ปีการศึกษา 2531

ปริญญานิพนธ์

เรื่อง

ระบบควบคุมวิทยุโทรศัพท์

โดย

นายไชยวัฒน์ เกษสุวรรณ เลขประจำตัว 28.1067

นายวินัย จิตตปาลกุล เลขประจำตัว 28.1221

อาจารย์ที่ปรึกษา

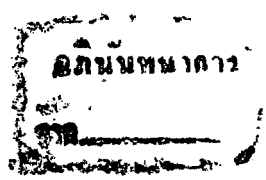
อาจารย์นิภา ลีลารุจิ

อาจารย์สมยศ จุณณะปิยะ

คณะวิศวกรรมศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้า

เจ้าคุณทหารลาดกระบัง



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

023155 -8.ค.2532

สารบัญ

		หน้า
บทคัดย่อ		
abstract		
บทที่ 1	บทนำ	1
บทที่ 2	ทฤษฎีและหลักการ	3
	2.1) ระบบควบคุมของสถานีแม่	5
	2.2) ระบบควบคุมของวิทยุโทรศัพท์	7
	2.3) โมเด็ม(MODEM)	9
	2.4) เฟสล็อคลูป(PHASE LOCK LOOP)	10
บทที่ 3	การออกแบบและคำนวณ	15
	3.1) เฟสล็อคลูป	15
	3.2) โมเด็ม	23
	3.3) ส่วนควบคุมและประมวลผล ของสถานีแม่	28
	3.4) ส่วนควบคุมและประมวลผล ของวิทยุโทรศัพท์	47
บทที่ 4	การทดลองและผลการทดลอง	64
บทที่ 5	บทวิจารณ์และสรุป	66
กิตติกรรมประกาศ		68
เอกสารอ้างอิง		69
ภาคผนวก		

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 1

บทนำ

(INTRODUCTION)

ปัจจุบันความเจริญก้าวหน้าทางเทคโนโลยี่เป็นไปอย่างรวดเร็ว ความเจริญก้าวหน้าจะแผ่ขยายไปได้ต้องอาศัยการสื่อสารและการคมนาคม จึงกล่าวได้ว่าการคมนาคมเป็นสื่อกลางของความเจริญก้าวหน้าทั้งมวล

การสื่อสารมีมากมายหลายประเภททั้งด้านวิทยุและด้านโทรศัพท์ ซึ่งมีการพัฒนาต่อเนื่องกันมาไม่หยุดยั้ง ที่จะกล่าวต่อไปนี้เป็นการสื่อสารประเภทหนึ่งซึ่งรวมการสื่อสารวิทยุและโทรศัพท์เข้าด้วยกันคือวิทยุโทรศัพท์ ซึ่งรวมข้อดีของทั้งระบบโทรศัพท์คือสามารถติดต่อได้กว้างขวาง และระบบวิทยุคือไม่ต้องใช้สายทำให้สะดวกในการใช้งาน

ในส่วนของวิทยุโทรศัพท์เองนั้นเราสามารถพบเห็นได้ตามบ้านทั่วไปซึ่งเป็นลักษณะโทรศัพท์ไม่มีสาย แต่ปัจจุบันมีการพัฒนาระบบวิทยุโทรศัพท์ขึ้นมามีลักษณะเป็นชุมสายของตัวเองซึ่งมีการเชื่อมต่อกับชุมสายโทรศัพท์ปกติ ทำให้มีข่ายการติดต่อกว้างขวางมากเหมาะสมในการเพิ่มคู่สายโทรศัพท์ในพื้นที่ซึ่งยากลำบากต่อการวางสายโทรศัพท์ซึ่งเรียกว่า ระบบโทรศัพท์เคลื่อนที่(Mobile Telephone)

เนื่องจากปัจจุบันเป็นยุคของคอมพิวเตอร์ ซึ่งมีบทบาทในงานต่างๆมากมายเพราะมีความรวดเร็วในการทำงานสูงและสามารถประมวลผลขบวนการที่ซับซ้อนต่างๆมาไว้ในระบบเล็กๆได้ ดังนั้นจุดประสงค์ของวิทยานิพนธ์ฉบับนี้จะเน้นการใช้ไมโครโปรเซสเซอร์ (Micro-Processor) ในการควบคุมระบบวิทยุโทรศัพท์ซึ่งจะใช้ความถี่ 27 MHz การควบคุมระบบจะเป็นไปอย่างอัตโนมัติโดยอาศัยโปรแกรมจากซิงค์เกิ้ลบอร์ด(single board) หรือไมโครคอมพิวเตอร์ ซึ่งการแก้ไขและพัฒนาโปรแกรมซึ่งเป็นซอร์ฟแวร์ (software) ย่อมสะดวก,รวดเร็วและดีกว่าการแก้ไขวงจรทางอิเล็กทรอนิกส์

<u>การทำงานพื้นฐานของระบบวิทยุโทรศัพท์</u>

การจำลองสร้างระบบหนึ่งๆนั้น เราจะต้องกำหนดฟังก์ชันเริ่มต้นของมันเพื่อเป็นแนวทางในการสร้างฟังก์ชันอื่นๆต่อไป

1) การติดต่อกับคู่สายโทรศัพท์ เป็นลักษณะพื้นฐานของระบบวิทยุโทรศัพท์ทั่วไป

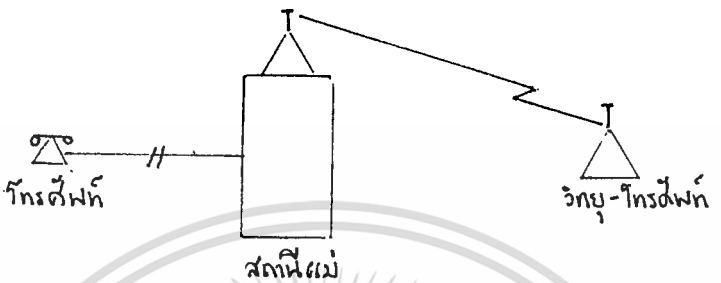
2) การติดต่อระหว่างวิทยุโทรศัพท์ด้วยกัน เป็นลักษณะของการทวนสัญญาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 2

ทฤษฎีและหลักการ

(Theory and Concept)



รูปที่ 2.1 ระบบพื้นฐานของวิทยุโทรศัพท์

จากรูปจะเห็นว่า วิทยุโทรศัพท์ประกอบไปด้วย 3 ส่วนใหญ่ๆ คือ

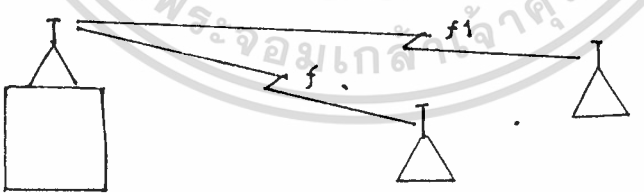
1) สถานีแม่

2) ตัววิทยุโทรศัพท์

3) คู่สายโทรศัพท์

ในการใช้งานวิทยุโทรศัพท์ในบ้านนั้น สถานีแม่จะมีเพียง 1 ช่องสัญญาณเท่านั้น เพื่อติดต่อกับตัววิทยุโทรศัพท์ที่ใช้ความถี่เดียวกัน

แต่ในระบบซึ่งมีข่ายงานในการติดต่อสูงนั้นจำเป็นจะต้องมีหลายช่องสัญญาณในการติดต่อ ถ้าวิทยุโทรศัพท์มีจำนวนมากก็จะเปลืองความถี่มากขึ้น ไม่ยืดหยุ่นต่อการขยายระบบ



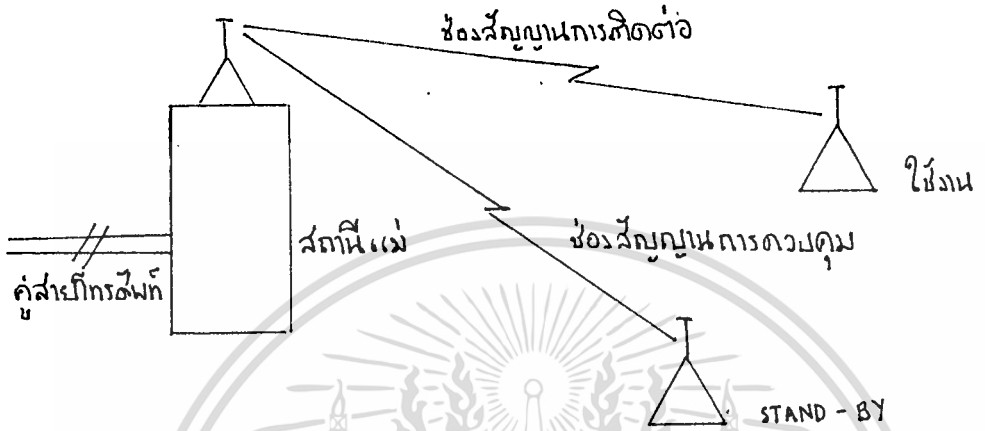
รูปที่ 2.2 การทำงานในลักษณะของการทวนสัญญาณ

จากรูป ทุกช่องสัญญาณจะมีความถี่เฉพาะ เพื่อไม่ให้เกิดการแทรกสอด (interference)ของสัญญาณ

ดังนั้นจึงมีการกำหนดสถานีแม่จำนวนหนึ่ง ซึ่งแต่ละสถานีมีความถี่เฉพาะตัว โดยมีจำนวนน้อยกว่าจำนวนของวิทยุโทรศัพท์ที่ใช้งาน ผู้ใช้เมื่อต้องการใช้งานจะกดปุ่มเลือกช่อง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สัญญาณหมายเลขต่างๆจนกว่าจะพบช่องสัญญาณที่ว่าง โดยทราบได้จาก tone ที่ตอบกลับมา ถ้าไม่มีการตอบกลับมาแสดงว่าไม่มีช่องสัญญาณที่ว่าง



รูปที่ 2.3 หลักการของระบบวิทยุโทรศัพท์

<u>หลักการของระบบวิทยุโทรศัพท์ที่สร้างขึ้นมา</u>

ระบบวิทยุโทรศัพท์ที่สร้างขึ้นมานี้มีข้อดีคือ ไม่จำเป็นต้องกดปุ่มเลือกช่องสัญญาณที่ว่างเพราะถ้าข่ายงานมีมากก็จะเสียเวลามาก ดังนั้นจะอาศัยหลักการดังนี้

1) สร้างช่องสัญญาณข่าวสาร ในการส่งสัญญาน digital ในการควบคุม เรียกว่า ช่องสัญญาณควบคุม(Control Channel) ซึ่งในส่วนนี้เป็นการสื่อสารแบบทางเดียว (Simplex) โดยสถานีแม่จะส่ง data ไปยังวิทยุโทรศัพท์อย่างเดียว ไม่มีการตอบกลับมา

2) ตัววิทยุโทรศัพท์เมื่อ stand-by จะรับสัญญาณจากช่องสัญญาณควบคุมอย่างเดียวเพื่อรับข่าวสาร

3) ตัววิทยุโทรศัพท์อาศัยข่าวสารจากช่องสัญญาณควบคุม ในการสร้างความถี่ของช่องสัญญาณที่จะใช้งาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.1) ระบบควบคุมของสถานีแม่

หน้าที่สำคัญของสถานีแม่คือ เป็นส่วนที่เชื่อมต่อระหว่างวิทยุโทรศัพท์ด้วยกันหรือระหว่างโทรศัพท์ธรรมดา กับวิทยุโทรศัพท์

ดังนั้นหน่วยประมวลผลกลางซึ่งมีหน้าที่ควบคุมสถานีแม่จะมีฟังก์ชันในการทำงานคือ

(1) รับข้อมูลในระบบจากส่วนอินเตอร์เฟส (Interface) ต่างๆ

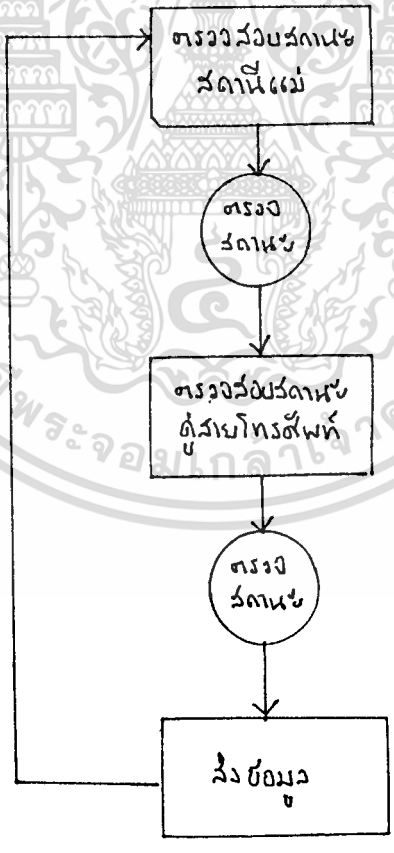
(2) เก็บสถานะของคู่สายโทรศัพท์นอก (External Line)

(3) เก็บสถานะของช่องสัญญาณการติดต่อสื่อสาร (Traffic Channel)

(4) ประมวลผลเพื่อส่งต่อไปยังช่องสัญญาณการควบคุม

นอกจากนั้น หน่วยประมวลผลนี้ยังมีหน้าที่ในการเก็บสถานะของตัววิทยุโทรศัพท์ทุกตัวไว้ด้วย

เราสามารถเขียนบล็อคไดอะแกรม ของส่วนควบคุมของสถานีแม่ได้ดังนี้



รูปที่ 2.4 โฟล์วชาร์ทของส่วนควบคุมสถานีแม่

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เราสามารถที่จะอธิบายส่วนต่างๆแยกได้เป็น

(1) หน่วยประมวลผลกลาง (Central Processing Unit:CPU) ทำหน้าที่คำนวณทางคณิตศาสตร์,ลอจิก และแยกแยะข้อมูล เพื่อควบคุมระบบทั้งหมดโดยผ่านหน่วยอินเตอร์เฟส(Interface)

(2) พอร์ตแบบขนาน (Parallel Port) เป็นหน่วยที่ติดต่ออยู่กับอุปกรณ์ภายนอก เพื่อส่งข้อมูลไปยังหน่วยประมวลผลกลาง ดังนั้นสามารถแบ่งหน้าที่ของพอร์ตแบบขนานได้เป็น

(2.1) รับสัญญาณจากคู่สายโทรศัพท์

(2.2) รับสัญญาณจากช่องสัญญาณการสื่อสาร

(3) พอร์ตอนุกรม (Serial Port) เป็นตัวแปลงข้อมูลแบบขนานให้เป็นอนุกรมโดยเฉพาะในการส่งข้อมูลทางวิทยุ

การส่งรับข้อมูลแบบอนุกรม(Serial Commmunication) เป็นการรับส่งข้อมูลเรียงลำดับกันไปเป็นบิทต่อเนื่องกันไปในลักษณะที่เรียกเป็นไทม์-ซีเควนเชียล (Time-Sequential) ซึ่งแบ่งออกได้เป็น 2 แบบคือ

(3.1) ซิงค์โครนัส คอมมูนิเคชั่น (Synchronous Communication)

(3.2) อะซิงค์โครนัส คอมมูนิเคชั่น (Asynchronous Communication)

ในวิทยานิพนธ์ฉบับนี้จะเป็นแบบอะซิงค์โครนัส ซึ่งเป็นระบบการส่ง-รับข้อมูลที่มีเทคนิคการใช้ซอร์ฟแวร์ในการส่งไม่มากนัก จะมีราคาถูกกว่า และความเร็วต่ำกว่าระบบซิงค์โครนัส

ข้อมูลที่จะส่งผ่านพอร์ตอนุกรมนี้จะแบ่งออกได้เป็น 3 ชุดคือ

(1) รหัสประจำตัวของตัววิทยุโทรศัพท์แต่ละตัว

(2) ข่าวสารในการติดต่อ

(3) ช่องสัญญาณการสื่อสารซึ่งถูกกำหนดให้ใช้ติดต่อ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.2) <u>ระบบควบคุมของวิทยุโทรศัพท์</u>

หน้าที่สำคัญของซีพียูซึ่งควบคุมวิทยุโทรศัพท์คือ

(1) รับดาต้าที่ส่งมาจากสถานีแม่เพื่อประมวลผล

(2) ตรวจสอบพาริตี้ของข้อมูลที่ส่งมา

(3) ส่งข้อมูลไปยังวงจรสังเคราะห์ความถี่ เพื่อให้ตรงกับช่องสัญญาณที่จะใช้งาน

(4) ตรวจรับการเรียกจากช่องสัญญาณควบคุม

(5) ควบคุมสวิทช์ในการเรียกออกของวิทยุโทรศัพท์

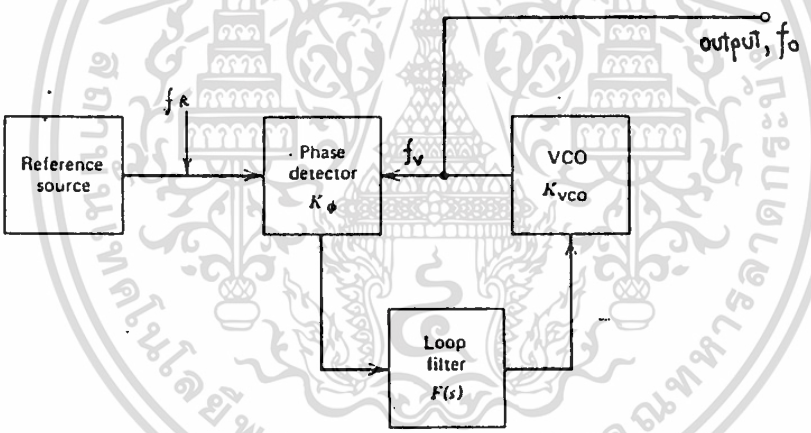
เนื่องจากลักษณะการทำงานที่ผ่านมาแล้วนั้นจะเห็นว่า ถ้าเราใช้ซีพียูในลักษณะเดียวกับสถานีแม่จะไม่เหมาะสม เนื่องจากสถานีแม่เป็นส่วนซึ่งมีข้อมูลผ่านออกมามาก การประมวลผลยุ่งยากซับซ้อน ดังนั้นทั้งระบบจึงมีลักษณะซึ่งใหญ่จึงไม่เหมาะสมกับวิทยุโทรศัพท์ซึ่งควรจะมีขนาดเล็กและมีการประมวลผลซึ่งไม่ซับซ้อนนัก ดังนั้นจึงเลือกใช้ไอซีประเภทไมโครคอนโทรลเลอร์ ไอซีประเภทนี้จะมีลักษณะเป็นไมโครคอมพิวเตอร์ซึ่งสมบูรณ์ในตัวเรียกว่า ชิงเกิลชิพคือภายในตัวเองจะประกอบไปด้วย RAM, ROM ขาอินเทอร์รัพท์ พอร์ทแบบขนาน และพอร์ทแบบอนุกรมจะเห็นว่าจะเหมาะสมกับระบบควบคุมขนาดเล็ก เราสามารถแสดงโฟล์วชาร์ตการทำงานของไมโครคอนโทรลเลอร์ได้ดังภาคผนวกที่ 2

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.4) <u>วงจรสังเคราะห์ความถี่แบบเฟสล็อคลูป</u>

วงจรสังเคราะห์ความถี่ (Frequency Synthesizer) เป็นผลรวมของวงจรซึ่งสร้างหรือสังเคราะห์ความถี่หลายค่าออกมาจากความถี่อ้างอิงค่าเดียว โดยความถี่ที่ต้องการสามารถเลือกได้โดยการกำหนดจากผู้ใช้ โดยความถี่ที่ต้องการนี้จะถูกสร้างมีค่าสูงกว่าความถี่อ้างอิง โดยความถี่นี้จะมีเสถียรภาพที่สูงมาก และมี spectrum purity ที่ดี

ระบบเฟสล็อคลูป (PLL) เป็นเทคนิคในการสร้างวงจรสังเคราะห์ความถี่ (Frequency Synthesizer) เราจะแสดงบล็อคไดอะแกรม ของระบบเฟสล็อคลูปได้ดังรูปที่ 2.7

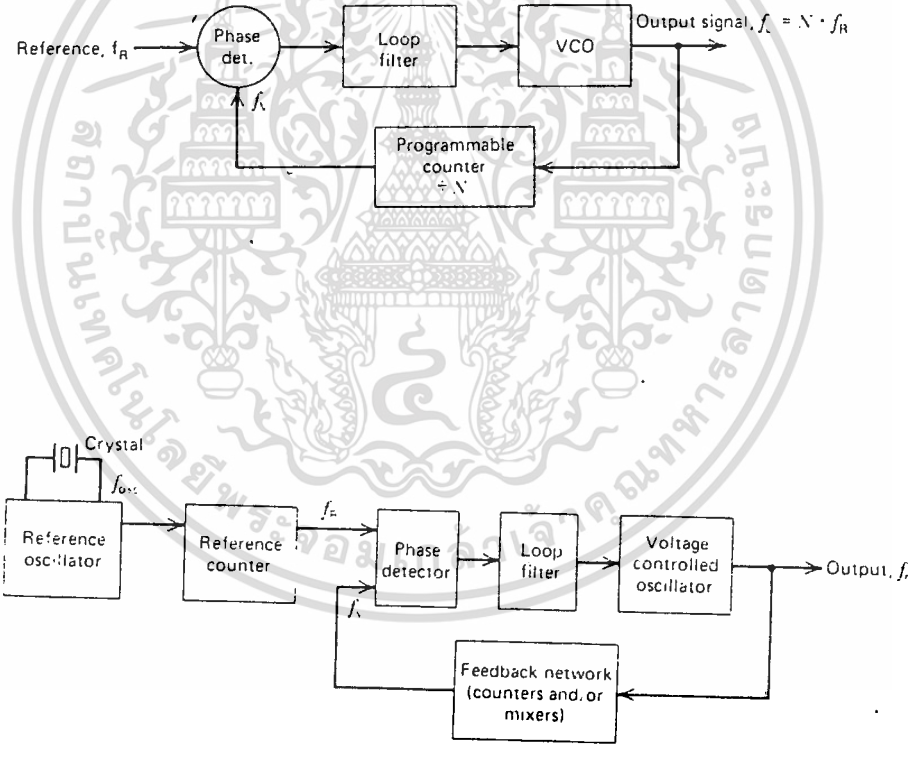


รูปที่ 2.7 บล็อคไดอะแกรมของระบบเฟสล็อคลูป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หลักการของเฟสล็อคลูป เป็นลักษณะของอิเล็คทรอนิคส์เซอร์โวลูป โดยมีอุปกรณ์พื้นฐานอยู่ 3 ส่วน คือ เฟสดีเทคเตอร์ ลูปฟิวเตอร์ และวีซีโอ(Voltage Control Oscillator: VCO) โดยความถี่อ้างอิงนั้นจะป้อนเข้าอินพุทขาหนึ่งของเฟสดีเทคเตอร์

เฟสดีเทคเตอร์จะเปรียบเทียบเฟสและ ความถี่ระหว่างความถี่อ้างอิง(ซึ่งจะต้องมีเสถียรภาพสูง เช่น จากชิ้นส่วนคริสตอล ฯลฯ) กับสัญญาณที่ได้จากวีซีโอ สัญญาณจากเฟสดีเทคเตอร์จะถูกกรอง (หรือขยาย) โดยลูปฟิลเตอร์ให้เรียบเพื่อความเหมาะสมในการควบคุมความถี่ของวีซีโอ ซึ่งจะเปลี่ยนไปจนกระทั่งไม่เกิดการแตกต่างของเฟสก็คือลูปเข้าสู่สภาวะล็อคนั่นเอง



รูปที่ 2.8 ระบบเฟสล็อคลูป ซึ่งเพิ่มส่วนหารความถี่เข้าไป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เราจะดัดแปลงเฟสล็อคลูปในการสร้างวงจรสังเคราะห์ความถี่ โดยการเพิ่มฟังก์ชั่นอีกส่วนหนึ่งก็คือ วงจรหารซึ่งจะถูกเพิ่มไปในส่วนป้อนกลับ ในหลักการเดียวกันเราจะเห็นว่าเอาท์พุทจะเป็นจำนวนเท่าของความถี่อ้างอิง เพราะว่าค่าที่กำหนดให้วงจรหารสามารถเปลี่ยนได้ ดังนั้นความถี่จากวีซีโอ จึงมีหลายค่าและมีผลต่างของความถี่แต่ละช่องเท่ากับความถี่อ้างอิง

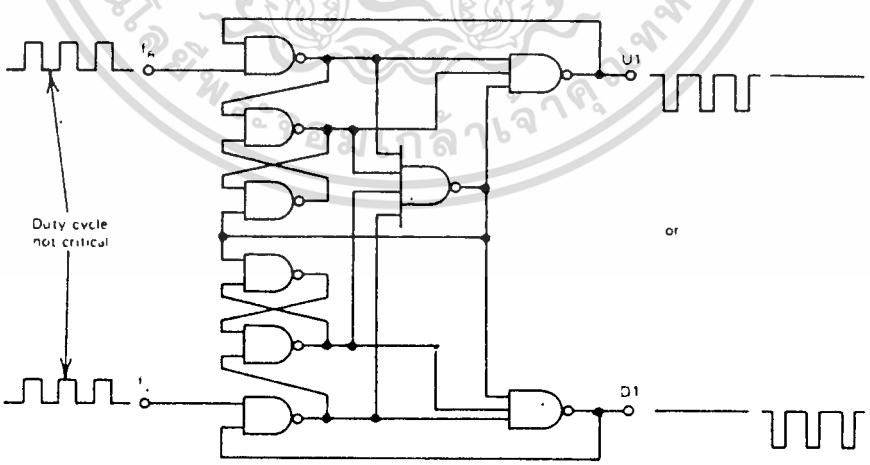
เราสามารถแยกอธิบายส่วนประกอบสำคัญของเฟสล็อคลูปได้เป็น

(1) วงจรสร้างความถี่อ้างอิง ความถี่ที่ถูกสร้างจะมีความแน่นอนสูง และเสถียรภาพที่ดี ดังนั้นวงจรที่เหมาะสมคือ วงจรที่สร้างความถี่จากคริสตอลซึ่งมีเสถียรภาพของความถี่สูงมาก

(2) วงจรหารความถี่อ้างอิง วงจรหารนี้จะหารความถี่ให้ได้ความถี่อ้างอิงที่ต้องการ

(3) วงจรเปรียบเทียบเฟส วงจรเปรียบเทียบเฟสจะสร้างสัญญาณเอาท์พุทซึ่งเกิดจากความต่างเฟสของความถี่อ้างอิงและความถี่จากส่วนป้อนกลับ สัญญาณเอาท์พุทนี้จะผ่านไปยังวงจรลูปฟิลเตอร์ เพื่อให้สัญญาณอยู่ในระดับที่คงที่เพื่อควบคุมความถี่ของวีซีโอ

วงจรเฟสดีเทคเตอร์สำหรับเฟสล็อคลูปมีอยู่ 2 ชนิดคือ ลิเนียร์หรืออนาล็อก และดิจิตอล

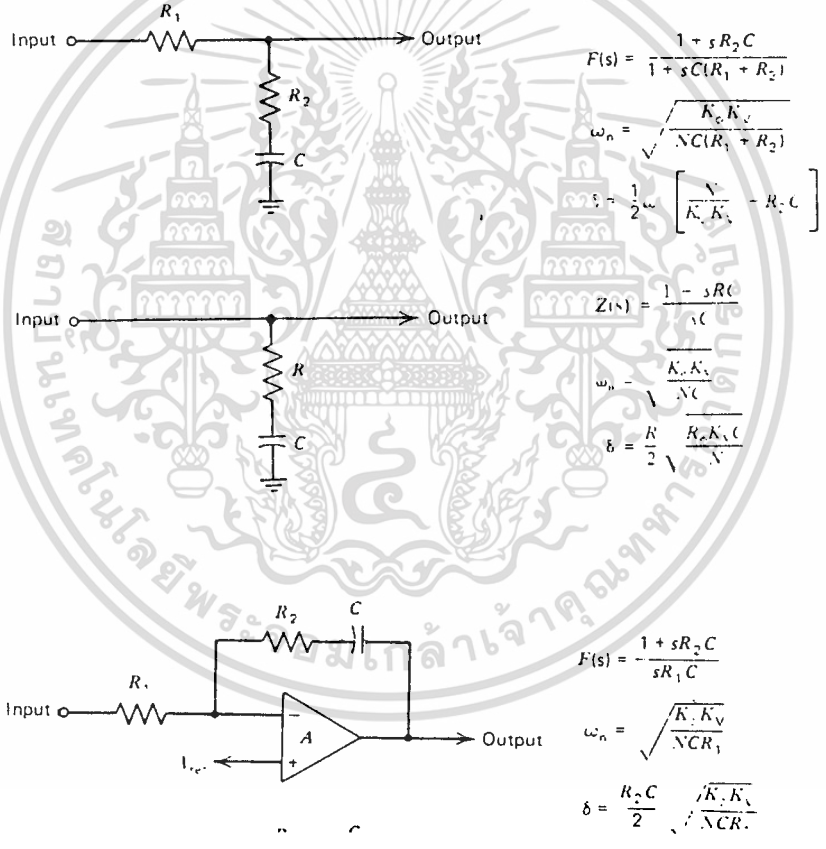


รูปที่ 2.9 วงจรดิจิตอลเฟสดีเทคเตอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

(4) วงจรลูปฟิลเตอร์ คุณสมบัติของลูปฟิลเตอร์ที่มีผลต่อการทำงานได้แก่ capture range bandwidth,capture time และ lock time เราสามารถควบคุมโดยการคำนวณด้วยการกำหนดค่า loop natural frequency, W_n และdamping factor , δ

ลูปฟิลเตอร์นี้เราสามารถใช้ได้ทั้งแอคทีฟ(active) หรือพาสซีฟ(passive) ฟิลเตอร์ คือ อุปกรณ์แอคทีฟจะมี noise เกิดขึ้น ทำให้เกิด error ขึ้นมา ส่วนพาสซีฟฟิลเตอร์แม้ไม่มีnoise แต่ไม่สามารถปรับระดับสัญญาณจากเฟสดีเทคเตอร์ ให้เหมาะสมกับวีซีโอ ได้เราจะแสดงวงจรของแอคทีฟ และพาสซีฟ ฟิลเตอร์ ได้ดังรูปที่ 2.10



รูปที่ 2.10 วงจรฟิลเตอร์ทั้งแบบพาสซีฟและแอคทีฟ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

(5) วงจรวีซีโอ (VCO) เป็นวงจรออสซิลเลเตอร์ที่ความถี่สามารถเปลี่ยนไปตามอินพุทโวลท์เตจ ซึ่งเราสามารถแบ่งออกเป็น 2 ประเภท คือ 1.) มัลติไวเบรเตอร์ (VCM) 2.) วงจรแอลซีออสซิลเลเตอร์ (VCO)

ในเฟสล็อคลูปค่าของอินพุทโวลท์เตจที่จะถูกควบคุมจากสัญญาณที่ผิดพลาดจากลูป-ฟิลเตอร์ โดยปกติแล้วในวงจรความถี่สูงนิยมใช้วีซีโอมากกว่าวีซีเอ็ม

(6) ส่วนป้อนกลับ ส่วนป้อนกลับระหว่างวีซีโอและเฟสดีเทคเตอร์จะมีคุณสมบัติดังต่อไปนี้

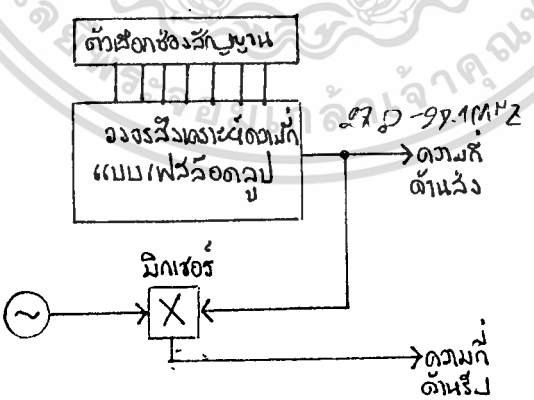
(6.1) เป็นส่วนอินเตอร์เฟสจากวีซีโอไปยังภาคต่อไป

(6.2) ปรับความถี่จากวีซีโอ ซึ่งมีค่าสูงไปยังความถี่ต่ำเพื่อความเหมาะสมในการป้อนเข้าเฟสดีเทคเตอร์

(6.3) ทำให้ระดับสัญญาณ ความกว้างของพัลส์ ฯลฯ เหมาะสมในการป้อนเข้าเฟสดีเทคเตอร์

(6.4) ใช้เป็นส่วนโปรแกรมในการกำหนดความถี่ที่ต้องการ

เราสามารถเขียนบล็อคไดอะแกรมแสดงวงจรสังเคราะห์ความถี่แบบเฟสล็อคลูปในวิทยานิพนธ์ฉบับนี้ได้ตามรูปดังต่อไปนี้.



