สำนักหอสมุดกลาง พระจอมเกล้าลาดกระบัง

การจัดการที่จอดรถ

CARS PARKING MANAGEMENT

T117474

โดย

นายเชิดชัย คำภู

นายอนุชา จารุสิน

เลขหมู่.....
เลขทะเบียน.....117474
วัน,เดือน,ปี.....- 5 ส.ค. 2554

b.....1233450X
i.....

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต

สาขาวิชาวิศวกรรมโทรคมนาคม

คณะวิศวกรรมศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2553

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การจัดการที่จอดรถ

CARS PARKING MANAGEMENT

โดย

นายเชิดชัย คำภู 49010213

นายอนุชา จารุสิน 50011838

อาจารย์ที่ปรึกษา

ผศ. สมภพ แก้วมีชัย

b 1233750X

i

โครงงานนี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต

สาขาวิชาวิศวกรรมโทรคมนาคม

คณะวิศวกรรมศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2553

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น ทั้งนี้ห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารนี้ทุกครั้งที่มีการนำไปใช้

ผ่านการตรวจชิ้นงานแล้ว

(ลงชื่อ).....ผู้ตรวจ

ผ่านการตรวจรูปเล่มแล้ว

(ลงชื่อ).....ผู้ตรวจ

ปริญญานิพนธ์ปีการศึกษา 2553

สาขาวิชาวิศวกรรมโทรคมนาคม

คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เรื่อง การจัดการที่จอดรถ

CARS PARKING MANAGEMENT

ผู้จัดทำ

1. นายเชิดชัย คำภู รหัสนักศึกษา 49010213
2. นายอนุชา จารุสิน รหัสนักศึกษา 50011838

..... อาจารย์ที่ปรึกษา

(ผศ.สมภพ แก้วมีชัย)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กิตติกรรมประกาศ

สำหรับโครงงานนี้จะสำเร็จลุล่วงไปไม่ได้ ถ้าไม่ได้รับการสนับสนุนค่าใช้จ่ายจาก บิดา มารดา รวมถึงคำแนะนำปรึกษาจาก คุณยงยุทธ หงษ์พงษ์ และอาจารย์ที่ปรึกษารวมถึงเพื่อนๆที่ช่วยสอนความรู้ใหม่ๆเกี่ยวกับการจัดทำโครงงานครั้งนี้ และทำให้โครงงานนี้ประสบความสำเร็จลุล่วงไปด้วยดี

นายเชิดชัย คำภู

นายอนุชา จารุสิน

ผู้จัดทำ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การจัดการที่จอดรถ

CARS PARKING MANAGEMENT

โดย นายเชิดชัย คำภู รหัส 49010213
นายอนุชา จารุสิน รหัส 50011838

อาจารย์ที่ปรึกษา ผศ.สมภพ แก้วมีชัย

บทคัดย่อ

โครงงานปริญญานิพนธ์นี้เป็นโครงงานที่ใช้ไมโครคอนโทรลเลอร์ตระกูลMCS-51 นำมาประยุกต์ใช้งานเพื่อแสดงตำแหน่งที่ว่างของลานจอดรถแต่ละชั้น และจำนวนของรถที่มีอยู่ภายในอาคารทั้งหมด โดยวงจรตรวจจับจะใช้แสงอินฟราเรดตรวจสอบตำแหน่งของที่จอดรถแต่ละตำแหน่งซึ่งจะมีตัวตรวจจับ แล้วนำสัญญาณที่ได้ส่งไปยังไมโครคอนโทรลเลอร์ตระกูล MCS-51 ซึ่งจะเก็บค่าของข้อมูลต่าง ๆ เอาไว้ โดยค่าทั้งหมดที่ได้จะมีการแสดงผลไปยังจอคอมพิวเตอร์ โดยที่ใช้โปรแกรมภาษาซีชาร์ป คำนวณค่าต่าง ๆ ที่รับมาจากไมโครคอนโทรลเลอร์และแสดงผลออกมาทางจอมอนิเตอร์

ABSTRACT

This study project, uses a MCS-51 microcontroller application to show available spaces in a car park on each floor and the total number of cars in a whole building. Fifty six sets of detector circuits using infrared rays scan each car park position and send obtained signals to the MCS-51 microcontroller. This collects values of data and all of these values express their results on a computer by C# language program , which then calculates these received values from the microcontroller to express out puts.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ

		หน้า
กิตติกรรมประกาศ		I
บทคัดย่อ		II
สารบัญ		III
สารบัญรูป		VI
สารบัญตาราง		VIII
บทที่ 1	**บทนำ**	**1**
	1.1 ความเป็นมาและความสำคัญของปัญหา	1
	1.2 วัตถุประสงค์	1
	1.3 ขอบเขตของโครงงาน	2
	1.4 รายละเอียดของโครงงาน	2
บทที่ 2	**ทฤษฎีและหลักการ**	**3**
	2.1 ทฤษฎีที่เกี่ยวกับเซ็นเซอร์ (SENSOR)	3
	2.1.1 โฟโต้ไดโอดอินฟราเรดไดโอด	3
	2.1.2 โฟโต้ทรานซิสเตอร์อินฟราเรด	4
	2.1.3 ออปโต้ไอโซเลเซอร์ (OPTO ISOLATOR	4
	2.1.4 อินฟราเรดเซ็นเซอร์	5
	2.1.5 ทฤษฎีที่เกี่ยวกับแสงอินฟราเรด	6
	2.1.6 ตัวส่งแสงอินฟราเรด	6
	2.1.7 ตัวรับแสงอินฟราเรด	8
	2.2 วงจรออปแอมป์	10
	2.2.1 วงจรเปรียบเทียบแรงดัน	10
	2.2.2 วงจรแรงดันตาม	11

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ (ต่อ)

2.3 ไมโครคอนโทรลเลอร์ MCS-51	11
2.3.1 ไมโครคอมพิวเตอร์	11
2.3.2 ไมโครคอนโทรลเลอร์	13
2.3.3 เหตุผลที่ต้องเลือกไอซีไมโครคอนโทรลเลอร์ MCS-51	14
2.3.4 ความหมายของไมโครคอนโทรลเลอร์ ไมโครคอนโทรลเลอร์ (MICROCONTROLLER)	15
2.3.5 โครงสร้างของไมโครคอนโทรลเลอร์	15
2.3.6 การจัดขาต่าง ๆ ของ MCS-51 ไอซี	18
2.3.7 โครงสร้างของพอร์ตอินพุทและเอาท์พุท	21
2.3.8 โครงสร้างหน่วยความจำ	24
2.3.9 หน่วยความจำภายนอก MCS-51	25
2.3.10 การติดต่อกับหน่วยความจำโปรแกรมภายนอก	26
2.3.11 การติดต่อกับหน่วยความจำข้อมูลภายนอก	28
2.3.12 การใช้งานพอร์ตสื่อสารอนุกรม	31
2.3.13 กระบวนการรับและส่งข้อมูลอนุกรมของ MCS-51	32
2.4 การสื่อสารแบบอนุกรม	38
2.4.1 มาตรฐาน RS-232C	39
2.4.2 ลักษณะของคอนเน็กเตอร์แบบ D- TYPE	39
2.4.3 รายละเอียดของสัญญาณ	41
2.4.4 ลักษณะสัญญาณอินพุตและเอาต์พุตของพอร์ต RS-232	41
2.4.5 อัตราเร็วในการรับส่งข้อมูลแบบอนุกรม	43
2.5 ความรู้เบื้องต้นเกี่ยวกับไอสแควร์ซี	44
2.5.1 หลักการของระบบบัสไอสแควร์ซี	44
2.5.2 ภาวะที่เกิดขึ้นบนระบบบัส	44
2.5.3 การทำงานบนบัส	45
2.5.4 การขยายพอร์ตด้วยไอซี PCF8574A	46

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ (ต่อ)

	2.6 โปรแกรมภาษาซีชาร์ป	48
	2.6.1 พื้นฐานโปรแกรมภาษาซีชาร์ป	48
	2.6.2 ตัวแปร (VARIABLES)	49
บทที่ 3	**การออกแบบและการจัดทำปริญญานิพนธ์**	**51**
	3.1 โครงสร้างของระบบในส่วนต่างๆของโครงงาน	54
	3.1.1 วงจรตรวจเซ็นเซอร์	54
	3.1.2 ไมโครคอนโทรลเลอร์ MCS-51	57
	3.2 การจัดขาต่าง ๆ ของ MCS-51	57
	3.3 ลานจอดรถจำลอง	58
	3.3.1 ในกรณีสีแดง	59
	3.3.2 ในกรณีสีเขียว	60
	3.3.3 ในกรณีสีเหลือง	61
บทที่ 4	**ผลการดำเนินงาน**	**63**
	4.1 ผลการทดลองวงจรตรวจจับ	63
	4.2 ทดสอบการทำงานของวงจร	64
	4.3 ทดสอบการทำงานของระบบ	68
บทที่ 5	**สรุปผลการดำเนินงาน**	**70**
	5.1 สรุปผล	70
	5.2 ปัญหาและแนวทางการแก้ไข	70
	5.3 แนวทางการพัฒนา	71
บรรณานุกรม		**72**
ภาคผนวก		**73**

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูป

รูปที่		หน้า
2.1	หลักการของตัวส่งและรับแสง	6
2.2	ตัวอย่างแอลอีดีภาคแสดงต่าง ๆ	7
2.3	ลักษณะทั่วไปของแอลดีอาร์และสัญลักษณ์	9
2.4	การทำงานและลักษณะสมบัติของโฟโต้ไดโอ	10
2.5	วงจรเปรียบเทียบแรงดัน	10
2.6	วงจรแรงดันตาม	11
2.7	โครงสร้างภายในของ MCS-51	17
2.8	ขาต่างๆของ AT89C51AC3	19
2.9	ขาของ MCS-51 ที่ใช้ต่อกับ XTAL	21
2.10	โครงสร้างพอร์ต1 ,3 และ 4 ของ MCS-51	22
2.11	โครงสร้างพอร์ต 0 ของ MCS-51	22
2.12	โครงสร้างพอร์ต 2 ของ MCS-51	23
2.13	การต่อพอร์ทเข้ากับระบบบัสภายในของ MCS-51	23
2.14	การจัดหน่วยความจำของ MCS-51	24
2.15	ตำแหน่งของหน่วยความจำแบบไบท์และแบบบิต	25
2.16	ไดอะแกรมกลุ่มสัญญาณที่ใช้อ่านข้อมูลวงจรของการสร้างสัญญาณ	26
2.17	ไดอะแกรมการอ่านข้อมูลจากหน่วยความจำโปรแกรมภายนอก	27
2.18	การต่อ MCS - 51 กับหน่วยความจำโปรแกรมภายนอก	28
2.19	ไดอะแกรมเวลาการอ่านและเขียนข้อมูลกับหน่วยความจำภายนอก	29
2.20	การต่อหน่วยความจำโปรแกรมกับMCS-51	30
2.21	การรีเซต MCS-51	30
2.22	ผังการทำงานโหมด 0	34
2.23	ผังการทำงานโหมด 1	35
2.24	ผังการทำงานโหมด 2	36

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูป (ต่อ)

		หน้า
2.25	ผังการทำงานโหมด 3	37
2.26	แผนผังคอนเน็กเตอร์ของ RS-232	40
2.27	ไดอะแกรมโครงสร้างทางฮาร์ดแวร์ของพอร์ตอนุกรม	42
2.28	ไอซี PCF8574A	46
3.1ก	โฟลว์ชาร์ตขั้นตอนการดำเนินงานของโครงการ	51
3.1ข	โฟลว์ชาร์ตขั้นตอนการดำเนินงานของโครงงาน (ต่อ)	52
3.2	โฟลว์ชาร์ตขั้นตอนการดำเนินงานไมโครคอนโทรลเลอร์ MCS-51	53
3.3	บล็อกไดอะแกรมของแบบจำลองระบบบอกตำแหน่งพื้นที่ว่างในอาคาร	54
3.4	วงจรภาคส่งแสงอินฟราเรด	54
3.5	วงจรภาครับของเซนเซอร์อินฟราเรด	55
3.6	ลายวงจรด้านล่างวงจรตรวจจับรวม 56 ตัว	55
3.7	แผงวงจรตรวจจับโดยใช้แสงอินฟราเรดทั้งหมด	56
3.8	แผงวงจรตรวจจับโดยใช้แสงอินฟราเรดที่ใช้งานจริง	56
3.9	ขาต่างๆของ AT89C51AC3	57
3.10	บอร์ดไมโครคอนโทรลเลอร์ MCS-51	58
3.11	ลานจอดรถจำลอง	58
3.12	โฟว์ชาร์ตกรณีไฟสีแดง	59
3.13	โฟว์ชาร์ตกรณีไฟสีเขียว	60
3.14	โฟว์ชาร์ตกรณีไฟสีเหลือง	61
4.1	วงจรตรวจจับสัญญาณ	63
4.2	การทดสอบวงจรตรวจจับ	67
4.3	ผลการทดสอบหน้าจอมอนิเตอร์กรณีมีที่ว่าง	68
4.4	ผลการทดสอบหน้าจอมอนิเตอร์กรณีมีการจอง	69

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญตาราง

หน้า

ตารางที่

2.1 ไมโครคอนโทรลเลอร์ตระกูล MCS-51 เบอร์ต่างๆ 17

2.2 บิตและหน้าที่ต่างๆ ของพอร์ต 3 20

2.3 โหมดการทำงานของพอร์ตสื่อสารอนุกรม 31

2.4 ผังคอนเน็กเตอร์ของ RS – 232 40

2.5 ข้อมูลในแอดเดรส 0000:041H ที่ใช้ในพอร์ตอนุกรม 43

4.1 การทดสอบการวัดค่าของวงจรเซ็นเซอร์ 64

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 1
บทนำ

1.1 กล่าวนำ

ในปัจจุบันนี้รถยนต์กลายเป็นปัจจัยหนึ่งที่มีความสำคัญต่อการดำรงชีวิตของคนเรา เนื่องจากการคมนาคมทำให้มีความสะดวกสบายในการเดินทางมากขึ้นประชาชนส่วนใหญ่นิยมใช้รถยนต์เป็นพาหนะในการเดินทางทำให้ตามอาคารสถานที่ต่างๆ ที่จำเป็นจะต้องไปติดต่อหรือใช้บริการ เช่น สถานที่ราชการ ห้างสรรพสินค้า หรือ สถานที่ประกอบธุรกิจต่างๆ โดยที่จะต้องใช้ยานพาหนะในการเดินทางและทางสถานที่ต่างๆ นั้นก็จะจัดเตรียมสถานที่เพื่ออำนวยความสะดวกในการจอดรถยนต์ไว้ให้ แต่ว่าในบางสถานที่นั้นมีพื้นที่จำกัดในการนำรถยนต์เข้าไปจอดเป็นเหตุทำให้ต้องเสียเวลาในการหาที่ที่ใช้ในการจอดรถยนต์ อาจเป็นเหตุให้เกิดความล่าช้าในการติดต่อหรือใช้บริการในสถานที่นั้นๆจากปัญหาดังกล่าวข้างต้นนั้น ทำให้ทางผู้จัดทำได้คิดค้นหาวิธีทางในการแก้ปัญหา โดยการจัดทำโครงงานแบบจำลองระบบบอกตำแหน่งพื้นที่ว่างในอาคารจอดรถอัตโนมัติ (Cars Parking Management) ซึ่งจะสามารถช่วยลดปัญหาการเสีย เวลาในการค้นหาตำแหน่งสำหรับจอดรถยนต์ ในบริเวณที่จอดรถของอาคารสถานที่นั้นๆได้ในระดับหนึ่ง ในโครงงานนี้ ยังสามารถให้ผู้ใช้บริการที่มาใช้บริการเป็นประจำจองพื้นที่จอดรถก่อนนำรถมาจอดได้ จึงเป็นการลดความล่าช้าสำหรับผู้ที่มาติดต่อหรือใช้บริการในสถานที่นั้นๆด้วย

1.2 วัตถุประสงค์ของโครงงาน

1.2.1 เพื่อศึกษาหลักการและการทำงานของอุปกรณ์ตรวจจับสัญญาณทางแสง

1.2.2 เพื่อศึกษาการเขียนโปรแกรมภาษาซีชาร์ป

1.2.3 เพื่อศึกษาการเชื่อมต่อคอมพิวเตอร์กับอุปกรณ์ภายนอกโดยใช้โปรแกรมภาษาซี

1.2.4 เพื่อศึกษาหลักการในการรับ-ส่งข้อมูลแบบดิจิตอล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1.2.5 เพื่อศึกษาการทำงานและการประยุกต์ใช้งานของไมโครคอนโทรลเลอร์ตระกูล MCS AT89C51

1.2.6 เพื่อสร้างแบบจำลองระบบบอกตำแหน่งพื้นที่ว่างในอาคารจอดรถอัตโนมัติ

1.3 ขอบเขตของโครงงาน

1.3.1 แบบจำลองของอาคารจอดรถยนต์สามารถจอดรถยนต์ได้ทั้งหมด 56 คัน โดยที่ทางเข้า-ออก ของอาคารให้ผ่านเข้า-ออกได้เฉพาะรถยนต์เท่านั้น

1.3.2 ระบบสามารถแสดงจำนวนที่ว่าง-ไม่ว่างทั้งหมดสำหรับที่จอดรถยนต์ภายในอาคารจอดรถยนต์โดยผ่านทางจอมอนิเตอร์ได้

1.3.3 ระบบสามารถจองตำแหน่งที่จอดรถได้โดยการใช้รหัส

1.3.4 ระบบจะมีการแสดงแผนผังของอาคารจอดรถไปที่จอมอนิเตอร์เพื่อแสดงตำแหน่งที่ว่างไม่ว่าง ซึ่งจะมีการติดตั้งจอภาพไว้ใกล้กับชุดโมเดลของอาคารจอดรถ

1.4 ประโยชน์ที่คาดว่าจะได้รับ

1.4.1 เพื่อเป็นประโยชน์ในการอำนวยความสะดวกแก่ผู้ใช้บริการของอาคารสถานที่นั้นๆจัดทำเพื่อผู้ที่ใช้รถยนต์เป็นพาหนะในการเดินทางเป็นหลัก

1.4.2 หากว่าผู้ที่เข้ามาหาที่จอดรถสามารถทราบว่ามีที่จอดรถเหลืออยู่หรือที่จอดรถเต็มแล้ว โดยดูได้จากจอมอนิเตอร์ที่ทางเข้า ถ้าหากว่าที่จอดเต็มแล้วก็ไม่จำเป็นต้องเข้าไปจอดโดยสามารถที่จะไปจอดที่จอดอื่นๆ ได้ ซึ่งจะทำให้ลดการเสียเวลาโดยเปล่าประโยชน์จากการนำรถเข้าไปจอดในที่ที่มีรถจอดเต็มอยู่แล้ว

1.4.3 แบบจำลองของอาคารจอดรถยนต์ที่จัดทำขึ้นนี้ อาจนำไปใช้เป็นแนวทางในการพัฒนาในการขยายพื้นที่ และนำไปใช้กับอาคารจอดรถที่ใช้งานจริงต่อไปในอนาคตได้

1.4.4 เป็นการนำความรู้ที่มีอยู่เดิม จากการศึกษาในห้องเรียนมาประยุกต์ใช้ให้ได้สิ่งประดิษฐ์ใหม่ ๆ ที่ทำให้เกิดประโยชน์ต่อไป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 2

ทฤษฎีที่สำคัญและหลักการ

2.1 ทฤษฎีที่เกี่ยวกับเซ็นเซอร์ (Sensor)

เซ็นเซอร์เป็นอุปกรณ์ชนิดหนึ่งซึ่งออกแบบมาให้สามารถใช้งานได้หลายรูปแบบ และยังสะดวกในการใช้งานอีกด้วย โดยสามารถประกอบเป็นวงจรหรือใช้งานร่วมกันกับอุปกรณ์ตัวอื่นได้ เซ็นเซอร์นั้นเปรียบเสมือนกับดวงตาซึ่งเป็นประสาทสัมผัสของคนเราที่มีความสำคัญยิ่ง ในโครงการนี้เราจะมองว่าการที่เราใช้เซ็นเซอร์เป็นตัวที่ตรวจจับสภาวะใดๆเช่น อุณหภูมิ สี แสง หรือวัตถุต่างๆ โดยอาศัยหลักการที่แตกต่างกันไปตามแต่ชนิดเช่น เซ็นเซอร์สีขาวดำ โดยอาศัยหลักการสะท้อนแสงของสีขาวและดำ ทางฟิสิกส์แล้วจะเห็นว่าสีขาวมีอัตราการสะท้อนแสงมากกว่าสีดำ เราจึงสามารถนำแสงสะท้อนมาเปรียบเทียบได้ โดยใช้ตัวเซ็นเซอร์คือ อุปกรณ์จำพวกโฟโต้ เช่น โฟโต้ไดโอด(Photo diode) โฟโต้ทรานซิสเตอร์ (Photo transistor) เป็นต้น ซึ่งมีความไวต่อแสงมาก โดยส่วนใหญ่ผลลัพธ์จะแสดงในรูปของความต้านทานที่เปลี่ยนไปตามสภาวะของตัวเซ็นเซอร์นั้นๆ เราสามารถแบ่งตัวเซ็นเซอร์ออกตามชนิดดังต่อไปนี้ ซึ่งในโครงงานนี้เราจะเลือกใช้เซ็นเซอร์ ชนิดอินฟราเรดเซ็นเซอร์ในการตรวจจับ

2.1.1 โฟโต้ไดโอดอินฟราเรดไดโอด

ทำงานด้วยแสงหรืออุปกรณ์ที่เปลี่ยนแปลงค่าความต้านทานตามแสง โดยจะเปลี่ยนแปลงความต้านทานตามความเข้มของแสงที่มาตกกระทบถ้าแสงมาตกกระทบน้อยจะมีความต้านทานสูง ถ้าแสงตกกระทบมากค่าความต้านทานจะต่ำ โฟโต้ไดโอดจะประกอบด้วยสารกึ่งตัวนำ P และ N ต่อชนกัน มีขาต่อออกมาใช้งาน 2 ขาคือขาแอโนด (A) และขาแคโถด (K) ส่วนวัสดุที่นำมาผลิตมีอยู่หลายชนิด เช่น ซิลิคอน เจอมันเนียม เซเลเนียม เป็นต้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.1.2 โฟโต้ทรานซิสเตอร์อินฟราเรด

เป็นอุปกรณ์สารกึ่งตัวนำทางแสงอีกชนิดหนึ่งที่ถูกสร้างและพัฒนามาใช้งานแทนโฟโต้ไดโอดเพราะมีข้อดีกว่าโฟโต้ไดโอดตรงที่ในการใช้งาสามารถทำการขยายสัญญาณที่ส่งออกได้แสงเพียงเล็กน้อยก็สามารถทำงานได้ดีการทำงานของโฟโตทรานซิสเตอร์ เมื่อจ่ายไบอัสตรงที่ขา C และขา E ของโฟโตทรานซิสเตอร์ และไม่มีแสงส่องมากระทบรอยต่อที่โฟโตทรานซิสเตอร์ตรงรอยต่อสาร PN ระหว่างขา C และขา B แล้ว โฟโตทรานซิสเตอร์จะไม่นำกระแส เมื่อมีแสงส่องมาตกกระทบรอยต่อสาร PN ระหว่างขา C และขา B จะทำให้เกิดกระแส Ib ไหลส่งผลให้โฟโตทรานซิสเตอร์ทำงาน เราสามารถหากระแสคอลเลคเตอร์ได้จากสมการดังนี้

$$I_c = h_{FE} I_\lambda \qquad (2.1)$$

เมื่อ I_c = กระแสคอลเลคเตอร์

I_λ = กระแสเบสที่เกิดขึ้นจากแสงอินฟราเรดมาตกกระทบ

h_{FE} = อัตราขยายกระของโฟโตทรานซิสเตอร์

ค่ากระแส I_c จะไหลเปลี่ยนแปลงตามการรับแสงอินฟราเรดที่ขาเบส

2.1.3 ออปโต้ไอโซเลเซอร์ (OPTO ISOLATOR)

เป็นอุปกรณ์เชื่อมต่อทางแสง ด้วยอุปกรณ์อิเล็กทรอนิกส์ล้วนๆการส่งผ่านแสงและการรับแสง จะอยู่ในอุปกรณ์เดียวกันทั้งหมดโดยใช้หลักการเปลี่ยนสัญญาณทางไฟฟ้าไปเป็นสัญญาณแสง และเปลี่ยนกลับจากสัญญาณแสงเป็นสัญญาณทางไฟฟ้าเหมือนเดิม จุดประสงค์จุดประสงค์ในการเชื่อมต่อวิธีนี้ก็เพื่อป้องกันการรบกวนของสัญญาณซึ่งกันและกันและต้องการให้เกิดการแยกกันของวงจรโดยเด็ดขาด แรงดันระหว่างอุปกรณ์ทั้งสองด้าน คืออุปกรณ์ทางด้านปล่อยแสงและอุปกรณ์ทางด้านรับแสงจะใช้แรงดันมากหรือน้อยขึ้นอยู่กับระยะห่างระหว่างตัวปล่อยแสงและตัวรับแสง ระยะห่างของอุปกรณ์ทั้งสองยิ่งมาก แรงดันที่ป้อนให้อุปกรณ์ทั้งสองจะต้องใช้มากขึ้น แต่มีขีดตรงค่าทนแรงดันของอุปกรณ์ ส่วนมากจะสร้างมาในรูปของตัว IC อุปกรณ์ตัวกำเนิดแสงจะเป็นพวก ไลต์อิมิตติ้งไดโอด (Light-Emitting Diode) ส่วนมากจะเป็นชนิดอินฟราเรด อุปกรณ์รับแสงพวกโฟโต้ดีเทคเตอร์ (Photo detector) จะนำมาใช้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

งานหลายชนิดด้วยกันคือ โฟโตทรานซิสเตอร์ทั้งแบบธรรมดาและแบบดาร์ลิงตัน หรืออาจเป็น โฟโต้เอสซีอาร์ (Photo Silicon Control Rectifier) เป็นต้น

2.1.4 อินฟราเรดเซ็นเซอร์

แสงอินฟราเรด คือแสงที่มีความยาวคลื่นต่ำกว่าแสงสีแดงลงไป อินฟราเรดเป็นแสงที่มีความยาวคลื่นอยู่ในช่วงประมาณ 780-3000 นาโนเมตร ดังนั้นจึงไม่สามารถมองเห็นได้ด้วยสายตา ของมนุษย์ ซึ่งคุณสมบัตินี้เอง จึงทำให้ เซ็นเซอร์ชนิดที่ใช้แสงอินฟราเรด เป็นที่นิยมนำมาใช้กันมาก โดยจะอาศัยหลักการของการสะท้อนของแสง กล่าวคือ ใช้อุปกรณ์ส่งแสง เป็นแหล่งกำเนิด ปล่อยแสงออกไป และเมื่อแสงกระทบกับวัตถุด้านหน้า มันก็จะสะท้อนแสงกลับมาเข้าที่ตัวรับแสง ส่วนอัตราของการสะท้อนกลับนั้น ขึ้นอยู่กับสี และสภาพความมัน ของวัตถุที่สะท้อน เช่น สีดำ จะมีอัตราการสะท้อนกลับ น้อยกว่าสีขาว , หรือสภาพพื้นผิวที่มีความราบเรียบเป็นมันวาว จะสามารถสะท้อนแสงได้ดีกว่า พื้นผิวที่มีลักษณะด้าน และขรุขระ เป็นต้น แสงอินฟราเรดเป็นแสงที่ไม่สามารถมองเห็นได้ด้วยตาเปล่า จึงเป็นที่นิยมที่จะนำมาใช้ในการสื่อสารหรือตรวจจับสิ่งของต่างๆ เพราะปัญหาการรบกวนของสัญญาณของแสงอื่นๆมีน้อยอีกทั้งการสร้างวงจรที่ใช้ในระบบอินฟราเรดก็ง่ายไม่มีความซับซ้อนมากนัก และความน่าเชื่อถือของสัญญาณที่ส่งก็มีความเชื่อถือสูง โดยในระบบอินฟราเรดจะต้องมีภาคส่ง และภาครับ ซึ่งการสร้างภาคส่งนั้นก็เพียงให้มีการส่งแสงออกมาในช่วงความถี่ที่สูงกว่าความถี่ทั่วๆไปของแสงธรรมดา คือต้องมากกว่า 20 kHz โดยจะใช้อินฟราเรดอิมิตติ้งไดโอด(Infrared Emitting Diode) เป็นตัวขับแสงอินฟราเรด ส่วนการสร้างภาครับนั้นเราก็จะใช้ โฟโต้ทรานซิสเตอร์เป็นตัวรับแสงโดยที่ทั้งภาครับและภาคส่งจะต้องมีระยะ ความสูงและทิศทางเท่ากัน เพราะถ้าไม่เท่ากันจะทำให้การตรวจจับสัญญาณได้ไม่ตรง แต่สำหรับโครงงานนี้จะใช้คุณสมบัติของตัวโฟโต้ทรานซิสเตอร์ ที่เมื่อมีแสงมาตกกระทบมันแล้วจะทำให้ปริมาณของกระแสที่วิ่งผ่านโฟโต้ทรานซิสเตอร์มีค่าเพิ่มขึ้นตามปริมาณความเข้มของแสงทำให้ความต่างศักดิ์ที่ตกคร่อมโฟโต้ทรานซิสเตอร์มีค่ามากตามไปด้วย และใช้ในการตรวจจับหาแหล่งที่มาของแสงเพื่อที่จะใช้บอกว่ามีการเคลื่อนไหว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.1.5 ทฤษฎีที่เกี่ยวกับแสงอินฟราเรด

รังสีอินฟราเรดมีลักษณะเช่นเดียวกับแสงธรรมชาติ รวมถึงคลื่นวิทยุ และรังสีเอกซ์ ซึ่งมีการแผ่รังสีออกมาในรูปคลื่นความถี่คุณสมบัติต่างๆ ของแสงอินฟราเรด มีดังนี้ คือ

1. เป็นแสงที่ไม่สามารถมองเห็นได้ด้วยตาเปล่า
2. เดินทางในอากาศได้ด้วยความเร็ว $3x10^8$ เมตรต่อวินาที
3. เกิดการสะท้อน และหักเหของแสงได้
4. มีความยาวคลื่นที่ใช้ประมาณ 780-1200 นาโนเมตร

รูปที่ 2.1 หลักการของตัวส่งและรับแสง

ในการส่งแสงอินฟราเรดจากตัวส่งแสงออกมายังตัวรับแสงจะต้องให้ตัวรับแสงวางอยู่ตรงกับตัวส่งแสงเพราะถ้าหากวางไม่ตรงกันตัวรับแสงจะไม่สามารถรับแสงได้เนื่องจากแสงเดินทางเป็นเส้นตรง ซึ่งจะแสดงหลักการของตัวส่งและรับแสง ดังรูปที่ 2.1

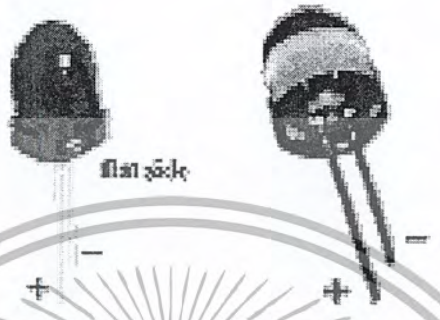
2.1.6 ตัวส่งแสงอินฟราเรด

แอลอีดี (LED)

แอลอีดีที่ให้แสงสีแดงนั้นเป็นไดโอดที่ทำมาจากสารกึ่งตัวนำที่เรียกว่า แกลเลียมอะเซไนด์ (GaAs) ซึ่งมีประสิทธิภาพการใช้งานและความเชื่อถือสูงจะปล่อยแสงออกมาอยู่ในย่านความยาวคลื่นประมาณ 940 นาโนเมตร คุณสมบัติทั่วไปของแอลอีดีก็เหมือนกับของไดโอดโดยในขณะไบแอสตรง มันจะให้กระแสไหลผ่านได้และจะเกิดแรงดันตกคร่อมตัวมัน มีค่าประมาณ 0.6-1 โวลต์ และในขณะไบแอสกลับแอลอีดีจะมีค่าแรงดันพังที่ต่ำกว่าไดโอดธรรมดาดังภาพที่ 2.2 นอกจากแอลอีดีที่เปล่งแสงสีแดงแล้ว เรายังพบแอลอีดีที่ให้แสงสีอื่นอีกมากมาย เช่น แสงสีเขียว สีเหลือง สีแสด เป็นต้น การที่แอลอีดีให้แสงสีแตกต่างกันนี้ขึ้นอยู่กับชนิดของเนื้อสารกึ่งตัวนำที่ใช้หรือขึ้นอยู่กับแถบพลังงานช่องว่างนั่นเอง เพราะความยาวคลื่นของแอลอีดีจะขึ้นอยู่กับ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

แถบพลังงานช่องว่างของสารโดยตรง ประโยชน์ของแอลอีดีในปัจจุบันมีมากมาย โดยสิ่งที่เห็นได้ชัดก็คือการใช้แอลอีดีเป็นอุปกรณ์จำพวกภาคแสดงผลของเครื่องคิดเลข เครื่องมือวัด ฯลฯ



รูปที่ 2.2 ตัวอย่างแอลอีดีภาคแสดงต่าง ๆ

แสงที่เปล่งออกมาจากตัวไดโอดเปล่งแสง (LED) แบ่งออกได้เป็น 2 ชนิด คือ แสงที่ตาคนมองเห็น และแสงที่ตามองไม่เห็น

1) แสงที่ตาคนมองเห็น ไดโอดเปล่งแสง (LED) ที่เปล่งแสงชนิดที่ตาคนมองเห็น 3 สี คือ สีแดง สีเขียว และสีเหลือง บางครั้งอาจออกสีส้ม ซึ่งก็คือสีเหลืองเข้ม ไดโอดเปล่งแสงชนิดนี้เรียกว่า ไลท์อีมิตติ้งไดโอด (Light Emitting Diode) หรือ LED เหมือนชื่อที่กล่าวมา

2) แสงที่คนมองไม่เห็น ไดโอดเปล่งแสง (LED) ที่เปล่งแสงที่ตาคนมองไม่เห็น เพราะเป็นแสงที่อยู่ในย่านอินฟราเรด (Infrared Light) ซึ่งขณะที่เปล่งแสงออกมาตาคนจะมองไม่เห็น ไดโอดเปล่งแสงชนิดนี้เรียกว่า อินฟราเรดอิมิตติ้งไดโอด (Infrared Emitting Diode) หรือ IRED แต่สามารถเรียกชื่อแบบรวมๆกันว่า ไดโอดเปล่งแสงหรือ LED แสงที่ถูกเปล่งออกมาจากไดโอดเปล่งแสง (LED) ที่มีสีต่างกันนั้นขึ้นอยู่กับเนื้อสารกึ่งตัวนำที่นำมาใช้ผลิต เมื่อใช้เนื้อสารกึ่งตัวนำต่างกัน เนื้อที่ใช้และสีที่ได้มีความแตกต่างกันดังนี้

1.ใช้สารแกลเลียมอาร์ซีไนด์ (Gallium Arsenide)ใช้ตัวย่อ GaAs ไดโอดเปล่งแสงอินฟราเรดออกมา

2. ใช้สารแกลเลียมอาร์ซีไนด์ ฟอสไฟต์ (Gallium Arsenide Phosphide) ใช้ตัวย่อ GaAsP ไดโอดเปล่งแสงสีแดง หรือสีเหลืองออกมา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3. ใช้สารแกลเลียมฟอสไฟต์ (Gallium Phosphide) ใช้ตัวย่อ GaP ไดโอดเปล่งแสง (LED) จะเปล่งแสงสีแดงหรือสีเขียวออกมา

ลักษณะไดโอดเปล่งแสงแบบอินฟราเรด

แสงที่เปล่งออกมาจากตัวไดโอดเปล่งแสง (LED) เกิดขึ้นตรงบริเวณรอยต่อของสารชนิด P และสารชนิด N เกิดการกระจัดกระจายออกรอบตัว ทำให้ความเข้มของแสงลดลงการใช้ไดโอดเปล่งแสง (LED) จำเป็นต้องมีความเข้มของแสงรวมไปในทิศทางใดทิศทางหนึ่งเฉพาะยิ่งถ้าเป็นไดโอดเปล่งแสง (LED) ที่เปล่งแสงอินฟราเรดออกมา ยิ่งมีความจำเป็นเพราะนิยมนำไปใช้กับระบบควบคุมระยะไกล (Remote Control) ต้องให้แสงเคลื่อนที่ไปในทิศทางเดียวดังนั้นในส่วนของตัวถังไดโอดเปล่งแสง (LED) จึงต้องมีส่วนสะท้อนแสง และเลนส์รวมแสงเพิ่มเข้าไป

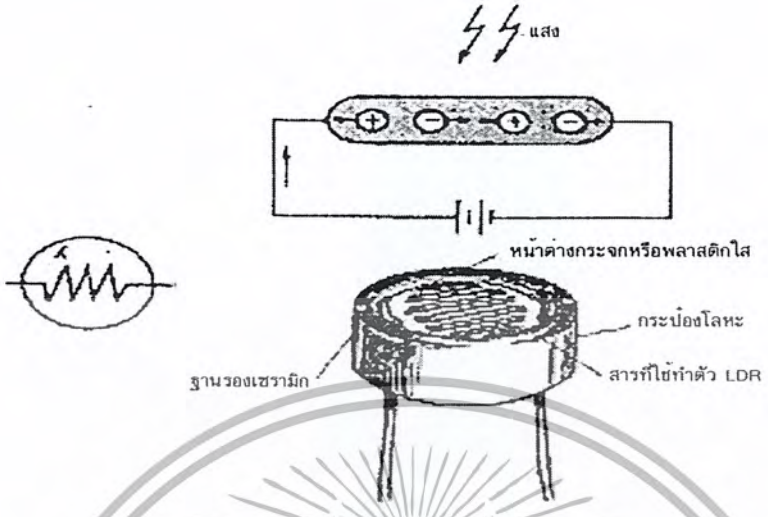
2.1.7 ตัวรับแสงอินฟราเรด

อุปกรณ์ตัวรับแสง หมายถึง อุปกรณ์ที่สามารถเปลี่ยนพลังงานแสงให้แปรค่ากับค่าของพลังงานทางไฟฟ้าได้ โดยตัวอุปกรณ์จะต้องประกอบด้วยแสงกึ่งตัวนำซึ่งจะนำมาต่อเชื่อมให้เกิดเป็นรอบต่อหรือเป็นเนื้อสารกึ่งตัวนำอย่างเดียวก็ได้

แอลดีอาร์

ตัวต้านทานไวแสงหรือแอลดีอาร์ แอลดีอาร์ (LDR) เป็นคำย่อของคำว่า light dependent resistor หรือค่าตัวต้านทานที่แปรค่าได้กับแสงนั่นเอง โดยเมื่อป้อนพลังงานอย่างพอเพียงให้กับ วาเลนซ์อิเลคตรอน ของสารกึ่งตัวนำจะทำให้เกิดการแตกตัวของโฮลและอิเลคตรอนเกิดขึ้นมากมาย พลังงานที่จะทำให้โฮล และอิเลคตรอนแตกตัวนี้จะต้องเป็นพลังงานที่มาจากภายนอก เช่น แสง ความร้อน ปริมาณของพลังงานที่ต้องการทำให้เกิดการแตกตัวของอิเลคตรอน-โฮล จะขึ้นอยู่กับชนิดของสารนั้น ๆ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



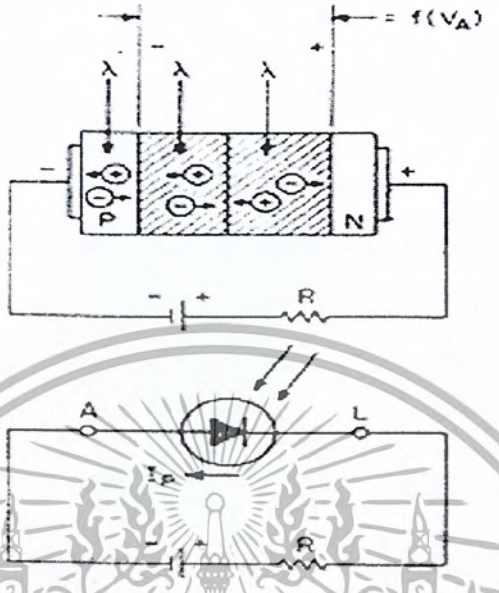
รูปที่ 2.3 ลักษณะทั่วไปของแอลดีอาร์และสัญลักษณ์

ลักษณะการทำงานของแอลดีอาร์จะเห็นได้ว่าเมื่อแสงมีความเข้มมาก ๆ มาตกกระทบ ก็จะทำให้ตัวมันนำกระแสไฟฟ้าได้ดีขึ้นเป็นผลทำให้เกิดค่าความต้านทานในตัวมันลดลง ข้อดีในการใช้งานของแอลดีอาร์ คือ มีความไวต่อแสงสูง ราคาถูก และมีขนาดเล็ก มีการเปลี่ยนแปลงค่าได้กว้าง แต่ก็ยังมีข้อเสียเมื่อเทียบกับอุปกรณ์รับแสงอื่น ๆ คือมีผลตอบสนองต่อความถี่ของแสงได้แคบดังรูปที่ 2.3

โฟโต้ไดโอด

โฟโต้ไดโอดเป็นอุปกรณ์ไวแสงชนิดหนึ่งซึ่งประกอบด้วยรอยต่อ P-N การใช้งานตัวโฟโต้ไดโอดจะให้กระแสไหลผ่านตัวมันที่ขึ้นอยู่กับแสงเมื่อโฟโต้ไดโอดชนิดซิลิกอน ถูกไบแอสกลับด้วยแรงดันค่าหนึ่งและมีแสงส่องไปที่บริเวณรอยต่อ ถ้าแสงที่ส่องมีความยาวคลื่นที่พอเหมาะจะมีกระแสไหลในวงจรโดยเป็นสัดส่วนกับความเข้มข้นของแสงที่ส่องผ่านบนอุปกรณ์นั้น ลักษณะทั่วไปขณะไบแอสตรงจะยังคงเหมือนกับไดโอดธรรมดาคือให้กระแสผ่านตัวมันเองได้ เมื่อแสงตกกระทบบริเวณรอยต่อจะทำให้เกิดการแตกตัวของโฮลและอิเลคตรอนอิสระ โฮลและอิเลคตรอนอิสระจะถูกแรงดันไบแอสกลับเหนี่ยวนำให้เกิดการไหลข้ามรอยต่อเป็นกระแสเนื่องจากแสงได้ โดยปกติค่ากระแสเนื่องจากแสงนี้มีค่าค่อนข้างต่ำมาก คืออยู่ในช่วง 1-10 uA เท่านั้น เนื่องจากไดโอดนี้ให้ค่าการเปลี่ยนแปลงของกระแสต่อแสงต่ำ ดังนั้นการใช้งานของโฟโต้ไดโอดจึงต้องมีตัวขยายกระแสเพิ่มเติม เช่น โฟโต้ทรานซิสเตอร์ ดังรูปที่ 2.4

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

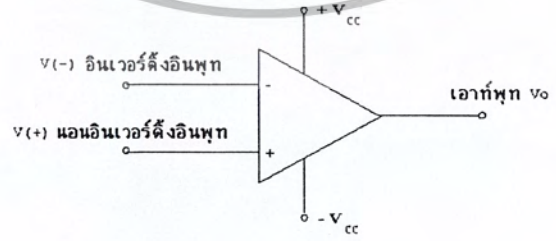


รูปที่ 2.4 การทำงานและลักษณะสมบัติของโฟโต้ไดโอ

2.2 วงจรออปแอมป์

2.2.1 วงจรเปรียบเทียบแรงดัน

วงจรเปรียบเทียบแรงดันเป็นวงจรที่ใช้ในการเปรียบเทียบแรงดันระหว่างขาอินเวอร์ติ้ง (Inverting) และขานอนอินเวอร์ติ้ง (Non-Inverting) ของออปแอมป์



