สำนักหอสมุดกลาง พระจอมเกล้าลาดกระบัง

บอร์ดแสดงผล 16 สี

16 - Colour LED Displayed Board

T042632

โดย

นางสาวนิลุบล พรสุขสวัสดิ์ รหัสประจำตัว 40010397

นายน้ำเหือง จันทร์มณี รหัสประจำตัว 40010404

อาจารย์ที่ปรึกษา

ผศ. พลผดุง ผดุงกุล

เลขหมู่.....
เลขทะเบียน 42632
วัน, เดือน, ปี 5 มิ.ย. 2545

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

ปริญญานิพนธ์สำหรับปริญญาวิศวกรรมศาสตรบัณฑิต

สาขาวิชาอิเล็กทรอนิกส์

คณะวิศวกรรมศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2543

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ปริญญานิพนธ์ปีการศึกษา 2543

ภาควิชา วิศวกรรมอิเล็กทรอนิกส์

คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เรื่อง บอร์ดแสดงผล 16 สี

ผู้จัดทำ

1. นางสาวนิลุบล พรสุขสวัสดิ์ รหัสประจำตัว 40010397
2. นายน้ำเหือง จันทร์มณี รหัสประจำตัว 40010404

..... อาจารย์ที่ปรึกษา

(ผศ. พลผดุง ผดุงกุล)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บอร์ดแสดงผล 16 สี

16 - Colour LED Displayed Board

นางสาวนิลุบล พรสุขสวัสดิ์ รหัสประจำตัว 40010397

นายน้ำเหือง จันทร์มณี รหัสประจำตัว 40010404

โครงงานนี้ได้รับการตรวจสอบแล้ว พร้อมที่จะทำการสอบได้

.....

(ผศ. พลผดุง ผดุงกุล)

อาจารย์ที่ปรึกษา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บอร์ดแสดงผล 16 สี

16 - Colour LED Displayed Board

โดย

นางสาวนิลุบล พรสุขสวัสดิ์ รหัสประจำตัว 40010397

นายน้ำเหือง จันทร์มณี รหัสประจำตัว 40010404

อาจารย์ที่ปรึกษา

ผศ. พลผดุง ผดุงกุล

ปริญญานิพนธ์สำหรับปริญญาวิศวกรรมศาสตรบัณฑิต

สาขาวิชาอิเล็กทรอนิกส์

คณะวิศวกรรมศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2543

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กิตติกรรมประกาศ

โครงงานนี้ใช้เวลาในการศึกษาหาข้อมูล รวบรวมข้อมูล และเก็บรายละเอียดปลีกย่อยต่างๆ ประมาณหนึ่งภาคการศึกษา ซึ่งแต่ละขั้นตอนของการทำงานได้พบอุปสรรคต่าง ๆ มากมาย แต่ก็สำเร็จลุล่วงไปได้ด้วยดี ทั้งนี้เพราะได้รับคำแนะนำจากอาจารย์ที่ปรึกษาและรุ่นพี่ปริญญาโท ซึ่งมีประโยชน์อย่างมาก และผู้จัดทำคิดว่าความรู้และประสบการณ์ในการทำงานครั้งนี้ จะมีประโยชน์อย่างมากในการทำงานครั้งต่อไป

ผู้จัดทำขอขอบพระคุณอาจารย์ที่ปรึกษา รุ่นพี่และเพื่อน ๆ ทุกคนและบุคคลอื่น ๆ ที่มิได้กล่าวถึงในที่นี้ ที่ให้ความช่วยเหลือเป็นอย่างดี จนทำให้งานในครั้งนี้สำเร็จได้อย่างที่ตั้งใจไว้

นิลุบล พรสุขสวัสดิ์

(นางสาวนิลุบล พรสุขสวัสดิ์)

น้ำเหือง จันทร์มณี

(นายน้ำเหือง จันทร์มณี)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บอร์ดแสดงผล 16 สี

นางสาวนิลุบล พรสุขสวัสดิ์
นายน้ำเหือง จันทร์มณี
ผศ. พลผดุง ผดุงกุล (อาจารย์ที่ปรึกษา)
ภาคการศึกษาที่ 2 ปีการศึกษา 2543

บทคัดย่อ

ปริญญานิพนธ์ฉบับนี้ เป็นส่วนหนึ่งของวิชา PROJECT II เสนอโครงงานบอร์ดแสดงผล 16 สี ซึ่งควบคุมการแสดงผลโดยฮาร์ดแวร์ เนื่องจากมีความเร็วสูงกว่าใช้ซอร์ฟแวร์ควบคุม และไม่ต้องใช้ MCS-51 ควบคุมตลอดเวลา ในการแสดงผลของบอร์ดแสดงผลนี้ จะประกอบไปด้วยส่วนประกอบหลัก 3 ส่วน ได้แก่ ส่วนรับข้อมูลการเลือกแสดงภาพ ส่วนแสดงผล และส่วนควบคุมการแสดงผล ส่วนรับข้อมูลการเลือกภาพใช้การสื่อสารแบบอนุกรมแบบ RS-232 และแสดงแป้นพิมพ์ตัวเลขที่ถูกกดบนหน้าจอคอมพิวเตอร์ โดยไมโครคอนโทรลเลอร์ MCS-51 จะเป็นศูนย์กลางในการควบคุมส่วนต่าง ๆ ทั้งหมด ส่วนแสดงผลนั้นจะประกอบด้วย LED 2 สี คือ สีแดงและสีเขียว จำนวนทั้งสิ้น 1024 ชิ้น ซึ่งโครงงานนี้จะทำการไบอัสให้กระแสเฉลี่ยแต่ละสีต่างกันทำให้เกิดการผสมของแสงสีได้ถึง 16 ระดับ และโดยการใช้ RAM จำนวน 3 ตัว ทำให้สามารถแสดงภาพต่อเนื่องได้ 3 ภาพในการเลือกการแสดงผลแต่ละครั้ง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

16 - Colour LED Displayed Board

Miss Nilubon Pornsuksawat

Mr. Namhueng Janmanee

Asst. Prof. Polphadung Phadungkul

2nd Semester, Educational Year 2000

Abstract

This thesis presents details of 16 Colors displaying board project. Using hardware to control scanning board is more advantage than using software because it is faster and more efficiency. This displayed unit has three major parts. PC keyboard data reciever , LED displaying controller and LED displaying board. The microcontroller MCS – 51 is programmed to control and link between these units so that the display can be performed. By varying bias current , this allows both red and green light of LED appear to be different shades. And when they are mix together , this results is various shade of light up to 16 colors . We design this board to be able to display motion pictures by using more than one RAM. However , with only Three RAMs that have been used , The displayed board can not show any movies that have nature.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ

กิตติกรรมประกาศ	I
บทคัดย่อ	II
Abstract	III
สารบัญ	IV
สารบัญรูปภาพ	VI
สารบัญตาราง	VII
บทที่ 1 บทนำ	1
บทที่ 2 หลักการและทฤษฎีที่เกี่ยวข้อง	2
2.1 ไมโครคอนโทรลเลอร์แบบชิพเดี่ยวตระกูล MCS-51	2
2.1.1 โครงสร้างภายในของ 8051	3
2.1.2 พอร์ทของ 8051	4
2.2 การเชื่อมโยง 8255 กับ MCS-51	5
2.2.1 8255 PROGRAMMABLE PERIPHERAL INTERFACE	5
2.2.2 โหมดการทำงาน	6
2.3 รายละเอียดต่างๆบนบอร์ด	8
2.3.1 หน่วยความจำ	8
2.4 SRAM	9
2.4.1 โครงสร้างของ SRAM	9
2.4.2 ลำดับการอ่านข้อมูลจาก SRAM	10
2.4.3 ลำดับการเขียนข้อมูลใน SRAM	11
บทที่ 3 รายละเอียดในการออกแบบและสร้าง	12
3.1 ส่วนรับข้อมูลจากคอมพิวเตอร์	12
3.2 ส่วนแสดงผล	12
3.3 ส่วนควบคุมการแสดงผล	13
3.3.1 ส่วน Software	13
3.3.2 ส่วน hardware	13
3.4 แผนผังการทำงานอย่างย่อ	19

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ (ต่อ)

บทที่ 4 ผลการทดลอง	20
บทที่ 5 สรุปผลและวิจารณ์	24
บรรณานุกรม	25
ภาคผนวก	26

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูปภาพ

รูปที่ 1.1 บล็อกไดอะแกรมของบอร์ดแสดงผล 16 สี 1

รูปที่ 2.1 แสดงโครงสร้างภายในของ MCS-51 3

รูปที่ 2.2 การจัดวางขาของ 8051 4

รูปที่ 2.3 การจัดวางขาและโครงสร้างของ 8255 6

รูปที่ 2.4 โครงสร้างของ SRAM 9

รูปที่ 2.5 โครงสร้างของ SRAM เส้นทางนำข้อมูลเข้าหรือออกจากชิปเป็นเส้นทางเดียวกัน 10

รูปที่ 2.6 แผนผังเวลาการอ่านข้อมูลจากหน่วยความจำ 11

รูปที่ 2.7 แผนผังเวลาการเขียนข้อมูลลงในหน่วยจำ 11

รูปที่ 3.1 ภาพบอร์ดแสดงผล 12

รูปที่ 3.2 แสดงวงจรเลือก row 15

รูปที่ 3.3 แสดงวงจรเลือก column 16

รูปที่ 3.4 แสดงวงจรที่ต่อมาจาก 8255 17

รูปที่ 3.5 แสดงวงจรส่วนควบคุมการทำงานของ RAM 18

รูปที่ 3.6 แผนผังการทำงานอย่างย่อของ MCS –51 ในการควบคุมบอร์ดแสดงผล 19

รูปที่ 4.1 แสดงภาพสัญญาณนาฬิกาที่สร้างจาก IC 555 20

รูปที่ 4.2 แสดงภาพสัญญาณที่ควบคุมการทำงานของ 74F244 21

รูปที่ 4.3 แสดงภาพสัญญาณเลือก column 21

รูปที่ 4.4 แสดงภาพสัญญาณ address บิต 0 ที่สร้างจาก 74LS569 22

รูปที่ 4.5 แสดงภาพสัญญาณ address บิต 9 ที่สร้างจาก 74LS569 22

รูปที่ 4.6 แสดงภาพสัญญาณที่ควบคุมการทำงานของ RAM 23

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญตาราง

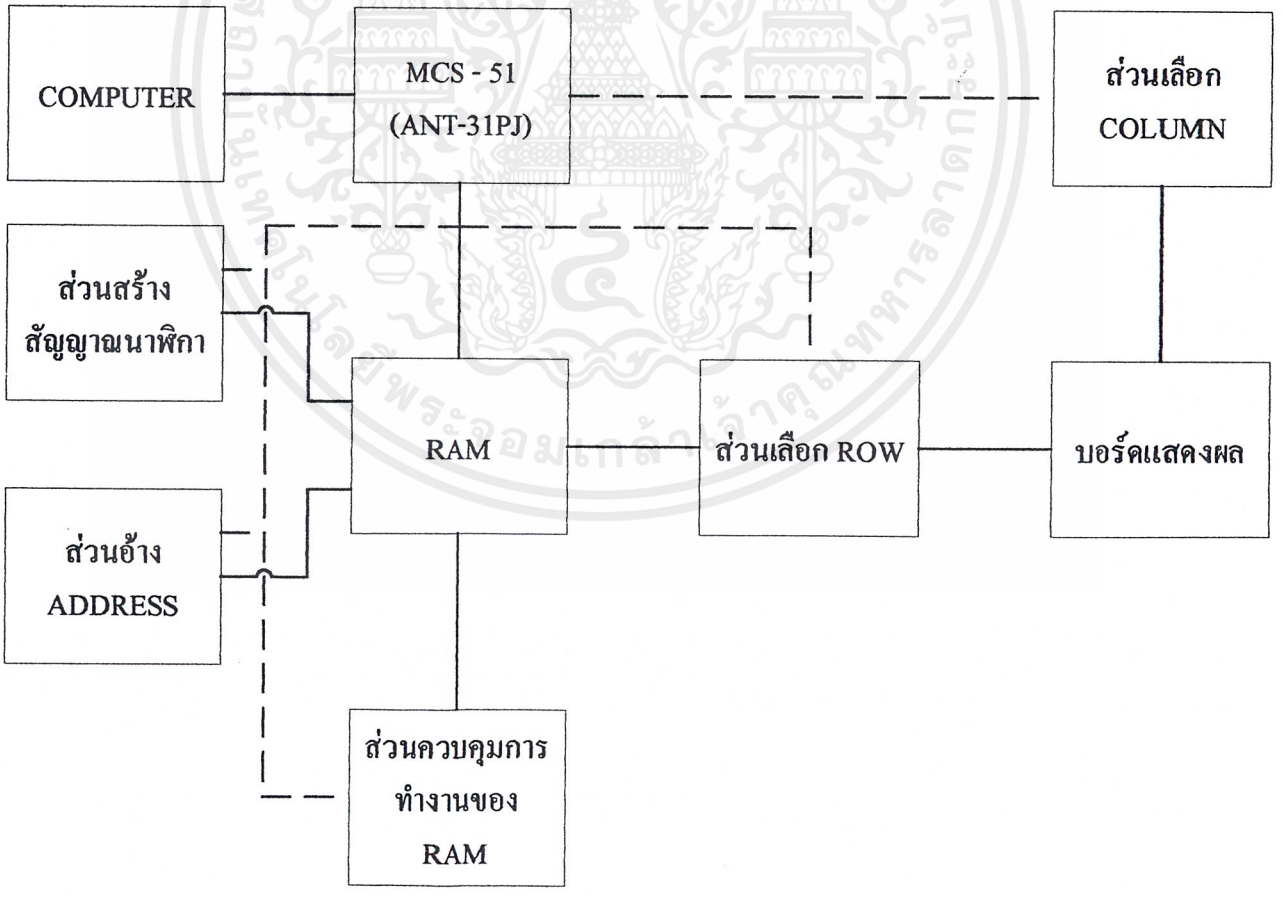
ตารางที่ 2.1 รายละเอียดของไมโครคอนโทรลเลอร์ตระกูล MCS-51 2
ตารางที่ 2.2 สรุปโหมดต่าง ๆ ของ 8255 7
ตารางที่ 2.3 ตารางความจริงของ 8255 8
ตารางที่ 2.4 การจัดหน่วยความจำของบอร์ด ANT-31PJ 8