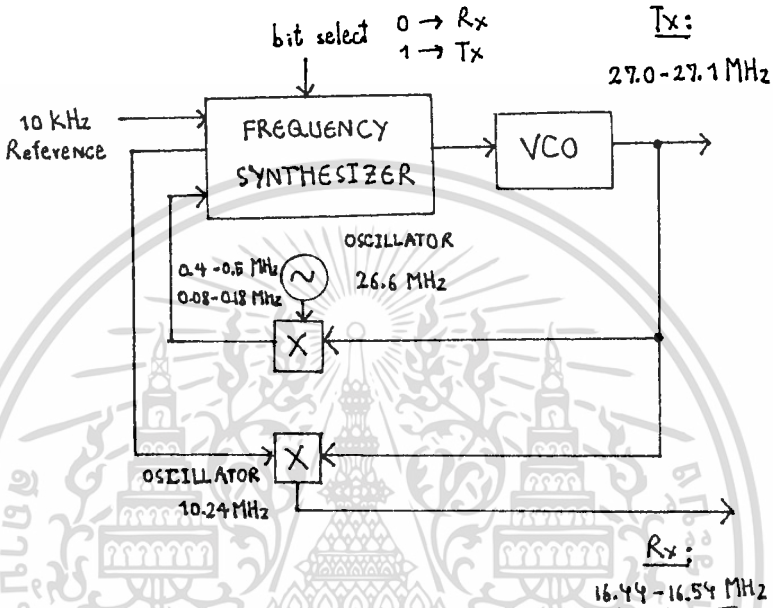
รูปที่ 2.11 บล็อคไดอะแกรมของวงจรสังเคราะห์ความถี่ที่ออกแบบ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 3

การออกแบบและคำนวณ

3.1) วงจรเฟสล็อคลูป



รูปที่ 3.1 บล็อคไดอะแกรมของวงจรเฟสล็อคลูปที่สร้าง

จากรูป วงจรสังเคราะห์ความถี่แบบเฟสล็อคลูปที่ใช้งานจะแบ่งช่องสัญญาณการติดต่อออกเป็น 2 กลุ่มเพื่อแยกเป็นความถี่ในการส่งและรับเป็นระบบดูเพล็กซ์(duplex) โดยได้กำหนดช่องสัญญาณในการใช้งานไว้ 10 ช่องสัญญาณคือ

ความถี่ส่ง : 27.00 - 27.10 MHz

ความถี่รับ : 26.68 - 26.78 MHz

(ทางด้านรับนั้นเมื่อมาผ่านวงจรมิกเซอร์กับความถี่ 10.24 Mhz จะได้ความถี่ซึ่งเมื่อไปผ่านวงจรมิกเซอร์กับความถี่ที่ส่งมาจากสถานีแม่จะได้ความถี่ไอเอฟ (Intermediate Frequency) 10.7 MHz ดังนั้นความถี่จากสถานีแม่จะเป็น

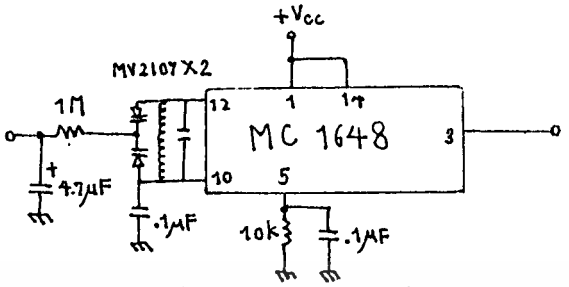
ความถี่ที่ส่งมาจากสถานีแม่ : 27.14 - 27.24 MHz)

เราจะแยกอธิบายการออกแบบส่วนต่างๆได้คือ

(1) วงจรโวลเตจคอนโทรลออสซิลเลเตอร์ (VCO)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ใช้ไอซีเบอร์ MC 1648 ซึ่งเป็นไอซีตระกูลอีซีแอล (ECL)



รูปที่ 3.2 วงจรวีซีโอที่ออกแบบ

จากรูป ความถี่ที่ได้จะคำนวณจากอุปกรณ์ที่ประกอบภายนอกได้แก่ ขดลวดเหนี่ยวนำ (L), คาปาซิเตอร์ (C) และวาริแคป โดยมีสูตรการคำนวณหาความถี่ดังนี้

$$F = 1/(2\pi\sqrt{LCt}) \quad Hz \qquad (3\text{-}1)$$

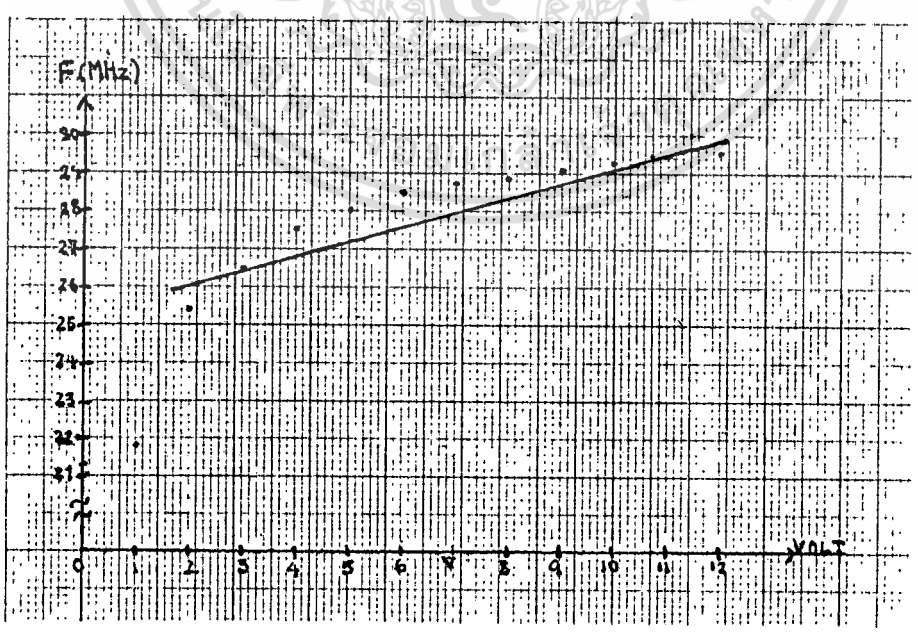
$$Ct = Cd + Cs + C \qquad (3\text{-}2)$$

Cd เป็นค่าคาปาซิแตนซ์ของวาริแคป ซึ่งเปลี่ยนตามโวลเตจที่ตกคร่อม

Cs เป็นค่าคาปาซิแตนซ์ที่ต่ออยู่ภายนอก

C เป็นค่าคาปาซิแตนซ์ของไอซี ประมาณ 6 PF

จากการทดลองปรับค่า L และ Cs ให้เหมาะสม เราสามารถพล็อตกราฟแสดงความสัมพันธ์ระหว่างอินพุทโวลเตจและความถี่เอาร์พุท ได้ดังรูปที่ 3.3



รูปที่ 3.3 กราฟแสดงความสัมพันธ์ระหว่างความถี่ และโวลท์เตจที่ควบคุม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

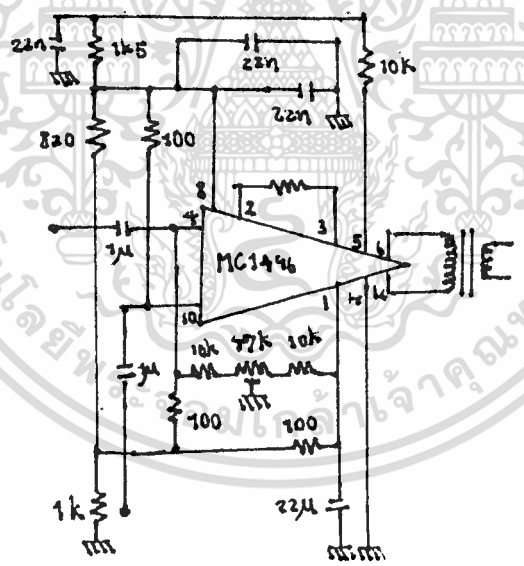
(2) วงจรมิกเซอร์ (MIXER)

ได้ใช้ไอซีเบอร์ MC 1496 ซึ่งเป็นไอซีบาลานซ์มอดดูเลเตอร์ (Balance Modulater) ซึ่งในวงจรที่ใช้งานจะมีวงจรมิกเซอร์อยู่ 2 ชุด ชุดหนึ่งใช้กับภาครับ ส่วนอีกชุดหนึ่งจะเป็นการลดความถี่ที่ใช้งานลงมา เพื่อผ่านไปยังวงจรสังเคราะห์ความถี่ โดยจะมิกซ์กับความถี่ 26.6 MHz ซึ่งจะใช้ความถี่ด้าน Lower Sideband ผลที่ได้คือ

ด้านส่ง ได้ความถี่ 0.4 - 0.5 MHz

ด้านรับ ได้ความถี่ 0.08 - 0.18 MHz

วงจรที่ใช้งานแสดงดังรูปที่ 3.4

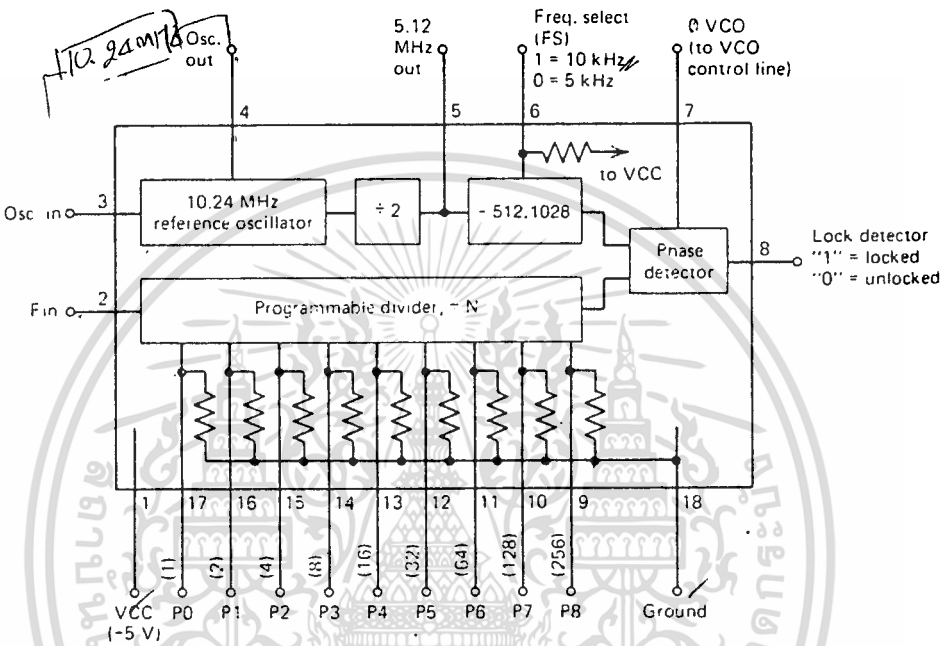


รูปที่ 3.4 วงจรมิกเซอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

(3) ไอซีสังเคราะห์ความถี่

ใช้ไอซีเบอร์ MC 145106 ซึ่งประกอบด้วยฟังค์ชั่นของวงจรสังเคราะห์ความถี่หลายส่วน คือ ตัวสร้างความถี่อ้างอิง , เฟสดีเทคเตอร์ และวงจรหารซึ่งสามารถโปรแกรมค่าได้ โดยมีบล็อคไดอะแกรมแสดงได้ดังรูปที่ 3.5



รูปที่ 3.5 แสดงโครงสร้างไอซีสังเคราะห์ความถี่เบอร์ MC 145106

เนื่องจากต้องการให้แต่ละช่องสัญญาณมีช่วงห่างอยู่ 10KHz ดังนั้นจึงป้อนลอจิก"1"แก่ขา 6 ซึ่งเป็นตัวเลือกวงจรหารของความถี่อ้างอิง

ทางด้านวงจรหารที่โปรแกรมได้นั้น จะเห็นได้ว่าเมื่อเราแปลงค่าฐานสิบของความถี่ทางด้านรับและด้านส่งไปเป็นค่าตัวเลขแบบฐานสอง จะได้

ช่อง 8 - 18 = 001000 - 010010 0.08-0.18MHz

ช่อง 40 - 50= 101000 - 110010 0.4-0.5MHz

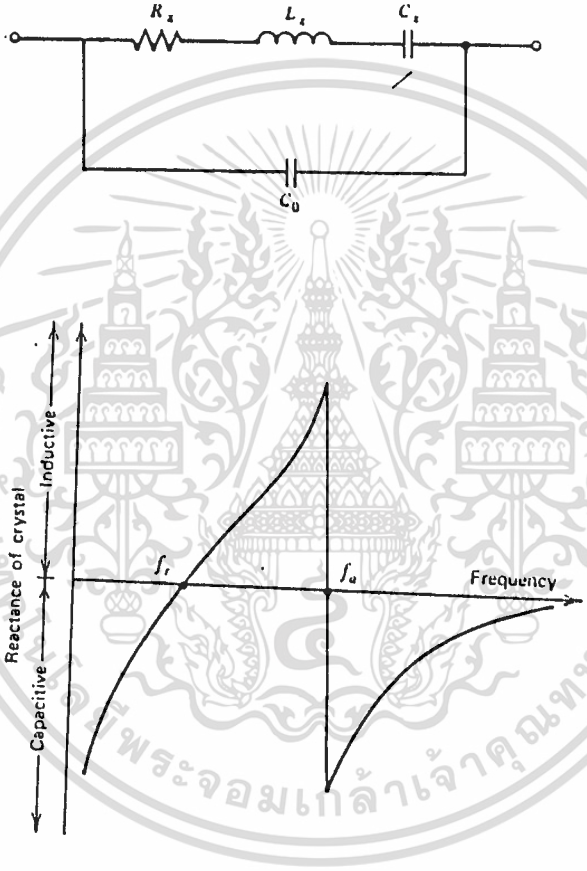
จะเห็นว่าต่างกันที่ดิจิตที่ 6 ดังนั้นเราจะใช้ส่วนนี้เป็นส่วนควบคุมความถี่ในการรับ-ส่ง

นอกจากนั้นเราสามารถใช้ความถี่ค่า 10.24 MHz จากคริสตอลเพื่อใช้สร้างความถี่อ้างอิงโดยได้จากขา 4

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

(4) วงจรออสซิลเลเตอร์

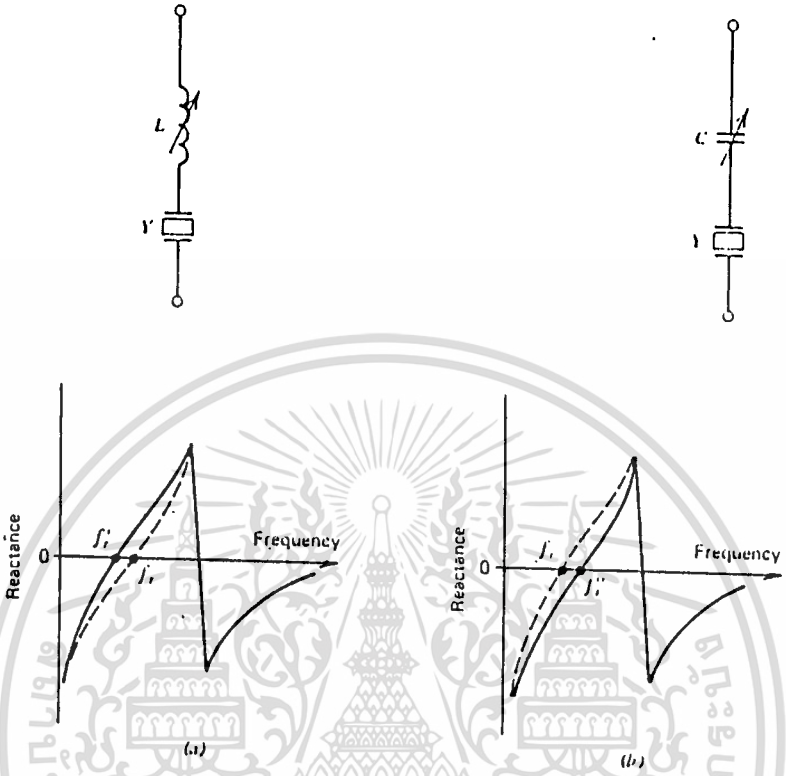
ใช้วงจรคริสตอลออสซิลเลเตอร์เพื่อเสถียรภาพที่ดีของวงจร การใช้แร่คริสตอลในส่วนป้อนกลับแทนวงจรแทงค์จูน เนื่องจากคริสตอลมีคุณสมบัติที่มีค่า Q สูงมาก เราจะแสดงวงจรเสมือน และผลของความถี่ของคริสตอลได้ดังรูปที่ 3.6



รูปที่ 3.6 แสดงวงจรเสมือนของคริสตอลและผลของความถี่ของคริสตอล

บางครั้งความถี่ในการใช้งานอาจจะไม่ตรงกับค่าของคริสตอล ที่มีจำหน่าย ดังนั้นต้องใช้เทคนิคในการพูลลิ่ง(Pulling) ความถี่ของคริสตอล

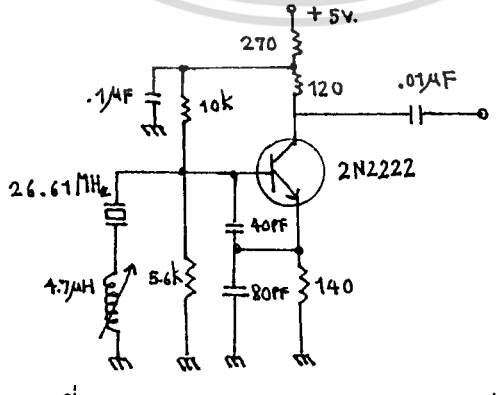
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.7 a) ผลของอินดัคทีฟพูลลิง b) ผลของคาพาซิทีฟพูลลิง

จากรูปจะแสดงผลของค่ารีแอคแตนซ์ กับความถี่รีโซแนนท์ของคริสตอล โดย รูป a) เป็นแบบอินดัคทีฟพูลลิ่ง b) เป็นแบบคาปาซิทีฟพูลลิ่ง

ดังนั้นวงจรคริสตอลออสซิลเลเตอร์ที่ใช้งาน เป็นดังรูปที่ 3.8

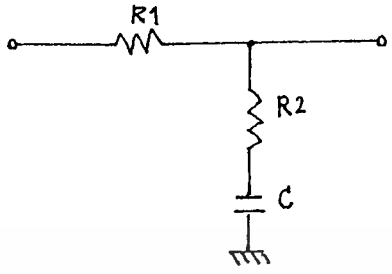


รูปที่ 3.8 วงจรคริสตอลออสซิลเลเตอร์ที่ออกแบบ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

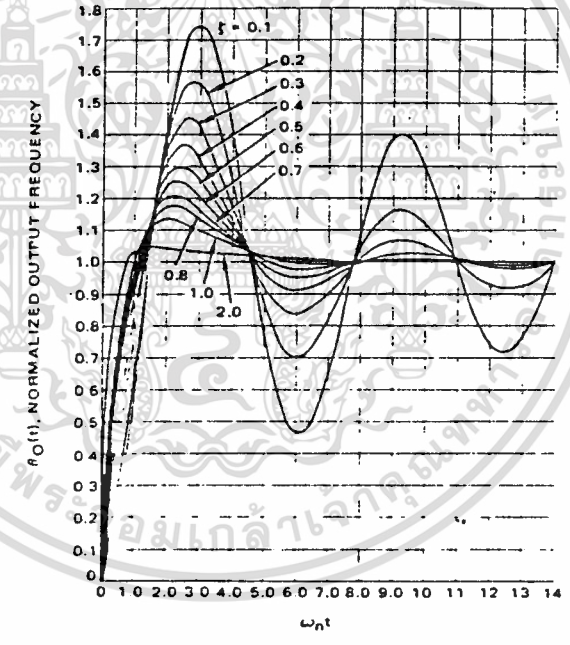
(5) วงจรลูปฟิลเตอร์

วงจรลูปฟิลเตอร์ที่เลือกใช้งานเป็นดังรูปที่ 3.9



รูปที่ 3.9 วงจรลูปฟิลเตอร์ที่ออกแบบ

ค่าของ$W_n t$ และ สามารถกำหนดได้จากกราฟแสดงความสัมพันธ์ดังรูปที่ 3.10



รูปที่ 3.10 กราฟแสดงความสัมพันธ์ของ $\theta(t)$ และ $W_n t$

กำหนดค่า $\delta = 0.8$, $W_n t = 4.5$

กำหนดค่า t = 1 ms ดังนั้น

$W_n = 4500$

N = ค่าที่ใช้หารในส่วนป้อนกลับ (สูงสุด)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

$= 50$

$K_v = 0.9$ Mhz/V (จากกราฟของวีซีโอจากหัวข้อที่ 1) (3-3)

$K_\phi = Vdd/4$ (3-4)

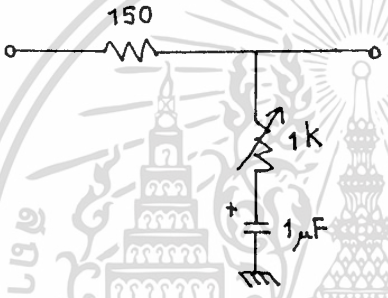
$= 5/4\pi = 0.4$

กำหนดค่า $C = 1\ \mu F$

จะได้ $R1 = 150\ \Omega$

$R2 = 216\ \Omega$ (ใช้ R ปรับค่าได้)

ดังนั้นวงจรลูปฟิลเตอร์ที่ออกแบบจะเป็นดังรูปที่ 3.11



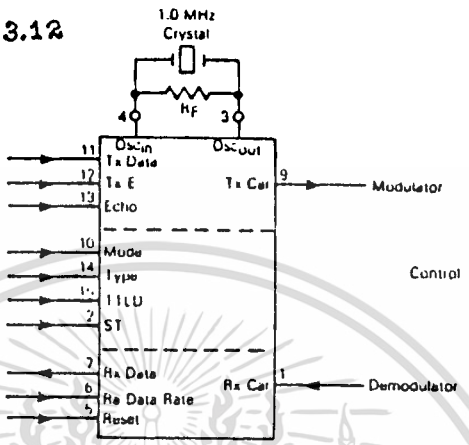
รูปที่ 3.11 วงจรลูปฟิลเตอร์ที่ใช้งาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.2) <u>โมเด็ม</u>

(1) ไอซีโมเด็ม

ใช้ไอซีเบอร์ MC 14412 ซึ่งเป็นโลว์สปีดโมเด็ม (Low Speed Modem) โดยการไบอัลขาต่างๆเป็นดังรูปที่ 3.12



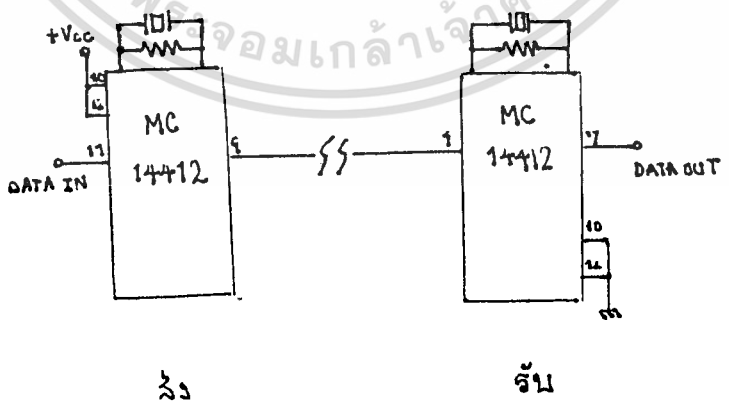
รูปที่ 3.12 โครงสร้างของไอซีโมเด็มเบอร์ MC 14412

รูปแบบการใช้งาน คือ

(1.1) ใช้มาตรฐานของ CCITT

(1.2) คู่ความถี่ที่ใช้เป็นของแชนแนล 1 โดยลอจิค 1 เป็นความถี่ 980 Hz และลอจิค 0 เป็นความถี่ 1180 Hz

จากการออกแบบวงจรไอซีทั้งด้านส่งและด้านรับ เป็นดังรูปที่ 3.13

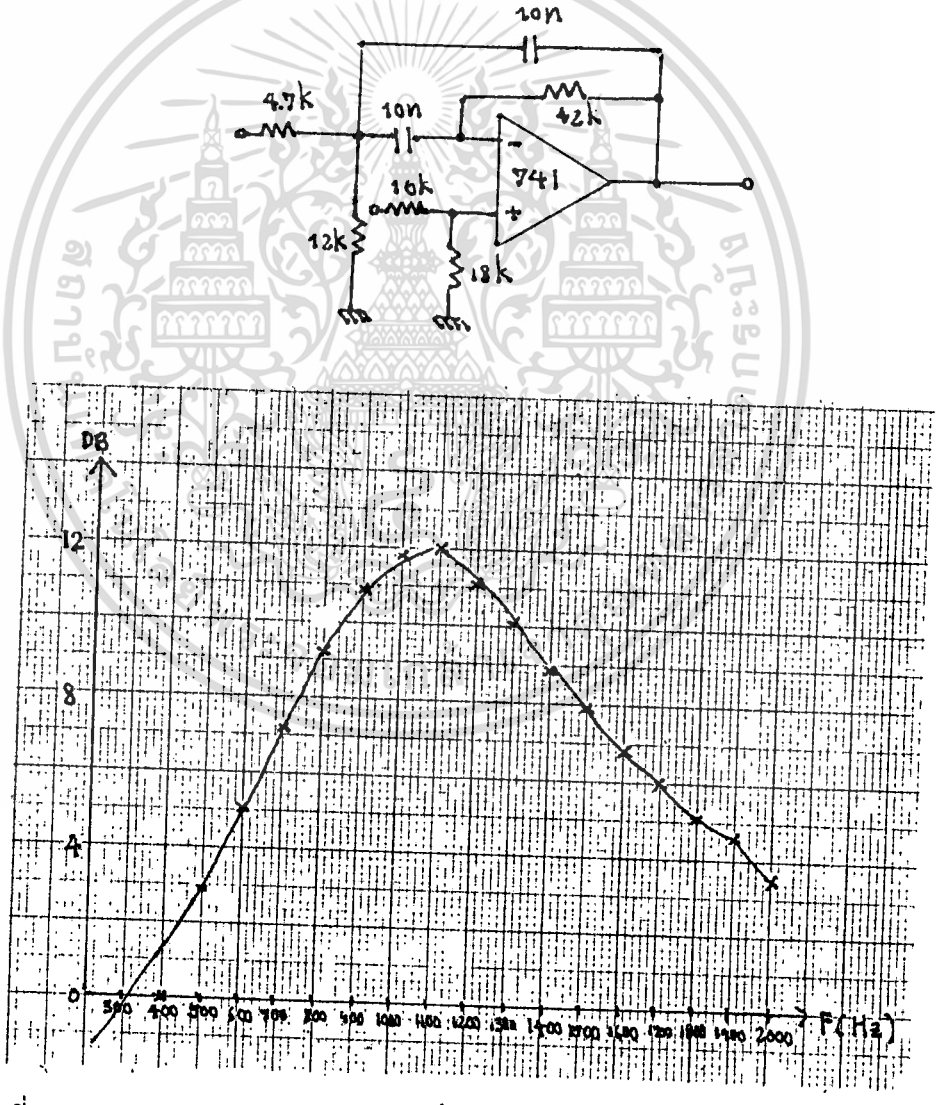


รูปที่ 3.13 การออกแบบโมเด็มด้านส่งและด้านรับ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

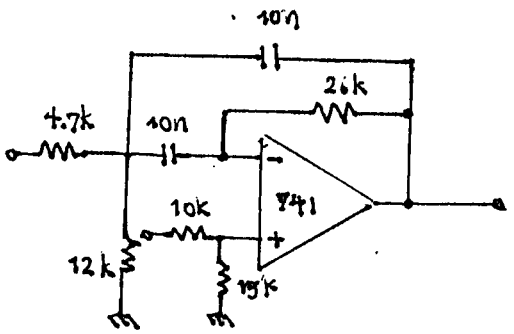
(2) วงจรฟิลเตอร์

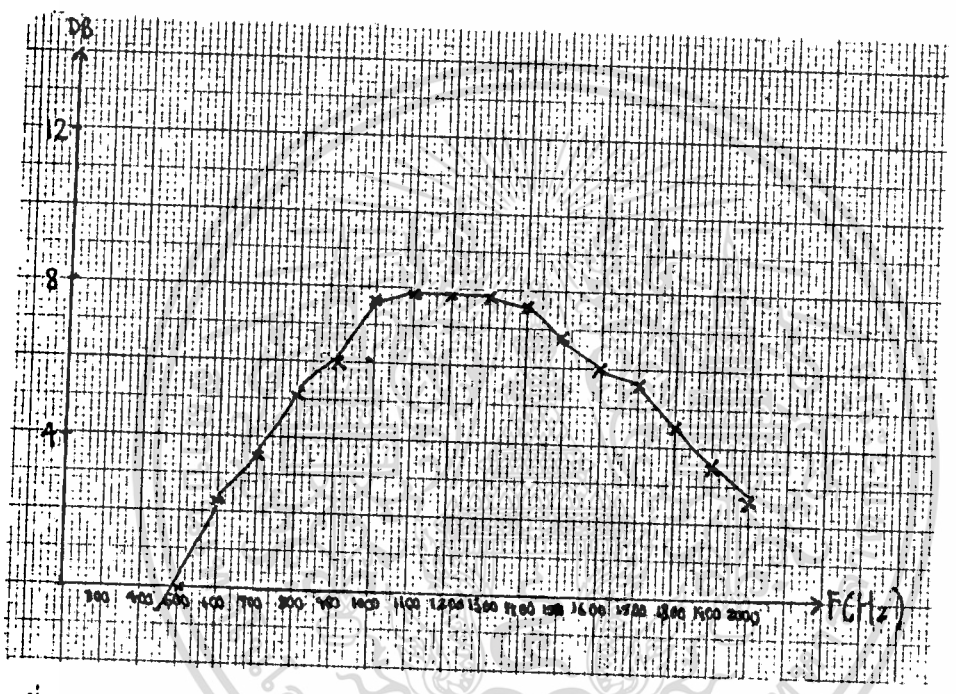
เนื่องจากวงจรฟิลเตอร์แบบพาสซีฟจะมีการลดทอนเกิดขึ้น ดังนั้นเราสามารถออกแบบวงจรเป็นแอคทีฟฟิลเตอร์ ซึ่งจะประกอบไปด้วยตัวต้านทาน,คาปาซิเตอร์ และอุปกรณ์แอคทีฟ คือ ออปแอมป์ ดังนั้นจึงลักษณะเป็นวงจรขยายไปในตัวด้วย วงจรฟิลเตอร์ที่ใช้งานทั้งในด้านส่งและรับ เป็นวงจรแบนด์พาสฟิลเตอร์ชนิดมัลติเปิ้ลฟิดแบ็ค (Multiple-Feedback Band Pass Filter) ในการออกแบบนั้นได้ใช้กราฟช่วยในการออกแบบ (เอกสารอ้างอิงหมายเลข 5) เราสามารถพล็อตกราฟแสดงผลตอบสนองต่อความถี่ของวงจรฟิลเตอร์ที่ใช้งานทั้งด้านรับและส่งได้ดังรูปที่ 3.14 และ 3.15