รูปที่ 2.5 วงจรเปรียบเทียบแรงดัน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

$$Vo = +Vsat \quad ; V(+) > V(-) \tag{2.2}$$

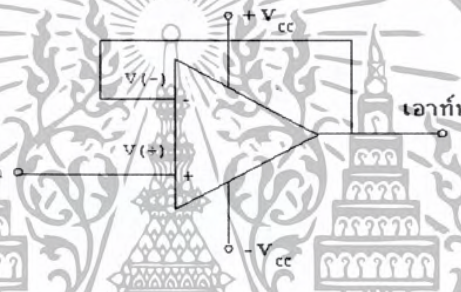
$$= -Vsat \quad ; V(+) < V(-) \tag{2.3}$$

เมื่อ Vo เป็นแรงดันเอาต์พุตของออปแอมป์

$V(+)$ เป็นแรงดันขาเข้าไม่กลับเฟส (+) ของออปแอมป์

$V(-)$ เป็นแรงดันขาเข้ากลับเฟส (-) ของออปแอมป์

2.2.2 วงจรแรงดันตาม



รูปที่ 2.6 วงจรแรงดันตาม

$$Vo = Vin \tag{2.4}$$

เมื่อ Vo เป็นแรงดันเอาต์พุตของออปแอมป์

Vin เป็นแรงดันขาเข้า

2.3 ไมโครคอนโทรลเลอร์ MCS-51

2.3.1ไมโครคอมพิวเตอร์

เมื่อเราเริ่มต้นเรียนรู้เรื่องคอมพิวเตอร์พิจารณาได้อย่างไรว่าชิ้นงานที่เห็นอยู่นั้นเป็นไมโครคอมพิวเตอร์ได้หรือไม่ ให้เราพิจารณาได้จาก องค์ประกอบ ของชิ้นงาน ซึ่งมีส่วนประกอบที่สำคัญดังนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1. หน่วยประมวลผลกลาง (CPU: Central Processing Unit) มีคุณสมบัติหลัก คือการประมวลผลข้อมูลการคำนวณทางคณิตศาสตร์และลอจิก เรารู้จักกันดีในชื่อของไอซีไมโครโพรเซสเซอร์ เช่น 8080, 80286, 80486 Pentium ฯลฯ ซึ่งเป็นของบริษัท Intel หรืออาจเป็น CPU รุ่นเก่าที่มีขนาด 8 บิต เช่นเบอร์ Z80 ที่เป็น CPUของบริษัท ZILOG เป็นต้น

2. หน่วยความจำ (Memory)

2.1 หน่วยความจำรอม (ROM: Read Only Memory) เป็นหน่วยความจำแบบถาวรที่มีการบันทึกข้อมูลไว้ ล่วงหน้าก่อนแล้วไม่สามารถเปลี่ยนแปลงข้อมูลหรือคำสั่งใดๆ ได้อีก ตัวอย่างเช่น ไอซีที่เป็นไบออส (BIOS) ของคอมพิวเตอร์ ในขณะที่เริ่มเปิดเครื่องไมโครคอมพิวเตอร์ครั้งแรก สังเกตได้ว่าจะมีการแสดงชื่อผู้ผลิตของบริษัทหรือ คุณสมบัติของเครื่องบนหน้าจอมอนิเตอร์ ถ้าหากเป็นเครื่องเล่นวีดีโอ จะเป็นตัวอักษรที่ทำหน้าที่แสดงผลเพื่อบอกให้ตั้งค่าข้อมูลต่างๆ ซึ่งไม่ว่าจะปิดแล้วเปิดกี่ครั้งตัวอักษรเดิมนั้นจะยังคงอยู่

2.2 หน่วยความจำแรม (RAM: Random Access Memory) เป็นหน่วยความจำที่ใช้เก็บข้อมูลไว้เพียงชั่วคราวอาจเป็นข้อมูลที่ CPU ต้องการประมวลผลในขณะนั้น และเมื่อ CPU ประมวลผลเรียบร้อยแล้วอาจลบหรือเปลี่ยนข้อมูลได้ บางครั้งเมื่อหยุดการจ่ายไฟให้กับวงจรจะทำให้ข้อมูลสูญหายไปได้ในทันที ตัวอย่าง เช่น ขณะที่เรากำลังพิมพ์งานแต่ยังไม่ได้บันทึกข้อมูลไว้ในส่วนใด ข้อมูลนี้จะถูกเก็บไว้ที่หน่วยความจำแรมก่อนหากเกิดเหตุการณ์ไฟฟ้าดับ จะทำให้ข้อมูลสูญหายไปได้ หรือการเก็บค่าของเวลาและอุณหภูมิของเครื่องไมโครเวฟ ที่สามารถเปลี่ยนแปลงค่าได้ตลอด หน่วยความจำแบบแรมนี้จะแตกต่างกับหน่วยความจำแบบรอม โดยหน่วยความจำแบบรอมจะไม่สามารถแก้ไข ข้อมูลได้ในขณะนั้น ในขณะที่หน่วยความจำแบบแรมไม่สามารถเก็บค่าข้อมูลไว้ได้ตลอดดังนั้นหากต้องการให้ข้อมูลคงอยู่ต้องใช้แบตเตอรี่สำรองไฟฟ้าไว้

3. หน่วยอินพุต (Input Unit) เป็นหน่วยที่ใช้สำหรับการรับสัญญาณข้อมูลจากภายนอกเช่น คีย์บอร์ด สแกนเนอร์ หรือที่รับสัญญาณมาจากอุปกรณ์เซนเซอร์ (Sensor) ซึ่งอาจเป็นค่าแรงเสียดทานของล้อรถยนต์ขณะเบรกการกดปุ่มสวิตซ์ตั้งเวลาของวีดีโอเทปฯลฯ กล่าวได้ว่าส่วนที่เป็นอินพุต คือส่วนที่ทำหน้าที่ป้อนข้อมูล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4. หน่วยเอาต์พุต (Output Unit) เป็นหน่วยที่ใช้สำหรับการแสดงผลของข้อมูลเช่นจอมอนิเตอร์ พรินเตอร์ เครื่องตัดสติกเกอร์หรืออุปกรณ์ประเภทแอลอีดี (LED) ลำโพง มอเตอร์ รีเลย์ หลอดไฟ ฯลฯ

2.3.2 ไมโครคอนโทรลเลอร์

ปัจจุบันการพัฒนาและการแข่งขันทางด้านเทคโนโลยีผลิตชิ้นส่วนสารกึ่งตัวนำที่นำไปสร้างเป็นไอซีมี ประสิทธิภาพสูงมากขึ้นและมีเทคโนโลยีที่เกิดจากการผลิตของบริษัทต่างๆ ซึ่ง ส่งผลให้การผลิตชิฟไอซีมีขนาดที่เล็กลง แต่มีประสิทธิภาพและคุณสมบัติต่างๆมากขึ้น ไอซีที่ถูกสร้างเป็นแบบ LSI (Large Scale Integrate Circuit) เป็นเทคโนโลยีการสร้างโดยการนำเอาทรานซิสเตอร์จำนวนมากมาสร้างเป็นไอซีดิจิตอลที่ซับซ้อนโดยทำขึ้นเพื่อหน้าที่เป็นหน่วยประมวลผลข้อมูลหรือ เรียกว่าไมโครโพรเซสเซอร์ (Microprocessor) ที่มีคุณสมบัติหลัก คือการประมวลผลข้อมูลการคำนวณทางคณิตศาสตร์ และลอจิก ถ้าหากมีการติดต่อกับหน่วยความจำที่เป็นแบบแรมแบบรอมหรืออุปกรณ์ภายนอกที่เป็นอินพุต-เอาต์พุตต้องมีการต่ออุปกรณ์อื่น ๆ ร่วมด้วย เพื่อทำหน้าที่เลือกอุปกรณ์ในการติดต่อหรือวงจรถอดรหัส (Decoder) ซึ่งสามารถทำงานได้ภายใต้การควบคุมของโปรแกรมและ ในการที่เรานำไมโครโพรเซสเซอร์มาเป็นตัวประมวลผลกลางมีหน่วยความจำแบบแรมพอร์ตอินพุตและเราเรียกว่า ไมโครคอมพิวเตอร์ เป็นสิ่งไม่คุ้มกับการลงทุนหากนำมาใช้ในงานควบคุมขนาดเล็ก และอาจต้องใช้เนื้อที่มากในการออกแบบ ดังนั้นการพัฒนาด้านเทคโนโลยีในการสร้างชิฟจึงมีการรวบรวมคุณสมบัติที่ต้องการใช้งานมาอยู่ในตัวเดียวกันคือ มีองค์ประกอบเกือบทุกอย่างของคอมพิวเตอร์อยู่ภายในตัวไอซีที่เราเรียกว่าไมโครคอมพิวเตอร์แบบชิฟเดี่ยวประกอบด้วยอุปกรณ์พื้นฐานเหมือนไมโครคอมพิวเตอร์ เช่น หน่วยประมวลผลกลางขนาดเล็ก (8บิต -16 บิต) และหน่วยประมวลผล ที่สามารถเข้าข้อมูลแบบบิตหน่วยความจำข้อมูลพื้นฐานแบบแรมขนาด 128 ไบต์ และบรรจุหน่วยความจำโปรแกรมประเภทรอม(บางเบอร์) สามารถใช้งานให้เป็นได้ทั้งอินพุตและเอาต์พุตมีวงจรสื่อสารอนุกรมแบบ ฟูลดูเพล็กซ์ วงจรCounter/Timer ที่อยู่ภายใน สามารถต่ออุปกรณ์ที่ใช้ในการสร้างวงจรกำเนิดสัญญาณนาฬิกาเช่น คริสตอล(Crystal) และตัวเก็บประจุก็สามารถใช้งานได้เป็นต้น เรา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เรียกกันทั่วๆไปว่า ไอซีไมโครคอนโทรลเลอร์(Microcontroller) ดังนั้นเมื่อเราต้องการใช้งานควบคุมขนาดเล็ก เช่น เตาไมโครเวฟ เครื่องซักผ้า เครื่องเล่นวีดีโอเทปและเครื่องใช้ไฟฟ้าอื่นๆ

2.3.3 เหตุผลที่ต้องเลือกไอซีไมโครคอนโทรลเลอร์ MCS-51

จากความก้าวหน้าของเทคโนโลยีจึงพบว่าอุปกรณ์บางส่วนที่นำมาควบคุมเครื่องใช้ไฟฟ้าระบบวัดคุมทางอิเล็กทรอนิกส์หรือ โครงงานในวารสารต่าง ๆจะนำเสนอในรูปแบบของไอซีไมโครคอนโทรลเลอร์ ซึ่งเทคโนโลยีดังกล่าวมีบริษัทต่างๆ ที่ผลิตออกมาแข่งขันด้านคุณสมบัติ และการใช้งานกันในตลาด อาทิเช่น Intel, Atmel, Microchip, Motorola ส่วนทางด้านการศึกษา และการเรียนรู้นั้นมีการพัฒนาทางด้าน การเรียนการสอนวิชาไมโครคอนโทรลเลอร์ โดยบริษัทต่างๆ แข่งขันกันเพื่อสร้างชุดฝึกตอบสนองต่อความต้องการเรียนรู้ ความสะดวกในการสร้างชุดทดลองของนักศึกษาและผู้ที่สนใจ ในขณะเดียวกันกันเรา ซึ่งเป็นผู้ตามเพื่อศึกษาเทคโนโลยีที่ทิ้งห่างทุกขณะ จึงต้องใช้ชุดทดลองที่ถูกกำหนดรูปแบบและราคา ทั้งที่อุปกรณ์ที่เป็นอุปกรณ์ร่วมกับเทคโนโลยีบางอย่าง เรากลับทิ้งไปและไม่ได้นำมาใช้งาน ยกตัวอย่าง เช่น Main Board ,CD-ROM หรือ Hard Disk ที่ใช้งานไม่ได้แล้ว เรานำอุปกรณ์ภายในมาใช้งานได้กี่ชิ้นบ้าง ตลอดจนตัวต้านทานและตัวเก็บประจุ ที่เป็นอุปกรณ์ติดบนผิวหน้าแบบ SMD (Surface Mounted Device) หน่วยความจำชนิดแรม และรอม แผ่นระบายความร้อนสเตปปิ้งมอเตอร์ (Stepping Motor) คอนเนคเตอร์ (Connector)ฯลฯซึ่งพร้อมที่จะให้เราใช้งานมากกว่าที่จะได้เพิ่มมาจากการซื้อชุดทดลองใหม่

ดังนั้น ไมโครคอนโทรลเลอร์จึงถูกเลือกใช้ในการทดลองด้วยเหตุผลในการใช้อุปกรณ์ที่ต่อร่วมในวงจรน้อยชิ้น ราคาถูก หาแหล่งข้อมูลได้ง่ายมีการพัฒนาประสิทธิภาพ ในการประมวลผล เทียบเท่าไมโครโพรเซสเซอร์ขนาด 8 บิต-16บิต และ ใช้ในการพัฒนากับงานที่ไม่ต้องการความซับซ้อนซึ่งจะทำให้มีความสะดวกมากขึ้นปัจจุบันมีไอซีของสองบริษัทที่เรานิยมใช้ในงานออกแบบ และสำหรับในการเรียนการสอนในสถาบันหลายแห่งคือ ไอซีไมโครคอนโทรลเลอร์ ซึ่งเป็นของบริษัท Microchip ตระกูล MCS-51 และไอซีไมโครคอนโทรลเลอร์ตระกูล MCS-51 จากหลายบริษัทที่ผลิตออกมามากมายหลายเบอร์ แต่คำสั่งจะใกล้เคียงกัน ดังนั้น การทดลองในหนังสือเล่มนี้จึงเลือกใช้งานไอซี ไมโครคอนโทรลเลอร์ของตระกูล MCS-51 ด้วยเหตุผลเพราะมีข้อมูล และตัวอย่างที่หลากหลายหามาใช้งานได้ง่าย แต่ก็ไม่ได้ตัดสินว่าดีกว่าไอซีในตระกูลอื่นๆ ดังนั้น ในการออกแบบชุดทดลองที่ใช้งานร่วมกับบอร์ดไมโครคอนโทรลเลอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ของ MCS-51 ที่จะแนะนำให้สร้างขึ้นนี้ สามารถที่จะนำมาใช้งานร่วมกับ ไอซีไมโครคอนโทรลเลอร์ได้ทุกเบอร์ และทุกตระกูล

2.3.4 ความหมายของไมโครคอนโทรลเลอร์ ไมโครคอนโทรลเลอร์

เป็นชื่อของอุปกรณ์อิเล็กทรอนิกส์แบบหนึ่งที่บรรจุ ความสามารถมากมายไม่ว่าจะเป็นหน่วยประมวลผล หน่วยคำนวณทางคณิตศาสตร์และลอจิก วงจรรับสัญญาณอินพุต วงจรขับสัญญาณออกทางเอาต์พุต หน่วยความจำ และวงจรกำเนิดสัญญาณนาฬิกา ทำให้ไมโครคอนโทรลเลอร์สามารถนำไปประยุกต์ ใช้งานแทนวงจรอิเล็กทรอนิกส์ที่ซับซ้อนได้เป็นอย่างดี โดยช่วยลดจำนวนของอุปกรณ์และขนาดของระบบลงในขณะที่มีขีดความสามารถสูงขึ้นภายใต้งบประมาณที่เหมาะสม

ไมโครคอนโทรลเลอร์มาจากคำ 2 คำรวมกัน คือ “ไมโคร(Micro)” ซึ่งหมายถึง ไมโคร โปรเซสเซอร์ (Microprocessor) ซึ่งเป็นอุปกรณ์ประมวลผลข้อมูลขนาดเล็ก ภายในประกอบด้วยหน่วยประมวลผลกลางหรือซีพียู (CPU : Central Processing Unit) หน่วยคำนวณทางคณิตศาสตร์และลอจิก (ALU : Arithmetic Logic Unit) วงจรเชื่อมต่อหน่วยความจำ และวงจรเชื่อมต่อสัญญาณนาฬิกา อีกคำหนึ่งคือคำว่า “คอนโทรลเลอร์(Controller)” หมายถึงอุปกรณ์ควบคุม ดังนั้น ไมโครคอนโทรลเลอร์ จึงเป็นอุปกรณ์ที่ใช้ในการควบคุม โดยที่สามารถเขียนโปรแกรมเพื่อกำหนดรูปแบบการควบคุมได้อย่างอิสระ

2.3.5 โครงสร้างของไมโครคอนโทรลเลอร์

ไมโครคอนโทรลเลอร์ตระกูล MCS-51 มีจำนวนมาก ขึ้นกับโครงสร้างภายในบางเบอร์มีหน่วยความจำภายในเป็นแบบรอม บางเบอร์เป็นแบบอีพรอม บางเบอร์มีแรมภายใน 128 ไบท์ บางเบอร์มี 256 ไบท์ เป็นต้น คุณสมบัติที่สำคัญของ MCS-51ที่ใช้คือ AT89C51AC3 มีดังนี้

- หน่วยความจำรอม 4 กิโลไบท์
- หน่วยความจำแรม 128 ไบท์
- พอร์ตอินพุทและเอาท์พุท ขนาด 8 บิต 5 พอร์ท
- Timer 16 บิต 3 ตัว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- สามารถอินเทอร์รัพท์ได้ 5 แหล่ง
- วงจรออสซิลเลเตอร์และวงจรนาฬิกาบนชิพ
- พอร์ตอนุกรมสามารถรับส่งข้อมูลแบบ Full Duplex ความเร็วสูง
- อ้างหน่วยความจำโปรแกรมภายนอกได้ 64 กิโลไบท์
- อ้างหน่วยความจำข้อมูลภายภายนอกได้ 64 กิโลไบท์
- สามารถประมวลผลทีละบิต
- อ้างหน่วยความจำแบบบิตได้ 210 ตำแหน่ง
- หนึ่งวัฏจักรคำสั่งกินเวลาประมาณ 1 ไมโครวินาทีขณะทำงานด้วย Clock 12 เมกะเฮิร์ท
- ติดต่อกับหน่วยความจำรอมได้สูงถึง 32 กิโลไบต์
- มีหน่วยความจำแรมภายในขนาด 128 ไบต์และ 256 ไบต์สำหรับเบอร์ 8051
- มีรีจิสเตอร์ 8 บิต 19 ตัว, 16 บิต 5 ตัว -มีพอร์ตอนุกรมในตัว
- สามารถติดต่อกับแหล่งกำเนิดอินเตอร์รัปต์ได้ 5 ชุดและ 9 ชุดสำหรับเบอร์ 8052
- มีพอร์ตอินพุตเอาต์พุต 32 บิต
- มีตัวตั้งเวลา ขนาด 16 บิต 2 ตัว และ 3 ตัว ในเบอร์ 8052
- มีคำสั่ง 111 คำสั่งสามารถอ้างแอดเดรสได้ 2 โหมด

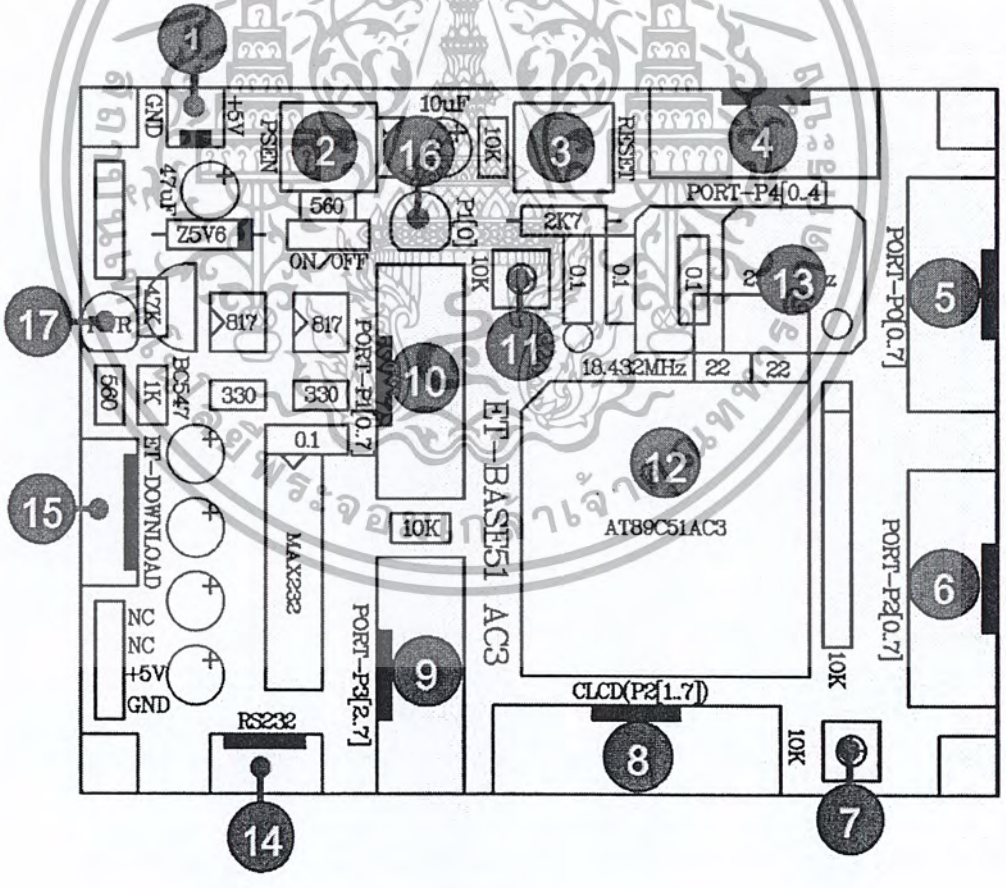
ตัวอย่าง ไมโครคอนโทรลเลอร์ตระกูล MCS-51 และลักษณะต่างๆ สามารถแสดงได้ ดังตารางที่ 1 สำหรับไดอะแกรมโครงสร้างภายในแสดงดังรูปที่ 2.7

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สำนักหอสมุดกลาง พระจอมเกล้าลาดกระบัง

ตารางที่ 2.1 ไมโครคอนโทรลเลอร์ตระกูล MCS-51 เบอร์ต่างๆ

เบอร์	หน่วยความจำโปรแกรมบนชิพ	หน่วยความจำข้อมูลบนชิพ	TIMERS
8051	4K ROM	128 bytes	2
8031	-	128 bytes	2
8751	4K EPROM	128 bytes	2
8052	8K ROM	256 bytes	3
8032	-	256 bytes	3
8752	8K EPROM	256 bytes	3



รูปที่ 2.7 โครงสร้างภายในของ MCS-51

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้
117474

โครงสร้างบอร์ด ET-BASE51 AC3 (AT89C51AC3)

หมายเลข 1 คือ ขั้วต่อแหล่งจ่ายไฟเลี้ยงวงจรของบอร์ด ใช้กับแหล่งจ่ายไฟตรง +5VDC

หมายเลข 2 เป็น Switch PSEN ใช้ร่วมกับ RESET สำหรับ Download แบบ Manual

หมายเลข 3 เป็น Switch RESET ใช้สำหรับ Reset การทำงานของ CPU

หมายเลข 4 เป็น Port-P4 มี ขนาด 5 Bit คือ P4.0-P4.4

หมายเลข 5 เป็น Port-P0 มี ขนาด 8 บิต

หมายเลข 6 เป็น Port-P2 มี ขนาด 8 บิต

หมายเลข 7 เป็น ตัวต้านทานสำหรับปรับค่าความสว่างให้ LCD

หมายเลข 8 เป็น Port-LCD ชนิด Character Typeใช้การเชื่อมต่อแบบ 4 บิต ผ่าน PortP2

หมายเลข 9 เป็น Port-P3 มีขนาด 6 บิต (P3.2-P3.7)

หมายเลข 10 เป็น Port-P1 มีขนาด 8 บิต

หมายเลข 11 เป็นตัวต้านทานสำหรับปรับค่าแรงดันอ้างอิงของ ADC (3V)

หมายเลข 12 คือ MCUเบอร์ AT89C51AC3 ซึ่งเป็น MCU ตระกูล MCS51จาก ATMEL

หมายเลข 13 คือ Oscillator Module ค่า 29.4912 MHz

หมายเลข 14 คือ ขั้วต่อ RS232 สำหรับใช้งานทั่วไป และ Download แบบ Manual

หมายเลข 15 คือ ขั้วต่อ ET-DOWNLOAD ใช้สำหรับ Download แบบ Auto

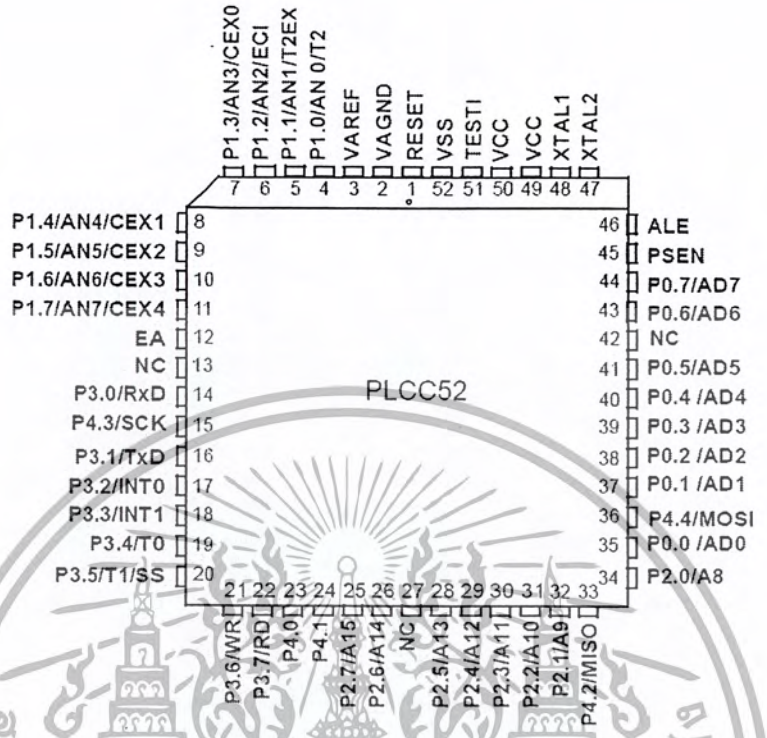
หมายเลข 16 เป็น LED Self Test (P1.0) ใช้สำหรับทดสอบการทำงานของบอร์ด

หมายเลข 17 เป็น LED Power ใช้แสดงสถานะของแหล่งจ่ายไฟ +5VDC

2.3.6 การจัดขาต่าง ๆ ของ MCS-51 ไอซี

ไมโครคอนโทรลเลอร์ 8051 โครงสร้าง มีขาทั้งหมด 52 ขา โดยขาต่างๆ เป็นขาพอร์ทอินพุท, ขาพอร์ทเอาท์พุท, ขาสัญญาณควบคุม, ขาตำแหน่งหน่วยความจำข้อมูลดังรูปที่ 2.8

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.8 ขาต่างๆของ AT89C51AC3

ความหมายขาต่างๆ มีดังนี้

1. พอร์ต 0 (Port 0) ได้แก่ขา 37-44 สามารถเป็นพอร์ตอินพุทและเอาท์พุทได้ ซึ่งในการติดต่อกับหน่วยความจำภายนอกสามารถเป็นขาบัสตำแหน่ง และบัสข้อมูลได้

2. พอร์ต 1 (Port 1) ได้แก่ขา 4-11เป็นพอร์ต 8 บิต สามารถอ้างทีละบิตได้ เป็น P1.0, P1.1,...etc

3. พอร์ต 2 (Port 2) ได้แก่ขา 25-34 สามารถทำงานได้ 2 หน้าที่ เป็นพอร์ต 8 บิต และเป็นขาแอดเดรส 8 บิต ในการอ้างหน่วยความจำภายนอก

4. พอร์ต 3 (Port 3) ได้แก่ขา 14-22 สามารถทำงานได้ 2 หน้าที่ เป็นพอร์ตอินพุทและเอาท์พุท และเป็นขาควบคุมต่างๆ ดังตารางที่ 2

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5. $\overline{\text{PSEN}}$ (Program Store Enable) ขา $\overline{\text{PSEN}}$ ได้แก่ขา45 เป็นขาที่ส่งสัญญาณออก ขานี้ทำการแอคทีฟเมื่อ MCS-51 ต้องการอ่าน Code โปรแกรมภายนอก โดยปกติถ้าหน่วยความจำภายนอกเป็นอีพรอมขา $\overline{\text{PSEN}}$ ต่อกับขา Output Enable $\overline{(\text{OE})}$ ของอีพรอม

ตารางที่ 2.2 บิตและหน้าที่ต่างๆ ของพอร์ต 3

บิต	ชื่อ	หน้าที่พิเศษ
P3.0	RXD	ใช้รับข้อมูลทางพอร์ทอนุกรม
P3.1	TXD	ใช้ส่งข้อมูลทางพอร์ทอนุกรม
P3.2	$\overline{\text{INT0}}$	อินเทอร์รัพท์ภายนอกหมายเลข 0
P3.3	$\overline{\text{INT1}}$	อินเทอร์รัพท์ภายนอกหมายเลข 1
P3.4	T0	ตัวจับเวลา / ตัวนับ ตัวที่ 0
P3.5	T1	ตัวจับเวลา / ตัวนับ ตัวที่ 1
P3.6	$\overline{\text{WR}}$	สัญญาณเขียนข้อมูลหน่วยความจำภายนอก
P3.7	$\overline{\text{RD}}$	สัญญาณอ่านข้อมูลหน่วยความจำภายนอก

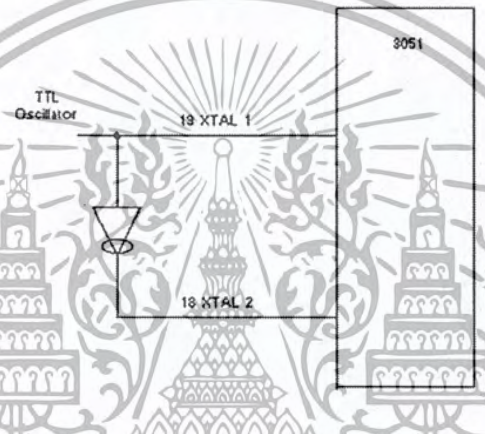
6. ALE (Address Latch Enable) เนื่องจากพอร์ท 0 สามารถเป็นขาอ้างตำแหน่ง และขาข้อมูล MCS-51 ซึ่งมีขา ALE ได้แก่ขา 46 ขานี้ใช้ผสมสัญญาณบัสตำแหน่งของพอร์ท 0 ในการใช้งานระบบ MCS-51 นั้น ต้องมีอุปกรณ์มาต่อกับพอร์ท 0 ที่ทำหน้าที่ Latch สัญญาณบัสตำแหน่ง เมื่อ MCS-51 ติดต่อกับ หน่วยความจำภายนอก MCS-51 ส่งสัญญาณบัสตำแหน่งออกมาก่อนทาง พอร์ท 0 จากนั้นส่งสัญญาณ ALE มา Latch อุปกรณ์ภายนอก ให้เก็บค่าบัสตำแหน่งของพอร์ท 0 ไว้เพื่อใช้พอร์ท 0 เป็นบัสข้อมูลต่อไป

7. $\overline{\text{EA}}$ (External Access) ขา$\overline{\text{EA}}$ ได้แก่ขา 12 ในสภาวะเป็นลอจิก "1" ใช้กับเบอร์ 8051/8052 เพื่อให้อ่านโปรแกรมจากหน่วยความจำภายใน และในสภาวะเป็นลอจิก "0" ทำให้ MCS-51 ทำโปรแกรมโดยอ่านจากหน่วยความจำโปรแกรมภายนอก (เมื่อ $\overline{\text{EA}}$ เป็น "0" ขา $\overline{\text{PSEN}}$ ทำการแอคทีฟ) หากเป็นเบอร์ 8031หรือ 8032 ขา $\overline{\text{EA}}$ เป็น "0" เสมอ เพราะไม่มีโปรแกรม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หน่วยความจำภายใน หากใช้เบอร์ 8051/8052 ซึ่งมีหน่วยความจำภายในและให้ขา $\overline{EA}$ เป็น " 0 " ซึ่งหยุดการทำงานของรอมภายในและอ่านโปรแกรมจากอีพรอมภายนอกแทน

8. RST (Reset) ขา RST ได้แก่ขา 1 เป็นขารีเซต MCS-51 โดยให้เป็นสภาวะลอจิก "1" อย่างน้อย 2 Machine Cycles จึงสามารถรีเซตระบบได้ ความถี่สัญญาณนาฬิกาบนชิพ (ON-Chip Oscillator Inputs) ได้แก่ขา 18-19 เป็นวงจรออสซิลเลเตอร์บนชิพ ซึ่งปกติทำการต่อคริสตัล ความถี่ 12 เมกกะเฮิร์ท กับตัวเก็บประจุ หรืออาจใช้สัญญาณนาฬิกาจากต้นกำเนิดชนิด ทีทีแอลต่อกับ XTAL1 และ XTAL2 ดังรูปที่ 2.9



รูปที่ 2.9 ขาของ MCS-51 ที่ใช้ต่อกับ XTAL

Power Connections ใน MCS - 51 นำแหล่งจ่ายไฟ 5 โวลต์ ต่อเข้ากับ Vcc (ขา 50) ส่วนขา Vss (ขา 52) ต่อลง กราวด์]

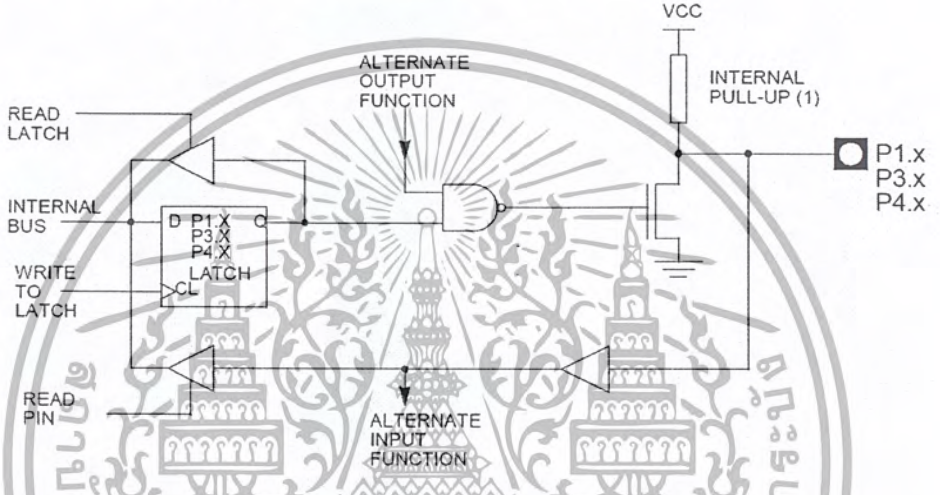
2.3.7 โครงสร้างของพอร์ตอินพุทและเอาท์พุท (I/O Port Structure)

ขาของพอร์ตแสดงโครงสร้างภายในได้ ดังรูปที่ 2-10 โดยมีโครงสร้างเป็น Field-Effect Transistor ต่ออยู่กับขภายนอก และมีความต้านทานต่อ Pull-Up อยู่สำหรับพอร์ต 1, 2, 3 เมื่อเป็นพอร์ต 0 ไม่มีตัวต้านทาน Pull-Up ภายใน เพราะต้องใช้เป็นขาบัสตำแหน่ง และบัสข้อมูล

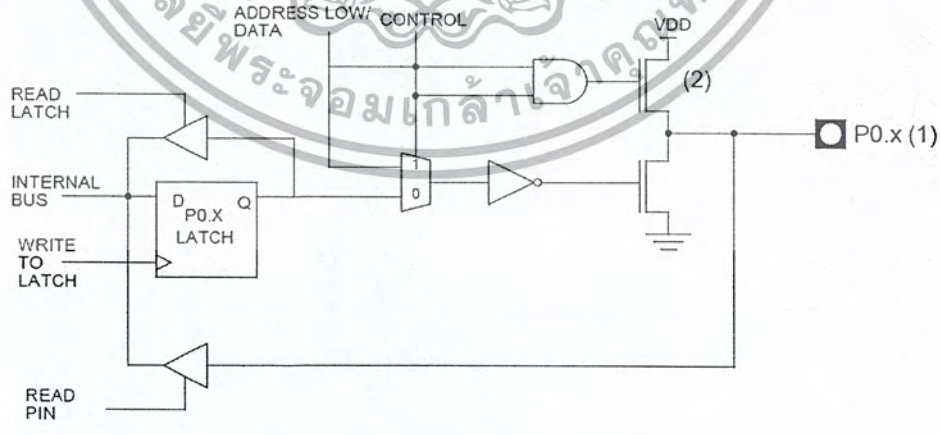
สำหรับขั้วต่อสัญญาณของพอร์ต I/O จาก MCU นั้น จะถูกออกแบบและจัดเตรียมไว้ผ่านทางขั้วต่อแบบ IDE Header ขนาด 10Pin (2x5) จำนวน 5 ชุด คือ PORT-P0, PORT-P1, PORT-P2,PORT-P3 และPORT-P4 ตามลำดับ โดยที่ขั้วต่อสัญญาณแต่ละชุด จะประกอบไปด้วยสัญญาณของ I/O ที่เชื่อมต่อมาจากขาสัญญาณของ MCU โดยตรงทั้งหมด โดยแต่ละพอร์ตจะมีสัญญาณพอร์ตละ 8 บิต ยกเว้น PORTP3และ PORT-P4 โดย PORT-P4 มีขนาด 5

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บิต (P4.0-P4.4) และ PORT-P3 ซึ่งจะมีเพียง 6 บิตเท่านั้นคือ P3.2-P3.7 ส่วน P3.0 และ P3.1 จะถูกสงวนไว้ใช้งานเป็นขาสัญญาณ RXD และ TXD สำหรับรับส่งข้อมูลของ RS232 ซึ่งสัญญาณทั้ง 2 เส้น (P3.0 และ P3.1) จะถูกเชื่อมต่อผ่านวงจร Line Driver (MAX232) สำหรับแปลงระดับสัญญาณจากระดับโลจิก TTL ของ MCU ให้เป็นสัญญาณแรงดันตามมาตรฐานของ RS232 โดยสัญญาณที่ได้รับการแปลงเป็นแบบ RS232 จะถูกเชื่อมต่อไปรอไว้ที่ขั้วต่อแบบCPA ขนาด 4 PIN (RS232) โดยการจัดเรียงสัญญาณของแต่ละชุด จะเป็นดังรูป 2.10

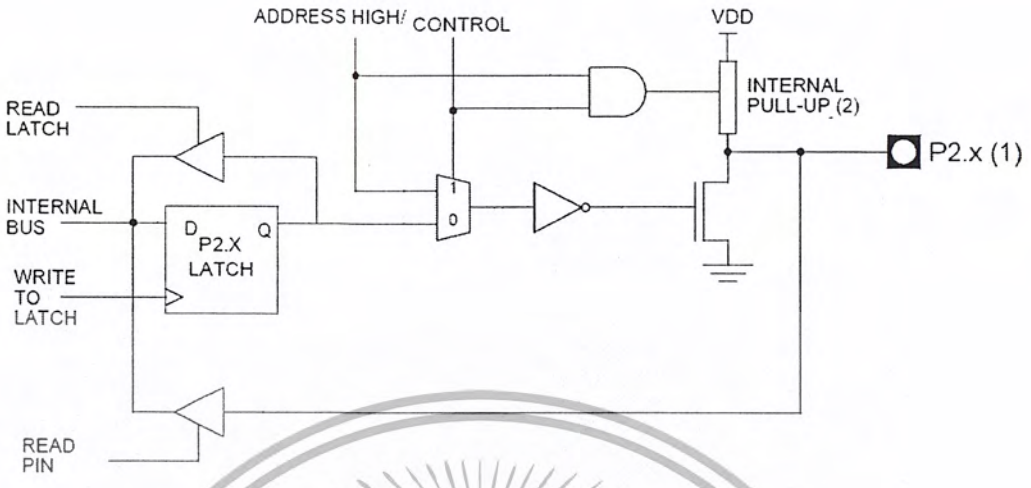


รูปที่ 2.10 โครงสร้างพอร์ต1 ,3 และ 4 ของ MCS-51



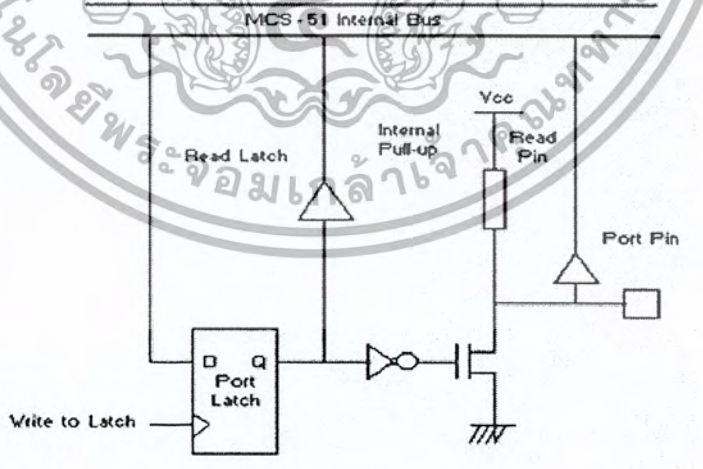
รูปที่ 2.11 โครงสร้างพอร์ต 0 ของ MCS-51

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.12 โครงสร้างพอร์ต 2 ของ MCS-51

พอร์ตนี้สามารถเป็นอินพุทและเอาท์พุทกับอุปกรณ์ภายนอกได้ ในการอ่านข้อมูลจากพอร์ตอ่านได้ 2 แบบ ได้แก่ Read Latch และ Read Pin โดย Read Latch หมายถึงการอ่านข้อมูลที่ถูก Latch เอาไว้เข้าสู่บัสภายในของ MCS-51 เช่นการทำคำสั่ง CPL P1.5 ถ้าเป็นการ Read Pin ซึ่งเป็นการใช้พอร์ตเป็นอินพุท โดยอ่านค่าจากขาของไอซีเข้าสู่บัสภายใน โดยการอ่านแบบ Read Latch และ Read Pinมีสัญญาณมาควบคุมที่บัฟเฟอร์ดังรูปที่ 2.13

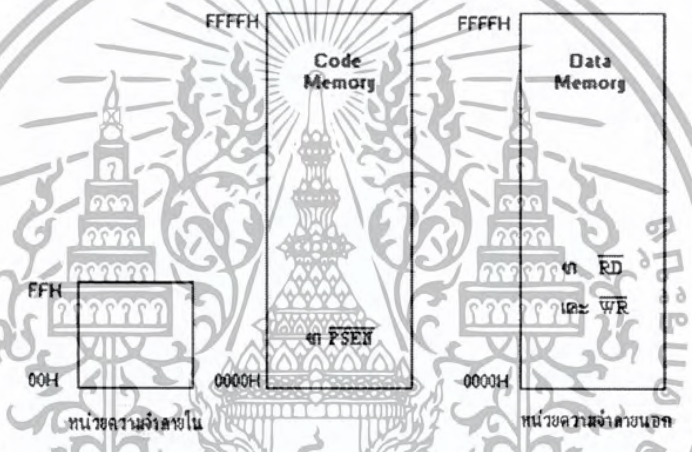


รูปที่ 2.13 การต่อพอร์ทเข้ากับระบบบัสภายในของ MCS-51

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.3.8 โครงสร้างหน่วยความจำ

หน่วยความจำสำหรับ MCS-51 มี 2 ชนิดได้แก่ หน่วยความจำที่ใช้เก็บโปรแกรมรอม กับหน่วยความจำที่ใช้เก็บข้อมูล ในการประมวลผลแรม MCS-51 บางเบอร์ เช่น 8051, 8052 มีหน่วยความจำภายในชิพ และ MCS-51 ทุกเบอร์สามารถอ้างหน่วยความจำ โปรแกรมภายนอกได้มากที่สุด 64 กิโลไบท์ สำหรับหน่วยความจำแรมภายใน ประกอบไปด้วยพื้นที่ใช้งานทั่วไป , รีจีสเตอร์ชุด , พื้นที่ใช้งานระดับบิต และรีจีสเตอร์ฟังก์ชั่นพิเศษ ไดอะแกรมของหน่วยความจำของ 8031 แสดงได้ดังรูปที่ 2.14



