพักข้อมูลภาครับมากเท่ากับจำนวน InputLen คำสั่ง Input จะส่งค่าว่าง ("") กลับออกมา ผู้ใช้งานสามารถตรวจสอบข้อมูลในตัวพักข้อมูลภาครับได้โดยใช้คุณสมบัติ InBufferCount โดยกำหนดให้มีข้อมูลอยู่ในบัฟเฟอร์ภาครับก่อนแล้วจึงค่อยอ่านข้อมูลจากบัฟเฟอร์ภาครับ

คุณสมบัตินี้มักใช้กับการอ่านค่าข้อมูลจากเครื่องมือหรือเครื่องจักรที่มีการกำหนดค่าขนาดความยาวของข้อมูลเอาไว้แล้ว ตัวอย่างโปรแกรมการอ่านค่าตัวอักษรออกมา 10 ตัวอักษร

```
Private Commandl_Click ( )
Dim CommData as String
MSComml . InputLen = 10      ' Specify a 10 character block for data.
CommData = MSComml . Input   ' Read data.
```

2.7.8 InputMode

ใช้กำหนดค่าและคืนค่าชนิดของข้อมูลที่รับโดยคำสั่ง Input มีรูปแบบการใช้งานคำสั่งคือ

object . InputMode [= value]

คุณสมบัติ InputMode ใช้กำหนดว่าข้อมูลชนิดไหนที่รับข้ามาผ่านคำสั่ง Input โดยข้อมูลจะเลือกได้ 2 ประเภทคือ

1) comlnputModeText สำหรับข้อมูลที่อยู่ในรูปข้อความตัวอักษรตามมาตรฐานแอนซี่(ANSI) โดยจะต้องกำหนดค่าเป็น " 0 " และค่าเริ่มต้นของการรับค่าข้อมูลก็จะเป็นค่านี้

2) ComlnputModeBinary สำหรับข้อมูล ซึ่งจะเก็บในรูปไบนารี่รวมกันอยู่เป็นไบต์ข้อมูล

ตัวอย่างการใช้งาน InputMode ต่อไปนี้จะทำการอ่านค่าข้อมูล 10 ไบต์จากพอร์ตอนุกรมและเก็บข้อมูลไว้ในตัวแปรแบบอาเรย์ ชนิดข้อมูลเป็นแบบไบต์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
Private Sub Commandl_Click ( )
        Dim Buffer as Variant
        Din Arr ( ) as Byte
        MSComml . CommPort = 1 ' Set and open port
        Mscomml . PortOpen = True
        MSComml . InputMode = comInputModeBinary ' Set InputMode to read binary data
        Do Until MSComml . InBufferCount < 10  ' Wait until 10 bytes are in the Input
                DoEvents
        Loop
        Buffer = MSComml . Input          ' Store binary data in buffer
        Arr = Buffer                      ' Assign to byte array for processing
End Sub
```

2.7.9 Output

ใช้ในการส่งขบวนของข้อมูลไปยังตัวพักข้อมูลภาคส่ง มีรูปแบบการใช้งานคือ

object . Output [= value]

ค่า value เป็นค่าของตัวอักษรที่เขียนไปยังตัวพักข้อมูลภาคส่ง คุณสมบัติ Output สามารถใช้ในการส่งข้อมูลตัวอักษรหรือข้อมูลไบนารี่ก็ได้ โดยการส่งข้อมูลเป็นรูปแบบตัวอักษรจะต้องกำหนดข้อมูลเป็นแบบแปรเปลี่ยนได้ และมีข้อมูลภายในเป็นแบบข้อความ สำหรับการส่งข้อมูลไบนารี่จะต้องกำหนดชนิดของข้อมูลเป็นแบบไบต์ (Byte) ตัวอย่างโปรแกรมการส่งค่าที่ป้อนจากคีย์บอร์ดไปยังพอร์ตอนุกรม โดยใช้คุณสมบัติ Output

```
Private Sub Form_KeyPress (KeyAscii As Integer )
        Dim Buffer as Variant
        MSComm1. CommPort = 1          ' Use COM1
        MSComm1. PortOpen = True       ' Open port
        Buffer = Chr$ (KeyAscii )
        MSComm1 . Output = Buffer      ' Send DATA
End Sub
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.7.10 OutBufferCont

คืนค่าจำนวนของข้อมูลตัวอักษรที่เก็บอยู่ในตัวพักข้อมูลภาคส่ง และสามารถใช้คำสั่งนี้เพื่อเคลียร์ตัวพักข้อมูลภาคส่งได้ด้วย มีรูปแบบการใช้งานคำสั่งคือ

object. OutBufferCount [= value]

ผู้ใช้งานสามารถเคลียร์ตัวพักข้อมูลภาคส่งได้โดยการกำหนดค่า OutBufferCount เท่ากับ "0"

2.7.11 OutBufferSize

กำหนดค่าและคืนค่าขนาดของตัวพักข้อมูลภาคส่ง ชนิดตัวแปรเป็นแบบไบต์ รูปแบบการใช้งานคำสั่งคือ

object. OutBufferSize [= object]

คุณสมบัติ OutBufferSize ใช้สำหรับการกำหนดขนาดของตัวพักข้อมูลภาคส่ง โดยค่าปกติที่ใช้งานจะมีค่าเท่ากับ 512 ไบต์ การกำหนดค่าตัวพักข้อมูลภาคส่งที่มากเกินไปจะทำให้ มีหน่วยความจะเหลือให้ใช้งานน้อย แต่อย่างไรก็ตามถ้ากำหนดค่าน้อยเกินไป จะทำให้เกิดข้อมูลล้นตัวพักข้อมูลขึ้นมาได้ ยกเว้นจะมีการใช้ แฮนด์เช็ก วิธีการที่ถูกต้องในการกำหนดค่าคือ ทดลองใช้ค่าเริ่มต้นคือค่า 512 ไบต์ดูก่อน ถ้าโปรแกรมทำงานแล้วเกิดการล้นของข้อมูลค่อยเพิ่มค่าของ OutBufferSize ให้มากขึ้น

2.7.12 ParityReplace

ใช้กำหนดและคืนค่าตัวอักษรที่ไปวางแทนในตำแหน่งที่เกิดข้อผิดพลาดจากพาริตี้มีรูปแบบการใช้งานคำสั่งคือ

object. ParityReplace [= value]

บิตพาริตี้ เป็นบิตที่ทางภาคส่งข้อมูลทำการส่งมาพร้อมกับข้อมูล เพื่อตรวจสอบข้อผิดพลาดของข้อมูล โดยเมื่อมีการใช้บิตพาริตี้ คอนโทรล เอ็มเอสคอมม์ จะทำการบวกบิตทุกบิตที่มีค่าลอจิก "1" ในแต่ละไบต์ และทำการตรวจสอบผลลัพธ์ว่าบิตที่อ่านได้นั้นมีจำนวนลอจิก "1 " เป็นเลขคู่หรือคี่ และตรงกับค่าที่กำหนดไว้แต่ต้นหรือไม่ ถ้าค่าที่นำมาบวกแล้วมีพาริตี้ไม่ตรงแสดงว่ารับส่งข้อมูลผิดพลาด การกำหนดค่า เริ่มต้นให้กับ ParityReplace นั้นกำหนดให้ใช้เครื่องหมาย (?) ไปวางไว้ที่ตำแหน่งที่เกิดพาริตี้ผิดพลาด ถ้ากำหนดค่า ParityReplace ให้เป็นค่าว่าง ("") จะเป็นการยกเลิกการใช้งาน ParityReplace และไม่มีการป้อนข้อมูลแทนเมื่อตรวจพบข้อผิดพลาด ParityReplace ใช้ชนิดข้อมูลเป็นแบบข้อความ แต่การกำหนด จะกำหนดได้เพียงไบต์เดียวเท่านั้น ซึ่งจะสามารถใช้ค่าใด ๆ ก็ได้ที่เป็นโค้ด แอนซี่ มีค่าอยู่ระหว่าง 0 – 255

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.7.13 DTREnable

ใช้ในการกำหนดสถานะลอจิกของขา DTR โดยสัญญาณของขา DTR จะส่งจากคอมพิวเตอร์ไปยังโมเด็มเพื่อแสดงว่าคอมพิวเตอร์พร้อมที่จะรับข้อมูลแล้ว ชนิดของข้อมูลเป็นแบบบูลีนมีรูปแบบการใช้งานคือ

object. DTREnable [= value]

ค่า value เป็นค่าสถานะ จริง หรือ เท็จ เพื่อกำหนดลอจิกของขา DTR ให้เป็น "0" หรือ "1" โดย

จริง หมายถึง ให้ขา DTR มีลอจิก " 1 "

เท็จ หมายถึง ให้ขา DTR มีลอจิก " 0 " (มีค่าปกติ)

เมื่อขา DTR ถูกกำหนดสถานะให้เป็น จริง ที่ขา DTR จะมีสถานะลอจิก " 1 " เมื่อทำการเปิดพอร์ตและจะมีสถานเป็น " 0 " เมื่อมีการปิดพอร์ต เมื่อขา DTR ถูกกำหนดสถานะ เป็นเท็จ ที่ขา DTR จะมีสถานะลอจิก " 0 " ตลอดเวลาไม่ว่าจะใช้คำสั่งเปิดพอร์ตหรือปิดพอร์ต สำหรับการใช้งานกับโมเด็ม การทำให้ขา DTR เป็นลอจิก " 0 " จะเป็นการวางหูโทรศัพท์หรือยกเลิกการติดต่อ

2.7.14 RTSEnable

ใช้เพื่อกำหนดสถานะลอจิกให้ขา RTS โดยขา RTS จะเป็นสัญญาณที่ส่งจากคอมพิวเตอร์ไปยังโมเด็มเพื่อร้องขอส่งข้อมูลชนิดของข้อมูลเป็นแบบบูลีนมีรูปแบบการใช้งานคือ

object. RTSEnable [= value]

ค่า Value เป็นสถานะ จริง หรือเท็จ เพื่อกำหนดลอจิก " 0 " หรือ " 1 " ให้ขา RTS โดย

จริง หมายถึง ให้ขา RTS มีลอจิก "1 "

เท็จ หมายถึง ให้ขา RTS มีลอจิก "0 " (เป็นค่าปกติ)

เมื่อขา RTSEnable ถูกกำหนดให้เป็น จริง ขา RTS จะมีสถานะลอจิก " 1 " เมื่อเปิดพอร์ตและมีสถานะลอจิก " 0 " เมื่อปิดพอร์ต

2.7.15 EOFEnable

เป็นการกำหนดให้ เอ็มเอสคอมม์ รอสัญลักษณ์แสดงส่วนท้ายสุดของไฟล์ (End of file : EOF) ระหว่างการรับอินพุตเข้ามา ถ้าพบสัญลักษณ์ EOF ภาคอินพุตจะหยุดรับข้อมูล และเหตุการณ์ OnComm จะถูกกระตุ้นให้ทำงาน คุณสมบัติ คอมม์อีเวนท์ จะมีค่าเท่ากับ 7 หรือ ComEvEOF มีรูปแบบการใช้งาน

object. EOFEnable [= value]

โดย value เป็นค่าสถานะ จริง หรือ เท็จ เพื่ออนุญาติหรือไม่อนุญาตการทำงานของเหตุการณ์ OnComm เมื่อ ตรวจพบสัญลักษณ์ EOF โดย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จริง หมายถึง เหตุการณ์ OnComm จะถูกกระตุ้นให้ทำงานด้วย EOF

เท็จ หมายถึง เหตุการณ์ OnComm จะไม่ถูกกระตุ้นให้ทำงานด้วย EOF (เป็นค่าปกติ) เมื่อ EOFnable กำหนดให้เป็น เท็จ ส่วนควบคุมจะไม่มีการตรวจสอบสัญลักษณ์ EOF

2.7.16 CTSHolding

ผู้ใช้งานสามารถตรวจสอบการทำงานของเขา CTS ได้ว่ามีสถานะลอจิก " 0 " หรือ " 1 " โดยค่าที่อ่านได้จะเป็นบูลีน จริง และ เท็จ ถ้าค่า CTSHolding เป็น จริง ขา CTS จะมีสถานะลอจิกเป็น " 1 " ถ้าค่า CTSHolding เป็นเท็จ ขา CTS จะมีสถานะลอจิกเป็น " 0 " มีรูปแบบการใช้งานคือ

object. CTSHolding

เมื่อขา CTS เป็นลอจิก " 0 " (CTSHolding = Fals) และเกิดไทม์เอาต์ คอนโทรล เอ็มเอสคอมม์ จะกำหนดให้คุณสมบัติ คอมม์อีเวนท์ มีค่าเป็น comEventCTSTO (Clear To Send Timeout) และกรตุ้นให้เกิดเหตุการณ์ OnComm

2.7.17 CDHolding

ผู้ใช้งานสามรถตรวจสอบการทำงานของขา (DCD) ได้ว่ามีสถานะลอจิกเป็น " 1 " หรือ " 0 " โดยค่าที่อ่านได้จะเป็นบูลีน จริง และ เท็จ ถ้าค่า CDHolding เป็น จริง ขา DCD จะมีสถานะลอจิก " 1 " ถ้าค่า CDHolding เป็น เท็จ ขา DCD จะมีสถานะลอจิก " 0 " มีรูปแบบการใช้งานคือ

object. CDHolding

เมื่อขา DCD มีลอจิก " 1 " (CDHolding = True) และเกิดไทม์เอาต์ คอนโทรล เอ็มเอสคอมม์ จะกำหนดให้คุณสมบัติ คอมม์อีเวนท์ มีค่าเป็น comEventCDTO (Carrier Detect Timeout Error) และกระตุ้นให้เกิดเหตุการณ์ OnComm

2.7.18 DSRHolding

ผู้ใช้งานสามารถตรวจสอบการทำงานของขา DSR ได้ว่ามีสถานะลอจิก " 1 " หรือ "0 " โดยค่าที่อ่านได้จะเป็นบูลีน จริง และเท็จ ถ้าค่า DSRHolding เป็น จริง ขา DSR จะมีสถานะลอจิก " 1" ถ้าค่า DSRHoldin เป็น เท็จ ขา DSR จะมีสถานะลอจิก " 0 " มีรูปแบบการใช้งานคือ

object. DSRHolding

เมื่อขา DSR เป็นลอจิก " 1 " (DSRHolding = True) และเกิดไทม์เอาต์ คอนโทรล เอ็มเอสคอมม์ จะกำหนดให้คุณสมบัติ คอมมีอีเวนท์ มีค่าเป็น comEventDSRTO (Data Set Ready Timeout) และกระตุ้นให้เกิดเหตุการณ์ OnComm

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.7.19 Handshaking

ใช้กำหนดคุณสมบัติและคืนค่ารูปแบบแฮนด์เช็กทางวงจรไฟฟ้ามีรูปแบบการใช้งานคำสั่งคือ

object . Handshaking [= value]

ค่าตัวแปร Value ที่ใช้กำหนดค่ากำหนดได้ 4 รูปแบบ ด้วยกันคือ

1) comNone ค่าที่กำหนดคือ 0 เป็นการกำหนดให้ไม่มีการแฮนด์เช็ก (เป็นค่าเริ่มต้น)
2) comXOnXOff ค่าที่กำหนดคือ 1 เป็นการกำหนดให้ใช้แฮนด์เช็กแบบ XON/OFF
3) comRTS ค่าที่กำหนดคือ 2 เป็นการกำหนดให้ใช้ขา RTS/CTS (Request To Send/Clear To Send)
4) comRTSXOnXOff ค่าที่กำหนดคือ 3 เป็นการกำหนดให้ใช้ทั้งแบบ Request To Send และ XON/XOFF

คุณสมบัติ Handshaking ใช้เพื่อกำหนดรูปแบบการสื่อสารภายใน ระหว่างที่ข้อมูลถูกส่งไปยังตัวพักข้อมูลภาครับ เมื่อข้อมูลตัวอักษรถูกส่งมาถึงพอร์ตอนุกรม อุปกรณ์สื่อสารข้อมูลจะทำการย้ายข้อมูลไปยังตัวพักข้อมูลภาครับ เพื่อที่จะให้โปรแกรมสามารถอ่านค่าไปใช้งานได้ ถ้าไม่มีตัวพักข้อมูลภาครับโปรแกรมที่ใช้งานจะต้องทำการอ่านค่าข้อมูลโดยตรงจากฮาร์ดแวร์ของพอร์ตอนุกรม ซึ่งผู้ใช้งานจะเกิดปัญหาข้อมูลสูญหายได้ เนื่องจากว่าการเปลี่ยนแปลงของข้อมูลที่ส่งเข้ามามีการเปลี่ยนแปลงอย่างรวดเร็ว คุณสมบัติ handshaking ช่วยให้ผู้ใช้งานแน่ใจได้ว่าข้อมูลที่ได้รับมานั้นไม่มีการสูญหายเมื่อตัวพักข้อมูลภาครับที่รับข้อมูลนั้นเกิดข้อมูลล้นโดยใช้วิธีการตรวจสอบความพร้อมของตัวพักข้อมูลภาครับที่รับข้อมูลหรือไม่ก่อนที่จะส่งข้อมูลมาให้

2.7.20 Break

ใช้ในการเซตและเคลียร์ค่าสัญญาณ Break ชนิดของข้อมูลเป็นแบบ บูลีนมีรูปแบบการใช้งานคือ

object. Break [= value]

โดย value เป็นค่าบูลีน ถ้า value = จริง หมายถึง การส่งสัญญาณ Break ออกไป ถ้า value = เท็จ หมายถึงการเคลียร์สัญญาณ Break เมื่อกำหนดให้สัญญาณ Break เป็น จริง จะเป็นการหยุดการส่งข้อมูลชั่วคราวจนกว่าจะมีการสั่งให้สัญญาณ Break เป็น เท็จ ตัวอย่าง เป็นวิธีการส่งสัญญาณ Break ออกไปเป็นช่วงเวลาสั้น ๆ ที่ 1/10 ของวินาที

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
MSComml. Break = True                    ' Set the Break condition.
Duration ! = Timer + .1                  ' Set duration to 1/10 second.
Do Until Timer > Duration !              ' wait for the duration to pass.
        Dummy = DoEvents ( )
Loop
MSComml. Break = False                   ' Clear the Break condition.
```

2.8 เหตุการณ์ OnComm

เหตุการณ์ OnComm จะถูกสร้างขึ้นเมื่อค่าของคุณสมบัติ คอมม์อีเวนท์ มีการเปลี่ยนแปลงเพื่อแสดงผลการเปลี่ยนแปลงเหล่านั้นแบบทันทีทันใดหรือแสดงข้อผิดพลาดที่เกิดขึ้น ตัวอย่าง โปรแกรมย่อม OnComm เพื่อนำเหตุการณ์ คอมมีอีเวนท์ มาแสดง

```
Private Sub MSComm OnComm ( )
        Select Case MSComml. CommEvent
        ' Handle each event or error by placing
        ' code below each case statement
        ' Errors
                Case comEventBreak          ' A Brreak was received.
                Case comEventCDTO           ' CD (RLSD) Timeout.
                Case comEventCTSTO          ' CTS Timeout.
                Case comEventDSRTO          ' DSR Timeout.
                Case comEventFrame          ' Framin Error
                Case comEventOverrun        ' Data Lost.
                Case comEventRxOver         ' Receive buffer overflow.
                Case comEventRxParity       ' Parity Error.
 Case comEventTxFull                  ' Transmit buffer full.
        ' Events
                Case comEvCD        ' Change in the CD line.
                Case comEvCTS       ' Change in the CTS line.
                Case comEvDSR       ' Change in the DSR line.
                Case comEvRing      ' Change in the Ring indicator.
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ค่าคงที่	ค่า	รายละเอียด
ComEventBreak	1001	ได้รับสัญญาณ Break
ComEventCTSTO	1002	ขา CTS เกิดไทม์เอาต์
ComEventDSRTO	1003	ขา DSR เกิดไทม์เอาต์
ComEventFrame	1004	เกิดข้อผิดพลาดที่เฟรมข้อมูล (Framing error)
ComEventOverrun	1006	พอร์ตอนุกรมเกิดโอเวอร์รัน (Port overrun)
ComEventCDTO	1007	ขา DCD เกิดไทม์เอาต์
ComEventRxOver	1008	บัฟเฟอร์รับข้อมูลเกิดโอเวอร์โฟลว์
ComEventRxParity	1009	เกิดข้อผิดพลาดที่พาริตี้ (Parity error)
ComEventTxFull	1010	บัฟเฟอร์ส่งข้อมูลเต็ม

ตารางที่ 6 ค่าคงที่สำหรับคุณสมบัติ Error

ค่าคงที่	ค่า	รายละเอียด
CominputmodeText	0	ข้อมูลที่รับมีคุณสมบัติเป็นข้อความ(ค่าปกติ)
CominputmodeBinary	1	ข้อมูลที่รับเข้ามาเป็นข้อมูลไบนารี่

ตารางที่ 7 ค่าคงที่สำหรับคุณสมบัติ InputMode

2.9 การใช้ เอ็มเอสคอมม์ เพื่อการติดต่ออุปกรณ์ภายนอก

จากรายละเอียดของ เอ็มเอสคอมม์ ที่กล่าวไปในตอนต้นนั้น จะเห็นได้ว่าวิธีการที่จะอ่านค่าหรือเขียนค่าไปยังขาสถานะและขาควบคุมของพอร์ตอนุกรมสามารถทำได้ง่ายมาก โดยใช้คำสั่งเหล่านี้

DTREnable สำหรับสั่งให้ขา DTR มีลอจิก"0" หรือ "1"

RTSEnable สำหรับสั่งให้ขา RTS มีลอจิก "0" หรือ "1"

CTSHolding สำหรับอ่านค่าสถานะจากขา CTS ว่ามีลอจิก "0" หรือ "1"

CDHolding สำหรับอ่านค่าสถานะจากขา DCD ว่ามีลอจิก "0" หรือ "1"

DSRHolding สำหรับอ่านค่าสถานะจากขา DSR ว่ามีลอจิก "0" หรือ "1"

Break สำหรับการสั่งให้ขา TxD มีลอจิก "0" หรือ "1"