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 1
บทนำ

ไมโครคอนโทรลเลอร์ได้เข้ามามีบทบาทในชีวิตประจำวันมากในปัจจุบัน ซึ่งจะเห็นได้ว่าไมโครคอนโทรลเลอร์จะใช้ในงานควบคุมต่างๆ สิ่งที่พิเศษในไมโครคอนโทรลเลอร์เมื่อเปรียบเทียบกับซีพียูในปัจจุบันก็คือ กลุ่มของส่วนที่ติดต่อกับอุปกรณ์ภายนอก (Peripheries) ซึ่งได้รวมไว้บนชิปแล้วทำให้องค์ประกอบภายนอกของไมโครคอนโทรลเลอร์ลดลง

โครงงานนี้มีจุดมุ่งหมายคือสร้างฮาร์ดแวร์เพื่อควบคุมการแสดงผลบนบอร์ด LED โดยนำไมโครคอนโทรลเลอร์ในตระกูล 8051 ของอินเทล ไปประยุกต์ใช้งาน มีการเขียนโปรแกรมควบคุมการทำงานด้วยภาษาแอสเซมบลี และออกแบบให้รับข้อมูลของคีย์บอร์ดที่ส่งผ่านทาง RS-232 นำไปแสดงผลบนจอ computer เพื่อที่จะใช้ในการเลือกรูปภาพที่จะนำไปแสดงบนบอร์ดแสดงผล



รูปที่ 1.1 บล็อกไดอะแกรมของบอร์ดแสดงผล 16 สี

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 2
หลักการและทฤษฎีที่เกี่ยวข้อง

2.1 ไมโครคอนโทรลเลอร์แบบชิพเดี่ยวตระกูล MCS-51

ไมโครคอนโทรลเลอร์แบบชิพเดี่ยวตระกูล MCS-51 นี้มีอยู่ด้วยกันหลายเบอร์ซึ่งมีรายละเอียดดังตารางที่ 2.1

ตารางที่ 2.1 รายละเอียดของไมโครคอนโทรลเลอร์ตระกูล MCS-51

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

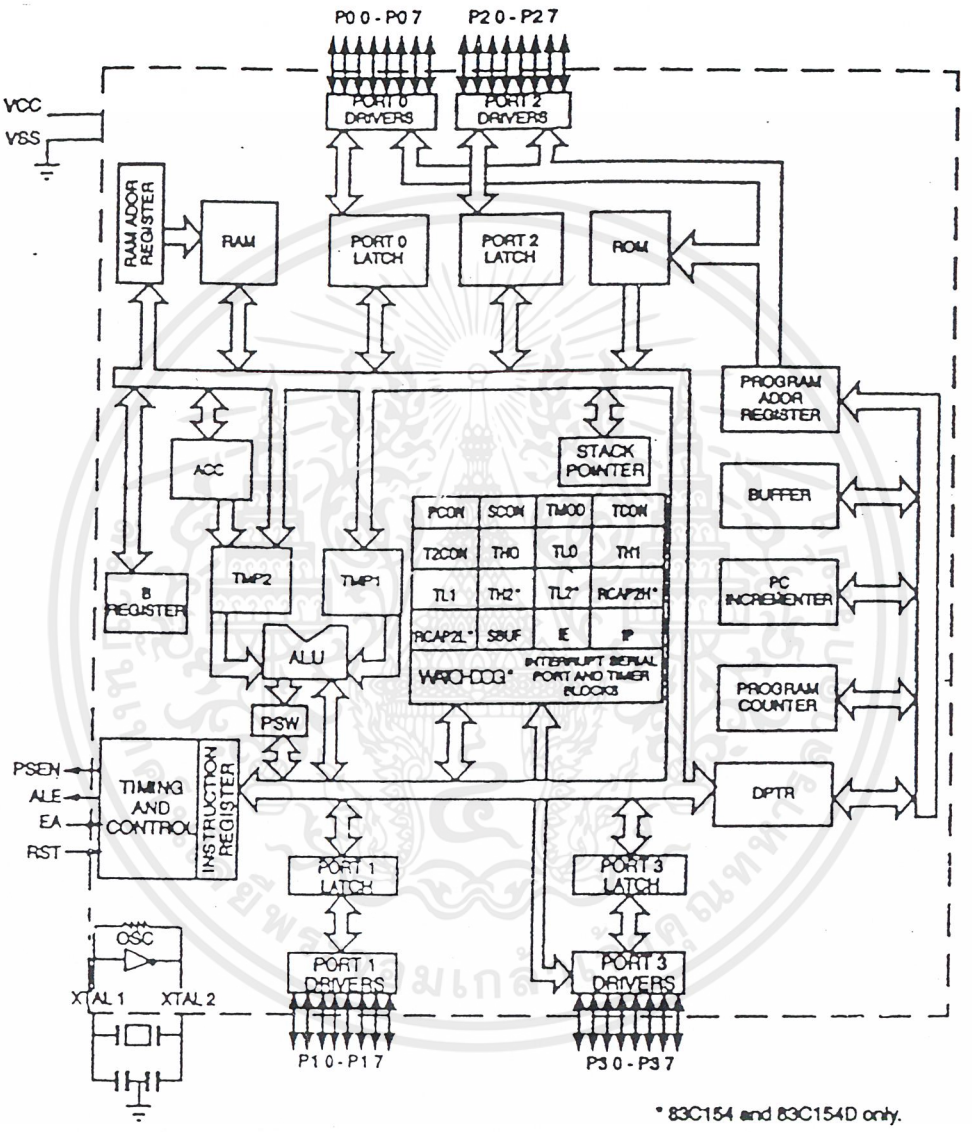
คุณสมบัติของไมโครคอนโทรลเลอร์ตระกูล MCS-51

- ต้องการแหล่งจ่ายไฟ +5 V ชุดเดียว
- มีหน่วยความจำโปรแกรม (Program memory) ขนาด 4 กิโลไบต์สำหรับเบอร์ 8051 และ 8031 สำหรับเบอร์ 8052 มีหน่วยความจำถึง 8 กิโลไบต์
- มีหน่วยความจำสำหรับเก็บข้อมูล (Data memory) ขนาด 128ไบต์ สำหรับเบอร์ 8052 ขึ้นไปมีถึง 256ไบต์
- มีหน่วยความจำสำหรับเก็บโปรแรมและข้อมูลแยกจากกันอย่างละ 64 กิโลไบต์
- มีไทม์เมอร์ เคาน์เตอร์ ขนาด 16 บิต 2ชุด (สำหรับ 8052 มี 3 ชุด) ทำงานได้ 4 โหมด
- รับอินเตอร์รัพท์ได้ 6 แหล่ง 5 เวคเตอร์ สำหรับเบอร์ 8052 ขึ้นไปมี 8 แหล่ง 6 เวคเตอร์
- มีพอร์ทรับส่งข้อมูลอนุกรม (UART) 2 พอร์ทแบบ Full Duplex เลือกได้ 4 โหมด
- มีคำสั่งในการทำ AND , OR หรือ COMPLEMENT ได้ทั้งแบบ 8 บิต และ 1 บิต

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

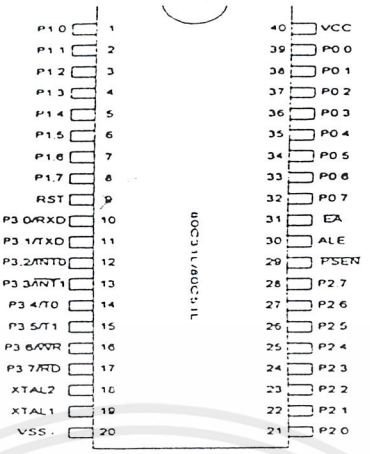
2.1.1 โครงสร้างภายในของ 8051

MCS-51 ใช้เทคโนโลยีในการผลิตเป็นแบบ NMOS และ CMOS เบอร์ 8032 และ 8052 จะมี ROM BASIC อยู่ภายใน จึงสะดวกสำหรับโปรแกรมเมอร์ที่จะเขียนโปรแกรมด้วยภาษาเบสิก



รูปที่ 2.1 แสดงโครงสร้างภายในของ MCS-51

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.2 การจัดวางขาของ 8051

2.1.2 พอร์ทของ 8051

8051 เป็นไมโครคอนโทรลเลอร์ขนาด 40 ขา ซึ่งมีขาต่าง ๆ ดังนี้

- Vcc (ขา 40) ต่อกับ +5 V
- Vss (ขา 20) เป็นขา GND
- พอร์ท 0 (ขา 32-39) มีทั้งหมด 8 บิต คือ (P0.7-P0.0)มีทั้งหมด 8 บิตคือ(P0.7-P0.0) ใช้งานได้ 2 หน้าที่ คือ แอดเดรสบัส และ ดาต้าบัสเมื่อต้องการติดต่อกับหน่วยความจำภายนอก หรือเป็นไอโอพอร์ท ถ้าต้องการให้เป็นอินพุทพอร์ทต้องส่งลอจิก "1" ไปยังพอร์ทนี้จะมีผลทำให้ Q ของ D-FF เป็น "0" ทำให้ FET ตัวล่างมีสถานะ OFF สัญญาณที่ใช้อ่านอินพุทพอร์ทแลทซ์ โดยส่งสัญญาณ READ LATCH ไปกระตุ้นที่ Tri-State Buffer ตัวบน และการอ่าน Port (pin) จะใช้สัญญาณ Read (pin)
- พอร์ท 1 (ขา 1-8) มีทั้งหมด 8 บิต คือ (P1.7-P1.0) มีโครงสร้างคล้าย พอร์ท 0 แต่จะใช้ความต้านทานภายในพลูอัพแทน Internal Pull Up Register
- พอร์ท 2 (ขา 21-28) มีทั้งหมด 8 บิต คือ (P2.7-P2.0) มีโครงสร้างคล้าย พอร์ท 0 โดยมี FET ตัวล่างตัวเดียว ส่วนด้านบนใช้ความต้านทานพลูอัพแทน (Internal Pull Up) พอร์ทนี้ทำงาน 2 หน้าที่ คือ สามารถใช้เป็นแอดเดรสบัสขนาด 8 บิต (A15-A8) และเป็นไอโอพอร์ทใช้งานทั่วไป เมื่อจะใช้งานเป็นอินพุทพอร์ทต้องส่งลอจิก "1" มาที่พอร์ทนี้ก่อน เพื่อบังคับให้ FET อยู่ในสภาวะ OFF
- พอร์ท 3 (ขา 10-17) มีทั้งหมด 8 บิต คือ (P3.7-P3.0) มีโครงสร้างคล้าย พอร์ท 1 ทำงานได้ 2 หน้าที่ คือ เป็นไอโอพอร์ท ถ้าจะโปรแกรมให้เป็นอินพุทพอร์ทต้องส่งลอจิก "1"

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

มาที่พอร์ทนี้ก่อน และอีกหน้าที่หนึ่งก็คือ ใช้ส่งสัญญาณควบคุมออกมา และรับสัญญาณเข้าไป สัญญาณต่าง ๆ มีดังนี้

P3.0/RXD (Serial Input Port) เป็นขาที่ใช้รับข้อมูลแบบอนุกรม (UART)

P3.1/TXD (Serial Input Port) เป็นขาที่ใช้รับข้อมูลแบบอนุกรม (UART)

P3.2/$\overline{\text{INT0}}$ (External Interrupt 0) ใช้รับสัญญาณการขัดจังหวะจากภายนอกเบอร์ 0

P3.3/$\overline{\text{INT1}}$ (External Interrupt 1) ใช้รับสัญญาณการขัดจังหวะจากภายนอกเบอร์ 1

P3.4/T0 (Counter 0 External Input) ขารับสัญญาณพัลซ์อินพุทเข้าไปยังวงจร Counter 0 (เป็นอินพุทโหมดเคาน์เตอร์)

P3.5/T1 (Counter 1 External Input) ขารับสัญญาณพัลซ์อินพุทเข้าไปยังวงจร Counter 1 (เป็นอินพุทโหมดเคาน์เตอร์)

P3.6/$\overline{\text{WR}}$ (External Data Memory Write Store) ขาสัญญาณควบคุมการเขียนข้อมูลลงหน่วยความจำภายนอก

P3.7/$\overline{\text{RD}}$ (External Data Memory Read Store) ขาสัญญาณควบคุมการอ่านข้อมูลจากหน่วยความจำภายนอก

- ALE(ขา30) เป็นขาส่งสโตรบสำหรับใช้ในการแลตซ์แอดเดรสไบต์ต่ำ (A7-A0) ที่ส่งออกมาจาก (พอร์ท 0) สัญญาณนี้จะแอคทีฟทุก ๆ 2 ครั้งใน 1 แมชชีนไซเคิล
- $\overline{\text{PSEN}}$(ขา29) เป็นขาสโตรบที่ใช้สำหรับอ่านข้อมูลจาก Program Memory ภายนอก สัญญาณนี้จะส่งออกมา 2 ครั้งในแต่ละแทชชีนไซเคิล แต่ถ้าเป็นการอ่าน Internal Program Memory จะไม่มีสัญญาณออกที่ขานี้
- $\overline{\text{EA}}$(ขา31) ใช้เลือกหน่วยความจำภายนอก

  ป้อน “0” จะอ่านโปรแกรมจากภายนอกชิพ

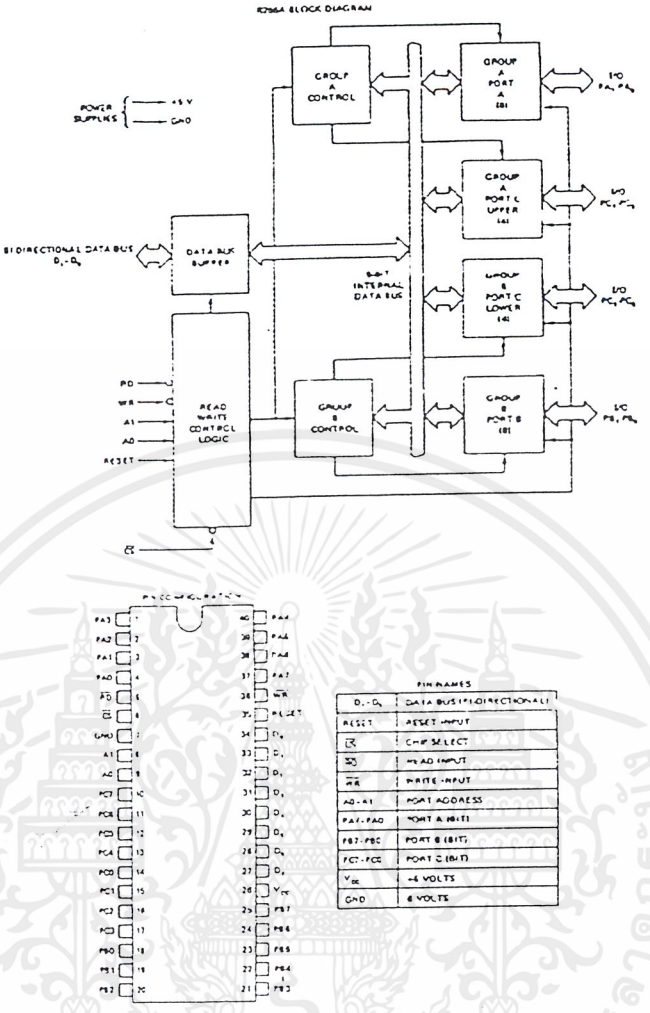
  ป้อน “1” จะอ่านโปรแกรมจากภายในชิพ
- RST(ขา9) ขารีเซ็ท จะรีเซ็ทได้ก็ต่อเมื่อป้อนลอจิก “1” เข้าที่ขานี้น่นอย่างน้อย 2 แมชชีนไซเคิล
- XTAL1(ขา19) ใช้ต่อคริสตัลภายนอกโดยเป็นอินพุทเข้าสู่วงจรออสซิเลเตอร์ภายใน
- XTAL2(ขา18) ใช้ต่อคริสตัลภายนอกโดยเป็นเอาท์พุทของวงจรออสซิเลเตอร์ภายใน