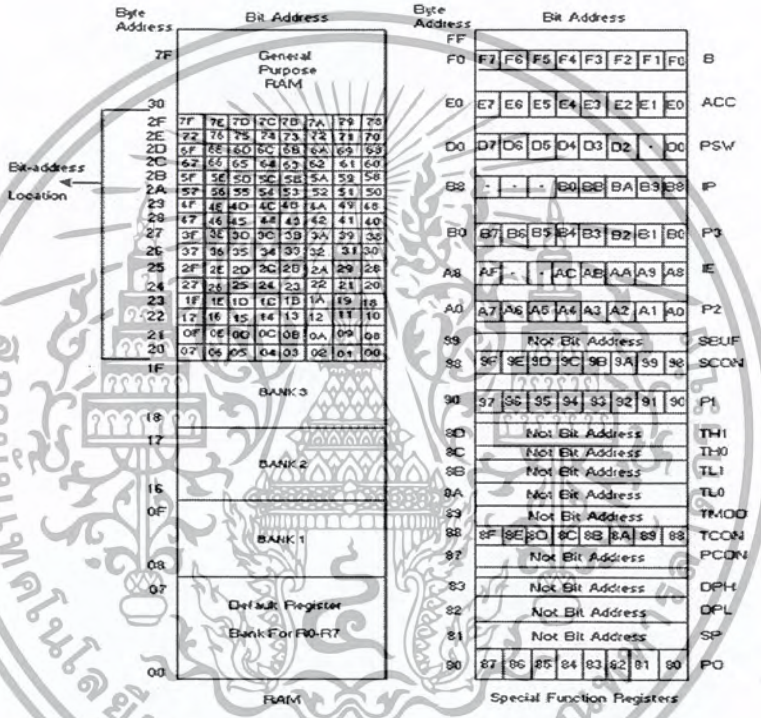
รูปที่ 2.14 การจัดหน่วยความจำของ MCS-51

ใน 8031 มีหน่วยความจำภายใน ตั้งแต่ตำแหน่ง 00H ถึง FFH และสามารถอ้างหน่วยความจำโปรแกรมภายนอกได้ 64 กิโลไบท์ ตำแหน่งที่อ่านข้อมูลจากหน่วยความจำโปรแกรมขา $\overline{PSEN}$ ทำการแอคทีฟ ซึ่ง 8031 สามารถอ้างหน่วยความจำภายนอกได้ 64 กิโลไบท์ ตำแหน่งโดยการติดต่อกับหน่วยความจำนี้ขา$\overline{RD}$ และ$\overline{WR}$ ทำการแอคทีฟ สำหรับหน่วยความจำข้อมูลภายในสามารถแบ่งได้ดังนี้

1. ชุดรีจีสเตอร์ 4 ชุดแต่ละชุดเรียกว่ารีจีสเตอร์ชุด ที่ตำแหน่ง 00H ถึง 1FH โดยแต่ละชุด ประกอบด้วยรีจีสเตอร์ R0 ถึง R7
2. หน่วยความจำที่สามารถเข้าถึงข้อมูลระดับบิตได้ ตำแหน่ง 20H ถึง 2FH
3. หน่วยความจำใช้งานทั่วไปตำแหน่ง 30H ถึง 7FH
4. รีจีสเตอร์ฟังก์ชั่นพิเศษ ตำแหน่ง 80H ถึง FFH

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

แผนผังการจัดการหน่วยความจำข้อมูลภายใน แสดงได้ดังรูปที่ 2-15 จากแผนผังแสดงถึง การอ้างตำแหน่ง หน่วยความจำภายในซึ่งอ้างได้ 2 แบบ ได้แก่ การอ้างไปที่ตำแหน่งของไบท์ (เขียนหมายเลขด้านนอก) หรือการอ้างไปที่ตำแหน่งของบิต (เขียนหมายเลขตำแหน่งด้านใน) โดยตำแหน่งของหน่วยความจำ ที่อ้างเป็นแบบบิตที่แน่นอน



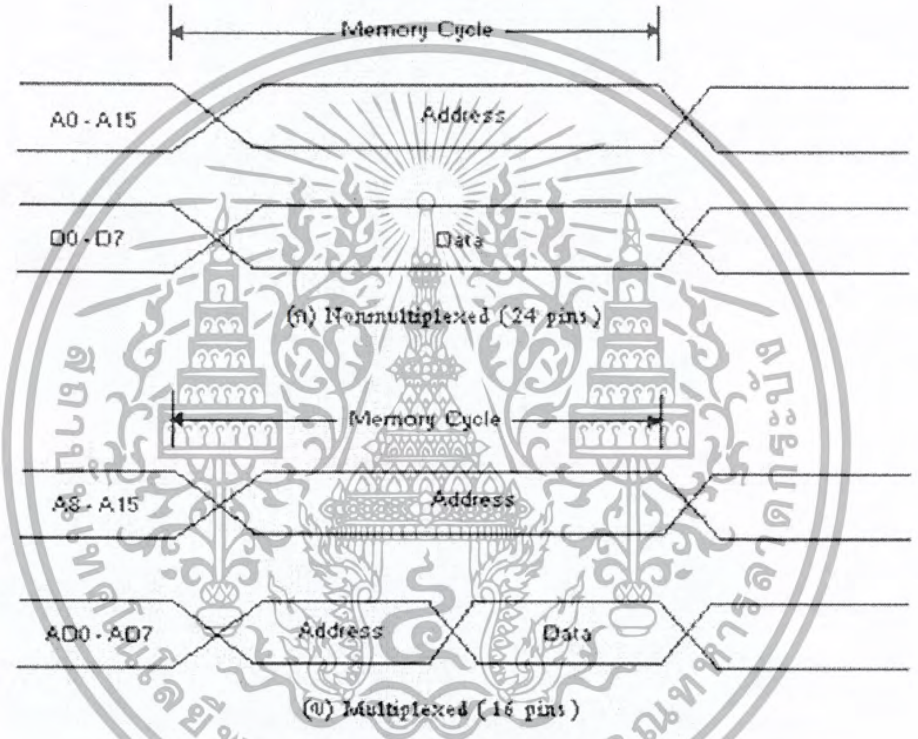
รูปที่ 2.15 ตำแหน่งของหน่วยความจำแบบไบท์และแบบบิต

2.3.9 หน่วยความจำภายนอก MCS-51

สามารถอ้างหน่วยความจำข้อมูลภายนอกได้ 64 กิโลไบท์ MCS-51 ใช้พอร์ท 0 ในการอ้างตำแหน่งหน่วยความจำ 8 บิตล่าง และใช้พอร์ท 0 เป็นพอร์ทข้อมูลด้วย โดยใช้ขา ALE เป็น Latch ข้อมูลพอร์ท 0 และใช้พอร์ท 2 เป็นขาอ้างตำแหน่ง 8 บิตบน (รวมขาอ้างตำแหน่ง 16 เส้น ซึ่งอ้างได้ 64 กิโลไบท์) นอกจากพอร์ท 0 ใช้งาน 2 หน้าที่ ในการติดต่อกับหน่วยความจำ ใช้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

วิธีมัลติเพล็กท์ระหว่างตำแหน่งกับข้อมูล ถ้าต้องการติดต่อกับหน่วยความจำที่เก็บข้อมูล 8 บิต และเก็บได้ 64 กิโลไบท์ ต้องใช้สายสัญญาณ 24 เส้นเป็นขาแอดเดรส 16 เส้น และขาข้อมูล 8 เส้น ดังรูปที่ 19 ถ้าใช้วิธีมัลติเพล็กท์ ใช้ขา A0 - A7 เป็นขาข้อมูล D0- D7 ใช้สายสัญญาณเพียง 16 เส้น รูปที่ 2-16 เมื่อต้องการติดต่อกับหน่วยความจำ ทำการส่งสัญญาณแอดเดรส A0-A15 ออก 16 เส้น จากนั้นขา A0-A7 เปลี่ยนเป็น D0- D7 ในการติดต่อกับหน่วยความจำภายนอก MCS - 51 ใช้วิธีนี้



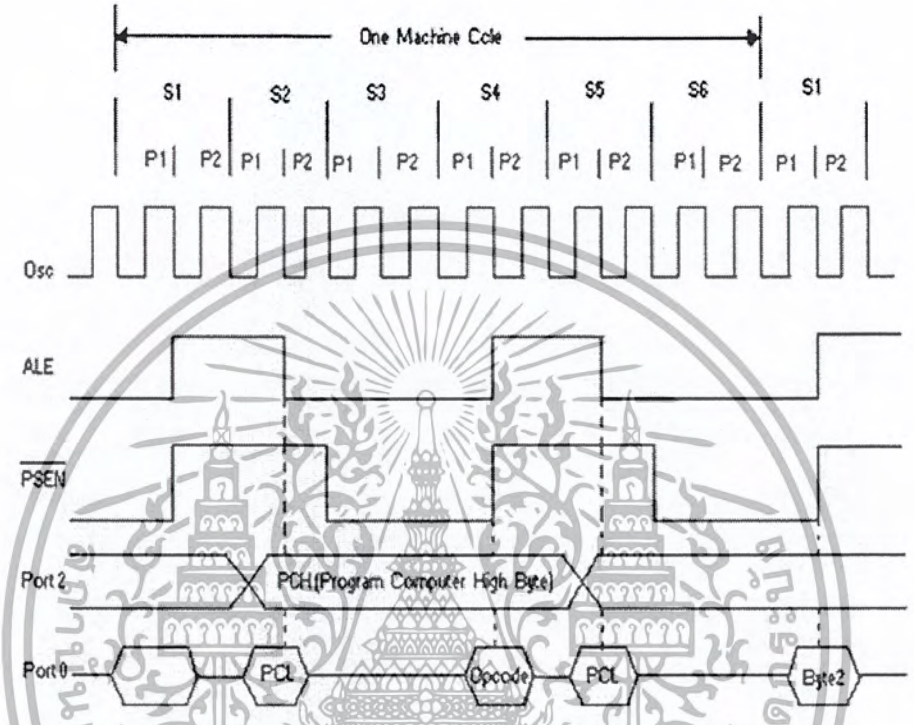
รูปที่ 2.16 ไดอะแกรมกลุ่มสัญญาณที่ใช้อ่านข้อมูล

2.3.10 การติดต่อกับหน่วยความจำโปรแกรมภายนอก

การอ่านข้อมูลที่หน่วยความจำโปรแกรมภายนอก MCS - 51 ส่งค่าตำแหน่งหน่วยความจำออกไป ซึ่งค่าตำแหน่งเก็บอยู่ใน PC ส่งออกทางพอร์ท 0 และพอร์ท 2 จากนั้นส่งขา ALE เป็นลอจิก "0" เพื่อLatch ขาแอดเดรส ของ 8 บิตต่ำคือพอร์ท 0 และส่งสัญญาณทางขา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

$\overline{PSEN}$ เป็นลอจิก " 0 " เพื่ออ่านข้อมูลซึ่งได้ Opcode เข้าไปทางขาบัสข้อมูล คือพอร์ท 0 ไดอะแกรมเวลาการอ่านข้อมูลจากหน่วยความจำภายนอก แสดงได้ดังรูปที่ 2.17

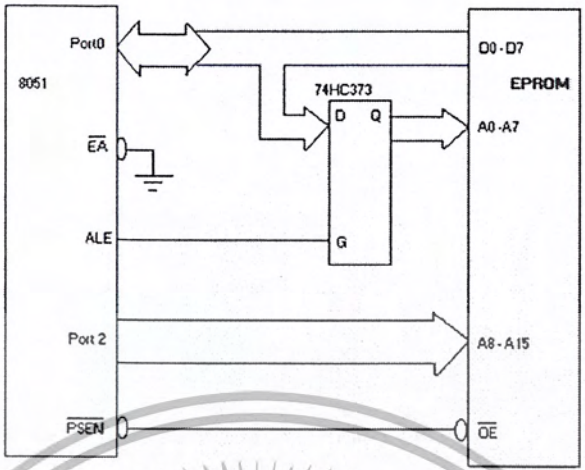


รูปที่ 2.17 ไดอะแกรมการอ่านข้อมูลจากหน่วยความจำโปรแกรมภายนอก

รูปที่ 2-17 ช่วงเวลาการทำงานของ MCS - 51 ใน S2 MCS - 51 ส่งค่าตำแหน่งหน่วยความจำโปรแกรม (ค่า PC) ทางพอร์ต 0 และ พอร์ท 2 จากนั้นส่งขา ALE เป็น " 0 " เพื่อ Latch อุปกรณ์ภายนอกคงค่าตำแหน่งไบท์ต่ำไว้ (พอร์ต 0) เพื่อใช้พอร์ต 0 เป็นขาข้อมูลต่อไป จากนั้นส่งขา $\overline{PSEN}$ เป็น " 0 " เพื่ออ่าน Opcode เข้าทางพอร์ต 0

การต่อหน่วยความจำกับ MCS - 51 แสดงได้ดังรูปที่ 21 โดยขา $\overline{EA}$ ต่อเป็น "0" เพื่อให้ MCS - 51 อ่านหน่วยความจำโปรแกรมภายนอกสำหรับการ มัลติเพล็กท์ ใช้ฟลิปฟลอป 8 ตัวเบอร์ 74373 เก็บค่าตำแหน่ง 8 บิตต่ำไว้ เมื่อ MCS - 51 ส่งค่าตำแหน่งพอร์ทออกไปจากนั้นส่งขา ALE ให้เป็น "0" ซึ่งใช้ขานี้ต่อกับ 74373 เพื่อ Latch ข้อมูลสำหรับขา $\overline{PSEN}$ ต่อกับขา Output Enable $\overline{OE}$ ของหน่วยความจำดังรูปที่ 2.18

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

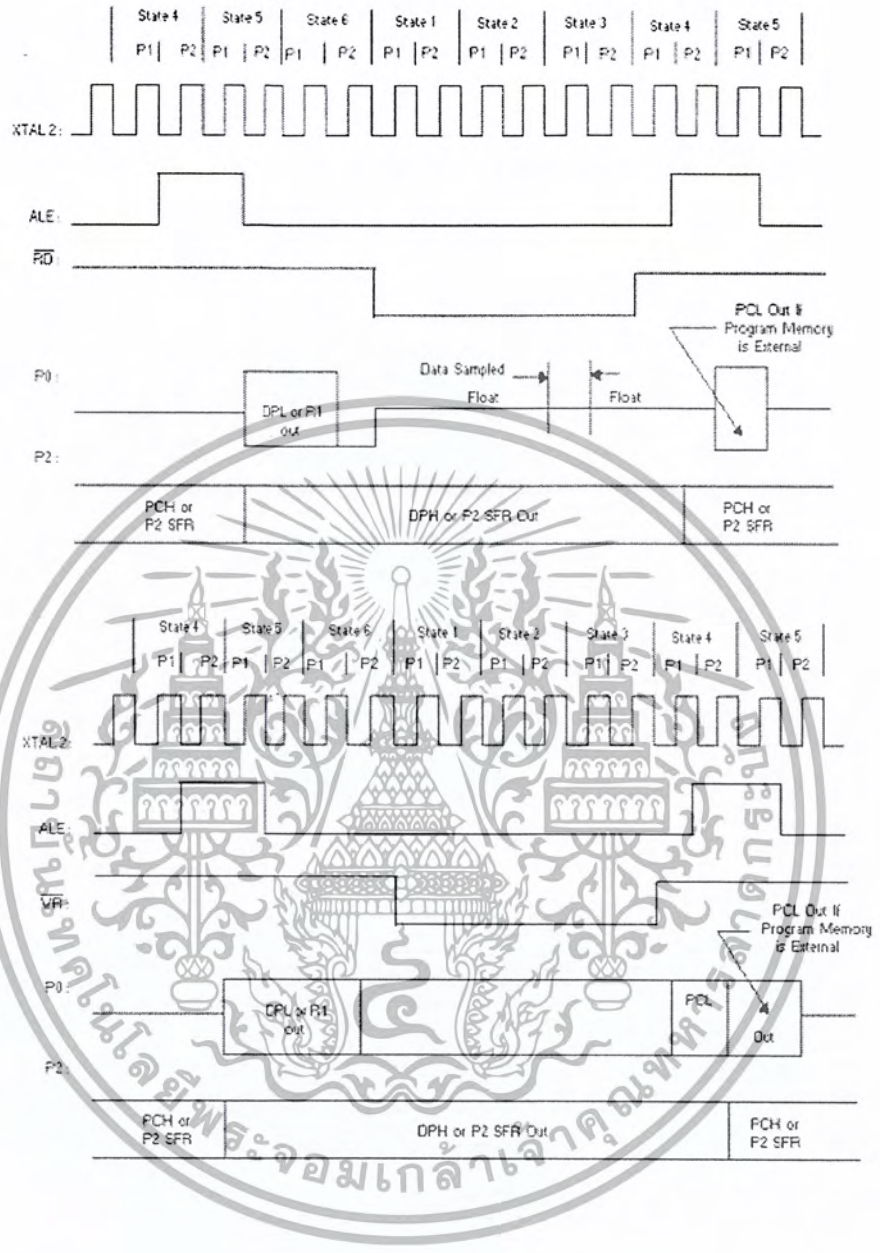


รูปที่ 2.18 การต่อ MCS - 51 กับหน่วยความจำโปรแกรมภายนอก

2.3.11 การติดต่อกับหน่วยความจำข้อมูลภายนอก

หน่วยความจำข้อมูลภายนอก MCS - 51 สามารถอ่าน และ เขียนได้ ในการติดต่อกับหน่วยความจำข้อมูลภายนอก MCS - 51 ส่งขาแอดเดรส ออกทางพอร์ท 0 และ พอร์ท 2 จากนั้นส่งขา ALE เพื่อไป Latch Address 8 บิตต่ำ โดยการอ่านเขียนข้อมูลนั้นใช้ขา $\overline{RD}$ หรือ P3.7 และขา $\overline{WR}$ หรือ P3.6 ตามลำดับ ไดอะแกรมเวลาการอ่านและเขียนข้อมูลกับหน่วยความจำภายนอก แสดงดังรูปที่ 2.19

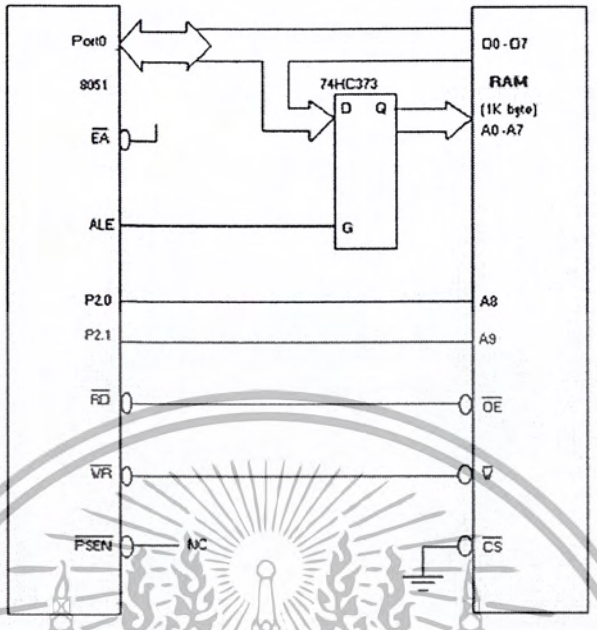
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.19 ไดอะแกรมเวลาการอ่านและเขียนข้อมูลกับหน่วยความจำภายนอก

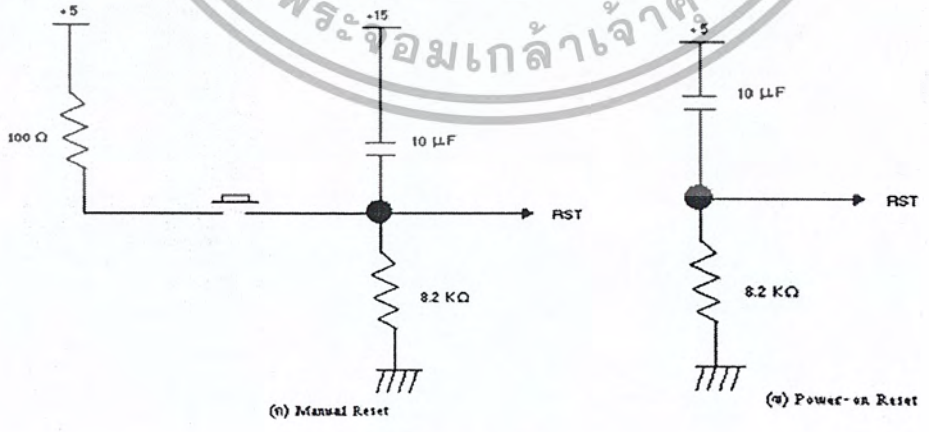
การเชื่อมต่อหน่วยความจำข้อมูลกับ MCS-51 ให้ 8051 ทำงานกับหน่วยความจำแสดงได้ดังรูปที่ 23 เป็นการเชื่อมต่อแรมขนาด 1 กิโลไบท์ ใช้ขา แอดเดรส เพียง 10 เส้น ดังนั้น A8 และ A9 ต่อกับ P2.0 และ P2.1 ส่วนขา $\overline{EA}$ ต่อกับลอจิก "1" เพื่อให้อ่านโปรแกรมจากรอมภายใน และขา $\overline{PSEA}$ ไม่ใช้เพราะไม่ได้ต่อรอมแสดงได้ดังรูปที่ 2.20

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.20 การต่อหน่วยความจำโปรแกรมกับMCS-51

Reset Operation การรีเซตหรือเริ่มต้นทำงานใหม่ของ MCS-51 ให้ลอจิก "1" ที่ขา RST เป็นเวลา 2 Machine Cycles (1 Machine Cycle เท่ากับ 12 Clock) จากนั้นให้กลับเป็นลอจิก "0" การรีเซตอาจทำได้โดยใช้สวิตซ์กด ดังรูปที่ 24 (ก) หรือใช้วิธี Power-up โดยใช้ตัว R-C ต่อเป็นวงจรดังรูปที่ 2-21



รูปที่ 2.21 การรีเซต MCS-51

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.3.12 การใช้งานพอร์ตสื่อสารอนุกรม

พอร์ตสื่อสารอนุกรมมีโครงสร้างการทำงานในแบบที่เรียกว่าฟูลดูเพล็กซ์ (Full Duplex) สามารถรับและส่งข้อมูลอนุกรมได้ในเวลาเดียวกันSerial Port Buffer (SBUF) ใช้เป็นบัฟเฟอร์สำหรับรับและส่งข้อมูลอนุกรมโดยมีอยู่ 2 ตัวพอร์ตสื่อสารอนุกรมสามารถโปรแกรมการทำงานได้หลายโหมดด้วยกันโดยเลือกทีละบิต SM1และ SM0 ซึ่งอยู่ในรีจิสเตอร์ควบคุม SCON การทำงานของทั้ง 4 โหมด ของพอร์ตสื่อสารอนุกรม มีดังตารางที่ 2.3

ตารางที่ 2.3 โหมดการทำงานของพอร์ตสื่อสารอนุกรม

SM0	SM1	โหมด	การทำงาน
0	0	0	Shift Register ความเร็วในการรับหรือส่งข้อมูลเท่ากับ (1/2) ของ CPU Osc
0	1	1	8 Bit UART ความเร็วในการรับหรือส่งข้อมูลกำหนดได้จาก Timer 1,2
1	0	2	9 Bit UART ความเร็วในการรับหรือส่งข้อมูล = (1/32) หรือ (1/64) เท่าของ CPU Osc โดยขึ้นกับบิต SMOD ใน PCON
1	1	3	9 Bit UART ความเร็วในการรับหรือส่งข้อมูลกำหนดที่ Timer 1,2

โหมด 0 : พอร์ตสื่อสารอนุกรม 8 บิต โดยการส่งข้อมูลเลื่อนออกทีละบิตโดยส่งบิต D0 ออกไปก่อน ทางขา R_XD เนื่องจากไม่มีการส่งบิตเริ่มต้น แต่จะส่ง Shift Clock ทางขา TxD (ความเร็ว 1/12 เท่าของ CPU Osc)

โหมด 1 : พอร์ตสื่อสารข้อมูลอนุกรม 10 บิต ข้อมูล 8 บิต 1 บิตเริ่มต้น และ 1 บิตสิ้นสุด และสามารถเปลี่ยนแปลงความเร็วในการส่งข้อมูลได้ โดยขึ้นกับบิต SMOD ใน PCON และอัตราโอเวอร์ไฟล์ของ ไทเมอร์ 1,2

$$\text{บอดเรท โหมด 1,3} = \frac{2^{\text{SMOD}} \times \text{CPU Osc}}{32 \times 12[256-(\text{TH1})]} \quad \text{โดยใช้ ไทเมอร์ 1} \qquad (1)$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

$$\text{บอดเรท โหมด 1,3} = \frac{\text{CPU Osc}}{32\times[65536-(\text{RCAP2H,RCAP2L})]} \quad \text{โดยใช้ไทเมอร์ 2} \qquad (2)$$

โหมด 2 : พอร์ตสื่อสารอนุกรม 11 บิต ข้อมูล 9 บิต 1 บิตเริ่มต้น และ 1 บิตสิ้นสุด (TB8 นิยมนำมาใช้ ส่งพาริตี้บิต) ความเร็วในการรับส่งข้อมูลเท่ากับ 1/32 หรือ 1/64 เท่าของ CPU Osc โดยขึ้นกับบิต SMOD ใน PCON

$$\text{บอดเรท (โหมด 2)} = \frac{(2^{\text{SMOD}})\text{CPU Osc}}{64} \qquad (3)$$

- บอดเรท (โหมด 2) = 1/32 CPU Osc เมื่อ SMOD = 1
- บอดเรท (โหมด 2) = 1/64 CPU Osc เมื่อ SMOD = 0

โหมด 3 : พอร์ตสื่อสารอนุกรม 11 บิต ข้อมูล 9 บิต 1 บิตเริ่มต้น และ 1 บิตสิ้นสุด เหมือนโหมด 2 ยกเว้นอัตราความเร็วจะขึ้นกับบิต SMOD ใน PCON และอัตราโอเวอร์ไฟล์ของ ไทเมอร์ 2

$$\text{บอดเรท โหมด 1, 3} = \frac{2^{\text{SMOD}}\times\text{CPU Osc}}{32\times12[256-(\text{TH1})]} \quad \text{โดยใช้ไทเมอร์ 1} \qquad (4)$$

$$\text{บอดเรท โหมด 1,3} = \frac{\text{CPU Osc}}{32\times[65536-(\text{RCAP2H,RCAP2L})]} \quad \text{โดยใช้ไทเมอร์ 2} \qquad (5)$$

2.3.13 กระบวนการรับและส่งข้อมูลอนุกรมของ MCS-51

การส่งข้อมูลออกทางพอร์ตอนุกรมของ MCS-51 จะเริ่มต้นขึ้นภายหลังเมื่อมีการเขียนข้อมูลลงไปใน SBUF ข้อมูลนี้ จะถูกเลื่อนทีละบิต และส่งสัญญาณออกไปภายนอกโดยอัตโนมัติ เมื่อข้อมูลเหล่านี้ได้ส่งออกไปครบถ้วนแล้วจะทำให้ค่าแฟลกซ์ T1 เป็น 1 เพื่อแจ้งให้ทราบว่าขณะนี้ SBUF ว่างและพร้อมที่จะส่งข้อมูลไบต์ต่อไปแล้ว ในกรณีที่ใช้เขียนข้อมูลใหม่ลง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ในรีจิสเตอร์ SBUF โดยไม่รอให้แฟลกซ์ T1 มีค่าเป็น 1 ก่อน จะมีผลทำให้ข้อมูลที่ส่งไปผิดพลาดได้

สำหรับการรับข้อมูลจากพอร์ตอนุกรมจะต้องเริ่มต้นโดยการกำหนดค่า REN (Receive Enable) ให้มีค่าเป็น 1 ก่อน หลังจากนั้นเมื่อมีข้อมูลภายนอกถูกส่งเข้ามายัง 8051 ทีละบิตจนครบ และเมื่อบิตสุดท้ายเลื่อนเข้ามาเรียบร้อยแล้ว ข้อมูลนั้นจะถูกย้ายมาเก็บไว้ยังรีจิสเตอร์ SBUF และแฟลกซ์ RI ก็จะมีค่าเป็น 1 (ถูกเซต)

1. พอร์ทอนุกรม (โหมด 0) การทำงานของพอร์ตอนุกรม (โหมด 0) เป็นการรับและส่งข้อมูลอนุกรมจำนวน 8 บิต โดยใช้เพียงขาสัญญาณ R_XD เท่านั้น (ขานี้ใช้งาน 2 หน้าที่ใช้ส่งและรับข้อมูล) ส่วนขาสัญญาณ T_XD จะนำไปใช้เพื่อเป็นขาสัญญาณนาฬิกาในการให้จังหวะการเลื่อนข้อมูลกับวงจร เลื่อนบิตภายนอก สำหรับอัตราเร็วจะถูกกำหนดไว้คงที่ที่ค่า 1/12 เท่าของ CPU Osc จากรูปที่ 25 แสดงให้เห็นถึงแผนภาพเวลาสัญญาณต่างๆ ในโหมด 0 เมื่อมีการรับหรือส่งข้อมูล 1 ไบต์ โดยสัญญาณนาฬิกาในการเลื่อนบิตนี้จะเกิดภายในตัว 8051 เอง เนื่องจากโหมดนี้ไม่มีการส่งบิตเริ่มต้นและบิตสิ้นสุด ดังนั้นจึงจำเป็นที่จะต้องส่งสัญญาณ Shift clock ออกไป เพื่อใช้ Synchronize ระหว่างฝ่ายรับและฝ่ายส่ง โดยจะใช้ขา T_XD ส่วนการรับข้อมูลจะรับข้อมูลเข้าทางขา R_XD และรับ Shift clock เข้าทางขา T_XD ถ้า CPU Osc มีค่าเท่ากับ 12 MHz ก็จะส่งได้ถึง 1 ล้านบอด ซึ่งโหมด 0 เป็นโหมดที่ส่งข้อมูลได้เร็วที่สุด รายละเอียดผังเวลาในการรับส่งดังแสดงในรูปที่ 2.22

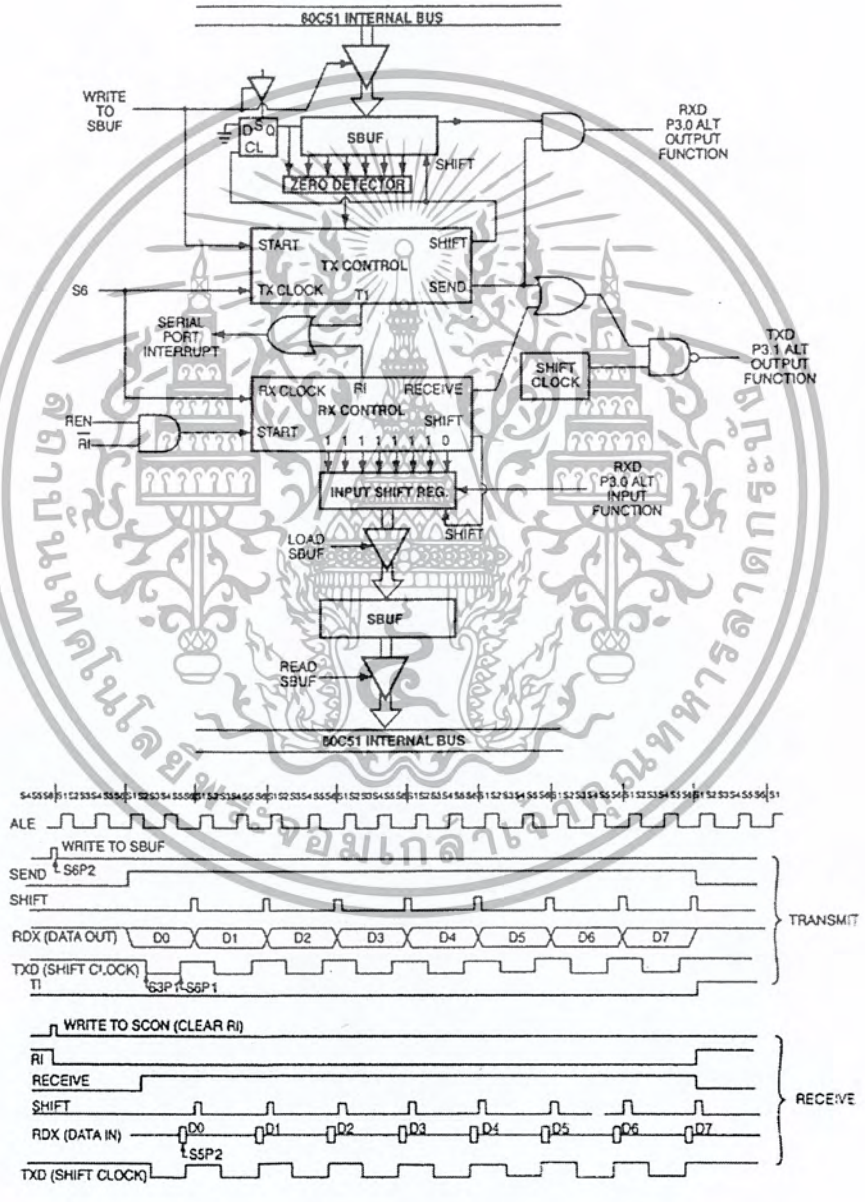
2. พอร์ตอนุกรม (โหมด 1) การทำงานในโหมด 1 เป็นการสื่อสารข้อมูลอนุกรมจำนวน 10 บิต ประกอบด้วยบิตเริ่มต้น 1 บิต ข้อมูลจำนวน 8 บิต และบิตสิ้นสุดอีก 1 บิต ดังแสดงในรูปที่ 26 โดยข้อมูลจะถูกส่งออกทาง T_XD และรับเข้ามาทางขาสัญญาณ R_XD ในส่วนของข้อมูล 8 บิต ที่ได้รับหรือทำการส่งออก จะเป็นบิตนัยสำคัญต่ำเป็นลำดับแรก ทางฝ่ายรับค่าของบิตสิ้นสุดจะส่งเข้ามาจัดเก็บไว้ในบิต RB8 ภายในรีจิสเตอร์ SCONสำหรับอัตราเร็วในการส่งข้อมูลของโหมด 1 นั้น สามามารถเลือกได้จากไทเมอร์ 1 ผังเวลาการทำงานแสดงดังในรูปที่ 2.23

3. พอร์ตอนุกรม (โหมด 2) โหมด 2 ใช้ทั้งหมด 11 บิต โดยแบ่งเป็น บิตเริ่มต้น 9 บิตข้อมูล และบิตสิ้นสุด โดยบิตที่ 9 ผู้ใช้สามารถกำหนดค่าเองได้ว่าจะส่งค่าอะไรออกไป โดยจะต้องนำไปใส่ไว้ในบิต TB8 ในรีจิสเตอร์ SCON ส่วนมากผู้ใช้มักจะนำบิตนี้มาใช้เป็นพาริตี้บิต โดยรับค่ามาจากพาริตี้แฟลกซ์ใน PSW ส่วนทางด้านรับคือ บิตที่ 9 จะถูกนำไปเก็บไว้ใน RB8

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

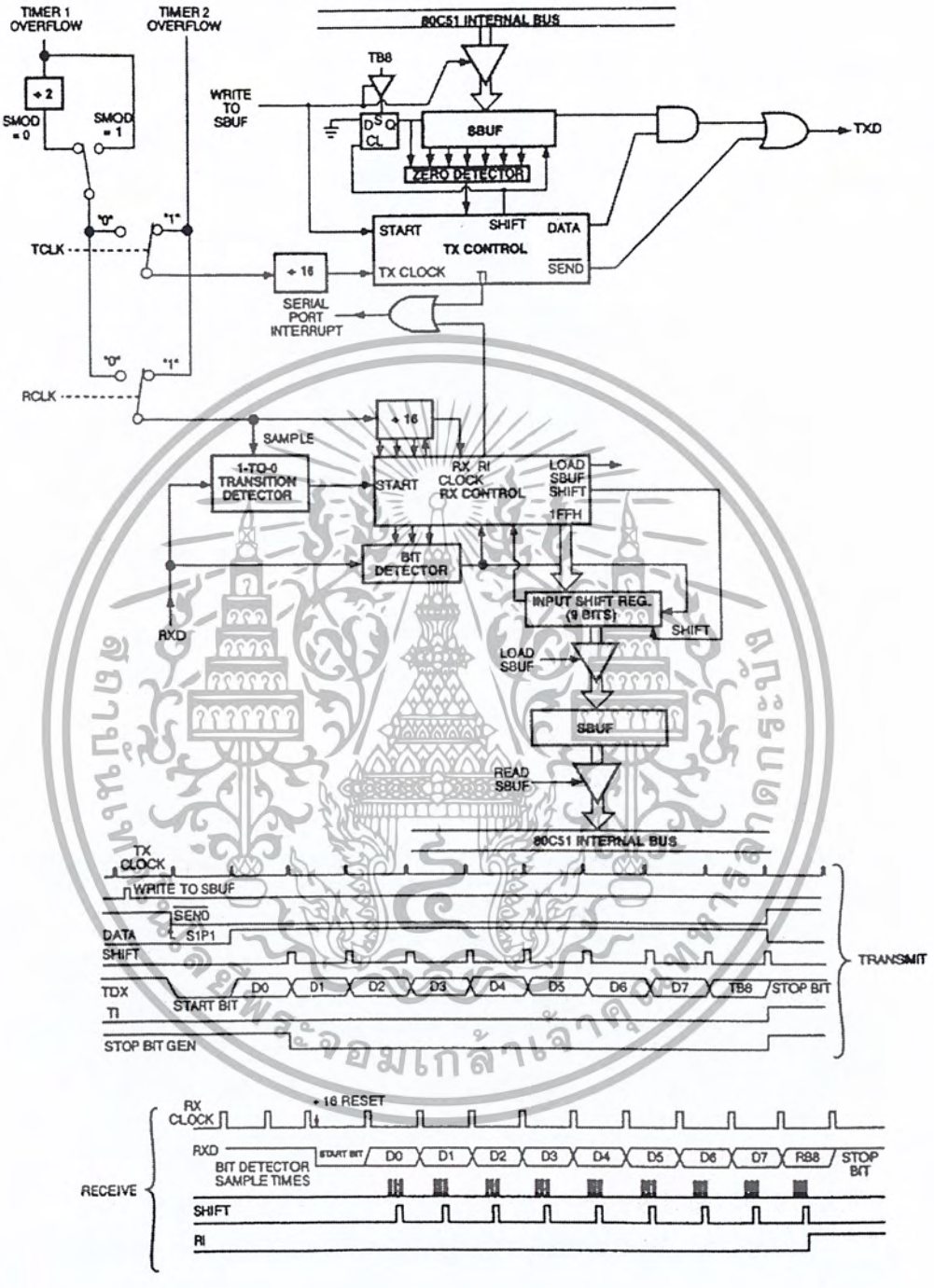
อัตราเร็วในการส่ง/รับข้อมูลกับ CPU Osc และค่า SMODซึ่งอยู่ในบิต 7 ใน SCON ผังเวลาการทำงานแสดงดังในรูปที่ 2.24

4. พอร์ตอนุกรม (โหมด 3) การทำงานเหมือนกับโหมด2 ทุกอย่างจะแตกต่างกันที่ความเร็วในการรับส่งข้อจะขึ้นอยู่กับอัตราโอเวอร์โฟลว์ของไทเมอร์ 1 หรือ ไทเมอร์ 2 โดยมีผังการทำงานแสดงดังในรูปที่ 2.25