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        Case comEvReceive    ' Received RThreshold # fo chars.
        Case comEvSend       ' Sthreshold number in the ' transmit buffer
        Case comEvEof        ' An EOF character was found in the input stream
    End Select
End Sub
```

2.8..1 ค่าคงที่คุณสมบัติของคอนโทรล เอ็มเอสคอมม์

ค่าคงที่	ค่า	รายละเอียด
Comnone	0	ไม่ใช้การตรวจสอบแฮนด์เชก
ComXonXoff	1	ใช้การตรวจสอบแฮนด์เชกแบบ XonXoff
ComRTS	2	ใช้การตรวจสอบแฮนด์เชกผ่านทางขา RTS และ CTS
ComRTSXonXoff	3	กำหนดการตรวจสอบแฮนด์เชกทั้งแบบ RTS,CTS และ Xon/Xoff

ตารางที่ 4 ค่าคงที่สำหรับคุณสมบัติ Handshake

ค่าคงที่	ค่า	รายละเอียด
ComEvSend	1	ส่งค่าเหตุการณ์ (send event)
ComEvReceive	2	รับค่าเหตุการณ์ (receive event)
ComEvCTS	3	มีการเปลี่ยนแปลงที่ขา CTS
ComEvDSR	4	มีการเปลี่ยนแปลงที่ขา DSR
ComEvCD	5	มีการเปลี่ยนแปลงที่ขา DCD
ComEvRing	6	ตรวจจับสัญญาณกระดิ่งของโทรศัพท์
ComEvEOF	7	ตรวจพบตำแหน่งท้ายสุดของไฟล์(End of file)

ตารางที่ 5 ค่าคงที่สำหรับคุณสมบัติ OnComm

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.10 ระบบส่งข้อมูลแบบแบบไอแสควซี (I^2C)

I^2C ย่อมาจาก Inter – IC Communication หมายถึง การติดต่อสื่อสารระหว่าง วงจรรวม ด้วยระบบเส้นทางแบบไอแสควซี พัฒนาขึ้นโดยบริษัท ฟิลลิปส์ (Philips) ด้วยจุดมุ่งหมายคือ ต้องการให้ไอซีหรือโมดูล สามารถ ติดต่อ สั่งงาน และควบคุมภายใต้สายสัญญาณเพียง สองเส้น เส้นหนึ่งคือสายข้อมูลอีกเส้นหนึ่งคือสายสัญญาณนาฬิกา การต่ออุปกรณ์ แต่ละตัวสามารถต่อขนานกันไปได้ ส่วนการกำหนดตำแหน่งสำหรับการติดต่ออุปกรณ์แต่ละตัวจะใช้รหัสข้อมูลและการกำหนดสภาวะลอจิกที่ขากำ หนดตำแหน่งของอุปกรณ์แต่ละตัว

สายข้อมูลบนบัส ไอแสควซี มีชื่อเรียกอย่างเป็นทางการว่า สายข้อมูลอนุกรม หรือ SDA (Serial Data line) ส่วนสายสัญญาณนาฬิกามีชื่อเรียกอย่างเป็นทางการว่า สายสัญญาณนาฬิกาอนุกรม หรือ SCL (Serial Clock line)

2.10.1 คุณสมบัติทั่วไปของระบบเส้นทางแบบไอแสควซี

สายข้อมูลอนุกรมและ เอสซีแอล เป็นสายสัญญาณ 2 ทิศทาง ต้องมีการต่อตัวต้านทานพูลอัพแรงดัน + 5 โวลต์ไว้ตลอดเวลา เพื่อให้สายมีสถานะลอจิกสูงเมื่อไม่มีการติดต่อใช้งาน และป้องกันสัญญาณรบกวนที่อาจมีเข้ามาในสายสัญญาณทั้งสองวงจรเอาต์พุตของอุปกรณ์ที่ต่ออยู่บนบัสไอแสควซี ต้องเป็นวงจรเดรนเปิด หรือคอลเล็กเตอร์เปิด ดังรูปที่ 20 อัตราการถ่ายทอดข้อมูลบนบัส สูงถึง 100 กิโลบิตต่อวินาที ในรูปแบบปกติ (Standard Mode)และสูงสุดถึง 400 กิโลบิตต่อวินาทีในรูปแบบความเร็วสูง (Fast Mode) อุปกรณ์ที่ต่อร่วมอยู่บนบัส ต้องมีค่าความจุไฟฟ้ารวมที่เกิดขึ้นระหว่างสายข้อมูลอนุกรมและเอสซีแอลไม่เกิน 400 pF การเข้าถึงอุปกรณ์บนบัสใช้ข้อมูลการเข้าถึง 2 ค่าคือ 7 บิต หรือ 10 บิต

ข้อเด่นของบัสไอแสควซีคือ สามารถเชื่อมอุปกรณ์ที่ใช้ไฟเลี้ยงไม่เท่ากัน ให้สามารถติดต่อสื่อสารด้วยกันได้ โดยให้ต่อสายข้อมูลอนุกรมและ สัญญาณนาฬิกา ของอุปกรณ์แต่ละตัวเข้าด้วยกัน และต่อตัวต้านทานพูลอัพเข้ากับแรงดัน +5 โวลต์ไว้เสมอ ดังรูปที่ 21 ในกรณีที่อาจมีแรงดันไฟกระชากขนาดใหญ่ปะปนเข้ามาในบัสไอแสควซี ที่ขาข้อมูลอนุกรม และสัญญาณนาฬิกา ของอุปกรณ์แต่ละตัวต้องต่อตัวต้านทานอนุกรมกับขาก่อนต่อเข้าสู่บัสไอแสควซี ดังรูปที่ 22

2.10.2 หลักการของระบบไอแสควซี

การกำหนดรูปแบบของการติดต่อบนบัสหรือเรียกว่า โปรโตคอล ของระบบบัสแบบไอแสควซี ซึ่งมีข้อตกลงพื้นฐานดังนี้

1) อุปกรณ์ที่เป็นผู้สร้างหรือส่งข้อมูลเรียกว่า ตัวส่ง
2) อุปกรณ์ที่เป็นผูรับเรียกว่าตัวรับ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3) อุปกรณ์บนบัสไอแสควซี สามารถเป็นได้ทั้งตัวรับและตัวส่ง หรืออย่างใดอย่างหนึ่งก็ได้

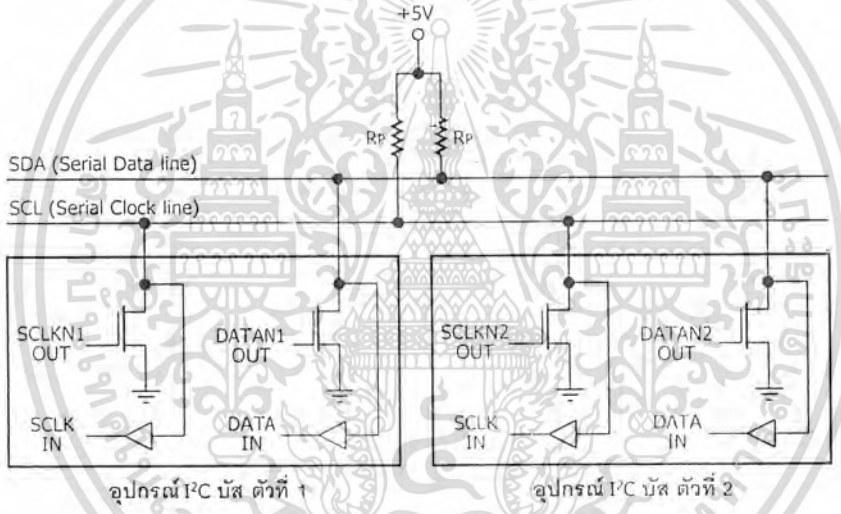
4) อุปกรณ์ที่ทำหน้าที่ควบคุมจังหวะการทำงานบนบัสเรียกว่ามาสเตอร์ (Master)

5) อุปกรณ์ที่ถูกควบคุมเรียกว่า สเลฟ (Slave)

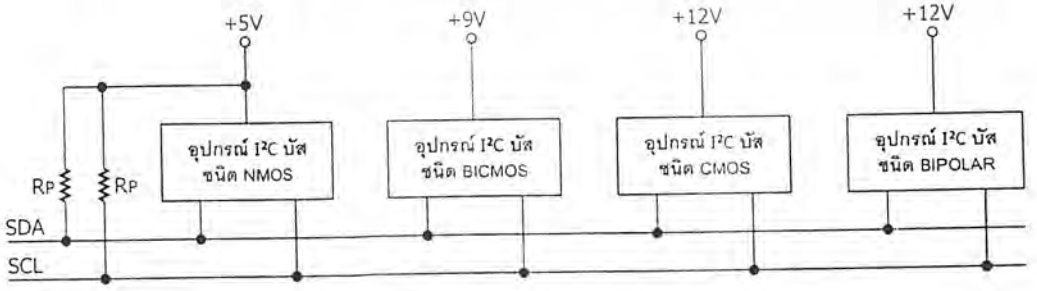
ข้อกำหนด 2 ประการสำคัญของการติต่อบนบัส ไอแสควซี คือ

1) การถ่ายทอดข้อมูลจะเกิดขึ้นได้เมื่อบัสว่างเท่านั้น

2) ในระหว่างการถ่ายทอดข้อมูล เมื่อใดก็ตามที่ สายสัญญาณนาฬิกา มีสถานะเป็นลอจิกสูง สายข้อมูลต้องรักษาข้อมูลไว้ ห้ามเปลี่ยนแปลงเด็ดขาด มิฉะนั้นสัญญารที่เกิดขึ้นจะได้รับการแปลเป็นสัญญาณควบคุมแทน

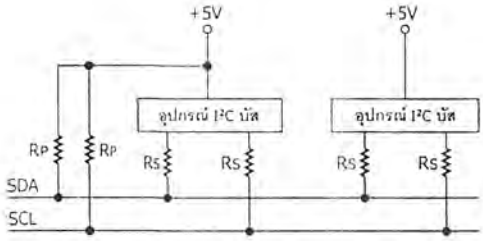


รูปที่ 20 โครงสร้างของวงจรเอาต์พุตของอุปกรณ์ที่ใช้การเชื่อมต่อบนระบบบัสไอแสควซี



รูปที่ 21 แสดงการเชื่อมต่ออุปกรณ์บนระบบบัสไอแสควซี ที่ใช้ไฟเลี้ยงไม่เท่ากัน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 22 การต่อตัวต้านทานเพื่อป้องกันแรงดันกระชากที่อาจปะปนเข้ามาในสายสัญญาณ ข้อมูลอนุกรมและสายสัญญาณนาฬิกา ของระบบบัสแบบไอแสควซี

2.10.3 สภาวะที่เกิดขึ้นบนบัสไอแสควซี

มี 5 สภาวะ คือ

1) **บัสว่าง** เกิดขึ้นเมื่อสถานะลอจิกบนสายข้อมูลอนุกรม และสัญญาณนาฬิกาเป็นลอจิกสูงทั้งคู่ซึ่งสามารถเริ่มต้นส่งข้อมูลได้

2) **เริ่มต้นการถ่ายทอดข้อมูล** เกิดขึ้นเมื่อสายข้อมูลอนุกรมมีการเปลี่ยนแปลงสถานะลอจิกจาก สูงไป ต่ำ ในขณะที่สายสัญญาณนาฬิกา มีสถานะลอจิกสูงเรียกสภาวะนี้ว่า สภาวะเริ่มต้น (START)

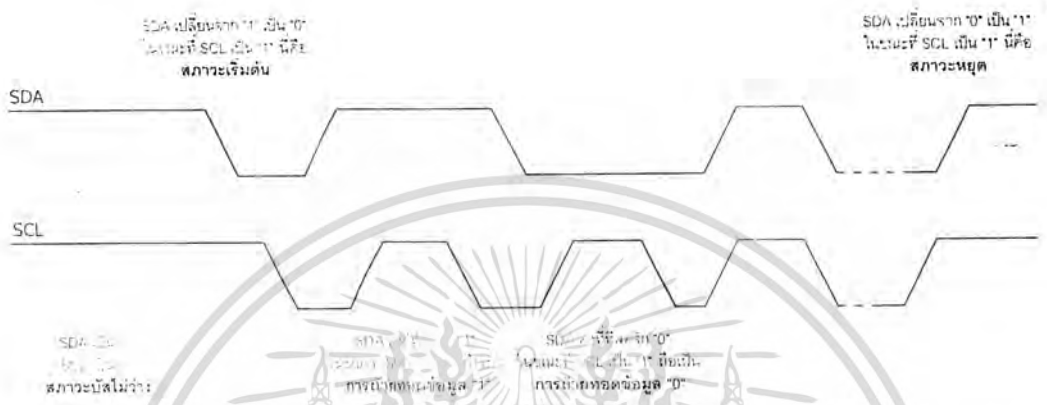
3) **หยุดการถ่ายทอดข้อมูล** เกิดขึ้นเมื่อสายข้อมูลอนุกรมมีการเปลี่ยนแปลงระดับลอจิกจากต่ำไปสูงในขณะที่สายสัญญาณนาฬิกา มีสถานะลอจิกสูงเรียกว่า สภาวะหยุด

4) **ข้อมูลดำรงอยู่บนบัส** เกิดขึ้นถัดจากสภาวะเริ่มต้น โดยสถานะลอจิกที่เกิดขึ้นบนสายข้อมูลอนุกรม ก็คือข้อมูลที่ทำการถ่ายทอด เมื่อ สัญญาณนาฬิกา เป็นลอจิกสูง สถานะที่สายข้อมูลอนุกรม ต้องคงที่เพื่อให้อุปกรณ์รับรู้ว่าข้อมูลจังหวะนั้นเป็น 0 หรือ 1 ข้อมูลอาจเกิดการเปลี่ยนแปลงได้ในขณะที่สายสัญญาณนาฬิกา เป็นลอจิกต่ำ แต่เมื่อใดก็ตามที่ต้องการถ่ายทอดข้อมูลอย่างสมบูรณ์ สถานะที่ขา ข้อมูลอนุกรมต้องคงที่ ตลอดช่วงเวลาที่สายสัญญาณนาฬิกามีสถานะลอจิก สูง หากเกิดการเปลี่ยนแปลงสถานะลอจิกในขณะที่สายสัญญาณนาฬิกา มีจิกสูงอยู่นั้น อุปกรณ์มาสเตอร์ที่ควบคุมการถ่ายทอดข้อมูลจะแปลความหมายเป็นสภาวะหยุดหรือสภาวะเริ่มต้นก็ได้ ทำให้ข้อมูลที่ทำการถ่ายทอดนั้นเกิดความผิดพลาดขึ้นได้

5) **รับรู้ข้อมูล** เกิดขึ้นหลังจากที่การถ่ายทอดข้อมูลจากตัวส่งมายังตัวรับเกิดขึ้นอย่างสมบูรณ์ โดยตัวส่งจะทำการส่งข้อมูลมาหนึ่งบิตเรียกว่า บิตรับรู้ มีสถานะเป็นลอจิกสูงหลังจากส่งข้อมูลมาครบถ้วน ส่วนอุปกรณ์มาสเตอร์จะทำการส่งสัญญาณรับรู้พิเศษซึ่งสัมพันธ์กับ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สัญญาณนาฬิกา อุปกรณ์สเลฟที่ถูกอ้างถึงในการติดต่อหรือกำลังติดต่ออยู่ในขณะนั้นก็จะกำเนิดบิตรับรู้ที่สถานะลอจิกต่ำเพื่อตอบสนองให้ทราบว่าได้รับข้อมูลเรียบร้อยแล้ว สภาวะทั้งหมดของระบบบัสแบบไอแสควซี แสดงดังรูปที่ 23



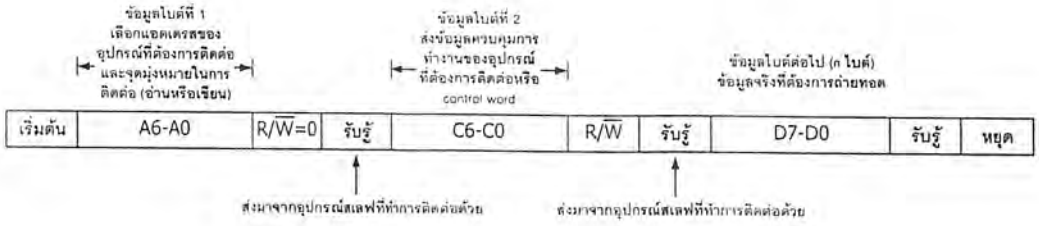
รูปที่ 23 แสดงสถานะต่างๆ บนระบบบัสไอแสควซี

2.10.4 การทำงานบนบัส ไอแสควซี

ก่อนที่จะเริ่มต้นการถ่ายทอดข้อมูลระหว่างอุปกรณ์ต่างๆที่ต่ออยู่บนบัสต้องมีการอ้างถึงอุปกรณ์เสียก่อน โดยการอ้างถึงมีอยู่ 2 รูปแบบคือ

1) การอ้างถึงแบบ 7 บิต ข้อมูลไบต์แรกที่เกิดขึ้นหลังจากสภาวะเริ่มต้นก็คือข้อมูลที่ใช้ในการอ้างถึงอุปกรณ์ที่ต้องการจะติดต่อโดยมีรูปแบบแสดงในรูปที่ 24 ใน 7 บิตบน รวมทั้งบิตนัยสำคัญสูงด้วยจะเป็นข้อมูลตำแหน่งของอุปกรณ์ที่ต้องการติดต่อ โดยแบ่งเป็น บิตกำหนดตำแหน่งคงที่ จำนวน 4 บิตซึ่งอุปกรณ์แต่ละตัวจะถูกกำหนดมาจากผู้ผลิตแก้ไขไม่ได้ ถัดมาอีก 3 บิต เป็นบิตกำหนดตำแหน่งที่สามารถโปแกรมได้ โดยผู้ใช้เป็นผู้กำหนดสถานะลอจิกที่ A0 – A2 ของอุปกรณ์ที่เชื่อมต่อกับบัสไอแสควซี ส่วนบิตนัยสำคัญต่ำสุด เป็นบิตกำหนดว่าต้องการที่จะเขียนหรืออ่านข้อมูลกับอุปกรณ์ตัวนั้น โดยหากบิตนัยสำคัญต่ำสุดเป็น 0 จะเป็นการเขียนข้อมูลลงอุปกรณ์ตัวนั้น ข้อมูลในไบต์ต่อมาคือรหัสควบคุมอุปกรณ์แต่ละตัวจะมีรหัสควบคุมที่ต่างกันไป และข้อมูลในไบต์ต่อมาคือข้อมูลที่ต้องการถ่ายทอดจริงๆ หลังจากที่มีการทอดข้อมูลในแต่ละไบต์ อุปกรณ์สเลฟที่ได้รับการติดต่อต้องส่งสัญญาณรับรู้ตอบกลับมาด้วยทุกครั้งเพื่อให้กระบวนการถ่ายทอดสัญญาณสามารถดำเนินการต่อไปได้ ในรูปที่ 25 แสดงรูปแบบข้ออนุกรมที่เกิดขึ้นในการติดต่อบนบัสไอแสควซี ของการอ้างถึงแบบ 7บิต

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 24 แสดงรูปแบบของข้อมูลอนุกรมที่เกิดขึ้นในการอ้างถึงแบบ 7 บิต ของระบบบัสแบบไอแสควซี

2) การอ้างถึงแบบ 10 บิต ในการอ้างถึงแบบนี้ ยังคงใช้รูปแบบข้อมูลอนุกรมที่เหมือนกับแบบ 7 บิต แต่จะมีข้อมูลเพิ่มเข้ามาเล็กน้อยโดยในข้อมูลไบต์แรกหลังจากเกิดสภาวะเริ่มต้น ต้องกำหนด 5 บิตบนมีข้อมูลเป็น 11110 ส่วน 2 บิตถัดมาเป็นบิต ตำแหน่งของอุปกรณ์ที่ต้องการติดต่อ ในบิตนัยสำคัญต่ำสุดของข้อมูลไบต์แรกยังคงเป็นการกำหนดว่า ต้องการอ่านหรือเขียนข้อมูลกับอุปกรณ์ตัวนั้นๆ ข้อมูลต่อมาเป็นข้อมูลตำแหน่งในไบต์ที่ 2 ของอุปกรณ์ที่ต้องการติดต่อด้วยข้อมูลในไบต์ถัดไปจึงเป็นข้อมูลควบคุม ข้อมูลหลังจากนั้นก็จะเป็นข้อมูลจริง ส่วนการตอบสนองของบิตรับรู้จะเหมือนกับการอ้างถึงแบบ 7 บิตเช่นกัน รูปแบบของของข้อมูลที่ใช้ในการอ้างถึงแบบ 10 บิตแสดงดังรูปที่ 25

รูปที่ 25 แสดงรูปแบบของของข้อมูลข้อมูลอนุกรมที่เกิดขึ้นในการอ้างถึงแบบ 10 บิตบนบัสแบบไอแสควซี

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.11 ขยายพอร์ตอินพุตเอาต์พุตให้แก่พอร์ตอนุกรมด้วยระบบบัส ไอแสควซี

ด้วยข้อจำกัดของจำนวนขาสัญญาณของพอร์ตอนุกรม หากต้องการให้สามารถตอบสนองระบบหรืองานที่ต้องการพอร์ตอินพุตเอาต์พุตจำนวนมากได้นั้นจำเป็นต้องต่ออุปกรณ์ภายนอกเพิ่มเติม เพื่อขยายจำนวนพอร์ตอินพุตเอาต์พุต เพื่อช่วยให้พอร์ตอนุกรมมีจำนวนพอร์ตใช้งานเพิ่มมากขึ้น โดยใช้สายสัญญาณของพอร์ตอนุกรมเพียง 3 เส้น แต่สามารถขยายพอร์ตอินพุตเอาต์พุตได้มากถึง 128 บิต

อุปกรณ์ที่แนะนำคือไอซีขยายพอร์ตเบอร์ PCF8574 และ PCF8574A โดยไอซีแต่ละเบอร์สามารถขายพอร์ตอินพุตเอาต์พุตได้ 8 ช่องหรือ 8 บิต สามารถต่อพ่วงกันได้มากถึง 8 ตัว ทำให้สามารถต่อพอร์ตอินพุตได้มากถึง 64 ช่องต่อไอซี 1 เบอร์ ถ้าหากนำ PCF8574 และ PCF8574A มาต่อพ่วงกัน จะเพิ่มจำนวนพอร์ตได้อีก 64 บิต รวมเป็น 128 บิต โดยการเชื่อมต่อนั้นต้องกระทำบนบัสไอแสควซีเท่านั้น

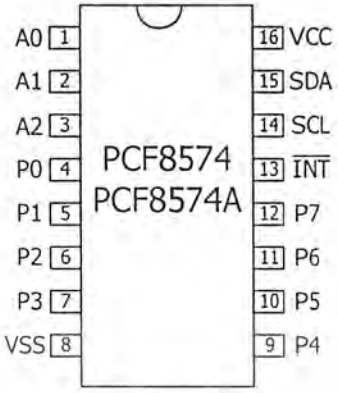
2.11.1 ข้อมูลเบื้องต้นของ PCF857/8574A

ไอซีPCF8574 และ PCF8574A มีคุณสมบัติดังนี้

1) ทำงานที่ระดับแรงดันตั้งแต่ 205V ถึง 6V
2) กินกระแสในสภาวะสแตนด์บายต่ำเพียง 10 UA
3) ใช้การเชื่อมต่อแบบบัส ไอแสควซี
4) มีเอาต์พุตสำหรับขัดจังหวะแบบเดรนเปิด
5) เอาต์พุตสามารถแลตช์หรือคงค่าไว้ได้ และขับ LED ได้โดยต่อวงจรในลักษณะกระแสซิงก์ หรือทำงานด้วยลอจิก " 0 "
6) สามารถกำหนดแอดเดรสของวงจรแต่ละตัวได้ทางฮาร์ดแวร์ ด้วยขา A0 – A2 ทำให้สามารถต่อพ่วงกันได้ถึง 8 ตัวต่อเบอร์
7) สามารถต่อพ่วงทั้ง PCF8574 และ P CF8574A ได้ ทำให้สามารถขยายพอร์ตได้มากถึง 128 บิต

การจัดขาและหน้าที่การทำงานของแต่ละขาของไอซี PCF8574 แสดงในรูปที่ 26 ขาพอร์ตทั้ง 8 ขา ของ PCF8574 สามารถกำหนดให้เป็นอินพุตหรือเอาต์พุตได้โดยอิสระโดยไม่จำเป็นต้องใช้คำสั่งควบคุมเพื่อเลือกให้เป็นขาเอาต์พุตหรือขาอินพุต ลักษณะวงจรภายในของพอร์ตอินพุตแสดงในรูปที่ 28 เมื่อจ่ายไฟให้กับ PCF8574 ครั้งแรก ขาพอร์ตทั้ง 8 ขา จะมีสถานะเป็น "1 " ซึ่งจะเป็นการจ่ายกระแสมาจากแหล่งจ่ายกระแสคงคงที่ภายในตัวไอซี ทำให้มีกระแสในขณะลอจิก "1 " นี้เพียง 100 UA เท่านั้น จึงไม่เหมาะสมที่จะใช้ PCF8574 ในการขับกระแสซอร์ส แต่จะเหมาะกับการนำ PCF8574 ไปขับกระแสซิงก์มากกว่า

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



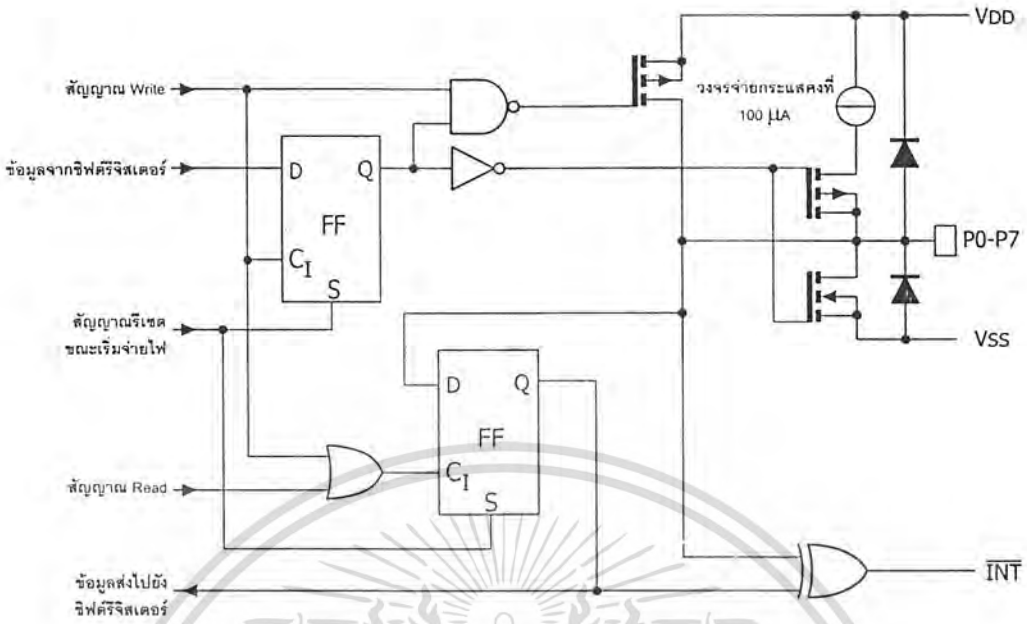
ชื่อ	ตำแหน่งขา	หน้าที่
A0	1	อินพุตแอดเดรสตัวที่ 1
A1	2	อินพุตแอดเดรสตัวที่ 2
A2	3	อินพุตแอดเดรสตัวที่ 3
P0	4	พอร์ตอินพุตเอาต์พุต 2 ทิศทางบิต 0
P1	5	พอร์ตอินพุตเอาต์พุต 2 ทิศทางบิต 1
P2	6	พอร์ตอินพุตเอาต์พุต 2 ทิศทางบิต 2
P3	7	พอร์ตอินพุตเอาต์พุต 2 ทิศทางบิต 3
VSS	8	กราวด์
P4	9	พอร์ตอินพุตเอาต์พุต 2 ทิศทางบิต 4
P5	10	พอร์ตอินพุตเอาต์พุต 2 ทิศทางบิต 5
P6	11	พอร์ตอินพุตเอาต์พุต 2 ทิศทางบิต 6
P7	12	พอร์ตอินพุตเอาต์พุต 2 ทิศทางบิต 7
INT	13	ขาเอาต์พุตอินเตอร์รัปต์(ทำงานที่ลอจิก 0)
SCL	14	ขาสัญญาณนาฬิกาสำหรับ I^2C บัส
SDA	15	ขาข้อมูลสำหรับ I^2C บัส
VDD	16	ไฟเลี้ยง

รูปที่ 26 การจัดขาของ PCF8574/8574A และหน้าที่การทำงานของแต่ละขา

เมื่อต้องการให้ขอพอร์ตเหล่านี้ทำหน้าที่เป็นอินพุตจะต้องส่งสัญญาณให้ขาเหล่านี้มีลอจิก "1 " เสียก่อน เมื่อขาอินพุตได้รับสัญญาณจากภายนอกป้อนเข้ามา PCF8574 จะสร้างสัญญาณขัดจังหวะ(INT) ป้อนให้ไมโครคอนโทรลเลอร์หรือคอมพิวเตอร์รับรู้แทนการต้องคอยตรวจสอบขาอินพุตอยู่ตลอดเวลา สัญญาณอินเตอร์รัปต์นี้จะถูกรีเซตเมื่อมีการอ่านค่าข้อมูลหรือมีการเปลี่ยนค่าของอินพุตไปสู่ค่าเดิม

สำหรับ PCF8574 และ PCF8574A มีการทำงานเหมือนกันทุกประการ แตกต่างกันเพียงค่าตำแหน่งที่ใช้ในการติดต่อเท่านั้น ด้วยเหตุนี้จึงสามารถนำไอซีทั้งสองเบอร์นี้มาต่อพ่วงกันเพื่อขยายพอร์ตได้มากถึงเบอร์ละ 8 ตัวรวมทั้งสิ้น 16 ตัว ส่งผลให้สามารถขยายพอร์ตได้สูงสุด 128 บิต

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 27 วงจรภายในของขาพอร์ตของวงจรรวม PCF8574/8574A

2.12 การควบคุมแรงดันไฟฟ้ากระแสสลับ

คือการนำสวิตช์อิเล็กทรอนิกส์ เช่น ไทริสเตอร์ หรือไทรแอก ไปใช้ในการแปลงผันกำลังไฟฟ้ากระแสสลับ จากแหล่งจ่ายไฟสลับที่มีแรงดันไฟฟ้าคงที่และความถี่คงที่ ให้ได้แรงดันไฟฟ้ากระแสสลับที่ปรับค่าได้แต่มีค่าความถี่คงที่ เป็นวิธีการแปลงผันไฟสลับเป็นไฟสลับแบบหนึ่ง นิยมใช้กันมากในงานอุตสาหกรรมหลากหลาย เช่น การให้ความร้อนแบบเหนี่ยวนำ (Induction heating) การควบคุมความสว่าง (Lighting controls) การเลื่อนแทปหม้อแปลง (Transformer tap changing) กระบวนการใช้ไฟฟ้าในงานเคมี (Primary transformer controls for electrochemical process) การให้ความร้อนในอุตสาหกรรม (Industrial heating) การควบคุมความเร็วของมอเตอร์เหนี่ยวนำ (Speed control of induction motor)

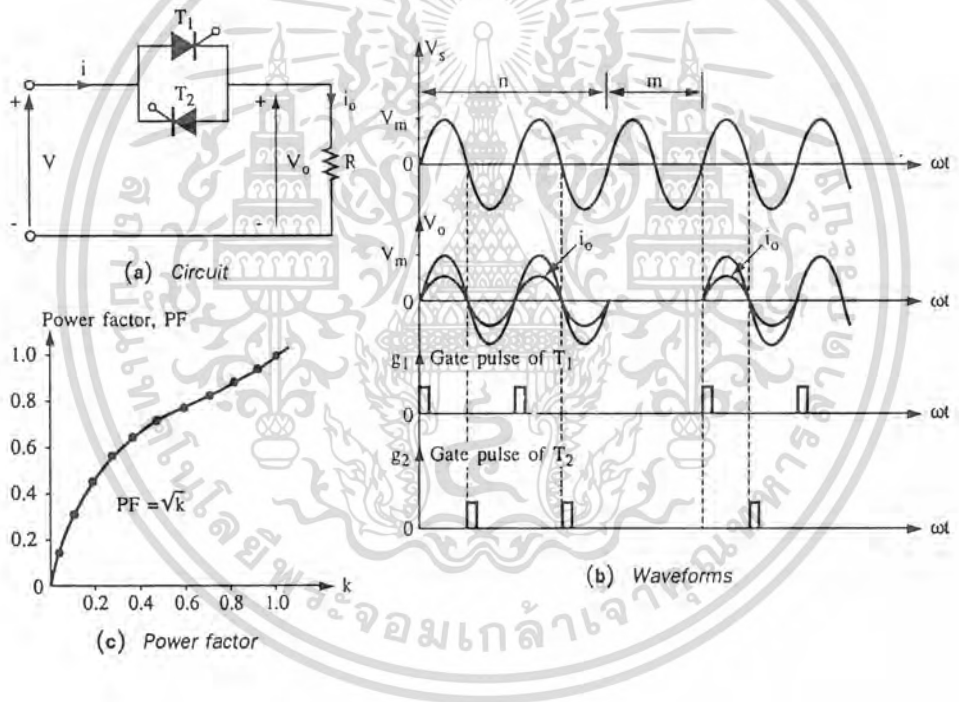
หลักการควบคุมสำหรับวงจรที่ใช้ควบคุมแรงดันไฟฟ้ากระแสสลับที่โหลด ทั้งกรณีใช้กับแหล่งจ่ายไฟสลับ 1 เฟสและ 3 เฟส คือเทคนิคการควบคุมการจุดชนวนของไทริสเตอร์ในวงจรแบ่งออกได้ 2 วิธีคือการควบคุมแบบปิด-เปิด (On-Off control) การควบคุมมุมเฟส (Phase-angle control)

2.12.1 การควบคุมแบบปิด-เปิด

หลักการควบคุมแบบปิด-เปิด อธิบายได้จากวงจรควบคุมไฟสลับแบบ 1 เฟส ดังรูป 28 (a) โดยวงจรจะใช้ไทริสเตอร์ 2 ตัวต่อกลับหลัง โดยให้ไทริสเตอร์ T_1 ทำงานในครึ่งวัฏ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จักรบวกของแหล่งจ่ายไฟฟ้าสลับ (V) และไทริสเตอร์ T_2 ทำงานในช่วงครึ่งวัฏจักรลบของแหล่งจ่ายไฟฟ้าสลับ (V) โดยไทริสเตอร์ทั้ง 2 ตัวจะได้รับสัญญาณจุดชนวนเกต g_1 และ g_2 ดังแสดงในรูป 30 (b) โดยสัญญาณ g_1 จะจุดชนวนเกตของ T_1 ที่แรงดันโหลดมีค่าเป็นศูนย์ ซึ่งเป็นเวลาเดียวกันกับ $\omega t = 0$ ของแต่ละวัฏจักรและสัญญาณจุดชนวนเกต g_2 จะจุดชนวนเกตของ T_2 ที่เวลา $\omega t = 0 + \pi$ แต่จะมีการกำหนดจำนวนสัญญาณจุดชนวน ว่าต้องการให้ไทริสเตอร์ทั้งสองทำงานจำนวน n วัฏจักรและให้หยุดทำงานจำนวน m วัฏจักร จากรูป 30 (b) กำหนดให้ไทริสเตอร์ทั้ง 2 ตัวทำงาน จำนวน 2 วัฏจักร และไม่ทำงานจำนวน 1 วัฏจักร (n = 2 วัฏจักร, m = 1 วัฏจักร)



รูปที่ 28 วงจรควบคุมแรงดันไฟสลับโดยวิธีควบคุมแบบ ปิด-เปิด

จากวงจรในรูป 28 (a) กำหนดให้

v = $V_m \sin \omega t$ (แรงดันชั่วขณะของแหล่งจ่ายไฟสลับ)

n = จำนวนวัฏจักรที่ไทริสเตอร์ทำงาน (ON-cycle)

m = จำนวนวัฏจักรที่ไทริสเตอร์ไม่ทำงาน (OFF-cycle)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

คำนวณค่าแรงดันไฟฟ้าอาร์.เอ็ม.เอส. ที่โหลดได้รับ ดังสมการต่อไปนี้

$$V_{o(RMS)} = \sqrt{\left[\frac{n}{2\pi(n+1)}\int_0^{2\pi} V_m^2 \sin\omega t d\omega t\right]}$$

$$= \frac{V_m}{\sqrt{2}} \times \sqrt{\frac{n}{n+m}}$$

$$\therefore V_{o(rms)} = V\sqrt{k} \qquad (1)$$

เมื่อ V = แรงดันไฟฟ้า อาร์เอ็มเอส ของแหล่งจ่ายไฟสลับ

$K = \frac{n}{n+m}$ คือค่าวัฏจักรหน้าที่ (Duty cycle)

ค่าตัวประกอบกำลังของแหล่งจ่ายไฟฟ้าคือ $\sqrt{k}$

ดังนั้น $$PF = \frac{Po}{Va} = \sqrt{k} \qquad (2)$$

เมื่อ $P_O = V_{o(RMS)} \times I_{o(RMS)}$

VA= กำลังไฟฟ้าด้านขาเข้า

กระแสไฟฟ้าใช้งานที่โหลด $I_{o(RMS)}$ คือ

$$I_{o(RMS)} = V_{o(RMS)} / R \qquad (3)$$

กระแสไฟฟ้าที่ไหลผ่านไทริสเตอร์แต่ละตัว (I_A) คือ

$$I_A = \frac{n}{2\pi(n+m)}\int_0^{\pi} (I_m) Sin\omega t d\omega t$$

$$= \frac{(Im)(n)}{\pi(n+m)}$$

$$= \frac{Im}{\pi}k \qquad (4)$$

กระแสไฟฟ้าใช้งานที่ผ่านไทริสเตอร์แต่ละตัว (I_R) คือ

$$I_R = \sqrt{\left(\frac{n}{2\pi(n+m)}\right)\int_0^{\pi} I^2{}_m Sin^2\omega t d\omega t}$$

$$= \frac{I_m}{2}\sqrt{k} \qquad (5)$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

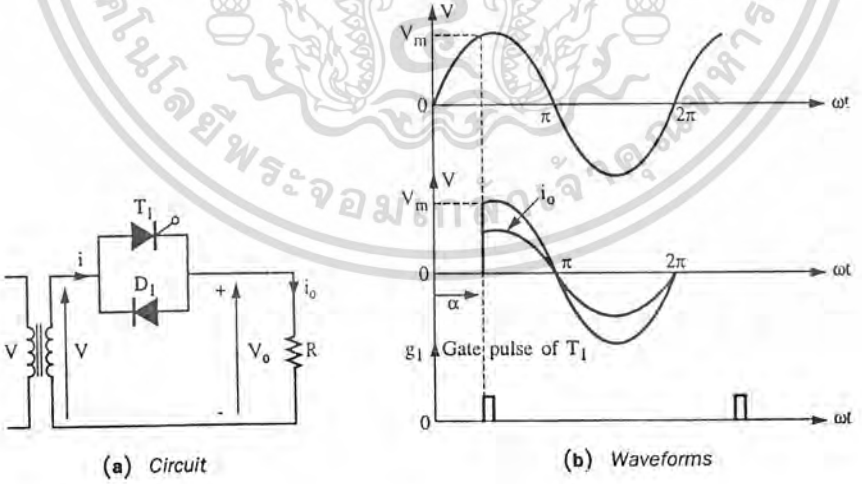
การควบคุมแบบ ปิด-เปิดนี้ ค่าแรงดันไฟฟ้าใช้งานด้านออกของวงจร และค่าตัวประกอบกำลังจะแปรไปตามค่า วัฏจักรหน้าที่ (k) ค่าตัวประกอบกำลังจะมีค่าเมื่อค่า $\sqrt{k}$ มค่าต่ำๆ ดังรูปที่ 28(a)

2.12.2 การควบคุมมุมเฟส

วงจรการใช้ไทริสเตอร์ควบคุมเฟส เพื่อควบคุมขนาดของแรงดันไฟฟ้าใช้งานด้านออกของวงจร อาจใช้ไทริสเตอร์ 2 ตัวต่อกลับกัน ดังรูป 28 (a) เรียกว่าวงจรควบคุมสองทาง (Bidirectional controller) หรืออาจใช้ไทริสเตอร์ 1 ตัวและไดโอด 1 ตัวแทนได้ เรียกว่าวงจรควบคุมทางเดียว (Unidirectional controller) ดังแสดงในรูป 29 (a)

หลักการควบคุมมุมเฟส คือ สร้างสัญญาณจุดชนวนเกตของไทริสเตอร์ในวงจร ให้ไทริสเตอร์ตัวที่นำกระแสในครึ่งวัฏจักรบวกของแหล่งจ่ายไฟสลับทำงานที่มุมจุดชนวนระหว่าง 0 ถึง 180^0 (0 - π) และควบคุมให้ไทริสเตอร์ที่นำกระแสในครึ่งวัฏจักรลบของแหล่งจ่ายไฟสลับทำงานที่มุมจุดชนวนระหว่าง 180 – 360^0 (π - 2π) ลักษณะของการควบคุมทางเดียวนั้น จะควบคุมขนาดแรงดันไฟฟ้าใช้งานขาออกของวงจรได้ในย่าน 70.7% - 100% จึงเหมาะที่จะใช้กับโหลดที่เป็นตัวต้านทานที่กระแสไฟฟ้าใช้งานต่ำๆ เท่านั้น

1.) วงจรควบคุมทางเดียว ลักษณะของรูปคลื่นแรงดันไฟฟ้าใช้งานที่โหลด $V_{o(RMS)}$ และกระแสด้านออก (I_o) แสดงในรูป 29 (b)



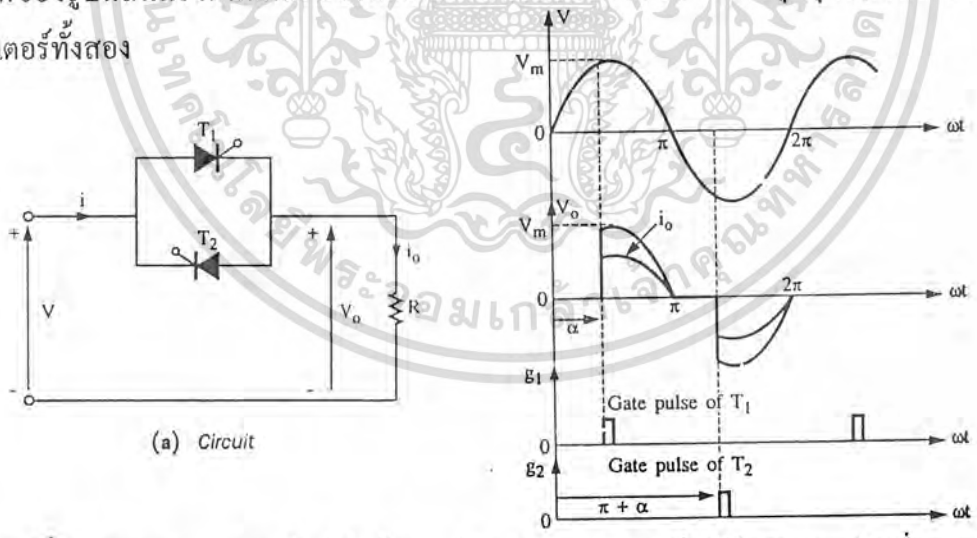
รูปที่ 29 วงจรควบคุมไฟสลับที่ใช้หลักการควบคุมมุมเฟส ทำการควบคุมทางเดียว และโหลดเป็นตัว ต้านทานที่มุมจุดชนวน (α)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากวงจรในรูป 29 ไทริสเตอร์ T_1 จะทำงานในครึ่งวัฏจักรบวก โดยมีสัญญาณจุดชนวนเกต (g_1) ที่ควบคุมได้ในย่าน $\alpha = 0$ ถึง $\alpha = \pi$ และไทริสเตอร์ T_1 เมื่อถูกจุดชนวนที่มุม α ใดๆ แล้วจะหยุดนำกระแสเองโดยธรรมชาติที่ $\omega t = \pi$ ในกรณีที่โหลดเป็นตัวต้านทานเพียงอย่างเดียว

2.)วงจรควบคุมสองทาง ลักษณะของวงจรควบคุมสองทางจะใช้ไทริสเตอร์ 2 ตัว ดังรูป 29 (a) โดยไทริสเตอร์ T_1 ทำงานในเวลา $\omega t = 0 - \pi$ และไทริสเตอร์ T_2 ทำงานในเวลา $\omega t = \pi - 2\pi$ โดยมีสัญญาณจุดชนวนเกตของไทริสเตอร์ T_1 และ T_2 (g_1 และ g_2) ดังแสดงในรูป 30 (b) ถ้าสัญญาณจุดชนวนเกตของไทริสเตอร์ T_1 เท่ากับ α สัญญาณจุดชนวนเกตของไทริสเตอร์ T_2 จะเท่ากับ $\pi + \alpha$ การวิเคราะห์หาค่าแรงดันไฟฟ้าใช้อาร์.เอ็ม.เอส.ที่โหลด และกระแสไฟฟ้าอาร์.เอ็ม.เอส. ที่โหลดนั้นขึ้นอยู่กับคุณลักษณะของโหลดชนิดต่างๆ ดังรายละเอียดต่อไปนี้

ก.) กรณีโหลดเป็นตัวต้านทาน เมื่อโหลดเป็นตัวต้านทานจะเป็นผลให้ไทริสเตอร์ T_1 ที่ถูกจุดชนวนเกตที่มุม $\omega t = \alpha$ เริ่มนำกระแส และหยุดนำกระแสเองโดยธรรมชาติ เมื่อแรงดันของแหล่งจ่ายเป็นศูนย์ที่มุม $\omega t = \pi$ ในเวลาถัดไป ไทริสเตอร์ T_2 จะถูกจุดชนวนให้นำกระแสที่มุม $\omega t = \pi + \alpha$ และจะหยุดนำกระแสเองเช่นกันที่มุม $\omega t = 2\pi$ ดังแสดงในรูป 30 (b) จะเป็นผลให้สามารถควบคุมขนาดแรงดันไฟสลับกลับที่โหลดได้ เพราะพื้นที่ของรูปคลื่นแรงดันตกคร่อมโหลดเปลี่ยนแปลงไปตามขนาดของมุมจุดชนวนเกตของไทริสเตอร์ทั้งสอง



รูปที่ 30 วงจรควบคุมแรงดันไฟสลับ 1 เฟส ควบคุมสองทางด้วยวิธีควบคุมเฟส เมื่อจุดชนวนเกตของ T_1ที่มุมαและ T_2ที่มุม $\pi+\alpha$มีโหลดเป็นตัวต้านทานเพียงเดียว

การคำนวณค่าแรงดันไฟฟ้า อาร์.เอ็ม.เอส.ที่โหลด ถ้ากำหนดให้ $v = V_m \sin \omega t$ และ$V = \frac{V_m}{\sqrt{2}}$ (แรงดันไฟฟ้า อาร์.เอ็ม.เอส.) สามารถคำนวณหาค่าแรงดันไฟสลับที่โหลดได้ดังนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

$$V_{o(rms)} = \sqrt{\frac{1}{\pi}\int_{\alpha}^{\pi} V_m^2 \sin^2 \omega t d\omega t}$$

$$= \left\{\frac{V_m^2}{\pi}\int_{\alpha}^{\pi}\frac{1}{2}(1-\cos 2\omega t)d\omega t\right\}^{\frac{1}{2}}$$

$$= \left\{\frac{V_m^2}{2\pi}\left(\pi-\alpha+\frac{1}{2}\sin 2\alpha\right)\right\}^{\frac{1}{2}} \therefore V_{o(rms)}$$

$$\therefore V_{o(rms)} = V\sqrt{\frac{1}{\pi}\left(\pi-\alpha+\frac{\sin 2\alpha}{2}\right)}$$

เนื่องจากโหลดเป็นตัวต้านทานเพียงอย่างเดียว ดังนั้นการหาค่ากระแสไฟฟ้าอาร์.เอ็ม.เอส. ที่โหลดจึงหาได้จากสมการคือ

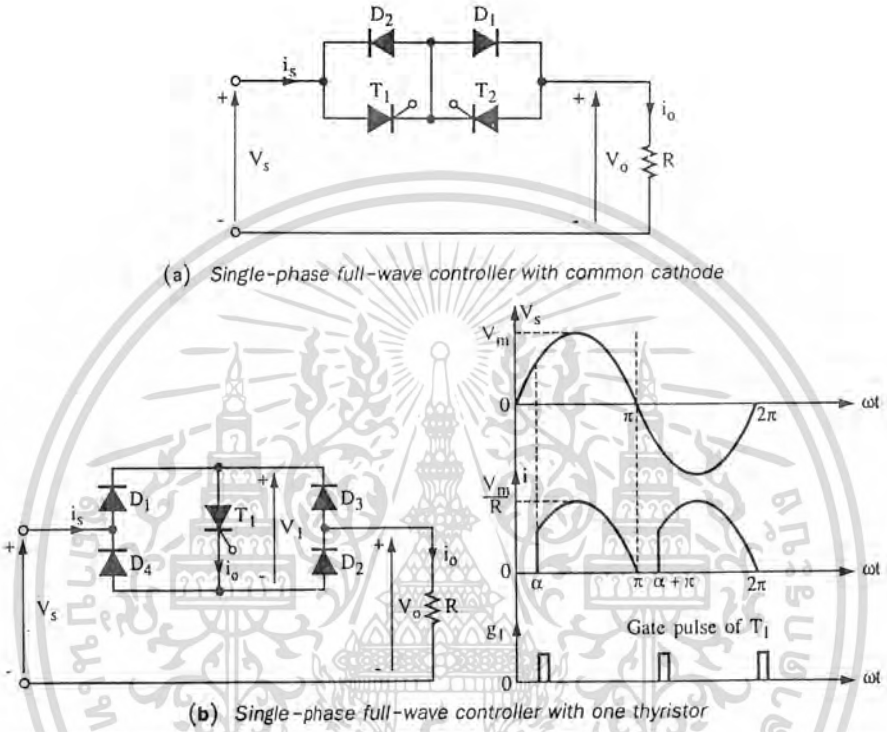
$$I_{o(RMS)} = \frac{V_{o(RMS)}}{R}$$

วงจรควบคุมแรงดันไฟสลับที่ควบคุมสองทาง ดังแสดงในรูป 30 (a) นั้น วงจรการจุดชนวนเกตต้องเป็นวงจรชนิดมีฉนวนกันระหว่างแคโทด เพราะถ้าไม่มีฉนวนกั้นไทริสเตอร์ทั้ง 2 ตัวไม่สามารถจะทำงานได้ เนื่องจากไทริสเตอร์ทั้ง 2 ตัวถูกลัดวงจรระหว่างแอโนดกับแคโทด ในกรณีที่วงจรจุดชนวนเกตไม่มีฉนวนกั้นระหว่างแคโทด ต้องหลีกเลี่ยงไปใช้วงจรกำลังแบบอื่นๆ เช่น วงจรควบคุมแรงดันไฟสลับที่ต่อแคโทดร่วม ดังแสดงในรูป 31 (a) วงจรนี้ต้องมีไดโอด D1 และ D2 ต่อเพิ่มเข้าในวงจร เพื่อให้กระแสในครึ่งวัฏจักรบวกไหลผ่าน T_1 และ D_1 และให้กระแสในครึ่งวัฏจักรลบไหลผ่าน T_2 และ D_2 หรือวงจรควบคุมแรงดันไฟสลับที่ใช้ไทริสเตอร์ตัวดียว ดังแสดงในรูป 31 (b) จะมีผลให้กระแสในครึ่งวัฏจักรบวกไหลผ่าน D_1 , T_1 และ D_2 เมื่อจุดชนวน T_1 ที่เวลา $\omega t = 0 - \pi$ และกระแสในวัฏจักรลบจะไหลผ่านโหลดได้โดยผ่าน D_3 , T_1 และ D_4 เมื่อจุดชนวน T_1 ที่เวลา $\omega t = \pi - 2\pi$ จะเห็นว่าไทริสเตอร์ T_1 นำกระแสทั้งในวัฏจักรบวกและวัฏจักรลบ

ข.) กรณีโหลดเป็นตัวต้านทานและตัวเหนี่ยวนำ เมื่อโหลดเป็นตัวต้านทานและเหนี่ยวนำ มุมเฟสของโหลด (ϕ) จะมีค่ามากกว่าศูนย์ ในกรณีนี้ค่าแรงดันไฟฟ้าใช้งานที่โหลดจะมีค่าสูงสุดเมื่อปรับมุมจุดชนวนเกต (α) เท่ากับมุมเฟสของโหลด (ϕ) สิ่งที่ต้องระวังสำหรับการสร้างสัญญาณจุดชนวนเกตในวงจรนี้คือ ถ้าเป็นสัญญาณพัลส์เดี่ยว (Single pulse) เมื่อปรับมุมจุดชนวน (α) เล็กกว่ามุมเฟสของโหลด (ϕ) จะเป็นผลให้ไทริสเตอร์นำกระแสได้เพียงตัวเดียวเท่านั้น แรงดันตกคร่อมโหลดจะกลายเป็นแรงดันไฟตรง ถ้านำวงจรไปประยุกต์ใช้งานในการควบคุมแรงดันด้านปฐมภูมิของหม้อแปลง จะมีผลให้ไทริสเตอร์หรือหม้อแปลงเสียหาย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ได้เมื่อปรับมุมจุดชนวน (α) น้อยกว่ามุมเฟสของโหลด (ϕ) ในทางปฏิบัติจะใช้พัลส์ต่อเนื่องในการจุดชนวนเกตแทนเพื่อป้องกันการเสียหายดังกล่าว ลักษณะของพัลส์ต่อเนื่องแสดงในรูป 31 (d) หรือเป็นพัลส์ที่มีความกว้างมาก (Long pulse) ดังรูป 31 (e)



รูปที่ 31 แสดงวงจรควบคุมไฟสลับแบบอื่นๆ

ลักษณะของวงจรเมื่อโหลดเป็นตัวเหนี่ยวนำแสดงในรูป 32 (a) , (c) และรูปคลื่นแรงดันไฟฟ้าใช้งานและกระแสไฟฟ้าอาร์.เอ็ม.เอส.ที่โหลดแสดงในรูป 32 (b)

การคำนวณค่าแรงดันไฟฟ้าอาร์.เอ็ม.เอส.ที่โหลด พิจารณาจากรูปคลื่นของวงจรในรูป 32 (b) จะเห็นว่ามุมจุดชนวนเกตของไทริสเตอร์คือมุม α และมุมหยุดนำกระแสของไทริสเตอร์คือ มุม β ดังนั้นสมการคำนวณค่าแรงดันไฟฟ้า อาร์เอ็มเอส ที่โหลดคือ

$$V_{o(rms)} = \sqrt{\frac{1}{\pi}\int_{\alpha}^{\beta} v_m^2 \sin^2 \omega t d\omega t}$$

$$= \left\{\frac{V_m^2}{2\pi}\int_{\alpha}^{\beta}(1-\cos 2\omega t)d\omega t\right\}^{\frac{1}{2}}$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

$$= V\left[\frac{1}{\pi}\left(\beta-\alpha+\frac{\sin 2\alpha}{2}-\frac{\sin 2\beta}{2}\right)\right]^{\frac{1}{2}}$$

$$\therefore V_{o(rms)} = V\sqrt{\frac{1}{\pi}\left(\beta-\alpha+\frac{\sin 2\alpha}{2}-\frac{\sin 2\beta}{2}\right)}$$

เมื่อ $\beta = \pi+\phi$

และ $\phi = \tan^{-1}\left(\frac{\omega L}{R}\right)$

การคำนวณค่ากระแสไฟฟ้า อาร์.เอ็ม.เอส. ที่โหลด

พิจารณาวงจรในรูปที่ 33 (a)จะได้ว่า

$$L\frac{di}{dt}+R_{i1} = \sqrt{2}\ y\sin\omega t \qquad (13)$$

แก้สมการที่ (13) เพื่อหาค่ากระแส i_1 ได้ว่า

$$i_1 = \frac{\sqrt{2}}{z}V\sin(\omega t-\phi)+A_1e^{-\left(\frac{Rt}{L}\right)}$$

หาค่าคงที่ A_1 เมื่อ $\omega t=\alpha$ และ $i_1=0$ จากสมการ i_1 จะได้ว่า

$$A_1 = \frac{\sqrt{2}}{Z}V\sin(\alpha-\phi)e^{\left(\frac{R}{L}\left(\frac{\alpha}{\omega}\right)\right)} \qquad (14)$$

แทนค่า (14) ลงในสมการ i_1 จะได้ว่า

$$i_1=\frac{\sqrt{2}}{Z}V\left[\sin(\omega t-\phi)-\sin(\alpha-\phi)e^{\left(\frac{R}{L}\left(\frac{\alpha}{\omega t}\right)\right)}\right] \qquad (15)$$

มุม β คือมุมหยุดนำกระแสของไทริสเตอร์ T_1 หาได้จากเงื่อนไขของกระแส i_1 โดยที่ $(\omega t=\beta)=0$ โดยแทนลงในสมการที่ (15) จะได้ความสัมพันธ์ดังนี้

$$\sin(\beta-\phi)=\sin(\alpha-\phi)e^{\left(\frac{R}{L}\left(\frac{\alpha-\beta}{\omega}\right)\right)} \qquad (16)$$

กระแสไฟฟ้า อาร์.เอ็ม.เอส. ที่ไหลผ่านไทริสเตอร์ได้จากสมการที่ (15)คือ

$$I_R=\sqrt{\frac{1}{2\pi}\int_{\alpha}^{\beta}i_1^2\,d\omega t}$$

$$\therefore I_R=\frac{V}{Z}\left[\frac{1}{\pi}\int_{\alpha}^{\beta}\left\{\sin(\omega t-\phi)-\sin(\alpha-\phi)\right\}e^{\left(\frac{R}{L}\left(\frac{\alpha}{\omega t}\right)\right)}\right]^{\frac{1}{2}} \qquad (17)$$

กระแสไฟฟ้า อาร์.เอ็ม.เอส. ที่ไหลผ่านโหลดจะหาได้จากสมการต่อไปนี้

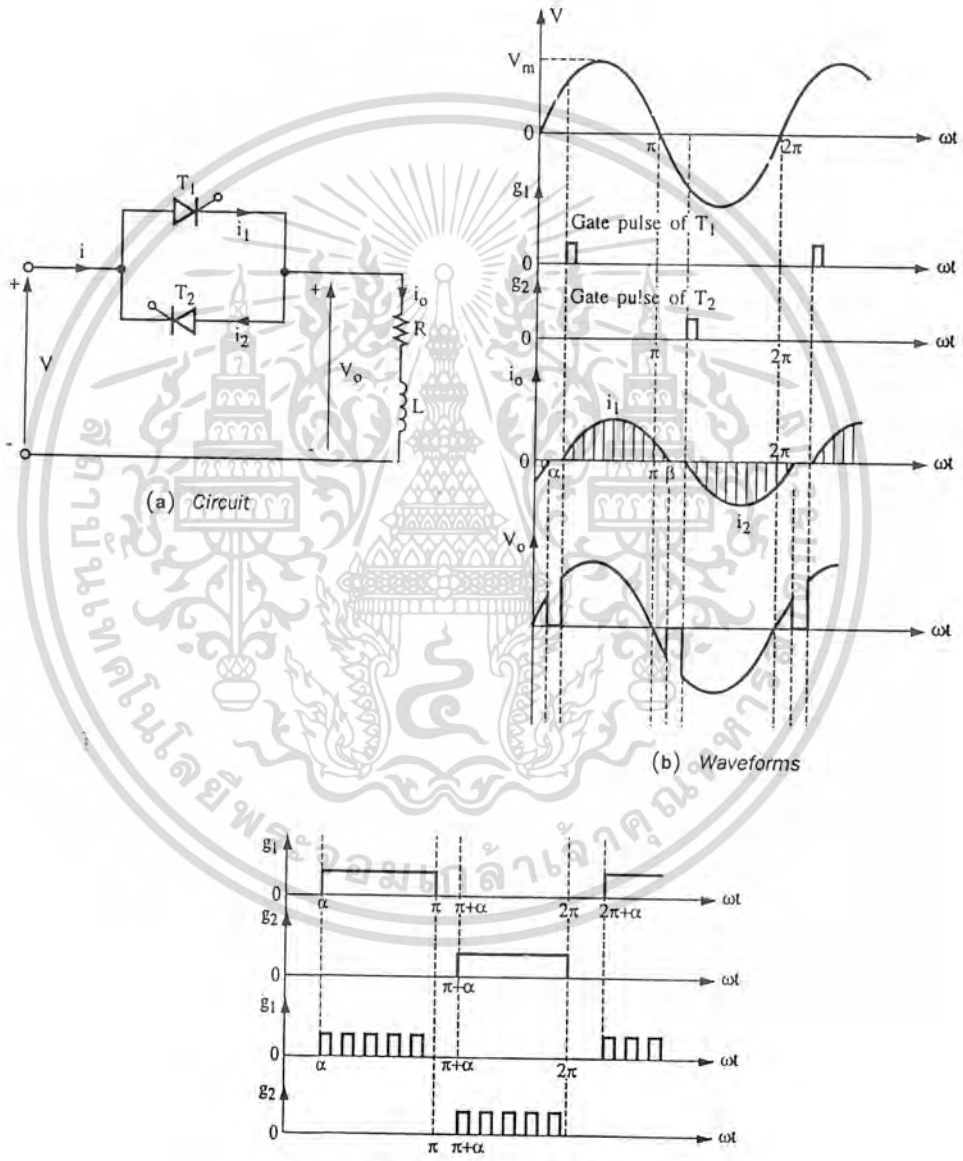
$$I_{o(rms)}=\sqrt{2}I_R \qquad (18)$$

กระแสเฉลี่ยที่ไหลผ่านไทริสเตอร์ สามารถที่จะหาได้จากสมการที่ (15)ได้ดังนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

$$I_A = \frac{1}{2\pi}\int_{\alpha}^{\beta} i_1 d\omega t$$

$$\therefore I_A = \frac{V}{\pi Z\sqrt{2}}\int_{\alpha}^{\beta}\left[\sin(\omega t - \phi) - \sin(\alpha - \phi)e^{\left(\frac{R}{L}\left(\frac{\alpha}{\omega t}\right)\right)}\right]d\omega t \qquad (19)$$



รูปที่ 32 วงจรควบคุมไฟสลับ 1 เฟส เมื่อโหลดเป็นตัวต้านทานและตัวเหนี่ยวนำและจุดชนวนเกต ที่มุม $\alpha > \Phi$ ด้วยพัลส์เดี่ยว และพัลส์แบบอื่น ๆ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ข้อสังเกต 1) ถ้ามุม $\alpha = \phi$ จากสมการ16 จะได้ว่า

