ภาควิชาครุศาสตร์วิศวกรรม

คณะครุศาสตร์อุตสาหกรรม

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ใบรับรองปริญญานิพนธ์

ชื่อหัวข้อ ชุดทดลองการเชื่อมต่อไมโครคอนโทรลเลอร์ MCS-51 กับอุปกรณ์ภายนอก

Microcontroller MCS-51 Interfacing Training System

ชื่อนักศึกษา 1. นายพีระ แจ้งศิริกุล รหัสประจำตัว 42035229

2. นายเพ็ชรพรชัย หน่ายคอน รหัสประจำตัว 42035230

3. นายอรรณพ อุยโต รหัสประจำตัว 42035246

หลักสูตร ครุศาสตร์อุตสาหกรรมบัณฑิต **สาขาวิชา** วิศวกรรมโทรคมนาคม

อาจารย์ที่ปรึกษา อาจารย์ปิยะ ศุภวราสุวัฒน์

อาจารย์ที่ปรึกษาร่วม อาจารย์ไพบูลย์ พวงวงศ์ตระกูล

คณะกรรมการสอบปริญญานิพนธ์	ลายมือชื่อ
1. อาจารย์ปิยะ ศุภวราสุวัฒน์	....
2. อาจารย์ไพบูลย์ พวงวงศ์ตระกูล	....
3. อาจารย์พงษ์เกียรติ เชษฐพิทักษ์สกุล	....
4. อาจารย์ปิยะ จิตธรรมมาภิรมย์	....
5. อาจารย์อมรชัย ชัยชนะ	....

วัน/เดือน/ปีที่สอบ วันพฤหัสบดีที่ 19 เมษายน พ.ศ. 2544 เวลา 10.00 น.

สถานที่สอบ ห้อง ค.311 คณะครุศาสตร์อุตสาหกรรม สจล.

ภาควิชารับรองแล้ว

ลงนาม..

(ผศ.วิสุทธิ์ อธิพรธรรม)

หัวหน้าภาควิชาครุศาสตร์วิศวกรรม

วันที่....๑....เดือน....พ.ค.....พ.ศ...๒๕๔๔...

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สำนักหอสมุดกลาง พระจอมเกล้าลาดกระบัง

ปริญญานิพนธ์

ชุดทดลองการเชื่อมต่อไมโครคอนโทรลเลอร์ MCS-51 กับอุปกรณ์ภายนอก

MICROCONTROLLER MCS-51 INTERFACING TRAINING SYSTEM

T040184

นายพีระ แจ้งศิริกุล

นายเพ็ชรพรชัย หน่ายคอน

นายอรรณพ อุยโต

ปริญญานิพนธ์ฉบับนี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรครุศาสตร์อุตสาหกรรมบัณฑิต

สาขาวิชาวิศวกรรมโทรคมนาคม

ภาควิชาครุศาสตร์วิศวกรรม คณะครุศาสตร์อุตสาหกรรม

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2543

ปพ.

เลขหมู่

เลขทะเบียน 40184

วัน, เดือน, ปี

b. 11092117

i.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ปริญญานิพนธ์

เรื่อง ชุดทดลองการเชื่อมต่อไมโครคอนโทรลเลอร์ MCS-51 กับอุปกรณ์ภายนอก

Microcontroller MCS-51 Interfacing Training System

วัตถุประสงค์

1. เพื่อศึกษาการทำงานของไมโครคอนโทรลเลอร์ตระกูล MCS-51
2. เพื่อออกแบบชุดทดลองไมโครคอนโทรลเลอร์ตระกูล MCS-51
3. เพื่อสร้างชุดทดลองไมโครคอนโทรลเลอร์ตระกูล MCS-51
4. เพื่อทดสอบการทำงานชุดทดลองไมโครคอนโทรลเลอร์ตระกูล MCS-51
5. เพื่อนำชุดทดลองไมโครคอนโทรลเลอร์ตระกูล MCS-51 ไปใช้งานได้จริง

ประโยชน์ที่คาดว่าจะได้รับ

1. เรียนรู้และเข้าใจการทำงานของไมโครคอนโทรลเลอร์ตระกูล MCS-51 ได้
2. เรียนรู้และเข้าใจการออกแบบชุดทดลองไมโครคอนโทรลเลอร์ตระกูล MCS-51 ได้
3. สามารถสร้างชุดทดลองไมโครคอนโทรลเลอร์ตระกูล MCS-51 ได้
4. สามารถทดสอบการทำงานชุดทดลองไมโครคอนโทรลเลอร์ตระกูล MCS-51 ได้
5. ได้ชุดทดลองการเชื่อมต่อไมโครคอนโทรลเลอร์ MCS-51 กับอุปกรณ์ภายนอก 1 ชุด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ชื่อหัวข้อ	ชุดทดลองการเชื่อมต่อไมโครคอนโทรลเลอร์ MCS-51 กับอุปกรณ์ภายนอก
นักศึกษา	นายพีระ แจ้งศิริกุล นายเพ็ชรพรชัย หน่ายคอน นายอรรณพ อุยโต
อาจารย์ที่ปรึกษา	อาจารย์ปิยะ ศุภวราสุวัฒน์
อาจารย์ที่ปรึกษาร่วม	อาจารย์ไพบูลย์ พวงวงศ์ตระกูล
หลักสูตร	ครุศาสตร์อุตสาหกรรมบัณฑิต
สาขาวิชา	วิศวกรรมโทรคมนาคม
ปีการศึกษา	2543

บทคัดย่อ

ปริญญานิพนธ์ฉบับนี้นำเสนอโครงงานชุดทดลองการเชื่อมต่อไมโครคอนโทรลเลอร์ MCS-51 กับอุปกรณ์ภายนอก ซึ่งมีวงจรเชื่อมต่อทั้งหมด 14 วงจร คือ วงจรแสดงผลโดยใช้ไดโอดเปล่งแสงแบบคาโถดร่วม, วงจรแสดงผลโดยใช้ไดโอดเปล่งแสงแบบอาโนดร่วม, วงจรแสดงผลแบบผลึกเหลวขนาด 16 ตัวอักษร, วงจรแสดงผลแบบ 7 ส่วน, วงจรแสดงผลแอลอีดีแบบเมตริกซ์, วงจรมอเตอร์ไฟฟ้ากระแสตรง, วงจรสเต็ปปิ้งมอเตอร์, วงจรคีย์สวิตช์, วงจรสวิตช์เดี่ยว, วงจรเชื่อมโยงทางแสง, วงจรแปลงสัญญาณแอนะลอกเป็นสัญญาณดิจิตอล, วงจรแปลงสัญญาณดิจิตอลเป็นสัญญาณแอนะลอก, วงจรควบคุมรีเลย์ และ วงจรการเชื่อมต่อแบบ I^2C การทดลองแต่ละวงจรสามารถทำได้โดยการโปรแกรมผ่านพอร์ตของ AT89S8252 โดยตรง การทดลองแบ่งเป็น 2 รูปแบบคือ ทดลองการใช้คำสั่งในการควบคุมการทำงานของวงจรภาคแสดงผล และทดลองการใช้คำสั่งในการควบคุมการทำงานของวงจรภาคแสดงผลเชื่อมต่อกับวงจรภาคอินพุต

โครงงานชุดทดลองนี้เหมาะสมที่จะใช้ประกอบการเรียนการสอนในรายวิชาไมโครโปรเซสเซอร์ เพื่อให้นักศึกษามีทักษะในการใช้งานไมโครคอนโทรลเลอร์เชื่อมต่อกับอุปกรณ์อินพุตและเอาต์พุตมากขึ้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Thesis Title	Microcontroller MCS-51 Interfacing Training System
Students	Mr. Peera Changsirikul
	Mr. Petpornchai Naikon
	Mr. Annop Auito
Advisor	Mr. Piya Supavarasuwat
Co-Advisor	Mr. Paiboon Pongwongtragull
Education Level	Bachelor of Science in Industrial Education
Program in	Telecommunication Engineering
Academic	2000

ABSTRACT

This thesis presents Microcontroller MCS-51 Interfacing Training System. There are 14 external interfaced circuits such as, common cathode LED, common anode LED, 16 digit LCD, 7-segments display, 8x8 dot metrix LED, DC-motor, stepping motor, key switch, single switch, opto-isolator, analog to digital converter, digital to analog converter, relay control and I^2C interface circuit .

To testing of each circuit is able to test by download program into Microcontroller AT89S8252 directly. The testing is divided for two types. The first type is the testing for using the instruction to control the operation of display section. The second testing is the testing for using the instruction to interface the display section to the input data section.

This project is suitable for Microcontroller subject in Bacherlor's degree level, it can improve the skill and knowledge of student well.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กิตติกรรมประกาศ

ปริญญานิพนธ์ฉบับนี้ สำเร็จลุล่วงลงได้ด้วยดี เนื่องมาจากความอนุเคราะห์ของอาจารย์ที่ปรึกษา อาจารย์ที่ปรึกษาร่วม และคณาจารย์ภาควิชาครุศาสตร์วิศวกรรมทุกท่านที่ได้กรุณาให้คำปรึกษา ข้อเสนอแนะและเอื้อเฟื้อเครื่องมือในการทดลองต่างๆ เพื่อนที่อนุเคราะห์อุปกรณ์หลายๆ อย่าง และสมาชิกในกลุ่มปริญญานิพนธ์ทุกคนที่ให้ความร่วมมือในการทำงานเป็นอย่างดีโดยเฉพาะคุณพ่อคุณแม่ที่ให้การสนับสนุนทางด้านเงินทุน และให้โอกาสทางการศึกษาจนมาถึงจุดนี้ได้ จึงขอขอบคุณไว้ ณ ที่นี้ด้วย

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ

เรื่อง	หน้า
บทคัดย่อภาษาไทย	I
บทคัดย่อภาษาอังกฤษ	II
กิตติกรรมประกาศ	III
สารบัญ	IV
สารบัญตาราง	VIII
สารบัญรูป	IX
บทที่ 1 บทนำ	1
1.1 ความเป็นมาและความสำคัญของวิทยานิพนธ์	1
1.2 ขีดความสามารถของโครงงาน	1
1.3 เนื้อหาโดยสังเขป	2
บทที่ 2 ทฤษฎีและหลักการ	3
2.1 กล่าวนำ	3
2.2 ความแตกต่างของไมโครคอนโทรลเลอร์กับไมโครโปรเซสเซอร์	3
2.3 คุณสมบัติทั่วไปของไมโครคอนโทรลเลอร์ตระกูล MCS-51	5
2.4 โครงสร้างภายนอกของไมโครคอนโทรลเลอร์ตระกูล MCS-51	6
2.5 คุณสมบัติเทคนิคของไมโครคอนโทรลเลอร์AT89S8252	9
2.6 โครงสร้างและการทำงานของพอร์ต	10
2.6.1 การใช้งานพอร์ตอินพุต	10
2.6.2 การใช้งานพอร์ตเอาต์พุต	10
2.7 ส่วนเชื่อมต่ออุปกรณ์ SPI	11
2.7.1 คุณสมบัติของ SPI ในไมโครคอนโทรลเลอร์ AT89S8252	12
2.7.2 ขาสัญญาณของการเชื่อมต่อแบบ SPI	13
2.7.3 การทำงานของ SPI	13
2.8 การติดต่อสื่อสารโดยระบบบัส I^2C	13
2.8.1 หลักการของบัส I^2C	14
2.8.2 สภาวะที่เกิดขึ้นบนบัส I^2C	14

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ (ต่อ)

เรื่อง	หน้า
2.8.3 การต่ออุปกรณ์ระบบบัส I^2C กับไมโครคอนโทรลเลอร์	15
2.8.4 การเขียนโปรแกรมติดต่อบนระบบบัส	15
2.9 ไดโอดเปล่งแสง	15
2.10 จอแสดงผลแบบ 7 ส่วน	17
2.11 จอแสดงผลแบบผลึกเหลว	21
2.11.1 ข้อดีของจอแสดงผลแบบผลึกเหลว	23
2.11.2 โครงสร้างของจอแสดงผลแบบผลึกเหลว	23
2.11.3 แบบต่าง ๆ ของจอแสดงผล	23
2.11.4 จอแสดงผลแบบผลึกเหลวแบบดอตเมตริกซ์	24
2.12 สเต็ปปิ้งมอเตอร์	24
2.12.1 โหมดเต็มขั้น	27
2.12.2 โหมดครึ่งขั้น	27
2.13 การแปลงสัญญาณดิจิตอลเป็นสัญญาณแอนะลอก	28
2.14 การเชื่อมโยงทางแสง	31
2.14.1 คุณสมบัติต่างของอุปกรณ์เชื่อมโยงทางแสง	31
2.14.2 วงจรเปลี่ยนระดับแรงดัน	33
2.15 แอลอีดีแบบเมตริกซ์	35
2.15.1 หลักการตรวจภาพบนแอลอีดีแบบเมตริกซ์	36
2.15.2 ปัญหาในการกวาดทางคอลัมน์	36
2.15.3 การคิดตัวอักษร	37
2.16 แป้นพิมพ์	37
2.16.1 การกระเด้งของหน้าสัมผัส	39
2.16.2 องค์ประกอบอันเกิดจากสวิตซ์ตัวอักษรบนแป้น	40
2.16.3 โปรแกรมสำหรับแป้นพิมพ์	40
2.16.4 การอ่านข้อมูลจากสวิตช์จำนวนมาก	41

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ (ต่อ)

เรื่อง	หน้า
บทที่ 3 การออกแบบ การสร้าง และการทำงาน	43
3.1 วงจรกำเนิดแรงดัน	43
3.2 การออกแบบวงจรควบคุม	44
3.3 การออกแบบวงจรอินเตอร์เฟส	45
3.3.1 การออกแบบวงจรสวิตซ์เดี่ยว	45
3.3.2 การออกแบบวงจรไดโอดเปล่งแสง	46
3.3.3 การออกแบบวงจรแสดงผลแบบ 7 ส่วน	47
3.3.4 การออกแบบวงจรแสดงผลแบบผลึกเหลว	47
3.3.5 การออกแบบวงจรควบคุมสเต็ปปิ้งมอเตอร์	48
3.3.6 การออกแบบวงจรมอเตอร์ไฟฟ้ากระแสตรง	49
3.3.7 การออกแบบวงจรแปลงสัญญาณแอนะลอกเป็นสัญญาณดิจิตอล	50
3.3.8 การออกแบบวงจรแปลงสัญญาณดิจิตอลเป็นสัญญาณแอนะลอก	50
3.3.9 การออกแบบวงจรเชื่อมโยงทางแสงออก	51
3.3.10 การออกแบบวงจรขับรีเลย์	52
3.3.11 การออกแบบด้วยวงจรขยายพอร์ตแบบระบบ I^2C	53
3.3.12 การออกแบบวงจรแสดงผลแอลอีดีแบบเมตริกซ์	53
บทที่ 4 การทดลอง และผลการทดลอง	55
4.1 การทดลองวงจรแหล่งจ่ายไฟ	55
4.2 การทดลองวงจรควบคุม	56
4.3 การทดลองวงจรอินเตอร์เฟส	57
4.3.1 การทดลองวงจรสวิตซ์เดี่ยว	57
4.3.2 การทดลองวงจรเชื่อมต่อไดโอดเปล่งแสงแบบคาโถดร่วม	60
4.3.3 การทดลองวงจรเชื่อมต่อไดโอดเปล่งแสงแบบอาโนดร่วม	61
4.3.4 การทดลองวงจรเชื่อมโยงแสงออก	63
4.3.5 การทดลองวงจรแสดงผลแบบผลึกเหลวขนาด 16 ตัวอักษร	64
4.3.6 การทดลองวงจรแสดงผลแบบ 7 ส่วน	67
4.3.7 การทดลองวงจรขับรีเลย์ 8 ช่อง	69

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ (ต่อ)

เรื่อง	หน้า
4.3.8 การทดลองวงจรขับสเต็ปปิ้งมอเตอร์	71
4.3.9 การทดลองวงจรมอเตอร์ไฟฟ้ากระแสตรง	72
4.3.10 การทดลองวงจรแสดงผลแอลอีดีแบบเมตริกซ์ 8x8	74
4.3.11 การทดลองวงจรสวิตซ์เมตริกซ์	76
บทที่ 5 บทสรุป ปัญหา แนวทางการแก้ไขและพัฒนา	79
5.1 บทสรุป	79
5.2 ปัญหา และแนวทางแก้ไข	79
5.3 แนวทางการพัฒนาโครงงาน	80
ภาคผนวก ก เครื่องต้นแบบ	81
ภาคผนวก ข รายการอุปกรณ์	91
ภาคผนวก ค ใบงานการทดลอง	104
ภาคผนวก ง รายละเอียดข้อมูล และคุณสมบัติของอุปกรณ์	178
บรรณานุกรม	207
ประวัติผู้แต่ง	208

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญตาราง

ตาราง	หน้า
ตารางที่ 2.1 หน้าที่พิเศษของแต่ละขาของพอร์ต P3	7
ตารางที่ 2.2 หน้าที่พิเศษของพอร์ต 1 ในไมโครคอนโทรลเลอร์ MCS-51 แบบเฟลช	11
ตารางที่ 2.3 รูปแบบของรหัสเพื่อการแสดงผลตัวเลขแบบ 7 ส่วน	20
ตารางที่ 2.4 การจ่ายกระแสให้กับขดลวดแต่ละเฟส	26
ตารางที่ 2.5 การป้อนกระแสไฟฟ้าให้แก่เฟสต่างๆ ของสเต็ปปิ้งมอเตอร์	27
ตารางที่ 2.6 การจ่ายกระแสให้กับขดลวดแบบโหมดครึ่งขั้น	27
ตารางที่ 2.7 เปรียบเทียบจำนวนของสายเมื่อต่อแบบธรรมดาและเมตริกซ์	41
ตารางที่ 4.1 ผลการทดลองวัดค่าแรงดันไฟตรงจากแหล่งจ่ายไฟ	55
ตารางที่ 4.2 ผลการทดลองวงจรควบคุม	57
ตารางที่ 4.3 ผลการทดลองวงจรสวิตซ์เดี่ยว	59
ตารางที่ 4.4 ผลการทดลองวงจรเชื่อมต่อไดโอดเปล่งแสงชนิดคาโถดร่วม	61
ตารางที่ 4.5 ผลการทดลองวงจรเชื่อมต่อไดโอดเปล่งแสงชนิดอาโนดร่วม	62
ตารางที่ 4.6 ผลการทดลองวงจรเชื่อมโยงทางแสงออก	64
ตารางที่ 4.7 ผลการทดลองวงจรขับโมดูลแบบผลึกเหลว	67
ตารางที่ 4.8 ผลการทดลองวงจรขับไดโอดเปล่งแสงแบบ 7 ส่วน	68
ตารางที่ 4.9 ผลการทดลองวงจรขับรีเลย์	70
ตารางที่ 4.10 ผลการทำงานของสเต็ปปิ้งมอเตอร์	72
ตารางที่ 4.11 ผลการทดลองวงจรมอเตอร์ไฟฟ้ากระแสตรง	73

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูป

รูป	หน้า
รูปที่ 2.1 โครงสร้างพื้นฐานของไมโครโปรเซสเซอร์	3
รูปที่ 2.2 โครงสร้างพื้นฐานของไมโครคอนโทรลเลอร์	4
รูปที่ 2.3 การจัดหาตำแหน่งขาต่างๆ ของไมโครคอนโทรลเลอร์ตระกูล MCS-51	7
รูปที่ 2.4 หน้าที่ของพอร์ตเมื่อคอนโทรลเลอร์ทำงานกับหน่วยความจำภายนอก	8
รูปที่ 2.5 โครงสร้างพื้นฐานของไมโครคอนโทรลเลอร์ AT89S8252	9
รูปที่ 2.6 รูปแบบของการเชื่อมต่อของ SPI อย่างง่าย	12
รูปที่ 2.7 โครงสร้างและสัญลักษณ์ของไดโอดเปล่งแสง	16
รูปที่ 2.8 การตรวจสอบไดโอดเปล่งแสง	17
รูปที่ 2.9 วงจรแสดงผลแบบ 7 ส่วน	18
รูปที่ 2.10 การจัดวางไดโอดเปล่งแสงเพื่อแสดงผลแบบ 7 ส่วน	18
รูปที่ 2.11 การแสดงผลแบบ 7 ส่วน แบบคาโถดร่วม	19
รูปที่ 2.12 ไดโอดเปล่งแสงแบบ 7 ส่วนและการแสดงค่าตัวเลขฐานสิบหก	19
รูปที่ 2.13 แมติคลิควิดคริสตอล ขณะยังไม่มีการไบอัส	21
รูปที่ 2.14 แมติคลิควิดคริสตอล ขณะที่มีการไบอัส	22
รูปที่ 2.15 การไบอัสจอแสดงผลแบบผลึกเหลว	22
รูปที่ 2.16 โครงสร้างของจอแสดงผลแบบผลึกเหลว	23
รูปที่ 2.17 สเต็ปปิ้งมอเตอร์ 4 เฟสแบบแม่เหล็กถาวรยูนิโพลาร์	25
รูปที่ 2.18 ผังเวลาการทำงานของการจ่ายกระแสให้กับขดลวดแต่ละเฟส	26
รูปที่ 2.19 ผังเวลาการทำงานของการจ่ายกระแสให้กับขดลวดแบบโหมดครึ่งคลื่น	28
รูปที่ 2.20 ลักษณะของการแปลงสัญญาณดิจิตอลเป็นแอนะลอก	28
รูปที่ 2.21 รูปคลื่นของสัญญาณแอนะลอก	29
รูปที่ 2.22 ผังการทำงานของการเชื่อมโยงทางแสงออก	31
รูปที่ 2.23 การจ่ายไบอัสให้กับอุปกรณ์เชื่อมโยงทางแสง	32
รูปที่ 2.24 การใช้งานของตัวเชื่อมโยงทางแสงซึ่งอินพุตเป็นไฟตรง	34
รูปที่ 2.25 เปลี่ยนแรงดันอินพุตขนาด 24 โวลต์เป็นเอาต์พุตขนาด 5 โวลต์	34
รูปที่ 2.26 การเปลี่ยนสัญญาณไฟสลับขนาด 12โวลต์เป็นระดับลอจิก 5 โวลต์	35
รูปที่ 2.27 การต่อแอลอีดีแบบเมตริกซ์	36

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูป (ต่อ)

รูป	หน้า
รูปที่ 2.28 วงจรแป้นพิมพ์แบบขาร่วมและแบบเมตริกซ์	38
รูปที่ 2.29 ผังการทำงานของวงจรเข้ารหัสแป้นพิมพ์	39
รูปที่ 2.30 ลักษณะแป้นพิมพ์ไอโพอีดิคอล	40
รูปที่ 3.1 วงจรแหล่งกำเนิดแรงดัน	43
รูปที่ 3.2 การออกแบบวงจรควบคุม	45
รูปที่ 3.3 การออกแบบวงจรสวิตซ์เดี่ยว	45
รูปที่ 3.4 การออกแบบวงจรไดโอดเปล่งแสงแบบคาโถดร่วม	46
รูปที่ 3.5 การออกแบบวงจรไดโอดเปล่งแสงแบบอาโนดร่วม	46
รูปที่ 3.6 การออกแบบวงจรแสดงผลแบบ 7 ส่วน	47
รูปที่ 3.7 การออกแบบวงจรแสดงผลแบบผลึกเหลว	48
รูปที่ 3.8 การออกแบบวงจรควบคุมสเต็ปปิ้งมอเตอร์	49
รูปที่ 3.9 วงจรมอเตอร์ไฟฟ้ากระแสตรง	49
รูปที่ 3.10 การออกแบบวงจรแปลงสัญญาณแอนะลอกเป็นสัญญาณดิจิตอล	50
รูปที่ 3.11 การออกแบบวงจรแปลงสัญญาณดิจิตอลเป็นสัญญาณแอนะลอก	51
รูปที่ 3.12 การออกแบบวงจรการเชื่อมโยงทางแสงออก	51
รูปที่ 3.13 การออกแบบวงจรขับรีเลย์	52
รูปที่ 3.14 การออกแบบวงจรขยายพอร์ตด้วยระบบ I^2C	53
รูปที่ 3.15 วงจรแสดงแอลอีดีเมตริกซ์	54
รูปที่ 4.1 วงจรจ่ายไฟ	56
รูปที่ 4.2 วงจรควบคุม	56
รูปที่ 4.3 วงจรสวิตซ์เดี่ยว 8 ตัว	58
รูปที่ 4.4 โปรแกรมการทดลองไดโอดเปล่งแสง	59
รูปที่ 4.5 วงจรเชื่อมต่อไดโอดเปล่งแสงชนิดคาโถดร่วม	60
รูปที่ 4.6 โปรแกรมไฟวิ่ง	61
รูปที่ 4.7 วงจรเชื่อมต่อไดโอดเปล่งแสงชนิดอาโนดร่วม	61

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูป (ต่อ)

รูป	หน้า
รูปที่ 4.8 โปรแกรมไฟวิ่ง	62
รูปที่ 4.9 วงจรเชื่อมโยงแสงออก	63
รูปที่ 4.10 โปรแกรมทดลองวงจรเชื่อมโยงทางแสงออก	63
รูปที่ 4.11 วงจรแสดงผลแบบผลึกเหลวขนาด 16 ตัวอักษร 1บรรทัด	64
รูปที่ 4.12 โปรแกรมขับโมดูลแบบผลึกเหลว	66
รูปที่ 4.13 วงจรแสดงผลแบบ 7 ส่วน	67
รูปที่ 4.14 โปรแกรมขับแอลอีดีแบบ 7 ส่วน	68
รูปที่ 4.15 วงจรขับรีเลย์ 8 ช่อง	69
รูปที่ 4.16 โปรแกรมขับรีเลย์	70
รูปที่ 4.17 วงจรขับสเต็ปปิ้งมอเตอร์	71
รูปที่ 4.18 โปรแกรมขับสเต็ปปิ้งมอเตอร์	72
รูปที่ 4.19 วงจรมอเตอร์ไฟฟ้ากระแสตรง	72
รูปที่ 4.20 โปรแกรมขับมอเตอร์ไฟฟ้ากระแสตรง	73
รูปที่ 4.21 วงจรแสดงผลแอลอีดีเมตริกซ์	74
รูปที่ 4.22 โปรแกรมแสดงผลแอลอีดีเมตริกซ์	76
รูปที่ 4.23 ผลการทดลองโปรแกรมแอลอีดีเมตริกซ์	76
รูปที่ 4.24 การทดลองวงจรสวิตซ์เมตริกซ์	77
รูปที่ 4.25 โปรแกรมการตรวจสอบค่าคีย์สวิตซ์	78

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 1
บทนำ

1.1 ความเป็นมาและความสำคัญของปริญญานิพนธ์

ไมโครคอนโทรลเลอร์เป็นส่วนหนึ่งของระบบอิเล็กทรอนิกส์ ที่มีการใช้งานในกระบวนการทำงานหลายๆ ด้าน เราจะสังเกตเห็นว่ามีการนำไมโครคอนโทรลเลอร์ MCS-51 มาใช้งานทางด้านการควบคุมหรืองานอื่นๆ ตามแต่ผู้ใช้ต้องการอยู่หลายงานด้วยกัน โดยเฉพาะมีการใช้งานการควบคุมหรือในงานอุตสาหกรรมอย่างกว้างขวาง โดยเฉพาะในปัจจุบันที่ต้องการงานที่มีขนาดเล็กแต่มีประสิทธิภาพการทำงานสูง ดังนั้นในสถาบันการศึกษาจึงได้มีการจัดการเรียนการสอนเกี่ยวกับวิชาไมโครคอนโทรลเลอร์ขึ้นแต่กระบวนการเรียนการสอนนอกจากการศึกษาจากหนังสือหรือตำราแล้ว สิ่งหนึ่งที่จะทำให้นักศึกษาเข้าใจมากขึ้นคือการทดลองปฏิบัติงานจริง และจำเป็นต้องมีชุดทดลองไมโครคอนโทรลเลอร์ประกอบการจัดการเรียนการสอน เพื่อเพิ่มทักษะให้กับผู้เรียนซึ่งจะทำให้ผู้เรียนเข้าใจยิ่งขึ้นจึงมีความคิดที่จะพัฒนาชุดทดลองการเชื่อมต่อไมโครคอนโทรลเลอร์กับอุปกรณ์ภายนอกเพื่อใช้ประกอบการเรียนการสอนให้มีประสิทธิภาพดียิ่งขึ้น