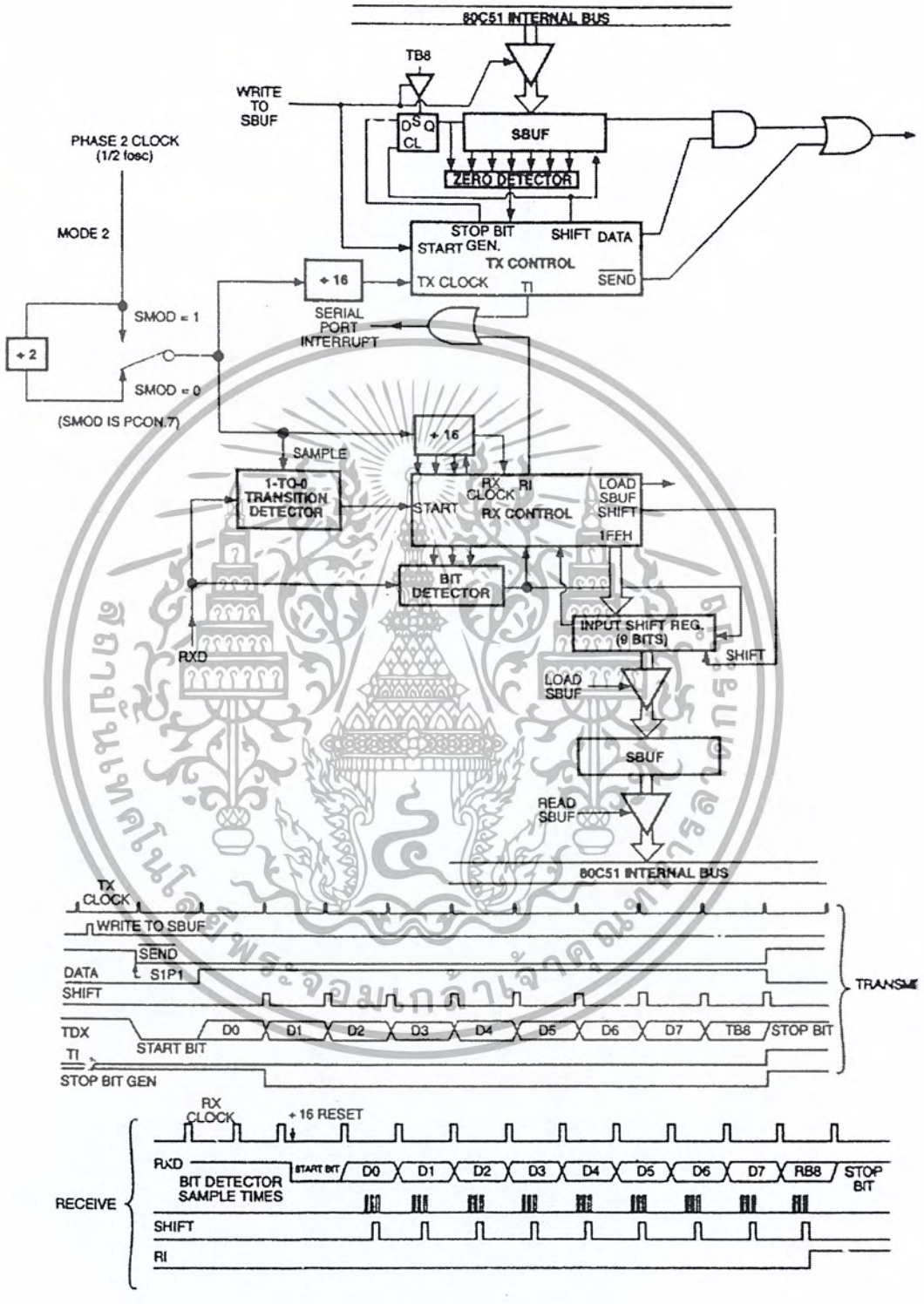
รูปที่ 2.22 ผังการทำงานโหมด 0

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



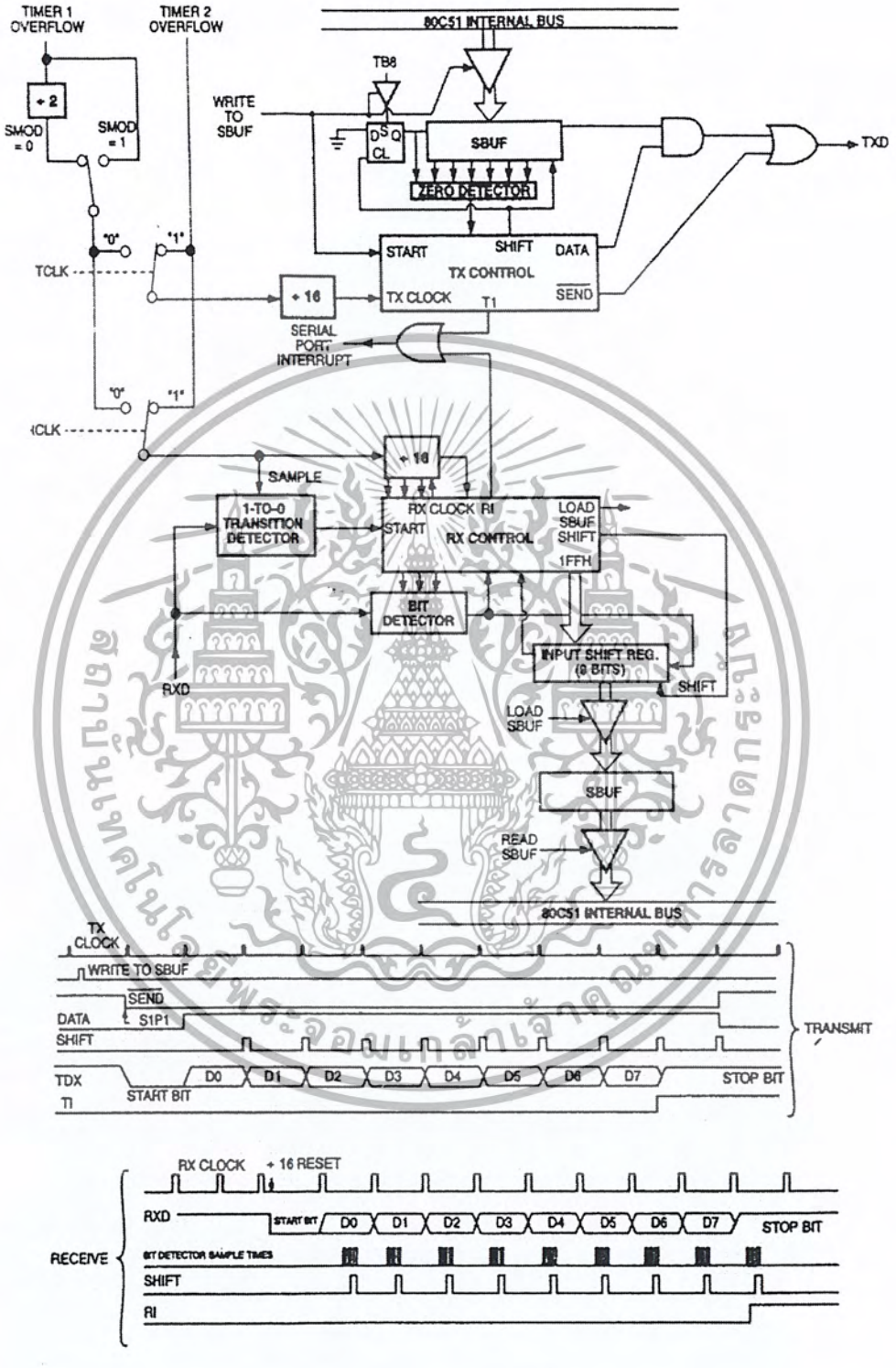
รูปที่ 2.23 ผังการทำงานโหมด 1

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.24 ผังการทำงานโหมด 2

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.25 ผังการทำงานโหมด 3

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.4 การสื่อสารแบบอนุกรม

ถึงแม้ว่าการสื่อสารแบบอนุกรมในเครื่องคอมพิวเตอร์นั้นจะมีความเร็วในการสื่อสารช้ากว่าแบบขนาน ทั้งนี้ก็เพราะว่าการเคลื่อนย้ายข้อมูลแบบอนุกรมนั้นเป็นการส่งข้อมูลครั้งละ 1 บิต แต่พอร์ตขนานนั้นสามารถส่งข้อมูลได้ครั้งละหลายๆบิตพร้อมกันส่งผลให้การสื่อสารข้อมูลแบบอนุกรมมีความเร็วต่ำกว่าแบบขนาน แต่การส่งข้อมูลแบบอนุกรมนั้นมีข้อที่เหนือกว่าการส่งข้อมูลแบบขนานคือ การสามาส่งข้อมูลได้ในระยะทางที่ไกลกว่าแบบขนาน อีกทั้งสายสัญญาณที่ใช้ยังมีน้อยกว่าการส่งข้อมูลแบบขนาน การสื่อสารแบบอนุกรมสามารถแบ่งออกเป็น 3 รูปแบบ ดังนี้

1. แบบ Simplex สามารถส่งข้อมูลได้อย่างเดียว เป็นการสื่อสารแบบทางเดียว
2. แบบ Half-Duplex สามารถส่งข้อมูลไปยังปลายทางและสามารถรับข้อมูลจากปลายทางได้ แต่ไม่สามารถทำการส่งและรับข้อมูลได้ในเวลาเดียวกัน
3. แบบ Full-Duplex สามารถรับและส่งข้อมูลได้ในเวลาเดียวกัน

นอกจากนี้แล้วยังสามารถแบ่งประเภทของการสื่อสารแบบอนุกรมตามลักษณะสัญญาณในการส่งได้อีก 2 แบบ

1. การสื่อสารแบบซิงโครนัส สำหรับการสื่อสารแบบซิงโครนัสนี้จะใช้สัญญาณนาฬิกาควบคุมการรับส่งสัญญาณ เช่น สายคีย์บอร์ดคอมพิวเตอร์ ส่วนอีกเส้นหนึ่งเป็นสายของข้อมูล (และมักจะมีสายกราวด์) สำหรับการสื่อสารแบบซิงโครนัสนี้เหมาะสำหรับการทำงานในระยะใกล้ ข้อมูลที่จะส่งมีไม่มากนัก เพราะถ้าระยะทางไกลขึ้นจะทำให้สัญญาณนาฬิกามีปัญหา อีกทั้งต้องมีสายหลายเส้นทำให้สิ้นเปลืองมาก
2. การสื่อสารแบบอะซิงโครนัสสำหรับการสื่อสารแบบอะซิงโครนัสนั้นจะใช้สายข้อมูลเพียงตัวเดียว แต่จะใช้รูปแบบการส่งข้อมูล หรือ bit Pattern เป็นตัวกำหนดว่าส่วนไหนเป็นส่วนเริ่มต้นข้อมูล ส่วนไหนเป็นตัวข้อมูล ส่วนไหนจะเป็นส่วนตรวจสอบความถูกต้องข้อมูล และส่วนไหนเป็นส่วนปิดท้ายของข้อมูล โดยต้องกำหนดให้สัญญาณนาฬิกาเท่ากันทั้งภาคส่งและภาครับ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.4.1 มาตรฐาน RS-232C

มาตรฐาน RS-232C เป็นมาตรฐานที่ได้รับการออกแบบมาเพื่อที่จะทำให้อุปกรณ์ต่อพ่วงจากผู้ผลิตต่างกันสามารถทำงานร่วมกันได้ มาตรฐานหลายชนิดได้รับการออกแบบขึ้นมา แต่มาตรฐานที่ได้รับความนิยมและใช้กันกว้างขวางมากอย่างที่สุดคือ มาตรฐาน RS-232C ซึ่งถูกประกาศใช้ในปี1969 โดยสมาคมอุตสาหกรรมอิเล็กทรอนิกส์ ในยุคแรกๆ การอินเตอร์เฟสแบบ RS-232C ถูกออกแบบสำหรับเชื่อมต่อเทอร์มินอล (DTE) กับโมเด็ม (DCE) ทั้งนี้ก็เพื่อป้องกันไม่ให้เกิดการส่งข้อมูลบนสายเส้นเดียวกัน

มาตรฐาน RS-232C ได้มีการแบ่งอุปกรณ์ออกเป็น 2 ประเภท ซึ่งอุปกรณ์ทั้งสองประเภทนี้ คือ

1. อุปกรณ์ DTE (Data Terminal Equipment) เป็นอุปกรณ์สำหรับส่งข้อมูล (เอาต์พุต)
2. อุปกรณ์ DCE (Data Communication Equipment) เป็นอุปกรณ์สำหรับรับข้อมูล (อินพุต)

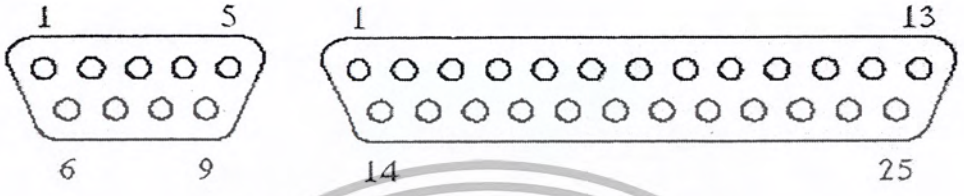
ตามมาตรฐาน RS-232C แล้วคอนเน็กเตอร์ของ DTE จะเป็นตัวผู้ ส่วนคอนเน็กเตอร์ของDCE จะเป็นตัวเมีย ซึ่งคอนเน็กเตอร์ที่นิยมใช้อยู่จะเป็นชนิด D-type แบบ 9 ขา และแบบ 25 ขา สัญญาณทางไฟฟ้าของ RS-232C กำหนดว่าถ้าเป็นลบมากกว่า -3 จะเป็น 1 และถ้าเป็น +4V ขึ้นไปจะเป็น 0 ความเร็วสูงสุดของข้อมูลสูงสุด ได้ถึง 20 Kbps สำหรับลอจิก 1 และลอจิก 0 จะมีระดับแรงดันตามที่กล่าวมาจะสามารถรับส่งข้อมูลได้ที่ความยาวของสายสัญญาณสูงสุด 50 ฟุต หรือ150 เมตร แต่ถ้าเราต้องการสื่อสารกับอุปกรณ์อื่นที่อยู่ห่างกันมากๆ จึงจำเป็นต้องใช้อุปกรณ์อื่นๆ เข้าช่วย เช่น การใช้โมเด็มเป็นต้น

2.4.2 ลักษณะของคอนเน็กเตอร์แบบ D- Type

หัวต่อแบบ D- Type ที่ใช้ในการสื่อสารแบบอนุกรมของเครื่องคอมพิวเตอร์นั้นจะมีอยู่ 2ลักษณะ คือ แบบ 9 ขา และแบบ 25 ขา บางครั้งเราจะเรียกว่า DB9 และ DB25 ซึ่งหัวต่อ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ทั้งสองชนิดจะมีลักษณะการทำงานของสัญญาณเหมือนกัน แต่การจัดเรียงไม่เหมือนกัน ดังภาพที่ 2.26 และตารางที่ 2.4



ภาพที่ 2.26 แผนผังคอนเน็กเตอร์ของ RS-232

ตารางที่ 2.4 ผังคอนเน็กเตอร์ของ RS – 232

D-Type 25 Pin	D-Type 9 Pin	สัญลักษณ์	ชื่อสัญญาณ
Pin 2	Pin 3	TD	Transmit Data
Pin 3	Pin 2	RD	Receive Data
Pin 4	Pin 7	RTS	Request To Send
Pin 5	Pin 8	CTS	Clear To Send
Pin 6	Pin 6	DSR	Data Set Ready
Pin 7	Pin 5	SG	Signal Ground
Pin 8	Pin 1	CD	Carrier Detect
Pin 20	Pin 4	DTR	Data Terminal Ready

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Pin 22	Pin 9	RI	Ring Indicator

2.4.3 รายละเอียดของสัญญาณสำหรับรายละเอียดของสัญญาณนั้นประกอบไปด้วย

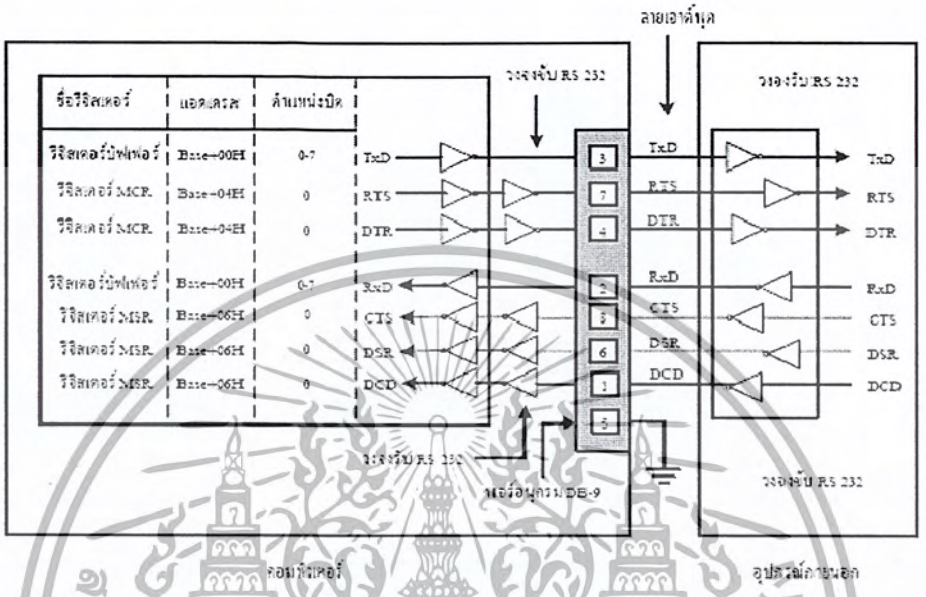
1. Transmit Data: TD ใช้สำหรับส่งข้อมูลอนุกรมออกจากคอมพิวเตอร์
2. Receive Data: RD ใช้สำหรับรับข้อมูลอนุกรมเข้ามายังคอมพิวเตอร์
3. Request To Send: RTS ใช้สำหรับส่งข้อมูลไปยังอุปกรณ์ปลายทางเพื่อร้องขอให้อุปกรณ์ปลายทางส่งข้อมูลกลับมา
4. Clear To Send: CTS ใช้สำหรับตรวจสอบว่าอุปกรณ์ที่เชื่อมต่อด้วยพร้อมที่จะรับข้อมูลหรือไม่ โดยจะคอยรับสัญญาณ RTS เมื่อทุกขาพร้อมก็จะทำการส่งข้อมูลออกทางขา TD
5. Data Set Ready: DSR ใช้สำหรับตรวจสอบการเชื่อมต่อกันระหว่างคอมพิวเตอร์กับอุปกรณ์ปลายทาง จะใช้คู่กับขา DTR
6. Signal Ground: SG เป็นกราวน์ของระบบ
7. Carrier Detect: CD ขานี้จะ Active เมื่อมีการส่งสัญญาณ Carrier จากโมเด็ม
8. Data Terminal Ready: DRT ใช้สำหรับบอกให้อุปกรณ์ปลายทางรับรู้ว่าต้องการติดต่อด้วยโดยขา DTR นี้ต้องเชื่อมต่อกับขา DSR ของอุปกรณ์ปลายทาง
9. Ring Indicator: RI ขานี้จะ Active เมื่อโมเด็มได้รับสัญญาณเรียกเข้าจากสายโทรศัพท์

2.4.4 ลักษณะสัญญาณอินพุตและเอาต์พุตของพอร์ต RS-232

สัญญาณเอาต์พุตที่จะใช้ควบคุม (RTS และ DTR) และจะมีสัญญาณแสดงสถานะทางอินพุต(CTS,DSR และ DCD) ของพอร์ตอนุกรม RS-232 จะถูกกลับสถานะภายในตัว UART ส่วนสัญญาณข้อมูลทั้งภาคส่ง และรับจะไม่ถูกกลับสถานะ UART จะให้ระดับสัญญาณที่เอาต์พุตออกมาเป็นแบบทีทีแอลเท่านั้น ดังนั้นเมื่อสัญญาณถูกส่งออกมาจาก UART จึงต้องส่งเข้าสู่วงจรขับเพื่อปรับระดับแรงดันให้ได้ระดับสัญญาณเป็นไปตามมาตรฐาน RS-232 ก่อนส่งออกไปจากคอมพิวเตอร์ สำหรับอุปกรณ์ต่อเชื่อมปลายทางก็จะต้องมีวงจรขับในลักษณะนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เช่นเดียวกันเพื่อให้ได้ระดับสัญญาณในระดับเดียวกัน แต่วงจรขับที่ใช้ทั้งภายในคอมพิวเตอร์และอุปกรณ์ต่อเชื่อมปลายทางนั้นจะถูกกลับสถานะดังแสดงเป็นบล็อกไดอะแกรมในภาพที่ 2.27



รูปที่ 2.27 ไดอะแกรมโครงสร้างทางฮาร์ดแวร์ของพอร์ตอนุกรม

แอดเดรสพื้นฐานของพอร์ตอนุกรม มี 4 ตำแหน่งดังนี้ คือ

COM1 : 3F8H

COM2 : 2F8H

COM3 : 3E8H

COM4 : 2E8H

เมื่อเริ่มเปิดเครื่องเพื่อใช้งานคอมพิวเตอร์ ไบออสภายในคอมพิวเตอร์ จะทำการตรวจสอบแอดเดรสของพอร์ตอนุกรมทั้งหมด ถ้าไบออสตรวจพบแอดเดรสของพอร์ตที่อนุกรม ไบออสจะนำแอดเดรสที่ตรวจพบ ไปเก็บไว้ในหน่วยความจำขนาด 2 ไบต์ สำหรับพอร์ตอนุกรม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

COM1 จะเก็บไว้แอดเดรส 0000:0400H และ0000:0401H ส่วนที่ตำแหน่งอื่นๆมีรายละเอียดดังตารางที่ 2.3

COM2 = 0000:0402H - 0000:0403H

COM3 = 0000:0404H - 0000:0405H

COM4 = 0000:0406H - 0000:0407H

นอกจากนี้ ที่หน่วยความจำแอดเดรส 0000:0411H ยังใช้สำหรับแสดงจำนวนของพอร์ตอนุกรมที่มีใช้อยู่ในคอมพิวเตอร์อีกด้วยโดยมีรายละเอียด ดังแสดงในตารางที่ 2.5

ตารางที่ 2.5 ข้อมูลในแอดเดรส 0000:041H ที่ใช้ในพอร์ตอนุกรม

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

2.4.5 อัตราเร็วในการรับส่งข้อมูลแบบอนุกรม

การที่อุปกรณ์ 2 อย่างจะติดต่อสื่อสารกันได้นั้น จะต้องทำงานด้วยอัตราเร็วเท่ากันซึ่งอัตราเร็วในการสื่อสารแบบอะซิงโครนัสคือ ค่าบอดเรต (Baud Rate) มีหน่วยเป็นบิตต่อวินาที ซึ่งอัตราเร็วในการสื่อสารแบบอนุกรมสำหรับมาตรฐานRS-232 นั้นมีใช้ดังนี้ 110 , 150 , 300 , 600 , 1200 , 2400 , 4800 , 9600 และ 1920

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.5 ความรู้เบื้องต้นเกี่ยวกับ I^2C

I^2C ย่อมาจาก Inter – IC Communication หมายถึง การติดต่อสื่อสารระหว่างไอซี

ตัวอย่างอุปกรณ์ที่สามารถใช้งานร่วมบนระบบบัสเช่น ไมโครคอนโทรลเลอร์ ไอซีขยายพอร์ต ไอซีแปลงADC/DAC ไอซี RTC ไอซีหน่วยความจำอีอีพรอม เป็นต้น

2.5.1 หลักการของระบบบัส I^2C

ระบบบัส I2C ประกอบด้วยสายสัญญาณสองเส้นคือ

1) SDA มาจาก Serial Data Line

2) SCL มาจาก Serial Clock Line

ภายในระบบบัส I2C ประกอบด้วยอุปกรณ์ที่ทำหน้าที่ต่างๆคือ แบ่งตามหน้าที่การจัดการข้อมูล

1) อุปกรณ์ที่เป็นผู้สร้างข้อมูลหรือส่งข้อมูล เรียกว่า ตัวส่ง

2) อุปกรณ์ที่เป็นผู้รับข้อมูล เรียกว่า ตัวรับ

แบ่งตามหน้าที่การจัดการระบบบัส

1) อุปกรณ์ที่ทำหน้าที่ควบคุมจังหวะการติดต่อ เรียกว่า มาสเตอร์

2) อุปกรณ์ที่ถูกควบคุมหรืออุปกรณ์ต่อพ่วง เรียกว่า สเลฟ

2.5.2 ภาวะที่เกิดขึ้นบนระบบบัส I^2C

1.บัสว่าง เป็นสภาวะที่มาสเตอร์ทำให้ SDA และ SCL เป็น“1” ทั้งคู่ เพื่อที่จะเริ่มต้นการสื่อสาร

2. สภาวะเริ่มต้น เกิดขึ้นเมื่อมาสเตอร์ทำให้ SDA เปลี่ยนจาก“1” เป็น “0” ก่อนแล้วจึงเปลี่ยน SCL จาก “1” เป็น“0”

3.สภาวะบิตข้อมูล เกิดขึ้นโดยตัวส่งทำให้ SDA เป็นบิตข้อมูล “0” หรือ “1” ก่อนแล้วมาสเตอร์ทำให้เกิดสัญญาณนาฬิกาขึ้นที่ SCL ตัวรับจึงรับข้อมูลในช่วงดังกล่าว

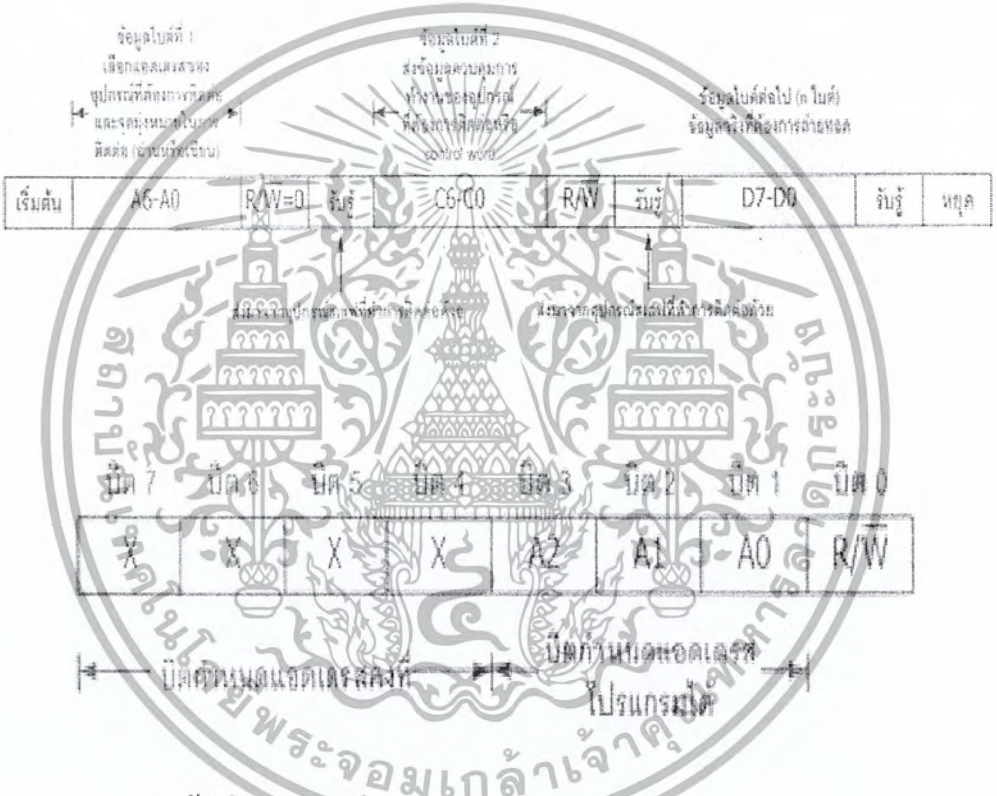
4.สภาวะบิตรับรู้ เกิดขึ้นโดยตัวส่งทำให้ SDA เป็น “1” ก่อนแล้วมาสเตอร์ทำให้เกิดสัญญาณนาฬิกาขึ้นที่ SCL เมื่อตัวรับรับรู้แล้วก็จะทำให้ SDA เป็น “0”

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

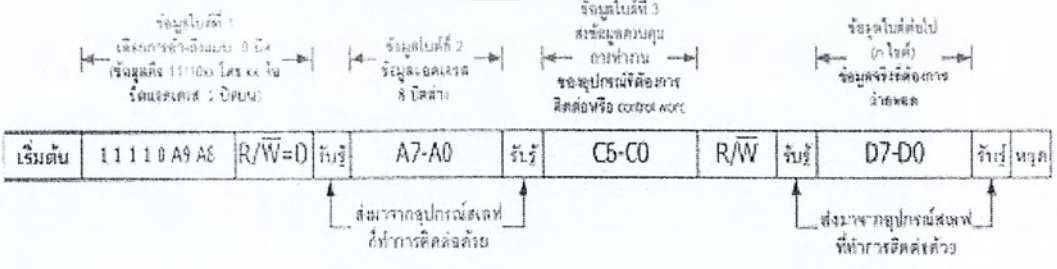
5.สภาวะหยุด เกิดขึ้นเมื่อมาสเตอร์ทำให้ SCL เปลี่ยนจาก“0” เป็น “1” ก่อน แล้วจึงเปลี่ยน SDA จาก “0” เป็น“1”

2.5.3 การทำงานบนบัส I^2C

การอ้างอิงแบบ 7 บิต

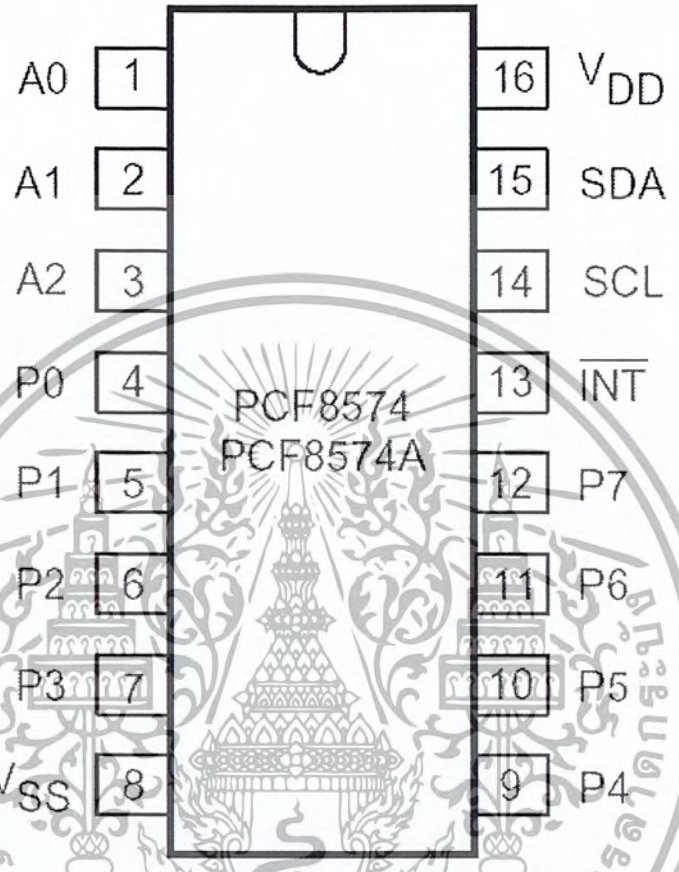


การอ้างอิงแบบ 10 บิต



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.5.4 การขยายพอร์ตด้วยไอซี PCF8574A



รูปที่ 2.28 ไอซี PCF8574A

- **A0,A1,A2** เป็นขาสัญญาณแอดเดรสใช้สำหรับกำหนดรหัสตำแหน่งของ PCF8574 ในระบบบัส ซึ่งสามารถทำการเชื่อมต่อ PCF8574 รวมกันในบัสเดียวกันได้ แต่ต้องกำหนดให้สภาวะลอจิกของขาสัญญาณทั้ง 3 เส้นนี้มีค่าต่างกัน

- **VSS**เป็นขาสัญญาณอ้างอิง หรือ GND

- **VDD** เป็นขาสัญญาณไฟเลี้ยง

- **INT** เป็นขาสัญญาณ Interrupt เมื่อทำงานจะให้สภาวะลอจิกเป็น “0”

- **SDA**เป็นขาสัญญาณข้อมูล (Data) แบบ 2 ทิศทาง ของการสื่อสาร I2C โดยขานี้จะเป็น Input เมื่อไมโครคอนโทรลเลอร์เขียนข้อมูลให้ PCF8574 และขานี้จะเป็นOutput เมื่อไมโครคอนโทรลเลอร์อ่านข้อมูลจาก PCF8574

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- **SCL** เป็นขาสัญญาณนาฬิกา (Clock) ของการสื่อสาร I2C โดยขานี้จะมีสภาวะเป็น Input ตลอด

- **P0-P7** เป็นขาสัญญาณ Port I/O โดยสามารถทำหน้าที่เป็นได้ทั้ง Input และ Output

INTERFACE DEFINITION TABLE

BYTE	BIT							
	7 (MSB)	6	5	4	3	2	1	0 (LSB)
I^2C slave address	L	H	H	H	A2	A1	A0	R/$\overline{W}$
I/O data bus	P7	P6	P5	P4	P3	P2	P1	P0

ขั้นตอนการเขียนข้อมูลควบคุม PCF8574A เป็น OUTPUT

1. ส่งสภาวะเริ่มต้น (Start Condition) ไปยังบัส เพื่อเริ่มต้นการสื่อสาร
2. ส่งรหัส Control Byte สำหรับการเขียนซึ่งก็คือ “01110000”
3. ส่งค่าOUTPUT ที่ต้องการเขียนข้อมูลไปให้ PCF8574A
4. ส่งสภาวะสิ้นสุด(Stop Condition)ไปยังบัสเพื่อจบการสื่อสาร

ขั้นตอนการอ่านข้อมูลPCF8574A เป็น INPUT

1. ส่งสภาวะเริ่มต้น (Start Condition) ไปยังบัส เพื่อเริ่มต้นการสื่อสาร
2. ส่งรหัสControl Byte สำหรับการเขียนซึ่งก็คือ “01110000”
3. ส่งค่าOUTPUT ที่ต้องการเขียนข้อมูลไปให้ PCF8574A เป็นค่า 0xFF เพื่อ Clear ค่า PCF8574A
4. ส่งสภาวะสิ้นสุด (Stop Condition) ไปยังบัสเพื่อจบการสื่อสาร
   - ส่งสภาวะเริ่มต้น (Start Condition) ไปยังบัส เพื่อเริ่มต้นการสื่อสาร
   - ส่งรหัส Control Byte สำหรับการอ่าน ซึ่งก็คือ “01110001”
   - อ่านค่า INPUT จาก PCF8574A
   - ส่งสภาวะสิ้นสุด (Stop Condition) ไปยังบัสเพื่อจบการสื่อสาร

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.6 โปรแกรมภาษาซีชาร์ป

2.6.1 พื้นฐานโปรแกรมภาษาซีชาร์ป

เป็นภาษาโปรแกรมเชิงวัตถุมีการทำงานบนดอตเน็ตเฟรมเวิร์ก พัฒนาโดยบริษัทไมโครซอฟท์ และมี Anders Hejlsberg เป็นหัวหน้าโครงการ โดยมีรากฐานมาจากภาษาซีพลัสพลัสและภาษาอื่นๆ (โดยเฉพาะภาษาเดลไฟและจาวา) โดยปัจจุบันภาษาซีชาร์ปเป็นภาษามาตรฐานรองรับโดย ECMA และ ISO โดยดอตเน็ตเฟรมเวิร์กเป็นแพลตฟอร์มใหม่ และเปลี่ยนแปลงไปอย่างสิ้นเชิง ที่ถูกสร้างขึ้นมาโดย บริษัท Microsoft เพื่อใช้สำหรับ การพัฒนา Application ดอตเน็ตเฟรมเวิร์ก ถูกออกแบบมาเพื่อให้สามารถ ถูกใช้จากภาษาใดๆ ก็ได้ รวมถึง ซีชาร์ป ด้วย รวมถึง ภาษา ภาษาซีพลัสพลัส, วิชวลเบสิก, เดลไฟ และอื่นๆ เพื่อให้สิ่งเหล่านี้เป็นไปได้ จึงเกิดภาษาเหล่านี้ ขึ้นมาในรูปของ Version เฉพาะสำหรับ ดอตเน็ตอีกด้วย ได้แก่ภาษา Managed ภาษาซีพลัสพลัส, วิชวลเบสิก, ซีชาร์ป, เดลไฟ8 เป็นต้นและมีอีกมากว่านี้ที่กำลังพัฒนา และปล่อยออกสู่ท้องตลาดอยู่ตลอดเวลาไม่ใช่เพียงแค่ภาษาทั้งหมดเหล่านี้ จะมีการเข้าถึง ดอตเน็ตเฟรมเวิร์กเท่านั้น แต่มันยังสามารถสื่อสาร กับภาษาอื่นๆ ได้อีกด้วย

ดอตเน็ตเฟรมเวิร์กพื้นฐานประกอบขึ้นด้วย Libary ของ Source Code ขนาดมหึมา ซึ่งเราเรียกใช้จากภาษา Client ของเรา เช่น ซีชาร์ป, ภาษาซีพลัสพลัส, ดอตเน็ต โดยการใช้เทคนิคเชิงวัตถุ (OOP) Libary ที่ว่านี้ ถูกแบ่งกลุ่มออกเป็น Modul ต่างๆ ดังนั้นเราจึงใช้ส่วนของมัน ตามผลลัพธ์ที่เราต้องการได้ เช่น Windows Application เป็นต้น จุดมุ่งหมายในที่นี้ก็คือระบบปฏิบัติการที่แตกต่างกันอาจจะสนับสนุน Modul เหล่านี้บาง Modul หรือทั้งหมดขึ้นอยู่กับคุณลักษณะของมัน เช่น PDA จะรวมเอาการสนับสนุน Function หน้าที่ที่เป็นแก่นของ ดอตเน็ตทั้งหมดเป็นต้น

ส่วน Libary ดอตเน็ตเฟรมเวิร์กกำหนดชนิด ข้อมูลพื้นฐานบางอย่างเอาไว้ ชนิดข้อมูลเป็นตัวแทนของข้อมูล และการแบ่งกฎเกณฑ์ทั้งหลายเหล่านี้ ที่จะส่งเสริมความสามารถ ในการสัมพันธ์ระหว่างภาษา โดยใช้ดอตเน็ตเฟรมเวิร์กสิ่งนี้ถูกเรียกว่า Common Type System (CTS) เช่นเดียวกับการจัดให้มี Libary .Net Common Language Runtime (CLR) ซึ่งรับผิดชอบในการจัดการ กับ ระบบ)ปฏิบัติการ ของ Application ทั้งหมดที่ถูกพัฒนาขึ้นมาด้วย Libary ดอตเน็ตเฟรมเวิร์ก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Borland ซีซาร์ป Builder เป็นภาษา ซีซาร์ป ที่ถูกพัฒนา โดย Borland สำหรับ ซีซาร์ป คือหนึ่งในภาษาที่สามารถนำมาสร้าง Application ที่ Run อยู่ใน ดอตเน็ต CLR ได้มันเป็นอีก วิวัฒนาการหนึ่งของภาษาซี และ ภาษาซีพลัสพลัสถูกสร้างขึ้นมาให้ทำงานกับ Platform ดอตเน็ตในฐานะที่มันถูกพัฒนาอกมาเป็นอันดับล่าสุด ภาษาซีซาร์ป จึงถูกออกแบบมาอย่างมีพื้นหลังมาจากซี และภาษาซีพลัสพลัสให้รับเอา ฟิวเจอร์ ที่ดีที่สุดจากภาษาอื่นๆ และลบล้างปัญหาที่มีอยู่ในภาษาเหล้านั้นให้หมดไป ความคล้ายคลึงกัน ของ Code ของ ซีซาร์ป กับ ภาษาซีพลัสพลัส และ Java นั้นอาจดูค่อนข้างสับสน และมันก็คล้ายกับการเขียนภาษาอังกฤษ น้อยกว่าภาษาอื่นๆ มาก แต่อย่างไรก็ตาม เราจะพบว่า ตัวเราเองอยู่ในโลกของการโปรแกรม ซีซาร์ป ที่มี Style อ่อนไหวนุ่มนวล และสามารถอ่าน Code ได้ง่ายและไม่สับสนมากนัก ซีซาร์ปนั้นไม่มีการแจ้งเตือน เกี่ยวกับช่องว่าง ที่อยู่ใน Code ไม่ว่าคุณจะใส่ช่องว่าง จำนวนมาก อักขระ Carriage return หรือ Tap เป็นอักขระที่รู้จักกันในชื่อว่า White space นั่นหมายความว่า เรามีอิสระในการ ที่จะจัดรูปแบบ Source Code ของเราได้ ถึงแม้ว่าการทำตามกฎที่แน่นอน สามารถช่วยให้เราทำสิ่งต่างๆ ให้อ่านได้ง่ายขึ้นก็ตาม Code ซีซาร์ปนั้นสร้างขึ้นจาก Statement ชุดหนึ่ง แต่ละ Statement จะจบด้วย ; เนื่องจาก White space นั้นถูกมองข้ามไป เราจึงสามารถมีหลายๆ Statement ในบรรทัดเดียวกันได้ แต่เพื่อความง่ายในการอ่าน มันจะมีประโยชน์ ถ้าเพิ่ม Carriage return เข้าไปหลัง ; ดังนั้นเราก็จะไม่มีหลายๆ Statement ในบรรทัดเดียวกันอย่างไรก็ตามมัน เป็นสิ่งที่ยอมรับได้และเป็นสิ่งปกติ ซีซาร์ปเป็นภาษา โครงสร้างบล็อก (Block structured) หมายความว่า ทุกๆ Statement ถือเป็นส่วนหนึ่งของ Block Code และ Block เหล่านี้ถูกจำกัดด้วย เครื่องหมาย { } เหมือนกับ ภาษาซีพลัสพลัสทุกประการ รูปแบบจึงเหมือนกับ ภาษาซีพลัสพลัสและมีโครงสร้างเหมือน ภาษาซีพลัสพลัสดังนั้นหากเราได้ศึกษาโครงสร้าง ของภาษาซีพลัสพลัสแล้ว ก็จะเข้าใจหากยังไม่เข้าให้ไปศึกษาโครงสร้างในภาษาซีพลัสพลัส

2.6.2 ตัวแปร

สำหรับตัวแปรใน ซีซาร์ปจะคล้ายกับภาษาซีพลัสพลัสมาก ดังนั้นสำหรับผู้ที่ได้ศึกษาเรื่องของภาษาซีพลัสพลัสมาแล้วก็จะเป็นไปได้ง่าย ในการศึกษาเรื่องนี้ ในการที่จะใช้ตัวแปรนั้น เราจะต้องประกาศตัวแปรก่อน นั่นคือจะต้องกำหนดชื่อ และชนิดของตัวแปรนั้น ทันทีที่เราได้ประกาศชนิดของตัวแปรเสร็จสิ้นเราจะ ใช้มันเป็นตัวจัดเก็บข้อมูล ตามชนิดที่เราได้ประกาศ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ไว้ได้ซึ่งมีรูปแบบการประกาศดังนี้Typename

ถ้าเราพยายามใช้ตัวแปรที่ยังไม่มีการประกาศเอาไว้ก่อนแล้ว Code ของเราจะไม่ถูก Compile แต่ Compiler จะบอกถึงสาเหตุที่แท้จริงที่เกิดขึ้น ดังนั้นจึงไม่ใช่ข้อผิดพลาดที่รายแรงเท่าใดนัก นอกจากนี้ การพยายามที่จะใช้ตัวแปรที่ไม่ได้กำหนดค่าเริ่มต้น ให้กับมัน มักจะเกิดข้อผิดพลาด แต่ก็อีกนั้นแหละตัว Compiler จะตรวจสอบพบมันได้ ตัวแปรอย่างง่ายๆ ที่จะแนะนำนี้ คือชนิดของตัวแปร เช่น ตัวเลข และ Boolean ที่ใช้เป็นรากฐานในการสร้าง Block สำหรับ Application ของเรา และสำหรับสิ่งอื่นๆ คือ ตัวแปรที่สลับซับซ้อน ตัวแปรอย่างง่าย ส่วนใหญ่แล้ว คือ ตัวเลข ซึ่งเมื่อเริ่มต้นนั้น มันดูเหมือนแปลกตาอยู่บ้าง แน่นอนว่าจำเป็นไหม ที่เราต้องการแค่ชนิดเดียวเท่านั้นที่ใช้เก็บตัวเลข

เหตุผลที่ตัวแปรชนิดตัวเลข มีอยู่มากมายนั้น เป็นไปตามกลไกของการจัดเก็บตัวเลข ในลักษณะอนุกรม ของ 0 และ 1 ในหน่วยความจำของคอมพิวเตอร์ สำหรับค่าจำนวนเต็ม เราเพียงแค่นำจำนวน Bit แล้วแทนที่จำนวนของเราในแบบ Binary ตัวแปรที่จัดเก็บ N Bit จะอนุญาตให้เราจัดเก็บจำนวนใดๆ ระหว่าง 0 ถึง 2n - 1 ได้ ถ้าจำนวนใดๆที่มากกว่าค่านี้ จะใหญ่เกินไป ที่จะใส่ไปในตัวแปรได้ ถ้าเราต้องการให้สามารถเก็บจำนวนได้มากขึ้น เราต้องใช้ Bit มากขึ้นด้วย ดังนั้น เราจึงจำเป็นจะต้องกำหนดตัวแปรให้เหมาะสม กับขนาดของข้อมูลที่เราจะใช้ ดังตารางต่อไปนี้ ซึ่งเป็นตารางตัวแปรชนิดตัวเลขจำนวนเต็ม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 3

การออกแบบและการจัดทำปริญญานิพนธ์

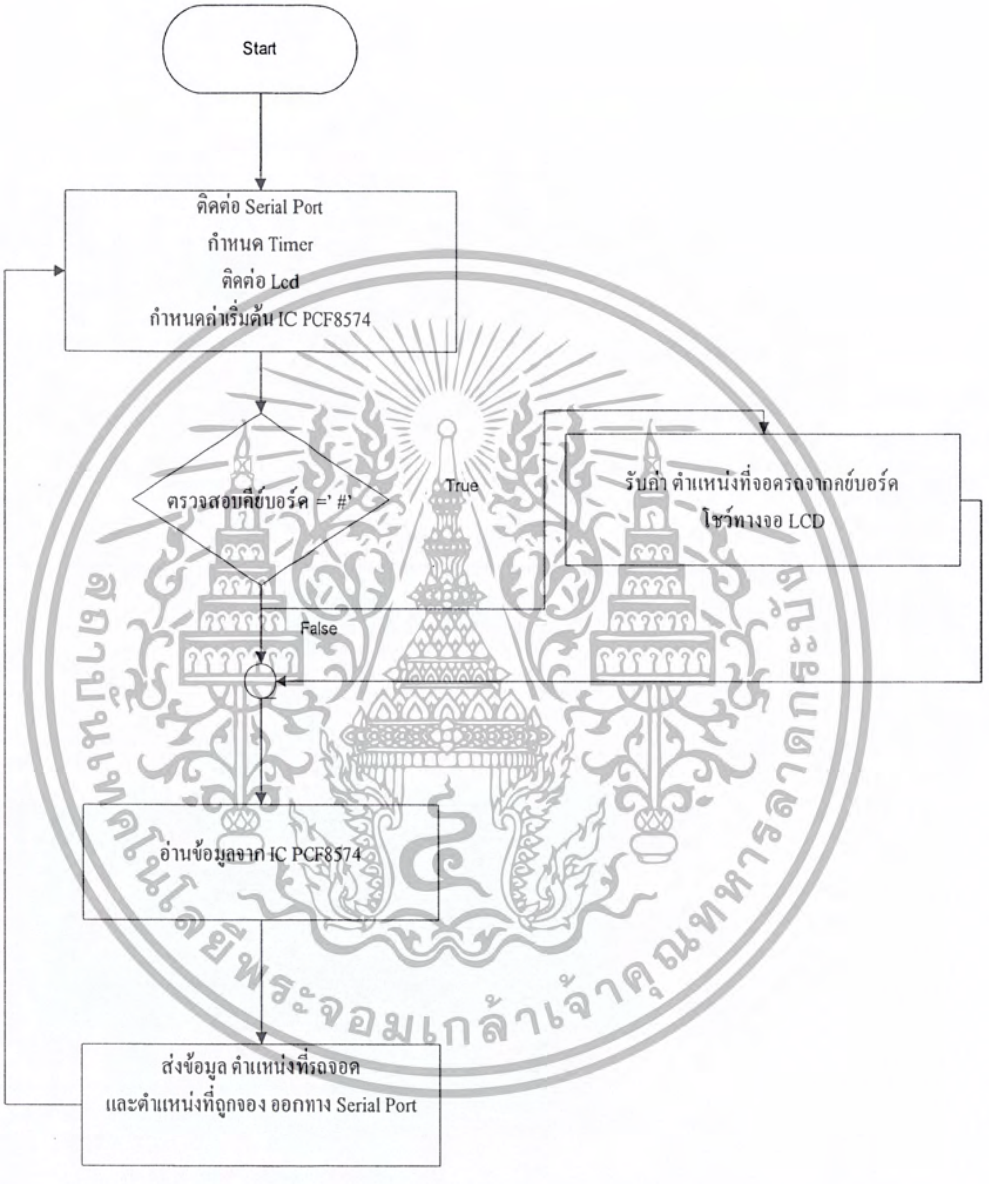
ทฤษฎีของระบบที่ทำการศึกษามา ซึ่งได้นำมาประกอบ การออกแบบและสร้างเป็นโครงงานปริญญานิพนธ์ซึ่งสามารถแยกออกเป็นส่วนประกอบหลักได้ดังนี้