รูปที่ 3.14 วงจรฟิลเตอร์ของโมเด็มด้านส่งและกราฟแสดงผลตอบสนองต่อความถี่

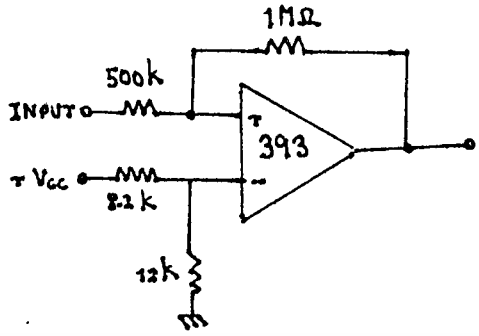
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้





รูปที่ 3.15 วงจรฟิลเตอร์ของโมเด็มด้านรับและกราฟแสดงผลตอบสนองต่อความถี่

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.17 วงจรคอมพาราเตอร์ที่ออกแบบ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

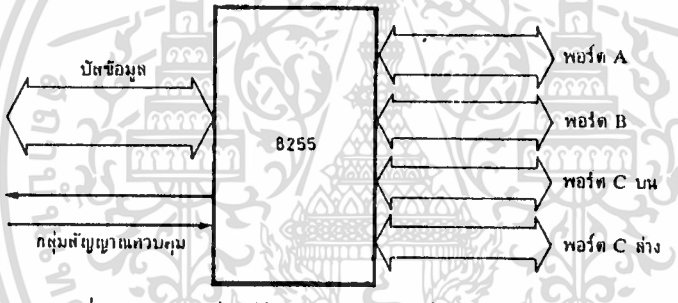
3.3) ส่วนควบคุม และประมวลผลของสถานีแม่

(1) พอร์ตแบบขนาน

ไอซีเบอร์ 8255 เป็นไอซีในตระกูลของซีพียู 8080 แต่นิยมนำมาประยุกต์ใช้งานร่วมกับไมโครโปรเซสเซอร์ Z80 เนื่องจากใช้งานได้ง่าย มีราคาถูกและหาได้ง่าย

ลักษณะโดยทั่วไป

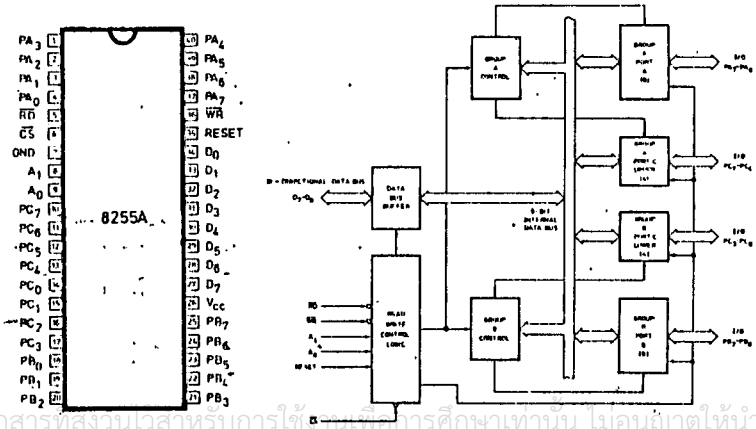
ไอซี 8255 มี40ขา ไอซีเบอร์นี้ได้ถูกออกแบบมาให้ใช้กับไมโครโปรเซสเซอร์ 8080 แต่ซีพียูเบอร์ Z80 มีความคล้ายคลึงกับซีพียู 8080 มาก จึงสามารถนำไอซีเบอร์นี้มาเป็นอุปกรณ์สนับสนุนให้กับซีพียู Z80ได้เช่นกัน ไอซี 8255 สามารถต่อเป็นพอร์ตให้ไมโครโปรเซส-เซอร์ได้ 3พอร์ต โดยมีโครงสร้างดังบล็อกไดอะแกรมรูปที่ 3.18



รูปที่ 3.18 บล็อคไดอะแกรมของไอซี 8255

พอร์ตทั้ง 3ของ 8255 เรียกว่าพอร์ต A,BและC เฉพาะพอร์ตCจะแยกออกเป็น 2ส่วน คือPC0-PC3เรียกว่าพอร์ตCล่างจำนวน4บิทและพอร์ทCบนคือPC4-PC7 พอร์ตทั้ง3จะทำหน้าที่เป็นพอร์ตอินพุตหรือเอาต์พุตหรือทั้ง2กรณี แล้วแต่โปรแกรมที่เรากำหนด

โครงสร้างวงจรภายในตัวไอซี 8255และการจัดวางขาต่างๆแสดงดังรูปที่ 3.19



รูปที่ 3.19 โครงสร้างภายในของไอซี 8255 และการวางขา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ขาสัญญาณต่างๆของ 8255

D0-D7 คือขาสัญญาณข้อมูล ขาเหล่านี้จะต่อเข้ากับระบบบัสข้อมูลของไมโครโปร-เซสเซอร์ เพื่อให้ไมโครโปรเซสเซอร์สามารถอ่านหรือเขียนข้อมูลจากพอร์ตผ่านทางบัสเหล่านี้

$\overline{CS}$ เป็นขาอินพุตซึ่งจะรับสัญญาณจากภายนอกเพื่อเลือกชิพ 8255 โดยเมื่อขานี้มีลอจิกเป็น 0 จะทำให้ 8255 ต่อเข้ากับระบบบัสของซีพียู

$\overline{RD}$ ขาสัญญาณการอ่าน เป็นสัญญาณที่ส่งมาจากซีพียู เมื่อสัญญาณที่ขานี้เป็น 0 และ $\overline{CS}$ เป็น 0 ด้วย 8255 จะทำตัวให้ซีพียูอ่านข้อมูลจากบัสในขณะเป็นพอร์ตอินพุต

$\overline{WR}$ ขาสัญญาณการเขียน จะแอคทีฟเมื่อสัญญาณ $\overline{WR}$ เป็น 0

A0-A1 ขาแอดเดรส ลอจิกของขาทั้งจะถอดรหัสออกได้4ค่าเพื่อกำหนดรีจิส-เตอร์ภายในที่เชื่อมต่อกับพอร์ตอินพุต-เอาต์พุตของ 8255

$\overline{RESET}$ ขารีเซ็ต เป็นสัญญาณที่ส่งมาจากภายนอก เข้ามาทำการรีเซ็ตสถานะต่างๆของ 8255

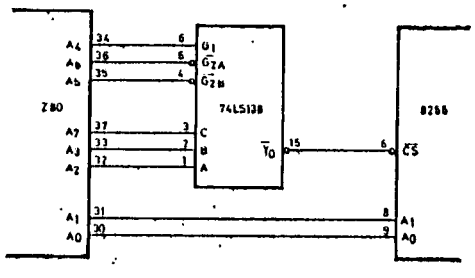
PA0-PA7 เป็นสายสัญญาณของพอร์ตA

PB0-PB7 เป็นสายสัญญาณของพอร์ตB

PC0-PC7 เป็นสายสัญญาณของพอร์ตCซึ่งแบ่งเป็น2กลุ่มคือ กลุ่มPC0-PC3และกลุ่ม PC4-PC7

การต่อ 8255 เข้ากับ Z80

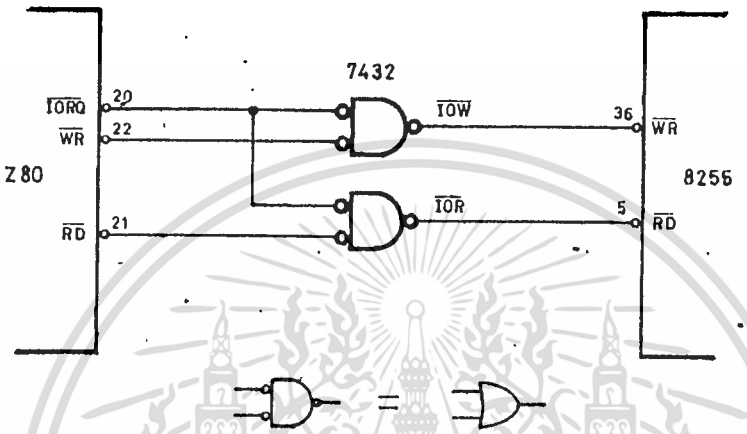
พิจารณาตัวอย่างการต่อ 8255 เป็นพอร์ตให้กับ Z80 โดยสมมุติให้ Z80 มองเห็น 8255 เป็นพอร์ตหมายเลข 10H, 11H, 12Hและ13H การเชื่อมต่อสายสัญญาณการเลือกแอดเดรสของพอร์ตทำได้ดังรูปที่ **3.20**



รูปที่ 3.20 การเลือกแอดเดรสของพอร์ท

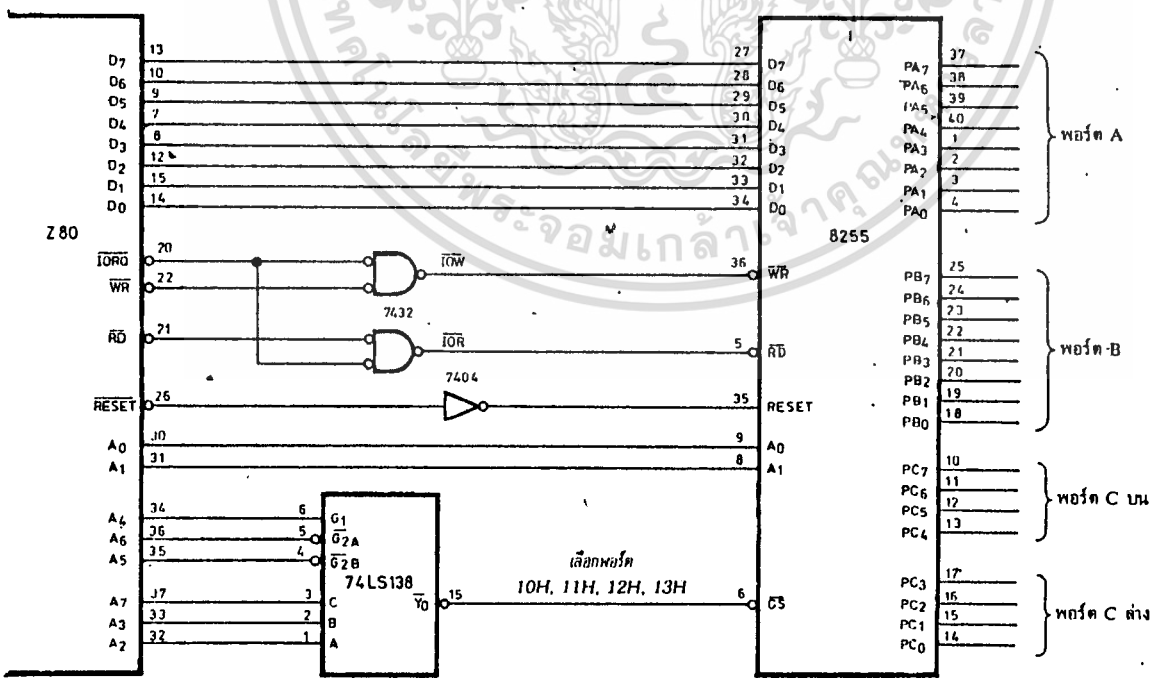
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สัญญาณที่จะควบคุม 8255 อีกชุดหนึ่ง คือสัญญาณควบคุมการเขียน($\overline{WR}$)และการอ่าน($\overline{RD}$) และเพื่อให้แยกกันระหว่างการเขียน-อ่านหน่วยความจำกับการเขียน-อ่านพอร์ตอินพุต-เอาต์พุตจึงต้องใช้ขา $\overline{IORQ}$ ร่วมด้วย ลักษณะการเชื่อมต่อของสัญญาณส่วนควบคุมแสดงดังรูปที่ 3.21



รูปที่ 3.21 การเชื่อมต่อของสัญญาณการเขียนและอ่านพอร์ท

ระบบการเชื่อมต่อ 8255 เข้ากับ Z80 ทั้งระบบแสดงดังรูปที่ 3.22



รูปที่ 3.22 การเชื่อมต่อ Z-80 และ 8255

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

<u>รีจิสเตอร์ภายในของ 8255</u>

การที่ 8255 มีพอร์ตที่ Z80 มองเห็น4พอร์ต โดยแต่ละพอร์ตจะเห็นเป็นเสมือนรีจิสเตอร์ที่เขียน-อ่านได้ และรีจิสเตอร์แต่ละตัวนี้จะถูกกำหนดด้วยแอดเดรสที่กำหนดไว้ เช่นกรณีกำหนดเป็นแอดเดรส10H, 11H, 12Hและ13H รีจิสเตอร์แต่ละตัวจะได้รับการกำหนดควบคู่กับ$\overline{RD}$และ$\overline{WR}$เพื่อแสดงความหมาย เช่นพอร์ต10Hเป็นพอร์ตA(A0="0",A1="0") ซึ่งเมื่อเขียนที่พอร์ตนี้($\overline{WR}$=0) จะเป็นการส่งข้อมูลเอาต์พุต และถ้าอ่านพอร์ตนี้($\overline{RD}$=0)จะเป็นการอินพุตจากพอร์ตนี้

การจะใช้งาน 8255 ต้องส่งรหัสควบคุม(control byte)เข้าไปยังพอร์ตข้อมูลควบคุม เพื่อควบคุมการทำงานของ 8255 โดยใช้สัญญาณควบคุมพอร์ตหมายเลข13H การทำงานของ 8255 มีอยู่3โหมดแล้วแต่โปรแกรมที่ส่งมาควบคุม ได้แก่โหมด 0, โหมด 1 และโหมด 2

การโปรแกรม 8255 คือการให้ค่ารหัสบิตต่างๆเข้าไปในรหัสควบคุมแล้วส่งไปยังรีจิสเตอร์ของพอร์ตควบคุม โดยความหมายของบิตต่างๆมีดังนี้

บิตD7 เป็นบิตแสดงรหัสคำสั่งควบคุม ถ้าบิตนี้เป็น"1"จะหมายถึงว่ารหัสควบคุมนี้จะมีผลต่อการเปลี่ยนแปลงการเซ็ตโหมดต่างๆของ 8255

บิตD6และD5 เป็นการเลือกโหมดของพอร์ตA

บิตD4 เป็นการกำหนดอินพุต-เอาต์พุตของพอร์ตA

บิตD3 เป็นบิตที่บอกการเซ็ตของพอร์ตCบน

บิตD2 เป็นบิตที่บอกการเซ็ตโหมดของพอร์ตB

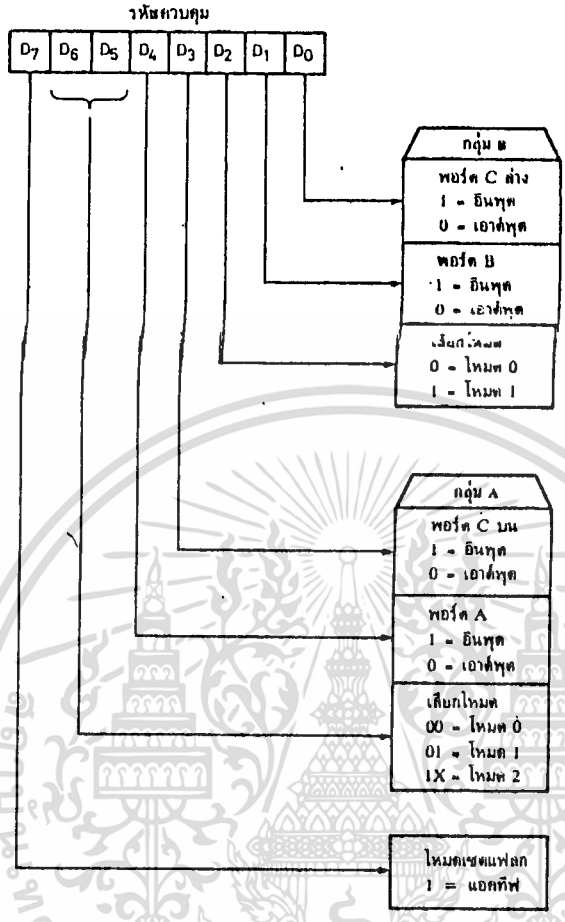
บิตD1 เป็นการกำหนดอินพุต-เอาต์พุตของพอร์ตB

บิตD0 เป็นการกำหนดอินพุต-เอาต์พุตของพอร์ตCล่าง

รายละเอียดความหมายของบิตต่างๆของรหัสควบคุม(control byte)พิจารณาได้ตามรูปที่ **3.23**

การโปรแกรม 8255 เริ่มโดยการกำหนดรหัสควบคุมที่ต้องการแล้วเอาต์พุตไปยังพอร์ตควบคุม ผลก็คือพอร์ตA,BและCจะมีสภาพเป็นไปตามที่เรากำหนด ซึ่งจะทำให้รับหรือส่งข้อมูลต่างๆโดยผ่านพอร์ตเหล่านี้ได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.23 รหัสควบคุมของไอซี 8255

การทำงานในโหมด0ของ 8255

โหมด0เป็นโหมดที่กำหนดให้พอร์ตทุกพอร์ตใน 8255 เป็นพอร์ตอินพุต-เอาต์พุตแบบพื้นฐาน รูปแบบความเป็นไปได้จึงมีทั้งสิ้น16รูปแบบ ตามลักษณะของพอร์ตA,B,CบนและCล่าง ลักษณะของรหัสควบคุมแต่ละแบบจะเป็นดังรูปที่ **3.24**

การทำงานในโหมด1ของ 8255

โหมด1เป็นโหมดที่ทำให้อินพุต-เอาต์พุตมีการตรวจสอบสัญญาณ(handshaking) โดยใช้อินพุต-เอาต์พุตของพอร์ตAและBเป็นหลัก และใช้พอร์ตCบนเป็นสัญญาณhandshakeของพอร์ตA ส่วนพอร์ตCล่างเป็นสัญญาณhandshakeของพอร์ตB

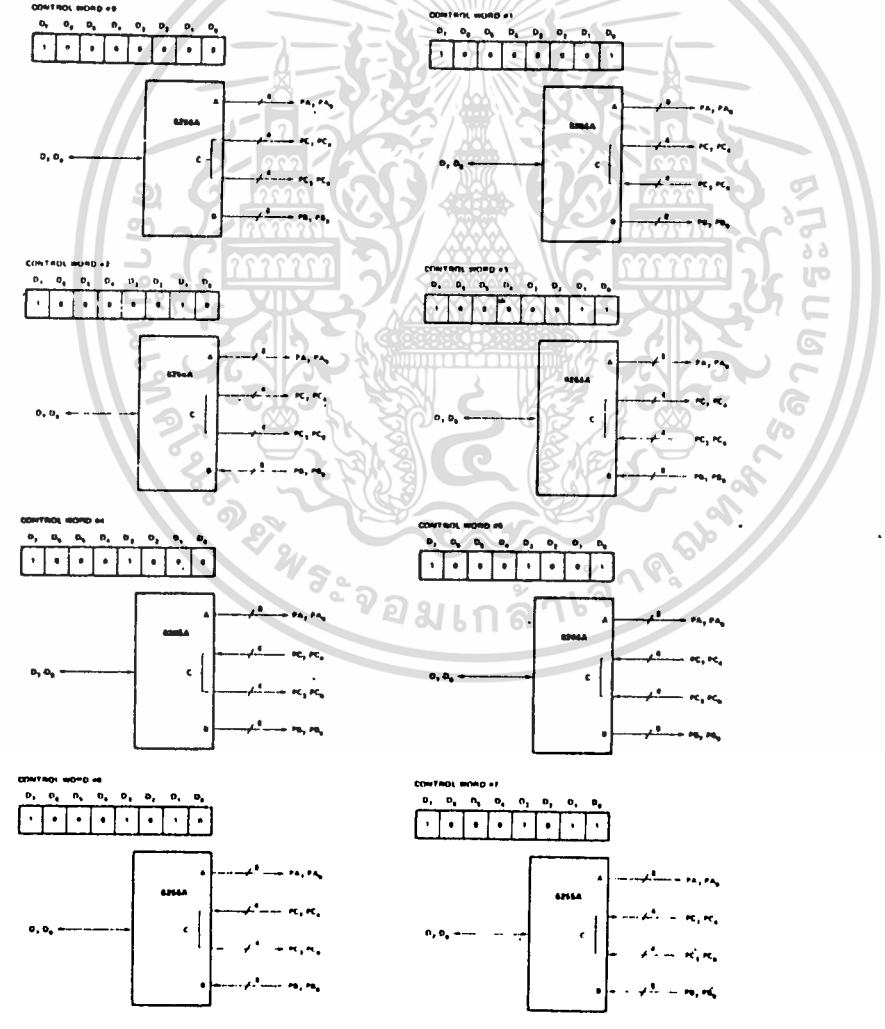
เหตุผลที่มีการใช้อินพุต-เอาต์พุตแบบhandshakeก็เพื่อให้การซิงโครไนซ์ระหว่างอุปกรณ์ภายนอกที่ทำงานได้ช้ากับการทำงานของคอมพิวเตอร์ที่ทำงานได้เร็ว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การทำงานในโหมด2ของ 8255

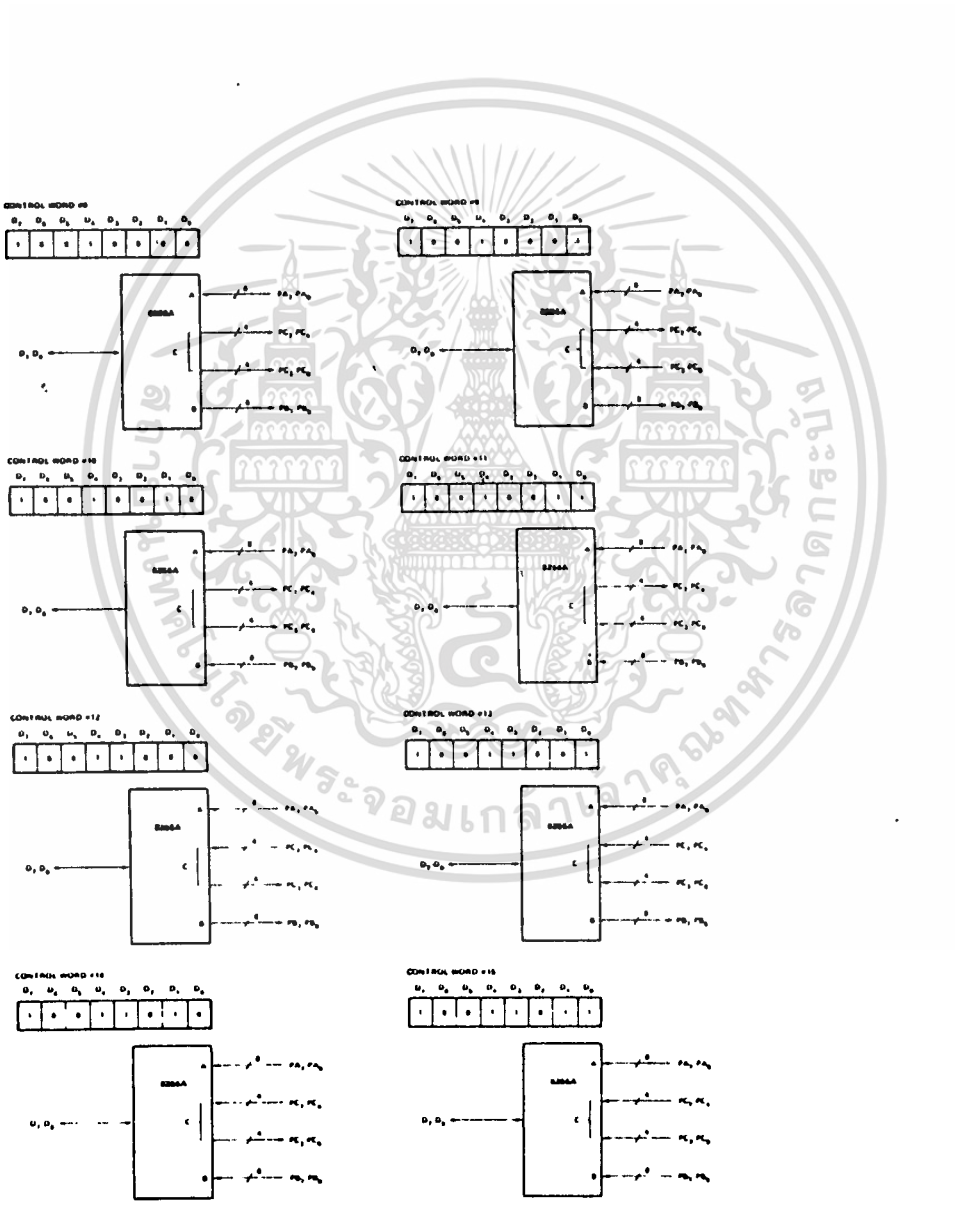
โหมด2นี้จะใช้ได้กับเฉพาะพอร์ตA ในโหมดนี้ 8255 จะใช้พอร์ตAทำหน้าที่เป็นพอร์ตแบบสองทิศทาง คือสามารถเป็นได้ทั้งอินพุตและเอาต์พุตซึ่งมีการตรวจสอบสัญญาณ (handshake)ทั้งคู่ ส่วนพอร์ตCจะทำหน้าที่เป็นสัญญาแตรวจสอบ

เมื่อพอร์ตAถูกโปรแกรมเป็นโหมด2แล้ว พอร์ตBจะต้องโปรแกรมเป็นโหมด0หรือโหมด1ก็ได้ ซึ่งทำให้ทำงานแยกอิสระจากกัน



รูปที่ 3.24 การโปรแกรมการทำงานของ 8255 ในโหมดต่างๆ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.24 (ต่อ)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

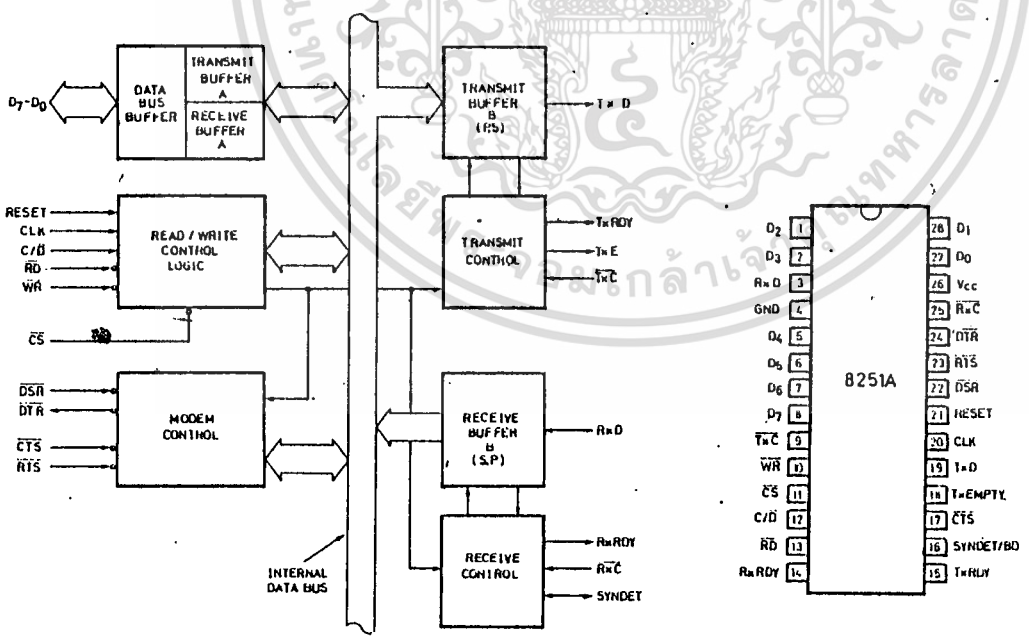
(2) พอร์ตอนุกรม

ไอซี 8251

8251 เป็นชิพที่มีใช้มานานแล้วตั้งแต่ยุคแรกๆของไมโครโปรเซสเซอร์จนถึงปัจจุบัน เป็นไอซีที่ได้รับความนิยมมากในการต่อพอร์ตใช้งานแบบอนุกรมกับไมโครโปรเซสเซอร์ตระกูลใดๆก็ตาม

ลักษณะโดยทั่วไป

ชิพ 8251 เป็นชิพที่ทำหน้าที่เป็นตัวรับและส่งข้อมูลระหว่างไมโครโปรเซสเซอร์กับอุปกรณ์อินพุต-เอาต์พุตแบบอนุกรม ชิพนี้ทำงานได้ทั้งโหมดอะซิงโครนัสและซิงโครนัส แต่ที่ประยุกต์ใช้กันมากเป็นแบบอะซิงโครนัส ชิพ 8251 เป็นไอซีขนาด28ขา มีการจัดขาและโครงสร้างภายในดังรูปที่ 3.25



รูปที่ 3.25 โครงสร้างภายในของไอซี 8251 และการวางขา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ขาสัญญาณต่างๆของ 8251

ขาสัญญาณที่ติดต่อกับซีพียู

DØ-D7 เป็นขาสัญญาณข้อมูลซึ่งต่อเข้ากับระบบบัสข้อมูลของซีพียู

RESET เป็นขาสัญญาณอินพุตซึ่งแอคทีฟที่ลอจิก1

CLK(clock) เป็นขาสัญญาณอินพุตที่ใช้เพื่อให้การถ่ายทอดข้อมูลภายในระบบมีการซิงโครไนซ์กัน ขาสัญญาณนี้ต่อเข้ากับ CLK ของ Z8Øได้

$\overline{WR}$ เป็นขาสัญญาณอินพุตใช้ในการเขียนข้อมูลมายังพอร์ต แอคทีฟที่ลอจิกØ

$\overline{RD}$ เป็นขาสัญญานอินพุตใช้เมื่อซีพียูต้องการอ่านข้อมูล แอคทีฟที่ลอจิกØ

C/$\overline{D}$ ขาสัญญาณcontrol/dataใช้ร่วมกับ $\overline{RD}$และ$\overline{WR}$ ถ้ามีค่าลอจิกเป็น1จะเป็นสัญญาณควบคุมใช้เขียน-อ่านรหัสควบคุม แต่ถ้าเป็นØจะใช้ในการเขียน-อ่านข้อมูล

$\overline{CS}$ ขาสัญญาณการเลือกชิพ8251 แอคทีฟที่ลอจิกØ

ขาสัญญาณที่ติดต่อกับโมเด็ม

$\overline{DSR}$ (Data Set Ready) เป็นสัญญาณอินพุตเข้า 8251 สถานะของ $\overline{DSR}$ ("Ø"หรือ"1")สามารถตรวจสอบได้โดยโปรแกรมของซีพียู จึงสามารถใช้สัญญาณ$\overline{DSR}$เป็นสัญญาณพอร์ตอินพุตขนาด1บิตได้ แต่ในระบบโมเด็ม $\overline{DSR}$จะเป็นสัญญาณตอบสนองความพร้อมจากโมเด็ม

$\overline{DTR}$ (Data Terminal Ready) เป็นสัญญาณเอาต์พุตจาก 8251 สถานะของ $\overline{DTR}$สามารถควบคุมโดยโปรแกรมจากซีพียูได้ จึงใช้เป็นพอร์ตเอาต์พุตขนาด1บิตได้ แต่ในงานโมเด็มจะใช้ $\overline{DTR}$เป็นสัญญาณแสดงบอกโมเด็มถึงความพร้อมของ 8251