1.2 ขีดความสามารถของโครงงาน

1) มีใบงานประกอบการทดลอง 13 ใบงาน

2) สามารถทดลองเชื่อมต่อกับพอร์ตอินพุตแบบสวิตช์เดี่ยว และสวิตช์เมตริกซ์ ได้

3) สามารถทดลองการเชื่อมต่อกับภาคแสดงผลแบบแอลอีดี, แอลอีดีเมตริกซ์, แอลอีดี 7 ส่วน และภาคแสดงผลแบบผลึกเหลว

4) สามารถทดลองการเชื่อมต่อกับพอร์ตเอาต์พุตมอเตอร์ไฟฟ้ากระแสตรง และสเต็ปปิ้งมอเตอร์ได้

5) สามารถทดลองการเชื่อมต่อกับพอร์ตเอาต์พุตระบบบัสแบบ I^2C, วงจรแปลงสัญญาณแอนะลอกเป็นสัญญาณดิจิตอล และวงจรแปลงสัญญาณดิจิตอลเป็นสัญญาณแอนะลอกได้

6) สามารถทดลองการเชื่อมต่อกับพอร์ตเอาต์พุตแบบวงจรควบคุมรีเลย์ และอุปกรณ์เชื่อมโยงทางแสงได้

7) สามารถประยุกต์ใช้ชุดทดลองกับไมโครคอนโทรลเลอร์ตระกูลอื่นได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1.3 เนื้อหาโดยสังเขป

เนื้อหาภายในปริญญานิพนธ์ฉบับนี้แบ่งออกเป็นบทต่างๆ เพื่อความสะดวกต่อการศึกษาและทำความเข้าใจในแต่ละบทจะประกอบด้วยเนื้อหาที่สำคัญดังนี้

บทที่ 2 ทฤษฎี และหลักการ ประกอบด้วยเนื้อหาในทางทฤษฎีที่เกียวกับวงจรได้แก่ทฤษฎีเกี่ยวกับไมโครคอนโทรลเลอร์ตระกูล MCS-51 โครงสร้างภายนอกของไมโครคอนโทรลเลอร์ตระกูล MCS-51 ไดโอดเปล่งแสง จอแสดงผลแบบ 7 ส่วนจอแสดงผลแบบผลึกเหลว จอแสดงผลแบบดอตเมตริกซ์ อุปกรณ์เชื่อมโยงทางแสง การแปลงสัญญาณแอนะลอกเป็นสัญญาณดิจิตอล การแปลงสัญญาณดิจิตอลเป็นสัญญาณแอนะลอก มอเตอร์ไฟฟ้ากระแสตรง สเต็ปปิ้งมอเตอร์ สวิตช์เดี่ยว สวิตช์เมตริกซ์ การเชื่อมต่อแบบ I^2C และรีเลย์

บทที่ 3 การออกแบบ การสร้าง และการทำงาน กล่าวถึง การออกแบบฮาร์ดแวร์ซึ่งได้แก่ภาคแสดงผลแบบแอลอีดีแบบคาโถดร่วมและอาโนดร่วม ภาคแสดงผลแบบแอลอีดี 7 ส่วน ภาคแสดงผลแบบผลึกเหลว วงจรแปลงสัญญาณแอนะลอกเป็นสัญญาณดิจิตอล และการแปลงสัญญาณดิจิตอลเป็นสัญญาณแอนะลอก วงจรควบคุมมอเตอร์ไฟฟ้ากระแสตรง วงจรควบคุมสเต็ปปิ้งมอเตอร์ วงจรสวิตช์เดี่ยว วงจรสวิตช์เมตริกซ์ วงจรเชื่อมต่อแบบ I^2C และ วงจรควบคุมรีเลย์

บทที่ 4 การทดลอง และผลการทดลอง กล่าวถึงการทดลองการทำงาน และผลการทดลองของวงจรภาคต่างๆ ได้แก่ วงจรกำเนิดแรงดัน วงจรควบคุม และวงจรอินเตอร์เฟส

บทที่ 5 บทสรุป ปัญหา แนวทางแก้ไข และพัฒนา กล่าวถึงผลการปฏิบัติงาน ปัญหาที่เกิดขึ้นขณะปฏิบัติงาน การแก้ไขปัญหาที่เกิดขึ้น แนวทางการแก้ไขปัญหา และแนวทางการพัฒนาโครงงาน

ภาคผนวก ก รูปต้นแบบ

ภาคผนวก ข รายการอุปกรณ์

ภาคผนวก ค ใบงานการทดลอง

ภาคผนวก ง รายละเอียดข้อมูล และคุณสมบัติของอุปกรณ์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 2
ทฤษฎี และหลักการ

2.1 กล่าวนำ

เนื้อหาของปริญญานิพนธ์ในบทนี้เป็นทฤษฎี และหลักการที่จะนำมาใช้ประกอบการสร้างโครงงาน โดยประกอบด้วยทฤษฎีที่เกี่ยวข้องกับวงจรแต่ละวงจรที่ใช้ในการสร้างโครงงาน ซึ่งจะได้กล่าวดังต่อไปนี้

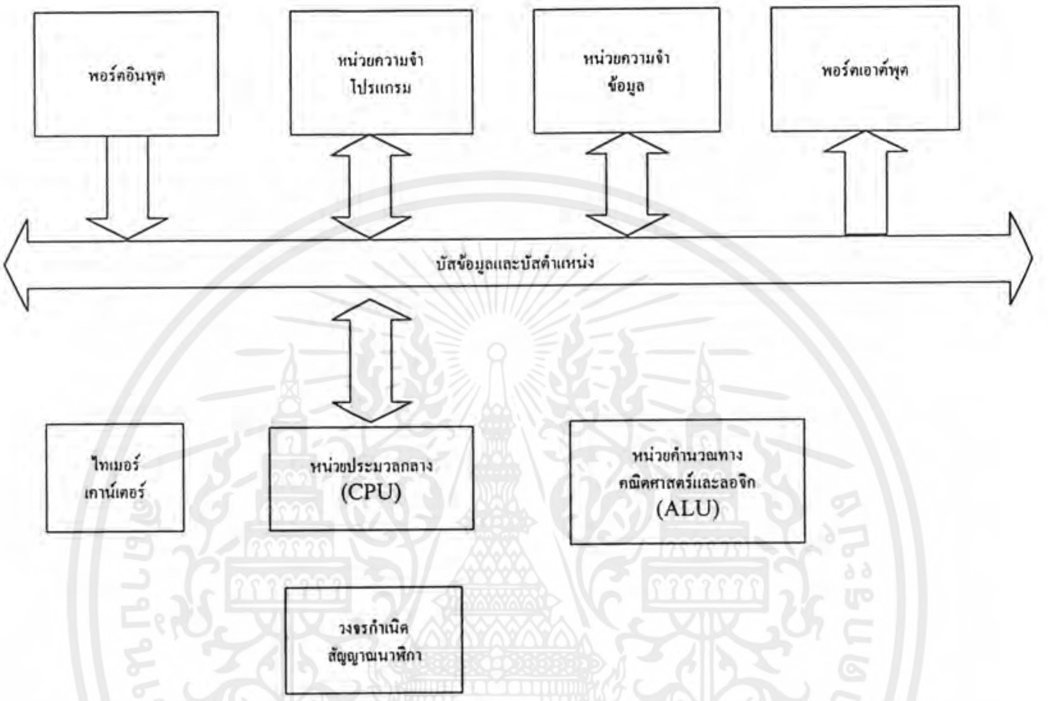
2.2 ความแตกต่างของไมโครคอนโทรลเลอร์กับไมโครโปรเซสเซอร์

รูปที่ 2.1 โครงสร้างพื้นฐานของไมโครโปรเซสเซอร์

โครงสร้างของไมโครโปรเซสเซอร์แสดงดังรูป 2.1 ซึ่งประกอบด้วยหน่วยประมวลผลกลาง, หน่วยคำนวณทางคณิตศาสตร์และลอจิก, บัสข้อมูล และบัสตำแหน่งสำหรับติดต่อกับหน่วยความจำภายนอก และวงจรกำเนิดสัญญาณนาฬิกานั่นหมายความว่า การใช้งานไมโครโปรเซสเซอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จะต้องเชื่อมต่อกับหน่วยความจำโปรแกรมภายนอก ถ้าหากมีความต้องการเชื่อมต่อกับอุปกรณ์อินพุต และเอาต์พุตต้องอาศัยอุปกรณ์ที่เรียกว่า ไอซีขยายพอร์ตทำให้การสร้างระบบควบคุมด้วยไมโครโปรเซสเซอร์จึงมีความต้องการอุปกรณ์จำนวนมาก ส่งผลให้ขนาดของระบบใหญ่



รูปที่ 2.2 โครงสร้างพื้นฐานของไมโครคอนโทรลเลอร์

จากรูปที่ 2.2 แสดงโครงสร้างของไมโครคอนโทรลเลอร์ จะเห็นได้อย่างชัดเจนว่า ภายในไมโครคอนโทรลเลอร์มีอุปกรณ์พื้นฐานเหมือนกับไมโครโปรเซสเซอร์หากแต่จะบรรจุหน่วยความจำโปรแกรม หน่วยความจำข้อมูล และพอร์ตอินพุตเอาต์พุตไว้ภายในพร้อมสรรพ ผู้ใช้งานจึงเพียงแต่เขียนโปรแกรมควบคุมลงบนตัวไมโครคอนโทรลเลอร์แล้วต่อวงจรที่ใช้ในการสร้างวงจรกำเนิดต้องการเข้ากับพอร์ของไมโครคอนโทรลเลอร์เพียงเท่านี้ก็สามารถใช้งานไมโครคอนโทรลเลอร์ได้แล้วส่งผลให้ขนาดและราคาของระบบลดลงอย่างมากและไมโครคอนโทรลเลอร์ก็สามารถเชื่อมต่อกับหน่วยความจำภายนอกได้เช่นกัน โดยพิจารณาให้หน่วยความจำภายนอกนั้นเป็นอุปกรณ์อินพุตเอาต์พุตตัวหนึ่งแล้วใช้ขาพอร์ตที่มีอยู่ทำการติดต่อ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.2.1 การเลือกใช้ไมโครโปรเซสเซอร์ และไมโครคอนโทรลเลอร์

ถ้าเป็นการสร้างระบบควบคุมขนาด 8 บิตมีความต้องการเชื่อมต่อกับอุปกรณ์ภายนอกไม่มากนัก (น้อยกว่า 10 แบบ) ควรเลือกใช้ไมโครคอนโทรลเลอร์และถ้าหากต้องการประมวลผลข้อมูลจำนวนมาก ต้องใช้หน่วยความจำโปรแกรมสูงถึง 8 กิโลไบต์ ทั้งนี้ยังมีความต้องการเก็บรักษาข้อมูลลงในหน่วยความจำข้อมูลมากเป็นกิโลไบต์ ควรออกแบบให้ไมโครคอนโทรลเลอร์ในระบบควบคุมนี้เชื่อมต่อกับหน่วยความจำภายนอก

ถ้าต้องการใช้งานกับข้อมูลมากกว่า 8 บิตตลอดเวลา และต้องการความเร็วในการทำงานสูงๆ สามารถติดต่อหน่วยความจำได้เป็นจำนวนมากๆ ติดต่อกับอุปกรณ์อินพุตเอาต์พุตได้จำนวนมากมายในเวลาเดียวกัน ควรเลือกใช้ไมโครโปรเซสเซอร์ ดังจะเห็นได้จากในเครื่องคอมพิวเตอร์ที่ใช้ไมโครโปรเซสเซอร์เป็นหัวใจหลักในการทำงาน ทำให้สามารถเชื่อมต่อกับหน่วยความจำได้มากเป็นหน่วยกิกะไบต์ขนาดของข้อมูลสูงสุด 64 บิต ความเร็วสูงเป็นหลายร้อยเมกะเฮิรตซ์เป็นต้น

สรุปได้ว่า ไมโครคอนโทรลเลอร์เหมาะสำหรับการสร้างระบบควบคุม ในขณะที่ไมโครโปรเซสเซอร์เหมาะสำหรับการสร้างระบบประมวลผลข้อมูลความเร็วสูงและระบบควบคุมที่มีขนาดใหญ่มากๆ

2.3 คุณสมบัติทั่วไปของไมโครคอนโทรลเลอร์ตระกูล MCS-51

คุณสมบัติทั่วไปของไมโครคอนโทรลเลอร์ MCS-51 มีดังนี้

1) เป็นไมโครคอนโทรลเลอร์ขนาด 8 บิต

2) มีวงจรออสซิลเลเตอร์ และวงจรผลิตสัญญาณนาฬิกาภายในไอซี

3) มีขาสัญญาณอินพุตเอาต์พุตขนาด 32 บิต

4) สามารถเชื่อมต่อหน่วยความจำข้อมูลภายนอก (External Data) โดยอ้างตำแหน่งหน่วยความจำได้ถึง 64 กิโลไบต์

5) สามารถเชื่อมต่อกับหน่วยความจำข้อมูลภายนอก (External Program Memory) โดยสามารถอ้างตำแหน่งหน่วยความจำได้ถึง 64 กิโลไบต์

6) มีหน่วยความจำโปรแกรมภายในตัว (On-Chip Program Memory) ขนาด 4 กิโลไบต์ โดยเฉพาะเบอร์ 8052 จะมีหน่วยความจำในส่วนนี้ถึง 8 กิโลไบต์ สำหรับเบอร์ 8031 และ M8032 จะไม่มีหน่วยความจำในส่วนนี้

7) มีหน่วยความจำข้อมูลภายในตัว (On-Chip Data Memory) ขนาด 12 ไบต์ โดยเฉพาะเบอร์ 8032 และ 8052 จะมีในส่วนนี้ถึง 256ไบต์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

8) มีหน่วยความจำบางส่วนสามารถเข้าถึงข้อมูลระดับบิตได้ด้วย ทำให้การควบคุมหรือการตรวจสอบสถานะบิตทำได้ง่าย ส่งผลให้การเขียนโปรแกรมทำได้ง่ายขึ้น

9) มีไทม์เมอร์/เคาน์เตอร์ (Timers/Counters) ขนาด 16 บิต จำนวน 2 ตัว โดยเฉพาะเบอร์8032 และ 8052 จะมีไทม์เมอร์/เคาน์เตอร์จำนวน 3 ตัว

10) การอินเตอร์รัพต์ทำได้ทำจาก 5 แหล่งกำเนิด โดยเฉพาะเบอร์ 8032 และ 8052 จะทำการอินเตอร์รัพต์ได้จาก 6 แหล่งกำเนิด โดยการอินเตอร์รัพต์ยังสามารถจัดระดับความสำคัญได้เป็น 2 ระดับ

11) มีพอร์ตสื่อสารอนุกรมภายในตัวเอง ซึ่งทำงานเป็นแบบฟูลดูเพล็กซ์ (Full Duplex)

12) มีคำสั่งในการคำนวณทางคณิตศาสตร์ และทางตรรกศาสตร์คำสั่งโดยส่วนใหญ่ใช้เวลาการทำงานเพียง 1 ไมโครวินาที เมื่อใช้คริสตอลความถี่ 12 เมกะเฮิรตซ์

13) ต้องการแหล่งจ่ายไฟ 5 โวลต์เพียงชุดเดียว

2.4 โครงสร้างภายนอกของไมโครคอนโทรลเลอร์ MCS-51

ไมโครคอนโทรลเลอร์ตระกูล MCS-51 ทุกเบอร์จะมีตำแหน่งขาที่เหมือนกัน ดังแสดงในรูปที่ 2.3 สำหรับหน้าที่การใช้งานของแต่ละขามีดังนี้

1) ขา Vcc เป็นขาป้อนแรงดันไฟเลี้ยง +5 โวลต์

2) ขา Vss เป็นกราวด์

3) ขาพอร์ต 0 (PORT 0) มี 8 ขาได้แก่ P0.0 - P0.7 เป็นขาพอร์ตอินพุตเอาต์พุตแบบ 2 ทิศทางสำหรับการใช้งานทั่วไป โดยถ้าใช้งานเป็นอินพุตต้องทำการเขียนค่าเป็น 1 ไปยังแต่ละบิตของพอร์ต เพื่อกำหนดให้ขาพอร์ตเหล่านั้นอยู่ในสถานะปล่อยลอย ซึ่งในสถานะนี้เองที่สามารถนำมาใช้เป็นพอร์ตอินพุตอิมพีแดนซ์สูงได้ นอกจากพอร์ตนี้จะใช้งานเป็นพอร์ตอินพุตเอาต์พุตแล้วมันยังถูกใช้งานในการติดต่อกับหน่วนความจำด้วย โดยทำหน้าที่ในการกำหนดตำแหน่งหน่วยความจำไบต์ต่ำ (A0 – A7) ซึ่งจะใช้เป็นแบบมัลติเพล็กซ์กับการรับส่งข้อมูลขนาด 8 บิต (D0 – D7)

4) ขาพอร์ต 1 มี 8 ขา ได้แก่ขา P1.0 – P1.7 เป็นขาพอร์ตอินพุตเอาต์พุตแบบ 2 ทิศทางสำหรับการใช้งานทั่วไป โดยถ้าใช้งานเป็นอินพุตต้องทำการเขียนค่าเป็น 1 ไปยังแต่ละบิตของพอร์ต เพื่อกำหนดให้เป็นพอร์ตอินพุต

5) ขาพอร์ต 2 (PORT 2) มี 8 ขา ได้แก่ขา P2.0 - P2.7 เป็นขาพอร์ตอินพุตเอาต์พุตแบบ 2 ทิศทางสำหรับการใช้งานทั่วไป โดยถ้าใช้งานเป็นอินพุตต้องทำการเขียนค่าเป็น 1 ไปยังแต่ละบิตของพอร์ต เพื่อกำหนดให้เป็นพอร์ตอินพุตนอกจากพอร์ตนี้จะใช้งานเป็นพอร์ตอินพุตเอาต์พุตแล้ว

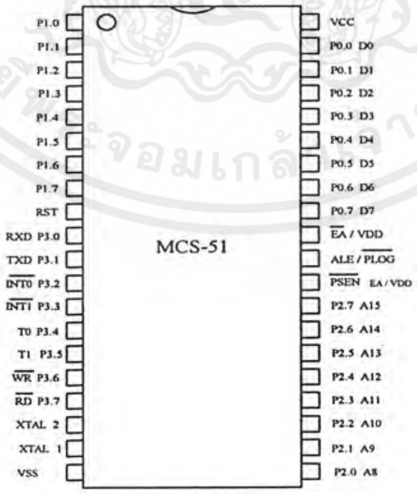
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ยังถูกใช้งานในการติดต่อกับหน่วนความจำภายนอกด้วยโดยทำหน้าที่ในการกำหนดตำแหน่งหน่วย ใช้เป็นพอร์ตอินพุตอิมพีแดนซ์สูงได้ นอกจากความจำไบต์สูง (A8 – A15)

ตารางที่ 2.1 หน้าที่พิเศษของแต่ละขาของพอร์ต P3

ขาพอร์ต	หน้าที่พิเศษ
P3.0	RXD (Serial Input Port)
P3.1	TXD (Serial Output Port)
P3.2	$\overline{INT}0$ (External Interrupt 0)
P3.3	$\overline{INT}1$ (External Interrupt 1)
P3.4	T0 (Timer 0 External Input)
P3.5	T1 (Timer 1External Input)
P3.6	$\overline{WR}$ (External Data Memory Write Strobe)
P3.7	$\overline{RD}$ (External Data Memory Read Strobe)

6) ขาพอร์ต 3 (PORT 3) มี 8 ขา ได้แก่ P3.1 P3.7 เป็นขาพอร์ตอินพุตเอาต์พุตแบบ 2 ทิศทางสำหรับการใช้งานทั่วไป โดยถ้าใช้งานเป็นอินพุตต้องทำการเขียนค่าเป็น 1 ไปยังแต่ละบิต



รูปที่ 2.3 การจัดตำแหน่งขาต่างๆ ของไมโครคอนโทรลเลอร์ตระกูล MCS-51

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

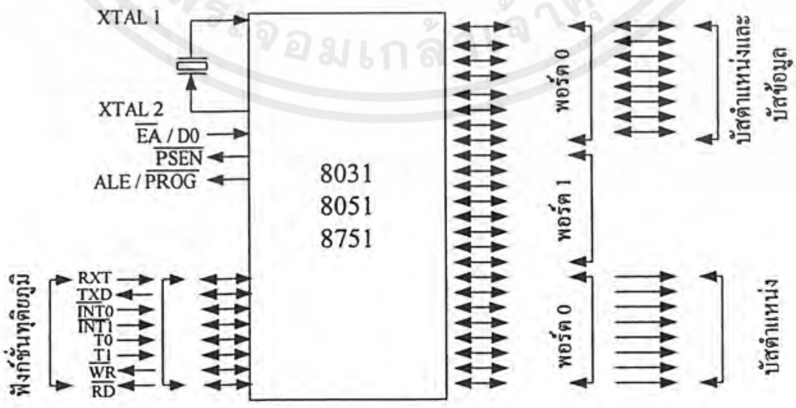
การกำหนดให้เป็นพอร์ตอินพุตนี้นอกจากจะใช้งานเป็นพอร์ตอินพุตเอาต์พุตแล้วยังถูกใช้งานในหน้าที่พิเศษต่างๆดังแสดงในตารางที่ 2.1

7) ขารีเซตใช้สำหรับการรีเซตการทำงานของไมโครคอนโทรลเลอร์ โดยการรีเซตต้องคงสถานะเป็น 1 อย่างน้อยนาน 2 รอบการทำงานของคำสั่ง ในขณะที่ออสซิลเลเตอร์ยังทำงานอยู่

8) ขา $ALE/\overline{PROG}$ เป็นขาสัญญาณที่ทำหน้าที่ควบคุมการค้างสภาวะ (Latch) ค่าตำแหน่งไบต์ต่ำ (Address Latch Enable) เมื่อติดต่อกับหน่วยความจำภายนอก นอกจากนี้ขานี้ยังทำหน้าที่เป็นอินพุตรับพัลส์ในการโปรแกรม ในส่วนของหน่วยความจำ EPROM สำหรับไมโครคอนโทรลเลอร์ตระกูล MCS-51 ที่มีหน่วยความจำตำแหน่งภายใน EPROM

9) ขา $\overline{PSEN}$ ทำหน้าที่ส่งสัญญาณเพื่อร้องขอติดต่อกับหน่วยความจำโปรแกรมภายนอก เมื่อไมโครคอนโทรลเลอร์ต้องการอ่านข้อมูลจากหน่วยความจำโปรแกรมภายนอกตัวไมโครคอนโทรลเลอร์ ต้องการอ่านข้อมูลจากหน่วยความจำโปรแกรมภายนอก ตัวไมโครคอนโทรลเลอร์จะส่งสัญญาณออกที่ขานี้ 2 ครั้งในแต่ละรอบการทำงานของคำสั่ง แต่ถ้าหากติดต่อกับหน่วยความจำข้อมูลภายนอก ขานี้จะไม่มีการส่งสัญญาณใดๆ ออกมาเลย

10) ขา $\overline{EA}/VPP$ เป็นขาสำหรับการเลือกใช้หน่วยความจำโปรแกรมจากภายในหรือจากภายนอก โดยถ้ามีสถานะเป็น 0 จะหมายถึงให้ไมโครคอนโทรลเลอร์รับคำสั่งจากหน่วยความจำภายนอกที่ตำแหน่งแอดเดรส 0- 0FFFH อย่างไรก็ตามถ้าบิตป้องกัน (Security Bit) ในหน่วยความจำ EPROM ถูกโปรแกรมไว้ ไมโครคอนโทรเลอร์จะไม่รับคำสั่งจากหน่วยความจำจากนี้ขานี้ยังทำหน้าที่รับแรงดันไฟสำหรับการโปรแกรม (Vpp) ขนาด 12 โวลต์ เพื่อใช้ใน ระหว่างการโปรแกรม EPROM



รูปที่ 2.4 หน้าที่ของพอร์ตเมื่อไมโครคอนโทรลเลอร์ทำงานกับหน่วยความจำภายนอก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

11) ขา XTAL1, XTAL2 เป็นขาอินพุตและเอาต์พุตของวงจรอินเวอร์ติ้งออสซิลเลเตอร์แอมปลิไฟเออร์ (Inverting Oscillator Amplifier) สำหรับใช้ต่อร่วมกับคริสตอลภายนอก

2.5 คุณสมบัติเทคนิคของไมโครคอนโทรลเลอร์ AT89S8252

1) เป็นไมโครคอนโทรลเลอร์ที่ใช้ซีพียูขนาด 8 บิต

2) ภายในมีหน่วยความจำโปรแกรมเป็นแบบเฟลชสามารถลบ และเขียนใหม่ได้ 1 พันครั้ง

3) หน่วยความจำข้อมูลพื้นฐานเป็นความจำแบบแรม 256 ไบต์ แบบอีพรอม 2 กิโลไบต์

4) ขาพอร์ตเป็นแบบ 2 ทิศทาง สามารถใช้งานได้เป็นทั้งอินพุต และเอาต์พุต

5) มีวงจรสื่อสารอนุกรมแบบฟูลดูเพล็กซ์

6) ไทเมอร์/เคาน์เตอร์ขนาด 16 บิต 3 ตัว

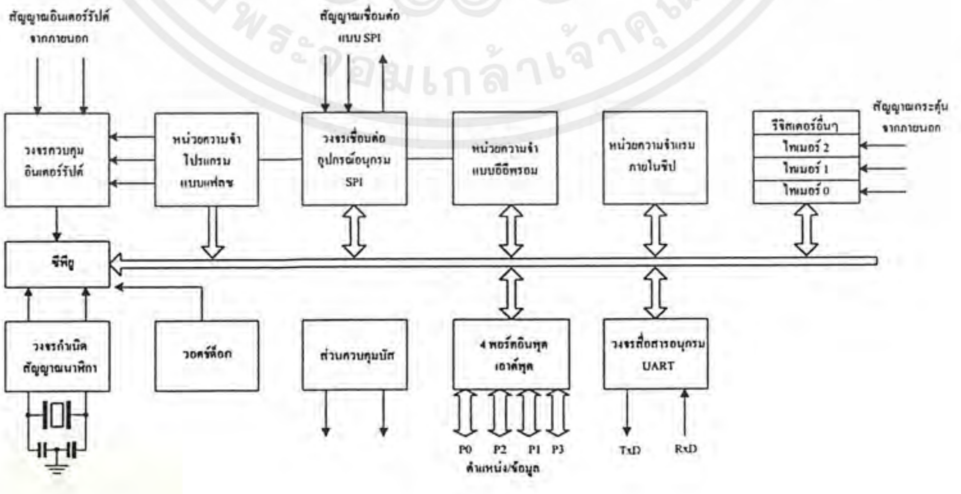
7) สามารถรองรับแหล่งกำเนิดอินเตอร์รัปต์ได้ 6 แหล่ง

8) มีวงจรกำเนิดสัญญาณนาฬิกาอยู่ภายในชิป

9) มีวงจรสื่อสารอนุกรมแบบ SPI

10) มีวอตช์ด็อกไทเมอร์ในตัว

โครงสร้างพื้นฐานของ AT89S8252 จะเห็นได้ว่ามีส่วนประกอบที่เพิ่มเติมแตกต่างจาก AT89C51 อยู่หลายส่วน เช่น วงจรเชื่อมต่ออนุกรมแบบ SPI ซึ่งภายในไมโครคอนโทรลเลอร์อนุกรมนี้ใช้ในการเขียนข้อมูลลงในหน่วยความจำโปรแกรมโดยไม่ต้องถอดตัวชิปออกไปจากระบบ หรือเรียกว่า การโปรแกรมในวงจร ไทเมอร์/เคาน์เตอร์ขนาด 16 บิต ที่เพิ่มเติมเข้ามาอีกหนึ่งตัวเป็นไทเมอร์ 2 และวงจรวอตช์ด็อกที่ใช้ในการตรวจสอบการทำงานผิดพลาดของซีพียู