2.2 การเชื่อมโยง 8255 กับ MCS-51

2.2.1 8255 PROGRAMMABLE PERIPHERAL INTERFACE

เป็นชิพขนาด 40 ขา มีอยู่ 3 พอร์ท คือ A,B,C เป็นพอร์ท 8 บิตที่สามารถโปรแกรมให้เป็นอินพุทหรือเอาท์พุทก็ได้ โดยที่พอร์ท C ยังแบ่งเป็น 4 บิทล่างและ 4 บิทบน โดยมีโครงสร้างตามรูปที่ 2.3

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.3 การจัดวางขาและโครงสร้างของ 8255

2.2.2 โหมดการทำงาน

การทำงานมีอยู่ด้วยกัน 3 โหมด ดังตารางที่ 2.2

โหมด 0 มีการทำงานแบบ BASIC I/O ไม่มี Handshake

โหมด 1 โหมดนี้ใช้พอร์ท A,B ในการรับหรือส่งข้อมูล และใช้พอร์ท C ในการตรวจสอบสัญญาณ (Handshake)

โหมด 2 โหมดนี้ใช้พอร์ท A ในการรับส่งข้อมูล 2 ทิศทาง และพอร์ท B ในการรับหรือส่งข้อมูลและใช้พอร์ท C บิต 0,1,2 ในการรับส่งข้อมูลบิต และบิต 4,5,6 เป็นสัญญาณ Handshake

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 2.2 สรุปโหมดต่าง ๆ ของ 8255

	MODE 0		MODE 1		MODE 2	
	IN	OUT	IN	OUT	GROUP A ONLY	
PA_0	IN	OUT	IN	OUT	←→	
PA_1	IN	OUT	IN	OUT	←→	
PA_2	IN	OUT	IN	OUT	←→	
PA_3	IN	OUT	IN	OUT	←→	
PA_4	IN	OUT	IN	OUT	←→	
PA_5	IN	OUT	IN	OUT	←→	
PA_6	IN	OUT	IN	OUT	←→	
PA_7	IN	OUT	IN	OUT	←→	
PB_0	IN	OUT	IN	OUT	——	MODE 0 OR MODE 1 ONLY
PB_1	IN	OUT	IN	OUT	——	
PB_2	IN	OUT	IN	OUT	——	
PB_3	IN	OUT	IN	OUT	——	
PB_4	IN	OUT	IN	OUT	——	
PB_5	IN	OUT	IN	OUT	——	
PB_6	IN	OUT	IN	OUT	——	
PB_7	IN	OUT	IN	OUT	——	
PC_0	IN	OUT	$INTR_B$	$INTR_B$	I/O	
PC_1	IN	OUT	IBF_B	$\overline{OBF}_B$	I/O	
PC_2	IN	OUT	$\overline{STB}_B$	$\overline{ACK}_B$	I/O	
PC_3	IN	OUT	$INTR_A$	$INTR_A$	$INTR_A$	
PC_4	IN	OUT	$\overline{STB}_A$	I/O	$\overline{STB}_A$	
PC_5	IN	OUT	IBF_A	I/O	IBF_A	
PC_6	IN	OUT	I/O	$\overline{ACK}_A$	$\overline{ACK}_A$	
PC_7	IN	OUT	I/O	$\overline{OBF}_A$	$\overline{OBF}_A$	

สัญญาณต่าง ๆ ของ 8255

D7 – D0 บัสข้อมูลเชื่อมโยงกับ CPU

A1 – A0 ใช้เลือกพอร์ท A,B,C และพอร์ทควบคุม

RESET เมื่อขานี้ได้รับสัญญาณกระตุ้นลอจิก "1" จะทำให้ 8255 ถูกรีเซ็ทมีผลทำให้ทุกพอร์ทเป็นอินพุททันที

PA7 – PA0 เป็นพอร์ทขนาน 8 บิต

PB7 – PB0 เป็นพอร์ทขนาน 8 บิต

PC7 – PC0 เป็นพอร์ทขนาน 8 บิต

$\overline{RD}$ ในการอ่านข้อมูลที่พอร์ทของ 8255 ต้องทำให้ขานี้เป็นลอจิก "0" พร้อมกับ $\overline{CS}$

$\overline{WR}$ ในการเขียนข้อมูลหรือโปรแกรมลงบน 8255 ต้องทำให้ขานี้เป็นลอจิก "0" พร้อมกับ $\overline{CS}$

$\overline{CS}$ เป็นขาเลือกชิพ 8255 ได้ มักจะต่อกับ I/O DECODER

เมื่อขา $\overline{WR}$,$\overline{RD}$,A0,A1,$\overline{CS}$ ทำงานทั้ง 5 ขา จะมีฟังก์ชันการทำงานดังตารางที่ 2.7

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 2.3 ตารางความจริงของ 8255

A_1	A_0	$\overline{RD}$	$\overline{WR}$	$\overline{CS}$	
					Input operation (READ)
0	0	0	1	0	Port A → data bus
0	1	0	1	0	Port B → data bus
1	0	0	1	0	Port C → data bus
					Output operation (WRITE)
0	0	1	0	0	Data bus → port A
0	1	1	0	0	Data bus → port B
1	0	1	0	0	Data bus → port C
1	1	1	0	0	Data bus → control
					Disable function
X	X	X	X	1	Data bus → 3-state
1	1	0	1	0	Illegal condition
X	X	1	1	0	Data bus → 3-state

2.3 รายละเอียดต่าง ๆ บนบอร์ด

2.3.1 หน่วยความจำ

การจัดหน่วยความจำของบอร์ด ANT-31PJ แบ่งได้เป็น 2 SOCKET โดยสามารถใส่หน่วยความจำได้สูงสุด 64 KBYTE และสามารถเลือกแบบเบอร์ต่าง ๆ ได้ดังนี้

ตารางที่ 2.4 การจัดหน่วยความจำของบอร์ด ANT-31PJ

SOCKET	ขนาดหน่วยความจำ	เบอร์ที่ใช้ได้	แนวทางการใช้
U2	8 – 32 K	2764,27128, 27256	สำหรับโปรแกรมมอนิเตอร์
U3	8 – 32 K	6264,62256	สำหรับเก็บข้อมูลหรือใช้ทดสอบโปรแกรม โดยสามารถเลือกตำแหน่ง ADDRESSให้เป็น0000H หรือ 8000H

การเลือก ADDRESS ของ U3 สามารถเลือกได้เป็น 2 ช่วง ดังนี้

1. ADDRESS 0000H – 1FFFH โดยจะมองเห็นหน่วยความจำเป็น DATA MEMORY เท่านั้น ซึ่งสามารถใช้งานได้ทั่วไป หรือจะใช้กับ BASIC 32 ก็ได้ (BASIC 32 คือ EPROM ที่มี BASIC อยู่ในตัว ทำให้สามารถพัฒนาโปรแกรมด้วยภาษา BASIC ได้)
2. ADDRESS 8000H – 9FFFH โดยจะมองเห็นหน่วยความจำเป็นทั้ง DATA MEMORY และ PROGRAM MEMORY ซึ่งใช้งานได้ทั่วไป ขณะที่ใช้กับ REM31 ได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ด้วย โดยจะทำให้การพัฒนาโปรแกรมทำได้อย่างสะดวก (REM31 คือ EPROM ที่มีโปรแกรมมอนิเตอร์ เพื่อช่วยให้พัฒนาโปรแกรมบนบอร์ดผ่านทางเครื่อง PC ได้

การเลือกเบอร์ของหน่วยความจำ หรือการเลือกตำแหน่งของ ADDRESS สามารถทำได้ด้วยตัว JUMPER ที่อยู่ทางด้านล่างของ SOCKET โดยกำหนดตัว JUMPER ให้ตรงกับตำแหน่งที่ต้องการ

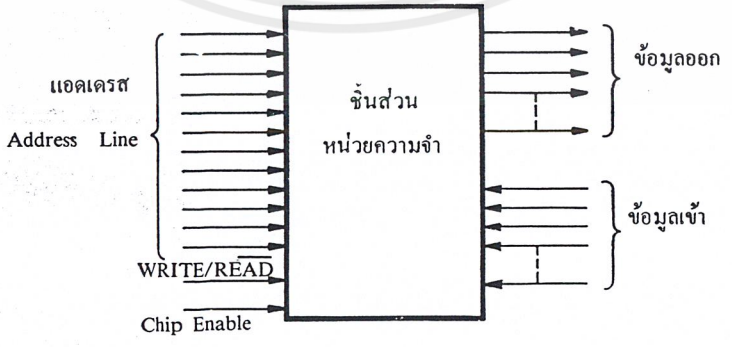
2.4 SRAM (Static Random Access Memory)

โดยทั่วไป SRAM เป็นระบบที่เก็บข้อมูลและสามารถเขียนข้อมูลใหม่ทับลงไปได้ง่าย แต่ถ้าไฟฟ้าดับข้อมูลก็จะสูญหายไปทันที โดยชิปที่มีขายทั่วไปและเป็นที่รู้จักกันดี ได้แก่ เบอร์ 2114 , 6116 และ 6264

2.4.1 โครงสร้างของ SRAM

รูปที่ 2.4 เป็นโครงสร้างพื้นฐานของ RAM ซึ่งมีส่วนสัญญาณที่เกี่ยวข้องกับการควบคุมอยู่หลายเส้น สัญญาณที่สำคัญ ได้แก่

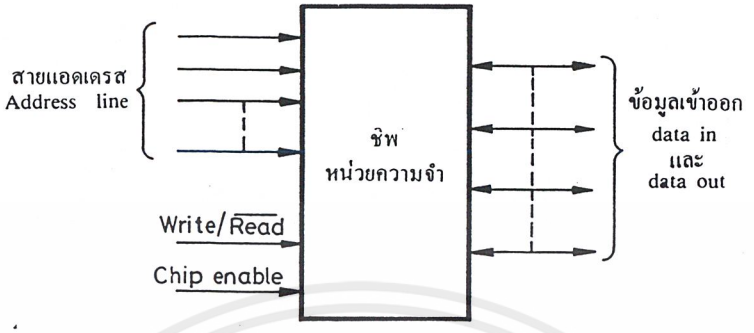
1. ข้อมูลออก (data out) เป็นข้อมูลออกจากตัวชิป เมื่อทำการอ่าน หรือ Read Operation
2. ข้อมูลเข้า (data in) เป็นข้อมูลที่จะถูกนำเข้าไปเก็บไว้ในชิป เมื่อทำการเขียนข้อมูล หรือ Write Operation
3. สายแอดเดรส (address lines) เป็นสัญญาณเลือกตำแหน่งที่จะทำการเขียนหรืออ่าน
4. เขียน/อ่าน (write/$\overline{\text{read}}$) คือสัญญาณควบคุมแสดงขบวนการที่จะทำ คือ เขียนข้อมูล หรืออ่านข้อมูล
5. ชิปอินาเบิ้ล (chip enable) นอกจากจะต้องเลือกตำแหน่งด้วยสายแอดเดรส และเลือกขบวนการ (อ่านหรือเขียน) แล้ว ชิปอินาเบิ้ล เป็นสัญญาณบังคับให้หน่วยความจำอันนี้ต้องรับสัญญาณควบคุมตัวอื่นๆ ทั้งหมด



รูปที่ 2.4 โครงสร้างของ SRAM

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ในบางครั้งเส้นทางนำข้อมูลเข้าหรือออกจากชิปเป็นเส้นทางเดียวกัน จะทำให้ลักษณะของชิปเป็นดังรูปที่ 2.5



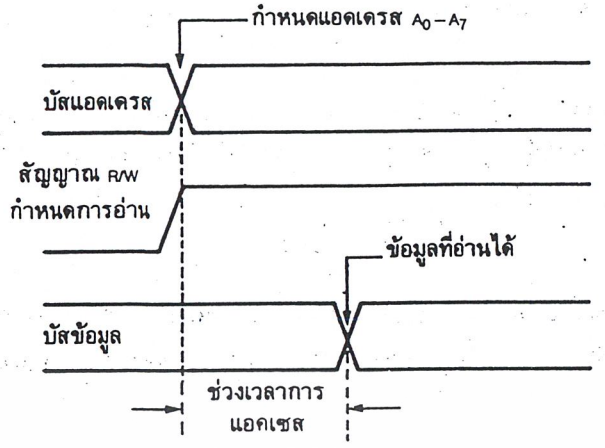
รูปที่ 2.5 โครงสร้างของ SRAM เส้นทางนำข้อมูลเข้าหรือออกจากชิปเป็นเส้นทางเดียวกัน

2.4.2 ลำดับการอ่านข้อมูลจาก RAM

ถ้าเราต้องการจะอ่านข้อมูลออกจาก RAM ที่มีข้อมูลพร้อมอยู่แล้วมาใช้งาน จะต้องมีลำดับการทำงานดังนี้

1. ขั้นแรก กำหนดแอดเดรสที่ต้องการให้กับ RAM วงจรภายใน RAM จะทำการถอดรหัสเพื่อกำหนดตำแหน่งที่แท้จริงสำหรับการอ่าน
2. กำหนดสัญญาณที่ขา R/$\overline{W}$ ให้ถูกต้องตามลอจิก โดยการอ่านหน่วยความจำบางชิป จะต้องกำหนดลอจิก "1" บางชิปอาจเป็นลอจิก "0" การกำหนดให้เป็นลอจิก "0" หรือลอจิก "1" จะต้องพิจารณาโดยดูจากคู่มือของชิปนั้น ๆ ประกอบด้วย
3. จะต้องให้ระบบรออยู่ชั่วขณะหนึ่ง ซึ่งเรียกช่วงเวลานี้ว่าช่วงเวลาการอ่าน (read access time) ในการอ่านนี้จะใช้สัญญาณเลือกชิปทำการเลือก โดยส่งสัญญาณ CE มาก่อน
4. ในช่วงเวลาขณะที่รอนี้ ข้อมูลที่ได้รับการอ่านจะมาปรากฏที่สายสัญญาณ D_{OUT} เพื่อให้รับข้อมูลออกไป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

