

การควบคุมอุปกรณ์ไฟฟ้าผ่านสายเพาเวอร์ไลน์
POWER LINE CONTROL



ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาอุตสาหกรรมศาสตรบัณฑิต

ภาควิชาวิศวกรรมสารสนเทศ

คณะวิศวกรรมศาสตร์

รพ.
พ.๕๕๖๖
๕๕๖๖

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2544

เลขหมู่.....
เลขทะเบียน... 46414
วัน, เดือน, ปี... 1 เม.ย. 2546

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

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านกึ่งการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หัวข้อปริญญานิพนธ์

การควบคุมอุปกรณ์ไฟฟ้าผ่านสายพาวเวอร์ไลน์

TITLE

POWER LINE CONTROL

โดย

นางสาวพรหทัย คาชดา รหัส 43015736

นายพัฒนพงษ์ พงษ์เนตร รหัส 43015737

อาจารย์ผู้ควบคุมปริญญานิพนธ์

รศ.ชวลิต เบญจางคประเสริฐ

ภาควิชา

วิศวกรรมสารสนเทศ

ปีการศึกษา

2544

ปริญญานิพนธ์ฉบับนี้ได้รับการอนุมัติให้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตร
อุตสาหกรรมศาสตรบัณฑิต คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหาร
ลาดกระบัง



(ร ศ.ชวลิต เบญจางคประเสริฐ)

อาจารย์ผู้ควบคุมปริญญานิพนธ์

ลิขสิทธิ์ของคณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หัวข้อปริญญานิพนธ์	การควบคุมอุปกรณ์ไฟฟ้าผ่านสายเคเบิลไร้
นักศึกษา	นางสาวพรหทัย ดาษดา รหัส 43015736 นายพัฒน์พงษ์ พงษ์เนตร รหัส 43015737
อาจารย์ผู้ควบคุมปริญญานิพนธ์	รศ.ชวลิต เบญจางคประเสริฐ
ระดับการศึกษา	ปริญญาอุตสาหกรรมศาสตรบัณฑิต
ภาควิชา	วิศวกรรมสารสนเทศ
ปีการศึกษา	2544

บทคัดย่อ

ปัจจุบันอุปกรณ์อำนวยความสะดวกที่ทำงานด้วยไฟฟ้า ได้เข้ามามีบทบาทในชีวิตประจำวันของ ซึ่งอุปกรณ์เหล่านี้ล้วนใช้พลังงาน ไฟฟ้าจากสายเคเบิลไร้ทั้งสิ้นแต่การเปิดปิดอุปกรณ์เหล่านี้ ต้องเปิดปิด ณ จุดที่ติดตั้งอยู่

ด้วยเหตุนี้จึงเกิดแนวคิดที่จะควบคุมอุปกรณ์ไฟฟ้า โดยการส่งสัญญาณควบคุมไปตามสายไฟฟ้ากระแสสลับ 220 โวลต์ เพื่อควบคุมอุปกรณ์ไฟฟ้า ณ จุดต่างๆ ได้โดยสะดวก ในการส่งสัญญาณควบคุมไปบนสายเคเบิลไร้ นั้นจะใช้ ไอซี TDA5051A เป็น ไอซีโมเด็ม และใช้ไมโครคอนโทรลเลอร์เป็นตัวควบคุม โดยสามารถควบคุมอุปกรณ์ผ่านพอร์ตต่างๆ ของไมโครคอนโทรลเลอร์ได้ ในโครงงานนี้แสดงให้เห็นการควบคุมอุปกรณ์ไฟฟ้า ผ่านสายเคเบิลไร้ตามจุดที่นำเสนอ 16 จุดได้อย่างมีประสิทธิภาพ

PROJECT TITLE POWER LINE CONTROL

STUDENT Miss Pornhathai Dassada 43015736

 Mr. Patpong Pongnet 43015737

ADVISOR Assoc.Prof. Chawalit Benjangkprasert

COURSE Bachelor of Industrial Technology in Electronics

DEPARTMENT Information Engineering

YEAR 2001

ABSTRACT

In the present day the electrical devices play an important role in personal daily life. All of them use the power supplies from AC power line. When we want to turn on or turn off them. We must turn on or turn off them at switch.

Therefore, there has been a notion to controls the electrical devices by sending the signal control into the power line. By this method, we can control the electrical devices comfortably. To transmit signal control into the power line, we use IC modem TDA 5051A to modulate and demodulate and we use the microcontroller to control the device. Which it can control any port of microcontroller. This project demonstrates the controllation of the electrical devices efficiently through the power line, for 16 proposed position.

กิตติกรรมประกาศ

ในการสร้างโครงการนี้ ทางผู้จัดทำได้รับความอนุเคราะห์ช่วยเหลือให้คำปรึกษา แนะนำ
ทางในการดำเนินงาน ตลอดจนความช่วยเหลือในเรื่องเครื่องมือที่ใช้ในการดำเนินงานต่าง ๆ จาก
รศ.ชวลิต เบญจางคประเสริฐ จนกระทั่งสำเร็จเป็นโครงการ ทีมงานผู้จัดทำโครงการขอกราบ
ขอบพระคุณท่านอาจารย์ที่ให้ความกรุณามา ณ ที่นี้

ทำยนี้ผู้จัดทำขอกราบขอบพระคุณ บิดา-มารดา ซึ่งให้การสนับสนุนทางการเงินและ
เป็นกำลังใจ แก่ผู้จัดทำเสมอมาจนสำเร็จการศึกษา



นางสาวพรหทัย ดายดา

นายพัฒนพงษ์ พงษ์เนตร

สารบัญ

	หน้า
บทคัดย่อภาษาไทย	I
บทคัดย่อภาษาอังกฤษ	II
กิตติกรรมประกาศ	III
บทที่ 1 บทนำ	1
บทที่ 2 ทฤษฎีและหลักการทํางาน	2
2.1 การสื่อสารแบบอนุกรม	2
2.2 โพรโทคอลการเชื่อมต่อข้อมูล Data Link Protocol	12
2.3 ไมโครคอนโทรลเลอร์ MCS-51	15
2.4 Visual Basic 6	37
บทที่ 3 การออกแบบและการสร้าง	41
3.1 หลักการทํางานของการควบคุมอุปกรณ์ไฟฟ้าผ่านสายเพาเวอร์ไลน์	41
3.2 ภาคมอเตอร์และคีมอเตอร์	43
3.3 ภาควัดค่าแรงดัน	44
3.4 ไมโครคอนโทรลเลอร์	45
3.5 การทํางานของภาคควบคุมอุปกรณ์ (ON-OFF)	46
บทที่ 4 การทดลองและผลการทดลอง	47
4.1 รูปสัญญาณที่ได้รับจากการมอดและคีมอเตอร์สัญญาณ	47
4.2 ทดลองวัดค่าแรงดันที่ผ่านวงจรคัปปลิงทางด้านรับ	50
บทที่ 5 สรุปและวิจารณ์ผลการทดลอง	51
5.1 สรุปผลการทดลอง	51
5.2 ปัญหาและแนวทางแก้ไข	51
5.3 การพัฒนาโครงการ	51
หนังสืออ้างอิง	52
ภาคผนวก ก.	53
ไฟล์เวิร์กทํางานของไมโครคอนโทรลเลอร์	54
โปรแกรมควบคุมการทํางานของไมโครคอนโทรลเลอร์	55
โปรแกรมที่ใช้ในการส่งควบคุมจากคอมพิวเตอร์	60

สารบัญ (ต่อ)

หน้า

ภาคผนวก ข.

69

วงจรรวมและ แผ่นวงจรพิมพ์ (PCB)

ภาพถ่ายโครงการ



สารบัญตาราง

	หน้า
ตารางที่ 2.1 สัญญาณต่าง ๆ ของพอร์ต P3	18
ตารางที่ 2.2 ลำดับความสำคัญของการอินเตอร์รัพต์	26
ตารางที่ 2.3 Vector Address ของการตอบสนองอินเตอร์รัพต์	29
ตารางที่ 2.4 การเลือกโหมดการทำงานของพอร์ตอนุกรม	33
ตารางที่ 4.1 ผลจากการวัดแรงดันที่ความยาวสายต่าง ๆ	50



บทที่ 1

ความมุ่งหมายของปริญญาานิพนธ์

1.1 บทนำ

อุปกรณ์อำนวยความสะดวกที่ทำงานด้วยไฟฟ้า มีมากมายหลายชนิด ซึ่งในปัจจุบันล้วนแต่เป็นสิ่งจำเป็นที่มนุษย์ขาดไม่ได้ อาทิเช่น โทรทัศน์ เครื่องปรับอากาศ พัดลม ตู้เย็น หลอดไฟ ฯลฯ ซึ่งอุปกรณ์เหล่านี้ล้วนใช้ไฟฟ้าเป็นแหล่งพลังงานโดยจ่ายไฟฟ้าผ่านสาย AC ทั้งสิ้น แต่การที่จะควบคุมการเปิดปิดอุปกรณ์ไฟฟ้าเหล่านี้ ล้วนต้องให้ผู้ใช้งานเดินไป ณ จุดที่ตั้งอุปกรณ์ ซึ่งหากจุดนั้นอยู่ไกลก็ย่อมจะไม่สะดวกนัก ดังนั้นหากเราใช้ระบบควบคุมอุปกรณ์ไฟฟ้าผ่านสาย AC โดยการมอดูเลตสัญญาณควบคุมกับสัญญาณ AC 50 Hz แล้วส่งไปตามสาย AC

จากหลักการดังกล่าวข้างต้น ทำให้สามารถควบคุมอุปกรณ์ไฟฟ้าต่าง ๆ ทุกจุดภายในบ้านผ่านสาย AC โดยไม่ต้องเดินสายไฟควบคุมเพิ่ม การควบคุมนั้นจะใช้ไมโครโปรเซสเซอร์เป็นตัวควบคุมแล้วใช้ IC Modem ในการมอดูเลตและมิมอดูเลต ซึ่งในการควบคุมผ่านสาย AC นี้ เรายังสามารถต่อเข้ากับระบบคอมพิวเตอร์ แล้วควบคุมผ่านคอมพิวเตอร์ได้อีกด้วย

1.2 วัตถุประสงค์ของการทำปริญญาานิพนธ์

ปริญญาานิพนธ์ฉบับนี้มีวัตถุประสงค์เพื่อศึกษาการทำงานของระบบควบคุมอุปกรณ์ไฟฟ้าภายในที่พักอาศัยผ่านสาย AC โดยใช้หลักการมอดูเลตสัญญาณควบคุมแล้วส่งสัญญาณไปบนไฟฟ้ากระแสสลับ โดยใช้ IC Modem TDA 5051 ซึ่งเป็นไอซีที่มีประสิทธิภาพสูงและมีคุณลักษณะเด่น ๆ มากมาย จึงทำให้ระบบที่ออกแบบนั้นมีเสถียรภาพและง่ายต่อการควบคุม เนื่องจากสามารถควบคุมโดยผ่านชุดควบคุมชุดเดียว อีกทั้งยังสะดวกในการติดตั้งเพราะไม่ต้องเดินสายควบคุมเพิ่ม และสามารถที่จะพัฒนาให้เข้ากับระบบคอมพิวเตอร์จึงสามารถพัฒนาให้ควบคุมผ่านทางระบบคอมพิวเตอร์ส่วนบุคคล จึงสามารถทำให้เป็นระบบอัตโนมัติ จะเห็นได้ว่าเป็นระบบที่อำนวยความสะดวกให้กับผู้ใช้เป็นอย่างมาก

1.3 ขอบเขตของโครงการ

การควบคุมอุปกรณ์ไฟฟ้าภายในบ้านผ่านสายไฟฟ้ากระแสสลับ 220 โวลต์ สามารถส่งสัญญาณไปควบคุมไมโครคอนโทรลเลอร์ให้มีเอาต์พุตที่พอร์ต 1, 2 ได้ตามต้องการ ซึ่งควบคุมอุปกรณ์ไฟฟ้าได้ทั้งหมด 16 จุด ต่อไมโครคอนโทรลเลอร์หนึ่งตัว

บทที่ 2

ทฤษฎีและหลักการ

2.1 การสื่อสารแบบอนุกรม

เครื่อง PC สามารถที่จะติดต่อสื่อสารกับอุปกรณ์คอมพิวเตอร์อื่น ๆ ที่อยู่ห่างออกไปเป็นพัน ๆ ฟุตได้ โดยติดต่อผ่านทางพอร์ตสื่อสารอนุกรม พอร์ตสื่อสารอนุกรมนั้นเป็นพอร์ตแบบสองทางที่ใช้เชื่อมเครื่อง PC ออกสู่โลกภายนอกโดยอาจจะต่อเข้ากับ โมเด็ม เครื่อง PC เครื่องมินิคอมพิวเตอร์ หรือเครื่องคอมพิวเตอร์เมนเฟรมก็ได้ และนอกจากนั้นพอร์ตอนุกรมนั้นยังเป็นช่องทางที่มักจะถูกใช้งานกับอุปกรณ์คอมพิวเตอร์อื่น ๆ ได้อีก เช่น เมาส์และเครื่องพิมพ์แบบอนุกรม

เครื่อง PC ทั่วไปจะมีพอร์ตขนาบสำหรับใช้งานกับเครื่องพิมพ์ติดมาพร้อมกับเครื่องอยู่แล้ว การต่อแบบขนาบนี้มีข้อดีคือ มีอัตราการส่งข้อมูลสูงกว่าการต่อแบบอนุกรม แต่ก็มีข้อเสียคือ มีข้อจำกัดในการต่อสายออกไปใช้งานไกล ๆ ซึ่งการใช้งานพอร์ตขนาบนั้นมักจะใช้งานอยู่ในระยะไม่เกินสิบฟุตจากเครื่องคอมพิวเตอร์เท่านั้น

เป็นที่น่าเสียดายที่ผู้ออกแบบเครื่อง PC ต้นแบบหรือที่รู้จักกันในชื่อของเครื่อง IBMPC/XT นั้นไม่ได้ให้ความสำคัญของการสื่อสารแบบอนุกรมของ PC จึงทำให้ PC รุ่นแรก ๆ ไม่มีส่วนวงจรที่ใช้เป็นพอร์ตชนิดนี้ติดมากับเครื่องด้วย ดังนั้นถ้าหากผู้ใช้เครื่องรุ่นดังกล่าวต้องการที่จะใช้งานพอร์ตนี้แล้ว ผู้ใช้รายนั้นก็จะต้องซื้อวงจรพอร์ตสื่อสารอนุกรมมาติดตั้งเพิ่มเติมเข้าไปในเครื่องเอง แต่สำหรับเครื่อง PC รุ่นใหม่ ๆ ผู้ผลิตมักจะสร้างส่วนที่เป็นวงจรสื่อสารอนุกรมติดมาพร้อมกับเครื่องด้วย แต่ถึงอย่างไรก็ตามพอร์ตสื่อสารอนุกรมดังกล่าวก็ยังมีประสิทธิภาพในการทำงานที่ไม่ดีนัก ดังนั้นถ้าหากว่าผู้ใช้ต้องการการสื่อสารข้อมูลที่มีประสิทธิภาพมากกว่านี้ ผู้ใช้ก็จะต้องซื้อพอร์ตสื่อสาร อนุกรมรุ่นพิเศษที่ใช้ชิป 16550 มาติดตั้งแทนพอร์ตเดิมที่มีอยู่ในระบบเอง

2.1.1 มาตรฐานการเชื่อมต่อแบบ RS-232-C

การเชื่อมต่ออนุกรมนั้นอาจจะกล่าวได้ว่า เป็นวิธีการเชื่อมต่อสื่อสารทั่วไปของอุปกรณ์ทางอิเล็กทรอนิกส์ต่าง ๆ ไม่ว่าจะมาจากผู้ผลิตรายใดก็ตามแต่เนื่องจากว่าผู้ผลิตสินค้าทางอิเล็กทรอนิกส์ต่างก็ออกแบบพอร์ตสื่อสารอนุกรมของตนขึ้นมา ทำให้มีความแตกต่างกันในรายละเอียดของพอร์ตหรือวิธีการในการรับส่งข้อมูล เพื่อให้แน่ใจว่าพอร์ตอนุกรมหนึ่ง ๆ นั้นสามารถที่จะติดต่อสื่อสารกับพอร์ตอนุกรมอื่น ๆ ได้โดยไม่มีปัญหาทางด้านเทคนิคของกรอุตสาหกรรม

อิเล็กทรอนิกส์ของสหรัฐ (EIA) ก็ได้กำหนดมาตรฐานของการเชื่อมต่อและคุณสมบัติของพอร์ตสื่อสารอนุกรมขึ้นมาในปี ค.ศ. 1969

ตัวองค์กร EIA เองนั้นนับได้ว่าเกี่ยวข้องโดยตรงกับอุตสาหกรรมอิเล็กทรอนิกส์ทั่วโลก ดังนั้นมาตรฐานอันดังกล่าวจึงกลายเป็นมาตรฐานสากลไปโดยปริยาย มาตรฐานอันหนึ่งที่ได้รับการรับรองโดย EIA แล้วก็เป็นมาตรฐานที่ 232 และ C หรือเป็นที่รู้จักกันในชื่อของ RS-232-C (Regulation Standard No. 232 Type C) มาตรฐานการเชื่อมต่อแบบนี้มีความคล้ายคลึงกับมาตรฐาน CCITT V.24 และคุณสมบัติทางไฟฟ้าใกล้เคียงกับมาตรฐาน Mil-Std-188C ของกระทรวงกลาโหมสหรัฐฯ ด้วย ซึ่งรายละเอียดของมาตรฐานต่าง ๆ นั้นจะกล่าวถึงอีกครั้งหนึ่ง

หลังจากนั้นในปี ค.ศ.1971 EIA ได้กำหนดมาตรฐานใหม่ที่มีประสิทธิภาพสูงกว่ามาตรฐาน RS-232-C ออกมาใหม่ ซึ่งก็ได้แก่มาตรฐาน RS-449, RS-423 โดยที่มาตรฐานใหม่เหล่านี้จะมีคุณสมบัติทางไฟฟ้าและคุณสมบัติทางกายภาพคล้าย ๆ กัน ดังนั้นจึงถูกเรียกรวม ๆ ไปว่าเป็นกลุ่มมาตรฐาน RS-449 แต่อย่างไรก็ตามทางบริษัท IBM และบริษัทผู้ผลิตเครื่องคอมพิวเตอร์ตระกูล PC อื่น ๆ ก็ไม่ได้ให้ความสนใจมาตรฐานใหม่นี้เท่าไรนักถึงแม้ว่ามันจะสามารถส่งข้อมูลได้เร็วและได้ระยะทางไกลกว่ามาตรฐาน RS-232-C ก็ตาม มีเพียงบริษัท Apple Computer เท่านั้นที่ได้นำเอามาตรฐานชุด RS-449 นี้ไปใช้กับผลิตภัณฑ์ Machintosh ของตน

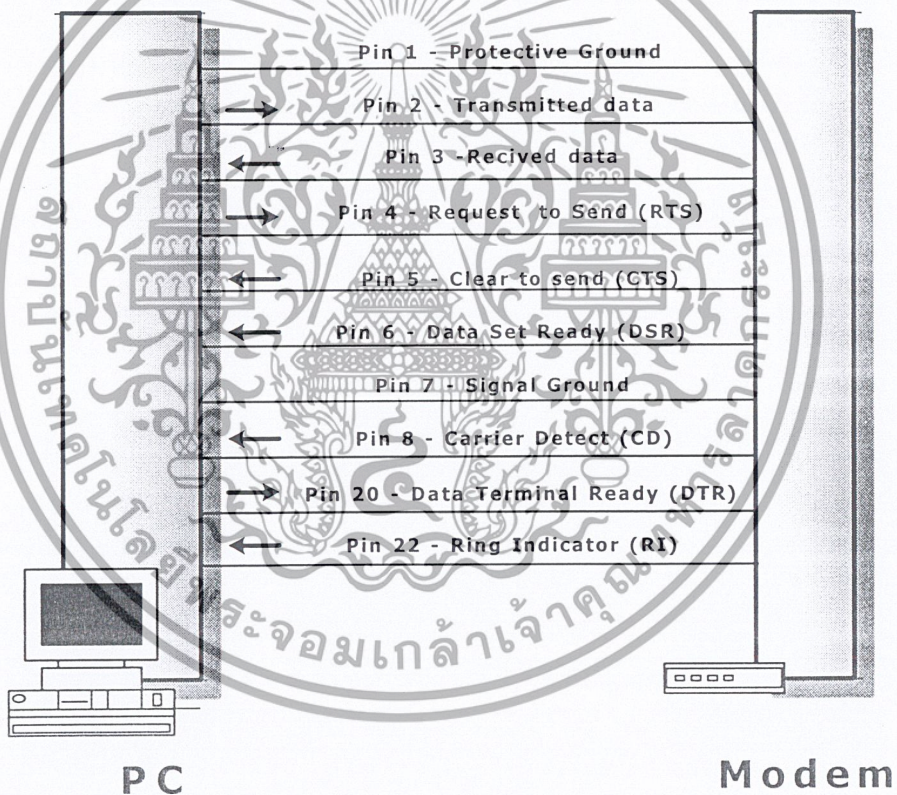
มาตรฐาน RS-232-C นั้นแบ่งการเชื่อมต่อออกเป็นสองลักษณะคือ การต่อกับอุปกรณ์เทอร์มินัล (DTE : Data Terminal Equipment) และการต่อกับอุปกรณ์สื่อสารข้อมูล (DCE : Data Communication Equipment) ซึ่งในกรณีทั่ว ๆ ไปนั้น DCE จะต้องต่อเข้ากับ DTC เสมอ ยกตัวอย่างเช่น การต่อกับโมเด็มเข้ากับเครื่อง PC โดยเครื่อง PC จะเป็นอุปกรณ์ DTE และโมเด็มก็จะเป็นอุปกรณ์ DCE



รูปที่ 2.1 รูปแสดงการต่ออุปกรณ์สื่อสารผ่านทางพอร์ตอนุกรม

RS-232-C เครื่อง PC จะส่งและรับข้อมูลจากโมเด็ม

การเชื่อมต่อตามมาตรฐาน RS-232-C นั้น โดยปกติจะใช้คอนเน็กเตอร์รูปตัว D ชนิด 25 ขา กำหนดให้ปลายสายสัญญาณด้านหนึ่งที่เป็นคอนเน็กเตอร์ตัวผู้ใช้ต่อกับอุปกรณ์ DCE และปลายของสัญญาณอีกด้านหนึ่งจะต้องเป็นคอนเน็กเตอร์ตัวเมียใช้ต่อกับอุปกรณ์ DTE แต่ว่ากฎและมาตรฐานนั้นมักจะถูกเปลี่ยนแปลงอยู่เสมอ ดังนั้นผู้ผลิตอุปกรณ์คอมพิวเตอร์อื่น ๆ ก็ได้ ออกแบบมาตรฐานคอนเน็กเตอร์แบบใหม่ออกมาเพื่อใช้แทนมาตรฐาน RS-232-C เดิม อย่างเช่น บริษัท IBM นั้นก็ได้ออกแบบคอนเน็กเตอร์แบบ 9 ขาขึ้นมาใช้แทนคอนเน็กเตอร์แบบ 25 ขา และได้กำหนดให้เป็นมาตรฐานบนเครื่อง IBM AT ซึ่ง IBM ในช่วงนั้นยังเป็นผู้นำทางด้าน การพัฒนาเครื่องคอมพิวเตอร์ส่วนบุคคลอยู่ ดังนั้นผู้ผลิตรายอื่น ๆ ก็ได้นำมาตรฐานคอนเน็กเตอร์แบบ ใหม่ของ IBM ไปใช้เป็นมาตรฐานในผลิตภัณฑ์ของตนด้วย



รูปที่ 2.2 สายนำสัญญาณที่ใช้ในมาตรฐาน RS-232-C

สายนำสัญญาณที่ใช้ในมาตรฐาน RS-232-C นั้นจะมีทั้งหมด 9 เส้นประกอบด้วยวงจรข้อมูล สองวงจร คือขานำสัญญาณหมายเลข 2 และ 3 ซึ่งจะใช้ในการส่งผ่านข้อมูลระหว่างเครื่อง PC และโมเด็มขานำสัญญาณหมายเลข 1 และ 7 เป็นกราวด์ส่วนวงจรชุด ที่สองจะประกอบไปด้วย

ขานำสัญญาณหมายเลข 4 และ 5 เป็นวงจรควบคุมสัญญาณที่เรียกว่า RTS และ CTS ซึ่งจะใช้ในการควบคุมการไหลของข้อมูลระหว่าง PC และโมเด็ม (Hardware Flow Control) ขานำสัญญาณหมายเลข 8 เป็นสัญญาณ Carrier Detect (CD) ใช้บอก PC ว่าใช้ขณะนั้นโมเด็มได้รับสัญญาณพาหะอยู่หรือไม่ขานำสัญญาณหมายเลข 20 (DTR : Data Terminal Ready) ใช้บอกโมเด็มว่าในขณะที่เทอร์มินัล (หรือ PC) พร้อมทั้งจะติดต่อกับโมเด็มหรือไม่ ส่วนขานำสัญญาณหมายเลข 22 จะใช้เป็นสัญญาณแสดงว่าในขณะที่โมเด็มได้รับสัญญาณกระดิ่งหรือไม่ ซึ่งมักจะไม่ได้ถูกใช้งานมากนัก

ในการสื่อสารแบบอนุกรมนี้ การที่จะทำให้อุปกรณ์อิเล็กทรอนิกส์สื่อสารกันได้ อย่างน้อยก็จำเป็นต้องใช้สายไฟฟ้าเชื่อมต่อระหว่างอุปกรณ์สองตัวไม่ต่ำกว่าสองเส้นแต่จะเห็นมาตรฐาน RS-232-C นั้นกำหนดให้คอนเน็คเตอร์มีขานำสัญญาณถึง 25 ขา ซึ่งอาจดูเหมือนว่าจะดูฟุ่มเฟือยเกินความจำเป็นจนเกินไป เหตุที่ต้องใช้คอนเน็คเตอร์แบบ 25 ขานี้ก็เพราะว่าอุปกรณ์สื่อสารแบบอนุกรมตามมาตรฐานนั้น จะต้องมีการไฟฟ้าที่เกี่ยวข้องกับการรับส่งข้อมูลอยู่สามวงจรด้วยกันก็คือ วงจรรับข้อมูลและวงจรควบคุม รูปที่ 3.2 จะแสดงผังการเชื่อมต่อแบบอนุกรมตามมาตรฐาน RS-232-C แบบคร่าว ๆ โดยจะแสดงเฉพาะการเชื่อมต่อสัญญาณที่สำคัญในการใช้งานเพียง 10 เส้น เท่านั้น จากรูปจะเห็นได้ว่า RS-232-C ดังที่จะกล่าวถึงรายละเอียดแยกเป็นประเภทดังต่อไปนี้

2.1.2 สัญญาณนำข้อมูล

ส่วนที่นับได้ว่าเป็นส่วนที่สำคัญที่สุดของมาตรฐาน RS-232-C ก็คือวงจรสำหรับส่งผ่านข้อมูลซึ่งมีอยู่สองวงจรถือวงจรที่ทำหน้าที่ส่งข้อมูลจาก DTE ไป DCE และอีกวงจรหนึ่งก็จะทำหน้าที่ตรงกันข้ามกับวงจรแรกก็คือวงจรที่ทำหน้าที่ส่งข้อมูลจาก DCE เพื่อส่งไปให้ DTE

จากรูปที่ 3.2 จะเห็นว่าเครื่อง PC นั้นจะส่งข้อมูลไปให้โมเด็มทางขานำสัญญาณหมายเลข 2 และรับข้อมูลจากโมเด็มทางขานำสัญญาณหมายเลข 3 โดยใช้ขานำสัญญาณหมายเลข 7 เป็นกราวด์ร่วมของทั้งสองวงจร

สัญญาณทางไฟฟ้าที่ขานำสัญญาณ หมายเลข 2, 3 และ 7

มาตรฐาน RS-232-C นั้นจะให้ความสำคัญกับวิธีการส่งผ่านข้อมูลบนสายนำสัญญาณทางไฟฟ้าที่ใช้แทนข้อมูลนั้นก็เป็นรูปแบบสัญญาณง่าย ๆ คือสัญญาณสี่เหลี่ยมที่ถูกสร้างจากไฟฟ้า กระแสตรงระดับแรงดันประมาณ 2.5 ถึง 3 โวลต์ แทนข้อมูล "0" และ -2.5 ถึง -3 โวลต์ จะแทนข้อมูล "1" สำหรับช่วงระดับแรงดันประมาณ -3 ถึง 3 โวลต์นั้นจะเป็นช่วงระดับแรงดันที่ใช้ในการแบ่งแยกกระดิ่งสถานะของสัญญาณระหว่างสถานะ "0" และสถานะ "1"

คุณสมบัติของสัญญาณแบบสี่เหลี่ยมนี้เมื่อถูกส่งออกไปในสายไฟฟ้าแบบธรรมดา มักจะเกิดความผิดเพี้ยนได้ง่าย เนื่องจากข้อกำหนดของสายนำสัญญาณหรืออาจจะมีสาเหตุมาจากสัญญาณรบกวนจากภายนอกอื่น ๆ ทำให้การรับข้อมูลมีระยะใช้งานอยู่ในช่วงที่จำกัด โดยเฉพาะการรับส่งข้อมูลที่มีความเร็วสูงนั้นคลื่นสี่เหลี่ยมจะเรียงชิดติดกัน จึงยังทำให้มีโอกาสเกิดความผิดพลาดของการรับข้อมูลได้ง่าย ดังนั้นมาตรฐาน RS-232-C จึงได้กำหนดความยาวของสายนำสัญญาณสำหรับการใช้งานที่ความเร็วต่างไว้อย่างเช่น ถ้าหากจะใช้การเชื่อมต่อตามมาตรฐาน RS-232-C ที่ความเร็วการส่งสูงข้อมูล 9600 BPS ก็จะมีระยะใช้งานได้ที่ระยะทางไม่น้อยกว่าห้าสิบฟุตเป็นต้น แต่อย่างไรก็ตามคุณภาพของสัญญาณก็จะขึ้นกับคุณภาพของสายนำสัญญาณด้วยเช่นกัน

ขานำสัญญาณหมายเลข 4 To Send และ 5 Request Clear To Send

วงจร RTS และ CTS นั้นเป็นวงจรที่ใช้สำหรับการควบคุมการส่งผ่านข้อมูลระหว่างเครื่อง PC และโมเด็ม โดยโมเด็มจะส่งสัญญาณ CTS สถานะ "ON" ให้แก่ PC เมื่อโมเด็มนั้นไม่พร้อมที่จะรับข้อมูลจากเครื่อง PC ได้ ในทำนองเดียวกันเครื่อง PC ก็จะทำให้สัญญาณ RTS สถานะ "ON" เมื่อ PC พร้อมที่จะรับข้อมูลและส่งสัญญาณ RTS สถานะ "OFF" เมื่อ PC ไม่พร้อมที่จะรับข้อมูลจากโมเด็ม วงจรควบคุมในส่วนนี้จะทำงานคล้ายกับสัญญาณไฟจราจรควบคุมการส่งผ่านข้อมูลระหว่างเครื่อง PC และโมเด็มนั่นเอง ซึ่งสัญญาณ RTS และ CTS นี้จะมีประโยชน์มากในการสื่อสารข้อมูลด้วยความเร็วสูง โดยสามารถป้องกันการเดิน และการสูญหายของข้อมูลได้เป็นอย่างดี

ขานำสัญญาณหมายเลข 20 Data Terminal Ready และขานำสัญญาณหมายเลข 6 Data Set Ready

สัญญาณ DTR จะใช้เป็นที่บอกโมเด็มทราบว่าเครื่อง PC นั้นกำลังอยู่ในสถานะที่พร้อมที่จะติดต่อสื่อสารกับ โมเด็มหรือโมและในกรณีเดียวกันสัญญาณ DSR นั้นก็จะใช้เป็นที่บอกให้เครื่อง PC ทราบว่าโมเด็มก็พร้อมที่จะติดต่อกับ PC หรือโม โดยที่สัญญาณ DSR นั้นจะอยู่ในสถานะ "ON" ก็ต่อเมื่อโมเด็มได้รับสัญญาณ DTR แล้ว

ขานำสัญญาณหมายเลข 8 Carrier Detect และ ขานำสัญญาณหมายเลข 22

Ring Indicator

สัญญาณ CD นี้จะใช้เป็นที่บอก PC ให้ทราบว่าโมเด็มนั้นกำลังเชื่อมต่อกับโมเด็มเครื่องอื่น ๆ และกำลังได้รับสัญญาณพาหะจากโมเด็มปลายทาง ส่วนสัญญาณ RI นั้นก็จะเป็นการบอกเครื่อง PC ให้ทราบว่าสัญญาณกระดิ่งโทรศัพท์โดยอัตโนมัติ (Auto-answer) อยู่ภายในตัว

เองแล้ว จึงไม่จำเป็นที่จะต้องใช้งานสัญญาณ RI นี้แต่ในบางกรณี โปรแกรมบางประเภทก็อาจจะใช้สัญญาณ RI นี้เป็นตัวกำหนดให้เริ่มทำการนับโปรแกรมอื่น ๆ ได้

จากที่ได้กล่าวมาข้างต้นแล้วว่าคอนเน็คเตอร์แบบ RS-232-C นั้นจะมีอยู่สองแบบก็คือแบบ 25 ขา และแบบ 9 ขา แต่อย่างไรก็ตามการอินเตอร์เฟส RS-232-C เข้ากับเครื่อง PC นั้นไม่ได้มีเฉพาะข้อกำหนดทางด้านกายภาพของคอนเน็คเตอร์เท่านั้น ด้วยเหตุนี้เองการสื่อสารผ่านพอร์ตสื่อสารอนุกรมของ PC นั้นจำเป็นที่จะต้องชิปที่ใช้สำหรับควบคุมสื่อสารข้อมูลโดยเฉพาะ ซึ่งชิปนี้อาจจะอยู่บนแผ่นวงจรหลัก หรือบนแผ่นวงจรควบคุมอุปกรณ์เอนกประสงค์ (Multi I/O Card) ก็ได้ โดยที่ชิปดังกล่าวที่ใช้กันทั่วไป ก็คือชิปควบคุมการสื่อสารเบอร์ 8250 ของบริษัท Western Digital และชิปที่ใหม่กว่าก็คือชิปเบอร์ 16550A หรือ อาจจะเป็นชิปอื่น ๆ ที่มีคุณสมบัติเท่าเทียมกัน ตัวชิปประเภทนี้จะทำหน้าที่อินเตอร์เฟสข้อมูลแบบอนุกรมเข้ากับระบบบัสข้อมูลของ PC ตามแผนภาพที่แสดงไว้ในรูปที่ 2.3



รูปที่ 2.3 แผนภาพการเชื่อมต่อของชิปควบคุมการสื่อสารเบอร์ 8250 บนเครื่อง PC

ซึ่งชิปนี้จะต่ออยู่กับ I/O ของ CPU และต่อกับขาสัญญาณ IRQ ของ CPU ด้วยโดยการต่อแบบนี้จะทำให้ CPU สามารถติดต่อกับวงจรอินเตอร์เฟสนี้ได้โดยตรง เนื่องจากว่าชิปนี้จะเชื่อมต่ออยู่กับบัสข้อมูลของ PC และวงจรสื่อสารแบบอนุกรมในแต่ละพอร์ตก็จะถูกกำหนดให้มีแอดเดรส I/O ประจำพอร์ตดังนั้นจึงทำให้ CPU สามารถอ่านเขียนข้อมูลที่พอร์ตได้โดยตรงเมื่อ CPU ต้องการ ในอดีตวิศวกรรมของบริษัท IBM ได้กำหนดให้เครื่อง IBM PC มีพอร์ตสื่อสาร

แบบอนุกรมได้เพียงสองพอร์ต และใช้งาน IRQ ได้เพียงสองหมายเลขเท่านั้น ดังนั้นเครื่อง IBM PC และเครื่องที่เข้ากันได้กับ IBM PC อื่น ๆ ในยุคนั้นจึงไม่สามารถใช้งานพอร์ตอนุกรมได้มากกว่าสองพอร์ต แต่ในปัจจุบันก็ได้มีการแก้ไขโดยการกำหนดให้เครื่องรุ่นใหม่อย่างเช่นเครื่อง IBM PC/AT หรือเครื่อง PS/2 สามารถมีจำนวนพอร์ตสื่อสารแบบอนุกรม (COM Port) ได้มากขึ้น และบางพอร์ตก็สามารถใช้ IRQ ให้ใช้พอร์ตละ 1 หมายเลข ซึ่งมาตรฐานของเครื่อง PC ทุกรุ่นจะกำหนดให้พอร์ตที่เรียกว่า COM1 สัญญาณขั้วจิ้งหว่านั้น CON1 จะใช้ IRQ4 และ COM2 จะใช้ IRQ3 และสำหรับเครื่อง PS/2 นั้นการกำหนดแอดเดรส และ PC ซึ่งรายละเอียดเกี่ยวกับการติดตั้งและกำหนดพอร์ตสื่อสารแบบอนุกรม

2.1.3 อัตราข้อมูลดิจิทัล (Digital Data Rates)

ข่าวสารดิจิทัลในรูปแบบอนุกรมจะเคลื่อนที่ในอัตราข้อมูลที่ต่างกัน การส่งข่าวสารจากที่หนึ่งไปอีกที่หนึ่งต้องใช้เวลาในการส่งแต่ละบิตออกไป การส่งแบบอนุกรมจะลดจำนวนสายส่งให้เหลือเพียงคู่เดียว คือ สายที่ส่งไป (Electrical Active) และสายส่งกลับ (Return Line) อัตราที่ข่าวสารถูกส่ง หรือรับเรียกว่า "อัตราบิต" (Bit Rate) มีหน่วยเป็น บิตต่อวินาที (Bit Per Second : dps) รหัสอักขระแต่ละระบบกำหนดจำนวนไบนารีบิตไว้โดยเฉพาะเรียกว่า "คำ" (Word) ซึ่งได้แก่ ชุดของกระแสไฟฟ้าสูงและต่ำเพื่อแทนอักขระต่าง ๆ กันในรหัส เช่น รหัสบอโดที่ใช้ 5 บิตแทนแต่ละอักขระจึงมีค่าของข้อมูลเป็น 5 บิต ในรหัสแอสกีคำจึงมี 7 บิต และ EBCDIC มี 8 บิต

อัตราบิตพื้นฐาน (Basic Data Rate) ซึ่งมีหน่วยเป็นจำนวนบิตต่อวินาที (bps) ได้แก่ อัตราที่ไบนารีบิตหรือวิธีต่าง ๆ ที่ใช้เร่งอัตราการส่งทั้งหมด การจัดกลุ่มของบิตข้อมูลไบนารีดังกล่าวคือ "สัญลักษณ์" ซึ่งได้แก่ การวัดกระแสไฟฟ้าที่เกิดขึ้นหนึ่งหน่วย อาจวัดเป็นโวลต์เตจจำนวนรอบ ของไซน์เวฟ ๗ ความถี่ที่ถูกกำหนดไว้แล้วคลื่นที่ถูกส่งไปในระยะหนึ่งจะถูกเปรียบเทียบกับคลื่น กระแส หรือความต้านทานที่ถูกกำหนดไว้ดังกล่าว สิ่งที่สำคัญในด้านการสื่อสารก็คือ สัญลักษณ์สามารถนำไปนาบิตมาจัดกลุ่มทำให้เกิดกระบวนการต่าง ๆ ได้เช่น ในรหัสแอสกีแทนที่จะใช้รหัส 7 บิต อาจสามารถใช้ระดับโวลต์เตจที่แตกต่างแทนแต่ละอักขระเช่น ค่า 1.2113 V คืออักขระ C และ 1.2114 V คืออักขระ D เนื่องจากแต่ละรหัสไบนารี 7 บิตแทนแต่ละอักขระ ดังนั้นจึงอาจกล่าวได้ว่ารหัสแอสกีกับระดับโวลต์เตจมีความสัมพันธ์ร่วมกันนั่นคือ 100 0011 (43H) สำหรับ C จะมีค่าเท่ากับ 1.2113 V และ 100 0100 (44H) สำหรับ D เท่ากับ 1.2114 V ความหมายก็คือ แต่ละ 7 บิตสามารถทำให้เกิดแต่ละอักขระ (ซึ่งในกรณีนี้คือ ระดับโวลต์เตจ) ได้ อักขระหนึ่งสามารถส่งออกไปได้จริงซึ่งทำให้สามารถส่งได้เร็วกว่าการส่งทีละบิตออกไป

ระบบในการเปลี่ยน 7 บิตของรหัสแอสกีให้เป็นแต่ละอักขระคือ จำนวนอัตราบิตหารด้วยจำนวนบิตต่ออักขระเช่น ข้อมูลที่ส่งออกไปในอัตรา 4900 dps นั้นในการส่งข้อความจริงส่งเพียง 700 อักขระต่อวินาที (เท่ากับ $4900/7$) ทั้งนี้เป็นเพราะอัตราของระบบการส่งถูกจำกัดโดย "ความกว้างแถบ" (Band Width) ของระบบ ตัวอย่างเช่น ความกว้างแถบของระบบโทรศัพท์โดยทั่วไปประมาณ 3 และอยู่ระหว่าง 300 ถึง 3 Hz นั้นอัตราบิตข้อมูล (Data Bit Rate) จึงถูกจำกัดโดย 3000Hz บนสุด (Upper End) ของความกว้างแถบลดลงเมื่อรหัส 7 บิตถูกเปลี่ยนเป็นอักขระ ซึ่งสามารถส่งออกไปได้ถึงอัตราที่ 3000Hz หรืออัตรา 7 บิตต่ออักขระ จะทำให้สามารถส่งข้อมูลที่ 3000 อักขระต่อวินาที \times 7 บิตต่ออักขระ หรือ 21000 dps

คำที่ใช้กันทั่วไปสำหรับค่าอักขระต่อวินาทีของระบบคือ BAUD ซึ่งความหมายกว้าง ๆ ของ ก็คืออัตราของการถ่ายทอดข่าวสาร (Rate of Information Transfer) ที่ใช้คำจำกัดความนี้ เพราะว่าอาจมีสิ่งต่าง ๆ เกิดขึ้นกับข้อมูลเดิม (Original Data) ก่อนที่มันจะอยู่ในรูปที่สามารถส่งออกไปได้จริง บางคนเข้าใจว่า "อัตราบิต" (Bit Rate) มีความหมายเดียวกัน แม้ว่าการสื่อสารและความเร็วต่ำ (Low Speed) ความแตกต่างระหว่าง 2 คำนี้อาจมีน้อย เช่น โมเด็ม 300 หรือ 1200 baud ของคอมพิวเตอร์ส่วนบุคคลอาจเรียกว่า โมเด็ม 300 หรือ 1200 ได้เพราะที่อัตราดังกล่าวหนึ่งสัญลักษณ์ที่ใช้สำหรับแต่ละบิตข้อมูลมีผลทำให้ค่าอัตราบิตและ baud เท่ากัน แต่ที่จริงแล้วทั้ง 2 คำมีความหมายต่างกัน

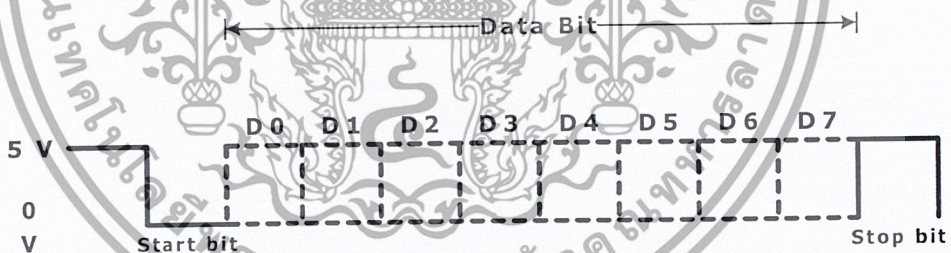
ข้อมูลแบบซิงโครนัสและอะซิงโครนัส (Synchronous and Asynchronous Data)