รูปที่ 3.1ก โฟลว์ชาร์ตขั้นตอนการดำเนินงานของโครงการ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 3.1ข. โฟลว์ชาร์ตขั้นตอนการดำเนินงานของโครงงาน

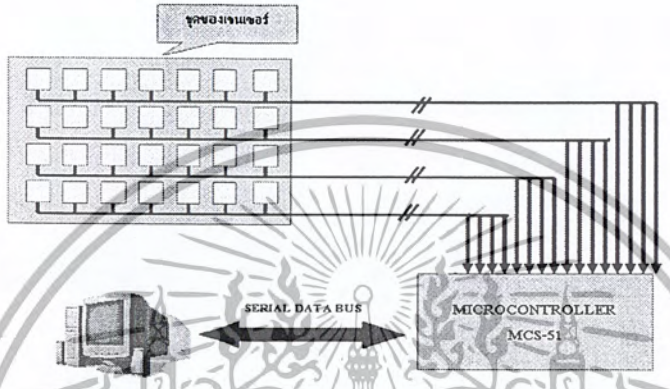


รูปที่ 3.2โฟลว์ชาร์ตขั้นตอนการดำเนินงานไมโครคอนโทรลเลอร์ MCS-51

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.1 โครงสร้างของระบบในส่วนต่างๆของโครงงาน

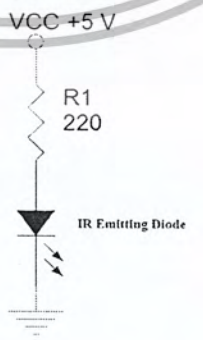
1. ชุดวงจรเซ็นเซอร์
2. ไมโครคอนโทรลเลอร์ MCS-51
3. แบบจำลองอาคารจอดรถ



รูปที่ 3.3 บล็อกไดอะแกรมของแบบจำลองระบบบอกตำแหน่งพื้นที่ว่างในอาคารจอดรถ

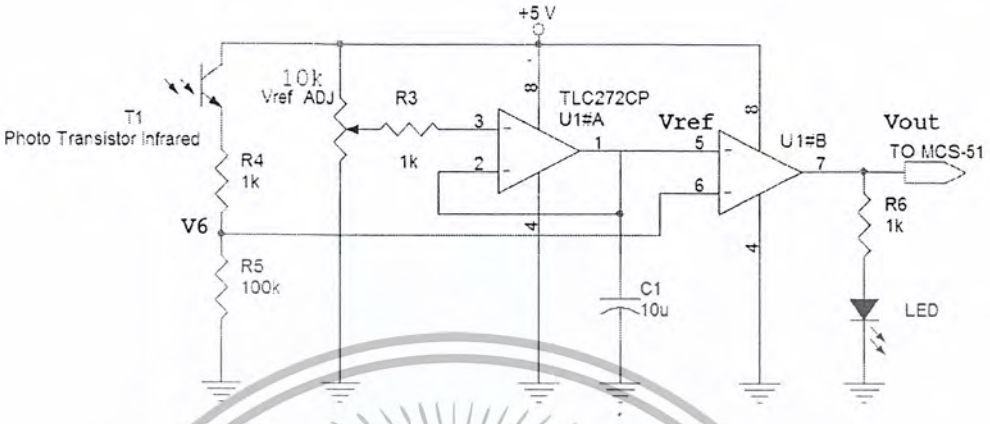
3.1.1 วงจรตรวจเซ็นเซอร์

ศึกษาและออกแบบสร้างวงจรตรวจจับ โดยให้สามารถใช้ได้ในอาคารจำลองโดยจะมีระยะห่างกันประมาณ 5-10 เซนติเมตรวงจรตรวจจับด้วยแสงอินฟราเรดจะใช้โฟโต้ไดโอดเป็นตัวตรวจจับโดยติดตั้งตัวตรวจจับนี้ไว้ตรงตำแหน่งรถทุกตำแหน่ง เมื่อมีรถเข้ามาจอดจะตัดลำแสงอินฟราเรดระหว่างตัวส่งและตัวรับทำให้ได้สัญญาณลอจิกมีสถานะเป็น “1” แต่ถ้าไม่มีรถเข้ามาจอดก็จะได้สัญญาณที่มีสถานะเป็นลอจิก “0”



รูปที่ 3.4 วงจรภาคส่งแสงอินฟราเรด

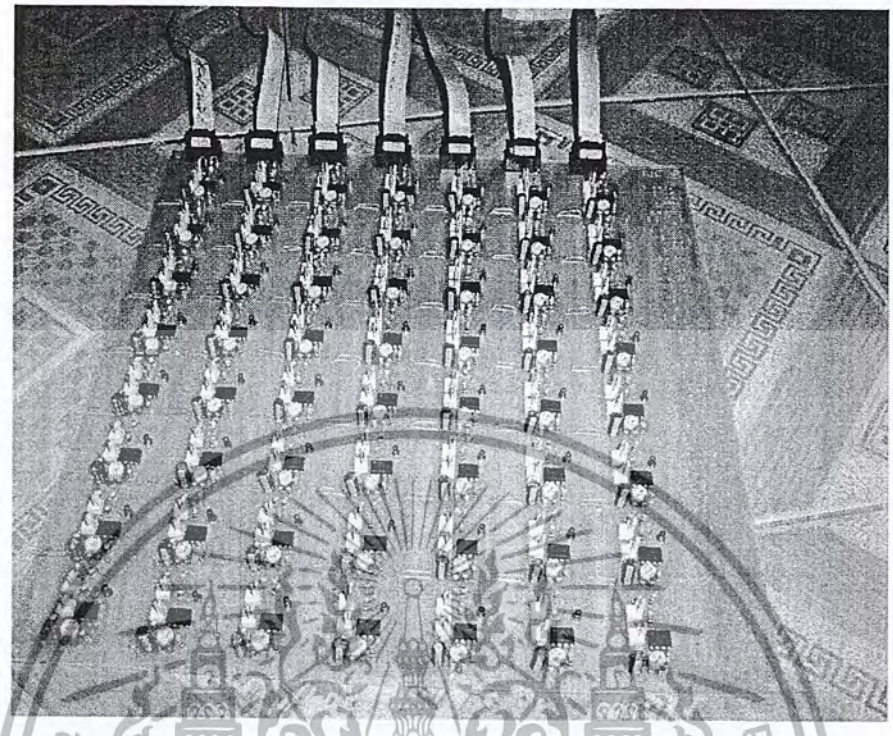
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



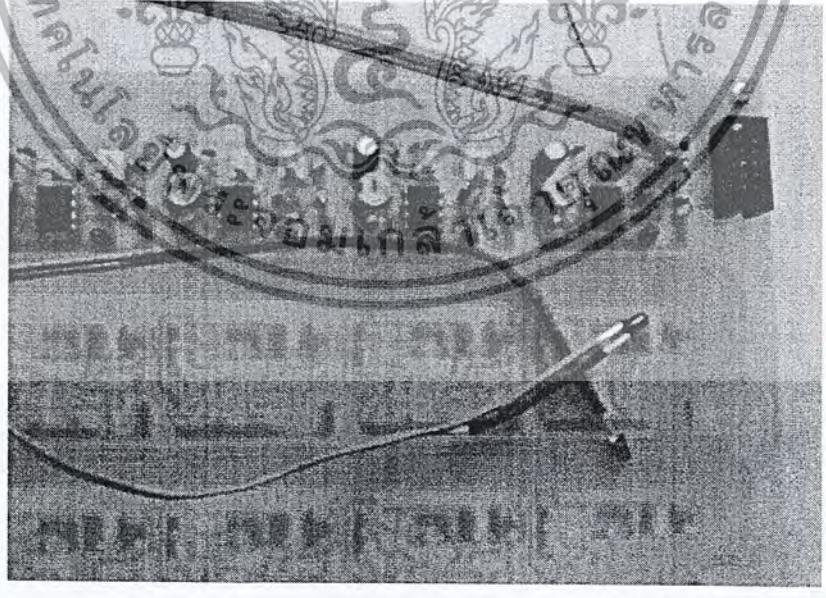
รูปที่ 3.5 วงจรภาครับของเซนเซอร์อินฟราเรด

รูปที่ 3.6 ลายวงจรด้านล่างวงจรตรวจจับรวม 56 ตัว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.7 แผงวงจรตรวจจับโดยใช้แสงอินฟราเรดทั้งหมด



รูปที่ 3.8 แผงวงจรตรวจจับโดยใช้แสงอินฟราเรดที่ใช้งานจริง

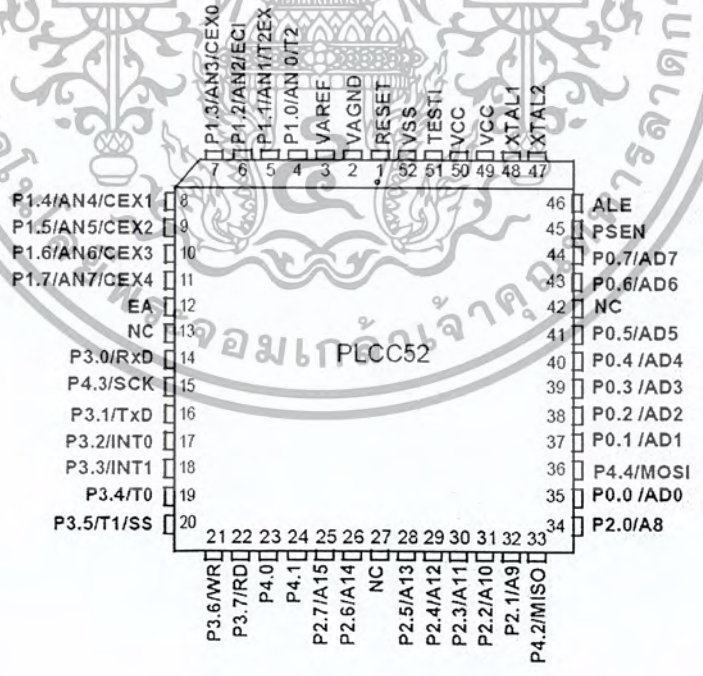
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.1.2 ไมโครคอนโทรลเลอร์ MCS-51

ไมโครคอนโทรลเลอร์ที่นำมาใช้งานเป็นบอร์ดสำเร็จรูปของ MCS-51 ซึ่งมีขนาดของหน่วยความจำโปรแกรมภายใน 128 Byte นำมาเพื่อใช้ในการรับและ ส่งข้อมูลจากอุปกรณ์ตรวจจับให้กับ เครื่องคอมพิวเตอร์ วงจรของบอร์ดแสดงดังภาพที่ 3.9 โดยการทำงานจะให้ไมโครคอนโทรลเลอร์ทำการเช็คสถานะของเซนเซอร์ทั้งหมด และส่งข้อมูลสถานะทั้งหมดไปยังคอมพิวเตอร์โดยการเชื่อมต่อแบบอนุกรม ผ่าน Serial Port และจะทำการรอรับการประมวลผลจำนวนของจำนวนข้อมูลที่ได้จากคอมพิวเตอร์ และ ส่งลอจิกออกมาเป็นเอาท์พุตเพื่อใช้เป็นสัญญาณไฟแสดงผล โดยขั้นตอนการทำงานของโปรแกรมควบคุมไมโครคอนโทรลเลอร์

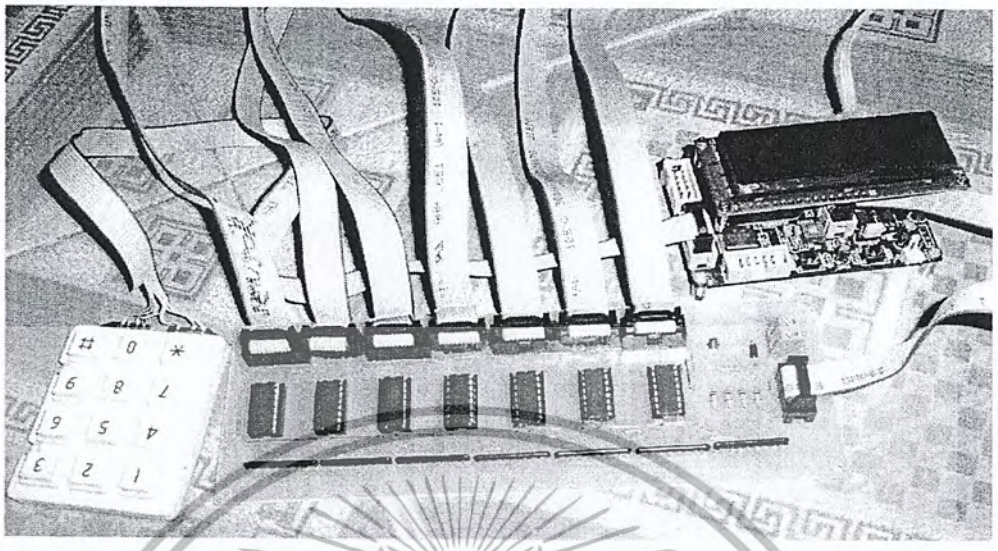
3.2 การจัดขาต่าง ๆ ของ MCS-51

ไอซี ไมโครคอนโทรลเลอร์ 8051 โครงสร้าง ไอซี มีขาทั้งหมด 55 ขา โดยขาต่างๆ เป็นขาพอร์ทอินพุท, ขาพอร์ทเอาท์พุท, ขาสัญญาณควบคุม, ขาตำแหน่งหน่วยความจำข้อมูลดังรูปที่ 3.9



รูปที่ 3.9 ขาต่างๆของ AT89C51AC3

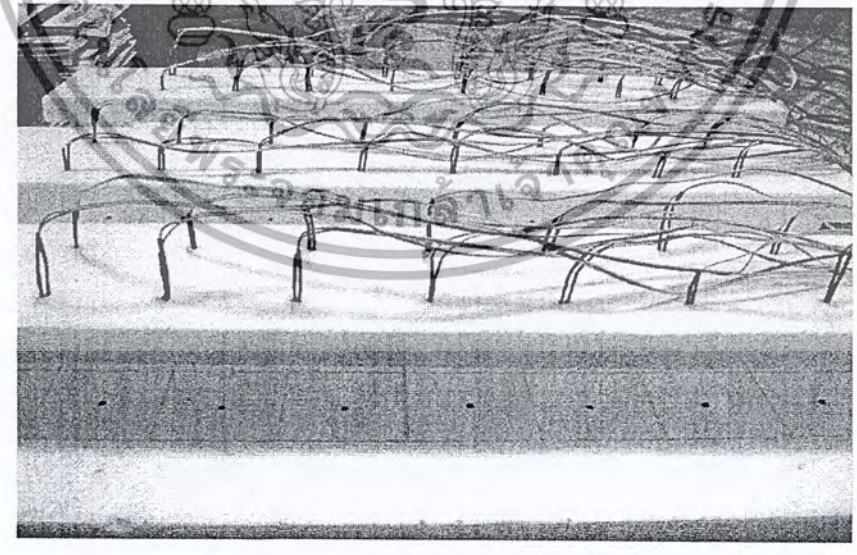
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.10 บอร์ดไมโครคอนโทรลเลอร์ MCS-51

3.3 ลานจอดรถจำลอง

ในโครงงานนี้ได้มีการจำลองอาคารหรือลานจอดรถไว้ที่ 56 คัน เพื่อเป็นการทดลอง และลานจอดรถที่ได้จำลองไว้มีลักษณะดังในรูปที่ 3.11



รูปที่ 3.11 ลานจอดรถจำลอง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โดยในโครงงานนี้เราได้กำหนดสถานะของช่องจอดรถในแต่ละช่องให้เหมือนกัน โดยกำหนดสถานะของช่องจอดรถตามสีต่างๆ ดังนี้

สีแดง แสดงถึง ช่องจอดนั้นไม่ว่าง

สีเขียว แสดงถึง ช่องจอดนั้นว่าง พร้อมที่จะให้การจอง

สีเหลือง แสดงถึง ช่องจอดนั้นได้มีการจองแล้ว และกำลังจะมีรถเข้าไปจอด

และในที่นี้ได้มีการออกแบบโฟว์ชาร์ตในกรณีการเข้าจอดรถในลานจอดรถรถจำลองนี้ ตามรูปดังต่อไปนี้

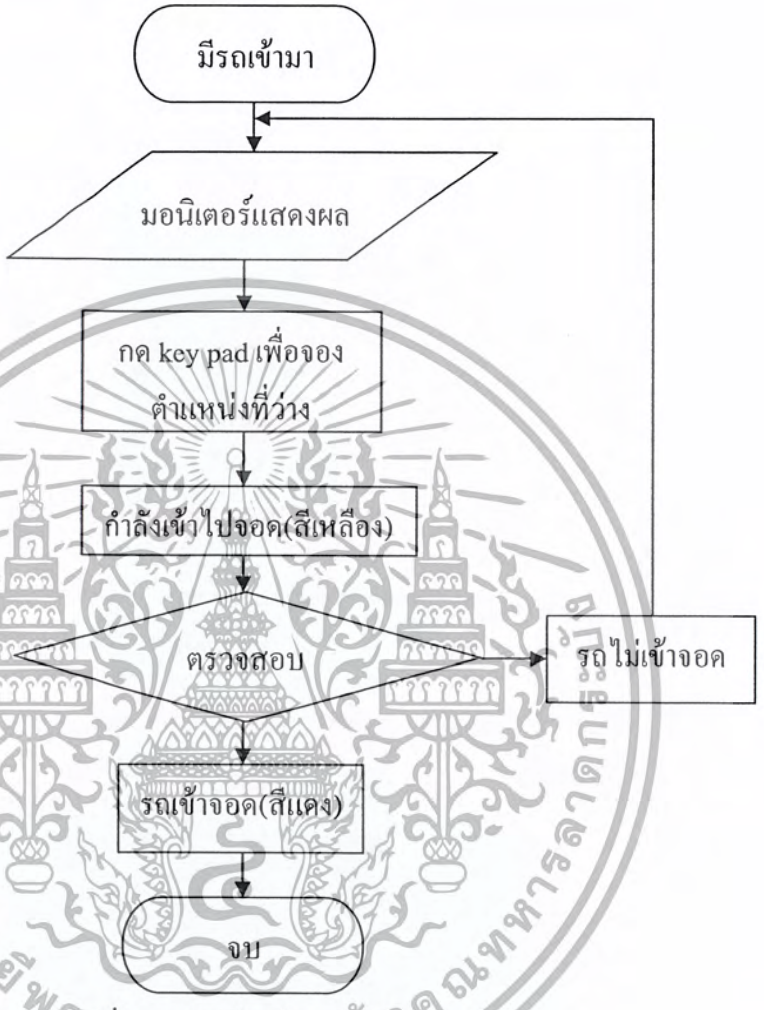
3.3.1ในกรณีสีแดง

รูปที่ 3.12 โฟว์ชาร์ตกรณีไฟสีแดง

ในกรณีทีรถเข้ามาเจอสถานะของช่องจอดรถเป็นสีแดงหมายถึง ช่องจอดรถนั้นมีรถจอดอยู่แล้ว ซึ่งรถที่เข้ามาจะต้องหาช่องจอดรถที่ว่างหรือช่องที่เป็นสีเขียว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.3.2 ในกรณีสีเขียว

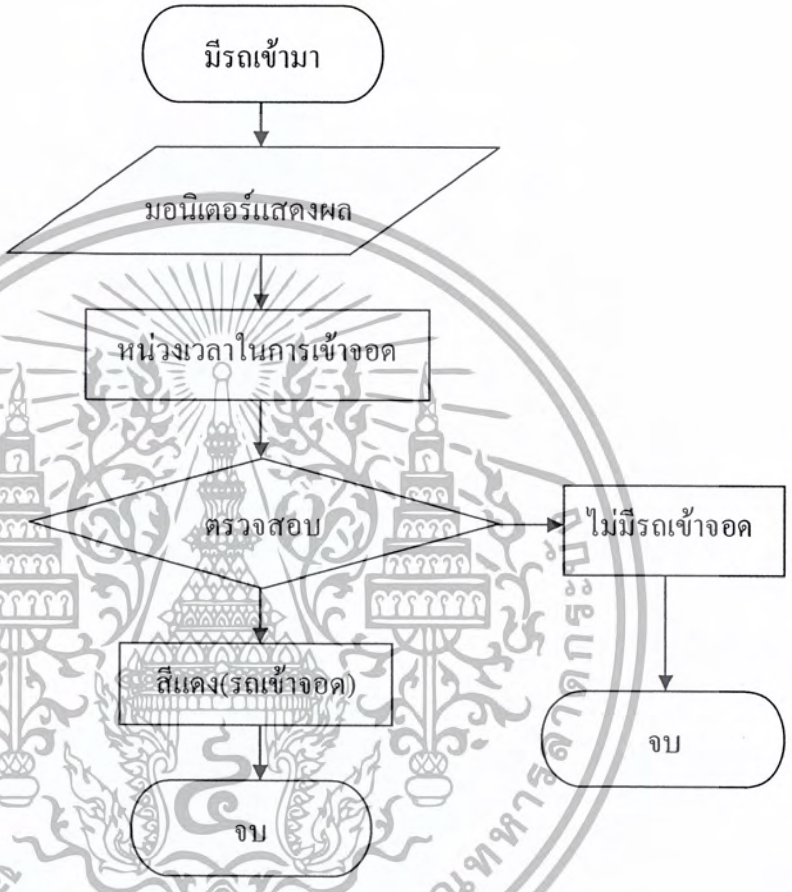


รูปที่ 3.13 โฟว์ชาร์ตกรณีไฟสีเขียว

ในกรณีนี้รถที่วิ่งเข้ามาจอดจะเห็นสถานะของช่องจอดรถนั้นๆ ว่าว่างซึ่งสามารถเข้าไปกดจองช่องจอดรถได้ และต้องกดช่องที่เป็นสีเขียวเท่านั้น เมื่อเข้าไปกดจองแล้วสถานะของช่องจอดรถนั้นจะแสดงสถานะใหม่เป็นสีแสด จะแสดงถึงช่องจอดนั้นมีเจ้าของแล้วและกำลังจะนำรถเข้ามาจอด เมื่อมีรถเข้ามาจอดแล้วสถานะของช่องจอดรถจะเปลี่ยนเป็นสีแดงหมายถึงมีรถเข้ามาจอดแล้ว แล้วเมื่อรถออกจากช่องจอดรถสถานะก็จะเป็นสีเขียว หมายถึงช่องจอดนี้ว่างแล้วพร้อมที่จะให้จองต่อไป ซึ่งตามความเป็นระเบียบและมีวินัยผู้ที่กดจองช่องนั้นแล้วต้องเข้าไปจอดให้ตรงตามช่องจอดรถที่จองไว้ และผู้ที่ไม่ได้กดจองก็ไม่ควรไปจอดช่องที่มีคนจองไว้แล้ว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.3.3 ในกรณีสีเหลือง



รูปที่ 3.14 โฟว์ชาร์ตกรณีไฟสีเหลือง

และในกรณีรถที่เข้ามาเจอสถานะของช่องจอดรถเป็นสีเหลืองหมายถึง ช่องจอดรถแถวนั้นทั้งแถวมีการจองหมดแล้ว ซึ่งถ้าไม่ใช่ผู้ที่เข้ามาจองไว้จะต้องหาช่องจอดรถแถวอื่นที่เป็นสีเขียวเท่านั้นและต้องกดจองไว้ก่อน เพื่อที่จะนำรถเข้าไปจอดให้ถูกช่อง ส่วนคนที่อยู่ในสถานะที่เข้ามาจองทั้งแถวจะต้องนำรถมาจอดไว้ตามเวลาที่หน่วงไว้ เพื่อไม่เป็นการสละสิทธิ์ในการเข้าจอด โดยสถานะสีเหลืองนี้จะมีการหน่วงเวลาไว้ตามความเหมาะสม ถ้าเกินเวลาไปช่องจอดที่ไม่มีรถเข้ามาจะเปลี่ยนสถานะเป็นสีเขียว และในกรณีที่มีการจองทั้งแถวอาจเป็นการให้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ช่องจอดทั้งแถวเป็นของกลุ่มบุคคลไปจำเป็นต้องใช้ช่องจอดรถจำนวนมากเพื่อมาทำธุระในอาคารนี้ เช่น การมาประชุมที่โรงแรม หรือการมาทำภารกิจ ณ อาคารที่มีอุปกรณ์ตัวนี้ติดตั้งอยู่เป็นต้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 4

ผลการดำเนินงาน

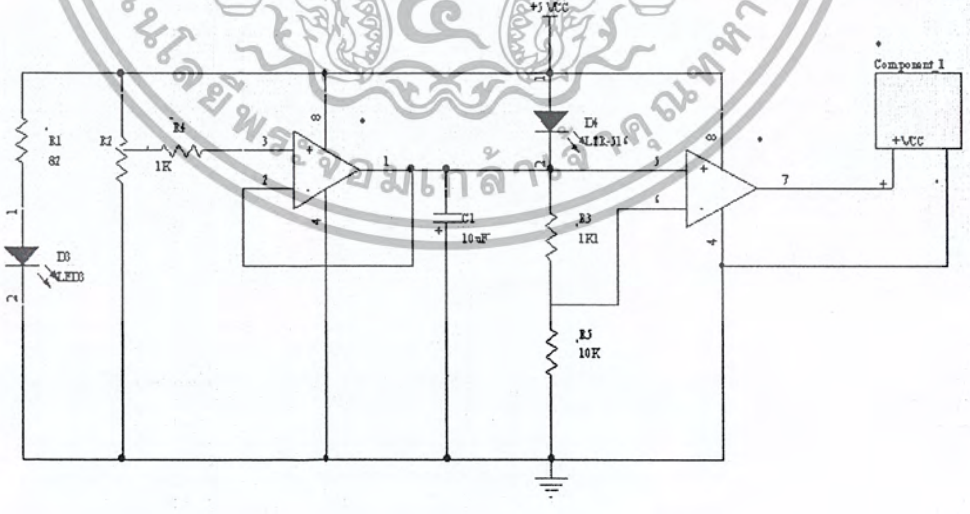
ผลที่ได้จากการทำโครงงานนี้ได้แบบจำลองระบบบอกตำแหน่งพื้นที่ว่าง ในอาคารจอดรถอัตโนมัติโดยแสดงผลไปยังจอคอมพิวเตอร์และสามารถแสดงตำแหน่งที่ว่างของที่จอดรถและแสดงตำแหน่งในลานจอดรถว่าตำแหน่งใดที่ว่างและสามารถเข้าไปจอดได้ หรือตำแหน่งใดที่ไม่ว่างไม่สามารถเข้าจอดได้แล้วเรายังสามารถจองตำแหน่งลานจอดรถล่วงหน้าได้ทั้งแถวอีกด้วย

4.1 ผลการทดลองวงจรตรวจจับ

1.ทำการทดสอบวงจรเพื่อทดสอบว่าสามารถใช้งานได้จริงและไม่ผิดพลาด

2.ทำการสร้างวงจรตรวจสอบสัญญาณดังรูป 4.1 และจ่ายไฟให้กับวงจร

3.ทำการปรับระยะตัวรับแสงและส่งแสงให้ตรงกัน



รูปที่ 4.1 วงจรตรวจจับสัญญาณ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.2 ทดสอบการทำงานของวงจร

ตารางที่ 4.1 การทดสอบการวัดค่าของวงจรเซ็นเซอร์

วงจรที่	Voltage Reference (V)	วัดค่า V(6)		วัดค่า V(out)	
		กรณีมีรถ (V)	กรณีไม่มีรถ(V)	กรณีมีรถ (V)	กรณีไม่มีรถ(V)
1	2.5	0.5	5	3.6	0
2	2.5	0.5	5	3.6	0
3	2.5	0.7	5	3.6	0
4	2.5	0.4	4.9	3.6	0
5	2.5	0.3	5	3.6	0
6	2.5	0.4	5	3.6	0
7	2.5	0.6	5	3.6	0
8	2.5	0.8	5	3.6	0
9	2.5	0.4	4.9	3.6	0
10	2.5	0.5	4.9	3.6	0
11	2.5	0.6	5	3.6	0
12	2.5	0.5	5	3.6	0
13	2.5	0.3	5	3.6	0
14	2.5	0.4	4.9	3.6	0
15	2.5	0.2	5	3.6	0
16	2.5	0.5	5	3.6	0
17	2.5	0.5	5	3.6	0
18	2.5	0.7	4.9	3.6	0
19	2.5	0.7	5	3.6	0
20	2.5	0.6	5	3.6	0
21	2.5	0.4	5	3.6	0

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

วงจรที่	Voltage Reference (V)	วัดค่า V(6)		วัดค่า V(out)	
		กรณีมีรถ (V)	กรณีไม่มีรถ(V)	กรณีมีรถ (V)	กรณีไม่มีรถ(V)
22	2.5	0.5	5	3.6	0
23	2.5	0.6	5	3.6	0
24	2.5	0.8	5	3.6	0
25	2.5	0.4	5	3.6	0
26	2.5	0.5	4.9	3.6	0
27	2.5	0.4	4.9	3.6	0
28	2.5	0.5	5	3.6	0
29	2.5	0.6	4.9	3.6	0
30	2.5	0.5	5	3.6	0
31	2.5	0.7	5	3.6	0
32	2.5	0.4	5	3.6	0
33	2.5	0.5	5	3.6	0
34	2.5	0.5	5	3.6	0
35	2.5	0.4	5	3.6	0
36	2.5	0.3	5	3.6	0
37	2.5	0.6	5	3.6	0
38	2.5	0.4	5	3.6	0
39	2.5	0.7	4.9	3.6	0
40	2.5	0.5	4.9	3.6	0
41	2.5	0.4	5	3.6	0
42	2.5	0.6	4.9	3.6	0
43	2.5	0.6	5	3.6	0
44	2.5	0.7	4.9	3.6	0

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

วงจรที่	Voltage Reference (V)	วัดค่า V(6)		วัดค่า V(out)	
		กรณีมีรถ (V)	กรณีไม่มีรถ(V)	กรณีมีรถ (V)	กรณีไม่มีรถ(V)
45	2.5	0.3	5	3.6	0
46	2.5	0.5	5	3.6	0
47	2.5	0.4	5	3.6	0
48	2.5	0.5	4.9	3.6	0
49	2.5	0.5	5	3.6	0
50	2.5	0.5	5	3.6	0
51	2.5	0.5	5	3.6	0
52	2.5	0.6	4.9	3.6	0
53	2.5	0.7	5	3.6	0
54	2.5	0.8	5	3.6	0
55	2.5	0.4	5	3.6	0
56	2.5	0.5	4.9	3.6	0

$$f_0(V_{(6)-}V_{ref}) = \begin{cases} V_{OH} \text{เมื่อ} (V_{(6)} - V_{ref}) > 0 \\ V_{OL} \text{เมื่อ} (V_{(6)} - V_{ref}) < 0 \end{cases}$$

V_{OH} คือ ระดับแรงดันเอาท์พุตสูง

V_{OL} คือ ระดับแรงดันเอาท์พุตต่ำ

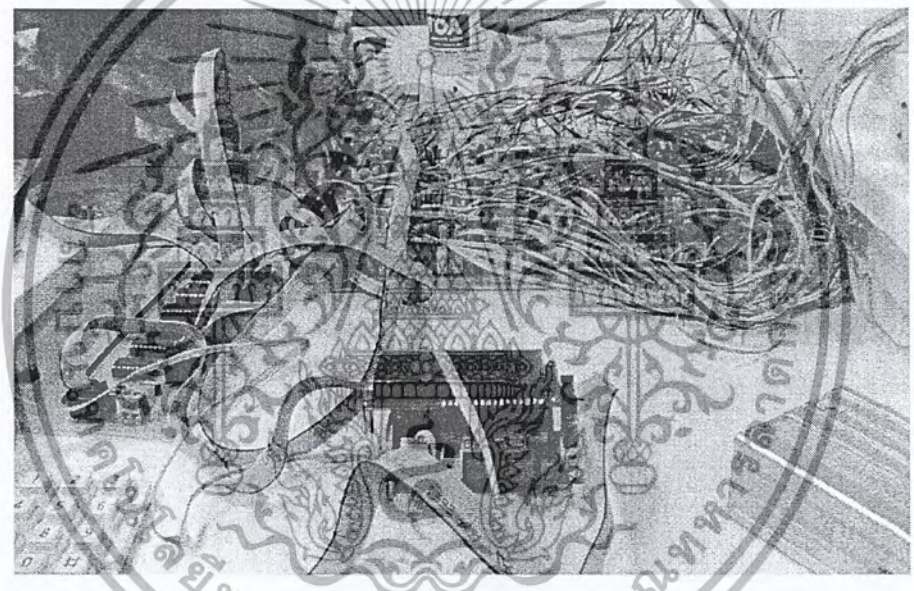
เมื่อวงจรภาคส่งแสงอินฟราเรดส่งแรงดันไฟฟ้าเข้ามา ภาครับแสงอินฟราเรดจะนำแรงดันไฟฟ้า (V_{in}) นี้มาเปรียบเทียบกับแรงดันอ้างอิง (V_{ref}) ซึ่งแรงดันอ้างอิงนี้คือการนำไฟ +5V เข้ามาป้อนให้กับออปแอมป์ตัวแรกหรือวงจรแรงดันตาม เพื่อที่จะนำไฟ +5V นี้มา Upset ให้เป็นไฟ 2.5V เพื่อเป็นแรงดันอ้างอิง โดยแรงดันไฟอ้างอิงนี้สามารถปรับค่าได้โดยเปลี่ยนค่าของ (R_{var}) และในวงจรเราตั้งไว้ที่ 10 กิโลโอห์ม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ในกรณีมีรถเข้ามานั้นแรงดันไฟฟ้า (V_{in}) จะมีค่าน้อยมากจนเกือบเป็นศูนย์ แล้วนำไปเปรียบเทียบกับแรงดันอ้างอิง (V_{ref}) เมื่อเปรียบเทียบกันแล้วพบว่า V_{out} จะออกมาประมาณ 3.6 V หมายความว่าไฟ LED ที่วงจรภาครับแสงอินฟราเรดจะติดคือมีรถเข้ามาจอด

ในกรณีไม่มีรถเข้ามานั้นแรงดันไฟฟ้า (V_{in}) จะมีค่าประมาณหรือเกือบ 5V เมื่อนำไปเปรียบเทียบกับแรงดันอ้างอิง (V_{ref}) เมื่อเปรียบเทียบกันแล้วพบว่า V_{out} จะออกมาเป็นศูนย์ หมายความว่าไฟ LED ที่วงจรภาครับแสงอินฟราเรดจะไม่ติดคือไม่มีรถเข้ามาจอด

วงจรเปรียบเทียบแรงดันจะให้ V_{out} ออกมาเป็น 3.6V เมื่อแรงดันอ้างอิง (V_{ref}) มากกว่าแรงดันไฟฟ้า (V_{in}) แต่ถ้าน้อยกว่า V_{out} จะออกมาเป็นศูนย์

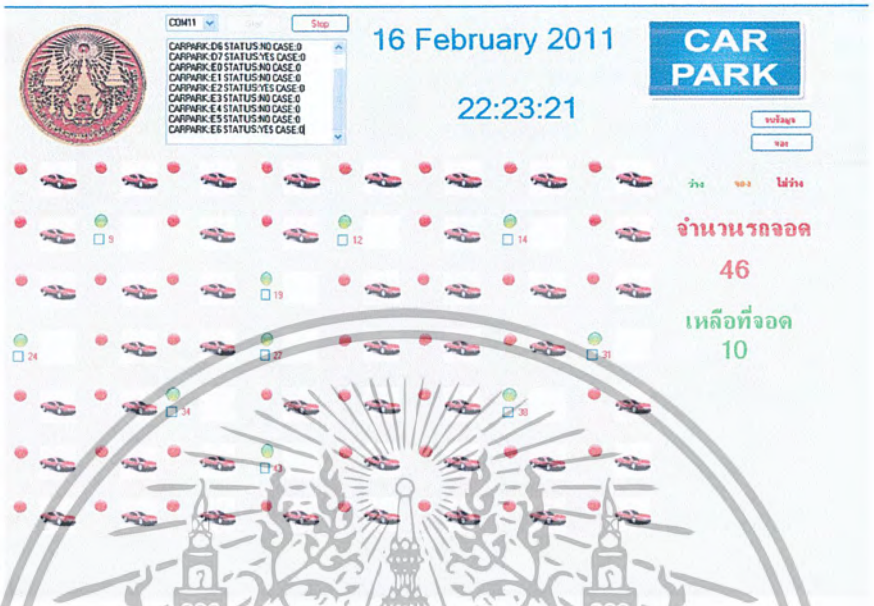


รูปที่ 4.2 การทดสอบวงจรตรวจจับ

จากการทดลองชุดวงจรตรวจจับทางแสงทั้ง 56 ตัว เมื่อตั้งสถานะหลอดไฟของชุดวงจรตรวจจับทางแสงแต่ละวงจรให้หลอดไฟติดเป็นลอจิกศูนย์เมื่อช่องจอดรถนั้นไม่มีว่างหรือไม่มีรถเข้ามาจอด และให้สถานะหลอดไฟดับเป็นลอจิกหนึ่ง ซึ่งหมายถึงมีรถเข้ามาจอดตรงช่องจอดรถ แล้ววงจรตรวจจับทางแสงก็ทำงาน และระยะที่อุปกรณ์ตรวจจับทางแสงทำงานอยู่ประมาณ 5 เซนติเมตร

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

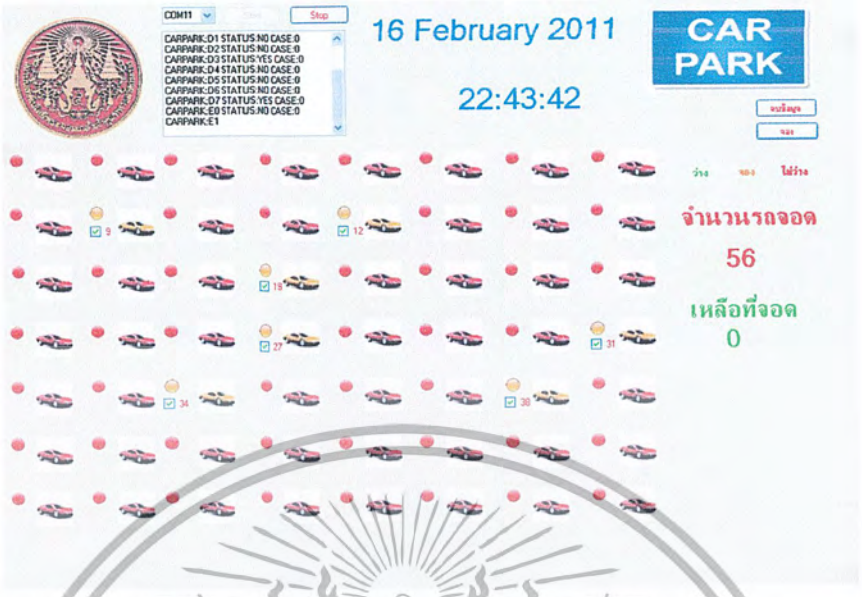
4.3 ทดสอบการทำงานของระบบ



รูปที่ 4.3 ผลการทดสอบหน้าจอมอนิเตอร์กรณีมีที่ว่าง

ในทำงานนี้สมมติให้รถที่วิ่งเข้ามาจอดจะเห็นสถานะของช่องจอดรถนั้นๆ ว่าว่าง ซึ่งสามารถเข้าไปกดจองช่องจอดรถได้ และต้องกดช่องที่เป็นสีเขียวเท่านั้น เมื่อเข้าไปกดจองแล้วสถานะของช่องจอดรถนั้นจะแสดงสถานะใหม่เป็นสีเหลือง จะแสดงถึงช่องจอดนั้นมีเจ้าของแล้วและกำลังจะนำรถเข้ามาจอด เมื่อมีรถเข้ามาจอดแล้วสถานะของช่องจอดรถจะเปลี่ยนเป็นสีแดงหมายถึงมีรถเข้ามาจอดแล้ว แล้วเมื่อรถออกจากช่องจอดรถสถานะก็จะเป็นสีเขียว หมายถึงช่องจอดนี้ว่างแล้วพร้อมที่จะให้จองต่อไป ซึ่งในที่นี้จะเห็นว่ามีช่องจอดที่ว่างให้จอดอยู่ 10 ที่ซึ่งต่อจากนี้จะเป็นกระบวนการนำไปสู่การกดจองช่องจอดรถ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.4 ผลการทดสอบหน้าจอมอนิเตอร์กรณีมีการจอง

และในกรณีรถที่เข้ามาเจอสถานะของช่องจอดรถเป็นสีเหลืองหมายถึง ช่องจอดรถแถวนั้นทั้งแถวมีการจองหมดแล้ว ซึ่งถ้าไม่ใช่ผู้ที่เข้ามาจองไว้จะต้องหาช่องจอดรถแถวอื่นที่เป็นสีเขียวเท่านั้นและต้องกดจองไว้ก่อน เพื่อที่จะนำรถเข้าไปจอดให้ถูกช่อง ส่วนคนที่อยู่ในสถานะที่เข้ามาจองทั้งแถวจะต้องนำรถมาจอดไว้ตามเวลาที่หน่วงไว้ เพื่อไม่เป็นการสละสิทธิ์ในการเข้าจอด โดยสถานะสีเหลืองนี้จะมีการหน่วงเวลาไว้ตามความเหมาะสม ถ้าเกินเวลาไปช่องจอดที่ไม่มีรถเข้ามาจะเปลี่ยนสถานะเป็นสีเขียว ซึ่งในการทดสอบนี้ได้ทำการตั้งให้มีช่องจอดว่างอยู่ 7 ช่อง แล้วได้ทำการกดจองเป็นจำนวน 7 ช่อง และในกรณีที่มีการจองทั้งแถวอาจเป็นการให้ช่องจอดทั้งแถวเป็นของกลุ่มบุคคลไปจำเป็นต้องใช้ช่องจอดรถจำนวนมากเพื่อมาทำธุระในอาคารนี้ เช่น การมาประชุมที่โรงแรม หรือการมาทำการกิจ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 5
สรุปผลการดำเนินงาน

ในการทำโครงงานปริญญานิพนธ์ การจัดการที่จอดรถ เพื่อศึกษาในด้านต่างๆที่เกี่ยวข้องกับโครงงาน เช่น การทำงานของอุปกรณ์ตรวจจับสัญญาณทางแสง การเขียนโปรแกรมด้วยภาษาซีชาร์ป การเชื่อมต่อคอมพิวเตอร์กับอุปกรณ์ภายนอกและการทำงานของไมโครคอนโทรลเลอร์ เป็นต้น โดยขอบเขตของโครงงานนี้คือ สามารถแสดงตำแหน่งของที่จอดรถยนต์แต่ละตำแหน่งภายในลานจอดรถยนต์และแสดงจำนวนรถยนต์ที่จอดอยู่ในแบบจำลองนี้สามารถจอดรถยนต์ได้ 56 คันและทางเข้าจะต้องแสดงว่าอาคารนี้มีจำนวนรถยนต์เข้าไปเต็มหรือไม่ เราสามารถที่จะสรุปการทำงาน และปัญหาต่าง ๆ ที่เกิดขึ้นระหว่างการทำโครงงานปริญญานิพนธ์ รวมถึงแนวทางแก้ไข และข้อเสนอแนะเพื่อที่จะได้นำโครงงานปริญญานิพนธ์นี้ไปพัฒนาต่อไปซึ่งจะแบ่งเป็นข้อ ๆ ได้คือ

5.1 สรุปผล

แบบจำลองสามารถนำรถเข้าไปจอดได้จำนวน 56 คัน ซึ่งวงจรตรวจจับสามารถตรวจจับรถที่ เข้ามาจอดได้โดยการควบคุมจากไมโครคอนโทรลเลอร์และส่งค่าผลจากวงจรตรวจจับไปยังคอมพิวเตอร์โดยผ่านพอร์ตอนุกรม บอกจำนวนรถที่เข้ามาจอด จำนวนที่จอดรถที่เหลือ และบอกตำแหน่งลานจอดรถว่าว่างหรือไม่ว่างและสามารถกดจองเพื่อนำรถเข้าไปจอดตำแหน่งนั้นได้

จากการทำโครงงานนี้ทำให้ได้รับความรู้ความเข้าใจแล้วยังจะนำความสามารถในการทำงานของไมโครคอนโทรลเลอร์มาประยุกต์ใช้งานในด้านต่างๆและในการเขียนโปรแกรมภาษาซีชาร์ปในการคำนวณและประมวลผลและการทำผังจอดรถที่แสดงเป็นสีสัญลักษณ์ต่าง ๆ ได้ หรือกราฟฟิกส์ต่าง ๆ ที่ทำให้มีภาพที่แสดงออกมามีความสวยงามมากขึ้น

5.2 ปัญหาและแนวทางการแก้ไข

ปัญหาและแนวทางแก้ไขที่เกิดขึ้นของการทำโครงงานปริญญานิพนธ์ การจัดการที่จอดรถ นี้คือ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1. ปัญหา การส่งแสงอินฟราเรดจากตัวส่งไปยังตัวรับระยะห่างของอุปกรณ์ทั้งสองทำให้ลำแสงที่ส่งไปเกิดการลดทอน และการรบกวนจากภายนอก แนวทางการแก้ไข เลือกตัวส่งที่มีกำลังส่งมากขึ้น (เปลี่ยนวงจรให้มีกำลังมากขึ้น) หรือความเข้มแสงที่มาก หรืออาจเลือกใช้เซ็นเซอร์ชนิดอื่นๆ ตามความเหมาะสมของสถานที่นั้นๆ

2. ปัญหา สถานะเอาท์พุตของวงจรตรวจจับไม่แน่นอนคือ ถ้าเอาต์พุทมีแรงดันระหว่าง 0.8 - 2.5 โวลท์ทำให้การผิดพลาดของข้อมูลเกิดขึ้นได้ แนวทางการแก้ไข โดยต่อเอาท์พุตผ่านบัฟเฟอร์ เพื่อแยกสถานะลอจิกให้ถูกต้อง

3. ปัญหา ความผิดพลาดที่เกิดจากอุปกรณ์ไม่ได้มาตรฐาน ทำให้การทำงานของวงจรผิดพลาดไม่ได้ตามที่ต้องการ แนวทางการแก้ไข ต้องมีการทดสอบอุปกรณ์ก่อนที่จะนำไปใช้งานเสียก่อน เพื่อความสะดวกและรวดเร็วต่อการติดตั้ง

5.3 แนวทางการพัฒนา

1.ใช้วงจรตรวจจับชนิดอื่น มาใช้แทนวงจรตรวจจับอินฟราเรด เช่น อุลตร้าโซนิค เพื่อให้ทำงานได้ดี และเหมาะสมขึ้น เพราะอินฟราเรดมีการรบกวนของแสง

2. ควรมีการขยายวงจรเพื่อให้สามารถรองรับชุดวงจรตรวจจับ ที่มีจำนวนมากขึ้นในสถานที่ที่ต่างกันออกไปพัฒนาจำนวนรถในระบบมากขึ้นด้วยการได้มาขึ้นด้วยการเพิ่มวงจรเซ็นเซอร์และทำการขยายพอร์ต MCS-51

3. สามารถใช้เป็นแนวทางในการพัฒนา ให้ใช้กับสถานที่จริงได้ในอนาคต หรือสามารถนำไปพัฒนาให้ดีกว่าที่เป็นอยู่จากเดิม อาจมีการติดตั้งที่กั้นประตูทางเข้า-ออก ที่ควบคุมด้วยสเตปมอเตอร์

4. พัฒนาระบบให้มีการเก็บข้อมูลจำนวนรถเป็นแบบฐานข้อมูล และพัฒนาให้มีการจองที่จอดรถผ่านเครือข่ายอินเตอร์เน็ต

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บรรณานุกรม

[1] ฉัททวุฒิ พืชผล, *Visual Basic 6*, บริษัท โปรวิชั่น จำกัด, กรุงเทพมหานคร, 2542.