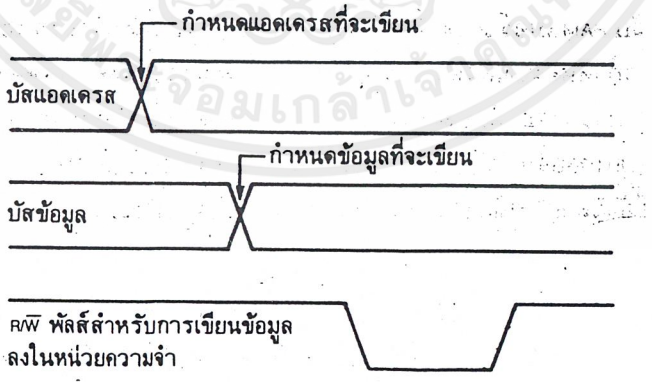


รูปที่ 2.6 แผนผังเวลาการอ่านข้อมูลจากหน่วยความจำ

2.4.3 ลำดับการเขียนข้อมูลใน RAM

การเขียนข้อมูลใน RAM มีลักษณะคล้ายกับการอ่าน แต่มีข้อแตกต่างอยู่บ้าง สำหรับลำดับการเขียนข้อมูลใน RAM มีดังนี้

1. ขั้นแรก จะกำหนดแอดเดรสให้กับหน่วยความจำตามตำแหน่งที่ต้องการจะเขียนข้อมูล
2. กำหนดสายสัญญาณข้อมูลเข้าโดยใช้สายสัญญาณ D_{IN}
3. ให้ระบบรอเวลาชั่วขณะหนึ่ง เรียกช่วงเวลานี้ว่า (write access time) เพื่อให้วงจรภายในได้รับการถอดรหัสตำแหน่งให้เรียบร้อยก่อน และสัญญาณ CE ต้องมารอก่อนแล้ว
4. หลังจากรอเวลา ให้กำหนดสัญญาณ $R/\overline{W}$ เพื่อการเขียน สัญญาณ $R/\overline{W}$ นี้จะเป็นพัลส์เล็ก ๆ ที่พอเพียงกับการเขียนข้อมูลลงใน RAM



รูปที่ 2.7 แผนผังเวลาการเขียนข้อมูลลงในหน่วยความจำ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 3

รายละเอียดในการออกแบบและการสร้าง

ในการออกแบบและการสร้างจะประกอบด้วย 2 ส่วน คือส่วนของวงจรฮาร์ดแวร์ (Hardware) และส่วนของโปรแกรมซอร์ฟแวร์ (Software) โดยมีจุดประสงค์และความสามารถในการทำงานได้ดังนี้

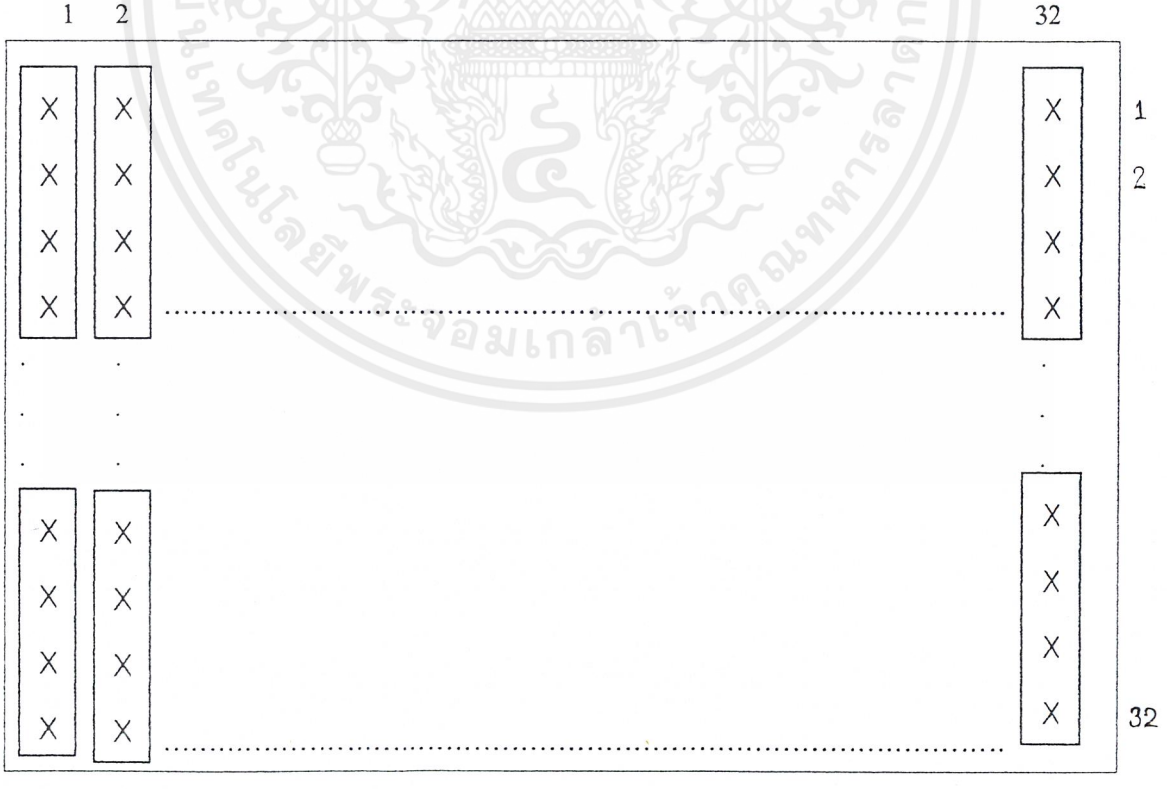
- ใช้ MCS-51 รับข้อมูลจากคีย์บอร์ด โดยผ่าน RS-232 และนำข้อมูลไปแสดงผลบนหน้าจอ computer
- ใช้คีย์บอร์ดในการเลือกรูปที่แสดงบนบอร์ดแสดงผลได้

ส่วนต่างๆ ของวงจรมีดังนี้

3.1 ส่วนรับข้อมูลจากคีย์บอร์ด

เราจะรับข้อมูลจากคีย์บอร์ดผ่านทาง RS –232 ซึ่งข้อมูลที่เป็นค่าแอสกีนั้นจะถูกส่งมาเก็บไว้ที่ Acc. ซึ่งหลังจากรับค่าเสร็จแล้ว โปรแกรมจะทำการตรวจสอบว่าเป็นคีย์ใดที่ถูกกด

3.2 ส่วนแสดงผล



รูปที่ 3.1 ภาพบอร์ดแสดงผล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ส่วนแสดงผลจะประกอบไปด้วย LED 2 สี จำนวนทั้งสิ้น 1024 ตัว โดยมีขนาด 32 X 32 โดยแต่ละ column นั้น ขาสีแดง และขาสีเขียวจะต่อถึงกันหมด และขา ground ของแต่ละ row ก็จะต่อถึงกันหมดด้วยเช่นเดียวกัน ในการแสดงผลของบอร์ดแสดงผลนั้น ชุดควบคุมจะไบอัสให้ LED ติดครั้งละ 4 dot หลังจากติดครบทั้ง 32 column แล้วอีก 4 row ต่อมาก็จะติดแทนไปเรื่อย ๆ จนกระทั่งครบ 32 row

3.3 ส่วนควบคุมการแสดงผล จะแบ่งได้เป็น 2 ส่วน

- ส่วน software
- ส่วน hardware

3.3.1 ส่วน solfware

ส่วนโปรแกรมภายใน MCS-51 หลังจากรับค่าแอสกีของคีย์ที่ถูกกดผ่านทาง RS-232แล้ว โปรแกรมจะทำการตรวจสอบรูปว่าเป็นรูปใดที่ถูกเลือก โดยรูปแบบของการแสดงผลแต่ละภาพจะถูกเก็บไว้ใน ROM ซึ่งขณะนี้มีให้เลือก 4 ภาพ เมื่อมีการกดคีย์เลือกภาพใดภาพหนึ่งออกมา โปรแกรมก็จะทำการเอาข้อมูลการแสดงผลของบอร์ด LED ภาพนั้น ออกไปสู่ SRAM ทีละตัวจนครบทั้ง 3 ตัว จากนั้น MCS-51 จะกระตุ้นให้ส่วนแสดงผลของ LED ทำงานในการควบคุมการทำงานของ RAM ทั้ง 3 ตัว ในขณะเขียนข้อมูลนั้นจะควบคุมโดยP1ของ MCS-51

3.2.2 ส่วน hardware

หลังจากเขียนข้อมูลลงใน RAM ทั้ง 3 ตัว แล้ว MCS-51 ก็จะสั่งให้ส่วนต่าง ๆ ซึ่งประกอบไปด้วย ส่วนควบคุมการทำงานของ 74F244 , ส่วนควบคุมการทำงานของส่วนเลือก column , ส่วนอ้าง address และส่วนควบคุมการทำงานของ RAM ซึ่งจะอธิบายการทำงานของแต่ละส่วนได้ดังนี้

ส่วนควบคุมการทำงานของ 74F244

จากรูปวงจรเลือก row จะเห็นว่าใช้ LINE DRIVER 74F244 ทำหน้าที่ไบอัสให้ LED ติด โดย 74F244 หนึ่งตัวให้เอ้าท์พุท 8 bit จะควบคุม LED ได้ 4 row (1 row มี 2 สี) โดยขาอินพุทจะรับข้อมูลจาก RAM โดยตรง เราใช้ 74F244 ทั้งหมด 8 ตัว โดยจะผลัดกันทำงานที่ละ 1 ตัว โดย LED ตัวใดจะติดนั้นจะขึ้นอยู่กับวงจรเลือก column ในการควบคุมการทำงานของ 74F244 นั้นจะใช้ CMOS 4017 ต่ออยู่กับขา 1 และขา 19 ของ 74F244 ซึ่งเราจะต่อวงจรให้ 4017 นับตั้งแต่ 0 – 7 ซึ่งในการที่จะให้ 74F244 ตัวถัดไปให้ทำงาน นั้นจะต้องมีการแสดงผลครบทั้ง 32 column นั่นคือจาก LED ที่แสดงแต่ละ column ใช้ 4 พัลส์ แต่ว่าจะต้องแสดงทั้งหมด 32 column สำหรับ 74F244 แต่ละตัว ดังนั้นในการเลือก 74F244 แต่ละตัวให้ทำงานต้องหน่วงเวลาโดยวงจรหาร 4 และ หาร 32 (หาร 4 และหาร 8)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ส่วนควบคุมการทำงานของส่วนเลือก column

การทำงานของวงจรเลือก column นั้นจะต้องมีความสัมพันธ์กับวงจรเลือก row โดยจากวงจรเลือก column นั้นเราจะต้องใช้ 4 พัลส์ ในการ DRIVE LED แต่ละ column ดังนั้น การเปลี่ยน column แต่ละครั้งจะต้องห่างกัน 4 พัลส์ เราใช้ CMOS 4017 ทำหน้าที่เป็นตัว counter ในการนับและเลือกไบอัสให้กับทรานซิสเตอร์แต่ละ column นั่นคือ เมื่อมีสัญญาณ colck เข้ามา 4 พัลส์ CMOS 4017 จะเปลี่ยน column การแสดงผล จะเห็นว่า CMOS 4017 ตัวแรกเป็นวงจรหาร 4 ส่วน อีก 4 ตัวเป็นวงจรขับทรานซิสเตอร์ 32 ตัว

ส่วนอ้าง address

จากรูปวงจรที่ต่อมาจาก 8255 เราจะเห็นว่าในการอ้าง address ให้แก่ RAM นั้น เราจะใช้ 74LS569 ซึ่งเป็นไบนารีเคาน์เตอร์ ขนาด 4 บิต จำนวนทั้งสิ้น 3 ตัว เพื่อที่จะสามารถอ้าง address ขนาด 10 บิต ให้แก่ RAM ทั้ง 3 ตัว โดยจะมีการต่อไปยังขา address ของ RAM โดยตรง ขณะที่ทำการเขียนข้อมูลลงใน RAM MCS-51จะสั่งให้ส่วนนี้เป็น High Z และเมื่อจะทำการอ่านข้อมูลจาก RAM MCS-51 จะสั่งให้ ไบนารีเคาน์เตอร์รีเซ็ต และให้ไบนารีเคาน์เตอร์ทำการนับขึ้นเพื่อทำหน้าที่อ้าง address ให้แก่ RAM