ข้อมูลแบบอนุกรมจะถูกส่งโดยรูปแบบใดรูปแบบหนึ่งคือแบบซิงโครนัสหรืออะซิงโครนัส ข้อมูลซิงโครนัสต้องการสัญญาณนาฬิกาที่สัมพันธ์กันระหว่างเครื่องส่งและเครื่องรับซึ่งเรียกว่า สัญญาณนาฬิกาข้อมูล (Data Clock) เพื่อให้การแปลข้อมูลที่ส่งและรหัสสอดคล้องกัน เครื่องรับจะจับสัญญาณนาฬิกาข้อมูลซึ่งอยู่ในกระแสข้อมูลแบบอนุกรมนี้ได้โดยวงจรพิเศษที่เรียกว่า "วงจรจับสัญญาณนาฬิกา" (Clock Recovery Circuits) เมื่อเครื่องรับจับสัญญาณนาฬิกาได้จึงจะทำให้ความสอดคล้องกันของบิตและอักขระเกิดขึ้น การสอดคล้องกันของบิต (Bit Synchronization) ได้แก่การเกี่ยวข้องกับทำให้เกิดอักขระ (Character Word) เริ่มต้นและสิ้นสุดเพื่อให้สามารถถอดรหัสและให้ความหมายของอักขระเหล่านั้นได้ การทำให้สอดคล้องกันทั้งหมดนี้ขึ้นอยู่กับสัญญาณนาฬิกาที่จับได้จากกระแสข้อมูลข่าวสาร

2.1.4 การสื่อสารกระแสข้อมูลแบบอนุกรม

แบ่งออกเป็น 2 แบบ คือ

1. การสื่อสารข้อมูลแบบอะซิงโครนัส (Asynchronous Transmission หรือ Start/Stop Transmission) เป็นการดำเนินการโดยอาศัยบิตปิดหัวท้ายหรือ "บิตเฟรม" (Framing Bit) พิเศษ เพื่อทำให้เกิดการเริ่มต้นและการสิ้นสุด (หยุด) ของคำอักขระข้อมูล สัญญาณนาฬิกาจะไม่ถูกจับกระแสข้อมูลถึงแม้ว่าสัญญาณนาฬิกาภายในของเครื่องส่งและเครื่องรับจะต้องมีความถี่เดียวกันจึงจะทำให้สามารถรับข้อมูลได้อย่างถูกต้อง เครื่องรับจะสนองตอบต่อกระแสข้อมูลก็ต่อเมื่อมันได้พบตัว "บิตเริ่มต้น" (Start Bit) อักขระข้อมูลจะถูกถอดรหัสและให้ความหมายหลังจากที่ได้รับ "บิตหยุด" (Stop Bit) และมีการยืนยันอีกครั้งหนึ่งเท่านั้นข้อมูลอะซิงโครนัสสามารถถูกจับได้ และทำให้สอดคล้องกับบิต และสอดคล้องกับอักขระได้ง่ายกว่า แต่ประสิทธิภาพการส่งข้อมูลจะลดลง เนื่องจากต้องมีการส่งบิตเฟรมซึ่งไม่ใช่เป็นข้อมูลข้อความเพิ่มขึ้นอีก แม้ว่าสัญญาณนาฬิกาจะไม่ถูกจับจากรหัสข้อมูล แต่สัญญาณนาฬิกาภายในที่เครื่องส่งและเครื่องรับต่างมีความถี่ใกล้เคียงกันมากที่สุด จึงจะทำให้ข้อมูลที่ผ่านการสุ่มตัวอย่างหรือที่เรียกว่าแซมเปิล (Sample) แล้วอยู่ในอัตราเดียวกันกับที่ถูกส่งออกมา อัตราการสุ่มตัวอย่าง (Sampling Rate) นี้จะทำให้บิตสอดคล้องกันในขณะที่การตรวจจับบิต (Detection) บิตเริ่มต้นและบิตหยุดจะทำให้อักขระสอดคล้องกัน



รูปที่ 2.4 แบบของการสื่อสารแบบอะซิงโครนัส

ในขณะที่ไม่มีสัญญาณในการส่งข้อมูล สัญญาณหรือจะมีแรงดันหรือเป็นสถานะการส่งเป็นแบบว่าง (Idle line) เพื่อต้องการให้ทราบว่ายังมีการติดต่อกันอยู่ทั้ง 2 ฝ่าย แต่เมื่อเริ่มต้นของการส่งข้อมูล ก็จะเป็นการเริ่มสถานะ "0" ไปหนึ่งช่วงเวลา (บิตเริ่มต้น) จากนั้นก็ตามด้วยบิตข้อมูล 1 อักขระ ซึ่งเป็นบิตนัยสำคัญต่ำสุด (LSB) ก่อน ตามหลังบิตข้อมูลจะเป็นบิตตรวจสอบ (Parity Bit) ซึ่งอาจเป็นแบบคู่หรือแบบคี่ก็ได้ซึ่งหมายความว่า ถ้าแบบคู่ (Even) คือจำนวน "1" ทั้งในบิตข้อมูลและบิตพาริตีรวมกันแล้วเป็นจำนวนคู่ซึ่งเครื่องส่งจะต้องคำนวณหาพาริตีบิตก่อนแล้วส่งไป ส่วนทางด้านรับก็จะคำนวณหาเช่นเดียวกัน ผลที่ได้จึงจะเป็นตามที่คาดไว้คือ ถ้าเราใช้

ระบบพาริตีคู่ ผลรวม "1" ต้องคู่เสมอ ในทำนองเดียวกันแบบคี่ (Odd) คือจำนวน "1" รวมกันแล้วจะเป็นจำนวนคี่ ส่วนบิตนั้นความกว้างเท่ากับ 1, 1.5 หรือ 2 พัลส์ของสัญญาณนาฬิกา

การสื่อสารข้อมูลแบบซิงโครนัส (Synchronous Transmission) เป็นการส่งข้อมูลเป็นกลุ่ม การส่งข้อมูลเป็นการส่งอักขระจำนวนมากซึ่งมีรูปแบบการส่งตามตารางข้างล่าง

SYN	SYN	STX	กลุ่มข้อมูล (Data หรือ Text)	ETB/EXT	BCC บิตตรวจสอบ
8 บิต	8 บิต	8 บิต	N x 8 บิต	8 บิต	16 บิต

เมื่อส่งข้อมูลต้องใส่อักขระ SYN ก่อนหน้ากลุ่มข้อมูลก็ตัวก็ได้ แต่ไม่ควรน้อยกว่า 2 ตัว ตามด้วย ETB บอกถึงการเริ่มต้นข้อมูลและเป็นกลุ่มข้อมูล ปิดท้ายกลุ่มข้อมูลด้วย ซึ่งบอกถึงสิ้นสุดกลุ่มข้อมูล หรือ ETX คือสิ้นสุดการส่งข่าวสาร และสุดท้ายเป็นกลุ่มของบิตตรวจสอบ ในการส่งแบบนี้สถานีรับต้องมีหน่วยความจำสำรอง (Buffer)

การหาค่าประสิทธิภาพ (efficiency) ของการส่งข้อมูลแบบอะซิงโครนัสและแบบซิงโครนัสหาได้จากสูตร

$$\text{ประสิทธิภาพ (efficiency)} = \frac{\text{จำนวนบิตข้อมูล (Data Bit)}}{\text{จำนวนบิตข้อมูล (Total Bit)}} \times 100\%$$

จะเห็นได้ว่าการส่งอักขระแบบอะซิงโครนัสจะใช้เวลามากกว่า เพราะมี บิตต้นทวน (Overhead Bit) ตัวอย่างข้างต้นเป็นการส่งเพียงอักขระเดียว การคิดประสิทธิภาพการส่งต้องพิจารณาอักขระของข้อความทั้งหมดมากกว่า 1 อักขระ โดยคิดจากจำนวนบิตข้อมูลรวมที่ใช้ในข้อความหารด้วยจำนวนบิตรวมทั้งหมด ข้อความได้แก่อักขระที่ใช้ตรวจสอบความผิดพลาดและอักขระต้นทวนอื่น ๆ ในการส่งแบบซิงโครนัสอักขระต้นทวนเหล่านี้ไม่ใช่อักขระข้อมูลจริง ๆ จึงไม่นับเป็นค่าตัวบิต แต่จะนับเป็นข่าวสารรวมจึงทำให้ประสิทธิภาพการส่งลดลง แต่สำหรับการส่งข้อความมาก ๆ นั้น ประสิทธิภาพที่ลดลงนี้จึงน้อยมาก

จากการเปรียบเทียบข้างต้นจะเห็นว่า แบบซิงโครนัสมีประสิทธิภาพสูงกว่าแต่ยังมีปัจจัยอื่น ๆ ที่ต้องนำมาพิจารณาประกอบด้วย เช่น ข้อมูลซิงโครนัสต้องมีการเพิ่มวงจรจับสัญญาณนาฬิกาด้วยโดยทั่วไปข้อมูลแบบอะซิงโครนัสซึ่งมีราคาถูกกว่านั้นควรใช้ในระบบที่มีอัตราข้อมูลต่ำ (Lower Data Rate) เมื่อเวลาไม่ใช่สิ่งวิกฤต ข้อมูลอะซิงโครนัสจึงใช้มากในคอมพิวเตอร์ประจำบ้าน (Home Computers) และโมเด็มที่ 300 หรือ 12000 bps ส่วนในอัตราข้อมูลที่สูง

(Higher Data Rates) จึงมักใช้ข้อมูลแบบซิงโครนัส ระบบข้อมูลซิงโครนัสนั้นเวลาที่เป็นสิ่งวิฤตใช้กันมากเช่นในระบบ Lan และข่ายงานแบบหลายจุด (Multipoint Networks)

2.2 โพรโทคอลการเชื่อมต่อข้อมูล Data Link Protocol

หน้าที่ทั่วไปของโปรโตคอล โปรโตคอลนอกจากจะสร้างกฎให้สถานีสนองตอบต่อการเชื่อมต่อข้อมูลและอักขระอื่น ๆ แล้วยังมีหน้าที่กำหนดสถานีทุติยภูมิที่สถานีแม่หรือสถานีปฐมภูมิเป็นผู้ควบคุมการสื่อสารเมื่อต้องการจะสื่อสารด้วย โดยในระบบการเชื่อมต่อหลายสถานีแบบ Multipoint System ซึ่งสถานีปฐมภูมิ 1 ตัว จะต่อกับสถานีทุติยภูมิหลายตัว สถานีปฐมภูมิสามารถเชื่อมการสื่อสารได้ 2 วิธี คือ (1) สถานีปฐมภูมิสามารถส่งคำถามหรือโพล (poll) ไปยังสถานีทุติยภูมิตัวใดตัวหนึ่งว่ามีข้อความจะส่งให้สถานีปฐมภูมิหรือไม่ (2) หรือสถานีแม่สามารถส่งการเลือก หรือ Selection ให้สถานีทุติยภูมิตัวใดตัวหนึ่งแจ้งว่าพร้อมที่จะรับข้อความหรือ Traffic หรือไม่ สถานีทุติยภูมิสามารถตอบสนองในโหมดใดโหมดหนึ่งคือ ส่ง (send) รับ (receive) หรือ local โดยแจ้งโหมดดังกล่าวตอบต่อโพลหรือซีเล็คชั่นของสถานีแม่ การสนองตอบต่อโพลส่วนใหญ่มีดังนี้

1. สถานีทุติยภูมิอยู่ใน โหมดส่ง และส่งข่าวสารออกไป
2. สถานีทุติยภูมิตอบรับ (Acknowledge หรือ ACK) ต่อโพลแต่อยู่ใน โหมดรับโดยมันจะแจ้ง สถานีปฐมภูมิว่าไม่มีข่าวสารที่จะส่งแต่ตัวเองพร้อมที่จะรับกระแสข้อมูล
3. สถานีทุติยภูมิ ปิดสาย (Off-Line) ซึ่งอยู่ใน โหมดโลคัล โดยมีจะแจ้งไปยังสถานีปฐมภูมิว่ามันอยู่ใน โหมดนี้

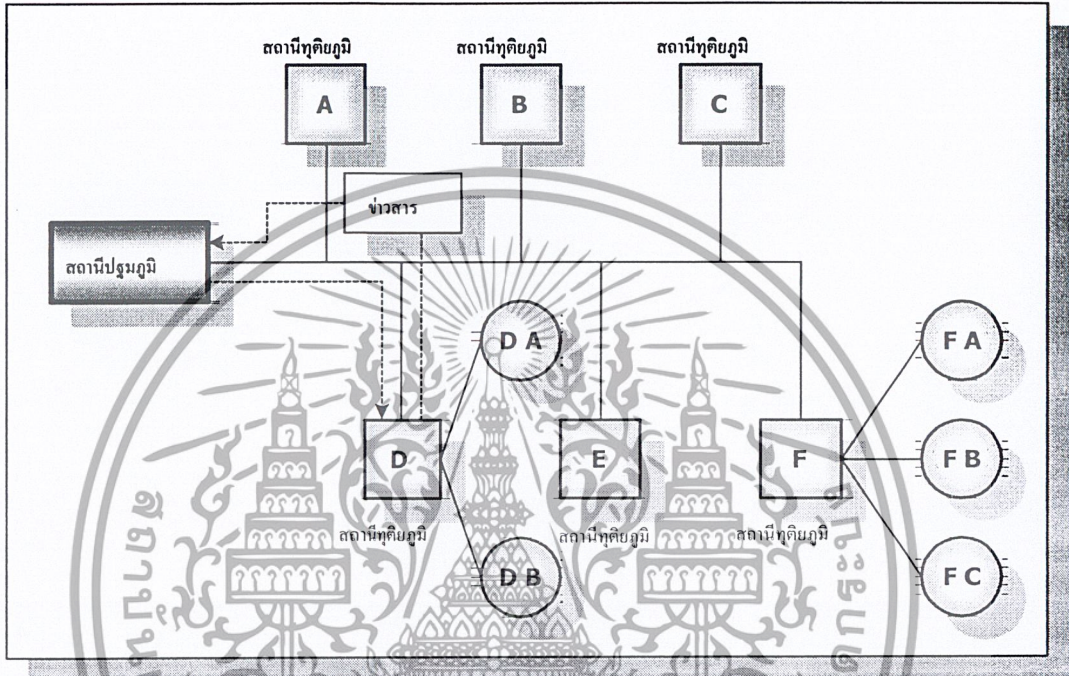
การตอบสนองต่อซีเล็คชั่นส่วนใหญ่จะคล้ายกัน มีต่างไปเล็กน้อยดังนี้

1. สถานีทุติยภูมิใน โหมดรับจะแจ้งไปว่ามันพร้อมที่จะรับกระแสข้อมูลจากสถานีปฐมภูมิ
2. สถานีทุติยภูมิอยู่ใน โหมดส่งและไม่พร้อมที่จะส่งข้อมูล
3. สถานีทุติยภูมิอยู่ใน โหมดโลคัล และไม่พร้อมที่จะส่งหรือรับกระแสข้อมูล

แต่ละโปรโตคอลจะรวมถึงการจัดการเกี่ยวกับ โพลและซีเล็คชั่นและการตอบสนองของสถานีทุติยภูมินอกจากนี้ยังกำหนดให้สถานีทุติยภูมิว่าตัวใดควรจะ โพลหรือซีเล็คชั่น สถานีทุติยภูมิสามารถกำหนดโพลให้กับสถานีทุติยภูมิรับครั้งละสถานีเดี่ยวนั้น

การกำหนดซีเล็คชั่นมี 3 แบบ คือ (1) แต่สถานีทุติยภูมิมีแอดเดรสในการซีเล็คชั่นแอดเดรสโดยเฉพาะ (Unique Selection Address) เพื่อที่จะให้สถานีปฐมภูมิเลือกเฉพาะสถานีใดสถานีหนึ่งเท่านั้น (2) การให้แอดเดรสเฉพาะกลุ่ม (Group Address) คือการกำหนดเฉพาะ

สถานีวิทยุกลุ่มหนึ่งให้สามารถถูกเลือกที่จะรับข่าวสารได้พร้อมกัน โดยสถานีอื่นไม่สามารถรับได้ (3) การให้แอดเดรสทั่วไป (Broadcast Address) โคนทุก ๆ สถานีวิทยุจะถูกเลือกเมื่อให้แอดเดรสแบบนี้ ดังนั้นสถานีปรุมภูมิสามารถส่งข่าวสารเดียวกันไปยังสถานีวิทยุทุก ๆ สถานีได้โดยใช้แอดเดรสนี้



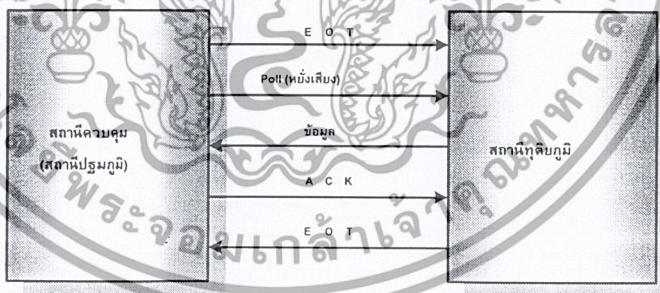
รูปที่ 2.5 การหึ่งเสียงของการเชื่อมโยงการสื่อสารหลายจุด

นอกจากจะสร้างรูปแบบในการโพลและซีเล็กชั่น และกำหนดแอดเดรสให้สถานีวิทยุแล้ว โพรโตคอลยังสามารถกำหนดรูปแบบข้อความและการสนองตอบต่ออักขระการเชื่อมข้อมูลด้วย ตามที่กล่าวมาแล้วว่ารหัสแอสกีประกอบด้วยอักขระ 3 ชนิด คือ ตัวอักษร กราฟฟิก และการเชื่อมข้อมูลอักขระ 2 ชนิดแรกเกี่ยวกับอักขระที่ถูกพิมพ์ออกมาและจะปรากฏบนจอภาพหรือกระดาษที่พิมพ์ไหน อย่างไรก็ตาม อักขระการเชื่อมข้อมูลจะถูกใช้ในการสร้างซอฟต์แวร์เชื่อมการสื่อสาร ได้แก่ ฟังก์ชันต่าง ๆ เช่น เริ่มต้นข้อความ (Start of Text หรือ STX) และสิ้นสุดการส่ง (End of Transmission หรือ EOT) อักขระเหล่านี้จะได้รับการตอบสนองตอบต่างกันขึ้นอยู่กับแต่ละโปรโตคอล แต่ส่วนใหญ่แล้วจะทำหน้าที่พื้นฐานที่จำเป็นคล้าย ๆ กันแต่โปรโตคอลแบบปิด

นำ (Bit oriented Protocols) จะทำให้บิตเดี่ยวหรือกลุ่มบิตที่เจาะจงทำหน้าที่เชื่อมข้อมูลการโพล (Polling)

เป็นการควบคุมการส่งข้อมูลในเครือข่ายการสื่อสารโดยการหยังเสียง ซึ่งจะจัดลำดับให้สถานีทุติยภูมิ ซึ่งมีหลายสถานี แล้วส่งสัญญาณไปสอบถามว่ามีข้อมูลจะส่งมาหรือไม่ตามลำดับการหยังเสียงหรือการสอบถามจะทำที่ละเครื่อง ซึ่งแต่ละสถานีอาจต้องการส่งข่าวสารหรือไม่มีการส่งข่าวสารก็ได้

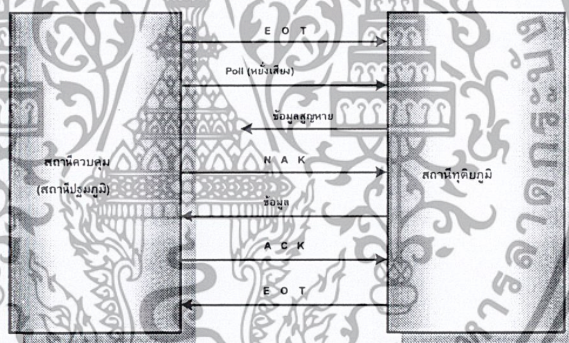
ในระบบการหยังเสียงนี้สถานีทุติยภูมิแต่ละตัวต้องมีรหัสประจำเครื่อง จากภาพข้างบนแสดงให้เห็นการหยังเสียงไปตามสายสื่อสารไปยังทุกสถานี และจะเห็นว่าสถานี D หลังถูกสอบถามและ D ก็ตอบรับโดยการส่งข้อมูลมายังสถานีปฐมภูมิ และสถานี D อาจจะมีสถานีลูกข่ายพ่วงอยู่ได้ ซึ่งเป็นการกำหนดตำแหน่งในระดับรองลงไป เมื่อมีการหยังเสียงเกิดขึ้นสถานีทุติยภูมิทุกสถานีจะรู้ว่ามีสัญญาณส่งมาแต่จะมีสถานีทุติยภูมิเพียงสถานีเดียวเท่านั้นที่จะรับสัญญาณนี้ได้คือเฉพาะสถานีที่มีตำแหน่งรหัสตรงกับตำแหน่งที่ส่งมาพร้อมกับสัญญาณหยังเสียง ถ้าในกรณีที่เป็นการหยังเสียงแบบกลุ่ม เช่น กลุ่ม F (FA, FB, BC) สามารถกระทำก็คือ ส่งการหยังเสียงไปยังเครื่อง F แล้วสถานี F จะส่งการหยังเสียงต่อไปยังสถานีพ่วงต่อไปแต่ถ้าต้องการหยังเสียงสถานีย่อยต่างกลุ่มกันจะทำไมได้เพราะสถานีย่อยของแต่ละกลุ่มอาจต้องการส่งข้อมูลพร้อมกันซึ่งจะมีการชนกันของข้อมูล ทำให้สูญเสีย



รูปที่ 2.6 แสดงถึงการหยังเสียงเพื่อรับข่าวสาร

หลักการทางาน สถานีปฐมภูมิหรือสถานีควบคุมจะเป็นผู้กำหนดสถานะของตัวเองให้เป็นผู้ส่งเองหรือเลือกให้สถานีอื่นเป็นผู้ส่งก็ได้ ซึ่งการหยังเสียงนี้อาจเป็นการกำหนดให้สถานีทุติยภูมิเป็นสถานีส่ง ซึ่งสถานีส่งนี้สามารถเลือกให้สถานีอื่น ๆ ให้เป็นสถานีอื่น ๆ ให้เป็นสถานีรับได้เพื่อจะได้ส่งข้อมูลไปให้ แต่ถ้าสมมติว่าสถานีลูกข่ายถูกหยังเสียงแล้วก็สามารถเลือกสถานีแม่ข่ายเป็นสถานีรับก็ได้

จากรูปเป็นการแสดงให้เห็นถึงลำดับการส่งสัญญาณเมื่อสมมติว่าไม่มีการผลิตพลาตเริ่มแรกสถานีปฐมภูมิหึ่งเสียงข่าสือสารโดยการส่งสัญญาณหึ่งเสียง ซึ่งสัญญาณหึ่งเสียงนี้จะมืรหัสตำแหน่งของทุคยภูมิอยู่ด้วยสถานีทุคยภูมิมีข้อมูล (ซึ่งรวมทั้งกลุ่มบิตตรวจสอบด้วย) ส่งมายังสถานีปฐมภูมิ (กำหนดให้เป็นผู้รับ) จากนั้นถ้าไม่มีข้อผิดพลาดใด ๆ เกิดขึ้นสถานีปฐมภูมิก็จะส่งสัญญาณ Acknowledge ซึ่งเป็นอักขระที่ผู้รับส่งตอบกลับให้ผู้ส่งเพื่อบอกให้รู้ว่าได้รับข่าวสารแล้วเมื่อสถานีส่ง (ในที่นี้คือสถานีทุคยภูมิ) ใ้รับสัญญาณนี้แล้วจะกลับมาด้วยสัญญาณ EOT (End of Transmission) ซึ่งเป็นอักขระที่บอกให้ทราบว่หมดข้อความที่จะส่งแล้ว และหมายถึงให้สายสื่อสารเข้าสู่สภาวะปกติให้พร้อมที่จะรับสัญญาณหึ่งเสียง ในกรณีที่มีการส่งข่าวสารเป็นบล็อคเมื่อสิ้นสุดบล็อคข้อมูลผู้ส่งต้องรอรับสัญญาณ ACK ก่อนที่จะส่งบล็อคต่อไป ถ้าในกรณีที่มีการส่งข้อมูลข่าวสารมีการผิดพลาดอาจเป็นเนื่องมาจากข้อมูลสูญหายระหว่างทาง ซึ่งจะทราบเมื่อหากทางด้านรับไม่ได้รับสัญญาณใด ๆ หลังจากเวลาผ่านไปชั่วขณะหนึ่งแล้วทางด้านรับก็จะส่งสัญญาณ NAK (Negative Acknowledge) มาดังรูปข้างล่าง

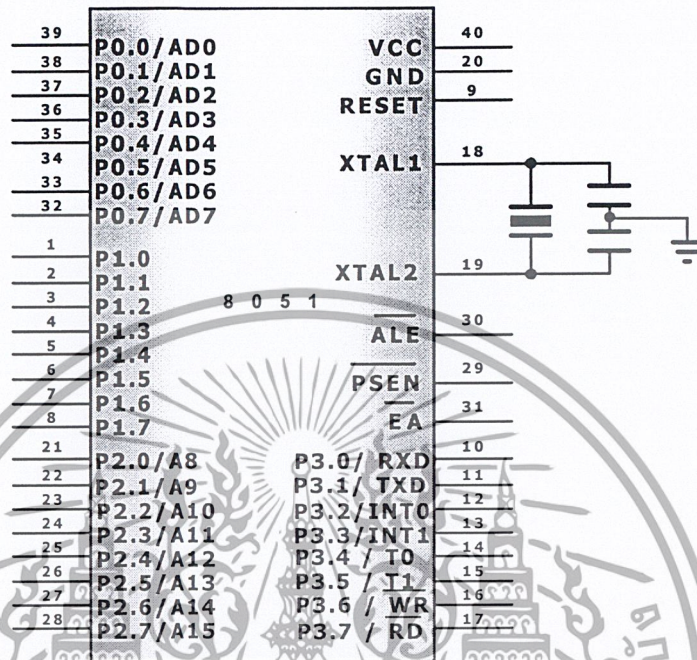


รูปที่ 2.7 แสดงถึงเมื่อการรับส่งข้อมูลไม่สำเร็จ โดยสถานีปฐมภูมิเป็นผู้ส่ง

2.3 ไมโครคอนโทรลเลอร์ MCS-51

ไมโครคอนโทรลเลอร์ในตระกูล MCS-51 ประกอบด้วยไมโครคอนโทรลเลอร์เบอร์ต่าง ๆ ทุกเบอร์มีสถาปัตยกรรมพื้นฐานที่เหมือนกัน เพียงแต่มิขนาดของหน่วยความจำภายในและภายนอกต่างกัน ไมโครคอนโทรลเลอร์ MCS-51 เป็นไมโครคอนโทรลเลอร์ขนาด 8 บิตที่มีอุปกรณ์สนับสนุนประกอบอยู่ภายใน หลายอย่างได้แก่ หน่วยความจำ สำหรับเก็บข้อมูล หน่วยความจำ สำหรับเก็บโปรแกรม ตัวตั้งเวลา/ตัวนับ อุปกรณ์รับส่งข้อมูลแบบอนุกรม เนื่องจากโครงสร้างของไมโครคอนโทรลเลอร์มีอุปกรณ์สนับสนุนประกอบอยู่ภายในนี้เอง ทำให้การใช้งานง่ายขึ้นและมี

ประสิทธิภาพมากขึ้น โดยไม่ต้องมีการเชื่อมต่ออุปกรณ์ภายนอกเพิ่มเติมมากเหมือนกับตัวไมโครโปรเซสเซอร์ทั่วไป



รูปที่ 2.8 ขามาตรฐานของไมโครคอนโทรลเลอร์ MCS-51

หน้าที่และการใช้งานของขาสัญญาณต่างๆ มีดังนี้

- VCC สำหรับต่อกับไฟเลี้ยง 5 โวลต์
- VSS สำหรับต่อกับกราวด์
- XTAL1 เป็นอินพุตของภาคขยายสัญญาณแบบอินเวอร์สของวงจรมัลติสัญญาณนาฬิกา
- XTAL2 เป็นเอาต์พุตของภาคขยายสัญญาณแบบอินเวอร์สของวงจรมัลติสัญญาณนาฬิกา
- RST ขาสัญญาณรีเซ็ต ไมโครคอนโทรลเลอร์จะถูกรีเซ็ตเมื่อสัญญาณที่ขา RST นี้มีค่าเป็นลอจิก 1 นานไม่ต่ำกว่า 2 แมกซ์ซีไอเกิล
- ALE/ \overline{PROC} : เป็นสัญญาณเอาต์พุต ซึ่งไมโครคอนโทรลเลอร์จะส่งออกไปเป็นพัลส์เพื่อเลขาท์ค่าตำแหน่งไบต์ต่ำที่อยู่พอร์ต P0 ในขณะที่ติดต่อกับหน่วยความจำภายนอก สัญญาณนี้จะถูกส่งออกไปด้วยอัตราคงที่คือ 1/6 เท่าของความถี่สัญญาณนาฬิกาซึ่งสามารถไปใช้เป็นสัญญาณนาฬิกาให้อุปกรณ์ภายนอกได้ สัญญาณพัลส์นี้จะถูกข้ามไป 1 พัลส์เมื่อมีการติดต่อกับหน่วยความจำข้อมูลภายนอก และสัญญาณนี้จะใช้เป็นอินพุตเพื่อควบคุมการโปรแกรม PROM ภายในตัวไมโครคอนโทรลเลอร์ด้วย

PSEN	เป็นเอาต์พุตสำหรับส่งสัญญาณสไตรป เพื่ออ่านข้อมูลจากหน่วยความจำโปรแกรมภายนอก เมื่อซีพียูอ่านรหัสคำสั่งจากหน่วยความจำภายนอก จะส่งสัญญาณสไตรปออกมา 2 ครั้งใน 1 แมกซ์ซินไซเคิลแต่สัญญาณสไตรปทั้ง 2 ครั้งจะถูกข้ามไปหากเป็นช่วงที่ซีพียูติดต่อกับ External Data Memory
EA	เป็นสัญญาณอินพุต ใช้สำหรับควบคุมให้ไมโครคอนโทรลเลอร์เลือกติดต่อกับโปรแกรมที่เก็บอยู่ในหน่วยความจำโปรแกรมภายใน หรือโปรแกรมที่เก็บอยู่ในหน่วยความจำภายนอกตัวไมโครคอนโทรลเลอร์ หากต้องการให้ซีพียูติดต่อกับหน่วยความจำโปรแกรมภายนอกต้องต่อสัญญาณเข้ากับลอจิก 0 ถ้าต้องการติดต่อกับหน่วยความจำโปรแกรมภายในก็ให้ขานี้เป็นลอจิก 1
Port 0	เป็นพอร์ตอินพุตเอาต์พุต 2 ทางแบบ Open drain ขนาด 8 บิต P0.1 - P0.7 เมื่อใช้เป็นเอาต์พุตสามารถต่อกับไอซี TTL ตระกูล LS ได้ 8 ตัว เมื่อต้องการใช้งานเป็นอินพุตต้องส่งค่าลอจิก 1 ออกไปก่อนเพื่อให้ลอยซึ่งจะเป็นอิมพีแดนซ์สูง พอร์ต P0 จะทำงานอีกหน้าที่หนึ่งคือ เป็นมัลติเพล็กซ์ของสัญญาณตำแหน่งด้านต่ำ และสัญญาณข้อมูลในการติดต่อกับหน่วยความจำภายนอก การทำงานในลักษณะนี้จะใช้การพูลอัพ จากภายในที่สามารถจ่ายกระแสให้กับอินพุตของ TTL ได้ 8 ตัว นอกจาก 2 หน้าที่ดังกล่าวแล้ว พอร์ต P0 ยังสามารถใช้เป็นตัวรับข้อมูลในช่วงการโปรแกรม EPROM และเป็นตัวส่งข้อมูลออกมในช่วงการตรวจสอบโปรแกรมภายใน ROM หรือ EPROM ซึ่งจะต้องใช้พูลอัพจากภายนอกในขณะที่ทำการตรวจสอบโปรแกรม
Port 1	เป็นพอร์ตอินพุตเอาต์พุตแบบ 2 ทาง ขนาด 8 บิต ที่มีพูลอัพอยู่ภายใน ในกรณีเอาต์พุตจะต่อกับอินพุตของ TTL ตระกูล LS ได้ 4 ตัว เมื่อต้องการใช้เป็นอินพุตต้องเริ่มต้นด้วยการส่งค่าลอจิก 1 ออกไปที่พอร์ตนี้ก่อน เพื่อทำให้เกิดพูลอัพภายใน เมื่อมีสัญญาณอินพุตเป็น 0 เข้ามาจะทำให้ พอร์ตจ่ายกระแสออกเนื่องจากการพูลอัพอยู่ภายใน นอกจากนี้พอร์ต P1 ยังทำหน้าที่รับตำแหน่งด้านต่ำในช่วงของการโปรแกรม EPROM และช่วงการตรวจสอบโปรแกรมใน ROM หรือ EPROM
Port 2	เป็นพอร์ตอินพุตเอาต์พุตแบบ 2 ทาง ขนาด 8 บิต ที่มีพูลอัพอยู่ภายใน ในกรณีเอาต์พุตจะต่อกับอินพุตของ TTL ตระกูล LS ได้ 4 ตัว เมื่อต้องการใช้เป็นอินพุตต้องเริ่มต้นด้วยการส่งค่าลอจิก 1 ออกไปที่พอร์ตนี้ก่อน เพื่อทำให้เกิดพูลอัพภายใน เมื่อมีสัญญาณอินพุตเป็น 0 เข้ามาจะทำให้ พอร์ตจ่ายกระแสออกเนื่องจากการ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การพูลอ์พอยู่ภายใน ในระหว่างการติดต่อกับโปรแกรมภายนอกหรือการติดต่อกับหน่วยความจำข้อมูลภายนอกที่มีการอ้างตำแหน่งแบบ 16 บิต พอร์ต P2 จะส่งตำแหน่งไบต์สูงออกไป ซึ่งการทำงานในลักษณะนี้จะมีการพูลอ์พภายในอยู่ในช่วงของการติดต่อกับหน่วยความจำข้อมูลภายนอกที่ใช้การอ้างตำแหน่งแบบ 8 บิต สัญญาที่ขาของพอร์ต P2 จะมีค่าเท่ากับรีจิสเตอร์ P2 ที่อยู่ใน SFR นอกจากนี้พอร์ต P2 ยังทำหน้าที่รับตำแหน่งไบต์สูง ในช่วงของการโปรแกรมและการตรวจสอบโปรแกรมใน ROM และ EPROM

พอร์ต	สัญญา	หน้าที่
P3.0	RXD	อินพุตของพอร์ตอนุกรม
P3.1	TXD	เอาต์พุตของพอร์ตอนุกรม
P3.2	INT0	สัญญาอินเตอร์รัพต์จากภายนอกตัวที่ 0
P3.3	INT1	สัญญาอินเตอร์รัพต์จากภายนอกตัวที่ 1
P3.4	TO	อินพุตจากภายนอกของตัวตั้งเวลา 0
P3.5	T1	อินพุตจากภายนอกของตัวตั้งเวลา 1
P3.6	WR	สัญญาเขียนข้อมูลออกไปภายนอก
P3.7	RD	สัญญาการอ่านข้อมูลจากภายนอกเข้ามา

ตารางที่ 2.1 สัญญาต่าง ๆ ของพอร์ต P3

Port 3

เป็นพอร์ตอินพุต เอาต์พุตแบบ 2 ทาง ขนาด 8 บิต ที่มีพูลอ์พอยู่ภายใน ในกรณีเอาต์พุตจะต่อกับอินพุตของ TTL ตระกูล LS ได้ 4 ตัว เมื่อต้องการใช้เป็นอินพุตต้องเริ่มต้นด้วยการส่งค่าลอจิก 1 ออกไปที่พอร์ตนี้ก่อน เพื่อทำให้เกิดพูลอ์พภายใน เมื่อมีสัญญาอินพุตเป็น 0 เข้ามาจะทำให้ พอร์ตจ่ายกระแสออกเนื่องจากการพูลอ์พอยู่ภายใน นอกจากนี้พอร์ต P3 ยังทำหน้าที่เป็นสัญญาอื่น ๆ อีก เมื่อต้องการใช้งานพอร์ต P3 ให้ทำหน้าที่เป็นสัญญาต่าง ๆ จะต้องเริ่มต้นด้วยการส่งค่าลอจิก 1 ออกไปแลทซ์ที่พอร์ต P3 ก่อนเพื่อให้เกิดการพูลอ์พภายใน หากเรากำหนดให้มีค่าลอจิก 0 จะทำให้สัญญาที่ขาต่าง ๆ มีค่าเป็น 0 ตลอดเวลา

2.3.1 หน่วยความจำของไมโครคอนโทรลเลอร์ MCS-51

หน่วยความจำของไมโครคอนโทรลเลอร์ MCS-51 จะแบ่งเป็น 3 กลุ่ม คือ หน่วยความจำสำหรับเก็บโปรแกรม หน่วยความจำสำหรับเก็บข้อมูล และรีจิสเตอร์ที่ทำหน้าที่เฉพาะ

1. หน่วยความจำสำหรับเก็บโปรแกรม

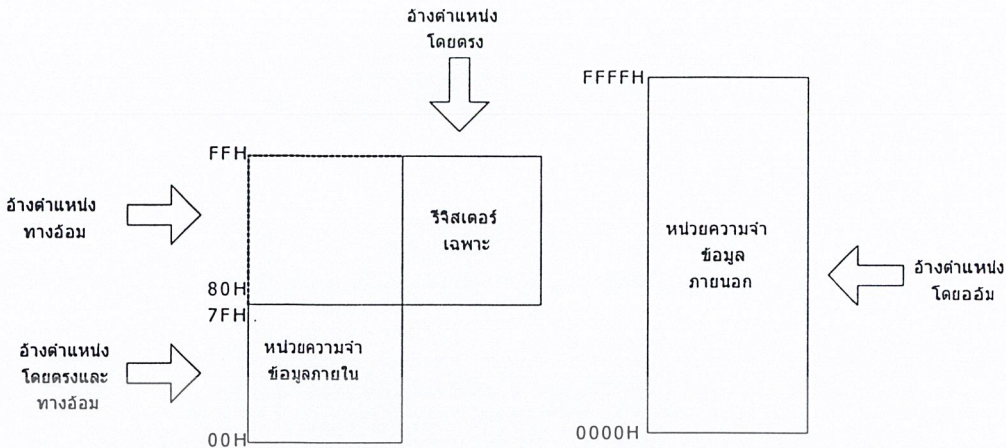
ใช้เก็บข้อมูลของโปรแกรมควบคุมการทำงานของไมโครคอนโทรลเลอร์ซึ่งหน่วยความจำจะเป็นลักษณะของ ROM การติดต่อกับหน่วยความจำโปรแกรมของไมโครคอนโทรลเลอร์ 8051 สามารถติดต่อกับหน่วยความจำโปรแกรมที่อยู่ภายในไมโครคอนโทรลเลอร์หรือโปรแกรมที่เก็บอยู่ภายนอกก็ได้ โดยมีจำนวนข้อมูลทั้งหมด 64 Kbytes สามารถเลือกใช้งานได้ 2 ลักษณะคือเลือกใช้หน่วยความจำภายในตัวไมโครคอนโทรลเลอร์ที่มีอยู่ 4 Kbyte รวมกับหน่วยความจำภายนอกอีก 60 Kbyte หรือเลือกใช้หน่วยความจำภายนอกทั้งหมด 64 Kbyte



รูปที่ 2.9 การเลือกใช้หน่วยความจำโปรแกรมของ 8051

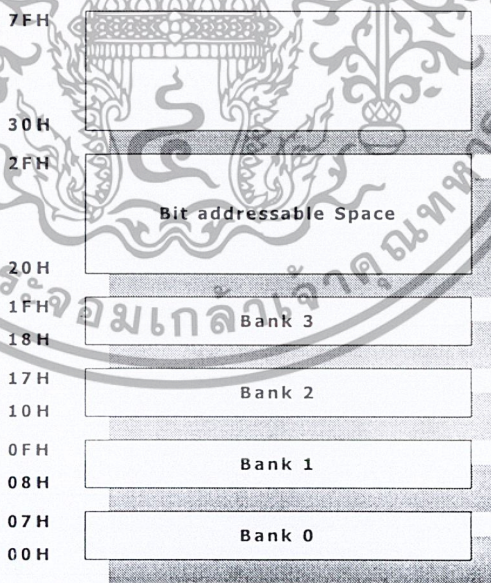
2. หน่วยความจำข้อมูล

หน่วยความจำข้อมูลทำหน้าที่เก็บข้อมูลต่าง ๆ ในขณะที่โปรแกรมทำงานไว้ชั่วคราว และทำหน้าที่เป็นสแตคบางส่วน ซึ่งก็คือหน่วยความจำข้อมูลแบบ RAM จัดแบ่งหน่วยความจำข้อมูลออกเป็น 2 ส่วนคือ หน่วยความจำข้อมูลภายในและหน่วยความจำข้อมูลภายนอก



รูปที่ 2.10 หน่วยความจำข้อมูลและการอ้างตำแหน่งหน่วยความจำ

หน่วยความจำข้อมูลภายในสำหรับเบอร์ 8051 มีหน่วยความจำข้อมูลภายใน 128 ไบต์ โดยเป็นหน่วยความจำที่ใช้งานอยู่ที่ตำแหน่ง 00H-07FH หน่วยความจำในตำแหน่งที่อยู่สูงขึ้นไปตำแหน่ง 80H-FFH จะมีส่วนซ้อนทับกันอยู่ระหว่างหน่วยความจำกับรีจิสเตอร์เฉพาะ การจัดแบ่งหน่วยความจำข้อมูลภายใน ยังสามารถแบ่งออกเป็นส่วนย่อยได้อีก 3 ส่วน



รูปที่ 2.11 การจัดแบ่งหน่วยความจำข้อมูลภายใน

ส่วนที่ 1 ตำแหน่ง 00F-7F จำนวน 32 ไบต์จะแบ่งออกเป็น 4 แบนด์แต่ละแบนด์มี 8 ตำแหน่ง พื้นที่ในแต่ละแบนด์ถูกใช้เป็นที่รีจิสเตอร์ที่ใช้งานทั่วไป โดย R0 อยู่ในตำแหน่งแรกของแบนด์และ R7 อยู่ในตำแหน่งสุดท้ายของแบนด์ การเลือกตำแหน่งที่ใช้งานของรีจิสเตอร์ R0-R7 จะกำหนดจากบิต RS0 และ RS1 ที่อยู่ในตัวรีจิสเตอร์ PSW

ส่วนที่ 2 ตำแหน่ง 20H-2FH เป็นส่วนที่สามารถใช้งานในลักษณะไบต์หรือบิตได้ คือเราสามารถอ้างตำแหน่งแบบบิตได้โดยตรง ซึ่งมีจำนวนทั้งหมด 128 บิต การอ้างตำแหน่งแบบบิตทำให้การทำงานได้รวดเร็วขึ้น ซึ่งเป็นคุณสมบัติของไมโครคอนโทรลเลอร์ MCS-51 ที่ถูกออกแบบมาเพื่องานควบคุมโดยเฉพาะ

ส่วนที่ 3 ตำแหน่ง 30H-7FH เป็นพื้นที่ที่ใช้งานทั่วไปการติดต่อกับข้อมูลในตำแหน่งต่าง ๆ ในส่วนนี้ สามารถอ้างตำแหน่งข้อมูลได้ในลักษณะของไบต์เท่านั้น ในพื้นที่ส่วนนี้เราอาจใช้เป็นสแต็กก็ได้

3. รีจิสเตอร์เฉพาะ (Special Function Register)

รีจิสเตอร์ต่าง ๆ ที่ทำหน้าที่เฉพาะอย่างซึ่งเรียกว่า Special Function Register อยู่ในตำแหน่งหน่วยความจำตั้งแต่ 80F – FFH การติดต่อกับรีจิสเตอร์เหล่านี้เราสามารถอ้างถึงตำแหน่งที่อยู่ได้โดยตรง (Direct addressing) และบางตัวสามารถอ้างตำแหน่งข้อมูลแบบบิตได้ โดยมีรีจิสเตอร์ต่าง ๆ ดังต่อไปนี้