$\overline{RTS}$ (Request To Send) เป็นสัญญาณเอาต์พุตจาก 8251 สถานะของ $\overline{RTS}$ สามารถควบคุมได้โดยโปรแกรมจากซีพียู เราจึงสามารถใช้ $\overline{RTS}$เป็นสัญญาณพอร์ตเอาต์พุตขนาด1บิตได้ แต่ถ้าใช้งานกับโมเด็ม $\overline{RTS}$จะเป็นสัญญาณบอกไปยังโมเด็มเพื่อเตรียมส่งข้อมูล

$\overline{CTS}$ (Clear To Send) เป็นสัญญาณเอาต์พุตจากภายนอก เมื่อ $\overline{CTS}$เป็น"Ø" ชิพ8251จึงจะส่งข้อมูลออกไปได้(TXENในรหัสควบคุมต้องเป็น"1"ด้วย)

ขาสัญญาณที่ติดต่อกับภาครับ-ส่ง

เป็นขาสัญญาณที่ใช้ในการรับ-ส่งข้อมูล แสดงสถานะการทำงานและควบคุมการทำงานของภาครับ-ส่ง

TxRDY (Transmitter Ready) มีความสัมพันธ์กับ Transmit Buffer A

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

และ B ดังนี้ TxRDY จะเป็น"1"เมื่อข้อมูลใน Transmit Buffer A ถูกส่งต่อไปยัง Transmit Buffer B แล้ว นั่นคือ Transmit Buffer B ว่าง พร้อมที่จะรับข้อมูลใหม่จากซีพียูได้ TxRDY จะเป็น"0"เมื่อ Transmit Buffer A รับข้อมูลจากซีพียู แต่มีข้อต้องระวังไว้ว่า บิตข้อมูล TxRDY ในไบต์แสดงสถานะจะไปออกยังสัญญาณ TXRDY ขา15ก็ต่อเมื่อ 8251 ได้รับสัญญาณอีนาเบิ้ลให้ส่งข้อมูลได้ หรือ TxEN ในไบต์แสดงสถานะเป็น"1" และ $\overline{CTS}$ เป็น"0"

TxEMPTY เป็น"1"เมื่อข้อมูลใน Transmit Buffer B ถูกเลื่อนออกไปทางขา TxD หมดแล้ว ซึ่งก็คือ Transmit Buffer B ว่างนั่นเอง TxEMPTY จะเป็น"0"เมื่อ Transmit Buffer B รับข้อมูลมาจาก Transmit Buffer A

$\overline{TxC}$ (Transmit clock) เป็นสัญญาณนาฬิกาฐานเวลาของข้อมูลที่ส่งออกไปในระบบซิงโครนัส ความถี่ของ $\overline{TxC}$ ก็คืออัตราส่ง (baud rate) ของการส่งเลย (1x) แต่ในระบบอะซิงโครนัส ความถี่ของ $\overline{TxC}$ จะสูงเป็น 1,16หรือ64เท่าของอัตราส่งจริง (baud rate)ก็ได้แล้วแต่จะกำหนดโดยโปรแกรม

RxRDY (Receiver Ready) เป็นสัญญาณเอาต์พุตไปยังซีพียู ใช้แสดงว่ามีการรับข้อมูลจากภายนอกเข้ามาอยู่ใน Receive Buffer A แล้ว พร้อมที่จะให้ซีพียูมาอ่านไปได้ ขาสัญญาณ RxRDY นี้อาจจะแทนเป็นสัญญาณอินเตอร์รัพต์ก็ได้ หรือในระบบ polling ตัวซีพียูอาจจะมาตรวจสอบสถานะของ RxRDY ก็ได้โดยขบวนการอ่าน และสัญญาณ RxRDY จะเป็น"0"ทันทีที่ซีพียูอ่านข้อมูลไปแล้ว

$\overline{RxC}$ (Receiver clock) เป็นสัญญาณนาฬิกาฐานเวลาของการรับข้อมูลจากภายนอก มีกฎเกณฑ์เช่นเดียวกับ $\overline{TxC}$ ในการใช้งานของ 8251 สัญญาณฐานเวลาของการรับและส่ง ($\overline{RxC}$ และ $\overline{TxC}$)จะเท่ากันหรือไม่ก็ได้

<u>การต่อ 8251 เข้ากับ Z80</u>

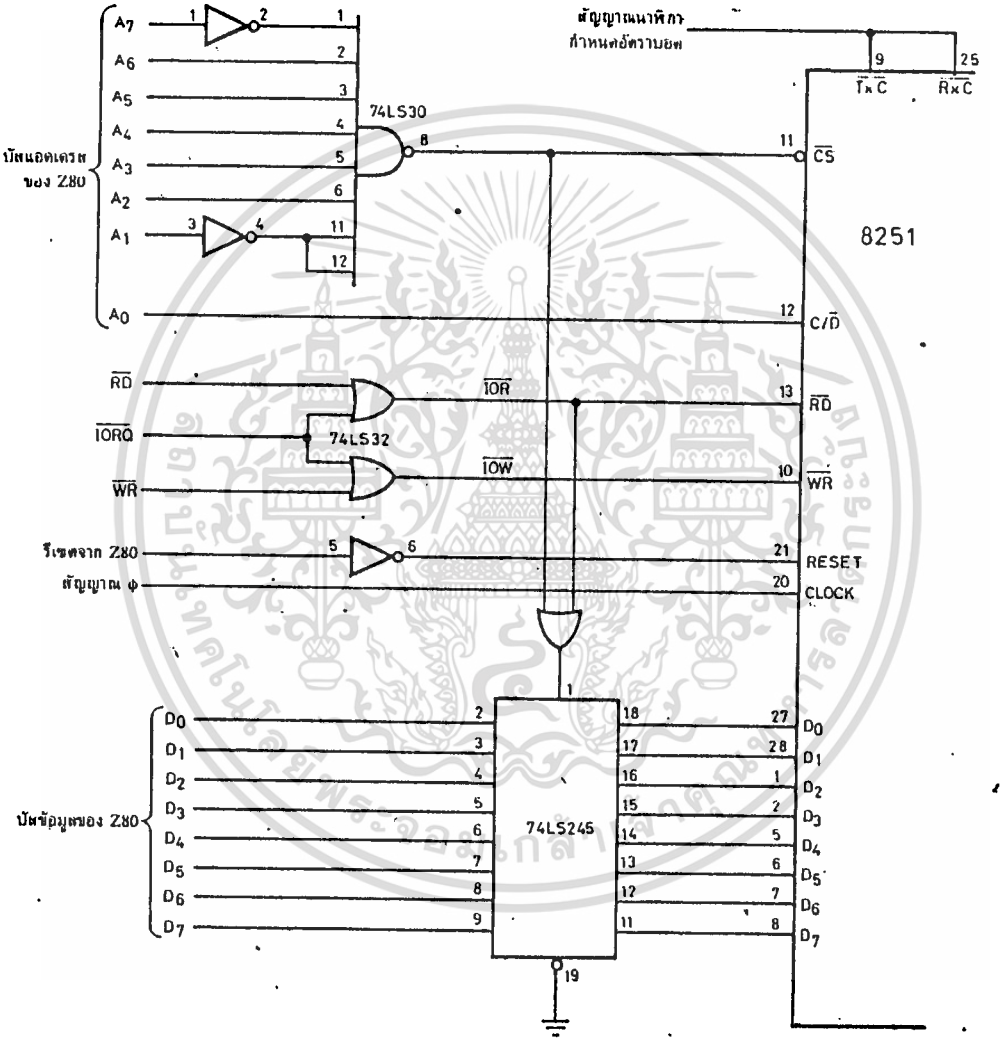
ตัวอย่างการเชื่อมโยงขาสัญญาณ 8251 เข้ากับ Z80 ปรากฏดังรูปที่

สำหรับสัญญาณนาฬิกาที่ป้อนเข้ามาจากภายนอกเพื่อใช้กำหนดอัตราบอดของ 8251 จะต้องมีค่าความถี่ตรงตามต้องการ โดยทั่วไปเรากำหนดการโปรแกรมตัวหารได้ภายใน 8251 ส่วนขา20(ขา CLK)จะต้องมีการป้อนสัญญาณนาฬิกาให้กับ 8251 โดยความถี่ของสัญญาณนี้จะต้องมีค่าไม่น้อยกว่า30เท่าของสัญญาณ $\overline{TxC}$ หรือ $\overline{RxC}$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การโปรแกรม 8251

ก่อนที่จะเข้าใจวิธีการโปรแกรม 8251 จะต้องทำความใจว่าเมื่อขาC/$\overline{D}$เป็น"1" หมายถึงการกำหนดพอร์ตควบคุม ซึ่งในตัวอย่างจากรูปที่2คือพอร์ต7DH โดยถ้าอ่านจากพอร์ต7DH จะได้ไบต์แสดงสถานะ แต่ถ้าอ่านไปยังพอร์ต7DHจะหมายถึงการเซ็ตคำสั่งควบคุมการทำงานของ 8251



รูปที่ 3.26 การเชื่อมต่อไอซี 8251 กับขาสัญญาณต่างๆ

การโปรแกรม 8251 คือการสั่งให้ 8251ทำงานตามรูปแบบที่ระบบเราต้องการ ก่อนที่เราเริ่มการรับ-ส่งทั้งหมด ซีพียูต้องออกคำสั่งมายัง 8251 โดยมีขั้นตอนและประเภทของคำสั่งคือ

1.) สัญญาณ RESET เป็นการเริ่มต้นก่อนคำสั่งใดๆ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.) ทำคำสั่งเลือกขบวนการทำงาน (mode select)

3.) ทำคำสั่งควบคุม (command)

หลังจากสัญญาณ RESET แล้ว ซีพียูต้องออกคำสั่งเลือก จบแล้วจะตามด้วยรหัส SYNC หรือรหัสคำสั่งควบคุม แล้วแต่ว่าจะเลือกการทำงานในแบบซิงโครนัสหรืออะซิงโครนัส โดยซีพียูจะต้องส่งรหัสคำสั่งหรือข้อมูลเป็นลำดับขั้นตอนดังตารางที่ 3.1

ขั้นที่	ขาสัญญาณ	รหัสคำสั่งหรือข้อมูลที่ส่งมาให้ 8251	
1	$C/\overline{D}$ = 1	Mode Instruction	; เลือกโหมด
2	$C/\overline{D}$ = 1	SYNC Charactor 1*	} ทำถ้าเลือกโหมดซิงโครนัส
3	$C/\overline{D}$ = 1	SYNC Charactor 2*	
4	$C/\overline{D}$ = 1	Command Instruction	; รหัสควบคุม
5	$C/\overline{D}$ = 0	Data.....	
6	$C/\overline{D}$ = 1	Command Instruction	ทำเมื่อต้องการเปลี่ยนรหัสควบคุมแต่ไม่เปลี่ยนโหมด
7	$C/\overline{D}$ = 0	Data....	
8	$C/\overline{D}$ = 1	Command Instruction	

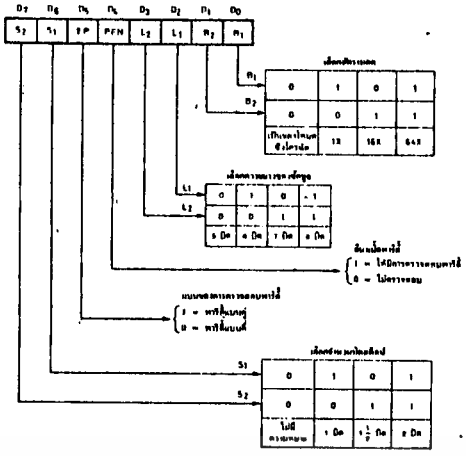
ตารางที่ 3.1

ขั้นตอนที่2และ3ให้ข้ามไปได้เลยถ้าเลือกโหมดอะซิงโครนัส และในขั้นตอนที่3 อาจไม่ต้องทำถ้าเลือกขบวนการซิงโครนัสแบบให้รหัสซิงค์ตัวเดียว สำหรับขั้นตอนที่5และ7หมายถึงการรับหรือส่งข้อมูลต่อเนื่องกันไปจนกว่าจะเปลี่ยนโหมดหรือรหัสควบคุม ส่วนขั้นตอนที่6หรือ8 จะทำถ้าต้องการเปลี่ยนรหัสควบคุมแต่ไม่เปลี่ยนแปลงโหมด

<u>คำสั่งเลือกโหมดและรหัสควบคุม</u>

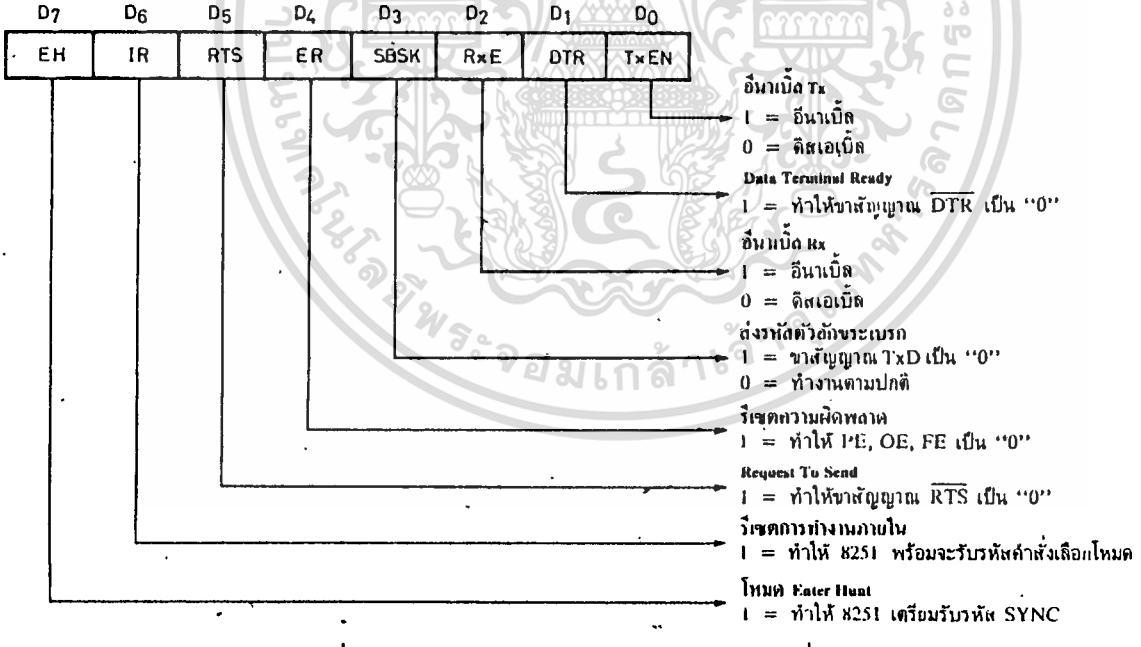
ลักษณะคำสั่งเลือกโหมดทั้ง8บิตของคำสั่งที่ซีพียูส่งมาให้ 8251 เพื่อเริ่มต้นการทำงานจะมีความหมายดังรูปที่3สำหรับโหมดแบบอะซิงโครนัส ต่อจากนั้นจึงเป็นการส่งรหัสควบคุมซึ่งมีความหมายดังรูปที่ 3.27

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.27 รหัสควบคุมของไอซี 8251 ชุดที่ 1

จากรูป สังเกตุการเลือกอัตราบอดของบิตD0และD1 ความหมายของ16xและ 64xคือการนำความถี่ของสัญญาณฐานเวลา $\overline{TxC}$ หรือ $\overline{RxC}$ มาหารด้วย16และ64ตามลำดับแล้วนำไปกำหนดอัตราบอด มีประโยชน์ในกรณีที่ต้องใช้วงจรกำเนิดสัญญาณนาฬิกาความถี่สูงๆแล้วต้องนำมาหารให้เหลือความถี่เท่ากับอัตราบอด การเลือก16xและ64xจะช่วยลดวงจรหารให้น้อยลงได้

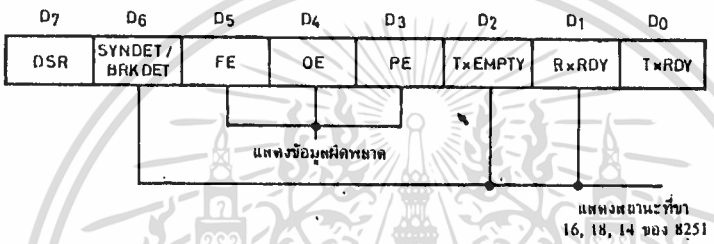


รูปที่ 3.28 รหัสควบคุมของไอซี 8251 ชุดที่ 2

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ไบต์แสดงสถานะ

ไบต์แสดงสถานะนี้มีขนาด8บิต แต่ละบิตจะแสดงสถานะของ 8251 โดยมีความหมายเป็น on กับ off ความหมายของแต่ละบิตเป็นดังรูปที่ 3.29



รูปที่ 3.29 ไบต์แสดงสถานะของไอซี 8251

TxRDY (DØ) จะเป็น"1"ทันทีที่ Transmit Buffer A ว่าง แสดงถึงความพร้อมที่จะรับข้อมูลไปส่งได้แล้วและยังแสดงถึงสัญญาณ TxRDY ที่ขา15อีกด้วย โดยจะมีลอจิกเหมือนขา15ถ้าขาสัญญาณ $\overline{CTS}$ ="Ø" แต่สัญญาณ TxRDY ที่ขา15จะเป็น"Ø"ทันทีที่ขา $\overline{CTS}$ ="1"

RxRDY แสดงถึงการได้รับข้อมูลเข้ามาครบไบต์แล้ว ให้ซีพียูมาอ่านไปได้ และยังมีลอจิกเหมือนกับสัญญาณ RxRDY ที่ขา14อีกด้วย

Parity Error (PE) เท่ากับ"1" เมื่อเกิดการผิดพลาดจากการตรวจสอบบิทพาริตี้ และจะถูกรีเซตเป็น"Ø"โดย ER ในรหัสควบคุมจากซีพียู

Overrun Error (OE) จะเป็น"Ø" ถ้าซีพียูไม่มีการอ่านข้อมูลที่ 8251 ได้รับเข้ามาไว้ใน Receive Buffer A แล้วมีข้อมูลใหม่เข้ามาอีก และ OE จะถูกรีเซตเป็น"Ø" โดย ER ในรหัสควบคุม

TxEMPTY แสดงลอจิกของสัญญาณ TxEMPTY ที่ขา 18 โดยจะเป็น"1" เมื่อ 8251 ว่างหรือไม่มีข้อมูลจะส่ง แต่จะกลับเป็น"Ø" เมื่อได้รับข้อมูลจากซีพียู ในขณะที่ภาคส่งได้รับการอีนาเบิ้ล

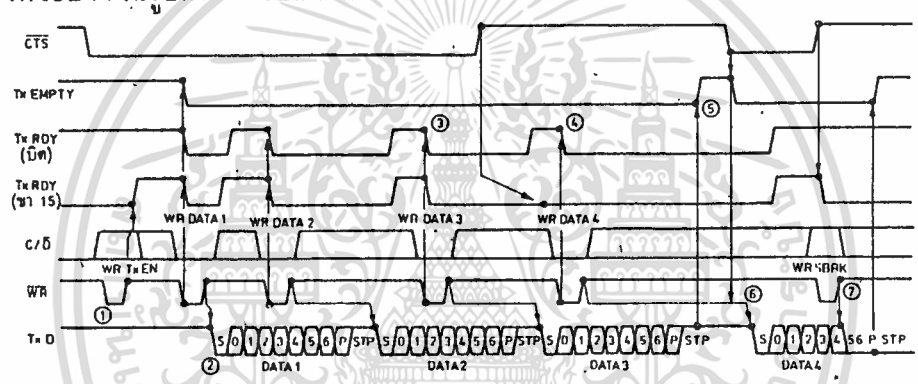
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Framing Error (FE) ใช้ในขบวนการอะซิงโครนัสเท่านั้น โดย FE จะเป็น "1" ถ้าข้อมูลที่รับมาขาดบิตสต๊อปและ FE จะถูกรีเซตเป็น"Ø" โดย ER ในรหัสควบคุม

ทั้ง PE,OE และ FE จะไม่ขัดขวางการทำงานของ 8251 แต่ประการใด เป็นเพียงการเตือนให้ทราบถึงว่าเกิดการผิดพลาดแล้วเท่านั้น

การส่งข้อมูลแบบอะซิงโครนัส

รูปที่ เป็นแผนผังเวลาแสดงการส่งข้อมูลขนาด 6 บิตบวก 1 บิตพาริตี้ บวก 2 บิตสต๊อป สมมติว่าซีพียูได้โปรแกรมเลือกโหมดเป็นแบบอะซิงโครนัสแล้ว ลำดับเวลาการทำงานของตัวอย่างในรูปที่3.30 เป็นดังนี้



รูปที่ 3.30 ไดอะแกรมเวลาของการส่งข้อมูล

1. เริ่มด้วยขา $\overline{CTS}$ เป็น "Ø" อยู่ก่อนแล้ว ซีพียูเขียนรหัสควบคุมมายัง 8251 ทำให้ขา C/$\overline{D}$ เป็น"1" โดยเป็นการส่งรหัสคำสั่ง TxEN เพื่ออีนาเบิ้ลภาคส่ง ทำให้ขา TxRDY เป็น"1" แสดงความพร้อมแล้ว ซีพียูจึงเริ่มส่งข้อมูลตัวแรก (DATA 1) ทำให้ขา TxEMPTY และบิต TxRDY (ในไบต์แสดงสถานะ) และขา TxRDY เปลี่ยนเป็น"Ø" ทั้งหมด

2. ข้อมูล DATA 1 แบบขนานที่เขียนเข้ามา จะถูกเลื่อนออกเป็นแบบอนุกรมทางขา TxD และ TxRDY ทั้งคู่ (ในไบต์แสดงสถานะและขา 15) จะกลับเป็น"1" แสดงว่า Transmit Buffer A ว่าง พร้อมที่จะรับข้อมูลตัวใหม่จากซีพียูอีก ซึ่งซีพียูจะส่งข้อมูลตัวที่ 2 (DATA 2) มาอีก ในขณะนั้น DATA 1 กำลังถูกเลื่อนออกไป เมื่อ DATA 1 ถูกเลื่อนออกไปหมด TxRDY จะกลายเป็น"1" อีกครั้ง

3. ซีพียู จึงส่งข้อมูลตัวที่ 3 (DATA 3) มาได้ ในขณะเดียวกัน DATA 2 จะถูกส่งไปยัง Transmit Buffer B เพื่อให้เริ่มเลื่อนออกไป เมื่อเลื่อน DATA 2 หมดแล้ว DATA 3

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ก็จะถูกเลื่อนออกตามไป ตอนนี้ถึงแม้ $\overline{CTS}$ จะเปลี่ยนเป็น"1" แล้วก็ตาม แต่ DATA 3 ถูกเขียนเข้าไปก่อนแล้ว จึงถูกส่งออกไป แต่ให้สังเกตุว่าเมื่อ $\overline{CTS}$ เป็น"1"และก่อนที่ซีพียูส่ง DATA 4 เข้ามาบิต TxRDY ในไบต์แสดงสถานะจะเป็น"1" แต่สัญญาณ TxRDY ที่ขา15ไม่เป็น "1"

4.เมื่อซีพียูส่ง DATA 4 มาทำให้บิต TxRDYในไบต์แสดงสถานะถูกรีเซตเป็น"Ø"

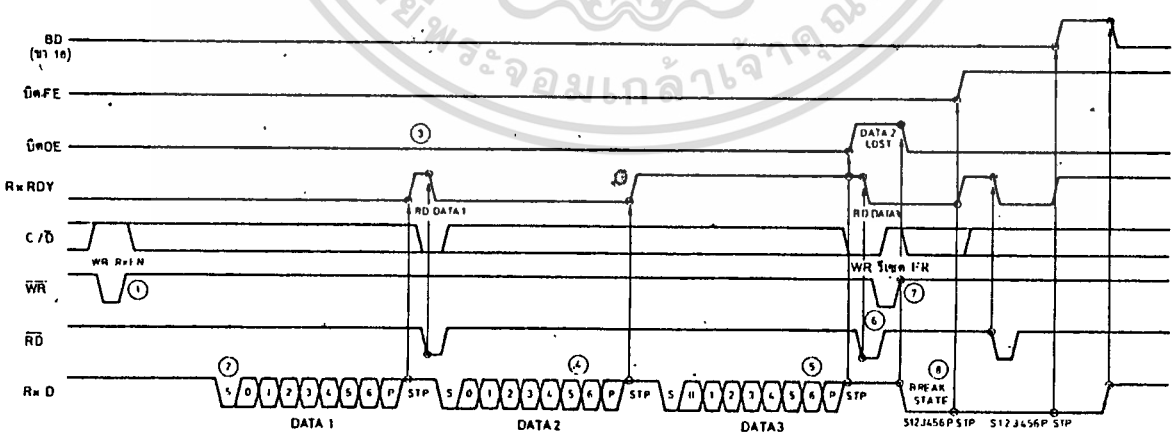
5.เมื่อ DATA 3 ถูกส่งออกไปแล้ว ขา TxEMPTY จะกลายเป็น"1" แต่ TxRDY ยังเป็น"Ø" ลักษณะเช่นนี้แสดงให้เห็นว่ามี DATA อยู่ใน Transmit Buffer A แต่ยังไม่ถูกส่งไปให้ Transmit Buffer B นั่นคือมี DATA ค้างอยู่ใน 8251 อีก 1 ตัว รอที่จะส่งออกไป

6.จนกระทั่งขา $\overline{CTS}$ กลับเป็น"Ø" อีก DATA 4 จึงถูกส่งออกไป

7.DATA 4 ถูกส่งออกมาได้ไม่กี่บิต ซีพียูก็ส่งรหัสควบคุม SBRK (D3) มาให้ 8251 หยุดหรือเบรก 8251 จึงหยุดการส่งตามรูป

<u>การรับข้อมูลแบบอะซิงโครนัส</u>

รูปที่3.31 แสดงให้เห็นสัญญาณต่างๆที่เกี่ยวข้องกันในการรับข้อมูลแบบอะซิงโครนัส จากรูปให้ข้อมูลเป็นขนาด 6 บิต มีบิตพาริตี้ 1 บิต,บิตสต๊อป 2 บิต รวมกับบิตสตาร์ตอีก 1 บิต เป็นความยาวทั้งหมด 10 บิต เหตุการณ์ต่างๆแสดงตามรูปดังนี้



รูปที่ 3.31 ไดอะแกรมเวลาของการรับข้อมูล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1.ซีพียูส่งรหัสควบคุม RxEN (D2) มายัง 8251 ทำให้ขา C/$\overline{D}$ เป็น"1" เพื่อให้ 8251 รับข้อมูลได้

2.ข้อมูลเริ่มมีเข้ามายัง 8251 ทางขา RxD เริ่มด้วยบิตสตาร์ต ตามด้วยข้อมูล และบิตพาริตี้ จบลงด้วยบิตสต๊อป 2 บิต

3.ที่บิตสต๊อปนี้เองเป็นการบอกถึงการจบ 1 ตัวอักษณะ RxRDY จึงเป็น"1"แสดงว่าได้รับข้อมูลเข้ามาครบ 1 ตัวแล้ว ซีพียูจะทำการอ่านมายัง 8251 (C/$\overline{D}$="Ø") รับข้อมูลไปได้เลย เมื่อซีพียูอ่านข้อมูลไปแล้ว RxRDY จะกลับเป็น"Ø"

4.ข้อมูลตัวที่ 2 ส่งเข้ามาอีก RxRDY จึงเป็น"1" อีก

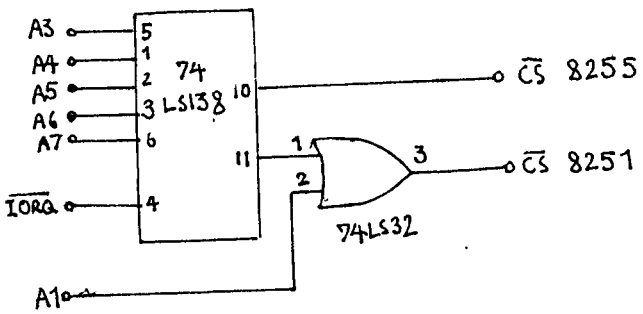
5.สมมติว่า RxRDY เป็น"1" แล้ว ซีพียูยังไม่ยอมมาอ่านข้อมูลไป แล้วมีข้อมูลตัวที่ 3 เข้ามาอีก ทำให้บิต OE (Overrun Error) ในไบต์แสดงสถานะเป็น"1" แสดงให้เห็นว่าข้อมูลบางตัวสูญหายไปเพราะถูกข้อมูลตัวถัดไปส่งมาทับ (DATA 2 หายไป)

6.ซีพียูทำการอ่านข้อมูลจาก 8251 จะได้ DATA 3 (DATA 2 จะถูกทับไปเสียแล้ว) และ OE จะไม่ขัดขวางการทำงานของซีพียูแต่ประการใด แต่ซีพียูจะรู้ว่าเกิดมีข้อผิดพลาดขึ้นแล้วได้โดยการอ่านไบต์แสดงสถานะ (ไม่ได้แสดงไว้ในรูปนี้)

7.ซีพียูสามารถรีเซต OE ในไบต์แสดงสถานะได้ โดยส่งรหัสควบคุม ER (D4) มารีเซตได้

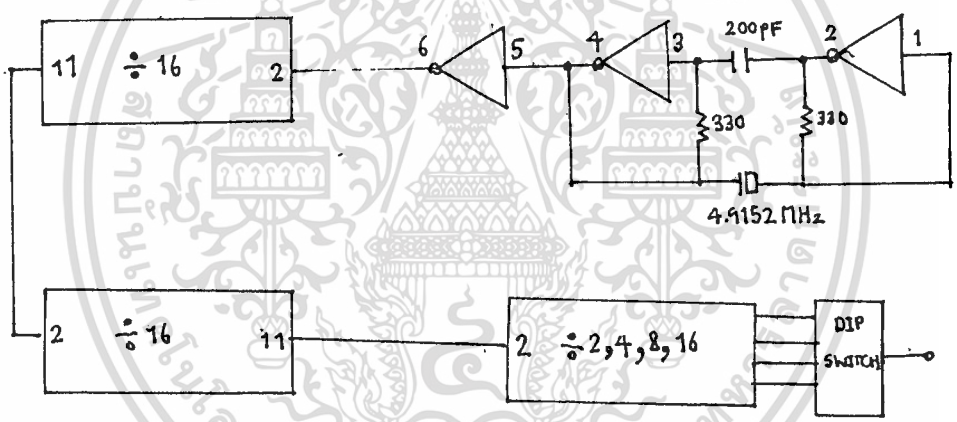
8.ในช่วงเวลานี้เป็นการตรวจพบว่าข้อมูลหรือตัวอักขระที่รับมาเกิดขาดหายไป 2 ตัว คือรับได้เป็น"Ø"ตลอดช่วงความยาว 2 ไบต์ติดต่อกัน จึงเกิดสถานะ break หรือหยุด ทำให้ขา BD (ขา 16) และบิต D6 ในไบต์แสดงสถานะเป็น"1"

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.32 วงจรดีโค้ดของพอร์ทอนุกรมและพอร์ทขนาน