รูปที่ 2.5 โครงสร้างพื้นฐานของไมโครคอนโทรลเลอร์ AT89S8252

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.6 โครงสร้าง และการทำงานของพอร์ต

ไมโครคอนโทรลเลอร์ MCS-51 แบบเฟลซมีพอร์ตให้ใช้งานทั้งสิ้น 4 พอร์ต คือ พอร์ต0 ถึงพอร์ต 3 มีขนาด 8 บิต เป็นพอร์ตแบบ 2 ทิศทาง สามารถเป็นได้ทั้งพอร์ตอินพุตสำหรับรับสัญญาณข้อมูลเข้า และเอาต์พุตสำหรับส่งสัญญาณข้อมูลออก ทุกพอร์ตของไมโครคอนโทรลเลอร์ MCS-51 แบบเฟลซมีวงแลตช์และวงจรขับตลอดจนบัฟเฟอร์อินพุตที่พอร์ต 0 และพอร์ต 2 จะใช้งานเป็นพอร์ตอินพุตและเอาต์พุตสำหรับงานทั่วไป และใช้ในการติดต่อกับหน่วยความจำภายนอก สำหรับพอร์ต 3 ทั้งพอร์ต และพอร์ต 1 บางขานอกจากจะใช้เป็นขาพอร์ตอินพุตเอาต์พุตตามปกติแล้ว ยังสามารถใช้งานในหน้าที่พิเศษได้อีก

2.6.1 การใช้งานพอร์ตอินพุต

เนื่องจากพอร์ตทั้งหมดของไมโครคอนโทรลเลอร์ MCS-51 แบบเฟลซสามารถเป็นได้ทั้งอินพุตและเอาต์พุตดังนั้นจึงมีความจำเป็นอย่างยิ่งต้องทำความเข้าใจถึงการกำหนดลักษณะการทำงานให้พอร์ตไมโครคอนโทรลเลอร์ MCS-51 แบบเฟลซ

ในการกำหนดให้เป็นพอร์ตอินพุต ต้องเริ่มต้นด้วยการเขียนข้อมูล “1” มาทีละบิตของพอร์ตที่ต้องการใช้งานเป็นพอร์ตอินพุต เพื่อหยุดการทำงานของเฟตที่ใช้ในการขับสัญญาณเอาต์พุตของพอร์ตนั้นๆ ทำให้ขาสัญญาณของพอร์ตเชื่อมต่อเข้ากับวงจรพูลอัปภายในโดยตรง ส่งผลให้ขาพอร์ตนั้นมีลอจิกเป็น “1” สามารถรับสัญญาณ “0” จากอุปกรณ์ได้ง่าย สัญญาณข้อมูลจากอุปกรณ์ภายนอกจะถูกส่งเข้ามาแล้วเก็บไว้ในวงจรบัฟเฟอร์ภายในพอร์ต แล้วรอให้ซีพียูมาอ่านค่าเข้าไป เมื่อเป็นเช่นนี้ อุปกรณ์ภายนอกที่เชื่อมต่อกับพอร์ตอินพุตของไมโครคอนโทรลเลอร์ MCS-51 แบบเฟลซควรกำหนดให้ทำงานในสภาวะลอจิก “0” จะดีหรือสะดวกที่สุด

2.6.2 การใช้งานเป็นพอร์ตเอาต์พุต

โดยปกติแล้วขาพอร์ตจะกำหนดให้มีลักษณะเป็นเอาต์พุตอยู่แล้ว ดังนั้นจึงสามารถส่งข้อมูลออกไปได้อย่างง่ายดายและตรงไปตรงมา คือเมื่อต้องการส่งข้อมูล “0” ออกไปทางเอาต์พุตก็ให้เขียนข้อมูล “0” ไปยังวงจรแลตช์ ซึ่งก็จะส่งต่อไปขับเฟต ทำให้เฟตทำงานที่ขาพอร์ตที่กำหนดให้ทำงานก็จะเกิดลอจิก “0” ขึ้น ในทางตรงกันข้ามหากต้องการส่งข้อมูล “1” ออกไป ก็ให้เขียนข้อมูล “1” ไปยังวงจรแลตช์ วงจรขับก็จะหยุดทำงาน ทำให้ที่ขาพอร์ตเชื่อมต่อกับขาวงจรพูลอัปภายในเกิดเป็นลอจิก “1” ที่ขาพอร์ตนั้น ซึ่งจะคล้ายกับการกำหนดให้เป็นขาอินพุตมาก เพียงแต่แตกต่างกันที่กระบวนการในการเคลื่อนย้ายข้อมูล โดยถ้าเป็นอินพุตจะมีสัญญาณมาอ่านข้อมูลที่บัฟเฟอร์ แต่ถ้าเป็นเอาต์พุตจะไม่มีการอ่านข้อมูลที่บัฟเฟอร์แต่อย่างใด เว้นแต่ในกรณีที่ต้องการตรวจสอบข้อมูลที่ส่งออกมาทางเอาต์พุต

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เมื่อใช้งานพอร์ตของไมโครคอนโทรลเลอร์ MCS-51 แบบเฟลซเป็นพอร์ตเอาต์พุต แต่ละขาของแต่ละพอร์ตมีความสามารถในการจ่ายกระแสซอร์ส ได้สูงสุด 10 มิลลิแอมป์ และทุกขารวมกันในพอร์ต 0 (ทั้ง 8 บิต) สูงสุด 26 มิลลิแอมป์ และ 15 มิลลิแอมป์ สำหรับพอร์ต 1-3 ในกรณีที่ใช้งานเป็นพอร์ตเอาต์พุตเพื่อไม่ให้เกิดปัญหาเกี่ยวกับความสามารถในการจ่ายกระแสจึงควรต่อวงจรบัฟเฟอร์ทางเอาต์พุตเพื่อช่วยในการขับกระแสอีกทางหนึ่ง

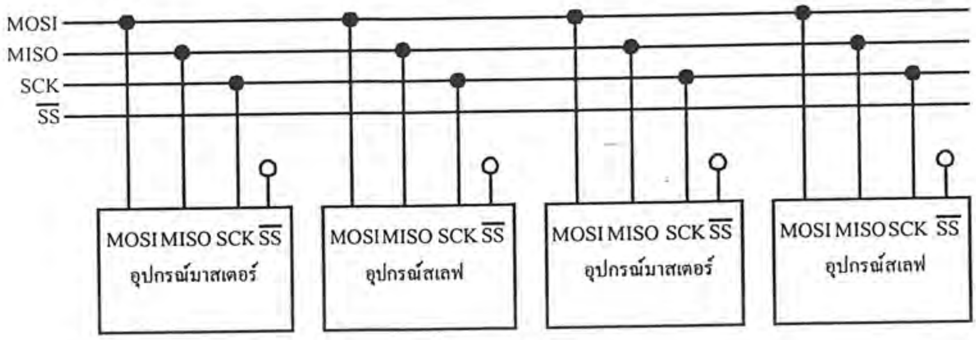
ตารางที่ 2.2 หน้าที่พิเศษของพอร์ต 1 ในไมโครคอนโทรลเลอร์ MCS-51 แบบเฟลซ

ขา	หน้าที่พิเศษ
P1.0	เป็นขาอินพุตนับค่าของไทเมอร์/เคาน์เตอร์ 2 และเป็นขาเอาต์พุตของการกำเนิดสัญญาณนาฬิกาด้วยไทเมอร์ 2 (Clock Out)
P1.1	ขา T2EX เป็นขาอินพุตทริกเกอร์สำหรับการแคปเจอร์/รีโหลดและควบคุมทิศทางของสัญญาณ
P1.4	ขา $\overline{SS}$ (Slave Select) เป็นขาเลือกการติดต่อในกรณีที่ไมโครคอนโทรลเลอร์เป็นอุปกรณ์สเลฟ ในระบบการติดต่อแบบ SPI
P1.5	ขา MOSI (Master data output, Slave data input) ใช้ในการติดต่อกับพอร์ต SPI
P1.6	ขา MISO (Master data input, Slave data output) ใช้ในการติดต่อกับพอร์ต SPI
P1.7	ขา SCK (Master clock output) เป็นขาสัญญาณนาฬิกาของการติดต่อกับพอร์ต

2.7 ส่วนเชื่อมต่ออุปกรณ์ SPI

ส่วนเชื่อมต่ออุปกรณ์ SPI นี้เป็นการเชื่อมต่อไมโครคอนโทรลเลอร์เข้ากับอุปกรณ์ภายนอกที่มีลักษณะของการถ่ายทอดข้อมูลในลักษณะอนุกรมซิงโครนัสความเร็วสูง ในการสื่อสารข้อมูลแบบซิงโครนัสจะต้องใช้สัญญาณนาฬิกาเพื่อกำหนดจังหวะในการถ่ายทอดข้อมูลร่วมกัน และต้องกำหนดสถานภาพของอุปกรณ์ในระบบอย่างชัดเจน เนื่องจากในการเชื่อมต่อแบบ SPI สามารถต่อพ่วงอุปกรณ์ได้เป็นจำนวนมาก โดยต้องมีการกำหนดตัวควบคุมหลักเป็นอุปกรณ์ตัวแม่และอุปกรณ์ที่นำมาต่อพ่วงในระบบเป็นอุปกรณ์ลูกหรือสเลฟดังแสดงในรูปที่ 2.6

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.6 รูปแบบของการเชื่อมต่อของ SPI อย่างง่าย

อุปกรณ์ตัวแม่หรือมาสเตอร์โดยส่วนใหญ่จะเป็นไมโครคอนโทรลเลอร์ และอุปกรณ์ตัวลูกหรือสเลฟมักเป็นไอซีที่ทำหน้าที่เฉพาะพิเศษ เช่น หน่วยความจำ ไอซีขยายพอร์ต หรือ UART เป็นต้น แต่ด้วยการใช้ความสามารถของ SPI อาจสร้างโครงข่ายระบบควบคุมแบบมัลติโปรเซสเซอร์ได้โดยการเชื่อมต่อไมโครคอนโทรลเลอร์หลายตัวเข้าด้วยกันโดยผ่านทางขาเชื่อมต่อของระบบ SPI หมายความว่า อุปกรณ์ตัวลูกก็คือไมโครคอนโทรลเลอร์หรือไมโครโปรเซสเซอร์อีกชุดหนึ่งหรือหลายๆ ชุดในกรณีต่อพ่วงกันมากกว่า 2 ตัว

2.7.1 คุณสมบัติของ SPI ในไมโครคอนโทรลเลอร์ AT89S8252

1) จัดการสื่อสารข้อมูลแบบฟูลดูเพล็ก ใช้สายสัญญาณในการถ่ายทอดข้อมูล 3 เส้นในลักษณะซิงโครนัส

2) สามารถทำงานเป็นได้ทั้งอุปกรณ์มาสเตอร์และสเลฟ

3) ความถี่ของการถ่ายทอดข้อมูลสูงสุด 1.5 เมกะเฮิรตซ์

4) สามารถเลือกให้ถ่ายข้อมูลในบิต LSB และ MSB ก่อนได้

5) เลือกการถ่ายข้อมูลได้ 4 อัตรา

6) สามารถสร้างสัญญาณอินเตอร์รัปต์เมื่อการถ่ายทอดข้อมูลเสร็จสิ้นลง

7) มีการป้องกันการเขียนข้อมูลชนกัน

8) เมื่อทำงานเป็นอุปกรณ์สเลฟสามารถออกจากโหมดประหยัดพลังงาน แบบไอเดิลได้เมื่อมีการร้องขอให้ติดต่อกับระบบ SPI

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.7.2 ขาสัญญาณของการเชื่อมต่อแบบ SPI

มีด้วยกัน 4 ขาคือ

1) ขา MOSI (Master Out Slave In) สำหรับอุปกรณ์มาสเตอร์ขานี้จะเป็นขาข้อมูลออกสำหรับส่งไปยังอุปกรณ์สเลฟ ในขณะที่ถ้าเป็นอุปกรณ์สเลฟ ขานี้จะเป็นขาข้อมูลเข้า

2) ขา MISO (Master In Slave Out) สำหรับอุปกรณ์มาสเตอร์ ขานี้จะเป็นขาข้อมูลเข้าจากอุปกรณ์สเลฟ ขานี้จะเป็นขาข้อมูลออกส่งไปยังอุปกรณ์มาสเตอร์

3) ขา SCK (SPI Clock) สำหรับอุปกรณ์มาสเตอร์ ขานี้จะเป็นขาส่งสัญญาณนาฬิกาออกไปยังอุปกรณ์สเลฟเพื่อกำหนดจังหวะการถ่ายทอดข้อมูลให้ตรงกัน ในขณะที่ถ้าเป็นอุปกรณ์สเลฟ ขานี้จะเป็นขารับสัญญาณนาฬิกาจากอุปกรณ์มาสเตอร์

4) ขา SS (Slave Select) ใช้เป็นขาเลือกอุปกรณ์สเลฟในกรณีที่มีการต่ออุปกรณ์สเลฟพ่วงกันหลายตัว ทำงานที่ลอจิก “0”

2.7.3 การทำงานของ SPI

การเขียนข้อมูลลงในรีจิสเตอร์ข้อมูลของอุปกรณ์สเลฟเริ่มต้นด้วยซีพียูทำการกระตุ้นให้วงจรกำเนิดสัญญาณนาฬิกาทำงาน ข้อมูลที่ต้องการเขียนจะออกจากอุปกรณ์มาสเตอร์ซึ่งในที่นี้ คือ ไมโครคอนโทรลเลอร์ ในลักษณะอนุกรมผ่านทางขา MOSI ไปเข้ายังขา MOSI ของอุปกรณ์สเลฟ หลังจากเลื่อนข้อมูลครบ 1 ไบต์ วงจรกำเนิดสัญญาณนาฬิกาของส่วน SPI จะหยุดทำงานแล้วทำการเซตบิต SPIF เพื่อแจ้งการถ่ายทอดข้อมูลสิ้นสุดลง ถ้าหากบิต SPIE ในรีจิสเตอร์ควบคุม SPI และบิต ES ในรีจิสเตอร์ SCON ได้รับการเซตไว้ด้วย จะเป็นการอีนาเบิลให้เกิดการอินเตอร์รัปต์ขึ้น

ขา SS เป็นขาสำหรับเลือกอุปกรณ์สเลฟ ถ้ามีการเชื่อมต่ออุปกรณ์สเลฟตัวใดเข้าสู่ระบบการติดต่อ SPI ที่นี้ต้องได้รับลอจิก “0” ในกรณีเป็นอุปกรณ์มาสเตอร์ต้องทำให้ขานี้มีลอจิกเป็น “1” หากอุปกรณ์สเลฟใดที่ขานี้ได้รับลอจิก “1” จะเป็นการดิสเอเบิลการติดต่อกับระบบ SPI ในทันที และขา MOSI จะสามารถใช้งานเป็นขาพอร์ตอินพุตได้เท่านั้น

2.8 การติดต่อสื่อสารโดยระบบบัส I^2C

I^2C ย่อมาจาก Inter-IC Communication หมายถึง การติดต่อสื่อสารระหว่างไอซีโดยระบบบัส I^2C ได้รับการพัฒนาจากบริษัทฟิลิปส์ ด้วยจุดมุ่งหมายหลัก คือ ต้องการให้ไอซีหรือโมดูลสามารถติดต่อ สั่งงาน และควบคุมภายใต้สายสัญญาณเพียง 2 เส้น เส้นหนึ่ง คือ สายข้อมูลอนุกรม หรือ SDA ส่วนสายสัญญาณอีกสายหนึ่ง คือ สายสัญญาณนาฬิกาหรือ SCL ของอุปกรณ์แต่ละตัวขนานกันหรือพ่วงไป ส่วนการกำหนดตำแหน่ง สำหรับอุปกรณ์แต่ละตัว จะใช้รหัสข้อมูลและการกำหนดสภาวะลอจิกที่ขาตำแหน่งของอุปกรณ์แต่ละตัว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สายข้อมูลบนระบบ I^2C มีชื่อเรียกเป็นทางการว่า สายอนุกรมหรือ สาย SDA (Serial Data line) ส่วนสายสัญญาณนาฬิกามีชื่อเรียกว่า สายสัญญาณนาฬิกาอนุกรมหรือ สาย SCL (Serial Clock line)

สาย SDA และ สาย SCL เป็นสายสัญญาณ 2 ทิศทาง ต้องมีการต่อตัวต้านทานพูลอัปกับแรงดัน 5 โวลต์อยู่ตลอดเวลา เพื่อให้สายมีสถานะเป็นลอจิกสูง ในขณะที่มีการติดต่อใช้งาน

อัตราการถ่ายทอดข้อมูลสูงถึง 100 กิโลบิตต่อวินาที หรือในโหมดปกติสูงถึง 400 กิโลบิตต่อวินาที การเข้าถึงระบบ จะใช้ข้อมูลสำหรับการเข้าถึง 2 ค่า คือ 7 บิต หรือ 10 บิต

ข้อเด่นอีกประการหนึ่งของบัส I^2C คือสามารถเชื่อมต่อกับอุปกรณ์ไปเลี้ยงไม่เท่ากันให้สามารถติดต่อถึงกันได้โดยอุปกรณ์บนระบบ

2.8.1 หลักการของบัส I^2C

การกำหนดการติดต่อต้องมีรูปแบบการติดต่อที่เป็นข้อตกลงพื้นฐานดังนี้

1) อุปกรณ์ที่สร้างข้อมูลหรือส่งข้อมูลเรียกว่า ตัวส่ง

2) อุปกรณ์ที่เป็นผู้รับข้อมูลเรียกว่า ตัวรับ

3) อุปกรณ์ที่ทำหน้าที่ควบคุมจังหวะการติดต่อบนบัส I^2C เรียกว่ามาสเตอร์

4) อุปกรณ์ที่ถูกควบคุมหรืออุปกรณ์ต่อพ่วงเข้าบนบัส I^2C เรียกว่า สเลฟ

ข้อกำหนดสำคัญของการติดต่อบนบัส I^2C สาย SDA และ สาย SCL คือ

1) การถ่ายทอดข้อมูลจะเกิดขึ้นได้เมื่อบัสว่างเท่านั้น

2) ในระหว่างการถ่ายทอดข้อมูลเมื่อใดก็ตามที่สาย SCL มีสถานะเป็นลอจิกสูง

2.8.2 สภาวะที่เกิดขึ้นบนบัส I^2C

มีด้วยกัน 5 สภาวะดังนี้

1) บัสว่าง (Bus Not Busy) สภาวะที่เกิดขึ้นเมื่อสถานะบนสาย SDA และ SCL เป็นลอจิกสูงทั้งคู่ หมายความว่า มีการถ่ายทอดข้อมูลสามารถเริ่มต้นขึ้นได้

2) เริ่มต้นการถ่ายทอดข้อมูล (Start data tranfer) เกิดขึ้นเมื่อ SDA มีการเปลี่ยนแปลงระดับลอจิกจากสูงไปต่ำ ในขณะที่สาย SCL มีสถานะเป็นลอจิกสูงเรียกสภาวะที่เกิดขึ้นว่า สภาวะเริ่มต้น

3) หยุดการถ่ายทอดข้อมูล (Stop data tranfer) เกิดขึ้นเมื่อสาย SDA มีการเปลี่ยนแปลงระดับลอจิกจากต่ำไปสูง ในขณะที่สาย SCL มีสถานะเป็นลอจิกสูงเรียกสภาวะที่เกิดขึ้นว่า สภาวะหยุด

4) ข้อมูลดำรงอยู่บนบัส (Data valid) สภาวะนี้เกิดขึ้นถัดจากสภาวะเริ่มต้นเกิดขึ้นเมื่อสาย SCL เป็นลอจิกสูงและสาย SDA คงที่ เพื่อให้รับรู้ว่ามีการส่งข้อมูล “0” หรือข้อมูล “1” เข้ามา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5) รับรู้ข้อมูล (Acknowledge) เกิดขึ้นหลังจากการถ่ายทอดข้อมูลจากตัวส่ง มายังตัวรับเกิดขึ้นอย่างสมบูรณ์ โดยตัวส่งจะทำการส่งข้อมูล 1 บิต เรียกว่า บิตรับรู้ มีสถานะเป็นลอจิกสูง หลังจากส่งข้อมูลมาครบถ้วน

2.8.3 การต่ออุปกรณ์ระบบบัส I^2C กับไมโครคอนโทรลเลอร์

สามารถทำได้โดยใช้ขาเพียง 2 ขา โดยกำหนดให้ขาหนึ่งเป็น SDA และ อีกขาหนึ่งเป็น SCL และต่อตัวต้านทาน 4.7 กิโลโอห์ม ที่ขาทั้ง 2 ขาก็สามารถติดต่อกับอุปกรณ์ระบบบัส I^2Cได้

2.8.4 การเขียนโปรแกรมติดต่อบัส I^2C

เริ่มต้นด้วยการสร้างสภาวะมาตรฐานของบัส ประกอบด้วย สภาวะเริ่มต้น, สภาวะสิ้นสุด การส่งข้อมูล, สภาวะหยุด, และสัญญาณนาฬิกาบนขา SCL

1) การสร้างสภาวะเริ่มต้น

1.1) ทำให้ขา SDA และ ขา SCL มีลอจิกเป็น "1"

1.2) ทำให้ขา SDA เป็น "0" โดย ขา SCL เป็น "1"

1.3) กำหนดให้ขาทั้งสองขาเป็น "0" ทั้งคู่ ซึ่งพร้อมในการติดต่อ

2) การสร้างสภาวะหยุด

1.1) กำหนดให้ขาทั้งสองขาเป็น "0" ทั้งคู่

1.2) ทำให้ขา SCL เป็น "0" โดยที่ ขา SDA ยังเป็น "1"

1.3) ทำให้ขา SDA เป็น "1" ซึ่งทำให้บัสกลับสู่สภาวะว่าง

3) การส่งข้อมูลลอจิก "0"

1.1) ทำให้ขา SDA เป็น "0" สำหรับส่งข้อมูลลอจิก "0"

1.2) ทำให้ขา SCL เป็น "1" สำหรับป้อนสัญญาณนาฬิกาโดยที่ ขา SDA ยังเป็น "0" อยู่

1.3) ทำให้ขา SCL กลับมาเป็น "0"

4) การส่งข้อมูลลอจิก "1"

1.1) ทำให้ขา SDA เป็น "1" สำหรับส่งข้อมูลลอจิก "1"

1.2) ทำให้ขา SCL เป็น "1" สำหรับป้อนสัญญาณนาฬิกาโดยที่ ขา SDA ยังเป็น "1"อยู่

1.3) ทำให้ขา SCL กลับมาเป็น "0"

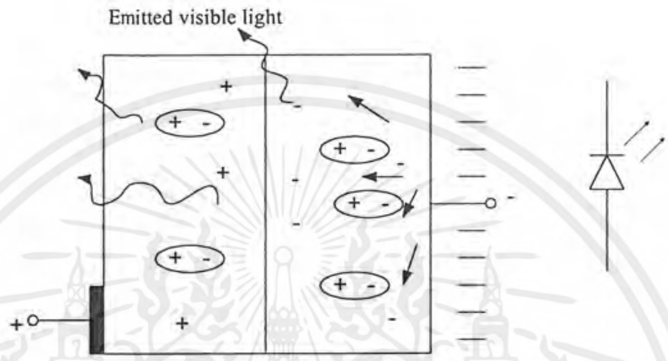
2.9 ไดโอดเปล่งแสง

ไดโอดเปล่งแสง (Light Emitting Diode LED) เป็นไดโอดที่สามารถเปล่งแสงสว่างออกมาด้วยคลื่นความถี่เดียวและมีเฟสต่อเนื่องได้ ซึ่งต่างจากแสงธรรมดากับที่คนเรามองเห็น ประกอบไป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ด้วยคลื่นที่มีความถี่และเฟสต่างๆ มารวมกัน

โครงสร้างของไดโอดเปล่งแสงเหมือนกับไดโอดทั่วไปที่ประกอบขึ้นจากการเอาสารพีและสารเอ็นมาประกบกัน โดยผิวข้างหนึ่งเป็นมันคล้ายกระจก เมื่อเราป้อนแรงดันไบอัสตรงให้แก่ไดโอดเปล่งแสงจะทำให้อิเล็กตรอนที่สารกึ่งตัวนำชนิดเอ็นมีพลังงานสูงขึ้นจนสามารถวิ่งข้ามรอยต่อไปรวมกับโฮลในสารพีได้ ทำให้เกิดพลังงานที่เราเรียกกันว่า "พลังงานโฟตอน" เปล่งออกมา



รูปที่ 2.7 โครงสร้าง และสัญลักษณ์ของไดโอดเปล่งแสง

ปรากฏการณ์ใหม่ที่เกิดขึ้นกับตัวไดโอดชนิดนี้บังเกิดขึ้นไม่ได้ถ้าหากไม่ใช้กระบวนการซึ่งเรียกว่า อิเล็กโตรลูมิเนสเซ็นส์ (Electroluminescence) ดังรูปที่ 2.7 คือต้องสร้างให้โลหะหน้าสัมผัสชนิดพีมีขนาดเล็กกว่าชนิดเอ็น อันจะทำให้พลังงานโฟตอนมีจำนวนมากที่สุด และที่สำคัญการที่จะทำให้พลังงานนี้เรืองแสงขึ้นมาสว่างนั้นจำเป็นต้องเติมสารเจือบางอย่างลงไปในซิลิกอนหรือเยอรมันเนียมด้วย สารนั้นก็คือแกเลี่ยมอาเซไนด์ฟอสไฟต์ (Gallium Arsenide Phosphide : GaAsP) หรือแกลเลี่ยมฟอสไฟต์ (Gallium Phosphide : GaP) และนี่เองที่ทำให้อิเล็กทรอนิกส์ทางด้านแสงได้รับการพัฒนาขึ้นอย่างรวดเร็ว จนได้อุปกรณ์อิเล็กทรอนิกส์ทางแสงเพิ่มขึ้นมาอย่างเช่น ไฟเบอร์ออฟติค ไอซีประเภทออฟติคอล รวมไปถึงการเกิดวงจรที่มีการเชื่อมโยงพลังงานต่างๆ ด้วยกรรมวิธี "ออฟโตไอโวเลเตอร์" (Opto Isolato)

หน่วยสำหรับการวัดความเข้มของแสงของไดโอดเปล่งแสง เราจะวัดกันเป็นแคนเดลลา (Candella) โดย 1 แคนเดลลามีความเข้มเท่ากับ 4π ลูเมนส์ ไดโอดเปล่งแสงอาจมีสีสันต่างๆ กันไป เช่น เขียว แดง เหลือง ส้ม และขาวและคาดหวังว่าระยะต่อไปเราอาจจะมีไดโอดแปล่งแสงสีน้ำเงินออกมาใช้ โดยทั่วไปไดโอดเปล่งแสงที่มีใช้กันในปัจจุบันใช้กับระดับแรงดัน 1.7 ถึง 3.3 โวลต์ ซึ่งจะเป็นค่าแรงดันที่ทำให้ไดโอดเปล่งแสงทำงานได้สมบูรณ์ที่สุดเมื่อมาใช้กับอุปกรณ์โซลิดสเตท

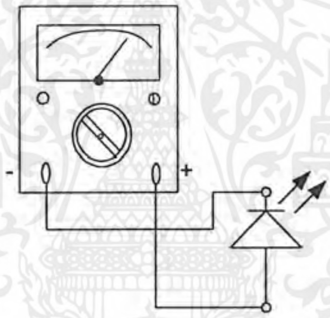
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สำนักหอสมุดกลาง พระจอมเกล้าลาดกระบัง

โดยจะมีความเร็วในการทำงานและความสว่างอยู่ในระดับพอดี โดยกำลังตั้งแต่ 10 ถึง 150 มิลลิวัตต์ อายุการใช้งานมากกว่า 100,000 ชั่วโมง