ส่วนควบคุมการทำงานของ RAM

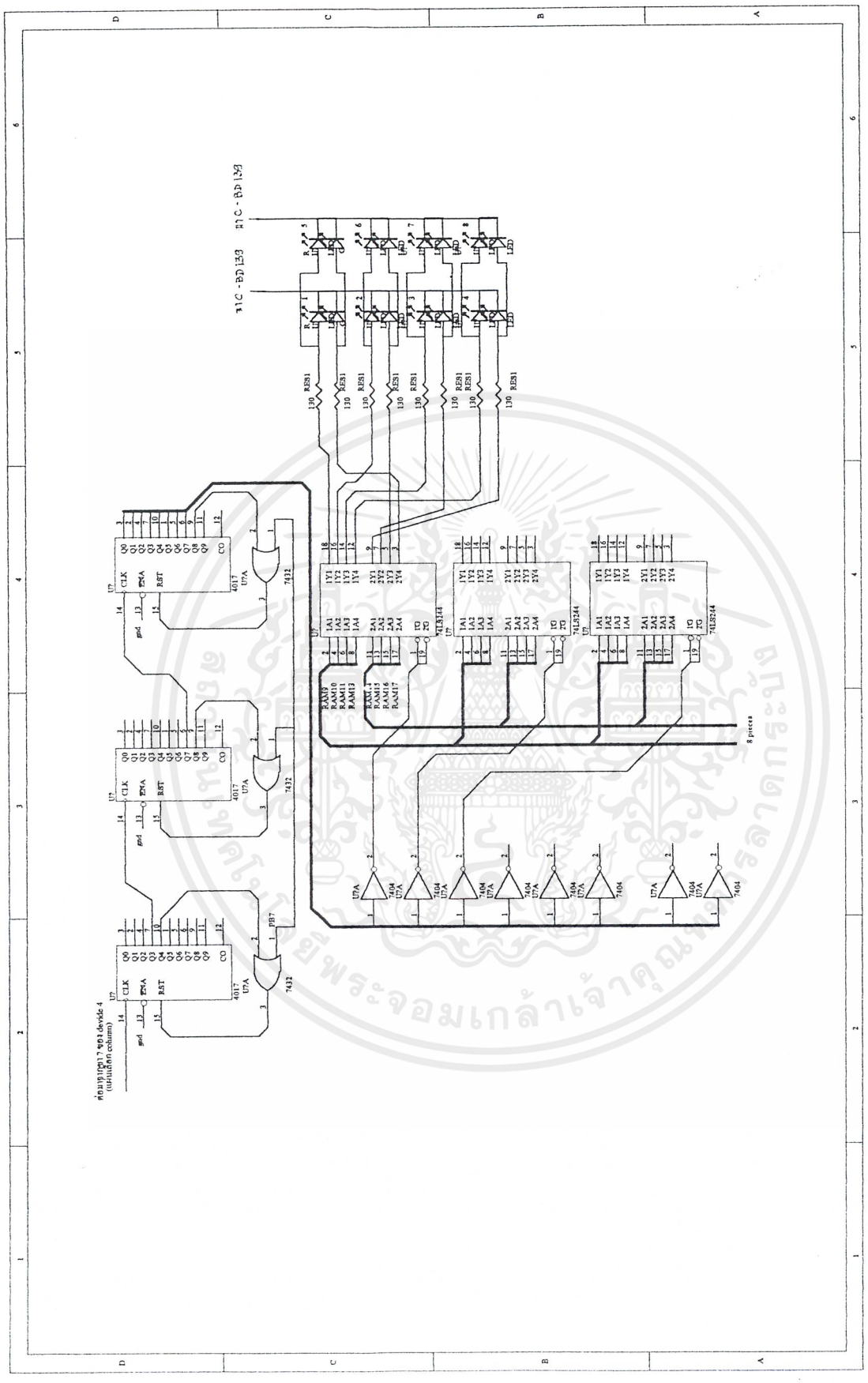
ในขณะที่ทำการอ่านข้อมูลจาก RAM การทำงานของ RAM จะขึ้นอยู่กับส่วนของ Hardware ซึ่งจะมีลักษณะการต่อดังรูปวงจรส่วนควบคุมการทำงานของ RAM โดย RAM แต่ละตัวจะผลัดกันทำงาน โดยเราจะใช้ 4017 ต่อกับขา 18 ของ RAM โดย 4017 จะนับตั้งแต่ 0 – 2 ซึ่งสัญญาณนาฬิกาที่มาป้อนให้กับ 4017 นั้นจะมาจากขา 13 ของ 74LS569 ตัวที่ 3 และผ่านวงจรหาร 4 และหาร 5 เพื่อที่ RAM แต่ละตัวจะได้ไม่ทำงานเร็วเกินไป จนไม่สามารถดูภาพได้ทัน และที่ต้องต่อ 7404 ก็เพราะว่า ขา 18 ของ RAM นั้นต้องการ ลอจิก "0" จึงจะทำงานได้

สำหรับสัญญาณนาฬิกาที่ใช้อ้างอิงให้ระบบการแสดงผลนั้น จะใช้วงจรสร้างพัลส์ของ 555 ซึ่งจะสามารถคำนวณจาก

ใน 1 วินาทีจะต้องแสดงภาพอย่างน้อย	25	ภาพ
ดังนั้นแต่ละภาพจะใช้เวลา 1/25 =	40	ms
การติดของ LED แต่ละครั้ง (4 dots) ต้องใช้	4	พัลส์
การแสดงทั้งบอร์ดต้องใช้ 4 X 1024/4 =	1024	พัลส์
ภายในเวลา 40 ms ต้องส่งพัลส์ออกมา	1024	พัลส์
ในเวลา 1 s ต้องส่งพัลส์ออกมา 1024/40 ms =	25	kHz

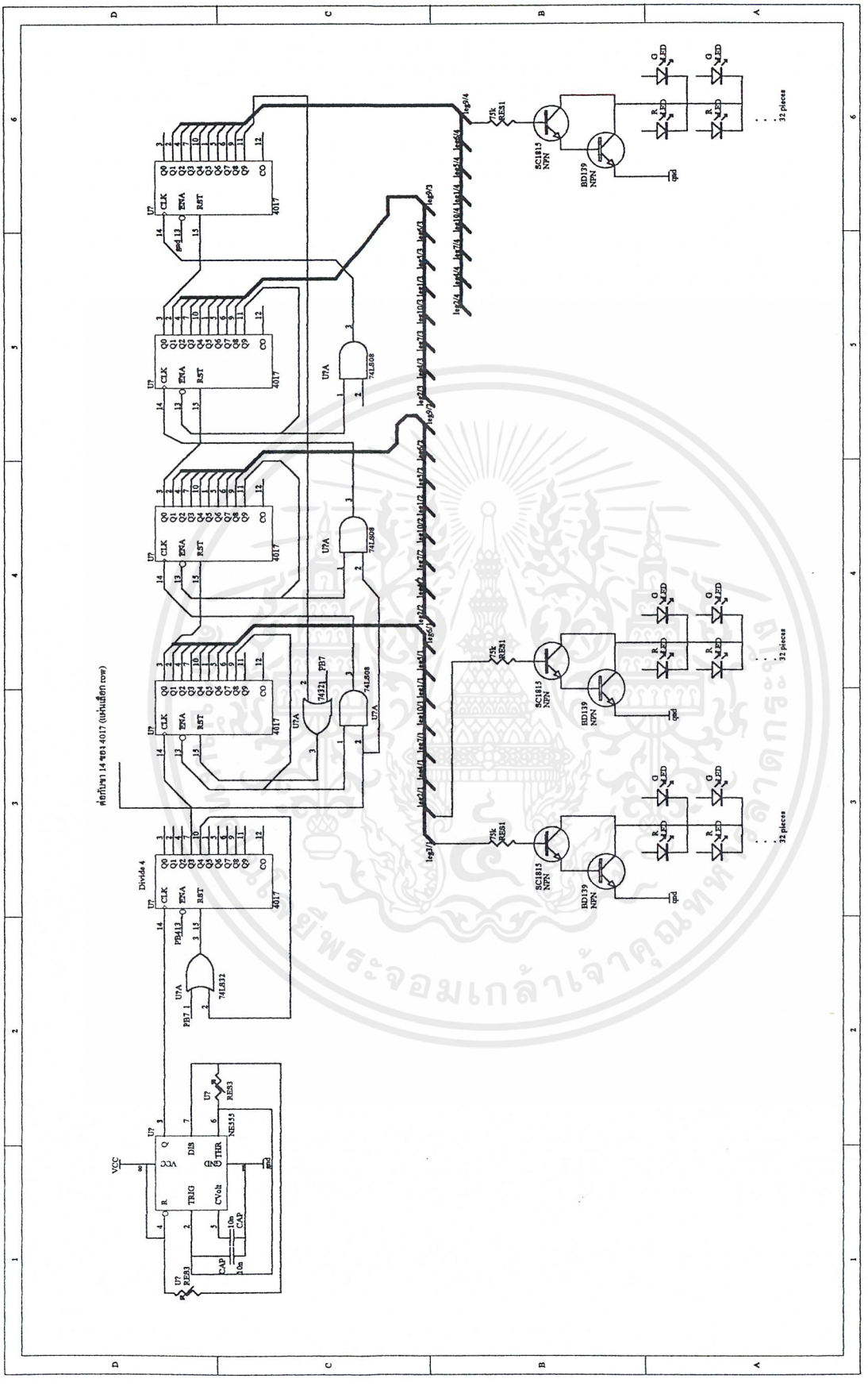
ต้องใช้ความถี่จากสัญญาณนาฬิกาอย่างน้อย 25 kHz

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.3 แสดงวงจรเลือก row

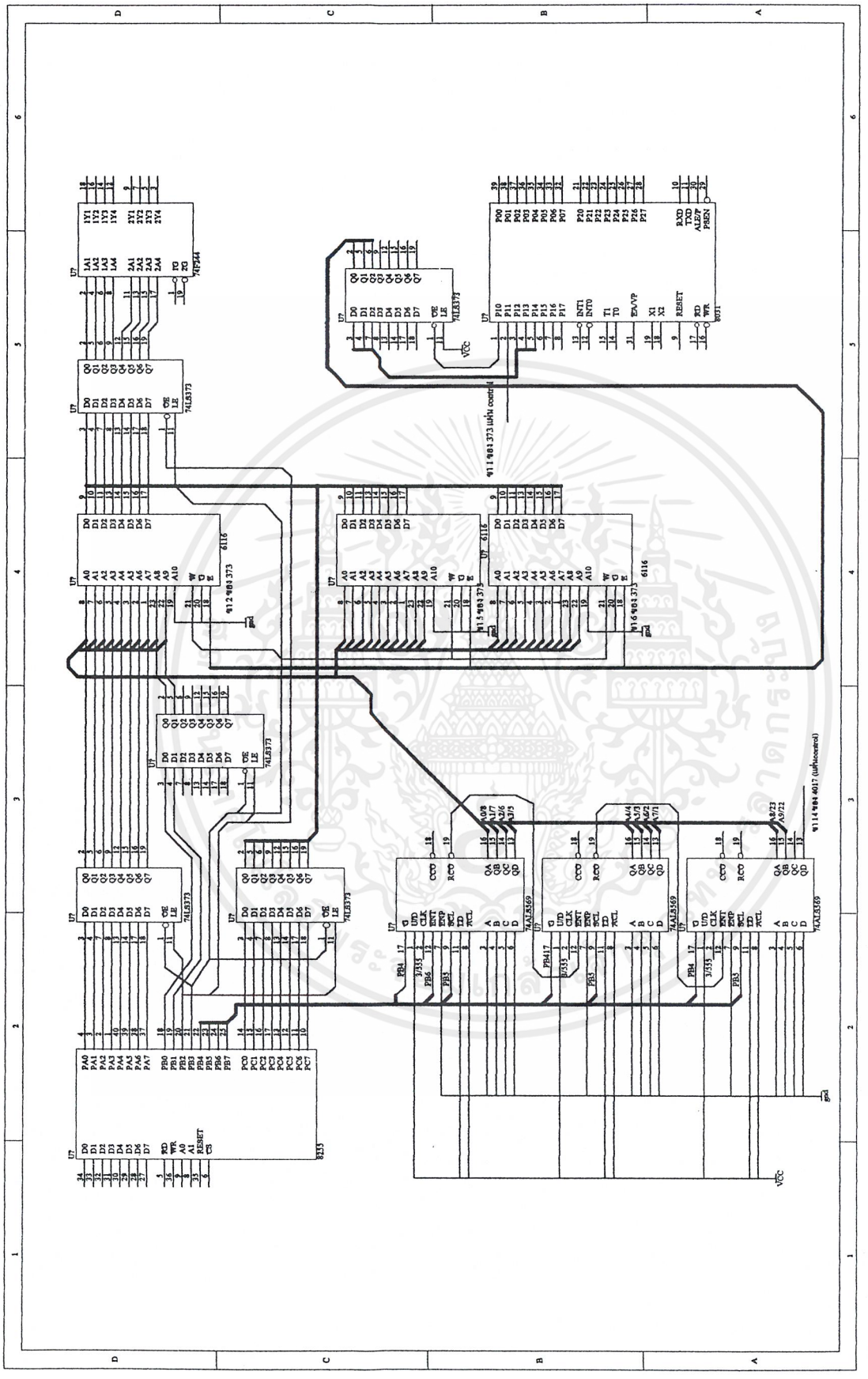
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.3 แสดงวงจรเลือก column