$$\sin(\beta - \phi) = \sin(\beta - \alpha) = 0$$

และ $\beta - \alpha = \gamma = \pi$ (20)

3) มุม γ คือจำนวนมุมนำกระแสของไทริสเตอร์จะมีค่าไม่เกิน π และถ้าทำการควบคุมมุมจุดชนวน (α) ไม่ให้น้อยกว่ามุมเฟสของโหลด (ϕ) ย่านการควบคุมมุมจุดชนวนการนำกระแสของไทริสเตอร์จะมีค่าดังสมการ คือ

$$\phi < \alpha < \pi \quad (21)$$

3) ถ้าจุดชนวนเกตที่มุม $\alpha < \phi$ ด้วยพัลส์ต่อเนื่องหรือพัลส์ที่มีความกว้างมาก กระแสไฟฟ้าอาร์.เอ็ม.เอส. ที่โหลดจะไม่เปลี่ยนแปลงไปตามค่าของมุม α

จะเห็นว่าการใช้สมการ (17) และ (19) หาค่ากระแสที่ไหลผ่านไทริสเตอร์จะยุ่งยากมาก เนื่องจากสมการดังกล่าวยังไม่อยู่ในรูปแบบที่สะดวกต่อการนำไปใช้งาน ในทางปฏิบัตินิยมใช้กราฟความสัมพันธ์ระหว่างค่ากระแสไฟฟ้าอาร์.เอ็ม.เอส.ต่อหน่วย และค่ากระแสเฉลี่ยต่อหน่วยกับมุมจุดชนวนเกตของไทริสเตอร์ช่วยในการคำนวณหาค่ากระแส I_R และ I_A ได้อย่างรวดเร็วดังรูป 34 พิจารณาจากสมการกระแส I_R กำหนดให้ค่าฐานของกระแสคือ $\frac{V_m}{Z}$ ดังนั้นจะได้ค่ากระแสไฟฟ้าใช้งานต่อหน่วย (I_{RN}) คือ

$$I_{RN} = \left[\int_{\alpha}^{\beta}\left[\sin(\omega t - \phi) - \sin(\alpha - \phi)e^{\left(\frac{R}{L}\left(\frac{\alpha}{\omega t}\right)\right)}\right]^2 d\omega t\right]^{\frac{1}{2}}$$

ซึ่งค่า I_{RN} สามารถหาได้จากกราฟรูปที่ 34(b)นั่นแสดงว่ากระแสไฟฟ้า อาร์.เอ็ม.เอส. ที่ผ่านไทริสเตอร์ (I_R) หาได้จาก

$$I_R = \frac{V_m}{Z} I_{RN}$$

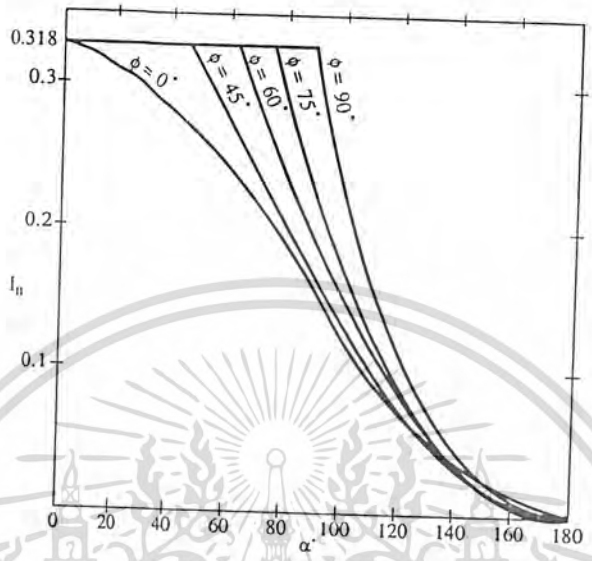
ในทำนองเดียวกันค่ากระแสเฉลี่ยต่อหน่วย (I_N) ที่ไหลผ่านไทริสเตอร์คือ

$$I_N = \frac{1}{2\pi}\int_{\alpha}^{\beta}\left[\sin(\omega t - \alpha) - \sin(\alpha - \phi)e^{\left(\frac{R}{L}\left(\frac{\alpha}{\omega t}\right)\right)}\right] d\omega t$$

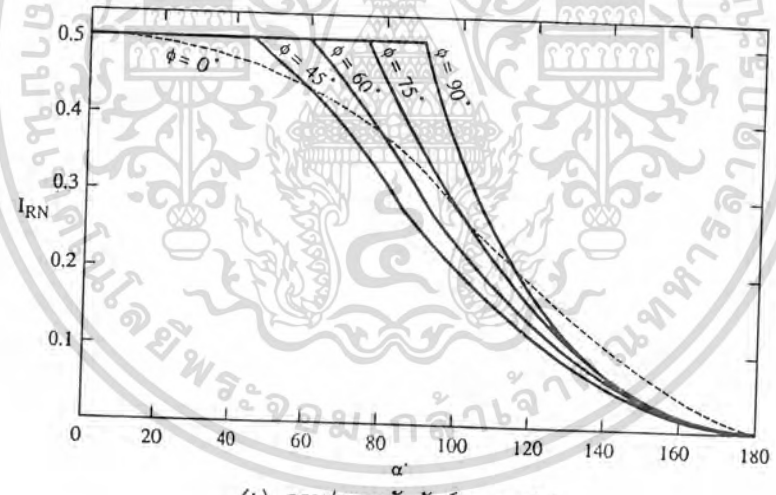
ดังนั้นสามรถหาค่ากระแส I_N ได้จากกราฟรูปที่34(a)ค่ากระแสเฉลี่ยที่ไหลผ่านไทริสเตอร์จะได้ดังนี้

$$I_A = \frac{V_m}{Z} I_n$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



(a) กราฟความสัมพันธ์ของ $I_N = f(\alpha)$



(b) กราฟความสัมพันธ์ของ $I_{RN} f(\alpha)$

รูปที่ 33 แสดงกราฟของ I_N และ I_{RN} ที่ใช้คำนวณค่ากระแสไทรสเตอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.13 การออกแบบและการพันหม้อแปลงกระแส

หม้อแปลงกระแส เรียกว่า "เคอเรนต์ทรานสฟอร์เมอร์" (CT : Current Transformer) จำเป็นจะต้องรู้องค์ประกอบหลายๆ อย่างที่เกี่ยวข้องพอสมควร เช่น

พิกัดกระแสสูงสุดที่ใช้งาน .....(A)

พิกัดกระแสสูงสุดของมิเตอร์ .....(A)

แรงดันตกคร่อมมิเตอร์ขณะวัดกระแสเต็มพิกัด .....(V)

การกำหนดพิกัดหรือขนาดของหม้อแปลงกระแส (CT : Current Transformer) กำหนดไว้ดังนี้

พิกัดกระแสสูงสุดที่ใช้งาน / พิกัดกระแสสูงสุดของมิเตอร์

เช่น 90/5, 100/5, 400/5, 500/5, 800/5, 1000/5, 1200/5, 1600/5, 2000/5, ฯลฯ

การหาแรงดันตกคร่อมมิเตอร์ขณะวัดกระแสเต็มพิกัด หาได้จาก

พิกัดกระแสสูงสุดของมิเตอร์ x ความต้านทานภายในของมิเตอร์

1) ขั้นตอนการออกแบบ

หาแรงดันตกคล่อมมิเตอร์ขณะใช้งานเต็มพิกัด = 5 x 0.5 = 2.5 V

คำนวณหากำลังของหม้อแปลงกระแสจากสูตร P= U . I .....(V-A)

P = 2.5 x 5 = 12.5 .....(V-A)

หาขนาดของพื้นที่หน้าตัดของแกนเหล็กจากสูตร A= 1.1 . /U . I .....(ตร.ซม.)

A = 1.1/2.5 x 5 = 3.89 .....(ตร.ซม.)

หาขนาดของแกนเหล็กโดยการหารากที่ 2 ของพื้นที่หน้าตัดจะได้ /3.89 = 1.97 .(ตร.ซม.) การพิจารณาเลือกขนาดของแกนเหล็ก ซึ่งควรเป็นแกนรูปตัว U , I ขนาด 2 ซม. แต่ในตลาดอาจไม่มีขายจึงควรหาแกนตัว E,I มาประยุกต์โดยตัดแบ่งครึ่ง และอัดซ้อนกันให้มีขนาดพื้นที่หน้าตัดตามที่คำนวณได้ในข้อ 3 โดยอาจเลือกแกนเหล็กขนาด 2.5 หรือ 3.125 ซม. ก็ได้

คำนวณหาจำนวนรอบของขดลวดที่ต่อกับมิเตอร์ ได้ดังนี้

U = 45/A x U = 45/3.89 x 2.5 = 28.9

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากจำนวนรอบที่คำนวณได้เมื่อกระแสที่ใช้งานเต็มพิกัด (Full Load) จะมีแรงดันตกคร่อมตัวหม้อแปลงกระแสเองทำให้แรงดันตกคร่อมมิเตอร์ลดลง การวัดค่ากระแสจะผิดพลาดอยู่ในเกณฑ์ต่ำกว่าความเป็นจริง จึงจำเป็นจะต้องเพิ่มแรงดันให้กับมิเตอร์ โดยการพันจำนวนรอบเพิ่มขึ้นตามหลักการส่งถ่ายกำลังไฟฟ้าสูงสุด (ความต้านทานภายในกับความต้านทานภายนอกจะเท่ากับแรงดันลดลงครึ่งหนึ่ง) แต่ประสิทธิภาพของหม้อแปลงกระแสจะดีกว่ามิเตอร์ที่ใช้วัด ดังนั้นในการพันจำนวนรอบเพิ่มขึ้นจะเพิ่มประมาณ 60- (80)% กรณีเพิ่ม 80% จะต้องพันขดลวดมิเตอร์

$$28.9 \times 1.8 = 52$$

หมายเหตุ หม้อแปลงกระแสขณะต่อใช้งานแต่ยังไม่ต่อเข้ามิเตอร์จะต้องต่อปลายสายทั้งสองข้างรวมกัน (Short Circuit)

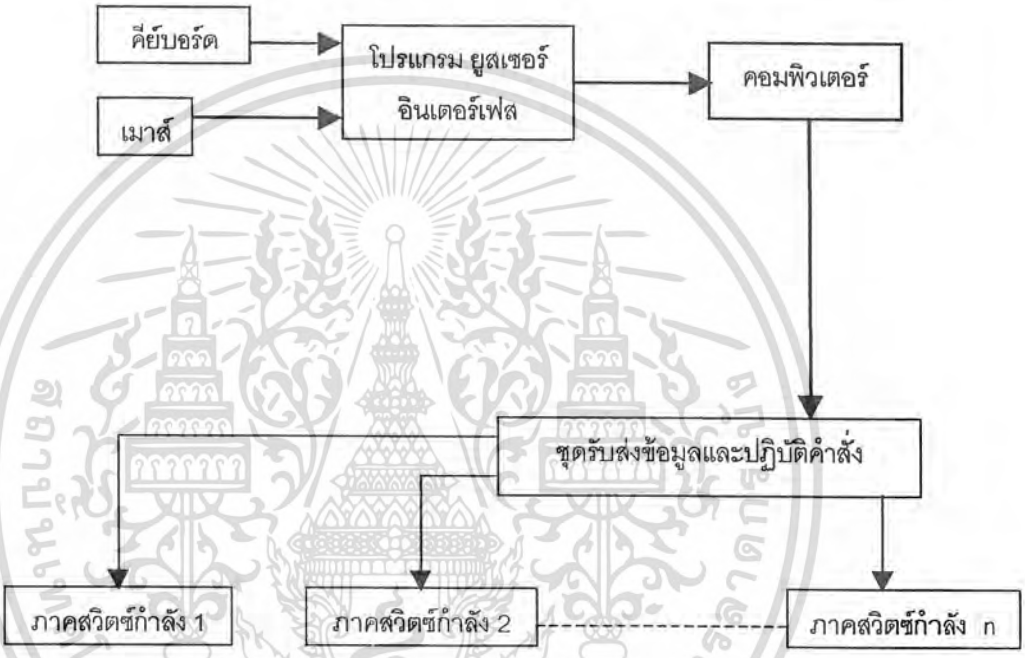
หาเบอร์ลวดที่จะใช้พันจากตารางการเทียบเบอร์ลวด ได้ดังนี้ #17 , #18 สำหรับขดลวดปฐมภูมิ (Primary Winding) จะเป็นสายประธานหรือสายเมน (Main) โดยร้อยสาย ผ่านแกนเหล็ก เราถือว่าเป็นการพันหนึ่งรอบ

หากเราต้องการนำเอาหม้อแปลงกระแสขนาด 100/5 ไปใช้กับมิเตอร์ขนาด 50 A เราสามารถทำได้ โดยการพันสายประธานหรือสายเมน (Main) รอบแกนเหล็กสองรอบและต้องจ่ายภาระทางไฟฟ้า ไม่เกิน 50 A

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 3
บล็อกไดอะแกรมและการทำงานของเครื่องควบคุมไฟเวที

บล็อกไดอะแกรมส่วนหลักๆ แสดงดังรูปที่ 34 ในบทนี้จะอธิบายถึงรายละเอียดปลีกย่อยในแต่ละชุดของบล็อกไดอะแกรม



รูปที่ 34 แสดงบล็อกไดอะแกรมหลักของเครื่องควบคุมไฟเวที

3.1 ภาค คีย์บอร์ด เมาส์ โปแกรมยูสเซอร์อินเตอร์เฟส และคอมพิวเตอร์

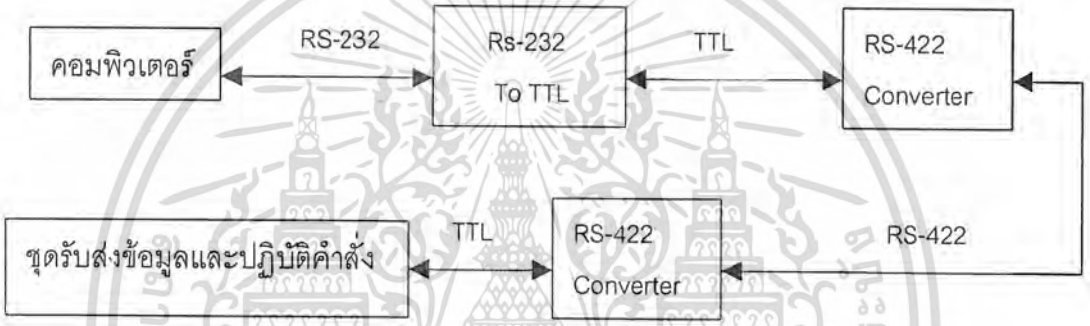
ในภาคนี้จะมีส่วนที่สำคัญคือโปแกรมยูสเซอร์อินเตอร์เฟส ซึ่งจะมีหน้าที่รับคำสั่งจากคีย์บอร์ด และ เมาส์ แล้วสั่งให้คอมพิวเตอร์ประมวลผลคำสั่งนั้นแล้ว ส่งออกพอร์ตอนุกรมของคอมพิวเตอร์เพื่อส่งให้ ชุดรับส่งข้อมูลและปฏิบัติคำสั่งและผู้ใช้จะสารถควบคุมหลอดไฟทั้งหมด รวมทั้งกำหนดคุณสมบัติต่างๆของหลอดไฟ เช่น แรงดันสูงสุดที่หลอดในดวงนั้นหรือกลุ่มนั้นทนได้รวมถึงการอ่านค่ารูปแบบอัติโนมัติที่จัดเก็บเป็นไฟล์ไว้มาสั่งให้เครื่องควบคุมไฟเวทีทำงานตามที่ได้จัดทำไว้ก่อนแล้ว จากโปรแกรมนี้

3.2 ภาคการติดต่อระหว่างคอมพิวเตอร์กับชุดรับส่งข้อมูล

การติดต่อระหว่างคอมพิวเตอร์กับชุดรับข้อมูลจะใช้การติดต่อแบบอนุกรม ทางพอร์ตอนุกรมของคอมพิวเตอร์ โดยปกติคอมพิวเตอร์จะส่งข้อมูลของพอร์ตอนุกรมเป็นมาตรฐาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สัญญาณแบบ RS-232 ซึ่งสามารถรับส่งข้อมูลอนุกรมในระยะทางได้เพียง ไม่เกิน 50 ฟุต และอัตราการรับส่งข้อมูลสูงสุดเพียง 20,000 บิตต่อวินาที จึงต้องนำสัญญาณนี้ผ่านวงจร แปลงสัญญาณจาก RS-232 เป็นRS-422 เพื่อให้สามารถรับส่งข้อมูลได้ไกลมากขึ้น และยังเป็นการประหยัดสายสัญญาณที่เดินจากคอมพิวเตอร์มายังชุดรับส่งสัญญาณด้วย เพราะใช้สายสัญญาณเพียง 4 เส้น และลดปัญหาเรื่องสัญญาณรบกวนได้ แต่เนื่องจากวงจรแปลงสัญญาณเป็น RS-422 ต้องการสัญญาณอินพุตเป็นระดับทีทีแอล การต่อใช้งานจริงจึงต้องแปลงสัญญาณ RS-232 เป็นสัญญาณในระดับ ทีทีแอลก่อนที่จะส่งให้กับภาค RS-422 Converter ลักษณะของการเชื่อมต่อบล็อกไดอะแกรมแสดงดังรูปที่ 35



รูปที่ 35 การแปลงสัญญาณ RS-232 เป็น RS-422

3.2.1 วงจร RS – 232 to RS – 422 Convertor

จากบล็อกไดอะแกรมรูปที่ 36 สัญญาณข้อมูลที่เป็นระดับ RS – 232 ซึ่งรับส่งอยู่ระหว่างคอมพิวเตอร์กับวงจรรวมเบอร์ MAX 232 โดย MAX 232 จะมีหน้าที่ แปลงสัญญาณระดับ RS – 232 ที่รับเข้ามาจากคอมพิวเตอร์ ไปเป็นสัญญาณในระดับ TTL และในเวลาเดียวกัน MAX – 232 สามารถที่จะแปลงข้อมูลระดับ TTL ไปเป็นระดับ RS – 232 ได้ด้วยเพราะการทำงานทั้ง 2 หน้าที่นั้นจะแยกจากกันอิสระ

วงจรรวม เบอร์ SN75176 มีหน้าที่แปลงสัญญาณระดับ TTL ไปเป็นระดับ RS – 422 หรือ แปลงสัญญาณระดับ RS – 422 ไปเป็น TLL การทำงานดังกล่าวจะต้องเลือกว่าจะให้วงจรรวม SN75176 ตัวนั้น ๆ ทำงานในแบบใด เพราะวงจรภายในของ SN75176 มีเพียงชุดเดียว เมื่อ คอมพิวเตอร์ส่งข้อมูลที่เป็น ระดับ RS – 232 ออกมา วงจรรวม MAX 232 จะทำการแปลงระดับสัญญาณข้อมูลดังกล่าวไปเป็นระดับ TTL แล้วส่งเข้าสู่วงจรรวม SN75176 ที่ทำหน้า ที่แปลงสัญญาณระดับ TTL ไปเป็นระดับ RS – 422 สัญญาณระดับดังกล่าวจะส่งไปยังภาครับที่เป็นชุดควบคุมภาคกำลังที่เวที โดยผ่านสาย twist pair แบบ UTP

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ในขณะเดียวกันหากชุดควบคุมภาคกำลัง ที่เวทีต้องการที่จะส่งข้อมูลให้คอมพิวเตอร์ ก็จะส่งข้อมูลในระดับ RS – 422 มาบนสายเกลียวคู่ ซึ่งมีชื่อว่า RX + , RX- สัญญาณดังกล่าวจะถูกส่งเข้าสู่วงจรรวม SN75176 ที่ทำหน้าที่แปลงสัญญาณระดับ RS – 422 เป็นระดับ TTL จากนั้นส่งข้าสู่วงจรรวม MAX 232 เพื่อแปลงระดับสัญญาณเป็น RS – 232 เพื่อส่งเข้าคอมพิวเตอร์ต่อไป

ในขณะที่สัญญาณมีการเข้าออกตัววงจรรวม MAX 232 จะสั่งให้ LED 1 กระพริบเพื่อแสดงการแลกเปลี่ยนข้อมูลระหว่างคอมพิวเตอร์ กับชุดควบคุมภาคกำลัง

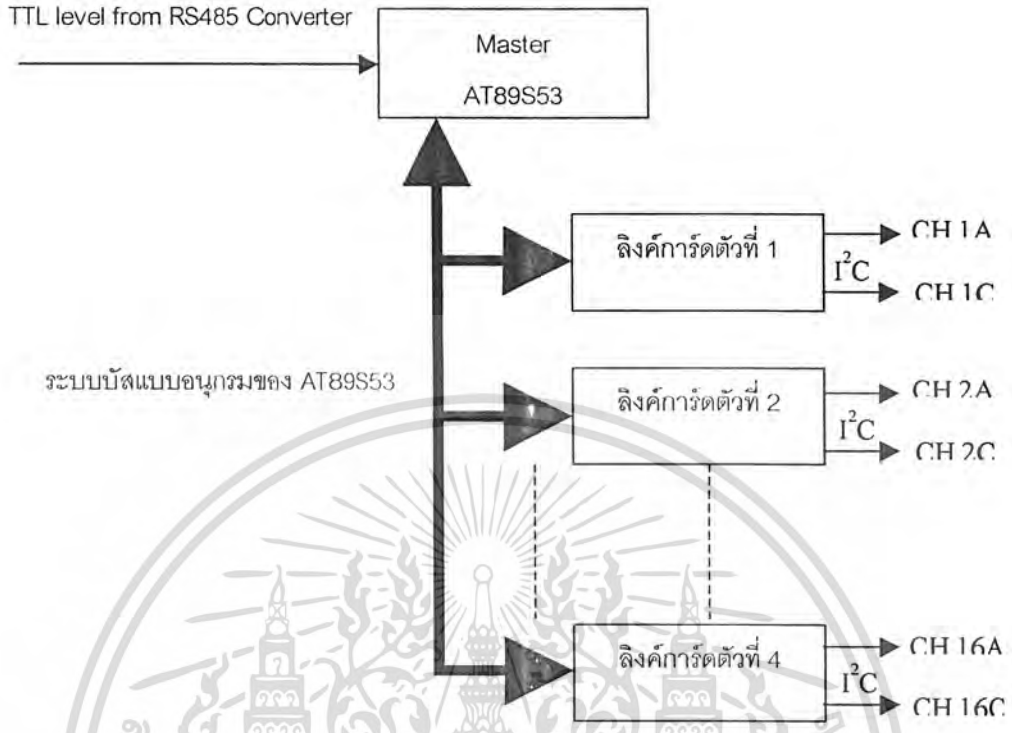
3.3 ภาคชุดรับส่งข้อมูลและปฏิบัติคำสั่ง

ชุดรับข้อมูลจะอยู่ที่หน้าเวที มีหน้าที่รับส่งข้อมูลระหว่างคอมพิวเตอร์ และจัดการเกี่ยวกับการสั่งชุดหรี่ไฟ ซึ่งเรียกว่า เพาเวอร์แพกซึ่งจะรับคำสั่งจากชุดรับข้อมูลนำมาสั่ง ภาคสวิตซ์กำลังเพื่อควบคุมกระแสและแรงดันที่จ่ายให้โหลดซึ่งเป็นหลอดไฟต่อไป

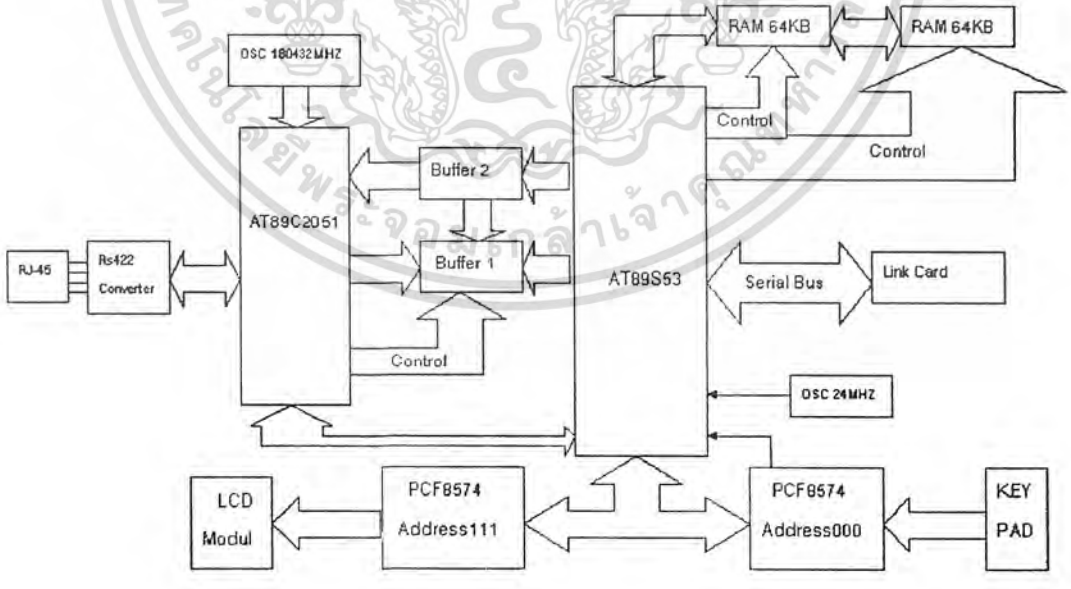
ส่วนประกอบหลักของชุดรับข้อมูลคือ ไมโครคอนโทรลเลอร์ เบอร์ AT89S53 ซึ่งในที่นี้เรียกว่ามาสเตอร (Master) ทำหน้าที่รับส่งข้อมูล กับคอมพิวเตอร์ และทำหน้าที่จัดการ การส่งข้อมูลให้กับชุดเพาเวอร์แพก การส่งข้อมูลให้ชุดเพาเวอร์แพกจะใช้ระบบบัสการส่งข้อมูลแบบไอแสควซีบัส โดยไมโครคอนโทรลเลอร์เบอร์ AT89S53 จะส่งสัญญาณข้อมูลให้กับวงจรสร้างสัญญาณ ไอแสควซีบัส ซึ่งในที่นี้เรียกว่า ลิงค์การ์ด (Link Card) เพื่อติดต่อกับชุดเพาเวอร์แพกต่อไป โดยลิงค์การ์ดจะมีไมโครคอนโทรลเลอร์เบอร์ AT89C2051 ทำหน้าที่จัดการในหน้าที่ต่างๆ ได้แก่ การแปลงสัญญาณแบบอนุกรมเป็น ไอแสควซีบัส และประมวลคำสั่งต่างๆที่รับมาจาก ตัวมาสเตอร์ เพื่อนำไปสั่งควบคุมชุดเพาเวอร์แพกต่อไป โดยมาสเตอร์สามารถที่จะรับและส่งข้อมูลหรือคำสั่งกับไมโครคอนโทรลเลอร์ได้ตลอดเวลา ในส่วนของชุดรับส่งสัญญาณและปฏิบัติคำสั่งนี้สามารถที่จะต่อลิงค์การ์ดได้ทั้งหมด 4 ตัว การต่อลิงค์การ์ดจะต่อแบบขนานกันไปได้ไม่เกิน 4 ตัว และในแต่ละการ์ดจะมีช่องสำหรับต่อบัสแบบไอแสควซีบัส ได้ 4 ช่องดังรูปที่ 35 การเชื่อมต่อของบล็อกไดอะแกรมในชุดรับส่งข้อมูลและปฏิบัติคำสั่งแสดงดังรูปที่ 36

เมื่อคอมพิวเตอร์ส่งข้อมูลผ่านมาทางสายเกลียวคู่ซึ่งเป็นข้อมูลระดับ RS – 422 สัญญาณข้อมูลดังกล่าวจะส่งเข้าสู่วงจรรวม SN75176 เพื่อแปลงระดับสัญญาณเป็น TTL และส่งเข้าสู่ไมโครคอนโทรลเลอร์ AT89C2051 เมื่อ AT89C2051 ได้รับข้อมูลแล้วจะส่งข้อมูลไปให้หน่วยพักข้อมูลที่ Buffer 1 และสั่งให้ Buffer 1 เก็บข้อมูล จากนั้น AT89C2051 จะส่งสัญญาณขัดจังหวะ AT 89S53 ไปบน Flag Bus เพื่อบอกให้ AT89S53 รับรู้ว่ามีข้อมูลสงมาให้ AT89S53 จะทำการสั่งเปิดเส้นทางข้อมูลของ Buffer 1 เพื่อให้ข้อมูลเข้ามาแล้วจะอ่านข้าไป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 36 การเชื่อมต่อภายในของบล็อกไดอะแกรมในชุดลิงค์การ์ดและปฏิบัติคำสั่ง



รูปที่ 37 การเชื่อมต่อภายในของบล็อกไดอะแกรมในชุดรับส่งข้อมูลและปฏิบัติคำสั่ง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ประมวลผลต่อไป เมื่อ AT89S53 อ่านข้อมูลเสร็จแล้วจะส่งสัญญาณสถานะไปบอกให้ AT89C2051 รับทราบว่าการอ่านข้อมูลเสร็จสิ้นแล้ว หาก AT89C2051 มีข้อมูล ชุดต่อไปที่ส่งให้อีก ก็จะทำการส่งสัญญาณต่าง ๆ เหมือนดังข้างต้นต่อไป

เมื่อ AT89S53 ต้องการส่งข้อมูลไปให้คอมพิวเตอร์ จะเริ่มต้นกระบวนการคือ AT89S53 จะส่งข้อมูลไปเก็บไวในหน่วยพักข้อมูลที่ 2 (Buffer 2) และตามด้วยการส่งสัญญาณไปขัดจังหวะ AT89C2051 เมื่อ AT89C2051 รับการขัดจังหวะ จะส่งสัญญาณไปเป็นเส้นทางข้อมูลของหน่วยพักข้อมูลที่ 2 แล้วอ่านข้อมูลส่งออกไปทาง พอร์ต อนุกรมเข้าสู่วงจรรวม SN75176 เพื่อแปลงระดับสัญญาณเป็น RS – 422 และส่งไปที่คอมพิวเตอร์ต่อไป

เมื่อมีการกด Key ใด ๆ PCF8574 จะส่งสัญญาณไปขัดจังหวะ AT89S53 จะอ่านข้อมูลทาง I2C Bus จาก PCF8574 ที่ตำแหน่ง 71H, แล้วนำข้อมูลไปประมวลผลต่อไป

เมื่อ AT89S53 จะส่งข้อมูลให้ LCD จะกระทำผ่าน PCF8574 ที่ตำแหน่ง 76H ด้วยระบบ I2C BUS ซึ่ง LCD จะใช้การติดต่อ ข้อมูลแบบครั้งละ 4 Bit เมื่อประหยัดจำนวนของ PCF8574

สำหรับการส่งข้อมูลระหว่าง AT89S53 กับ I2C BUS Drive จะใช้วิธีการส่งข้อมูลทางพอร์ตอนุกรม ด้วยความเร็ว 750 kbps

3.4 ภาคสวิตช์กำลัง

ภาคสวิตช์กำลังจะประกอบด้วยส่วนสำคัญ 2 ส่วนคือ

3.4.1 วงจรแปลงสัญญาณดิจิตอลเป็นอานาลอก

อุปกรณ์หลักที่ทำหน้าที่นี้คือวงจรรวมเบอร์ TDA8444 ซึ่งการต่อใช้งานจะใช้กับบัส ระบบไอแสควซี ซึ่ง TDA8444 หนึ่งตัวมีเอาต์พุตที่แยกอิสระกันถึง 8 ช่อง และเมื่อต่อในระบบบัสแบบ I^2C จะสามารถต่อ TDA8444 นี้ได้ถึง 8 ตัวในบัสเดียวกัน ดังนั้นจะพบว่าบัส ไอแสควซีหนึ่งเส้นสามรถที่จะควบคุมหลอดไฟได้ทั้งหมด 64 ช่อง นอกจากนี้ลิงค์การ์ด 1 ตัว จะมีบัสแบบ I ไอแสควซี อยู่ 4 บัส ทำให้ลิงค์การ์ด 1 ตัวสามารถที่ต่อ หลอดไฟได้ทั้งหมด 256 ช่อง

คุณสมบัติของ TDA8444

เป็นวงจร D/A ขนาด 6 Bit จำนวน 8 ช่อง

สามารถปรับการแกว่งของแรงไฟที่เอ้าพุตได้

เอาต์พุตเป็นแบบวงจร พุชพูล

มีวงจรป้องกันเอาต์พุตลัดวงจร

ย่านการทำงานไฟเลี้ยงกว้าง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

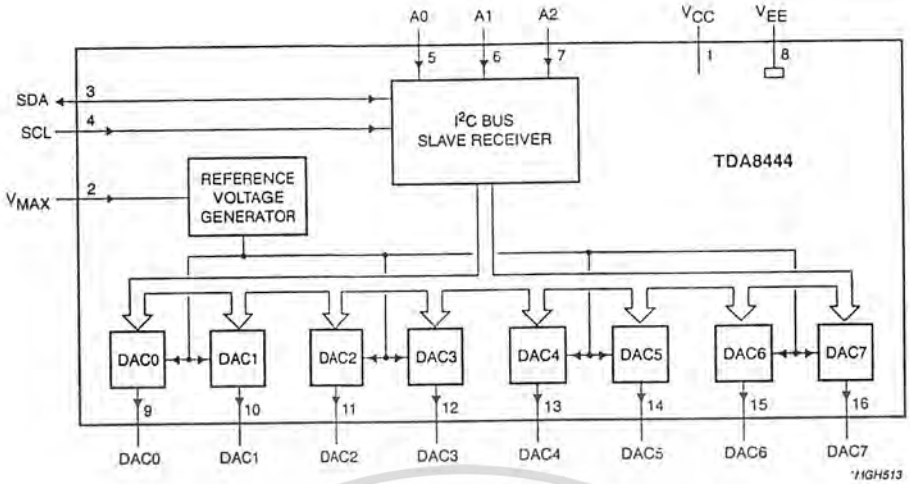


Fig.1 Block diagram.

รูปที่ 38 บล็อกไดอะแกรมภายใน TDA8444

3..4.2 การติดต่อกับ TDA844

TAD8444 จะทำงานเป็นเพียงตัวรับข้อมูลได้อย่างเดียวเท่านั้น โดยมีรูปแบบของขบวนสัญญาณ ดังรูปที่ 39

Table 1 I²C-bus format (see note 1)

S	0 1 0 0 A2 A1 A0 0	A	I3 I2 I1 I0 SD SC SB SA	A	X X D5 D4 D3 D2 D1 D0	A	P

Note

1. S = START condition; A2 to A0 = programmable address bits; A = Acknowledge; I3 to I0 = Instruction bits; SD to SA = subaddress bits; X = don't care; D5 to D0 = data bits; P = STOP condition.

รูปที่ 39 รูปแบบข้อมูลที่ใช้ติดต่อกับ TDA8444

หมายเลขตำแหน่งที่สามารถกำหนดได้ คือ

TDA8444 และ TDA8444AT : 40H,42H,44H,46H,4ah,4CH และ 46H

TDA8444T : 48H,4AH,4CH และ 46H (ขา A2 ต้องเป็น 1 เสมอ)

คำสั่งที่ใช้ได้คือ 00H ถึง 0FH, F0H ถึง FFH

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ถ้าหากป้อนคำสั่งนอกเหนือจาก 4 คำสั่งนี้ TDA8444 จะไม่ปฏิบัติตามคำสั่งนั้น ๆ แต่จะยังคงสร้างสัญญาณบิตรับรู้

สำหรับคำสั่ง 0 จะมีผลคือเมื่อป้อนข้อมูลที่ต้องการแปลง เพียงข้อมูลเดียว TDA8444 จะทำการแปลงออกทาง เอ้าพุตตั้งแต่ช่องที่ระบุเป็นคำสั่ง และจะเพิ่มหมายเลขช่องอัตโนมัติ เช่น หากป้อนข้อมูลให้คำสั่งเป็น 0 และตำแหน่งของช่องเอาต์พุตเป็น 02H และข้อมูลที่ต้องการแปลงคือ 32 จะได้ผลคือ TDA8444 จะทำการแปลงข้อมูล 32 เป็นระดับแรงไฟที่ ช่อง 02 ไปจนถึง ช่อง 07

ส่วนคำสั่ง F จะมีผลคือการป้อนข้อมูลในแต่ละชุด TDA8444 จะทำการข้อมูลดังกล่าวให้ออกที่ช่องเอาต์พุตเฉพาะที่ระบุมา ตามคำสั่งเพียงช่องเดียวเท่านั้น เช่น เมื่อป้อนขบวนคำสั่งดังต่อไปนี้ คำสั่งคือ F และตำแหน่งช่องเป็น 07 ข้อมูลคือ 20 ผลที่ได้คือ TDA8444 จะทำการแปลงข้อมูล 20 เป็นระดับออกที่ช่อง 07 เพียงช่องเดียว

3.5 วงจรควบคุมความสว่าง

ทำหน้าที่จำกัดกระแสและแรงดันซึ่งจะแปรผันตามค่าแรงดันควบคุมที่ส่งมาจากวงจรแปลงสัญญาณดิจิตอลเป็นแอนาลอก อุปกรณ์หลักของวงจรนี้คือ วงจรรวมเบอร์ TCA785 ซึ่งเป็นไอซีที่ใช้สำรับงานควบคุมเฟสโดยเฉพาะซึ่ง การทำงานจะอาศัยหลักการของการทริกขาเกตของเอสซีอาร์ให้นำกระแสที่เฟสต่างๆ ของคลื่นแรงดันไฟสลับที่จ่ายให้กับหลอดไฟ

บล็อกไดอะแกรมการเชื่อมต่อภายในของชุดเพาเวอร์แพกแสดงดังรูปที่ 40

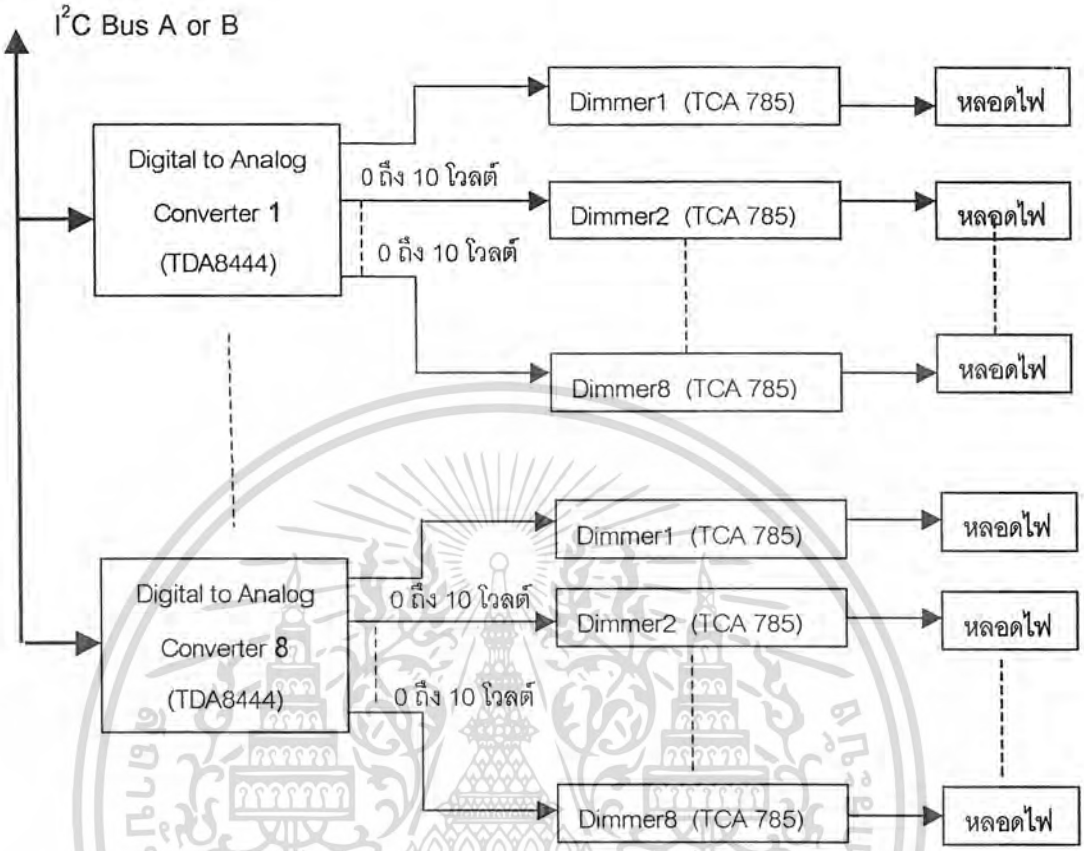
3.6 การนำไอซีสำเร็จรูปมาใช้ในส่วนของวงจรควบคุม

ในส่วนของวงจรควบคุมนั้นก็ คือ วงจรที่ใช้ในการสร้างสัญญาณจุดชนวนให้แก่ ไทริสเตอร์ เป็นวงจรที่แยกออกมาจาก ส่วนของวงจรกำลัง วงจรควบคุมนี้อาจ สร้างมาจาก วงจร Transistor หรือจาก OP-AMP ในส่วนของ ชุดควบคุมความสว่าง ในโครงงานนี้จะใช้ IC ที่มีหน้าที่ในการควบคุมเฟสโดยเฉพาะ คือ IC เบอร์ TCA 785 ซึ่ง TCA 785 นี้เป็นอุปกรณ์หลักในการสร้างสัญญาณจุดชนวนให้กับไทริสเตอร์

3.6.1 คุณสมบัติของ TCA 785

เป็นวงจรรวมที่ผลิตขึ้นใช้ในงาน การควบคุมเฟสโดยเฉพาะ สามารถใช้ได้กับระบบไฟหนึ่ง เฟส และระบบไฟ สาม เฟสสามารถปรับมุมจุดชนวนเกต ได้ตั้งแต่ 0 องศา ถึง 180 องศา และยังสามารถสร้างลักษณะของพัลส์ได้หลายรูปแบบ ตามแต่ลักษณะของโหลด ใช้งานได้ดีในย่านความถี่ระหว่าง 10 HZ ถึง 500 Hz ต้องการไฟเลี้ยงกระแสตรงขนาด 8 โวลต์ ถึง 18

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 40 บล็อกไดอะแกรมการเชื่อมต่อภายในของชุดเพาเวอร์แพก

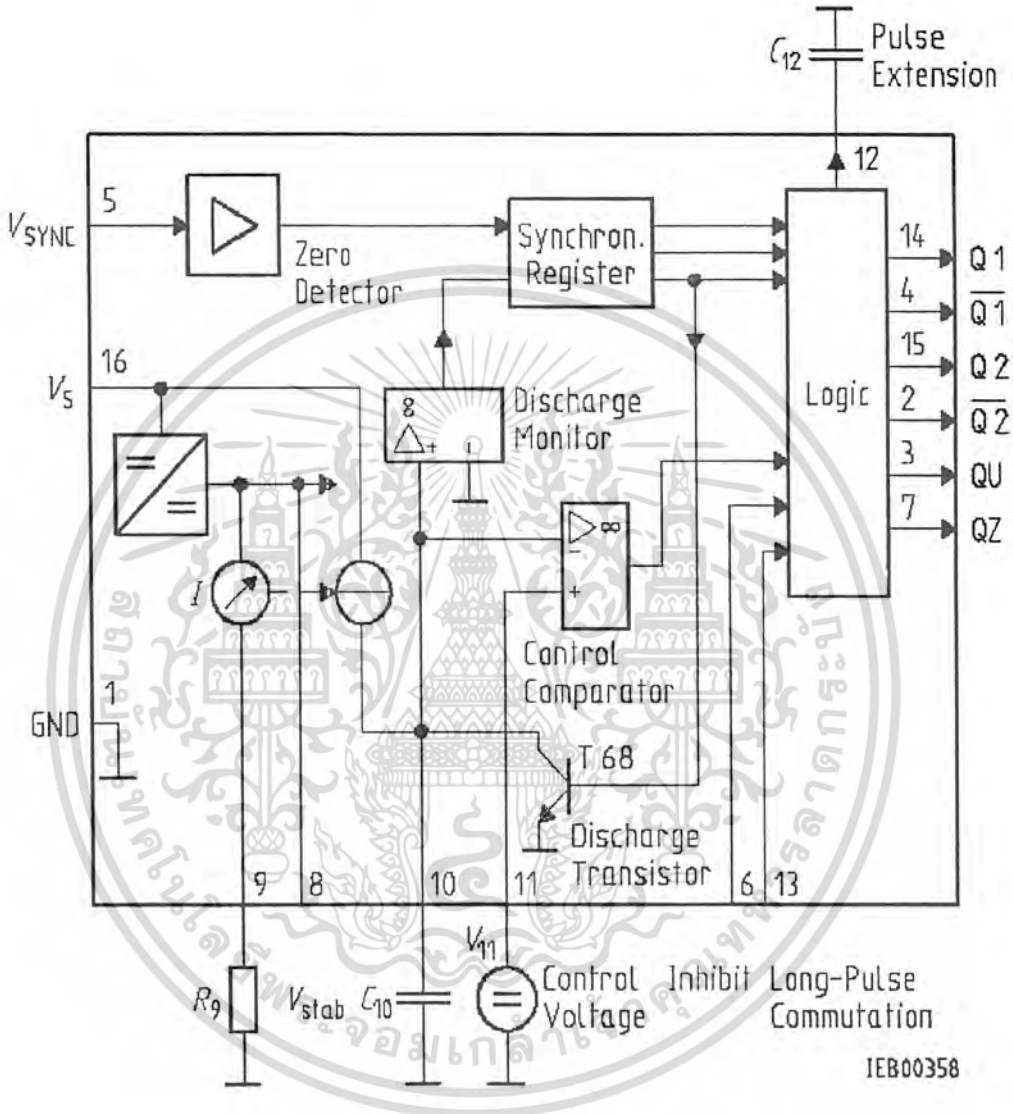
โวลต์ ใช้งานได้ในอุณหภูมิ –25องศาเซลเซียส ถึง +85องศาเซลเซียส กินกระแสไฟฟ้าประมาณ 4.5 mA. – 10mA อินเตอร์เฟสได้โดยตรงกับลอจิกเกตที่ใช้แรงดัน +15 โวลต์ (เช่น CMOS) สามารถจ่ายกระแสไฟฟ้าออกที่ขา 14 และ 15 ได้ประมาณ 250 mA. สัญญาณทางด้านออกมีทั้งแบบปกติ และ แบบกลับสัญญาณ มี Inhibit Function ควบคุมการกำเนิดสัญญาณ และสามารถใช้งานในลักษณะ การตรวจจับจุด0ได้อีก ด้วย

3.6.2 การทำงานของ TCA-785

สร้างภายในของ TCA 785 ซึ่งเป็นวงจรรวมแบบ LSI มี 16 ขา บรรจุแบบตัวถังพลาสติก ขา 16 คือ $+V_s$ รับแรงดันไบอัสในย่าน +8 โวลต์ ถึง +18 โวลต์ โดยขา 1 เป็นจุดกราวด์ แรงดันควบคุมภายในเป็นแรงดันอ้างอิง คือ V_{ref} ประมาณ 3.1V วัดได้ที่ขา 8 โดยตัว C_8 ทำหน้าที่ป้องกันสัญญาณรบกวน ขา 5 คือขาที่รับแรงดันซิงโครไนซ์ (V_{syn}) ซึ่งเป็นแรงดันไฟสลับจากแหล่งจ่ายไฟสลับที่ต่อกับวงจรภาคกำลัง โดยที่ต่อนิวตรอนกับขา 1 , R_9 คือตัวต้านทานที่ใช้ในการสร้าง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สัญญาณลาดเอียง และ C_{10} คือตัวเก็บประจุสัญญาณลาดเอียงต่อกับขา 9 และ 10 ตามลำดับ ค่าของ C_{10} มีค่าในช่วง 500 pf ถึง 1 uf และ R_9 ที่เหมาะสมในช่วง 3 กิโลโอห์ม ถึง 300 กิโลโอห์ม



รูปที่ 41 แสดงโครงสร้างภายในของวงจรรวมเบอร์ TCA785

ค่า R_9 และ C_{10} จะเป็นตัวกำหนดขนาดของสัญญาณลาดเอียง (V_{10}) ถ้า R_9 และ C_{10} มีค่ามากความลาดเอียงของ V_{10} จะมีค่ามากตามไปด้วย ขา 11 คือขาที่ต่อแรงดันควบคุมเป็นแรงดันไฟตรงที่สามารถปรับค่าได้แรงดันควบคุม (V_{11}) นี้จะถูกป้อนเข้าขาบวก(+) ของออปแอมป์เปรียบเทียบสัญญาณควบคุมในรูปที่ 39 โดยเทียบกับแรงดัน V_{10} เพื่อกำหนดขนาด ของมุมจุดชนวนทางด้านออกของวงจร ดังแสดงในรูปที่ 40 จะเห็นว่าสัญญาณทางด้านออกของวงจรอยู่ที่

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ขา 14 และ ขา 15 โดยพัลส์ที่ขา 15 จะทำงานในช่วงเวลา $\omega_t = 0$ องศา ถึง 180 องศา และ พัลส์ที่ขา 14 จะทำงานในช่วงเวลาที่ $\omega_t = 180$ องศา ถึง 360 องศา ขนาดความกว้างของพัลส์ทางด้านออกคือ β ปกติถ้าไม่ต่อ C_{12} จะมีค่า β = 30us แต่สามารถเปลี่ยนค่า β ได้โดยใช้ค่าC_{12}ต่อเข้าที่ขา12ดังแสดงความสัมพันธ์ของค่า C_{12} กับค่าของ β ในตารางที่ 8 สำหรับ ขา 2 และ ขา 4 คือ สัญญาณกลับของ V_{15} และ V_{14} สัญญาณลักษณะพัลส์เดี่ยวที่มีค่า β = 30 us จะเกิดได้โดยไม่ต้องทำการต่อ C_{12} เข้าที่ขา 12 แต่ถ้าต้องการ พัลส์ทางด้านออกเป็นพัลส์ยาวสามารถทำได้โดยต่อขา12 กับจุดกราวด์และถ้าต้องการให้สัญญาณออก ที่ขา 2 และขา 4 เป็นพัลส์ยาวด้วยจะต้องต่อขา 13 เข้ากับจุดกราวด์ด้วย ดังรูปคลื่นที่แสดงในรูปที่ 42

C_{12} (pf)	0	150	220	330	680	1000
Pulse width (us)	30	93	137	205	422	620

ตารางที่ 8 แสดงความสัมพันธ์ของความกว้างของพัลส์ด้านออกกับค่า C_{12}

3.6.3 ลักษณะของพัลส์ด้านออกชนิดต่างๆ

การสร้างพัลส์ด้านออกให้มีรูปร่างแตกต่างกันนั้น ขึ้นอยู่กับชนิดของโหลดในวงจรที่ไทริสเตอร์นั้นต่อควบคุมโหลดอยู่ ลักษณะของพัลส์มีหลายรูปแบบดังรูปที่ 42 มีรายละเอียดดังต่อไปนี้

1) Short pulse (β = 30 – 100 us) สำหรับโหลดที่เป็นตัวต้านทาน

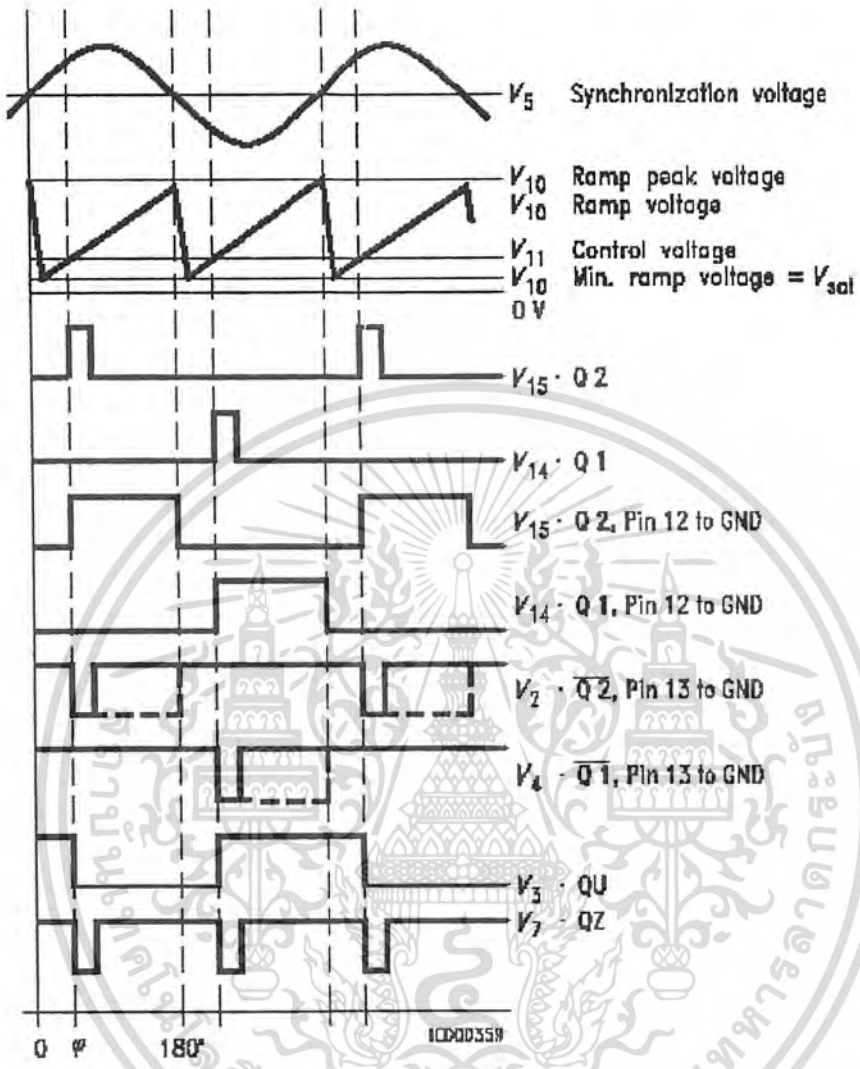
2) Long pulse (β = 100 us – 1 ms) สำหรับโหลดที่เป็นตัวต้านทานที่กินกระแสสูงๆ หรือโหลดที่เป็นตัวเหนี่ยวนำ

3) Continuous pulse (β = 180องศา – alpha) สำหรับโหลดที่เป็นตัวเหนี่ยวนำที่มีค่าเหนี่ยวนำ (L) สูงมากๆ

4) Combined pulse สำหรับวงจรที่มีการเปลี่ยนแปลงของกระแสเกต ($\frac{di}{dt}$ = 1 จนถึง 3 A / uS) หรือใช้สำหรับจุดชนวน ไทริสเตอร์ที่มีการต่อแบบอนุกรมกัน

5) Double pulse ใช้สำหรับวงจรจุดชนวนไทริสเตอร์ในวงจร 3 เฟสแบบบริดจ์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 42 รูปคลื่นสัญญาณด้านเข้าและด้านออกที่สำคัญของวงจรรวมเบอร์ TCA-785

3.6.4 หลักการทำงานของชุดควบคุมความสว่าง

ดังที่ได้กล่าวมาแล้วว่าชุดควบคุมความสว่างนั้น ประกอบด้วย วงจรกำลัง และวงจรควบคุม จากหลักการที่กล่าวถึงมาแล้วในตอนต้น เมื่อนำมาออกแบบวงจรควบคุมความสว่างจึงได้มาเป็นวงจรดังรูปที่ 43 และรูปที่ 44 จากวงจรในรูปที่ 43 สัญญาณซิงค์จากหม้อแปลงทางด้านทุติยภูมิจะผ่าน R_2 เพื่อทำการลดขนาดของสัญญาณเอาต์พุตที่จะนำมาใช้จะได้มาจากขา 14 (Q_1) และ ขา 15 (Q_2) ของวงจรรวมเบอร์ TCA 785 ทั้งสองขานี้จะทำงานสลับกัน คนละครึ่งรอบของสัญญาณไซน์ โดยต่อผ่าน ไดโอด D_3 และ D_4 เพื่อป้องกันไม่ให้แรงดันที่อาจไหลย้อนกลับมาจาก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

อีกเอาต์พุตหนึ่ง ขนาดของแรงดันที่จุดนี้จะถูกจำกัดด้วย R_4 ก่อนที่จะเข้า LED ของออปโตคัปเปลอร์ IC_3 และ IC_4 เบอร์ 4N40 โดยกำหนดให้มีกระแสไหลผ่าน LED ไม่เกิน15mA ตัวต้านทานปรับค่าได้ VR_1 จะทำหน้าที่ในการปรับความลาดชันของสัญญาณลาดเอียง(Ramp Signal)ภายใน TCA 785 ซึ่งจะมีผลต่อการปรับมุมทริกเอสซีอาร์ให้เป็นไปได้เร็วหรือช้า ในขณะที่ทำการปรับแรงดันควบคุมที่ขา 11 ของ TCA785 โดยแรงดันที่ป้อนให้ขานี้ได้มาจากวงจรแปลงสัญญาณดิจิตอลเป็นแอนาลอกโดยวงจรรวมเบอร์ TDA8444 ออปโตคัปเปลอร์เบอร์ 4N40 นั้นทำหน้าที่ในการแยกไฟแรงดันต่ำออกจากไฟแรงดันสูง ด้วยการเชื่อมโยงทางแสงกับโฟโต้SCR โดยจัดให้มีขดลวดทุติยภูมิของหม้อแปลงได้แรงดันไฟสลับ 12 โวลต์ จำนวนสองชุด จากหม้อแปลงตัวเดียวกันกับที่ใช้เป็นแหล่งจ่ายญญาณซิงค์ให้กับ TCA785 ขดลวดทุติยภูมิทั้งสองชุดนี้นั้นจะทำหน้าที่เป็นแหล่งจ่ายไฟให้สัญญาณทริกทั้งรอบบวก และรอบลบของสัญญาณไซน์ ลักษณะของการต่อขดลวดทุติยภูมิทั้งสองชุดนั้นจะมีหลักเกณฑ์ในการต่อโดยเฉพาะซึ่งจะขอกล่าวถึงในหัวข้อถัดไป ในส่วนของวงจรทางด้านไฟแรงดันสูงนั้นจะมีค่าของตัวต้านทานที่สามารถปรับเปลี่ยนได้ ให้สอดคล้องกับคุณสมบัติของเอสซีอาร์ แต่ละตัวได้นั่นก็คือ RG ซึ่งตัวต้านทาน RG นี้จะทำหน้าที่ในการจำกัดกระแสเกตที่จะป้อนให้กับเอสซีอาร์ สำหรับซีเนอร์ไดโอด ZD_1 และ ZD_2 นั้นใช้ในการรักษาะดับแรงดันให้มีความคงที่เพื่อไม่ให้เกินอัตราที่จะทนได้ ทางด้านขาเกตของ เอสซีอาร์

3.6.5 การหาขั้วของหม้อแปลง

โดยปกติทั่วไปแล้วนั้นหม้อแปลงเป็นอุปกรณ์ที่ใช้กับไฟฟ้ากระแสสลับไม่จำเป็นที่จะต้องคำนึงถึงขั้ว แต่สำหรับในวงจรของชุดควบคุมความสว่างนี้นั้น มีความจำเป็นที่จะต้องทำการหาขั้วของหม้อแปลง เนื่องมาจากได้มีการจัดวงจรการทริก SCR แบบตัวต่อตัวเพื่อที่จะทำการลดกำลังงานที่จะสูญเสียไปที่ขาเกต ดังนั้นจึงต้องมีการกำหนดขั้วให้กับหม้อแปลง จากวงจรในรูปที่ 4.4 ที่สัญลักษณ์ของหม้อแปลงจะมีจุดสีดำปรากฏอยู่ ซึ่งหมายถึงการเปรียบเทียบกันแล้วว่าที่ขดลวดขดต่างๆมีความต่างเฟสกันอยู่ประมาณศูนย์องศา มีวิธีการขั้วของหม้อแปลงดังต่อไปนี้

1) ทดลองจ่ายแรงดันไฟฟ้ากระแสสลับเข้าที่ขดลวดปฐมภูมิ ในที่นี้คือ ขด 220V

2) นำขดลวดทุติยภูมิ ในที่นี้คือขด 12 โวลต์ ทั้งสองขดโดยนำขั้วใดขั้วหนึ่งของแต่ละขดมาต่อกันแล้วเอาโวลต์มิเตอร์วัดแรงดันที่ปลายที่เหลือของขดลวด ถ้าแรงดันที่วัดได้มีค่าประมาณ 24โวลต์ แสดงว่าปลายทั้งสองของขดลวดที่วัดอยู่นั้นมีขั้วต่างกัน แต่ถ้าค่าแรงดันที่วัดได้นั้นมีค่าประมาณศูนย์โวลต์ แสดงว่าปลายของขดลวดที่วัดอยู่นั้นมีขั้วที่เหมือนกัน ในขั้นตอนต่อไปต้องใช้ความระมัดระวังในการหาขั้วของหม้อแปลง เนื่องจากจะต้องทำการหาขั้วของขดลวดปฐมภูมิ ขดลวด 220 โวลต์ โดยใช้วิธีการเดียวกันกับการหาขั้วที่ขดลวดทุติยภูมิ คือวัดดูว่า

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ถ้าแรงดันที่วัดอยู่นั้นมีค่ามากกว่า 220 โวลต์ จะมีขั้วที่ต่างกัน แต่ถ้าแรงดันที่วัดได้มีค่าลดลงต่ำกว่า 220 โวลต์ จะมีขั้วเหมือนกัน เวลานำไปใช้งานให้ปลดจุดที่ต่อกันในตอนหาขั้วออกก่อน ไม่เช่นนั้นหม้อแปลงจะไม่แยกไฟแรงดันสูงออกจากไฟแรงดันต่ำ

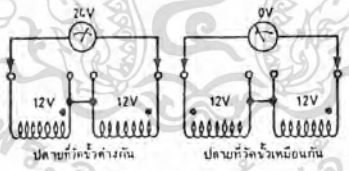
3.6.6 การหาค่า RG ที่ใช้ในการจำกัดกระแสเกต

เนื่องจากเอสซีอาร์ มีโครงสร้างที่ขาเกตเปราะบางฉะนั้นแล้วจึงต้องมีการระมัดระวังเป็นพิเศษ ในวงจรในรูปที่ 4.3 นั้น ไดโอด D_6 และ D_7 จะมีหน้าที่ในการป้องกันแรงดันย้อนกลับที่อาจจะหลงเข้ามาทำอันตรายกับขาเกตได้ด้วยสาเหตุใดก็ตาม D_6 และ D_7 จะทำการลัดวงจรที่ขาเกตทันที นอกจากนี้ยังมีอุปกรณ์ที่มีหน้าที่ในการป้องกันวงจรเกตนั่นก็คือ RG จะช่วยในการจำกัดกระแส และแรงดันให้มีขนาดที่เหมาะสม จากรูปที่ 4.5 นั้นเป็นบางส่วนของวงจรทริก ทำการเขียนให้ดูง่ายขึ้น เพื่อนำมาใช้ในการคำนวณด้วยรูปทางขวามือ สามารถแสดงสมการที่ใช้ในการคำนวณหาค่า RG ได้ดังนี้

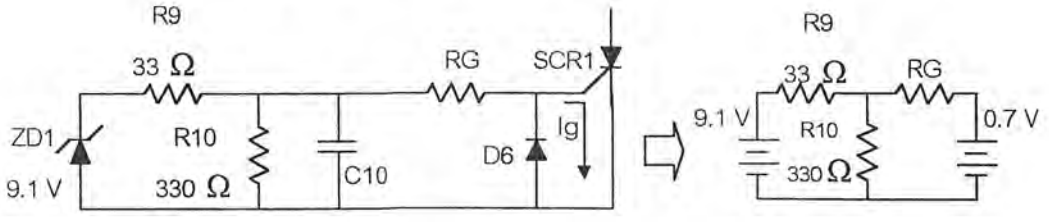
$$R_9 I + R_{10}(I - I_G) = 9.1$$

$$R_{10}(I_G - I) + R_G I_G = -0.7$$

ซึ่ง I_G สามารถหาค่าได้จากคู่มือของ SCR

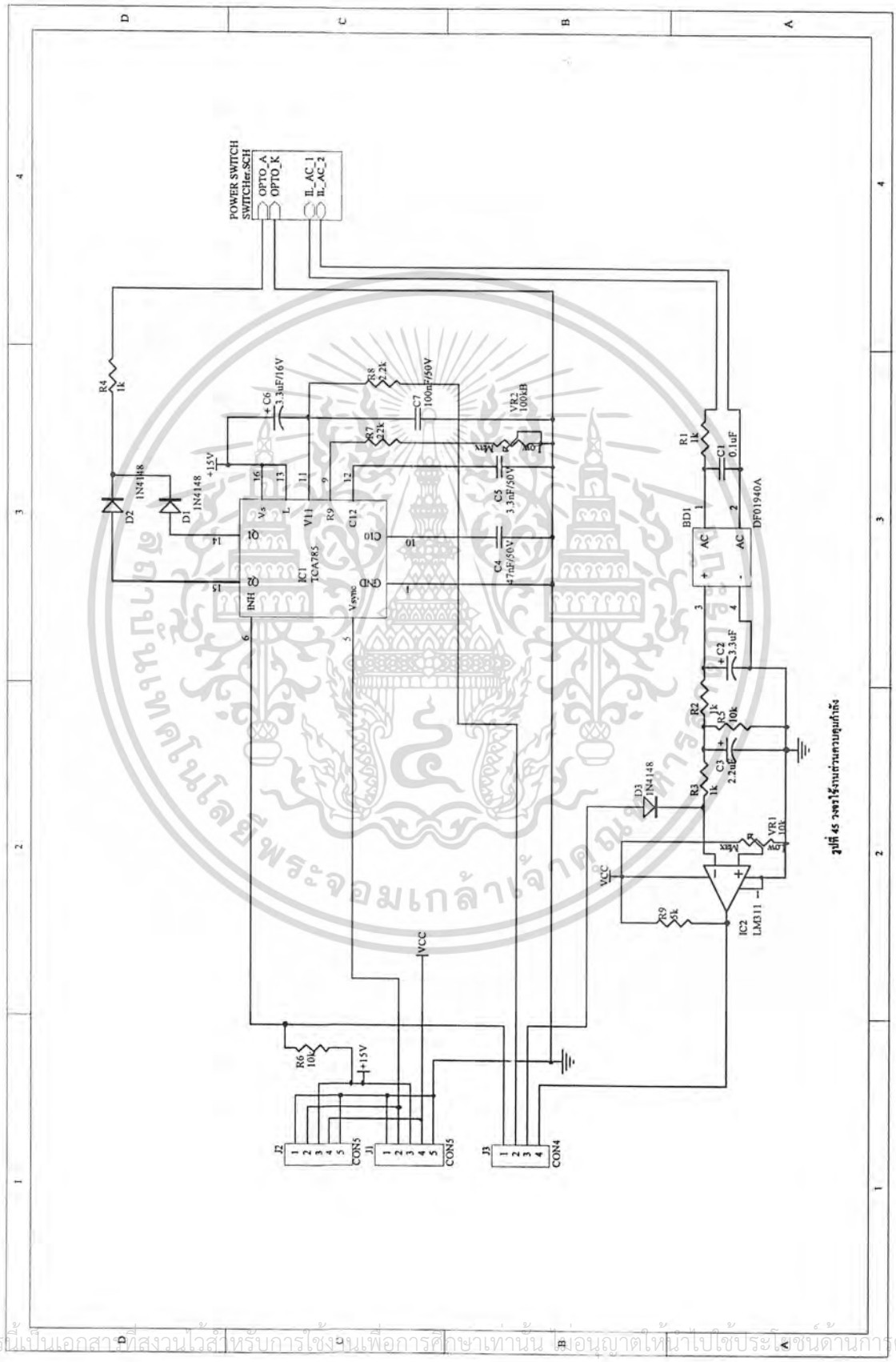


รูปที่ 43 แสดงวิธีการตรวจหาขั้วของหม้อแปลง



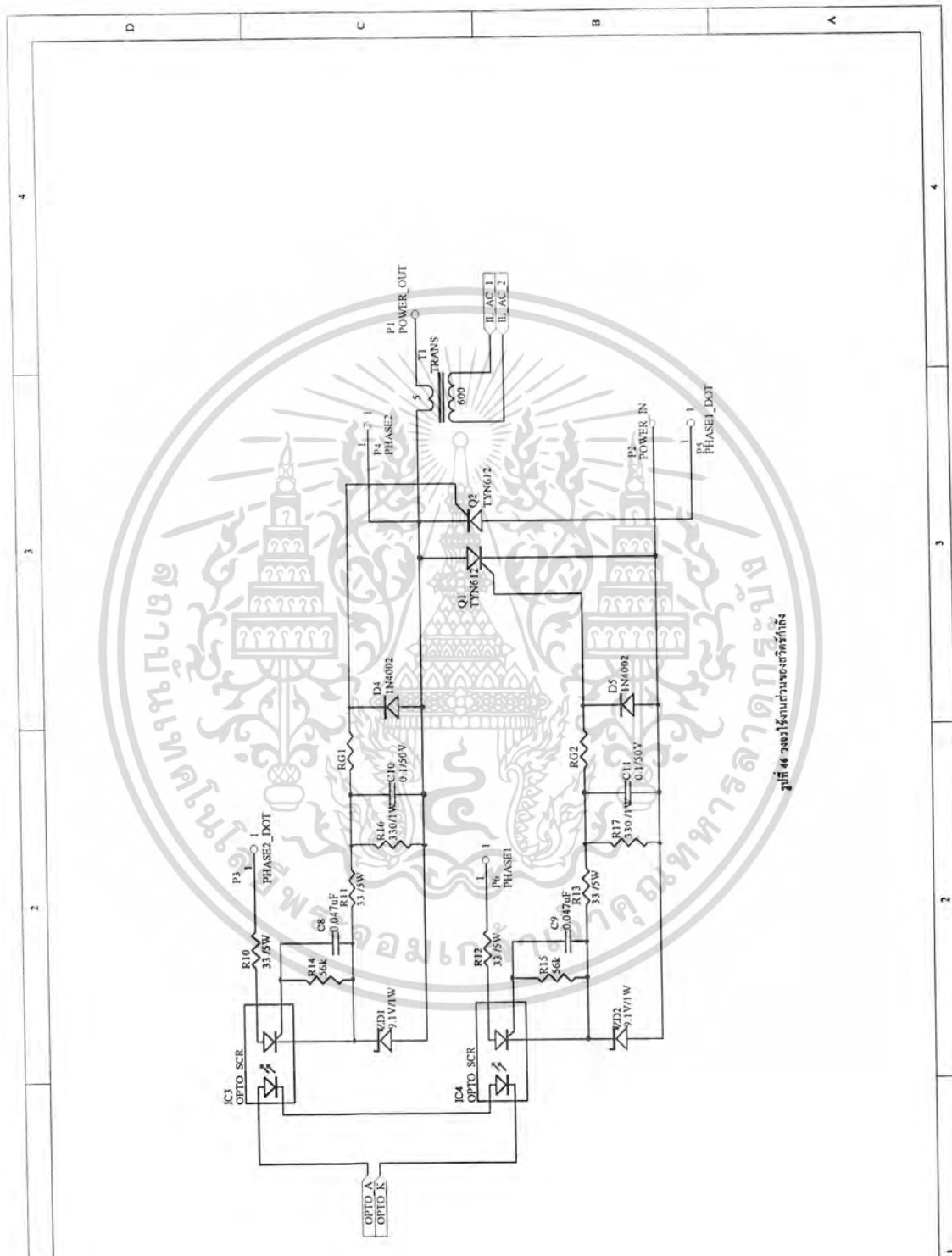
รูปที่ 44 แสดงการคำนวณหาค่า RG

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 45 วงจรไร้สายส่วนควบคุมกำลัง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 46 วงจรไร้งานส่วนของวัตรักลัง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.7 ซอฟต์แวร์ส่วนของ Main CPU

ซอพต์แวร์ในส่วนนี้จะเขียนด้วยภาษาแอสเซมบลี้จะมีซอฟต์แวร์อยู่ด้วยกันทั้งหมด 3 ชุด ด้วยกันคือ

1. ชุดของไมโครคอนโทรลเลอร์ จะทำหน้าที่ในการรับส่งข้อมูล ระหว่างคอมพิวเตอร์กับ ไมโครคอนโทรลเลอร์ เบอร์ AT89S53
2. ชุดของไมโครคอนโทรลเลอร์ เบอร์ AT89S53 จะเป็นส่วนของโปรแกรมที่ทำหน้าที่ในการจัดการกับหน่วยความจำ และการประมวลผลข้อมูลที่จะติดต่อกับ ภาค I^2C Bus Driver
3. ชุดของไมโครคอนโทรลเลอร์ เบอร์ AT89C2051 ที่จะทำหน้าที่ในการสร้างสัญญาณข้อมูลบน I^2C Bus

3.8 ซอฟต์แวร์ส่วนของหน้าจอควบคุมที่คอมพิวเตอร์

จะทำหน้าที่ในการรับข้อมูลจากผู้ใช้ แล้วนำมาทำการประมวลผล หลังจากนั้นจะทำการส่งข้อมูลออกไปโดยทำการส่งข้อมูลออกทางพอร์ตอนุกรมของคอมพิวเตอร์ เขียนด้วยโปแกรมวิชวลเบสิก

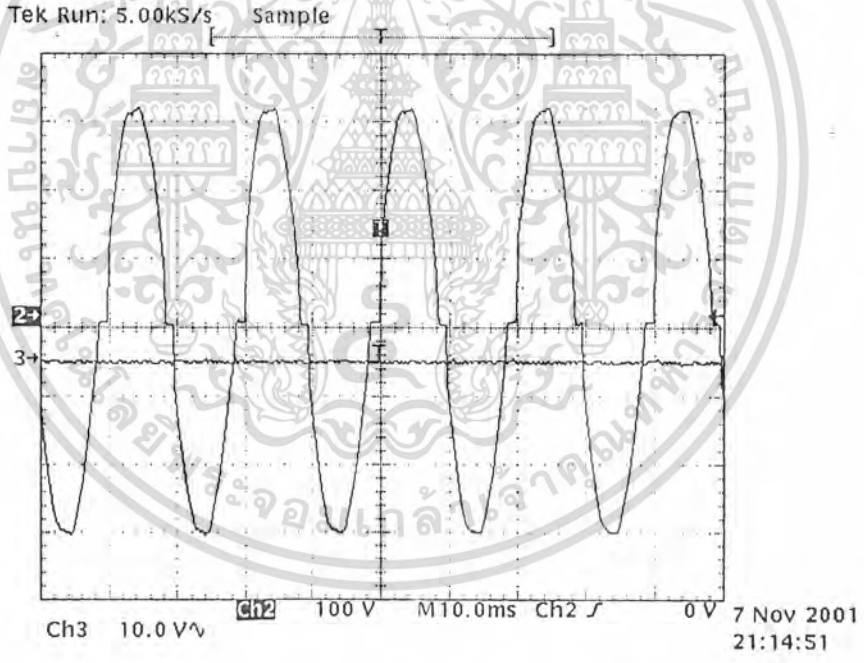
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 4

ผลการทดลอง

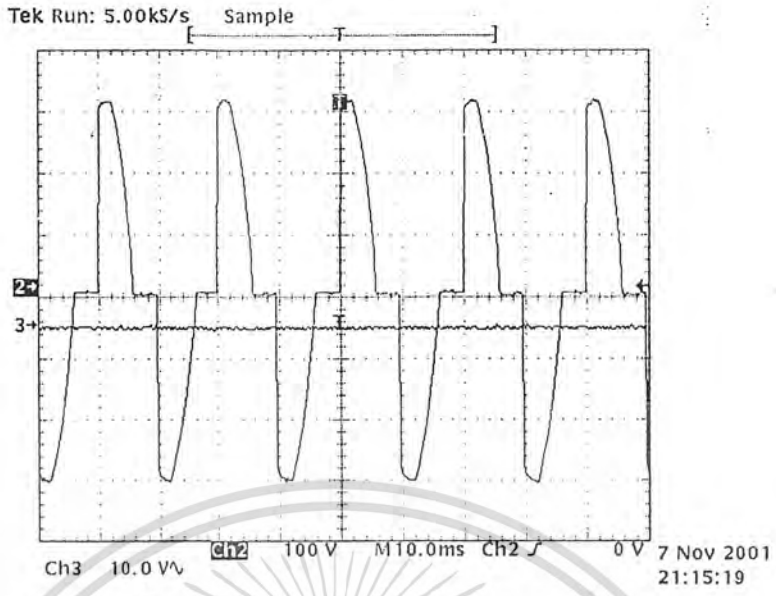