ปัจจุบันไดโอดเปล่งธรรมดา สามารถทำให้มีแสงสองสีในตัวเดียวกันเราเรียกว่า ไดโอดเปล่งแสงแบบ Two-Lead เมื่อเรากลับไบอัสในแต่ละครั้งจะมีสีสันที่แตกต่างกันออกมา เช่นครั้งแรกเป็นสีเขียว เมื่อเรากลับไบอัสมันจะกลายเป็นสีแดง ซึ่งถ้าเราดูตัวมันจากภายนอกแล้ว มันก็คือไดโอดเปล่งแสงตัวใสๆ เหมือนตัวอื่นๆ และมีขาใช้งานเพียง 2 ขาเท่านั้นคือ ขาที่เป็น A1 และ A2

วิธีการทดสอบไดโอดเปล่งแสง สามารถทดสอบได้ว่าดีหรือเสียได้ ด้วยการจ่ายไบอัสให้กับไดโอดเปล่งแสง โดยตั้งมิเตอร์ Rx1 วัด เข็มมิเตอร์จะขึ้นครั้งหนึ่งไม่ขึ้นครั้งหนึ่งเหมือนกับการวัดไดโอดทั่วไป ครั้งที่ขึ้นก็คือลักษณะไบอัสตรง ไดโอดเปล่งแสงจะเปล่งแสงออกมาให้เราเห็นฉะนั้นเวลาวัดเราอาจจะสังเกตจากแสงเอาก็ได้

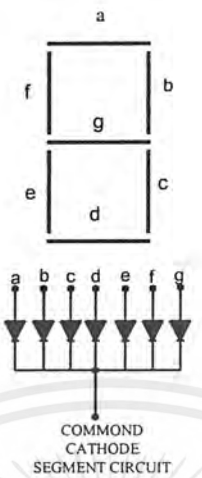


รูปที่ 2.8 การตรวจสอบไดโอดเปล่งแสง

2.10 จอแสดงผลแบบ 7 ส่วน

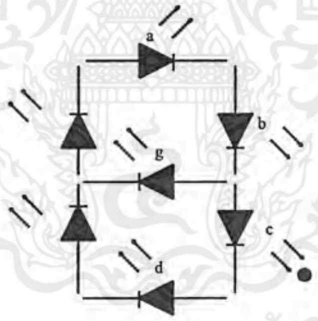
ประกอบด้วยไดโอดเปล่งแสงนำมาต่อกันเป็นรูปเลข 8 มีขาร่วม 1 ขา (มีทั้งคาโถดร่วมและอาโนดร่วม) และขาของไดโอดเปล่งแสงแต่ละตัวแสดงดังรูปที่ 2.7 แสดงถึงรูปร่างและวงจรเสมือน ซึ่งจะใช้ในตัวอย่างนี้เป็นแบบคาโถดร่วม ถ้ามีการใช้งานมากกว่า 1 อัน เราต้องใช้วิธีการสแกนด้วยความถี่สูงจนการมองเห็นของมนุษย์ไม่สามารถจับการกระพริบได้ อัตราการกวาดในแต่ละครั้งจะหน่วงไว้ 10 มิลลิวินาที หรือมากกว่าและค่าจะถูกด้วยจำนวนแสดงที่ใช้เพื่อหาช่วงเวลา ที่ใช้ทำให้หลอดสว่างของแต่ละอัน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้
40184



รูปที่ 2.9 วงจรแสดงผลแบบ 7 ส่วน

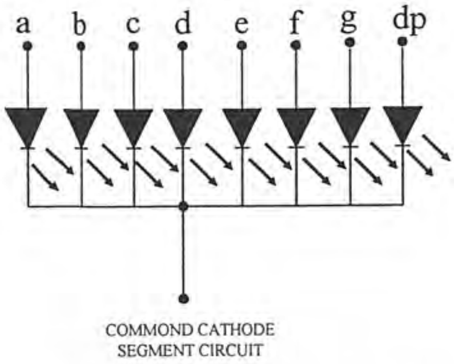
ส่วนแสดงผลเป็นชุดของไดโอดเปล่งแสงแบบ 7 ส่วน 6 หลัก จอแสดงผลแบบ 7 ส่วนแต่ละหลักจะมีโครงสร้างการจัดวางไดโอดเปล่งแสงเป็นดังรูป 2.10



รูปที่ 2.10 การจัดวางไดโอดเปล่งแสงเพื่อแสดงผลแบบ 7 ส่วน

จากรูปที่ 2.10 ไดโอดเปล่งแสงทั้ง 8 ตัวจะมีชื่อกำกับอยู่ คือ a, b, c, d, e, f, g และ dp จะเรียกไดโอดเปล่งแสงแต่ละตัวว่าเป็นส่วน a, b, c, d, e, f, g และ dp การแสดงผลแบบ 7 ส่วนของไดโอดเปล่งแสงในรูปที่ 2.10 มีการจัดเป็นคาโถดร่วมคือ ได้นำเอาขาคาโถดของไดโอดเปล่งแสงทุกตัวต่อรวมกันเข้าเป็นจุดร่วม ดังรูป 2.11

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

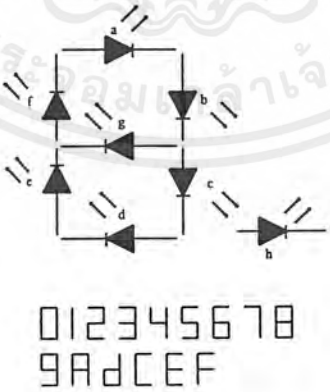


รูปที่ 2.11 การแสดงผลแบบ 7 ส่วนแบบคาโถดร่วม

การแสดงผลดังกล่าวจะแสดงผลได้ทีละ 1 หลักเท่านั้น แต่ก็สามารถแสดงผลให้เห็นได้ทั้ง 6 หลัก เหมือนการแสดงผลพร้อมกันได้ ด้วยวิธีมัลติเพล็กซ์ คือการแสดงหลักที่ 1, 2, 3 จนถึง 5 ทีละหลักวนไปจนครบรอบแล้วแสดงหลักที่ 1 ใหม่ ถ้าเขียนคำสั่งให้แสดงวนด้วยอัตราที่เหมาะสมจะทำให้เหมือนกับการแสดงพร้อมกันทั้ง 6 หลัก วิธีการนี้เหมือนกับการฉายภาพยนตร์ที่เป็นภาพนิ่งแต่มีการเลื่อนภาพด้วยอัตราประมาณ 25 ภาพต่อวินาทีก็จะเห็นเป็นภาพเคลื่อนไหว

2.10.1 การนำข้อมูลออกที่ภาคแสดงผลแบบ 7 ส่วน

ไดโอดเปล่งแสง เป็นอุปกรณ์ที่ใช้แสดงผลของเอาต์พุตอย่างง่ายๆ อย่างหนึ่ง และภาคแสดงผลแบบ 7 ส่วนจะใช้สำหรับแสดงตัวเลขฐานสิบ 0 ถึง 9 หรือแสดงเลขฐานสิบหก 0 ถึง F การกำหนดรูปแบบของตัวเลข และชื่อของแต่ละส่วน แสดงในรูป 2.12



รูปที่ 2.12 ภาคแสดงผลแบบ 7 ส่วน และการแสดงตัวเลขของฐานสิบฐานสิบหก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ถ้ากำหนดให้แสดงผลแบบ 7 ส่วนนี้เป็นแบบคาโถดร่วม และถ้าให้พอร์ตเอาต์พุตบิต D0 ต่ออยู่กับส่วน a บิต D1 ต่ออยู่กับส่วน b ตามลำดับ เมื่อบิตใดมีระดับเป็น 1 จะทำให้ส่วนนั้นติด ดังนั้นการแสดงตัวเลข สามารถเขียนเป็นรหัสเลขฐานสิบหกได้ดังตาราง วิธีการคือเมื่อต้องการให้การแสดงผลแบบ 7 ส่วนติดเป็นเลขใด ก็ส่งรูปแบบนั้นออกทางพอร์ตเอาต์พุตซึ่งแสดงในตารางที่ 2.3

ตารางที่ 2.3 รูปแบบของรหัสเพื่อการแสดงผลตัวเลขแบบ 7 ส่วน

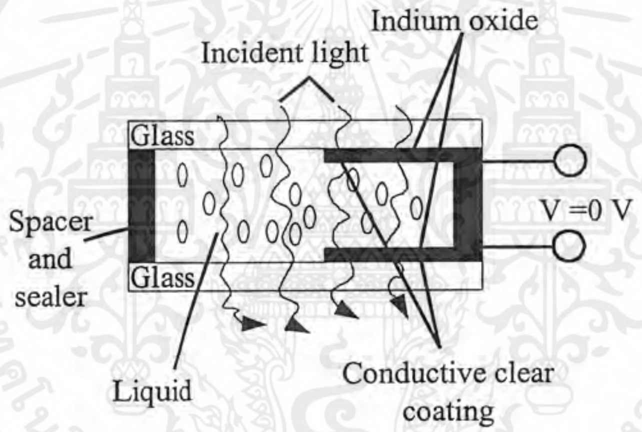
เลข	D7	D6	D5	D4	D3	D2	D1	D0	รหัส
	.	g	f	e	d	c	b	a	
0	0	0	1	1	1	1	1	1	3FH
1	0	0	0	0	0	1	1	0	06H
2	0	1	0	1	1	1	1	1	5BH
3	0	1	0	0	1	1	1	1	4FH
4	0	1	1	0	0	1	1	0	66H
5	0	1	1	0	1	1	0	1	6DH
6	0	1	1	1	1	1	0	1	7DH
7	0	0	0	0	0	1	1	1	07H
8	0	1	1	1	1	1	1	1	7FH
9	0	1	0	0	1	1	1	1	6FH
A	0	1	1	1	0	1	1	1	77H
B	0	1	1	1	1	1	0	0	7CH
C	0	0	1	1	1	0	0	1	39H
D	0	1	1	1	1	1	1	0	5EH
E	0	1	1	1	1	1	0	1	79H
F	0	1	1	1	0	0	0	1	71H

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.11 จอแสดงผลแบบผลึกเหลว

จอแสดงผลแบบผลึกเหลว (Liquid Cystal Display) ซึ่งเราจะใช้สำหรับวงจรแสดงผลระดับตัวเลขในเครื่องใช้อิเล็กทรอนิกส์ในยุคปัจจุบัน ซึ่งอยู่ในภาคแสดงผล (Display) ของเครื่องคิดเลข, นาฬิกา, หน้าปัทม์เครื่องรับวิทยุ-โทรทัศน์, เครื่องวัดและทดสอบระบบดิจิตอล, วงจรแสดงทางดิจิตอล ฯลฯ

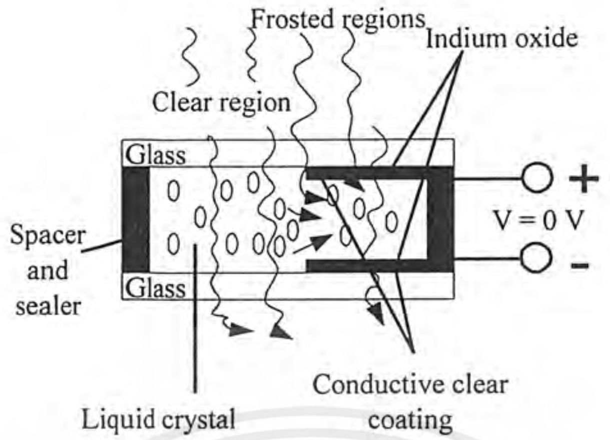
ผลึกเหลว (Liquid Cystal) อันเป็นตัวหลักของการแสดงผลแบบผลึกเหลวนี้ เป็นสสารชนิดหนึ่งซึ่งมีสภาพการไหลเหมือนอย่างของเหลว แต่การรวมตัวของโมเลกุลของมันจะรวมตัวกันแบบโมเลกุลของของแข็ง ซึ่งการรวมตัวแบบนี้เรียกว่าการรวมตัวแบบ นีแมติคลิควิดคริสตอล (Nematic Liquid Cystal) ดังแสดงโครงสร้างและโมเลกุลที่อยู่ภายในจะเป็นโมเลกุลตามที่ปรากฏในรูปที่ 2.13 และ 2.14



รูปที่ 2.13 แมติคลิควิดคริสตอลขณะที่ยังไม่มีการไบอัส

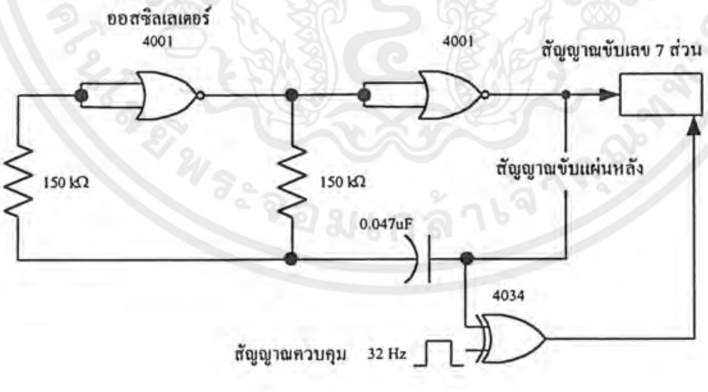
เมื่อเราจ่ายไบอัสเข้าไป (โดยทั่วไปจอแสดงผลแบบผลึกเหลวที่มีจำหน่ายในปัจจุบันจะให้แรงเคลื่อนไบอัสอยู่ในระดับ 6 หรือ 20 โวลต์สำหรับจอแสดงผลแบบผลึกเหลวขนาดใหญ่) จะทำให้เกิดความต่างศักย์ขึ้นภายใน ลักษณะนี้เองที่ทำให้โมเลกุลภายในของผลึกเกิดการสะท้อนแสงผ่านหลอดแก้วดาร์คไลท์ออกมา สำหรับอายุการใช้งานของจอแสดงผลแบบผลึกเหลวจะมีอายุการใช้งานโดยเฉลี่ยไม่ต่ำกว่า 10,000 ชั่วโมง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.14 แมติคลิควิดคริสตอล เมื่อจ่ายไบอัสเข้าไป

จอแสดงผลแบบผลึกเหลวที่ใช้อยู่ในปัจจุบันมีอยู่ 2 ชนิดคือ ชนิดไดนามิคสแกทเตอร์ริ่ง (Dynamic Scattering) กับชนิดฟีลด์เอฟเฟ็ค (Field-effect) จอแสดงผลแบบผลึกเหลวชนิดหลังนิยมใช้มากกว่าชนิดแรกเพราะกินกระแสเพียง 20% ของชนิดแรกและยังสามารถให้แสงได้ชัดแม้ในขณะที่แสงสว่างภายนอกจะมาก นอกจากนี้แล้วจอแสดงผลแบบผลึกเหลวชนิดเอฟเฟ็คยังแยกละเอียดออกไปอีกเป็นชนิดสะท้อนแสง, ชนิดให้แสงสว่างผ่านได้ และชนิดสเฟล็คตีฟ



รูปที่ 2.15 การให้ไบอัสจอแสดงผลแบบผลึกเหลว

การป้อนไบอัสให้แก่จอแสดงผลแบบผลึกเหลวต้องเป็นไฟฟ้ากระแสสลับที่มีความถี่พอเหมาะ และมีขนาดแรงเคลื่อนถูกต้อง แต่ไม่จำเป็นจะต้องเป็นรูปไซน์เสมอไป สัญญาณกระแสสลับนี้จะต้องมีค่าเฉลี่ยเป็น “0” หรือค่าดิวตี้ไซเกิล 50% ถ้าไม่เป็น “0” จริง จะทำให้อายุการใช้งานของ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จอแสดงผลแบบผลึกเหลวสั้นลง ในทางปฏิบัติโดยทั่วไปเราใช้วงจรอิเล็กทรอนิกส์ผลิตความถี่ และป้อนสัญญาณไฟฟ้าเข้าไปยังแผ่นแก้วที่เราเรียกว่าแบล็คเพลน และอีกส่วนหนึ่งก็จะเป็นส่วนของภาคแสดงผล

ปัจจุบันจอแสดงผลแบบผลึกเหลวได้ถูกนำมาใช้ในการแสดงผลมากขึ้น เนื่องจากราคาถูกลง และใช้งานได้ง่ายซึ่งเราอาจจะพบเห็นได้ทั่วไปในรูปของ นาฬิกา, เครื่องคิดเลข, เกมส์กด จนถึงเครื่องคอมพิวเตอร์

2.11.1 ข้อดีของจอแสดงผลแบบผลึกเหลว

1) บางเบาพกพาได้สะดวก

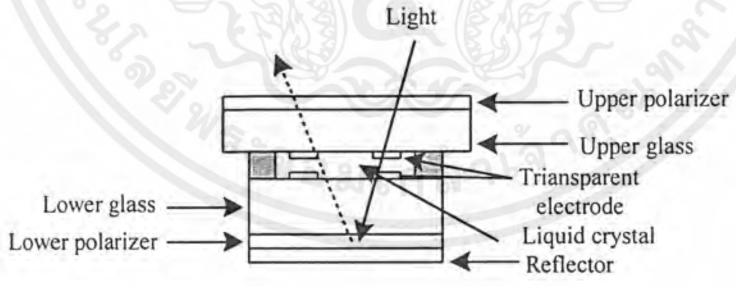
2) ใช้พลังงานน้อยเช่น อาจจะให้แรงดันต่ำขนาด 2 หรือ 3 โวลต์ และกระแสเพียงเล็กน้อยก็สามารถใช้งานได้แล้ว

3) ราคาถูก

4) ใช้งานในช่วงอุณหภูมิที่กว้างได้

2.11.2 โครงสร้างของจอแสดงผลแบบผลึกเหลว

ประกอบด้วยแผ่นแก้ว 2 แผ่นประกบกันโดยเว้นช่องตรงกลางไว้ 6 ถึง 10 ไมโครเมตร ผิวด้านในของแผ่นแก้วเคลือบด้วยตัวนำไฟฟ้าชนิดใสที่ไว้แสดงตัวอักษร ระหว่างตัวนำไฟฟ้าชนิดใสกับผลึกเหลว จะมีชั้นสารที่ทำให้โมเลกุลของผลึกรวมตัวกันในทิศทางของแสงที่ส่องมา เรียกว่า (Alignment Layer) และผลึกเหลวที่ใช้โดยทั่วไปจะเป็นแบบนีเมติก (Nematic)



รูปที่ 2.16 โครงสร้างของจอแสดงผลแบบผลึกเหลว

2.11.3 แบบต่างๆของจอแสดงผล

จอแสดงผลแบบผลึกเหลวสามารถแสดงผลให้เราเห็นได้โดยมีหลักการ 3 แบบคือ

1) แบบสะท้อน (Reflective Mode) จะมีสารประเภทโลหะเคลือบอยู่ที่แผ่นหลังของจอแสดงผลแบบผลึกเหลวแบบนี้จะเหมาะสำหรับที่มีแสงสว่างเพียงพอ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2) แบบส่งผ่าน (Transmissive Mode) แบบนี้จะวางหลอดไฟไว้ด้านหลังทำให้อ่านค่าแสดงผลได้ชัดเจน

3) แบบส่งผ่าน/สะท้อน (Transflective Mode) เป็นการรวมตัวกันระหว่าง 2 แบบที่ผ่านเข้ามาด้วยกัน

2.11.4 จอแสดงผลแบบผลึกเหลวแบบดอตเมตริกซ์

ในแต่ละแบบจะมีส่วนประกอบใหญ่ๆ แบ่งได้เป็น

1) ดอตเมตริกซ์ เป็นตัวแสดงผลให้เรามองเห็นในลักษณะการปิดและการเปิดตัวเองกับนั่นเองคือ ส่วนที่เป็นตัวกระจกบรรจุผลึก

2) ตัวขับ (Driver) เป็นตัวรับสัญญาณมาจากส่วนควบคุมเพื่อมาขับผลึกจอแสดงผลแบบผลึกเหลว

3) ตัวควบคุม (Controller) เป็นตัวรับสัญญาณข้อมูลจากอุปกรณ์ภายนอกมาและจัดการควบคุม จอแสดงผลแบบผลึกเหลวโมดูล ให้ทำงานแสดงผลต่างๆ เช่น การลบจอภาพ การเกิดตัวอักษร, การเลื่อนข้อมูลบนจอแสดงผลแบบผลึกเหลว เป็นต้น โดยมีเบอร์ไอซีที่นิยมใช้กันคือ HD44780 ซึ่งจะใช้ในแบบคุณสมบัติ คือจอแสดงผลแบบผลึกเหลวที่แสดงข้อมูลในรูปตัวอักษร ส่วนเบอร์ HD61830 ซึ่งจะใช้ในแบบกราฟฟิก ก็คือ จอแสดงผลแบบผลึกเหลวที่แสดงข้อมูลในลักษณะรูปภาพหรือตัวอักษรก็ได้

2.12 สเต็ปปิ้งมอเตอร์

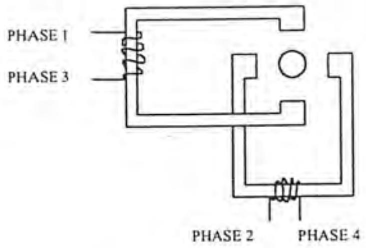
สเต็ปปิ้งมอเตอร์ ถือว่าเป็นอุปกรณ์เอาต์พุตอย่างหนึ่ง ซึ่งสามารถควบคุมได้ด้วยไมโครคอมพิวเตอร์ ลักษณะการทำงานของสเต็ปปิ้งมอเตอร์ จะเคลื่อนที่เป็นขั้น ซึ่งอาจเป็นขั้นละ 18, 5, 7.5 องศา ก็แล้วแต่ชนิดของมอเตอร์ส่วนใหญ่สเต็ปปิ้งมอเตอร์จะใช้ในงานควบคุมระบบดิจิตอล เช่น อุปกรณ์ประกอบคอมพิวเตอร์ต่างๆ เช่น พริ้นซ์เตอร์, X-Y พล็อตเตอร์, ดิสก์ไดรว์ฟ ตลอดจนอุปกรณ์ในระบบงานอิเล็กทรอนิกส์อุตสาหกรรม หรือเครื่องมือวัดและควบคุมอื่นๆ

สเต็ปปิ้งมอเตอร์ มีส่วนประกอบสำคัญ 2 คือ

1) โรเตอร์ (ส่วนที่หมุนได้) จะเป็นแม่เหล็กถาวร หรืออื่นๆ

2) สเตเตอร์ (ส่วนที่อยู่กับที่) เป็นขดลวดหลายๆ ขด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.17 สเต็ปปิ้งมอเตอร์ 4 เฟสแบบแม่เหล็กถาวรยูนิโพลาร์

สเต็ปปิ้งมอเตอร์ ที่เห็นกันในปัจจุบันแบ่งได้ 3 แบบ คือ

1) แบบแม่เหล็กถาวร (PM = Permanent Magnet)

2) แบบแปรค่ารีลักแตนซ์ (VR = Varible Reluctance)

3) แบบลูกผสม (H = Hybrid)

สเต็ปปิ้งมอเตอร์แบบแม่เหล็กถาวรนั้นจะมีสเตเตอร์ที่พันขดลวดไว้หลายๆ โพล มีโรเตอร์เป็นรูปทรงกระบอกฟันเลื่อย และโรเตอร์ทำด้วยแม่เหล็กถาวร เพื่อป้องกันไฟกระแสตรงให้กับขดสเตเตอร์ จะให้เกิดแรงผลักต่อโรเตอร์ด้วยอิเล็กโตรแมกไนติกฟอร์ส จะทำให้มอเตอร์หมุน มอเตอร์แบบแม่เหล็กถาวรจะเกิดแรงฉุดยึดให้โรเตอร์หยุดอยู่กับที่แม้จะไม่ได้ป้อนเข้าขดลวด

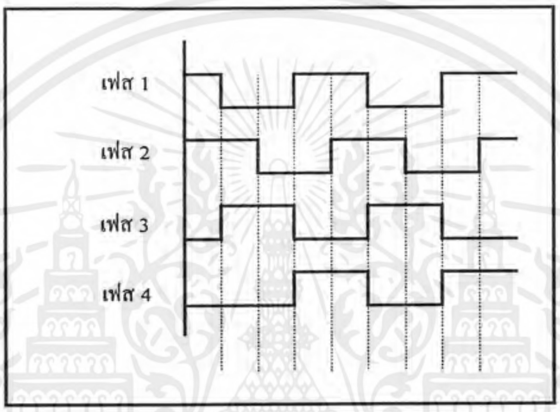
ส่วนมอเตอร์แบบแปรค่ารีลักแตนซ์ จะมีการหมุนโรเตอร์ได้อย่างอิสระ แม้จะไม่ได้จ่ายไฟให้ โรเตอร์ของมันทำจากสารฟอร์สโรแมกเนติกขนาดกำลังอ่อน และมีลักษณะเป็นฟันเลื่อยรูปทรงกระบอก โดยจะมีความสัมพันธ์โดยตรงกับจำนวนโพลในสเตเตอร์ จึงมีหน้าที่กำหนดมุมที่หมุนไปแต่ละครั้งเพื่อป้อนไฟเข้าขดลวดสเตเตอร์ แรงบิดที่เกิดขึ้นจะไปหมุนโรเตอร์ให้หมุนในเส้นทางของอำนาจแม่เหล็กที่มีค่ารีลักแตนซ์ต่ำที่สุด ตำแหน่งที่เกิดแน่นอนและมีเสถียรภาพ แต่จะเกิดขึ้นได้หลายๆ จุด ดังนั้นเมื่อป้อนไฟเข้าขดลวดต่างๆ ในมอเตอร์ต่างขดกันไป ก็จะทำให้มอเตอร์หมุนไปในตำแหน่งต่างๆ กันโรเตอร์ของมอเตอร์แบบแปรค่ารีลักแตนซ์จะมีความเฉื่อยของโรเตอร์น้อยจึงมีความเร็วสูงกว่ามอเตอร์แบบแม่เหล็กถาวร

สำหรับมอเตอร์แบบลูกผสม จะเป็นลูกผสมของมอเตอร์แบบแปรค่ารีลักแตนซ์กับมอเตอร์แบบแม่เหล็กถาวรโดยจะมีสเตเตอร์ ซึ่งคล้ายกับที่ใช้ในมอเตอร์แบบแปรค่ารีลักแตนซ์ สำหรับโรเตอร์มีหมวกหุ้มปลาย ซึ่งมีลักษณะของสารแม่เหล็กที่มีกำลังสูงโดยการควบคุมขนาดรูปร่างของหมวกแม่เหล็กอย่างดี ทำให้ได้มุมการหมุนแต่ละครั้งน้อย และแม่นยำ ข้อดีก็คือ ให้แรงบิดสูง และมีขนาดกระทัดรัด และให้แรงฉุดยึดโรเตอร์นิ่งกับที่ตอนไม่จ่ายไฟ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 2.4 การจ่ายกระแสให้กับขดลวดแต่ละเฟส

ขั้น	เฟส			
	1	2	3	4
1	1	1	0	0
2	0	1	1	0
3	0	1	1	1
4	1	0	0	1



รูปที่ 2.18 ผังเวลาการทำงานของการจ่ายกระแสให้กับขดลวดแต่ละเฟส

จากลักษณะของมุมโรเตอร์หมุนกับกระแสไฟฟ้าที่ป้อนแก่เฟสต่างๆ เราสามารถสั่งงานให้สเต็ปปิ้งมอเตอร์ หมุนได้ 3 อย่างคือ

1) แบบจ่ายกระแสไฟฟ้าให้เฟสเดียววนเวียนกันไปเรียกว่า One Excitation หรือการขับครึ่งขั้น (Half Drive) คือ $\phi1,\phi2,\phi3,\phi4$ การ One Excitation แบบนี้แรงบิดจะน้อย

2) แบบจ่ายกระแสไฟฟ้าให้พร้อมกันทีเดียว 2 เฟส เรียกว่า Two Excitation หรือการขับเต็มขั้น(Full Step) คือ $\phi1\phi2,\phi2\phi3,\phi3\phi4,\phi4\phi1$ หมุนเวียนกันไปแบบนี้แรงบิดจะมาก