[2] นคร ภักดีชาติ และ ชัยวัฒน์ ลิ้มพรวิจิตร, *การทดลองและใช้งานไมโครคอลโทรเลอร์MCS-51 ด้วยโปรแกรมภาษาซี*, บริษัท อินโนเวตีฟ เอ็กเพอร์เมนต์ จำกัด, กรุงเทพหมานคร, 2521.

[3] พจนาฏ สุวรรณมณี, *เซ็นเซอร์และทรานสดิวเซอร์เบื้องต้น*, สมาคมส่งเสริมเทคโนโลยี (ไทย-ญี่ปุ่น), กรุงเทพมหานคร, 2545.

[4] ดอนสัน ปงผาบ, *ไมโครคอนโทรลเลอร์และการประยุกต์ใช้งาน*, สมาคมส่งเสริมเทคโนโลยี (ไทย-ญี่ปุ่น), กรุงเทพมหานคร, 2553.

[5] http://www.th.wikipedia.org/

[6] http://suranat.tripod.com/u3c2.html

[7] http://61.19.247.211/~adisak51/page06_1.html

[8] http://www.vcharkarn.com/vcafe/65718

[9] http://www.oknation.net/blog/ec/2008/02/06/entry-3

[10] http:// www.thaiall.com/tc/index.php

[11] http://www.sut.ac.th/Engineering/Telecom/about/student2/ST51.htm

[12] http://203.170.167.149/it/elearning/mcstext/stumCS51.htm

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Features

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

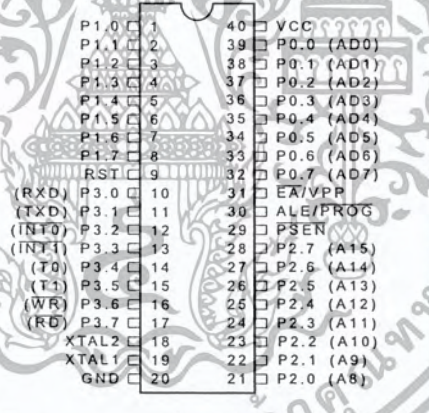
Description

The AT89C51 is a low-power, high-performance CMOS 8-bit microcomputer with 4 Kbytes of Flash Programmable and Erasable Read Only Memory (PEROM). The device is manufactured using Atmel's high density nonvolatile memory technology and is compatible with the industry standard MCS-51™ instruction set and pinout. The on-chip Flash allows the program memory to be reprogrammed in-system or by a conventional nonvolatile memory programmer. By combining a versatile 8-bit CPU with Flash on a monolithic chip, the Atmel AT89C51 is a powerful microcomputer which provides a highly flexible and cost effective solution to many embedded control applications.

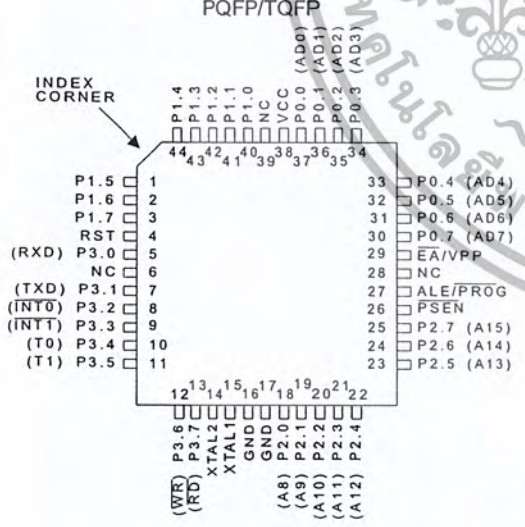
(continued)

Pin Configurations

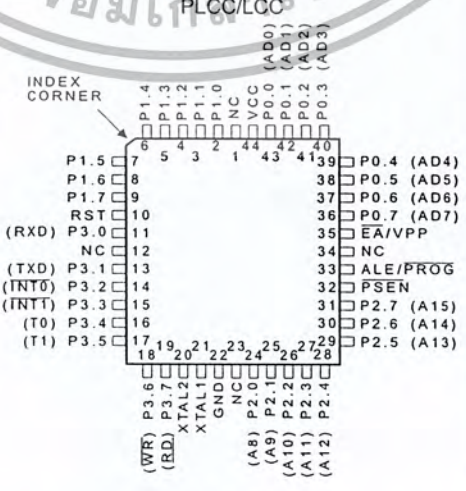
PDIP/Cerdip



PQFP/TQFP



PLCC/LCC



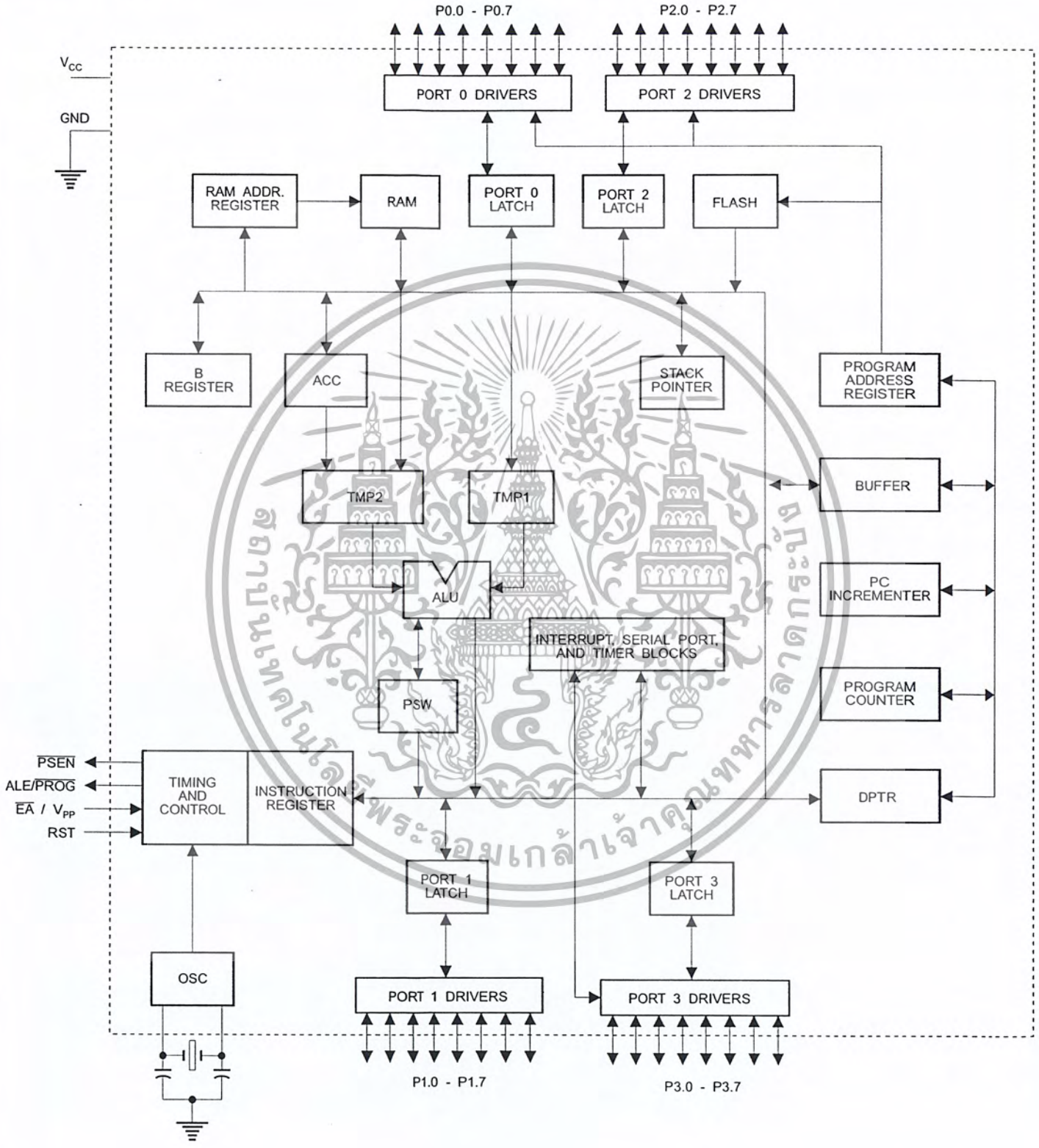
8-Bit Microcontroller with 4 Kbytes Flash

AT89C51

0265E

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Block Diagram



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Description (Continued)

The AT89C51 provides the following standard features: 4 Kbytes of Flash, 128 bytes of RAM, 32 I/O lines, two 16-bit timer/counters, a five vector two-level interrupt architecture, a full duplex serial port, on-chip oscillator and clock circuitry. In addition, the AT89C51 is designed with static logic for operation down to zero frequency and supports two software selectable power saving modes. The Idle Mode stops the CPU while allowing the RAM, timer/counters, serial port and interrupt system to continue functioning. The Power Down Mode saves the RAM contents but freezes the oscillator disabling all other chip functions until the next hardware reset.

Pin Description

V_{CC}

Supply voltage.

GND

Ground.

Port 0

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

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

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

Port 1

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

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

Port 2

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

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

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

Port 3

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

Port 3 also serves the functions of various special features of the AT89C51 as listed below:

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

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

RST

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

ALE/$\overline{PROG}$

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

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

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

$\overline{PSEN}$

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

(continued)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Pin Description (Continued)

When the AT89C51 is executing code from external program memory, PSEN is activated twice each machine cycle, except that two PSEN activations are skipped during each access to external data memory.

EA/VPP

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

EA should be strapped to VCC for internal program executions.

This pin also receives the 12-volt programming enable voltage (VPP) during Flash programming, for parts that require 12-volt VPP.

XTAL1

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

XTAL2

Output from the inverting oscillator amplifier.

Oscillator Characteristics

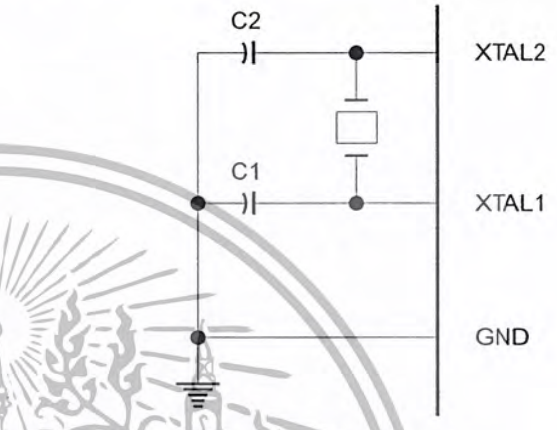
XTAL1 and XTAL2 are the input and output, respectively, of an inverting amplifier which can be configured for use as an on-chip oscillator, as shown in Figure 1. Either a quartz crystal or ceramic resonator may be used. To drive the device from an external clock source, XTAL2 should be left unconnected while XTAL1 is driven as shown in Figure 2. There are no requirements on the duty cycle of the external clock signal, since the input to the internal clocking circuitry is through a divide-by-two flip-flop, but minimum and maximum voltage high and low time specifications must be observed.

Idle Mode

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

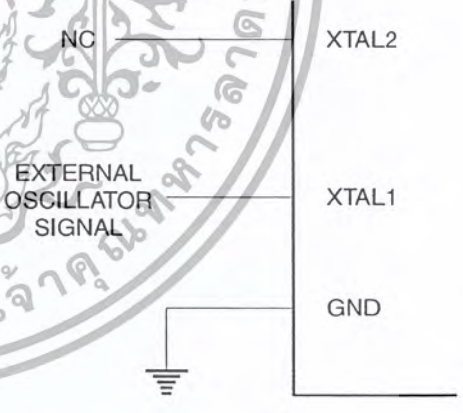
It should be noted that when idle is terminated by a hardware reset, the device normally resumes program execution, from where it left off, up to two machine cycles before the internal reset algorithm takes control. On-chip hard-

Figure 1. Oscillator Connections



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

Figure 2. External Clock Drive Configuration



Status of External Pins During Idle and Power Down

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

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ware inhibits access to internal RAM in this event, but access to the port pins is not inhibited. To eliminate the possibility of an unexpected write to a port pin when Idle is terminated by reset, the instruction following the one that invokes Idle should not be one that writes to a port pin or to external memory.

Power Down Mode

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

Program Memory Lock Bits

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

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

Lock Bit Protection Modes

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

Programming the Flash

The AT89C51 is normally shipped with the on-chip Flash memory array in the erased state (that is, contents = FFH) and ready to be programmed. The programming interface accepts either a high-voltage (12-volt) or a low-voltage (V_{CC}) program enable signal. The low voltage programming mode provides a convenient way to program the AT89C51 inside the user's system, while the high-voltage programming mode is compatible with conventional third party Flash or EPROM programmers.

The AT89C51 is shipped with either the high-voltage or low-voltage programming mode enabled. The respective top-side marking and device signature codes are listed in the following table.

	V_{PP} = 12 V	V_{PP} = 5 V
Top-Side Mark	AT89C51 xxxx yyww	AT89C51 xxxx-5 yyww
Signature	(030H)=1EH (031H)=51H (032H)=FFH	(030H)=1EH (031H)=51H (032H)=05H

The AT89C51 code memory array is programmed byte-by-byte in either programming mode. *To program any non-blank byte in the on-chip Flash Memory, the entire memory must be erased using the Chip Erase Mode.*

Programming Algorithm: Before programming the AT89C51, the address, data and control signals should be set up according to the Flash programming mode table and Figures 3 and 4. To program the AT89C51, take the following steps.

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

$\overline{Data}$ Polling: The AT89C51 features $\overline{Data}$ Polling to indicate the end of a write cycle. During a write cycle, an at-

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

(continued)

Programming the Flash (Continued)

tempted read of the last byte written will result in the complement of the written datum on PO.7. Once the write cycle has been completed, true data are valid on all outputs, and the next cycle may begin. Data Polling may begin any time after a write cycle has been initiated.

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

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

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

Reading the Signature Bytes: The signature bytes are read by the same procedure as a normal verification of locations 030H,

031H, and 032H, except that P3.6 and P3.7 must be pulled to a logic low. The values returned are as follows.

(030H) = 1EH indicates manufactured by Atmel
(031H) = 51H indicates 89C51
(032H) = FFH indicates 12 V programming
(032H) = 05H indicates 5 V programming

Programming Interface

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

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

Flash Programming Modes

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

Notes: 1. The signature byte at location 032H designates whether V_{PP} = 12 V or V_{PP} = 5 V should be used to enable programming.

2. Chip Erase requires a 10 ms $\overline{\text{PROG}}$ pulse.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Figure 3. Programming the Flash

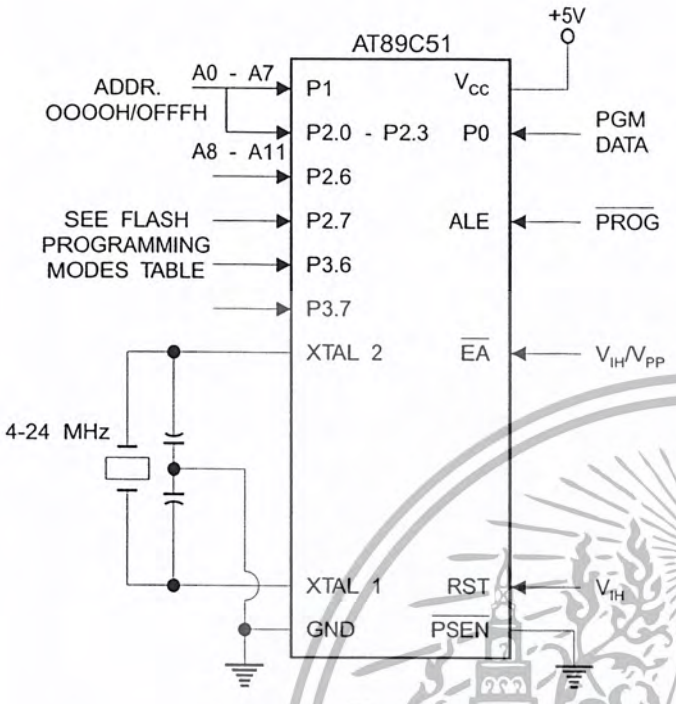
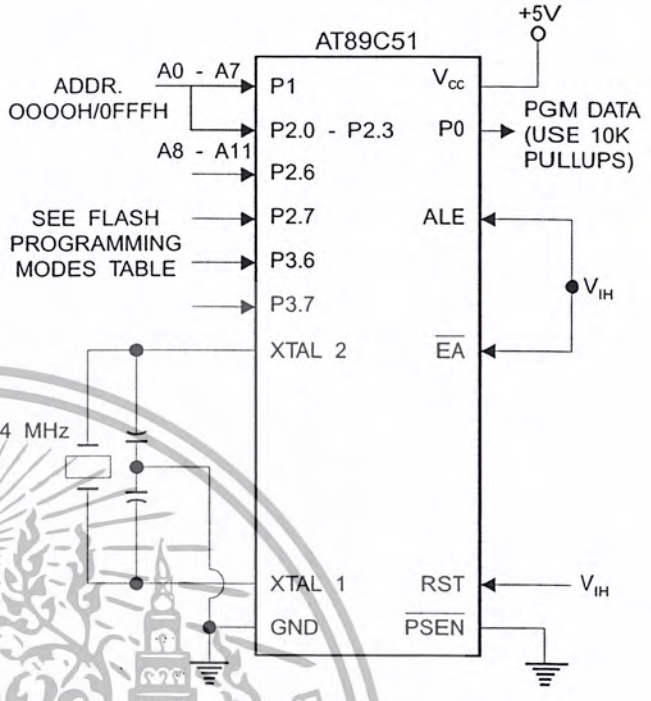


Figure 4. Verifying the Flash



Flash Programming and Verification Characteristics

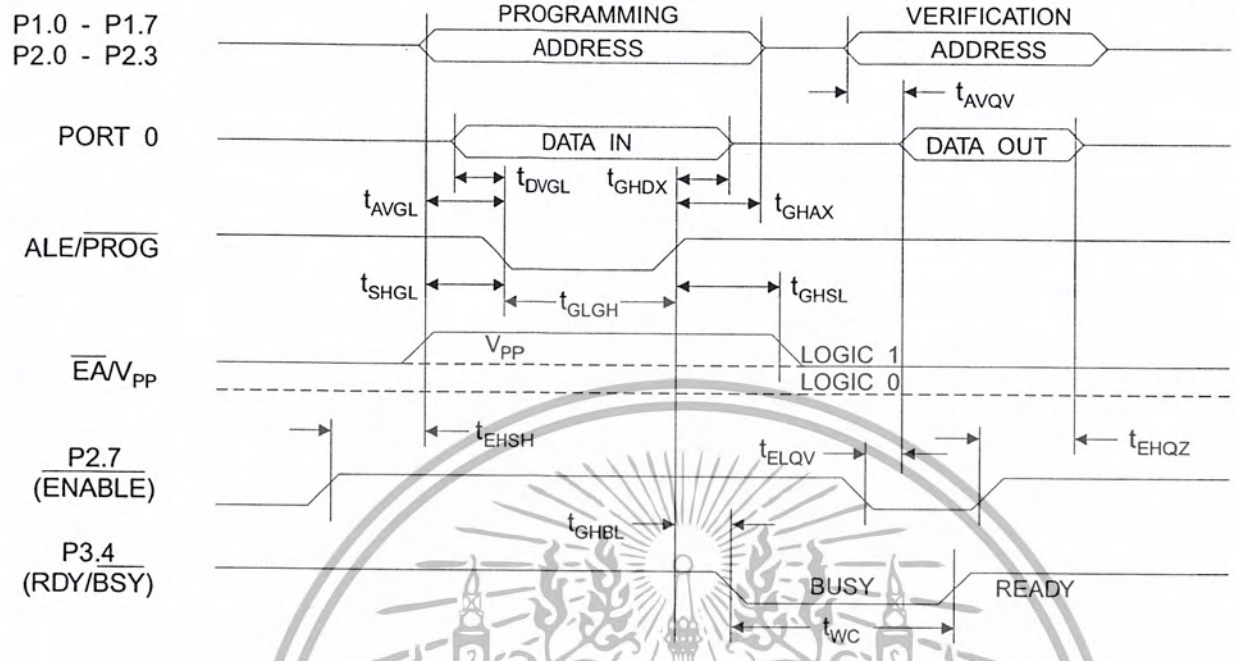
T_A = 21°C to 27°C, V_{CC} = 5.0 ± 10%

Symbol	Parameter	Min	Max	Units
V_{PP}(1)	Programming Enable Voltage	11.5	12.5	V
I_{PP}(1)	Programming Enable Current		1.0	mA
$1/t_{CLCL}$	Oscillator Frequency	4	24	MHz
t_{AVGL}	Address Setup to $\overline{PROG}$ Low	$48t_{CLCL}$		
t_{GHAX}	Address Hold After $\overline{PROG}$	$48t_{CLCL}$		
t_{DVGL}	Data Setup to $\overline{PROG}$ Low	$48t_{CLCL}$		
t_{GHDX}	Data Hold After $\overline{PROG}$	$48t_{CLCL}$		
t_{EHSH}	P2.7 ($\overline{ENABLE}$) High to V_{PP}	48tCLCL		
t_{SHGL}	V_{PP} Setup to $\overline{PROG}$ Low	10		µs
t_{GHSL}(1)	V_{PP} Hold After $\overline{PROG}$	10		µs
t_{GLGH}	$\overline{PROG}$ Width	1	110	µs
t_{AVQV}	Address to Data Valid		48tCLCL	
t_{ELQV}	$\overline{ENABLE}$ Low to Data Valid		$48t_{CLCL}$	
t_{EHQV}	Data Float After $\overline{ENABLE}$	0	$48t_{CLCL}$	
t_{GHBL}	$\overline{PROG}$ High to $\overline{BUSY}$ Low		1.0	µs
t_{WC}	Byte Write Cycle Time		2.0	ms

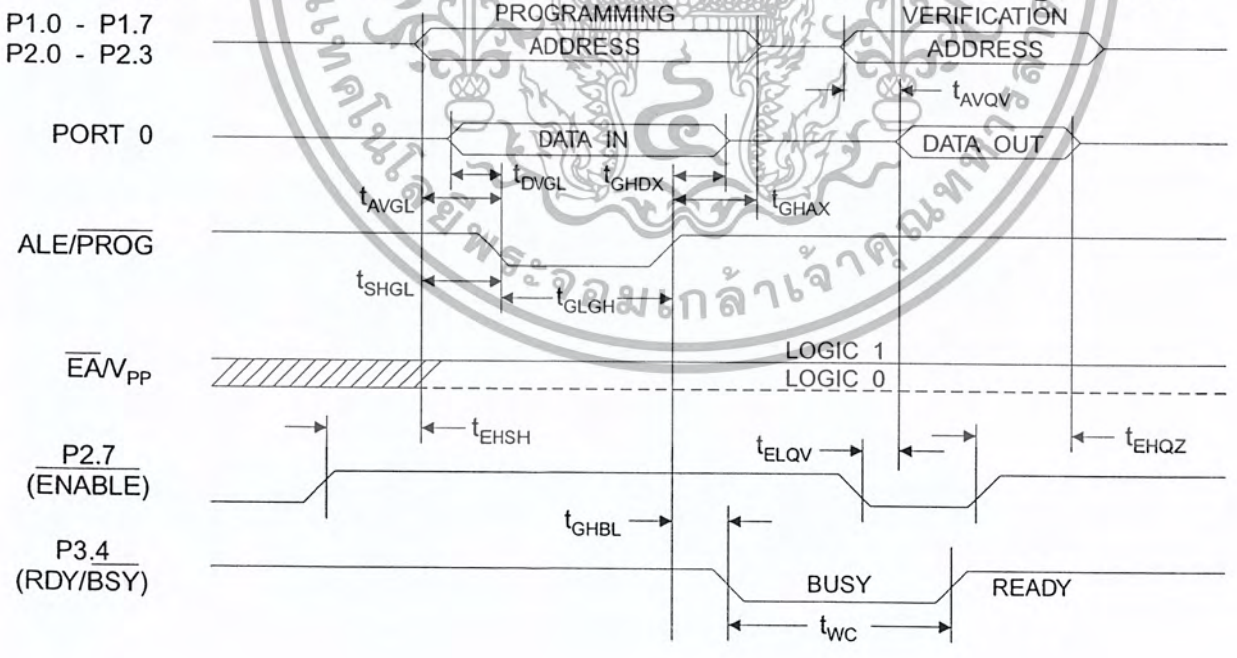
Note: 1. Only used in 12-volt programming mode.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Flash Programming and Verification Waveforms - High Voltage Mode



Flash Programming and Verification Waveforms - Low Voltage Mode



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Absolute Maximum Ratings*

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

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

D.C. Characteristics

T_A = -40°C to 85°C, V_{CC} = 5.0 V ± 20% (unless otherwise noted)

Symbol	Parameter	Condition	Min	Max	Units
V_{IL}	Input Low Voltage	(Except $\overline{EA}$)	-0.5	0.2 V_{CC}-0.1	V
V_{IL1}	Input Low Voltage ($\overline{EA}$)		-0.5	0.2 V_{CC}-0.3	V
V_{IH}	Input High Voltage	(Except XTAL1, RST)	0.2 V_{CC}+0.9	V_{CC}+0.5	V
V_{IH1}	Input High Voltage	(XTAL1, RST)	0.7 V_{CC}	V_{CC}+0.5	V
V_{OL}	Output Low Voltage[1] (Ports 1,2,3)	I_{OL} = 1.6 mA		0.45	V
V_{OL1}	Output Low Voltage[1] (Port 0, ALE, PSEN)	I_{OL} = 3.2 mA		0.45	V
V_{OH}	Output High Voltage (Ports 1,2,3, ALE, PSEN)	I_{OH} = -60 µA, V_{CC} = 5 V ± 10%	2.4		V
		I_{OH} = -25 µA	0.75 V_{CC}		V
		I_{OH} = -10 µA	0.9 V_{CC}		V
V_{OH1}	Output High Voltage (Port 0 in External Bus Mode)	I_{OH} = -800 µA, V_{CC} = 5 V ± 10%	2.4		V
		I_{OH} = -300 µA	0.75 V_{CC}		V
		I_{OH} = -80 µA	0.9 V_{CC}		V
I_{IL}	Logical 0 Input Current (Ports 1,2,3)	V_{IN} = 0.45 V		-50	µA
I_{TL}	Logical 1 to 0 Transition Current (Ports 1,2,3)	V_{IN} = 2 V		-650	µA
I_{LI}	Input Leakage Current (Port 0, EA)	0.45 < V_{IN} < V_{CC}		±10	µA
RRST	Reset Pulldown Resistor		50	300	KΩ
C_{IO}	Pin Capacitance	Test Freq. = 1 MHz, T_A = 25°C		10	pF
I_{CC}	Power Supply Current	Active Mode, 12 MHz		20	mA
		Idle Mode, 12 MHz		5	mA
	Power Down Mode[2]	V_{CC} = 6 V		100	µA
		V_{CC} = 3 V		40	µA

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

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

A.C. Characteristics

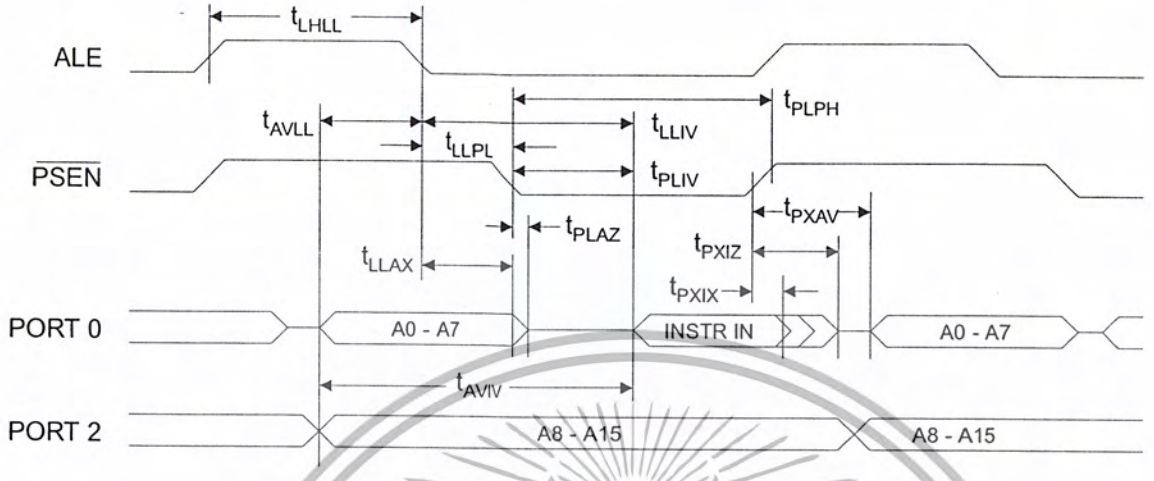
(Under Operating Conditions; Load Capacitance for Port 0, ALE/$\overline{PROG}$, and $\overline{PSEN}$ = 100 pF; Load Capacitance for all other outputs = 80 pF)

External Program and Data Memory Characteristics

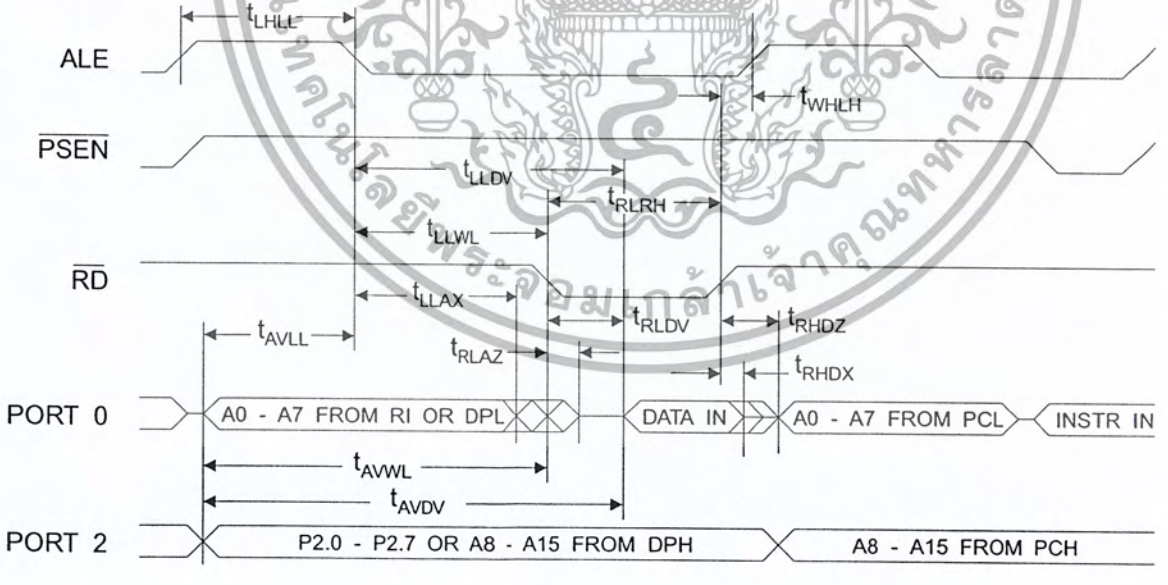
Symbol	Parameter	12 MHz Oscillator		16 to 24 MHz Oscillator		Units
		Min	Max	Min	Max	
$1/t_{CLCL}$	Oscillator Frequency			0	24	MHz
t_{LHLL}	ALE Pulse Width	127		$2t_{CLCL}-40$		ns
t_{AVLL}	Address Valid to ALE Low	28		$tCLCL-13$		ns
t_{LLAX}	Address Hold After ALE Low	48		$tCLCL-20$		ns
t_{LLIV}	ALE Low to Valid Instruction In		233		$4tCLCL-65$	ns
t_{LLPL}	ALE Low to $\overline{PSEN}$ Low	43		$t_{CLCL}-13$		ns
t_{PLPH}	$\overline{PSEN}$ Pulse Width	205		$3t_{CLCL}-20$		ns
t_{PLIV}	$\overline{PSEN}$ Low to Valid Instruction In		145		$3t_{CLCL}-45$	ns
t_{PXIX}	Input Instruction Hold After $\overline{PSEN}$	0		0		ns
t_{PXIZ}	Input Instruction Float After $\overline{PSEN}$		59		$t_{CLCL}-10$	ns
t_{PXAV}	$\overline{PSEN}$ to Address Valid	75		$t_{CLCL}-8$		ns
t_{AVIV}	Address to Valid Instruction In		312		$5t_{CLCL}-55$	ns
t_{PLAZ}	$\overline{PSEN}$ Low to Address Float		10		10	ns
t_{RLRH}	$\overline{RD}$ Pulse Width	400		$6t_{CLCL}-100$		ns
t_{WLWH}	$\overline{WR}$ Pulse Width	400		$6t_{CLCL}-100$		ns
t_{RLDV}	$\overline{RD}$ Low to Valid Data In		252		$5t_{CLCL}-90$	ns
t_{RHDX}	Data Hold After $\overline{RD}$	0		0		ns
t_{RHDZ}	Data Float After $\overline{RD}$		97		$2t_{CLCL}-28$	ns
t_{LLDV}	ALE Low to Valid Data In		517		$8tCLCL-150$	ns
t_{AVDV}	Address to Valid Data In		585		$9tCLCL-165$	ns
t_{LLWL}	ALE Low to $\overline{RD}$ or $\overline{WR}$ Low	200	300	$3t_{CLCL}-50$	$3t_{CLCL}+50$	ns
t_{AVWL}	Address to $\overline{RD}$ or $\overline{WR}$ Low	203		$4t_{CLCL}-75$		ns
t_{QVWX}	Data Valid to $\overline{WR}$ Transition	23		$t_{CLCL}-20$		ns
t_{QVWH}	Data Valid to $\overline{WR}$ High	433		$7t_{CLCL}-120$		ns
t_{WHQX}	Data Hold After $\overline{WR}$	33		$t_{CLCL}-20$		ns
t_{RLAZ}	$\overline{RD}$ Low to Address Float		0		0	ns
t_{WHLH}	$\overline{RD}$ or $\overline{WR}$ High to ALE High	43	123	$t_{CLCL}-20$	$t_{CLCL}+25$	ns

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

External Program Memory Read Cycle

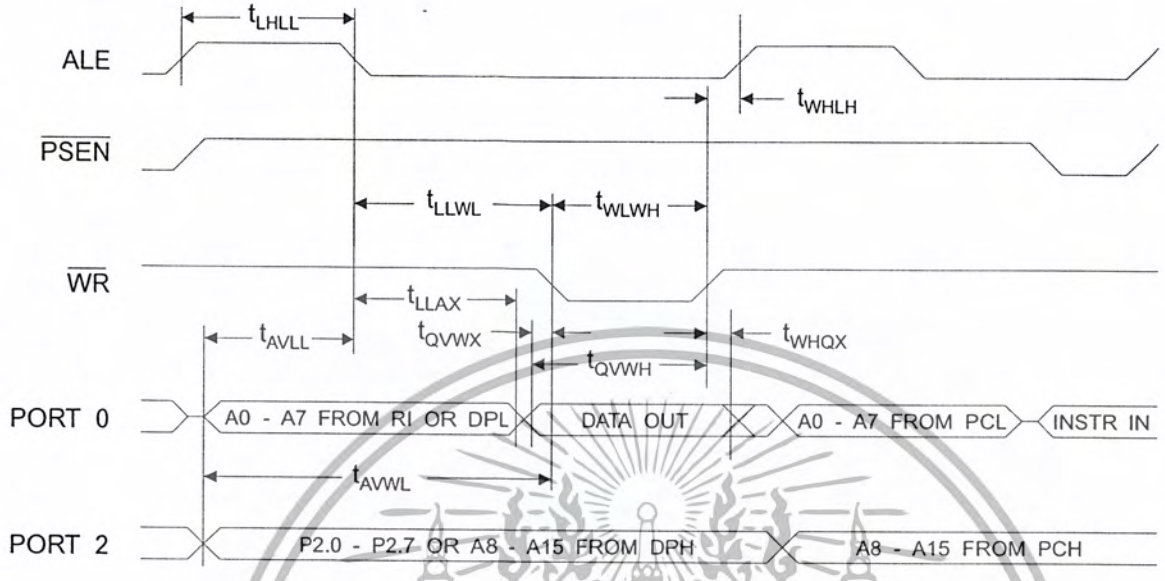


External Data Memory Read Cycle



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

External Data Memory Cycle



External Clock Drive Waveforms

External Clock Drive

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

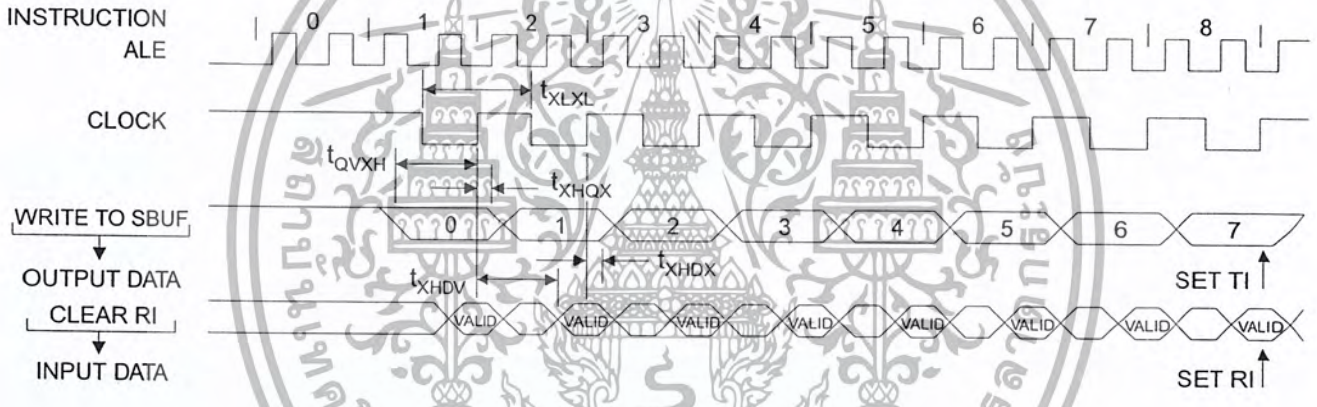
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช

Serial Port Timing: Shift Register Mode Test Conditions

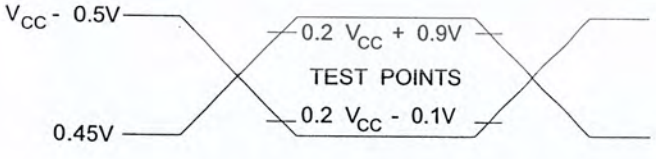
(V_{CC} = 5.0 V ± 20%; Load Capacitance = 80 pF)