ACC : แอคคิวมูเลเตอร์มีขนาด 8 บิตทำหน้าที่เช่นเดียวกับแอกคิวมูเลเตอร์ของซีพียูอื่น ๆ คือใช้เป็นตัวกระทำร่วมทางคณิตศาสตร์ของเลข 2 จำนวน เช่น การ บวก ลบ คูณ หาร และทำหน้าที่เป็นตัวเก็บผลลัพธ์ที่ได้จากการคำนวณทางคณิตศาสตร์ นอกจากนี้ ACC ยังทำหน้าที่ในการรับส่งข้อมูลระหว่างไมโครคอนโทรลเลอร์กับอุปกรณ์อินพุต หรือเอาต์พุตภายนอก หรือหน่วยความจำข้อมูลภายนอกด้วย ซึ่งการส่งข้อมูลจะต้องนำข้อมูลที่ต้องการส่งออกไปภายนอกมาเก็บไว้ที่แอกคิวมูเลเตอร์ก่อนแล้วจึงส่งค่าที่อยู่ในแอกคิวมูเลเตอร์ส่งออกไป ในการรับข้อมูลจากภายนอกเข้ามายังไมโครคอนโทรลเลอร์จะรับเข้ามาเก็บไว้ที่แอกคิวมูเลเตอร์ก่อนแล้วจึงนำไปใช้งานต่อไปได้

Register B : เป็นรีจิสเตอร์ที่ใช้งานเฉพาะในคำสั่งการคูณและการหารเท่านั้น โดยจะทำงานร่วมกับแอกคิวมูเลเตอร์ ในการเก็บผลลัพธ์ของการคูณและเศษที่ได้จากการหาร

SP : Stack Pointer มีขนาด 8 บิต ทำหน้าที่ชี้ตำแหน่งสแต็ก ซึ่งไมโครคอนโทรลเลอร์ MCS-51 ใช้พื้นที่ของหน่วยความจำภายในเป็นสแต็กโดยตำแหน่งเริ่มต้นหลังจากการรีเซตจะเป็น 07 การทำงานของ SP จะเพิ่มขึ้นหรือลดลงอย่างอัตโนมัติเมื่อมีการเก็บข้อมูลลงในสแต็กหรือนำข้อมูลออกจากสแต็ก การเก็บข้อมูลลงในสแต็กจะเก็บได้ครั้งละ 8 บิต โดยใช้คำสั่ง PUSH โดยค่าของ SP จะเพิ่มขึ้น 1 แล้วเก็บข้อมูลลงในสแต็กที่ตำแหน่งที่ SP ชี้อยู่ ซึ่งจะทำให้ตำแหน่งแรกของการ

เก็บข้อมูลของสแตคคือ 08 การนำข้อมูลออกจากสแตค ทำได้จากการทำคำสั่ง POP โดยจะดึงข้อมูลออกจากสแตคในตำแหน่งที่ SP ชี้อยู่ออกไปกำหนดให้กับหน่วยความจำที่รับข้อมูล แล้วค่าของสแตคจะลดลง 1 หลังจากมีการนำข้อมูลออกจากสแตคไปแล้ว และเมื่อมีการทำคำสั่งการเรียกโปรแกรมย่อย ACALL หรือ LCALL หรือการตอบสนองการอินเทอร์รัพต์จะมีการเก็บค่าของรีจิสเตอร์ PC ลงในสแตค และเมื่อทำคำสั่งการกลับจากโปรแกรมย่อย RET หรือ RETI จะมีการดึงค่าจากสแตคมากำหนดให้กับ PC ตามเดิมในการใช้งานเราสามารถกำหนดพื้นที่ของสแตคไปอยู่ในตำแหน่งใด ๆ ในหน่วยความจำข้อมูลภายในตัว ไมโครคอนโทรลเลอร์ได้โดยการกำหนดค่าให้กับ SP ตามที่ต้องการแต่ไม่ควรใช้ในตำแหน่งเริ่มต้น 00 เนื่องจากเป็นตำแหน่งเดียวกับรีจิสเตอร์ R0 ถึง R7 ในแบงก์ 0 และไม่ควรกำหนดตำแหน่งสูงเกินไป เพราะจะทำให้พื้นที่ของสแตคน้อยซึ่งอาจไม่เพียงพอกับใช้งาน

DPTR : Data Pointer เป็นรีจิสเตอร์ขนาด 16 บิตใช้สำหรับเป็นตัวชี้ตำแหน่งของหน่วยความจำภายนอกหรือตำแหน่งของอุปกรณ์อื่นพุดที่ไมโครคอนโทรลเลอร์ต้องการติดต่อด้วย และใช้เป็นตัวกำหนดตำแหน่งเริ่มต้นของตารางในการทำงานเกี่ยวกับ Loop up table รีจิสเตอร์ DPTR ประกอบด้วยรีจิสเตอร์ขนาด 8 บิต 2 ตัวคือรีจิสเตอร์ DPH และ DPL ซึ่งสามารถเลือกการใช้งานในลักษณะ 8 บิต 2 ตัวหรือ 16 บิต 1 ตัวก็ได้

P0-P3 : พอร์ต P0 – พอร์ต P3 เป็นรีจิสเตอร์ขนาด 8 บิต ซึ่งค่าที่อยู่ในรีจิสเตอร์เหล่านี้จะเป็นค่าเดียวกับค่าของสัญญาณที่ขาต่าง ๆ ของพอร์ต ดังนั้นการส่งข้อมูลออกไปที่พอร์ตจะทำใจได้ โดยการกำหนดค่าให้กับรีจิสเตอร์ P0 – P3 ในกรณีที่ใช้พอร์ตเป็นอินพุตเราสามารถอ่านสัญญาณที่ต่อกับขาของพอร์ตได้จากการอ่านค่าของรีจิสเตอร์ P0 – P3 เช่นกัน

SBUF : เป็นรีจิสเตอร์ขนาด 8 บิต ทำหน้าที่เป็นบัฟเฟอร์ของการรับส่งข้อมูลแบบอนุกรม โดยทั้งบัฟเฟอร์ด้านรับข้อมูลและบัฟเฟอร์ด้านส่งข้อมูล บัฟเฟอร์ด้านรับข้อมูลจะทำหน้าที่เก็บข้อมูลที่ส่งเข้ามาให้กับไมโครคอนโทรลเลอร์ทางพอร์ตอนุกรม เมื่อข้อมูลเข้ามาครบจำนวนบิตแล้วค่าที่อยู่ในบัฟเฟอร์ตัวนี้จะถูกไมโครคอนโทรลเลอร์อ่านออกไปอีกทีหนึ่ง สำหรับบัฟเฟอร์ด้านส่งข้อมูลจะรับข้อมูลขนาด 8 บิตที่ไมโครคอนโทรลเลอร์ต้องการส่งออกไปทางพอร์ตอนุกรม โดยข้อมูลในบัฟเฟอร์จะถูกส่งออกไปยังพอร์ตอนุกรมทีละบิตต่อไป

IP : Interrupt Priority Control เป็นรีจิสเตอร์ขนาด 8 บิตทำหน้าที่ควบคุมการจัดลำดับความสำคัญของการอินเทอร์รัพต์ ซึ่งจะอธิบายรายละเอียดในเรื่องอินเทอร์รัพต์ต่อไป

IE : Interrupt Enable เป็นรีจิสเตอร์ขนาด 8 บิตทำหน้าที่ควบคุมการส่งสัญญาณและการตอบรับอินเทอร์รัพต์ของไมโครคอนโทรลเลอร์ ซึ่งจะอธิบายรายละเอียดในเรื่องอินเทอร์รัพต์ต่อไป

TMOD : Time Mode Control เป็นรีจิสเตอร์ขนาด 8 บิตทำหน้าที่ควบคุมการเลือกโหมดการทำงานของ Time/Counter0 และ Time/Counter1

TCON : Time Control เป็นรีจิสเตอร์ขนาด 8 บิตใช้สำหรับควบคุมการทำงานของ Time/Counter0 และ Time/Counter1

T2CON : เป็นรีจิสเตอร์ขนาด 8 บิตใช้สำหรับควบคุมการทำงานของ Time/Counter2

SCON : Serial Control เป็นรีจิสเตอร์ขนาด 8 บิตใช้สำหรับควบคุมการทำงานของพอร์ตอนุกรมซึ่งจะอธิบายรายละเอียดในเรื่องการรับส่งข้อมูลแบบอนุกรม

PCON : เป็นรีจิสเตอร์ทำหน้าที่ควบคุมการใช้กำลังไฟของตัวไมโครคอนโทรลเลอร์

PSW : Program Status Word เป็นรีจิสเตอร์ขนาด 8 บิตทำหน้าที่แสดงสถานะการทำงานของโปรแกรมหรือเรียกว่าแฟลก ซึ่งจะมีการเปลี่ยนแปลงหลังจากมีการทำงานในคำสั่งต่าง ๆ ที่มีผลต่อแฟลก และยังใช้เป็นตัวเลือกตำแหน่งแ่งค์ของรีจิสเตอร์ R0 – R 7 อีกด้วย ผลของบิตต่าง ๆ สามารถนำไปเป็นเงื่อนไขในการกระโดดได้ ค่าของบิตต่าง ๆ ใน PSW สามารถเซตและเคลียด้วยคำสั่งทางซอฟต์แวร์ได้ แฟลกต่าง ๆ ของรีจิสเตอร์ PSW จะอยู่ในตำแหน่งของบิตต่าง ๆ ดังนี้

PSW.7	PSW.6	PSW.5	PSW.4	PSW.3	PSW.2	PSW.1	PSW.0
CY	AC	FO	RS1	RS0	OV	-	P

-CY คือ Carry Flag เป็นแฟลกตัวทศทำหน้าที่หลายอย่าง เช่นเป็นตัวทศในกรณีของการบวก ทำหน้าที่เป็นตัวยืมในกรณีของการลบ

-AC คือ Auxiliary Carry Flag เป็นแฟลกตัวทศช่วยระหว่าง บิตที่ 3 และบิตที่ 4 ซึ่งแฟลกนี้จะถูกเซตเป็น 1 หากทำการบวกเลขแล้วมีการทศจากบิตที่ 3 ไปยังบิตที่ 4 หากไม่มีการทศแฟลกนี้จะ เป็น 0

-FO คือ Flag 0 เป็นแฟลกที่ใช้งานทั่วไปซึ่งเราสามารถใช้เป็นแฟลกสถานะ (Status flag) ของโปรแกรมได้โดยการเซตหรือรีเซตด้วยคำสั่งทางซอฟต์แวร์

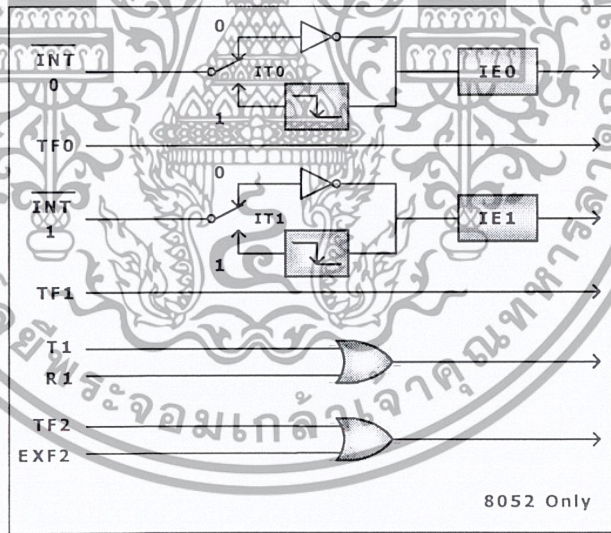
-RS1 และ RS0 คือ Register Bank Select เป็นตัวกำหนดการเลือกพื้นที่ใช้งานของกลุ่มรีจิสเตอร์ว่าอยู่ในแบงก์ไหน

-OV คือ Overflow Flag เป็นแฟลกที่แสดงค่าโอเวอร์โฟลว์ซึ่งจะถูกเซตหรือเคลียจากการทำงานของคำสั่งทางคณิตศาสตร์ ถ้าผลลัพธ์ที่ได้มีค่ามากกว่า +127 หรือน้อยกว่า -128 แฟลกนี้จะมีถูกเซต

-P คือ Parity Flag เป็นแฟล็กที่แสดงจำนวนบิตที่เป็น 1 ในแอสกีวูเลเตอร์ โดยแฟล็ก P จะถูกเซตเป็น 1 เมื่อแอสกีวูเลเตอร์มีจำนวนบิตที่เป็น 1 เป็นคี่และถูกเคลียเมื่อมีจำนวนบิต 1 เป็นคู่

2.3.2 การใช้งานอินเทอร์รัพต์ของ MCS-51

การติดต่อกันระหว่างไมโครคอนโทรลเลอร์กับอุปกรณ์ภายนอกสามารถทำได้ 2 ลักษณะ คือ ใช้วิธีการโพลลิง (Polling) ซึ่งไมโครคอนโทรลเลอร์ต้องคอยตรวจสอบอุปกรณ์ภายนอกอยู่ตลอดเวลาว่ามีข้อมูลที่ต้องการติดต่อกับไมโครคอนโทรลเลอร์หรือไม่ ซึ่งวิธีนี้จะทำให้ไมโครคอนโทรลเลอร์ต้องเสียเวลาไปกับการตรวจสอบนานมาก หากมีอุปกรณ์จำนวนมากต่ออยู่กับ ไมโครคอนโทรลเลอร์ การติดต่อแบบที่สองเป็นการใช้อินเทอร์รัพต์ซึ่งเป็นวิธีการที่ไมโครคอนโทรลเลอร์ไม่ต้องคอยตรวจสอบอุปกรณ์ภายนอกอยู่ตลอด แต่จะให้อุปกรณ์ภายนอกส่งสัญญาณอินเทอร์รัพต์เข้ามาให้ไมโครคอนโทรลเลอร์เมื่ออุปกรณ์ภายนอกต้องการติดต่อกับไมโครคอนโทรลเลอร์ เมื่อไมโครคอนโทรลเลอร์ได้รับสัญญาณอินเทอร์รัพต์แล้วจึงจะทำการรับส่งสัญญาณส่งข้อมูลกับอุปกรณ์ภายนอกนั้น ๆ ต่อไป



รูปที่ 2.12 แหล่งกำเนิดสัญญาณอินเทอร์รัพต์ของ MCS-51

แหล่งกำเนิดสัญญาณอินเทอร์รัพต์ของ 8051 มี 5 แหล่งแต่ละแหล่งจะมาจากอุปกรณ์ต่าง ๆ ดังนี้

สัญญาณรบกวนอินเทอร์รัพต์	ความหมาย
$\overline{INT0}$	สัญญาณรบกวนอินเทอร์รัพต์จากภายนอกทางขา P3.2
$\overline{INT1}$	สัญญาณรบกวนอินเทอร์รัพต์จากภายนอกทางขา P3.3
TF0	สัญญาณรบกวนอินเทอร์รัพต์จาก Timer0 ที่อยู่ในตัวไมโครคอนโทรลเลอร์
TF1	สัญญาณรบกวนอินเทอร์รัพต์จาก Timer1 ที่อยู่ในตัวไมโครคอนโทรลเลอร์
RI, TI	สัญญาณรบกวนอินเทอร์รัพต์จากพอร์ตอนุกรมที่อยู่ในตัวไมโครคอนโทรลเลอร์
TF2, Exf2	สัญญาณรบกวนอินเทอร์รัพต์จาก Timer2

การควบคุมอินเทอร์รัพต์

สัญญาณต่างๆ ของการรบกวนอินเทอร์รัพต์ทั้งหมด เราสามารถเซตหรือเคลียได้ด้วยซอฟต์แวร์เช่นเดียวกับฮาร์ดแวร์ ดังนั้นการรบกวนอินเทอร์รัพต์จึงสามารถสร้างหรือยกเลิกได้ด้วยซอฟต์แวร์ แผลงกำเนิดสัญญาณอินเทอร์รัพต์ทั้งหมด เราสามารถควบคุมให้ทำการร้องขอ อินเทอร์รัพต์หรือไม่ก็ได้ โดยการเซตหรือเคลียบิตต่างๆ ที่อยู่ในรีจิสเตอร์ IE (Interrupt Enable) ภายในรีจิสเตอร์ IE มีบิต EA ที่ทำหน้าที่ควบคุมให้ไมโครคอนโทรลเลอร์ตอบรับการรบกวนอินเทอร์รัพต์หรือไม่ตอบรับทั้งหมด หากเรากำหนดให้บิต EA เป็น 0 ไมโครคอนโทรลเลอร์ จะไม่ตอบรับการรบกวนอินเทอร์รัพต์ทั้งหมด

IE : INTERRUPT ENABLE REGISTER (Bit addressable)

IE.7 IE.6 IE.5 IE.4 IE.3 IE.2 IE.1 IE.0

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

- EA IE.7 (disable all interrupt) หากกำหนดให้บิตนี้มีค่าเป็น 0 จะไม่มีการตอบรับอินเทอร์รัพต์ทั้งหมด หากต้องการให้มีการตอบรับอินเทอร์รัพต์ จะต้องเซตให้ EA เป็น 1 และเซตบิตควบคุมแต่ละอินเทอร์รัพต์ที่ต้องการใช้อินเทอร์รัพต์ให้เป็น 1 โดยใช้คำสั่ง เซตหรือเคลียบิต
- IE.6 ไม่ใช้งาน

- ET2 IE.5 ควบคุมให้ Timer2 หรือ Capture ส่งอินเทอร์รัพต์หรือไม่
(1=ส่งอินเทอร์รัพต์)
- ES IE.4 ควบคุมให้พอร์ตอนุกรมส่งอินเทอร์รัพต์หรือไม่ (1 = ส่งอินเทอร์รัพต์)
- ET1 IE.3 ควบคุมให้ Timer1 ส่งอินเทอร์รัพต์เมื่อถึงค่าที่ตั้งไว้หรือไม่
(1 = ส่งอินเทอร์รัพต์)
- EX1 IE.2 ควบคุมให้ตอบรับอินเทอร์รัพต์จากภายนอก $\overline{INT1}$ หรือไม่
(1 = ตอบรับอินเทอร์รัพต์)
- ETO IE.1 ควบคุมให้ Timer0 ส่งอินเทอร์รัพต์เมื่อถึงค่าที่ตั้งไว้หรือไม่
(1 = ส่งอินเทอร์รัพต์) ควบคุมให้ตอบรับอินเทอร์รัพต์จากภายนอก $\overline{INT0}$
หรือไม่ (1 = ตอบรับอินเทอร์รัพต์)

หากต้องการใช้สัญญาณอินเทอร์รัพต์จากแหล่งกำเนิดใด จะต้องเซตให้บิตควบคุมเป็น 1 และต้องเซตให้ EA เป็น 1 ด้วย

การจัดลำดับความสำคัญของอินเทอร์รัพต์

การจัดลำดับความสำคัญของการอินเทอร์รัพต์ของไมโครคอนโทรลเลอร์ จัดได้ 2 ระดับ แตกต่างกันโดยการเซตค่าในรีจิสเตอร์ควบคุมการจัดลำดับความสำคัญของการอินเทอร์รัพต์ ในกรณีที่ไม่มีการจัดลำดับความสำคัญของการอินเทอร์รัพต์หรือจัดให้มีความสำคัญในระดับเดียวกัน ไมโครคอนโทรลเลอร์จะจัดให้มีความสำคัญของสัญญาณอินเทอร์รัพต์ที่อยู่ในระดับเดียวกัน ให้ตามลำดับความสำคัญจากสูงไปต่ำดังแสดงในตาราง

IE0	External Interrupt 0
TF0	Timer 1
IE1	External Interrupt 1
TF1	Timer 1
RI หรือ TI	Serial Port
TF2 หรือ EXF2	Timer 2

ตารางที่ 2.2 ลำดับความสำคัญของการอินเทอร์รัพต์

การกำหนดลำดับความสำคัญของการอินเทอร์รัพต์กำหนดได้จากบิตต่าง ๆ ของรีจิสเตอร์ IP โดยบิตที่มีค่าเป็น 1 จะมีลำดับความสำคัญสูงกว่า

IP : INTERRUPT RRIORITY REGISTER

IP.7	IP.6	IP.5	IP.4	IP.3	IP.2	IP.1	IP.0
-	-	PT2	PS	PT1	PX1	PT0	PX0

- IP.7 ไม่ใช้

- IP.6 ไม่ใช้

PI2 IP.5 กำหนดความสำคัญของ Timer 2

PS IP.4 กำหนดความสำคัญของพอร์ตอนุกรม

PT1 IP.3 กำหนดความสำคัญของ Timer 1

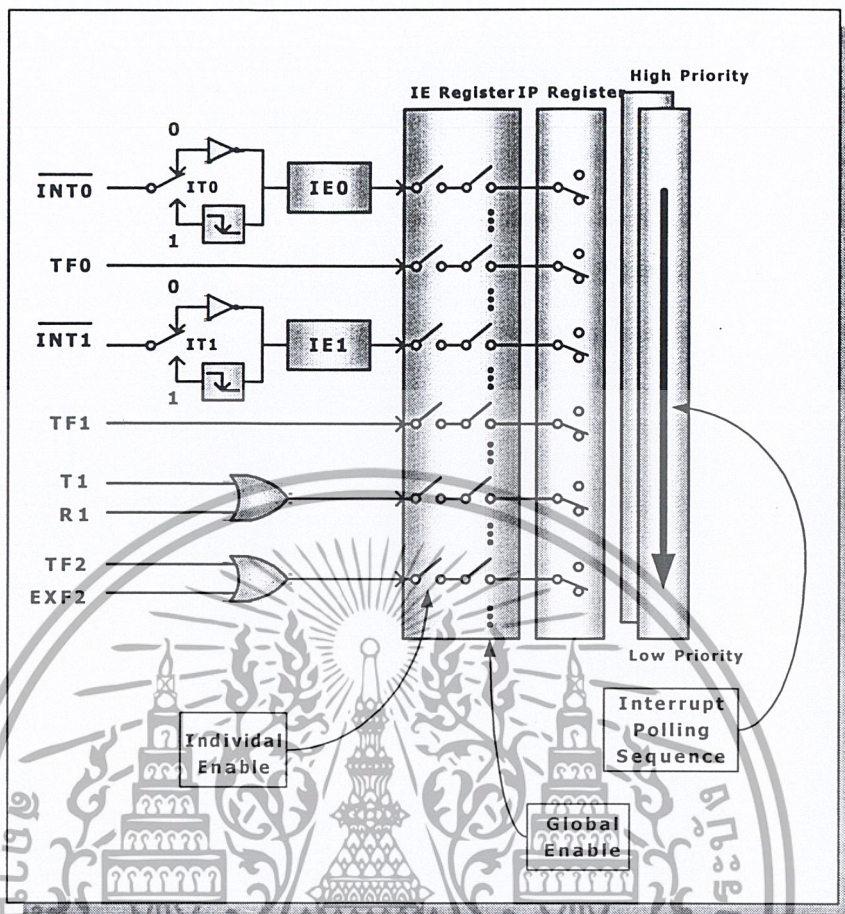
PX1 IP.2 กำหนดความสำคัญของสัญญาณ INT1

PT0 IP.1 กำหนดความสำคัญของ Timer 0

PX0 IP.0 กำหนดความสำคัญของสัญญาณ INTO

เมื่อมีการกำหนดลำดับความสำคัญของการอินเทอร์รัพต์แตกต่างกันแล้ว หากมีการร้องขออินเทอร์รัพต์เกิดขึ้นพร้อมกันจากแหล่งกำเนิด 2 แหล่งกำเนิดที่มีระดับความสำคัญแตกต่างกัน ไมโครคอนโทรลเลอร์จะตอบรับการร้องขออินเทอร์รัพต์ที่มีระดับความสำคัญสูงกว่าสามารถอินเทอร์รัพต์ซ้อนในขณะที่ไมโครคอนโทรลเลอร์กำลังทำโปรแกรมตอบสนองการอินเทอร์รัพต์ของสัญญาณที่มีความสำคัญต่ำกว่าได้ ในขณะที่ไมโครคอนโทรลเลอร์กำลังทำโปรแกรมตอบสนองการอินเทอร์รัพต์อยู่ จะไม่สามารถตอบรับการร้องขออินเทอร์รัพต์จากแหล่งกำเนิดระดับที่ต่ำกว่าหรือระดับเดียวกันได้อีก จนกว่าจะจบโปรแกรมตอบสนองการอินเทอร์รัพต์ก่อน

วงจรการควบคุมการอินเทอร์รัพต์ของ MCS-51 ทำโดยการควบคุมการตัดต่อสวิตช์ต่าง ๆ ดังแสดงในรูป 6-2 ซึ่งควบคุมการตัดต่อสวิตช์ด้วยซอฟต์แวร์จากการเซตหรือเคลือบิตของรีจิสเตอร์ IE และ IP



รูปที่ 2.13 ระบบควบคุมการอินเทอร์รัพต์

หากต้องการให้ไมโครคอนโทรลเลอร์ตอบรับการร้องขออินเทอร์รัพต์ จะต้องมีการกำหนดการทำงานตามขั้นตอนต่อไปนี้

1. เซตบิต EA (Enable all) ในรีจิสเตอร์ IE.7 ให้เป็น 1
2. เซตบิตควบคุมอินเทอร์รัพต์ของแต่ละแหล่งกำเนิดที่ต้องการใช้งานอินเทอร์รัพต์ในรีจิสเตอร์ IE ให้มีค่าเป็น 1
3. กำหนดลำดับความสำคัญของการอินเทอร์รัพต์ (หากต้องการ) โดยกำหนดในรีจิสเตอร์ IP
4. เขียนโปรแกรมตอบสนองการร้องขออินเทอร์รัพต์ของแต่ละแหล่งกำเนิดลงในตำแหน่งต่าง ๆ ตาม Vector Address ในตาราง

ตารางแสดงตำแหน่งต่าง ๆ ของโปรแกรมตอบสนองอินเทอร์รัพต์ของแต่ละแหล่งกำเนิด ซึ่งจะต้องนำคำสั่งเริ่มต้นของโปรแกรมตอบสนองอินเทอร์รัพต์ไปบรรจุไว้ หากมีการใช้อินเทอร์รัพต์ทั้งหมดจะพบว่าความหมายของโปรแกรมตอบสนองอินเทอร์รัพต์ของแต่ละโปรแกรมจะมีความ

ขาวเพียง 8 ไบต์ เท่านั้น หากโปรแกรมมีความยาวมากกว่านี้ ต้องนำไปเขียนเป็นโปรแกรมย่อยไว้ภายนอก แล้วใช้วิธีการเรียกโปรแกรมย่อยมาทำงาน

Interrupt Souece	Vector Address
IE0	0003H
TF0	000BH
IE1	0013H
TE1	001BH
RI & T1	0023H
TF2 & EXF2	002BH

ตารางที่ 2.3 Vector Address ของการตอบสนองอินเทอร์รัพต์

สัญญาณอินเทอร์รัพต์จากภายนอก

แหล่งกำเนิดสัญญาณอินเทอร์รัพต์จากภายนอกมี 2 สัญญาณคือ $\overline{INT0}$ และ $\overline{INT1}$ ซึ่งอยู่ที่ขา P3.2 และ P3.3 เราสามารถเลือกลักษณะสัญญาณร้องขออินเทอร์รัพต์ได้สองลักษณะคือ เลือกใช้สัญญาณ แอคทีฟที่ระดับลอจิก 0 หรือเลือกใช้สัญญาณแอคทีฟที่การเปลี่ยนแปลงของสัญญาณจาก 1 ไป 0 ซึ่งการเลือกทำได้โดยการกำหนดค่าบิตของ IT0 และ IT1 ที่อยู่ในรีจิสเตอร์ TCON หากรีเซตให้ค่าของบิต IT0 เป็น 0 จะเป็นการกำหนดการเลือกใช้สัญญาณแอคทีฟที่ระดับลอจิก 0 หากเซตให้ IT0 = 1 เป็นการเลือกใช้สัญญาณแอคทีฟที่การเปลี่ยนระดับสัญญาณจาก 1 ไป 0 การตรวจสอบการร้องขอการอินเทอร์รัพต์ของไมโครคอนโทรลเลอร์จะกระทำในทุก ๆ เมกซ์ซีนไซเกิลภายในตัวไมโครคอนโทรลเลอร์มีแฟลกการร้องขออินเทอร์รัพต์จากสัญญาณภายนอกทั้งสองอยู่ที่บิต IE0 และ IE1 ในรีจิสเตอร์ TCON โดยแฟลกการร้องขออินเทอร์รัพต์จะถูกเซตเมื่อมีสัญญาณอินเทอร์รัพต์จากภายนอกเข้ามาตามเงื่อนไขที่กำหนด

ในกรณีที่เรากำหนดให้สัญญาณอินเทอร์แอคทีฟเมื่อมีการเปลี่ยนค่าจาก 1 ไป 0 แฟลกการร้องขออินเทอร์รัพต์จะถูกเซตเมื่อสัญญาณที่ขา $\overline{INT0}$ หรือ $\overline{INT1}$ เปลี่ยนจาก 1 ไป 0 (ค่าของลอจิก 1 หรือ ลอจิก 0 จะต้องนานกว่า 12 คาบเวลาของสัญญาณนาฬิกาเพื่อให้ไมโครคอนโทรลเลอร์สามารถตรวจสอบได้ทัน ค่าของแฟลกการร้องขออินเทอร์รัพต์ IE0 หรือ IE1 จะถูกเคลียจากฮาร์ดแวร์อัตโนมัติเมื่อไมโครคอนโทรลเลอร์ตอบรับการร้องขออินเทอร์รัพต์ โดยการเรียกโปรแกรมย่อยของการตอบสนองการอินเทอร์รัพต์มาทำงาน

หากเรากำหนดให้สัญญาณอินเทอร์รัพต์แอกทีฟที่ลอจิก 0 สัญญาณการร้องขออินเทอร์รัพต์ที่ส่งเข้ามา จะต้องค้างเป็น 0 ไว้จนกว่าไมโครคอนโทรลเลอร์จะทำการตอบรับการร้องขออินเทอร์รัพต์ การตอบรับการร้องขออินเทอร์รัพต์ของไมโครคอนโทรลเลอร์ ทำโดยการเรียกโปรแกรมย่อยการตอบสนองอินเทอร์รัพต์มาทำงาน เมื่อไมโครคอนโทรลเลอร์เรียกโปรแกรมการตอบสนองอินเทอร์รัพต์มาทำงานแล้ว ต้องเลิกส่งสัญญาณการร้องขออินเทอร์รัพต์ก่อนที่โปรแกรมการตอบสนองจะจบลง หรือก่อนที่การร้องขออินเทอร์รัพต์จากแหล่งกำเนิดอื่นจะเข้ามา เพื่อป้องกันไม่ให้เกิดการอินเทอร์รัพต์ซ้ำ

การใช้งานสัญญาณร้องขออินเทอร์รัพต์จากภายนอก INTO และ INT1 (ขา P3.2 และ P3.3) จะต้องเซตบิตละให้เป็น 1 ในตอนเริ่มต้นก่อนเพื่อทำให้เป็นอิมพีแดนซ์สูงโดยใช้คำสั่ง SETB P3.2 และ SEB P3.3

การใช้งานสัญญาณ INTO

เนื่องจากตำแหน่งโปรแกรมการตอบสนองการอินเทอร์รัพต์ของสัญญาณ เริ่มต้นที่ตำแหน่ง 0003 และโปรแกรมหลักหลังการรีเซตไมโครคอนโทรลเลอร์เริ่มต้นทำงานที่ตำแหน่ง 0000 ดังนั้นคำสั่งแรกของโปรแกรมหลักที่อยู่ในตำแหน่ง 0000 ต้องเป็นคำสั่งการกระโดดข้ามส่วนของโปรแกรมตอบสนองอินเทอร์รัพต์ไป เพื่อไม่ให้มีการทำโปรแกรมตอบสนองอินเทอร์รัพต์ก่อนที่จะมีการอินเทอร์รัพต์ก่อนที่จะมีการอินเทอร์รัพต์ในส่วนของโปรแกรมหลักจะต้องมีการกำหนดให้ไมโครคอนโทรลเลอร์ตอบรับการร้องขออินเทอร์รัพต์จากสัญญาณ INTO โดยกำหนดให้บิตที่ 7 (EA) ของรีจิสเตอร์ IE = 1 และต้องเซตให้บิตที่พอร์ต P3.2 = 1 ด้วย

ในส่วนของโปรแกรมตอบสนองอินเทอร์รัพต์ทั้งหมด จะต้องมีการเก็บค่าของรีจิสเตอร์ที่ใช้งานในโปรแกรมตอบสนองอินเทอร์รัพต์ทั้งหมดลงสแตคและดึงกลับออกมาก่อนจบการทำงาน ของโปรแกรม เพื่อไม่ให้ข้อมูลของโปรแกรมหลักเปลี่ยนแปลงไป เนื่องจากสัญญาณอินเทอร์รัพต์จะเกิดขึ้นในเวลาใดก็ได้ ดังนั้นค่าของรีจิสเตอร์ต่างๆ ที่โปรแกรมตอบสนองการอินเทอร์รัพต์ใช้ จะเปลี่ยนค่าไปจากเดิมเมื่อจบโปรแกรมตอบสนองอินเทอร์รัพต์

สัญญาณอินเทอร์รัพต์จาก Timer0 และ Timer1

แหล่งกำเนิดสัญญาณอินเทอร์รัพต์ TFO และ TF1 เป็นแหล่งกำเนิดที่มาจากตัวตั้งเวลา/ตัวนับ Timer0 และ Timer1 ตามลำดับ TFO และ TF1 จะเกิดขึ้นเมื่อค่าเวลาของ Timer0 และ Timer1 ที่ตั้งไว้ครบกำหนดแล้ว หรือนับได้ครบจำนวนที่ตั้งไว้ในกรณีที่ทำหน้าที่เป็นตัวนับ โดย TFO และ TF1 จะถูกเซตโดยฮาร์ดแวร์ภายในตัวไมโครคอนโทรลเลอร์ และค่าของแฟล็ก TFO และ TF1 จะถูกเคลียร์อัตโนมัติโดยฮาร์ดแวร์เมื่อไมโครคอนโทรลเลอร์ตอบรับการร้องขอ

อินเทอร์รัพต์ โดยการเรียกโปรแกรมย่อยที่อยู่ในตำแหน่งเริ่มต้นที่ OBH (สำหรับ Timer0) หรือ 1BH (สำหรับ Timer1) มาทำงาน

สัญญาณอินเทอร์รัพต์จากพอร์ตอนุกรม

แหล่งกำเนิดสัญญาณอินเทอร์รัพต์จากพอร์ตอนุกรมมีสัญญาณอินเทอร์รัพต์เกิดจาก 2 แหล่งคือ จาก RI (ด้านรับ) และ TI (ด้านส่ง) เป็นแหล่งกำเนิดที่มาจากบัฟเฟอร์ทางด้านรับและส่งออกข้อมูลของพอร์ตอนุกรมที่อยู่ในไมโครคอนโทรลเลอร์ สัญญาณทั้งสองจะถูกนำมา OR กันเพื่อรวมเป็นสัญญาณเดียว ทำให้มีสัญญาณร้องขออินเทอร์รัพต์ของพอร์ตอนุกรมอยู่ที่ตำแหน่ง 23H ดังนั้นในส่วนของโปรแกรมตอบสนองการอินเทอร์รัพต์ของพอร์ตอนุกรม จะต้องมีการตรวจสอบก่อนว่าการขออินเทอร์รัพต์เกิดขึ้นจาก RI หรือ TI โดยใช้คำสั่งการตรวจสอบบิต RI หรือ TI ที่อยู่ในรีจิสเตอร์ SCON เมื่อทราบที่มาของการร้องขออินเทอร์รัพต์แล้วจึงไปทำโปรแกรมสำหรับการรับหรือการส่งข้อมูลต่อไป หลังจากตรวจสอบแหล่งกำเนิดสัญญาณร้องขออินเทอร์รัพต์แล้วจะต้องทำการเคลียบิต RI หรือ TI ด้วยซอฟต์แวร์ก่อนที่จะจบโปรแกรมตอบสนองการอินเทอร์รัพต์ ตัวอย่างการใช้งานจะอยู่ในเรื่องการรับส่งข้อมูลของพอร์ตอนุกรม

2.3.3 การใช้งานพอร์ตอนุกรม

การสื่อสารอนุกรมเป็นการส่งข้อมูลครั้งละบิตแบบต่อเนื่องกันไปโดยส่งบิตต่ำออกไปก่อนแล้วตามด้วยบิตสูงซึ่งแตกต่างไปจากการส่งข้อมูลแบบขนานที่ส่งข้อมูลทุกบิตออกไปพร้อมกัน

การส่งข้อมูลแบบอนุกรมมีโครงสร้างที่สำคัญหลายอย่างเช่นความเร็วในการส่งข้อมูลซึ่งเรียกว่า Baud rate มีหน่วยเป็น บิตต่อวินาที

ภายในไมโครคอนโทรลเลอร์ MCH-51 มีพอร์ตการสื่อสารแบบอนุกรม (Serial port) ที่สามารถรับและส่งข้อมูลแบบ Full Duplex อยู่ 1 พอร์ต การส่งข้อมูลแบบ Full Duplex หมายถึงว่าพอร์ตอนุกรมสามารถรับและส่งข้อมูลได้ในเวลาเดียวกัน การควบคุมการทำงานของพอร์ตอนุกรมทางด้านส่งและรับข้อมูลทำโดยการกำหนดค่าให้รีจิสเตอร์ควบคุมการทำงานของพอร์ตอนุกรมชื่อ SCON

รีจิสเตอร์ควบคุมการทำงานของพอร์ตอนุกรม SCON (Serial Port Control Register) SCON เป็นรีจิสเตอร์เฉพาะที่ทำหน้าที่ควบคุมโหมดการทำงานของพอร์ตอนุกรม และเป็นที่ยึดข้อมูลบิตที่ 9 ของการรับและส่งข้อมูล (บิต TB8 และ RB8) และมีแฟลกของการร้องขออินเทอร์รัพต์ของพอร์ตอนุกรมรวมอยู่ด้วย บิตต่าง ๆ ดังแสดงในตาราง การควบคุมการทำงานเราจะ

กำหนดบิตต่าง ๆ ในรีจิสเตอร์ตัวนี้ด้วยคำสั่งการเซ็ทหรือเคลียร์บิตก็ได้เนื่องจากรีจิสเตอร์ SCON
อ้างตำแหน่งแบบบิตได้

Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
SM0	SM1	SM2	REN	TB8	Rb6	T1	RI

SCON : Serial Port Control Register

ความหมายของบิตต่าง ๆ ในรีจิสเตอร์ SCON เป็นดังนี้

SM0 และ SM1 : เป็นบิตกำหนดโหมดการทำงานของพอร์ตอนุกรมซึ่งมี 4 โหมด ดังนี้

SM0	SM1	โหมด	การทำงาน	อัตรารับส่ง
0	0	0	Shift register	$F_{osc}/12$
0	1	1	8 bit UART	Variable
1	0	2	9 bit UART	$F_{osc}/32$ หรือ $f_{osc}/64$
1	1	30	9 bit UART	Variable

การเลือกโหมดการทำงานของพอร์ตอนุกรม

SM2 : เป็นบิตควบคุมให้ทำงานในลักษณะการเชื่อมต่อไมโครคอนโทรลเลอร์หลายตัวเข้าด้วยกัน
สำหรับการใช้งานในโหมด 2 หรือโหมด 3 เป็นดังนี้

SM2 = 1 จะทำให้แฟลคอินเตอร์รัพต์ทางด้านรับ (R1) ไม่ถูกเซตเมื่อรับข้อมูลเข้ามาแล้ว
มีค่าบิตที่ 9 เป็น 0 (อยู่ในบิต RB8) สำหรับการทำงานในโหมด 1 ถ้าเซต SM2 = 1 แฟลค
อินเตอร์รัพต์ทางด้านรับ (แฟลค RI) จะไม่ถูกเซตหากข้อมูลที่รับเข้ามาไม่มี stop bit การใช้งาน
ในโหมด 0 ต้องกำหนดให้ SM2 = 0

REN : เซตหรือรีเซตด้วยซอฟต์แวร์เป็นตัวควบคุมการรับข้อมูลของพอร์ตอนุกรมดังนี้

- 1 = ให้มีการรับข้อมูล
- 2 = ไม่ให้มีการรับข้อมูล

TB8 : เป็นบิตข้อมูลบิตที่ 9 ที่ต้องการส่งในโหมด 2 และ 3 สามารถเซตหรือเคลียร์

RB8 : เป็นบิตเก็บข้อมูลที่รับเข้ามาบิตที่ 9 ในโหมด 2 หรือ 3 สำหรับการทำงานใน
โหมด 1 หากกำหนดให้บิต SM2 = 0 บิต RB8 จะเป็นค่าของ stop bit ที่รับเข้ามาสำหรับ
โหมด 0 ไม่มีการใช้ RB8

T1 : แฟลคของการอินเตอร์รัพต์ด้านส่งข้อมูล แฟลคนี้จะถูกเซตด้วยฮาร์ดแวร์เมื่อจบการ
ส่งข้อมูลบิตที่ 8 ใน โหมด 0 หรือเมื่อเริ่มต้นส่ง stop bit ในโหมด 1, 2 หรือ 3 เราต้อง
เคลียร์แฟลคนี้ด้วยซอฟต์แวร์ เมื่อจบโปรแกรมตอบสนองการอินเตอร์รัพต์ของการส่งข้อมูลแล้ว

RI : แพลทของอินเทอร์เฟซด้านรับข้อมูล แพลทนี้จะถูกเซตด้วยฮาร์ดแวร์เมื่อจบการส่งข้อมูลบิตที่ 8 ใน โหมด 0 ถูกรับเข้ามาในครั้งแรกในโหมด 1, 2 และ 3 เราต้องเคลียร์แพลทนี้ด้วยซอฟต์แวร์ เมื่อจบโปรแกรมตอบสนองการอินเทอร์เฟซของการส่งข้อมูลแล้ว

Mode	รีจิสเตอร์ SCON	บิต SM2 Variation
0	10H	Single processor
1	50H	Environment
2	90H	(SM = 2)
3	D0H	
0	NA	
1	70H	Multiprocessor
2	B0H	Environment
3	F0H	(SM = 1)

ตารางที่ 2.4 การเลือกโหมดการทำงานของพอร์ตอนุกรม

การรับส่งข้อมูลของพอร์ตอนุกรมจะมีรีจิสเตอร์ที่ทำหน้าที่รับและส่งข้อมูลอยู่ 1 ตัว คือรีจิสเตอร์ SBUF การส่งข้อมูลออกไปยังพอร์ตอนุกรมของไมโครคอนโทรลเลอร์ทำได้โดยการใช้ข้อมูลลงในรีจิสเตอร์ SBUF เช่นกัน

การทำงานของพอร์ตอนุกรมของไมโครคอนโทรลเลอร์สามารถแบ่งออกได้เป็น 4 โหมด คือ โหมด 0, 1, 2 และ 3 แต่ละโหมดมีการทำงานดังนี้

โหมด 0

ข้อมูลขนาด 8 บิต แบบอนุกรมรับเข้ามาทางขา RXD และข้อมูล 8 บิตส่งออกแบบอนุกรมทางขา TXD การรับและส่งข้อมูลจะเริ่มจากบิตต่ำ (LSB) ก่อน อัตราการรับส่งข้อมูล (Baud rate) จะเป็น 1/12 เท่าของสัญญาณนาฬิกา

โหมด 1

ใช้การรับและส่งข้อมูลแบบ 10 บิต ซึ่งข้อมูลอนุกรม 10 บิต เข้ามาทางขา RTD แล้วข้อมูลขนาด 10 บิตส่งออกแบบอนุกรมทางขา TDX โดยข้อมูล 10 ประกอบด้วย 1 Start bit (ค่า 0), 8 บิตข้อมูล (การรับ/ส่งจะเริ่มจากบิตต่ำก่อน) และ 1 Stop bit ที่รับเข้ามาไปเก็บใน

บิต ที่อยู่ในรีจิสเตอร์ SCON อัตราการรับส่งข้อมูลในโหมดนี้สามารถกำหนดได้ตามต้องการ โหมด 2