4.1 ผลการทดลองวงจรควบคุมความสว่าง

สามารถทำการควบคุมความสว่างของหลอดไฟได้โดย เปลี่ยนมุมจุดชนวนที่ กำหนด ให้กับไทริสเตอร์ การเปลี่ยนมุมจุดชนวนของไทริสเตอร์ทำได้โดยการปรับแรงดันควบคุมที่ขา 11 ของไอซี TCA785 ถ้าต้องการให้หลอดไฟมีความสว่างสูงสุด ให้ป้อนแรงดันควบคุมที่ขา 11 เท่ากับ ศูนย์โวลท์ แสดงว่าไทริสเตอร์ถูกจุดชนวนที่มุมศูนย์องศา ที่หลอดไฟจะได้รับแรงดันไฟฟ้าเต็มที่มีค่าเท่ากับ 220 โวลท์ ถ้าป้อนแรงดันควบคุมเท่ากับ 5 โวลท์ ไทริสเตอร์จะถูกจุดชนวนที่มุม 90 องศา หลอดไฟจะมีความสว่างลดลงเหลือเพียงครึ่งหนึ่ง หรือเหลือเพียง 50 เปอร์เซ็นต์ และถ้าต้องการให้หลอดไฟดับสนิท ให้ป้อนแรงดันควบคุมเท่ากับ 10 โวลท์

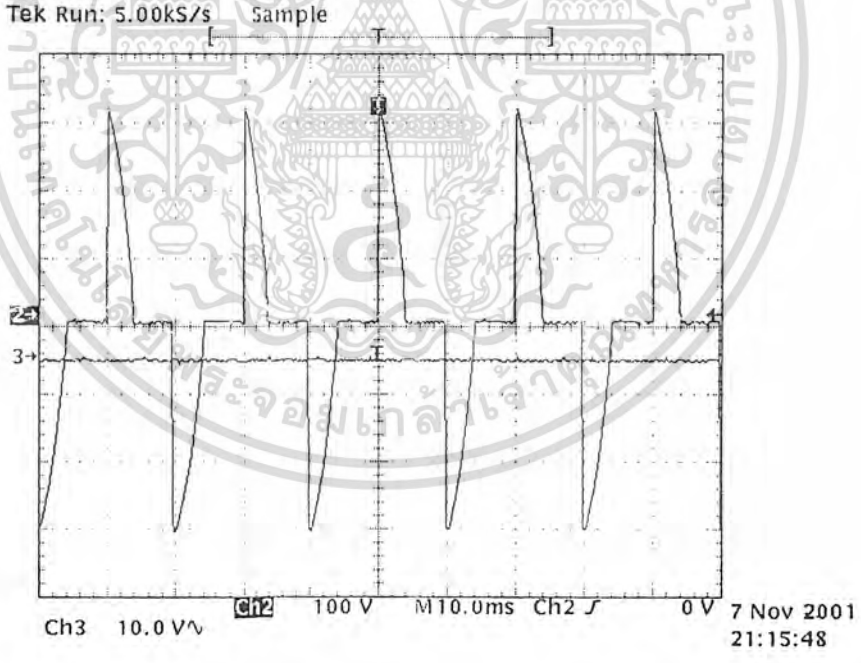


รูปที่ 47 รูปคลื่นแรงรันตกคร่อมโหลดเมื่อกระตุ้นเอสซีอาร์ที่มุมประมาณ 0 องศา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

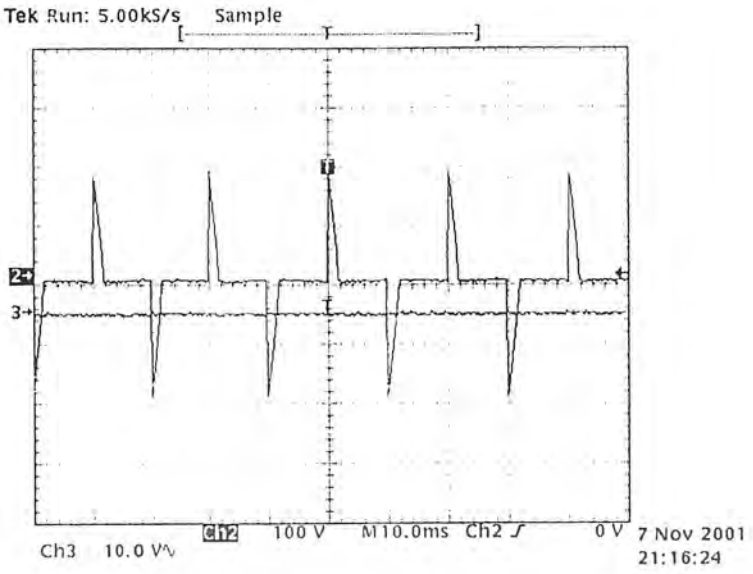


รูปที่ 48 รูปคลื่นแรงรันตกคร่อมโหลดเมื่อกระตุ้นเอสซีอาร์ที่มุมประมาณ 45 องศา

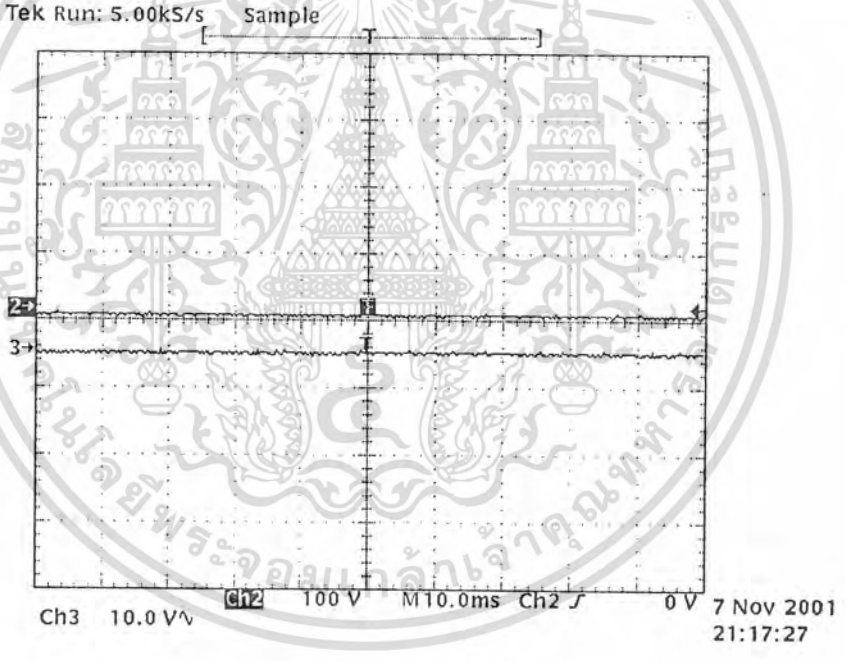


รูปที่ 49 รูปคลื่นแรงรันตกคร่อมโหลดเมื่อกระตุ้นเอสซีอาร์ที่มุมประมาณ 90 องศา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่50รูปคลื่นแรงรันตกคร่อมโหลดเมื่อกระตุ้นเอสซีอาร์ที่มุมประมาณ 135 องศา



รูปที่51รูปคลื่นแรงรันตกคร่อมโหลดเมื่อกระตุ้นเอสซีอาร์ที่มุมประมาณ 180 องศา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.2หน้าจอควบคุมที่คอมพิวเตอร์และการใช้งาน

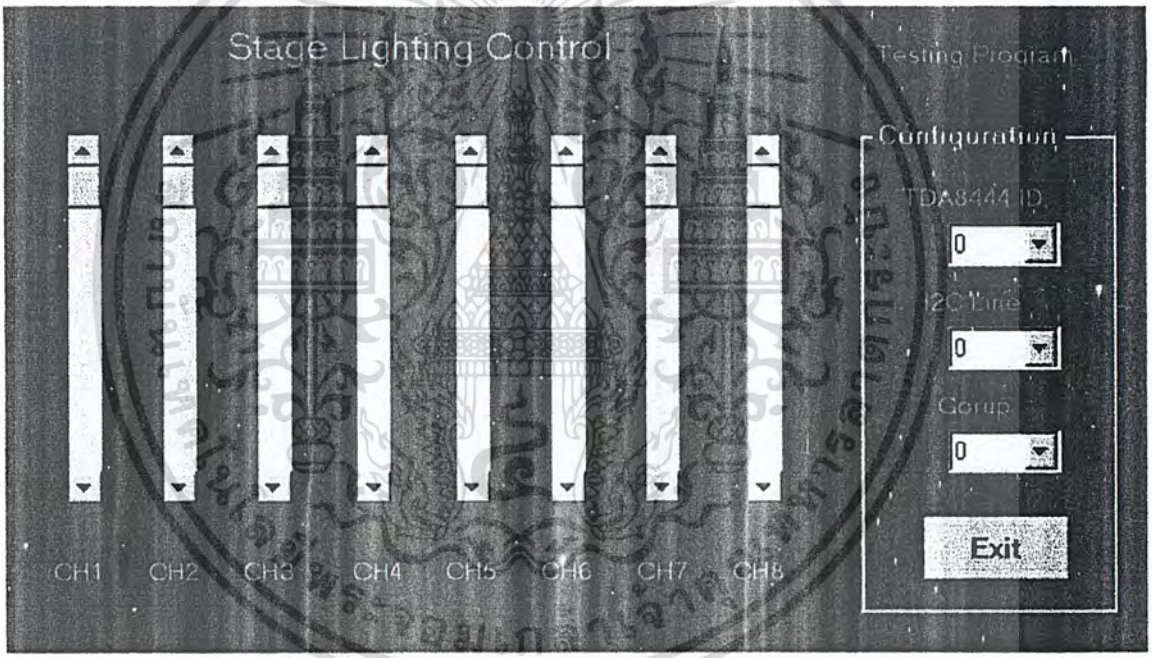
หน้าจอควบคุมที่คอมพิวเตอร์ประกอบด้วยส่วนต่างๆดังต่อไปนี้

1.แถบไสลด์มีจำนวนทั้งหมด 8 ช่อง ประกอบไปด้วย CH1 – CH8ใช้ในการควบคุมความสว่างโดยการเลื่อนขึ้นหรือลง

2.TDA8444 IDใช้สำหรับกำหนดให้หน้าจอนี้ควบคุมดิมเมอร์แพกชุดใด

3.I2C Lineใช้สำหรับกำหนดให้หน้าจอนี้เลือกใช้ช่องสัญญาณในการส่งข้อมูลไปยังดิมเมอร์แพกชุดใด

4.Groupใช้สำหรับกำหนดให้หน้าจอนี้ควบคุมดิมเมอร์แพกกลุ่มใด



รูปที่52 แสดงภาพหน้าจอควบคุมที่คอมพิวเตอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.2.1ขั้นตอนการใช้งานโปรแกรมควบคุม

1.ทำการติดตั้งโปรแกรม STL.exe โดย

1.1ถ้ามีโปรแกรมวิชวลเบสิก-6 สามารถทำการเปิดโปรแกรม STL.exe ได้ทันที

1.2ถ้าไม่มีโปรแกรมวิชวลเบสิก-6 ให้ทำการติดตั้ง ไฟล์mscomm32.ocxไว้ใน window และ system ที่อยู่ใน window

2.ต่อสายสัญญาณ RS-232 จากคอมพิวเตอร์เข้ากับวงจรคอนเวอร์เตอร์

3.ต่อสายสัญญาณจากวงจรคอนเวอร์เตอร์เข้ากับชุดคอนโทรลยูนิต

4.ต่อสายสัญญาณจากชุดคอนโทรลยูนิต เข้ากับชุดดิมเมอร์แพก

5.ปรับดิพสวิตช์ของชุดดิมเมอร์แพกไปที่ตำแหน่ง0000

6.เปิดโปรแกรม STL.exe

7.กำหนดให้ TDA8444 ID ไว้ที่ตำแหน่ง 0

8.กำหนดให้ I2C Line และ Group ไว้ที่ 0

9.ทำการต่อโหลดเข้ากับชุดดิมเมอร์แพก

10.เปิดสวิตช์ของวงจรคอนเวอร์เตอร์ คอนโทรลยูนิตและดิมเมอร์แพก พร้อมใช้งาน

4.3 สรุป

จากการทำงานของโครงงานสามารถทำงานได้อย่างถูกต้องตามคำสั่ง แต่ถ้าต้องการให้มีการส่งคำสั่งด้วยความเร็วที่สูงขึ้น เพื่อต้องการให้สามารถที่จะทำการควบคุมความสว่างเป็นไปอย่างรวดเร็ว ไม่สามารถทำได้เนื่องมาจากสาเหตุที่เกี่ยวกับช่วงเวลาในการแปลงข้อมูลของวงจรแปลงสัญญาณดิจิตอลเป็นอนาล็อกยังมีความเร็วต่ำอยู่ อีกส่วนหนึ่งเป็นผลมาจากความเร็วในการทำงานของซีพียูหลักยังถือว่ามีความเร็วต่ำอยู่ ดังนั้นการที่จะทำให้การควบคุมความสว่างเป็นไปอย่างรวดเร็ว รับข้อมูลได้อย่างต่อเนื่อง จะต้องทำให้ซีพียูมีความเร็วในการทำงานเพิ่มมากขึ้น หรืออาจเพิ่มจำนวนซีพียูหลักให้มีจำนวนมากขึ้นเพื่อทำการแบ่งหน้าที่การทำงาน

4.4 แนวทางการพัฒนา

1.ออกแบบซอฟต์แวร์ให้มีความรัดกุมเพิ่มมากขึ้น

2.เพิ่มจำนวนของซีพียู หรือเพิ่มความเร็วในการทำงานของซีพียู

3.หากต้องการให้ชุดควบคุมความสว่างสามารถจ่ายกำลังงานได้มากขึ้นให้ทำการเปลี่ยนเอสซีอาร์ให้มีพิกัดกระแสทำงานที่มากขึ้น

4.ในส่วนของวงจรตรวจสอบกระแสเกินหากต้องการให้ตรวจสอบกระแสเกินมากว่า 10แอมแปร์ให้เปลี่ยนเคอร์เรนท์ทรานส์ฟอร์เมอร์ให้มีพิกัดกระแสที่สูงขึ้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5.อาจเพิ่มวงจรแปลงสัญญาณดิจิตอลเป็นอนาล็อกให้วงจรตรวจสอบกระแสเกิน เพื่อให้สามารถกำหนดแรงดันอ้างอิงได้โดยตรงจากซอฟต์แวร์

6.ทำการปรับปรุงออกแบบให้เคอร์เรนท์ทรานส์ฟอร์เมอร์ให้มีความเป็นเชิงเส้นมากขึ้นโดยอาจเพิ่มจำนวนรอบของขดลวดทุติยภูมิให้มากขึ้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

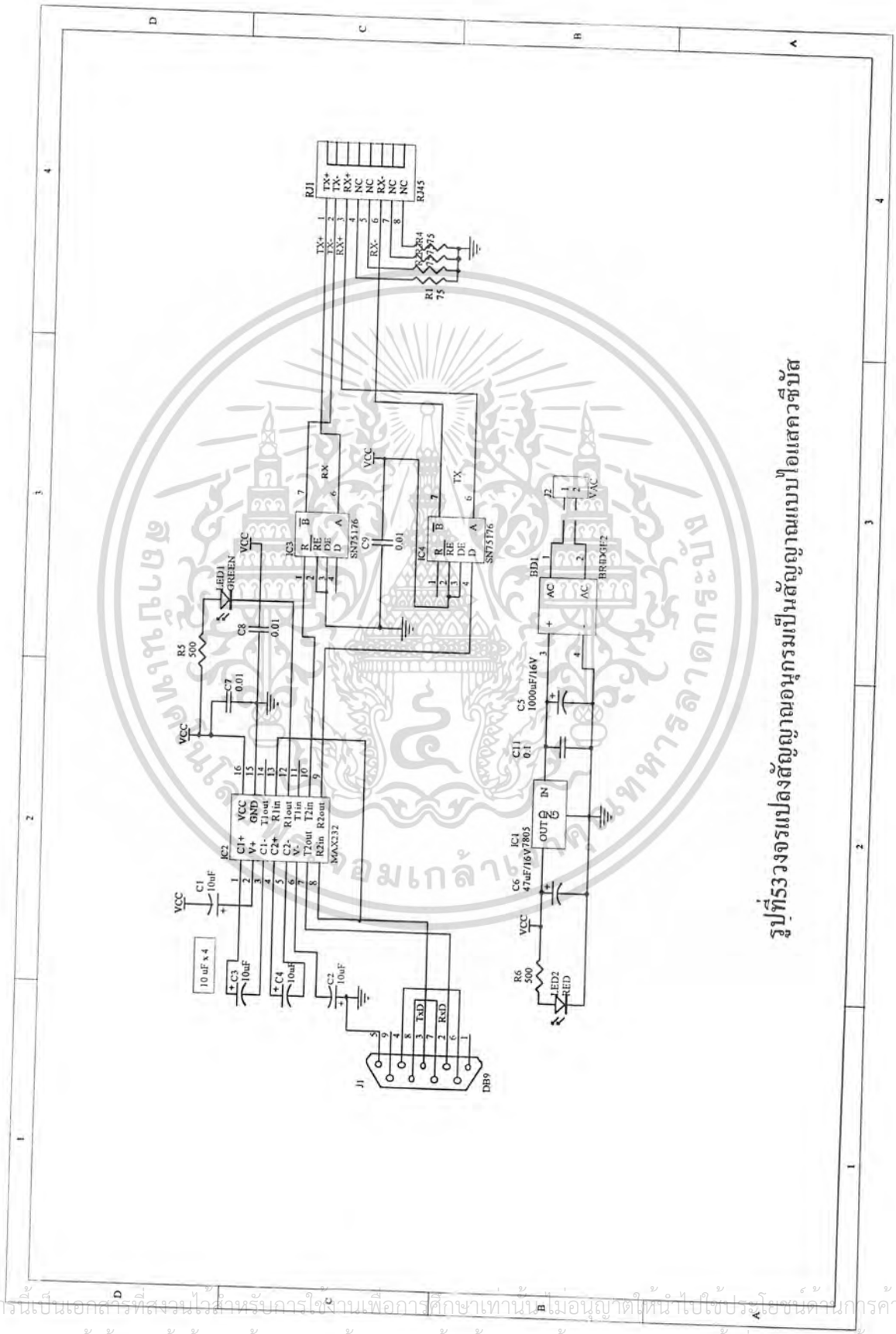
บรรณานุกรม

1. Dominiquc Paret , Carl Fanger ,**The I^2C Bus From Theory to Practice** , Philips Semiconductor , Zurich Switzerland : JOHN WILEY & SONS
2. Frederick F. Driscoll , **Data Communication** : Harcourt Brace Jovanovich Publisher
3. Data Handbook IC12 2000 , **Display Drivers and I^2C – bus Peripherals** : Philips
4. กฤษดา ใจเย็น , อรรถพล บุญยะโภคา , ชัยวัฒน์ ลิ้มพรจิตรวิไล , **เรียนรู้และปฏิบัติการเชื่อมต่อคอมพิวเตอร์กับอุปกรณ์ภายนอก ,กรุงเทพ** : บริษัท อินโนเวตีฟ เอ็กเพอริเมนต์ จำกัด
5. **คู่มือไอซี ไมโครคอนโทรลเลอร์ MCS – 51** : บริษัท อีทีที จำกัด
6. ชัยยงค์ แก้วมงคล , นภัทร วัจนเทพินทร์ , **อิเล็กทรอนิกส์กำลัง 1 (วงจรคอนเวอร์เตอร์)** , กรุงเทพ : บริษัท สกายบุคส์ จำกัด , พิมพ์ครั้งที่ 1 พฤษภาคม 2540
7. อุดมศักดิ์ ยั่งยืน , **POWER ELECTRONICS I** , กรุงเทพ : ภาควิชาวิศวกรรมไฟฟ้า คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าธนบุรี , เมษายน 2528
8. **วารสารเซมิคอนดัคเตอร์** ฉบับที่ 104 มกราคม 2534 ,หน้าที่ 200 – 210

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

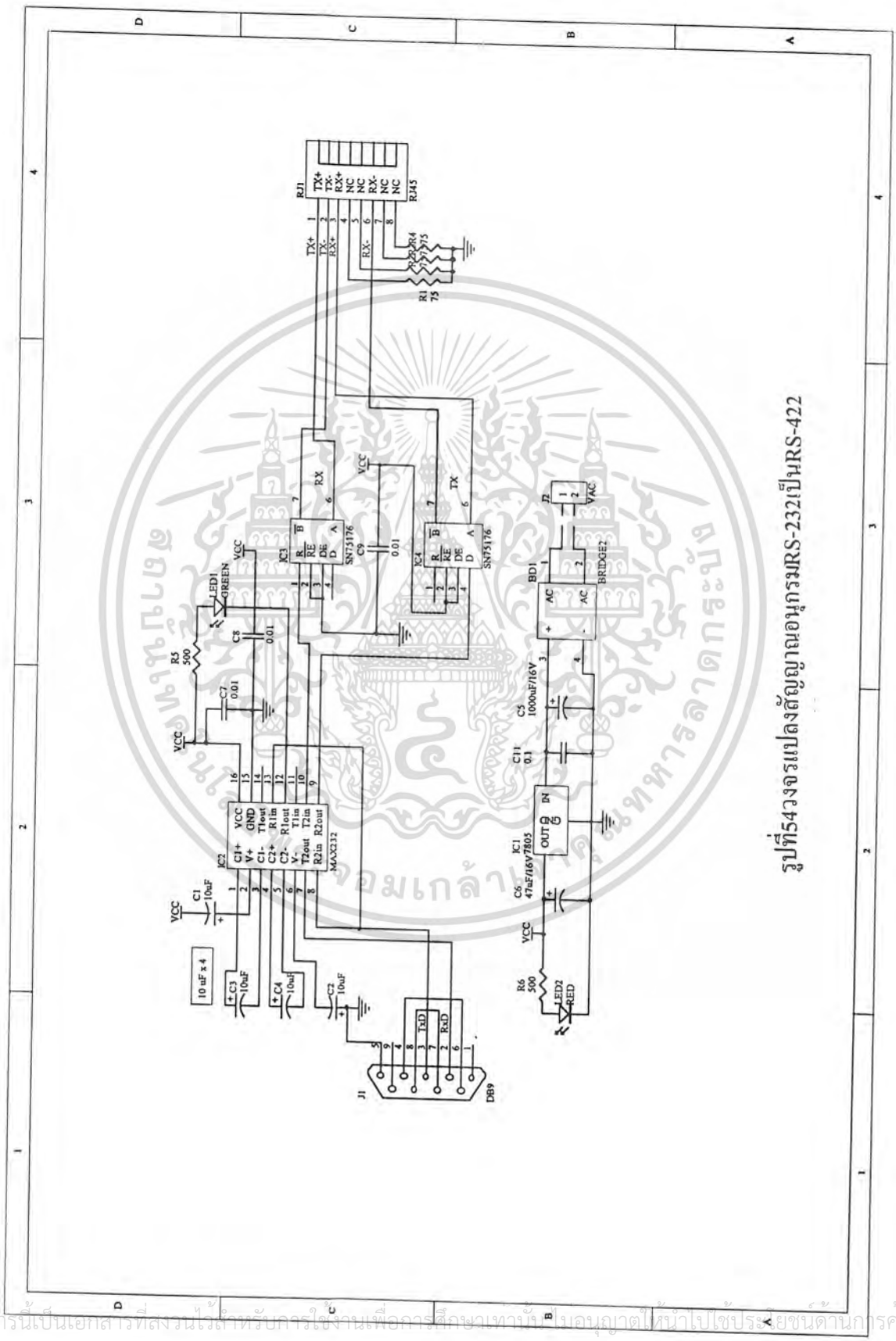
ภาคผนวก ก.
วงจรใช้งานทั้งหมดของเครื่องควบคุมไฟเวที

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



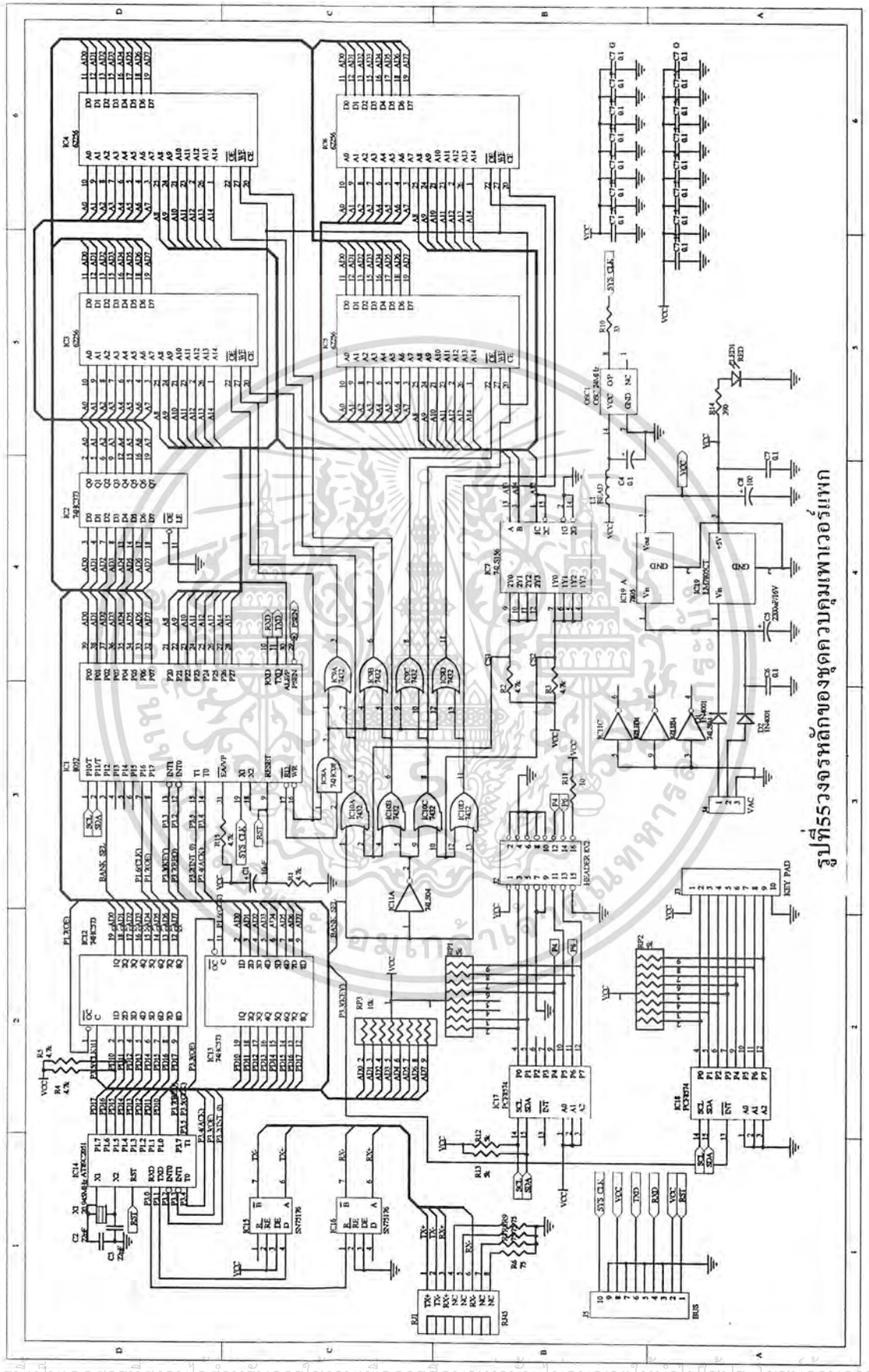
รูปที่ วงจรแปลงสัญญาณอนุกรมเป็นสัญญาณแบบไอแสควชัน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



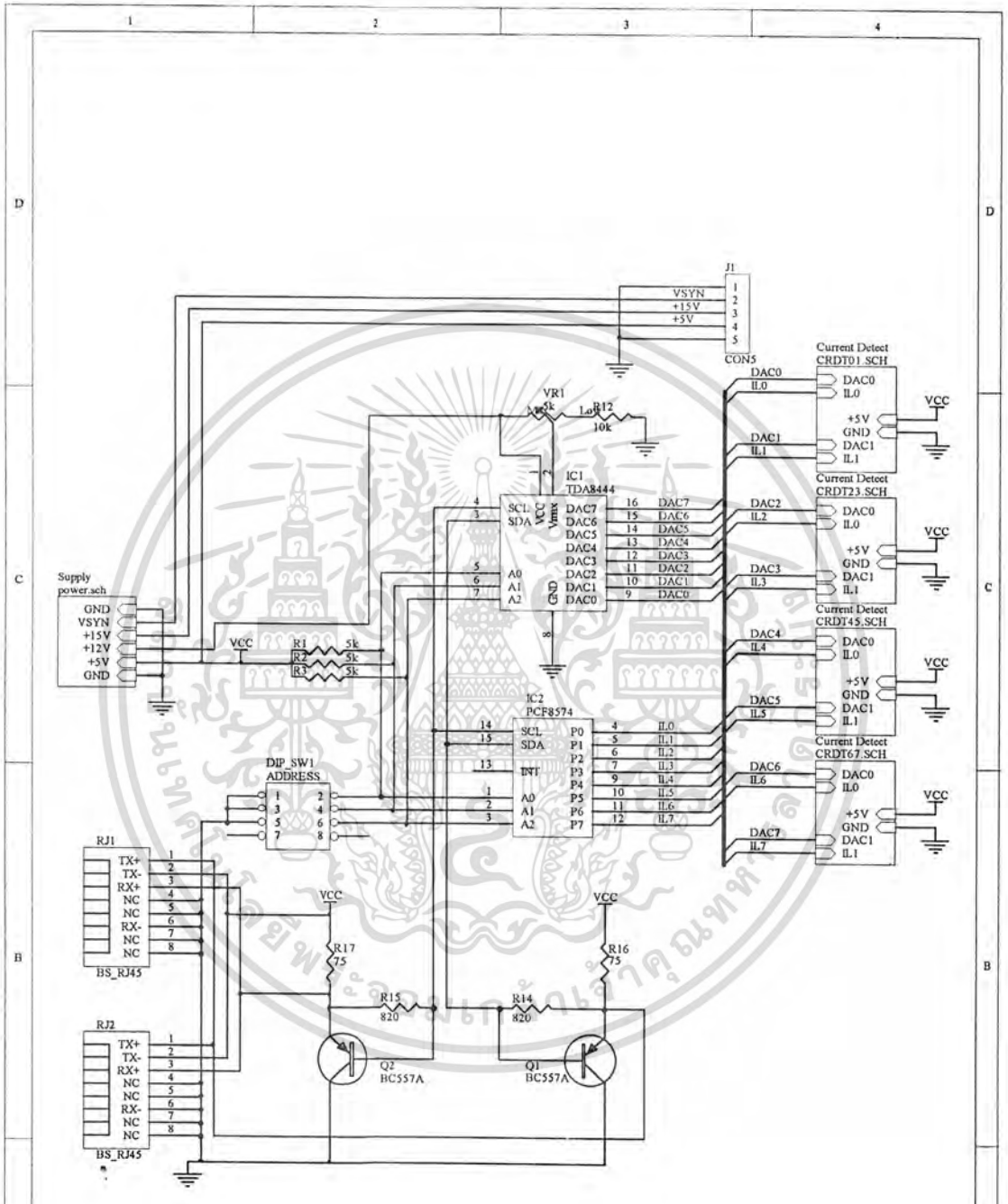
รูปที่ 54 วงจรแปลงสัญญาณอนุกรม RS-232 เป็น RS-422

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



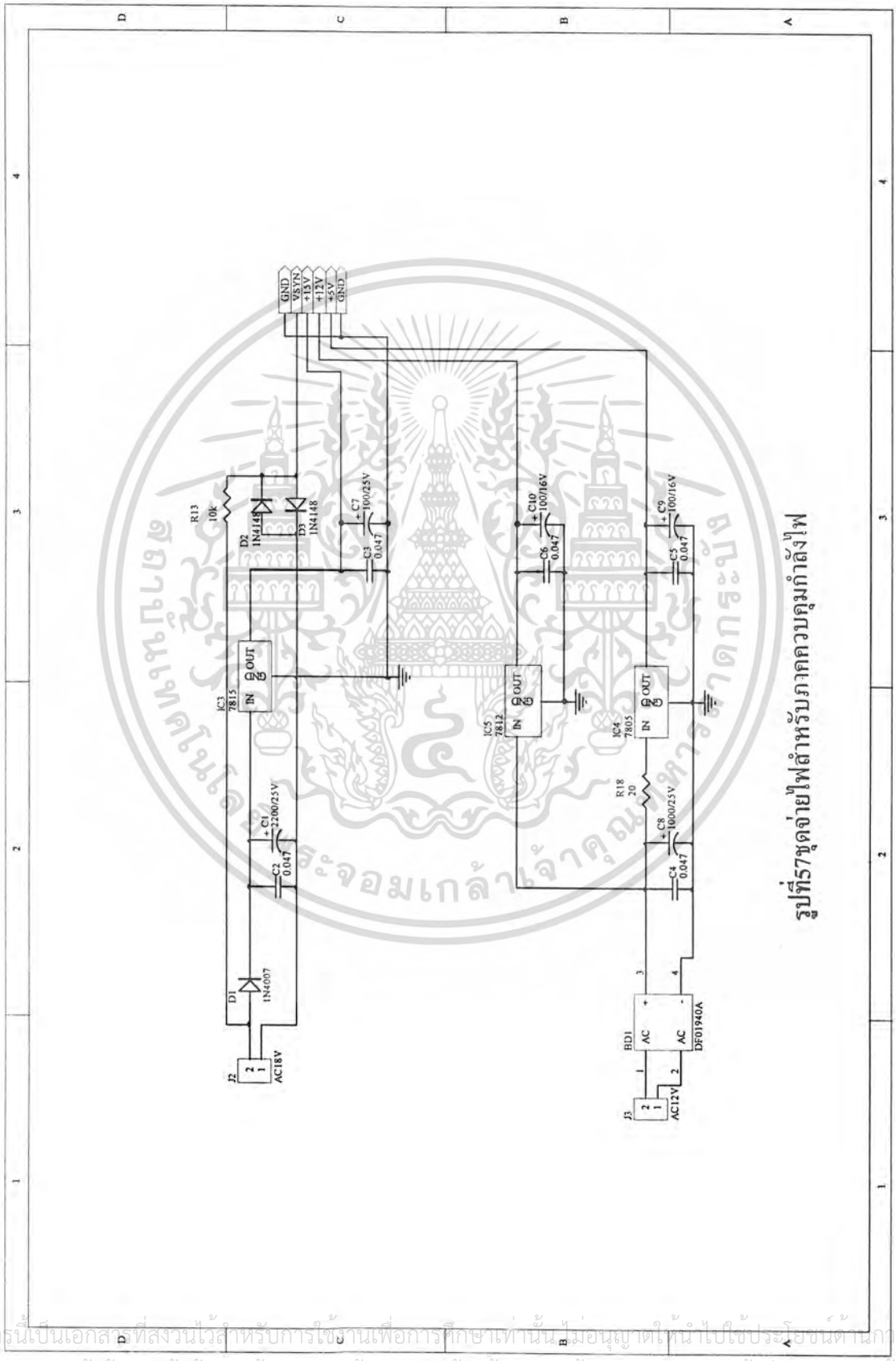
รูปที่ 55 วงจรหลักของชุดควบคุมพาเวอร์แมก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



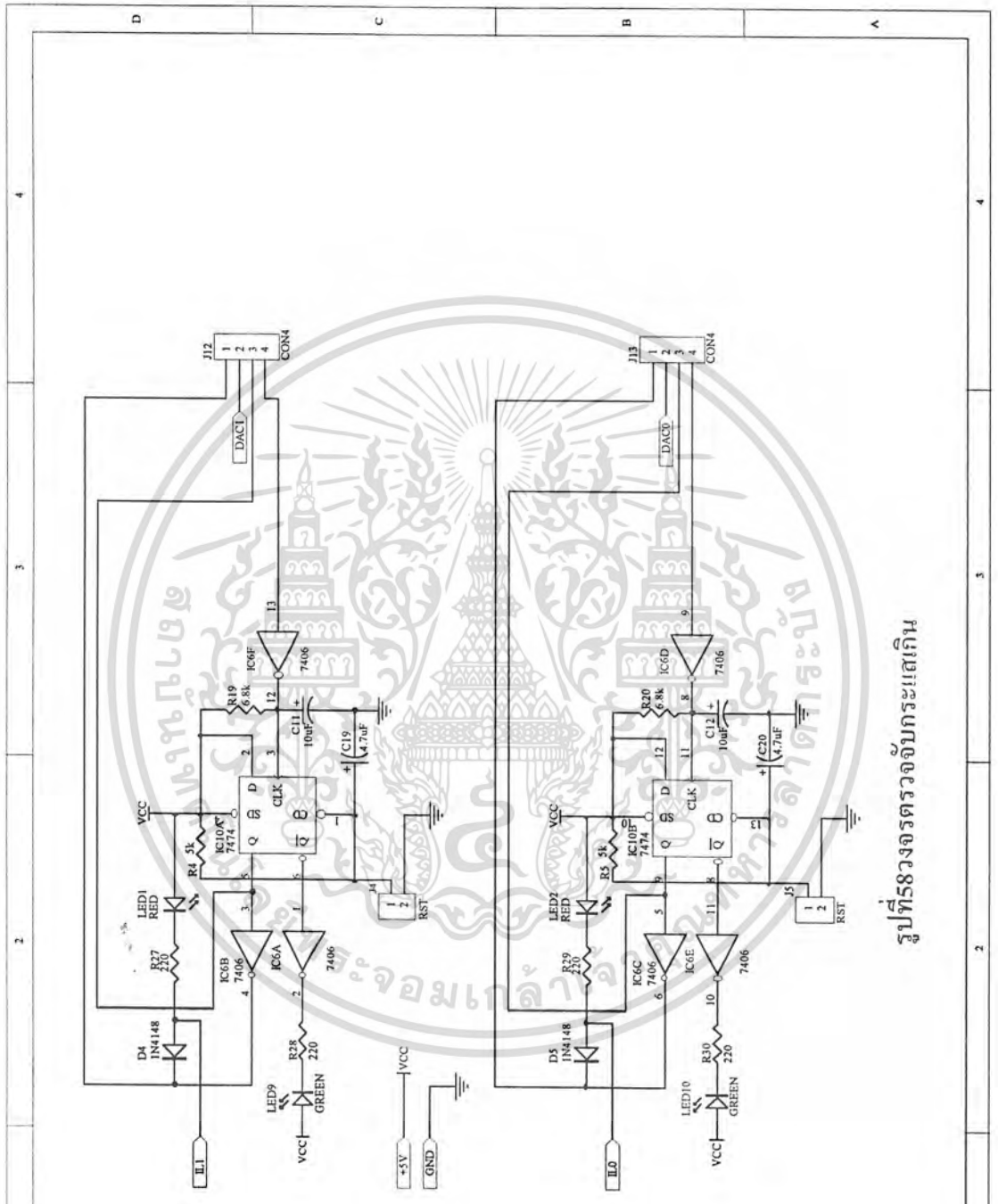
รูปที่56วงจรชุดรับข้อมูลควบคุมไฟและตรวจสอบกระแส

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่57ชุดจ่ายไฟสำหรับภาคควบคุมกำลังไฟ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่5วงจรตรวจจับกระแสเกิน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

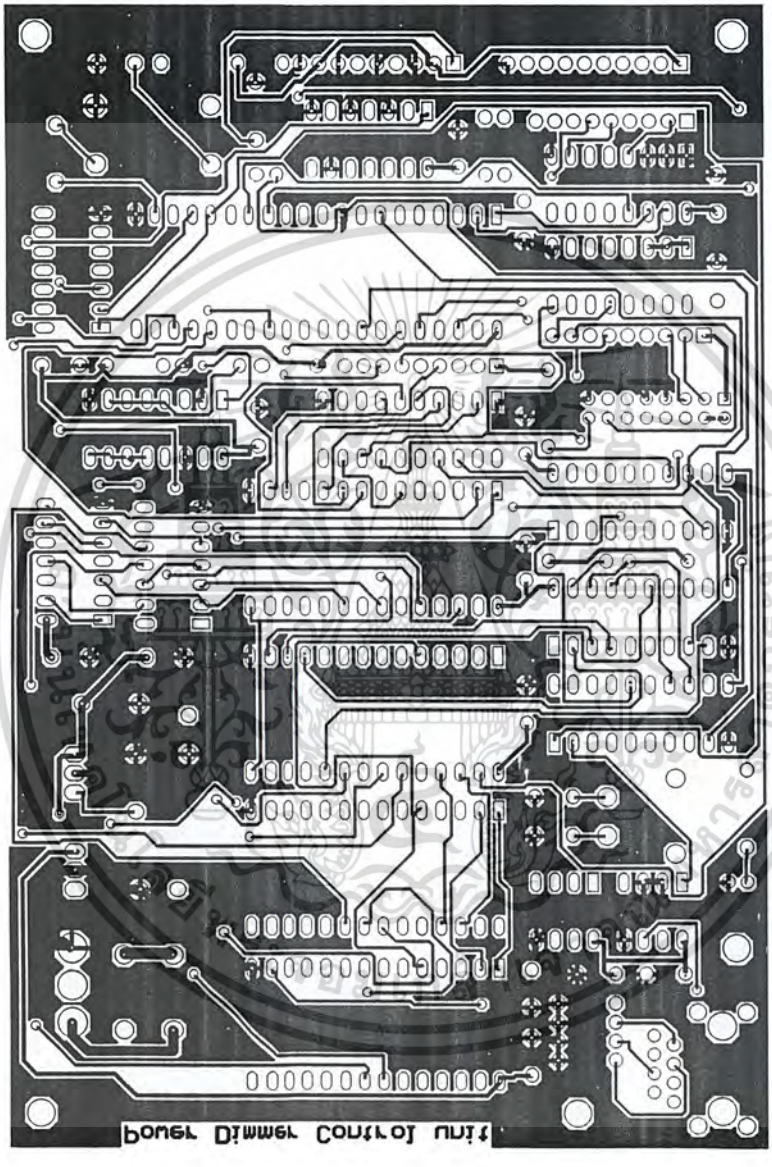
ภาคผนวก ข.

แผ่นวงจรพิมพ์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่59แผ่นวงจรพิมพ์ชุดควบคุมหลัก

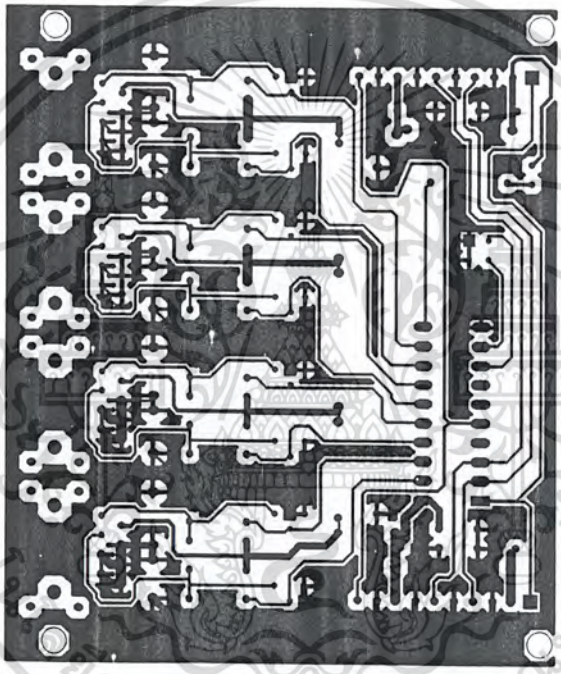
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่60แผงวงจรพิมพ์แปลงสัญญาณอนุกรมRS-232เป็นRS-422

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

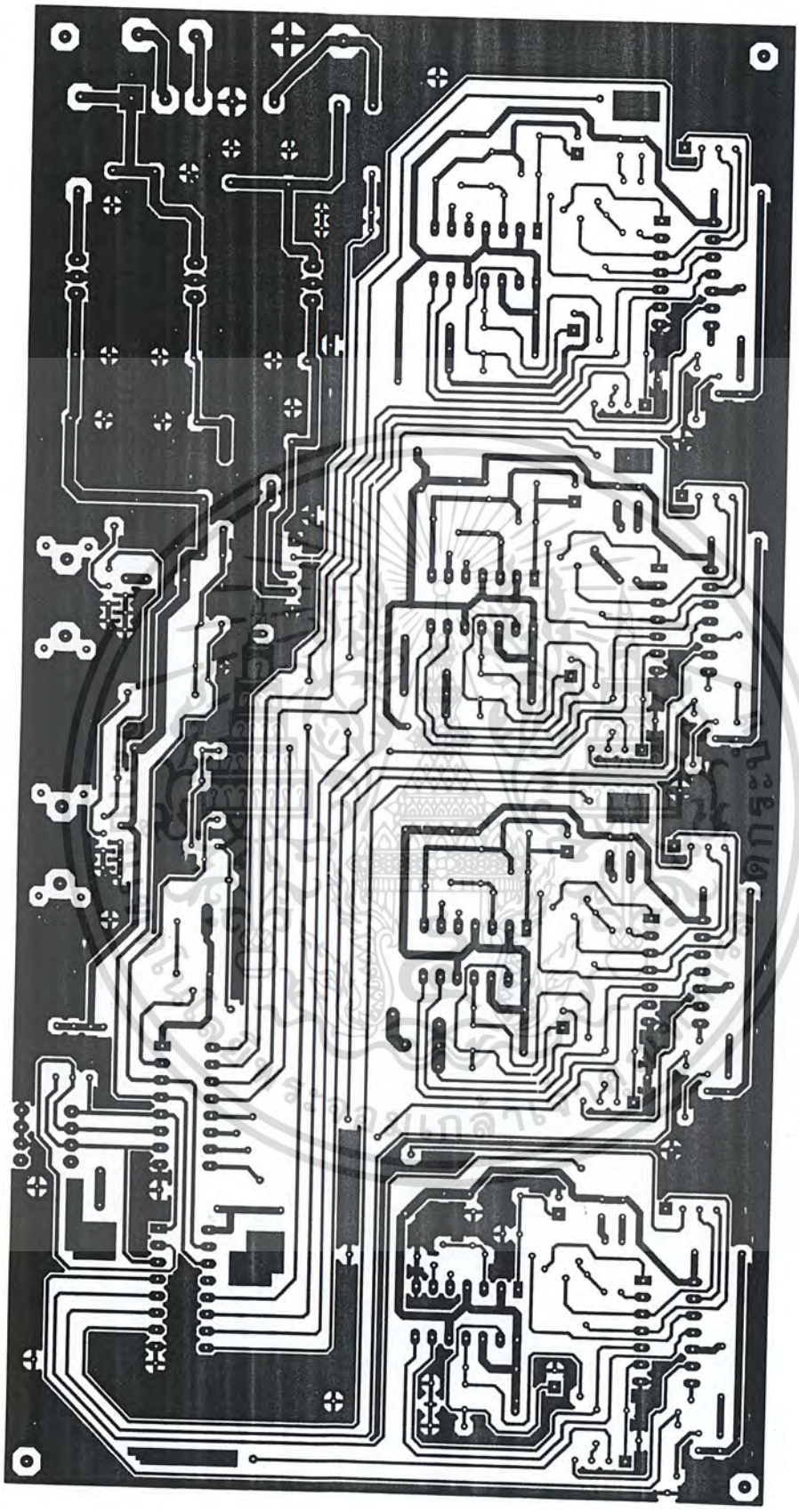


รูปที่ 61 แผ่นวงจรพิมพ์แปลงสัญญาณอนุกรมเป็นสัญญาณแบบโอเอสดาชิบัส

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ ข-2 ตัวอย่างแผ่นวงจรพิมพ์ที่ผลิตขึ้นสำหรับการทดสอบ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



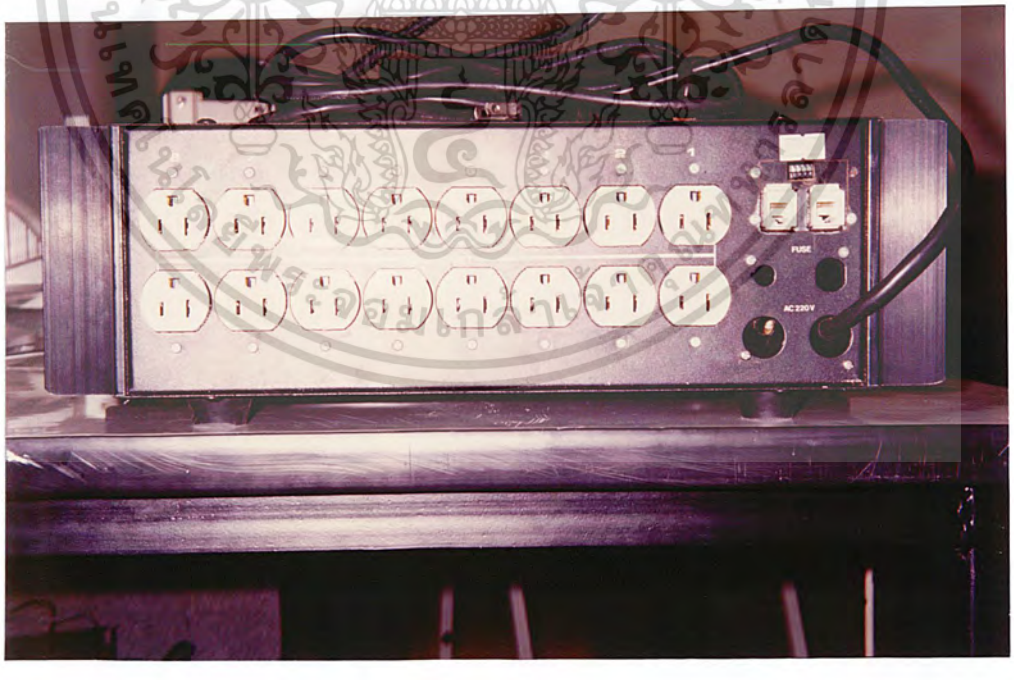
รูปที่63แผ่นวงจรพิมพ์ชุดแปลงสัญญาณดิจิตอลเป็นอนาลอก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก ค.
รูปถ่ายทั้งหมดของเครื่องควบคุมไฟเวที

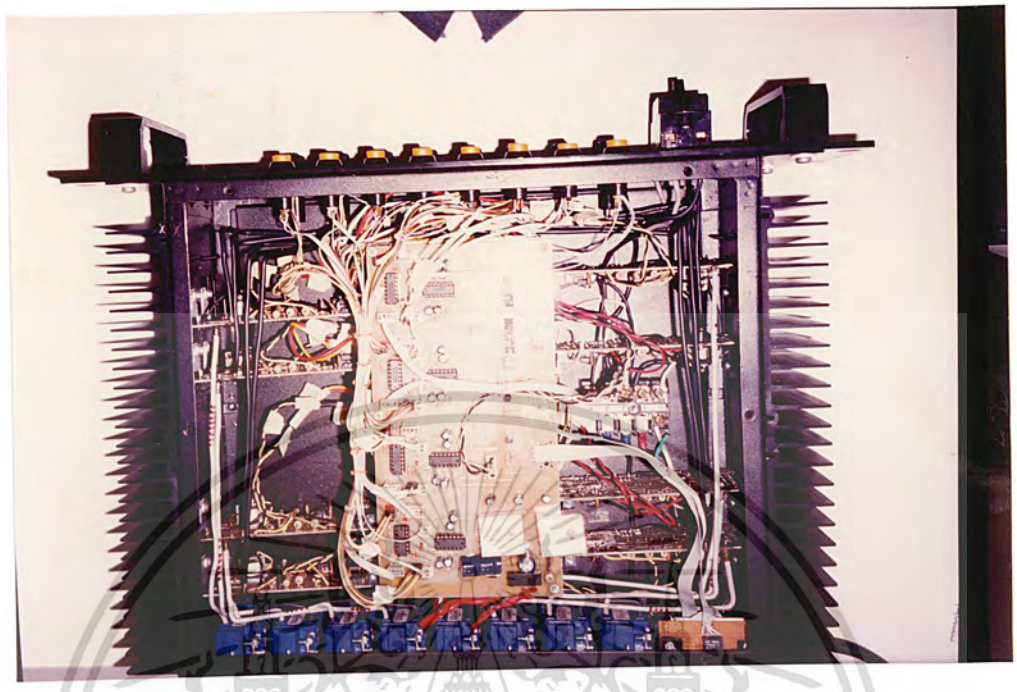
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่64 รูปถ่ายด้านหน้าของชุดดิมเมอร์แพก



รูปที่65 รูปถ่ายด้านหลังของชุดดิมเมอร์แพก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

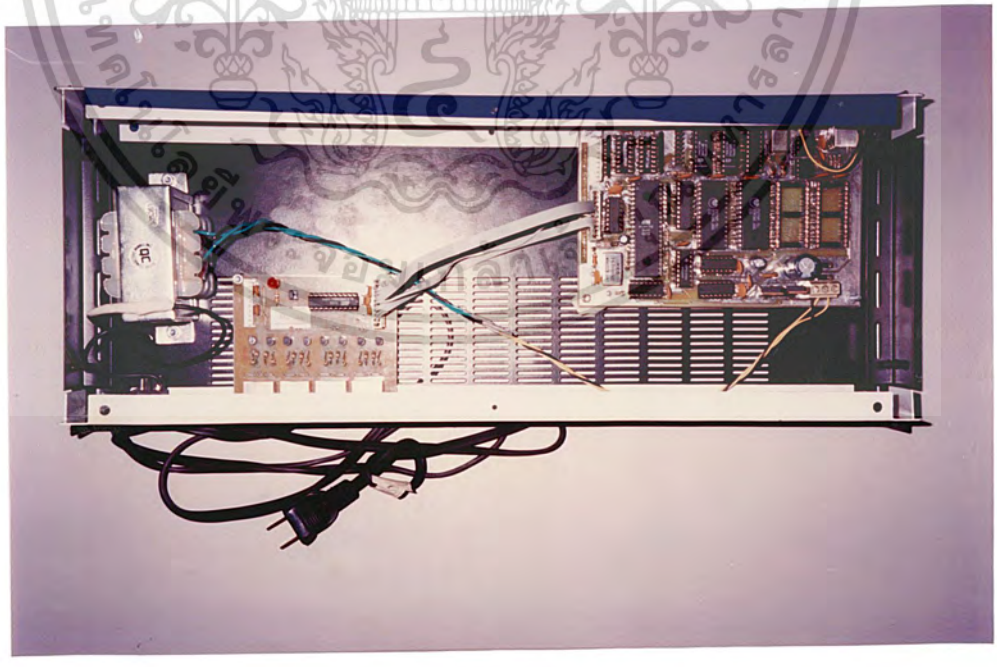


รูปที่66 รูปถ่ายด้านในของชุดดิมเมอร์แพก

รูปที่67 รูปถ่ายด้านหน้าของชุดคอนโทรลยูนิต

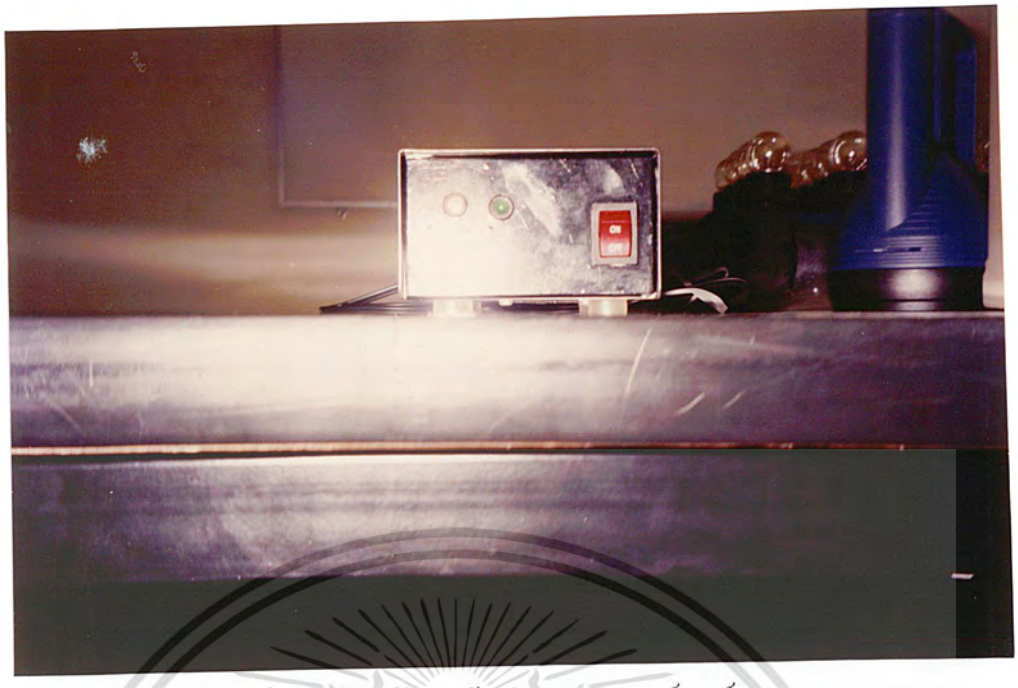
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่68 รูปถ่ายด้านหลังของชุดคอนโทรลยูนิต

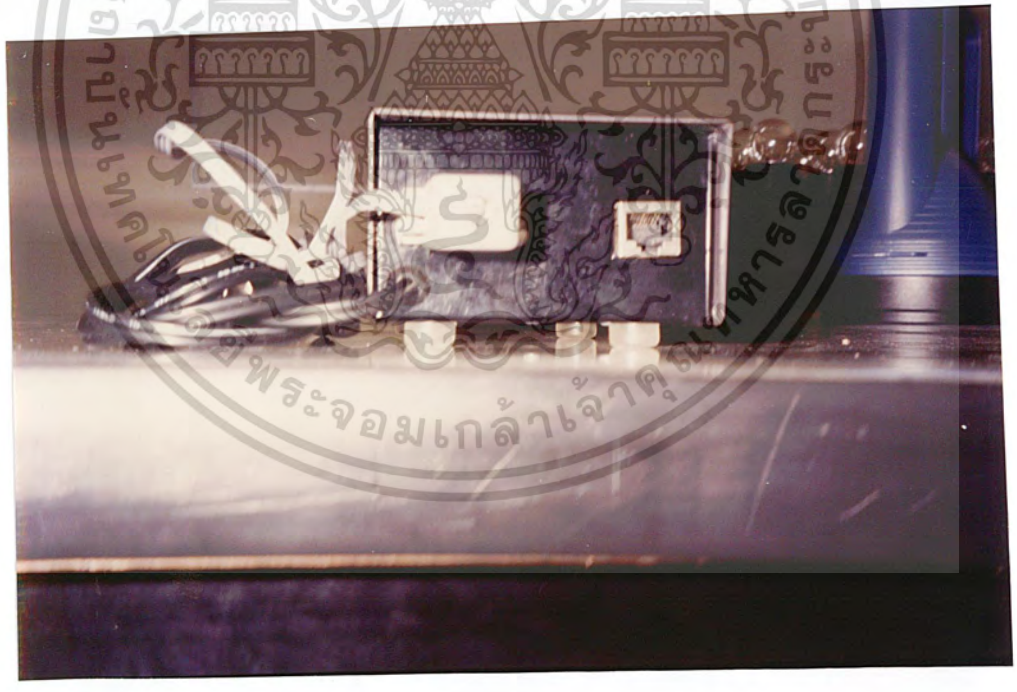


รูปที่69 รูปถ่ายด้านในของชุดคอนโทรลยูนิต

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

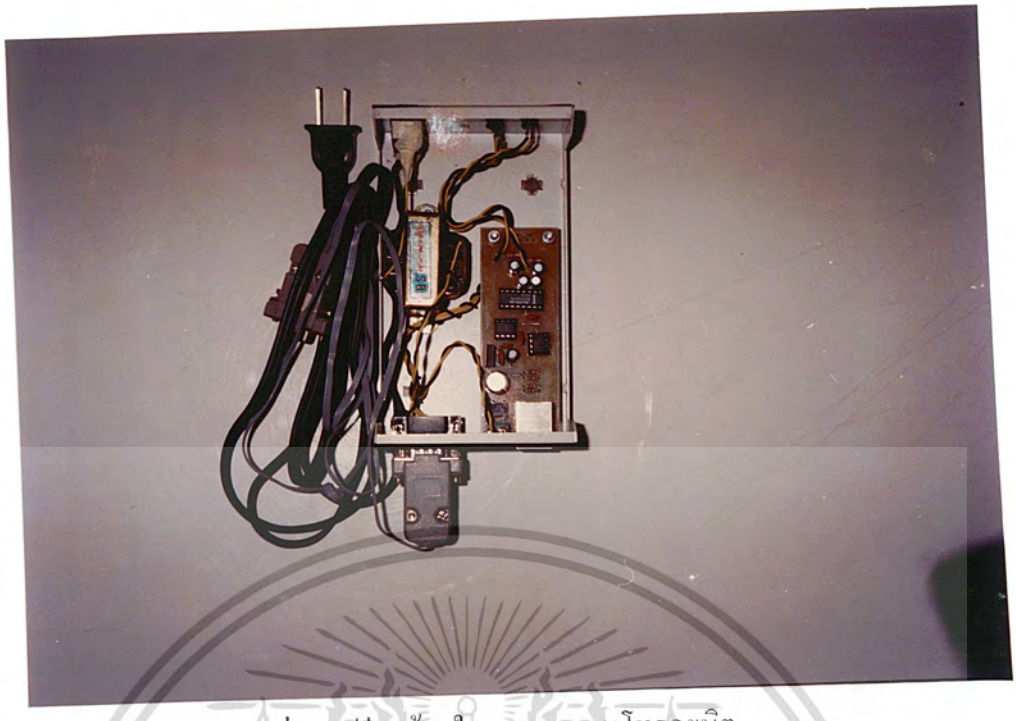


รูปที่70 .รูปถ่ายด้านหน้าของชุดคอนเวอร์เตอร์



รูปที่71 .รูปถ่ายด้านหลังของชุดคอนเวอร์เตอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่72.รูปถ่ายด้านในของชุดคอนโทรลยูนิต

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก ง.
โปรแกรมในส่วนของไมโครคอนโทรลเลอร์
และ วิชวลเบสิก-6

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
VERSION 5.00
Object = "{648A5603-2C6E-101B-82B6-000000000014}#1.1#0"; "MSCOMM32.OCX"
Begin VB.Form frmtest
   BackColor      =  &H008000000&
   Caption        =  "State Lighting Control"
   ClientHeight   =  4875
   ClientLeft     =  330
   ClientTop      =  600
   ClientWidth    =  8625
   LinkTopic      =  "Form1"
   MaxButton      =  0   'False
   ScaleHeight    =  4875
   ScaleWidth     =  8625
   Begin VB.Timer Timer1
      Interval       =  50
      Left           =  0
      Top            =  6000
   End
   Begin VB.Frame Frame1
      BackColor      =  &H008000000&
      Caption        =  "Configuration"
      BeginProperty Font
         Name           =  "MS Sans Serif"
         Size           =  9.75
         Charset        =  222
         Weight         =  700
         Underline      =  0   'False
         Italic         =  0   'False
         Strikethrough  =  0   'False
      EndProperty
      ForeColor      =  &H00FFFF00&
      Height         =  3735
      Left           =  6480
      TabIndex       =  16
      Top            =  840
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
Width           =  1935
Begin VB.ComboBox cmbGroup
   Height       =  330
   ItemData     =  "stITESTDIM.frx":0000
   Left         =  660
   List         =  "stITESTDIM.frx":001C
   TabIndex     =  25
   Text         =  "0"
   Top          =  2340
   Width        =  855
End
Begin VB.ComboBox cmbLine
   Height       =  330
   ItemData     =  "stITESTDIM.frx":0038
   Left         =  660
   List         =  "stITESTDIM.frx":0048
   TabIndex     =  23
   Text         =  "0"
   Top          =  1560
   Width        =  855
End
Begin VB.CommandButton Command17
   Caption      =  "Exit"
   BeginProperty Font
      Name      =  "MS Sans Serif"
      Size      =  9.75
      Charset   =  222
      Weight    =  700
      Underline =  0   'False
      Italic    =  0   'False
      Strikethrough =  0   'False
   EndProperty
   Height       =  495
   Left         =  480
   TabIndex     =  20
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    Top             =   3000
    Width           =   1095
  End
  Begin VB.ComboBox cmbTdaID
    Height          =   330
    ItemData        =   "stITESTDIM.frx":0058
    Left            =   660
    List            =   "stITESTDIM.frx":0074
    TabIndex        =   18
    Text            =   "0"
    Top             =   780
    Width           =   855
  End
  Begin VB.Label Label4
    Alignment       =   2  'Center
    AutoSize        =   -1  'True
    BackColor       =   &H008000000&
    Caption         =   "Gorup"
    BeginProperty Font
      Name          =   "MS Sans Serif"
      Size          =   9.75
      Charset       =   222
      Weight        =   700
      Underline     =   0   'False
      Italic        =   0   'False
      Strikethrough =   0   'False
    EndProperty
    ForeColor       =   &H00FFFF00&
    Height          =   240
    Index           =   3
    Left            =   570
    TabIndex        =   26
    Top             =   2040
    Width           =   675
  End
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
Begin VB.Label Label4
   Alignment      =  2  'Center
   AutoSize       =  -1  'True
   BackColor      =  &H00800000&
   Caption        =  "I2C Line"
   BeginProperty Font
      Name        =  "MS Sans Serif"
      Size        =  9.75
      Charset     =  222
      Weight      =  700
      Underline   =  0   'False
      Italic      =  0   'False
      Strikethrough  =  0   'False
   EndProperty
   ForeColor      =  &H00FFFF00&
   Height         =  240
   Index          =  2
   Left           =  480
   TabIndex       =  24
   Top            =  1260
   Width          =  855
End
Begin VB.Label Label4
   Alignment      =  2  'Center
   AutoSize       =  -1  'True
   BackColor      =  &H00800000&
   Caption        =  "TDA8444 ID."
   BeginProperty Font
      Name        =  "MS Sans Serif"
      Size        =  9.75
      Charset     =  222
      Weight      =  700
      Underline   =  0   'False
      Italic      =  0   'False
      Strikethrough  =  0   'False
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    EndProperty
    ForeColor     = &H00FFFF00&
    Height        = 240
    Index         = 1
    Left          = 300
    TabIndex      = 17
    Top           = 480
    Width         = 1335
  End
End
Begin VB.VScrollBar VScroll8
  Height        = 2775
  LargeChange   = 10
  Left          = 5640
  Max           = 63
  TabIndex      = 7
  TabStop       = 0  'False
  Top           = 960
  Width         = 255
End
Begin VB.VScrollBar VScroll7
  Height        = 2775
  LargeChange   = 10
  Left          = 4875
  Max           = 63
  TabIndex      = 6
  TabStop       = 0  'False
  Top           = 960
  Width         = 255
End
Begin VB.VScrollBar VScroll6
  Height        = 2775
  LargeChange   = 10
  Left          = 4170
  Max           = 63
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    TabIndex      = 5
    TabStop       = 0  'False
    Top           = 960
    Width         = 255
  End
  Begin VB.VScrollBar VScroll5
    Height        = 2775
    LargeChange   = 10
    Left          = 3450
    Max           = 63
    TabIndex      = 4
    TabStop       = 0  'False
    Top           = 960
    Width         = 255
  End
  Begin VB.VScrollBar VScroll4
    Height        = 2775
    LargeChange   = 10
    Left          = 2685
    Max           = 63
    TabIndex      = 3
    TabStop       = 0  'False
    Top           = 960
    Width         = 255
  End
  Begin VB.VScrollBar VScroll3
    Height        = 2775
    LargeChange   = 10
    Left          = 1920
    Max           = 63
    TabIndex      = 2
    TabStop       = 0  'False
    Top           = 960
    Width         = 255
  End
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
Begin MSCommLib.MSComm MSComm1
   Left        =  6720
   Top         =  6000
   _ExtentX    =  1005
   _ExtentY    =  1005
   _Version    =  393216
   DTREnable   =  -1  'True
End
Begin VB.VScrollBar VScroll2
   Height      =  2775
   LargeChange =  10
   Left        =  1200
   Max         =  63
   TabIndex    =  1
   TabStop     =  0  'False
   Top         =  960
   Width       =  255
End
Begin VB.VScrollBar VScroll1
   Height      =  2775
   LargeChange =  10
   Left        =  480
   Max         =  63
   TabIndex    =  0
   TabStop     =  0  'False
   Top         =  960
   Width       =  255
End
Begin VB.Label logo
   BackStyle   =  0  'Transparent
   Caption     =  "Industrial Electronics Kmitl"
   BeginProperty Font
      Name        =  "AngsanaUPC"
      Size        =  9
      Charset     =  222
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    Weight          = 700
    Underline       = 0  'False
    Italic          = 0  'False
    Strikethrough   = 0  'False
  EndProperty
  ForeColor       = &H00800000&
  Height          = 255
  Left            = 8640
  TabIndex        = 22
  Top             = 5280
  Width           = 4000
End
Begin VB.Label Label2
  Alignment       = 2  'Center
  AutoSize        = -1  'True
  BackColor       = &H00800000&
  Caption         = "Testing Program"
  BeginProperty Font
    Name          = "MS Sans Serif"
    Size          = 9.75
    Charset       = 222
    Weight        = 700
    Underline     = 0  'False
    Italic        = 0  'False
    Strikethrough = 0  'False
  EndProperty
  ForeColor       = &H00FFFF00&
  Height          = 240
  Left            = 6600
  TabIndex        = 21
  Top             = 240
  Width           = 1755
End
Begin VB.Label Label1
  Alignment       = 2  'Center
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
  AutoSize      = -1  'True
  BackColor     = &H00800000&
  Caption       = "Stage Lighting Control"
  BeginProperty Font
    Name        = "MS Sans Serif"
    Size        = 14.25
    Charset     = 222
    Weight      = 700
    Underline   = 0   'False
    Italic      = 0   'False
    Strikethrough = 0   'False
  EndProperty
  ForeColor     = &H00FFFF00&
  Height        = 360
  Left          = 1680
  TabIndex      = 19
  Top           = 120
  Width         = 3135
End
Begin VB.Label Label11
  Alignment     = 2  'Center
  AutoSize      = -1  'True
  BackColor     = &H00800000&
  Caption       = "CH8"
  BeginProperty Font
    Name        = "MS Sans Serif"
    Size        = 9.75
    Charset     = 222
    Weight      = 700
    Underline   = 0   'False
    Italic      = 0   'False
    Strikethrough = 0   'False
  EndProperty
  ForeColor     = &H00FFFF00&
  Height        = 240
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
  Left        = 5520
  TabIndex    = 15
  Top         = 4140
  Width       = 465
End
Begin VB.Label Label10
  Alignment   = 2 'Center
  AutoSize    = -1 'True
  BackColor   = &H00800000&
  Caption     = "CH7"
  BeginProperty Font
    Name      = "MS Sans Serif"
    Size      = 9.75
    Charset   = 222
    Weight    = 700
    Underline = 0 'False
    Italic    = 0 'False
    Strikethrough = 0 'False
  EndProperty
  ForeColor   = &H00FFFF00&
  Height      = 240
  Left        = 4800
  TabIndex    = 14
  Top         = 4140
  Width       = 465
End
Begin VB.Label Label9
  Alignment   = 2 'Center
  AutoSize    = -1 'True
  BackColor   = &H00800000&
  Caption     = "CH6"
  BeginProperty Font
    Name      = "MS Sans Serif"
    Size      = 9.75
    Charset   = 222
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    Weight        = 700
    Underline     = 0  'False
    Italic        = 0  'False
    Strikethrough = 0  'False
  EndProperty
  ForeColor     = &H00FFFF00&
  Height        = 240
  Left          = 4080
  TabIndex      = 13
  Top           = 4140
  Width         = 465
End
Begin VB.Label Label8
  Alignment     = 2  'Center
  AutoSize      = -1  'True
  BackColor     = &H00800000&
  Caption       = "CH4"
  BeginProperty Font
    Name        = "MS Sans Serif"
    Size        = 9.75
    Charset     = 222
    Weight      = 700
    Underline   = 0  'False
    Italic      = 0  'False
    Strikethrough = 0  'False
  EndProperty
  ForeColor     = &H00FFFF00&
  Height        = 240
  Left          = 2640
  TabIndex      = 12
  Top           = 4140
  Width         = 465
End
Begin VB.Label Label7
  Alignment     = 2  'Center
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    AutoSize        = -1  'True
    BackColor       = &H00800000&
    Caption         = "CH5"
    BeginProperty Font
      Name          = "MS Sans Serif"
      Size          = 9.75
      Charset       = 222
      Weight        = 700
      Underline     = 0   'False
      Italic        = 0   'False
      Strikethrough = 0   'False
    EndProperty
    ForeColor       = &H00FFFF00&
    Height          = 240
    Left            = 3360
    TabIndex        = 11
    Top             = 4140
    Width           = 465
  End
  Begin VB.Label Label6
    Alignment       = 2  'Center
    AutoSize        = -1  'True
    BackColor       = &H00800000&
    Caption         = "CH2"
    BeginProperty Font
      Name          = "MS Sans Serif"
      Size          = 9.75
      Charset       = 222
      Weight        = 700
      Underline     = 0   'False
      Italic        = 0   'False
      Strikethrough = 0   'False
    EndProperty
    ForeColor       = &H00FFFF00&
    Height          = 240
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    Left            = 1080
    TabIndex        = 10
    Top             = 4140
    Width           = 465
  End
  Begin VB.Label Label5
    Alignment       = 2 'Center
    AutoSize        = -1 'True
    BackColor       = &H00800000&
    Caption         = "CH3"
    BeginProperty Font
      Name          = "MS Sans Serif"
      Size          = 9.75
      Charset       = 222
      Weight        = 700
      Underline     = 0 'False
      Italic        = 0 'False
      Strikethrough = 0 'False
    EndProperty
    ForeColor       = &H00FFFF00&
    Height          = 240
    Left            = 1800
    TabIndex        = 9
    Top             = 4140
    Width           = 465
  End
  Begin VB.Label Label4
    Alignment       = 2 'Center
    AutoSize        = -1 'True
    BackColor       = &H00800000&
    Caption         = "CH1"
    BeginProperty Font
      Name          = "MS Sans Serif"
      Size          = 9.75
      Charset       = 222
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
      Weight        = 700
      Underline     = 0  'False
      Italic        = 0  'False
      Strikethrough = 0  'False
    'EndProperty
    ForeColor     = &H00FFFF00&
    Height        = 240
    Index         = 0
    Left          = 360
    TabIndex      = 8
    Top           = 4140
    Width         = 465
  End
End
Attribute VB_Name = "frmtest"
Attribute VB_GlobalNameSpace = False
Attribute VB_Creatable = False
Attribute VB_PredeclaredId = True
Attribute VB_Exposed = False
Option Explicit
Public TDA_ID As Integer
Public dPh As Integer
Public Line12c As Integer
Public Group As Integer
Private Sub Command1_Click()
        MSComm1.Output = Chr(0) & Chr(0) & Chr(63)
        VScroll1.Value = 63
End Sub


Private Sub cmbGroup_Change()
        Group = Val(cmbGroup.Text)
End Sub

Private Sub cmbLine_Change()
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        Line12c = Val(cmbLine.Text) * 64
        dPh = dPh And 63
        dPh = dPh Or Line12c

End Sub

Private Sub cmbTdaID_Change()
        TDA_ID = Val(cmbTdaID.Text) * 8
        dPh = dPh And 248
        dPh = dPh Or TDA_ID
End Sub

Private Sub Command17_Click()
    Unload Me
End Sub

 Private Sub Form_Load()
        MSComm1.CommPort = 2
        MSComm1.Settings = "4800,N,8,1"
        MSComm1.InputMode = comInputModeText
        MSComm1.RThreshold = 1
        MSComm1.PortOpen = True
        TDA_ID = Val(cmbTdaID.Text) * 8
        Line12c = Val(cmbLine.Text) * 64
        Group = Val(cmbGroup.Text)
        dPh = TDA_ID Or Line12c
        MSComm1.Output = Chr(0) & Chr(0) & Chr(63)
        MSComm1.Output = Chr(0) & Chr(1) & Chr(63)
        MSComm1.Output = Chr(0) & Chr(2) & Chr(63)
        MSComm1.Output = Chr(0) & Chr(3) & Chr(63)
        MSComm1.Output = Chr(0) & Chr(4) & Chr(63)
        MSComm1.Output = Chr(0) & Chr(5) & Chr(63)
        MSComm1.Output = Chr(0) & Chr(6) & Chr(63)
        MSComm1.Output = Chr(0) & Chr(7) & Chr(63)
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
End Sub

Private Sub VScroll1_Scroll()

        dPh = dPh Or 0
        MSComm1.Output = Chr(0) & Chr(0) & Chr(VScroll1.Value)

End Sub

Private Sub VScroll2_Scroll()
        dPh = dPh Or 1
        MSComm1.Output = Chr(0) & Chr(1) & Chr(VScroll2.Value)

End Sub

Private Sub VScroll3_Scroll()
        dPh = dPh Or 2
        MSComm1.Output = Chr(0) & Chr(2) & Chr(VScroll3.Value)
End Sub

Private Sub VScroll4_Scroll()
        dPh = dPh Or 3
        MSComm1.Output = Chr(0) & Chr(3) & Chr(VScroll4.Value)
End Sub

Private Sub VScroll5_Scroll()
        dPh = dPh Or 4
        MSComm1.Output = Chr(0) & Chr(4) & Chr(VScroll5.Value)

End Sub
Private Sub VScroll6_Scroll()
        dPh = dPh Or 5
        MSComm1.Output = Chr(0) & Chr(5) & Chr(VScroll6.Value)
End Sub
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
Private Sub VScroll7_Scroll()
        dPh = dPh Or 6
        MSComm1.Output = Chr(0) & Chr(6) & Chr(VScroll7.Value)
End Sub

Private Sub VScroll8_Scroll()
        dPh = dPh Or 7
        MSComm1.Output = Chr(0) & Chr(7) & Chr(VScroll8.Value)
End Sub
```


เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
;////////////////////////////////////////////////////////////
;  X051A.asm                                                 /
;  use to  X 051 A reciver and transmitter for serial data   /
;  Last modify : 4/8/44                                      /
;  Programmer : Supreecha Promrattana                        /
;  Compiler   : SXA51                                        /
;                                                            /
;************************************************************

BANK0       EQU   0E7H

IO_EN       BIT   P3.3          ;ENABLE 74LS573 FOR INPUT DATA
ACK         BIT   P3.4          ;ACKNOWLEDGE
CLK         BIT   P3.5          ;CLOCK PULSE FOR 74LS573
REQ         BIT   P3.7          ;REQUES TO SEND

RDY_RCV           EQU   020H
RCV_BUF           EQU   021H          ;3 byte 021 - 023
CH_BUF      EQU   026H
;************************************************************

            ORG   0000H
            AJMP  START

;------------------------------------------------------------
        ; INT_0 Servie routine
        ; use for recieve data from main CPU

            ORG   0003H
            CLR   ACK
            CLR   IO_EN
            MOV   P1,#0FFH
            MOV   SBUF,P1
            JB    TI,$
            SETB  IO_EN
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                SETB    ACK
                RETI

        ;
        ;Serial port interrupt service routine
        ;

                ORG     0023H
                JB      RI,RCV_SERIAL           ;Rx or Tx
                CLR     TI
                RETI
;---------------------------------------------------------------
RCV_SERIAL:     MOV     @R0,SBUF
                INC     R0
                CJNE    R0,#RCV_BUF+3,END_RX
                MOV     R0,#RCV_BUF
                PUSH    P1
                PUSH    P3
                MOV     P1,RCV_BUF
                ACALL   SND_TO_MAIN
                MOV     P1,RCV_BUF+1
                ACALL   SND_TO_MAIN
                MOV     P1,RCV_BUF+2
                ACALL   SND_TO_MAIN
                POP     P3
                POP     P1
END_RX:                 CLR     RI
                RETI

;---------------------------------------------------------------


;////////////////////////////////////////////////////////////
;************************************************************
;               Main program
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
;*****************************************************************

START:      SETB    IO_EN
            CLR     CLK
            SETB    REQ
            SETB    ACK
            SETB    INT0
            MOV     SP,#128-32
            MOV     R0,#128         ;128 BYTE OF INTERNAL RAM
CLR_RAM:    MOV     @R0,#00
            DJNZ    R0,CLR_RAM      ;CLEAR INTERNAL MEMORY
            MOV     IE,#091H        ;Enable  interrupt
                                    ;enable INT_0 and Serial port
                                    ;other disable
            MOV     IP,#011H        ;Serial port and INT0 height priority
            CLR     IT0             ;0 Level interrupt

;
; Set serial port baud rate
;Frequency 20.945 MHz
;Desired baud rate 115200 bps
;

;           MOV     SCON,#070H      ;Serial port mode 1 8 Bit SM2=0 REN=1
;           ORL     PCON,#80H       ;SMOD=1
;           MOV     TMOD,#20H       ;TIMER 1 MODE 2
;           MOV     TH1,#0FFH       ;Reload value for desired baud rate
;           MOV     P1,#0FFH        ;SET P1 to Height Impedance
;           SETB    TR1             ;Turn on timer 1

;
; Set serial port baud rate
;Frequency 20.945 MHz
;Desired baud rate 9600 bps
;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
;       MOV     SCON,#070H      ;Serial port mode 1 8 Bit SM2=0 REN=1
;       ORL     PCON,#80H       ;SMOD=1
;       MOV     TMOD,#20H       ;TIMER 1 MODE 2
;       MOV     TH1,#0F5H       ;Reload value for desired baud rate
;       MOV     P1,#0FFH        ;SET P1 to Height Impedance
;       SETB    TR1             ;Turn on timer 1

;
; Set serial port baud rate
;Frequency 18.432 MHz
;Desired baud rate 19200 bps
;

;       MOV     SCON,#070H      ;Serial port mode 1 8 Bit SM2=0 REN=1
;       ORL     PCON,#80H       ;SMOD=1
;       MOV     TMOD,#20H       ;TIMER 1 MODE 2
;       MOV     TH1,#0FBH       ;Reload value for desired baud rate
;       MOV     P1,#0FFH        ;SET P1 to Height Impedance
;       SETB    TR1             ;Turn on timer 1

;
; Set serial port baud rate
;Frequency 18.432 MHz
;Desired baud rate 9600 bps
;

;       MOV     SCON,#070H      ;Serial port mode 1 8 Bit SM2=0 REN=1
;       ORL     PCON,#80H       ;SMOD=1
;       MOV     TMOD,#20H       ;TIMER 1 MODE 2
;       MOV     TH1,#0F6H       ;Reload value for desired baud rate
;       MOV     P1,#0FFH        ;SET P1 to Height Impedance
;       SETB    TR1             ;Turn on timer 1
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
; Set serial port baud rate
;Frequency 18.432 MHz
;Desired baud rate 4800 bps
;
        MOV     SCON,#070H      ;Serial port mode 1 8 Bit SM2=1 REN=1
        ORL     PCON,#80H       ;SMOD=1
        MOV     TMOD,#20H       ;TIMER 1 MODE 2
        MOV     TH1,#0ECH       ;Reload value for desired baud rate
        MOV     P1,#0FFH        ;SET P1 to Height Impedance
        SETB    TR1             ;Turn on timer 1

;*****************************************************************
        MOV     R0,#RCV_BUF
        SJMP    $
;*****************************************************************

SND_TO_MAIN: SETB   CLK             ;PULSE P3.5
        NOP
        NOP
        CLR     CLK
        CLR     REQ
        JB      ACK,$
        SETB    REQ
        SETB    ACK
        RET


        END
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
SCL_A           BIT     P1.1
SDA_A           BIT     P1.0
SCL_B           BIT     P1.3
SDA_B           BIT     P1.2
SCL_C           BIT     P1.5
SDA_C           BIT     P1.4
SCL_D           BIT     P1.7
SDA_D           BIT     P1.6

;BIT addressing
SLAVE_ADDR      EQU     020H    ;keep address of X051B
FLAG            EQU     021H
I2C_IN_DATA     EQU     022H    ;keep data input from I2C
DPL_BUF                 EQU     023H
I2C_OUT_DATA    EQU     024H    ;keep data for output of I2C

;Ram area
I2C_ADDR        EQU     25H
I2C_SUB_ADDR    EQU     26H

;CONSTANT
I2C_BUSY        EQU     FLAG.0  ;1 if i2c process not complete
NEW_REQ                 EQU     FLAG.1  ;1 if Main CPU request to send data
;/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\
                ORG     0000H
                MOV     P1,#0FFH
                MOV     R0,#128
CLR_RAM:        MOV     @R0,#0
                DJNZ    R0,CLR_RAM
                MOV     R0,#DPL_BUF
                MOV     P3,#0FFH
                MOV     A,P3
                RR      A
                RR      A
                RR      A
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        RR      A
        ANL     A,#03H
        MOV     SLAVE_ADDR,A
        MOV     IE,#10010000B
        MOV     IP,#00010000B
        SJMP    MAIN

;]]]]]]]]]]]]]]]]]]]]]]]]]]]]]]]]]]]]]]]]]]]]]]]]]]]]]]]]]]]]]]]]]]]]]]]]]]]]]]]

        ORG     0023H

        JNB     RI,END_TX
        PUSH    ACC
RCV:    JB      SM2,GET_ID

GET_DATA:
        MOV     @R0,SBUF
        CLR     RI
        INC     R0
        POP     ACC
        RETI

GET_ID:
        MOV     A,SBUF
        CLR     RI
        SETB    TB8
        CJNE    A,SLAVE_ADDR,END_RX
        JNB     I2C_BUSY,ACK
        POP     ACC
        MOV     SBUF,#255
        RETI

ACK:
        CLR     SM2
        POP     ACC
        MOV     SBUF,SLAVE_ADDR
        RETI
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
END_RX:         POP     ACC
                RETI
                ;
END_TX:         CLR     TI
                RETI
                ;

;/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\

MAIN:   ;       MOV     SCON,#10110000B         ;MODE 2 SM2=1
        ;       ORL     PCON,#80H       ;SMOD=1 OSC/32 BAUD RATE

                MOV     TMOD,#20H
                MOV     TH1,#0FFH
                MOV     SCON,#0F0H
                ANL     PCON,#01111111B
                SETB    ES
                SETB    REN
                SETB    TR1

                MOV     R0,#DPL_BUF

START:          CJNE    R0,#I2C_OUT_DATA+1,$
                CLR     P3.7
                SETB    I2C_BUSY
                SETB    SM2
                MOV     R0,#DPL_BUF
                MOV     A,@R0
                ORL     A,#11110000B
                CLR     ACC.3
                MOV     I2C_SUB_ADDR,A
                MOV     A,@R0
                RR      A
                RR      A
                ANL     A,#00001110B
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                SETB    ACC.6
                MOV     I2C_ADDR,A
                JNB     DPL_BUF.7,TDA_AB
                JNB     DPL_BUF.6,TDA_C

TDA_D:          ACALL   TDA_WR_D
                CLR     I2C_BUSY
                SETB    P3.7
                SJMP    START
                ;
TDA_C:          ACALL   TDA_WR_C
                CLR     I2C_BUSY
                SETB    P3.7
                SJMP    START
                ;
TDA_AB:                 JNB     DPL_BUF.6,TDA_A
TDA_B:          ACALL   TDA_WR_B
                CLR     I2C_BUSY
                SETB    P3.7
                SJMP    START

TDA_A:          ACALL   TDA_WR_A
END_I2C:        CLR     I2C_BUSY
                SETB    P3.7
                SJMP    START


;-------------------------------------------------------------------

CLR_ALL:        MOV     I2C_ADDR,#01110000B
                MOV     I2C_OUT_DATA,#0FFH
                MOV     R2,#8
CLR_PCF:        ACALL   PCF_WR_A
                ACALL   PCF_WR_B
                ACALL   PCF_WR_C
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                ACALL   PCF_WR_D
                INC     I2C_ADDR
                INC     I2C_ADDR
                DJNZ    R2,CLR_PCF

                MOV     I2C_ADDR,#01000000B
                MOV     I2C_OUT_DATA,#63
                MOV     I2C_SUB_ADDR,#11110000B
                MOV     R2,#8
                MOV     R1,#8
CLR_TDA:        ACALL   TDA_WR_A
                ACALL   TDA_WR_B
                ACALL   TDA_WR_C
                ACALL   TDA_WR_D
                INC     I2C_SUB_ADDR
                DJNZ    R1,CLR_TDA
                INC     I2C_ADDR
                INC     I2C_ADDR
                DJNZ    R2,CLR_TDA
                RET

;////////////////////////////////////////////////////////////
;------------------------------------------------------------
;
;               TDA8444 Write at I2C A
;               INPUT : I2C_ADDR        > 0100 0XXX
;                       I2C_SUB_ADDR    > 1111 XXXX
;                       I2C_OUT_DATA    > 00XX XXXX
;               REG   : ACC
;------------------------------------------------------------