Symbol	Parameter	12 MHz Osc		Variable Oscillator		Units
		Min	Max	Min	Max	
t_{XLXL}	Serial Port Clock Cycle Time	1.0		$12t_{CLCL}$		µs
t_{QVXH}	Output Data Setup to Clock Rising Edge	700		$10t_{CLCL}$-133		ns
t_{XHQX}	Output Data Hold After Clock Rising Edge	50		$2t_{CLCL}$-33		ns
t_{XHDX}	Input Data Hold After Clock Rising Edge	0		0		ns
t_{XHDV}	Clock Rising Edge to Input Data Valid		700		10tCLCL-133	ns

Shift Register Mode Timing Waveforms

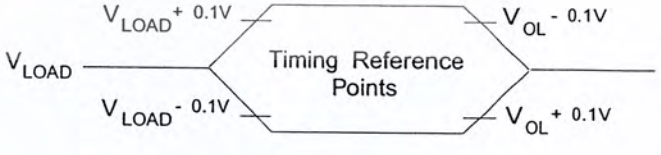


AC Testing Input/Output Waveforms [1]



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

Float Waveforms [1]



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

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Ordering Information

Speed (MHz)	Power Supply	Ordering Code	Package	Operation Range
12	5 V ± 20%	AT89C51-12AC AT89C51-12JC AT89C51-12PC AT89C51-12QC	44A 44J 40P6 44Q	Commercial (0°C to 70°C)
		AT89C51-12AI AT89C51-12JI AT89C51-12PI AT89C51-12QI	44A 44J 40P6 44Q	Industrial (-40°C to 85°C)
		AT89C51-12AA AT89C51-12JA AT89C51-12PA AT89C51-12QA	44A 44J 40P6 44Q	Automotive (-40°C to 125°C)
	5 V ± 10%	AT89C51-12DM AT89C51-12LM	40D6 44L	Military (-55°C to 125°C)
		AT89C51-12DM/883 AT89C51-12LM/883	40D6 44L	Military/883C Class B, Fully Compliant (-55°C to 125°C)
16	5 V ± 20%	AT89C51-16AC AT89C51-16JC AT89C51-16PC AT89C51-16QC	44A 44J 40P6 44Q	Commercial (0°C to 70°C)
		AT89C51-16AI AT89C51-16JI AT89C51-16PI AT89C51-16QI	44A 44J 40P6 44Q	Industrial (-40°C to 85°C)
		AT89C51-16AA AT89C51-16JA AT89C51-16PA AT89C51-16QA	44A 44J 40P6 44Q	Automotive (-40°C to 125°C)
20	5 V ± 20%	AT89C51-20AC AT89C51-20JC AT89C51-20PC AT89C51-20QC	44A 44J 40P6 44Q	Commercial (0°C to 70°C)
		AT89C51-20AI AT89C51-20JI AT89C51-20PI AT89C51-20QI	44A 44J 40P6 44Q	Industrial (-40°C to 85°C)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Ordering Information

Speed (MHz)	Power Supply	Ordering Code	Package	Operation Range
24	5 V ± 20%	AT89C51-24AC AT89C51-24JC AT89C51-24PC AT89C51-24QC	44A 44J 44P6 44Q	Commercial (0°C to 70°C)
		AT89C51-24AI AT89C51-24JI AT89C51-24PI AT89C51-24QI	44A 44J 44P6 44Q	Industrial (-40°C to 85°C)

Package Type	
44A	44 Lead, Thin Plastic Gull Wing Quad Flatpack (TQFP)
40D6	40 Lead, 0.600" Wide, Non-Windowed, Ceramic Dual Inline Package (Cerdip)
44J	44 Lead, Plastic J-Leaded Chip Carrier (PLCC)
44L	44 Pad, Non-Windowed, Ceramic Leadless Chip Carrier (LCC)
40P6	40 Lead, 0.600" Wide, Plastic Dual Inline Package (PDIP)
44Q	44 Lead, Plastic Gull Wing Quad Flatpack (PQFP)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ET-BASE51 AC3 (AT89C51AC3)

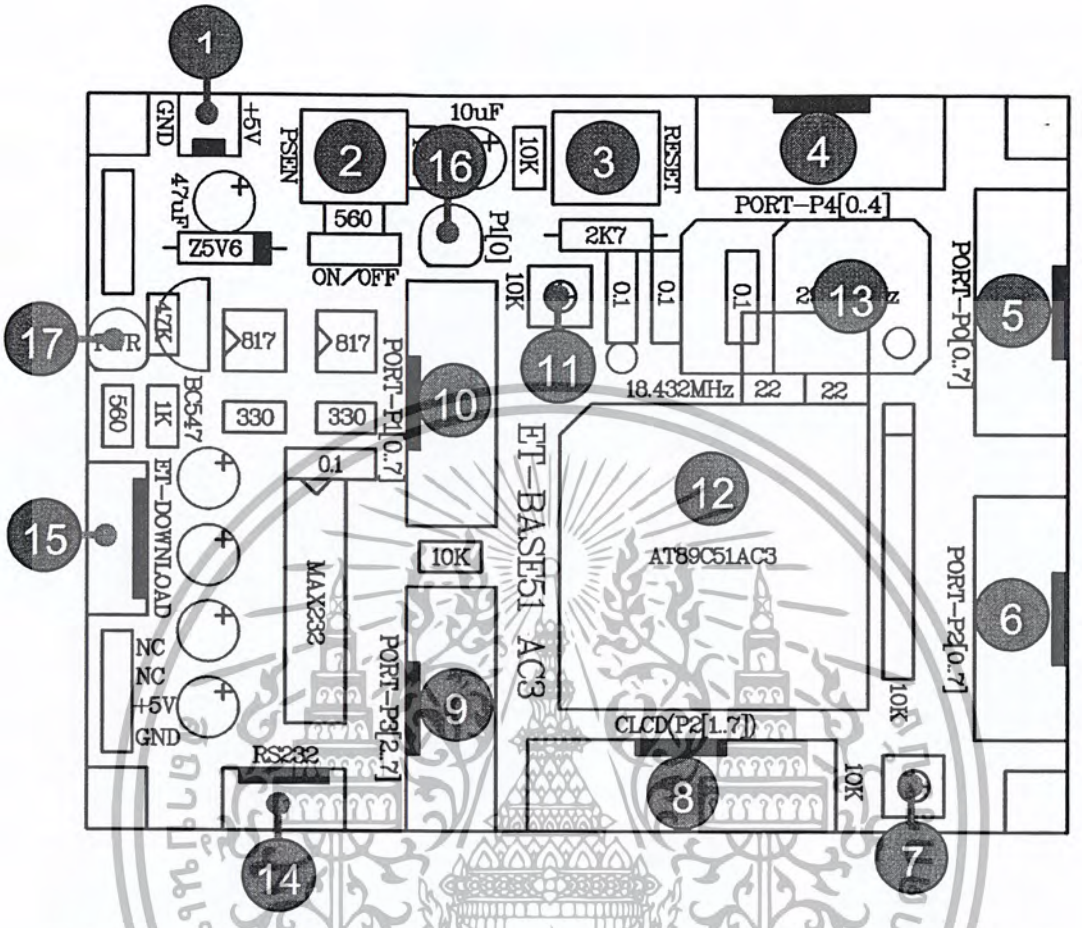
ET-BASE51 AC3 เป็นบอร์ดไมโครคอนโทรเลอร์ในตระกูล MCS51 ขนาด 52 Pin ซึ่งเลือกใช้ไมโครคอนโทรเลอร์ เบอร์ AT89C51AC3 ของ ATMEL เป็น MCU ประจำบอร์ด โดย MCU รุ่นนี้จะบรรจุอยู่ภายในตัวถังแบบ 52 Pin PLCC โดย MCU ตัวนี้จะมีจุดเด่น คือ เรื่องของความเร็วในการประมวลผล ซึ่งสามารถทำงานได้ด้วยความถี่สูงสุด 60MHz ที่ 12 Clock / 1 Machine Cycle นอกจากนี้แล้วยังมีความเพียบพร้อมด้วยอุปกรณ์พื้นฐานต่างๆที่จำเป็นต่อการใช้งาน ไม่ว่าจะเป็นหน่วยความจำสำหรับเก็บข้อมูลแบบ EEPROM ขนาด 2 KByte หรือหน่วยความจำใช้งานแบบ RAM ซึ่งมีมากถึง 2304 Byte (2048+256) ส่วนในด้านของอุปกรณ์ Peripheral นั้นก็นับว่าครบถ้วนเหมาะแก่การนำไปประยุกต์ใช้งานเกี่ยวกับการควบคุมและประมวลผลต่างๆได้เป็นอย่างดี โดยจะมีทั้ง SPI, UART, Watchdog, Timer/Counter, PWM และ ADC โดยการออกแบบโครงสร้างของบอร์ดนั้นจะเน้นเรื่องขนาดของบอร์ดให้มีขนาดเล็กเพื่อให้ง่ายต่อการนำไปประยุกต์ใช้งาน และสะดวกต่อการพัฒนาโปรแกรม

คุณสมบัติของบอร์ด

- เลือกใช้ MCU ตระกูล MCS51 เบอร์ AT89C51AC3 ของ ATMEL เป็น MCU ประจำบอร์ด โดยเลือกใช้แหล่งกำเนิดสัญญาณนาฬิกาแบบ Oscillator Module ค่า 29.4912 MHz ซึ่งสามารถกำหนดการทำงานของ MCU ให้ทำงานในโหมดความเร็ว 2 เท่า (X2 Mode) ได้ ทำให้ MCU สามารถประมวลผลด้วยความเร็วสูงสุดที่ 58.9824 MHz โดยคุณสมบัติเด่นๆของ MCU ได้แก่
  - มีหน่วยความจำ Flash สำหรับเขียนโปรแกรมขนาด 64KByte
  - มี EEPROM ขนาด 2KByte สำหรับเก็บข้อมูล และ สามารถเขียนซ้ำได้กว่า 1 ล้านครั้ง
  - มีพอร์ต I/O ขนาด 8 บิต จำนวน 5 พอร์ต (P0,P1,P2,P3 และ P4(5Bit))
  - มี RAM ใช้งาน 2304 Byte (ERAM 2048 Byte + IRAM 256 Byte)
  - มีวงจรสื่อสารอนุกรม UART จำนวน 1 พอร์ต และมีวงจรสื่อสาร SPI จำนวน 1 พอร์ต
  - มีวงจร Timer/Counter ขนาด 16 บิต จำนวน 3 ชุด
  - มีวงจร ADC ขนาด 10บิต จำนวน 8 ช่อง (ใช้ Port-P1 โดยกำหนดจากโปรแกรม)
  - มีวงจร Watchdog, Power-ON Reset, Capture/Compare ,PWM
- มีขั้วต่อสัญญาณ I/O แบบ TTL แบบ Header 2x5 จำนวน 5 ชุด (P0,P1,P2,P3 และ P4)
- มีขั้วต่อ LCD แบบ Header 2x7 รองรับการเชื่อมต่อกับ LCD Character (เชื่อมต่อแบบ 4 บิต)
- มีขั้วต่อใช้งาน RS232 สำหรับใช้งาน และ ET-DOWNLOAD สำหรับ Download ผ่าน RS232
- มี LED แสดงสถานะแหล่งจ่าย Power และ Self-Test สำหรับใช้ทดสอบการทำงานของบอร์ด
- ใช้แหล่งจ่ายไฟขนาด +5VDC
- ขนาด PCB Size เล็กเพียง 8 x 6 cm.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โครงสร้างบอร์ด ET-BASE51 AC3 (AT89C51AC3)



- หมายเลข 1 คือ ขั้วต่อแหล่งจ่ายไฟเลี้ยงวงจรของบอร์ด ใช้กับแหล่งจ่ายไฟตรง +5VDC
- หมายเลข 2 เป็น Switch PSEN ใช้ร่วมกับ RESET สำหรับ Download แบบ Manual
- หมายเลข 3 เป็น Switch RESET ใช้สำหรับ Reset การทำงานของ CPU
- หมายเลข 4 เป็น Port-P4 มี ขนาด 5 Bit คือ P4.0-P4.4
- หมายเลข 5 เป็น Port-P0 มี ขนาด 8 บิต
- หมายเลข 6 เป็น Port-P2 มี ขนาด 8 บิต
- **หมายเลข 7 เป็น ตัวต้านทานสำหรับปรับค่าความสว่างให้ LCD**
- **หมายเลข 8** เป็น Port-LCD ชนิด Character Type ใช้การเชื่อมต่อแบบ 4 บิต ผ่าน Port-P2
- **หมายเลข 9** เป็น Port-P3 มีขนาด 6 บิต (P3.2-P3.7)
- **หมายเลข 10** เป็น Port-P1 มีขนาด 8 บิต
- **หมายเลข 11** เป็นตัวต้านทานสำหรับปรับค่าแรงดันอ้างอิงของ ADC (3V)
- **หมายเลข 12** คือ MCU เบอร์ AT89C51AC3 ซึ่งเป็น MCU ตระกูล MCS51 จาก ATMEL

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

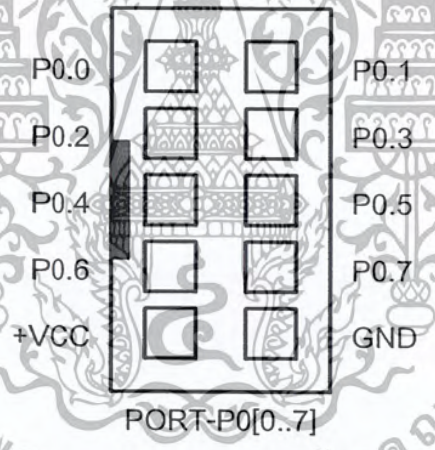
- **หมายเลข** 13 คือ Oscillator Module ค่า 29.4912 MHz
- **หมายเลข** 14 คือ ขั้วต่อ RS232 สำหรับใช้งานทั่วไป และ Download แบบ Manual
- **หมายเลข** 15 คือ ขั้วต่อ ET-DOWNLOAD ใช้สำหรับ Download แบบ Auto
- **หมายเลข** 16 เป็น LED Self Test (P1.0) ใช้สำหรับทดสอบการทำงานของบอร์ด
- **หมายเลข** 17 เป็น LED Power ใช้แสดงสถานะของแหล่งจ่ายไฟ +5VDC

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ขั้วต่อสัญญาณต่าง ๆ

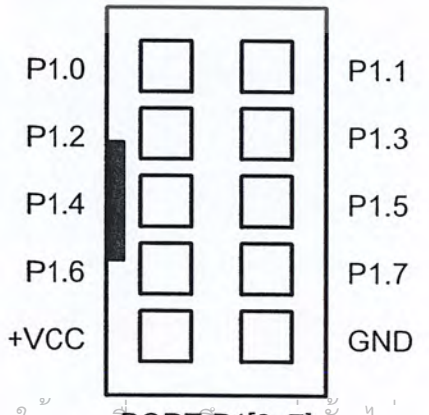
สำหรับขั้วต่อสัญญาณของพอร์ต I/O จาก MCU นั้น จะถูกออกแบบและจัดเตรียมไว้ผ่านทางขั้วต่อแบบ IDE Header ขนาด 10Pin (2x5) จำนวน 5 ชุด คือ PORT-P0, PORT-P1, PORT-P2,PORT-P3 และ PORT-P4 ตามลำดับ โดยที่ขั้วต่อสัญญาณแต่ละชุด จะประกอบไปด้วยสัญญาณของ I/O ที่เชื่อมต่อมาจากขาสัญญาณของ MCU โดยตรงทั้งหมด โดยแต่ละพอร์ตจะมีสัญญาณพอร์ตละ 8 บิต ยกเว้น PORT-P3 และ PORT-P4 โดย PORT-P4 มีขนาด 5 บิต (P4.0-P4.4) และ PORT-P3 ซึ่งจะมีเพียง 6 บิตเท่านั้น คือ P3.2-P3.7 ส่วน P3.0 และ P3.1 จะถูกสงวนไว้ใช้งานเป็นขาสัญญาณ RXD และ TXD สำหรับรับส่งข้อมูลของ RS232 ซึ่งสัญญาณทั้ง 2 เส้น (P3.0 และ P3.1) จะถูกเชื่อมต่อผ่านวงจร Line Driver (MAX232) สำหรับแปลงระดับสัญญาณจากระดับโลจิก TTL ของ MCU ให้เป็นสัญญาณแรงดันตามมาตรฐานของ RS232 โดยสัญญาณที่ได้รับการแปลงเป็นแบบ RS232 จะถูกเชื่อมต่อไปรอไว้ที่ขั้วต่อแบบ CPA ขนาด 4 PIN (RS232) โดยการจัดเรียงสัญญาณของแต่ละชุด จะเป็นดังรูป

พอร์ต P0 มีขนาด 8 บิต



PORT-P0[0..7]

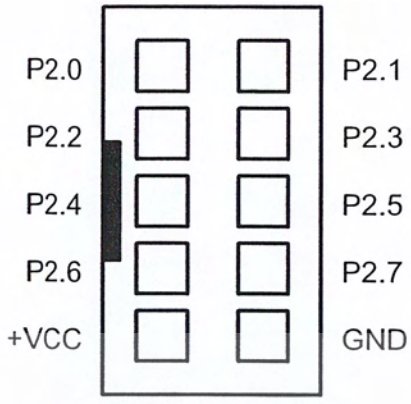
พอร์ต P1 มีขนาด 8 บิต



PORT-P1[0..7]

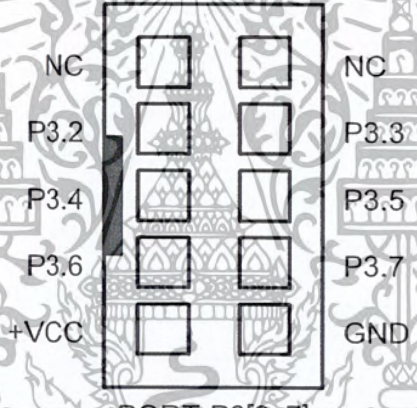
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

พอร์ต P2 มีขนาด 8 บิต



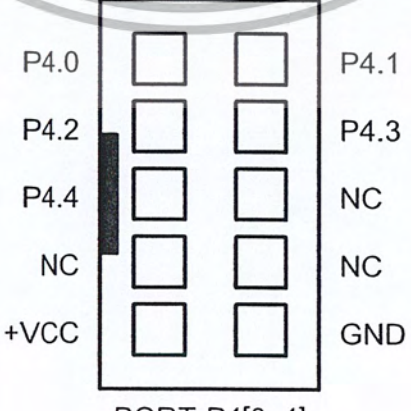
PORT-P2[0..7]

พอร์ต P3 มีขนาด 6 บิต



PORT-P3[2..7]

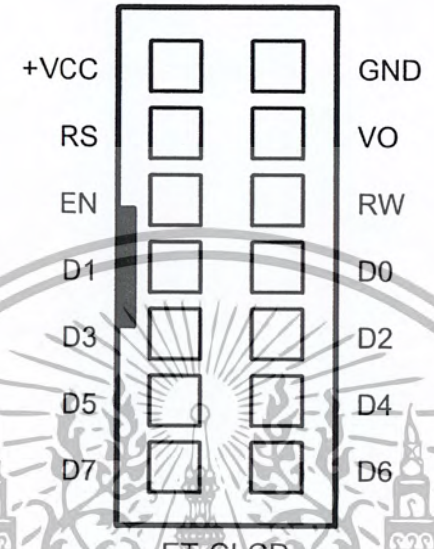
พอร์ต P4 มีขนาด 5 บิต

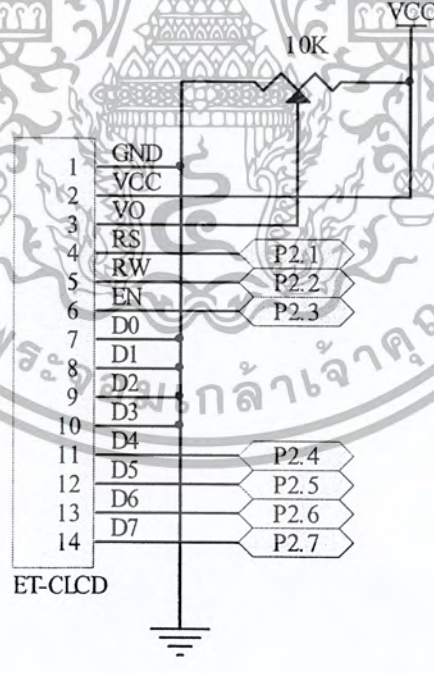


PORT-P4[0..4]

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

พอร์ต CLCD ใช้กับ Character LCD โดยเชื่อมต่อแบบ 4 บิต โดยสัญญาณที่ใช้เชื่อมต่อกับ LCD จะเป็นสัญญาณชุดเดียวกับที่ต่อไปยังขั้วต่อของ PORT-P2 โดยในการเชื่อมต่อสายสัญญาณจากขั้วต่อของพอร์ต LCD ไปยังจอแสดงผล LCD นั้น ให้ยึดชื่อสัญญาณเป็นจุดอ้างอิง โดยให้ต่อสัญญาณที่มีชื่อตรงกันเข้าด้วยกันให้ครบทั้ง 14 เส้น



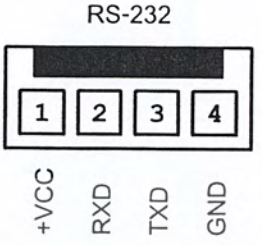


1	2	3	4	5	6	7	8	9	10	11	12	13	14
GND	+VCC	VO	RS	RW	EN	D0	D1	D2	D3	D4	D5	D6	D7

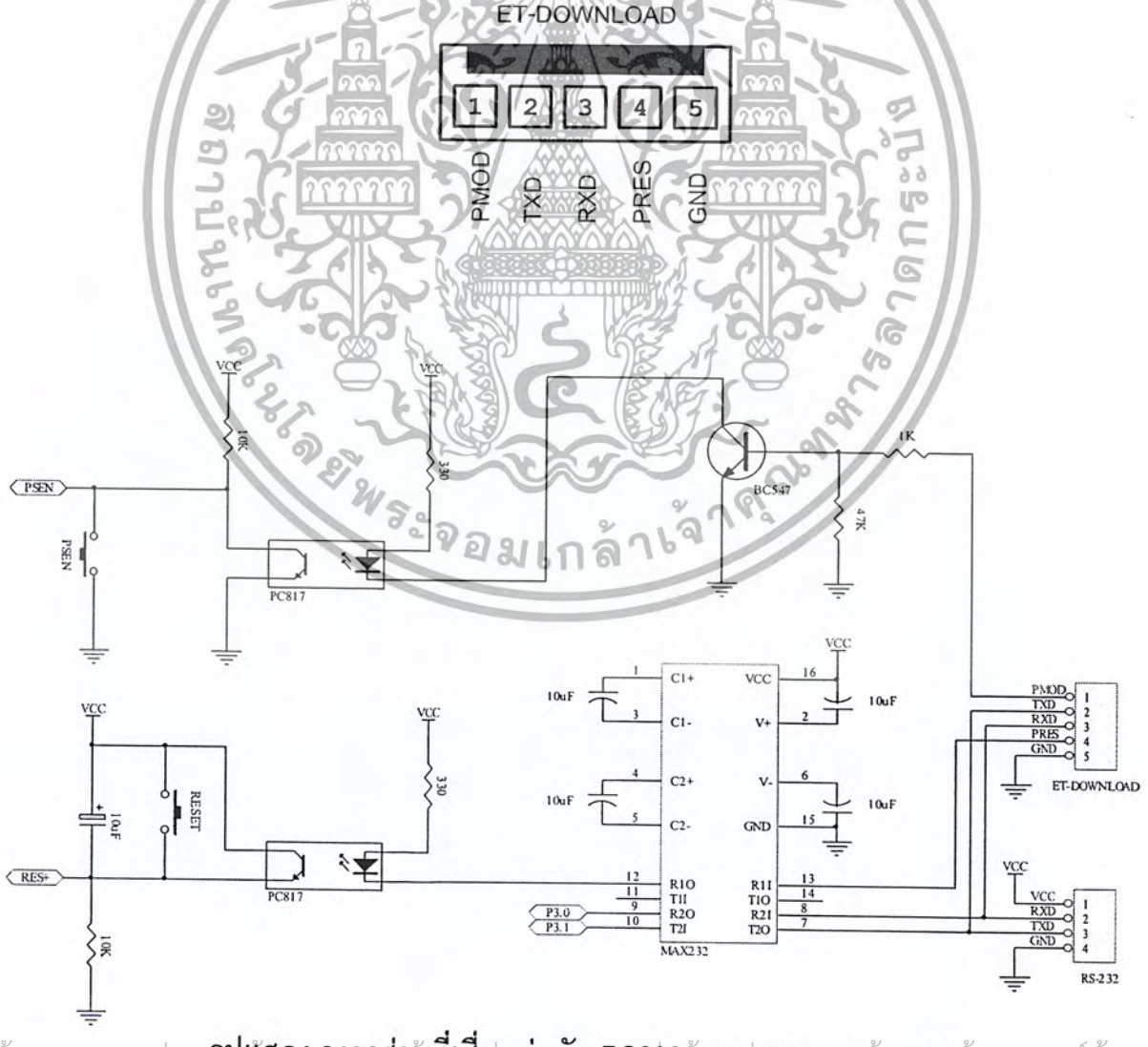
แสดงการจัดเรียงขาสัญญาณของ Character LCD มาตรฐาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

พอร์ต RS232 เป็นสัญญาณ RS232 ซึ่งผ่านวงจรแปลงระดับสัญญาณ MAX232 เรียบร้อยแล้ว สามารถใช้เชื่อมต่อกับสัญญาณ RS232 เพื่อรับส่งข้อมูล นอกจากนี้แล้วยังสามารถใช้งาน ร่วมกับ Switch PSEN และ Switch RESET เพื่อทำการ Download แบบ Manual ได้ด้วย



พอร์ต ET-DOWNLOAD เป็นขั้วต่อสำหรับใช้ Download Hex File ให้กับ MCU แบบ**อัตโนมัติ** โดยใช้งานร่วมกับโปรแกรม FLIP V2.4.4 ของ ATMEL



รูปแสดง วงจรส่วนที่เชื่อมต่อกับ RS232 **และ** ET-DOWNLOAD

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การ Download Hex file ให้กับ MCU ของบอร์ด

การ Download Hex File ให้กับหน่วยความจำ Flash ของ MCU ในบอร์ดนั้น จะใช้โปรแกรมชื่อ "FLIP" ของ ATMEL ซึ่งจะติดต่อกับ MCU ผ่าน Serial Port ของคอมพิวเตอร์ PC โดยโปรแกรมดังกล่าวสามารถดาวน์โหลดได้จาก WWW.ATMEL.COM โดยไม่เสียค่าใช้จ่ายใดๆ แต่สำหรับในกรณีที่ซื้อบอร์ดจาก อีทีที นั้น โปรแกรมดังกล่าวจะจัดเตรียมไว้ให้ในแผ่น CD ROM อยู่แล้ว

โปรแกรม FLIP (Flexible In-system Programmer) เป็นโปรแกรมสำหรับพัฒนาระบบของไมโครคอนโทรลเลอร์ของ ATMEL โดยสามารถใช้สนับสนุนการพัฒนาโปรแกรมของไมโครคอนโทรลเลอร์ตระกูล MCS51 ในกลุ่มที่ใช้การพัฒนาแบบ ISP ซึ่งรวมถึงเบอร์ AT89C51AC3 ด้วย โดยโปรแกรมจะทำงานภายใต้ระบบปฏิบัติการของ Windows9X/Me/NT/2000 และ Windows XP โดยสนับสนุนการเชื่อมต่อกับระบบฮาร์ดแวร์ที่ใช้การเชื่อมต่อแบบ RS232 หรือ CAN หรือ USB ซึ่งวิธีการเชื่อมต่อของโปรแกรม FLIP กับระบบฮาร์ดแวร์ของไมโครคอนโทรลเลอร์นั้น จะขึ้นอยู่กับความสามารถของตัวไมโครคอนโทรลเลอร์ที่จะนำมาทำการพัฒนาว่าสามารถใช้การติดต่อสื่อสารด้วยวิธีใดได้บ้าง แต่สำหรับไมโครคอนโทรลเลอร์เบอร์ AT89C51AC3 นั้นจะสามารถใช้การเชื่อมต่อผ่านทางพอร์ตอนุกรม RS232 เท่านั้น ไม่สามารถเชื่อมต่อผ่านระบบการสื่อสารของ CAN หรือ USB ได้ โดยโปรแกรม FLIP จะใช้สำหรับ Download ข้อมูลให้กับหน่วยความจำของไมโครคอนโทรลเลอร์ที่ทำงานใน Monitor Mode เพื่อให้ผู้ใช้สั่งจัดการกับหน่วยความจำภายในตัว CPU ไม่ว่าจะเป็นการ สั่งลบข้อมูล(Erase) สั่งตรวจสอบข้อมูลในหน่วยความจำ(Blank Check) สั่งโปรแกรมข้อมูลให้กับหน่วยความจำโปรแกรมของ CPU (Program) สั่งเปรียบเทียบข้อมูลจาก Buffer กับหน่วยความจำในตัว CPU (Verify) หรือสั่งอ่านข้อมูลจากหน่วยความจำของ CPU (Read) เป็นต้น

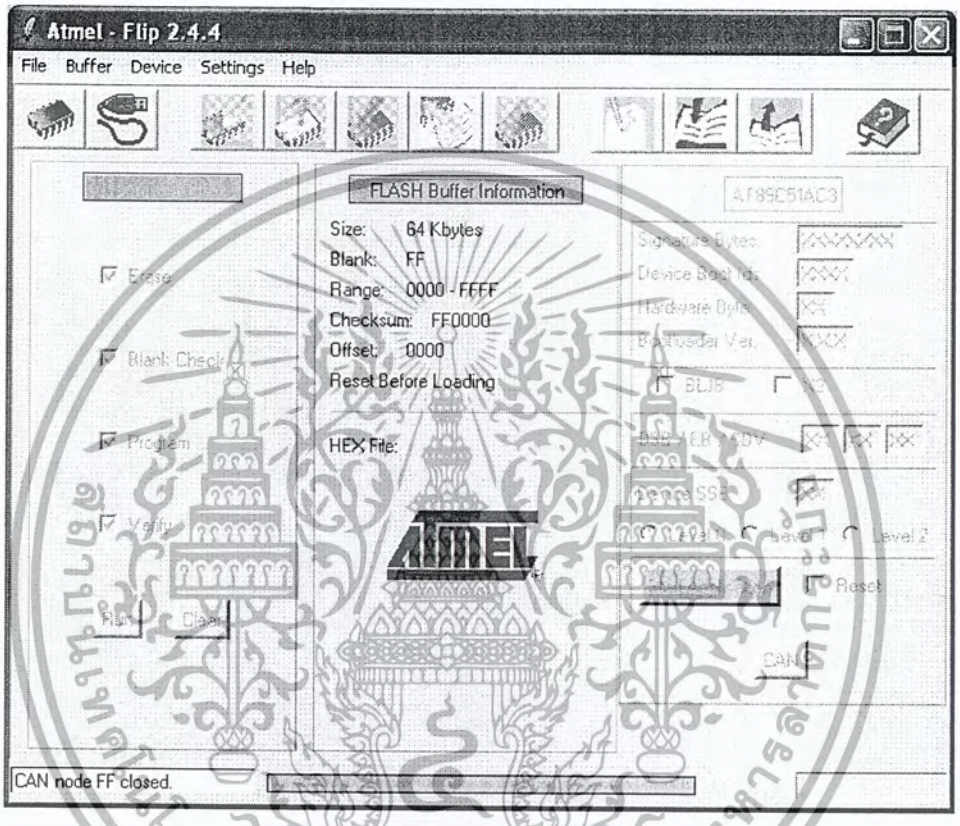
ซึ่งเมื่อต้องการให้โปรแกรม FLIP ติดต่อกับ CPU ใน Monitor Mode นั้น จะต้องสั่ง Reset ให้ CPU เข้าทำงานใน Monitor Mode ก่อนเสียก่อน ซึ่งหลักการสำหรับ Reset ให้ CPU เข้าทำงานใน Monitor Mode จะต้องกำหนดให้ขาสัญญาณ PSEN มีสภาวะเป็น "0" ในขณะที่ CPU หลุดพ้นจากสภาวะของการ Reset ซึ่งตามปรกติแล้วหลังการ Reset ทุกครั้ง CPU จะตรวจสอบสภาวะของขาสัญญาณ PSEN ว่าเป็น "0" หรือไม่ถ้าไม่ใช่ก็จะทำงานในโหมดการทำงานปรกติแต่ถ้าใช่ก็จะตรวจสอบสถานะของสัญญาณอื่นๆที่เกี่ยวข้องกับการทำงานใน Monitor Mode ถ้าเงื่อนไขอื่นๆถูกต้องก็จะเข้าทำงานใน Monitor Mode ทันที สำหรับบอร์ด รุ่น ET-BASE51 AC3 (AT89C51AC3) นั้น การที่จะสั่ง Reset ให้ CPU ของ ATMEL เข้าทำงานใน Monitor Mode เพื่อสั่ง Download HEX File จาก PC ให้กับบอร์ดจะสามารถทำได้ 2 แบบ คือ

- การ Download แบบ Manual โดยวิธีการนี้จะใช้กับสาย RS232 แบบ 4 Pin ร่วมกับ Switch PSEN และ Switch RESET ในการสั่ง Download
- การ Download แบบ Auto โดยวิธีการนี้ จะใช้สาย ET-DOWNLOAD แบบ 5 Pin ในการสั่ง Download

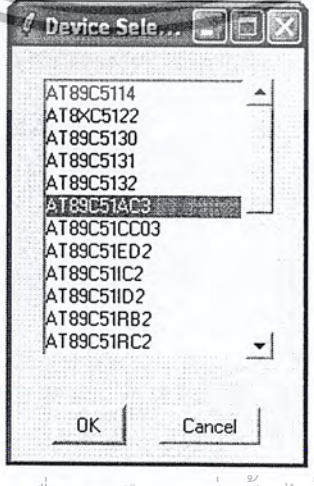
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ลำดับขั้นตอนการ Download HEX File ด้วยโปรแกรม FLIP 2.4.4 แบบ Manual

1. ต่อสายสัญญาณ RS232 จาก Com Port ของเครื่องคอมพิวเตอร์ PC เข้ากับขั้ว RS232 แบบ 4 Pin ของบอร์ด
2. จ่ายไฟเลี้ยงวงจรให้บอร์ด ซึ่งจะสังเกตุเห็น LED แสดงสถานะของ PWR สีแดงติดสว่างอยู่
3. สั่ง Run โปรแกรม FLIP V2.4.4 ซึ่งจะได้ผลดังรูป



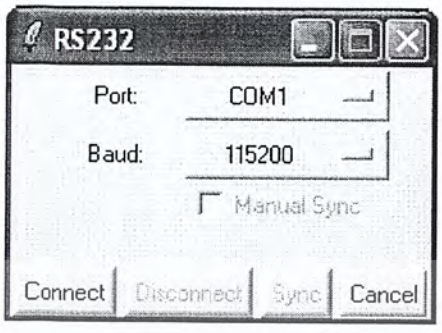
4. สั่งเลือกกำหนดเบอร์ของ MCU ที่ติดตั้งไว้ในบอร์ด โดยเลือก Device ⟶ Select ซึ่งต้องเลือกกำหนดให้ตรงกับที่ทำการติดตั้งไว้จริงๆในบอร์ดด้วย ดังตัวอย่าง (AT89C51AC3)



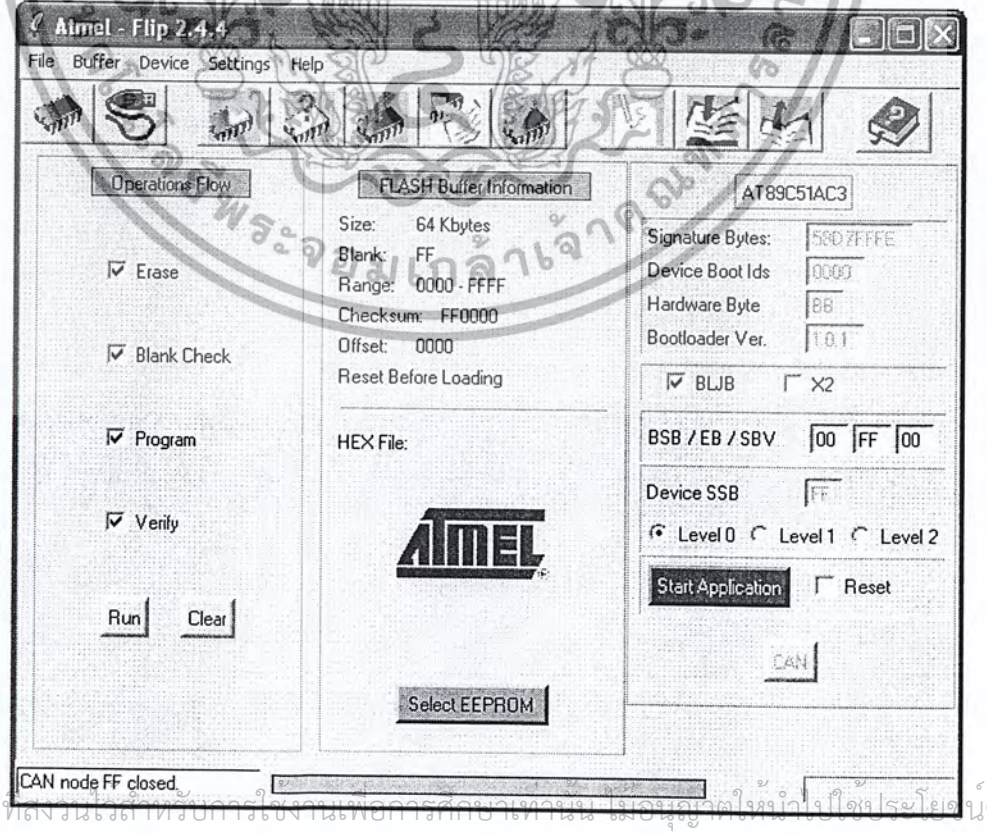
รูปแสดง การเลือกกำหนดเบอร์ CPU ของ ET-BASE51 AC3 (AT89C51AC3)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5. คลิกเมาส์ที่คำสั่ง Setting ⟶ Communication ⟶ RS232 จากนั้นเลือกกำหนด Comport ให้ตรงกับที่ต่อสายไว้จริง ดังรูป (ในตัวอย่างใช้ Com1)

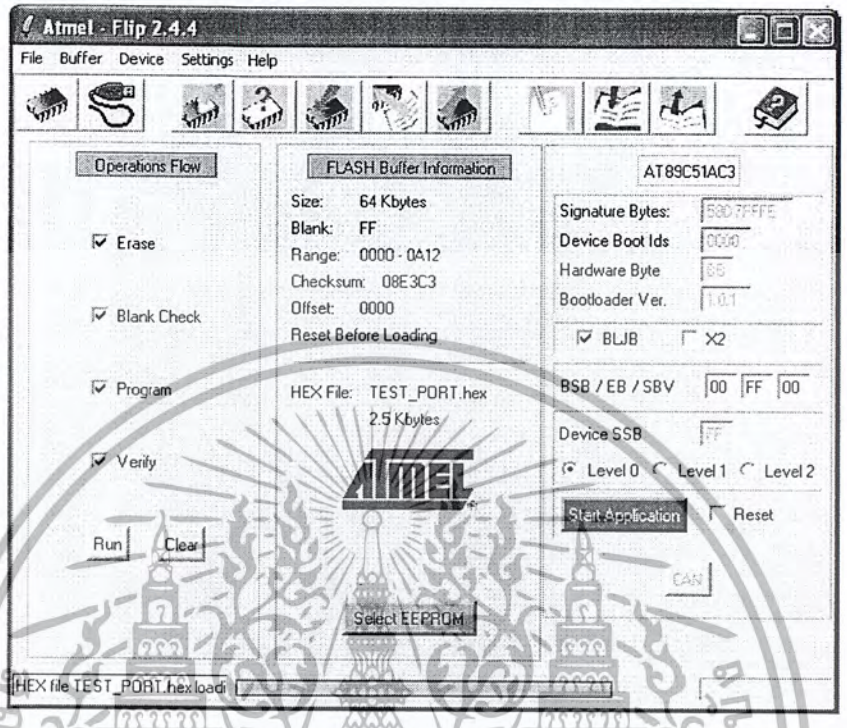


6. ทำการรีเซ็ต MCU ให้เข้าทำงานใน Monitor โดยมีลำดับขั้นตอนดังนี้
   a) กดสวิตช์ PSEN ค้างไว้เพื่อกำหนดสถานะขาสัญญาณ PSEN ให้เป็น "0"
   b) กดสวิตช์ RESET เพื่อส่งสัญญาณ RESET ให้กับ CPU โดยสวิตช์ PSEN ต้องกดค้างอยู่เช่นเดิม
   c) ปล่อยสวิตช์ RESET เพื่อปล่อยให้ CPU พ้นจากสภาวะการ Reset (สวิตช์ PSEN ยังกดค้างอยู่)
   d) ปล่อยสวิตช์ PSEN เป็นลำดับสุดท้าย
7. คลิกเมาส์ที่ปุ่ม Connect เพื่อทำการติดต่อสื่อสารกับ MCU ใน Monitor Mode ซึ่งจะได้ผลดังรูป

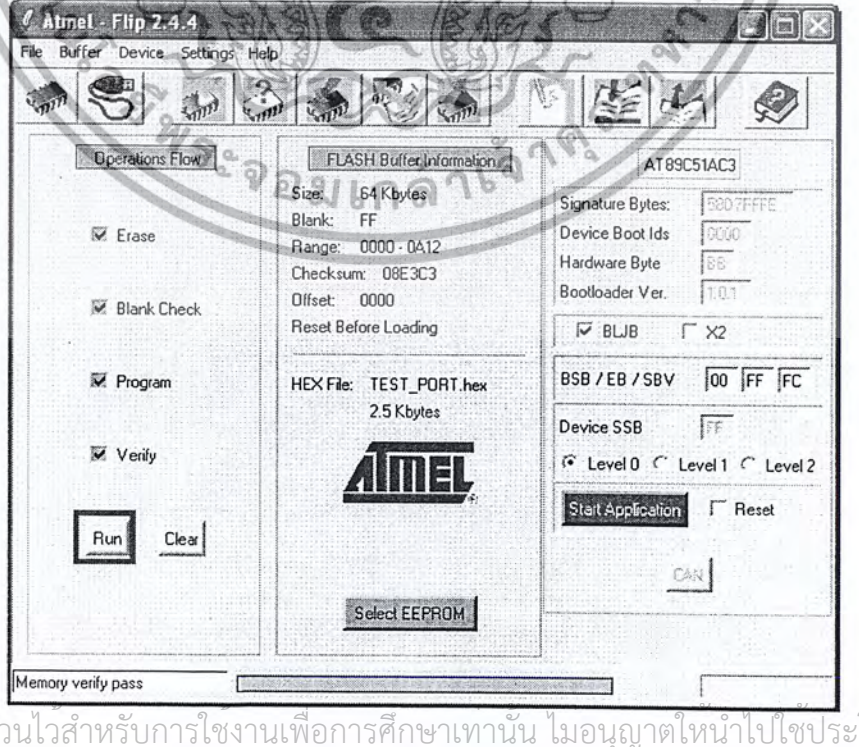


เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

8. สั่งเปิด Hex File ที่ต้องการจะ Download ให้กับ MCU มารอไว้ใน Buffer ของโปรแกรม FLIP โดยใช้คำสั่ง "File ⟶ Load Hex File..."