ใช้การรับข้อมูลขนาด 11 บิต ข้อมูลแบบอนุกรมรับเข้ามาทางขา RXD และส่งออกไปทางขา TXD ซึ่งข้อมูล 11 บิตประกอบด้วย 1 Start bit (ค่า 0), 8 บิต (การรับ/ส่งเริ่มจากบิตก่อน), 1 บิต ข้อมูลที่กำหนดค่าได้ และ 1 Stop bit (ค่า 1) สำหรับด้านส่งข้อมูลบิตที่ 9 จะกำหนดไว้ใน TB8 ที่อยู่ในรีจิสเตอร์ SCON ซึ่งเราสามารถกำหนดเป็น 1 หรือ 0 ก็ได้ ในการทำงานเราอาจใช้บิตข้อมูลบิตที่ 9 เป็นบิตตรวจสอบการส่งข้อมูลแบบพาริตีคู่ ในกรณีของการรับข้อมูลบิตที่ 9 ที่รับเข้ามาไปเก็บในบิต RB8 ที่อยู่ในรีจิสเตอร์ SCON ส่วน Stop bit จะไม่มีการนำมาเก็บอัตราการรับและส่งข้อมูลในโหมดนี้จะเลือกใช้ความเร็วได้ 2 ค่าคือ 1/30 หรือ 1/64 เท่าของสัญญาณนาฬิกาข้อมูล

โหมด 3

การทำงานในโหมด 3 จะมีลักษณะคล้ายกับโหมด 2 แต่ในโหมด 3 เราสามารถกำหนดอัตราการรับและส่งข้อมูลได้ตามต้องการ

การทำงานของพอร์ตอนุกรมทั้ง 4 โหมด การส่งข้อมูลและเริ่มต้นเมื่อมีการกำหนดข้อมูลให้กับรีจิสเตอร์ ซึ่งการกำหนดค่าให้กับรีจิสเตอร์ SUBF จะใช้คำสั่งการโอนย้ายข้อมูลเช่น MOVE SBUF, #15H หรือ MOVE SUBF, @R1 ก็ได้ การรับข้อมูลในโหมด 0 จะเริ่มต้นรับข้อมูลเมื่อค่าของบิต RI = 0 และ REN = 1 ส่วนในโหมดอื่น ๆ การรับข้อมูลจะเริ่มต้นเมื่อกำหนดบิต REN = 1 และมี Start bit เข้ามาที่ขา RXD การกำหนดอัตรารับส่ง

อัตรารับส่งข้อมูลแบบอนุกรมเป็นความเร็วในการส่งข้อมูลในหน่วย บิต/วินาที ซึ่งเราสามารถกำหนดอัตรารับส่งที่แตกต่างกันได้ในโหมด 1 และ 3 สำหรับอัตรารับส่งในโหมด 0 จะคงที่ส่วนในโหมด 2 เลือกได้ 2 อัตราความเร็ว

อัตรารับส่งข้อมูลในโหมด 0 จะใช้สัญญาณนาฬิกาของระบบไมโครคอนโทรลเลอร์เป็นฐาน คือ

$$\text{อัตรารับส่งโหมด 0} = \text{ความถี่สัญญาณนาฬิกาของไมโครคอนโทรลเลอร์} / 12$$

อัตราการรับส่งในโหมด 2 เลือกได้ 2 อัตรา ขึ้นอยู่กับบิต SMOD ที่อยู่ในรีจิสเตอร์ PCON เป็นตัวเลือก

$$\text{ถ้าบิต SMOD} = 0$$

อัตรารับส่งโหมด 2 = 1/64 เท่าของสัญญาณนาฬิกา

ถ้า SMOD = 1

อัตรารับส่งโหมด 2 = 1/32 เท่าของสัญญาณนาฬิกา

สำหรับอัตราการรับและส่งข้อมูลในโหมด 1 และโหมด 3 จะกำหนดได้ตามต้องการ โดยใช้อัตราการเกิดโอเวอร์โฟลวของ หรือเป็นตัวกำหนด (8052 มี Timer เพิ่มขึ้น) อัตราการรับและส่งข้อมูลสามารถกำหนดได้จาก Timer1 หรือ Timer0 หรือจากทั้ง 2 ตัว (ตัวหนึ่งสำหรับส่งอีกตัวหนึ่งสำหรับข้อมูล) ความเร็วในการรับและส่งข้อมูลอาจกำหนดไม่เท่ากันได้

การใช้ Timer1 เป็นตัวสร้างอัตราการรับส่ง

เมื่อใช้ Timer1 เป็นตัวสร้างอัตราการรับส่งข้อมูลของพอร์ตอนุกรมในโหมด 1 หรือโหมด 3 ต้องควบคุมไม่ให้ Timer1 ส่งสัญญาณอินเตอร์รัพต์ อัตราการรับและส่งข้อมูลมาจากอัตราการเกิดโอเวอร์โฟลวของ Timer1 และค่าของบิต SMOD ที่อยู่ในรีจิสเตอร์ PCON (รีจิสเตอร์ PCON ไม่สามารถอ้างเซตบิต 7 ของ PCON สำหรับการเคลียให้ใช้การ AND เช่น ANL PCON, #01111111B เท่ากับการเคลียบิต 7 ของ) เราคำนวณอัตรารับและส่งข้อมูลได้ดังนี้

$$\text{อัตรารับส่งในโหมด 1 และโหมด 3} = \frac{2^{\text{smod}}}{32} * (\text{Timer1 Overflow rate})$$

เมื่อใช้ Timer1 สร้างอัตราการรับและส่งข้อมูลจะต้องทำให้ไม่มีการร้องขออินเตอร์รัพต์ที่เกิดจาก Timer1 (โดยการควบคุมที่รีจิสเตอร์ IE) เราจะใช้ Timer1 ทำงานในลักษณะของตัวจับเวลาหรือตัวนับอย่างใดอย่างหนึ่งได้ในโหมดต่างๆ ทั้ง 3 โหมด สำหรับการใช้งานทั่วไปเราจะกำหนดให้ Timer1 ทำงานในลักษณะของตัวจับเวลาในโหมด (มีค่าการควบคุมใน) ซึ่งจะได้อัตราการรับและส่งข้อมูลดังนี้

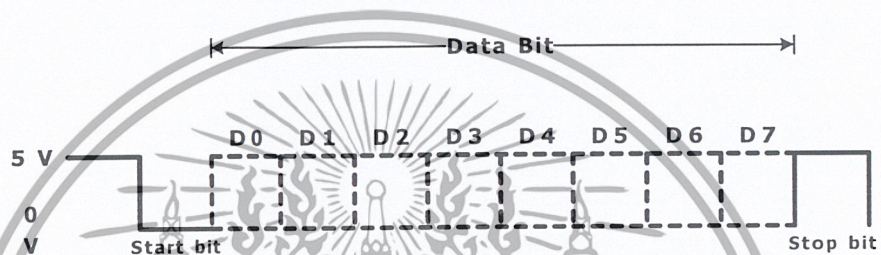
$$\text{อัตรารับส่งในโหมด 1 และโหมด 3} = \frac{2^{\text{smod}}}{32} * \frac{\text{Oscillator}}{12 * [256 - (\text{TH1})]}$$

หากเราต้องการให้ Timer1 มีอัตราการรับส่งต่ำมาก ๆ เราทำให้ Timer1 ทำงานในลักษณะของตัวจับเวลาแบบ 16 บิต (มีค่าของการควบคุมใน TMOD = 0010XXXX) และให้มี

การอินเทอร์รัพต์จาก Timer1 โดยให้โปรแกรมตอบสนองอินเทอร์รัพต์ของ Timer1 ทำการกำหนดค่าเริ่มต้นใหม่ ให้กับตัวจับเวลา ซึ่งเป็นการทำงานที่เป็นแบบ 16 บิต

การรับส่งข้อมูลในโหมด 1

ในโครงการใช้การรับส่งข้อมูลในโหมด 1 ซึ่งยกตัวอย่างเฉพาะการส่งข้อมูลจำนวน 10 บิต จะส่งออกไปที่ขา TXD และรับข้อมูลเข้ามาทางขา RDX ข้อมูล 10 บิต ประกอบด้วย 1 Start Bit (ค่าลอจิก 0) ข้อมูลจำนวน 8 บิต (LSB ส่งหรือรับก่อน) และ 1 Stop Bit (ค่าลอจิก 1)



รูปที่ 2.14 สัญญาณของการส่งข้อมูลของโหมด 1

ในกรณีของการรับข้อมูล Stop Bit จะถูกนำเข้าไปเก็บในบิต RBB ที่อยู่ในรีจิสเตอร์ SCON ความเร็วในการรับส่งข้อมูลจะกำหนดโดยอัตราการเกิดโอเวอร์โฟลวของ Timer1 หรือ Timer2 หรือกำหนดจากทั้ง 2 ตัว (ตัวหนึ่งสำหรับด้านส่ง ตัวหนึ่งสำหรับด้านรับ) รูป 8-11 แสดงไดอะแกรม เวลารับส่งข้อมูลของพอร์ตอนุกรมในโหมด 1

การส่งข้อมูลจะเริ่มขึ้นหลังจากกำหนดค่าลงในรีจิสเตอร์ข้อมูล SBUF ด้านส่งจะมีสัญญาณ Write to SBUF ควบคุมให้มีการใส่ค่า 1 ลงในตำแหน่งของบิตที่ 9 ของ SHIFT REGISTER ด้านส่ง และเริ่มทำงานส่งข้อมูลของ TX CONTROL UNIT การส่งข้อมูลออกจะเริ่มขึ้นในช่วงเวลา S1P1 ของเมซซิงไซเคิลและบิตต่อไปตามจังหวะของตัวกำหนดอัตราส่งที่มาจากโอเวอร์โฟลว์ของ Timer1 หรือ Timer2 ที่หารด้วย 16

การส่งข้อมูลจะเริ่มด้วยการส่งสัญญาณ SEND ให้แอกทีฟ เมื่อเริ่มต้นการส่งข้อมูลบิตแรกออกไปที่ TXD และบิตของข้อมูลจะถูกส่งตามออกไปโดยการทำงานของ Shift register ซึ่ง จะทำการเลื่อนข้อมูลไป 1 ตำแหน่งตามจังหวะของอัตราส่งที่เข้ามายัง TX CLOCK เมื่อบิตข้อมูลทั้งหมดถูกเลื่อนออกไปทางขวา จะมี 01 เข้ามาแทนที่ทางด้านซ้าย จนกระทั่งบิตข้อมูลสูงสุด (MSB) ออกไปอยู่ที่เอาต์พุตของ Shift Register ทางซ้ายของบิตที่ 9 เป็น 0 ทั้งหมด เมื่อเกิด

เงื่อนไขดังกล่าวจะทำให้ TX CONTROL UNIT ทำการเลื่อนข้อมูลจะจบในช่วง CLOCK ถูกที่ 10 ของสัญญาณการส่งข้อมูล หลังจากมีสัญญาณ write to SBUF

การรับข้อมูลจะเริ่มขึ้นเมื่อสัญญาณที่ขา RXD เปลี่ยนจาก 1 ไป 0 โดยซีพียู จะมีการตรวจสอบด้วยความเร็ว 16 เท่า ของอัตราความเร็วในการรับข้อมูลเมื่อซีพียูตรวจสอบได้ว่าการส่งข้อมูลเข้ามาจะทำการรีเซตวงจรรอรับ 16 ทันทเพื่อเริ่มต้นใหม่ให้สัมพันธ์กับข้อมูลที่เข้ามาและค่า 1FFH จะถูกกำหนดให้กับ Shift Register ด้านรับ

การรับข้อมูลจะทำการตรวจสอบบิตข้อมูลที่ขา TXD 3 ครั้งในช่วงบิต RCLK ที่ 7, 8 และ 9 เข้ามาในวงจรรอรับ 16 โดยจะใช้ค่าที่อ่านจาก 2 ใน 3 ครั้งเป็นข้อมูล การทำเช่นนี้ก็เพื่อจัดสัญญาณรบกวนออกหากการตรวจสอบว่าในช่วงที่บิตที่ 0 ของ RCLK ข้อมูลที่ RXD ไม่เป็น 0 วงจรด้านรับจะรีเซตกลับไปรอการเปลี่ยนค่าที่ขา RXD จาก 1 ไป 0 ใหม่ การทำเช่นนี้เพื่อป้องกันการผิดพลาดของ START BIT หากค่า Start Bit ถูกต้อง จะทำการเลื่อนข้อมูลเข้าไปใน Shift Register และรอรับบิตต่อไป โดยบิตข้อมูลจะเข้ามาทางขวาของ Shift Register และค่า 1 ที่กำหนดไปตอนเริ่มจะเริ่มจะถูกเลื่อนไปทางซ้ายตามลำดับเมื่อ Start Bit ที่เข้ามาถูกเลื่อนไปทางซ้ายมือสุดของ Shift Register จะทำให้ RX CONTROL BOX ทำให้การเลื่อนอีก 1 เป็นครั้งสุดท้าย และนำค่าใน Shift Register ไปเก็บไว้ในรีจิสเตอร์ SBUF และ RB8 และการเซตแฟล็ก RI จะเกิดขึ้นหลังจากการเลื่อนข้อมูลครั้งสุดท้ายแล้วต่อเมื่อเราได้กำหนด

- 1) RI = 0 (จะต้องเคลียร์ RI หลังจากทำโปรแกรมอ่านข้อมูลไปแล้ว) และ
- 2) SM2 = 0 หรือได้รับ STOP BIT = 1

หากเงื่อนไขทั้งสองไม่จริงข้อมูลที่รับเข้ามาจะถูกยกเลิก ถ้าเงื่อนไขทั้ง 2 เป็นจริง STOP BIT จะถูกนำไปเก็บ ใน RB8 และข้อมูล 8 บิต ถูกนำไปเก็บใน SBUF และ RI ถูกเซต วงจรรับข้อมูลจะกลับไปตรวจสอบที่ขา RXD ต่อเมื่อรอรับข้อมูลบิตใหม่ต่อไป

2.4 Visual Basic 6

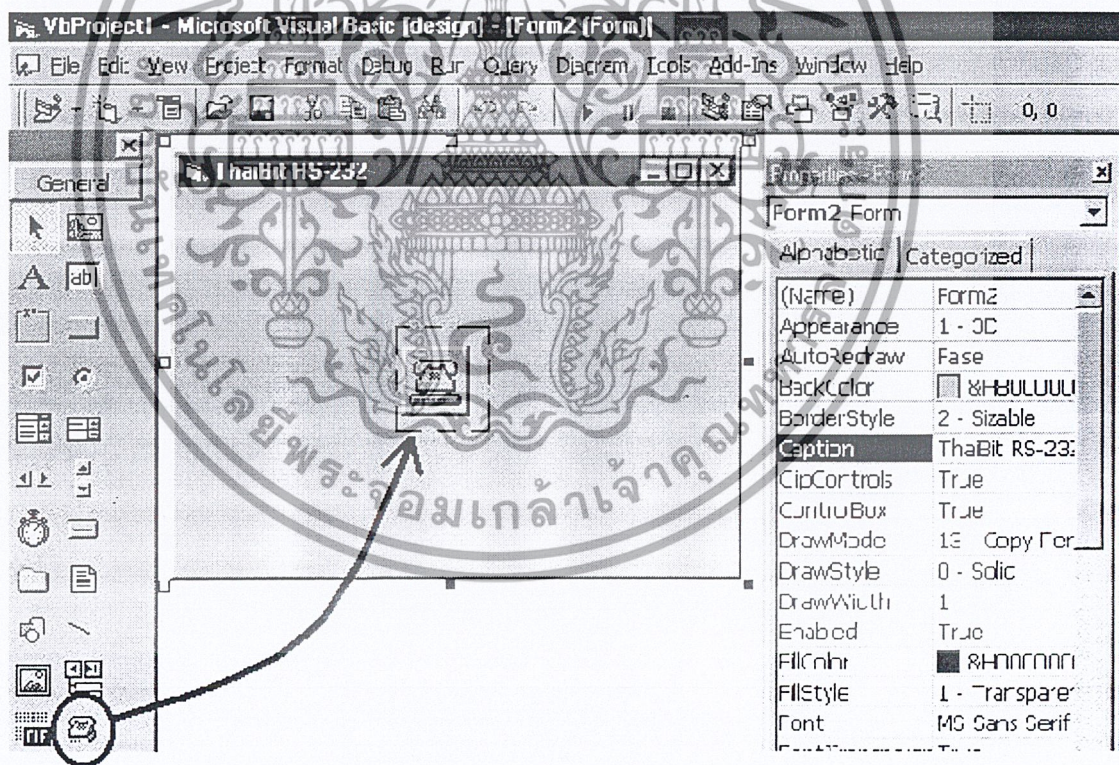
เป็นโปรแกรมที่ใช้ในการส่งข้อมูลออกพอร์ตอนุกรมของคอมพิวเตอร์ ซึ่งในโครงการนี้จะใช้คอมพิวเตอร์สั่งเปิดปิดอุปกรณ์ไฟฟ้าในตำแหน่งที่ต้องการ ตำแหน่งที่ต้องการควบคุมจะถูกส่งออกไปยังพอร์ตอนุกรมแล้วส่งต่อไปยัง RS-232 เพื่อนำสัญญาณนี้ใช้ในการส่งไปบนสายไฟกระแสสลับเพื่อให้ควบคุมอุปกรณ์ไฟฟ้าที่ต้องการได้

Visual Basic เป็นภาษาที่ได้รับความนิยมนำมาใช้ในการพัฒนาโปรแกรมบน Windows เนื่องจากเป็นภาษาคอมพิวเตอร์ที่ใช้เทคโนโลยีในลักษณะ Visualize ซึ่งเพียงแค่เลือก Control ที่

เหมาะสมแล้วว่าดลงบน Form ก็สามารถสร้างจอภาพที่ใช้สำหรับติดต่อกับผู้ใช้ รวมทั้งการใช้เทคนิคการเขียนโปรแกรมแบบ Event-driven ซึ่งเป็นการเขียนโปรแกรมเพื่อกำหนดขั้นตอนการทำงานให้กับ Control ต่าง ๆ ที่สร้างขึ้นตามเหตุการณ์ (Event) ต่าง ๆ ที่เกิดขึ้น เช่น การเลื่อนเมาส์ หรือการรับข้อมูลจากคีย์บอร์ด ฯลฯ ประกอบกับภาษาที่ใช้เขียนโปรแกรมเป็นภาษา BASIC ซึ่งเป็นภาษาคอมพิวเตอร์ ที่ผู้ใช้นคอมพิวเตอร์ส่วนบุคคลส่วนใหญ่คุ้นเคย จึงส่งผลให้การพัฒนาโปรแกรมบน Windows ด้วย Visual Basic มีขั้นตอนน้อย กระทำได้ง่าย และสะดวกต่อการใช้งาน

2.4.1 การเขียนโปรแกรมติดต่อกับ Serial Port

การเขียนโปรแกรมติดต่อกับ I/O Serial Port สามารถทำได้โดยใช้ Control ที่ชื่อ MSComm ซึ่งจะได้นำ Control MSComm เข้ามาในแถบเครื่องมือ (Toolbox) ของ Project ที่เรากำลังเขียนอยู่ก่อน เพื่อให้สามารถนำ Control MSComm มาใช้งานได้ โดยต้องกำหนด Custom Control เข้าไปที่เมนู Project-->Components แล้วเลือกที่ช่อง MSComm ก็จะปรากฏเป็นรูปไอคอนโทรศัพท์ที่สี่เหลี่ยม ให้คลิกที่ไอคอนลากนำมาไว้บน Form ใน Project ของโปรแกรมโดยสามารถทำตามวิธีที่กล่าวมา



รูปที่ 2.15 แสดงการลาก Control ชื่อ MSComm จาก Toolbox มาไว้บน Form

การเขียนโปรแกรมติดต่อกับ Serial Port สามารถทำได้ 2 วิธี คือ

1.การติดต่อแบบอินเทอร์รัพต์

ขบวนการอินเทอร์รัพต์ อุปกรณ์รอบข้างเกือบทุกชิ้นจะต้องปฏิบัติงานอยู่เพื่อส่งสัญญาณ ไปให้ แก่ซีพียูเสมอ ถ้าอุปกรณ์นั้นพร้อมที่จะรับส่ง ที่เคยเจอจากการทำโครงการอุปกรณ์ จะส่งเป็นรหัส แอสกีเราจะเขียนโปรแกรมอินเทอร์รัพต์ โดยเมื่อที่ข้อมูลเข้ามาจะทำให้มี CommEvent กับ OnComm Event

2.การติดต่อแบบโพลลิ่ง

ในระบบพีซีการโพลมบ้างที่ใช้การส่งผ่านข้อมูลระหว่าง Terminal กับ CPU กรณีข้อมูล เป็นประเภทไบท์ที่ส่งจากคีย์บอร์ด โดยวิธีการนี้จะตรวจสอบ คีย์บอร์ดว่ามีข้อมูลส่งมาหรือเปล่า โดยจะตรวจสอบตลอดเวลา การทำงานกับข้อมูลที่ได้รับเข้ามาจะตรวจสอบด้วยความเร็วที่สูงกว่า อัตราความเร็วข้อมูลที่ส่งเข้ามาทาง คีย์บอร์ด การที่ CPU ส่งสัญญาณออกไปตรวจสอบพบว่ามีข้อมูลที่ต้องส่งเข้ามา เรียกว่า"Wet Poll" ซึ่งจะเสียช่วงเวลา 90 เปอร์เซ็นต์ คาบเวลาที่เสียไปนั้น เรา เลี่ยงไปใช้เทคนิค การโพลแบบ "Round Robin" แทน แต่ในVBเราจะใช้การตรวจสอบข้อมูลที่มา จาก Serial Port ตลอด โดยจะใช้ Control Timer เข้ามาช่วยในการเขียน โปรแกรมซึ่งสามารถตรวจสอบ ได้ถึงระดับ 1 มิลลิวินาที หรือจะใช้ Do...Loop ก็ได้

ในตัวคอนโทรล MSComm มี Event ที่ใช้เพียง Event เดียวเท่านั้นเอง ก็คือ OnComm Event ซึ่งจะใช้ในการติดต่อแบบอินเทอร์รัพต์ การเขียน โปรแกรมติดต่อกับ Serial Port แบบธรรมดา จะใช้ comEvent เพียง comEvReceive,comEvSend ถ้าเป็นการติดต่อสื่อสารแบบโมเด็มจะใช้หลาย ตัวในการตรวจสอบสัญญาณ

2.4.2 องค์ประกอบในการใช้ MSComm

การตั้งค่าติดต่อกับพอร์ต

- ComPort คือ เราต้องกำหนดหมายเลข Port ที่ใช้ต่อRS-232 (Com1,Com2)
- Setting คือ เราต้องกำหนดอัตรา Baud,Parity,Data(จำนวนบิต),Stop ตัวอย่าง 1200,n,8,1 เป็นต้น
- HandShaking คือ เราจะกำหนดได้ 4 แบบ 1.comNone 2.comXonXoff 3. comRTS 4.comTRSXonXoff

การใช้ Buffer ในการรับส่งข้อมูล

- InBuffersize คือ การกำหนด Buffer ในการรับข้อมูลเข้ามา
- OutBuffersize คือ การกำหนด Buffer ในการส่งข้อมูลออกไป
- Rthreshold คือ การที่เรากำหนดการเกิด Event-driven ในการรับข้อมูลเข้ามา

- Sthreshold คือ การที่เรากำหนดการเกิด Event-driven ในการรับข้อมูลออกไป
 - InputLen คือ จำนวนของข้อมูลที่ไปอ่านใน Buffer รับข้อมูล
 - EOFEnable คือ การที่บอกว่าสิ้นสุดของไฟล์(EOF) End of File
- ด้านฮาร์ดแวร์
- ParityReplace คือ ค่าของคาเลกเตอร์ที่จะแทนในเมื่อเกิด Parity Error
 - NullDiscard คือ การกำหนดให้รับหรือไม่รับ NULL CHARACTER
 - RTSEnable คือ ทำให้มีสัญญาณ RTS (Request To Send)
 - DTSEnable คือ ทำให้มีสัญญาณ DTR(Data Terminal Ready)

ตัวอย่างการกำหนดคุณสมบัติของ MSComm Control ให้สามารถติดต่อกับพอร์ตได้

1. Property ชื่อ CommPort คือ เลือกคอมพอร์ตที่เราจะต่อใช้งาน มาดูการเขียน โค้ดกัน
MSComm1.CommPort=1 ในที่นี้เลือกจะใช้ Com1อยู่ที่ด้านหลังเครื่องคอมพิวเตอร์
2. Property ชื่อ Settings คือ การตั้งค่าของการรับส่งข้อมูล ซึ่งจะต้องรู้ด้วยว่าอัตราบอด ของอุปกรณ์ที่จะติดต่อด้วยเป็นเท่าไร โดยมีรายละเอียดการใส่ต่าง ๆ ค่าดังนี้
MSComm1.Settings="Baud(อัตราการรับส่งข้อมูล),Parity(ถ้าไม่ใช้ใส่ N,จำนวนบิตข้อมูล,บิตสตอป การเขียน โค้ด MSComm1.Settings="1200,N,8,1"
- 3.Property ชื่อ InputLen คือ กำหนดขนาดขนาดของข้อมูลเข้ามาให้ไปอ่านข้อมูลทั้งหมดที่อยู่ในบัฟเฟอร์ มาดูการเขียน โค้ดกัน MSComm1.InputLen=1
4. Property ชื่อ PortOpen คือ จะเปิดให้พอร์ตใช้งานหรือไม่ ถ้าเปิด =True ถ้าปิด =False มาดูการเขียน โค้ดกัน MSComm1.PortOpen=True
5. Property ชื่อ Rthreshold คือ ทำให้เกิดการกระตุ้นด้วย Event-driven เมื่อมีข้อมูลในบัฟเฟอร์รับข้อมูล (Comport) มันทำให้เกิด CommEvent ใน OnComm Event มาดูการเขียน โค้ด MSComm1.Rthreshold =1 จากรายละเอียดที่กล่าวมา เราจะมาเขียนใน โพรซีเจอร์ VB ซึ่งจะไว้ที่ Sub Form_Load()

2.4.3 วิธีของการรับส่งข้อมูลจาก Serial Port

จากวิธีเขียน โค้ดด้านบนเป็นการกำหนดค่าเริ่มต้นให้กับคอมพอร์ตและเปิดใช้การรับและส่งของพอร์ต RS-232 ดังนั้นก็สามารถจะรับและส่งข้อมูลทางพอร์ตได้ โดยใช้ Property ดังนี้
Output จะเป็นการส่งข้อมูลไปที่พอร์ตก็คือใช้ MSComm1.out = ข้อมูลที่ต้องการส่ง ซึ่งข้อมูลที่ส่งไปนั้นเป็นค่าแอสกี ส่วน Property Input เป็นส่วนของการรับข้อมูลจากพอร์ต แต่ในส่วนนี้จะต้องนำคำสั่งไปเขียนที่ Event Property OnComm จะอยู่ใน Sub MSComm_OnComm ซึ่งจะอ่านข้อมูลเข้ามาจากทางพอร์ต RS232 นั้นเอง

บทที่ 3

การออกแบบและการสร้าง

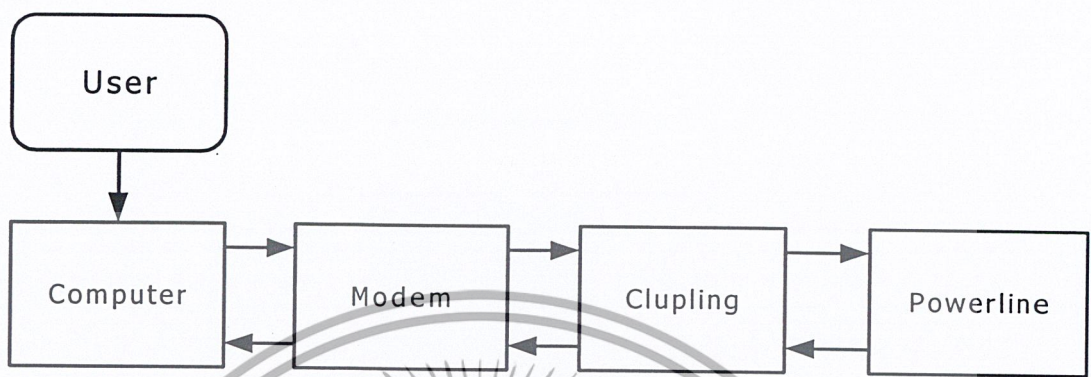
3.1 หลักการทำงานของ การควบคุมอุปกรณ์ไฟฟ้าผ่านสายพาวเวอร์ไลน์

การทำงาน โดยรวมของการควบคุมอุปกรณ์ไฟฟ้าผ่านสายไฟกระแสสลับ 220 โวลต์ ในภาคส่งจะใช้คอมพิวเตอร์ในการส่งควบคุมอุปกรณ์ไฟฟ้าในตำแหน่งที่เราต้องการซึ่งได้แสดงไว้ทั้งหมด 16 ตำแหน่ง โดยจะส่งสัญญาณควบคุมออกทางพอร์ตอนุกรมของคอมพิวเตอร์ สัญญาณที่ออกจากคอมพิวเตอร์ต้องแปลงแรงดันเป็นสัญญาณ TTL ก่อนโดยผ่านไอซี MAX 232 ต่อไปก็จะนำสัญญาณนี้ไปมอดูเลต ซึ่งจะใช้ไอซี TDA5051A ในการมอดูเลตสัญญาณควบคุม สัญญาณที่ผ่านการมอดูเลตแล้วจะผ่านวงจรขับปลั๊กก่อนที่จะส่งไปบนสายไฟกระแสสลับ เพื่อเป็นการกันไฟกระแสสลับไม่ให้เข้ามายังภาคส่งได้

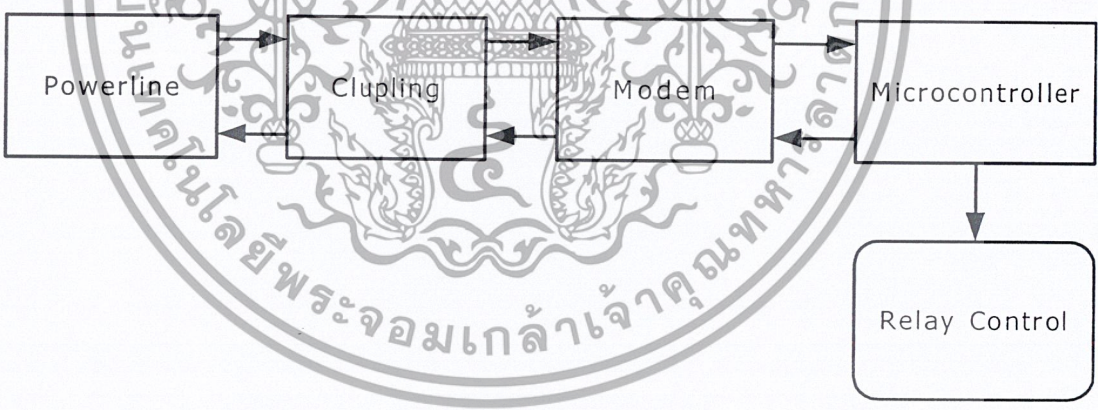
ส่วนทางภาครับก็จะมีวงจรขับปลั๊กอีกตัวหนึ่งต่ออยู่กับไฟกระแสสลับ 220 โวลต์ เพื่อกันไฟกระแสสลับไม่ให้เข้ามาทำความเสียหายแก่ภาครับ สัญญาณที่ผ่านวงจรขับปลั๊กมาแล้ว จะส่งไปดีมอดสัญญาณซึ่งจะใช้ไอซี TDA5051A เป็นตัวดีมอดสัญญาณสัญญาณที่ผ่านการดีมอดแล้วจะส่งต่อไปยังไมโครคอนโทรลเลอร์ ไมโครคอนโทรลเลอร์จะเป็นตัวประมวลผลว่าจะเปิดปิดสวิตซ์ตัวไหน โดยการส่งสัญญาณไปที่พอร์ตต่าง ๆ ของตัวไมโครคอนโทรลเลอร์เพื่อไปควบคุมตัวรีเลย์ให้สามารถเปิดปิดสวิตซ์ได้ตามต้องการ

ในส่วนของการเขียนโปรแกรมในภาคส่งซึ่งก็คือคอมพิวเตอร์จะใช้โปรแกรม Visual Basic ในการควบคุมไมโครคอนโทรลเลอร์ โดยจะมีการตรวจสอบระบบก่อนที่มีการส่งสัญญาณควบคุมทำให้สามารถทราบได้ว่าระบบพร้อมที่จะทำงานหรือไม่ สำหรับในส่วนติดต่อกับผู้ใช้ก็จะมีเมนูที่ทำให้สะดวกต่อการควบคุมอุปกรณ์ สำหรับไมโครคอนโทรลเลอร์ก็มีการเขียน โปรแกรมที่จะตอบสนองสัญญาณควบคุมจากคอมพิวเตอร์และสามารถควบคุมพอร์ตต่าง ๆ ของไมโครคอนโทรลเลอร์ได้ตามต้องการ

ส่วนสำคัญในการควบคุมอุปกรณ์ไฟฟ้าผ่านสายไฟกระแสสลับ 220 โวลต์ก็คือ ไอซี TDA5051A ซึ่งเป็นไอซีโมเด็มซึ่งทำหน้าที่เป็นทั้งตัวมอดูเลตและดีมอดูเลตสัญญาณ หากขาดไอซีตัวนี้แล้วการส่งสัญญาณไปบนสายไฟกระแสสลับก็ไม่สามารถเกิดขึ้นได้



Block diagram ของภาครับ



Block diagram ของภาคส่ง

รูปที่ 3.1 Block diagram การทำงานของการควบคุมอุปกรณ์ไฟฟ้าผ่านสายเพาเวอร์ไลน์

3.2 ภาคมอดดูเลตและดีมอดดูเลต

จะทำการมอดดูเลตในภาคส่งและดีมอดดูเลตในภาครับ โดยใช้ TDA 5051 เป็นอุปกรณ์สำคัญ TDA 5051 เป็น ASK MODEM ที่สร้างขึ้นเพื่อส่งข้อมูลไปบนสายไฟ AC มีอุปกรณ์ดับปลิงภายนอกอีกเล็กน้อยก็จะทำงานได้แล้ว การส่งข้อมูลไปบนสายไฟ AC นั้นจะเปลี่ยนข้อมูลโดยวิธีการมอดดูเลตแบบ ASK

TDA 5051 จะทำงานเป็นทั้งภาครับและภาคส่งอย่างมีประสิทธิภาพโดยการใช้วิธีการมอดดูเลตและดีมอดดูเลตด้วยเบาค์เรตต่ำ



รูปที่ 3.2 แสดงวงจรที่ใช้ในการมอดดูเลตและดีมอดดูเลต

คุณลักษณะหลักของไอซี

- ภายในไอซีเป็น CMOS ใช้พลังงานน้อยมีเสถียรภาพที่อุณหภูมิสูง
- ใช้กับไฟ 5 โวลต์ซึ่งเป็นค่าเดียวกับตัวคอนโทรลเลอร์
- ต้องใช้ความถี่ภายนอกป้อนโดยใช้คริสตอล ความถี่ที่ป้อนนี้จะใช้เป็นความถี่ในการทำงานและเป็นความถี่เอาท์พุทที่อาจต่อไปยังไมโครคอนโทรลเลอร์

- มีการตรวจสอบอย่างสมบูรณ์แล้ว ทั้งก่อนและหลังบรรจุ
- มีเบรคเรต 600 สำหรับการใช้ทั่วไปและเบรคเรตสูงสุด 1200
- การขยายที่อินพุทเป็นแบบ Automatic Gain Control สามารถขยายสัญญาณเล็ก ๆ

ได้ในตอนที่สาย AC มีโหลดเยอะ

การเลือกความถี่ Carrier Frequency

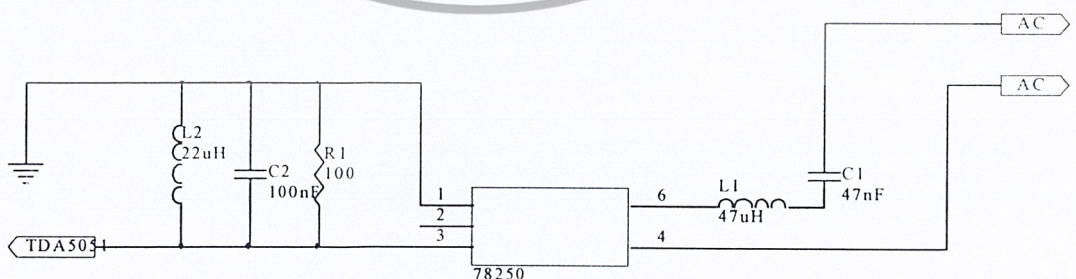
ความถี่ที่ใช้งานของโมเด็มกำหนดโดยการเลือกความถี่จากคริสตอล มีค่าอยู่ในช่วง 6.5MHz ถึง 8.886 MHz

ความถี่ออสซิลเลต	ความถี่ Clock Output	ความถี่ Carrier และ Detection
Fosc	Fosc/2	Fosc/64

3.3 ภาคคัปปลิงสัญญาณ

จะอยู่ที่ในภาครีบและภาคส่งสัญญาณ วงจรคัปปลิงจะเป็นตัวกันไฟกระแสสลับ 220 โวลต์ไม่ให้เข้ามาทำความเสียหายแก่ระบบ สัญญาณที่ต้องการเท่านั้นที่สามารถผ่านเข้าออกได้ วงจรพื้นฐานของภาคคัปปลิงคือ วงจรกรองความถี่สูงผ่าน ซึ่งวงจรนี้จะไม่ลดทอนความถี่พาหะแต่จะลดทอนไฟกระแสสลับ 220 โวลต์ นอกจากนี้วงจรภาคคัปปลิงยังใช้คุณสมบัติของวงจร bandpass filter ด้วย วงจร bandpass filter จะมีความถี่ f_c เท่ากับความถี่ carrier ของตัว TDA 5051 ที่มอดสัญญาณ ในด้านส่ง bandpass filter จะกำจัดความถี่ฮาร์โมนิกส์ทิ้งไป

ภาคคัปปลิงประกอบด้วย 1.) วงจรรีโซแนนซ์แบบอนุกรมประกอบด้วย L1, C1
2.) ตัวทรานฟอเมอร์คัปปลิงเป็น Hightfrequency Transformer ใช้ เบอร์ 78250 3.) วงจรรีโซแนนซ์แบบอนุกรมประกอบด้วย L2, C2



รูปที่ 3.3 แสดงวงจรคัปปลิง

3.4 ไมโครคอนโทรลเลอร์

ไมโครคอนโทรลเลอร์เป็นตัวที่รับสัญญาณที่ผ่านการคิมอดแล้วมาจากตัว TDA5051 เพื่อนำมาประมวลผลว่าจะเปิดปิดสวิทซ์ตัวใด โดยจะรับสัญญาณเข้ามาที่ขา 10 คือขา RX ซึ่งขานี้จะทำหน้าที่รับอินพุตเข้ามาแบบอนุกรม ซึ่งได้กล่าวรายละเอียดเกี่ยวกับการใช้งานพอร์ตอนุกรมไว้อย่างละเอียดแล้วในบทที่ 2 สัญญาณที่รับมานี้จะเป็นรหัสแอสกีที่ส่งมาจากคอมพิวเตอร์ เราจะนำรหัสแอสกีที่รับได้มาเขียน โปรแกรมเพื่อควบคุมพอร์ตต่าง ๆ ของไมโครคอนโทรลเลอร์ สัญญาณที่ส่งออกจากพอร์ตจะนำไปควบคุมรีเลย์เพื่อให้เปิดปิดอุปกรณ์ไฟฟ้าได้ตามต้องการ



รูปที่ 3.4 แสดงการต่อไมโครคอนโทรลเลอร์

3.5 การทำงานของภาคควบคุมอุปกรณ์ (ON-OFF)

ภาคนี้จะประกอบด้วยวงจรขั้วรีเลย์ เราสามารถสั่งควบคุมอุปกรณ์ไฟฟ้ากระแสสลับต่าง ๆ ได้โดยสั่งผ่านทางรีเลย์ โดยวงจรควบคุมรีเลย์จะต่อจากพอร์ตของไมโครคอนโทรลเลอร์ซึ่งอยู่ในภาครับที่พอร์ต 1, 2 มีการต่อวงจรขั้วรีเลย์ทั้งหมด 16 ชุด

การทำงานจะเริ่มจากการส่งสัญญาณเพื่อเลือกตำแหน่งของรีเลย์ที่ต้องการเปิดปิดจากคอมพิวเตอร์ ซึ่งจะมีสวิทช์ให้เลือกทั้งหมด 16 ตัว เมื่อมีการกดสวิทช์ที่ตำแหน่งใดคอมพิวเตอร์จะรหัสแอสกีที่ตรงกับสวิทช์ตำแหน่งนั้นออกไปยังไมโครคอนโทรลเลอร์ที่ภาครับส่ง ไมโครคอนโทรลเลอร์จะทำการประมวลผลว่ารหัสแอสกีที่รับเข้ามาตรงกับตำแหน่งใด ซึ่งก็คือพอร์ตของไมโครคอนโทรลเลอร์จะถูกทำการคอมพ्लीเมนต์ค่าที่ส่งออกพอร์ต สถานะที่ขาพอร์ตของไมโครคอนโทรลเลอร์ก็จะเป็น 1, 0 สลับกันไป



รูปที่ 3.5 วงจรรีเลย์

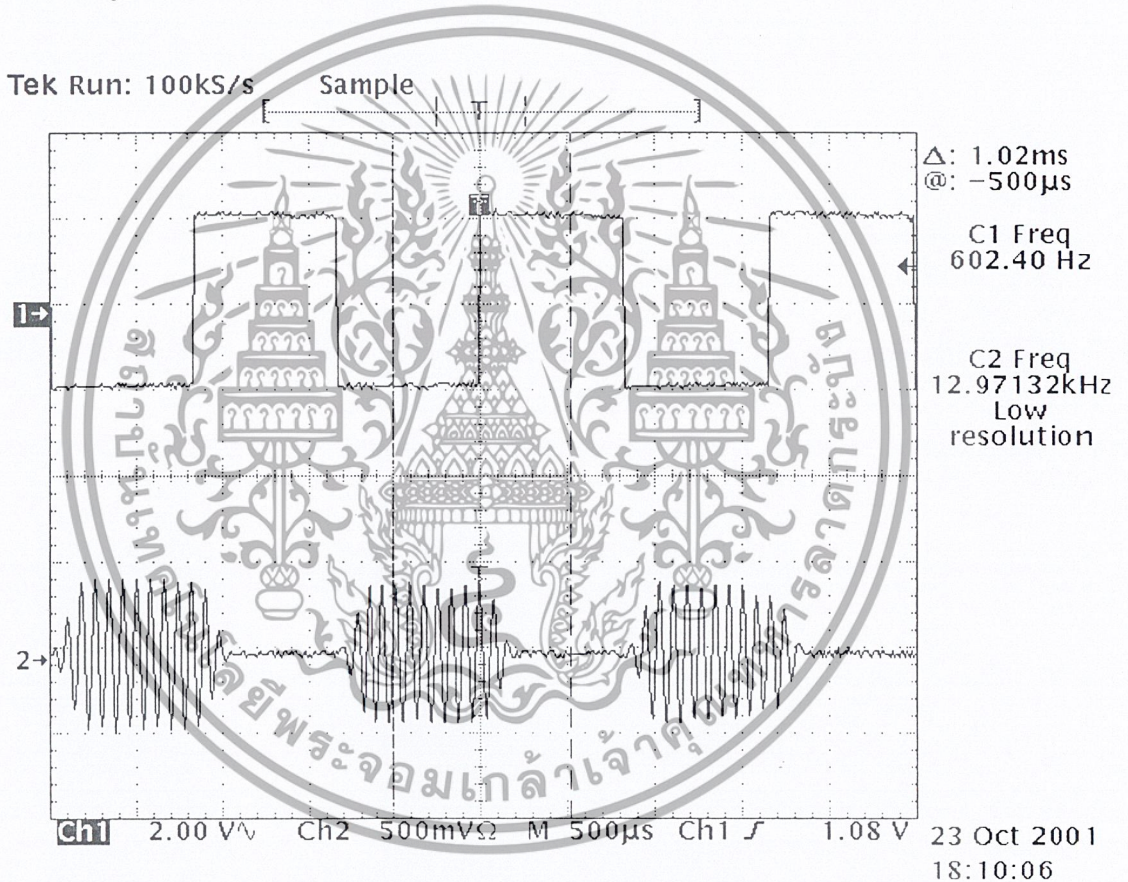
เมื่อขาพอร์ตเป็นมีสถานะเป็นลอจิก 0 ไฟขา B จะเป็น 0 ทรานซิสเตอร์ Q1 อยู่ในสถานะ OFF จะทำให้ไฟเลี้ยงรีเลย์ยังไม่ครบวงจรรีเลย์จะยังไม่ทำงาน เมื่อขาพอร์ตเป็นมีสถานะเป็นลอจิก 1 ทรานซิสเตอร์ Q1 อยู่ในสถานะ ON จะทำให้ไฟเลี้ยงรีเลย์ยังครบวงจรของรีเลย์สามารถทำงานได้ หากเราจะเปิดปิดอุปกรณ์ไฟฟ้าตัวใดก็นำเอาสายไฟกระแสสลับเส้นใดเส้นหนึ่งมาต่อที่รีเลย์ขา 1, 2 ก็จะสามารถเปิดปิดอุปกรณ์นั้นได้