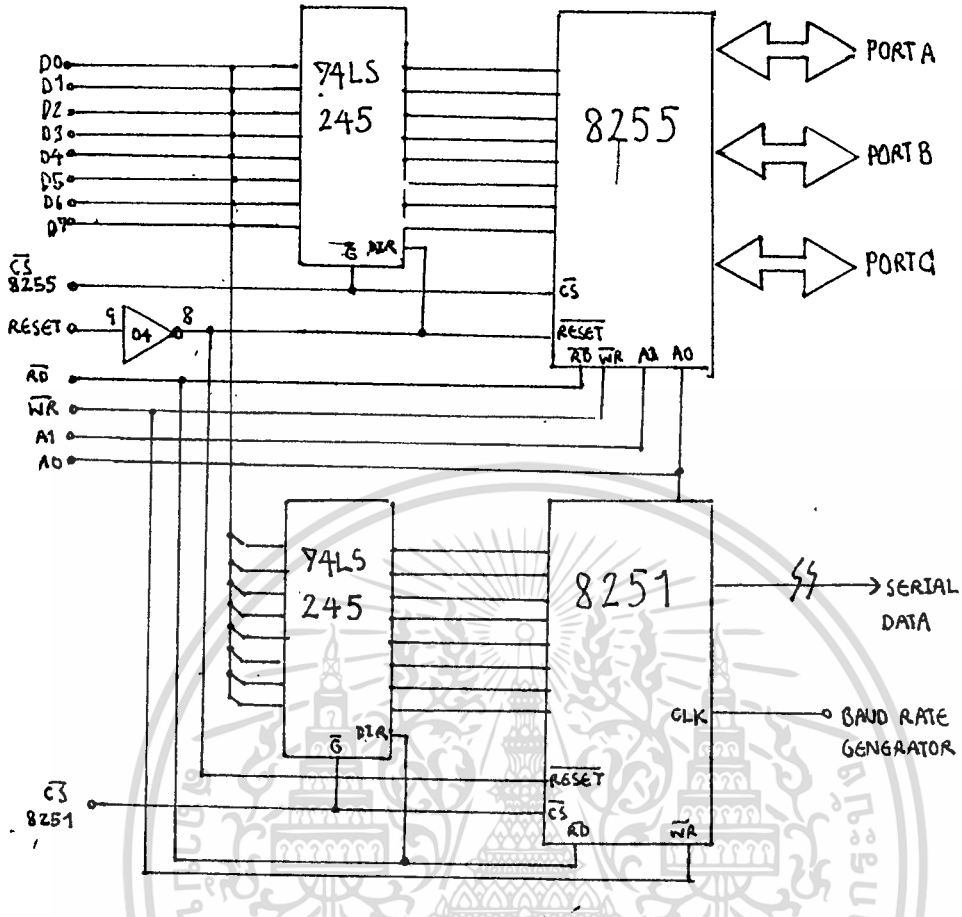
จากรูป แสดงการใช้ไอซีเบอร์ 74LS138 เป็นตัวดีโค้ดแอดเดรสของพอร์ตแบบอนุกรม และขนานที่ใช้งาน



รูปที่ 3.33 วงจรสร้างสัญญาณนาฬิกากำหนดอัตราบอดเรตของ 8251

จากรูป เป็นวงจรสร้างสัญญาณนาฬิกาของอัตราบอดเรตของไอซี 8251 โดยการใช้แร่คริสตอลเป็นตัวควบคุมความถี่ มีไอซีเบอร์ 74LS161 เป็นวงจรหาร และที่เอาท์พุทของไอซีตัวที่สามมีสวิทเลือกสัญญาณนาฬิกา 4 ความถี่

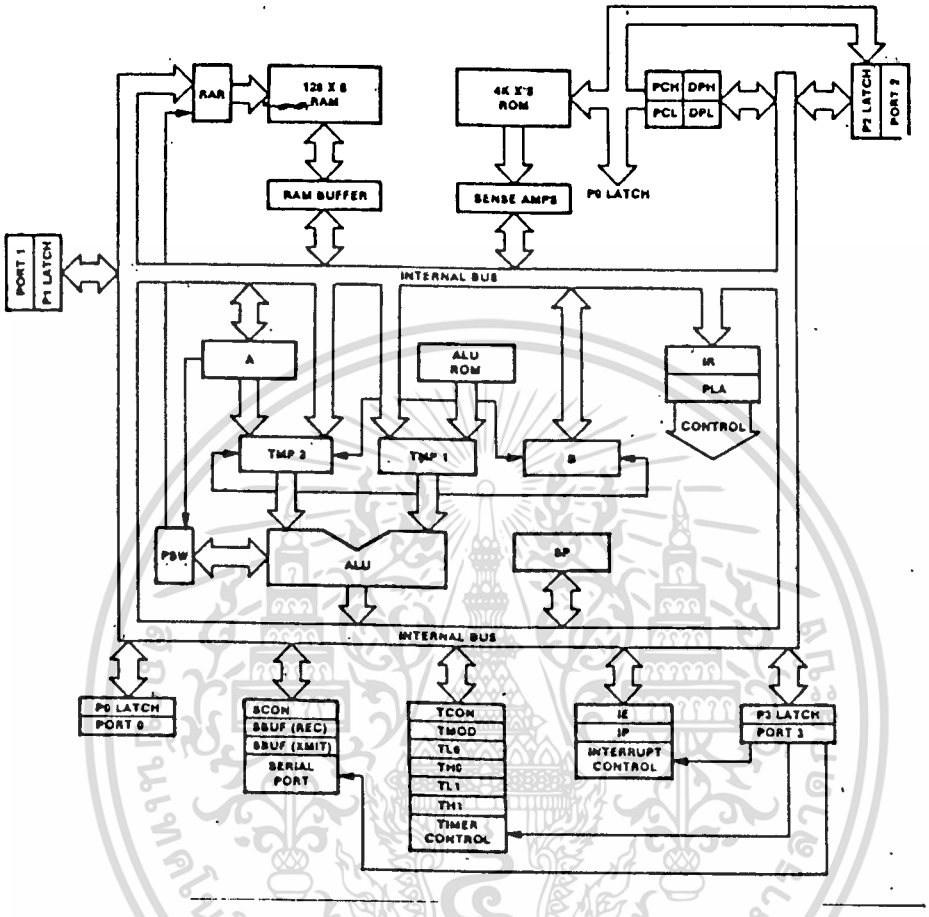
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.34 วงจรของส่วนควบคุมสถานีแม่ที่ใช้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.4) ส่วนควบคุม และประมวลผลของตัวไมโครคอนโทรลเลอร์



รูปที่ 3.35 โครงสร้างภายในของไมโครคอนโทรลเลอร์ตระกูล MCS-51

เราจะอธิบายส่วนสำคัญของไมโครคอนโทรลเลอร์ตระกูล MCS-51 เป็น

(1) หน่วยความจำ (2) คอนโทรลรีจิสเตอร์ (3) อินเทอร์รัพท์

(4) การติดต่อแบบอนุกรม (5) โปรแกรมการทำงาน

(1) หน่วยความจำ

ในตระกูล 8051 นี้จะแบ่งหน่วยความจำเป็น 3 ส่วนคือ

(1.1) หน่วยความจำโปรแกรม 64 Kbyte

(1.2) หน่วยความจำข้อมูลภายนอก 64 kbyte

(1.3) หน่วยความจำข้อมูลภายใน 384 byte

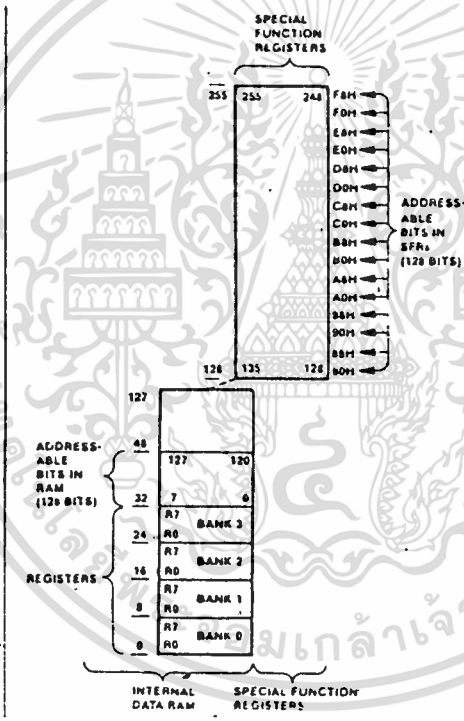
หน่วยความจำโปรแกรมนั้น ไบท์ต่ำขนาด 4 Kbyte จะเป็นส่วนของรอมหรือ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

อีพรอมในกรณีไอซีเบอร์ 8051 หรือ 8751 ซึ่งต้องต่อขา EA กับไฟเลี้ยง ส่วนในกรณีของเบอร์ 8031 ซึ่งไม่มีหน่วยความจำในตัวจะเป็นหน่วยความจำภายนอกทั้งหมด แต่ขา EA จะต้องต่อกับกราวด์

เมื่อมีการรีเซ็ทระบบเกิดขึ้น ซีพียูจะเริ่มทำงานที่ตำแหน่ง 0000 ซึ่งตำแหน่ง 0000-0002 จะเป็นส่วนอินนิเชี่ยล (Initialization) โปรแกรม และตั้งแต่ 0003-0042 จะถูกจองไว้สำหรับส่วนอินเทอร์รัพท์ 5 ชุด ซึ่งจะอยู่ในตำแหน่งต่างๆ

หน่วยความจำข้อมูลภายในจะถูกแบ่งเป็น 2 ส่วนคือ ส่วนของหน่วยความจำแรม 256 byte ซึ่งจะเป็นรีจิสเตอร์ของฟังก์ชันพิเศษอีก 128 byte แสดงได้ดังรูปที่ 3.36



รูปที่ 3.36 หน่วยความจำข้อมูลภายในของ MCS-51

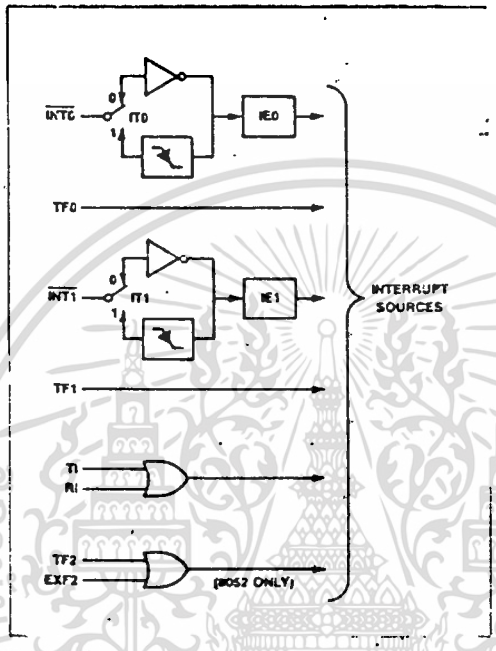
หน่วยความจำแรม 128 byte นั้นตำแหน่ง 0-31 จะเป็นส่วนของรีจิสเตอร์ 8 บิท 4 ชุด ส่วนของสแต็ค (stack) สามารถวางไว้ในตำแหน่งต่างๆ ภายในแรมได้

ส่วนของรีจิสเตอร์ของฟังก์ชันพิเศษนั้นจะอยู่ในตำแหน่ง 128-255 ซึ่งจะมีรีจิสเตอร์พิเศษอยู่ 20 ชุด ประกอบไปด้วย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

(2) ระบบอินเทอร์รัพท์

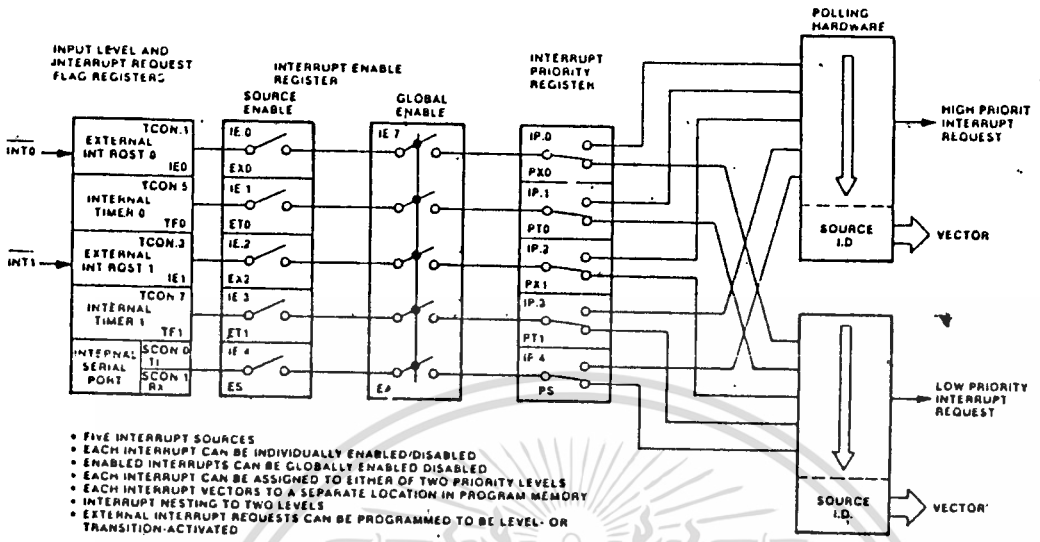
ส่วนตอบสนองของระบบอินเทอร์รัพท์มาจาก 5 ส่วนคือ จากขาอินเทอร์รัพท์ภายนอก 2 ชุด (INT0, INT1), จากไทม์เมอร์/เคาร์เตอร์ 2 ชุด และจากพอร์ตอนุกรมอีก 1 ชุด เราจะแสดงอินเทอร์รัพท์ทั้ง 5 ชุดได้ดังรูปที่ 3.38



รูปที่ 3.38 อินเตอร์รัพต์ 5 ชุดของ MCS-51

ส่วนอินเทอร์รัพท์เวคเตอร์ของแต่ละชุดจะอยู่ในตำแหน่งต่างๆ ในหน่วยความจำโปรแกรม ซึ่งสามารถกำหนดการทำงานและเลือกระดับความสำคัญของอินเทอร์รัพท์ได้จากซอร์ฟแวร์ของบิทในรีจิสเตอร์พิเศษ IE และ IP เราสามารถที่จะแสดงโครงสร้างระบบอินเทอร์รัพท์ได้ดังรูปที่ 3.39

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.39 โครงสร้างของระบบอินเตอร์รัพต์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

(3) คอนโทรลรีจิสเตอร์

มีรายละเอียดดังนี้

PSW

CY	AC	FØ	RS1	RSØ	OV	X	P

CY เป็นแคร์รี่แฟลก (Carry Flag)

AC เป็นอ็อกซิลารี่แฟลก(Auxillary Flag)สำหรับการทำงานแบบบีซีดี (BCD)

FØ เป็นแฟลกศูนย์ (Zero Flag)

RS1 เป็นบิทที่ใช้รีจิสเตอร์แบงค์ (Register Bank) สามารถตั้งและลบได้ด้วยซอร์ฟแวร์

RSØ ใช้งานร่วมกับรีจิสเตอร์ RS1

รูปแบบการตั้งการทำงานของรีจิสเตอร์ RSO,RS1

Ø,Ø เลือกหน่วยความจำแรมช่วง ØØh-Ø7h Ø,1 เลือกหน่วยความจำแรมช่วง Ø8h-Øfh

1,Ø เลือกหน่วยความจำแรมช่วง 1Øh-17h 1,1 เลือกหน่วยความจำแรมช่วง 18h-1Fh

OV แสดงโอเวอร์โฟล์วแฟลก

X สำรอง

P แสดงพาริตี้ (Parity Flag) ตั้งและลบได้ด้วยซอร์ฟแวร์

IE

EA	X	X	ES	ET1	EX1	ETØ	EXØ

EA เป็นการควบคุมการรับอินเทอร์รัพท์ระบบ ถ้าเป็น Ø จะไม่รับอินเทอร์รัพท์ ถ้าเป็น 1 การรับอินเทอร์รัพท์ จะขึ้นกับแต่ละตัวซึ่งอิสระต่อกัน

ES เป็นการควบคุมการรับอินเทอร์รัพท์ของพอร์ตอนุกรม

ET1 เป็นตัวควบคุมการรับอินเทอร์รัพท์ของไทม์เมอร์1

EX1 เป็นตัวควบคุมการรับอินเทอร์รัพท์ของขาอินเทอร์รัพท์ภายนอก1

ETØ เป็นตัวควบคุมการรับอินเทอร์รัพท์ของไทม์เมอร์Ø

EXØ เป็นตัวควบคุมการรับอินเทอร์รัพท์ของขาอินเทอร์รัพท์ภายนอกØ

X สำรอง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

IP

X	X	X	PS	PT1	PX1	PTØ	PXØ

PS เป็นการทำให้อินเทอร์รัพท์ของพอร์ตอนุกรม มีความสำคัญกว่าอันอื่น

PT1 เป็นการทำให้อินเทอร์รัพท์ของไทม์เมอร์ 1 มีความสำคัญกว่าอันอื่น

PX1 เป็นการทำให้อินเทอร์รัพท์ของขาอินเทอร์รัพท์ 1 มีความสำคัญกว่าอันอื่น

PTØ เป็นการทำให้อินเทอร์รัพท์ของไทม์เมอร์ Ø มีความสำคัญกว่าอันอื่น

PXØ เป็นการทำให้อินเทอร์รัพท์ของขาอินเทอร์รัพท์ Ø มีความสำคัญกว่าอันอื่น

X สำรอง

SCON

SMØ	SM1	SM2	REN	TB8	RB8	T1	R1

SMØ,1 เป็นตัวเลือกโหมดควบคุม

ØØ โหมด Ø เป็นชิพรีจิสเตอร์ มีอัตราการส่ง = $F_{osc}/12$

Ø1 โหมด 1 เป็นการส่ง 8 บิท มีอัตราการส่งที่แปรค่าได้

1Ø โหมด 2 เป็นการส่ง 9 บิท มีอัตราการส่งเท่ากับ = $F_{osc}/64$ หรือ $F_{osc}/32$

11 โหมด 3 เป็นการส่ง 9 บิท มีอัตราการส่งที่แปรค่าได้

SM2 เป็นตัวทำให้มีการติดต่อแบบ Multiprocessor ถ้าถูกเซ็ตจะทำให้ RI ไม่ทำงาน

REN ถ้าถูกเคลียร์จะทำให้ไม่สามารถส่งข้อมูล

TB8 เป็นบิทที่ 9 ของข้อมูลที่จะส่งไปในโหมด 2 หรือ 3

RB8 เป็นบิทที่ 9 ของข้อมูลที่รับมาในโหมด 2 หรือ 3

T1 เป็นตัวตรวจว่ามีการอินเทอร์รัพท์จากการส่งข้อมูล โดยจะเซ็ตเมื่อการส่งสมบูรณ์ และจะถูกเคลียร์ทางซอร์ฟแวร์

R1 เป็นตัวตรวจว่ามีการอินเทอร์รัพท์จากการรับข้อมูล โดยจะเซ็ตเมื่อการรับสมบูรณ์ และจะถูกเคลียร์ทางซอร์ฟแวร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

TMOD

Gate	C/T	M1	MØ	Gate	C/T	M1	MØ

Gate ถ้ามีค่าเป็น Ø จะเป็นตัวที่จะทำให้ไทม์เมอร์ หรือเคาร์เตอร์ทำงานได้

C/T เป็นตัวเลือกว่าจะใช้งานเป็นไทม์เมอร์ หรือเคาร์เตอร์

M1,M2 เป็นตัวเลือกโหมดในการทำงาน

ØØ เป็นการใช้งานไทม์เมอร์/เคาร์เตอร์ ขนาด 13 บิท

Ø1 เป็นการใช้งานไทม์เมอร์/เคาร์เตอร์ ขนาด 16 บิท

1Ø เป็นการใช้งานไทม์เมอร์/เคาร์เตอร์ ขนาด 8 บิท โดยเป็นลักษณะเรียกค่าซ้ำโดยอัตโนมัติ (Auto-Reload)

11 ในไทม์เมอร์ Ø เป็นการใช้ไทม์เมอร์/เคาร์เตอร์ขนาด 8 บิท โดย TLØ จะถูกควบคุมตามมาตรฐาน ส่วน THØ จะถูกควบคุมจาก TR1 ในไทม์เมอร์ 1 จะไม่ใช้

TCON

TF1	TR1	TFØ	TRØ	IE1	IT1	IEØ	ITØ

TF1 เป็นแฟลกโอเวอร์โฟล์วของไทม์เมอร์/เคาร์เตอร์ การเซ็ตและเคลียร์ทางฮาร์ดแวร์

TR1 เป็นตัวควบคุมไทม์เมอร์ 1 ให้ทำงานหรือหยุดทำ การเซ็ตและเคลียร์ทางซอร์ฟแวร์

TFØ เป็นแฟลกโอเวอร์โฟล์วของไทม์เมอร์/เคาร์เตอร์ Ø

TRØ เป็นตัวควบคุมไทม์เมอร์ Ø

IE1 เป็นตัวตรวจว่ามีการรับสัญญานอินเทอร์รัพท์จากขาอินเทอร์รัพท์ 1 หรือไม่

IT1 เป็นบิทที่กำหนดการรับสัญญานอินเทอร์รัพท์ขา 1 ว่าจะรับช่วงขาลงหรือช่วงสถานะเป็น Ø

IEØ เป็นตัวตรวจว่ามีการรับสัญญานอินเทอร์รัพท์จากขาอินเทอร์รัพท์ Ø หรือไม่

ITØ เป็นบิทที่กำหนดการรับสัญญานอินเทอร์รัพท์ขา Ø ว่าจะรับช่วงขาลงหรือช่วงสถานะเป็น Ø

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4) การติดต่อแบบอนุกรม

พอร์ตอนุกรมของไมโครคอนโทรลเลอร์ตระกูล MCS-51 สามารถสื่อสารในระบบ UART (Universal Asynchronous Receiver and Transmitter) โดยพอร์ตอนุกรมนี้สามารถติดต่อในแบบฟูลดูเพล็กซ์ซึ่งโปรแกรมให้ฟังก์ชันการทำงานได้เป็น 4 โหมด คือ

โหมด 0 : เป็นการสื่อสารแบบซิงโครนัส

โหมด 1 : เป็นการสื่อสารแบบ UART โดย 1 เฟรมมี 10 บิท และสามารถเลือกความเร็วในการสื่อสารได้

โหมด 2 : เป็นการสื่อสารแบบ UART โดย 1 เฟรมมี 11 บิท และอัตราเร็วในการสื่อสารคงที่

โหมด 3 : เป็นการสื่อสารแบบ UART โดย 1 เฟรมมี 11 บิท และสามารถเลือกความเร็วในการสื่อสารได้

รีจิสเตอร์ที่ควบคุมพอร์ตอนุกรม

รีจิสเตอร์หน้าที่พิเศษ SCON ซึ่งไม่ได้มีเฉพาะบิทในการควบคุมเท่านั้น ยังเป็นที่อยู่ของบิทที่ 9 ทั้งในการรับและส่ง และยังรวมบิทอินเทอร์รัพท์ของพอร์ตอนุกรมด้วย

การกำหนดบอดเรต

ในโหมด 0 อัตราบอดเรตคงที่ คือเท่ากับ 1/12 ของความถี่ของออสซิลเลเตอร์

ในโหมด 2 นั้นจะขึ้นกับค่า SMOD ในรีจิสเตอร์พิเศษ PCON คือ

$$\text{บอดเรต} = 2^{\text{mod}} * \text{ความถี่ของออสซิลเลเตอร์}/64$$

ส่วนในโหมด 1 และ 3 นั้น บอดเรตจะมาจากอัตราการเกิดโอเวอร์โฟลว์ในไทม์เมอร์1 โดย

$$\text{บอดเรต} = [2^{\text{mod}}/32] * [\ \text{ความถี่ออสซิลเลเตอร์}/12*(256-TH1)\ ]$$

ตารางต่อไปนี้จะแสดงการกำหนดอัตราบอดเรตในโหมด 1 และ 3

BAUD RATE	f_{osc}	SMOD	TIMER 1		
			C/T	MODE	RELOAD VALUE
MODE 0 MAX: 1MHZ	12 MHZ	X	X	X	X
MODE 2 MAX. 375K	12 MHZ	1	X	X	X
MODES 1,3 62.5K	12 MHZ	1	0	2	FFH
19.2K	11.059 MHZ	1	0	2	FDH
9.6K	11.059 MHZ	0	0	2	FDH
4.8K	11.059 MHZ	0	0	2	FAH
2.4K	11.059 MHZ	0	0	2	F4H
1.2K	11.059 MHZ	0	0	2	E8H
137.5	11.986 MHZ	0	0	2	1DH
110	6 MHZ	0	0	2	72H
110	12 MHZ	0	0	1	FEEBH

ตารางที่ 3.2

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ในโครงงานนี้ได้เลือกการใช้งานพอร์ตอนุกรมในโหมด 3 คือเป็น UART 11 bit โดยสามารถกำหนดอัตราบอดเรตได้ ซึ่งสามารถอธิบายรายละเอียดได้ดังนี้

บิททั้ง 11 bit นั้นจะประกอบไปด้วย 1 สต็าทบิท, 1 สต็อปบิท, 8 ดาต้าบิท รวมทั้งบิทที่ 9 ซึ่งสามารถโปรแกรมได้ โดยบิทนี้ในการส่งจะต้องโหลดเข้าบิท TB8 และในการรับก็จะถูกโหลดเข้าบิท RB8 ใน SCON ซึ่งโหมดนี้ต่างจากโหมด 1 เพียงบิท 9 นี้เอง

ในการส่งนั้นจะต้องดาต้าเข้ารีจิสเตอร์ SBUF จากนั้นก็จะผ่านชิพรีจิสเตอร์ พร้อมทั้งบิท TB8 จากนั้นก็จะทำการเซ็ตบิท T1 ใน SCON แสดงว่าส่งดาต้าไปหมดเฟรมแล้ว

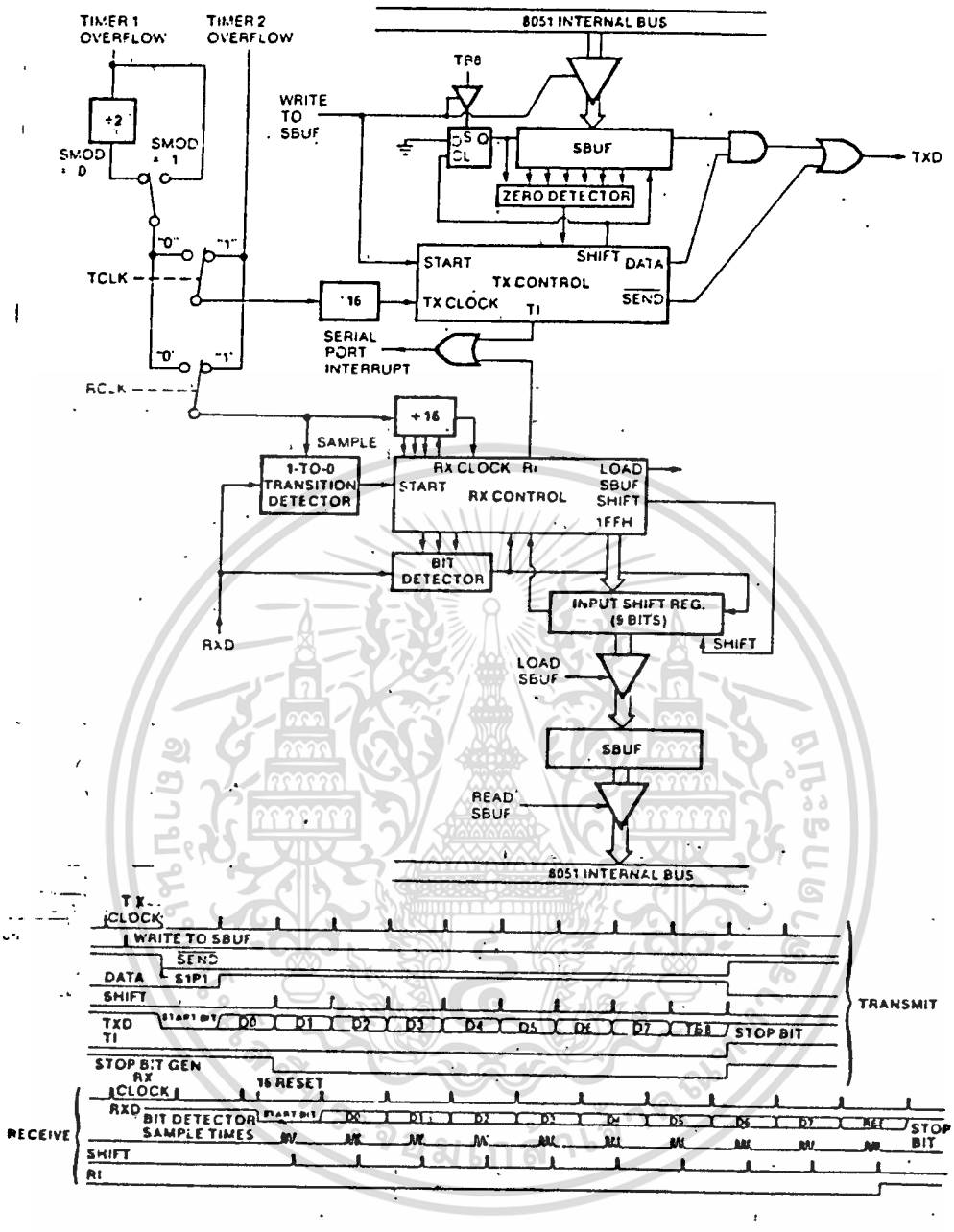
ในด้านรับนั้นก็จะรับดาต้าแล้วผ่านชิพรีจิสเตอร์เข้ารีจิสเตอร์ SBUF และ RB1 พร้อมทั้งเซ็ต R1 แต่สภาวะต่อไปนี้ต้องเกิดขึ้น

1) RI=0 และ 2) SM2=0 หรือ บิทที่ 9 ที่รับเป็น 1

ถ้าสภาวะไม่เป็นไปตามที่กำหนด ข้อมูลที่รับมาจะถือว่าสูญหาย

รูปต่อไปนี้จะแสดงการทำงานของพอร์ตอนุกรมในโหมดที่ 3

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.40 การทำงานของพอร์ทอนุกรม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5) โปรแกรมการทำงานที่สำคัญ

โปรแกรมการใช้งานที่สำคัญได้แก่

1) โปรแกรมควบคุมการรับของพอร์ตอนุกรม

2) โปรแกรมเช็คพาริตี้

3) โปรแกรมการเช็คอินเทอร์รัพท์ภายนอก

<u>โปรแกรมควบคุมการรับของพอร์ตอนุกรม</u>

ในโครงงานนี้มีข้อกำหนดของข้อมูลที่รับมา คือ

1) ทำงานในโหมด 3 (ข้อมูลประกอบด้วย 1 สต๊าทบิท, 8 ดาต้าบิท, 1 พาริตี้บิท และ 1 สต๊อปบิท)

2) อัตราบอดเรตเป็น 300 บอด

3) ไทม์เมอร์1 ทำงานในลักษณะโหลดค่าโดยอัตโนมัติ (8-bit Auto-Reload Timer/Counter)