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

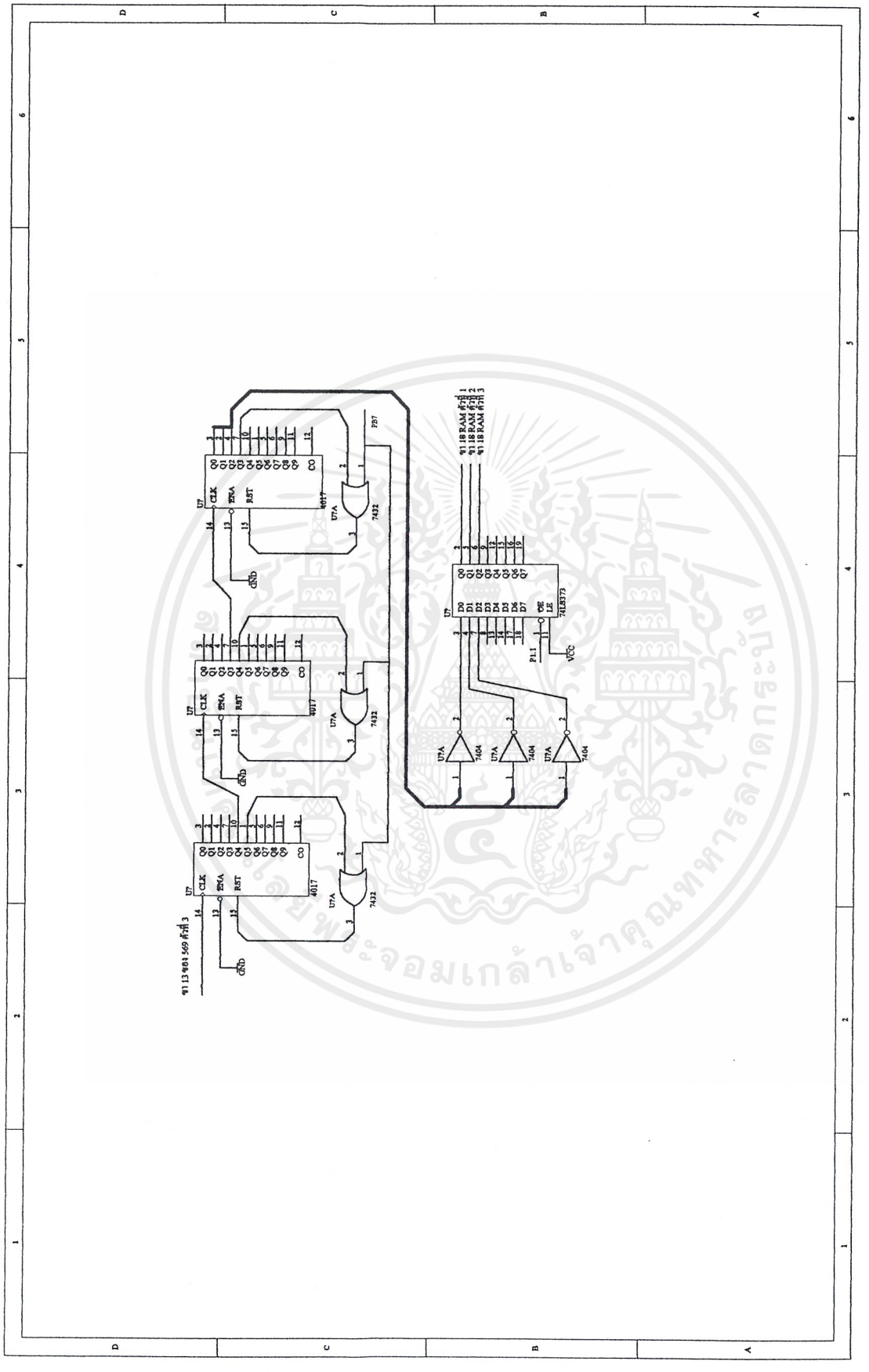
สำนักหอสมุดกลาง พระจอมเกล้าลาดกระบัง



รูปที่ 3.4 แสดงวงจรที่ต่อมาจาก 8255

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

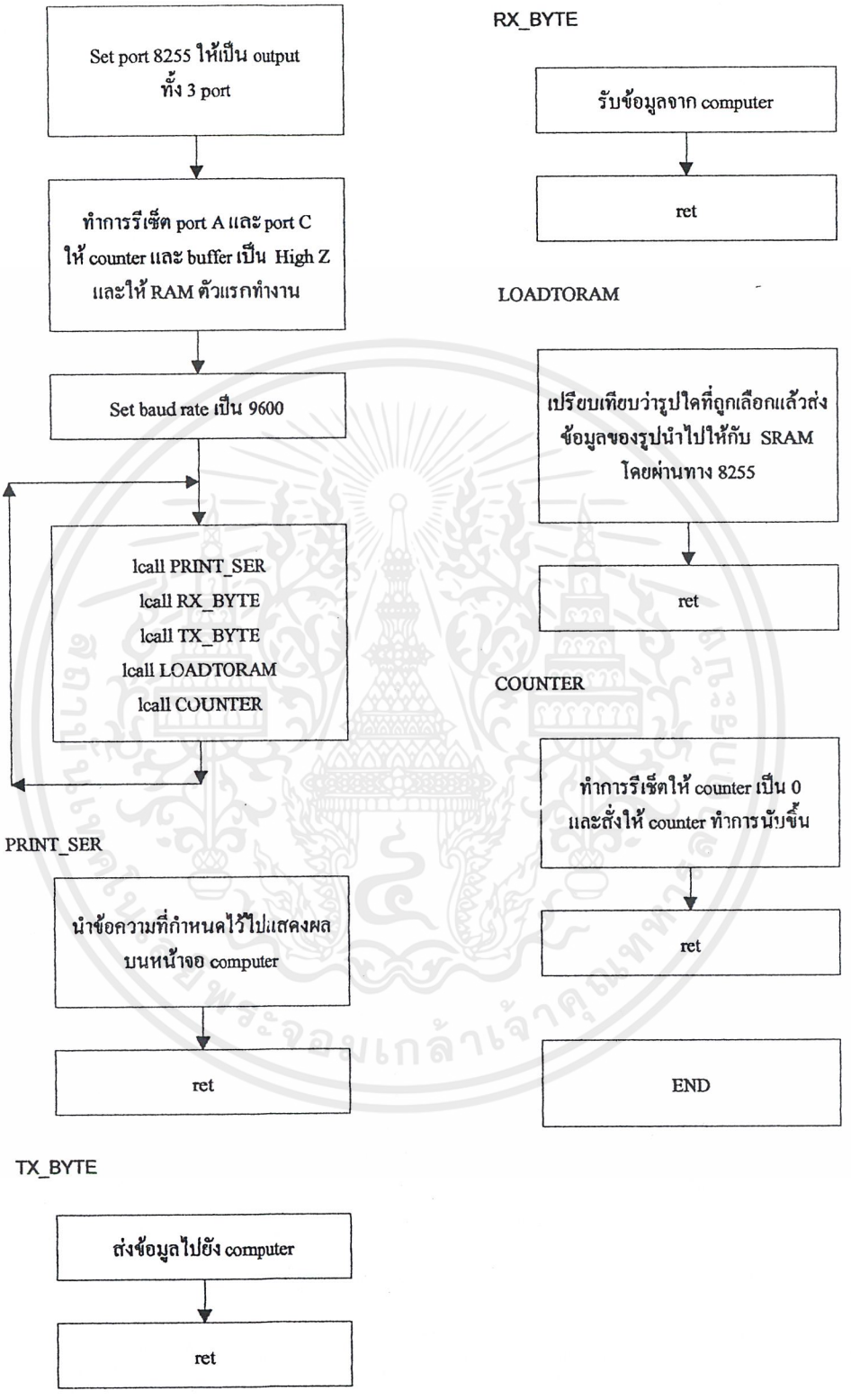
42632



รูปที่ 3.5 แสดงวงจรส่วนควบคุมการทำงานของ RAM

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.4 แผนผังการทำงานอย่างย่อของ MCS –51 ในการควบคุมบอร์ดแสดงผล



รูปที่ 3.6 แผนผังการทำงานอย่างย่อของ MCS –51 ในการควบคุมบอร์ดแสดงผล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 4

ผลการทดลอง

การทดสอบสัญญาณต่าง ๆ

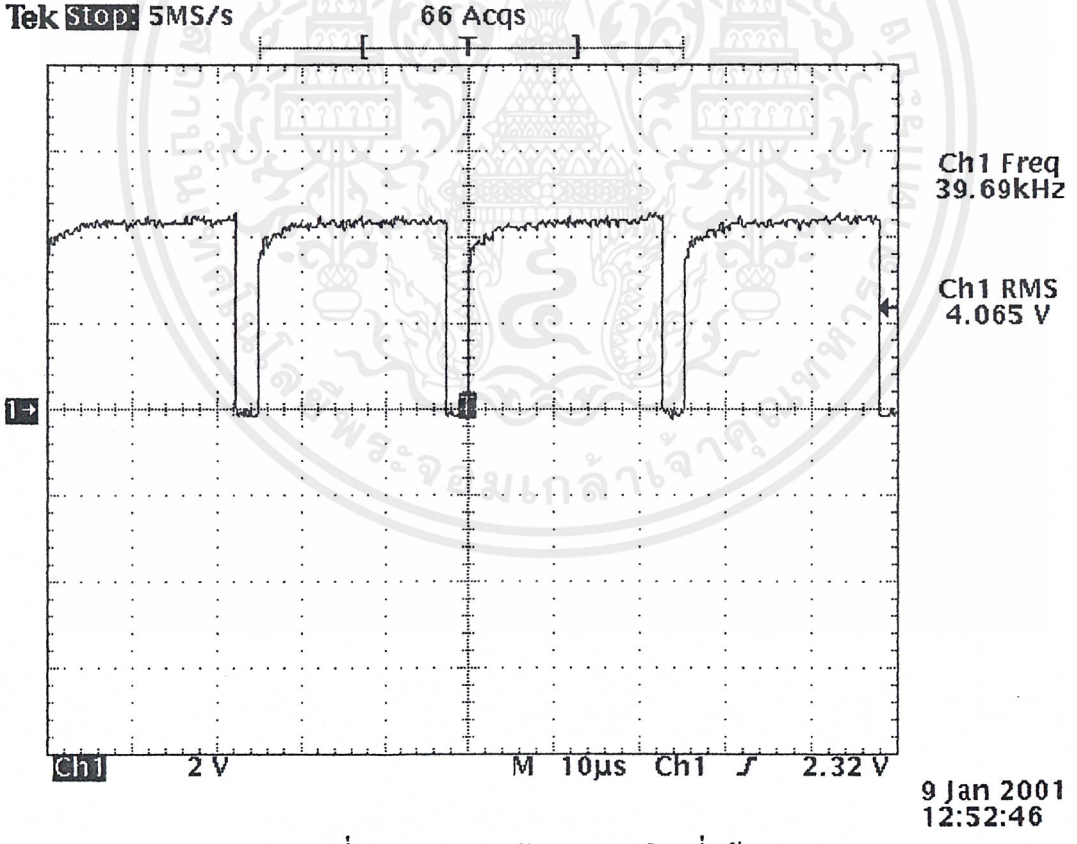
วิธีการทดสอบ

1. ป้อนไฟตรงขนาด 5 V ให้กับวงจร
2. ใช้ Storage scope จับสัญญาณนาฬิกา (pin 1)และสัญญาณข้อมูลของคีย์บอร์ด (pin 2)
3. ทำการกดคีย์ต่างๆ ที่อยู่บนคีย์บอร์ดเพื่อเลือกภาพที่จะแสดง

ผลการทดสอบสัญญาณ

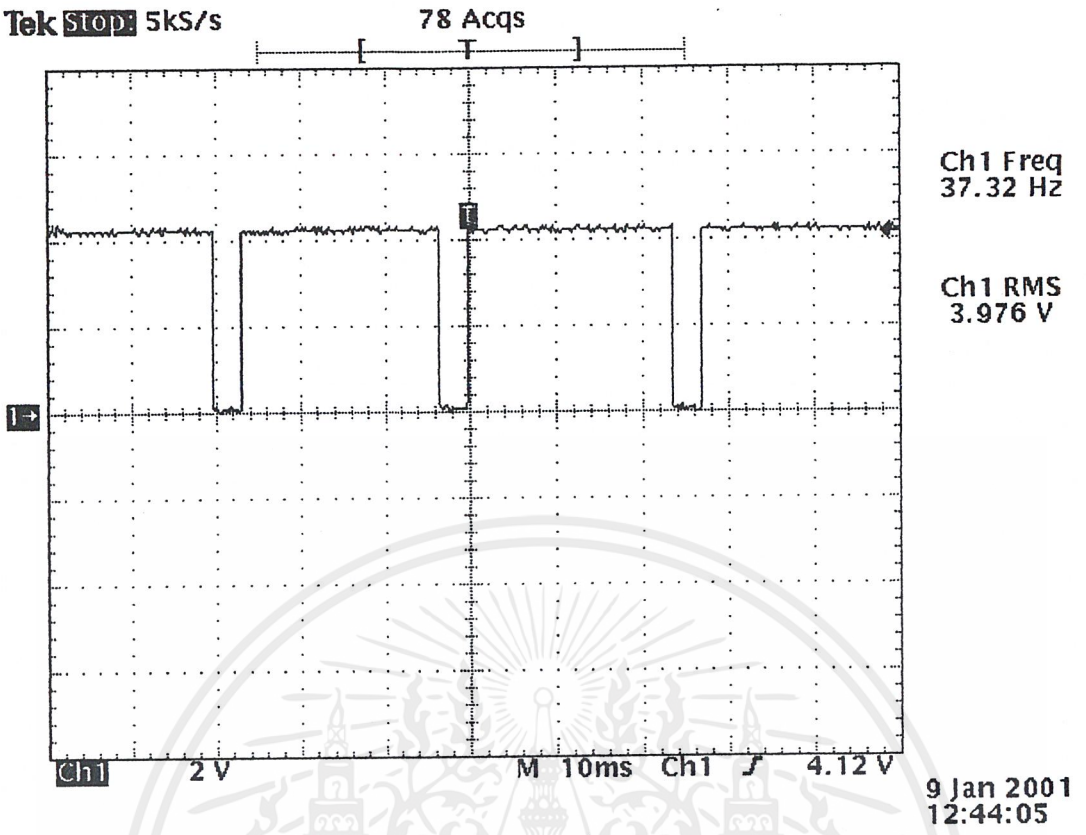
สัญญาณเอาท์พุทของส่วนต่าง ๆ ที่ได้จากการกดคีย์บอร์ดนั้นจะมีลักษณะเป็น pulse โดยจะมีความถี่แตกต่างกัน