3) แบบจ่ายไฟฟ้าให้ทีละ 1 เฟส สลับกับ 2 เฟส เรียกว่า One-Two Excitation หรือ Half Step เหมือนรูปแสดงมุมของโรเตอร์ แต่แบบนี้จำนวนขั้นจะเพิ่มขึ้นเป็น 2 เท่าของแบบแรก แต่แรงเฉลี่ยจะน้อย จากการจ่ายกระแสไฟฟ้าให้เฟสทั้ง 3 อย่าง เราสามารถสั่งให้สเต็ปปิ้งมอเตอร์หมุนทวนเข็มได้โดยการมองการจ่ายกระแสไฟฟ้าย้อนกลับ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 2.5 การป้อนกระแสไฟฟ้าให้แก่เฟสต่างๆ ของสเต็ปปิ้งมอเตอร์

$\phi 4$	$\phi 3$	$\phi 2$	$\phi 1$
0	0	0	1
0	0	1	0
0	1	0	0
1	0	0	0

2.12.1 โหมดเต็มขั้น (Full Step Mode)

แบ่งการขับออกเป็น 2 แบบคือ

1) การขับแบบเวฟยูนิโพลาร์ เป็นการป้อนกระแสให้กับขดลวดแต่ละขดลวดของสเต็ปปิ้งมอเตอร์ทีละขดเรียงลำดับกันไป กระแสที่ไหลในขดลวดจะไหลในทิศทางเดียวกันทุกขด ลักษณะการขับแบบนี้จะให้แรงบิดน้อย

2) การขับแบบ 2 เฟส เป็นการป้อนกระแสให้แก่ขดลวด 2 ขด

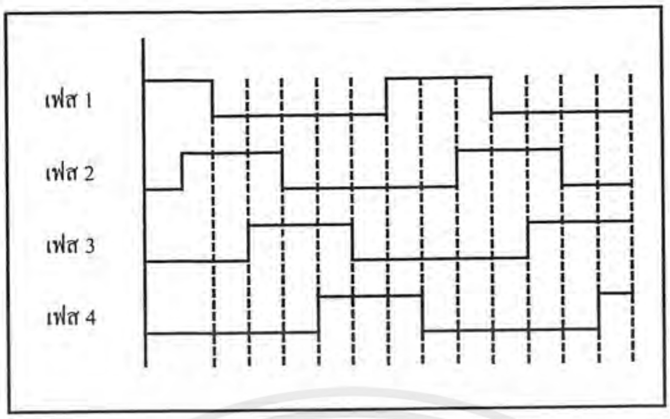
2.12.2 โหมดครึ่งขั้น (Half Step Mode)

เป็นการนำเอาวิธีขับแบบเวฟยูนิโพลาร์ผสมกับแบบการขับแบบ 2 เฟสในการหมุนจะหมุนครั้งละครึ่งขั้น และจำนวนองศาที่ได้จากการหมุนแต่ละครั้งจะเท่ากับครึ่งหนึ่งของการหมุนแบบโหมดเต็มขั้น การป้อนกระแสจะเป็นไปตามตาราง และไทม์มิ่งไดอะแกรมต่อไปนี้

ตารางที่ 2.6 การจ่ายกระแสให้กับขดลวด แบบโหมดครึ่งขั้น

ขั้น	เฟส			
	1	2	3	4
1	1	1	0	0
2	0	1	0	0
3	0	1	1	0
4	0	0	1	0
5	0	0	1	1
6	0	0	0	1
7	1	0	0	1
8	1	0	0	0

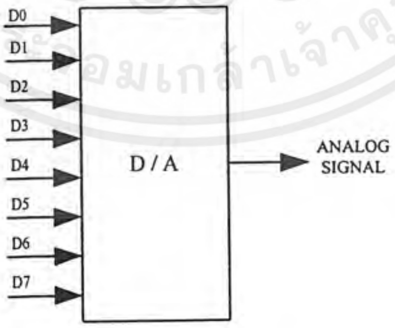
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.19 ผังการทำงานของการจ่ายกระแสให้กับขดลวด แบบโหมดครึ่งขั้น

2.13 การแปลงสัญญาณดิจิตอลเป็นสัญญาณแอนะลอก

ไมโครคอนโทรลเลอร์จะมีการทำงานในเชิงของดิจิตอลเสมอซึ่งมีระดับสัญญาณเป็นลอจิก 0 และ 1 แต่ในการประยุกต์ใช้งานจริงมักจะต้องเกี่ยวข้องกับสัญญาณที่เป็นแอนะลอก ซึ่งสัญญาณที่มีความแตกต่างหลายระดับ สัญญาณเหล่านี้ได้แก่ อุณหภูมิ ความดัน น้ำหนักเสียง ในทางอิเล็กทรอนิกส์เราสามารถเปลี่ยนแรงดันให้อยู่ในรูปสัญญาณเหล่านี้ได้ และระบบคอมพิวเตอร์ต้องเปลี่ยนสัญญาณ 0 และ 1 ให้เป็นสัญญาณแรงดันได้เช่นเดียวกัน ซึ่งสัญญาณ 0 และ 1 นี้จะอยู่ในรูปของข้อมูลทางคอมพิวเตอร์

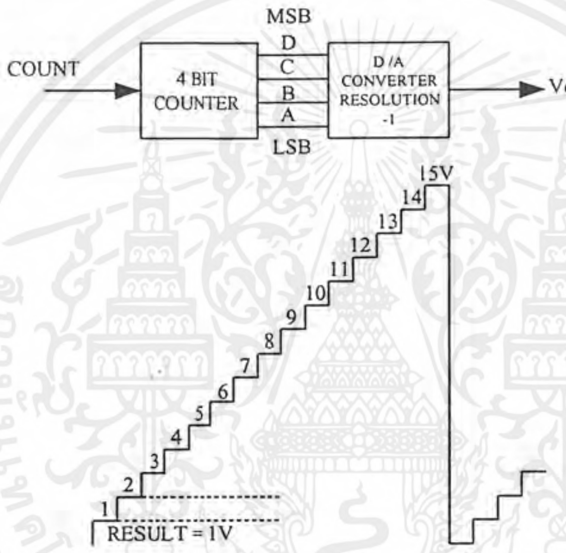


รูปที่ 2.20 ลักษณะของการแปลงสัญญาณดิจิตอลเป็นสัญญาณแอนะลอก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.13.1 คุณสมบัติทั่วไปของวงจรแปลงสัญญาณดิจิตอลเป็นสัญญาณแอนะลอก

1) การจำแนกแยกแยะ (Resolution) คือความสามารถในการแบ่งแยกระดับของสัญญาณ หรืออีกนัยหนึ่งก็คือจำนวนบิตของสัญญาณดิจิตอลนั่นเอง เช่น ขนาด 8 บิต จะหมายความว่า สามารถแยกสัญญาณแอนาล็อกไว้เป็น 256 ระดับ (2^8) ถ้าขนาด 12 บิต ก็จะแยกสัญญาณได้เป็น 4096 ระดับ ค่าการจำแนกแยกแยะบางครั้งก็จะแสดงอยู่ในรูปของเปอร์เซ็นต์โดยขนาด 8 บิตจะมีค่าเท่ากับ 100/256 คือ 0.39 %



รูปที่ 2.21 รูปคลื่นของสัญญาณแอนะลอก

รูปที่ 2.21 เป็นรูปแสดงสัญญาณแอนะลอกที่ได้ เมื่อสัญญาณดิจิตอลอินพุตเป็นสัญญาณจากวงจรนับ 4 บิต เริ่มนับตั้งแต่ “0000” ถึง “1111” สัญญาณแอนะลอกที่ได้จะมีค่าจำแนกแยกแยะเท่ากับ 1 โวลต์ ดังนั้นเมื่อสัญญาณจากวงจรนับนับไป 1 ครั้งก็จะทำให้แรงดันเอาต์พุตเปลี่ยนไป 1 โวลต์ ถ้าวงจรนับ นับวนไปเรื่อยๆ จาก “0000” ถึง “1111” แล้วมาเริ่มนับที่ “0000” ใหม่ ก็จะทำให้ได้สัญญาณแอนะลอกที่เรียกว่า สัญญาณขั้นบันได

ค่าการจำแนกแยกแยะไม่เพียงแต่จะเป็นค่าแสดงถึงจำนวนของแรงดันหรือกระแสต่อ 1 ขั้น เท่านั้น แต่ยังเป็นตัวแสดงถึงแรงดันหรือกระแสสูงสุดของเอาต์พุตอีกด้วย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2) ค่าแรงดันสูงสุด คือ ค่าแรงดันสูงสุดที่จะเป็นไปได้สัญญาณแอนะลอก สมมติว่าใช้วงจรแปลงสัญญาณดิจิตอลเป็นสัญญาณแอนะลอกขนาด 8 บิตจะเขียนเป็นสมการได้ดังนี้

$$Vo = Vref\left(\frac{A1}{2}+\frac{A2}{4}+\frac{A3}{8}+\frac{A4}{16}+\frac{A5}{32}+\frac{A6}{64}+\frac{A7}{128}+\frac{A8}{256}\right) \text{ โวลต์} \qquad \textbf{(2.1)}$$

A1 ถึง A8 คือสัญญาณดิจิตอลมีค่าเป็น 0 หรือ 1 ตามข้อมูลในที่นี้ถ้าสมมติ Vref = 10 โวลต์ ค่าสูงสุดที่เป็นไปได้คือ ให้ A1 ถึง A8 = 1 ดังนี้

$$Vo = 10 \text{ X } \frac{255}{256} = 9.9609 \text{ โวลต์} \qquad \textbf{(2.2)}$$

นั่นหมายความว่าสัญญาณแอนะลอกจะมีค่าสูงสุดน้อยกว่าแรงดันอ้างอิงอยู่ 1 ระดับเสียง และในที่นี้ความแตกต่างของแต่ละระดับจะเท่ากับ

$$Vdiff = \frac{10}{256} = 0.03906 \text{ โวลต์} \qquad \textbf{(2.3)}$$

3) ค่าความเที่ยงตรง (Accuracy) คือค่าเปรียบเทียบระหว่างแรงดันจริงๆ ที่ได้กับ โวลต์ที่กำหนดให้เป็นถ้าวงจรแปลงสัญญาณดิจิตอลเป็นสัญญาณแอนะลอกมีคุณสมบัติเป็น 10 โวลต์ และ +1.5% นั่นหมายถึง วงจรแปลงสัญญาณดิจิตอลเป็นสัญญาณแอนะลอกชุดนี้จะมีโอกาสผิดพลาดได้สูงคือ 0.0012 X 10 = 0.015 โวลต์ ปกติค่าความเที่ยงตรงในทางจุดปกติแล้วไม่ควรจะมีค่ามากกว่าครึ่งหนึ่งของระดับสัญญาณแอนะลอก หรือเท่ากับ +1/2 ของบิตที่ต่ำที่สุด (LSB) ในที่นี้คือ 0.039/2 = 0.0195 โวลต์

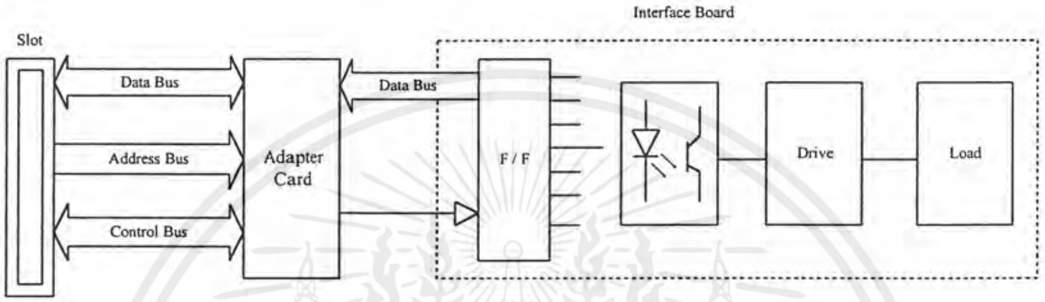
4) ค่าความเป็นเชิงเส้น (Linearity)คือค่าความคงที่ในการเปลี่ยนระดับของสัญญาณมีหน่วยเป็นเปอร์เซ็นต์ ปกติวงจรแปลงสัญญาณดิจิตอลเป็นสัญญาณแอนะลอกควรจะมีค่าการเปลี่ยนระดับเป็นเส้นตรง แต่ในความจริงอาจจะมีการเบี่ยงเบนเกิดขึ้นได้ ในทางจุดปกติไม่ควรมีค่ามากกว่า +1/2 LSB ซึ่งก็คือค่าเปอร์เซ็นต์ของการจำแนกแยกแยะหารด้วย 2 คือ 0.39122 = 0.019%

5) ค่าตั้งเวลา (Setting Time) คือ ค่าเวลาที่ใช้ในการเปลี่ยนระดับสัญญาณเท่ากับ ½ ตัวอย่างเช่น DAC 0808 จะมีค่าเท่ากับ 150 นาโนวินาที ค่านี้มีความสำคัญต่อความเร็วในการเปลี่ยนหรืออีกนัยหนึ่งก็คือ ความถี่ของสัญญาณแอนะลอกนั่นเอง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.14 การเชื่อมโยงทางแสง

ในการควบคุมอุปกรณ์เอาต์พุต นอกจากใช้งานแมกคานิกส์รีเลย์แล้วยังสามารถใช้อุปกรณ์ทางแสง ในการควบคุมการปิดเปิดสวิตช์อิเล็กทรอนิกส์ ซึ่งคุณสมบัติอย่างหนึ่งของการใช้แสง และอิเล็กทรอนิกส์คือ ความเร็วในการเปิด-ปิด จะทำได้เร็วกว่ามากเมื่อเทียบกับรีเลย์แบบแมกคานิกส์



รูปที่ 2.22 ผังการทำงานของการเชื่อมโยงทางแสงออก

ในส่วนของวงจรเชื่อมโยงทางแสงออกเป็นวงจรทางแสงออกเป็นวงจรที่ใช้แสงในการควบคุมการเปิดปิดของอิเล็กทรอนิกส์สวิตช์ แต่ก่อนจะเข้าถึงการทำงานของวงจร จะกล่าวถึงการทำงานของอุปกรณ์การเชื่อมโยงทางแสง ดังนี้

อุปกรณ์การเชื่อมโยงทางแสง เป็นอุปกรณ์เดี่ยวที่ประกอบด้วยแหล่งกำเนิดแสง และตัวตรวจจับแสง โดยที่ทั้งสองส่วนนี้แยกจากกันและกันโดยมีฉนวนโปร่งใสคั่นกลาง และชิ้นส่วนทั้งหมดจะถูกบรรจุอยู่ในตัวถังทึบแสง

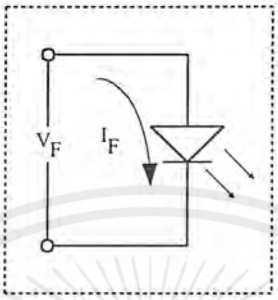
แหล่งกำเนิดแสงสำหรับต่อเชื่อมโยงทางแสงส่วนมากแล้วจะใช้ไดโอดเปล่งแสงอินฟาเรด (Infrared Emitting Diode) ที่ทำจากสารแกลเลี่ยมอาร์เซไนต์ (Gallium Arsenide : GaAs) ตัวตรวจจับหรืออุปกรณ์ภาคเอาต์พุตนั้นอาจเป็น โฟโต้ทรานซิสเตอร์ , โฟโต้ดาร์ลิงตัน , โฟโต้เอสซีอาร์

2.14.1 คุณสมบัติต่างๆ ของอุปกรณ์เชื่อมโยงทางแสง

ในที่นี้จะกล่าวถึงเฉพาะตัวแปรทางด้านไฟฟ้ากระแสตรงเท่านั้น ตัวแปรทางด้านกระแสตรงสามารถแบ่งออกเป็นอินพุต เอาต์พุต และอัตราส่วนของการส่งผ่านกระแส (Current Transfer Ratio : CTR) อัตราส่วนของการส่งผ่านกระแสนั้น เป็นอัตราส่วนระหว่างกระแสอินพุตต่อกระแสเอาต์พุตของตัวเชื่อมโยงทางแสง ซึ่งค่านี้จะขึ้นอยู่กับประสิทธิภาพของไดโอดเปล่งแสงอินฟราเรด และช่องว่างระหว่างชิ้นส่วนทางอินพุตและเอาต์พุต โดยที่พื้นที่, ความไว (Sensitivity) และอัตราการขยายตัวของตัวตรวจจับก็มีบทบาทที่สำคัญเช่นกัน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตัวแปรอินพุตทางด้านไฟฟ้ากระแสตรงซึ่งเป็นตัวแปรทางด้านไฟฟ้าของไดโอดเปล่งแสงอินฟราเรดได้แก่ ค่ากระแสของไดโอดเมื่อได้รับการไบอัสตรง (I_F) ค่าแรงดันตกคร่อมเมื่อได้รับไบอัสตรง (V_F) และค่าแรงดันสูงสุดที่ทนได้เมื่อได้รับไบอัสกลับ (V_R)



รูปที่ 2.23 การจ่ายไบอัสให้กับอุปกรณ์เชื่อมโยงทางแสง

เนื่องจากตัวแปรเอาต์พุตทางด้านไฟฟ้ากระแสตรงและพารามิเตอร์การส่งถ่าย (Transfer Parameter) นั้นจะแตกต่างกันโดยขึ้นอยู่กับชนิดของชิ้นส่วนที่เป็นตัวตรวจรับที่ใช้ในตัวเชื่อมโยงทางแสง

ตัวเชื่อมโยงทางแสงที่ใช้โฟโต้ทรานซิสเตอร์ และโฟโต้ดาร์ลิงตัน มีหลักการทำงานเหมือนกับรอยต่อระหว่างขาคอลเลคเตอร์กับขาเบสที่ถูกทำให้กว้างขึ้น แสงที่ตกกระทบรอยต่อจะทำให้เกิดคู่ของอิเล็กตรอนและโฮลขึ้นมา เกิดการนำกระแสได้ ตัวแปรสำหรับเชื่อมโยงทางแสงชนิดโฟโต้ดาร์ลิงตันและโฟโต้ทรานซิสเตอร์ มีดังนี้

IC	: เป็นกระแสสูงสุดที่ไหลต่อเนื่องผ่านขาคอลเล็คเตอร์ (เอาต์พุต)
$V_{(BR)C\ BO}$	: เป็นแรงดันพังทลายสูงสุดจากขาคอลเล็คเตอร์ไปยังขาเบส
$V_{(BR)CEO}$	: เป็นแรงดันพังทลายสูงสุดจากขาคอลเล็คเตอร์ไปยังขาอิมิตเตอร์
$V_{(BR)ECO}$	: เป็นแรงดันพังทลายสูงสุดจากขาคอลเล็คเตอร์ไปยังขาคอลเล็คเตอร์

ตัวเชื่อมโยงทางแสงที่ใช้สวิตช์สองทิศทางทำงานเมื่อมีแสงมากระตุ้นภาคเอาต์พุตนั้นถูกออกแบบมาสำหรับใช้ในงานซึ่งต้องการแยกการทริกหรือกระตุ้นตัวไตรแอค และเอสซีอาร์ การสวิตช์ทางด้านไฟกระแสสลับที่มีขนาดกระแสต่ำและการแยกกันทางไฟฟ้ามีค่าสูง สำหรับอุปกรณ์นี้มีตัวแปรที่สำคัญ ดังนี้คือ

$IT_{(RMS)}$	: เป็นค่ากระแสอาร์เอ็มเอส RMS สูงสุดขณะที่อยู่ในสถานะที่ทำงาน (On State)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

V_{DRM} : เป็นค่าแรงดันซ้ำๆ เมื่ออยู่ในสถานะหยุดทำงาน (Repetitive Off-State Terminal Voltage)

V_{TM} : เป็นแรงดันยอดสูงสุด (Peak Voltage)

ตัวแปรของการส่งผ่านของตัวเชื่อมโยงทางแสงนั้นเป็นการจัดอัตราส่วนของการส่งกระแสระหว่างชิ้นส่วนอินพุตและเอาต์พุต สำหรับตัวเชื่อมโยงทางแสงที่ใช้อุปกรณ์โฟโต้ทรานซิสเตอร์และอุปกรณโฟโต้ดาร์ลิงตันเท่านั้น มีค่าตัวแปรที่สำคัญคือ

CTR : เป็นอัตราส่วนต่ำสุดระหว่างกระแสเอาต์พุตของคอลเล็คเตอร์สูงสุด ต่อกระแสไดโอดที่ค่า V_{CE} และ I_F ที่กำหนด

V_{CE}(Sat) : เป็นแรงดันอิ่มตัวระหว่างขาคอลเล็คเตอร์และขาอิมิตเตอร์

สำหรับตัวเชื่อมโยงทางแสงที่ใช้สวิตช์สองทางซึ่งทำงานเมื่อมีแสงมากระตุ้น และแบบที่ใช้เอสซีอาร์นั้นมีตัวแปรที่สำคัญดังนี้

IFT : เป็นค่ากระตุ้นไดโอดเปล่งแสงอินฟราเรดสูงสุด ซึ่งต้องการให้สถานะของเอาต์พุตค้างได้

IH : เป็นกระแสยึด (Holding Current) ซึ่งต้องการสำหรับเอาต์พุตเพื่อที่จะคงสถานะค้างเอาไว้

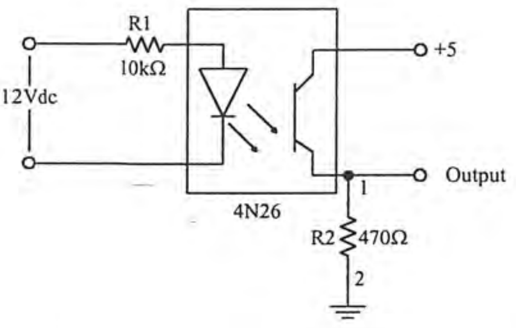
2.14.2 วงจรเปลี่ยนระดับแรงดัน

เมื่อต้องการให้วงจรทางด้านดิจิตอลรับสัญญาณอินพุตจากอุปกรณ์ภายนอก ซึ่งมีค่าแรงดันสูงจึงจำเป็นต้องเปลี่ยนระดับแรงดันของสัญญาณให้อยู่ในระดับลอจิก 5 โวลต์ ถ้าสัญญาณอินพุตเป็นสัญญาณไฟตรงก็สามารถที่จะเชื่อมต่อกับวงจรลอจิกได้โดยใช้ตัวเชื่อมโยงทางแสง โดยไม่ต้องเกี่ยวข้องกับทางกระแสไฟฟ้าระหว่างวงจรทั้งสองส่วน (วงจรทั้งสองไม่ใช้กราวด์ร่วมกัน)

ประโยชน์ของการแยกวงจรทั้งสองออกจากกันนี้ก็คือ สัญญาณรบกวนใดๆ หรือแรงดันที่เป็นยอดแหลม (Spike Voltage) ที่เกิดจากสายกราวด์ของวงจรภายนอกจะไม่เข้าไปเกิดในสายกราวด์ของวงจรลอจิกเลยนอกจากนี้ตัวเชื่อมต่อทางแสงยังสามารถใช้เพื่อเปลี่ยนสัญญาณไฟกระแสสลับ เป็นระดับสัญญาณ 5 โวลต์ ได้ ในขณะเดียวกันก็จะแยกวงจรลอจิกออกจากแรงไฟสลับที่มีค่าสูงด้วย

ในรูปที่ 2.24 แสดงถึงการใช้งานของตัวเชื่อมโยงทางแสงซึ่งเป็นอินพุตเป็นไฟตรงแรงดัน 12 โวลต์ จะถูกเปลี่ยนให้อยู่ระดับลอจิก 5 โวลต์ จะเห็นว่าในวงจรใช้ตัวเชื่อมโยงทางแสงเบอร์ 4N26 เมื่ออินพุตมีแรงดันเป็น 12 โวลต์ จะทำให้เอาต์พุตของตัวเชื่อมโยงทางแสงเป็นระดับลอจิก "1" ในขณะเดียวกันสัญญาณรบกวนที่มีลักษณะเหมือนกัน (Common-Mode Noise) จะถูกกำจัดไปโดยไดโอดที่อินพุตของตัวเชื่อมโยงทางแสง

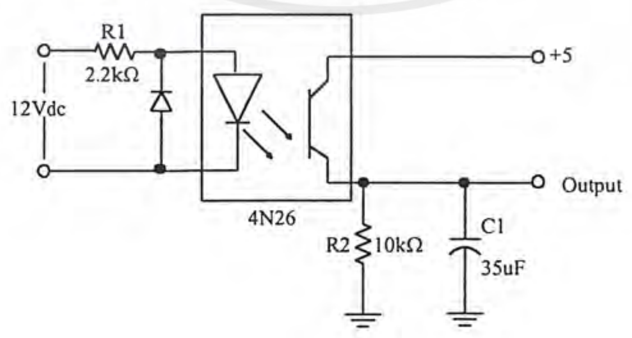
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.24 การใช้งานของตัวเชื่อมโยงทางแสงซึ่งอินพุตเป็นไฟตรงแรงดัน 12 โวลต์

เมื่อสัญญาณขนาด 12 โวลต์ ถูกป้อนเข้าสู่อินพุตของตัวเชื่อมโยงทางแสงจะเกิดกระแสไหลผ่านความต้านทาน R1 และไดโอดเปล่งแสงอินฟราเรด กระแสที่เกิดขึ้นจะไปทำให้ไดโอดเปล่งแสงอินฟราเรดสว่างขึ้น และแสงที่เกิดขึ้นจะไปตกกระทบรอยต่อระหว่างคอลเล็คเตอร์และเบสของทรานซิสเตอร์ ทำให้มันทำงานมีกระแสไหลจาก +5 โวลต์ผ่านทรานซิสเตอร์ ทำให้เกิดแรงดันตกคร่อมความต้านทาน R2 ได้เอาต์พุตอยู่ในระดับลอจิก “1” โดยที่สัญญาณเอาต์พุตนี้สามารถใช้ขับอินพุตของลอจิกเกตต่างๆ ได้

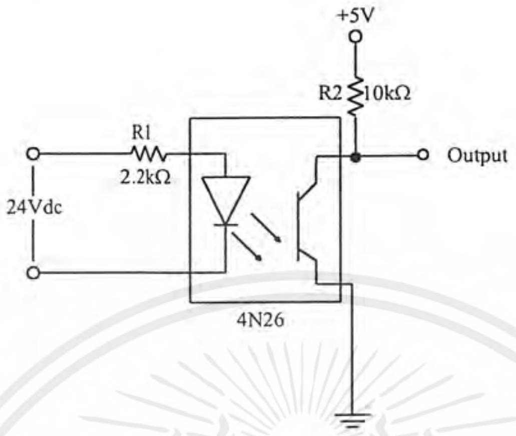
เมื่อแรงดัน 12 โวลต์ ออกจะทำให้ทรานซิสเตอร์หยุดทำงาน และความต้านทาน R2 ดึงเอาต์พุตให้มีค่าเป็น “0” วงจรในรูปที่ 2.25 จะเปลี่ยนแรงดันอินพุตขนาด 24 โวลต์ไปเป็นเอาต์พุตขนาด 5 โวลต์ ที่มีขั้วกลับกัน (Invert) ซึ่งหมายความว่า เมื่ออินพุตมีค่าสูงจะทำให้เอาต์พุตมีค่าต่ำ เมื่อป้อนสัญญาณขนาด 24 โวลต์เข้าไป จะเกิดกระแสไหลผ่านไดโอดเปล่งแสงอินฟราเรด แสงที่เปล่งออกมาจะไปตกกระทบกับโฟโต้ทรานซิสเตอร์เกิดการนำกระแสขึ้นเนื่องจากเอาต์พุตนั้นต่อมาจากขาคอลเล็คเตอร์จึงได้ค่าแรงดันระดับลอจิก “0” เมื่อต่อเข้าอินพุตของลอจิกเกท



รูปที่ 2.25 การเปลี่ยนแรงดันอินพุตขนาด 24 โวลต์ ไปเป็นเอาต์พุตขนาด 5โวลต์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เมื่อนำสัญญาณออกไปจะทำให้โฟโต้ทรานซิสเตอร์หยุดนำกระแสและความต้านทาน R2 จะดึงให้เอาต์พุตเป็นลอจิก “1”