บทที่ 4

การทดลองและผลการทดลอง

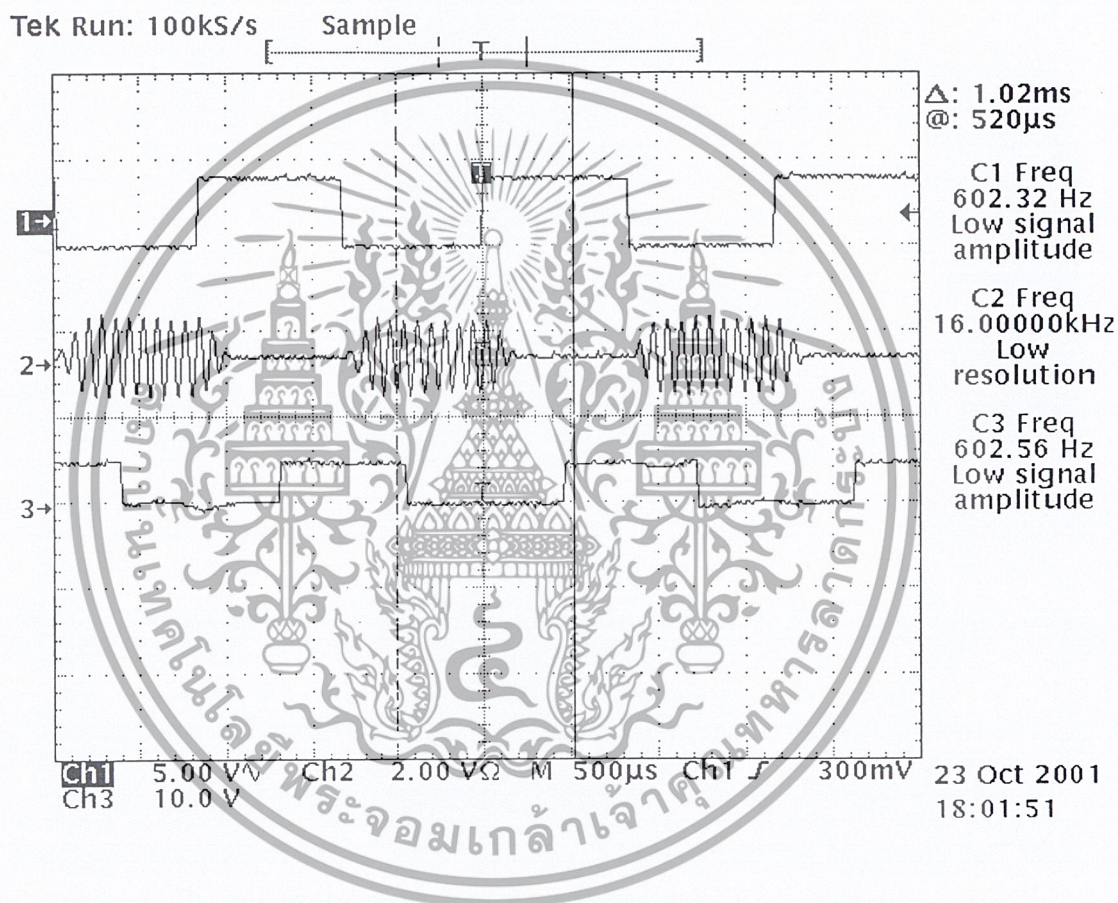
4.1 รูปสัญญาณที่ได้รับจากการมอดและดีมอดสัญญาณ

จากการทดลองป้อนสัญญาณ TTL ความถี่ 600 Hz เข้าที่ขา DATA INPUT ของตัว TDA5051 แล้ววัดสัญญาณที่ขา TX OUT ซึ่งเป็นสัญญาณที่มอดดูเลตแล้ว จากรูปที่ 4.1 สัญญาณจากออสซิลอสโคปช่องที่ 1 เป็นสัญญาณ TTL ความถี่ 600 Hz ช่องที่ 2 เป็นรูปสัญญาณที่ได้จากการมอดดูเลต



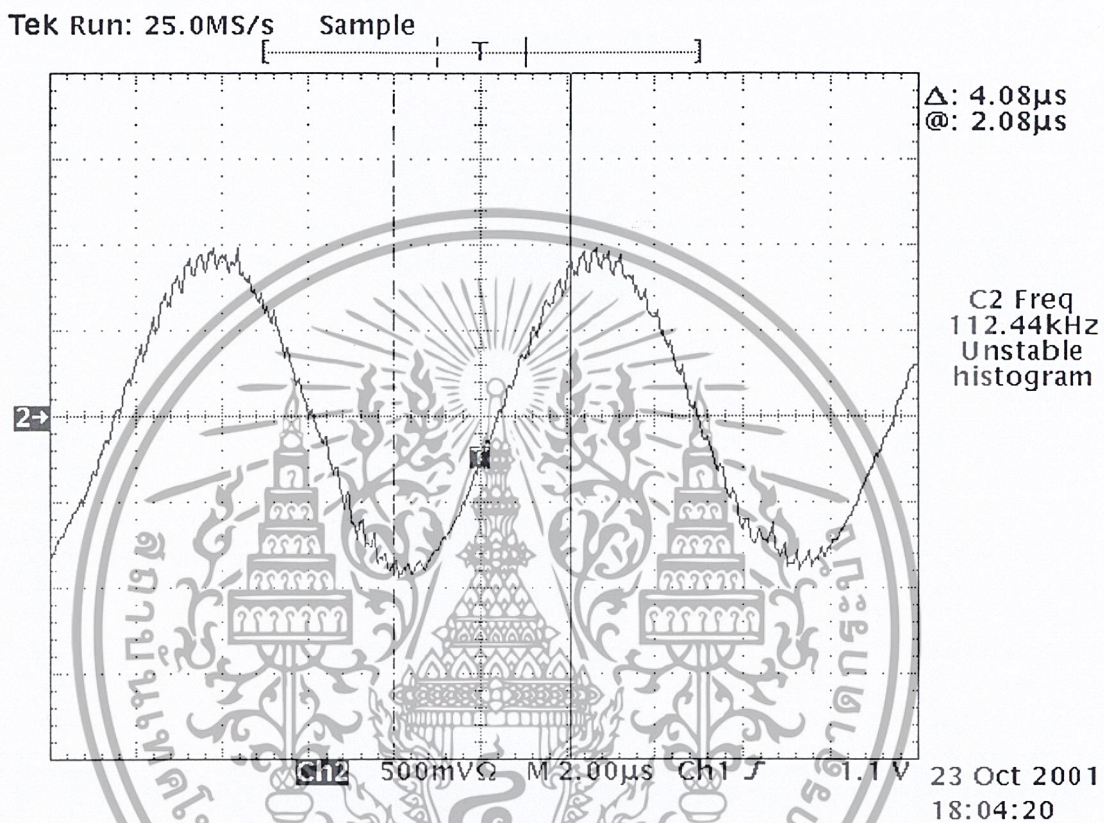
รูปที่ 4.1 รูปสัญญาณที่ได้จากการมอดดูเลต

รูปที่ 4.2 เป็นการทดสอบการตีมอตคูเลตสัญญาณ วัดที่ขาDATAOUT ของตัว TDA5051A ซึ่งจะเห็นว่าตัว TDA5051A สามารถตีมอตสัญญาณออกมาได้ สัญญาณที่ได้จากการตีมอตคูเลตจะเกิดการดีเลย์ขึ้น แต่สัญญาณนี้ก็ยังสามารถใช้งานได้ สัญญาณจากออสซิลโลสโคปช่องที่ 1 เป็นสัญญาณ TTL ความถี่ 600 Hz สัญญาณช่องที่ 2 เป็นสัญญาณที่ได้จากการมอตคูเลตสัญญาณช่องที่ 3 เป็นสัญญาณที่ได้จากการตีมอตคูเลต



รูปที่ 4.2 รูปสัญญาณที่ได้จากการมอตคูเลตและตีมอตคูเลต

รูปที่ 4.3 แสดงความถี่พาหะที่ใช้ในการมอดูเลตสัญญาณ ซึ่งจากทฤษฎีความถี่พาหะจะมีค่าเท่ากับ ความถี่ของคริสตอลหารด้วย 64 ซึ่งจากกราฟทดลองจะมีความถี่พาหะเท่า 112.44 KHz



รูปที่ 4.3 แสดงรูปความถี่ CARRIER ที่ใช้ในการมอดูเลต

4.2 ทดลองวัดค่าแรงดันที่ผ่านวงจรคัปปลิ่งทางด้านรับ

โดยการวัดแรงดันที่ผ่านวงจรคัปปลิ่งทางรับแล้วซึ่งจะมีรูปสัญญาณเหมือนกับสัญญาณที่ผ่านการมอดูเลตแล้ว วัดที่ความยาวต่างกันจะให้ค่าแรงดันออกมาต่างกัันดังตาราง

ระยะทางจากภาคส่ง (เมตร)	ค่าแรงดัน (Vp-p)
0	3.6
5	3.6
50	1.4
200	0.1
250	0.1

ตารางที่ 4.1 ผลจากการวัดแรงดันที่ความยาวสายต่าง ๆ

จากตารางจะเห็นว่าค่าแรงดันจะลดลงไปตามความยาวของสาย ซึ่งจากการทดลองแม้ว่าแรงดันที่ได้จะมีค่าลดลงแต่รูปสัญญาณที่ได้ยังคงมีลักษณะเหมือนตอนที่ออกมาจาก TXOUT สัญญาณรบกวนน้อยมาก

บทที่ 5

สรุปและวิจารณ์ผลการทดลอง

5.1 สรุปผลการทดลอง

ปริญญานิพนธ์ฉบับนี้เสนอผลงานเกี่ยวกับการควบคุมอุปกรณ์ไฟฟ้าภายในบ้าน ผ่านสายไฟกระแสสลับ ซึ่งจากการทดลองส่งสัญญาณควบคุมจากคอมพิวเตอร์ไปบนสายไฟกระแสสลับ โดยผ่าน IC MODEM แล้วทำการควบคุมไมโครคอนโทรลเลอร์ สามารถทำการควบคุมพอร์ตต่าง ๆ ของ ไมโครคอนโทรลเลอร์ได้ และระยะทางที่สามารถส่งได้ก็ครอบคลุมพื้นที่บ้าน 1 หลัง โดยอาศัยวงจรขยายสัญญาณช่วย โดยรวมแล้วการควบคุมทำได้เป็นอย่างดีตามความต้องการ

5.2 ปัญหาและแนวทางแก้ไข

จากการทดลองส่งสัญญาณควบคุม ทำให้ทราบถึงปัญหาที่เกิดขึ้นและทำการแก้ไขได้ดังนี้

1. ขนาดของสัญญาณเมื่อส่งไประยะทางไกล ๆ จะมีการลดทอนของสัญญาณลง และหากมีระยะทางไกลมาก ๆ ระดับขนาดสัญญาณจะลง แต่เนื่องจากรูปคลื่นสัญญาณที่ได้ยังคงมีลักษณะที่ไม่ผิดเพี้ยน จึงแก้ไขได้โดยการใช้วงจรขยายสัญญาณช่วย
2. ในส่วนของการส่งสัญญาณดิจิตอลอินพุตจากคอมพิวเตอร์ หากว่าส่วนของ IC MAX232 ซึ่งทำหน้าที่แปลงแรงดันจากคอมพิวเตอร์ให้เป็นแรงดันที่อยู่ในระดับ TTL เกิดการเสียหาย อาจทำให้แรงดันไฟจากพอร์ตอนุกรมของคอมพิวเตอร์เข้ามายังขา DATAIN ซึ่งจะมีระดับแรงดันที่มากกว่า แรงดันสูงสุดของไอซี จะทำให้ไอซีเสียหาย สามารถแก้ไขได้โดยการใช้ซีเนอร์ไดโอดควบคุมระดับแรงดันไฟ

5.3 การพัฒนาโครงการ

โครงการที่สร้างขึ้นนี้ สามารถทำงานได้ตามขีดความสามารถในวัตถุประสงค์ที่ได้กำหนดไว้ แต่สามารถเพิ่มขีดความสามารถในการทำงานขึ้นได้อีกดังเช่นต่อไปนี้

1. ขยายระยะทางในการส่งโดยใช้วงจรขยายสัญญาณที่มีประสิทธิภาพ
2. เพิ่มจำนวนภาครับได้อีกโดยการพัฒนาที่ซอฟต์แวร์
3. นอกจากนี้ยังสามารถประยุกต์การควบคุมไมโครคอนโทรลเลอร์ไปใช้งานได้หลากหลาย

หนังสืออ้างอิง

1. วรพจน์ กรแก้ว, ชัยวัฒน์ ลิ้มพรจิตรวิไล, เรียนรู้และปฏิบัติการไมโครคอนโทรลเลอร์ MCS-51, Innovative Experiment Co., Ltd.
2. อุดม จีนประดับ, ไมโครคอนโทรลเลอร์ MCS-51, สถาบันเทคโนโลยีพระจอมเกล้าพระนครเหนือ, พิมพ์ครั้งที่ 1
3. สุรชัย เพิ่มสินทวี และ ธเนศ สุวรรณผ่อง, การสื่อสารข้อมูลด้วย MODEM, บริษัท ซีเอ็ดยูเคชั่น จำกัด(มหาชน)
4. คารณิ ทองไทยนันท์, การสื่อสารข้อมูล, บริษัท พัฒนาวิชาการ(2535) จำกัด, พิมพ์ครั้งที่ 1
5. สมยศ จุณณะปิยะ, การประยุกต์ใช้งานไมโครคอนโทรลเลอร์, สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง, พิมพ์ครั้งที่ 3
6. www.Thaiio.com





เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โฟลว์ชาร์ตการทำงานของไมโครคอนโทรลเลอร์



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โปรแกรมควบคุมการทำงานของไมโครคอนโทรลเลอร์

```

ORG 0000H
MOV SCON,#50H
MOV IE,#00
MOV PCON,#00H
MOV TMOD,#20H
MOV TH1,#0E8H
SETB TR1
MOV P1,#00H
MOV P2,#00H
MOV A,#5AH      ;'Z' TO COMPUTER
MOV SBUF,A
CLR TI
JNB TI,$
RX: CLR RI
   JNB RI,$
   MOV A,SBUF
   CJNE A,#41H,TO_B ;1ST CONTROL
   CPL P1.0
   MOV A,#61H      ;'a' TO COMPUTER
   MOV SBUF,A
   CLR TI
   JNB TI,$
   LJMP RX
TO_B: CJNE A,#42H,TO_C ;2ND CONTROL
      CPL P1.1
      MOV A,#62H      ;'b' TO COMPUTER
      MOV SBUF,A

```

```

CLR TI
JNB TI,$
LJMP RX
TO_C: CJNE A,#43H,TO_D ;3RD CONTROL
CPL P1.2
MOV A,#63H ;'c' TO COMPUTER
MOV SBUF,A
CLR TI
JNB TI,$
LJMP RX
TO_D: CJNE A,#44H,TO_E ;4TH CONTROL
CPL P1.3
MOV A,#64H ;'d' TO COMPUTER
MOV SBUF,A
CLR TI
JNB TI,$
LJMP RX
TO_E: CJNE A,#45H,TO_F ;5TH CONTROL
CPL P1.4
MOV A,#65H ;'e' TO COMPUTER
MOV SBUF,A
CLR TI
JNB TI,$
LJMP RX
TO_F: CJNE A,#46H,TO_G ;6TH CONTROL
CPL P1.5
MOV A,#66H ;'f' TO COMPUTER
MOV SBUF,A
CLR TI
JNB TI,$

```

```

LJMP RX
TO_G: CJNE A,#47H,TO_H ;7TH CONTROL
      CPL P1.6
      MOV A,#67H ;'g' TO COMPUTER
      MOV SBUF,A
      CLR TI
      JNB TI,$
      LJMP RX
TO_H: CJNE A,#48H,TO_I ;8TH CONTROL
      CPL P1.7
      MOV A,#68H ;'h' TO COMPUTER
      MOV SBUF,A
      CLR TI
      JNB TI,$
      LJMP RX
TO_I: CJNE A,#49H,TO_J ;9TH CONTROL
      CPL P2.0
      MOV A,#69H ;'i' TO COMPUTER
      MOV SBUF,A
      CLR TI
      JNB TI,$
      LJMP RX
TO_J: CJNE A,#4AH,TO_K ;10TH CONTROL
      CPL P2.1
      MOV A,#6AH ;'j' TO COMPUTER
      MOV SBUF,A
      CLR TI
      JNB TI,$
      LJMP RX
TO_K: CJNE A,#4BH,TO_L ;11TH CONTROL

```

```

CPL P2.2
MOV A,#6BH      ;'k' TO COMPUTER
MOV SBUF,A
CLR TI
JNB TI,$
LJMP RX
TO_L:  CJNE A,#4CH,TO_M    ;12TH CONTROL
CPL P2.3
MOV A,#6CH      ;'l' TO COMPUTER
MOV SBUF,A
CLR TI
JNB TI,$
LJMP RX
TO_M:  CJNE A,#4DH,TO_N    ;13TH CONTROL
CPL P2.4
MOV A,#6DH      ;'m' TO COMPUTER
MOV SBUF,A
CLR TI
JNB TI,$
LJMP RX
TO_N:  CJNE A,#4EH,TO_O    ;14TH CONTROL
CPL P2.5
MOV A,#6EH      ;'n' TO COMPUTER
MOV SBUF,A
CLR TI
JNB TI,$
LJMP RX
TO_O:  CJNE A,#4FH,TO_P    ;15TH CONTROL
CPL P2.6
MOV A,#6FH      ;'o' TO COMPUTER

```

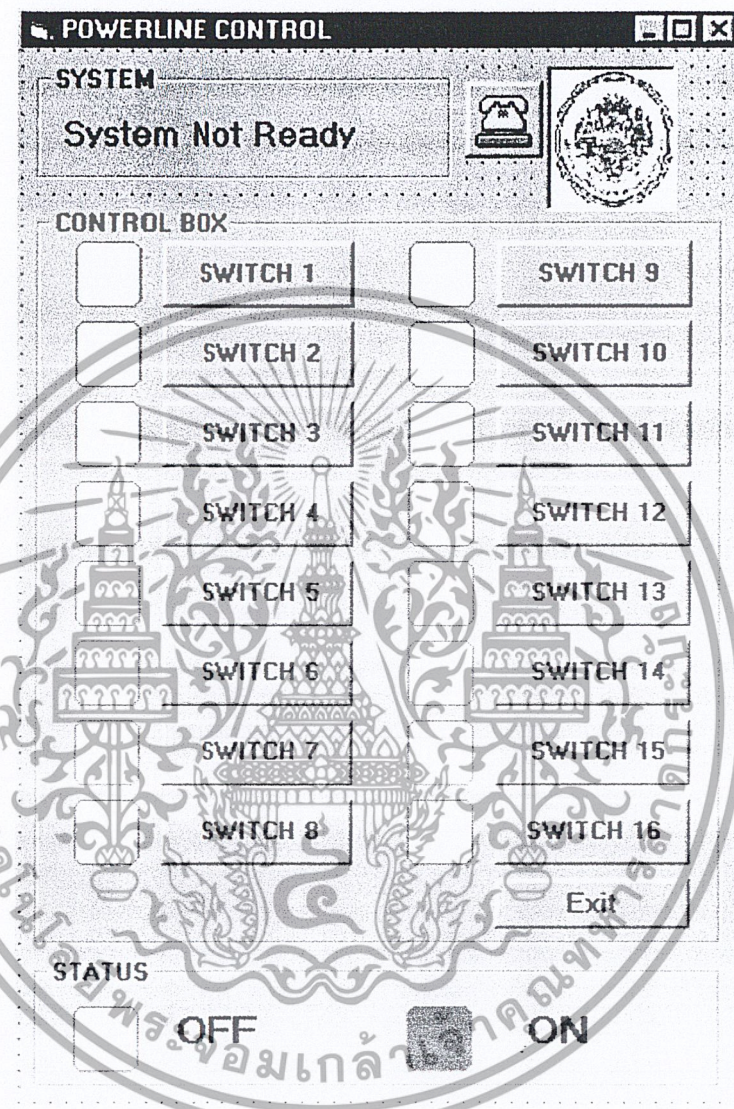
```

MOV SBUF,A
CLR TI
JNB TI,$
LJMP RX
TO_P: CJNE A,#50H,TO_RX      ;16TH CONTROL
CPL P2.7
MOV A,#70H      ;'p' TO COMPUTER
MOV SBUF,A
CLR TI
JNB TI,$
LJMP RX
TO_RX: LJMP RX
END

```



โปรแกรมที่ใช้ในการสั่งควบคุมจากคอมพิวเตอร์



Dim sw1, sw2, sw3, sw4, sw5, sw6, sw7, sw8, sw9, sw10, sw11, sw12, sw13, sw14, sw15, sw16

As Boolean

Option Explicit

```
Private Sub Command1_Click()
```

```
MSComm1.Output = "A"
```

```
End Sub
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
Private Sub Command10_Click()
```

```
MSComm1.Output = "J"
```

```
End Sub
```

```
Private Sub Command11_Click()
```

```
MSComm1.Output = "K"
```

```
End Sub
```

```
Private Sub Command12_Click()
```

```
MSComm1.Output = "L"
```

```
End Sub
```

```
Private Sub Command13_Click()
```

```
MSComm1.Output = "M"
```

```
End Sub
```

```
Private Sub Command14_Click()
```

```
MSComm1.Output = "N"
```

```
End Sub
```

```
Private Sub Command15_Click()
```

```
MSComm1.Output = "O"
```

```
End Sub
```

```
Private Sub Command16_Click()
```

```
End
```

```
End Sub
```

```
Private Sub Command17_Click()
```

```
MSComm1.Output = "P"
```

End Sub

Private Sub Command2_Click()

MSComm1.Output = "B"

End Sub

Private Sub Command3_Click()

MSComm1.Output = "C"

End Sub

Private Sub Command4_Click()

MSComm1.Output = "D"

End Sub

Private Sub Command5_Click()

MSComm1.Output = "E"

End Sub

Private Sub Command6_Click()

MSComm1.Output = "F"

End Sub

Private Sub Command7_Click()

MSComm1.Output = "G"

End Sub

Private Sub Command8_Click()

MSComm1.Output = "H"

End Sub

```
Private Sub Command9_Click()
```

```
MSComm1.Output = "1"
```

```
End Sub
```

```
Private Sub Form_Load()
```

```
Dim Check As Variant
```

```
MSComm1.CommPort = 1
```

```
MSComm1.Settings = "1200,n,8,1"
```

```
MSComm1.InputLen = 1
```

```
MSComm1.PortOpen = True
```

```
MSComm1.RThreshold = 1
```

```
End Sub
```

```
Private Sub MSComm1_OnComm()
```

```
Dim Check As Variant
```

```
Check = MSComm1.Input
```

```
If Check = "Z" Then
```

```
Label1.Caption = "System Ready"
```

```
End If
```

```
If Check = "a" Then
```

```
sw1 = Not sw1
```

```
If sw1 = True Then
```

```
Shape1.FillColor = &HFF&
```

```
End If
```

```
If sw1 = False Then
```

```
Shape1.FillColor = &HFFFFFF
```

```
End If
```

```
End If
```

```
If Check = "b" Then
```

```
sw2 = Not sw2
```

```

If sw2 = True Then
Shape2.FillColor = &HFF&
End If
If sw2 = False Then
Shape2.FillColor = &HFFFFFF
End If
End If
If Check = "c" Then
sw3 = Not sw3
If sw3 = True Then
Shape3.FillColor = &HFF&
End If
If sw3 = False Then
Shape3.FillColor = &HFFFFFF
End If
End If
If Check = "d" Then
sw4 = Not sw4
If sw4 = True Then
Shape4.FillColor = &HFF&
End If
If sw4 = False Then
Shape4.FillColor = &HFFFFFF
End If
End If
If Check = "e" Then
sw5 = Not sw5
If sw5 = True Then
Shape5.FillColor = &HFF&
End If

```



```

If sw5 = False Then
Shape5.FillColor = &HFFFFFF
End If
End If
If Check = "f" Then
sw6 = Not sw6
If sw6 = True Then
Shape6.FillColor = &HFF&
End If
If sw6 = False Then
Shape6.FillColor = &HFFFFFF
End If
End If
If Check = "g" Then
sw7 = Not sw7
If sw7 = True Then
Shape7.FillColor = &HFF&
End If
If sw7 = False Then
Shape7.FillColor = &HFFFFFF
End If
End If
If Check = "h" Then
sw8 = Not sw8
If sw8 = True Then
Shape8.FillColor = &HFF&
End If
If sw8 = False Then
Shape8.FillColor = &HFFFFFF
End If

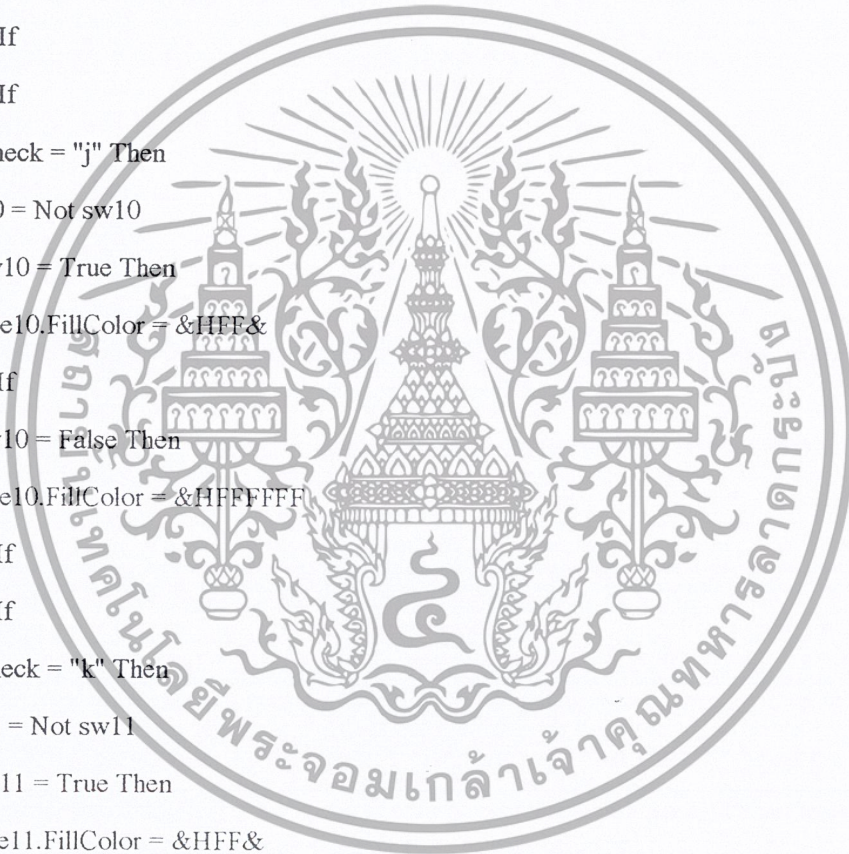
```



```

End If
If Check = "i" Then
sw9 = Not sw9
If sw9 = True Then
Shape9.FillColor = &HFF&
End If
If sw9 = False Then
Shape9.FillColor = &HFFFFFF
End If
End If
If Check = "j" Then
sw10 = Not sw10
If sw10 = True Then
Shape10.FillColor = &HFF&
End If
If sw10 = False Then
Shape10.FillColor = &HFFFFFF
End If
End If
If Check = "k" Then
sw11 = Not sw11
If sw11 = True Then
Shape11.FillColor = &HFF&
End If
If sw11 = False Then
Shape11.FillColor = &HFFFFFF
End If
End If
If Check = "l" Then
sw12 = Not sw12

```



```

If sw12 = True Then
Shape12.FillColor = &HFF&
End If

If sw12 = False Then
Shape12.FillColor = &HFFFFFF
End If

End If

If Check = "m" Then
sw13 = Not sw13

If sw13 = True Then
Shape13.FillColor = &HFF&
End If

If sw13 = False Then
Shape13.FillColor = &HFFFFFF
End If

End If

If Check = "n" Then
sw14 = Not sw14

If sw14 = True Then
Shape14.FillColor = &HFF&
End If

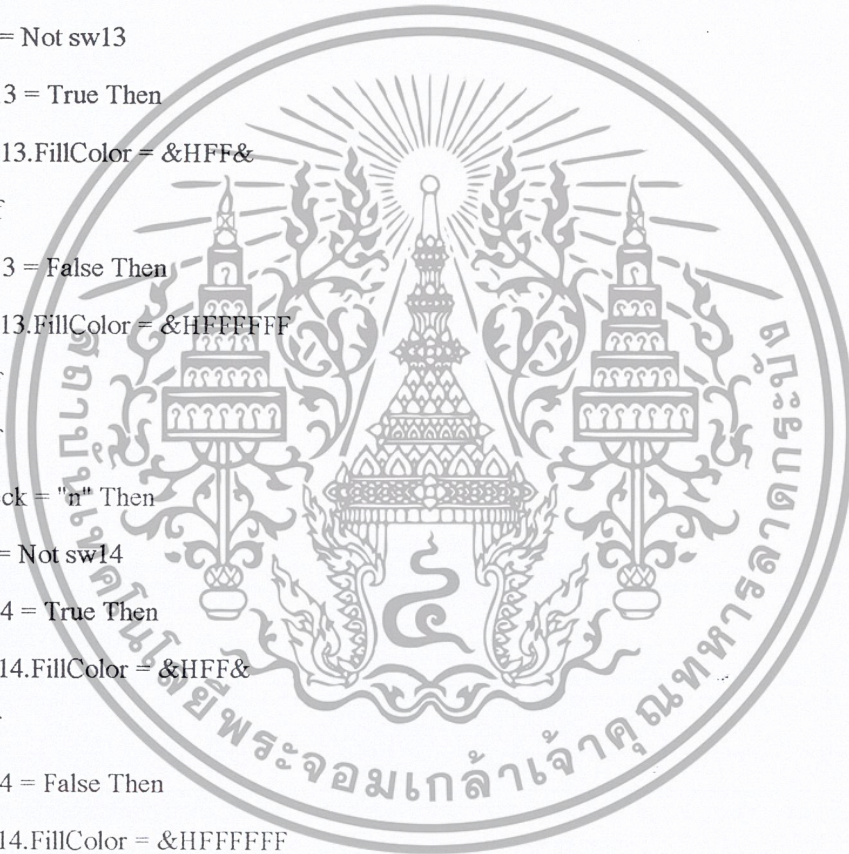
If sw14 = False Then
Shape14.FillColor = &HFFFFFF
End If

End If

If Check = "o" Then
sw15 = Not sw15

If sw15 = True Then
Shape15.FillColor = &HFF&
End If

```



```

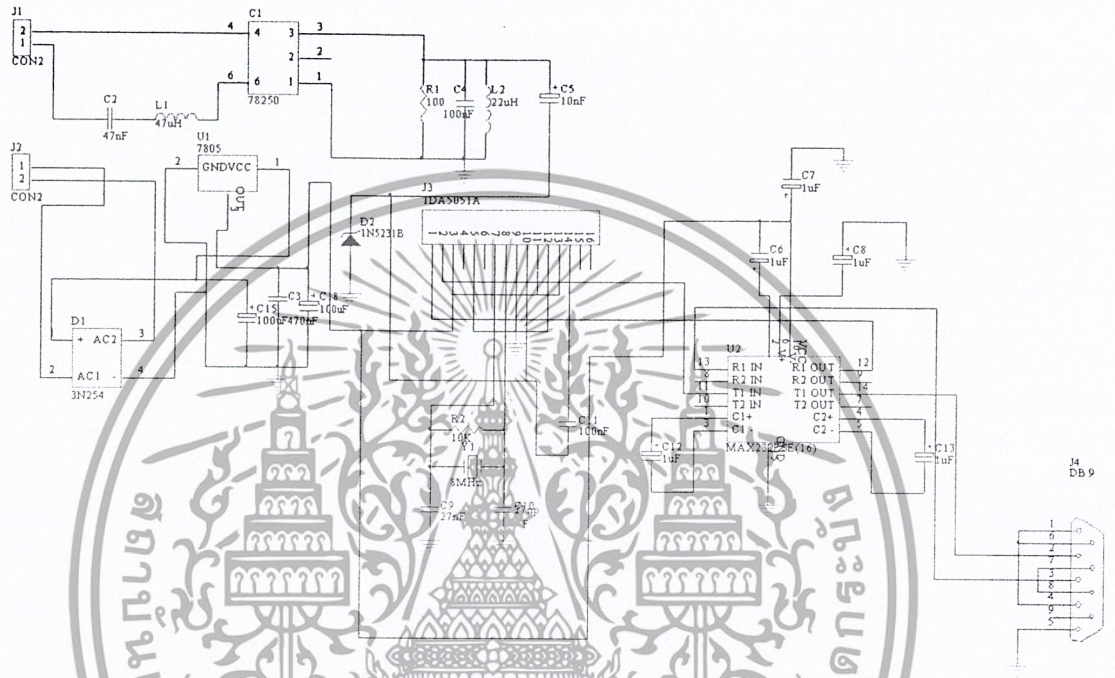
If sw15 = False Then
Shape15.FillColor = &HFFFFFFF
End If
End If
If Check = "p" Then
sw16 = Not sw16
If sw16 = True Then
Shape16.FillColor = &HFF&
End If
If sw16 = False Then
Shape16.FillColor = &HFFFFFFF
End If
End If
End Sub

```



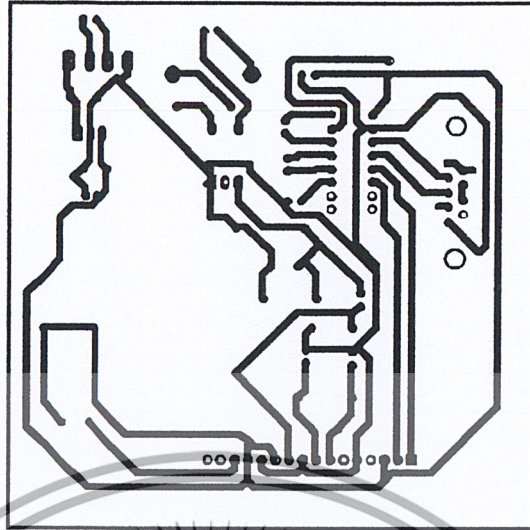


เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

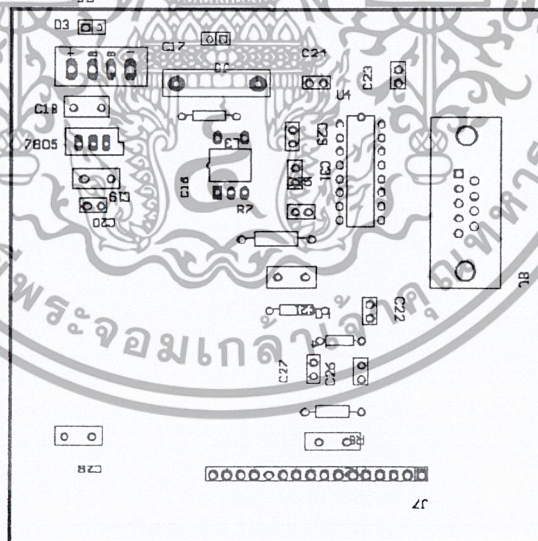


รูปวงจรถ่ายส่ง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปลายวงจรของวงจรภาคส่ง



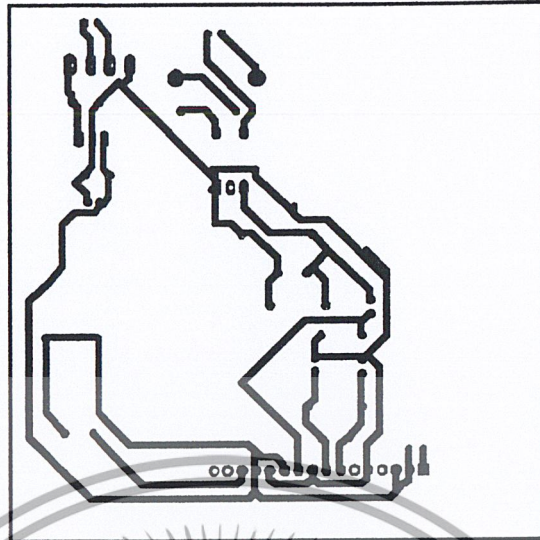
รูปการวางอุปกรณ์ของภาคส่ง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

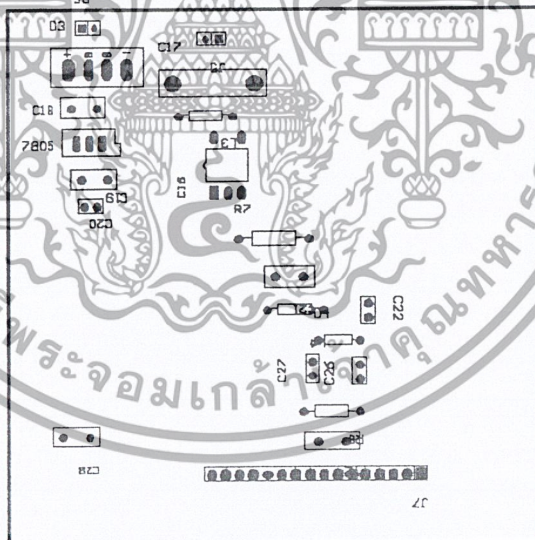


รูปวงจรของภาครับ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

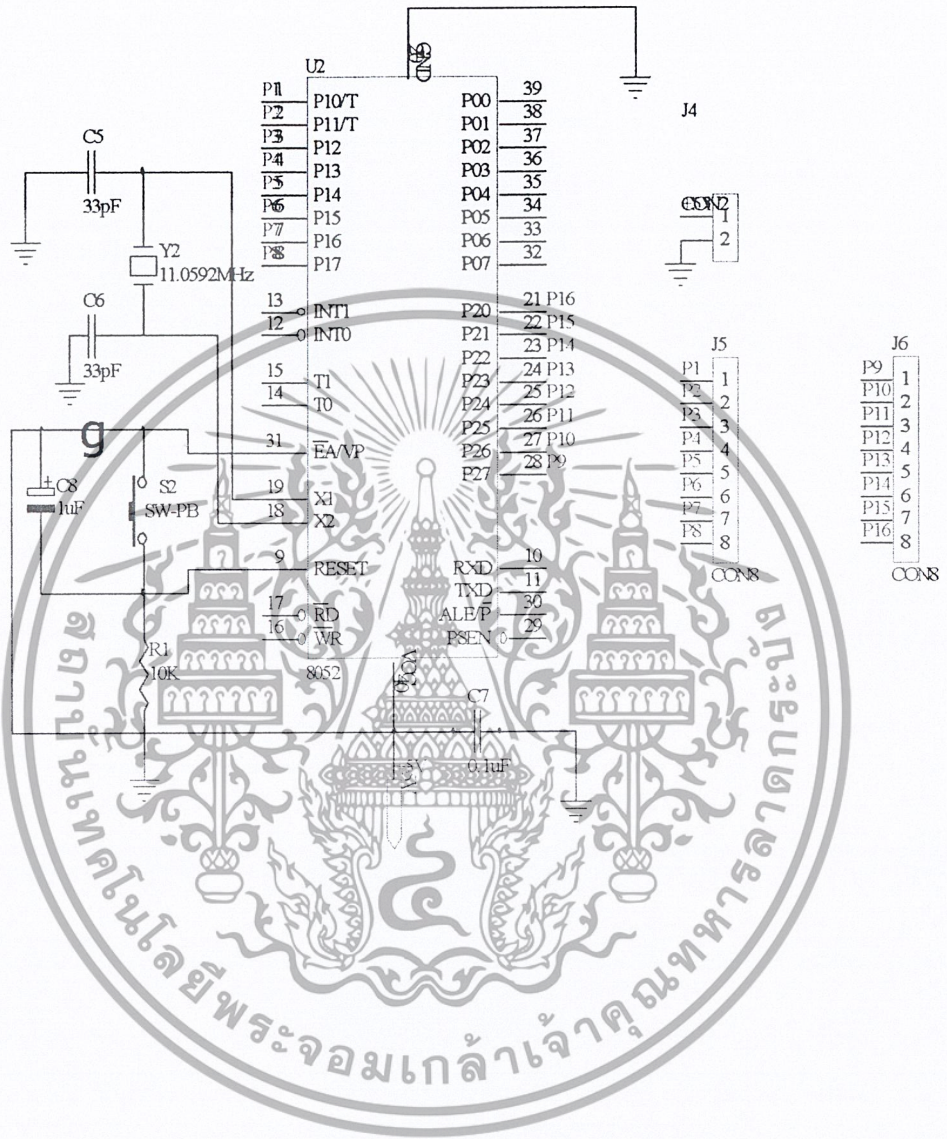


รูปลายวงจรของภากรับ



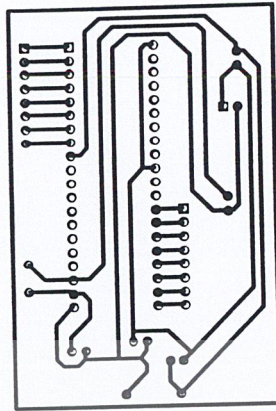
รูปวงจรการวางอุปกรณ์ของรับ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปร่างไมโครคอนโทรลเลอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