ตัวอย่างสัญญาณที่ได้จาก Storage scope แสดงดังรูปที่ 4.1-4.6

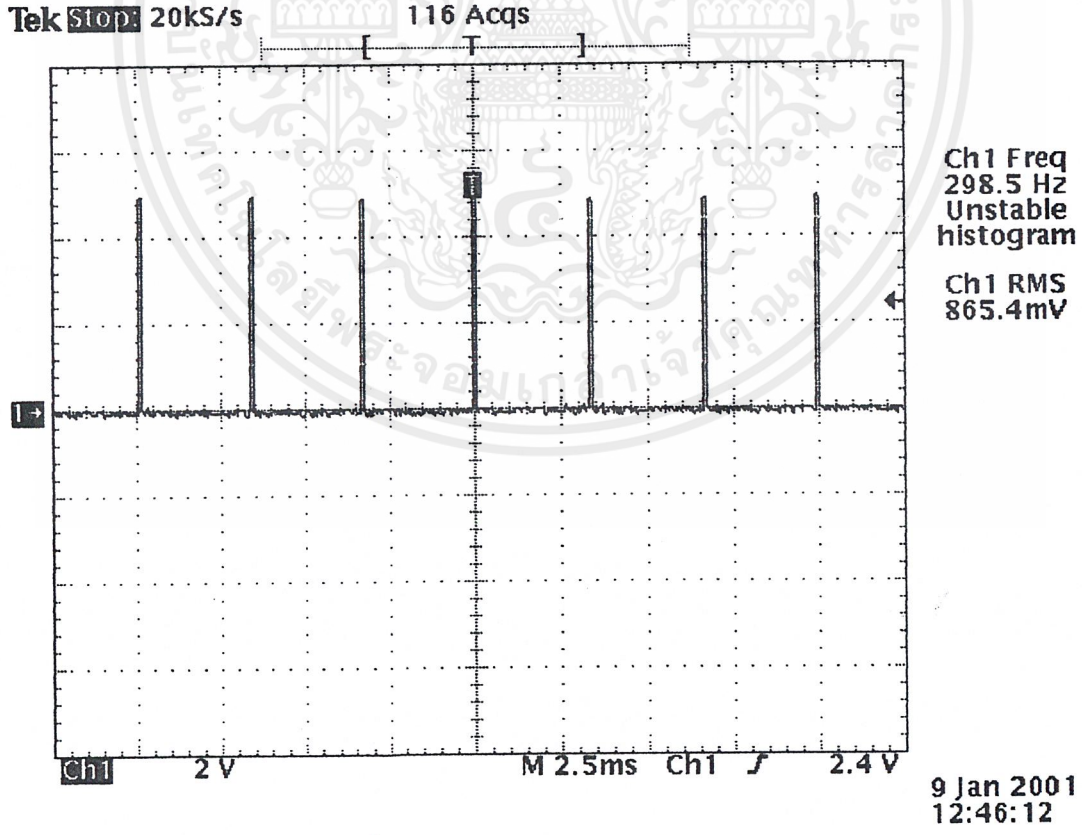


รูปที่ 4.1 แสดงภาพสัญญาณนาฬิกาที่สร้างจาก IC 555

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

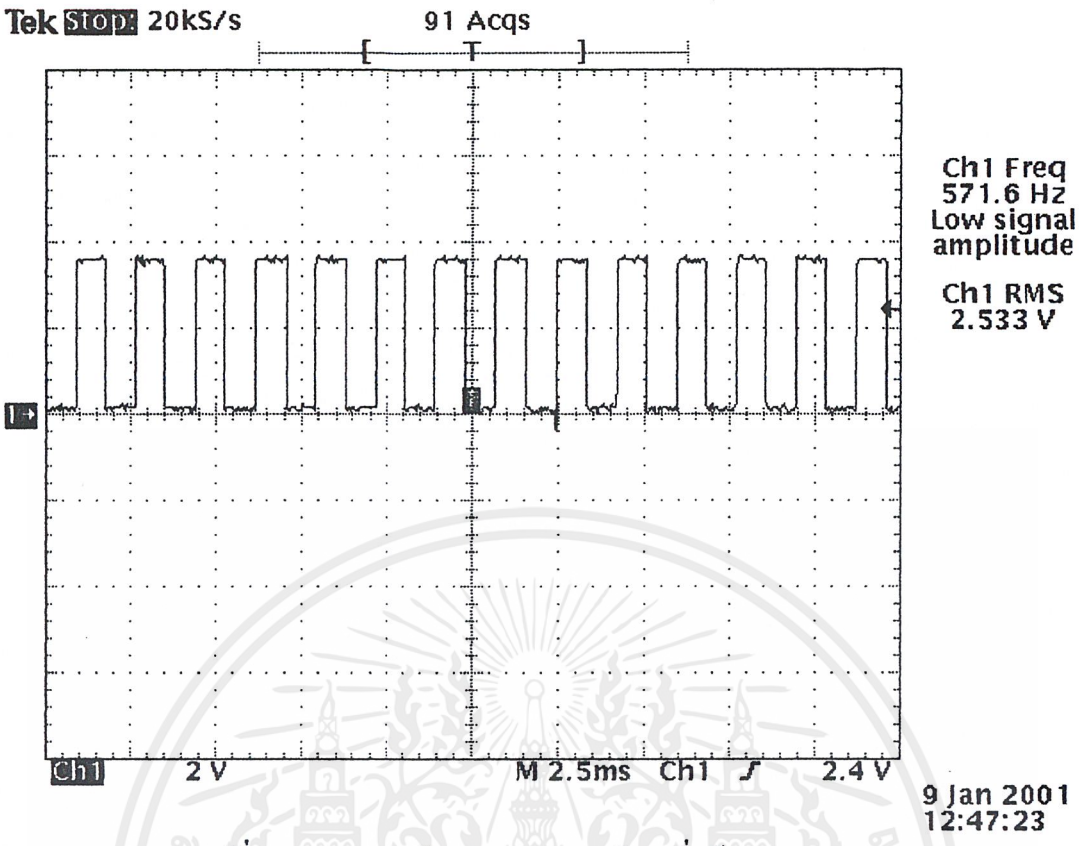


รูปที่ 4.2 แสดงภาพสัญญาณที่ควบคุมการทำงานของ 74F244

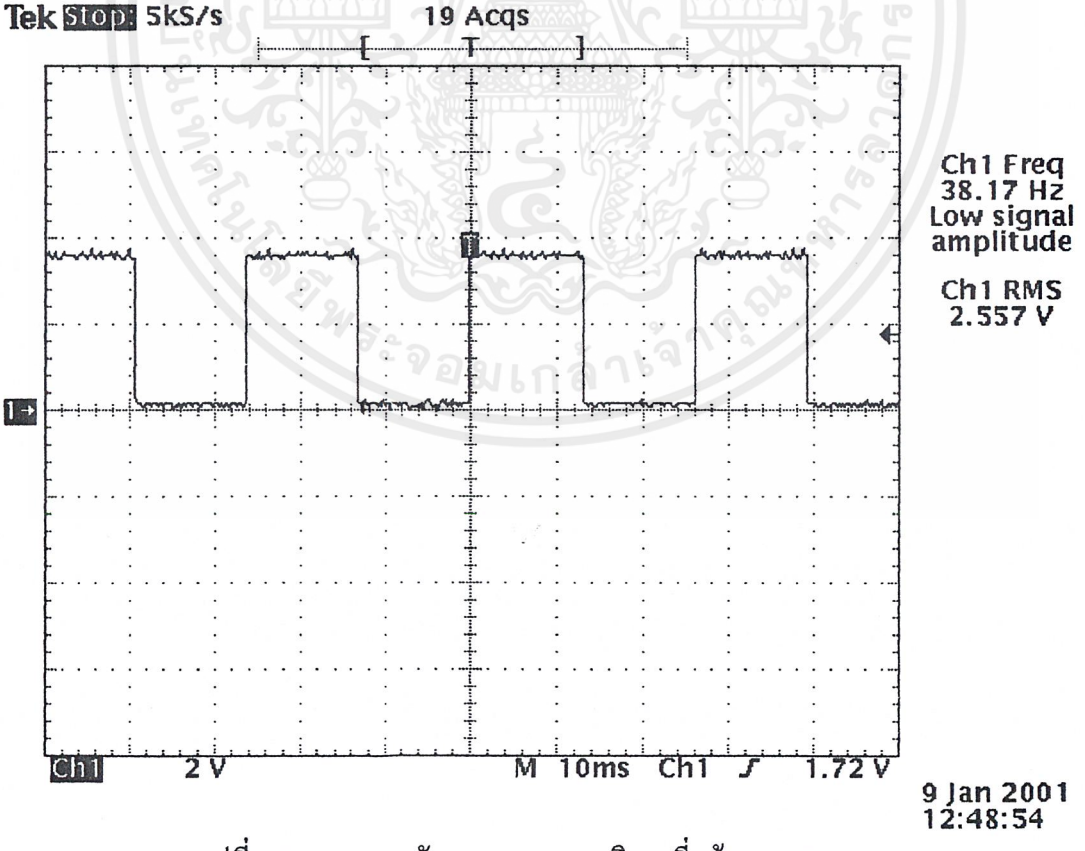


รูปที่ 4.3 แสดงภาพสัญญาณเลือก column

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

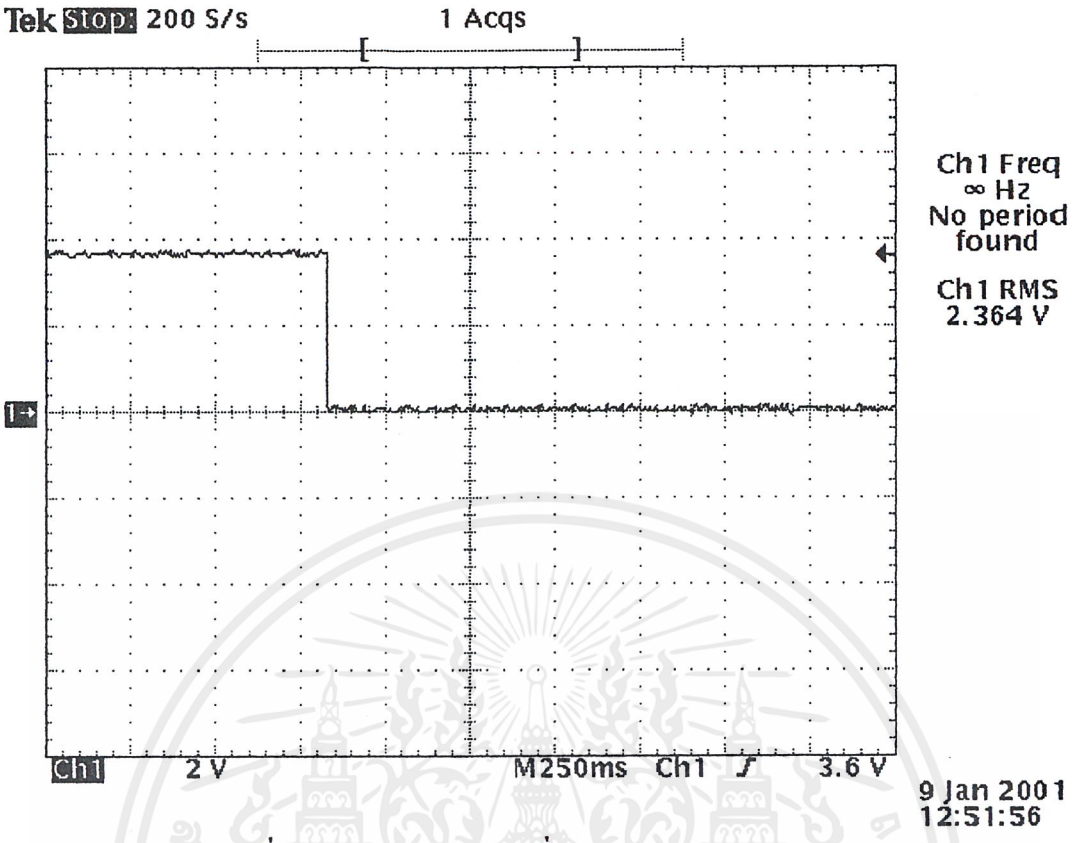


รูปที่ 4.4 แสดงภาพสัญญาณ address บิต 0 ที่สร้างจาก 74LS569



รูปที่ 4.5 แสดงภาพสัญญาณ address บิต 9 ที่สร้างจาก 74LS569

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.6 แสดงภาพสัญญาณที่ควบคุมการทำงานของ RAM

ผลตอบสนองต่อความถี่สัญญาณนาฬิกาของบอร์ดแสดงผล LED

ความถี่ต่ำสุดที่ภาพไม่กระพริบ 25 kHz

ความถี่สูงสุดที่สามารถตอบสนองได้ 380 kHz

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 5

สรุปผลและวิจารณ์

1. ความถี่ที่ LED สามารถตอบสนองได้

ความถี่ต่ำสุดที่ภาพไม่กระพริบ 25 kHz

ความถี่สูงสุดที่สามารถตอบสนองได้ 380 kHz

2. ถ้าใช้ความถี่ 380 kHz ซึ่งเป็นความถี่สูงสุดที่บอร์ดแสดงผล LED สามารถตอบสนองได้ จะสามารถขยายบอร์ดได้ใหญ่ที่สุด คือ

ใน 1 วินาทีจะต้องแสดงภาพอย่างน้อย	25	ภาพ
ดังนั้นแต่ละภาพจะใช้เวลา 1/25 =	40	ms
การติดของ LED แต่ละครั้ง (4 dots) ต้องใช้	4	พัลส์
การแสดงทั้งบอร์ด (A dots)ต้องใช้ 4 X A/4 =	A	พัลส์
ภายในเวลา 40 ms ต้องส่งพัลส์ออกมา	A	พัลส์
ในเวลา 1 ms ต้องส่งพัลส์ออกมา	A/40 ms	kHz

$$A/40\ ms = 380\ k$$

$$A = 15200\ dots$$

3. บอร์ดแสดงผลนี้สามารถปรับปรุงให้สามารถแสดงภาพเคลื่อนไหวที่ละเอียดขึ้นได้โดยการเพิ่มจำนวน RAM ให้มากขึ้น

4. โครงงานนี้สามารถปรับปรุงให้สามารถแสดงภาพเคลื่อนไหวได้มากขึ้น โดยส่งข้อมูลมาจาก computer โดยตรง

5. เนื่องจาก LED ที่ใช้มีราคาค่อนข้างถูก จึงทำให้คุณภาพของบอร์ดที่ได้ไม่ดีเท่าที่ควร

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บรรณานุกรม

สุมทร วิทูสุรพจน์ , “ การใช้งานไมโครคอนโทรลเลอร์ตระกูล 8051 “ , คณะวิศวกรรมศาสตร์ มหาวิทยาลัยสงขลานครินทร์ , 180 หน้า , 2537

ร.อ. ธัชพงษ์ นิลอุบล , “ ไมโครคอนโทรลเลอร์ เล่ม 1 ” , เอ็มแอนด์อี , 194 หน้า , 2541

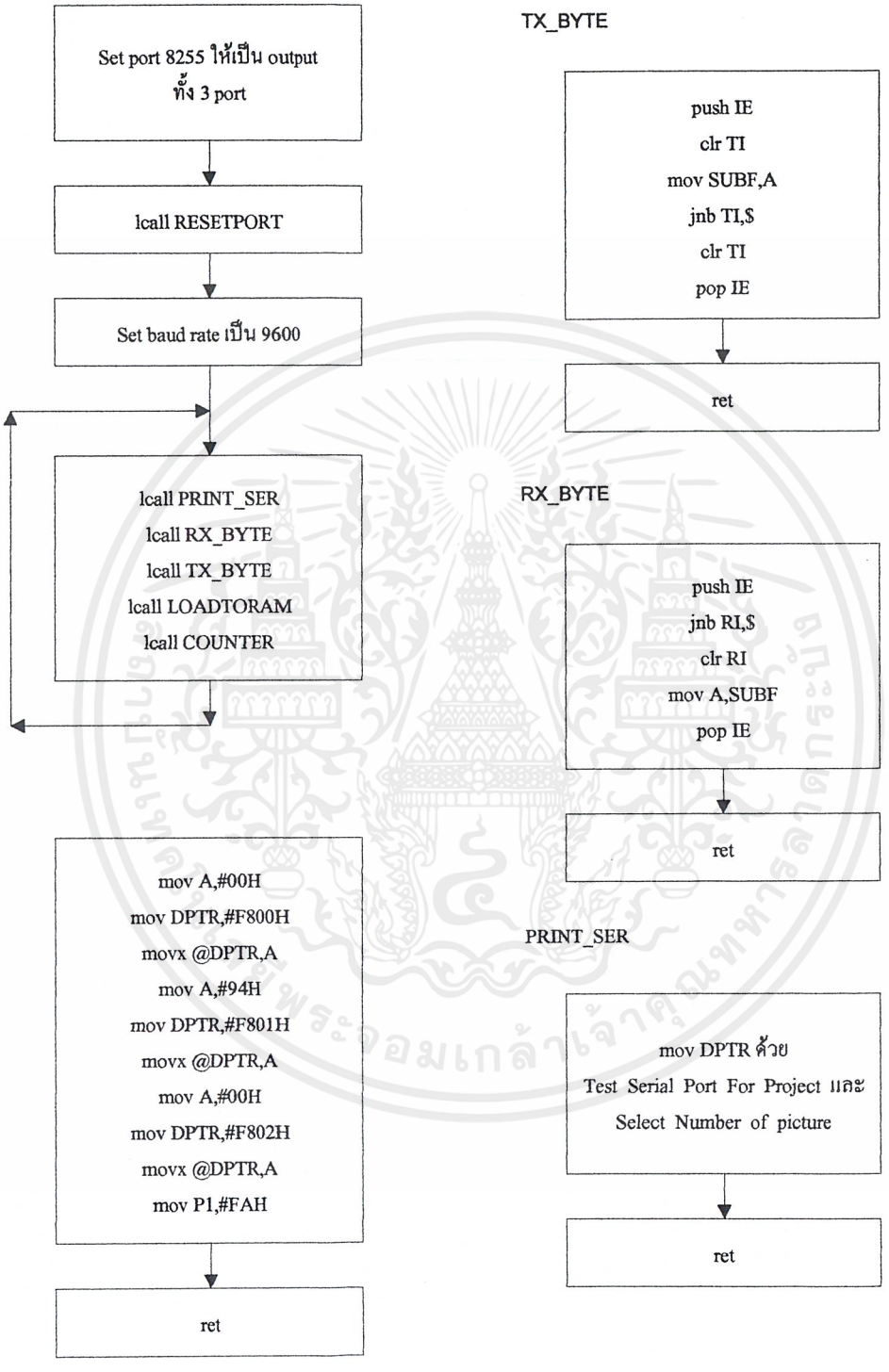
รศ. สมยศ อุณณะปิยะ , .” การประยุกต์ใช้งานไมโครคอนโทรลเลอร์ตระกูล MCS-51 ” , คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง , 248 หน้า ,2541