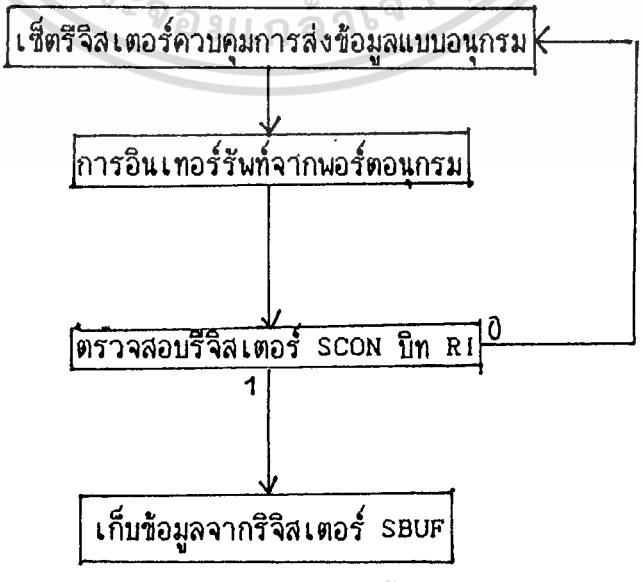
การคำนวณหาค่าของไทม์เมอร์1 จากสูตร

$$\text{บอดเรต} = [2^{\text{SMOD}}/32]*[6E6/12*(256-TH1)]$$

ให้ $S^{MOD} = 0$, บอดเรต = 300 บอด จะได้

$$TH1 = HCC$$

โฟลว์ชาร์ตแสดงการทำงานของการควบคุมของพอร์ตอนุกรมเป็นดังรูป



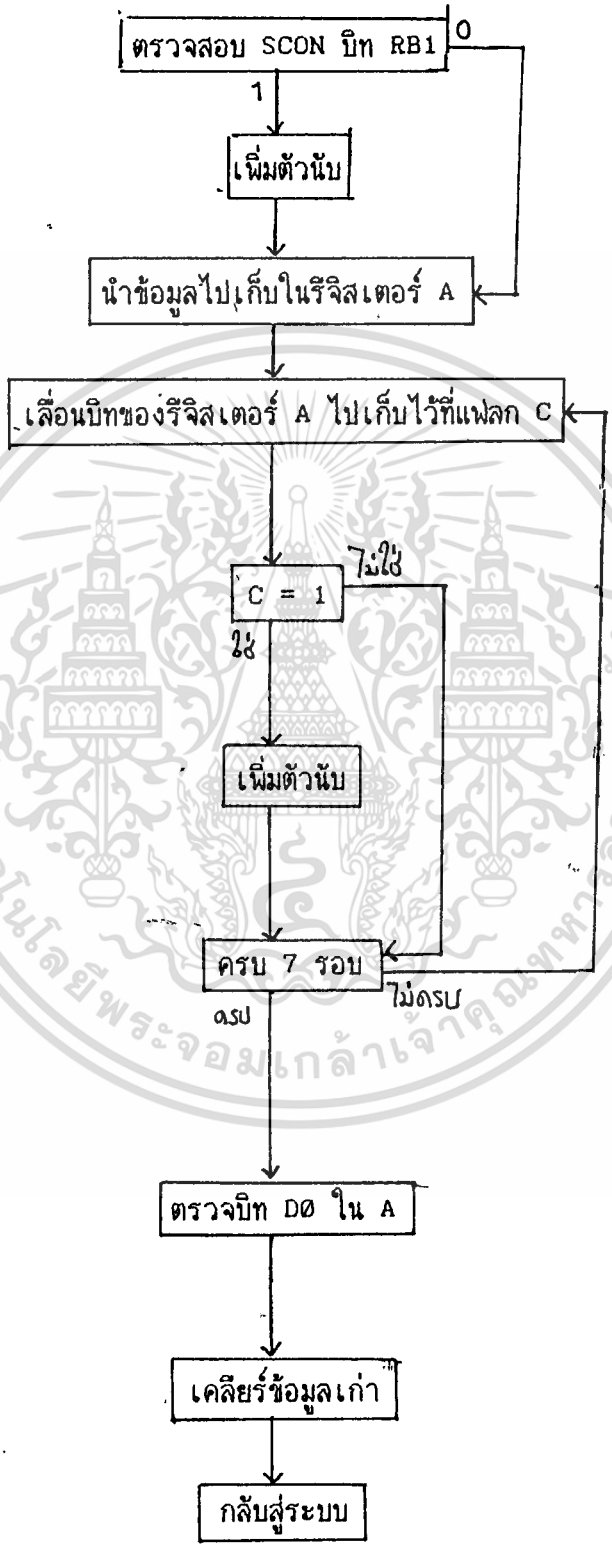
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
( PROGRAM MCS 8031 RECEIVE DATA)
      MOV   TH1 , HCC-,        ( SET BUAD RATE )
      MOV   TCON , H40,        ( SET TIME 1 )
      MOV   TMOD , H20,        ( SET T1 AUTO RELOAD )
      MOV   SMOD , H00'        ( SET SMOD FOR BOUD RATE)
      MOV   SCON , H50         ( SET SCON)
      MOV   IE  , H90          ( ENABLE RECEIVE )
(PROGRAM INTERUPT RECEIVE)
      JB    SCON , H01         (CHECK BIT RI)
      RET   I
      MOV   A   , SBUFF        (RECEIVE DATA)
```

<u>โปรแกรมเช็คพาริตี้</u>

ได้กำหนดให้ใช้พาริตี้คี่ (ODD PARITY) มีโฟลว์ชาร์ตการทำงานดังรูป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



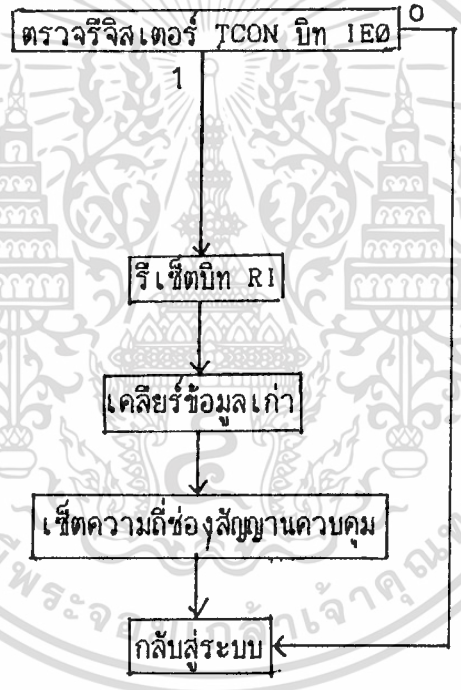
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
(PROGRAMME CHECK ODD PARITY)
                MOV   R2 , ØØH      (CLEAR COUNTER)
                MOV   C  , RB1      (LOAD PARITY)
                JNC   NO 1
                INC   R2
    NO1     ;   MOV   R3 , Ø7H
                MOV   A , R1
    CHECK   ;   RLC   A
                JNC   NO2
                INC   R2
    NO 2    ;   DJNZ R3, CHECK
                MOV   A , R2
                MOV   C , A (Ø)
                JC    ODD PARITY    (CHECK PARITY DO)
                MOV   RØ ,  ØØH
ODD PARITY  ;   RET
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

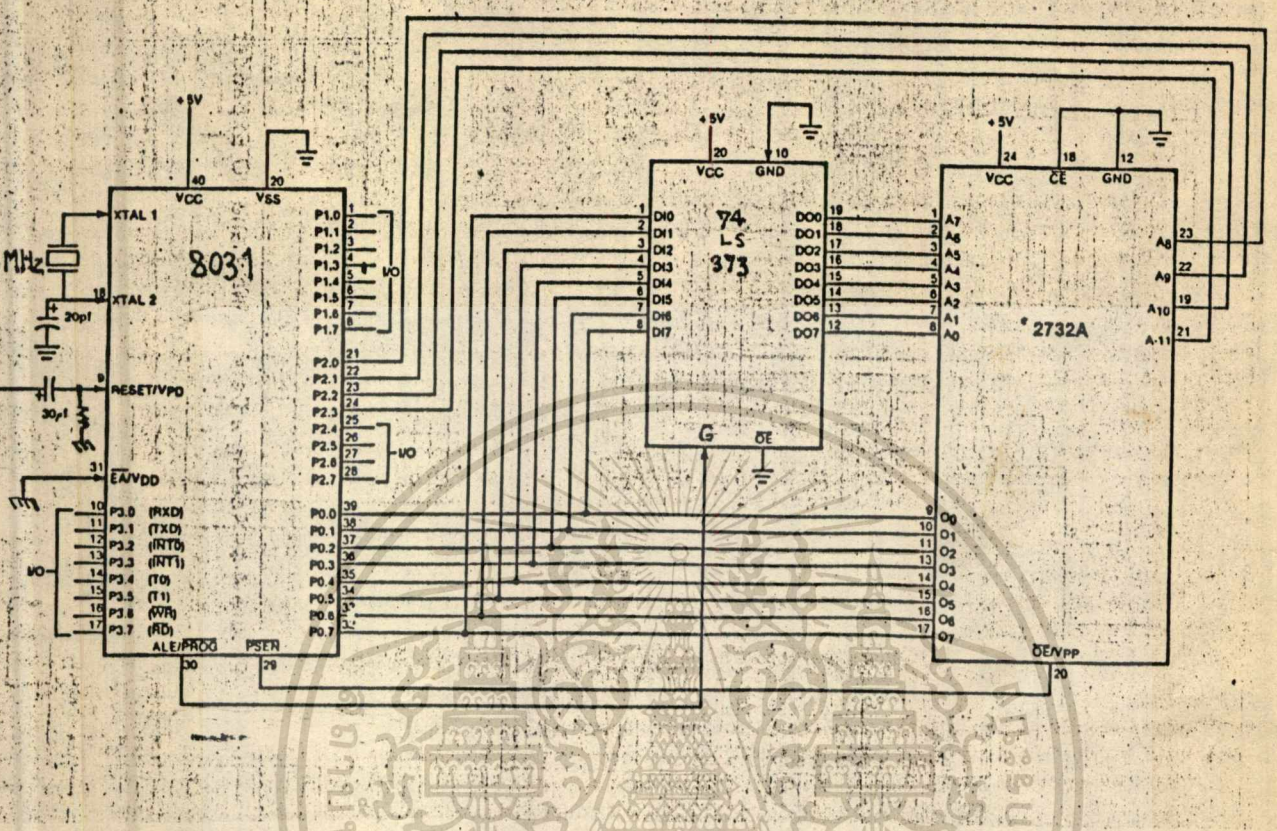
โปรแกรมการเช็คอินเทอร์รัพท์ภายนอก

ขา INTO ได้ต่อไว้ด้วยสวิทส์เป็นตัวควบคุมการรับสายและ การต่อออกของตัววิทยุโทรศัพท์ โดยถ้าสถานะเป็นโลว์จะเกิดการอินเทอร์รัพท์ ดังนั้นจะเป็นการเคลียร์ระบบแสดงว่า ไม่มีการใช้งาน โดยจะควบคุมจากรีจิสเตอร์ TMOD ว่าจะให้อินาเบิ้ลในจุดใด เราสามารถแสดงโฟลว์ชาร์ตการทำงานได้ดังรูป




```
(PROGRAMME EXTERNAL INTERURT)
            JB    TCON , Ø1H      (CHECK INTERUPT)
            RET
            CLR   TCON
            MOV   RØ , HØØ
            MOV   PORT 1 , HØ8    (SET DATA CHANEL)
            MOV   IE , 9ØH        (ENABLE SERIAL ,  DISABLE INTO)
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.41 วงจรของส่วนควบคุมวิทยุโทรศัพท์ที่ใช้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 4

การทดลองและผลการทดลอง

ภาคการศึกษาแรก

ได้ออกแบบสร้างส่วนควบคุมของสถานีแม่ และตัววิทยุโทรศัพท์

4.1) ผลการทดลองของส่วนควบคุมสถานีแม่ สามารถอ่านค่าจากพอร์ตแบบขนาน 8255 แล้วส่งออกทางพอร์ตอนุกรม 8251 ได้โดยกำหนดอัตราเร็วที่ใช้ส่งไว้ 600 บอด

4.2) ผลการทดลองโมเด็ม เมื่อทดลองส่งพัลซ์เทรนความถี่ 150 Hz โมเด็มด้านรับ สามารถรับสัญญาณที่ส่งมาได้อย่างถูกต้อง แต่เมื่อทดลองส่งความถี่ 300 Hz ได้เกิดอาการสั่นไปมาของสัญญาณที่ได้จากโมเด็มทางด้านรับ

4.3) ผลการทดลองส่วนควบคุมของตัววิทยุโทรศัพท์ ทดสอบไอซี 8035 (เป็นไมโครคอนโทรลเลอร์ตระกูล MCS-48 ซึ่งได้ใช้ในภาคการศึกษาแรก) โดยการอ่านสัญญาณนาฬิกาที่ป้อนให้แก่ขา T0 แล้วแสดงออกทางเอาร์ตพุทพอร์ต โดยให้แอลอีดีติด-ดับสลับกัน ปรากฏว่า เกิดค่าผิดพลาดขึ้นประมาณ 1 ครั้งในการติด-ดับ 10 ครั้ง (เป็นเพราะโปรแกรมการอ่านข้อมูลอนุกรมไม่ดี และเนื่องจาก 8035 ไม่ได้มีขาที่ทำหน้าที่รับสัญญาณอนุกรมโดยตรง ดังนั้นจึงอาศัยโปรแกรมการอ่านเพียงอย่างเดียว)

ภาคการศึกษาที่ 2

ได้ปรับปรุงส่วนควบคุมของวิทยุโทรศัพท์ทั้งหมด และออกแบบสร้างวงจรสังเคราะห์ความถี่ชนิดเฟสล็อคลูป

4.4) ผลการทดลองของวงจรสังเคราะห์ความถี่ ขณะกำลังทำปริญญานิพนธ์ วงจรในภาคต่างๆ ของวงจรสังเคราะห์ความถี่สามารถทำงานได้คือ วีซีโอ, มิกเซอร์ และออส-ซิลเลเตอร์ แต่เมื่อทดลองต่อรวมทั้งระบบแล้วไม่สามารถทำงานได้ เมื่อทดลองเฉพาะส่วนวีซีโอ กับมิกเซอร์นั้น ปรากฏว่ามีสัญญาณรบกวนจากวงจรมิกเซอร์คือ สัญญาณพาหะทั้งสองตัวจากวงจรออสซิลเลเตอร์ และวีซีโอ มีระดับสูงมาก ซึ่งกำลังแก้ไขวงจรอยู่ ผลการทดลองของวีซีโอได้แสดงไว้ในหัวข้อ 3.

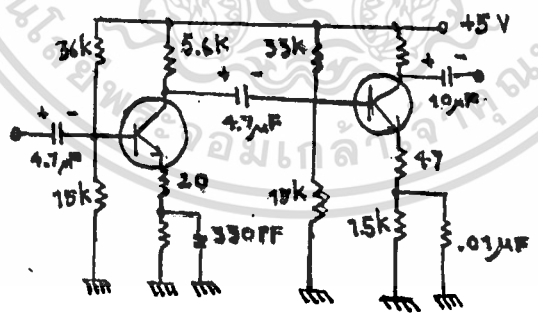
4.5) ผลการทดลองส่วนควบคุมของวิทยุโทรศัพท์ ทดลองโปรแกรมทดสอบไอซี 8031 (เป็นไมโครคอนโทรลเลอร์ตระกูล MCS-51 ซึ่งเปลี่ยนจากภาคการศึกษาที่ 1)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 5
บทวิจารณ์และสรุป

จากการที่ต้องการแก้ไขส่วนควบคุมของวิทยุโทรศัพท์ให้สมบูรณ์ โดยการเปลี่ยนไอซีไมโครคอนโทรลเลอร์จาก 8035 มาเป็น 8031 เนื่องจาก 8031 มีพอร์ตอนุกรมไว้ใช้งานด้วยไม่จำเป็นต้องอาศัยโปรแกรมทั้งหมดในการรับข้อมูลแบบอนุกรม เช่นเดียวกับ 8035 ดังนั้นโปรแกรมของระบบจึงกะทัดรัดขึ้น ส่วนการที่จะใช้วงจรทางฮาร์ดแวร์ซึ่งเป็นรีโมตคอนโทรลในการติดต่อกับตัววิทยุโทรศัพท์แต่ละตัวก็ไม่จำเป็น เพราะได้เปลี่ยนวิธีการเป็นการควบคุมมาจากข้อมูลที่ส่งผ่านช่องสัญญาณควบคุมจะมีหมายเลขของวิทยุโทรศัพท์แต่ละเครื่องมาด้วย ซึ่งเป็นการลดวงจรทางฮาร์ดแวร์อีกด้วย

ส่วนของวงจรสังเคราะห์ความถี่นั้นตอนนี้ยังแก้ไขการรบกวนของวงจรคริสตอลออสซิลเลเตอร์ และวีซีโอที่เอาร์พุท โดยจะทดลองสร้างวงจรแบนด์รีเจ็คฟิลเตอร์ ซึ่งตอนนี้ยังไม่ได้ทำ ส่วนระดับสัญญาณจากมิกเซอร์ที่จะไปเข้าไอซีสังเคราะห์ความถี่นั้น ปรากฏว่า ระดับสัญญาณที่ต้องการต่ำมากไม่สามารถทำให้วงจรทำงานได้ ดังนั้นจึงออกแบบวงจรขยายคอมมอนอี--มิตเตอร์สองเสตท ดังรูป



แล้วผ่านไปยังไอซี 74HC04 ซึ่งทำหน้าที่เป็นบัฟเฟอร์เพื่อป้อนเข้าไอซีสังเคราะห์ความถี่ต่อไป จากการทดลอง ปรากฏว่า ระดับสัญญาณสูงพอที่จะทำให้ไอซีสังเคราะห์ความถี่ทำงานได้ ดังนั้นปัญหาสำคัญตอนนี้อยู่ที่เอาร์พุทของมิกเซอร์

ทางด้านสถานีแม่ ส่วนควบคุมได้ต่อกับซิงเกิ้ลบอร์ดแล้วสามารถทำงานได้ดี แต่ติดปัญหาอยู่ที่การแก้ไขโปรแกรมค่อนข้างลำบากดังนั้นตอนนี้จึงทดลองทำงานที่จะไปอินเตอร์-

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เฟสกับคอมพิวเตอร์ IBM-PC เพื่อความสะดวกในการพัฒนาโปรแกรม

แนวความคิดและข้อเสนอแนะ เพื่อเป็นแนวทางในการพัฒนาระบบควบคุมวิทยุ-โทรศัพท์สรุปได้คือ

1) ทำส่วนติดต่อทางวิทยุ และติดต่อกับคู่สายโทรศัพท์ซึ่งที่จริงแล้วมีโครงการว่า จะทำในภาคการศึกษาที่ 2 นี้แต่เราได้เสียเวลาในการเปลี่ยนแปลงส่วนควบคุมของตัววิทยุโทรศัพท์ทั้งหมด ดังนั้นจึงยกเลิกไป

2) ข้อมูลที่ติดต่อกันระหว่างวิทยุโทรศัพท์และสถานีแม่ ควรเป็นแบบฟูลดูเพล็กซ์ เพื่อความสะดวกในการเขียนโปรแกรมและประหยัดเวลาในการทำงานของสถานีแม่ ที่จะต้องบริการตัววิทยุโทรศัพท์ในทุกขั้นตอนแม้จะไม่มีการใช้งานก็ตาม

3) หน่วยประมวลผลในสถานีแม่ ควรใช้คอมพิวเตอร์แทนชิงเกิ้ลบอร์ดเพื่อสะดวกในการพัฒนาโปรแกรม

4) โปรแกรมหลักของทั้งสถานีแม่และตัววิทยุโทรศัพท์ ยังสามารถพัฒนาได้อีกเพื่อเพิ่มขีดความสามารถ ให้กับตัววิทยุโทรศัพท์เอง

5) ขยายช่องสัญญาณในการติดต่อของวงจรสังเคราะห์ความถี่ให้มากขึ้น เพื่อขอบข่ายในการติดต่อจะได้กว้างขึ้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กิตติกรรมประกาศ

ขอขอบคุณ อาจารย์นิภา ลีลารุจิ , อาจารย์สมยศ จุณณะปิยะ ที่ได้ให้คำแนะนำอันเป็นประโยชน์ และสนับสนุนการทำงานมาโดยตลอด จึงกราบขอบพระคุณอย่างสูงไว้ ณ ที่นี้

ขอขอบคุณ อาจารย์ภาควิชาโทรคมนาคมทุกท่าน ที่ได้ให้ความร่วมมือ และช่วยเหลืออำนวยความสะดวกในการทำวิทยานิพนธ์ด้วยดีมาตลอด

ท้ายสุดนี้ ขอขอบคุณเพื่อนๆทุกท่าน ที่ได้มีส่วนช่วยเหลือ และแนะนำโครงงานชิ้นนี้จนประสบความสำเร็จ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เอกสารอ้างอิง

1) JOHN B. PEATMAN , " DESIGN WITH MICROCONTROLLER ", McGRAW-HILL,INC. , 1988

2) JAMES W. COFFRON , " Z-80 APPLICATION " , SYBEX,INC. , 1983

3) WES HAYWARD , " SOLID STATE DESIGN FOR THE RADIO AMATURE " , AMERICAN RADIO RELAY LEAGUE,INC. , 1977

4) ARTURE H. SEIDMAN , " INTEGRATED CIRCUITS APPLICATION HANDBOOK " , JOHN WILEY & SON , 1983

5) JOHN L. HILBURN , DAVID E. JOHNSON , " MANUAL OF ACTIVE FILTER DESIGN " , McGRAW-HILL,INC. , 1973

6) DATABOOK , " MCS-51 FAMILY OF SINGLE CHIP MICRO-COMPUTERS USER'S MANUAL " , INTEL CORPORATION , 1981

7) บุญศักดิ์ พิทักษ์เลิศกุล, ประกิจ ธาราประเสริฐ , " เพจเจอร์ " , วิทยานิพนธ์สำหรับวิศวกรรมศาสตร์บัณฑิต คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าลาดกระบัง , พ.ศ.2530

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก

ภาคผนวกที่ 1

แสดงโฟล์วชาร์ตการทำงานของสถานีแม่

ภาคผนวกที่ 2

แสดงโฟล์วชาร์ตการทำงานของวิทยุโทรศัพท์ และโปรแกรมทั้งหมด

ภาคผนวกที่ 3

วงจรสังเคราะห์ความถี่แบบเฟสล็อคลูป

ภาคผนวกที่ 4

ดาต้าชีทของ MCS-51 , 8251

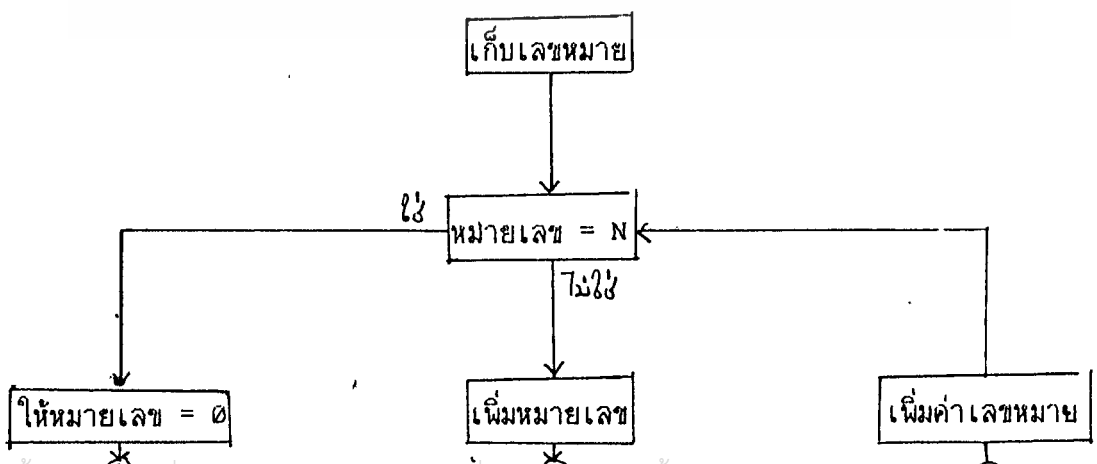
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวกที่ 1

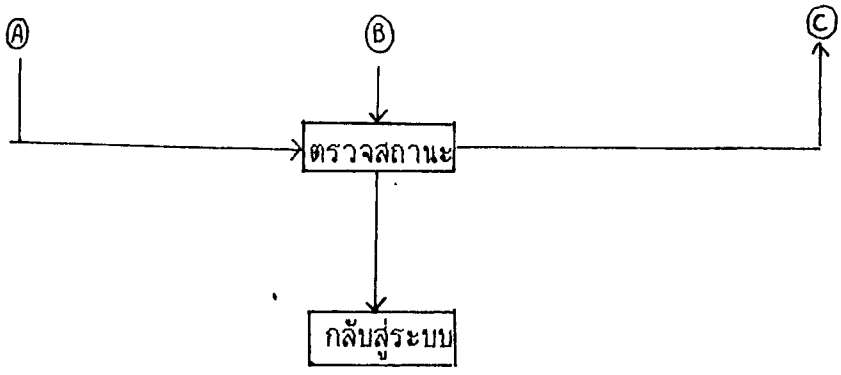
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

(โปรแกรมหลักของสถานีแม่)

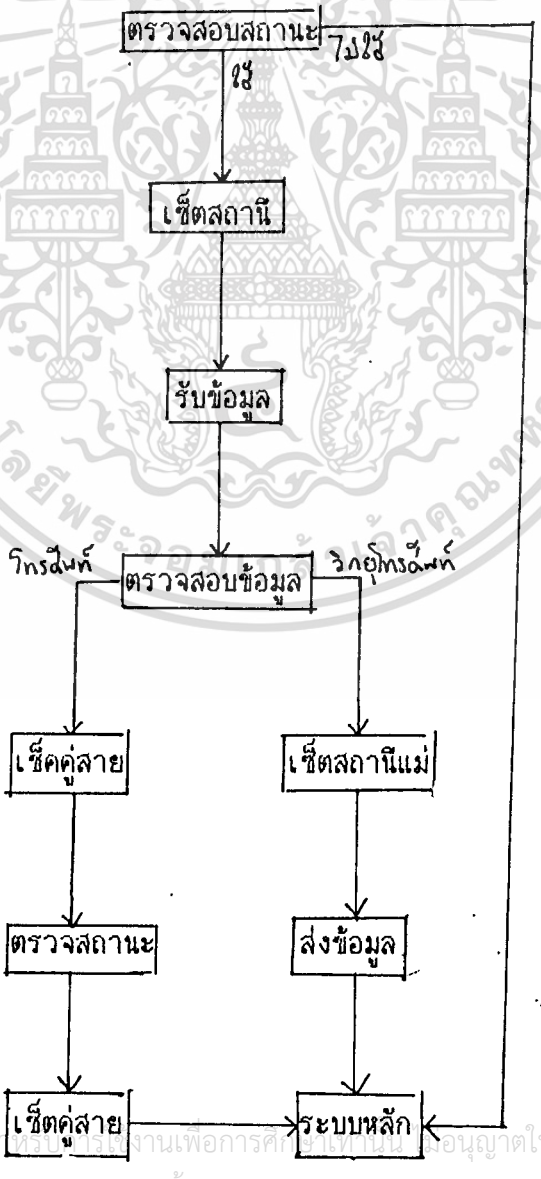
(โปรแกรมตรวจสอบสถานะ)



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

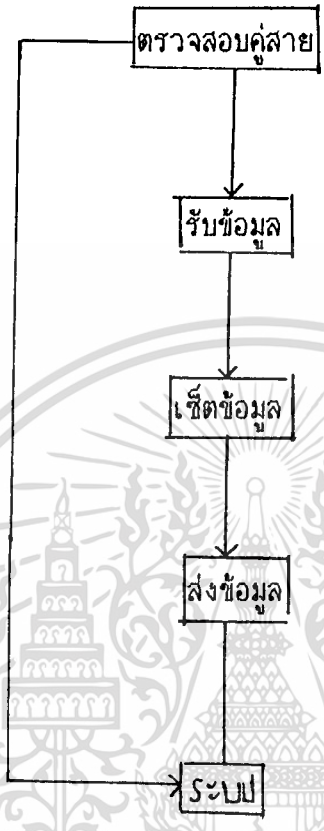


(โปรแกรมตรวจสอบสถานีแม่)



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

(โปรแกรมตรวจคู่สาย)

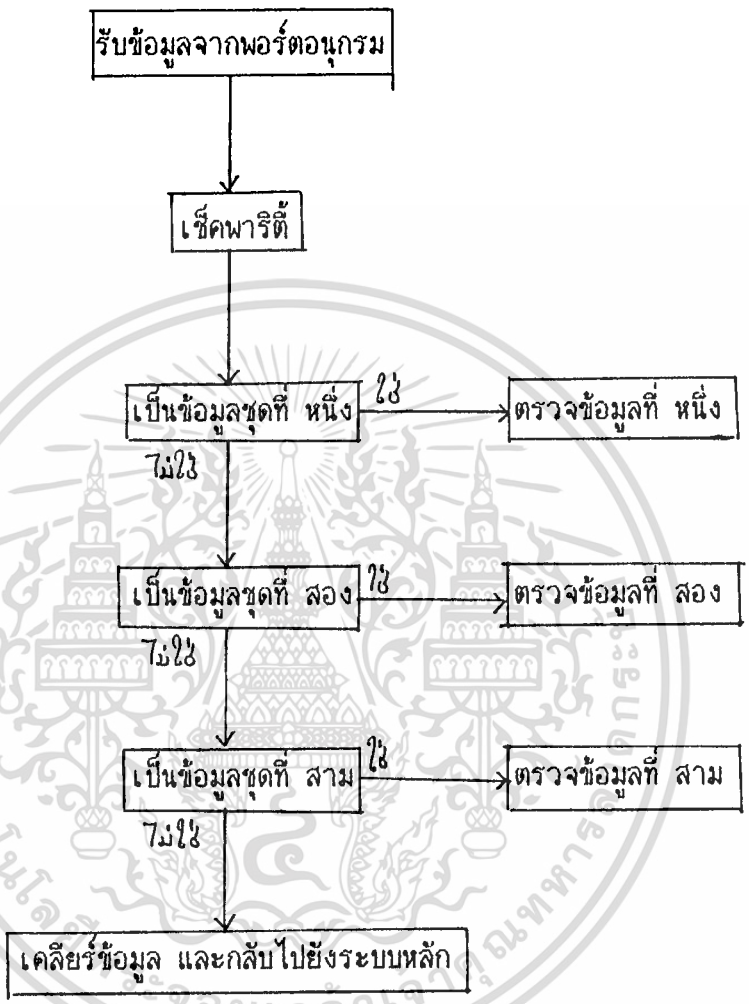


เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

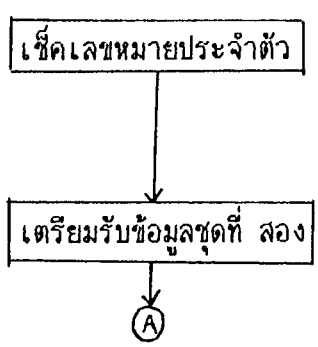
ภาคผนวกที่ 2

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

(โปรแกรมรับข้อมูลจากพอร์ตอนุกรม)

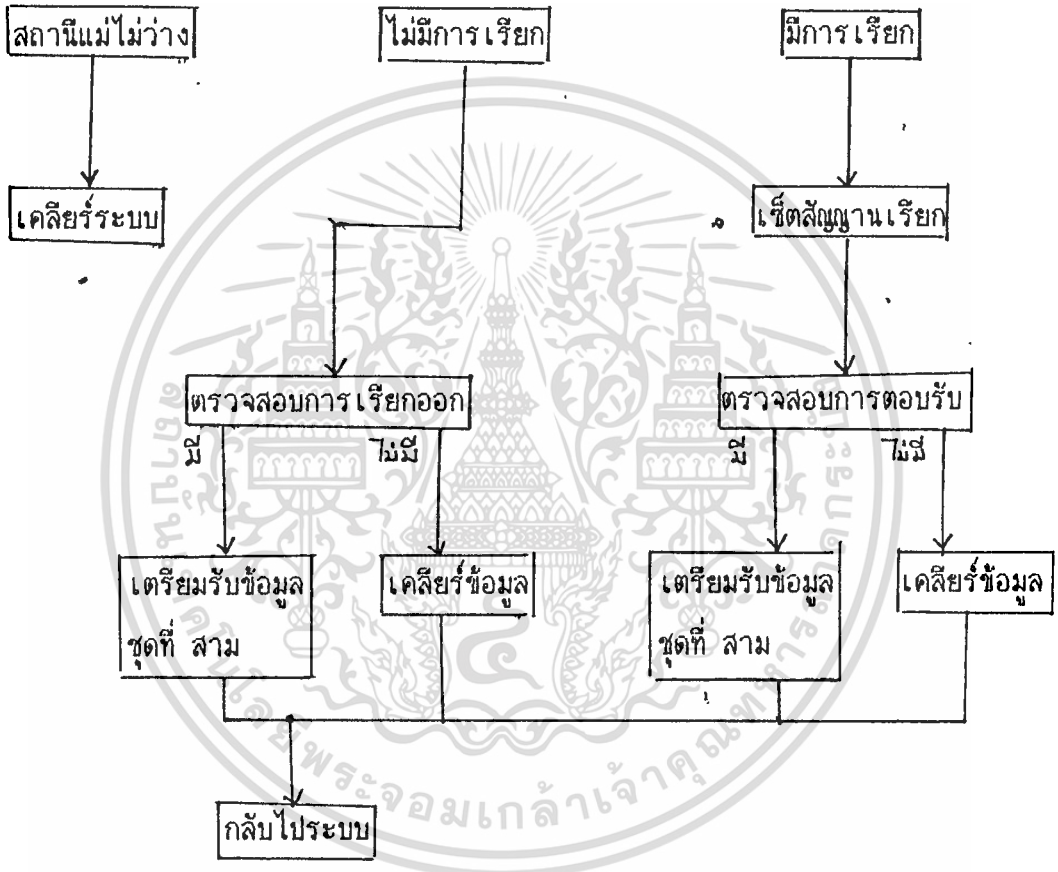


(โปรแกรมตรวจข้อมูลที่ หนึ่ง)