รูปที่ 2.26 การเปลี่ยนสัญญาณไฟสลับขนาด 12 โวลต์ ให้เป็นระดับลอจิก 5 โวลต์

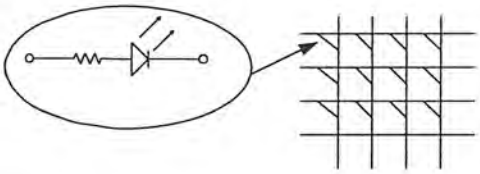
วงจรในรูปที่ 2.26 เป็นการเปลี่ยนสัญญาณไฟสลับขนาด 12 โวลต์ ให้เป็นระดับลอจิก 5 โวลต์ เมื่อป้อนสัญญาณอินพุตขนาด 12 โวลต์ เข้าไปจะทำให้กระแสไหลผ่านไดโอดเปล่งแสงอินฟราเรดในช่วงครึ่งไซเกิลบวก และผ่านไดโอด D1 ในช่วงครึ่งไซเกิลลบ นอกจากนี้ไดโอด D1 ยังทำหน้าที่จำกัดแรงดันไบอัสกลับ (VR) ไม่ให้เกิน 0.7 โวลต์อีกด้วย

ในระหว่างครึ่งไซเกิลบวกนั้นโฟโต้ดาร์ลิงตันจะนำกระแส ซึ่งจะทำให้เกิดแรงดันไฟตรงแบบกระเพื่อมตกคร่อมความต้านทาน R2 ซึ่งจะถูกกรองให้เรียบโดยใช้ตัวเก็บประจุ C1 แรงดันที่ตกคร่อมตัวเก็บประจุ C1 จะทำให้อินพุตของเกตมีค่าสูง และเมื่ออินพุตไฟสลับถูกนำออกไปจะทำให้โฟโต้ดาร์ลิงตันหยุดนำกระแส แรงดันที่ตกคร่อมตัวเก็บประจุ C1 จะตกลงไป เนื่องจากตัวเก็บประจุจะคายประจุผ่านความต้านทาน R2 ตอนนี้ความต้านทาน R2 จะดึงให้อินพุตของเกทมีค่าต่ำ

2.15 แอลอีดีแบบเมตริกซ์

จะกล่าวถึงการแสดงอย่างง่ายๆ ก่อนนั่นคือ การแสดงแบบแถวเดียว โดยเราจะควบคุมสัญญาณเพียงด้านเดียว คือ ส่งข้อมูลแล้วหน่วงไว้ระยะหนึ่งแล้วจึงส่งสัญญาณตัวต่อไปก็จะทำให้เราเห็นเป็นไฟวิ่งได้จากนั้นเราสามารถให้เห็นเป็นตัวอักษรหรือรูปภาพได้ ถ้าเราเพิ่มลอจิกการควบคุมให้กับมันซึ่งแทนที่จะควบคุมการติดดับเพียงด้านเดียวก็เป็น 2 ด้านดังรูปที่ 2.27

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.27 การต่อไดโอดเปล่งแสงแบบเมตริกซ์

จากรูปที่ 2.27 ถ้าเราส่งข้อมูล “1” ออกไปทางแถวและให้คอลัมน์เป็น “0” ไดโอดเปล่งแสงจะถูกไบอัสตรง และติดสว่างตามบิตข้อมูลที่ส่งออกไปทางแถว จากหลักการนี้เราสามารถควบคุมไดโอดเปล่งแสงทุกดวง ในแผงเมตริกซ์ให้ติดหรือดับได้ตามต้องการ

2.15.1 หลักการเกิดภาพบนแอลอีดีแบบเมตริกซ์

เนื่องจากการมองเห็นของนัยน์ตาของมนุษย์นั้นจะมีการคงภาพคือเห็นภาพนั้นอยู่นาน 1/6 วินาที ถึงแม้ว่าภาพนั้นจะไม่ปรากฏแล้วก็ตาม จากหลักการของการมองเห็นนี้ ถ้าเราสามารถทำให้ไดโอดเปล่งแสงติด และดับตามจุดต่างๆได้เร็วกว่า 1/6 เราจะเห็นว่าไดโอดเปล่งแสงที่ตำแหน่งนั้นติดค้างอยู่

แต่เนื่องจากซีพียูทำงานด้วยความเร็วสูงการที่จะให้ไดโอดเปล่งแสงติดและเปลี่ยนตำแหน่งการติดว่างไปเรื่อยๆ ตามความเร็วของซีพียูแล้วจะทำให้ไม่ปรากฏภาพ แต่จะเห็นว่าไดโอดเปล่งแสงนั้นติดสว่างเรืองๆทุกดวง เหตุที่เป็นเช่นนี้ก็เพราะว่า ความสัมพันธ์ของกระแสความอิ่มตัวของทรานซิสเตอร์ รวมทั้งกระแสที่ทำให้ไดโอดเปล่งแสงติดสว่าง เป็นปัจจัยทำให้เราเห็นไดโอดเปล่งแสงติดทุกดวง ด้วยเหตุนี้เมื่อทำให้ไดโอดเปล่งแสงดวงใดติดแล้วควรทำการหน่วงเวลา เพื่อให้มีกระแสเพียงพอที่จะทำให้ไดโอดเปล่งแสงติดสว่าง ยิ่งหน่วงเวลานานเท่าไรยิ่งทำให้กระแสเข้าใกล้จุดสูงสุด แต่เนื่องจากเราต้องทำการกวาดหลายๆ คอลัมน์ถ้าทำการหน่วงเวลามากเกินไปจะทำให้ไดโอดเปล่งแสงในลำดับถัดไปมีวงรอบในการนำกระแสเป็นช่วงๆ ซึ่งระยะเวลาในการนำกระแสของไดโอดเปล่งแสงแต่ละดวงห่างไกลจากความคงแสงของตามนุษย์ ซึ่งมีผลทำให้เป็นภาพกระพริบหรือพริ้ว ในการกวาดแบบคอลัมน์นี้จะเห็นว่าเวลาที่จะทำให้ ไดโอดเปล่งแสงแต่ละดวงติดนานเท่าไรนั้นขึ้นอยู่กับจำนวนคอลัมน์ของไดโอดเปล่งแสงด้วย

2.15.2 ปัญหาในการกวาดทางคอลัมน์

ถ้าจำนวนคอลัมน์มีมาก เวลาในการนำกระแสของไดโอดเปล่งแสง แต่ละคอลัมน์ก็มีน้อยตามไปด้วยทำให้เกิดการพริ้ว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ดังนั้นสามารถแก้ไขได้โดยเมื่อจำนวนคอลัมน์มากจนเกินไปเราควรเปลี่ยนมาเป็นการกวาดทางแถวแทนซึ่งจะทำให้เวลาของไดโอดเปล่งแสงแต่ละดวงที่จะติดมีเวลาคงที่มากยิ่งขึ้น และความสว่างคงที่มากขึ้น

2.15.3 การออกแบบตัวอักษร

การคิดข้อมูลที่จะนำแสดงเป็นตัวอักษร มิได้ถูกกำหนดเป็นมาตราฐานหรือเป็นทฤษฎีแต่ขึ้นอยู่กับฮาร์ดแวร์นั้นๆ เมื่อศึกษาวงแสดงขนาด 8X8 ตัว จะพบว่ากำหนดให้พอร์ต 1 ของ AT89S8252 เป็นพอร์ตสำหรับจ่ายแหล่งจ่ายกระแส (Source Current ผ่าน 74LS245) และทางพอร์ตที่เป็นตัวรับกระแสซิงค์ (Sink Current ผ่านทรานซิสเตอร์)

ฉะนั้นข้อมูลที่ทำให้ไดโอดเปล่งแสงติดได้ คือ ข้อมูลที่ออกมาจากพอร์ต 1 ซึ่งเป็นลอจิกสูง และข้อมูลพอร์ต 2 ต้องเป็นลอจิกต่ำ แต่เนื่องจากพอร์ต 2 มีทรานซิสเตอร์อยู่ 1 ตัว เป็นตัวรับกระแสซิงค์จึงต้องคิดข้อมูลเป็นลอจิกบวก คือ “1” (ไดโอดเปล่งแสงติด)

2.16 แป้นพิมพ์

แป้นพิมพ์นับเป็นสิ่งสำคัญอย่างหนึ่งเพราะแป้นพิมพ์เป็นอุปกรณ์อินพุตที่ทำให้ผู้ใช้สามารถติดต่อกับเครื่องควบคุมในการทำงานต่างๆได้ โดยแป้นพิมพ์จะประกอบด้วยส่วนสำคัญ 3 ส่วน คือ

1) ส่วนของสวิตช์แป้นพิมพ์

2) ส่วนของวงจรเข้ารหัส โดยส่วนนี้จะทำหน้าที่เข้ารหัสเพื่อรับรู้ตำแหน่งตัวอักษรบนแป้นและสามารถให้ค่าอักษรบนแป้นของแต่ละตัวเมื่อมีการกดตัวอักษรบนแป้น ที่เรียกว่ารหัสตัวอักษรบนแป้น (Keycode) ได้

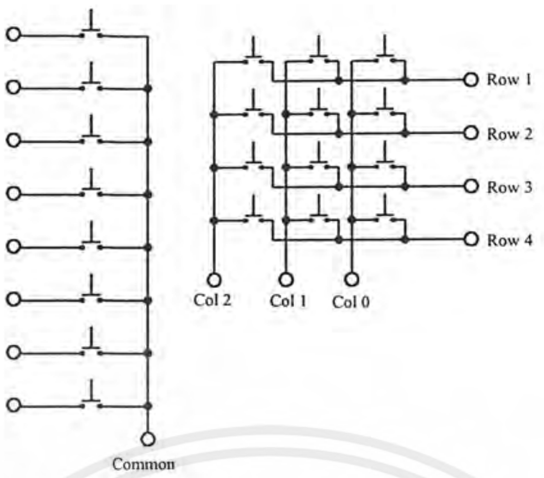
3) ส่วนวงจรถอดรหัสจะเป็นส่วนที่ทำหน้าที่เปลี่ยนรหัสตัวอักษรบนแป้น ให้รหัสที่นำไปใช้งานได้เช่นรหัส ASCII, รหัส BCD, รหัส HEX เป็นต้น

วงจรแป้นพิมพ์ แบ่งได้เป็น 2 ลักษณะใหญ่ๆ คือ

1) แบบขาร่วม

2) แบบเมตริกซ์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

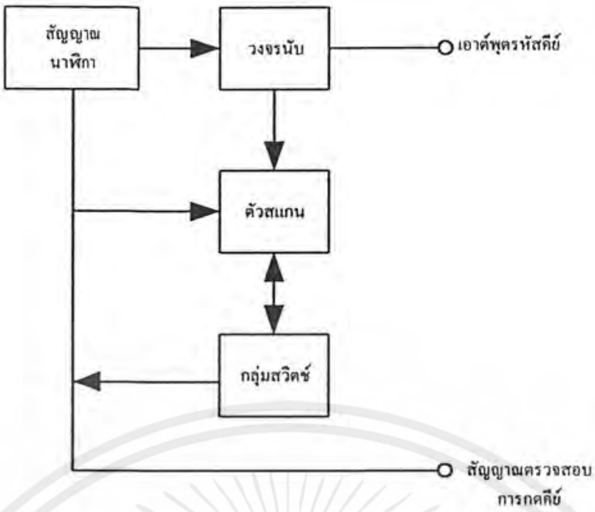


รูปที่ 2.28 วงจรแป้นพิมพ์แบบขาร่วมและแบบเมตริกซ์

จะเห็นว่าแบบขาร่วมจะเป็นแบบที่เข้าใจง่ายและการรับรู้การกดตัวอักษรบนแป้นจะเป็นไปในลักษณะเฉพาะตัวของแต่ละตัวอักษรบนแป้นซึ่งทำให้ง่ายต่อการออกแบบสะดวก และประหยัดในกรณีที่ใช้สวิตช์ไม่มากนัก ส่วนแบบเมตริกซ์ จะใช้สายในการต่อเข้ากับวงจรเข้ารหัสได้น้อยกว่า โดยให้ระบุจำนวนจุดของตัวอักษรบนแป้น ได้จำนวนมากซึ่งจำนวนตัวอักษรบนแป้น จะขึ้นอยู่กับสายทางด้านแถวและสายทางด้านคอลัมน์ แต่การออกแบบวงจรเข้ารหัสจะยุ่งยากขึ้นโดยสามารถแสดงผังการทำงานของวงจรเข้ารหัสแป้นพิมพ์ไว้ดังรูป 2.29

สัญญาณนาฬิกาจะใช้ตรวจว่าตัวอักษรบนแป้นใดถูกกด โดยเมื่อไม่มีการกดตัวอักษรบนแป้น ตัวกวาดจะส่งสัญญาณกวาดไปที่สวิตช์ และเมื่อมีการกดตัวอักษรบนแป้น สัญญาณนาฬิกาจะหยุดการทำงาน ทำให้ตัวกวาดหยุดการกวาด วงจรนับจะหยุดนับและคงค่าสุดท้ายเอาไว้ (รหัสตัวอักษรบนแป้นที่ถูกกด) จนกว่าจะมีการปล่อยตัวอักษรบนแป้น ถึงจะรับค่าแป้นพิมพ์ตัวใหม่ได้ ซึ่งลักษณะการกวาดแบบเมตริกซ์นี้ จะเริ่มกวาดจากแถวแรกไปยังแถวสุดท้ายและคอลัมน์แรกไปยังคอลัมน์สุดท้ายพร้อมกัน ดังนั้นเมื่ออักษรบนแป้นใดถูกกดก็จะทำให้ทาบตัวอักษรบนแป้นที่ถูกกดว่าอยู่แถวและคอลัมน์ใดได้จากส่วนของวงจรนับ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.29 ผังการทำงานของวงจรเข้ารหัสแป้นพิมพ์

2.16.1 การกระเด้งของหน้าสัมผัส

เนื่องจากเมื่อมีการกดตัวอักษรบนแป้นพิมพ์สวิตช์ก็จะทำให้หน้าสัมผัสต่อ แต่การต่อของหน้าสัมผัสนี้จะไม่เป็นไปตามอุดมคติ ด้วยเหตุว่าหน้าสัมผัสจริงๆไม่ได้ราบเรียบหรือสะอาดอยู่เสมอ และแรงกดของนิ้วบนแป้นกดสวิตช์ ก็ยังแตกต่างกันไป ทำให้เกิดการสะบัดหรือแกว่งของแป้นกดชั่วระยะหนึ่งหลังจากกดสวิตช์ลงไปแล้วเรียกว่าเป็นการกระเด้งของหน้าสัมผัส จึงทำให้ตัวตรวจจับตัวอักษรบนแป้นที่กดตรวจว่าตัวอักษรบนแป้นดังกล่าวมีการกดหลายครั้ง ทั้งๆที่ความจริงผู้ใช้ต้องการกดเพียงครั้งเดียวซึ่งช่วงการเกิดของสัญญาณที่ไม่ต้องการนี้จะมีช่วงเวลาประมาณ 10 มิลลิวินาที เราสามารถแก้การดีเบาซ์ ของสวิตช์ได้หลายวิธี

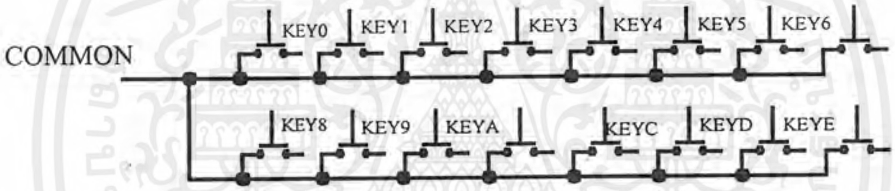
ในแบบอาร์เอสแลต ขณะสวิตช์ไม่ถูกเปลี่ยนทิศทาง เอาต์พุตแนนด์เกทตัวบนจะมีลอจิก “1” เนื่องจากมีขาอินพุตใดอินพุตหนึ่งเป็น “0” ก็จะทำให้อินพุตของแนนด์เกทตัวล่างเป็น “1” ทั้งคู่ ดังนั้นเอาต์พุตดีเบาซ์ก็จะมีลอจิกเป็น “0” แต่เมื่อมีการเปลี่ยนทิศทางสวิตช์ ทำให้หน้าสัมผัสสวิตช์ด้านแนนด์เกตบนตัวลอย ส่วนหน้าสัมผัสสวิตช์แนนด์เกท ตัวล่างจะมีการต่อและจะมีการกระเด้งที่หน้าสัมผัสนี้เกิดขึ้น แต่ช่วงขณะที่แนนด์เกตตัวล่างได้รับอินพุตเป็น “0” ครั้งแรก เอาต์พุตดีเบาซ์ก็จะเป็น “1” และเป็นผลให้แนนด์เกตตัวบนมีอินพุตเป็น “1” ทั้งคู่ทำให้เอาต์พุตตัวบนเป็น “0” และเป็นผลทำแนนด์ตัวล่างมีอินพุตค้างเป็น “0” อยู่ด้านหนึ่งเนื่องจากเอาต์พุตแนนด์เกตตัวบน ดังนั้นแม้สวิตช์ทางด้านแนนด์เกตตัวล่างจะกระเด้ง อย่างไรก็ตามเอาต์พุตที่ได้ก็จะไม่เปลี่ยนแปลงเพราะอินพุตเป็น “0” ค้างอยู่ด้านหนึ่งแต่วิธีนี้จำเป็นต้องใช้สวิตช์เป็นแบบ 2 ทาง จึงจะแก้การกระเด้งได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ดังนั้นถ้าสวิตช์เราเป็นแบบทิศทางเดียวหรือแบบกดติดปล่อยดับแล้ว การแก้การกระเด้งขึ้นอยู่กับค่าความต้านทานและตัวเก็บประจุ เช่น เมื่อเรากดสวิตช์ จะทำให้ตัวเก็บประจุถูกคายประจุ ผ่าน R1 จนถึงระดับหนึ่ง ถึงจะทำให้เอาต์พุตของชมิตทริกเกอร์ เป็นลอจิก "1" คือจะไม่เป็น "1" ทันทีทันใด ซึ่งช่วงที่เป็น "1" นี้ก็คือช่วงที่เลยเวลาของการกระเด้งไปแล้ว และในขณะสวิตช์เปิดก็จะไม่เป็น "0" ทันทีเพราะจะมีการเก็บประจุของตัวเก็บประจุผ่าน R2 จนถึงช่วงเวลาหนึ่งถึงจะได้ลอจิก "0" เพราะคุณสมบัติของชมิตทริกเกอร์ด้วย

2.16.2 องค์ประกอบอันเกิดจากสวิตช์ตัวอักษรบนแป้น

ตัวอักษรบนแป้นโดยทั่วไปหน้าสัมผัสจะมีการสั่นทั้งขณะเปิดและขณะปิด ซึ่งมีเวลาเป็นมิลลิวินาที เมื่อมีการกดปล่อยตัวอักษรบนแป้นอย่างรวดเร็วพัลซ์ที่เกิดขึ้น จะต้องตรวจสอบได้ด้วยไมโครคอนโทรลเลอร์ ตัวอักษรบนแป้นอาจทำงานเพียงสัมผัส หรือสร้างสัญญาณการกดตัวอักษรบนแป้นด้วยอาร์เอสฟลิปฟลอบหรือซอฟแวร์



รูปที่ 2.30 ลักษณะแป้นพิมพ์ไฮโพธีติคอล (Hypothetical)

2.16.3 โปรแกรมสำหรับแป้นพิมพ์

โปรแกรมที่ดีจะต้องตอบสนองการทำงานของมนุษย์ผ่านทางตัวอักษรบนแป้นบอร์ด โดยทำตามนี้

1) การกดปล่อยตัวอักษรบนแป้น เวลาที่ใช้หน่วงจะมีระยะเวลานานกว่า รายละเอียดจากโรงงานที่ผลิตเพื่อรอคาบการกดปล่อยทั้งสองทาง

2) สำหรับตัวอักษรบนแป้นหลายๆ ตัว ตัวอักษรบนแป้นที่ได้รับการกดก่อนเท่านั้นจะได้รับการอ่านค่าตัวอักษรบนแป้น

3) การค้างของตัวอักษรบนแป้น ตัวอักษรบนแป้นที่กดก่อนจะได้รับการอ่านค่า จบเวลาการกดตัวอักษรบนแป้นจะไม่มีตัวอักษรบนแป้นใดได้รับการอ่านค่าจนกว่าจะพบว่าตัวอักษรบนแป้นทั้งหมดกลับสู่สถานะเดิม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4) การกดปล่อยตัวอักษรบนแป้นอย่างรวดเร็ว ในการออกแบบตัวอักษรบนแป้นจะได้รับการกวาดด้วยอัตราที่สูงกว่าปฏิกิริยาของการกระทำของมนุษย์

ลักษณะโปรแกรมที่สำหรับอ่านค่าตัวอักษรบนแป้นมีอยู่ 2 ลักษณะ คือ

1) การหน่วงเวลาโดยใช้โปรแกรมลูปเพื่อรอรับค่าตัวอักษรบนแป้น

2) การอ่านจะมีขึ้นก็ต่อเมื่อมีการกดตัวอักษรบนแป้นเท่านั้น (Interrupt driven)

2.16.4 การอ่านข้อมูลจากสวิตช์จำนวนมาก

การต่อสวิตช์แบบปกติจะใช้ 1 บิตต่อสวิตช์ 1 ตัว แต่ในการใช้งานจริงของระบบคอมพิวเตอร์ต้องการใช้สวิตช์หรือตัวอักษรบนแป้นเป็นจำนวนมาก ดังนั้นถ้าใช้ 1 สวิตช์ต่อ 1 บิต จะทำให้สิ้นเปลืองพอร์ตเป็นจำนวนมาก เช่นถ้าต้องการใช้ 64 สวิตช์ จะต้องใช้พอร์ตแบบขนาน 8 บิตถึง 8 พอร์ต วิธีการต่อสวิตช์แบบที่ประหยัดพอร์ตและเป็นที่นิยมใช้แบบหนึ่ง คือ การต่อแบบเมตริกซ์ ซึ่งจะทำให้ลดจำนวนพอร์ตลง ถ้าพิจารณาจากจำนวนของสวิตช์ และจำนวนของสายเมื่อต่อแบบธรรมดา และจำนวนของสายเมื่อต่อแบบสายเมื่อต่อแบบเมตริกซ์ จะเป็นดังตารางที่ 2.7

ตารางที่ 2.7 เปรียบเทียบจำนวนของสายเมื่อต่อแบบธรรมดาและเมตริกซ์

จำนวนสวิตช์	จำนวนสาย	
	ต่อแบบธรรมดา	ต่อแบบเมตริกซ์
1X1	1	1
2X2	4	4
3X3	9	6
4X4	16	8
5X5	25	10
8X8	64	16

จากตารางที่ 2.7 เห็นได้ชัดว่า เมื่อจำนวนของสวิตช์มากขึ้นก็จะทำให้ประหยัดสายข้อมูลมากขึ้นไปด้วย ข้อยุ่งยากทางด้านฮาร์ดแวร์จะลดลง และการตรวจสอบสวิตช์ที่กดจะใช้วิธีการทางซอฟแวร์ช่วยจัดการ

ลักษณะการทำงานของวงจรแป้นพิมพ์ของวงจรแป้นพิมพ์ขนาด 3X3 ตัว

หลักการทำงานของโปรแกรมจะใช้วิธีการกวาด เพื่อหาว่าสวิตช์ใดถูกกดโดยอาจเริ่มต้นที่แถวที่ 0 โดยกำหนดให้แถวที่ 0 มีระดับเป็น 0 ส่วนแถวอื่นๆ มีระดับเป็น 1 แล้วทำการตรวจสอบ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ทีละคอลัมน์ โดยเริ่มจากคอลัมน์ที่ 0 ตรวจสอบว่ามีระดับเป็น 0 หรือไม่ถ้ามีระดับเป็น 0 แสดงว่าสวิตช์ที่ตำแหน่งนี้ถูกกด ถ้าไม่มีก็เลื่อนไปคอลัมน์ถัดไป และจะทำการเลื่อนและตรวจสอบทั้งทางด้านคอลัมน์และแถวต่อๆ ไปจนกว่าจะหมดหรือจนกระทั่งพบสวิตช์ที่ถูกกด และเมื่อพบว่าสวิตช์ใดถูกกดแล้วต้องทราบว่าสวิตช์นั้นคือสวิตช์ตำแหน่งใดเพื่อที่จะได้ตีความหมาย หรือสร้างรหัสขึ้นมาตามสวิตช์ตำแหน่งนั้นๆ

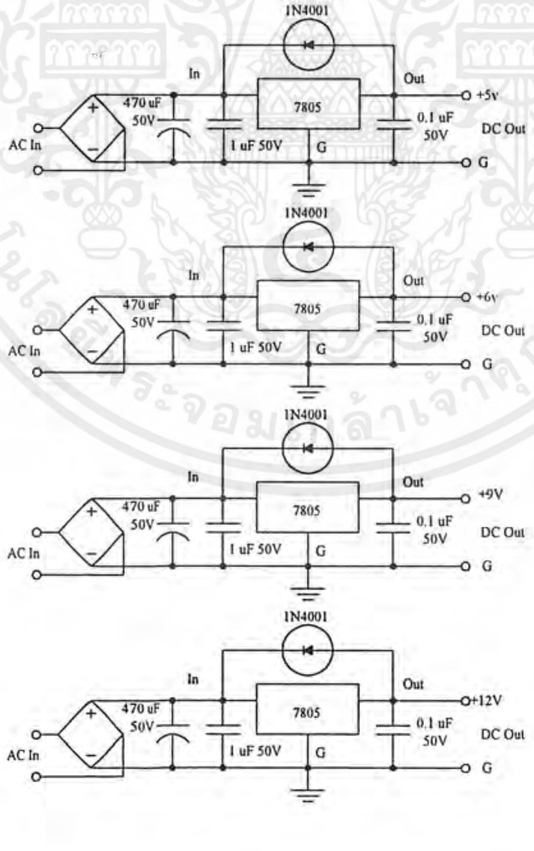
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 3
การออกแบบ การสร้าง และการทำงาน

ในการออกแบบและสร้างชุดทดลองการเชื่อมต่อไมโครคอนโทรลเลอร์ MCS-51 กับอุปกรณ์ภายนอก จะแบ่งออกเป็น 3 ส่วน คือ ส่วนแหล่งกำเนิดแรงดัน ส่วนวงจรควบคุม และส่วนวงจรอินเตอร์เฟส โดยอธิบายเป็นส่วนได้ดังนี้

3.1 วงจรแหล่งกำเนิดแรงดัน

ในชุดทดลองมีแหล่งกำเนิดแรงดันอยู่ 4 ชุด มีขนาดแรงดัน 5 โวลต์, 6 โวลต์, 9 โวลต์ และ 12 โวลต์ โดยส่วนของแรงดัน 5 โวลต์ จะจ่ายให้วงจรรวมทั้งหมด ส่วนแรงดัน 6 โวลต์, 9 โวลต์ และ 12 โวลต์ จะป้อนให้กับอุปกรณ์ภายนอก เช่น สเต็ปปิ้งมอเตอร์ อุปกรณ์รีเลย์ เป็นต้น



รูปที่ 3.1 วงจรแหล่งกำเนิดแรงดัน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.1.1 หลักการทำงาน

วงจรแหล่งกำเนิดแรงดันจะรับค่าแรงดันไฟฟ้ากระแสสลับ 220 โวลต์ ผ่านหม้อแปลงขนาด 3 แอมแปร์ มี 4 ชุดผ่านวงจรเร็กติไฟเออร์ โดยชุดที่หนึ่ง ผ่านไอซีเบอร์ 7805 เอาต์พุตที่ได้มีค่า 5 โวลต์ ชุดที่สอง ผ่านไอซีเบอร์ 7806 เอาต์พุตที่ได้มีค่า 6 โวลต์ ชุดที่สาม ผ่านไอซีเบอร์ 7809 เอาต์พุตที่ได้มีค่า 9โวลต์ ชุดที่สี่ ผ่านไอซีเบอร์ 7812 เอาต์พุตที่ได้มีค่า 12 โวลต์