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก

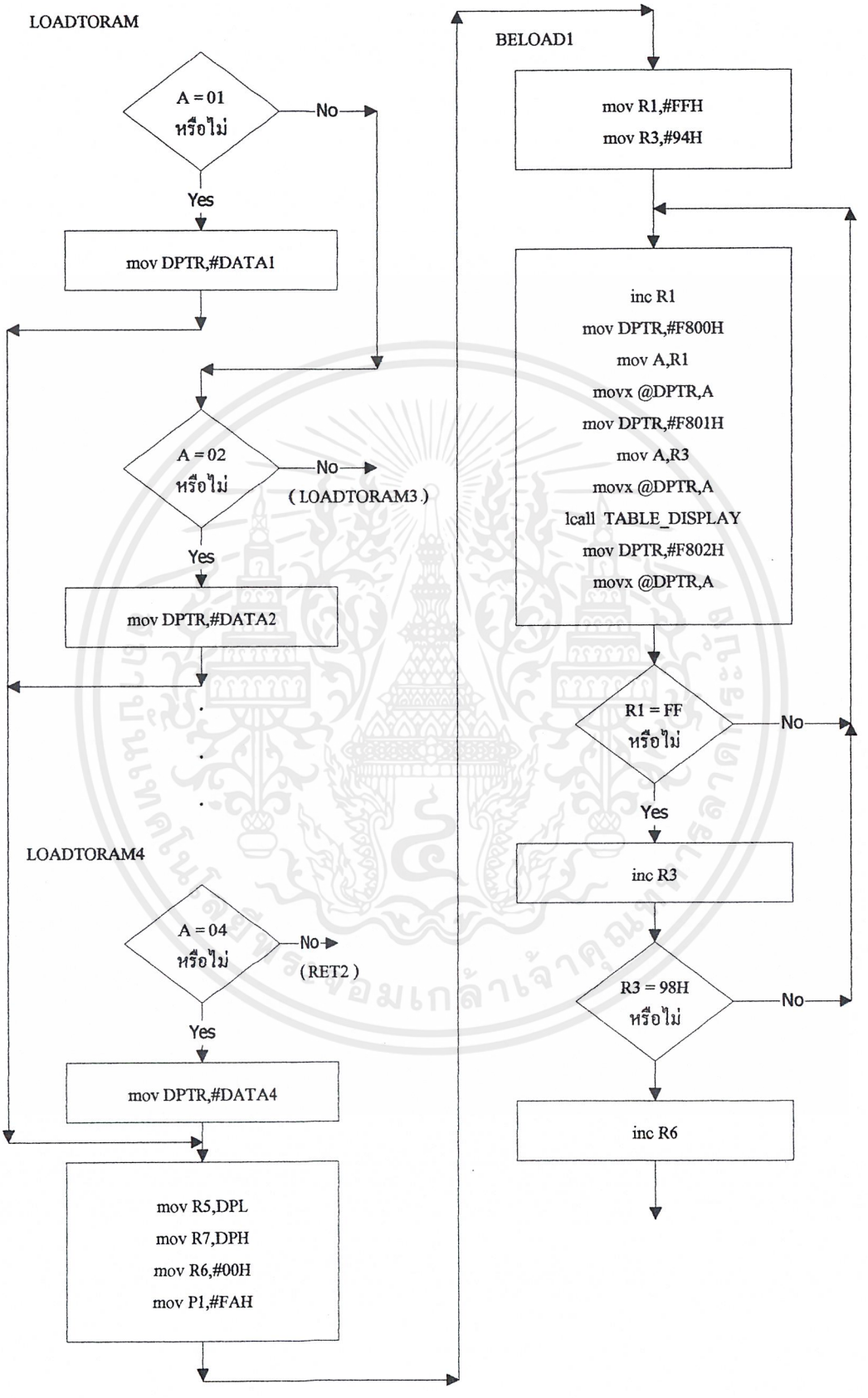
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก ก แผนผังการทำงานของ MCS –51 ในการควบคุมบอร์ดแสดงผล



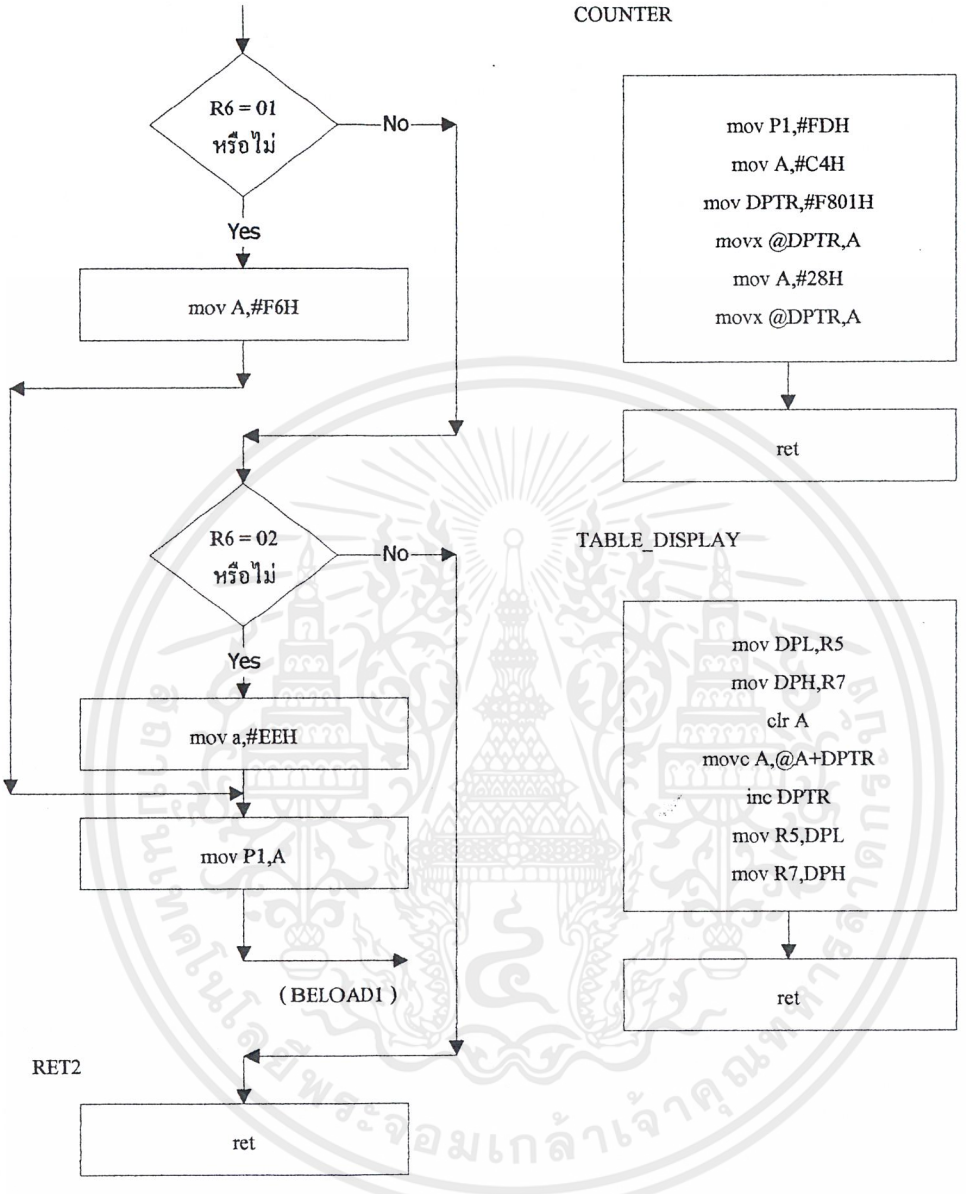
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

แผนผังการทำงานของ MCS –51 ในการควบคุมบอร์ดแสดงผล (ต่อ)



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

แผนผังการของ MCS – 51 ในการควบคุมบอร์ดแสดงผล (ต่อ)



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก ข รายละเอียดโปรแกรมการควบคุมบอร์ดแสดงผล

```
;/***************************/;
;/*  Program for RS-232   */;
;/***************************/;
PORTA   EQU 0F800H
PORTB   EQU 0F801H
PORTC   EQU 0F802H
        ORG 0000H
BEGIN:          MOV DPTR,#0F803H
                MOV A,#80H
                MOVX @DPTR,A
                LCALL RESETPORT
MAIN:           MOV A,#0FDH         ;Set baud rate 9600
                MOV TH1,A
                MOV TL1,A
                MOV TMOD,#00100000B ;Timer1 Mode1
                CLR ES              ;Disable serial interupt
                CLR ET1             ;Disable timer1 interupt
                SETB TR1            ;Set timer1 control
                MOV SCON,#01010000B ;Serial mode1
LOOP:           LCALL PRINT_SER
                LCALL RX_BYTE
                LCALL TX_BYTE
                LCALL LOADTORAM
                LCALL COUNTER
                SJMP LOOP
  ;*******************************
   ;*    RESETPORT          *
   ;*******************************
RESETPORT:  MOV A,#00H
                MOV DPTR,#PORTA
                MOVX @DPTR,A
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รายละเอียดโปรแกรมการควบคุมบอร์ดแสดงผล (ต่อ)

```
                MOV DPTR,#PORTB
                MOV A,#10010100B
                MOVX @DPTR,A
                MOV A,#00H
                MOV DPTR,#PORTC
                MOVX @DPTR,A
                MOV P1,#11111010B
                RET
   ;*****************************;
   ;* send 1-byte to RS-232  *;
   ;* input : ACC          *;
   ;* output: Serial Port   *;
   ;*****************************;
TX_BYTE:    PUSH IE
            CLR TI
            MOV SBUF,A
            JNB TI,$
            CLR TI
            POP IE
            RET
   ;*****************************;
   ;* Recieve data from RS-232 *;
   ;*   input : Serial Port   *;
   ;*   output: Acc         *;
   ;*****************************;
RX_BYTE:    PUSH IE
            JNB RI,$
            CLR RI
            MOV A,SBUF
            POP IE
            RET
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รายละเอียดโปรแกรมการควบคุมบอร์ดแสดงผล (ต่อ)

```
   ;*****************************;
   ;* Print data to serial port *;
   ;*    to display buffer      *;
   ;* Usage : LCALL PRINT_SER   *;
   ;*      : DB 'XXXX',00       *;
   ;* Register : ACC            *;
   ;* Last byte must be 00      *;
   ;*****************************;
PRINT_SER:   MOV DPTR,#TABLE
PRINT1:      CLR A
             MOVC A,@A+DPTR
             CJNE A,#00H,PRINT2
             SJMP PRINT3
PRINT2:      LCALL TX_BYTE
             INC DPTR
             SJMP PRINT1
PRINT3:      RET
TABLE:       DB 0DH,0AH
             DB "   Test Serial Port for Project",0DH,0AH
             DB " Select Number of Pictures (1-4) :"
             DB 00H
   ;*******************************
   ;     load data to ram
   ;*******************************
LOADTORAM: CJNE A,#'1',LOADTORAM2
             MOV DPTR,#DATA1
             SJMP BELOAD
LOADTORAM2:CJNE A,#'2',LOADTORAM3
             MOV DPTR,#DATA2
             SJMP BELOAD
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รายละเอียดโปรแกรมการควบคุมบอร์ดแสดงผล (ต่อ)

```
LOADTORAM3:CJNE A,#'3',LOADTORAM4
           MOV DPTR,#DATA3
           SJMP BELOAD
LOADTORAM4:CJNE A,#'4',RET2
           MOV DPTR,#DATA4
BELOAD:    MOV R5,DPL
           MOV R7,DPH
           MOV R6,#00H
           MOV P1,#11111010B
BELOAD1:   MOV R1,#0FFH
           MOV R3,#10010100B   ;bit2 = LE OF 74LS373
ADDRESS:   INC R1
           MOV DPTR,#PORTA
           MOV A,R1
           MOVX @DPTR,A
           MOV DPTR,#PORTB
           MOV A,R3
           MOVX @DPTR,A
           LCALL TABLE_DISPLAY
           MOV DPTR,#PORTC
           MOVX @DPTR,A
           CJNE R1,#0FFH,ADDRESS
           INC R3
           CJNE R3,#10011000B,ADDRESS
           INC R6
           CJNE R6,#01H,ADDRESS1
           MOV A,#11110110B
           SJMP ADDRESS2
ADDRESS1:  CJNE R6,#02H,RET2
           MOV A,#11101110B
           SJMP ADDRESS2
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รายละเอียดโปรแกรมการควบคุมบอร์ดแสดงผล (ต่อ)

```
ADDRESS2:    MOV P1,A
             LJMP BELOAD1
RET2:        RET
;#####################################
;        reset and count up
;#####################################
COUNTER:     MOV P1,#11111101B
             MOV A,#11001000B
             MOV DPTR,#PORTB    ;reset counter
             MOVX @DPTR,A
             MOV A,#00101000B   ;up counter
             MOVX @DPTR,A
             RET
;@@@
```