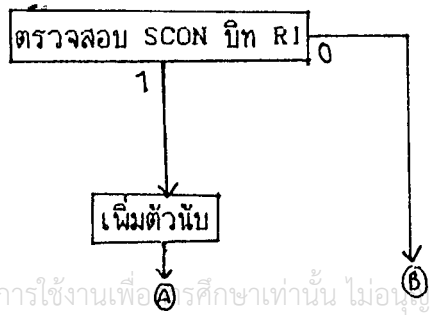


เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

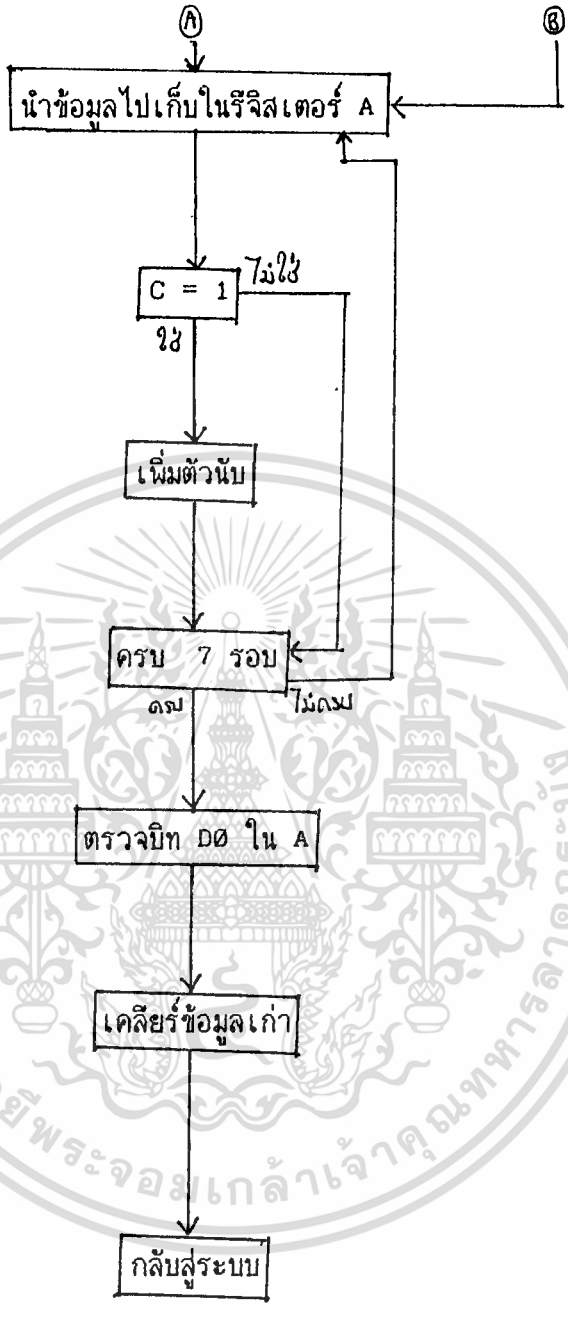
(โปรแกรมตรวจข้อมูลชุดที่ สอง)



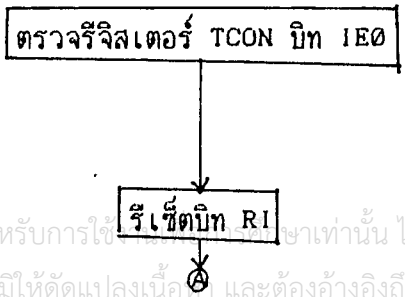
(โปรแกรมเช็คพาริตี้)



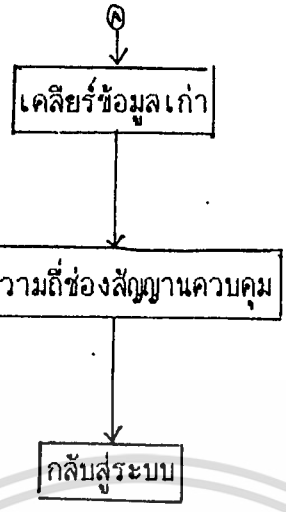
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



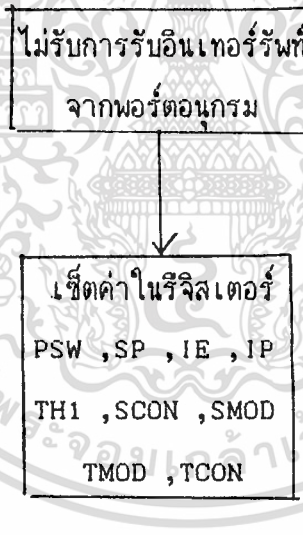
(โปรแกรมเซ็คการอินเทอร์รัพท์จากภายนอก)



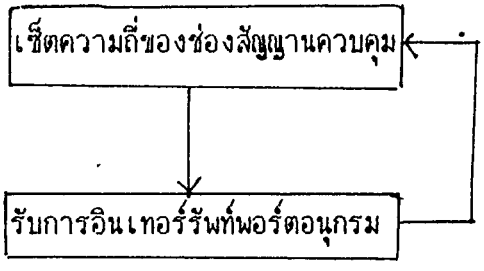
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



(โปรแกรมระบบหลัก)



(โปรแกรมระบบ)



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
{ BEGINNING PROGRAM }
            ACALL  MAIN
            ACALL  IE0
            RETI
            ACALL  INTO
            RETI
            ACALL  IE1
            RETI
            ACALL  INT1
            RETI
            JB  SCON , H01
            RETI
            CLR  SCON
            MOV  R1 , SBUFF
            ACALL  CK.PART , TY
            CLR  BIT , NORINGING
FRAME1 ,    CJNE  R0 , 00H , FRAME2
            MOV  A , R1
            SWAP  A
            ANL  A , 0FH
            CJNE  A , 00H , ERROR1
            MOV   A , R1
            ANL  A , 0FH
            CJNE  A , CODE , ERROR1
            INC  R0
 ERROR1,    RETI
 FRAME1,    CJNE  R0 , 01H , FRAME2
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                MOV   A , R1
                SWAP  A
                ANL   A , ØFH
                CJNE  A , Ø1H ,ERROR2
DATA CHECK ,INC   RØ
                MOV   A , R1
                ALN   A , ØFH
NOT BUSY ,      CJNE  A , ØØH , NOT BUSY
                SET   BIT , BUSY BASE
                MOV   RØ , ØØH
                RET I
NOT BUSY ,      CLR   BIT , NOT BUSY
CALLED    ,     CJNE  A , Ø1H , NO CALLED
                SET   BIT  RING
                MOV   IE , IE + SE.
                NOP
                RET I
                CJNE  A , Ø2H , ERROR2
NO CALLED ,     MOV   IE , IE + SE
                NOP
ERROR2    ,     RETI
FRAME3    ,     CJNE  RØ , Ø2H , ERROR3
                MOV   RØ , ØØH
                MOV   A , R1
                SWAP  A
                ALN   A , ØFH
                CJNE  A , Ø2H , ERROR3
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                MOV   A , R1
                ANL   A , ØFH
                MOV   PORT1 , A
                MOV   IE , IE P DIS SE
ERROR3      , RETI
{ PROGRAM CK PARITY }
                MOV   R2 , ØØH
                MOV   C , RB1
                JNC   NOTHING 1
                INC   R2
NOTHING 1   , MOV   R3 , Ø7H
                MOV   A , R1
CHECK       , RLC   A
                JNC   NOTHING 2
                INC   R2
NOTHING 2   , DJNZ  R3 , CHECK
                MOV   A , R2
                MOV   C , A(DØ)
                JC    ODD PARITY
                MOV   RØ , ØØH
ODD PARITY , RET
{ PROGRAM EXTERNAL INTERRUPTØ }
                JB    TCON , Ø1H
                RET
                CLR   TCON
                MOV   RØ , ØØH
                MOV   PORT1 , Ø8H
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        MOV   IE, IE + DIS. SE
        RET
{ PROGRAM   INITIALIZATION }
        MOV   IE,ØØH
        MOV   SP,2ØH
        MOV   PSW,ØØH
        MOV   TH1,CCH
        MOV   TCON,4ØH
        MOV   TMØD,2ØH
        MOV   SMØD,ØØH
        MOV   IP,11H
        MOV   PORT1,Ø8
RETURN, MOV   RØ,ØØH
        MOV   IE,9ØH
        SJMP  RETURN
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวกที่ 3

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวกที่ 4

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

intel

PRELIMINARY

8031/8051/8751
SINGLE-COMPONENT 8-BIT MICROCOMPUTER

- 8031 - Control Oriented CPU With RAM and I/O
- 8051 - An 8031 With Factory Mask-Programmable ROM
- 8751 - An 8031 With User Programmable/Erasable EPROM

- 4K x 8 ROM/EPROM
- 128 x 8 RAM
- Four 8-Bit Ports, 32 I/O Lines
- Two 16-Bit Timer/Event Counters
- High-Performance Full-Duplex Serial Channel
- External Memory Expandable to 128K
- Compatible with MCS-80®/MCS-85® Peripherals
- Boolean Processor
- MCS-48® Architecture Enhanced with:
  - Non-Paged Jumps
  - Direct Addressing
  - Four 8-Register Banks
  - Stack Depth Up to 128-Bytes
  - Multiply, Divide, Subtract, Compare
- Most Instructions Execute in 1μs
- 4μs Multiply and Divide

The Intel® 8031/8051/8751 is a stand-alone, high-performance single-chip computer fabricated with Intel's highly-reliable +5 Volt, depletion-load, N-Channel, silicon-gate HMOS technology and packaged in a 40-pin DIP. It provides the hardware features, architectural enhancements and new instructions that are necessary to make it a powerful and cost effective controller for applications requiring up to 64K bytes of program memory and/or up to 64K bytes of data storage.

The 8051/8751 contains a non-volatile 4K x 8 read-only program memory; a volatile 128 x 8 read/write data memory; 32 I/O lines; two 16-bit timer/counters; a five-source, two-priority-level, nested interrupt structure; a serial I/O port for either multi-processor communications, I/O expansion, or full duplex UART; and on-chip oscillator and clock circuits. The 8031 is identical, except that it lacks the program memory. For systems that require extra capability, the 8051 can be expanded using standard TTL compatible memories and the byte oriented MCS-80 and MCS-85 peripherals.

The 8051 microcomputer, like its 8048 predecessor, is efficient both as a controller and as an arithmetic processor. The 8051 has extensive facilities for binary and BCD arithmetic and excels in bit-handling capabilities. Efficient use of program memory results from an instruction set consisting of 44% one-byte, 41% two-byte, and 15% three-byte instructions. With a 12 MHz crystal, 58% of the instructions execute in 1μs, 40% in 2μs and multiply and divide require only 4μs. Among the many instructions added to the standard 8048 instruction set are multiply, divide, subtract and compare.

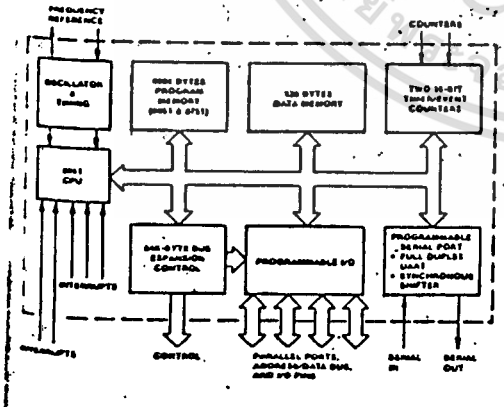


Figure 1.
Block Diagram

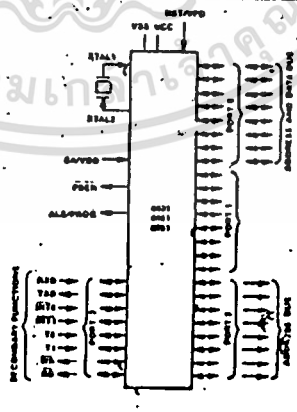


Figure 2.
Logic Symbol

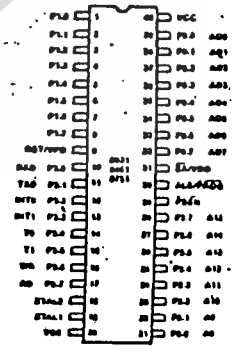


Figure 3. Pin
Configuration

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

8051 Family Pin Description

V_{SS}
Circuit ground potential.

V_{CC}
+5V power supply during operation, programming and verification.

PORT 0

Port 0 is an 8-bit open drain bidirectional I/O port. It is also the multiplexed low-order address and data bus when using external memory. It is used for data input and output during programming and verification. Port 0 can sink/source two TTL loads.

PORT 1

Port 1 is an 8-bit quasi-bidirectional I/O port. It is used for the low-order address byte during programming and verification. Port 1 can sink/source one TTL load.

PORT 2

Port 2 is an 8-bit quasi-bidirectional I/O port. It also emits the high-order address byte when accessing external memory. It is used for the high-order address and the control signals during programming and verification. Port 2 can sink/source one TTL load.

PORT 3

Port 3 is an 8-bit quasi-bidirectional I/O port. It also contains the interrupt, timer, serial port and $\overline{RD}$ and $\overline{WR}$ pins that are used by various options. The output latch corresponding to a secondary function must be programmed to a one (1) for that function to operate. Port 3 can sink/source one TTL load. The secondary functions are assigned to the pins of Port 3, as follows:

- RXD/data (P3.0). Serial port's receiver data input (asynchronous) or data input/output (synchronous).
- TXD/clock (P3.1). Serial port's transmitter data output (asynchronous) or clock output (synchronous).
- $\overline{INT0}$ (P3.2). Interrupt 0 input or gate control input for counter 0.
- $\overline{INT1}$ (P3.3). Interrupt 1 input or gate control input for counter 1.
- T0 (P3.4). Input to counter 0.
- T1 (P3.5). Input to counter 1.
- $\overline{WR}$ (P3.6). The write control signal latches the data byte from Port 0 into the External Data Memory.
- $\overline{RD}$ (P3.7). The read control signal enables External Data Memory to Port 0.

RST/V_{PD}

A low to high transition on this pin (at approximately 3V) resets the 8051. If V_{PD} is held within its spec (approximately +5V), while V_{CC} drops below spec, V_{PD} will provide standby power to the RAM. When V_{PD} is low, the RAM's current is drawn from V_{CC}. A small internal resistor permits power-on reset using only a capacitor connected to V_{CC}.

ALE/$\overline{PROG}$

Provides Address Latch Enable output used for latching the address into external memory during normal operation. Receives the program pulse input during EPROM programming.

$\overline{PSEN}$

The Program Store Enable output is a control signal that enables the external Program Memory to the bus during normal fetch operations.

$\overline{EA}$/VDD

When held at a TTL high level, the 8051 executes instructions from the internal ROM/EPROM when the PC is less than 4096. When held at a TTL low level, the 8051 fetches all instuctions from external Program Memory. The pin also receives the 21V EPROM programming supply voltage.

XTAL1

Input to the oscillator's high gain amplifier. A crystal or external source can be used.

XTAL2

Output from the oscillator's amplifier. Required when a crystal is used.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ABSOLUTE MAXIMUM RATINGS*

Ambient Temperature Under Bias ..... 0°C to 70°C
Storage Temperature ..... −65°C to +150°C
Voltage on Any Pin With
Respect to Ground (V_{SS}) ..... −0.5V to +7V
Power Dissipation ..... 2 Watts

*NOTICE: Stresses above 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 above 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.

D.C. CHARACTERISTICS (T_A = 0°C to 70°C; V_{CC} = 5V ± 5%; V_{SS} = 0V)

Symbol	Parameter	Min	Typ	Max	Units	Test Conditions
V_{IL}	Input Low Voltage (All except XTAL1)	−0.5		0.8	V	
V_{IL1}	Input Low Voltage (XTAL1)	−0.5		TBD	V	
V_{IH}	Input High Voltage (All Except XTAL1, RST/V_{PD})	2.0		V_{CC}+0.5	V	
V_{IH1}	Input High Voltage (XTAL1)	TBD		V_{CC}+0.5	V	
V_{IH2}	Input High Voltage (RST)	3.0		V_{CC} + 0.5	V	
V_{IH3}	Input High Voltage (V_{PD})	4.5		5.5	V	Power Down Only (V_{CC} = 0)
V_{OL}	Output Low Voltage (All Outputs Except Port 0)			0.45	V	I_{OL}=2 mA
V_{OL1}	Output Low Voltage (Port 0)			0.45	V	I_{OL}=4 mA
V_{OH}	Output High Voltage (All Outputs Except Port 0, ALE and $\overline{PSEN}$)	2.4			V	I_{OH}=−100 μA
V_{OH1}	Output High Voltage (ALE and $\overline{PSEN}$, Port 0 in External Bus Mode)	2.4			V	I_{OH}=−400 μA
I_{LO}	Pullup Resistor Current (P1, P2, P3)			500	μA	.45V≤ V_{IN} ≤V_{CC}
I_{LO1}	Output Leakage Current (P0)			±10	μA	.45V≤V_{IN}≤V_{CC}
I_{CC}	Power Supply Current (All Outputs Disconnected)			150	mA	T_A=25°C
I_{PD}	Power Down Supply Current			20	mA	T_A=25°C, V_{PD}=5V, V_{CC}=0V
C_{IO}	Capacitance Of I/O Buffer			10	pF	fc=1MHz

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

A.C. CHARACTERISTICS (T_A = 0°C to 70°C; V_{CC} = 5V ± 5%; V_{SS} = 0V; C_L for Port 0, ALE and $\overline{PSEN}$ Outputs = 150 pF; C_L for All Other Outputs = 80 pF)

Program Memory Characteristics

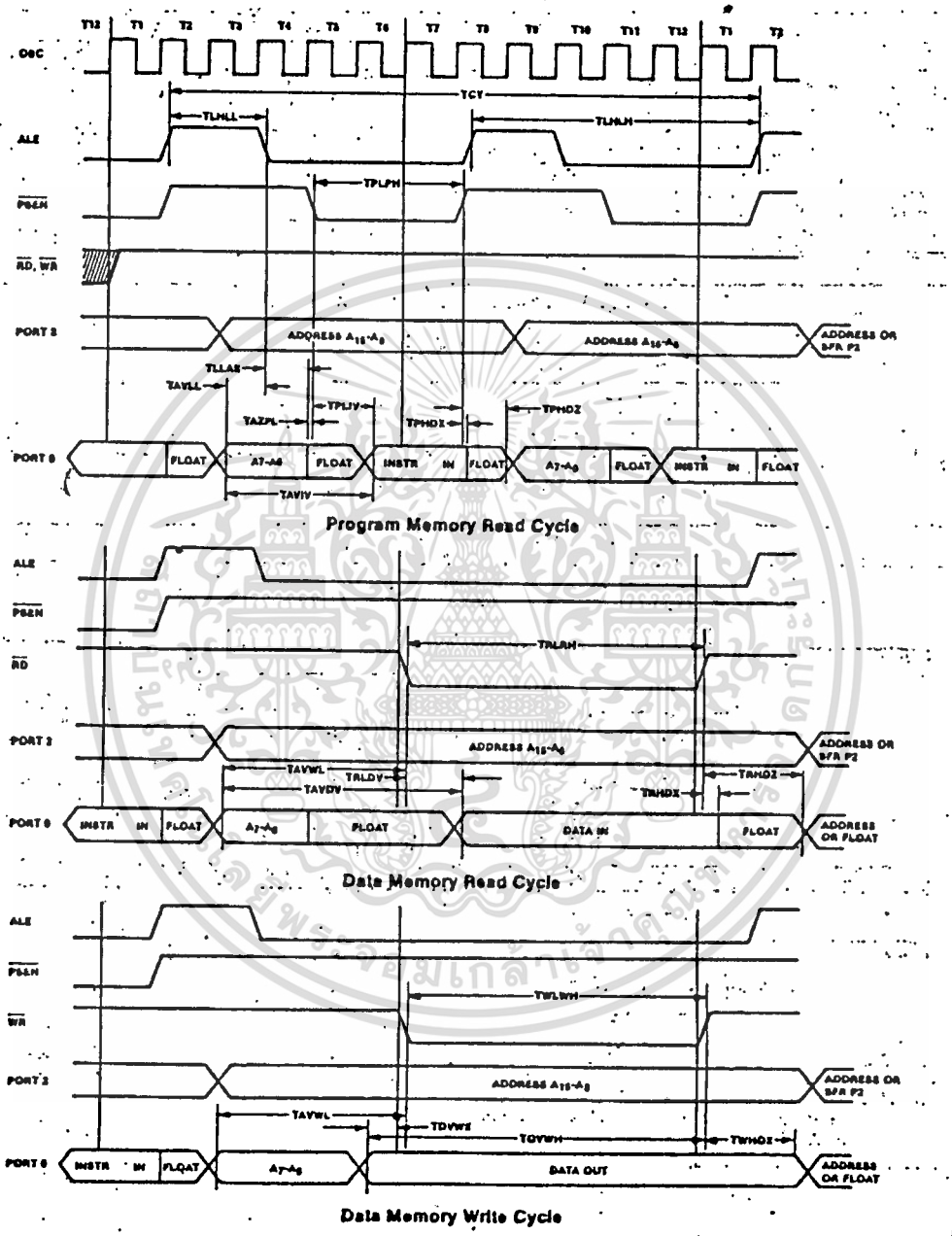
Symbol	Parameter	12 MHz Clock			Variable Clock 1/TCLCL=1.2 MHz to 12 MHz		
		Min	Max	Units	Min	Max	Units
TCLCL	Oscillator Period	83		ns			ns
TCY	Min Instruction Cycle Time	1.0		μs	12TCLCL	12TCLCL	ns
TLHLL	ALE Pulse Width	140		ns	2TCLCL-30		ns
TAVLL	Address Set Up To ALE	60		ns	TCLCL-25		ns
TLLAX	Address Hold After ALE	50		ns	TCLCL-35		ns
TPLPH	$\overline{PSEN}$ Width	230		ns	3TCLCL-20		ns
TLHLH	$\overline{PSEN}$, ALE Cycle Time	500		ns	6TCLCL		ns
TPLIV	$\overline{PSEN}$ To Valid Instruction In		150	ns		3TCLCL-100	ns
TPHDX	Input Data Hold After $\overline{PSEN}$	0		ns	0		ns
TPHDZ	Input Data Float After $\overline{PSEN}$		75	ns		TCLCL-10	ns
TAVIV	Address To Valid Instr In		320	ns >		5TCLCL-100	ns
TAZPL	Address Float To $\overline{PSEN}$	0		ns	0		ns

External Data Memory Characteristics

Symbol	Parameter	12 MHz Clock			Variable Clock		
		Min	Max	Units	Min	Max	Units
TRLRH	$\overline{RD}$ Pulse Width	400		ns	6TCLCL-100		ns
TWLWH	$\overline{WR}$ Pulse Width	400		ns	6TCLCL-100		ns
TRLDV	$\overline{RD}$ To Valid Data In		250	ns		5TCLCL-170	ns
TRHDX	Data Hold After $\overline{RD}$	0		ns	0		ns
TRHDZ	Data Float After $\overline{RD}$		100	ns		2TCLCL-70	ns
TAVDV	Address To Valid Data In		600	ns		9TCLCL-150	ns
TAVWL	Address To $\overline{WR}$ or $\overline{RD}$	200		ns	4TCLCL-130		ns
TDVWX	Data Valid To $\overline{WR}$ Transition			ns			ns
TQVWH	Data Setup Before $\overline{WR}$	400		ns	7TCLCL-180		ns
TWHQX	Data Held After $\overline{WR}$	80		ns	2TCLCL-90		ns

NOTE:
There are 2 to 8 ALE cycles per instruction. Clocks and state timing are shown on the timing diagram for reference purposes only. They are not accessible outside the package. TCY is the minimum instruction cycle time which consists of 12 oscillator clocks or two ALE cycles. Address setup and hold time from ALE are the same for data and program memory.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

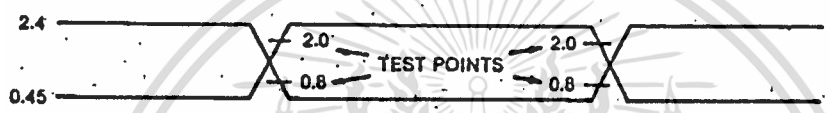


เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

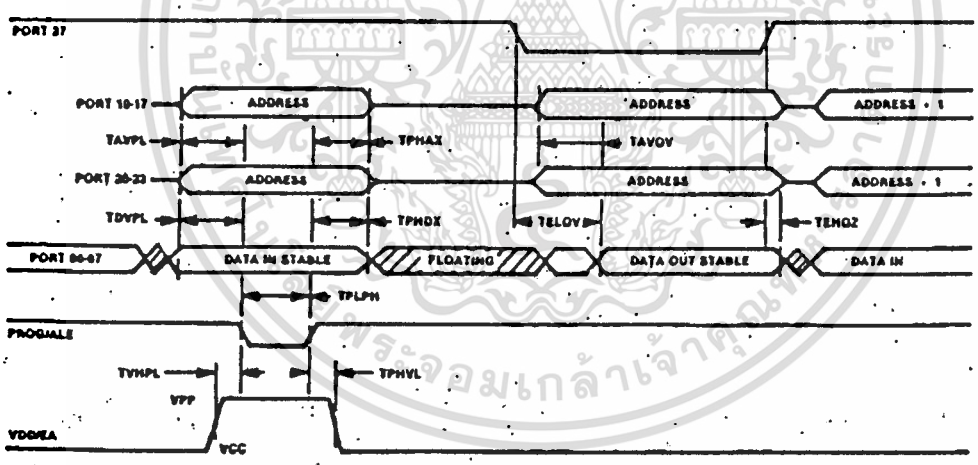
PROGRAM VERIFY

Symbol	Parameter	12 MHz Clock		Variable Clock	
		Min.	Max.	Min.	Max.
TDVPL	Data Setup to PROG	10μs		3 TCY + 10μs	
TPHDX	Data Hold from PROG	10μs		3 TCY + 10μs	
TAVQV	Address to Data Valid		10μs		3 TCY + 10μs
TELQV	Output Enable (P27) to Data Valid		10μs		3 TCY + 10μs
TEHQZ	Output Enable Off to Data Float	0	10μs	0	3 TCY + 10μs
TVHPL	VDD Setup to PROG	10μs		10μs	
TPHVL	VDD Hold after PROG	10μs		10μs	
TPLPH	PROG Width	49ms	51ms	49ms	51ms

Input and Output Waveforms for A.C. Tests



PROGRAM VERIFY



NOTES: 1) PSEN = VIL, VPD/RST = VIH1
2) MSB of address is Port 23, LSB is Port 0
3) All levels are VIL, VIH, VOL, VOH except VDD/EA.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

intel

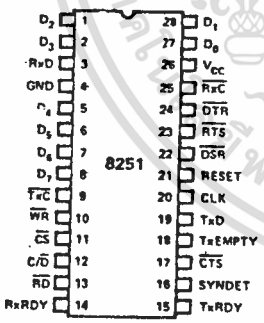
Silicon Gate MOS 8251

PROGRAMMABLE COMMUNICATION INTERFACE

- **Synchronous and Asynchronous Operation**
  - **Synchronous:**
    - 5-8 Bit Characters
    - Internal or External Character Synchronization
    - Automatic Sync Insertion
  - **Asynchronous:**
    - 5-8 Bit Characters
    - Clock Rate — 1, 16 or 64 Times Baud Rate
    - Break Character Generation
    - 1, 1½, or 2 Stop Bits
    - False Start Bit Detection
- **Baud Rate —DC to 56k Baud (Sync Mode)**
  DC to 9.6k Baud (Async Mode)
- **Full Duplex, Double Buffered, Transmitter and Receiver**
- **Error Detection — Parity, Overrun, and Framing**
- **Fully Compatible with 8080 CPU**
- **28-Pin DIP Package**
- **All Inputs and Outputs Are TTL Compatible**
- **Single 5 Volt Supply**
- **Single TTL Clock**

The 8251 is a Universal Synchronous/Asynchronous Receiver/Transmitter (USART) Chip designed for data communications in microcomputer systems. The USART is used as a peripheral device and is programmed by the CPU to operate using virtually any serial data transmission technique presently in use (including IBM Bi-Sync). The USART accepts data characters from the CPU in parallel format and then converts them into a continuous serial data stream for transmission. Simultaneously it can receive serial data streams and convert them into parallel data characters for the CPU. The USART will signal the CPU whenever it can accept a new character for transmission or whenever it has received a character for the CPU. The CPU can read the complete status of the USART at any time. These include data transmission errors and control signals such as SYNDET, TxEMPT. The chip is constructed using N-channel silicon gate technology.

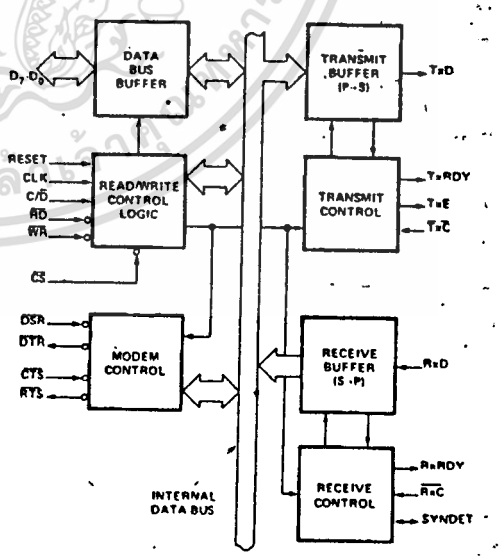
PIN CONFIGURATION



Pin Name	Pin Function
D_7-D_0	Data Bus (8 bits)
C/$\overline{D}$	Control or Data is to be Written or Read
$\overline{RD}$	Read Data Command
$\overline{WR}$	Write Data or Control Command
$\overline{CS}$	Chip Enable
CLK	Clock Pulse (TTL)
RESET	Reset
$\overline{TxC}$	Transmitter Clock
TxD	Transmitter Data
$\overline{RxC}$	Receiver Clock
RxD	Receiver Data
RxRDY	Receiver Ready (has character for 8080)
TxRDY	Transmitter Ready (ready for char. from 8080)

Pin Name	Pin Function
$\overline{DSR}$	Data Set Ready
$\overline{DTR}$	Data Terminal Ready
SYNDET	Sync Detect
$\overline{RTS}$	Request to Send Data
$\overline{CTS}$	Clear to Send Data
TxE	Transmitter Empty
VCC	+5 Volt Supply
GND	Ground

BLOCK DIAGRAM



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

8251 BASIC FUNCTIONAL DESCRIPTION

General

The 8251 is a Universal Synchronous/Asynchronous Receiver/Transmitter designed specifically for the 8080 Microcomputer System. Like other I/O devices in the 8080 Microcomputer System its functional configuration is programmed by the systems software for maximum flexibility. The 8251 can support virtually any serial data technique currently in use (including IBM "bi-sync").