3.2 การออกแบบวงจรควบคุม

วงจรควบคุม เป็นวงจรที่ใช้ควบคุมการทำงานทั้งหมดของชุดทดลองกับคอมพิวเตอร์ผ่านพอร์ตอนุกรม และควบคุมการทำงานของวงจรอินเตอร์เฟสทุกวงจร หน้าที่หลักของวงจรควบคุม คือ ทำหน้าที่บันทึกโปรแกรมภาษาเครื่องลงบนไมโครคอนโทรลเลอร์ และประมวลผลคำสั่งตามโปรแกรมที่เขียนขึ้น โดยมีการส่งผ่านข้อมูลแบบ SPI โดยใช้ไมโครคอนโทรลเลอร์ AT89S8252 เป็นตัวควบคุมการทำงาน มีหน่วยความจำโปรแกรมภายใน 8 กิโลไบต์ มีหน่วยความจำข้อมูล 256 ไบต์ หน่วยความจำอีอีพรอม 2 กิโลไบต์ ในวงจรใช้ไอซี PC817 ทำหน้าที่รีเซตวงจร

3.2.1 หลักการทำงาน

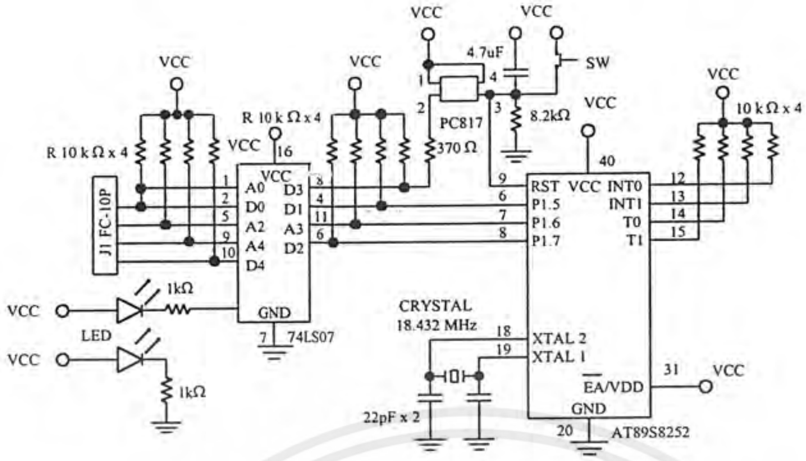
หน่วยความจำข้อมูลสามารถที่จะโปรแกรมโดยผ่านเส้นทางข้อมูลอนุกรม SPI ซึ่ง RST จะป้อนแรงดันสูงสุด ในการอินเตอร์เฟสแบบอนุกรมจะประกอบไปด้วยขา SCK, MOSI (Input) และ MISO (Output) หลังจาก RST มีค่าเป็น 1 ก็จะสามารถทำการโปรแกรมที่ต้องการได้ ก่อนที่จะทำการโปรแกรมต้องทำการลบข้อมูลภายในก่อน ซึ่งบอร์ดที่ใช้เป็นการลบข้อมูลโดยอัตโนมัติ

ตำแหน่งหน่วยความจำ 0000H-1FFFH ใช้สำหรับหน่วยความจำโปรแกรม

ตำแหน่งหน่วยความจำ 000H-7FFH ใช้สำหรับหน่วยความจำข้อมูล

วงจรอินเตอร์เฟสของชุดทดลองนี้จะมีทั้งหมด 14 วงจรซึ่งจะได้กล่าวถึงรายละเอียดของการออกแบบและการทำงานของแต่ละวงจรต่อไป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

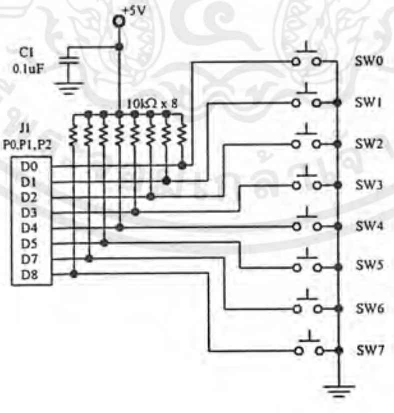


รูปที่ 3.2 การออกแบบวงจรควบคุม

3.3 การออกแบบวงจรอินเตอร์เฟส

3.3.1 การออกแบบวงจรสวิตช์เดี่ยว

การออกแบบวงจรจะใช้สวิตช์เดี่ยวทั้งหมด 8 ตัวเป็นอินพุตให้กับ MCS-51 การใช้งานนั้นจะทำการต่อเข้ากับพอร์ตของ MCS-51 โดยตรง จะขึ้นอยู่กับการเลือกใช้งาน จะมีวงจรการออกแบบแสดงได้ดังรูปที่ 3.3

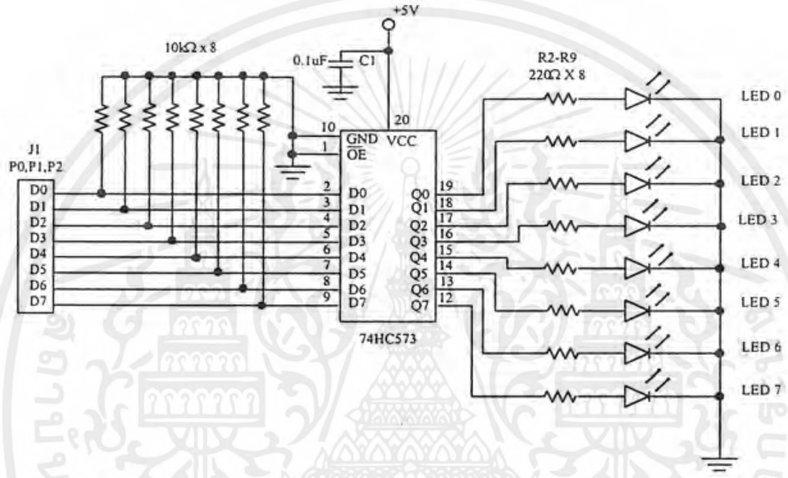


รูปที่ 3.3 การออกแบบวงจรสวิตช์เดี่ยว

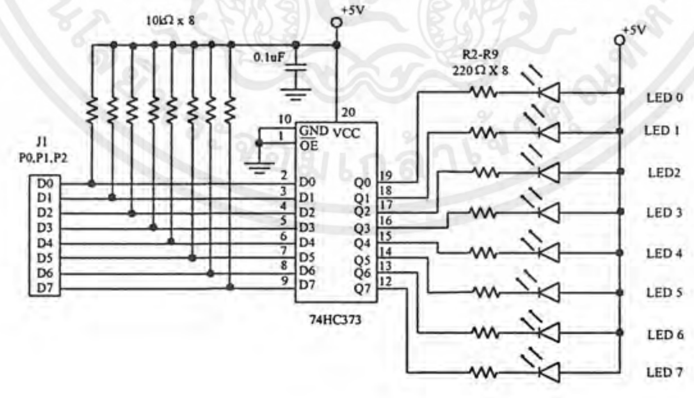
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.3.2 การออกแบบวงจรไดโอดเปล่งแสง

วงจรนี้จะมีสวิตช์เป็นตัวอินพุต เพื่อส่งข้อมูลขนาด 8 บิตไปยังพอร์ตของ AT89S8252 ซึ่งจะส่งข้อมูลผ่าน 74HC573 ที่ทำหน้าที่เป็นบัฟเฟอร์ ที่ผ่านการควบคุมจากการถอดรหัส (Decode) การออกแบบวงจรไดโอดเปล่งแสงทำการออกแบบอยู่ 2 วงจร โดยแบ่งเป็นวงจรไดโอดเปล่งแสงแบบคาโถดร่วม และวงจรไดโอดเปล่งแสงแบบอาโนดร่วม การออกแบบวงจรแสดงได้ดังรูป 3.4 และรูปที่ 3.5



รูปที่ 3.4 การออกแบบวงจรไดโอดเปล่งแสงแบบคาโถดร่วม

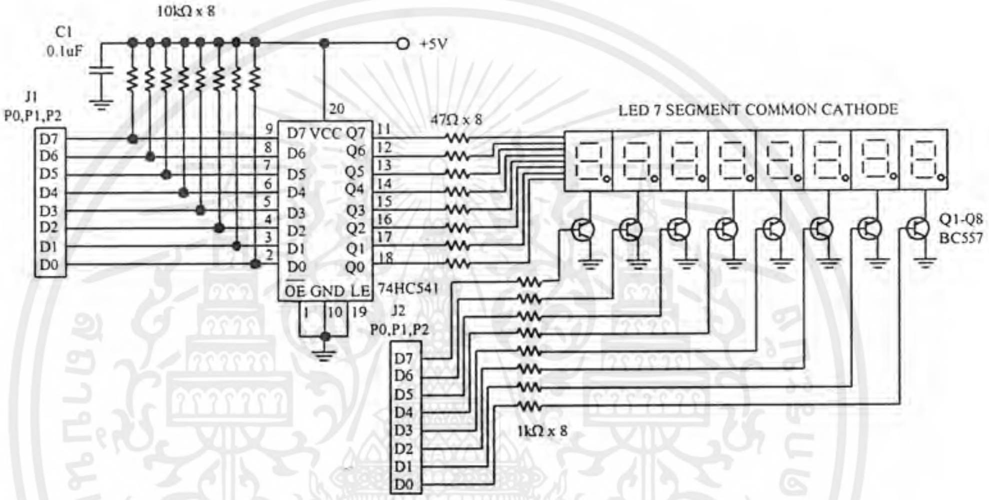


รูปที่ 3.5 การออกแบบวงจรไดโอดเปล่งแสงแบบอาโนดร่วม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.3.3 การออกแบบวงจรแสดงผลแบบ 7 ส่วน

การแสดงผลแบบ 7 ส่วน เป็นตัวแสดงผลอีกแบบหนึ่งที่แสดงผลออกมาเป็นตัวเลข ภายในประกอบด้วยแอลอีดี 8 ดวง เป็นส่วน a, b, c, d, e, f, g และ จุดทศนิยม (dp) การแสดงผลของวงจรแสดงผลแบบ 7 ส่วน จำเป็นต้องอาศัยการเขียนโปรแกรมจากไมโครคอนโทรลเลอร์ MCS-51 จะจ่ายไฟให้แก่ขาร่วมของแอลอีดีแบบ 7 ส่วนทีละหลักโดยผ่านทรานซิสเตอร์ การออกแบบวงจรแสดงได้ดังรูปที่ 3.6



รูปที่ 3.6 การออกแบบวงจรแสดงผลแบบ 7 ส่วน

3.3.4 การออกแบบวงจรแสดงผลแบบผลึกเหลว

วงจรแสดงผลแบบผลึกเหลว จะออกแบบให้ต่อแบบเอาต์พุตซึ่งมีสายสัญญาณข้อมูล 10 เส้นจากพอร์ต โดยอาศัยไอซีส่งผ่านข้อมูล 74HC541 โดย 8 เส้นเป็นข้อมูล และอีก 2 เส้นเป็นขา E และ RS

จังหวะการทำงานของแอลซีดีโมดูล

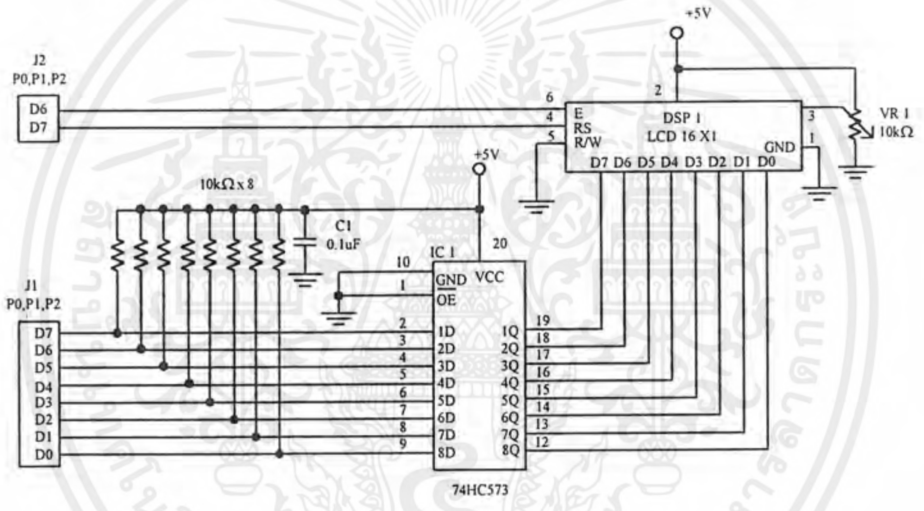
ในการติดต่อกับแอลซีดีโมดูล จะต้องมีการหน่วงเวลาหลังจากที่ทำการส่งรหัสคำสั่ง หรือข้อมูลเนื่องจากต้องรอให้คอนโทรลเลอร์ภายในแอลซีดีโมดูล แปลความหมายของรหัสคำสั่ง และทำงานตามคำสั่งให้เรียบร้อยก่อน จากนั้นจึงจะรับข้อมูลหรือดำเนินการต่อไป

ดังนั้น ในการใช้งานแอลซีดีโมดูล ผู้เขียนโปรแกรมต้องมีโปรแกรมเพื่อหน่วงเวลารอให้แอลซีดีโมดูล พร้อมทำงาน โดยเมื่อเริ่มจ่ายไฟให้แก่แอลซีดีโมดูล ต้องรอประมาณ 10 มิลลิวินาที เพื่อให้แอลซีดีโมดูล ทำการเตรียมความพร้อมหรือ อินิเชียล (initial) หลังจากนั้นก็จะกำหนดลอจิก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ให้กับขา RS ของแอลซีดีโมดูล แล้วต้องหน่วงเวลาประมาณ 2 มิลลิวินาที เพื่อให้คอนโทรลเลอร์ในแอลซีดีโมดูล แปลความหมายของลอจิกที่ขา RS ว่า ข้อมูลต่อไปที่จะได้รับนั้นเป็นรหัสคำสั่ง หรือเป็นข้อมูลที่ต้องการแสดงผล จากนั้นจะเป็นการส่งข้อมูลมารอที่บัสข้อมูล D0 – D7 (กรณีทำงานในโหมด 8 บิต) ขั้นตอนต่อไป จะเป็นการส่งสัญญาณพัลส์ไปที่ขา E เพื่ออีนาเบิลแอลซีดีโมดูล ให้รับข้อมูลจากบัสข้อมูลเข้าไป โดยพัลส์ที่ป้อนเข้าที่ขา E ของแอลซีดีโมดูล ต้องเป็นพัลส์ขอบขาขึ้น จากนั้นทำการหน่วงเวลา 2 มิลลิวินาที

วงจรนี้จะใช้การเขียนโปรแกรมในการสร้างสัญญาณให้แสดงผลออกทางหน้าจอแสดงผลแบบผลึกเหลว ซึ่งสามารถแสดงผลได้ครั้งละ 1 บรรทัด จำนวน 16 ตัวอักษร การออกแบบวงจรแสดงได้ดังรูปที่ 3.7

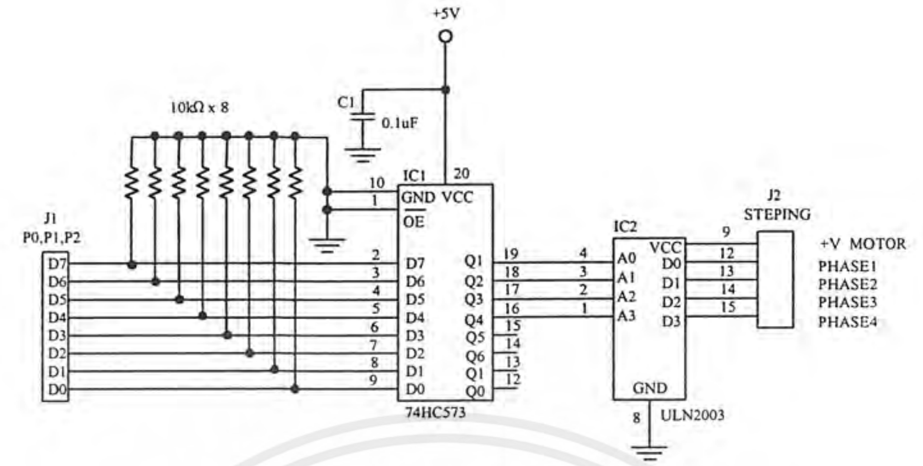


รูปที่ 3.7 การออกแบบวงจรแสดงผลแบบผลึกเหลว

3.3.5 การออกแบบวงจรควบคุมสเต็ปปิ้งมอเตอร์

การออกแบบวงจรขับสเต็ปปิ้งมอเตอร์ จะใช้การส่งผ่านข้อมูล ผ่านไอซีแลตช์ 74HC573 แล้วส่งต่อไปยังไอซีไดร์เวอร์กระแสสูง แบบคอลเล็คเตอร์เปิดเบอร์ ULN2003 ด้วยการใช้ไอซีแบบนี้ทำให้สามารถเลือกแรงดันขับสเต็ปปิ้งมอเตอร์ได้ ตั้งแต่ 5 – 30 โวลต์ โดย ULN2003 มีความสามารถจ่ายกระแสได้สูงสุด 500 มิลลิแอมป์ ต่อขา จากวงจรจะมี 4 เฟสคือ เฟส 1 , 2 , 3 และ 4 ถ้าต้องการให้สเต็ปปิ้งมอเตอร์ขยับก็ทำการส่งข้อมูลที่เป็น “1” ให้กับวงจร การออกแบบวงจรแสดงได้ดังรูปที่ 3.8

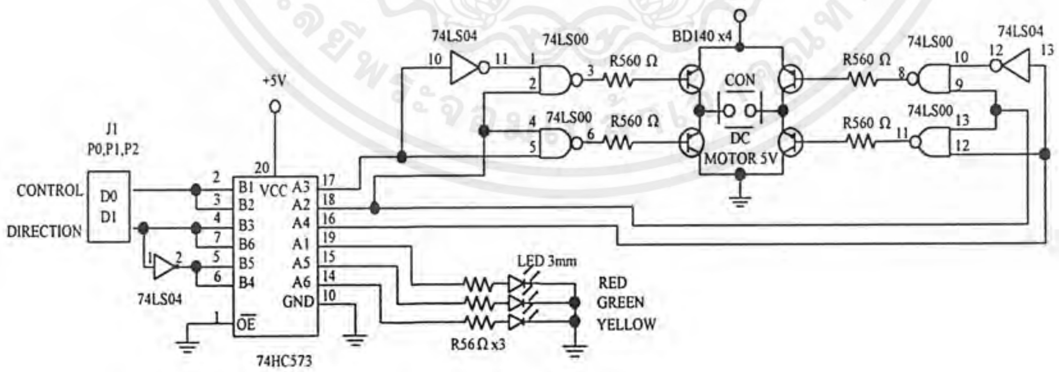
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.8 การออกแบบวงจรควบคุมสเต็ปปิ้งมอเตอร์

3.3.6 การออกแบบวงจรมอเตอร์ไฟฟ้ากระแสตรง

การควบคุมความเร็วของมอเตอร์ให้ความเร็วคงที่ตามที่เราต้องการเป็นเรื่องที่ทำได้ค่อนข้างยาก ถ้าใช้การควบคุมแบบธรรมดาทั่วไป คือ การควบคุมแรงดันที่จ่ายให้กับมอเตอร์ การจะควบคุมความเร็วของมอเตอร์ให้ได้ความเร็วคงที่ ต้องควบคุมกระแสที่จ่ายให้กับมอเตอร์ ซึ่งวิธีนี้เรียกว่า การควบคุมแบบพัลส์วิดท์โมดูเลชั่น (Pulse Width Modulation : PWM) เป็นการควบคุมกระแสที่จ่ายให้กับมอเตอร์



รูปที่ 3.9 วงจรมอเตอร์ไฟฟ้ากระแสตรง

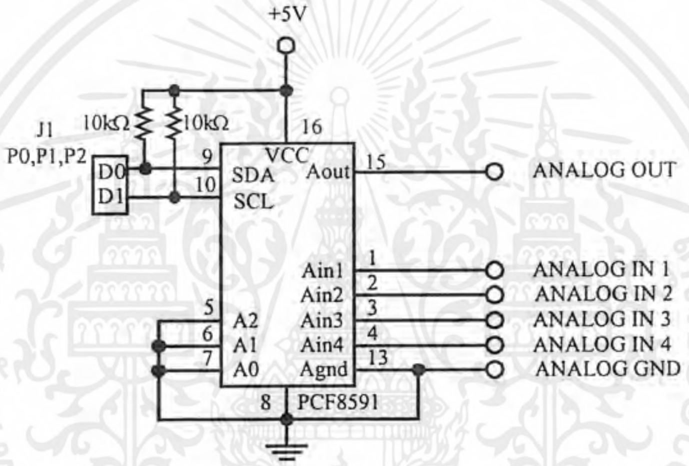
การออกแบบจะใช้ AT89S8252 ในการควบคุมการปิดเปิดวงจรขับและการควบคุมทิศทางการหมุนของมอเตอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.3.7 การออกแบบวงจรแปลงสัญญาณแอนะลอกเป็นสัญญาณดิจิตอล

คุณสมบัติของวงจรแปลงสัญญาณแอนะลอกเป็นสัญญาณดิจิตอล จะมีลักษณะเหมือนกับวงจรแปลงสัญญาณดิจิตอลเป็นสัญญาณแอนะลอกและจะมีคุณสมบัติพิเศษอีก 1 อย่าง คือ การเปลี่ยนแปลงเวลา หมายถึง ช่วงเวลาที่วงจรแปลงสัญญาณแอนะลอกเป็นสัญญาณดิจิตอล ใช้ในการแปลงสัญญาณแอนะลอกเป็นสัญญาณดิจิตอลได้ 1 ค่า

การออกแบบวงจรแปลงสัญญาณแอนะลอกเป็นสัญญาณดิจิตอลนี้ จะใช้การส่งข้อมูลผ่านไอซีเบอร์ PCF8591 การออกแบบวงจรได้ดังรูปที่ 3.10

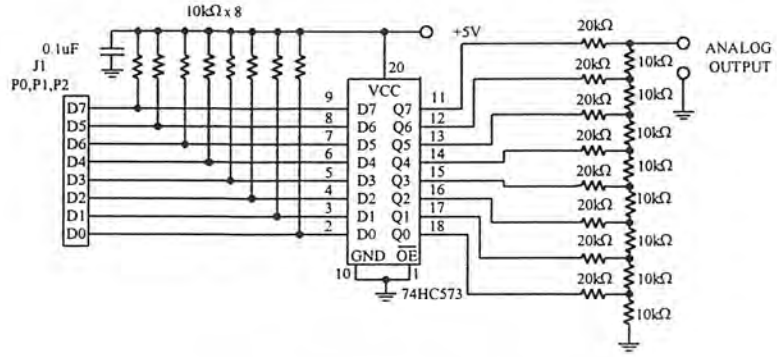


รูปที่ 3.10 การออกแบบวงจรแปลงสัญญาณแอนะลอกเป็นดิจิตอล

3.3.8 การออกแบบวงจรแปลงสัญญาณดิจิตอลเป็นสัญญาณแอนะลอก

การออกแบบวงจรแปลงสัญญาณดิจิตอลเป็นสัญญาณแอนะลอกแบบ R-2R จะใช้ตัวความต้านทานต่อขนานกันหลายสาขา กระแสในวงจรจะถูกเฉลี่ยกันในวงจรด้วยอัตราส่วนของตัวความต้านทานคงที่ คือ R และ 2R ในการใช้งานจะต่อเข้ากับพอร์ตใดก็ได้ โดยผ่านวงจรแลตช์ 74HC573 เพื่อได้ข้อมูลดิจิตอลออกไปยังวงจร ซึ่งวงจรจะแปลงสัญญาณเป็นสัญญาณแอนะลอกให้เอง การออกแบบวงจรแสดงได้ดังรูปที่ 3.11

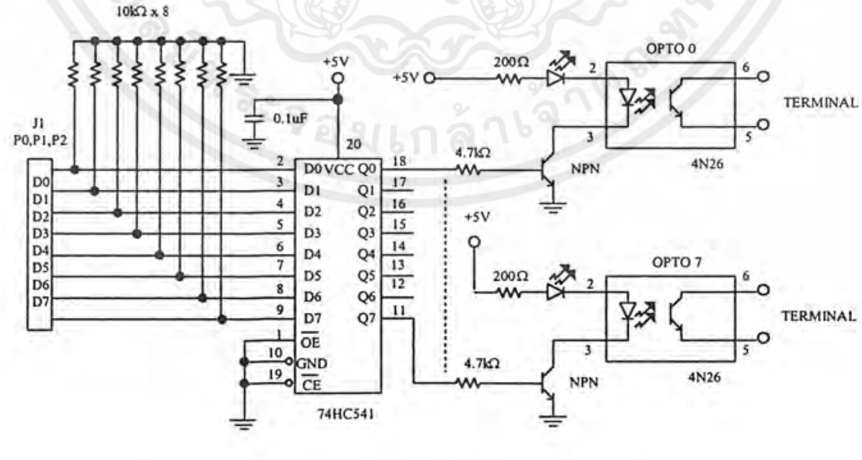
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.11 การออกแบบวงจรแปลงสัญญาณดิจิตอลเป็นสัญญาณแอนะลอก

3.3.9 การออกแบบวงจรอุปกรณ์เชื่อมโยงทางแสงออก

วงจรเชื่อมโยงทางแสงออก เป็นวงจรที่ใช้ควบคุมการทำงานของอุปกรณ์ทางไฟฟ้า เพราะอุปกรณ์เชื่อมโยงทางแสงสามารถแยกระบบไมโครคอนโทรลเลอร์กับระบบไฟฟ้า โดยอาศัยแสง ซึ่งภายในตัวของมัน จะมีหลอดไดโอดเปล่งแสงเป็นตัวเปล่งแสงอยู่ภายใน และจะมีตัวโฟโต้ทรานซิสเตอร์ทำหน้าที่เป็นสวิตช์แสง เมื่อมีแรงดันให้กับหลอดไดโอดเปล่งแสง แสงที่เกิดขึ้นจากไดโอดเปล่งแสงจะทำให้ตัวทรานซิสเตอร์นำกระแสได้ ถ้าปลดแรงดันออกจากไดโอดเปล่งแสง ทรานซิสเตอร์ก็จะหยุดนำกระแส ดังนั้นเราสามารถออกแบบวงจรเชื่อมโยงทางแสงเข้าและออกได้ดังรูปที่ 3.12



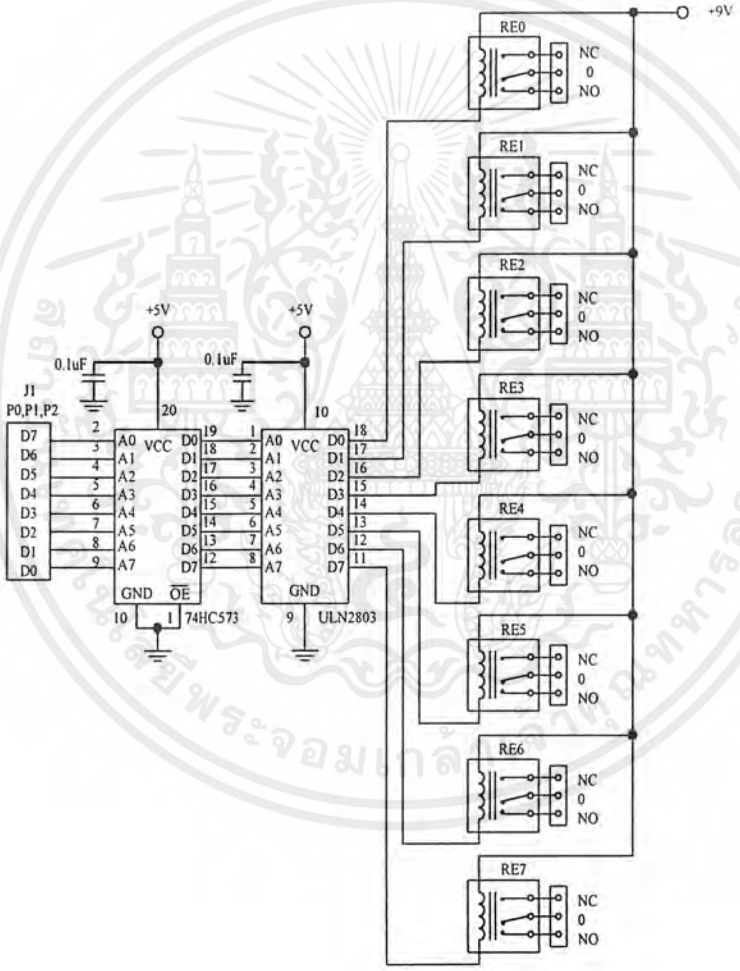
รูปที่ 3.12 การออกแบบวงจรเชื่อมโยงทางแสงออก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.3.10 การออกแบบวงจรขับรีเลย์