TDA_WR_A:       ACALL   I2C_CONNECT_A
                ACALL   TDA_INS_WR_A
                ACALL   I2C_DATA_WR_A
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                ACALL  I2C_STOP_A
                RET

;////////////////////////////////////////////////////////////
;------------------------------------------------------------
;
;               PCF8574 Write and Read at I2C A
;               INPUT : I2C_ADDR        > 0111 0XXX
;                       I2C_OUT_DATA    > XXXX XXXX
;               OUTPUT: I2C_IN_DATA
;               REG   : ACC
;------------------------------------------------------------

PCF_WR_A:       LCALL  I2C_CONNECT_A
                LCALL  I2C_DATA_WR_A
                LCALL  I2C_STOP_A
                RET

I2C_DATA_RD_A:  PUSH   ACC
                CLR    A
                MOV    R6,#8                ;Set loop 8 time

I2C_DAT_RD_1_A: LCALL  I2C_DELAY
                SETB   SCL_A
                LCALL  I2C_DELAY
                MOV    C,SDA_A              ;Get SDA to Carry flag
                RLC    A                    ;Rotate ACC. to Left with carry
                CLR    SCL_A
                DJNZ   R6,I2C_DAT_RD_1_A    ;Do until 8 time
                MOV    I2C_IN_DATA,A        ;Mov data to Buffer I2C_IN_DATA
                POP    ACC
                RET

;
; I2C Start Condition
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
;
I2C_START_A:    SETB    SCL_A
                SETB    SDA_A
                ACALL   I2C_DELAY
                CLR     SDA_A
                ACALL   I2C_DELAY
                CLR     SCL_A
                RET
;
; I2C Stop Condition
;
I2C_STOP_A:     CLR     SDA_A
                ACALL   I2C_DELAY
                SETB    SCL_A
                ACALL   I2C_DELAY
                SETB    SDA_A
                RET

I2C_DATA_WR_A:          PUSH    ACC
                MOV     A,I2C_OUT_DATA
                SJMP    I2C_CONNCT_1_A

TDA_INS_WR_A: PUSH      ACC
                MOV     A,I2C_SUB_ADDR
                SJMP    I2C_CONNCT_1_A
;
;       I2C CONNECT THE SLAVE
;       INPUT   : I2C_ADDR ;ADDRESS OF I2C PHERIPHERAL
;       REG     : ACC R6

I2C_CONNECT_A:          PUSH    ACC
                MOV     A,I2C_ADDR
                ACALL   I2C_START_A
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                ;
I2C_CONNCT_1_A:     MOV     R6,#8
I2C_CONNCT_2_A:     RLC     A
                    MOV     SDA_A,C
                    ACALL   I2C_CLK_A
                    DJNZ    R6,I2C_CONNCT_2_A
                    ;
                    SETB    SDA_A
                    ACALL   I2C_DELAY
                    SETB    SCL_A
                    JB      SDA_A,$+4
                    NOP
                    NOP
                    CLR     SCL_A
                    POP     ACC
                    RET

I2C_CLK_A:          ACALL   I2C_DELAY
                    SETB    SCL_A
                    ACALL   I2C_DELAY
                    CLR     SCL_A
                    RET

;//////////////////////////////////////////////////////////
;-----------------------------------------------------------
;
;                   TDA8444 Write at I2C B
;                   INPUT : I2C_ADDR        > 0100 0XXX
;                           I2C_SUB_ADDR    > 1111 XXXX
;                           I2C_OUT_DATA    > 00XX XXXX
;                   REG   : ACC
;-----------------------------------------------------------

TDA_WR_B:           ACALL   I2C_CONNECT_B
                    ACALL   TDA_INS_WR_B
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                ACALL   I2C_DATA_WR_B
                ACALL   I2C_STOP_B
                RET
;////////////////////////////////////////////////////////////////
;---------------------------------------------------------------
;
;               PCF8574 Write and Read at I2C B
;               INPUT : I2C_ADDR        > 0111 0XXX
;                       I2C_OUT_DATA    > XXXX XXXX
;               OUTPUT: I2C_IN_DATA
;               REG   : ACC
;---------------------------------------------------------------

PCF_WR_B:       LCALL   I2C_CONNECT_B
                LCALL   I2C_DATA_WR_B
                LCALL   I2C_STOP_B
                RET

I2C_DATA_RD_B:  PUSH    ACC
                CLR     A
                MOV     R6,#8                   ;Set loop 8 time

I2C_DAT_RD_1_B: LCALL   I2C_DELAY
                SETB    SCL_B
                LCALL   I2C_DELAY
                MOV     C,SDA_B                 ;Get SDA to Carry flag
                RLC     A                       ;Rotate ACC. to Left with carry
                CLR     SCL_B
                DJNZ    R6,I2C_DAT_RD_1_B       ;Do until 8 time
                MOV     I2C_IN_DATA,A           ;Mov data to Buffer I2C_IN_DATA
                POP     ACC
                RET

        ;
        ; I2C Start Condition
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        ;
I2C_START_B:  SETB   SCL_B
              SETB   SDA_B
              ACALL  I2C_DELAY
              CLR    SDA_B
              ACALL  I2C_DELAY
              CLR    SCL_B
              RET
        ;
        ; I2C Stop Condition
        ;
I2C_STOP_B:   CLR    SDA_B
              ACALL  I2C_DELAY
              SETB   SCL_B
              ACALL  I2C_DELAY
              SETB   SDA_B
              RET

I2C_DATA_WR_B:      PUSH   ACC
              MOV    A,I2C_OUT_DATA
              SJMP   I2C_CONNCT_1_B

TDA_INS_WR_B: PUSH   ACC
              MOV    A,I2C_SUB_ADDR
              SJMP   I2C_CONNCT_1_B
;
;       I2C CONNECT THE SLAVE
;       INPUT  : I2C_ADDR ;ADDRESS OF I2C PHERIPHERAL
;       REG    : ACC R6

I2C_CONNECT_B:      PUSH   ACC
              MOV    A,I2C_ADDR
              ACALL  I2C_START_B
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                ;
I2C_CONNCT_1_B:     MOV     R6,#8
I2C_CONNCT_2_B:     RLC     A
                    MOV     SDA_B,C
                    ACALL   I2C_CLK_B
                    DJNZ    R6,I2C_CONNCT_2_B
                    ;
                    SETB    SDA_B
                    ACALL   I2C_DELAY
                    SETB    SCL_B
                    JB      SDA_B,$+4
                    NOP
                    NOP
                    CLR     SCL_B
                    POP     ACC
                    RET

I2C_CLK_B:          ACALL   I2C_DELAY
                    SETB    SCL_B
                    ACALL   I2C_DELAY
                    CLR     SCL_B
                    RET

;////////////////////////////////////////////////////////////
;------------------------------------------------------------
;
;                   TDA8444 Write at I2C C
;                   INPUT : I2C_ADDR        > 0100 0XXX
;                           I2C_SUB_ADDR    > 1111 XXXX
;                           I2C_OUT_DATA    > 00XX XXXX
;                   REG   : ACC
;------------------------------------------------------------

TDA_WR_C:           ACALL   I2C_CONNECT_C
                    ACALL   TDA_INS_WR_C
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                ACALL   I2C_DATA_WR_C
                ACALL   I2C_STOP_C
                RET
;////////////////////////////////////////////////////////////////
;----------------------------------------------------------------
;
;               PCF8574 Write and Read at I2C C
;               INPUT : I2C_ADDR        > 0111 0XXX
;                       I2C_OUT_DATA    > XXXX XXXX
;               OUTPUT: I2C_IN_DATA
;               REG   : ACC
;----------------------------------------------------------------

PCF_WR_C:       LCALL   I2C_CONNECT_C
                LCALL   I2C_DATA_WR_C
                LCALL   I2C_STOP_C
                RET

I2C_DATA_RD_C:  PUSH    ACC
                CLR     A
                MOV     R6,#8                   ;Set loop 8 time

I2C_DAT_RD_1_C: LCALL   I2C_DELAY
                SETB    SCL_C
                LCALL   I2C_DELAY
                MOV     C,SDA_C                 ;Get SDA to Carry flag
                RLC     A                       ;Rotate ACC. to Left with carry
                CLR     SCL_C
                DJNZ    R6,I2C_DAT_RD_1_C       ;Do until 8 time
                MOV     I2C_IN_DATA,A           ;Mov data to Buffer I2C_IN_DATA
                POP     ACC
                RET
        ;
        ; I2C Start Condition
        ;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
I2C_START_C:    SETB    SCL_C
                SETB    SDA_C
                ACALL   I2C_DELAY
                CLR     SDA_C
                ACALL   I2C_DELAY
                CLR     SCL_C
                RET
        ;
        ; I2C Stop Condition
        ;

I2C_STOP_C:     CLR     SDA_C
                ACALL   I2C_DELAY
                SETB    SCL_C
                ACALL   I2C_DELAY
                SETB    SDA_C
                RET

I2C_DATA_WR_C:          PUSH    ACC
                MOV     A,I2C_OUT_DATA
                SJMP    I2C_CONNCT_1_C

TDA_INS_WR_C: PUSH      ACC
                MOV     A,I2C_SUB_ADDR
                SJMP    I2C_CONNCT_1_C
;
;       I2C CONNECT THE SLAVE
;       INPUT   : I2C_ADDR ;ADDRESS OF I2C PHERIPHERAL
;       REG     : ACC R6

I2C_CONNECT_C:          PUSH    ACC
                MOV     A,I2C_ADDR
                ACALL   I2C_START_C
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
I2C_CONNCT_1_C:     MOV    R6,#8
I2C_CONNCT_2_C:     RLC    A
                    MOV    SDA_C,C
                    ACALL  I2C_CLK_C
                    DJNZ   R6,I2C_CONNCT_2_C

                    SETB   SDA_C
                    ACALL  I2C_DELAY
                    SETB   SCL_C
                    JB     SDA_C,$+4
                    NOP
                    NOP
                    CLR    SCL_C
                    POP    ACC
                    RET

I2C_CLK_C:          ACALL  I2C_DELAY
                    SETB   SCL_C
                    ACALL  I2C_DELAY
                    CLR    SCL_C
                    RET

;////////////////////////////////////////////////////////////
;------------------------------------------------------------
;
;                   TDA8444 Write at I2C D
;                   INPUT : I2C_ADDR        > 0100 0XXX
;                           I2C_SUB_ADDR    > 1111 XXXX
;                           I2C_OUT_DATA    > 00XX XXXX
;                   REG   : ACC
;------------------------------------------------------------

TDA_WR_D:           ACALL  I2C_CONNECT_D
                    ACALL  TDA_INS_WR_D
                    ACALL  I2C_DATA_WR_D
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                ACALL   I2C_STOP_D
                RET
;////////////////////////////////////////////////////////////
;-------------------------------------------------------------
;
;               PCF8574 Write and Read at I2C D
;               INPUT : I2C_ADDR        > 0111 0XXX
;                       I2C_OUT_DATA    > XXXX XXXX
;               OUTPUT: I2C_IN_DATA
;               REG   : ACC
;-------------------------------------------------------------

PCF_WR_D:       LCALL   I2C_CONNECT_D
                LCALL   I2C_DATA_WR_D
                LCALL   I2C_STOP_D
                RET

I2C_DATA_RD_D:  PUSH    ACC
                CLR     A
                MOV     R6,#8                   ;Set loop 8 time

I2C_DAT_RD_1_D: LCALL   I2C_DELAY
                SETB    SCL_D
                LCALL   I2C_DELAY
                MOV     C,SDA_D                 ;Get SDA to Carry flag
                RLC     A                       ;Rotate ACC. to Left with carry
                CLR     SCL_D
                DJNZ    R6,I2C_DAT_RD_1_D       ;Do until 8 time
                MOV     I2C_IN_DATA,A           ;Mov data to Buffer I2C_IN_DATA
                POP     ACC
                RET
;
; I2C Start Condition
;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
I2C_START_D:    SETB    SCL_D
                SETB    SDA_D
                ACALL   I2C_DELAY
                CLR     SDA_D
                ACALL   I2C_DELAY
                CLR     SCL_D
                RET
        ;
        ; I2C Stop Condition
        ;

I2C_STOP_D:     CLR     SDA_D
                ACALL   I2C_DELAY
                SETB    SCL_D
                ACALL   I2C_DELAY
                SETB    SDA_D
                RET

I2C_DATA_WR_D:          PUSH    ACC
                MOV     A,I2C_OUT_DATA
                SJMP    I2C_CONNCT_1_D

TDA_INS_WR_D: PUSH      ACC
                MOV     A,I2C_SUB_ADDR
                SJMP    I2C_CONNCT_1_D
;
;       I2C CONNECT THE SLAVE
;       INPUT   : I2C_ADDR ;ADDRESS OF I2C PHERIPHERAL
;       REG     : ACC R6

I2C_CONNECT_D:          PUSH    ACC
                MOV     A,I2C_ADDR
                ACALL   I2C_START_D

I2C_CONNCT_1_D:         MOV     R6,#8
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
I2C_CONNCT_2_D:     RLC     A
                    MOV     SDA_D,C
                    ACALL   I2C_CLK_D
                    DJNZ    R6,I2C_CONNCT_2_D

                    SETB    SDA_D
                    ACALL   I2C_DELAY
                    SETB    SCL_D
                    JB      SDA_D,$+4
                    NOP
                    NOP
                    CLR     SCL_D
                    POP     ACC
                    RET

I2C_CLK_D:          ACALL   I2C_DELAY
                    SETB    SCL_D
                    ACALL   I2C_DELAY
                    CLR     SCL_D
                    RET

I2C_DELAY:          MOV     R7,#2           ;10 uS
I2C_DELAY_1:        NOP
                    DJNZ    R7,I2C_DELAY_1
                    RET

DELAY_XX:           MOV     R5,#5
DELAY_XX_1:         MOV     R4,#0FFH
DELAY_XX_2:         LCALL   I2C_DELAY
                    NOP
                    DJNZ    R4,DELAY_XX_2
                    DJNZ    R5,DELAY_XX_1
                    RET

                    END
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
SCL             BIT     P1.0
SDA             BIT     P1.1

ACK             BIT     P3.4
REQ             BIT     P3.5
CLK             BIT     P1.6
IO_EN           BIT     P1.7
BANK_SEL        BIT     P1.2
                ;
                ; Variable
                ;

;USE BIT ADDRESSING RAM
CH_CNT                  EQU     020H
I2C_OUT_DATA    EQU     021H
BYTE_CNT        EQU     022H
CRNT_SLAVE      EQU     023H
FLAG            EQU     024H
;FIRST 128 BYTE  RAM AREA
I2C_ADDR        EQU     030H
I2C_IN_DATA     EQU     031H
KEY_DATA        EQU     032H
HIGH_BYTE       EQU     033H

DPH_BUF                 EQU     034H    ;**** Change that after
DPL_BUF                 EQU     DPH_BUF+1
DATA_BUF        EQU     DPL_BUF+1
TIME_BUF        EQU     DATA_BUF+1
GROUP_DPH       EQU     TIME_BUF+1
GROUP_DPL       EQU     GROUP_DPH+1
GROUP_DATA      EQU     GROUP_DPL+1
temp            equ     GROUP_DATA+1
;BIT
EN_LCD          BIT     I2C_OUT_DATA.1
RS_LCD          BIT     I2C_OUT_DATA.0
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
GRP_COMPL    BIT    CH_CNT.3        ;Group complete BIT
X051B_BUSY   BIT    FLAG.0

        ;
        ; CONSTANT
        ;
PCF_LCD_ID   EQU    01111110B  ;PCF8573 Slave address at 07H  WR mode
PCF_KEY_ID   EQU    01110000B
BANK0        EQU    11100101B

;*****************************************************************

             ORG    0000H
             LJMP   MAIN

        ;
        ; X051A Request service data transfer at P0
        ;
             ORG    0003H                ;INT0 VECTER
             PUSH   ACC
             SETB   ACK
             LJMP   DATA_PROCESS

        ;
        ;KEY INTERRUPT SERVCIE ROUTINE
        ;

             ORG    0013H                ;INT1 VECTER
             LCALL  KEY_RD
             MOV    A,#43H
             LCALL  SET_DDRAM
             MOV    A,HIGH_BYTE
             LCALL  WR_DAT_LCD
             MOV    A,KEY_DATA
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                LCALL   WR_DAT_LCD
                SETB    INT1
                JNB     INT1,$-2
                NOP
                RETI

;---------------------------------------------------------------
;////////////////// Main  Program ////////////////////////////////
;---------------------------------------------------------------

MAIN:           SETB    IO_EN
                CLR     CLK
                SETB    REQ
                SETB    ACK
                SETB    INT0
                MOV     SP,#128-32
                MOV     IE,#10000001B           ;Enable INT0 INT1
                MOV     IP,#00000001B           ;INT0 and INT1 hi priority
                ANL     PSW,#BANK0              ;SELECT Rn to Bank 0
                SETB    IT0                     ;INT1 edge type interrupt
                CLR     IT1                     ;INT0 zero type interrupt
                MOV     R0,#128                 ;128 BYTE OF INTERNAL RAM
CLR_RAM:        MOV     @R0,#00
                DJNZ    R0,CLR_RAM              ;CLEAR INTERNAL MEMORY

;               MOV     SCON,#10110000B                 ;MODE 2 SM2=1
;               ORL     PCON,#80H

                MOV     TMOD,#20H
                MOV     TH1,#0FFH
                MOV     SCON,#0F0H
                ANL     PCON,#01111111B
                CLR     ES
                SETB    REN
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                SETB    TR1

                SETB    INT1
                SETB    INT0

;---------------------------------------------------------------------
yyy:            mov     dph,#00000000b
                mov     dpl,#00000000b
                clr     bank_sel

scan:           acall   serial
                jb      x051b_busy,scan
                inc     dpl
scan_1:         acall   serial
                jb      x051b_busy,scan_1
                inc     dpl
scan_2:         acall   serial
                jb      x051b_busy,scan_2
                inc     dpl
scan_3:         acall   serial
                jb      x051b_busy,scan_3
                inc     dpl
scan_4:         acall   serial
                jb      x051b_busy,scan_4
                inc     dpl
scan_5:         acall   serial
                jb      x051b_busy,scan_5
                inc     dpl
scan_6:         acall   serial
                jb      x051b_busy,scan_6
                inc     dpl
scan_7:         acall   serial
                jb      x051b_busy,scan_7
                mov     dptr,#0000
                sjmp    scan
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                mov     a,#63

new:            movx    @dptr,a
new_1:          lcall   serial
                jb      x051b_busy,new_1
                inc     dptr
                movx    @dptr,a
new_2:          lcall   serial
                jb      x051b_busy,new_2
                inc     dptr
                movx    @dptr,a
new_3:          lcall   serial
                jb      x051b_busy,new_3
                inc     dptr
                movx    @dptr,a
new_4:          lcall   serial
                jb      x051b_busy,new_4
                inc     dptr
                movx    @dptr,a
new_5:          lcall   serial
                jb      x051b_busy,new_5
                inc     dptr
                movx    @dptr,a
new_6:          lcall   serial
                jb      x051b_busy,new_6
                inc     dptr
                movx    @dptr,a
new_7:          lcall   serial
                jb      x051b_busy,new_7
                inc     dptr
                movx    @dptr,a
new_8:          lcall   serial
                jb      x051b_busy,new_8
                mov     dptr,#00
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                dec     a
                lcall   delay_xx
                lcall   delay_xx
                cjne    a,#0,new
                mov     a,#63
                sjmp    new

delay_xx:       mov     r4,#2
delay_xx_1:     mov     r3,#0ffh
delay_xx_2:     lcall   i2c_delay
                djnz    r3,delay_xx_2
                djnz    r4,delay_xx_1
                ret

;---------------------------------------------------------------
;///////////////////////////////////////////////////////////////
;                       '0123456789012345678'
MESSAGE:        DB      'Test_LCD AND KEY pad'
SND_MSG:        DB      'StageLightingControl'
MESSAGE1:       DB      'Receive=   Send=   '
NUMBER_TAB:     DB      030H,031H,032H,033H,034H,035H,036H,037H,038H,039H

;///////////////////////////////////////////////////////////////
;---------------------------------------------------------------
;       Serial service routine
;       INPUT : DPTR @DPTR
;       Reg   : ACC DPTR
;    OUTPUT : SBUF
;---------------------------------------------------------------

SERIAL:         PUSH    ACC
                CLR     X051B_BUSY
                MOV     A,DPH
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        ANL     A,#03H
        MOV     CRNT_SLAVE,A
        SETB    SM2
        SETB    TB8
        MOV     SBUF,A
        JNB     TI,$
        CLR     TI
        JNB     RI,$
        MOV     A,SBUF
        CLR     RI
        CJNE    A,CRNT_SLAVE,BUSY
        MOV     SBUF,DPL
        JNB     TI,$
        CLR     TI
        CLR     BANK_SEL
        MOVX    A,@DPTR
        MOV     SBUF,A
        JNB     TI,$
        CLR     TI
        POP     ACC
        RET

BUSY:   SETB    X051B_BUSY
        POP     ACC
        RET

;-------------------------------------------------------------------
;///////////////////////////////////////////////////////////////////
;-------------------------------------------------------------------
;
;               DATA PROCESSING
;
;///////////////////////////////////////////////////////////////////
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
DATA_PROCESS:   JNB     BYTE_CNT.1,NEW_BYTE
                JNB     BYTE_CNT.0,BYTE_3
                ANL     BYTE_CNT,#11111100B
                POP     ACC
                RETI

BYTE_3:         PUSH    DPH
                PUSH    DPL
                MOV     P0,#0FFH
                CLR     IO_EN
                MOV     A,P0
                SETB    IO_EN
                MOV     DPH,DPH_BUF
                MOV     DPL,DPL_BUF
;               CLR     BANK_SEL
                MOVX    @DPTR,A
                POP     DPL
                POP     DPH
                POP     ACC
                ANL     BYTE_CNT,#11111100B
                CLR     ACK
                SETB    INT0
                JNB     INT0,$-2
                SETB    ACK
                RETI


NEW_BYTE:       JNB     BYTE_CNT.0,BYTE_1
                MOV     P0,#0FFH
BYTE_2:         CLR     IO_EN
                MOV     DPL_BUF,P0
                SETB    IO_EN
                CLR     BYTE_CNT.0
                SETB    BYTE_CNT.1
                SJMP    END_PROCESS
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
BYTE_1:         MOV     P0,#0FFH
                CLR     IO_EN
                MOV     DPH_BUF,P0
                SETB    IO_EN
                SETB    BYTE_CNT.0

END_PROCESS:    POP     ACC
                CLR     ACK
END_PROCESS_1:  SETB    INT0
                JNB     INT0,END_PROCESS_1
                SETB    ACK
                RETI

;////////////////////////////////////////////////////////
;                                                      /
;               Send DATA to X051A                     /
;               input : A                             /
;               output : p0                           /
;////////////////////////////////////////////////////////
SND_X051A:      MOV     P0,A
                SETB    CLK             ;PULSE
                NOP
                NOP
                NOP
                NOP
                NOP
                CLR     CLK
                JNB     ACK,$
                CLR     REQ
                NOP
                SETB    REQ
                SETB    ACK
                RET
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
;////////////////////////////////////////////////////////////////
;/                                                              /
;/                        I2C Routine                           /
;/                                                              /
;////////////////////////////////////////////////////////////////
        ;
        ; PCF8574 at address 0111111x that connect to LCD port
        ; PCF8574 at address 0111000x that connect to Key switch port
        ;

;*******************************************************************************
;       INITIAL LCD 4 BIT INTERFACE
;-------------------------------------------------------------------------------

INIT_LCD:       MOV     A,#33H
                LCALL   WR_INS_LCD
                MOV     A,#32H
                LCALL   WR_INS_LCD
                MOV     A,#28H          ;FUNCTION SET
                LCALL   WR_INS_LCD
                MOV     A,#06H          ;ENTRY MODE
                LCALL   WR_INS_LCD
                MOV     A,#01H          ;CLEAR DISPLAY
                LCALL   WR_INS_LCD
                MOV     A,#0EH          ;08H Display OFF ,no cursor and not blink (0Ch for ON)

                LCALL   WR_INS_LCD      ;If 0fH then blink and show cursor
                RET

;*******************************************************************************
;       SET ADDRESS OF DDRAM
;       INPUT = A
;-------------------------------------------------------------------------------
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
SET_DDRAM:    SETB    ACC.7
              LCALL   WR_INS_LCD
              RET

;*****************************************************************************
;             WRITE instruction to LCD USE PCF8574 at address 07H
;             INPUT  = A
;             REG    = I2C_OUT_DATA , A ,B
;-----------------------------------------------------------------------------

WR_INS_LCD:   MOV     B,A             ;Keep A
              ANL     A,#0F0H
              MOV     I2C_OUT_DATA,A
              LCALL   PCF_LCD_WR
              LCALL   ENABLE_LCD
              MOV     A,B
              SWAP    A
              ANL     A,#0F0H
              MOV     I2C_OUT_DATA,A
              LCALL   PCF_LCD_WR

ENABLE_LCD:   SETB    EN_LCD
              LCALL   PCF_LCD_WR
              CLR     EN_LCD
              LCALL   PCF_LCD_WR
              RET


;*****************************************************************************
;             WRITE DATA to LCD USE PCF8574 at address 07H
;             INPUT  = A
;             REG    = I2C_OUT_DATA , A ,B
;-----------------------------------------------------------------------------
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
WR_DAT_LCD:  MOV   B,A              ;Keep A
             ANL   A,#0F0H
             MOV   I2C_OUT_DATA,A
             SETB  RS_LCD           ;SET RS to Select Data Register
             LCALL PCF_LCD_WR
             LCALL ENABLE_LCD
             MOV   A,B
             SWAP  A
             ANL   A,#0F0H
             MOV   I2C_OUT_DATA,A
             SETB  RS_LCD           ;SET RS to Select Data Register
             LCALL PCF_LCD_WR
             LCALL ENABLE_LCD
             RET

;-------------------------------------------------------------------------------
;            OUTPUT = KEY_DATA
;
KEY_RD:      LCALL  PCF_KEY_RD
             MOV    KEY_DATA,I2C_IN_DATA
             MOV    I2C_OUT_DATA,#0F0H
             LCALL  PCF_KEY_WR
             LCALL  PCF_KEY_RD
             MOV    A,I2C_IN_DATA
             ORL    A,KEY_DATA
             MOV    HIGH_BYTE,#0

KEY_1:       CJNE   A,#0EEH,KEY_2       ;KEY 1  PRESS
             MOV    KEY_DATA,#31H
             SJMP   END_KEY

KEY_2:       CJNE   A,#0DEH,KEY_3       ;KEY 2  PRESS
             MOV    KEY_DATA,#32H
             SJMP   END_KEY
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
KEY_3:      CJNE    A,#0BEH,KEY_4       ;KEY 3  PRESS
            MOV     KEY_DATA,#33H
            SJMP    END_KEY

KEY_4:      CJNE    A,#07EH,KEY_5       ;KEY 4  PRESS
            MOV     KEY_DATA,#34H
            SJMP    END_KEY

KEY_5:      CJNE    A,#0EDH,KEY_6       ;KEY 5  PRESS
            MOV     KEY_DATA,#35H
            SJMP    END_KEY

KEY_6:      CJNE    A,#0DDH,KEY_7       ;KEY 6  PRESS
            MOV     KEY_DATA,#36H
            SJMP    END_KEY

KEY_7:      CJNE    A,#0BDH,KEY_8       ;KEY 7  PRESS
            MOV     KEY_DATA,#37H
            SJMP    END_KEY

KEY_8:      CJNE    A,#07DH,KEY_9       ;KEY 8  PRESS
            MOV     KEY_DATA,#38H
            SJMP    END_KEY

KEY_9:      CJNE    A,#0EBH,KEY_10      ;KEY 9  PRESS
            MOV     KEY_DATA,#39H
            SJMP    END_KEY

KEY_10:     MOV     HIGH_BYTE,#031H

            CJNE    A,#0DBH,KEY_11              ;KEY 10 PRESS
            MOV     KEY_DATA,#30H
            SJMP    END_KEY
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
KEY_11:     CJNE   A,#0BBH,KEY_12          ;KEY 11 PRESS
            MOV    KEY_DATA,#31H
            SJMP   END_KEY

KEY_12:     CJNE   A,#07BH,KEY_13          ;KEY 12 PRESS
            MOV    KEY_DATA,#32H
            SJMP   END_KEY

KEY_13:     CJNE   A,#0E7H,KEY_14          ;KEY 13 PRESS
            MOV    KEY_DATA,#33H
            SJMP   END_KEY

KEY_14:     CJNE   A,#0D7H,KEY_15          ;KEY 14 PRESS
            MOV    KEY_DATA,#34H
            SJMP   END_KEY

KEY_15:     CJNE   A,#0B7H,KEY_16          ;KEY 15 PRESS
            MOV    KEY_DATA,#35H
            SJMP   END_KEY

KEY_16:     CJNE   A,#077H,END_KEY         ;KEY 16 PRESS
            MOV    KEY_DATA,#36H

END_KEY:    MOV    I2C_OUT_DATA,#0FH
            LCALL  PCF_KEY_WR

            RET


;-------------------------------------------------------------------------------
;           OUTPUT = I2C_IN_DATA
;

PCF_KEY_RD: MOV    I2C_ADDR,#PCF_KEY_ID+1
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                LCALL   I2C_CONNECT
                LCALL   I2C_DATA_RD
                LCALL   I2C_ACK_BIT     ;Send acknowledge
                LCALL   I2C_STOP        ;Send stop condition
                RET

;-------------------------------------------------------------------------------
;               INPUT = I2C_OUT_DATA
;

PCF_KEY_WR:     MOV     I2C_ADDR,#PCF_KEY_ID
                SJMP    PCF8574_WR
                ;
PCF_LCD_WR:     MOV     I2C_ADDR,#PCF_LCD_ID
                ;
PCF8574_WR:     PUSH    PSW
                SETB    PSW.3
                SETB    PSW.4           ;SET Rn to BANK 4
                LCALL   I2C_CONNECT
                LCALL   I2C_DATA_WR
                LCALL   I2C_STOP
                POP     PSW
                RET

I2C_DATA_RD:    PUSH    ACC
                PUSH    PSW
                SETB    PSW.3
                SETB    PSW.4           ;SET Rn to BANK 4
                CLR     A
                MOV     R6,#8                   ;Set loop 8 time
I2C_DATA_RD_1:          LCALL   I2C_DELAY
                SETB    SCL
                LCALL   I2C_DELAY
                MOV     C,SDA                   ;Get SDA to Carry flag
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        RLC     A                   ;Rotate ACC. to Left with carry
        CLR     SCL
        DJNZ    R6,I2C_DATA_RD_1    ;Do until 8 time
        MOV     I2C_IN_DATA,A       ;Mov data to Buffer I2C_IN_DATA
        POP     PSW
        POP     ACC
        RET
;
; I2C Start Condition
;
I2C_START:  SETB    SCL
        SETB    SDA
        LCALL   I2C_DELAY
        CLR     SDA
        LCALL   I2C_DELAY
        CLR     SCL
        RET
;
; I2C Stop Condition
;
I2C_STOP:   CLR     SDA
        LCALL   I2C_DELAY
        SETB    SCL
        LCALL   I2C_DELAY
        SETB    SDA
        RET

I2C_ACK_BIT:    SETB    SDA
        LCALL   I2C_DELAY
        LCALL   I2C_CLK
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                RET

I2C_DATA_WR:    PUSH    ACC
                MOV     A,I2C_OUT_DATA
                SJMP    I2C_CONNECT_1
;
;       I2C CONNECT THE SLAVE
;       INPUT   : I2C_ADDR ;ADDRESS OF I2C PHERIPHERAL
;       REG     : ACC R6

I2C_CONNECT:    PUSH    ACC
                LCALL   I2C_START
                MOV     A,I2C_ADDR

I2C_CONNECT_1:          MOV     R6,#8
I2C_CONNECT_2:          RLC     A
                MOV     SDA,C
                LCALL   I2C_CLK
                DJNZ    R6,I2C_CONNECT_2

                SETB    SDA
                LCALL   I2C_DELAY
                SETB    SCL
                JB      SDA,$
                CLR     SCL
                POP     ACC
                RET

I2C_CLK:        LCALL   I2C_DELAY
                SETB    SCL
                LCALL   I2C_DELAY
                CLR     SCL
                RET
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
I2C_DELAY:    MOV   R7,#5          ;10 uS
I2C_DELAY_1:  NOP
              NOP
              DJNZ  R7,I2C_DELAY_1
              RET