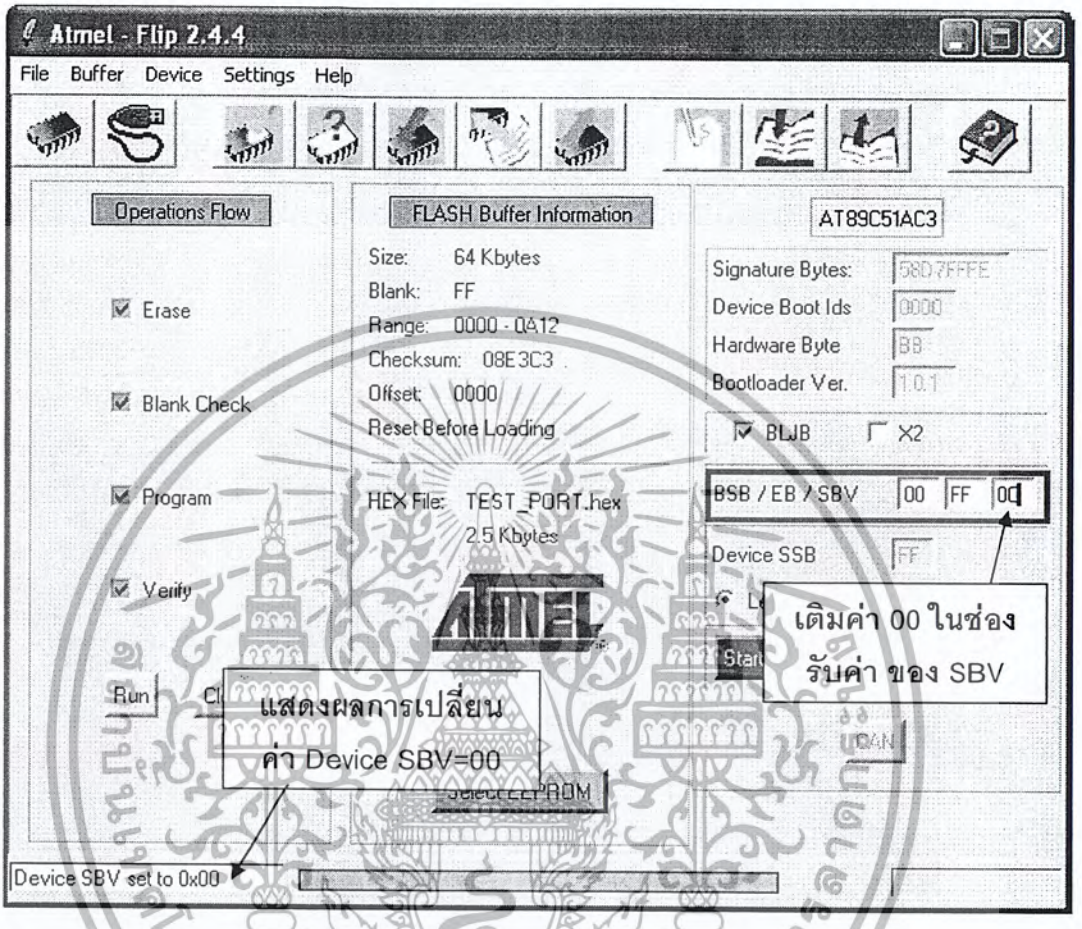


9. คลิกเมาส์ที่หน้าตัวเลือกคำสั่งใน Tab ของ Operation Flow ให้ครบทุกคำสั่ง ซึ่งได้แก่ Erase, Bank Check, Program, Verify จากนั้นคลิกเมาส์ที่ปุ่มคำสั่ง Run และรอจนการทำงานของโปรแกรมเสร็จเรียบร้อยดังรูป



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

10. ตรวจสอบค่า Device BSB และ SBV ว่ามีค่าเป็น 00 ทั้งหมดแล้วหรือยัง ซึ่งถ้ายังไม่เป็น 00 ให้ทำการแก้ไขค่าให้เป็น 00 โดยคลิกเมาส์ในช่องตัวเลขแล้วพิมพ์ค่า 00 แทนที่ลงไปทั้ง 2 ช่องดังรูป

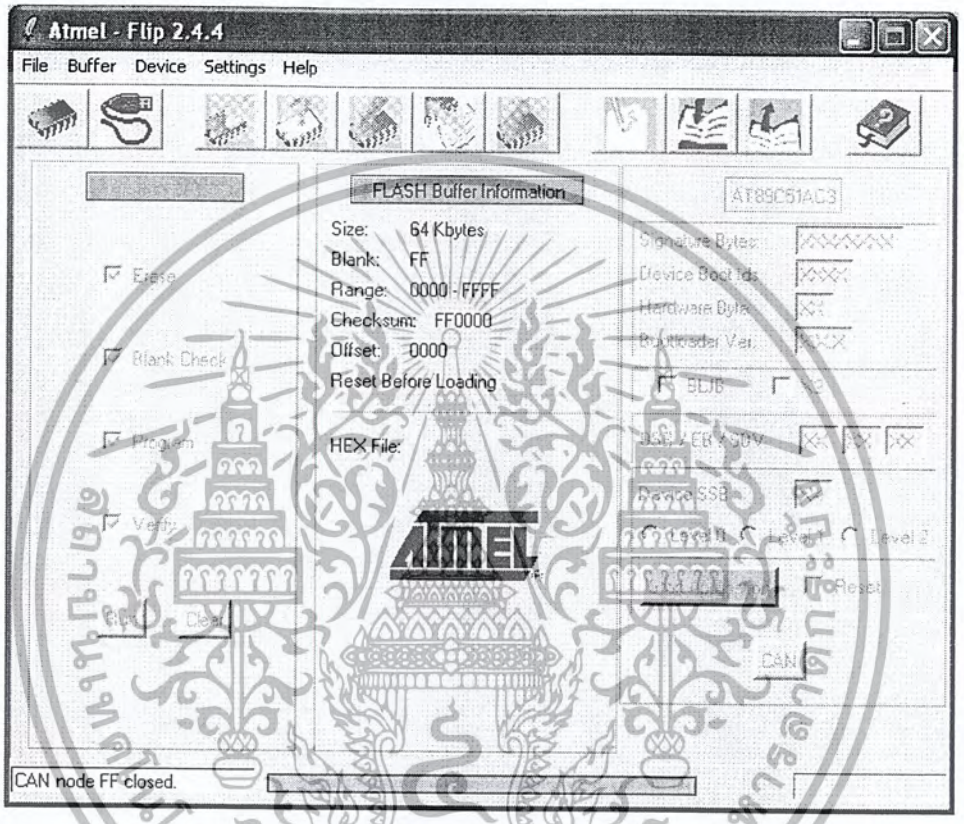


11. ทำการกดสวิตช์ Reset ให้กับบอร์ดเพื่อให้บอร์ดเริ่มต้นทำงานตามโปรแกรมที่ได้ทำการ Download ไปให้ ซึ่งถ้าไม่เกิดความผิดพลาดใดๆจะเห็น MCU เริ่มต้นทำงานทันที

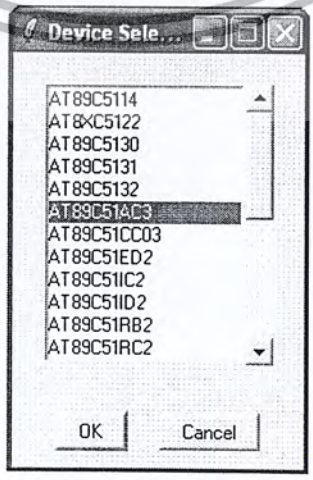
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ลำดับขั้นตอนการ Download HEX File ด้วยโปรแกรม FLIP 2.4.4 แบบ Auto Download

1. ต่อสายสัญญาณ RS232 จาก Com Port ของเครื่องคอมพิวเตอร์ PC เข้ากับขั้ว ET-DOWNLOAD แบบ 5 Pin ของบอร์ด
2. จ่ายไฟเลี้ยงวงจรให้บอร์ด ซึ่งจะสังเกตุเห็น LED แสดงสถานะของ PWR สีแดงติดสว่างอยู่
3. สั่ง Run โปรแกรม FLIP V2.4.4 ซึ่งจะได้ผลดังรูป



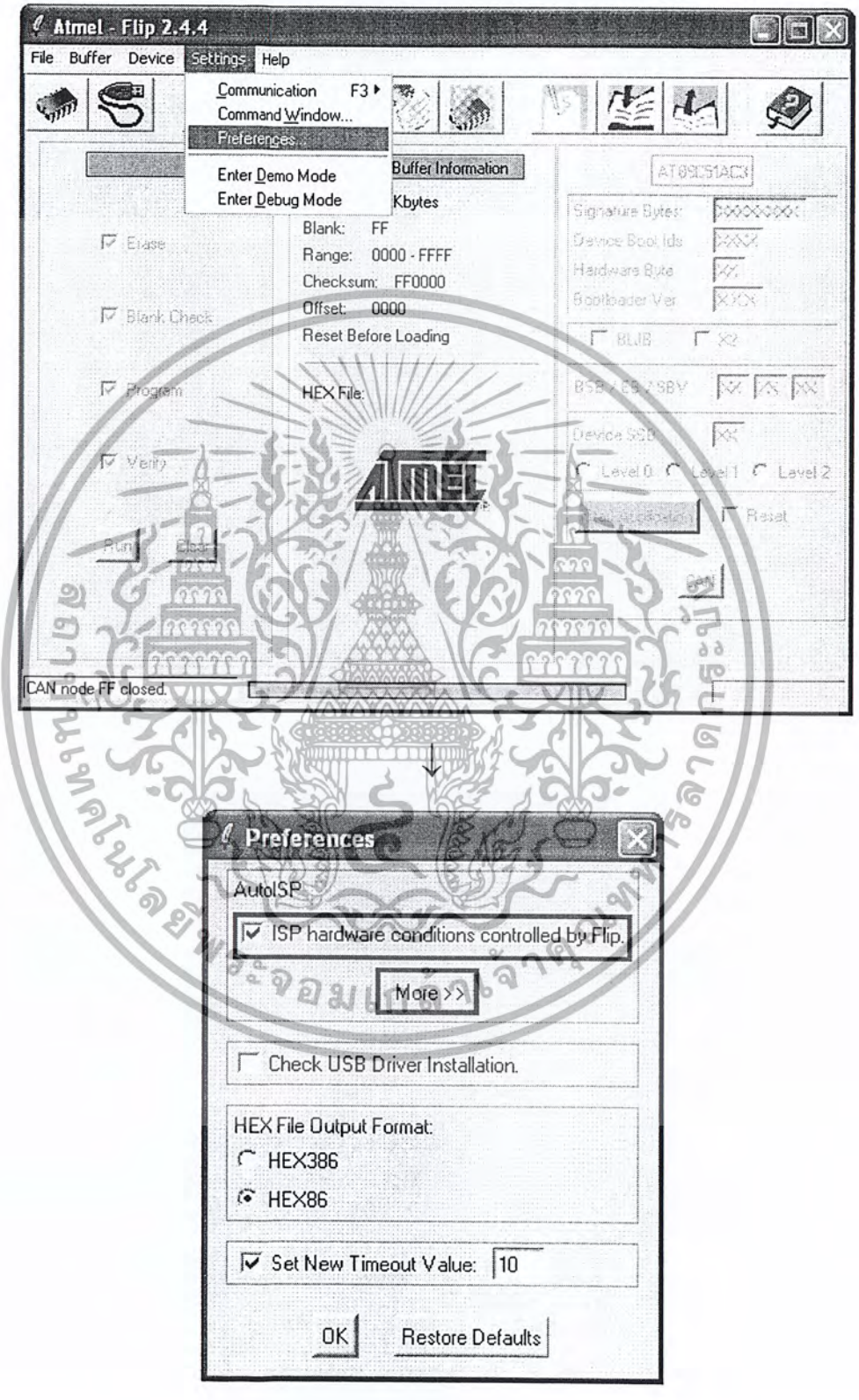
4. สั่งเลือกกำหนดเบอร์ของ MCU ที่ติดตั้งไว้ในบอร์ด โดยเลือก Device ⟶ Select ซึ่งต้องเลือกกำหนดให้ตรงกับที่ทำการติดตั้งไว้จริงๆในบอร์ดด้วย ดังตัวอย่าง (AT89C51AC3)



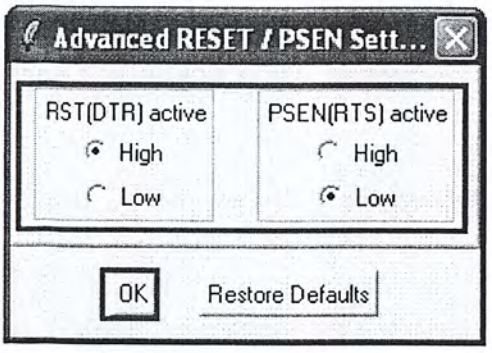
รูปแสดง การเลือกกำหนดเบอร์ CPU ของ ET-BASE51 AC3 (AT89C51AC3)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

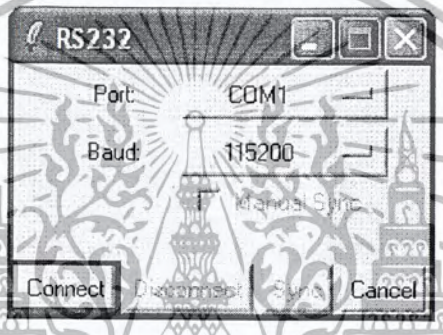
5. ทำการกำหนดค่า Option ของการสื่อสาร RS232 สำหรับใช้ Download แบบอัตโนมัติ โดยให้เลือกคลิกเมาส์ที่ Setting → Preferences... แล้วเลือกกำหนดค่าดังรูป



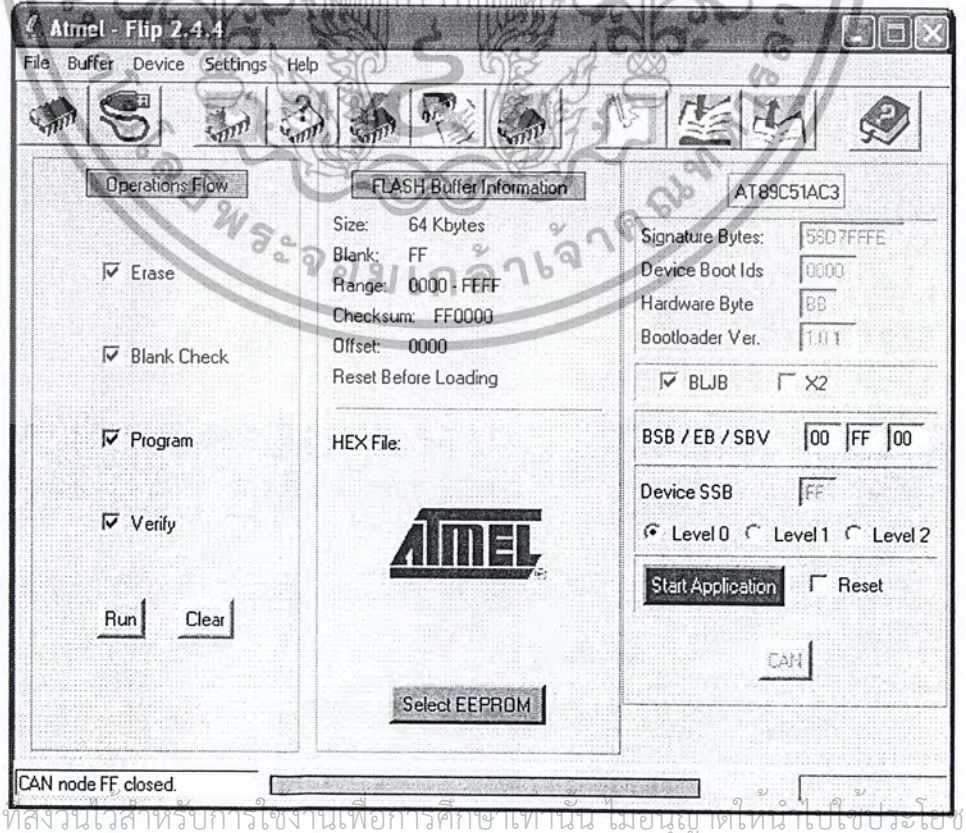
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



6. คลิกเมาส์ที่คำสั่ง Setting ⟶ Communication ⟶ RS232 จากนั้นเลือกกำหนด Comport ให้ตรงกับที่ต่อสายไว้จริง ดังรูป (ในตัวอย่างใช้ Com1)

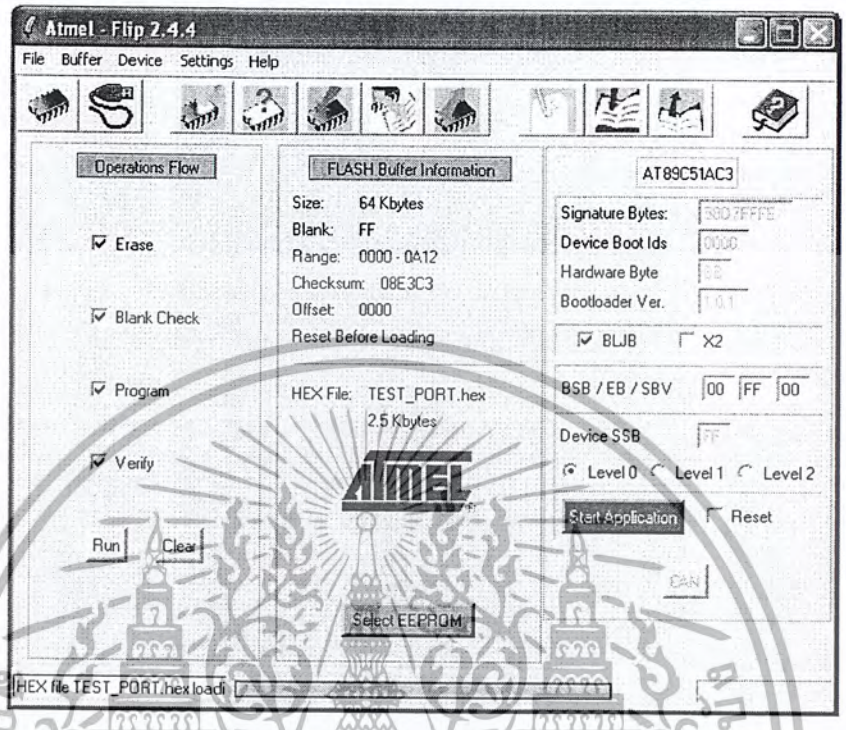


7. คลิกเมาส์ที่ปุ่ม Connect เพื่อทำการติดต่อสื่อสารกับ MCU ใน Monitor Mode ซึ่งจะได้ผลดังรูป

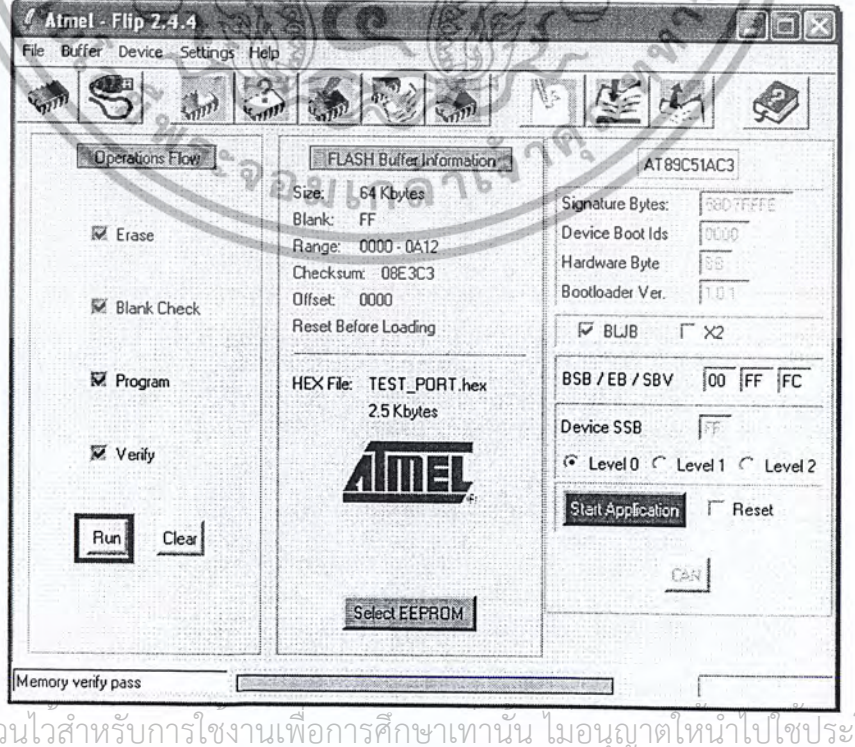


เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

8. สั่งเปิด Hex File ที่ต้องการจะ Download ให้กับ MCU มารอไว้ใน Buffer ของโปรแกรม FLIP โดยใช้คำสั่ง "File ⟶ Load Hex File..."

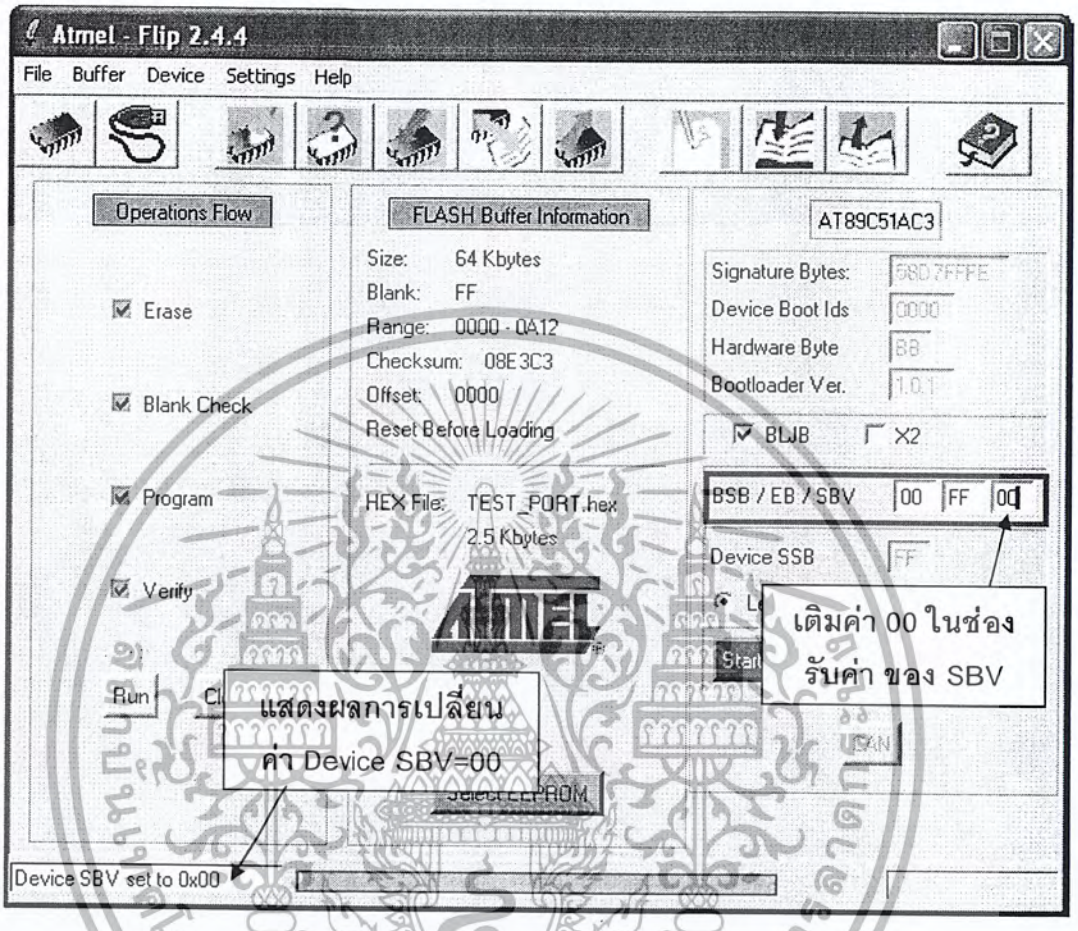


9. คลิกเมาส์ที่หน้าตัวเลือกคำสั่งใน Tab ของ Operation Flow ให้ครบทุกคำสั่ง ซึ่งได้แก่ Erase, Bank Check, Program, Verify จากนั้นคลิกเมาส์ที่ปุ่มคำสั่ง Run และรอจนการทำงานของโปรแกรมเสร็จเรียบร้อยดังรูป



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

10. ตรวจสอบค่า Device BSB และ SBV ว่ามีค่าเป็น 00 ทั้งหมดแล้วหรือยัง ซึ่งถ้ายังไม่เป็น 00 ให้ทำการแก้ไขค่าให้เป็น 00 โดยคลิกเมาส์ในช่องตัวเลขแล้วพิมพ์ค่า 00 แทนที่ลงไปทั้ง 2 ช่องดังรูป



11. ทำการคลิกเมาส์ที่ "Start Application" หรือกดสวิตช์ Reset ให้กับบอร์ดเพื่อให้บอร์ดเริ่มต้นทำงานตามโปรแกรมที่ได้ทำการ Download ไปให้ ซึ่งถ้าไม่เกิดความผิดพลาดใดๆจะเห็น MCU เริ่มต้นทำงานทันที

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ปัญหาต่าง ๆในขณะใช้งานโปรแกรม FLIP และแนวทางการแก้ไข

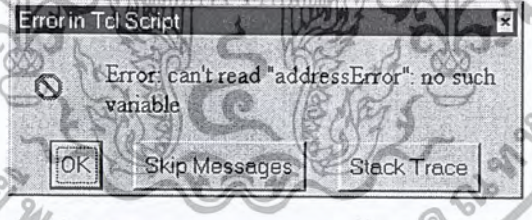
ในบางครั้งเมื่อเรียกใช้คำสั่งต่างๆของโปรแกรม FLIP แล้ว อาจเกิดความผิดพลาดบางประการขึ้น ซึ่งอาจไม่ใช่ปัญหาที่เกิดจากความบกพร่องของระบบฮาร์ดแวร์ แต่อาจเกิดการการกำหนดพารามิเตอร์บางอย่างในโปรแกรมไม่ถูกต้องหรือข้ามขั้นตอนบางประการไป ซึ่งเมื่อโปรแกรม FLIP ไม่สามารถปฏิบัติตามคำสั่งที่ผู้ใช้งานสั่งไปได้สำเร็จจะแสดงอาการ Error ต่างๆให้ทราบ ซึ่งพอสรุปได้ดังนี้

1. Time Out Error เป็นความผิดพลาดที่เกิดจากการที่โปรแกรม FLIP ไม่สามารถทำการสื่อสารกับ CPU ใน Monitor Mode ได้ ซึ่งอาจเกิด หลายสาเหตุ เช่น
   - การต่อสายสัญญาณระหว่างขั้วต่อ RS232 ของบอร์ด ET-BASE51 AC3 กับขั้วต่อพอร์ตสื่อสารอนุกรม RS232 ของคอมพิวเตอร์ยังไม่เรียบร้อยหรือต่อไม่ตรงกับที่กำหนดตัวเลือกไว้ในโปรแกรม หรือการกำหนดรูปแบบและตัวเลือกต่างๆในการสื่อสารไม่ถูกต้อง เมื่อพบปัญหานี้ให้ลองทำการตรวจสอบค่าต่างๆในการสื่อสารใน "Setting → Preferences.. และ Setting → Communication → RS232"
   - ยังไม่ได้รีเซ็ตให้ CPU เข้าทำงานใน Monitor Mode รอไว้ก่อนที่จะสั่งงานโปรแกรมในกรณี Download แบบ Manual หรือบอร์ดยังไม่พร้อมทำงาน เช่น ยังไม่ได้จ่ายไฟเลี้ยงให้บอร์ด
   - กำหนดค่า Baudrate เร็วเกินไป ซึ่งในกรณีที่ใช้งานกับเครื่องคอมพิวเตอร์ที่มีความเร็วมากๆ นั้น ควรกำหนดค่า Baudrate ในการสื่อสารให้ช้าลง ซึ่งอาจใช้ค่า 19200 หรือ 9600 ก็พอ เพราะถ้ากำหนดให้ความเร็วมากเกินไป เมื่อโปรแกรม FLIP ส่งข้อมูลให้กับ CPU แบบต่อเนื่องนั้น อาจทำให้ CPU ไม่สามารถประมวลผลคำสั่งหรือข้อมูลต่างๆที่ส่งไปให้ทันก็จะทำให้เกิดความผิดพลาดบ่อยครั้งขึ้น
2. Software Security Bit Set. Cannot access device Data เป็นความผิดพลาดที่เกิดจากการนำ CPU ที่มีการสั่ง Lock Bit ของ Security Bit ไว้ก่อนแล้ว จึงมาสั่ง Program หรือ Verify หรือ Read ในภายหลังโดยยังไม่ได้สั่งลบข้อมูลเก่าออกเสียก่อน ซึ่งให้แก้ปัญหาด้วยการสั่งลบข้อมูล (Erase) เสียก่อนแล้วจึงสั่งเขียนข้อมูลใหม่อีกครั้งหนึ่ง
3. The board reply is not correct เป็นความผิดพลาดที่เกิดจากการสื่อสารข้อมูลระหว่างโปรแกรม FLIP กับ ไมโครคอนโทรลเลอร์ เกิดความผิดพลาดในลักษณะของ Framing Error ขึ้น ซึ่งปัญหาอาจเกิดจากการกำหนดค่า Baudrate ไม่ถูกต้องกับค่าความถี่ของ Crystal ที่ใช้กับบอร์ด
4. The RS232 port could not be opened เป็นความผิดพลาดที่เกิดจากโปรแกรม FLIP ไม่สามารถสั่งเปิดการทำงานของพอร์ตสื่อสารอนุกรม RS232 ของเครื่องคอมพิวเตอร์ PC ได้ ซึ่งอาจเกิดจากการกำหนดหมายเลข Comport ในโปรแกรมที่เลือกไว้ไม่มีอยู่จริง หรือมีโปรแกรมอื่นเรียกใช้งาน Comport นั้นค้างอยู่ หรือเรียกใช้งานโปรแกรม FLIP ในขณะที่กำลังสั่งเปิดโปรแกรมอื่นๆที่มีการใช้งาน Comport

เอกสารนี้เป็นลิขสิทธิ์ของ ETT CO.,LTD ห้ามมิให้ทำการคัดลอก ดัดแปลง เผยแพร่ หรือนำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

อยู่ด้วย ซึ่งให้ลองปิดโปรแกรม FLIP แล้วสั่งเปิดโปรแกรมใหม่ดู ถ้ายังเกิดปัญหาเดิมอยู่อีกอาจลองตรวจสอบสาเหตุอื่นๆที่เกี่ยวข้องและทำการแก้ไข

5. Check sum error เป็นความผิดพลาดที่เกิดจากการที่ CPU รับข้อมูลที่ส่งไปจากคอมพิวเตอร์ PC ไม่ครบถูกต้องทั้งหมด ซึ่งปัญหาอาจเกิดจากการกำหนดความเร็วในการสื่อสาร Baudrate เร็วเกินไป หรือกำหนดไว้ไม่เหมาะสมกับค่าความถี่ Crystal ให้ลองเปลี่ยนค่า Baudrate ให้ช้าลงกว่าเดิม ซึ่งค่าที่เหมาะสมได้แก่ 9600,19200 และ 38400 แต่ถ้าคอมพิวเตอร์ไม่เร็วมากนักก็อาจกำหนดเป็น 57600 หรือ 115200 ก็ได้ แต่ถ้ากำหนดค่าสูงๆแล้วเกิด Error ควรลดค่า Baudrate ให้ช้าลงกว่าเดิม
6. การสั่ง Load HEX ไม่ได้ เป็นความผิดพลาดที่เกิดจากการที่โปรแกรม FLIP ไม่สามารถอ่านข้อมูลใน HEX File ออกมาได้ ซึ่งอาจเกิดจากไฟล์ที่สั่งโหลดนั้น ไม่ใช่ไฟล์แบบ Intel HEX เนื่องจากโปรแกรม FLIP สามารถใช้งานกับไฟล์แบบ Intel HEX เท่านั้น ส่วนไฟล์ในรูปแบบอื่นๆจะไม่สามารถนำมาใช้งานกับโปรแกรมนี้ได้ ส่วนปัญหาอีกประการหนึ่งที่มักพบอยู่บ่อยๆ คือโปรแกรม FLIP ไม่สามารถอ่าน HEX File ได้ทั้งๆที่ไฟล์ที่สั่งให้อ่านนั้นเป็นไฟล์แบบ Intel HEX อยู่แล้ว ซึ่งที่พบอยู่บ่อยๆก็ได้แก่ HEX File ที่สั่งแปลโดยใช้โปรแกรม Assembler ของ SXA51.EXE เนื่องจาก HEX File ที่ได้จากการแปลของโปรแกรมตัวนี้จะเกิดบรรทัดว่างอยู่ในไฟล์ในส่วนเริ่มต้นบรรทัดแรกด้วย 1 บรรทัด ซึ่งตามรูปแบบของ HEX File แล้ว ในแต่ละบรรทัดของไฟล์จะต้องเริ่มต้นด้วยเครื่องหมายโคลอน (:) แล้วตามด้วยข้อมูลต่างๆในบรรทัดนั้น แต่เมื่อบรรทัดแรกเป็นบรรทัดว่างโปรแกรมจึงแสดง Error ว่าไม่ใช่ HEX File โดยโปรแกรม FLIP จะแสดง Error ให้ทราบดังนี้



สำหรับวิธีการแก้ไขปัญหานี้ให้ใช้โปรแกรม Text Editor เปิด HEX File ที่ได้จากการแปลของ SXA51.EXE แล้วตัดบรรทัดว่างในไฟล์นั้นทิ้งไปแล้วสั่งบันทึกใหม่ก็จะสามารถนำไปใช้ได้แล้ว

```
:10000000758921759850D2BCC2ACC2A9C2AB74FB31
:1000                                AF1200880D0A0AE4
:1000                                043502D4A52359A
:10003000314143322056312026205632202831329 9
:100040002D436C6F636B290D0A507265737320 41E9
:00000001FF
```

บรรทัดว่างที่ได้จากการแปลของโปรแกรม SXA51

รูปแสดง ลักษณะของ HEX File ที่ได้จาก SXA51 ซึ่งจะเกิดบรรทัดว่างอยู่ 1 บรรทัด

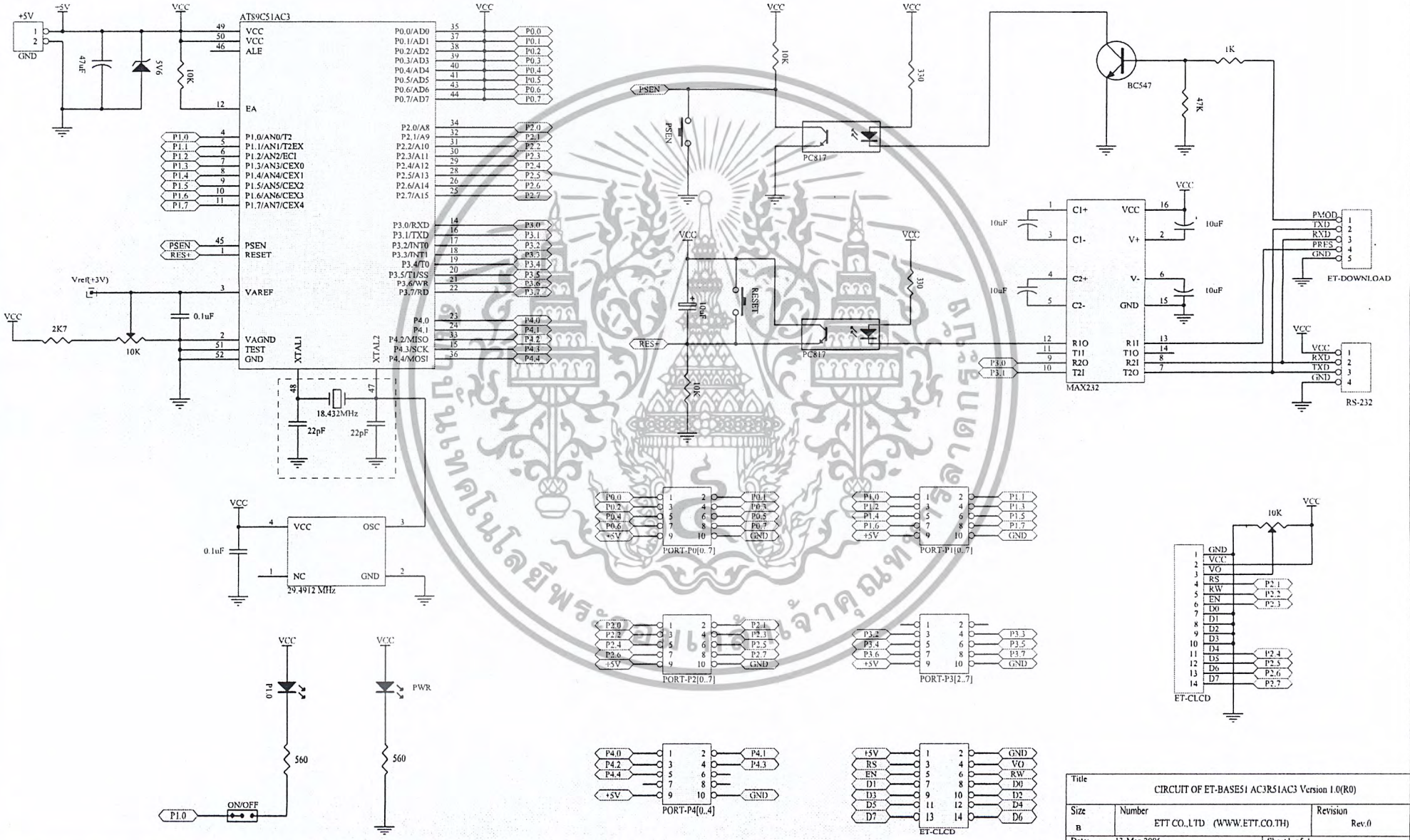
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
:10000000758921759850D2BCC2ACC2A9C2AB74FB31
:10001000F58DF58BD28CD28ED2AF1200880D0A0AE4
:1000200044454D4F20544553542043502D4A52359A
:10003000314143322056312026205632202831329 9
:100040002D436C6F636B290D0A507265737320041E9
:00000001FF
```

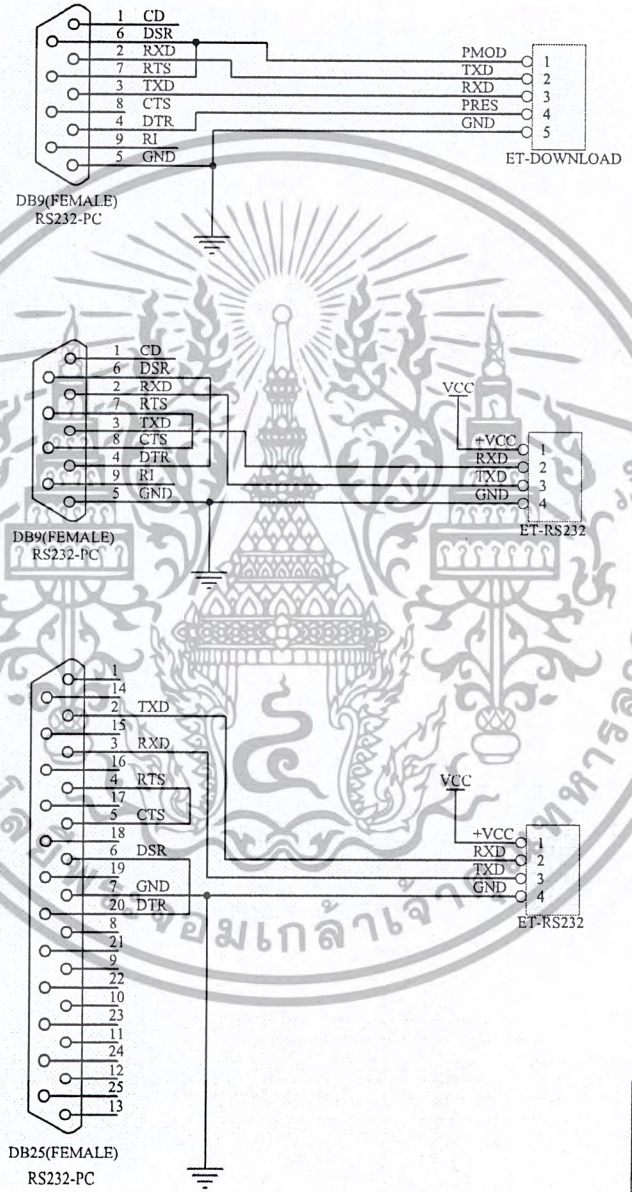
รูปแสดง ลักษณะของ HEX File ที่สามารถใช้กับโปรแกรม FLIP ได้หลังตัดบรรทัดว่างทิ้งไปแล้ว

7. **เมื่อสั่งโปรแกรมข้อมูลให้กับ CPU เรียบร้อยแล้วหลังจากรีเซ็ตบอร์ดแล้วไม่ทำงาน** ซึ่งปัญหานี้อาจเกิดจากสาเหตุความผิดพลาดหลายประการ ซึ่งพอสรุปได้ดังนี้ คือ
   - โปรแกรมที่เขียนขึ้นไม่ถูกต้องยังไม่สามารถทำงานได้เอง ซึ่งปัญหานี้ผู้ใช้ต้องหาทางตรวจสอบและแก้ไขความผิดพลาดที่เกิดขึ้นเอง
   - ยังไม่ได้มีการสั่ง Load HEX เข้ามารอไว้ยัง Buffer แล้วสั่งโปรแกรม (Program Device) ซึ่งโปรแกรม FLIP จะนำข้อมูลที่อยู่ใน Buffer เขียนไปยังหน่วยความจำของโปรแกรม
   - สวิตช์ PSEN อาจถูกกดค้างอยู่ จึงทำให้การรีเซ็ตบอร์ดทุกๆครั้งนั้น CPU จะเข้าไปทำงานใน Monitor Mode เสมอ ซึ่งปัญหานี้สามารถตรวจสอบได้โดยการวัดระดับลอจิกที่ขาสัญญาณ PSEN ของ CPU ซึ่งอยู่ที่ขา 45 (PLCC-52) ซึ่งควรมีสภาวะเป็น "1" ถ้าไม่มีการกดสวิตช์ PSEN ไว้ และควรมีสภาวะเป็น "0" ถ้ามีการกดสวิตช์ PSEN ไว้
   - สวิตช์ RESET อาจถูกกดค้างอยู่ จึงทำให้ CPU ไม่สามารถหลุดพ้นจากสภาวะการรีเซ็ตได้ ซึ่งปัญหานี้สามารถตรวจสอบได้โดยการวัดระดับลอจิกที่ขาสัญญาณ RESET ของ CPU ซึ่งอยู่ที่ขา 1 (PLCC-52) ซึ่งควรมีสภาวะเป็น "0" ถ้าไม่มีการกดสวิตช์ RESET ไว้ และควรมีสภาวะเป็น "1" ถ้ามีการกดสวิตช์ RESET ไว้
   - ค่าของ Device BSB และ SBV ยังไม่ได้ถูกกำหนดให้มีค่าเป็น 00H ไว้ ซึ่งจะทำให้โปรแกรมกระโดดไปทำงานยังตำแหน่งที่ชี้โดย Device SBV แทน ซึ่งถ้าค่าของ Device SBV ไม่ใช่ศูนย์ก็จะเหมือนกับว่าโปรแกรมไม่ทำงาน ซึ่งการแก้ไข ปัญหานี้ หลังจากสั่งโปรแกรมข้อมูลให้กับ CPU เรียบร้อยแล้ว ควรกำหนดให้ค่าของ Device BSB และ Device SBV มีค่าเป็น 00H ไว้ทั้งคู่

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



Title	CIRCUIT OF ET-BASE51 AC3R51AC3 Version 1.0(R0)	
Size: B	Number: ETT CO.,LTD (WWW.ETT.CO.TH)	Revision: Rev.0
Date: 13-Mar-2006		Sheet 1 of 1
File: D:\My Circuit\MyCircuit.Ddb		Drawn By: Eakachai Makarn



Title	ETT CO.,LTD (WWW.ETT.CO.TH)		
Size: A4	Number: CABLE RS232	Revision: 1.0	
Date: 31-Mar-2006		Sheet1 of 1	
File: D:\My Circuit\RS232CABLE.ddb		Drawn By:	