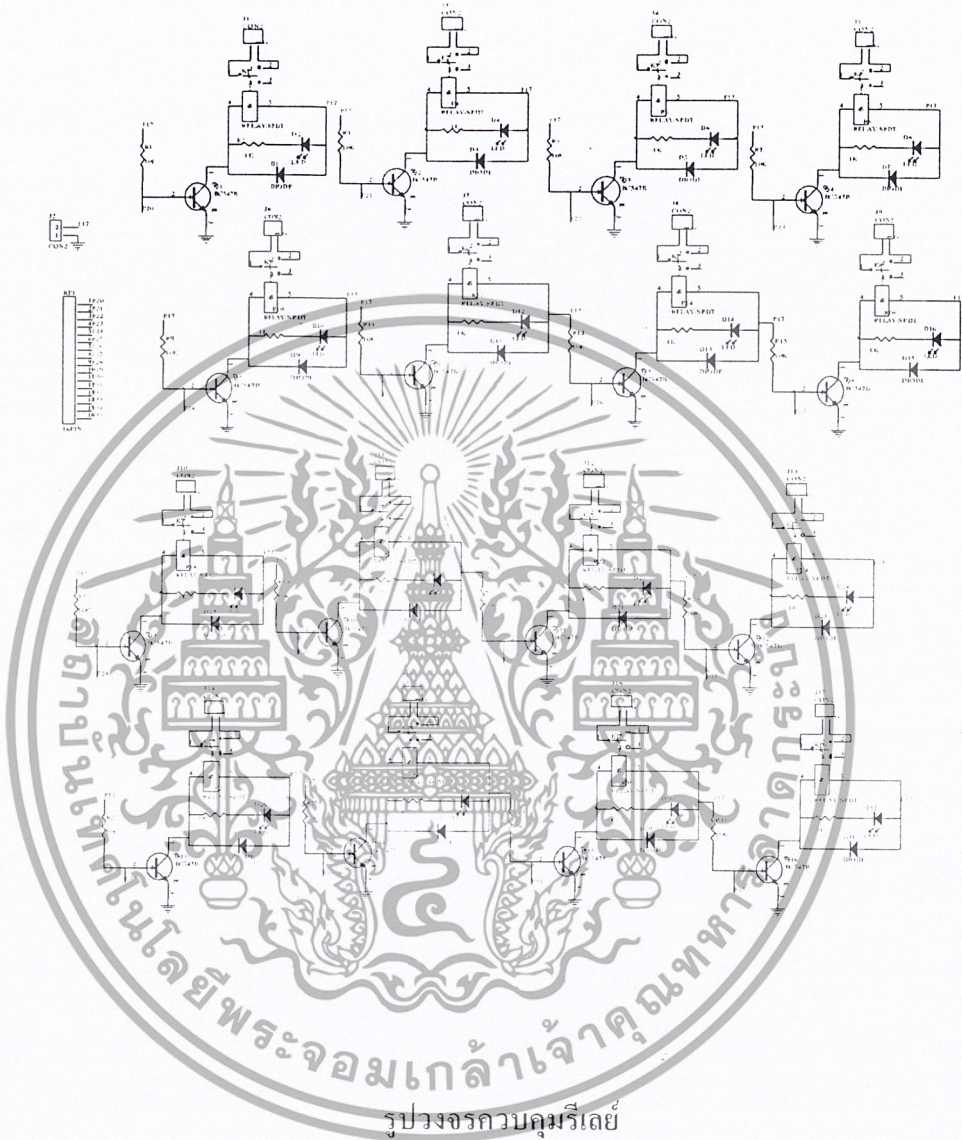


รูปลายวงจรไมโครคอนโทรลเลอร์

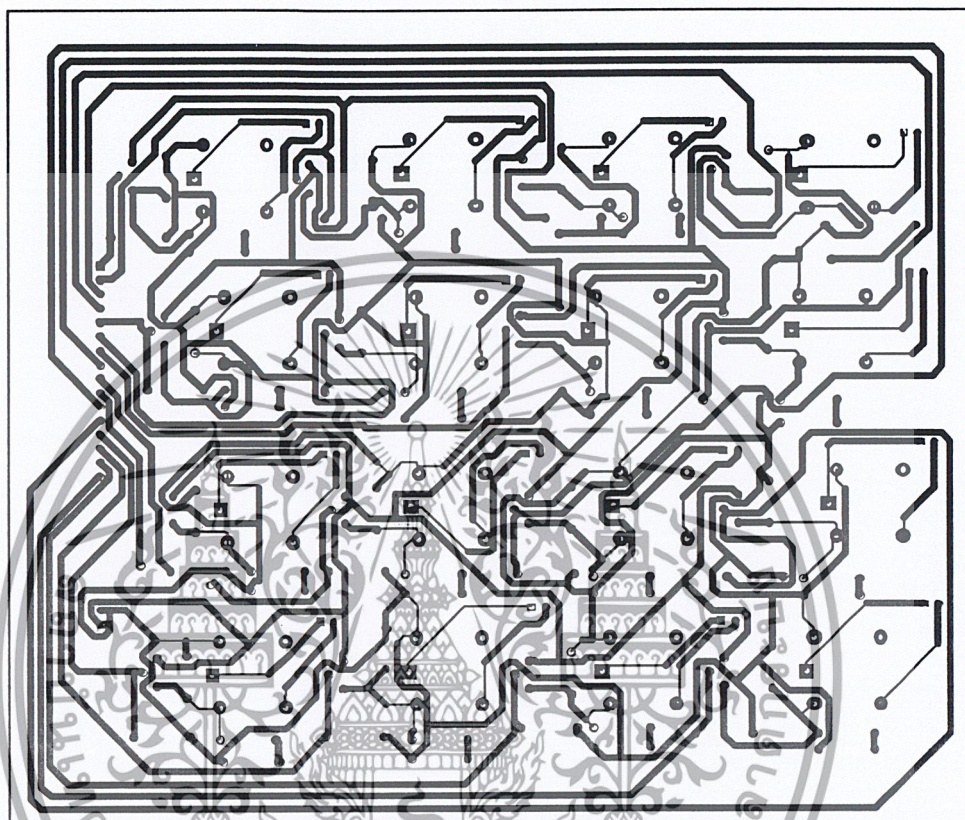


รูปการวางอุปกรณ์ของวงจรไมโครคอนโทรลเลอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

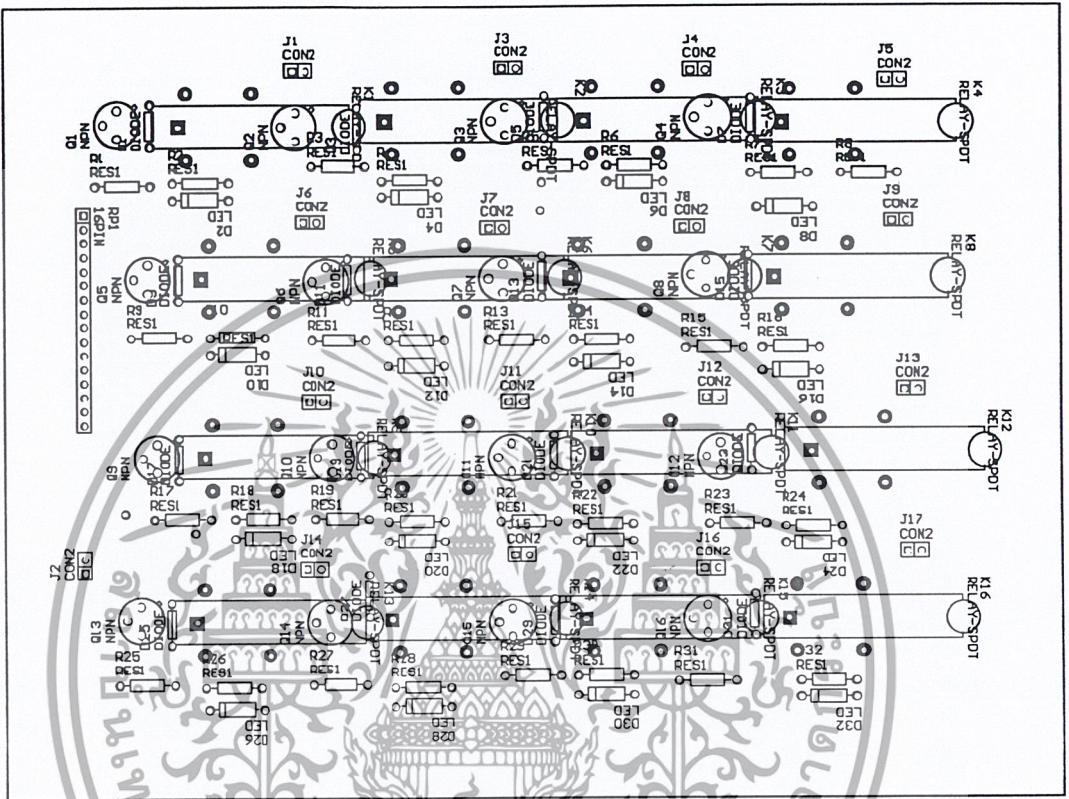


เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
 ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปสายทองแดงของวงจรควบคุมรีเลย์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

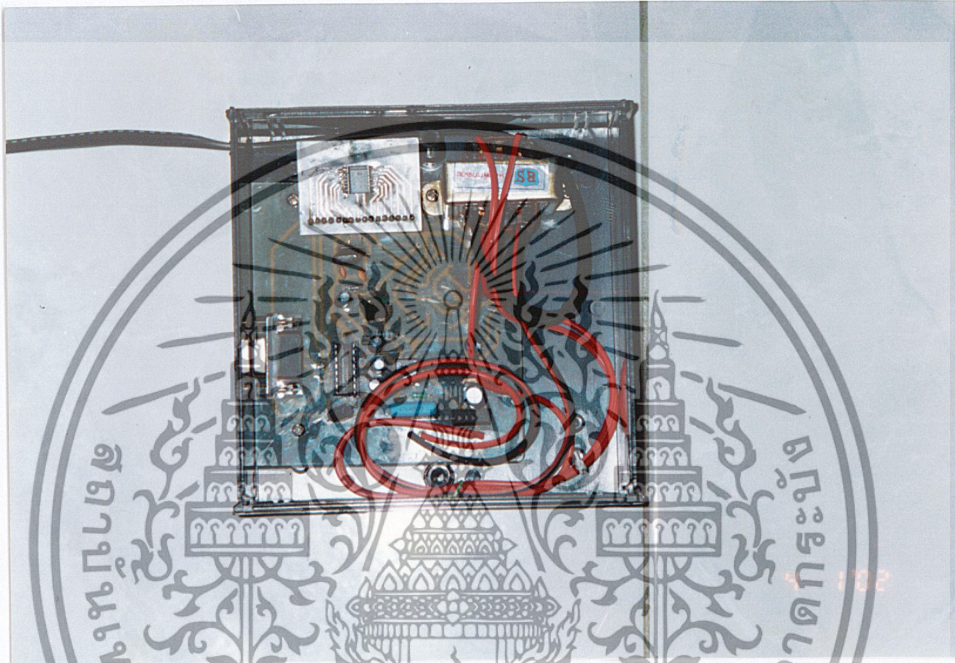


รูปการวางอุปกรณ์ของวงจรควบคุมรีเลย์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาพถ่ายโครงงาน

1.ภาพโครงงานในส่วนของภาคส่ง



รูปที่ 1

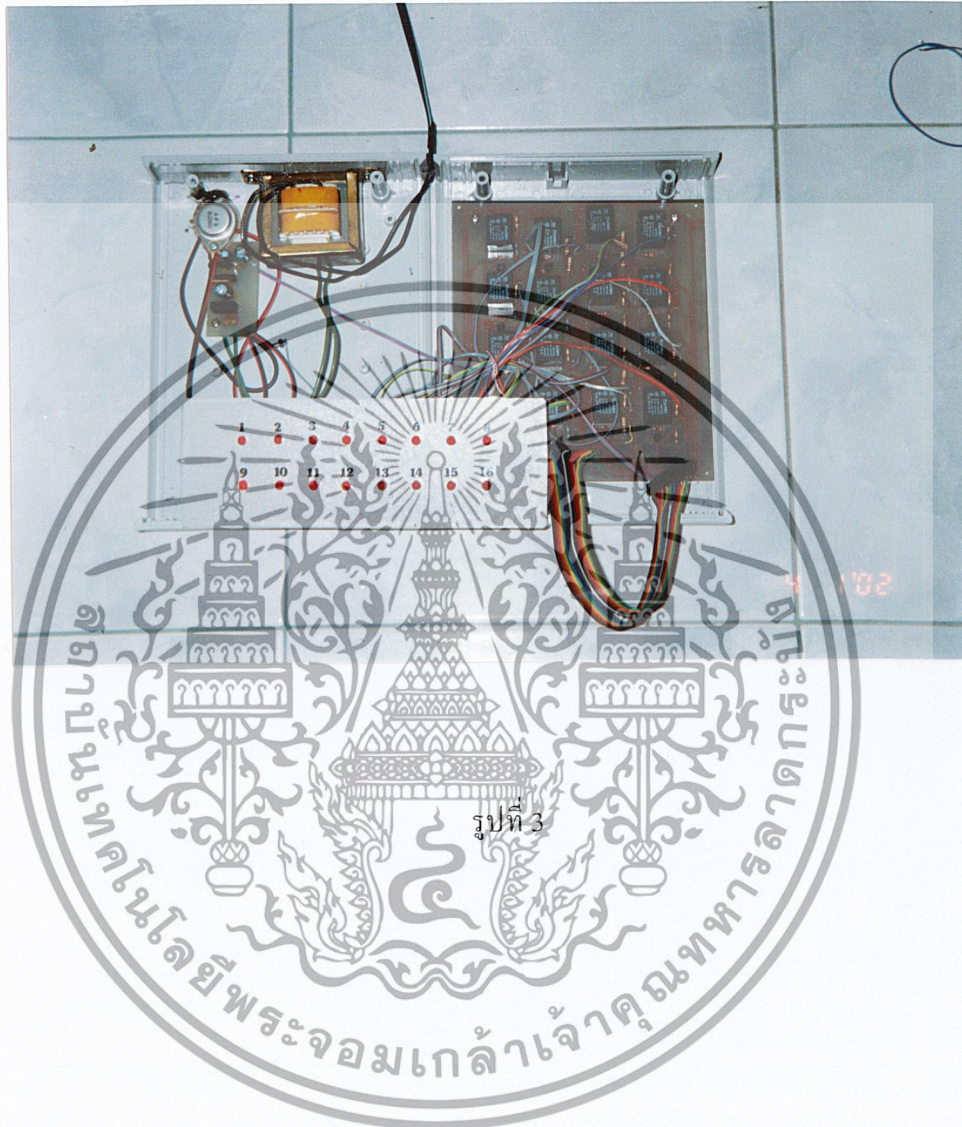
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.ภาพโครงงานในส่วนของภากรับ



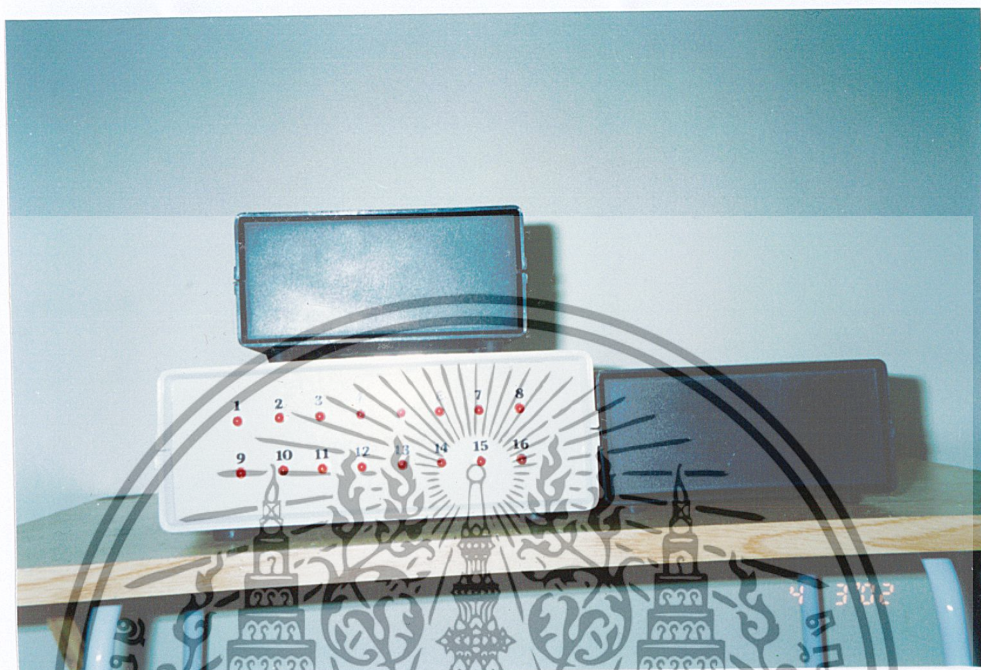
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.ภาพโครงงานของภาคควบคุมอุปกรณ์ (ON-OFF)



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.ภาพโครงการที่ประกอบสำเร็จพร้อมใช้งาน



รูปที่ 4

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

DATA SHEET



TDA5051A Home automation modem

Product specification
File under Integrated Circuits, IC11

1998 Jan 08

Philips
Semiconductors



PHILIPS

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Home automation modem

TDA5051A

FEATURES

- Full digital carrier generation and shaping
- Modulation/demodulation frequency set by clock adjustment, from microcontroller or on-chip oscillator
- High clock rate of 6 bits D/A (Digital to Analog) converter for rejection of aliasing components
- Fully integrated output power stage with overload protection
- Automatic gain control at receiver input
- 8-bit A/D and narrow digital filtering
- Digital demodulation delivering baseband data
- Easy compliance with EN50065-1 with simple coupling network
- Few external components for low cost applications
- SO16 plastic package.

APPLICATIONS

- Home appliance control (air conditioning, shutters, lighting, alarms and so on)
- Energy/heating control
- ASK (Amplitude Shift Keying) data transmission using the home power network.

GENERAL DESCRIPTION

The TDA5051A is a modem IC, specifically dedicated to ASK transmission by means of the home power supply network, at 600 or 1200 baud data rate. It operates from a single 5 V supply.

QUICK REFERENCE DATA

SYMBOL	PARAMETER	CONDITIONS	MIN.	TYP.	MAX.	UNIT
V_{DD}	supply voltage		4.75	5.0	5.25	V
$I_{DD(tot)}$	total supply current	$f_{osc} = 8.48$ MHz				
	reception mode			28	38	mA
	transmission mode ($DATA_{IN} = 0$)	$Z_L = 30 \Omega$		47	68	mA
	power down mode			19	25	mA
T_{amb}	operating ambient temperature		0		70	°C
f_{cr}	carrier frequency	note 1	95	132.5	148.5	kHz
f_{osc}	oscillator frequency		6.08	8.48	9.504	MHz
$V_{O(rms)}$	output carrier signal on CISPR16 load (RMS value)		120		122	dB μ V
$V_{I(rms)}$	input signal (RMS value)	note 2	82	–	122	dB μ V
THD	total harmonic distortion on CISPR16 load with coupling network			–55	–	dB
Z_L	load impedance		1	30	–	Ω
BR	baud rate			600	1200	bits/s

Notes

1. Frequency range corresponding to the EN50065-1 band. However the modem can operate at any lower oscillator frequency.
2. The minimum value can be improved by using an external amplifier (see application diagrams, Figs 22 and 23).

Home automation modem

TDA5051A

ORDERING INFORMATION

TYPE NUMBER	PACKAGE		
	NAME	DESCRIPTION	VERSION
TDA5051AT	SO16	plastic small outline package: 16 leads; body width 7.5 mm	SOT162-1



Home automation modem

TDA5051A

BLOCK DIAGRAM

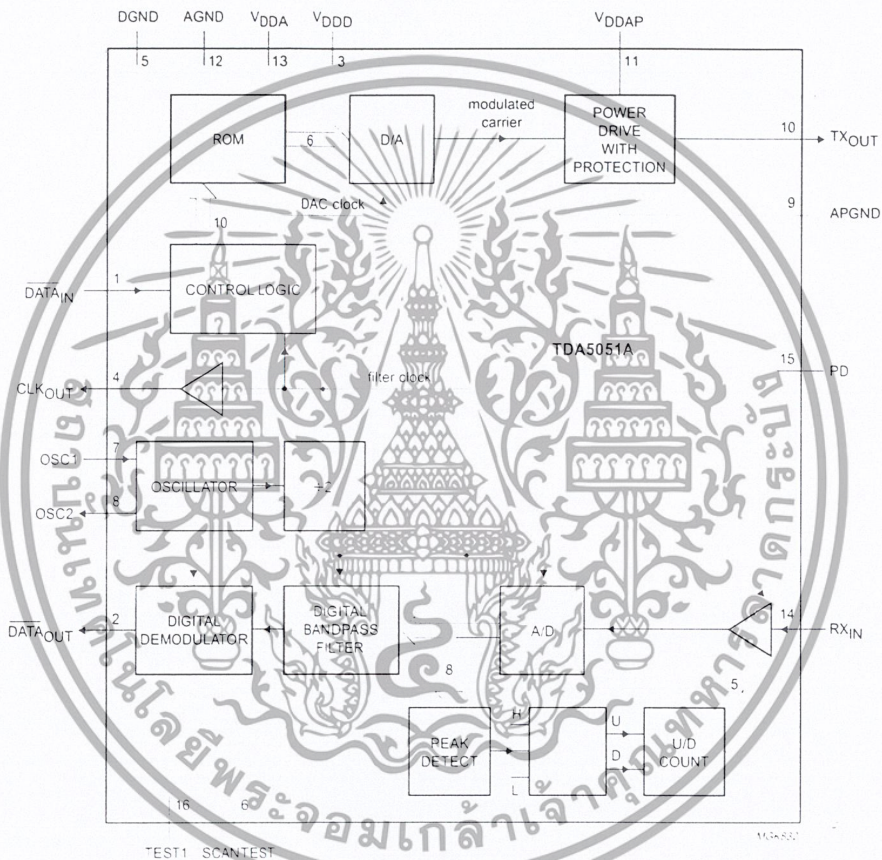


Fig.1 Block diagram

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Home automation modem

TDA5051A

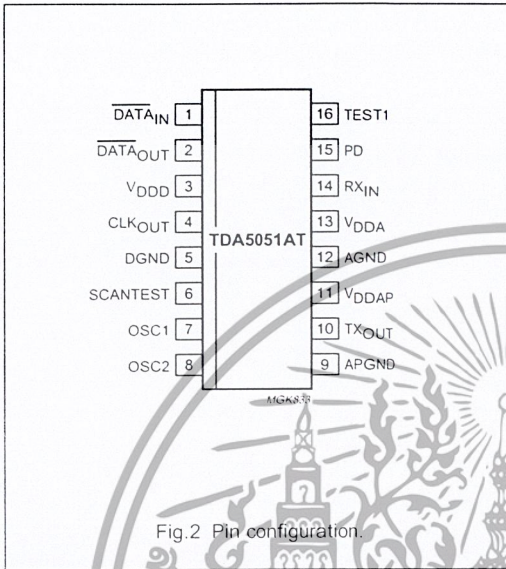


Fig.2 Pin configuration.

PINNING

SYMBOL	PIN	DESCRIPTION
DATA _{IN}	1	digital data input (active LOW)
DATA _{OUT}	2	digital data output (active LOW)
V _{DDD}	3	digital supply voltage
CLK _{OUT}	4	clock output
DGND	5	digital ground
SCANTEST	6	test input (LOW in application)
OSC1	7	oscillator input
OSC2	8	oscillator output
APGND	9	analog ground for power amplifier
TX _{OUT}	10	analog signal output
V _{DDAP}	11	analog supply voltage for power amplifier
AGND	12	analog ground
V _{DDA}	13	analog supply voltage
RX _{IN}	14	analog signal input
PD	15	power-down input (active HIGH)
TEST1	16	test input (HIGH in application)

Home automation modem

TDA5051A

FUNCTIONAL DESCRIPTION

Both transmission and reception stages are controlled either by the master clock of the microcontroller, or by the on-chip reference oscillator connected to a crystal. This holds for the accuracy of the transmission carrier and the exact trimming of the digital filter, thus making the performance totally independent of application disturbances such as component spread, temperature, supply drift and so on.

The interface with the power network is made by means of a LC network (see Fig.18). The device includes a power output stage able to feed a 120 dB μ V (RMS) signal on a typical 30 Ω load.

To reduce power consumption, the IC is disabled by a power-down input (pin PD): in this mode, the on-chip oscillator remains active and the clock continues to be supplied at pin CLK_{OUT}. For low-power operation in reception mode, this pin can be dynamically controlled by the microcontroller (see Section "Power-down mode").

When the circuit is connected to an external clock generator (see Fig.6), the clock signal must be applied at pin OSC1 (pin 7); OSC2 (pin 8) must be left open. Use of the on-chip clock circuitry is shown in Fig.7.

All logic inputs and outputs are compatible with TTL/CMOS levels, providing an easy connection to a standard microcontroller I/O port.

The digital part of the IC is fully scan-testable. Two digital inputs, SCANTEST and TEST1, are used for production test: these pins must be left open in functional mode (correct levels are internally defined by pull-up/down resistors).

Transmission mode

The carrier frequency is generated by the scanning of a ROM memory under the control of the microcontroller clock or the reference frequency provided by the on-chip oscillator, thus providing strict stability with respect to environmental conditions. High frequency clocking rejects the aliasing components to such an extent that they are filtered by the coupling LC network and do not cause any significant disturbance. The data modulation is applied through pin DATA_{IN}

and smoothly applied by specific digital circuitry to the carrier (shaping). Harmonic components are limited in this process, thus avoiding unacceptable disturbance of the transmission channel (according to CISPR16 and EN50065-1 recommendations). A -55 dB total harmonic distortion is reached when using the typical LC coupling network (or an equivalent filter).

The D/A converter and the power stage are set in order to provide a maximum signal level of 122 dB μ V (RMS) at the output.

The output of the power stage (TX_{OUT}) always has to be connected to a decoupling capacitor, because of a DC level of 0.5V_{DD} at this pin, present even when the device is not transmitting. This pin also has to be protected against overvoltage and negative transient signals. The DC level of TX_{OUT} can be used to bias an unipolar transient suppressor, as shown in the application diagram (see Fig.18).

Direct connection to the mains is done through a LC network for low-cost applications. However, a HF signal transformer could be used when power-line insulation has to be performed.

CAUTION

In transmission mode, the receiving part of the circuit is **not disabled** and the detection of the transmitted signal is normally performed. In this mode, the gain chosen before the beginning of the transmission is stored, and the **AGC is internally set to -6 dB** as long as DATA_{IN} is LOW. Then, the old gain setting is **automatically restored**.

Home automation modem

TDA5051A

Receiving mode

The input signal received by the modem is applied to a wide range input amplifier with Automatic Gain Control (AGC) (-6 to +30 dB). This is basically for noise performance improvement and signal level adjustment that ensures a maximum sensitivity of the A/D converter. Then an 8-bit A/D conversion is performed, followed by digital bandpass filtering, in order to meet the CISPR normalization and to comply with some additional limitations encountered in current applications. After digital demodulation, the baseband data signal is made available after pulse shaping.

The signal pin (RX_{IN}) is a high-impedance input, which has to be protected and DC decoupled for the same reasons as with pin TX_{OUT}. The high sensitivity (82 dB μ V) of this input requires an efficient 50 Hz rejection filter (realized by the LC coupling network) also used as an anti-aliasing filter for the internal digital processing (see Fig. 18).



Home automation modem

TDA5051A

Data format

TRANSMISSION MODE

The data input ($\overline{\text{DATA}}_{\text{IN}}$) is active LOW: this means that a burst is generated on the line (pin TX_{OUT}) when pin $\overline{\text{DATA}}_{\text{IN}}$ is LOW.

Pin TX_{OUT} is in high-impedance state as long as the device is not transmitting. Successive logic 1s are treated in a NRZ mode (see pulse shape description in Figs 8 and 9).

RECEIVING MODE

The data output (pin $\overline{\text{DATA}}_{\text{OUT}}$) is active LOW: this means that the data output is LOW when a burst is received. Pin $\overline{\text{DATA}}_{\text{OUT}}$ remains LOW as long as a burst is received.

Power-down mode

Power-down input (pin PD) is active HIGH: this means that the power consumption is minimal when pin PD is HIGH. All functions, except clock generation, are disabled then.

LIMITING VALUES

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

SYMBOL	PARAMETER	MIN.	MAX.	UNIT
V _{DD}	supply voltage	4.5	5.5	V
f _{osc}	oscillator frequency		12	MHz
T _{stg}	storage temperature	-50	+150	°C
T _{amb}	operating ambient temperature	+10	+80	°C
T _j	junction temperature		125	°C

HANDLING

Inputs and outputs are protected against electrostatic discharge in normal handling. However, to be totally safe, it is desirable to take normal precautions appropriate to handling MOS devices.

Home automation modem

TDA5051A

CHARACTERISTICS

$V_{DD} = V_{DDA} = 5\text{ V} \pm 5\%$; $T_{amb} = 0\text{ to }+70\text{ }^{\circ}\text{C}$; V_{DD} connected to V_{DDA} ; DGND connected to AGND.

SYMBOL	PARAMETER	CONDITIONS	MIN.	TYP.	MAX.	UNIT
Supply						
V_{DD}	supply voltage		4.75	5	5.25	V
$I_{DD(RX/TX)(tot)}$	total analog + digital supply current; TX or RX mode	$V_{DD} = 5\text{ V} \pm 5\%$	–	28	38	mA
$I_{DD(PD)(tot)}$	total analog + digital supply current; power-down mode	$V_{DD} = 5\text{ V} \pm 5\%$; PD = HIGH	–	19	25	mA
$I_{DD(PAMP)}$	power amplifier supply current in transmission mode	$V_{DD} = 5\text{ V} \pm 5\%$; $Z_L = 30\ \Omega$; $DATA_{IN} = \text{LOW}$	–	19	30	mA
$I_{DD(PAMP)(max)}$	maximum power amplifier supply current in transmission mode	$V_{DD} = 5\text{ V} \pm 5\%$; $Z_L = 1\ \Omega$; $DATA_{IN} = \text{LOW}$	–	76	–	mA
$DATA_{IN}$ input, PD input; $DATA_{OUT}$ output, CLK_{OUT} output						
V_{IH}	HIGH-level input voltage		$0.2V_{DD} + 0.9$	–	$V_{DD} + 0.5$	V
V_{IL}	LOW-level input voltage		–0.5	–	$0.2V_{DD} - 0.1$	V
V_{OH}	HIGH-level output voltage	$I_{OH} = -1.6\text{ mA}$	2.4	–	–	V
V_{OL}	LOW-level output voltage	$I_{OL} = 1.6\text{ mA}$	–	–	0.45	V
OSC1 input and OSC2 output (OSC2 only used for driving external quartz crystal; must be left open when using an external clock generator)						
V_{IH}	HIGH-level input voltage		$0.7V_{DD}$	–	$V_{DD} + 0.5$	V
V_{IL}	LOW-level input voltage		–0.5	–	$0.2V_{DD} - 0.1$	V
V_{OH}	HIGH-level output voltage	$I_{OH} = -1.6\text{ mA}$	2.4	–	–	V
V_{OL}	LOW-level output voltage	$I_{OL} = 1.6\text{ mA}$	–	–	0.45	V
Clock						
f_{osc}	oscillator frequency		6.080	–	9.504	MHz
$\frac{f_{osc}}{f_{cr}}$	ratio between oscillator and carrier frequency		–	64	–	
$\frac{f_{osc}}{f_{CLKOUT}}$	ratio between oscillator and clock output frequency		–	2	–	
Transmission mode						
f_{cr}	carrier frequency	$f_{osc} = 8.48\text{ MHz}$	–	132.5	–	kHz
t_{su}	set-up time of the shaped burst	$f_{osc} = 8.48\text{ MHz}$; see Fig.8	–	170	–	μs
t_h	hold time of the shaped burst	$f_{osc} = 8.48\text{ MHz}$; see Fig.8	–	170	–	μs

Home automation modem

TDA5051A

SYMBOL	PARAMETER	CONDITIONS	MIN.	TYP.	MAX.	UNIT
$t_{W(DI)(min)}$	minimum pulse width of \overline{DATA}_{IN} signal	$f_{osc} = 8.48$ MHz; see Fig.8	–	190	–	μ s
$V_{o(rms)}$	output carrier signal (RMS value)	$Z_L =$ CISPR16 $\overline{DATA}_{IN} =$ LOW	120	–	122	$\text{dB}\mu\text{V}$
$I_{o(max)}$	power amplifier maximum output current (peak value)	$Z_L = 1 \Omega$; $\overline{DATA}_{IN} =$ LOW	–	160	–	mA
Z_o	output impedance of the power amplifier		–	5	–	Ω
V_o	output DC level at TX_{OUT}		–	2.5	–	V
THD	total harmonic distortion on CISPR16 load with the coupling network (measured on the first ten harmonics)	$V_{o(rms)} = 121$ $\text{dB}\mu\text{V}$ on CISPR16 load; $f_{osc} = 8.48$ MHz; $\overline{DATA}_{IN} =$ LOW (no modulation); see Figs 3 and 16	–	–55	–	dB
B_{-20dB}	bandwidth of the shaped output signal (at –20 dB) on CISPR16 load with the coupling network	$V_{o(rms)} = 121$ $\text{dB}\mu\text{V}$ on CISPR16 load; $f_{osc} = 8.48$ MHz; $\overline{DATA}_{IN} = 300$ Hz; duty factor = 50%; see Fig.4	–	3000	–	Hz
Reception mode						
$V_{i(rms)}$	analog input signal (RMS value)		82	–	122	$\text{dB}\mu\text{V}$
V_i	DC level at pin RX_{IN}		–	2.5	–	V
Z_i	RX_{IN} input impedance		–	50	–	$\text{k}\Omega$
R_{AGC}	automatic gain control range		–	36	–	dB
$t_{c(AGC)}$	automatic gain control time constant	$f_{osc} = 8.48$ MHz; see Fig.5	–	296	–	μ s
$t_{d(dem)(su)}$	demodulation delay set-up time	$f_{osc} = 8.48$ MHz; see Fig.15	–	350	400	μ s
$t_{d(dem)(h)}$	demodulation delay hold time	$f_{osc} = 8.48$ MHz; see Fig.15	–	420	470	μ s
B_{det}	detection bandwidth	$f_{osc} = 8.48$ MHz	–	3	–	kHz
BER	bit error rate	$f_{osc} = 8.48$ MHz; 600 baud; S/N = 35 dB; signal 76 $\text{dB}\mu\text{V}$; see Fig.17	–	1	–	1×10^{-4}

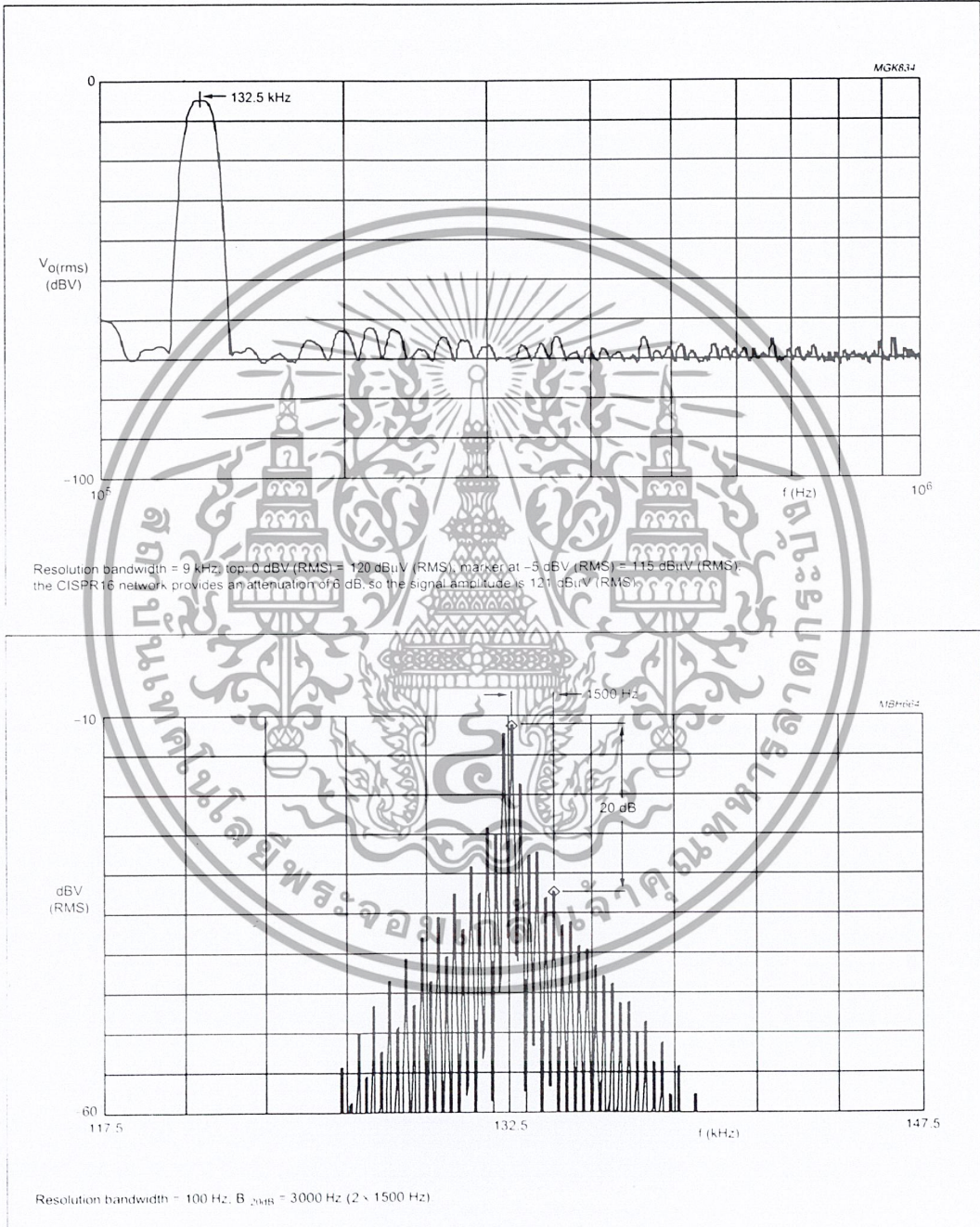
Home automation modem

TDA5051A

SYMBOL	PARAMETER	CONDITIONS	MIN.	TYP.	MAX.	UNIT
Power-up timing						
$t_{d(pu)(TX)}$	delay between power-up and \overline{DATA}_{IN} in transmission mode	XTAL = 8.48 MHz; C1 = C2 = 27 pF; $R_p = 2.2\text{ M}\Omega$; see Fig.10	–	1	–	μs
$t_{d(pu)(RX)}$	delay between power-up and \overline{DATA}_{OUT} in reception mode	XTAL = 8.48 MHz; C1 = C2 = 27 pF; $R_p = 2.2\text{ M}\Omega$; $f_{RXIN} = 132.5\text{ kHz}$; 120 dB μV sinewave; see Fig.11	–	1	–	μs
Power-down timing						
$t_{d(pd)(TX)}$	delay between PD = 0 and \overline{DATA}_{IN} in transmission mode	$f_{osc} = 8.48\text{ MHz}$; see Fig.12	–	10	–	μs
$t_{d(pd)(RX)}$	delay between PD = 0 and \overline{DATA}_{OUT} in reception mode	$f_{osc} = 8.48\text{ MHz}$; $f_{RXIN} = 132.5\text{ kHz}$; 120 dB μV sinewave; see Fig.13	–	500	–	μs
$t_{active(min)}$	minimum active time with $T = 10\text{ ms}$ power-down period in reception mode	$f_{osc} = 8.48\text{ MHz}$; $f_{RXIN} = 132.5\text{ kHz}$; 120 dB μV sinewave; see Fig.14	–	1	–	μs

Home automation modem

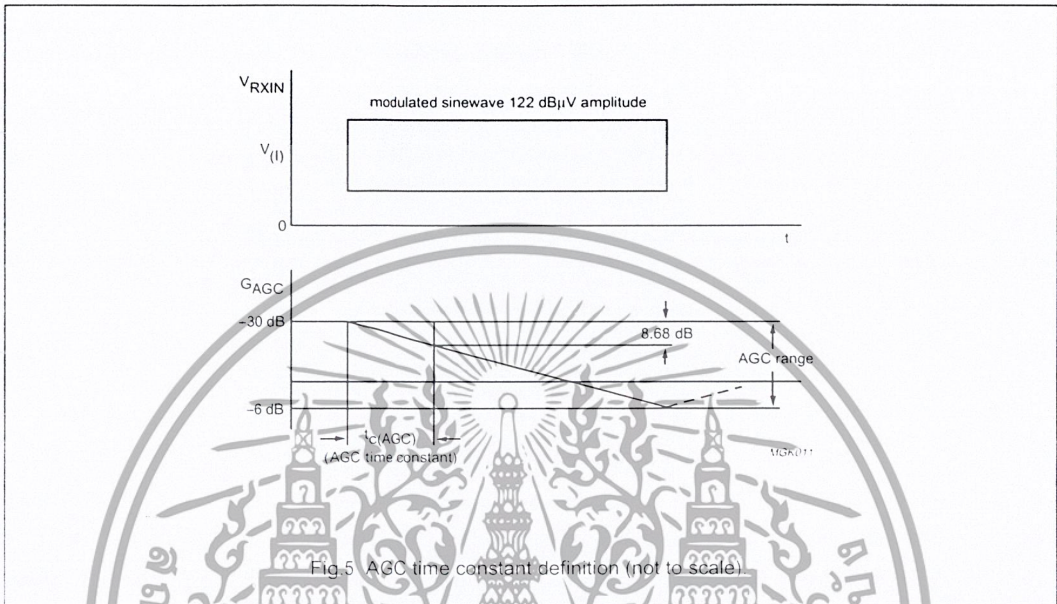
TDA5051A



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Home automation modem

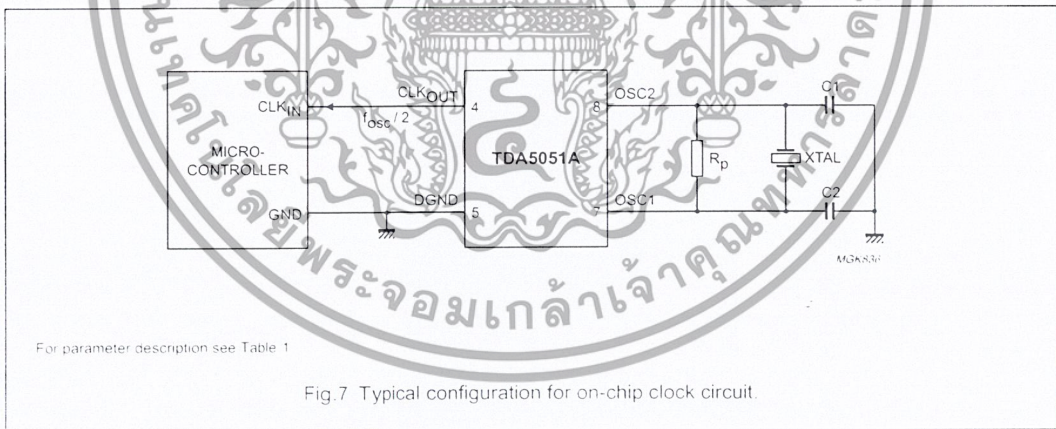
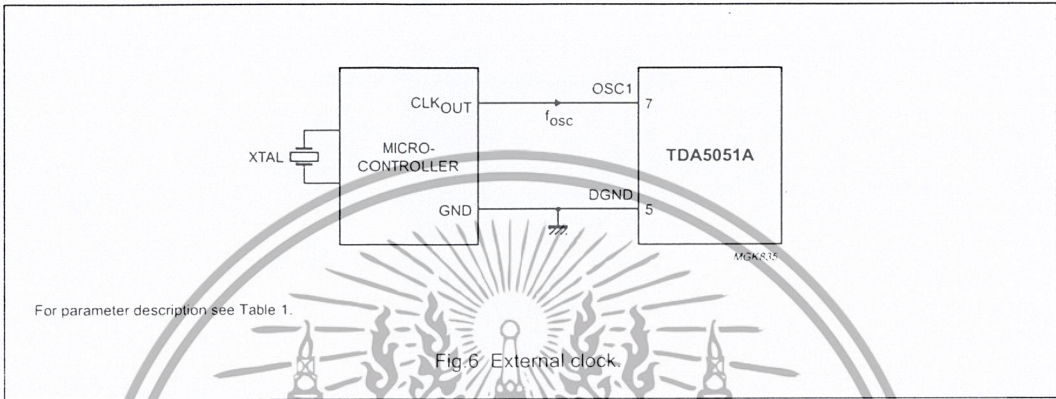
TDA5051A



Home automation modem

TDA5051A

configuration for clock



Home automation modem

TDA5051A

Table 1 Clock oscillator parameters

f_{osc} OSCILLATOR FREQUENCY	f_{cr} CARRIER FREQUENCY	$\frac{1}{2}f_{osc}$ CLOCK OUTPUT FREQUENCY	EXTERNAL COMPONENTS
6.080 to 9.504 MHz	95 to 148.5 kHz	3.040 to 4.752 MHz	C1 = C2 = 27 to 47 pF; R _p = 2.2 to 4.7 M Ω ; XTAL = standard quartz crystal

Table 2 Calculation of parameters depending on the clock frequency

SYMBOL	PARAMETER	CONDITIONS	UNIT
f_{osc}	oscillator frequency	with on-chip oscillator: frequency of the crystal quartz; with external clock: frequency of the signal applied at OSC1	Hz
f_{CLKOUT}	clock output frequency	$\frac{1}{2}f_{osc}$	Hz
f_{cr}	carrier frequency/digital filter tuning frequency	$\frac{1}{64}f_{osc}$	Hz
t_{su}	set-up time of the shaped burst	$\frac{23}{f_{cr}}$ or $\frac{1472}{f_{osc}}$	s
t_h	hold time of the shaped burst	$\frac{23}{f_{cr}}$ or $\frac{1472}{f_{osc}}$	s
$t_{W(DI)(min)}$	minimum pulse width of DATA _{IN} signal	$t_{su} + \frac{1}{f_{cr}}$	s
$t_{W(burst)(min)}$	minimum burst time of V _{ORC} signal	$t_{W(DI)(min)} + t_r$	s
$t_{c(AGC)}$	AGC time constant	$\frac{2514}{f_{osc}}$	s
$t_{su(demod)}$	demodulation set-up time	$\frac{3200}{f_{osc}}$ (=max.)	s
$t_{h(demod)}$	demodulation hold time	$\frac{3800}{f_{osc}}$ (=max.)	s