              END
```


เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

DATA SHEET

PCF8574
Remote 8-bit I/O expander for I²C-bus

Product specification
Supersedes data of September 1994
File under Integrated Circuits, IC12

1997 Apr 02

Philips Semiconductors

PHILIPS

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

CONTENTS

1 FEATURES
2 GENERAL DESCRIPTION
3 ORDERING INFORMATION
4 BLOCK DIAGRAM
5 PINNING
6 CHARACTERISTICS OF THE I²C-BUS
6.1 Bit transfer
6.2 Start and stop conditions
6.3 System configuration
6.4 Acknowledge
7 FUNCTIONAL DESCRIPTION
7.1 Addressing
7.2 Interrupt
7.3 Quasi-bidirectional I/Os
8 LIMITING VALUES
9 HANDLING
10 DC CHARACTERISTICS
11 I²C-BUS TIMING CHARACTERISTICS
12 PACKAGE OUTLINES
13 SOLDERING
13.1 Introduction
13.2 DIP
13.2.1 Soldering by dipping or by wave
13.2.2 Repairing soldered joints
13.3 SO and SSOP
13.3.1 Reflow soldering
13.3.2 Wave soldering
13.3.3 Repairing soldered joints
14 DEFINITIONS
15 LIFE SUPPORT APPLICATIONS
16 PURCHASE OF PHILIPS I²C COMPONENTS

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

FEATURES

Operating supply voltage 2.5 to 6 V

Low standby current consumption of 10 μA maximum

I²C to parallel port expander

Open-drain interrupt output

8-bit remote I/O port for the I²C-bus

Compatible with most microcontrollers

Latched outputs with high current drive capability for directly driving LEDs

Address by 3 hardware address pins for use of up to 8 devices (up to 16 with PCF8574A)

DIP16, or space-saving SO16 or SSOP20 packages.

2 GENERAL DESCRIPTION

The PCF8574 is a silicon CMOS circuit. It provides general purpose remote I/O expansion for most microcontroller families via the two-line bidirectional bus (I²C).

The device consists of an 8-bit quasi-bidirectional port and an I²C-bus interface. The PCF8574 has a low current consumption and includes latched outputs with high current drive capability for directly driving LEDs. It also possesses an interrupt line ($\overline{INT}$) which can be connected to the interrupt logic of the microcontroller. By sending an interrupt signal on this line, the remote I/O can inform the microcontroller if there is incoming data on its ports without having to communicate via the I²C-bus. This means that the PCF8574 can remain a simple slave device.

The PCF8574 and PCF8574A versions differ only in their slave address as shown in Fig.9.

ORDERING INFORMATION

TYPE NUMBER	PACKAGE		
	NAME	DESCRIPTION	VERSION
PCF8574P; PCF8574AP	DIP16	plastic dual in-line package; 16 leads (300 mil)	SOT38-1
PCF8574T; PCF8574AT	SO16	plastic small outline package; 16 leads; body width 7.5 mm	SOT162-1
PCF8574TS	SSOP20	plastic shrink small outline package; 20 leads; body width 4.4 mm	SOT266-1

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4 BLOCK DIAGRAM

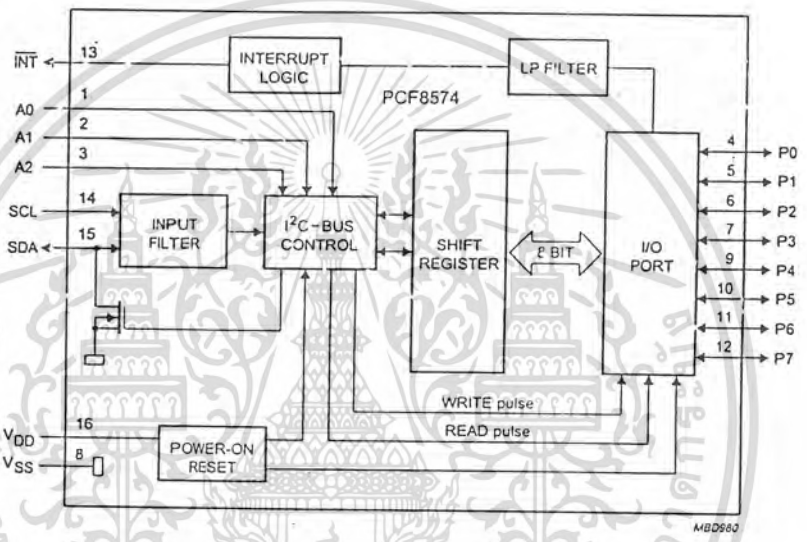


Fig.1 Block diagram (SOT38-1 and SOT162-1).

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5 PINNING

SYMBOL	PIN		DESCRIPTION
	DIP16; SO16	SSOP20	
A0	1	6	address input 0
A1	2	7	address input 1
A2	3	9	address input 2
P0	4	10	quasi-bidirectional I/O 0
P1	5	11	quasi-bidirectional I/O 1
P2	6	12	quasi-bidirectional I/O 2
P3	7	14	quasi-bidirectional I/O 3
V_{SS}	8	15	supply ground
P4	9	16	quasi-bidirectional I/O 4
P5	10	17	quasi-bidirectional I/O 5
P6	11	19	quasi-bidirectional I/O 6
P7	12	20	quasi-bidirectional I/O 7
$\overline{INT}$	13	1	interrupt output (active LOW)
SCL	14	2	serial clock line
SDA	15	4	serial data line
V_{DD}	16	5	supply voltage
n.c.	–	3	not connected
n.c.	–	8	not connected
n.c.	–	13	not connected
n.c.	–	18	not connected

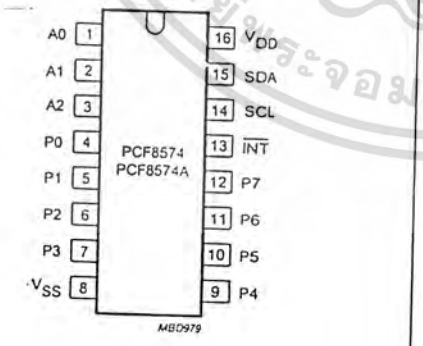


Fig.2 Pin configuration (DIP16; SO16).

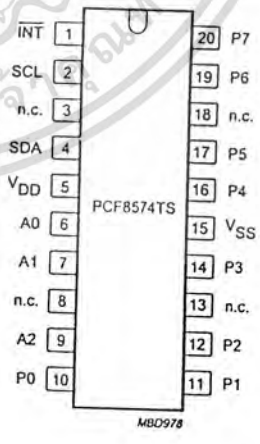


Fig.3 Pin configuration (SSOP20).

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

6 CHARACTERISTICS OF THE I²C-BUS

The I²C-bus is for 2-way, 2-line communication between different ICs or modules. The two lines are a serial data line (SDA) and a serial clock line (SCL). Both lines must be connected to a positive supply via a pull-up resistor when connected to the output stages of a device. Data transfer may be initiated only when the bus is not busy.

6.1 Bit transfer

One data bit is transferred during each clock pulse. The data on the SDA line must remain stable during the HIGH period of the clock pulse as changes in the data line at this time will be interpreted as control signals (see Fig.4).

6.2 Start and stop conditions

Both data and clock lines remain HIGH when the bus is not busy. A HIGH-to-LOW transition of the data line, while the clock is HIGH is defined as the start condition (S). A LOW-to-HIGH transition of the data line while the clock is HIGH is defined as the stop condition (P) (see Fig.5).

6.3 System configuration

A device generating a message is a 'transmitter', a device receiving is the 'receiver'. The device that controls the message is the 'master' and the devices which are controlled by the master are the 'slaves' (see Fig.6).

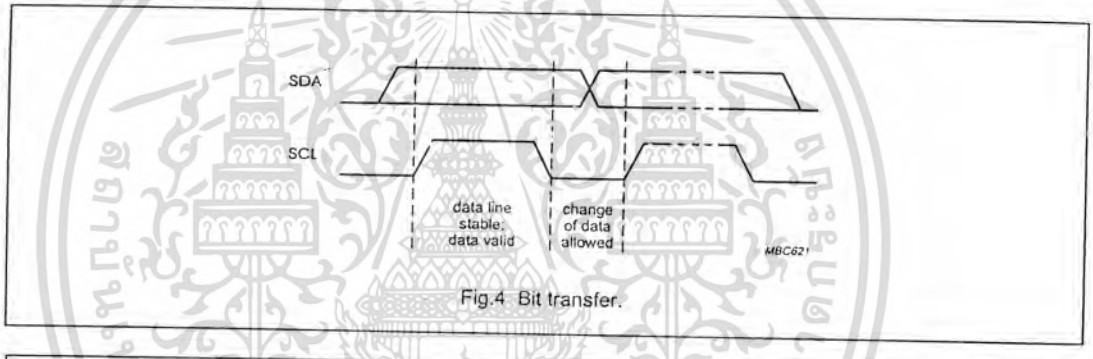


Fig.4 Bit transfer.

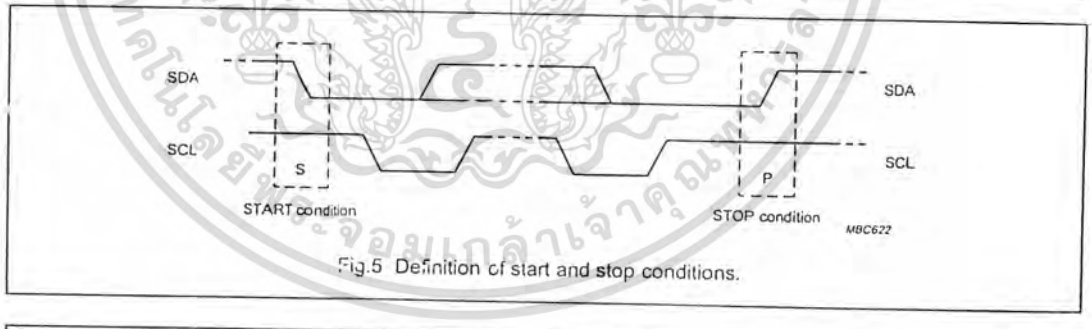


Fig.5 Definition of start and stop conditions.

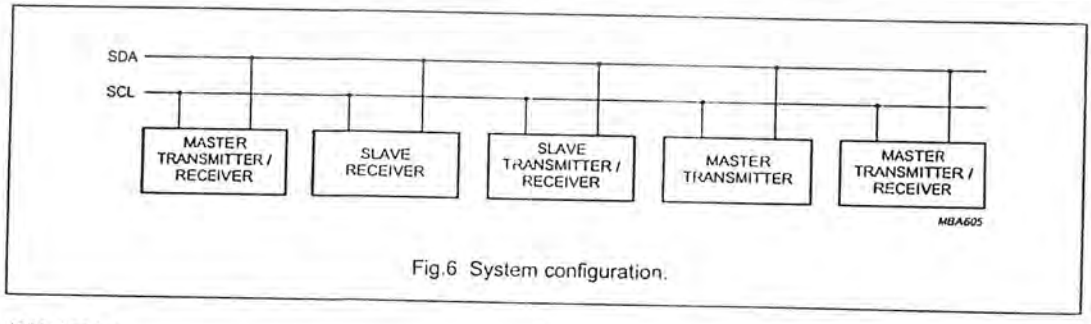


Fig.6 System configuration.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

6.4 Acknowledge

The number of data bytes transferred between the start and the stop conditions from transmitter to receiver is not limited. Each byte of eight bits is followed by one acknowledge bit. The acknowledge bit is a HIGH level put on the bus by the transmitter whereas the master generates an extra acknowledge related clock pulse.

A slave receiver which is addressed must generate an acknowledge after the reception of each byte. Also a master must generate an acknowledge after the reception of each byte that has been clocked out of the slave transmitter. The device that acknowledges has to pull down the SDA line during the acknowledge clock pulse, so that the SDA line is stable LOW during the HIGH period of the acknowledge related clock pulse, set-up and hold times must be taken into account.

A master receiver must signal an end of data to the transmitter by **not** generating an acknowledge on the last byte that has been clocked out of the slave. In this event the transmitter must leave the data line HIGH to enable the master to generate a stop condition.

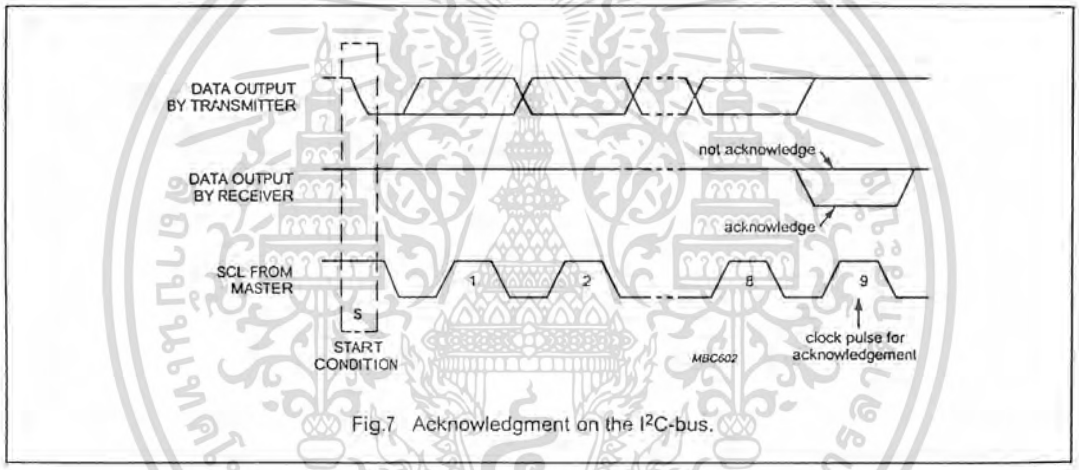


Fig.7 Acknowledgment on the I²C-bus.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

7 FUNCTIONAL DESCRIPTION

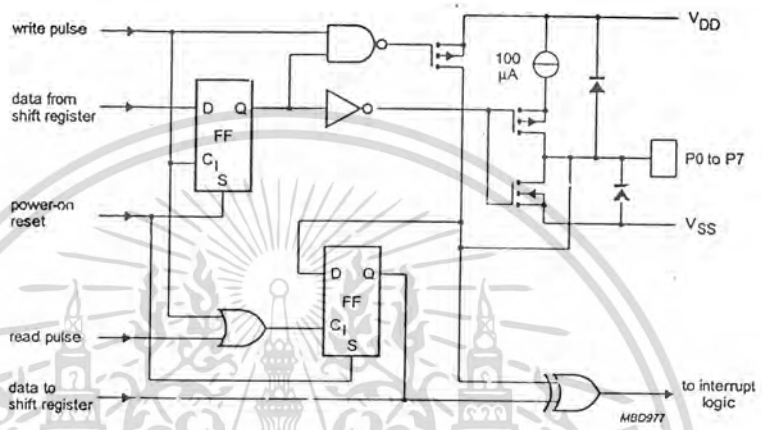
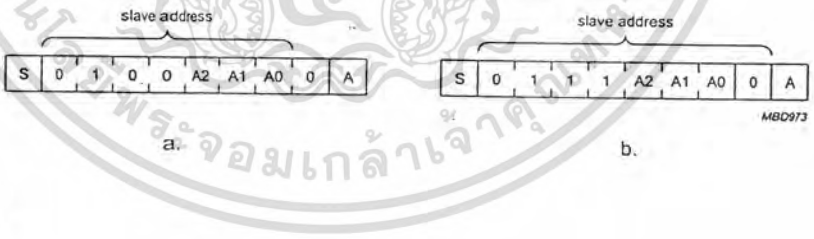


Fig.8 Simplified schematic diagram of each I/O.

7.1 Addressing

For addressing see Figs 9, 10 and 11.



(a) PCF8574.
(b) PCF8574A.

Fig.9 PCF8574 and PCF8574A slave addresses.

Each of the PCF8574's eight I/Os can be independently used as an input or output. Input data is transferred from the port to the microcontroller by the READ mode (see Fig.11). Output data is transmitted to the port by the WRITE mode (see Fig.10).

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

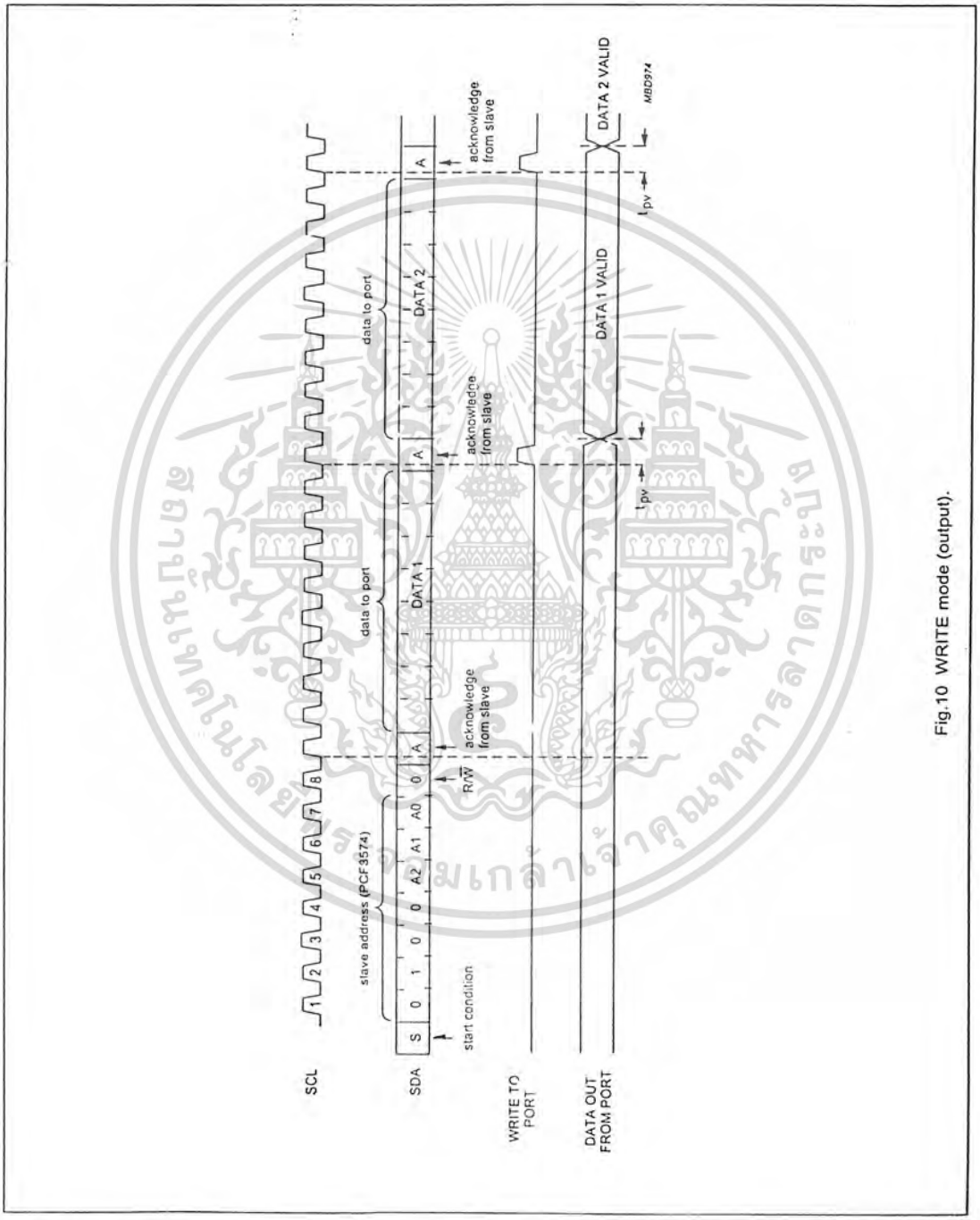
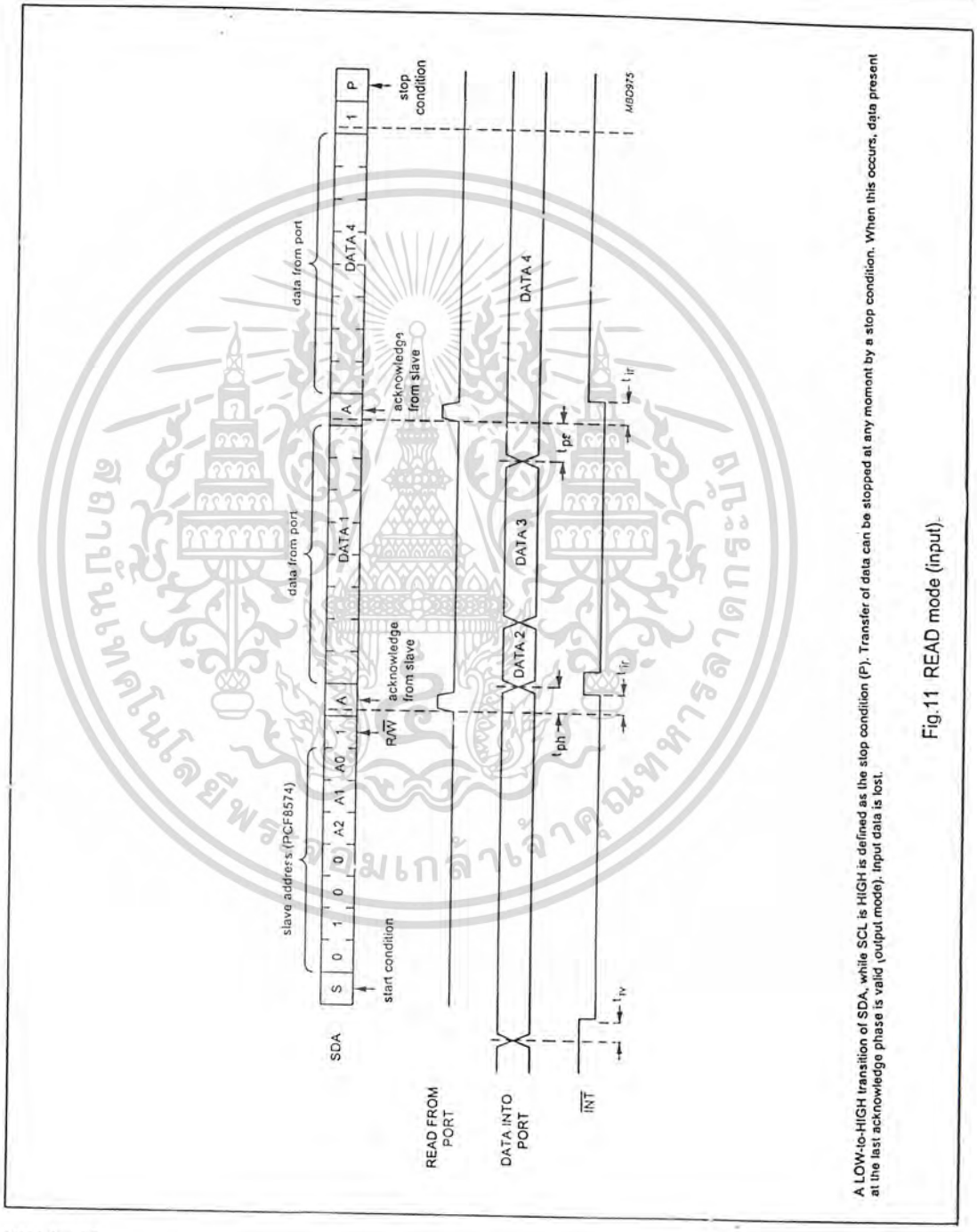


Fig.10 WRITE mode (output).

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



A LOW-to-HIGH transition of SDA, while SCL is HIGH is defined as the stop condition (P). Transfer of data can be stopped at any moment by a stop condition. When this occurs, data present at the last acknowledge phase is valid (output mode). Input data is lost.

Fig.11 READ mode (input).

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

7.2 Interrupt (see Figs 12 and 13)

The PCF8574 provides an open drain output ($\overline{INT}$) which can be fed to a corresponding input of the microcontroller. This gives these chips a type of master function which can initiate an action elsewhere in the system.

An interrupt is generated by any rising or falling edge of the port inputs in the input mode. After time t_{iv} the signal $\overline{INT}$ is valid.

Resetting and reactivating the interrupt circuit is achieved when data on the port is changed to the original setting or data is read from or written to the port which has generated the interrupt.

Resetting occurs as follows:

- In the READ mode at the acknowledge bit after the rising edge of the SCL signal
- In the WRITE mode at the acknowledge bit after the HIGH-to-LOW transition of the SCL signal
- Interrupts which occur during the acknowledge clock pulse may be lost (or very short) due to the resetting of the interrupt during this pulse.

Each change of the I/Os after resetting will be detected and, after the next rising clock edge, will be transmitted as $\overline{INT}$. Reading from or writing to another device does not affect the interrupt circuit.

7.3 Quasi-bidirectional I/Os (see Fig.14)

A quasi-bidirectional I/O can be used as an input or output without the use of a control signal for data direction. At power-on the I/Os are HIGH. In this mode only a current source to V_{DD} is active. An additional strong pull-up to V_{DD} allows fast rising edges into heavily loaded outputs. These devices turn on when an output is written HIGH, and are switched off by the negative edge of SCL. The I/Os should be HIGH before being used as inputs.

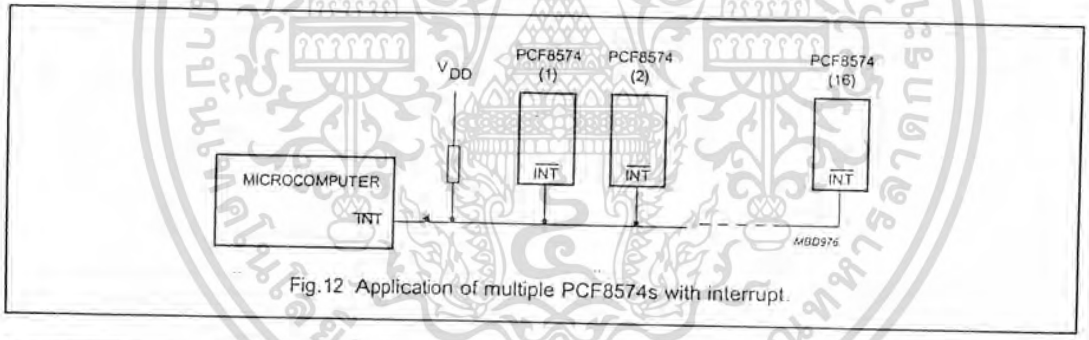


Fig.12 Application of multiple PCF8574s with interrupt.

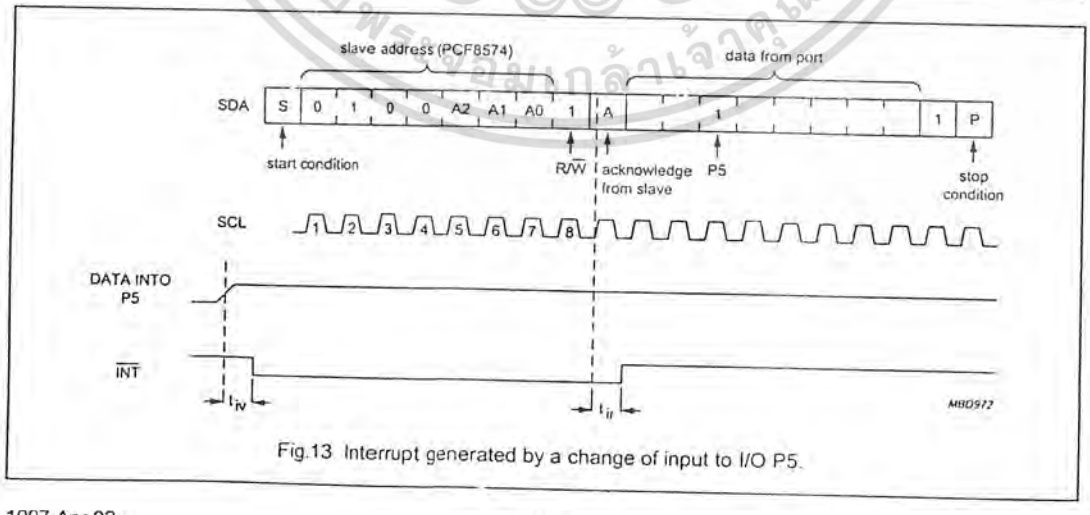


Fig.13 Interrupt generated by a change of input to I/O P5.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

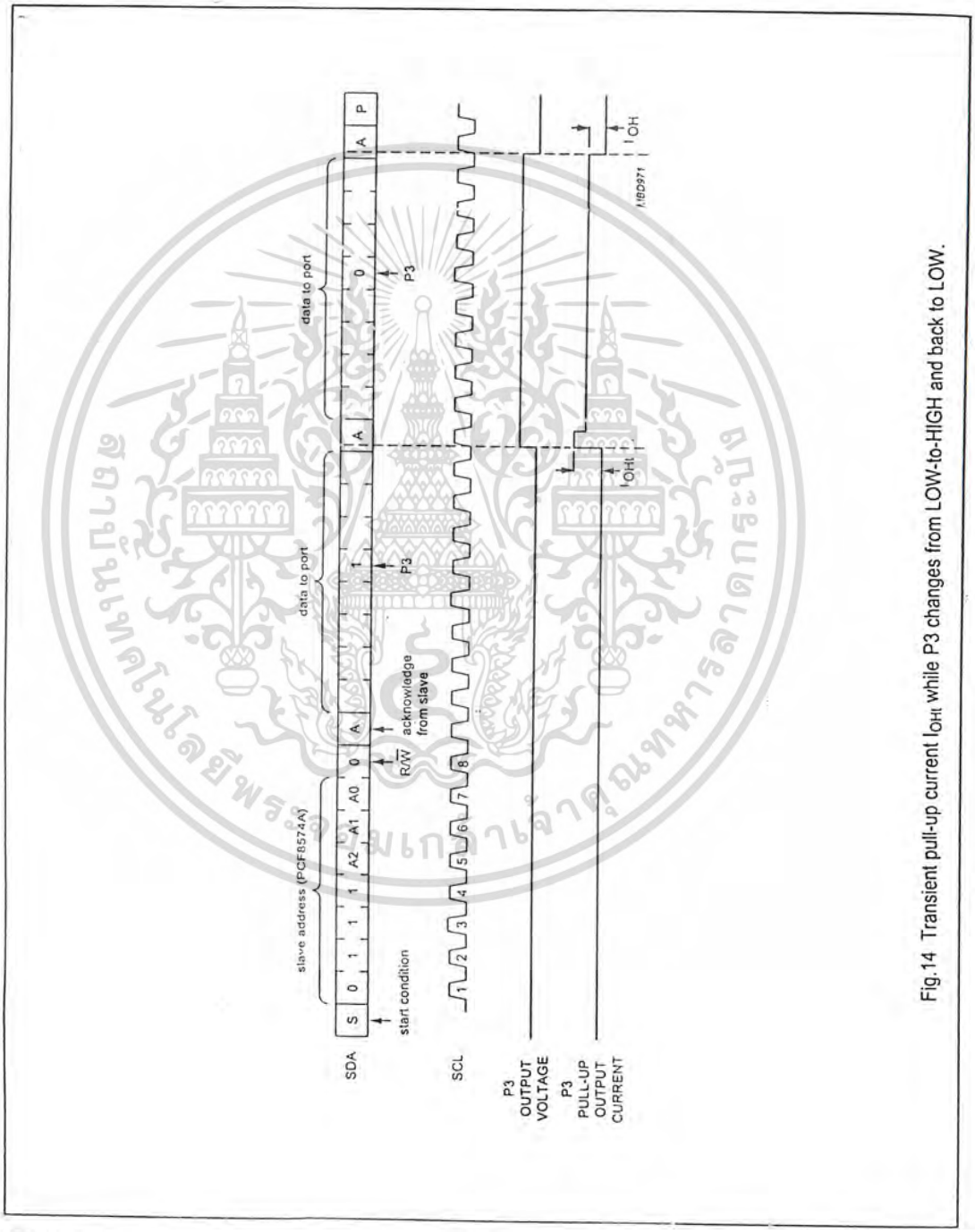


Fig.14 Transient pull-up current I_{OHt} while P3 changes from LOW-to-HIGH and back to LOW.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

8 LIMITING VALUES

In accordance with the Absolute Maximum Rating System (IEC 134).

SYMBOL	PARAMETER	MIN.	MAX.	UNIT
V_{DD}	supply voltage	−0.5	+7.0	V
V_I	input voltage	V_{SS} − 0.5	V_{DD} + 0.5	V
I_I	DC input current	−	±20	mA
I_O	DC output current	−	±25	mA
I_{DD}	supply current	−	±100	mA
I_{SS}	supply current	−	±100	mA
P_{tot}	total power dissipation	−	400	mW
P_O	power dissipation per output	−	100	mW
T_{stg}	storage temperature	−65	+150	°C
T_{amb}	operating ambient temperature	−40	+85	°C

9 HANDLING

Inputs and outputs are protected against electrostatic discharge in normal handling. However, to be totally safe, it is desirable to take precautions appropriate to handling MOS devices. Advice can be found in Data Handbook IC12 under *"Handling MOS Devices"*.

10 DC CHARACTERISTICS

V_{DD} = 2.5 to 6 V; V_{SS} = 0 V; T_{amb} = −40 to +85 °C; unless otherwise specified.

SYMBOL	PARAMETER	CONDITIONS	MIN.	TYP.	MAX.	UNIT
Supply						
V_{DD}	supply voltage		2.5	−	6.0	V
I_{DD}	supply current	operating mode; V_{DD} = 6 V; no load; V_I = V_{DD} or V_{SS}; f_{SCL} = 100 kHz	−	40	100	µA
I_{stb}	standby current	standby mode; V_{DD} = 6 V; no load; V_I = V_{DD} or V_{SS}	−	2.5	10	µA
V_{POR}	Power-on reset voltage	V_{DD} = 6 V; no load; V_I = V_{DD} or V_{SS}; note 1	−	1.3	2.4	V
Input SCL; input/output SDA						
V_{IL}	LOW level input voltage		−0.5	−	+0.3V_{DD}	V
V_{IH}	HIGH level input voltage		0.7V_{DD}	−	V_{DD} + 0.5	V
I_{OL}	LOW level output current	V_{OL} = 0.4 V	3	−	−	mA
I_L	leakage current	V_I = V_{DD} or V_{SS}	−1	−	+1	µA
C_i	input capacitance	V_I = V_{SS}	−	−	7	pF

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

SYMBOL	PARAMETER	CONDITIONS	MIN.	TYP.	MAX.	UNIT
I/Os						
V_{IL}	LOW level input voltage		−0.5	−	$+0.3V_{DD}$	V
V_{IH}	HIGH level input voltage		$0.7V_{DD}$	−	$V_{DD} + 0.5$	V
$I_{IHL(max)}$	maximum allowed input current through protection diode	$V_I \geq V_{DD}$ or $V_I \leq V_{SS}$	−	−	±400	μA
I_{OL}	LOW level output current	V_{OL} = 1 V; V_{DD} = 5 V	10	25	−	mA
I_{OH}	HIGH level output current	$V_{OH} = V_{SS}$	30	−	300	μA
I_{OHt}	transient puli-up current	HIGH during acknowledge (see Fig.14); $V_{OH} = V_{SS}$; V_{DD} = 2.5 V	−	−1	−	mA
C_i	input capacitance		−	−	10	pF
C_o	output capacitance		−	−	10	pF
Port timing; $C_L \leq 100$ pF (see Figs 10 and 11)						
t_{pv}	output data valid		−	−	4	μs
t_{su}	input data set-up time		0	−	−	μs
t_h	input data hold time		4	−	−	μs
Interrupt $\overline{INT}$ (see Fig.13)						
I_{OL}	LOW level output current	V_{OL} = 0.4 V	1.6	−	−	mA
I_L	leakage current	$V_I = V_{DD}$ or V_{SS}	−1	−	+1	μA
TIMING; $C_L \leq 100$ pF						
t_{iv}	input data valid time		−	−	4	μs
t_{ir}	reset delay time		−	−	4	μs
Select inputs A0 to A2						
V_{IL}	LOW level input voltage		−0.5	−	$+0.3V_{DD}$	V
V_{IH}	HIGH level input voltage		$0.7V_{DD}$	−	$V_{DD} + 0.5$	V
I_{LI}	input leakage current	pin at V_{DD} or V_{SS}	−250	−	+250	nA

Note

1. The Power-on reset circuit resets the I²C-bus logic with $V_{DD} < V_{POR}$ and sets all I/Os to logic 1 (with current source to V_{DD}).

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

11 I²C-BUS TIMING CHARACTERISTICS

SYMBOL	PARAMETER	MIN.	TYP.	MAX.	UNIT
I²C-BUS TIMING (see Fig.15; note 1)					
f_{SCL}	SCL clock frequency	–	–	100	kHz
t_{SW}	tolerable spike width on bus	–	–	100	ns
t_{BUF}	bus free time	4.7	–	–	µs
$t_{SU;STA}$	START condition set-up time	4.7	–	–	µs
$t_{HD;STA}$	START condition hold time	4.0	–	–	µs
t_{LOW}	SCL LOW time	4.7	–	–	µs
t_{HIGH}	SCL HIGH time	4.0	–	–	µs
t_r	SCL and SDA rise time	–	–	1.0	µs
t_f	SCL and SDA fall time	–	–	0.3	µs
$t_{SU;DAT}$	data set-up time	250	–	–	ns
$t_{HD;DAT}$	data hold time	0	–	–	ns
$t_{VD;DAT}$	SCL LOW to data out valid	–	–	3.4	µs
$t_{SU;STO}$	STOP condition set-up time	4.0	–	–	µs

Note

1. All the timing values are valid within the operating supply voltage and ambient temperature range and refer to V_{IL} and V_{IH} with an input voltage swing of V_{SS} to V_{DD}.

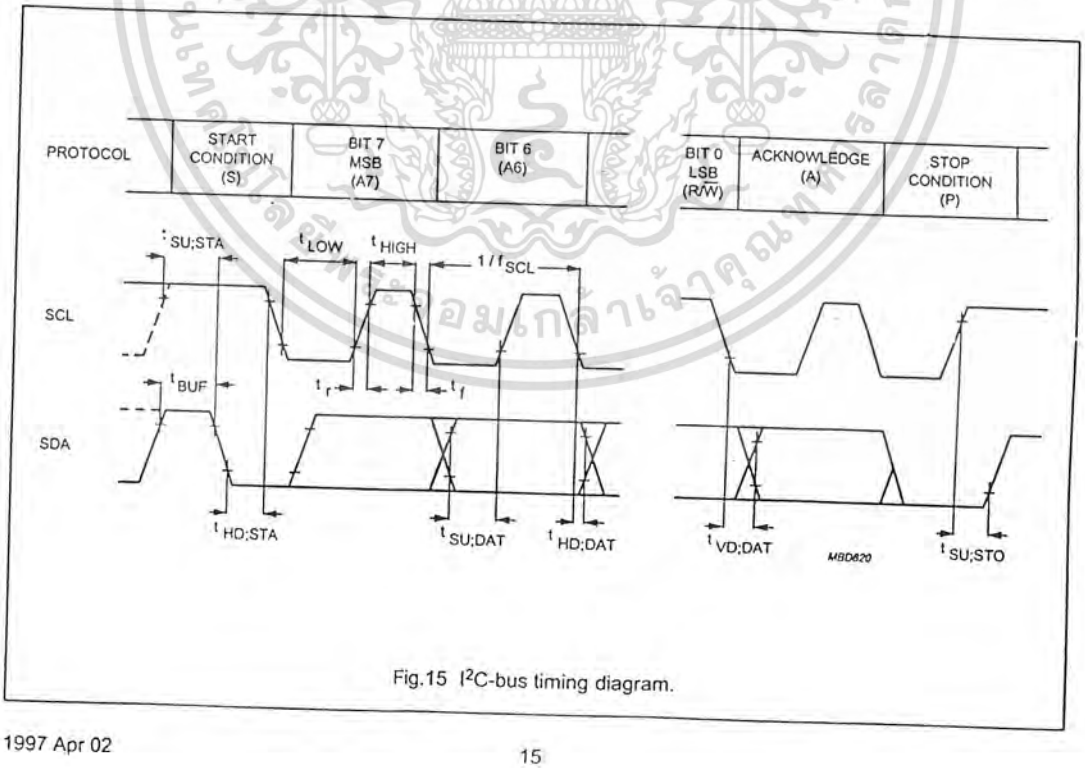
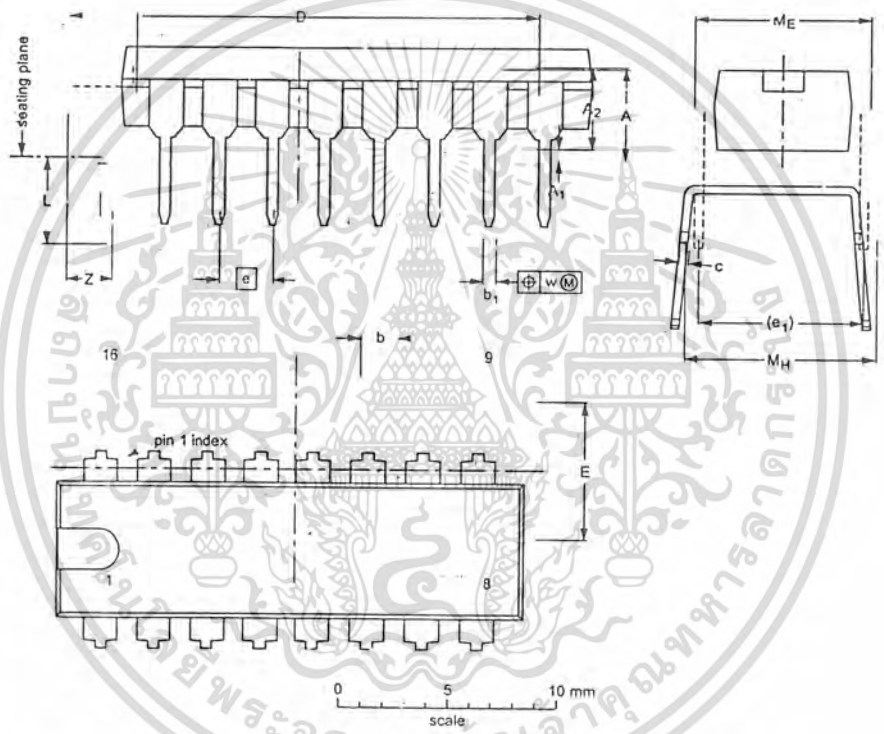


Fig.15 I²C-bus timing diagram.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

12 PACKAGE OUTLINES

DIP16: plastic dual in-line package; 16 leads (300 mil); long body SOT38-1



DIMENSIONS (inch dimensions are derived from the original mm dimensions)

UNIT	A max.	A_1 min.	A_2 max.	b	b_1	c	D (1)	E (1)	e	e_1	L	M_E	M_H	w	Z (1) max.
mm	4.7	0.51	3.7	1.40 1.14	0.53 0.38	0.32 0.23	21.8 21.4	6.48 6.20	2.54	7.62	3.9 3.4	8.25 7.80	9.5 8.3	0.254	2.2
inches	0.19	0.020	0.15	0.055 0.045	0.021 0.015	0.013 0.009	0.86 0.84	0.26 0.24	0.10	0.30	0.15 0.13	0.32 0.31	0.37 0.33	0.01	0.087

Note

1. Plastic or metal protrusions of 0.25 mm maximum per side are not included.

OUTLINE VERSION	REFERENCES				EUROPEAN PROJECTION	ISSUE DATE
	IEC	JEDEC	EIAJ			
SOT38-1	050G09	MO-001AE				~~92-10-02~~ 95-01-19

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Octuple 6-bit DACs with I²C-bus

TDA8444; TDA8444T; TDA8444AT

FEATURES

- Eight DACs with 6-bit resolution
- Adjustable common output swing
- Push-pull outputs
- Outputs short-circuit protected
- Three programmable slave address bits
- Large supply voltage range
- Low temperature coefficient.

GENERAL DESCRIPTION

The interface circuit is a bipolar IC in a DIP16, SO16, or SO20 package made in an I2L-compatible 13 V process.

The TDA8444 contains eight programmable 6-bit DAC outputs, an I²C-bus slave receiver with three (two for SO16) programmable address bits and one input (V_{MAX}) to set the maximum output voltage. Each DAC can be programmed separately by a 6-bit word to 64 values, but V_{MAX} determines the maximum output voltage for all DACs. The resolution will be approximately $\frac{1}{64}V_{MAX}$.

At power-on all DACs are set to their lowest value.

QUICK REFERENCE DATA

SYMBOL	PARAMETER	CONDITIONS	MIN.	TYP.	MAX.	UNIT
V_{CC}	supply voltage		4.5	12	13.2	V
I_{CC}	supply current	V_{CC} = 12 V	–	14	–	mA
P	power dissipation		–	170	–	mW
V_{VMAX}	input effective voltage		1	–	V_{CC} – 2.0	V
$V_{o(DACn)}$	DAC output voltage	$V_{MAX} = V_{CC}$	0.1	–	V_{CC} – 0.5	V
$V_{o(DACn)(max)}$	maximum DAC output voltage	$1 < V_{MAX} < V_{CC} - 2.0$	–	V_{MAX} + 0.3	–	V
$I_{source(min)}$	minimum DAC source current	data = 1FH	2	–	–	mA
$I_{sink(min)}$	minimum DAC sink current	data = 1FH	2	–	–	mA

ORDERING INFORMATION

TYPE NUMBER	PACKAGE		
	NAME	DESCRIPTION	VERSION
TDA8444	DIP16	plastic dual in-line package; 16 leads (300 mil); long body	SOT38-1
TDA8444T	SO16	plastic small outline package; 16 leads; body width 7.5 mm	SOT162-1
TDA8444AT	SO20	plastic small outline package; 20 leads; body width 7.5 mm	SOT163-1

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

BLOCK DIAGRAM

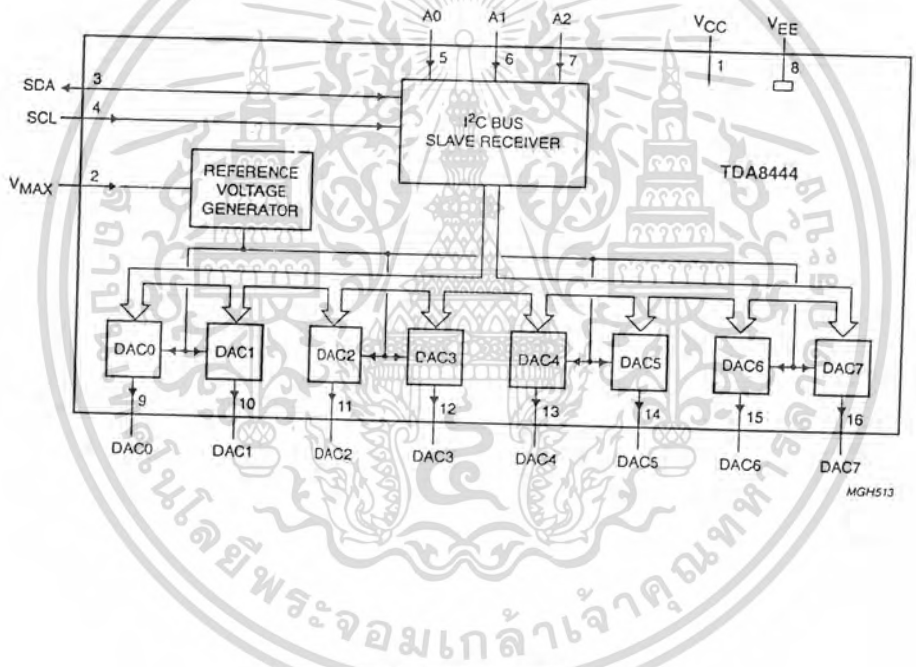


Fig.1 Block diagram.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

PINNING

SYMBOL	PIN			DESCRIPTION
	TDA8444 (DIP16)	TDA8444T (SO16)	TDA8444AT (SO20)	
V_{CC}	1	1	1	supply voltage
V_{MAX}	2	2	2	control input for DAC maximum output voltage
SDA	3	3	3	I²C-bus serial data input/output
SCL	4	4	4	I²C-bus serial clock
A0	5	6	7	programmable address bit 0 for I²C-bus slave receiver
A1	6	7	8	programmable address bit 1 for I²C-bus slave receiver
A2	7	–	9	programmable address bit 2 for I²C-bus slave receiver
V_{EE}	8	8	10	ground
DAC0	9	9	11	analog voltage output 0
DAC1	10	10	13	analog voltage output 1
DAC2	11	11	14	analog voltage output 2
DAC3	12	12	15	analog voltage output 3
DAC4	13	13	16	analog voltage output 4
DAC5	14	14	17	analog voltage output 5
DAC6	15	15	18	analog voltage output 6
DAC7	16	16	20	analog voltage output 7