In a communication environment an interface device must convert parallel format system data into serial format for transmission and convert incoming serial format data into parallel system data for reception. The interface device must also delete or insert bits or characters that are functionally unique to the communication technique. In essence, the interface should appear "transparent" to the CPU, a simple input or output of byte-oriented system data.

Data Bus Buffer

This 3-state, bi-directional, 8-bit buffer is used to interface the 8251 to the 8080 system Data Bus. Data is transmitted or received by the buffer upon execution of INput or OUTput instructions of the 8080 CPU. Control words, Command words and Status information are also transferred through the Data Bus Buffer.

Read/Write Control Logic

This functional block accepts inputs from the 8080 Control bus and generates control signals for overall device operation. It contains the Control Word Register and Command Word Register that store the various control formats for device functional definition.

RESET (Reset)

A "high" on this input forces the 8251 into an "Idle" mode. The device will remain at "Idle" until a new set of control words is written into the 8251 to program its functional definition.

CLK (Clock)

The CLK input is used to generate internal device timing and is normally connected to the Phase 2 (TTL) output of the 8224 Clock Generator. No external inputs or outputs are referenced to CLK but the frequency of CLK must be greater than 30 times the Receiver or Transmitter clock inputs for synchronous mode (4 times for asynchronous mode).

$\overline{WR}$ (Write)

A "low" on this input informs the 8251 that the CPU is outputting data or control words, in essence, the CPU is writing out to the 8251.

$\overline{RD}$ (Read)

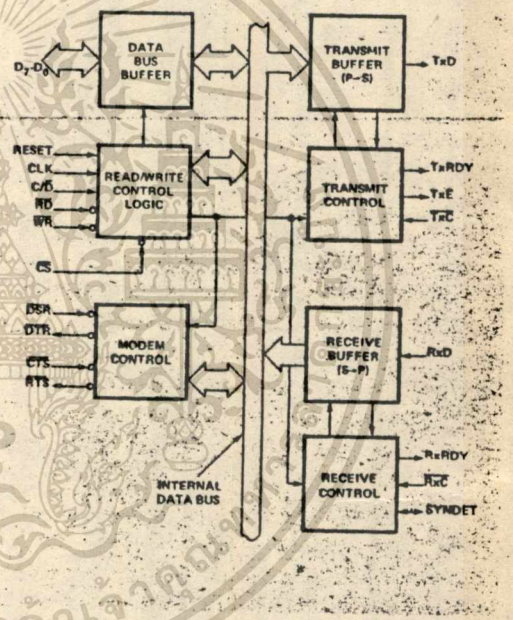
A "low" on this input informs the 8251 that the CPU is inputting data or status information, in essence, the CPU is reading from the 8251.

C/$\overline{D}$ (Control/Data)

This input, in conjunction with the $\overline{WR}$ and $\overline{RD}$ inputs informs the 8251 that the word on the Data Bus is either a data character, control word or status information.
1 = CONTROL 0 = DATA

$\overline{CS}$ (Chip Select)

A "low" on this input enables the 8251. No reading or writing will occur unless the device is selected.



C/$\overline{D}$	$\overline{RD}$	$\overline{WR}$	$\overline{CS}$	
0	0	1	0	8251 → DATA BUS
0	1	0	0	DATA BUS → 8251
1	0	1	0	STATUS → DATA BUS
1	1	0	0	DATA BUS → CONTROL
X	X	X	1	DATA BUS → 3-STATE

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Modem Control

The 8251 has a set of control inputs and outputs that can be used to simplify the interface to almost any Modem. The modem control signals are general purpose in nature and can be used for functions other than Modem control, if necessary.

$\overline{DSR}$ (Data Set Ready)

The $\overline{DSR}$ input signal is general purpose in nature. Its condition can be tested by the CPU using a Status Read operation. The $\overline{DSR}$ input is normally used to test Modem conditions such as Data Set Ready.

$\overline{DTR}$ (Data Terminal Ready)

The $\overline{DTR}$ output signal is general purpose in nature. It can be set "low" by programming the appropriate bit in the Command Instruction word. The $\overline{DTR}$ output signal is normally used for Modem control such as Data Terminal Ready or Rate Select.

$\overline{RTS}$ (Request to Send)

The $\overline{RTS}$ output signal is general purpose in nature. It can be set "low" by programming the appropriate bit in the Command Instruction word. The $\overline{RTS}$ output signal is normally used for Modem control such as Request to Send.

$\overline{CTS}$ (Clear to Send)

A "low" on this input enables the 8251 to transmit data (serial) if the Tx EN bit in the Command byte is set to a "one."

Transmitter Buffer

The Transmitter Buffer accepts parallel data from the Data Bus Buffer, converts it to a serial bit stream, inserts the appropriate characters or bits (based on the communication technique) and outputs a composite serial stream of data on the TxD output pin.

Transmitter Control

The Transmitter Control manages all activities associated with the transmission of serial data. It accepts and issues signals both externally and internally to accomplish this function.

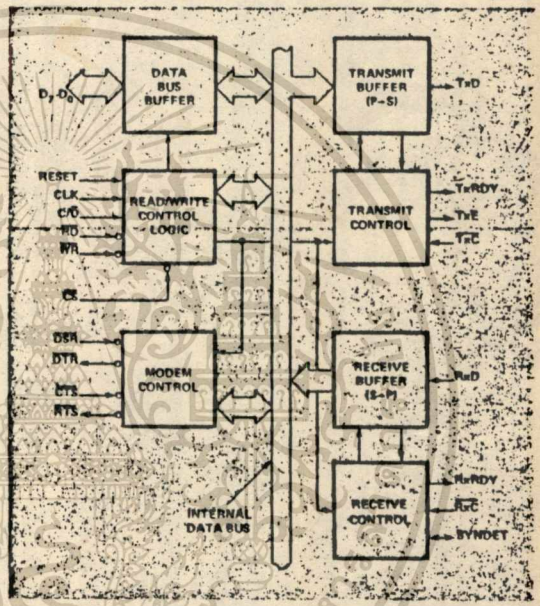
TxRDY (Transmitter Ready)

This output signals the CPU that the transmitter is ready to accept a data character. It can be used as an interrupt to the system or for the Polled operation the CPU can check TxRDY using a status read operation. TxRDY is automatically reset when a character is loaded from the CPU.

TxE (Transmitter Empty)

When the 8251 has no characters to transmit, the TxE output will go "high". It resets automatically upon receiving a character from the CPU. TxE can be used to indicate the end of a transmission mode, so that the CPU "knows" when to "turn the line around" in the half-duplexed operational mode.

In SYNChronous mode, a "high" on this output indicates that a character has not been loaded and the SYNC character or characters are about to be transmitted automatically as "fillers".



$\overline{TxC}$ (Transmitter Clock)

The Transmitter Clock controls the rate at which the character is to be transmitted. In the Synchronous transmission mode, the frequency of $\overline{TxC}$ is equal to the actual Baud Rate (1X). In Asynchronous transmission mode, the frequency of $\overline{TxC}$ is a multiple of the actual Baud Rate. A portion of the mode instruction selects the value of the multiplier; it can be 1x, 16x or 64x the Baud Rate.

For Example:

If Baud Rate equals 110 Baud,
$\overline{TxC}$ equals 110 Hz (1x)
$\overline{TxC}$ equals 1.76 kHz (16x)
$\overline{TxC}$ equals 7.04 kHz (64x).
If Baud Rate equals 9600 Baud,
$\overline{TxC}$ equals 614.4 kHz (64x).

The falling edge of $\overline{TxC}$ shifts the serial data out of the 8251.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Receiver Buffer

The Receiver accepts serial data, converts this serial input to parallel format, checks for bits or characters that are unique to the communication technique and sends an "assembled" character to the CPU. Serial data is input to the RxD pin.

Receiver Control

This functional block manages all receiver-related activities.

RxRDY (Receiver Ready)

This output indicates that the 8251 contains a character that is ready to be input to the CPU. RxRDY can be connected to the interrupt structure of the CPU or for Polled operation the CPU can check the condition of RxRDY using a status read operation. RxRDY is automatically reset when the character is read by the CPU.

$\overline{RxC}$ (Receiver Clock)

The Receiver Clock controls the rate at which the character is to be received. In Synchronous Mode, the frequency of $\overline{RxC}$ is equal to the actual Baud Rate (1x). In Asynchronous Mode, the frequency of $\overline{RxC}$ is a multiple of the actual Baud Rate. A portion of the mode instruction selects the value of the multiplier; it can be 1x, 16x or 64x the Baud Rate.

For Example:
If Baud Rate equals 300 Baud,
$\overline{RxC}$ equals 300 Hz (1x)
$\overline{RxC}$ equals 4800 Hz (16x)
$\overline{RxC}$ equals 19.2 kHz (64x).
If Baud Rate equals 2400 Baud,
$\overline{RxC}$ equals 2400 Hz (1x)
$\overline{RxC}$ equals 38.4 kHz (16x)
$\overline{RxC}$ equals 153.6 kHz (64x).

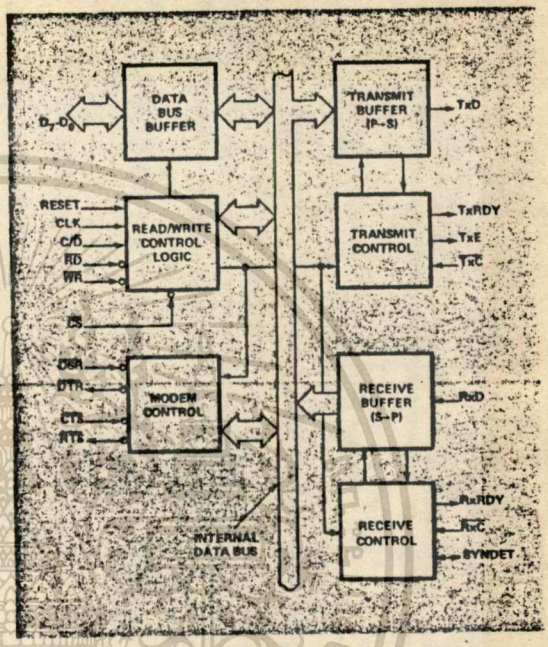
Data is sampled into the 8251 on the rising edge of $\overline{RxC}$.

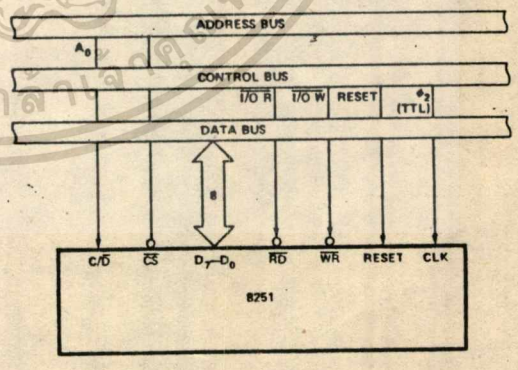
NOTE: In most communications systems, the 8251 will be handling both the transmission and reception operations of a single link. Consequently, the Receive and Transmit Baud Rates will be the same. Both $\overline{TxC}$ and $\overline{RxC}$ will require identical frequencies for this operation and can be tied together and connected to a single frequency source (Baud Rate Generator) to simplify the interface.

SYNDET (SYNC Detect)

This pin is used in SYNChronous Mode only. It is used as either input or output, programmable through the Control Word. It is reset to "low" upon RESET. When used as an output (internal Sync mode), the SYNDET pin will go "high" to indicate that the 8251 has located the SYNC character in the Receive mode. If the 8251 is programmed to use double Sync characters (bi-sync), then SYNDET will go "high" in the middle of the last bit of the second Sync character. SYNDET is automatically reset upon a Status Read operation.

When used as an input, (external SYNC detect mode), a positive going signal will cause the 8251 to start assembling data characters on the falling edge of the next $\overline{RxC}$. Once in SYNC, the "high" input signal can be removed. The duration of the high signal should be at least equal to the period of $\overline{RxC}$.





8251 Interface to 8080 Standard System Bus

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

DETAILED OPERATION DESCRIPTION

General

The complete functional definition of the 8251 is programmed by the systems software. A set of control words must be sent out by the CPU to initialize the 8251 to support the desired communications format. These control words will program the: BAUD RATE, CHARACTER LENGTH, NUMBER OF STOP BITS, SYNCHRONOUS or ASYNCHRONOUS OPERATION, EVEN/ODD PARITY etc. In the Synchronous Mode, options are also provided to select either internal or external character synchronization.

Once programmed, the 8251 is ready to perform its communication functions. The TxRDY output is raised "high" to signal the CPU that the 8251 is ready to receive a character. This output (TxRDY) is reset automatically when the CPU writes a character into the 8251. On the other hand, the 8251 receives serial data from the MODEM or I/O device, upon receiving an entire character the RxRDY output is raised "high" to signal the CPU that the 8251 has a complete character ready for the CPU to fetch. RxRDY is reset automatically upon the CPU read operation.

The 8251 cannot begin transmission until the TxEN(Transmitter Enable) bit is set in the Command Instruction and it has received a Clear To Send (CTS) input. The TxD output will be held in the marking state upon Reset.

Programming the 8251

Prior to starting data transmission or reception, the 8251 must be loaded with a set of control words generated by the CPU. These control signals define the complete functional definition of the 8251 and must immediately follow a Reset operation (internal or external).

The control words are split into two formats:

1. Mode Instruction
2. Command Instruction

Mode Instruction

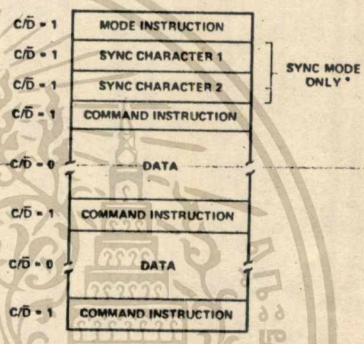
This format defines the general operational characteristics of the 8251. It must follow a Reset operation (internal or external). Once the Mode instruction has been written into the 8251 by the CPU, SYNC characters or Command instructions may be inserted.

Command Instruction

This format defines a status word that is used to control the actual operation of the 8251.

Both the Mode and Command instructions must conform to a specified sequence for proper device operation. The Mode Instruction must be inserted immediately following a Reset operation, prior to using the 8251 for data communication.

All control words written into the 8251 after the Mode Instruction will load the Command Instruction. Command Instructions can be written into the 8251 at any time in the data block during the operation of the 8251. To return to the Mode Instruction format a bit in the Command Instruction word can be set to initiate an internal Reset operation which automatically places the 8251 back into the Mode Instruction format. Command Instructions must follow the Mode Instructions or Sync characters.

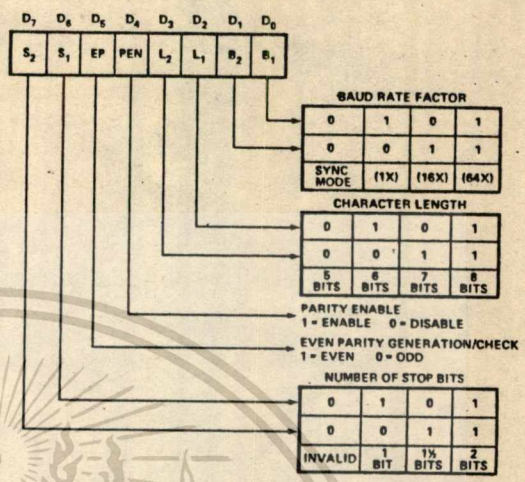


Typical Data Block

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Mode Instruction Definition

The 8251 can be used for either Asynchronous or Synchronous data communication. To understand how the Mode Instruction defines the functional operation of the 8251 the designer can best view the device as two separate components sharing the same package. One Asynchronous the other Synchronous. The format definition can be changed "on the fly" but for explanation purposes the two formats will be isolated.



Mode Instruction Format, Asynchronous Mode

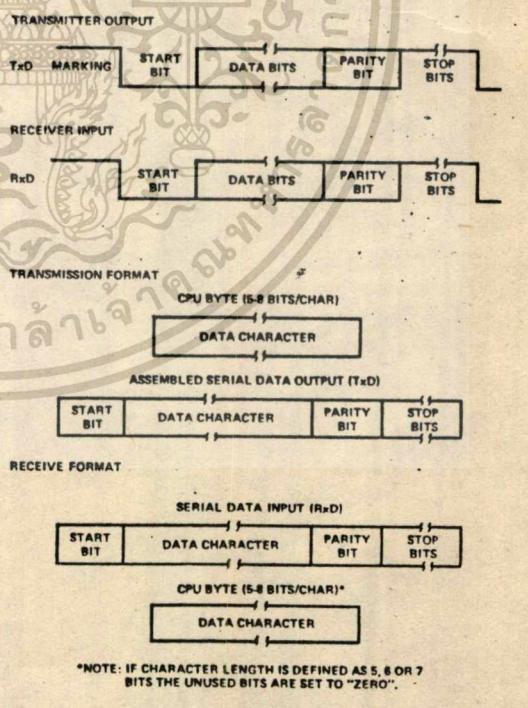
Asynchronous Mode (Transmission)

Whenever a data character is sent by the CPU the 8251 automatically adds a Start bit (low level) and the programmed number of Stop bits to each character. Also, an even or odd Parity bit is inserted prior to the Stop bit(s), as defined by the Mode Instruction. The character is then transmitted as a serial data stream on the TxD output. The serial data is shifted out on the falling edge of $\overline{TxC}$ at a rate equal to 1, 1/16, or 1/64 that of the $\overline{TxC}$, as defined by the Mode Instruction. BREAK characters can be continuously sent to the TxD if commanded to do so.

When no data characters have loaded into the 8251 the TxD output remains "high" (marking) unless a Break (continuously low) has been programmed.

Asynchronous Mode (Receive)

The RxD line is normally high. A falling edge on this line triggers the beginning of a START bit. The validity of this START bit is checked by again strobing this bit at its nominal center. If a low is detected again, it is a valid START bit, and the bit counter will start counting. The bit counter locates the center of the data bits, the parity bit (if it exists) and the stop bits. If parity error occurs, the parity error flag is set. Data and parity bits are sampled on the RxD pin with the rising edge of $\overline{RxC}$. If a low level is detected as the STOP bit, the Framing Error flag will be set. The STOP bit signals the end of a character. This character is then loaded into the parallel I/O buffer of the 8251. The RxRDY pin is raised to signal the CPU that a character is ready to be fetched. If a previous character has not been fetched by the CPU, the present character replaces it in the I/O buffer, and the OVERRUN flag is raised (thus the previous character is lost). All of the error flags can be reset by a command instruction. The occurrence of any of these errors will not stop the operation of the 8251.



Asynchronous Mode

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Synchronous Mode (Transmission)

The TxD output is continuously high until the CPU sends its first character to the 8251 which usually is a SYNC character. When the $\overline{CTS}$ line goes low, the first character is serially transmitted out. All characters are shifted out on the falling edge of $\overline{TxC}$. Data is shifted out at the same rate as the $\overline{TxC}$.

Once transmission has started, the data stream at TxD output must continue at the $\overline{TxC}$ rate. If the CPU does not provide the 8251 with a character before the 8251 becomes empty, the SYNC characters (or character if in single SYNC word mode) will be automatically inserted in the TxD data stream. In this case, the TxEMPTY pin is raised high to signal that the 8251 is empty and SYNC characters are being sent out. The TxEMPTY pin is internally reset by the next character being written into the 8251.

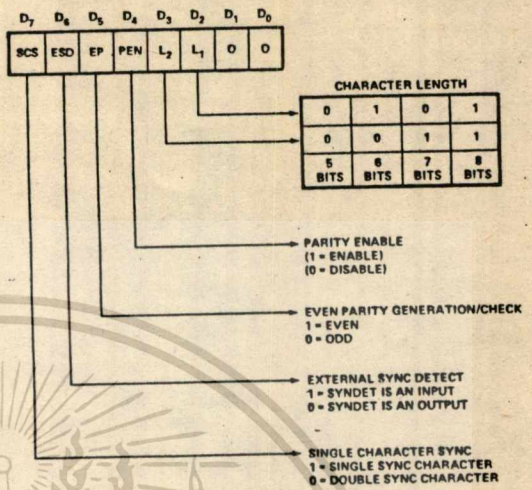
Synchronous Mode (Receive)

In this mode, character synchronization can be internally or externally achieved. If the internal SYNC mode has been programmed, the receiver starts in a HUNT mode. Data on the RxD pin is then sampled in on the rising edge of $\overline{RxC}$. The content of the Rx buffer is continuously compared with the first SYNC character until a match occurs. If the 8251 has been programmed for two SYNC characters, the subsequent received character is also compared; when both SYNC characters have been detected, the USART ends the HUNT mode and is in character synchronization. The SYNDET pin is then set high, and is reset automatically by a STATUS READ.

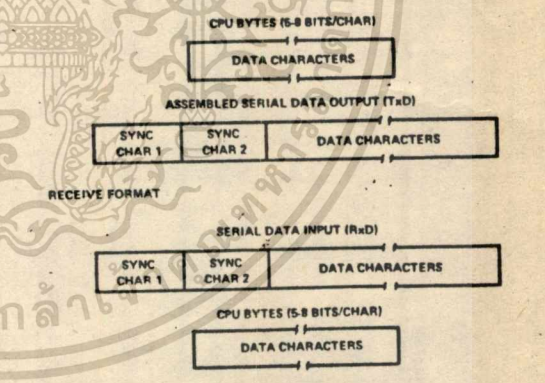
In the external SYNC mode, synchronization is achieved by applying a high level on the SYNDET pin. The high level can be removed after one $\overline{RxC}$ cycle.

Parity error and overrun error are both checked in the same way as in the Asynchronous Rx mode.

The CPU can command the receiver to enter the HUNT mode if synchronization is lost.



Mode Instruction Format, Synchronous Mode



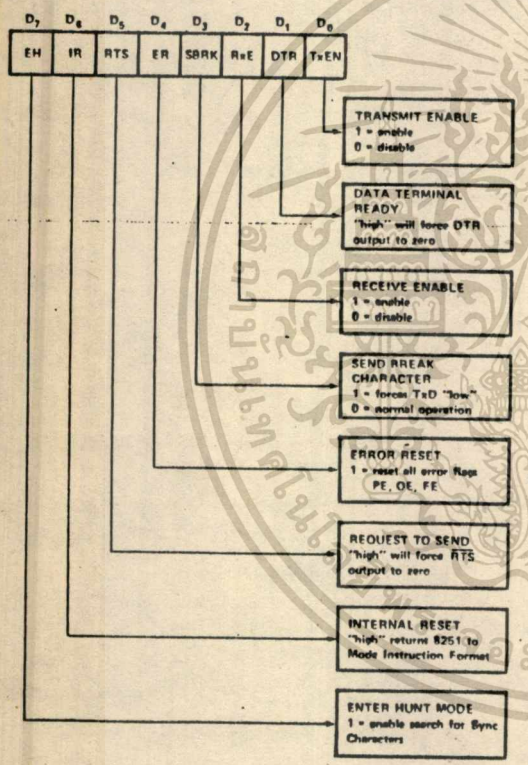
Synchronous Mode, Transmission Format

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

COMMAND INSTRUCTION DEFINITION

Once the functional definition of the 8251 has been programmed by the Mode Instruction and the Sync Characters are loaded (if in Sync Mode) then the device is ready to be used for data communication. The Command Instruction controls the actual operation of the selected format. Functions such as: Enable Transmit/Receive, Error Reset and Modem Controls are provided by the Command Instruction.

Once the Mode Instruction has been written into the 8251 and Sync characters inserted, if necessary, then all further "control writes" ($C/\overline{D}$ = 1) will load the Command Instruction. A Reset operation (internal or external) will return the 8251 to the Mode Instruction Format.



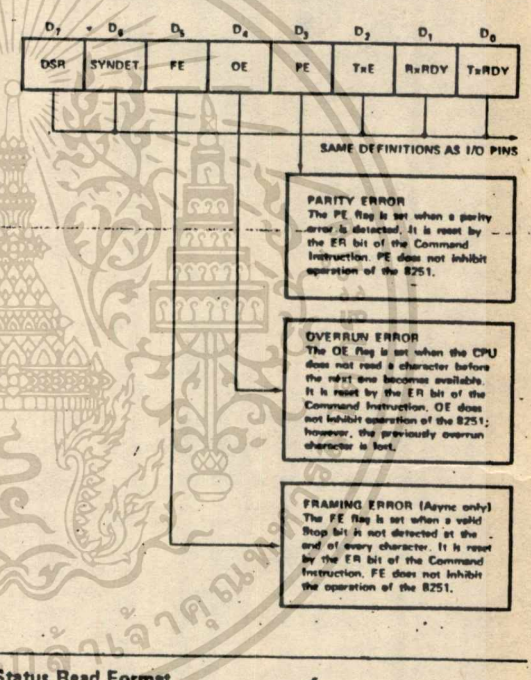
Command Instruction Format

STATUS READ DEFINITION

In data communication systems it is often necessary to examine the "status" of the active device to ascertain if errors have occurred or other conditions that require the processor's attention. The 8251 has facilities that allow the programmer to "read" the status of the device at any time during the functional operation.

A normal "read" command is issued by the CPU with the C/D input at one to accomplish this function.

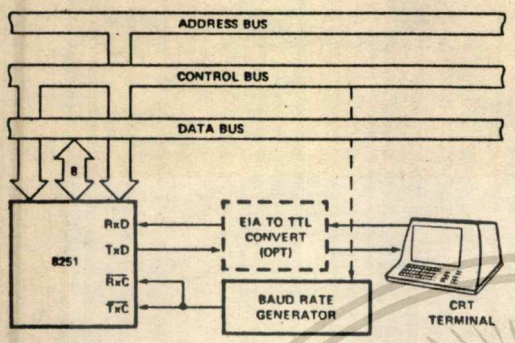
Some of the bits in the Status Read Format have identical meanings to external output pins so that the 8251 can be used in a completely Polled environment or in an interrupt driven environment.



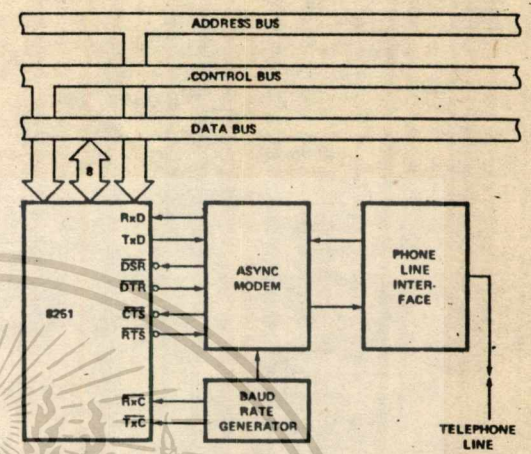
Status Read Format

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

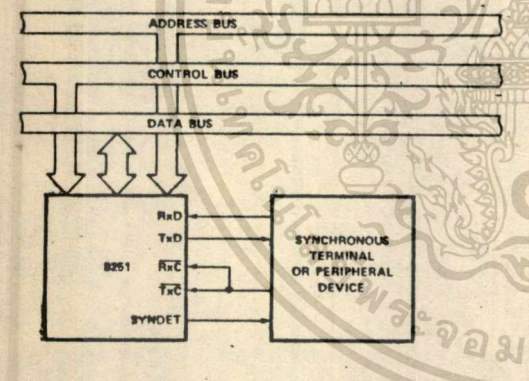
APPLICATIONS OF THE 8251



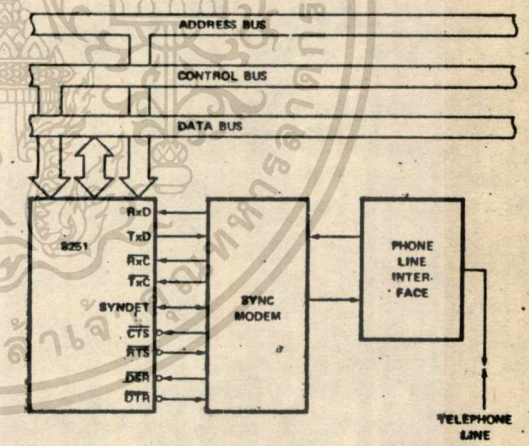
Asynchronous Serial Interface to CRT Terminal, DC-9600 Baud



Asynchronous Interface to Telephone Lines



Synchronous Interface to Terminal or Peripheral Device



Synchronous Interface to Telephone Lines

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

D.C. Characteristics: $T_A = 0°C$ to $70°C$; $V_{CC} = +5V \pm 5\%$; $V_{SS} = 0V$

Symbol	Parameter	Min.	Typ.	Max.	Unit	Test Conditions
V_{IL}	Input Low Voltage			.8	V	
V_{IH}	Input High Voltage	2.0			V	
V_{OL}	Output Low Voltage			.4	V	I_{OL}=2.0mA (DB_{0-7}), 1.6mA (Others)
V_{OH}	Output High Voltage	2.4			V	I_{OH}=150μA (DB_{0-7}), 100μA (Others)
I_{CC}	Power Supply Current		80		mA	
I_{LI}	Input Load Current		10		μA	V_{IN}=0V to 5.25V
I_{LOL}	Output Leakage Current (DB Low)		-100		μA	V_{OUT}=0.4V
I_{LOH}	Output Leakage Current (DB High)		+10		μA	V_{OUT}=V_{CC}

A.C. Characteristics: $T_A = 0°C$ to $70°C$; $V_{CC} = +5V \pm 5\%$; $V_{SS} = 0V$

Symbol	Parameter	Min.	Typ.	Max.	Unit	Test Conditions
t_{CY}	Clock Period		480		ns	
t_{CW}	Clock Pulse Width		220		ns	
t_{CD}	Clock Pulse Delay		260		ns	
t_R, t_F	Clock Rise and Fall Time		50		ns	
t_{WR}	Write Pulse Width		400		ns	
t_{DS}	Data Set-Up Time for $\overline{WRITE}$		300		ns	
t_{DH}	Data Hold Time for $\overline{WRITE}$		20		ns	
t_{DD}	Data Delay from $\overline{READ}$		350		ns	C_L = 150pF
t_{FD}	$\overline{READ}$ to Data Floating		160		ns	C_L = 150pF
t_{AW}	Address Stable before $\overline{WRITE}$	0			ns	
t_{WA}	Address Hold Time for $\overline{WRITE}$		40		ns	
t_{AR}	Address Stable before $\overline{READ}$		0		ns	
t_{RA}	Address Hold Time for $\overline{READ}$		0		ns	
t_{DTx}	TxD Delay from Rising Edge of TxC		300		ns	
t_{SRx}	Rx Data Set-Up Time to Sampling Pulse		500		ns	
t_{HRx}	Rx Data Hold Time to Sampling Pulse		6		CLK Period	
f_{Tx}	Transmitter Input Clock Frequency 1X Baud Rate 16X and 64X Baud Rate		 56 615		 KHz KHz	
f_{Rx}	Receiver Input Clock Frequency 1X Baud Rate 16X and 64X Baud Rate		 56 615		 KHz KHz	
t_{Tx}	TxRDY Delay from Center of Data Bit		6		CLK Period	C_L=50pF
t_{Rx}	RxRDY Delay from Center of Data Bit		6		CLK Period	
t_{IS}	Internal SYNDET Delay from Center of Data Bit		6		CLK Period	
t_{ES}	External SYNDET Set-Up Time before Rising Edge of RxC		6		CLK Period	

THESE SPECIFICATIONS ARE SUBJECT TO CHANGE WITHOUT NOTICE
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้