Home automation modem

TDA5051A

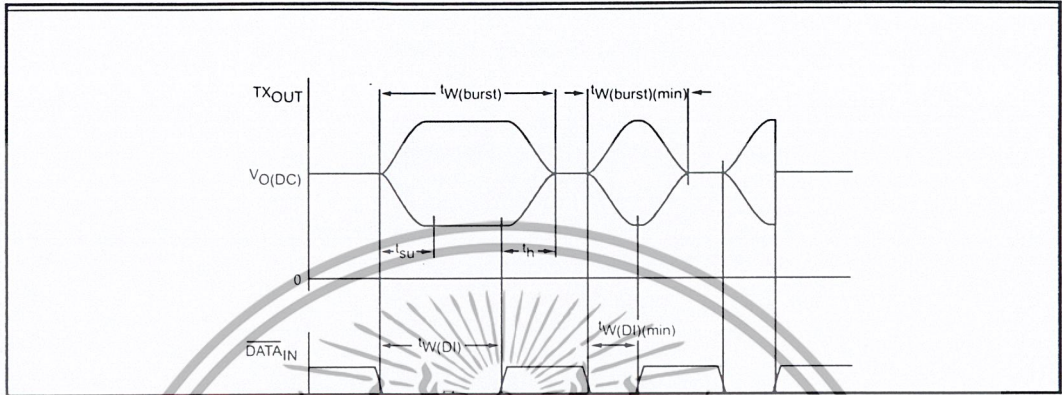


Table 3 Relationship between DATA_{IN} and TX_{OUT}

PD	DATA _{IN}	TX _{OUT}
1	X ¹⁾	high impedance
0	1	high impedance (after t _h)
0	0	active with DC offset

Note

1. X = don't care

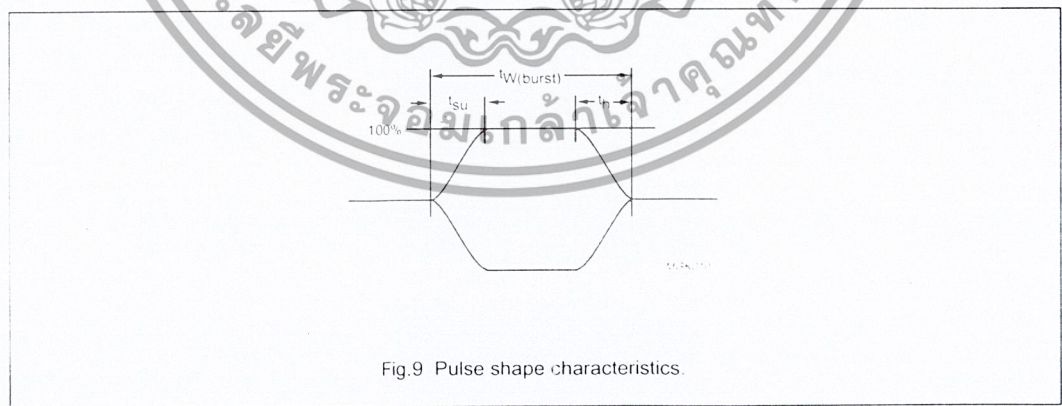


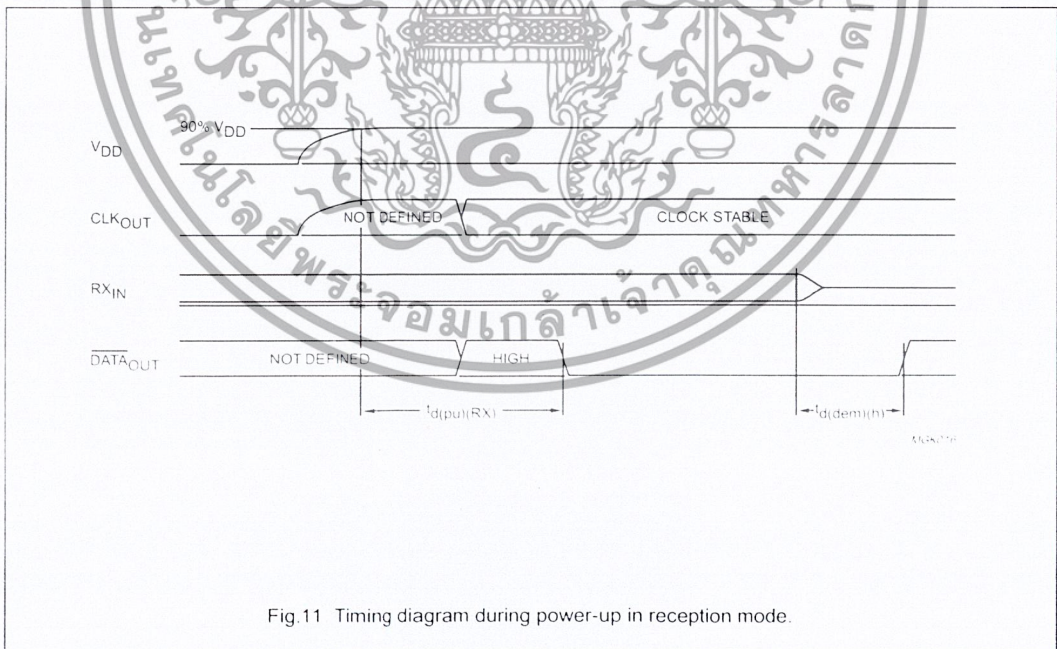
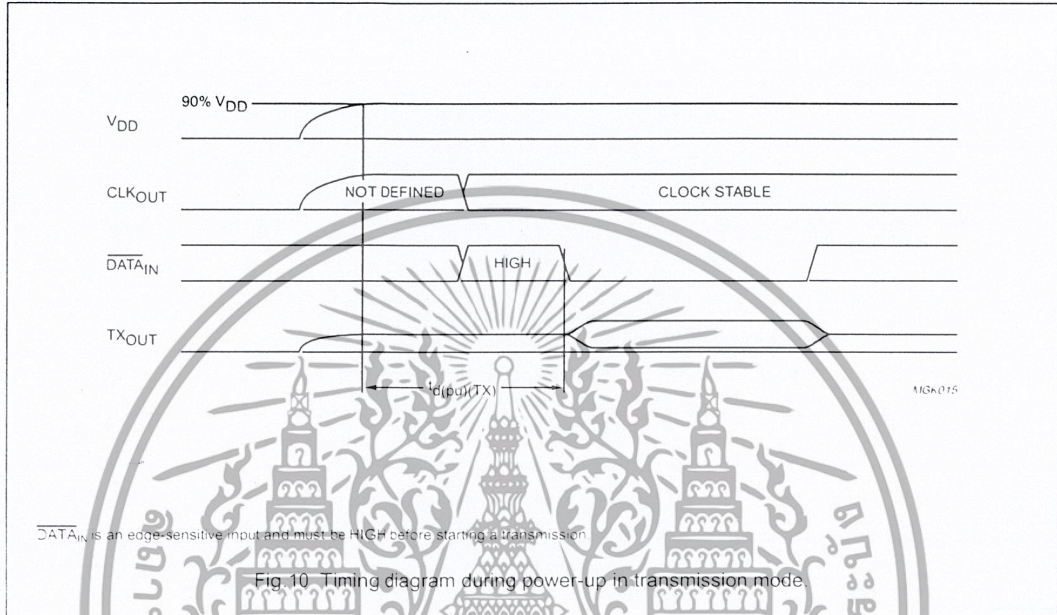
Fig.9 Pulse shape characteristics

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Home automation modem

TDA5051A

Timing diagrams



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Home automation modem

TDA5051A

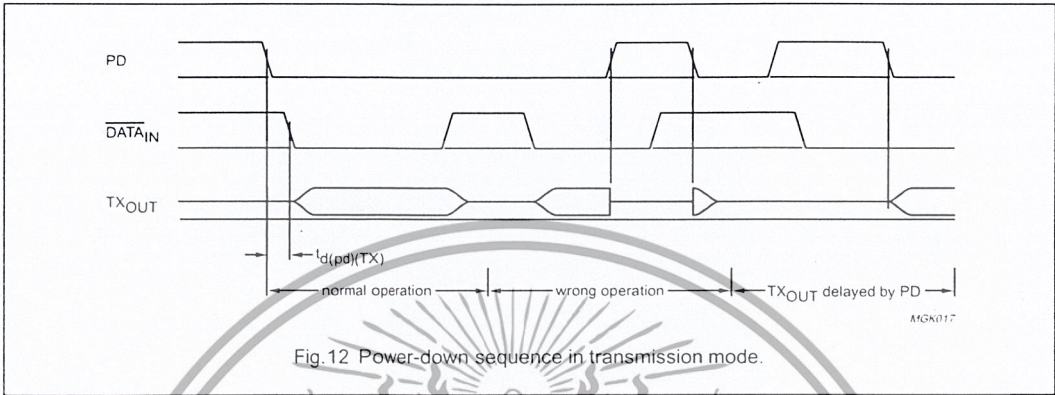


Fig.12 Power-down sequence in transmission mode.

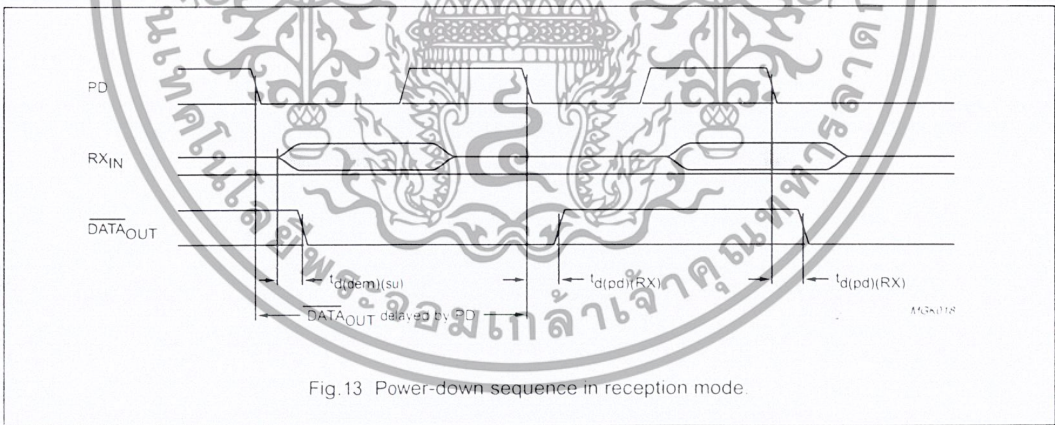


Fig.13 Power-down sequence in reception mode

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Home automation modem

TDA5051A

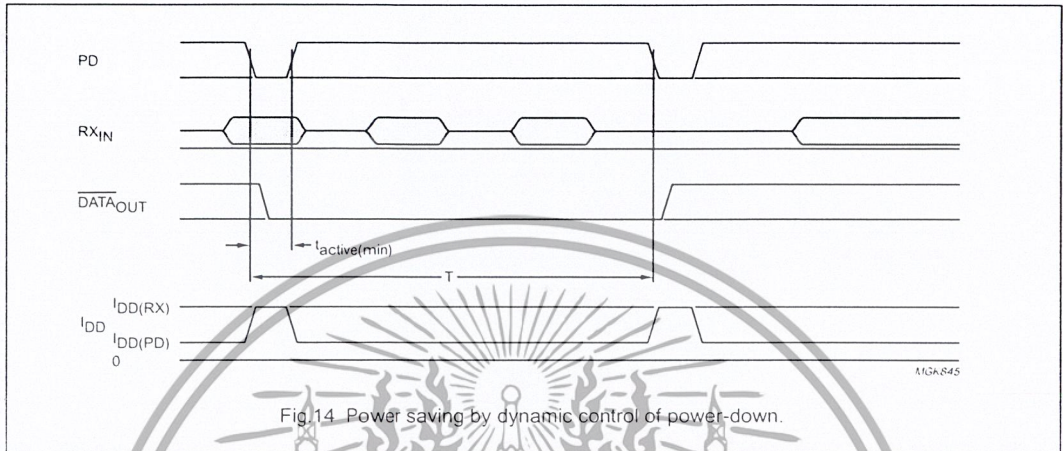


Fig.14 Power saving by dynamic control of power-down.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Home automation modem

TDA5051A

TEST INFORMATION

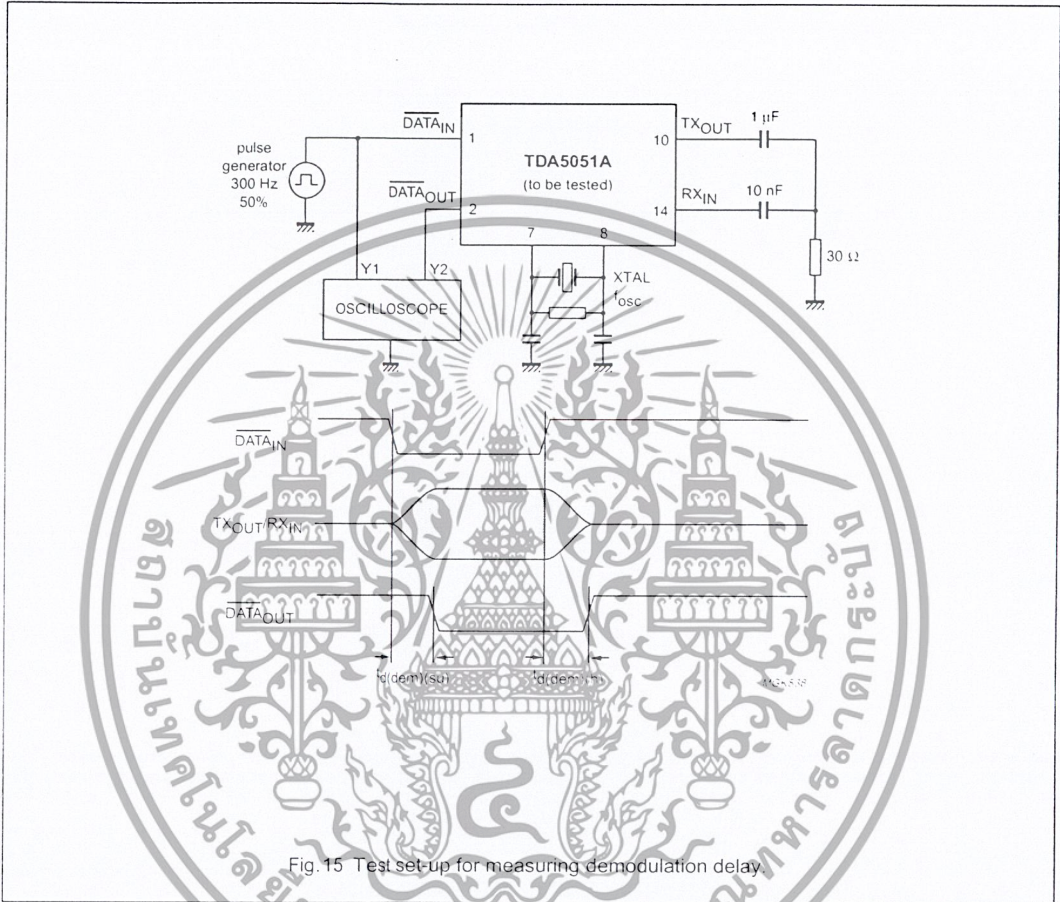


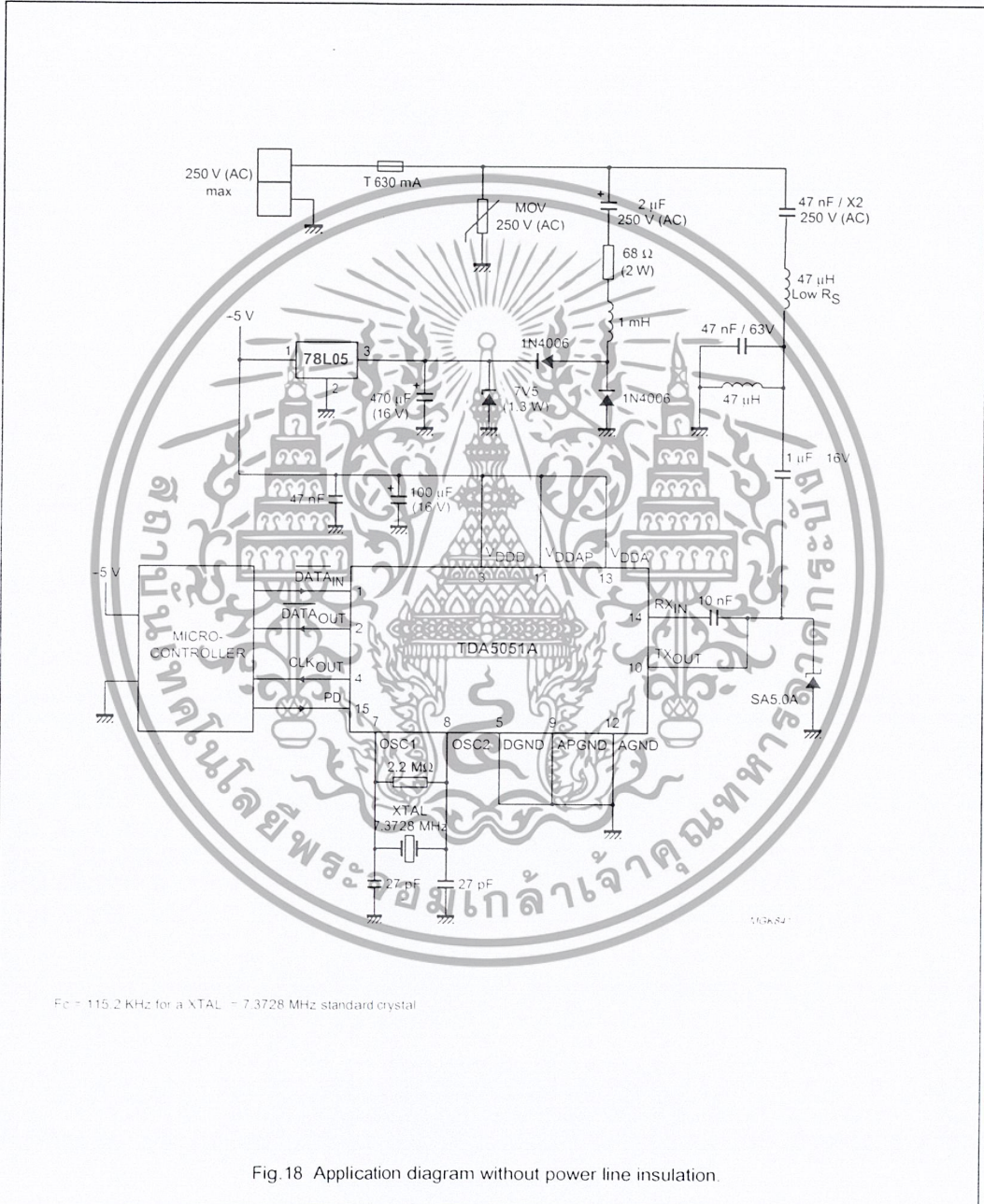
Fig. 15 Test set-up for measuring demodulation delay

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Home automation modem

TDA5051A

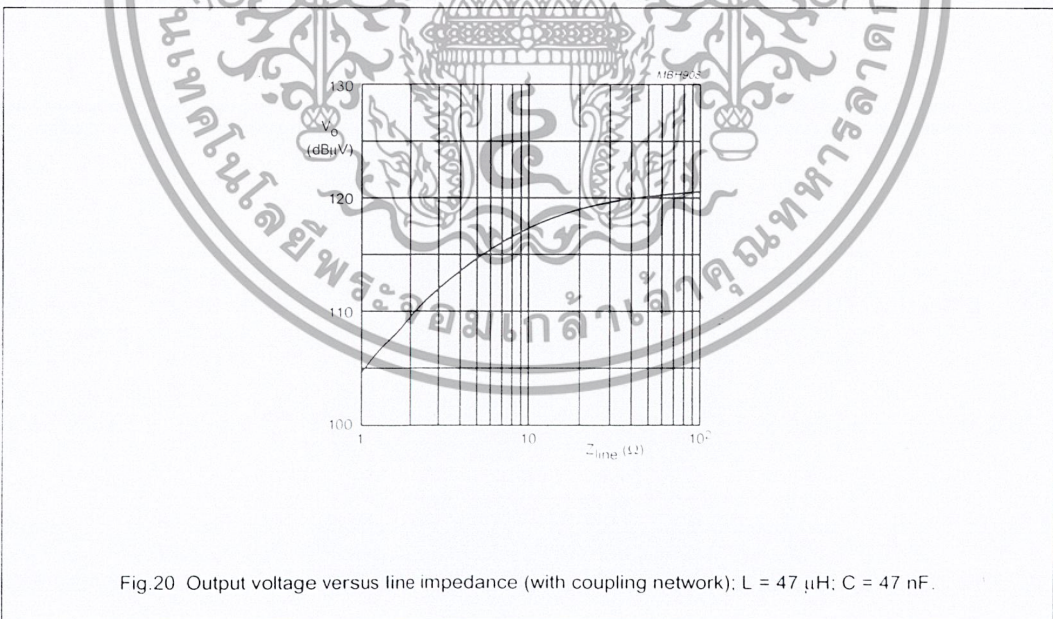
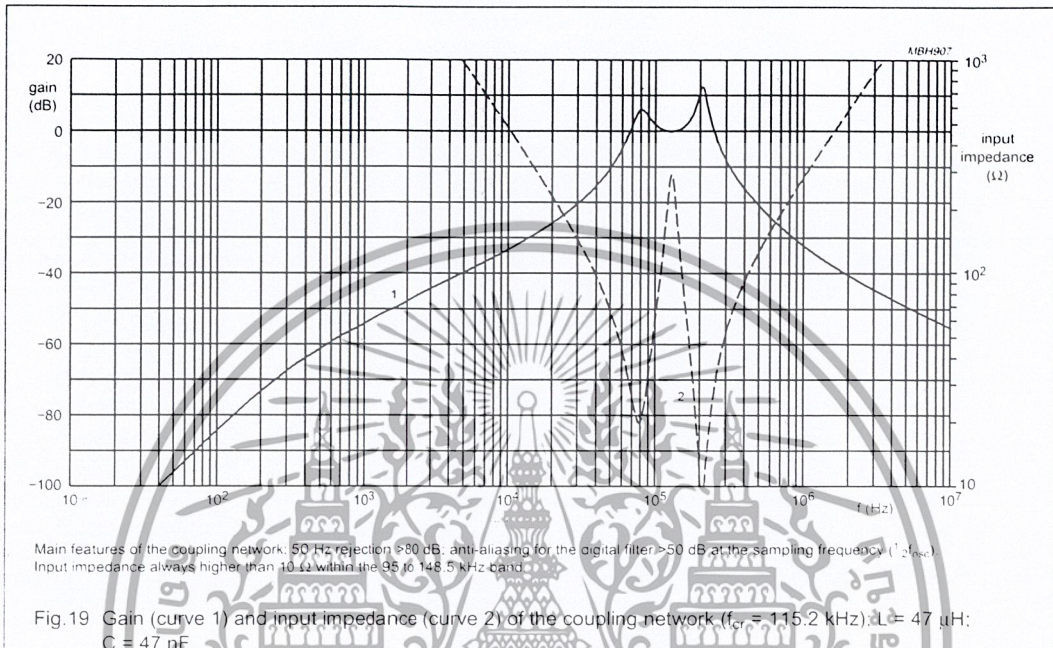
APPLICATION INFORMATION



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Home automation modem

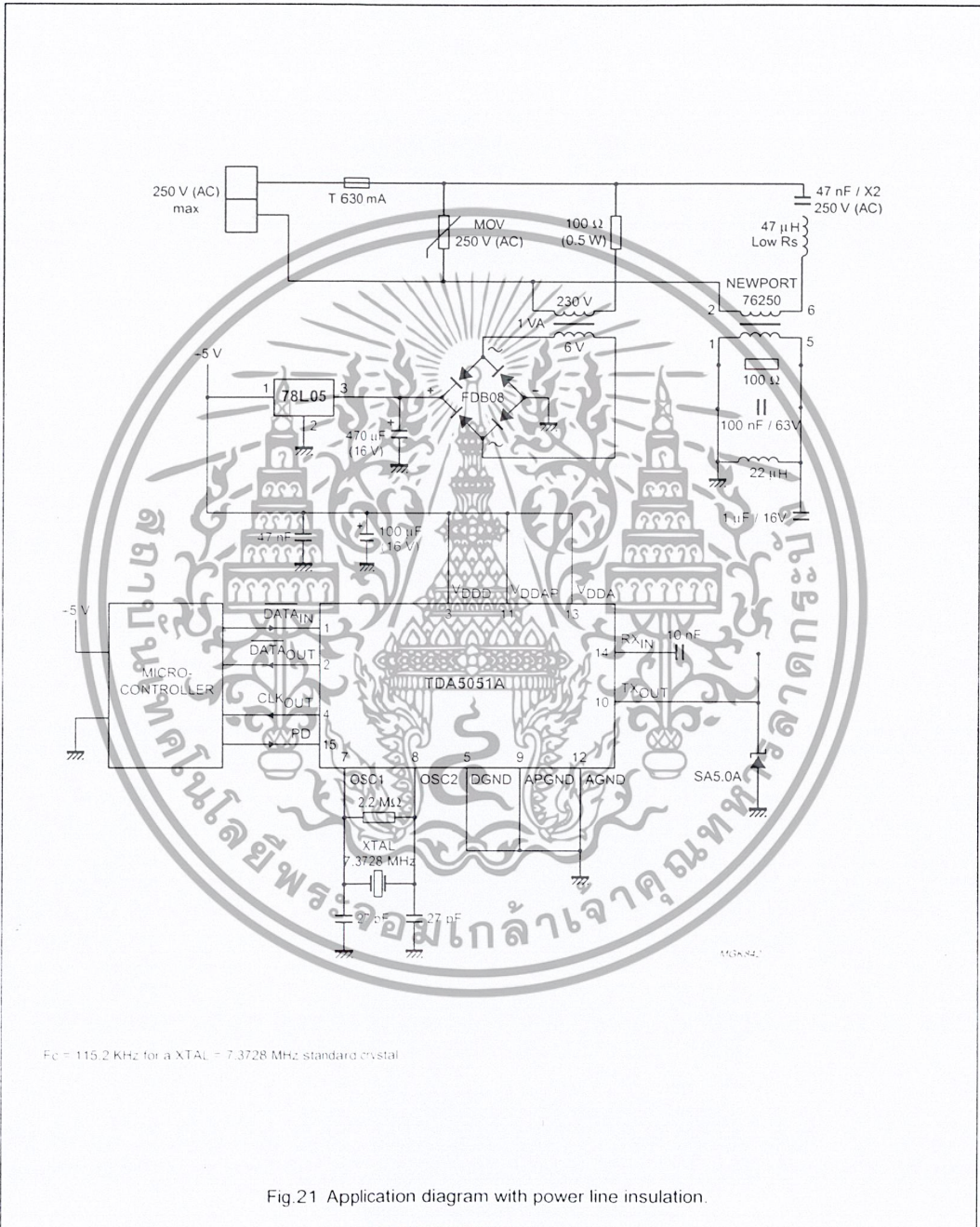
TDA5051A



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Home automation modem

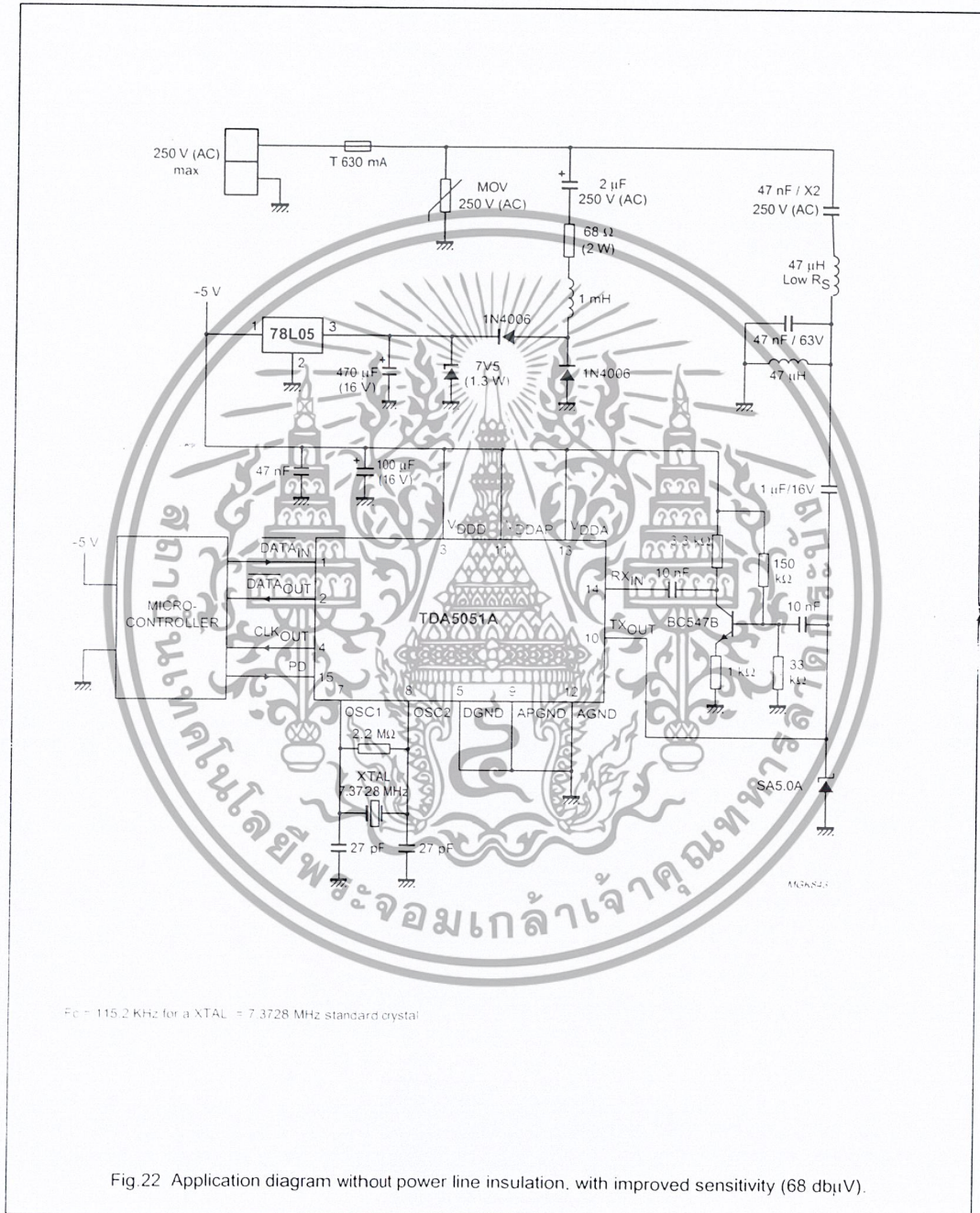
TDA5051A



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Home automation modem

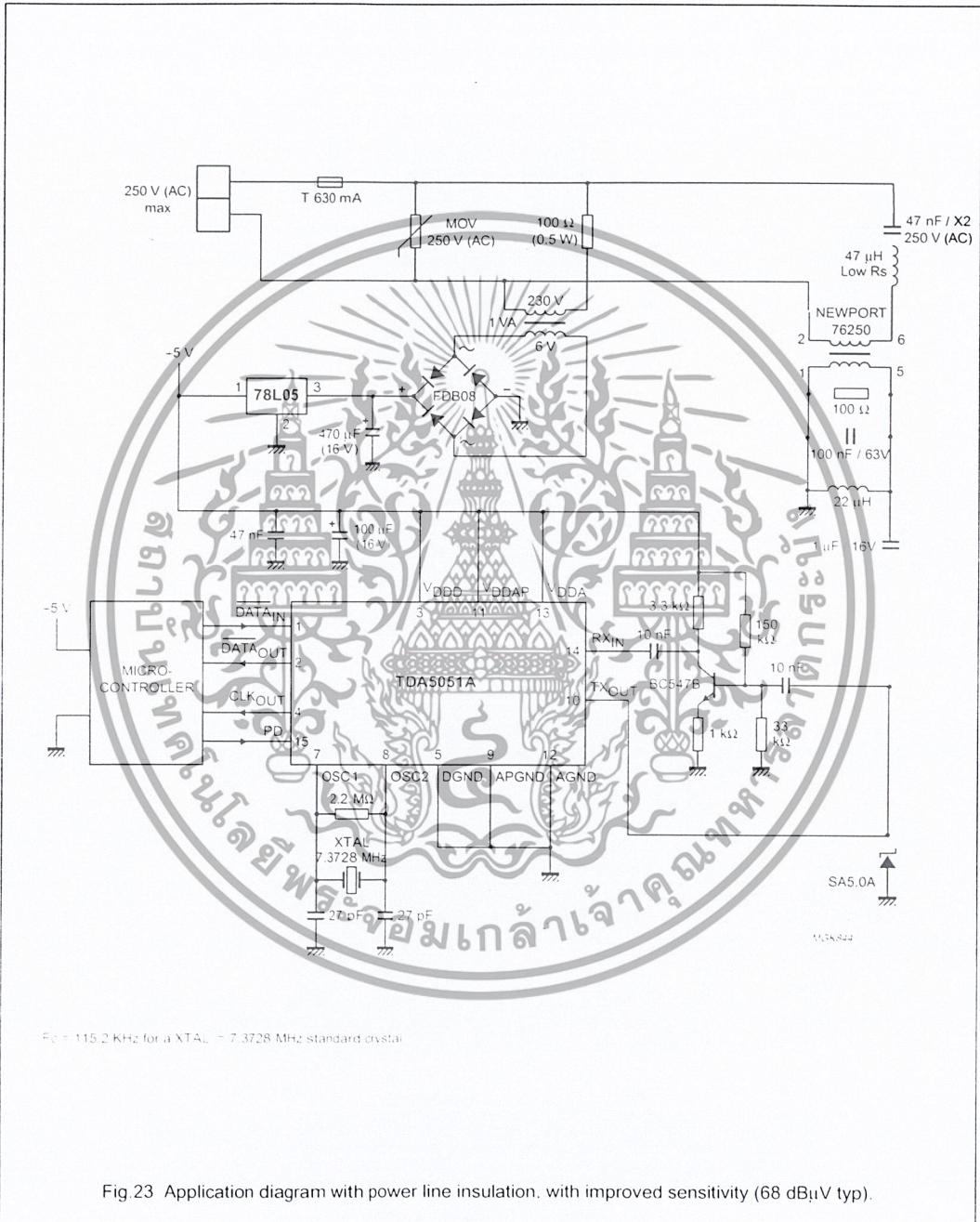
TDA5051A



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Home automation modem

TDA5051A



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

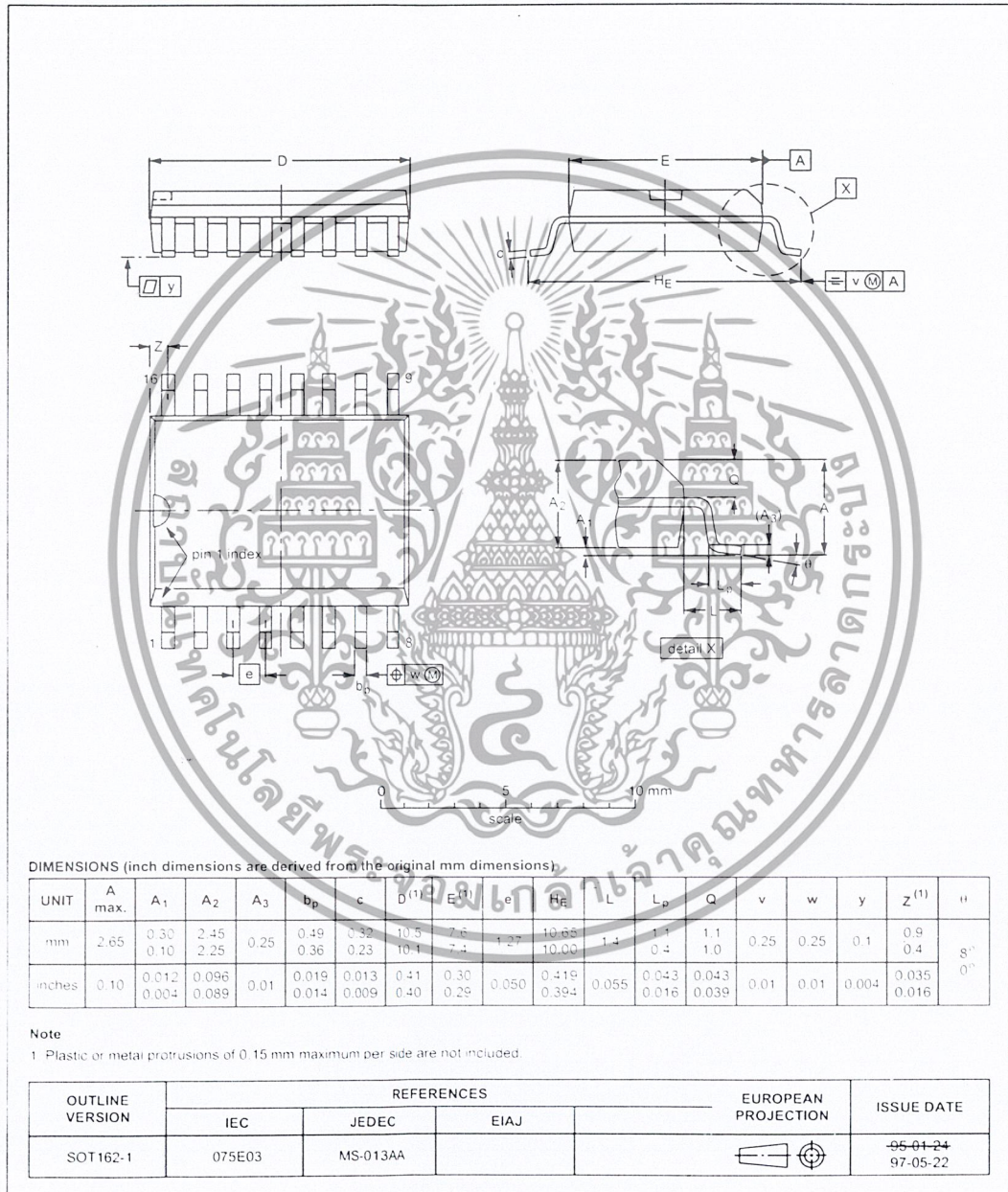
Home automation modem

TDA5051A

PACKAGE OUTLINE

SO16: plastic small outline package; 16 leads; body width 7.5 mm

SOT162-1



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Home automation modem

TDA5051A

SOLDERING**Introduction**

There is no soldering method that is ideal for all IC packages. Wave soldering is often preferred when through-hole and surface mounted components are mixed on one printed-circuit board. However, wave soldering is not always suitable for surface mounted ICs, or for printed-circuits with high population densities. In these situations reflow soldering is often used.

This text gives a very brief insight to a complex technology. A more in-depth account of soldering ICs can be found in our "IC Package Databook" (order code 9398 652 90011).

Reflow soldering

Reflow soldering techniques are suitable for all SO packages.

Reflow soldering requires solder paste (a suspension of fine solder particles, flux and binding agent) to be applied to the printed-circuit board by screen printing, stencilling or pressure-syringe dispensing before package placement.

Several techniques exist for reflowing; for example, thermal conduction by heated belt. Dwell times vary between 50 and 300 seconds depending on heating method. Typical reflow temperatures range from 215 to 250 °C.

Preheating is necessary to dry the paste and evaporate the binding agent. Preheating duration: 45 minutes at 45 °C.

Wave soldering

Wave soldering techniques can be used for all SO packages if the following conditions are observed:

- A double-wave (a turbulent wave with high upward pressure followed by a smooth laminar wave) soldering technique should be used
- The longitudinal axis of the package footprint must be parallel to the solder flow
- The package footprint must incorporate solder thieves at the downstream end.

During placement and before soldering, the package must be fixed with a droplet of adhesive. The adhesive can be applied by screen printing, pin transfer or syringe dispensing. The package can be soldered after the adhesive is cured.

Maximum permissible solder temperature is 260 °C, and maximum duration of package immersion in solder is 10 seconds, if cooled to less than 150 °C within 6 seconds. Typical dwell time is 4 seconds at 250 °C.

A mildly-activated flux will eliminate the need for removal of corrosive residues in most applications.

Repairing soldered joints

Fix the component by first soldering two diagonally- opposite end leads. Use only a low voltage soldering iron (less than 24 V) applied to the flat part of the lead. Contact time must be limited to 10 seconds at up to 300 °C. When using a dedicated tool, all other leads can be soldered in one operation within 2 to 5 seconds between 270 and 320 °C.

Home automation modem

TDA5051A

DEFINITIONS

Data sheet status	
Objective specification	This data sheet contains target or goal specifications for product development.
Preliminary specification	This data sheet contains preliminary data; supplementary data may be published later.
Product specification	This data sheet contains final product specifications.
Limiting values	
Limiting values given are in accordance with the Absolute Maximum Rating System (IEC 134). Stress above one or more of the limiting values may cause permanent damage to the device. These are stress ratings only and operation of the device at these or at any other conditions above those given in the Characteristics sections of the specification is not implied. Exposure to limiting values for extended periods may affect device reliability.	
Application information	
Where application information is given, it is advisory and does not form part of the specification.	

LIFE SUPPORT APPLICATIONS

These products are not designed for use in life support appliances, devices, or systems where malfunction of these products can reasonably be expected to result in personal injury. Philips customers using or selling these products for use in such applications do so at their own risk and agree to fully indemnify Philips for any damages resulting from such improper use or sale.