n.c.	–	5	5, 6, 12, 19	not connected

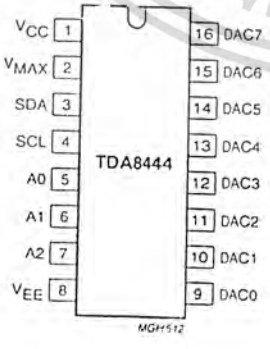


Fig.2 Pin configuration (TDA8444; DIP16).

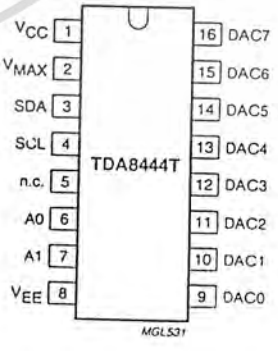


Fig.3 Pin configuration (TDA8444T; SO16).

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

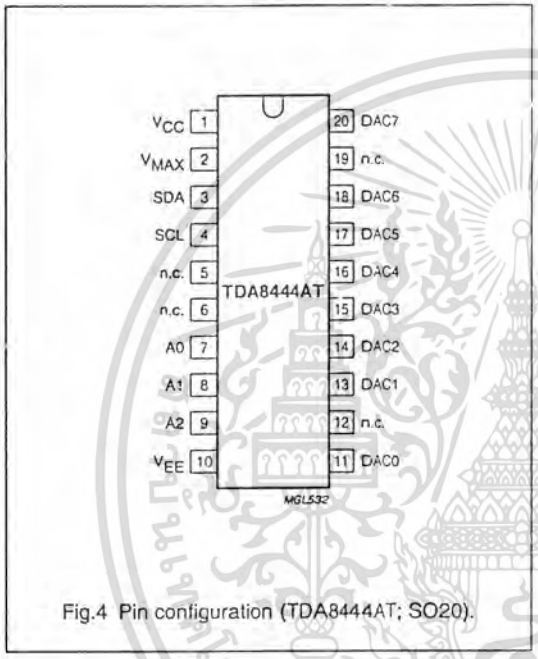


Fig.4 Pin configuration (TDA8444AT; SO20).

FUNCTIONAL DESCRIPTION

I²C-bus interface

The I²C-bus interface is a receive-only slave, which accepts data according the format shown in Table 1.

Table 1 I²C-bus format (see note 1)

S	0	1	0	0	A2	A1	A0	0	A	I3	I2	I1	I0	SD	SC	SB	SA	A	X	X	D5	D4	D3	D2	D1	D0	A	P

Note

1. S = START condition; A2 to A0 = programmable address bits; A = Acknowledge; I3 to I0 = Instruction bits; SD to SA = subaddress bits; X = don't care; D5 to D0 = data bits; P = STOP condition.

Valid addresses are:

TDA8444 and TDA8444AT: 40H, 42H, 44H, 46H, 48H, 4AH, 4CH and 4EH

TDA8444T: 48H, 4AH, 4CH and 4EH (A2 is always logic 1).

All other addresses cannot be acknowledged by the circuit. The actual slave address depends on the programmable address bits A2, A1 and A0. This way up to eight circuits can be used on one I²C-bus.

Valid instructions are: 00H to 0FH; F0H to FFH.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

The circuit will not react to other combinations of the 4 instruction bits I3 to I0 than 0 or F, but will still generate an acknowledge. The difference between instruction 0 and F is only important when more than one data byte is sent within one transmission. Instruction 0 causes the data bytes to be written into the DAC-latches with consecutive subaddresses starting with the subaddress given in the instruction byte (auto-increment of subaddress), while instruction F will cause a consecutive writing of the data bytes into the same DAC-latch whose subaddress was given in the instruction byte. In case of only one data byte the DAC-latch with the subaddress equal to the subaddress in the instruction byte will receive the data.

Valid subaddresses are: 0H to 7H.

The subaddresses correspond to DAC0 to DAC7. The Auto-Increment (AI) function of instruction 0, however, works on all possible subaddresses 0 to F in such a way that next to subaddress F, subaddress 0 will follow, and so on.

The data will be latched into the DAC-latch on the positive-going edge of the acknowledge related clock pulse.

The specification of the SCL and SDA I/O meets the I²C-bus specification. For protection against positive voltage pulses on pins 3 and 4, zener diodes are connected between these pins and V_{EE}. This means that normal bus line voltage should not exceed 5.5 V.

The address inputs A0, A1 and A2 can be easily programmed by either a connection to V_{EE} (An = 0) or V_{CC} (An = 1). If the inputs are left floating the result will be An = 1.

V_{MAX}

The V_{MAX} input gives a means of compressing the DAC output voltage swing. The maximum DAC output voltage will be equal to $V_{MAX} + V_{DAC(min)}$, while the 6-bit resolution is maintained. This enables a higher voltage resolution for smaller output swings.

DACs

The DACs consist of a 6-bit data-latch, current switches and an opamp. The current sources connected to the switches have values with weights 2^0 to 2^5. The sum of the switched on currents is converted by the opamp into a voltage between approximately 0.5 and 10.5 V if $V_{MAX} = V_{CC} = 12$ V. The DAC outputs are short-circuit protected against V_{CC} and V_{EE}. Capacitive load on the DAC outputs should not exceed 2 nF in order to prevent possible oscillations at certain levels. The temperature coefficient for each of the outputs remains in all possible conditions well below 0.1 LSB per Kelvin.

LIMITING VALUES

In accordance with the Absolute Maximum Rating System (IEC 134).

SYMBOL	PARAMETER	MIN.	MAX.	UNIT
V_{CC}	supply voltage	−0.5	+18	V
I_{CC}	supply current	−10	+40	mA
$P_{(max)}$	maximum power dissipation	−	500	mW
$V_{i(n)}$	input voltage	−0.5	+5.9	V
	pins SDA and SCL	−0.5	+5.9	V
	pins V_{MAX}, A0 to A2 and DAC0 to DAC7	−0.5	V_{CC} + 0.5	V
I_n	current in all pins except V_{CC} and V_{EE}	−	±10	mA
T_{stg}	storage temperature	−65	+150	°C
T_{amb}	operating ambient temperature	−20	+70	°C

QUALITY SPECIFICATION

In accordance with *"SNW-FQ-611-E"*.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

THERMAL CHARACTERISTICS

SYMBOL	PARAMETER	CONDITIONS	VALUE	UNIT
$R_{th(j-a)}$	thermal resistance from junction to ambient	in free air		
	TDA8444		75	K/W
	TDA8444T	note 1	100	K/W
	TDA8444AT	note 1	85	K/W

Note

1. When mounted on a Printed-Circuit Board (PCB).

CHARACTERISTICS

V_{CC} = 12 V; T_{amb} = 25 °C; unless otherwise specified.

SYMBOL	PARAMETER	CONDITIONS	MIN.	TYP.	MAX.	UNIT
Supply						
V_{CC}	supply voltage		4.5	12	13.2	V
I_{CC}	supply current	$V_{MAX} = V_{CC}$ = 12 V; data = 00H	12	14	19	mA
P	power dissipation		–	170	250	mW
V_{rst}	power reset voltage		1	–	4	V
Pin V_{MAX}						
$V_{i(VMAX)}$	input effective voltage		1	–	V_{CC} – 2.0	V
I_i	input current	$V_{MAX} = V_{CC}$	–	–	10	µA
		V_{MAX} = 1 V	–	–	10	µA
Pins SDA and SCL						
V_I	input voltage		0	–	5.5	V
V_{IL}	LOW-level input voltage		–	–	1.0	V
V_{IH}	HIGH-level input voltage		3.0	–	–	V
I_{IL}	LOW-level input current	$V_{SDA} = V_{SCL}$ = –0.3 V	–	–	–10	µA
I_{IH}	HIGH-level input current	$V_{SDA} = V_{SCL}$ = 6 V	–	–	±10	µA
PIN SDA						
V_{OL}	LOW-level output voltage	I_L = 3 mA	–	–	0.4	V
$I_{o(sink)}$	output sink current		3	8	–	mA
Address bits (A0 to A2)						
V_I	input voltage		0	–	V_{CC}	V
V_{IL}	LOW-level input voltage		–	–	1.0	V
V_{IH}	HIGH-level input voltage		2.2	–	–	V
I_{IL}	LOW-level input current	$V_{An} = V_{EE}$	–10	–15	–	µA
I_{IH}	HIGH-level input current	$V_{An} = V_{CC}$	–	–	1	µA

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

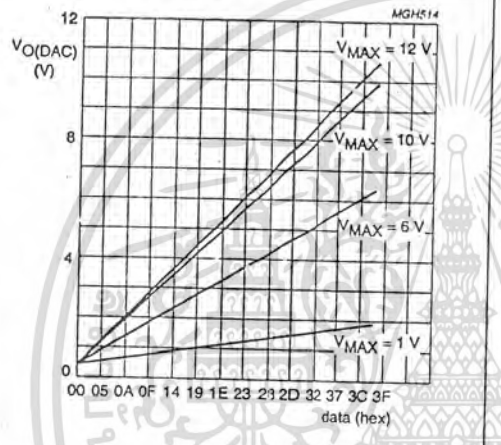
SYMBOL	PARAMETER	CONDITIONS	MIN.	TYP.	MAX.	UNIT
DACs (DAC0 to DAC7)						
V_o	DAC output voltage	$V_{MAX} = V_{CC}$	0.1	–	$V_{CC} - 0.5$	V
$V_{o(min)}$	minimum output voltage	data = 00H; $I_L = -2$ mA	0.1	0.28	0.5	V
$V_{o(max)}$	maximum output voltage	data = 3FH; $I_L = -2$ mA				
		$V_{MAX} = V_{CC}$	10.0	10.5	11.5	V
		$1 < V_{MAX} < 10$ V	–	note 1	–	V
$I_{o(sink)}$	output sink current	$V_{DAC} = V_{CC}$; data = 1FH	2	8	15	mA
$I_{o(source)}$	output source current	$V_{DAC} = V_{EE}$; data = 1FH	−2	–	−6	mA
Z_o	output impedance	$-2 \leq I_L \leq +2$ mA; data = 1FH	–	4	50	Ω
DNL	differential non-linearity	$V_{MAX} = V_{CC}$; $I_L = -2$ mA	–	–	±0.5	LSB
INL	integral non-linearity	$V_{MAX} = V_{CC}$; $I_L = -2$ mA	–	–	±0.5	LSB
ΔG_{FS}	DC gain match at full-scale	data = 3FH; $I_L = -2$ mA	–	–	5	%
$\Delta G/\Delta$data	DC gain versus other DAC data change	data = 3FH; $I_L = -2$ mA	–	<±0.5	–	LSB
TC	temperature coefficient	data = 3FH; $I_L = -2$ mA	–	<±0.1	–	LSB/K

Note

1. The output voltage is typically: $\frac{V_{swing}}{(V_{CC} - 2.0)} \times V_{MAX} + V_{o(00H)}$ with $V_{swing} = V_{o(3FH)} - V_{o(00H)}$ for $V_{MAX} = V_{CC}$.

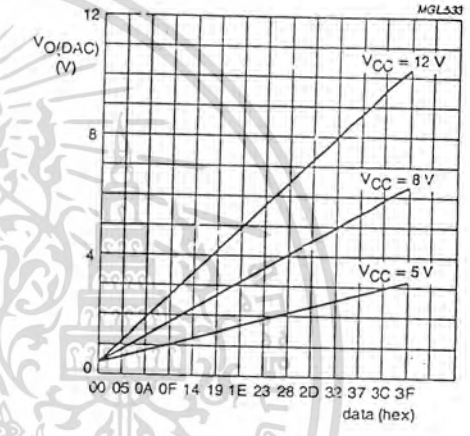
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

TEST AND APPLICATION INFORMATION



V_{CC} = 12 V.

Fig.5 DAC output voltage as a function of programmed value.



$V_{MAX} = V_{CC}$.

Fig.6 DAC output voltage as a function of programmed value.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

SIEMENS

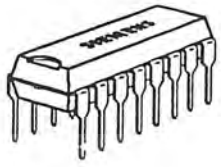
Phase Control IC

TCA 785

Bipolar IC

Features

- Reliable recognition of zero passage
- Large application scope
- May be used as zero point switch
- LSL compatible
- Three-phase operation possible (3 ICs)
- Output current 250 mA
- Large ramp current range
- Wide temperature range

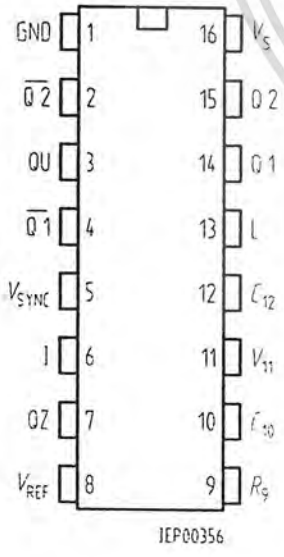


P-DIP-16-1

Type	Ordering Code	Package
TCA 785	Q67000-A2321	P-DIP-16-1

This phase control IC is intended to control thyristors, triacs, and transistors. The trigger pulses can be shifted within a phase angle between 0 ° and 180 °. Typical applications include converter circuits, AC controllers and three-phase current controllers.

This IC replaces the previous types TCA 780 and TCA 780 D.



IEP00356

Pin Configuration
(top view)

Pin Definitions and Functions

Pin	Symbol	Function
1	GND	Ground
2	$\overline{Q2}$	Output 2 inverted
3	Q U	Output U
4	$\overline{Q2}$	Output 1 inverted
5	V_{SYNC}	Synchronous voltage
6	I	Inhibit
7	Q Z	Output Z
8	V_{REF}	Stabilized voltage
9	R_9	Ramp resistance
10	C_{10}	Ramp capacitance
11	V_{11}	Control voltage
12	C_{12}	Pulse extension
13	L	Long pulse
14	Q 1	Output 1
15	Q 2	Output 2
16	V_S	Supply voltage

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่อนุญาตให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Functional Description

The synchronization signal is obtained via a high-ohmic resistance from the line voltage (voltage V_5). A zero voltage detector evaluates the zero passages and transfers them to the synchronization register.

This synchronization register controls a ramp generator, the capacitor C_{10} of which is charged by a constant current (determined by R_9). If the ramp voltage V_{10} exceeds the control voltage V_{11} (triggering angle φ), a signal is processed to the logic. Dependent on the magnitude of the control voltage V_{11}, the triggering angle φ can be shifted within a phase angle of 0° to 180°.

For every half wave, a positive pulse of approx. 30 µs duration appears at the outputs Q 1 and Q 2. The pulse duration can be prolonged up to 180° via a capacitor C_{12}. If pin 12 is connected to ground, pulses with a duration between φ and 180° will result.

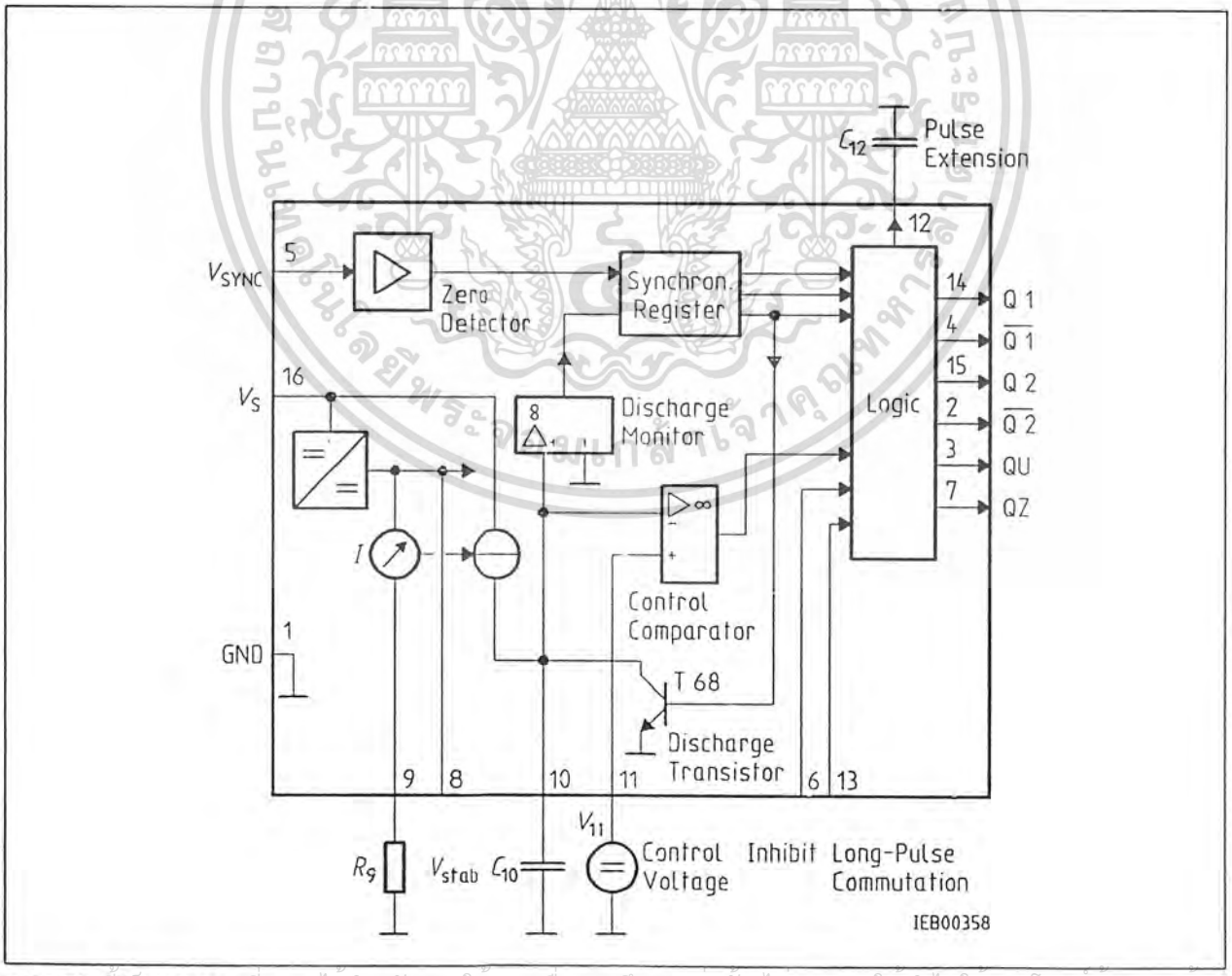
Outputs $\overline{Q1}$ and $\overline{Q2}$ supply the inverse signals of Q 1 and Q 2.

A signal of φ +180° which can be used for controlling an external logic,is available at pin 3.

A signal which corresponds to the NOR link of Q 1 and Q 2 is available at output Q Z (pin 7).

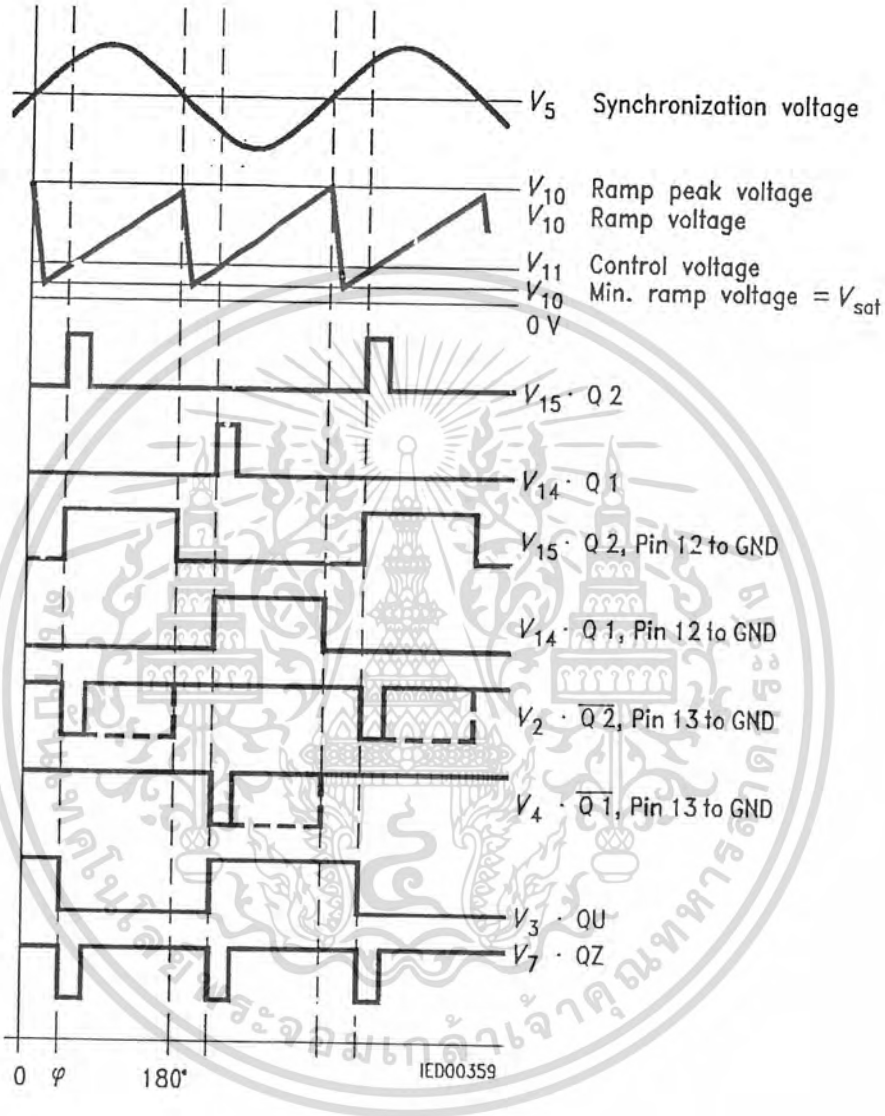
The inhibit input can be used to disable outputs Q1, Q2 and $\overline{Q1}$, $\overline{Q2}$.

Pin 13 can be used to extend the outputs $\overline{Q1}$ and $\overline{Q2}$ to full pulse length (180° – φ).



Block Diagram

เอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



Pulse Diagram

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Absolute Maximum Ratings

Parameter	Symbol	Limit Values		Unit
		min.	max.	
Supply voltage	V_S	− 0.5	18	V
Output current at pin 14, 15	I_Q	− 10	400	mA
Inhibit voltage	V_6	− 0.5	V_S	V
Control voltage	V_{11}	− 0.5	V_S	V
Voltage short-pulse circuit	V_{13}	− 0.5	V_S	V
Synchronization input current	V_5	− 200	± 200	µA
Output voltage at pin 14, 15	V_Q		V_S	V
Output current at pin 2, 3, 4, 7	I_Q		10	mA
Output voltage at pin 2, 3, 4, 7	V_Q		V_S	V
Junction temperature	T_j		150	°C
Storage temperature	T_{stg}	− 55	125	°C
Thermal resistance system - air	$R_{th\,SA}$		80	K/W

Operating Range

Supply voltage	V_S	8	18	V
Operating frequency	f	10	500	Hz
Ambient temperature	T_A	− 25	85	°C

Characteristics

$8 \le V_S \le 18$ V; -25 °C $\le T_A \le 85$ °C; $f = 50$ Hz

Parameter	Symbol	Limit Values			Unit	Test Circuit
		min.	typ.	max.		
Supply current consumption S1 ... S6 open $V_{11} = 0$ V $C_{10} = 47$ nF; $R_9 = 100$ kΩ	I_S	4.5	6.5	10	mA	1
Synchronization pin 5 Input current R_2 varied	$I_{5\,rms}$	30		200	µA	1
Offset voltage	ΔV_5		30	75	mV	4
Control input pin 11 Control voltage range	V_{11}	0.2		$V_{10\,peak}$	V	1
Input resistance	R_{11}		15		kΩ	5

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Characteristics (cont'd)
$8 \le V_S \le 18$ V; -25 °C $\le T_A \le 85$ °C; $f = 50$ Hz

Parameter	Symbol	Limit Values			Unit	Test Circuit
		min.	typ.	max.		
Ramp generator						
Charge current	I_{10}	10		1000	µA	
Max. ramp voltage	V_{10}			$V_2 - 2$	V	1
Saturation voltage at capacitor	V_{10}	100	225	350	mV	1.6
Ramp resistance	R_9	3		300	kΩ	1
Sawtooth return time	t_f		80		µs	1
Inhibit pin 6						
switch-over of pin 7						
Outputs disabled	V_{6L}		3.3	2.5	V	1
Outputs enabled	V_{6H}	4	3.3		V	1
Signal transition time	t_r	1		5	µs	1
Input current	I_{6H}		500	800	µA	1
$V_6 = 8$ V						
Input current	$-I_{6L}$	80	150	200	µA	1
$V_6 = 1.7$ V						
Deviation of I_{10}	I_{10}	-5		5	%	1
R_9 = const.						
$V_S = 12$ V; $C_{10} = 47$ nF						
Deviation of I_{10}	I_{10}	-20		20	%	1
R_9 = const.						
$V_S = 8$ V to 18 V						
Deviation of the ramp voltage						
between 2 following						
half-waves, V_S = const.	$\Delta V_{10\,max}$		± 1		%	
Long pulse switch-over						
pin 13						
switch-over of S8						
Short pulse at output	V_{13H}	3.5	2.5		V	1
Long pulse at output	V_{13L}		2.5	2	V	1
Input current	I_{13H}			10	µA	1
$V_{13} = 8$ V						
Input current	$-I_{13L}$	45	65	100	µA	1
$V_{13} = 1.7$ V						
Outputs pin 2, 3, 4, 7						
Reverse current	I_{CEO}			10	µA	2.6
$V_Q = V_S$						
Saturation voltage	V_{sat}	0.1	0.4	2	V	2.6
$I_Q = 2$ mA						

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Characteristics (cont'd)
$8 \le V_S \le 18$ V; -25 °C $\le T_A \le 85$ °C; $f = 50$ Hz

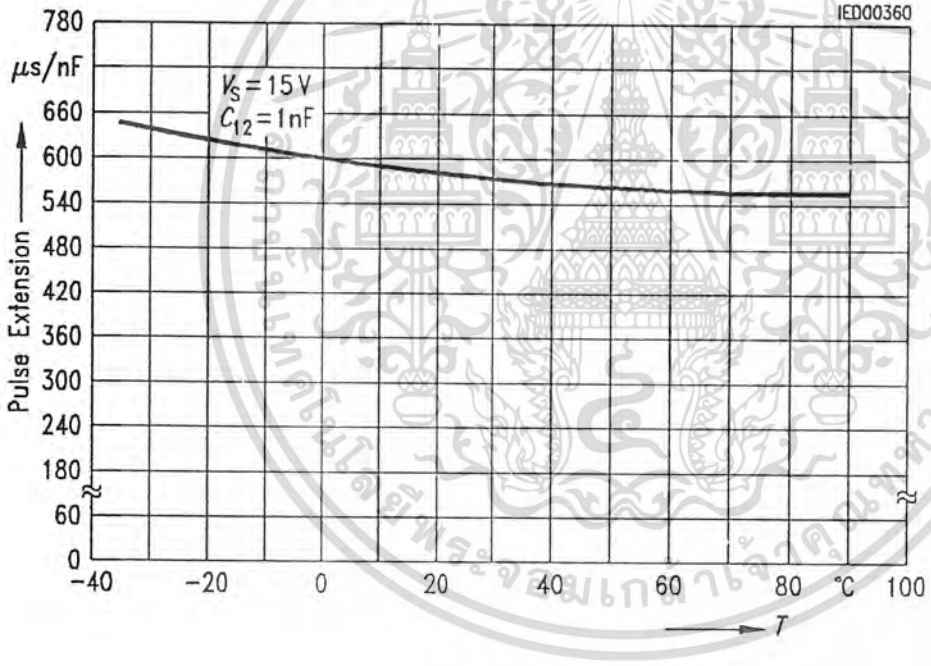
Parameter	Symbol	Limit Values			Unit	Test Circuit
		min.	typ.	max.		
Outputs pin 14, 15						
H-output voltage $-I_Q = 250$ mA	$V_{14/15\,H}$	$V_S - 3$	$V_S - 2.5$	$V_S - 1.0$	V	3.6
L-output voltage $I_Q = 2$ mA	$V_{14/15\,L}$	0.3	0.8	2	V	2.6
Pulse width (short pulse) S9 open	t_p	20	30	40	µs	1
Pulse width (short pulse) with C_{12}	t_p	530	620	760	µs/nF	1
Internal voltage control						
Reference voltage Parallel connection of 10 ICs possible	V_{REF}	2.8	3.1	3.4	V	1
TC of reference voltage	α_{REF}		2×10^{-4}	5×10^{-4}	1/K	1

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Application Hints for External Components

		min	max	
Ramp capacitance	C_{10}	500 pF	1 µF[1]	The minimum and maximum values of I_{10} are to be observed
Triggering point	$t_{Tr} = \frac{V_{11} \times R_9 \times C_{10}}{V_{REF} \times K}$		[2]	
Charge current	$I_{10} = \frac{V_{REF} \times K}{R_9}$		[2]	Ramp voltage $V_{10\,max} = V_S - 2\,V$ $V_{10} = \frac{V_{REF} \times K \times t}{R_9 \times C_{10}}$ [2]

Pulse Extension versus Temperature

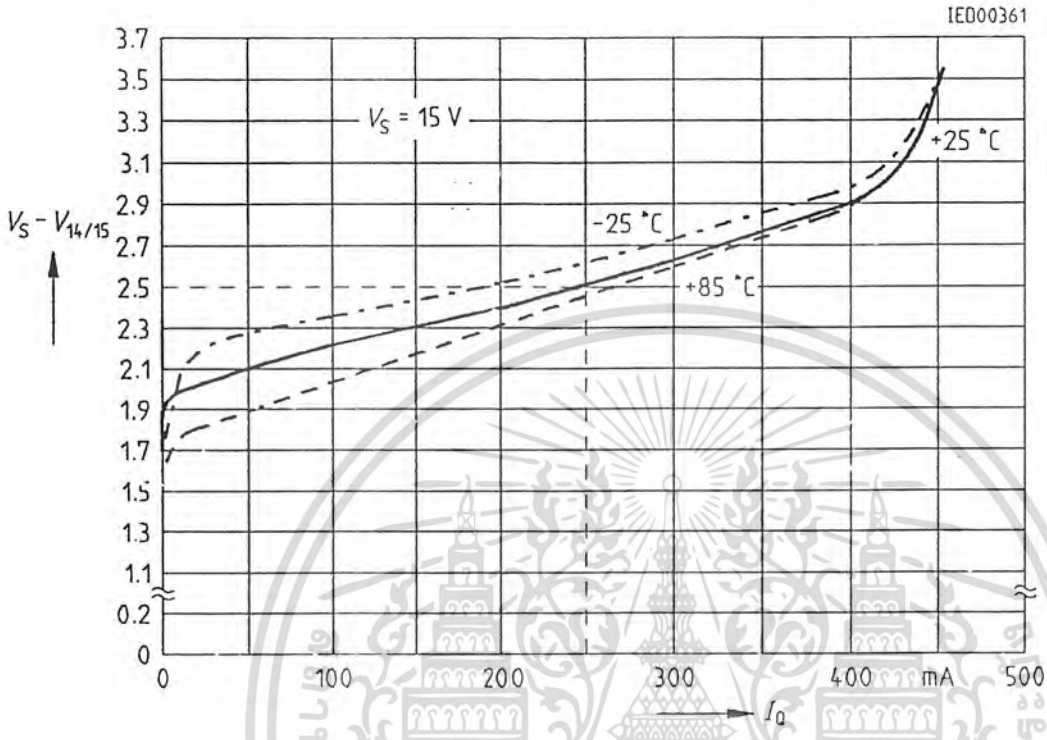


[1] Attention to flyback times

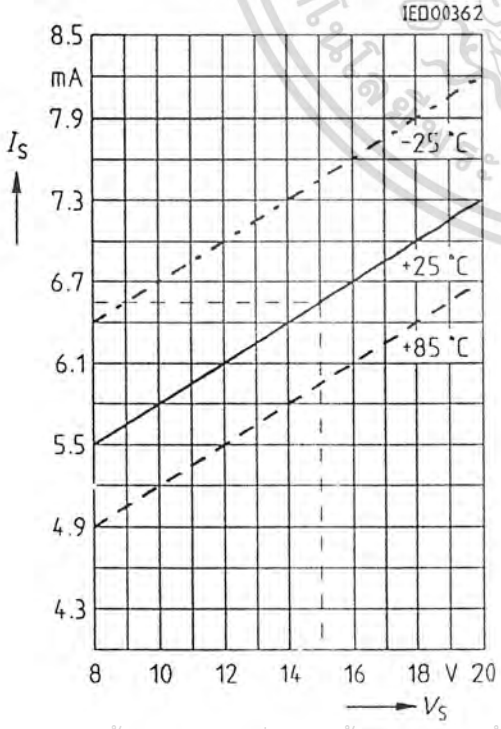
[2] $K = 1.10 \pm 20\,\%$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

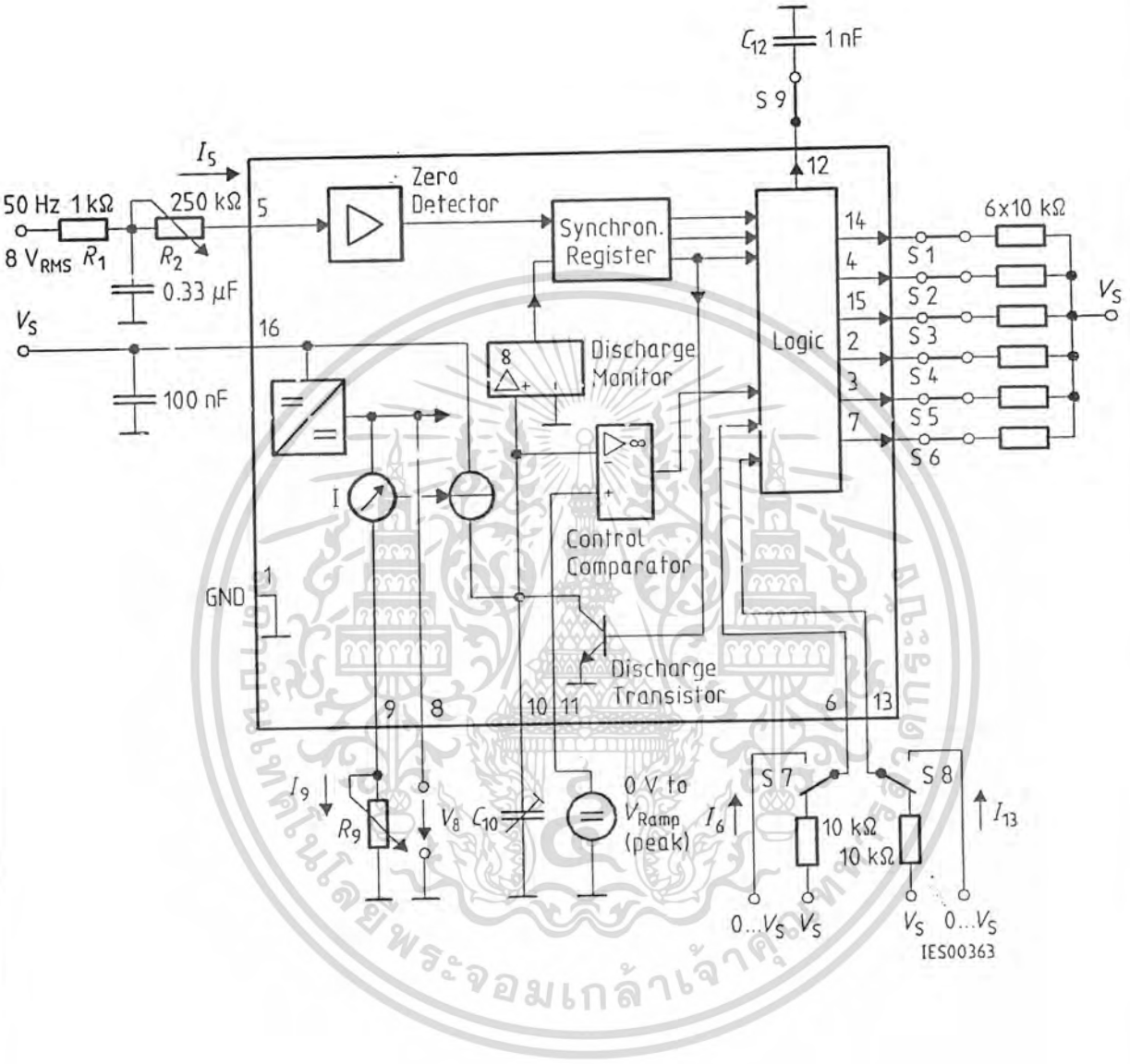
Output Voltage measured to + V_S



Supply Current versus Supply Voltage



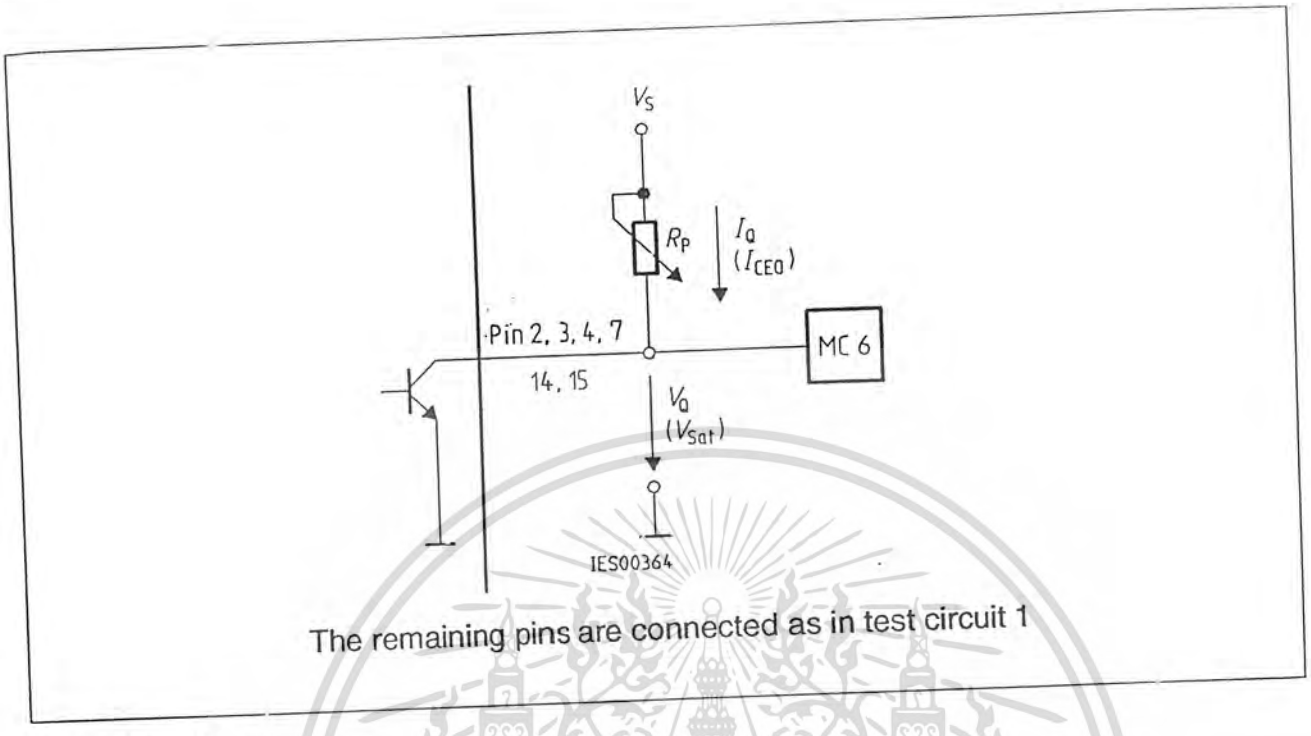
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



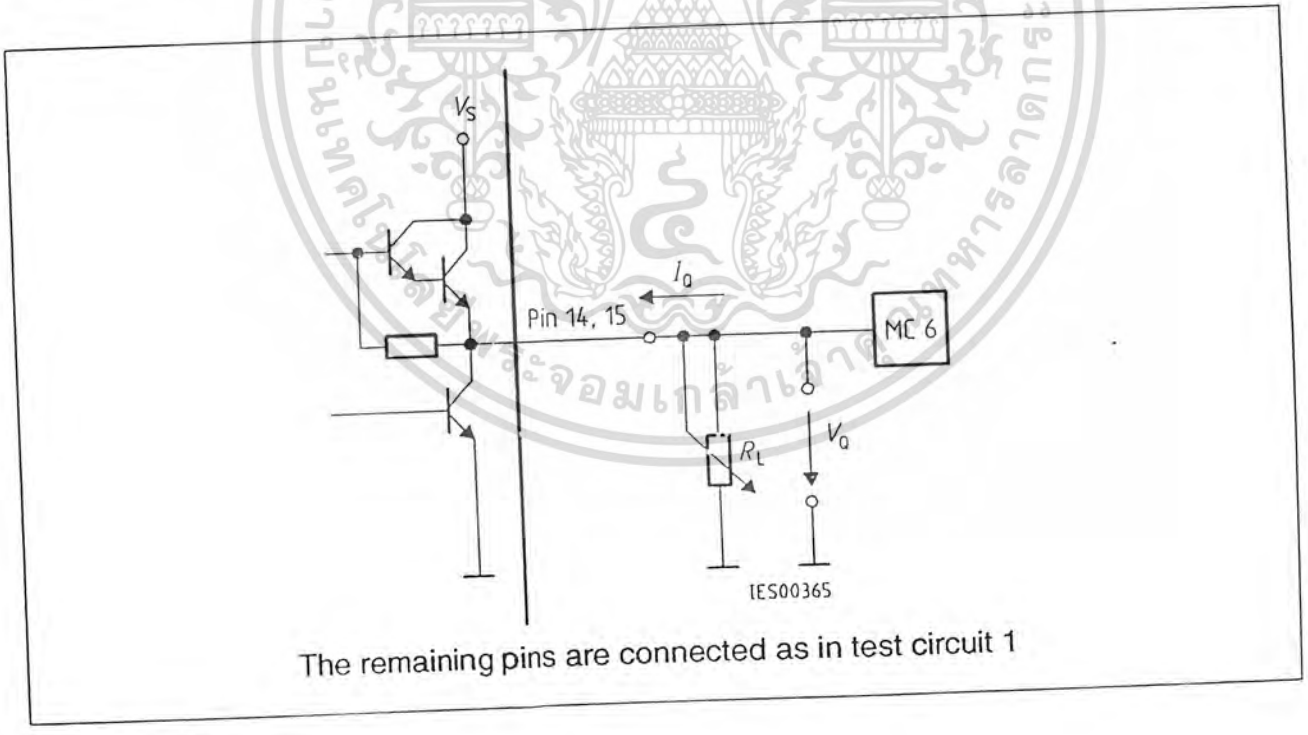
It is necessary for all measurements to adjust the ramp with the aid of C_{10} and R_9 in the way that $3\,V \leq V_{ramp\ max} \leq V_S - 2\,V$
e.g. C_{10} = 47 nF; 18 V: R_9 = 47 kΩ; 8 V: R_9 = 120 kΩ

Test Circuit 1

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

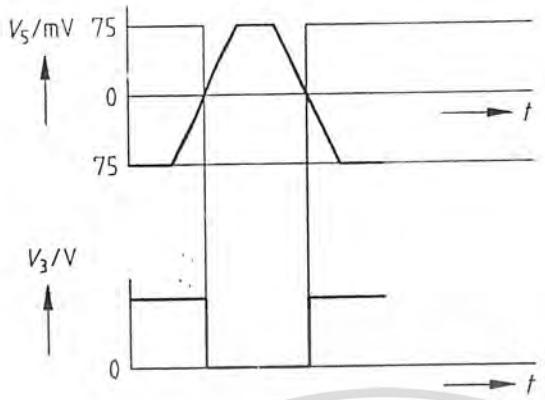


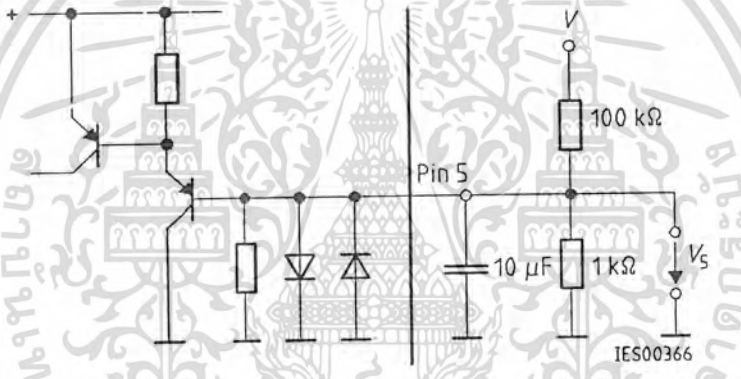
Test Circuit 2



Test Circuit 3

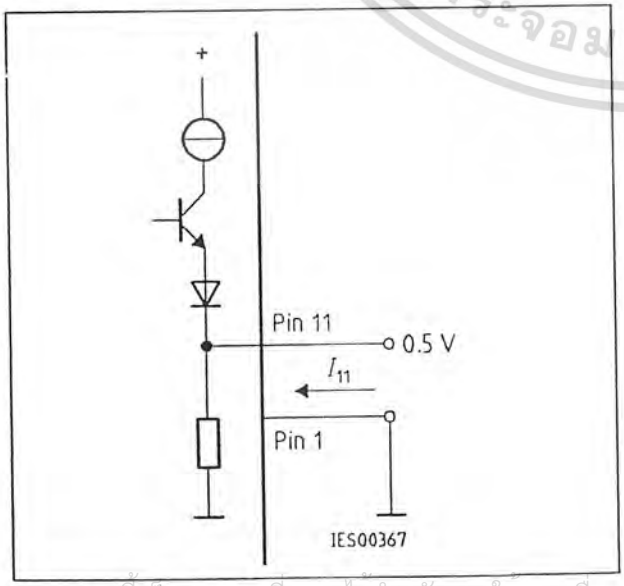
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



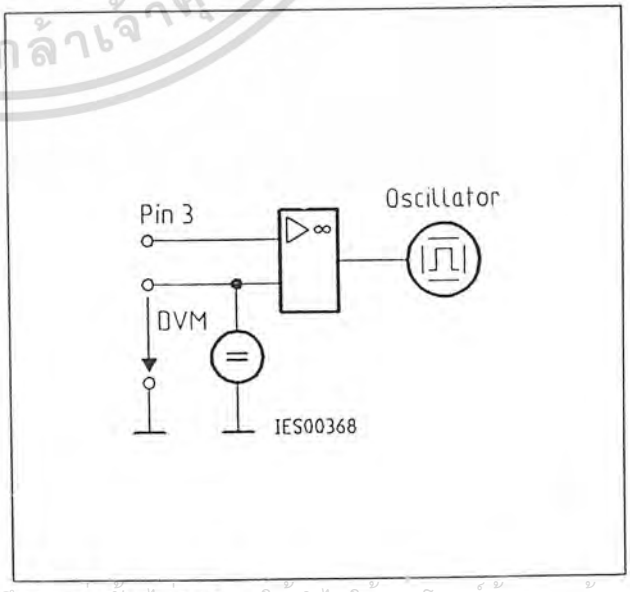


Remaining pins are connected as in test circuit 1
The 10 µF capacitor at pin 5 serves only for test purposes

Test Circuit 4

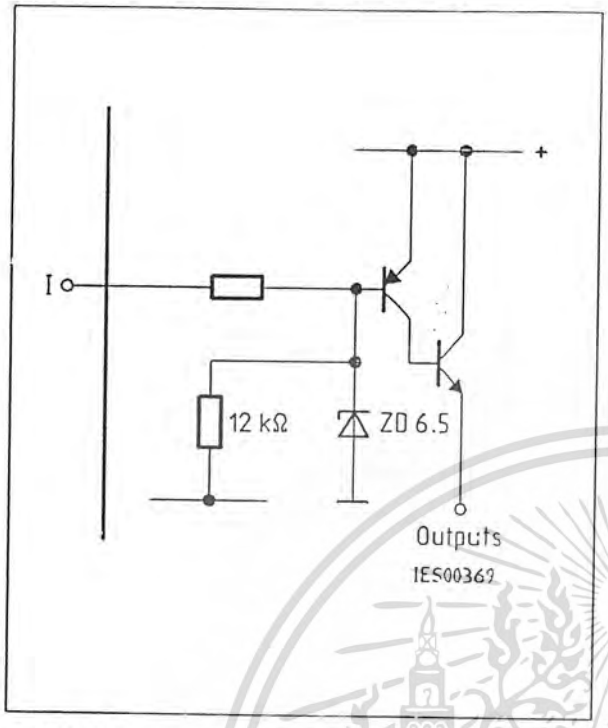


Test Circuit 5

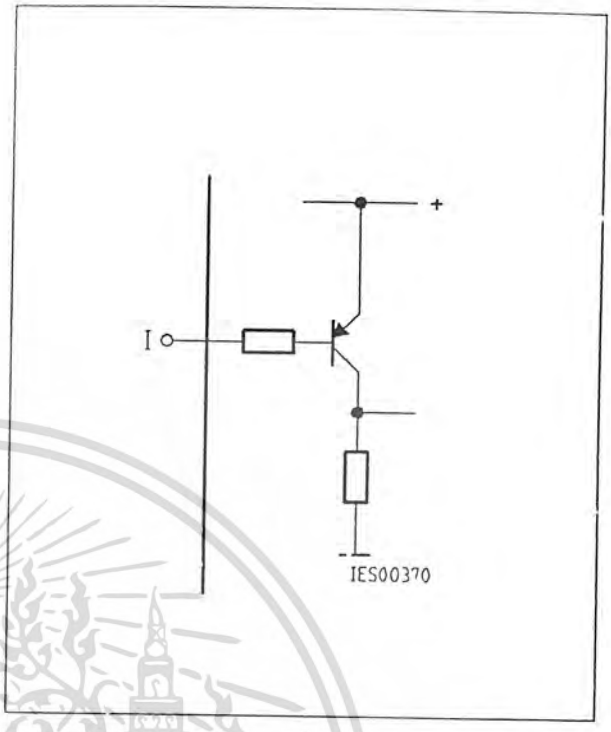


Test Circuit 6

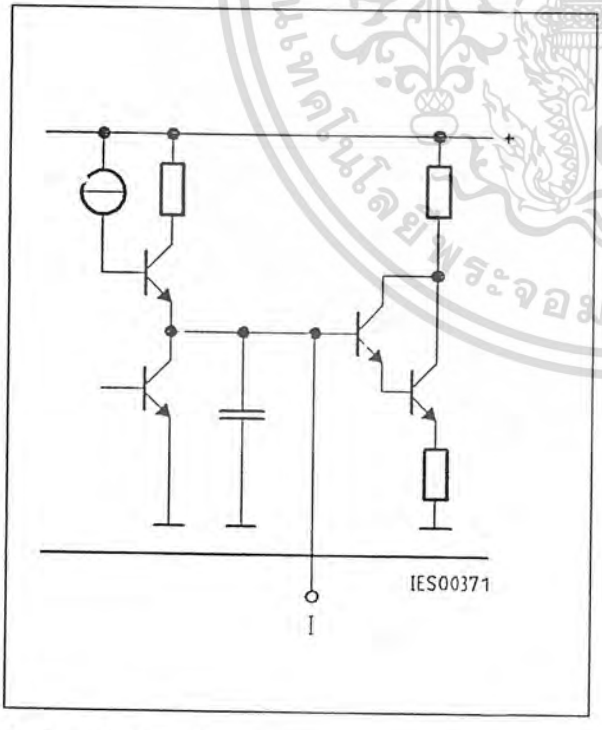
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



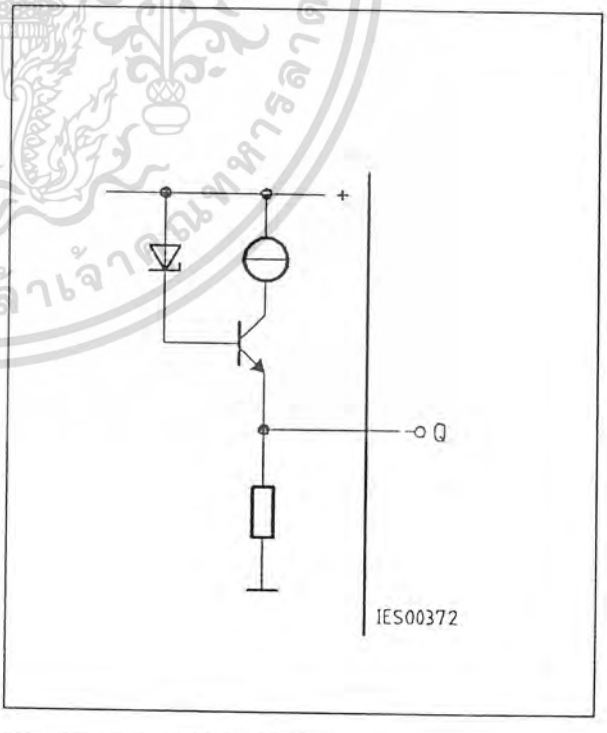
Inhibit 6



Long Pulse 13

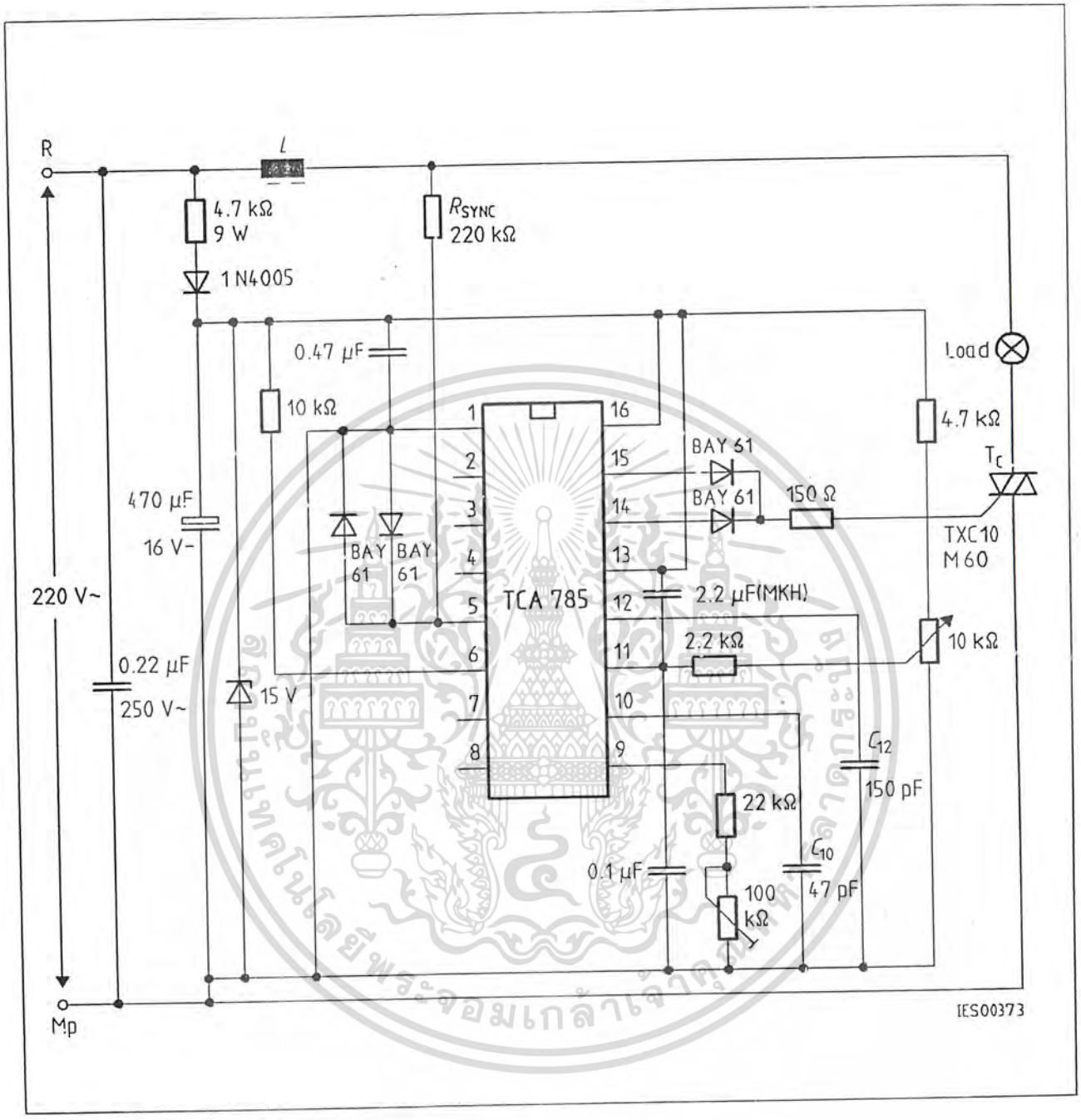


Pulse Extension 12



Reference Voltage 8

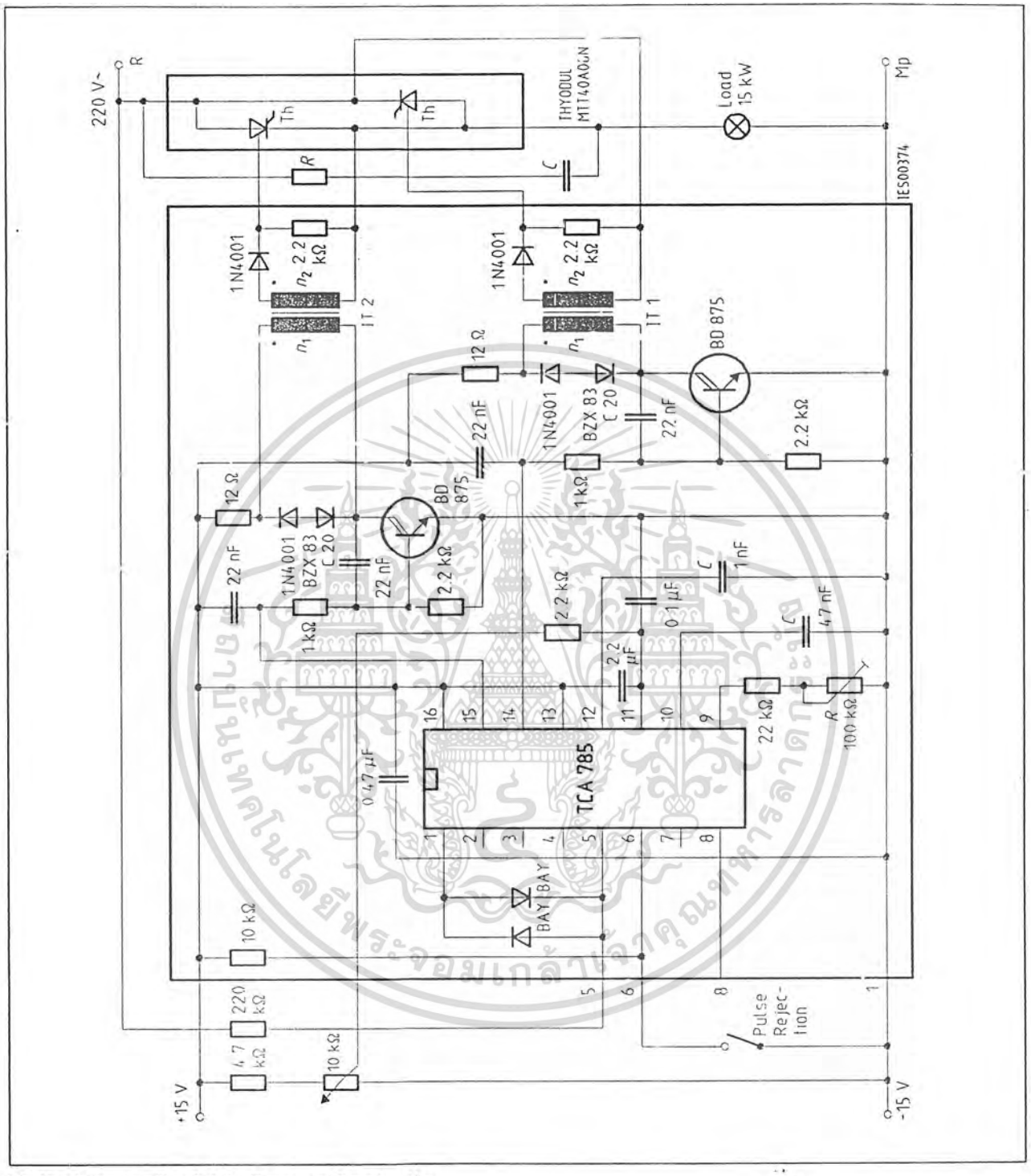
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



Application Examples
Triac Control for up to 50 mA Gate Trigger Current

A phase control with a directly controlled triac is shown in the figure. The triggering angle of the triac can be adjusted continuously between 0° and 180° with the aid of an external potentiometer. During the positive half-wave of the line voltage, the triac receives a positive gate pulse from the IC output pin 15. During the negative half-wave, it also receives a positive trigger pulse from pin 14. The trigger pulse width is approx. 100 µs.

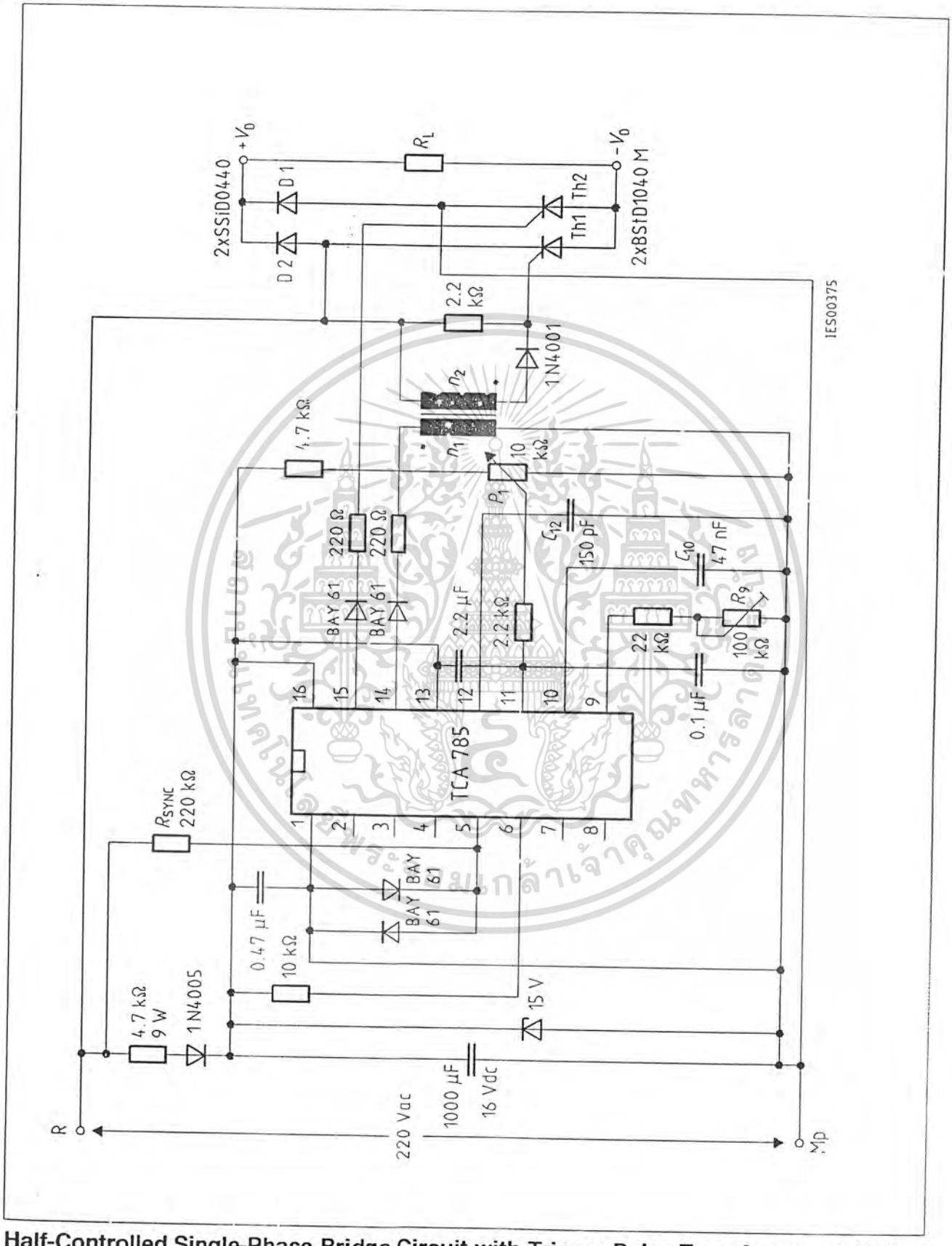
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



Fully Controlled AC Power Controller
Circuit for Two High-Power Thyristors

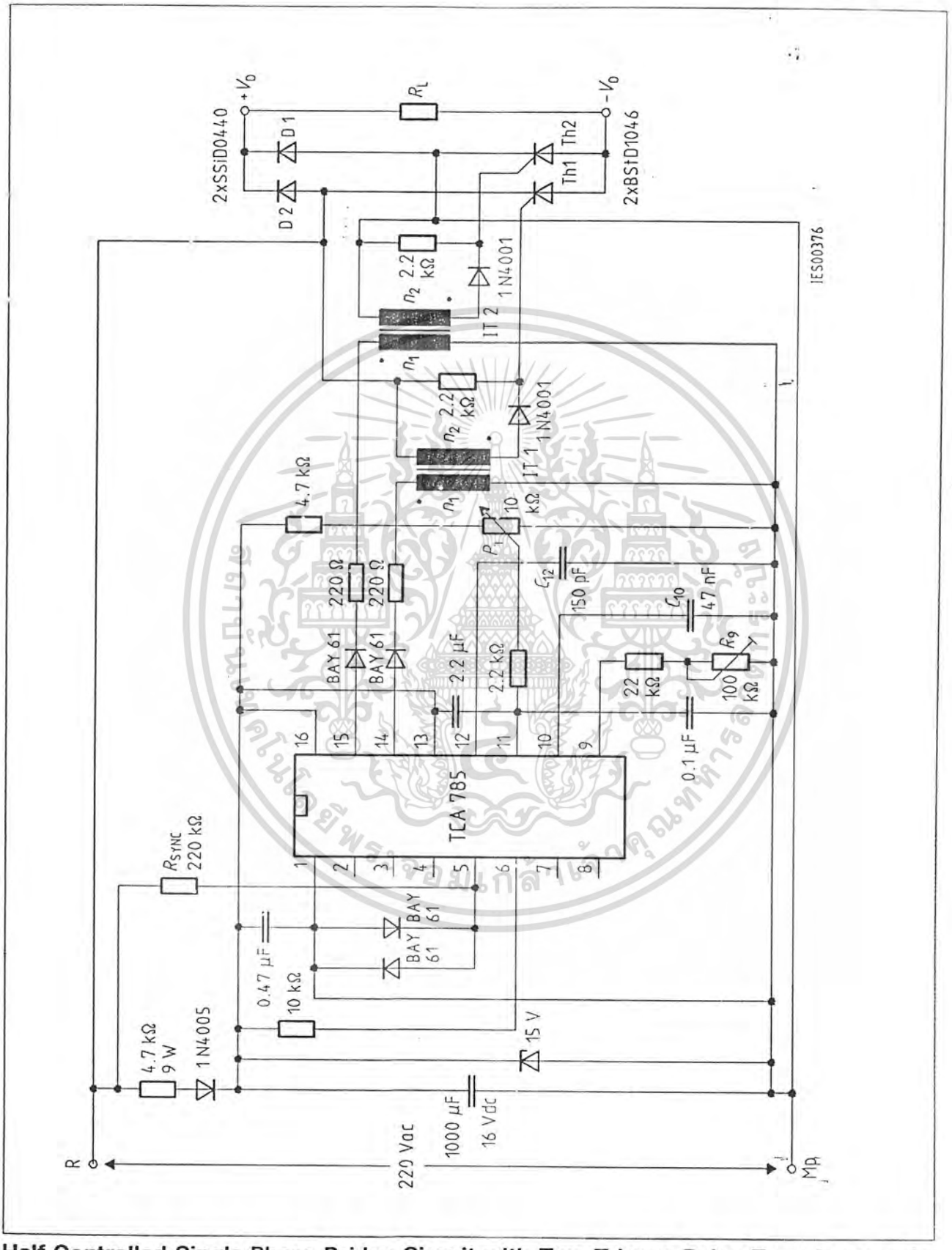
Shown is the possibility to trigger two antiparalleled thyristors with one IC TCA 785. The trigger pulse can be shifted continuously within a phase angle between 0° and 180° by means of a potentiometer. During the negative line half-wave the trigger pulse of pin 14 is fed to the relevant thyristor via a trigger pulse transformer. During the positive line half-wave, the gate of the second thyristor is triggered by a trigger pulse transformer at pin 15.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่อนุญาตให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



Half-Controlled Single-Phase Bridge Circuit with Trigger Pulse Transformer and Direct Control for Low-Power Thyristors

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่อนุญาตให้ดัดแปลง เนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



Half-Controlled Single-Phase Bridge Circuit with Two Trigger Pulse Transformers for Low-Power Thyristors

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้