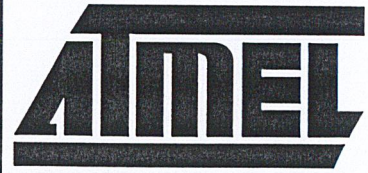
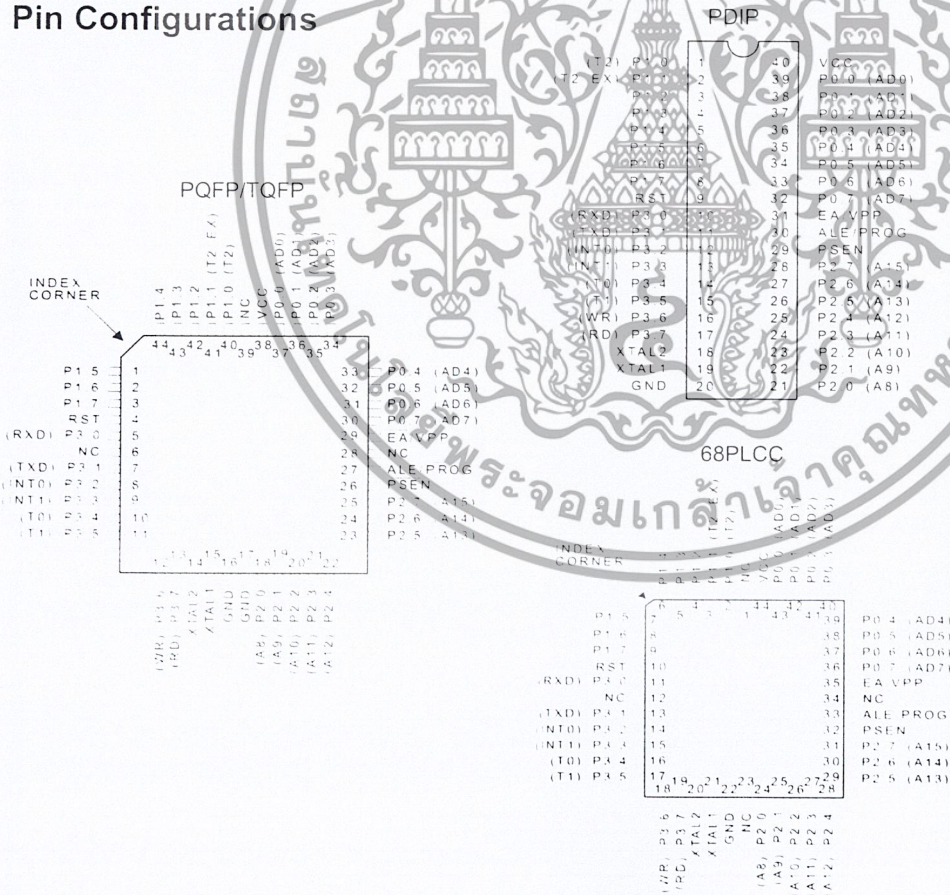
Features

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

Description

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

Pin Configurations



8-Bit Microcontroller with 8K Bytes Flash

AT89C52

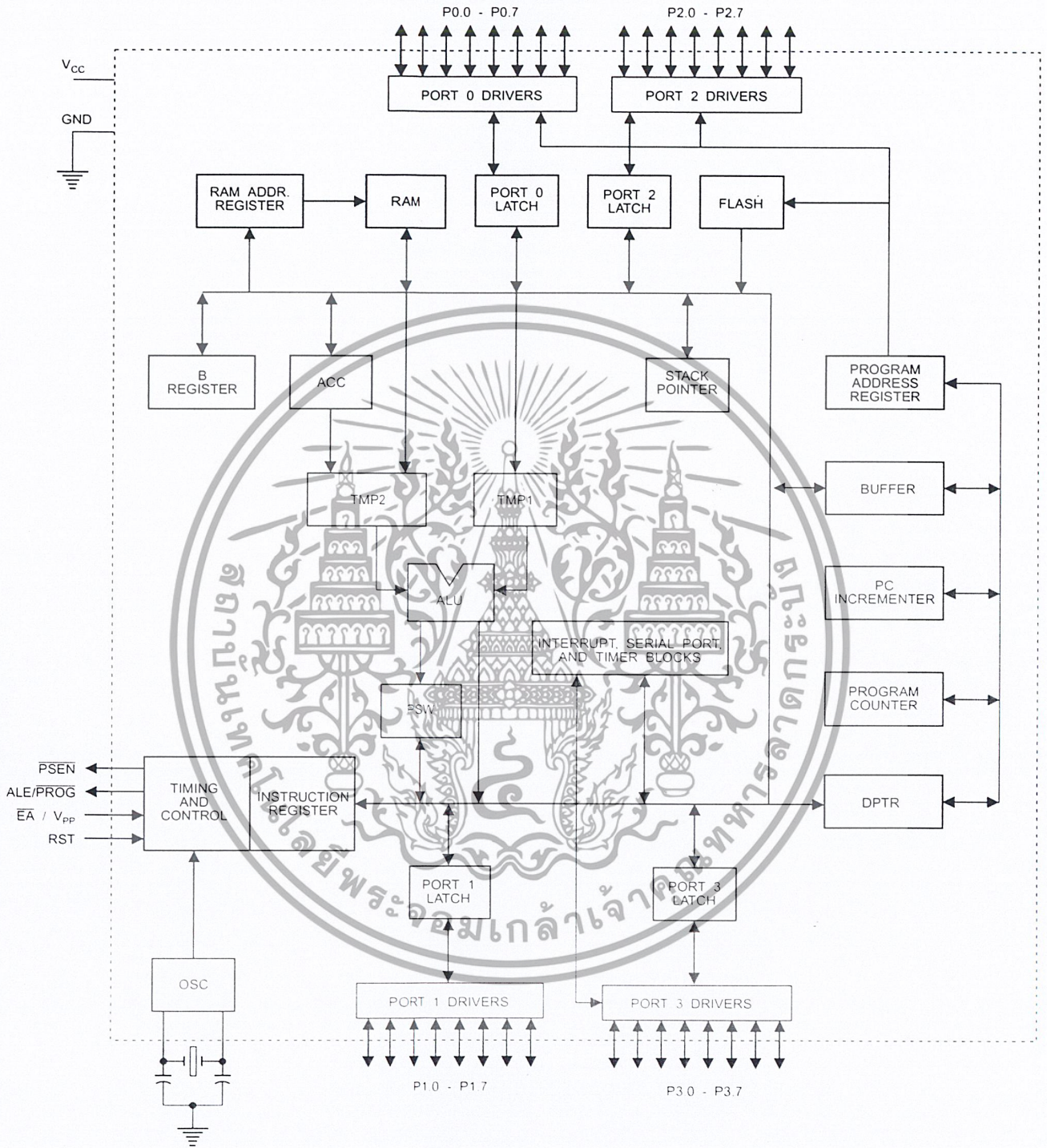
(continued)

0313F-A-12/97



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Block Diagram



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
 ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

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

Pin Description

V_{CC}
Supply voltage.

GND
Ground.

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

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

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

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

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

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

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

Port 2

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

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

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

Port 3

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

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

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

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

RST

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

ALE/PROG

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

In normal operation, ALE is emitted at a constant rate of 1/6 the oscillator frequency and may be used for external timing or clocking purposes. Note, however, that one ALE





pulse is skipped during each access to external data memory.

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

PSEN

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

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

$\overline{\text{EA}}/\text{V}_{\text{PP}}$

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

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

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

XTAL1

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

XTAL2

Output from the inverting oscillator amplifier.

Table 1. AT89C52 SFR Map and Reset Values

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

Special Function Registers

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

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

User software should not write 1s to these unlisted locations, since they may be used in future products to invoke

new features. In that case, the reset or inactive values of the new bits will always be 0.

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

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

Table 2. T2CON—Timer/Counter 2 Control Register

T2CON Address = 0C8H		Reset Value = 0000 0000B						
Bit Addressable								
Bit	TF2	EXF2	RCLK	TCLK	EXEN2	TR2	C/T $\bar{2}$	CP/RL $\bar{2}$
	7	6	5	4	3	2	1	0

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

Data Memory

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

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

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

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

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

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

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



Timer 0 and 1

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

Timer 2

Timer 2 is a 16-bit Timer/Counter that can operate as either a timer or an event counter. The type of operation is selected by bit $C/T2$ in the SFR T2CON (shown in Table 2).

Timer 2 has three operating modes: capture, auto-reload (up or down counting), and baud rate generator. The modes are selected by bits in T2CON, as shown in Table 3.

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

Table 3. Timer 2 Operating Modes

RCLK +TCLK	CP/RL2	TR2	MODE
0	0	1	16-Bit Auto-Reload
0	1	1	16-Bit Capture
1	X	1	Baud Rate Generator
X	X	0	(Off)

In the Counter function, the register is incremented in response to a 1-to-0 transition at its corresponding external input pin, T2. In this function, the external input is sampled during S5P2 of every machine cycle. When the samples

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

Capture Mode

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

Auto-Reload (Up or Down Counter)

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

Figure 1. Timer in Capture Mode

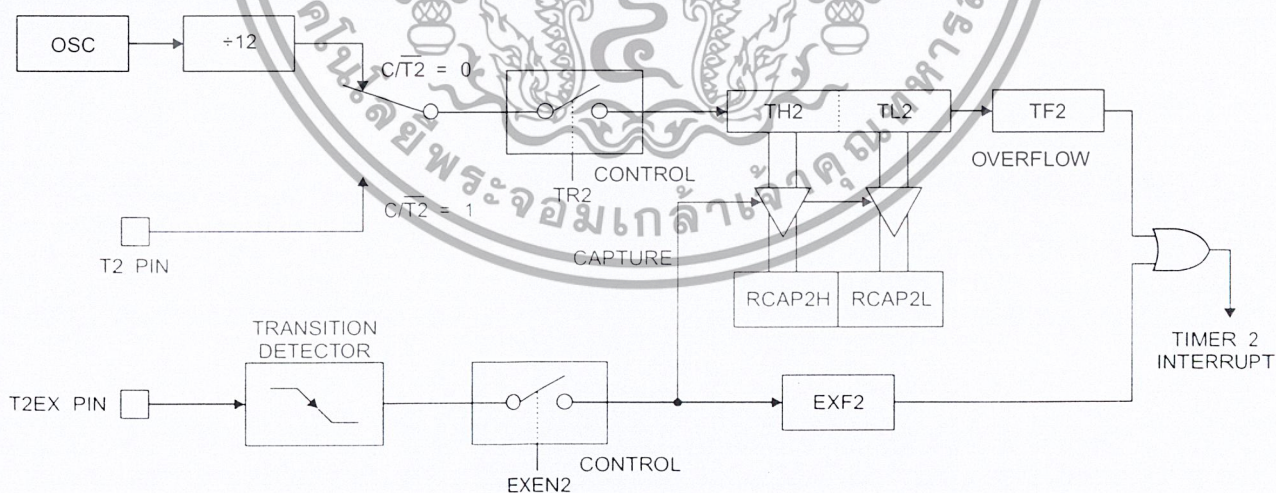


Figure 2 shows Timer 2 automatically counting up when DCEN = 0. In this mode, two options are selected by bit EXEN2 in T2CON. If EXEN2 = 0, Timer 2 counts up to 0FFFFH and then sets the TF2 bit upon overflow. The overflow also causes the timer registers to be reloaded with the 16-bit value in RCAP2H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

Setting the DCEN bit enables Timer 2 to count up or down, as shown in Figure 3. In this mode, the T2EX pin controls

the direction of the count. A logic 1 at T2EX makes Timer 2 count up. The timer will overflow at 0FFFFH and set the TF2 bit. This overflow also causes the 16-bit value in RCAP2H and RCAP2L to be reloaded into the timer registers, TH2 and TL2, respectively.

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

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

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

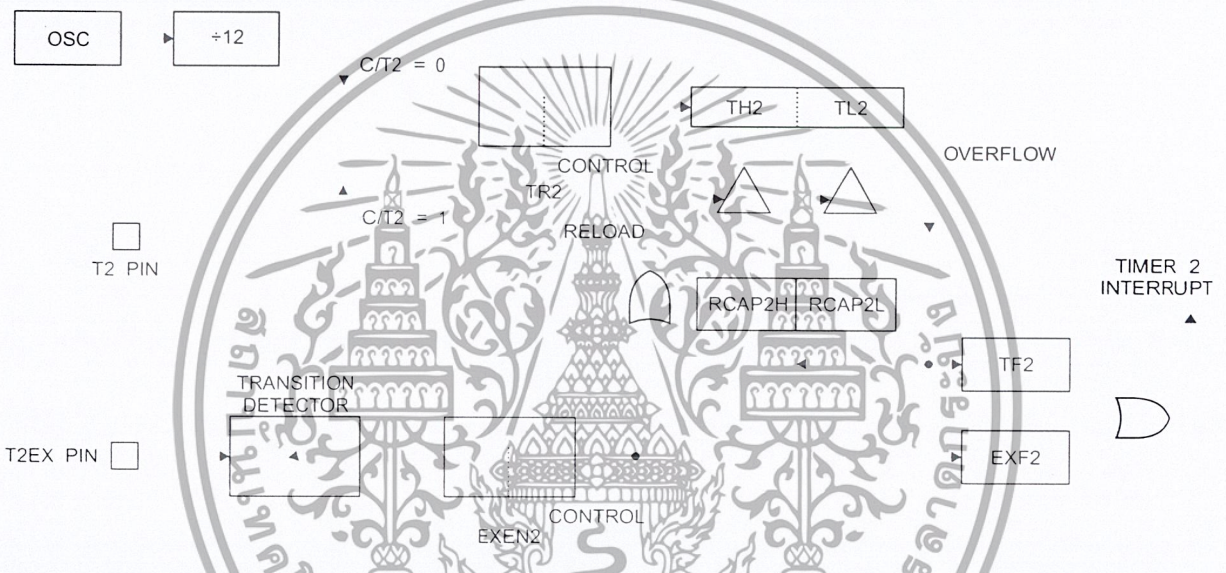


Table 4. T2MOD—Timer 2 Mode Control Register

T2MOD Address = 0C9H		Reset Value = XXXX XX00B						
Not Bit Addressable								
Bit	7	6	5	4	3	2	T2OE	DCEN

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



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

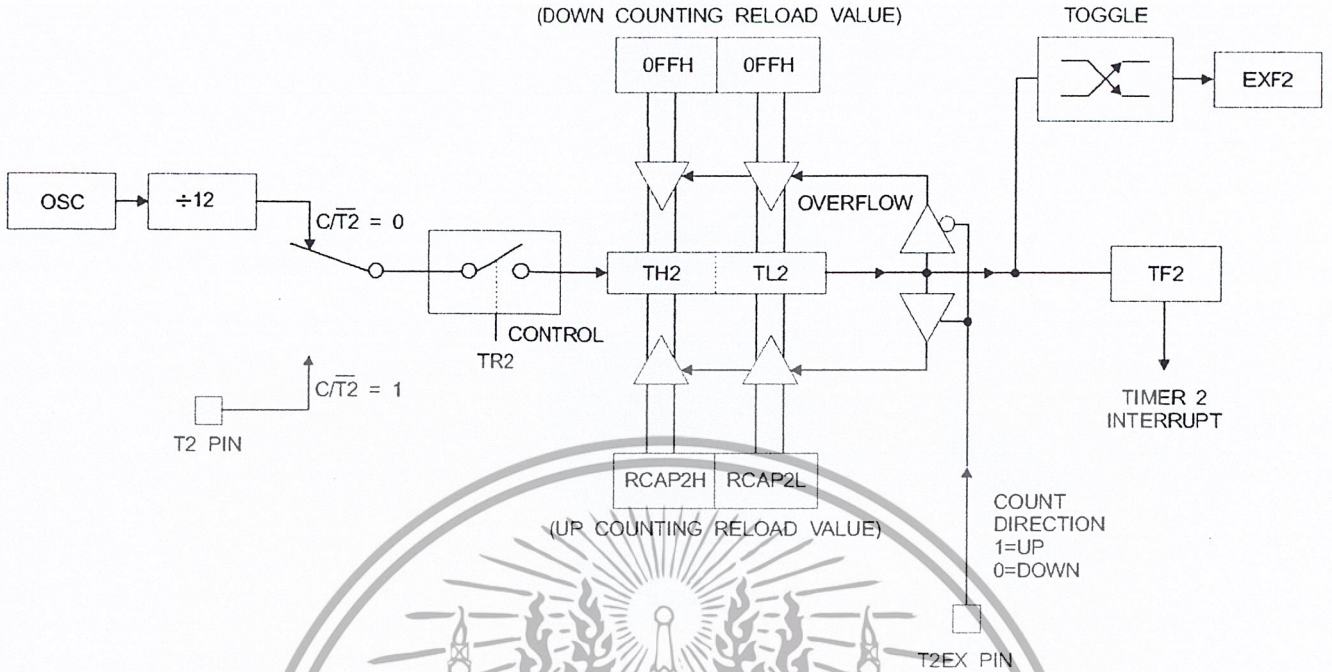
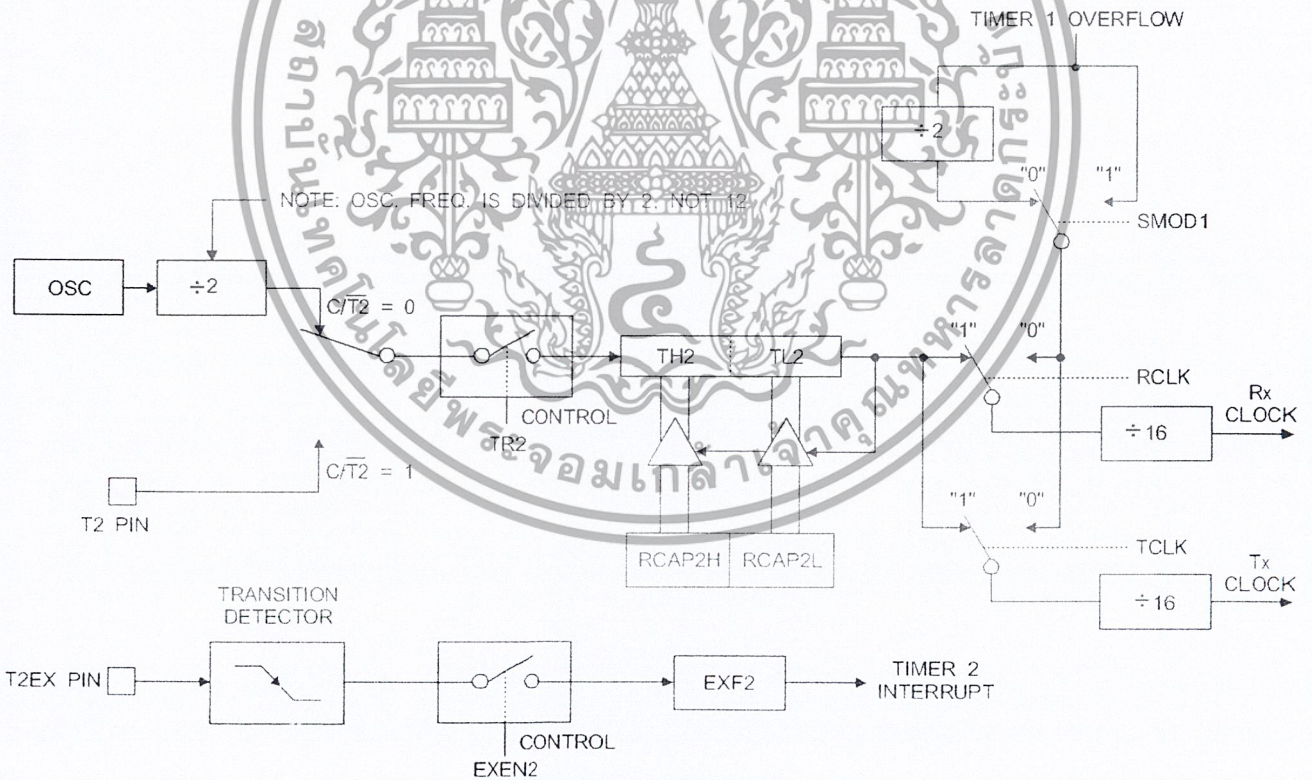


Figure 4. Timer 2 in Baud Rate Generator Mode



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Programmable Clock Out

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

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

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

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

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

UART

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

Interrupts

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

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

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

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

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

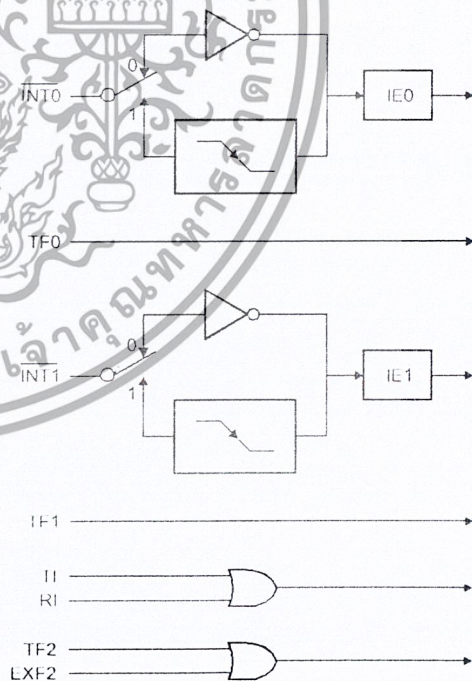
Table 5. Interrupt Enable (IE) Register

(MSB)							(LSB)
EA	—	ET2	ES	ET1	EX1	ET0	EX0
Enable Bit = 1 enables the interrupt.							
Enable Bit = 0 disables the interrupt.							

Symbol	Position	Function
EA	IE.7	Disables all interrupts. If EA = 0, no interrupt is acknowledged. If EA = 1, each interrupt source is individually enabled or disabled by setting or clearing its enable bit.
—	IE.6	Reserved.
ET2	IE.5	Timer 2 interrupt enable bit.
ES	IE.4	Serial Port interrupt enable bit.
ET1	IE.3	Timer 1 interrupt enable bit.
EX1	IE.2	External interrupt 1 enable bit.
ET0	IE.1	Timer 0 interrupt enable bit.
EX0	IE.0	External interrupt 0 enable bit.

User software should never write 1s to unimplemented bits, because they may be used in future AT89 products.

Figure 6. Interrupt Sources



Oscillator Characteristics

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

Idle Mode

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

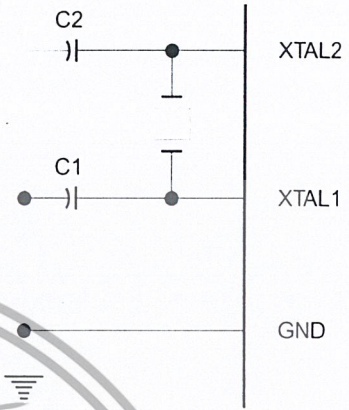
Note that when idle mode is terminated by a hardware reset, the device normally resumes program execution from where it left off, up to two machine cycles before the internal reset algorithm takes control. On-chip hardware inhibits access to internal RAM in this event, but access to the port pins is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

Power Down Mode

In the power down mode, the oscillator is stopped, and the instruction that invokes power down is the last instruction executed. The on-chip RAM and Special Function Registers retain their values until the power down mode is terminated. The only exit from power down is a hardware reset. Reset redefines the SFRs but does not change the on-chip RAM. The reset should not be activated before V_{CC} is

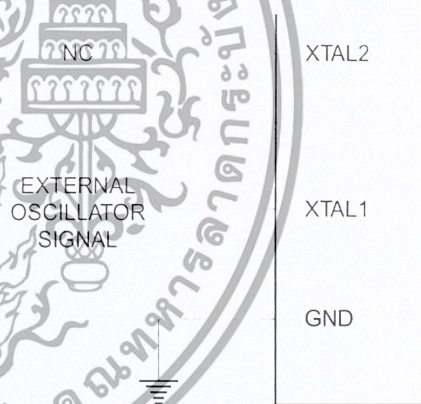
restored to its normal operating level and must be held active long enough to allow the oscillator to restart and stabilize.

Figure 7. Oscillator Connections



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

Figure 8. External Clock Drive Configuration



Status of External Pins During Idle and Power Down Modes

Mode	Program Memory	ALE	PSEN	PORT0	PORT1	PORT2	PORT3
Idle	Internal	1	1	Data	Data	Data	Data
Idle	External	1	1	Float	Data	Address	Data
Power Down	Internal	0	0	Data	Data	Data	Data
Power Down	External	0	0	Float	Data	Data	Data



Program Memory Lock Bits

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

Lock Bit Protection Modes

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

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

Programming the Flash

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

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

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

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

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

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

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

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

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

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

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

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

Programming Interface

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

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

Flash Programming Modes

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

Note: 1. Chip Erase requires a 10-ms PROG pulse.



Figure 9. Programming the Flash Memory

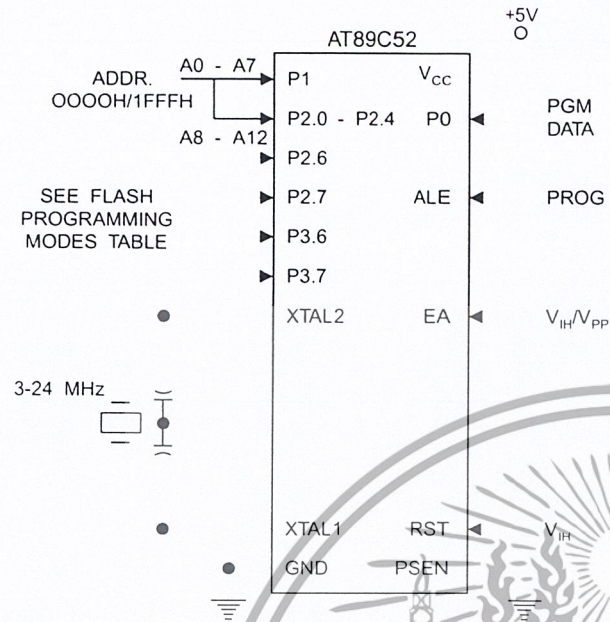
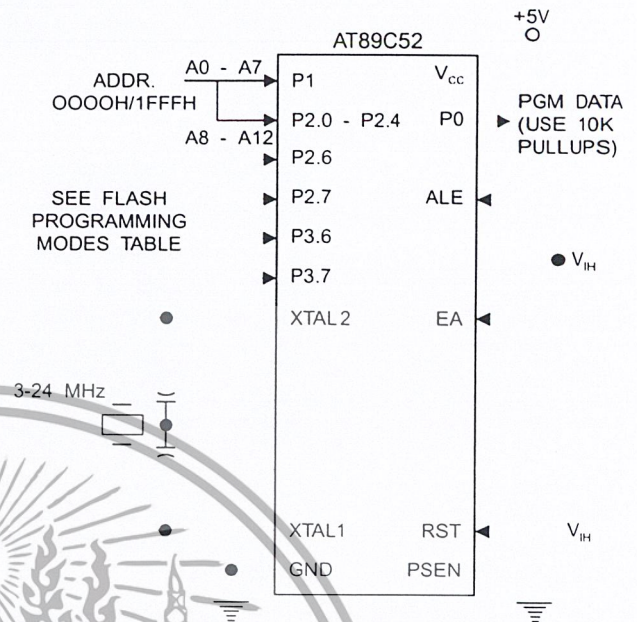


Figure 10. Verifying the Flash Memory



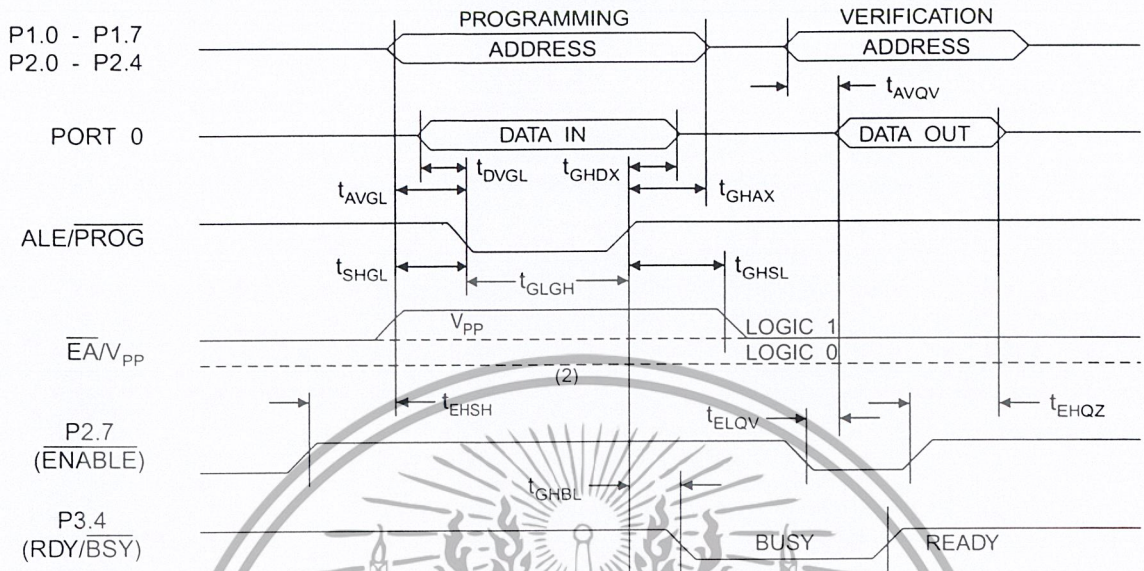
Flash Programming and Verification Characteristics

$T_A = 0^\circ\text{C to } 70^\circ\text{C}$, $V_{CC} = 5.0 \pm 10\%$

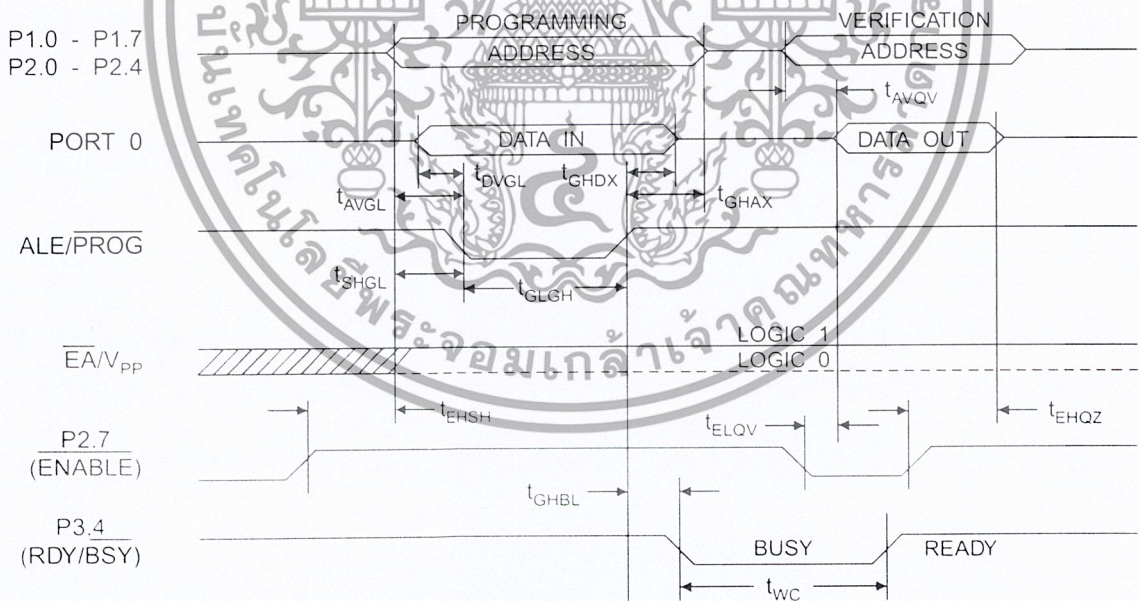
Symbol	Parameter	Min	Max	Units
$V_{PP}^{(1)}$	Programming Enable Voltage	11.5	12.5	V
$I_{PP}^{(1)}$	Programming Enable Current		1.0	mA
$1/t_{CLCL}$	Oscillator Frequency	3	24	MHz
t_{AVGL}	Address Setup to PROG Low	$48t_{CLCL}$		
t_{GHAX}	Address Hold After PROG	$48t_{CLCL}$		
t_{DVGL}	Data Setup to PROG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

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

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



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



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



Absolute Maximum Ratings*

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

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

DC Characteristics

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

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

Notes: 1. Under steady state (non-transient) conditions, I_{OL} must be externally limited as follows:

Maximum I_{OL} per port pin: 10 mA

Maximum I_{OL} per 8-bit port:

Port 0: 26 mA Ports 1, 2, 3: 15 mA

Maximum total I_{OL} for all output pins: 71 mA

If I_{OL} exceeds the test condition, V_{OL} may exceed the related specification. Pins are not guaranteed to sink current greater than the listed test conditions.

2. Minimum V_{CC} for Power Down is 2V.

AC Characteristics

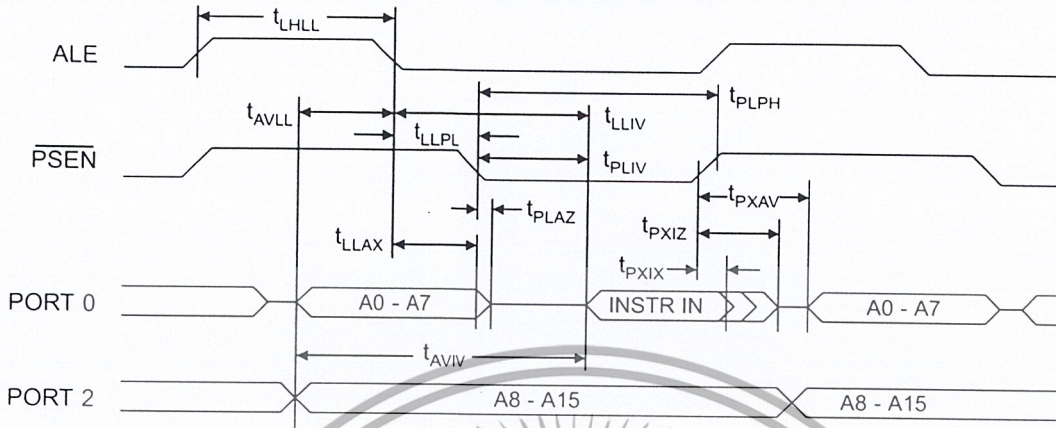
Under operating conditions, load capacitance for Port 0, ALE/ $\overline{\text{PROG}}$, and $\overline{\text{PSEN}}$ = 100 pF; load capacitance for all other outputs = 80 pF.

External Program and Data Memory Characteristics

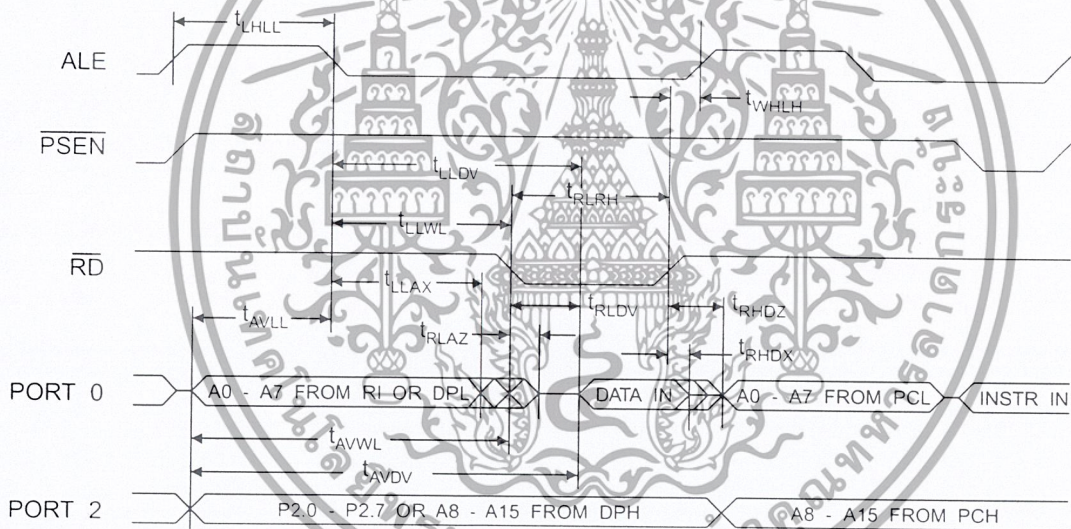
Symbol	Parameter	12 MHz Oscillator		Variable Oscillator		Units
		Min	Max	Min	Max	
$1/t_{\text{CLCL}}$	Oscillator Frequency			0	24	MHz
t_{LHLL}	ALE Pulse Width	127		$2t_{\text{CLCL}}-40$		ns
t_{AVLL}	Address Valid to ALE Low	43		$t_{\text{CLCL}}-13$		ns
t_{LLAX}	Address Hold After ALE Low	48		$t_{\text{CLCL}}-20$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



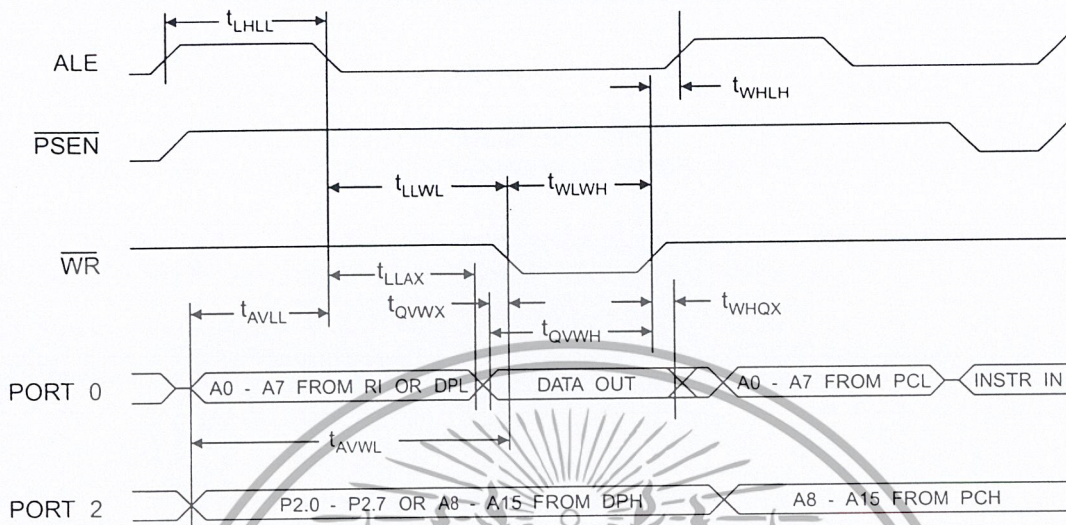
External Program Memory Read Cycle



External Data Memory Read Cycle



External Data Memory Write Cycle



External Clock Drive Waveforms



External Clock Drive

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



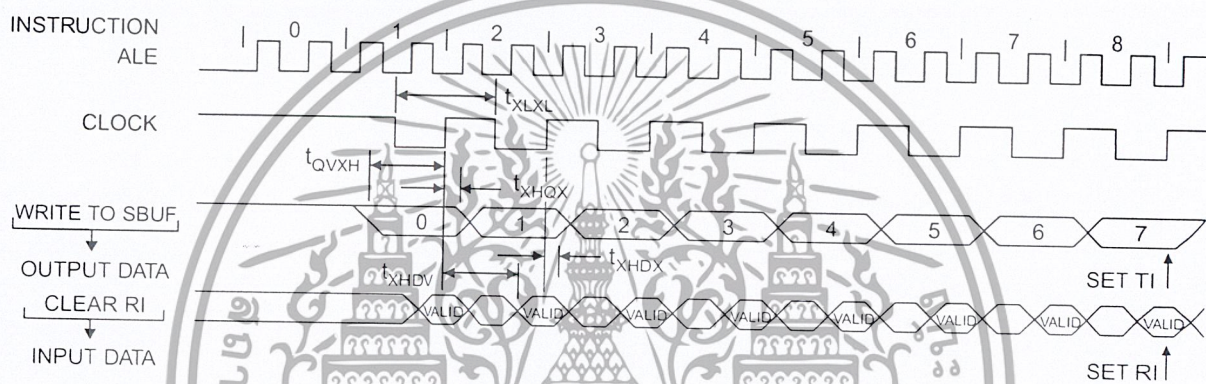
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Serial Port Timing: Shift Register Mode Test Conditions

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

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

Shift Register Mode Timing Waveforms



AC Testing Input/Output Waveforms (1)

Float Waveforms (1)

$V_{CC} - 0.5V$

$0.2 V_{CC} + 0.9V$

$V_{LOAD} + 0.1V$

$V_{OL} - 0.1V$

TEST POINTS

V_{LOAD}

Timing Reference Points

0.45V

$0.2 V_{CC} - 0.1V$

$V_{LOAD} - 0.1V$

$V_{OL} + 0.1V$

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

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

Ordering Information

Speed (MHz)	Power Supply	Ordering Code	Package	Operation Range
12	5V ± 20%	AT89C52-12AC	44A	Commercial (0°C to 70°C)
		AT89C52-12JC	44J	
		AT89C52-12PC	40P6	
		AT89C52-12QC	44Q	
		Industrial (-40°C to 85°C)	AT89C52-12AI	44A
			AT89C52-12JI	44J
			AT89C52-12PI	40P6
			AT89C52-12QI	44Q
		Automotive (-40°C to 105°C)	AT89C52-12AA	44A
			AT89C52-12JA	44J
			AT89C52-12PA	40P6
			AT89C52-12QA	44Q
16	5V ± 20%	AT89C52-16AC	44A	Commercial (0°C to 70°C)
		AT89C52-16JC	44J	
		AT89C52-16PC	40P6	
		AT89C52-16QC	44Q	
		Industrial (-40°C to 85°C)	AT89C52-16AI	44A
			AT89C52-16JI	44J
			AT89C52-16PI	40P6
			AT89C52-16QI	44Q
		Automotive (-40°C to 105°C)	AT89C52-16AA	44A
			AT89C52-16JA	44J
			AT89C52-16PA	40P6
			AT89C52-16QA	44Q
20	5V ± 20%	AT89C52-20AC	44A	Commercial (0°C to 70°C)
		AT89C52-20JC	44J	
		AT89C52-20PC	40P6	
		AT89C52-20QC	44Q	
		Industrial (-40°C to 85°C)	AT89C52-20AI	44A
			AT89C52-20JI	44J
			AT89C52-20PI	40P6
			AT89C52-20QI	44Q

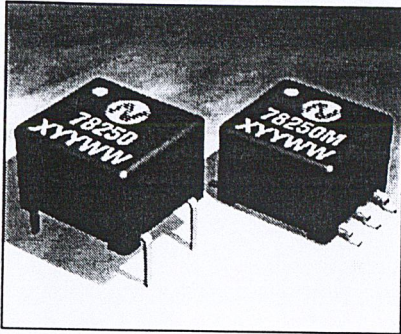
Ordering Information

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



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

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่นอนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



FEATURES

- Maxim Compatible
- Isolation to 4kV
- Surface Mount Options
- Toroidal Construction
- Standard Pinout
- UL 94V-0 Package Material
- Fully Encapsulated
- Low Profile
- Industrial Temperature Range

DESCRIPTION

The 78250 series of converter transformers are specifically designed for use with Maxim chipsets to provide isolated RS232 interfaces. Carefully controlled turns ratios ensure consistent performance whilst a toroidal construction minimises EMI. The surface mount devices are fully compatible with CECC00802 to 280°C which allows them to be placed and reflowed with IC's thus reducing time and cost in production.

SELECTION GUIDE 78250/78250M

Parameter	Conditions	MIN	TYP	MAX	Units
Primary Inductance, L_p	10kHz, 100mV	1.0	2.0	2.5	mH
Leakage Inductance, L_l	100kHz, 100mV		2.0	3.0	μ H
Interwinding Capacitance, C_{ww}	100kHz, 100mV		69	90	pF
D.C. Resistance, R_{DC}	<0.1VDC		1.0	2.0	Ω
Volt-time Product, E_T	5kHz, 5V	50			V μ s

SELECTION GUIDE 78250V/78250MV

Parameter	Conditions	MIN	TYP	MAX	Units
Primary Inductance, L_p	10kHz, 100mV	1.0	2.0	2.5	mH
Leakage Inductance, L_l	100kHz, 100mV		35	40	μ H
Interwinding Capacitance, C_{ww}	100kHz, 100mV		9	12	pF
D.C. Resistance, R_{DC}	<0.1VDC		1.4	1.8	Ω
Volt-time Product, E_T	5kHz, 5V	50			V μ s

ABSOLUTE MAXIMUM RATINGS

Operating free air temperature range	-40°C to 85°C
Storage temperature range	-50°C to 125°C
Lead temperature 1.5mm from case for 10 seconds	300°C
Peak current I_p	300mA
Isolation voltage 78250(M) (flash tested for 1 second)	1500Vrms
Isolation voltage 78250V(M) (flash tested for 1 second)	4000Vrms

All specifications typical at $T_A=25^\circ\text{C}$

www.dc-dc.com

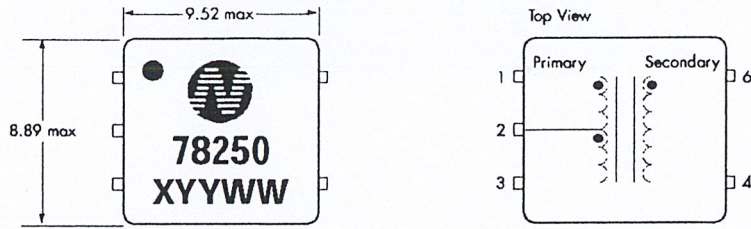
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับกรใช้ภายในเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

78250 SERIES

Converter Transformers

MECHANICAL DIMENSIONS

6 Pin DIP



6 Pin SM



* Pin 5 is connected to secondary centre tap.
All dimensions in mm XX.XX ± 0.25mm. All pins on a 2.54mm pitch and within ± 0.25mm of true position.

C&D Technologies (NCL) Limited reserve the right to alter or improve the specification, internal design or manufacturing process at any time, without notice. Please check with your supplier or visit our web site to ensure that you have the current and complete specification for your product before use.

© C&D Technologies (NCL) Limited 2000 NMP 78250.2

No part of this publication may be copied, transmitted or stored in a retrieval system or reproduced in any way including, but not limited to, photography, photocopy, magnetic or other recording means, without prior written permission from C&D Technologies (NCL) Limited.

Instructions for use are available from www.dc-dc.com

C&D Technologies (NCL) Ltd
Lanners Drive, Blakelands North
Milton Keynes MK14 5BU, England
Tel: +44 (0)1908 615232
Fax: +44 (0)1908 617545
email: info@cdtechnocl.com

[www: http://www.dc-dc.com](http://www.dc-dc.com)

C&D Technologies (NCL), Inc.
8917 Glenwood Avenue, Raleigh
NC 27612, USA
Tel: +1 (919) 571-9405
Fax: +1 (919) 571-9262
email: info@us.cdtechnocl.com

C&D TECHNOLOGIES
Power Solutions

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้