วงจรขับรีเลย์ เป็นวงจรที่ใช้ขับรีเลย์จำนวน 8 ช่อง การทำงานมีการจ่ายแรงดันไฟตรงให้กับรีเลย์ 9 โวลต์ เพื่อเป็นไฟเลี้ยงให้กับตัวรีเลย์

เมื่อรับอินพุตจากพอร์ตของ MCS-51 โดยจัมพ์เลือกระหว่างพอร์ต P0, P1 หรือ P2 ซึ่งวงจรขับรีเลย์จะมีบัฟเฟอร์ไอซีเบอร์ 74HC573 โดยมีขาที่ 11 เป็นตัวเลือก แล้วส่งผ่านข้อมูลไปยังรีเลย์ การออกแบบวงจรแสดงได้ดังรูป 3.13

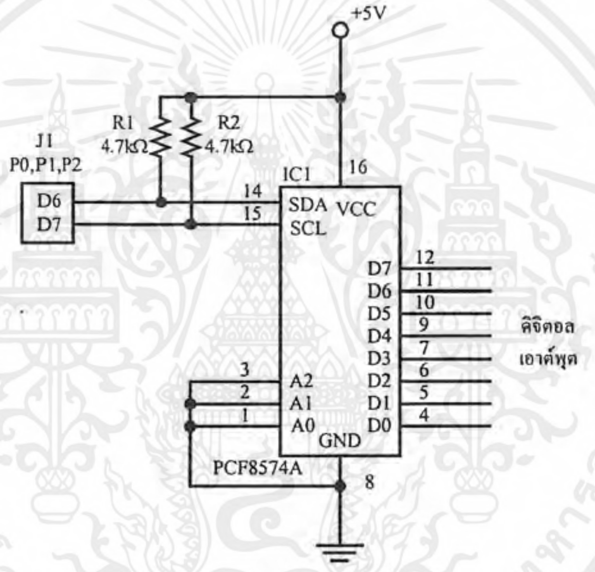


รูปที่ 3.13 การออกแบบวงจรขับรีเลย์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.3.11 การออกแบบวงจรขยายพอร์ตด้วยระบบ I^2C

วงจร I^2C คือการติดต่อสื่อสารระหว่าง IC โดยติดต่อสั่งงานภายใต้สายสัญญาณเพียง 2 เส้น โดยเส้นหนึ่ง คือสัญญาณนาฬิกา การออกแบบวงจรจะใช้ IC เบอร์ PCF8574 เป็นอุปกรณ์ขยายพอร์ต การต่อวงจรใช้งานสามารถติดต่อ สั่งงาน และควบคุมภายใต้สายสัญญาณเพียง 2 เส้น เส้นหนึ่ง คือ สายข้อมูลอนุกรมหรือ SDA ส่วนสายสัญญาณอีกสายหนึ่ง คือ สายสัญญาณนาฬิกาหรือ SCL ของอุปกรณ์แต่ละตัวขนานกันหรือพ่วงไป ส่วนการกำหนดตำแหน่ง สำหรับอุปกรณ์แต่ละตัวจะใช้รหัสข้อมูลและการกำหนดสภาวะลอจิกที่ขาตำแหน่งของอุปกรณ์แต่ละตัว การออกแบบวงจรแสดงได้ดังรูปที่ 3.14

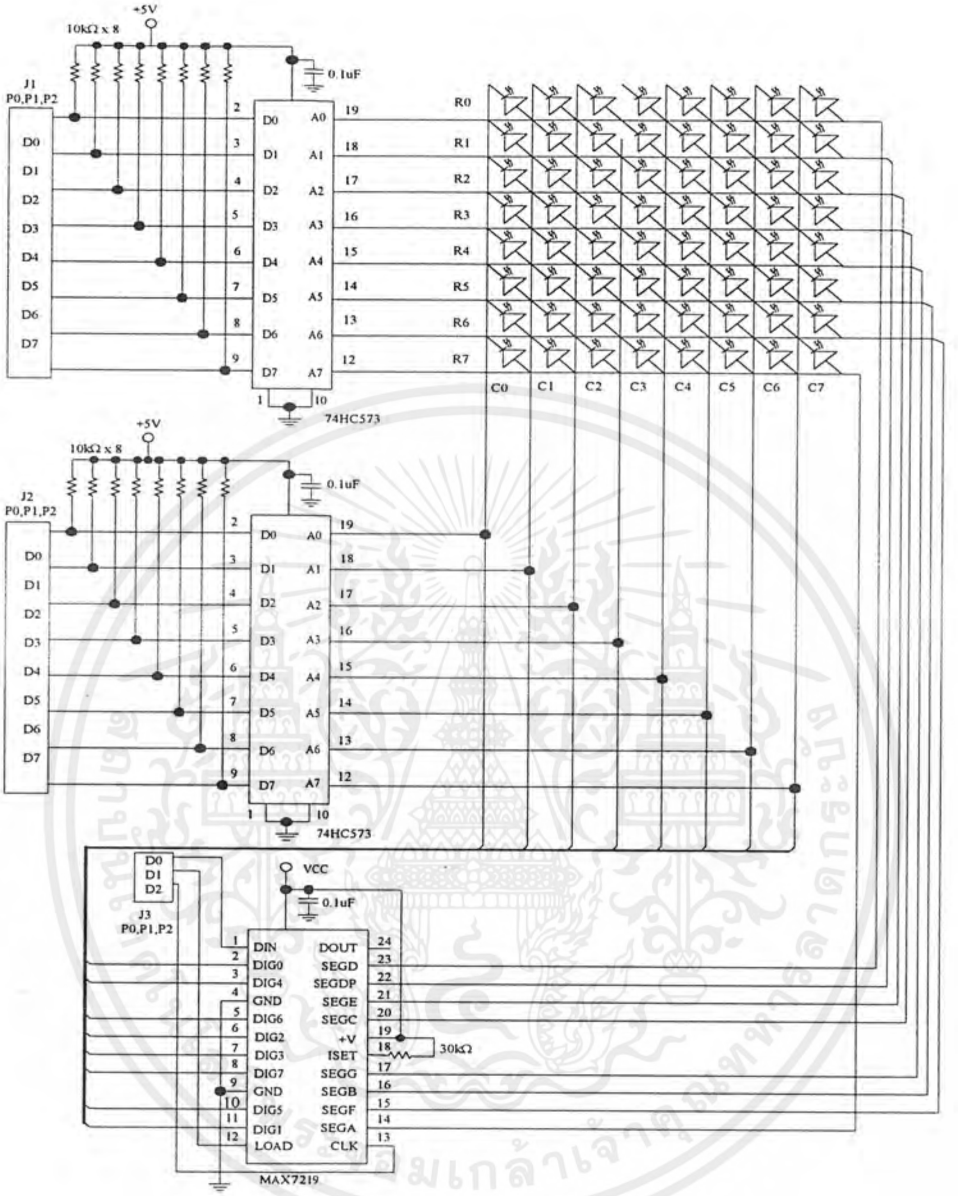


รูปที่ 3.14 การออกแบบวงจรขยายพอร์ตด้วยระบบบัส I^2C

3.3.12 การออกแบบวงจรแสดงผลแอลอีดีแบบเมตริกซ์

วงจรแอลอีดีแบบเมตริกซ์ จะใช้ แอลอีดี 8x8 จุด รวมเป็น 64 ตัว และใช้ไอซีอีก 2 ตัวเบอร์ 74HC573 เป็นตัวสำหรับเช็คแถวและหลัก และไอซีอีก 1 ตัว เบอร์ MAX7219 ที่สามารถทำงานได้ทั้งแถว และหลักในตัวเดียว การออกแบบวงจรแสดงได้ดังรูปที่ 3.15

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.15 วงจรแสดงผลแอลอีดีแบบเมตริกซ์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 4

การทดลอง และผลการทดลอง

ในส่วนของบทนี้จะกล่าวถึงการทดลอง และผลการทดลอง ซึ่งในการทดลองวงจรที่ออกแบบทั้งหมดจะใช้ บอร์ดควบคุมที่ทำการออกแบบคอมพิวเตอร์ และซอฟต์แวร์ CP-S8252

4.1 การทดลองวงจรแหล่งจ่ายไฟ

แหล่งจ่ายไฟประกอบด้วยแหล่งจ่ายแรงดันไฟตรงทั้งหมด 4 ชุดด้วยกัน คือ

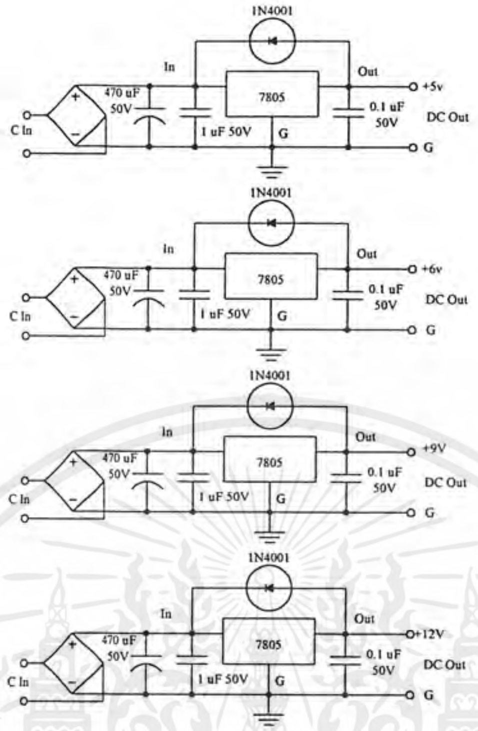
1) ไอซี 7805 ซึ่งแปลงแรงดันไฟตรง 5 โวลต์
2) ไอซี 7806 ซึ่งแปลงแรงดันไฟตรง 6 โวลต์
3) ไอซี 7809 ซึ่งแปลงแรงดันไฟตรง 9 โวลต์
4) ไอซี 7812 ซึ่งแปลงแรงดันไฟตรง 12 โวลต์

ชุดจ่ายแรงดันไฟตรง 5 โวลต์ ใช้สำหรับจ่ายแรงดันไฟตรงให้กับบอร์ดควบคุม และบอร์ดอินเตอร์เฟส ส่วน 6 โวลต์, 9โวลต์, และ 12 โวลต์ เป็นแหล่งจ่ายแรงดันไฟสำรองให้กับอุปกรณ์เชื่อมต่อภายนอกเช่น สเต็ปปิ้งมอเตอร์, อุปกรณ์รีเลย์ เป็นต้น จากการทดลองวัดแรงดันไฟตรงแหล่งจ่ายแรงดันไฟตรงได้ผลดังตารางที่ 4.1

ตารางที่ 4.1 ผลการทดลองวัดค่าแรงดันไฟตรงจากแหล่งจ่ายไฟ

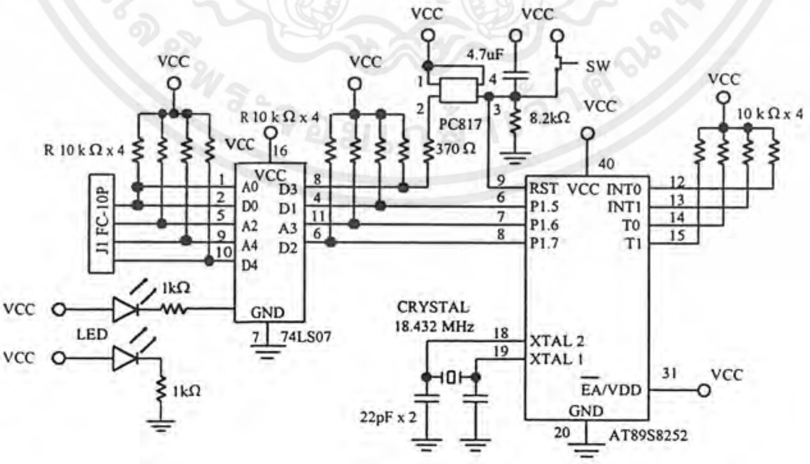
แหล่งจ่ายแรงดันไฟตรง	แรงดันที่วัดได้ (โวลต์)
7805	4.9
7806	5.8
7809	8.9
7812	11.8

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.1 วงจรแหล่งจ่ายไฟ

4.2 การทดลองวงจรควบคุม



รูปที่ 4. 2 วงจรควบคุม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

วงจรควบคุมจะประกอบด้วยไมโครคอนโทรลเลอร์ AT89S8252 ซึ่งเป็นตัวควบคุมโดยมีการส่งผ่านข้อมูลแบบ SPI โดยการดาวน์โหลดข้อมูลลงในไมโครคอนโทรลเลอร์ได้โดยไม่ต้องโปรแกรมจากเครื่องโปรแกรมภายนอก

จากการทดลองนำวงจรควบคุมมาเชื่อมต่อกับวงจรแสดงผลโดยใช้ไดโอดเปล่งแสงโดยต่อที่พอร์ต P0 พอร์ต P1 และพอร์ต P2 เมื่อทำการดาวน์โหลดข้อมูลต่างๆ ซึ่งได้ผลการทดลองเป็นไปตามตารางที่ 4.2

ตารางที่ 4.2 ผลการทดลองวงจรควบคุม

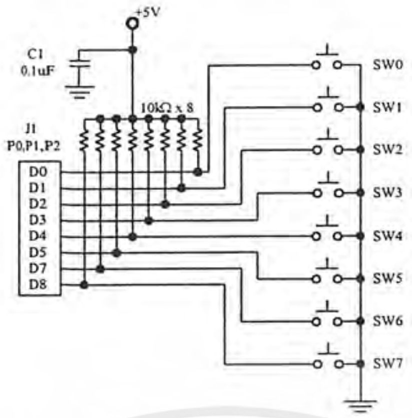
ข้อมูลอินพุต	ข้อมูลเอาต์พุต
00H	00H
01H	01H
02H	02H
04H	04H
08H	08H
10H	10H
20H	20H
40H	40H
80H	80H

4.3 การทดลองวงจรอินเตอร์เฟส

4.3.1 การทดลองวงจรสวิตช์เดี่ยว

จากรูปที่ 4.3 วงจรจะประกอบด้วยสวิตช์เดี่ยวทั้งหมด 8 ตัวได้ทำการทดลองโดยการนำสวิตช์ต่อกับไดโอดเปล่งแสง และป้อนแรงดันไฟตรง 5 โวลต์ให้กับสวิตช์ได้ผลคือเมื่อ สวิตช์อยู่ในสภาวะปิดหรือลอจิกต่ำ ผลที่ได้คือไดโอดเปล่งแสงอยู่ในสภาวะดับ เมื่อเปลี่ยนสภาวะให้เป็นสภาวะหรือจิกสูง ผลที่ได้คือไดโอดเปล่งแสงอยู่ในสภาวะสว่าง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.3 วงจรสวิตซ์เดี่ยว 8 ตัว

ทำการเชื่อมต่อเป็นอินพุตให้กับวงจรควบคุมและแสดงผลด้วยวงจรแสดงผลไดโอดเปล่งแสง ทำการเขียนโปรแกรมดังนี้

```
                ORG     0000H
START:          CLR     A
                MOV     P0,#0FFH
CH_KEY:         MOV     A,P0
                ANL     A,#0FFH
CH_KEY1:        CJNE    A,#0FEH,CH_KEY2
                JMP     KEY1
CH_KEY2:        CJNE    A,#0FDH,CH_KEY3
                JMP     KEY2
CH_KEY3:        CJNE    A,#0FAH,CH_KEY4
                JMP     KEY3
CH_KEY4:        CJNE    A,#0F7H,CH_KEY5
                JMP     KEY4
CH_KEY5:        CJNE    A,#0EFH,CH_KEY6
                JMP     KEY5
CH_KEY6:        CJNE    A,#0DFH,CH_KEY7
                JMP     KEY6
CH_KEY7:        CJNE    A,#0AFH,CH_KEY8
                JMP     KEY7
CH_KEY8:        CJNE    A,#7FH,CH_KEY
                JMP     KEY8
;*******************************************
KEY1:           MOV     P2,#01H
                ACALL   DELAY_1S
                JMP     START
KEY2:           MOV     P2,#02H
                ACALL   DELAY_1S
                JMP     START
KEY3:           MOV     P2,#04H
                ACALL   DELAY_1S
                JMP     START
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
KEY4:           MOV     P2,#08H
                ACALL   DELAY_1S
                JMP     START
KEY5:           MOV     P2,#10H
                ACALL   DELAY_1S
                JMP     START
KEY6:           MOV     P2,#20FH
                ACALL   DELAY_1S
                JMP     START
KEY7:           MOV     P2,#40FH
                ACALL   DELAY_1S
                JMP     START
KEY8:           MOV     P2,#80H
                ACALL   DELAY_1S
                JMP     START
;*********************************************
DELAY:          MOV     7,#010H
DELAY_1:        MOV     6,#0E6H
DELAY_2:        NOP
                DJNZ    R6,DELAY_2
                DJNZ    R7,DELAY_1
                RET
DELAY_1S:       MOV     5,#100
DELAY_1S_1:     ACALL   DELAY
                DJNZ    R5,DELAY_1S_1
                RET
                END
```

รูปที่ 4.4 โปรแกรมการทดลองไดโอดเปล่งแสง

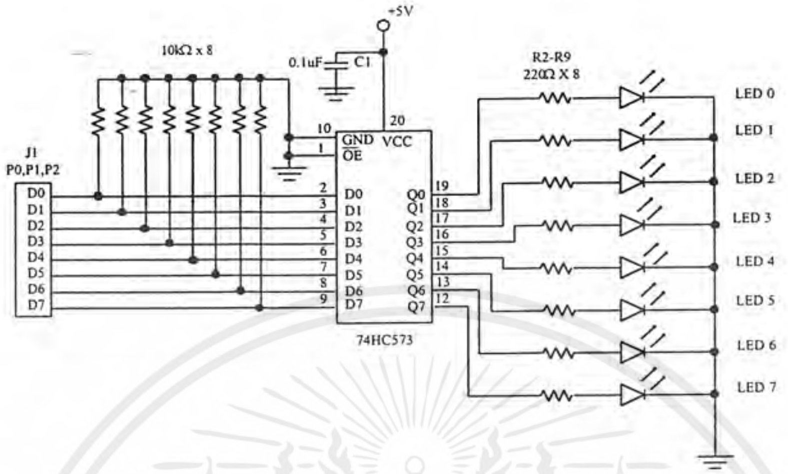
จากโปรแกรมจะได้ผลการทดลองดังตารางที่ 4.3

ตารางที่ 4.3 ผลการทดลองวงจรสวิตซ์เดี่ยว

สวิตซ์								เอาต์พุต
0	0	0	0	0	0	0	1	01H
0	0	0	0	0	0	1	1	03H
0	0	0	0	0	1	1	1	07H
0	0	0	0	1	1	1	1	0FH
0	0	0	1	1	1	1	1	1FH
0	0	1	1	1	1	1	1	3FH
0	1	1	1	1	1	1	1	7FH
1	1	1	1	1	1	1	1	FFH

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.3.2 การทดลองวงจรเชื่อมต่อไดโอดเปล่งแสงชนิดคาโถดร่วม



รูปที่ 4.5 วงจรเชื่อมต่อไดโอดเปล่งแสงชนิดคาโถดร่วม

จากรูปที่ 4.5 เป็นวงจรเชื่อมต่อโดยใช้ไดโอดเปล่งแสงชนิดคาโถดร่วม จะประกอบด้วย ไอซี 74HC541 ซึ่งเป็นไอซีค้างสภาวะทำหน้าที่เป็นตัวเลือกให้ข้อมูลออกจากวงจรไปแสดงผลที่ไดโอดเปล่งแสง โดยมีความต้านทาน 220 โอห์ม เป็นตัวจำกัดกระแสไหลผ่านไดโอดเปล่งแสงถ้าต้องการให้ไดโอดเปล่งแสงตัวใดสว่างก็ป้อนลอจิก “1” และถ้าต้องการให้ไดโอดเปล่งแสงตัวใดดับก็ป้อนลอจิก “0” ได้ผลการทดลองดังตารางที่ 4.5

จากวงจรสามารถนำมาเขียนเป็นโปรแกรมไฟวิ่งซ้ายได้ดังนี้

```
            ORG         0000H
            MOV         P2,A
            MOV         R3,#00H
            MOV         DPTR,#DATA
LOOP:       MOVC        A,@A+DPTR
            MOV         P2,A
            ACALL       DELAY
            INC         R3
            MOV         A,R3
            CJNE        R3,#08H,LOOP
            SJMP        START
DATA:       DB      01H,02H,04H,08H,10H,20H,40H,80H
DELAY:      MOV         7,#010H
DELAY_1:    MOV         6,#0E6H
DELAY_2:    NOP
            NOP
            DJNZ        R6,DELAY_2
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        DJNZ        R7,DELAY_1
        RET
        END
```

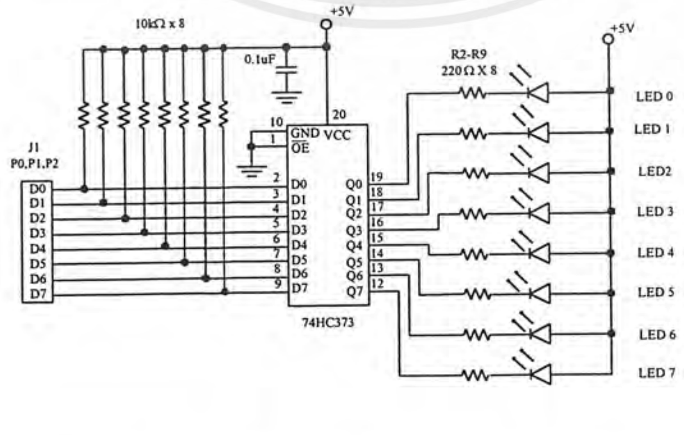
รูปที่ 4.6 โปรแกรมไฟวิ่งซ้าย

จากโปรแกรมจะได้ผลการทดลองดังตารางที่ 4.4

ตารางที่ 4.4 ผลการทดลองวงจรเชื่อมต่อไดโอดเปล่งแสงชนิดคาโถดร่วม

อินพุต	เอาต์พุต	ตำแหน่งบิต							
		0	1	2	3	4	5	6	7
00	00	0	0	0	0	0	0	0	0
01	01	0	0	0	0	0	0	0	1
03	03	0	0	0	0	0	0	1	1
80	80	1	0	0	0	0	0	0	0
C0	C0	1	1	0	0	0	0	0	0
F0	F0	1	1	1	1	0	0	0	0
FF	FF	1	1	1	1	1	1	1	1

4.3.3 การทดลองวงจรเชื่อมต่อไดโอดเปล่งแสงชนิดอาโนดร่วม



รูปที่ 4.7 วงจรเชื่อมต่อไดโอดเปล่งแสงชนิดอาโนดร่วม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากรูปที่ 4.7 เป็นวงจรเชื่อมต่อโดยใช้ไดโอดเปล่งแสงชนิดอาโนดร่วม จะประกอบด้วยไอซี 74HC541 เป็นไอซีขยายกระแสซึ่งจะขยายกระแสให้วงจรไดโอดเปล่งแสง โดยวงจรจะมีความต้านทาน 220 โอห์ม เป็นตัวจำกัดกระแสไหลผ่านไดโอดเปล่งแสงถ้าต้องการให้ ไดโอดเปล่งแสงตัวใดสว่างก็ป้อนลอจิก "0" และถ้าต้องการให้ไดโอดเปล่งแสงตัวใดดับก็ป้อนลอจิก "1"

จากวงจรสามารถนำมาเขียนเป็นโปรแกรมไฟวิ่งซ้ายได้ดังนี้

```
            ORG       0000H
            MOV       P2,A
            MOV       R3,#0FFH
            MOV       DPTR,#DATA
LOOP:       MOVC      A,@A+DPTR
            MOV       P2,A
            ACALL     DELAY
            INC       R3
            MOV       A,R3
            CJNE      R3,#08H,LOOP
            SJMP      START
DATA:       DB     0FEH,0FDH,0FBH,0F7H,0EFH,0DFH,0BFH,7FH
;*******************************************************
DELAY:      MOV         7,#010H
DELAY_1:    MOV         6,#0E6H
DELAY_2:    NOP
            DJNZ        R6,DELAY_2
            DJNZ        R7,DELAY_1
            RET
            END
```

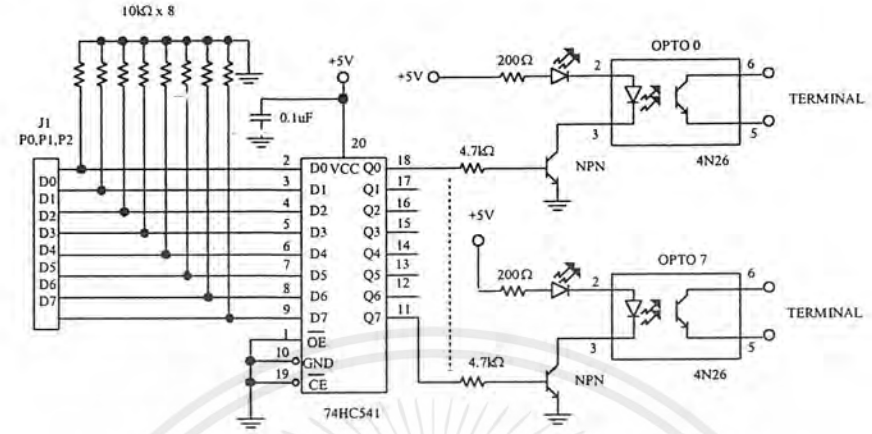
รูปที่ 4.8 โปรแกรมไฟวิ่งซ้าย

ตารางที่ 4.5 ผลการทดลองวงจรเชื่อมต่อไดโอดเปล่งแสงชนิดอาโนดร่วม

อินพุต	เอาต์พุต	ตำแหน่งบิต							
		0	1	2	3	4	5	6	7
FF	00	0	0	0	0	0	0	0	0
FE	01	0	0	0	0	0	0	0	1
FB	03	0	0	0	0	0	0	1	1
7F	80	1	0	0	0	0	0	0	0
3F	C0	1	1	0	0	0	0	0	0
0F	F0	1	1	1	1	0	0	0	0
00	FF	1	1	1	1	1	1	1	1

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.3.4 การทดลองวงจรเชื่อมโยงแสงออก



รูปที่ 4.9 วงจรเชื่อมโยงแสงออก

จากรูปที่ 4.9 การทำงานจะใช้การเชื่อมต่อทางแสงโดยใช้ไดโอดเปล่งแสงภายในเบอร์ 4N26 ซึ่งจะถูกควบคุมโดยทรานซิสเตอร์ 2N3904 แบบเอ็นพีเอ็น ทำการป้อนลอจิกต่ำให้กับวงจรเชื่อมโยงแสงออกผลที่ได้ก็คือ ไดโอดเปล่งแสงดับ ทำการป้อนลอจิกสูงให้กับวงจรเชื่อมโยงแสงออกผลที่ได้ก็คือไดโอดเปล่งแสงสว่าง เมื่อนำมาเชื่อมต่อกับวงจรควบคุม แล้วทำการป้อนข้อมูล ถ้าข้อมูลที่ส่งออกมาทางพอร์ตมีค่าเป็นลอจิกสูงจะทำให้ไดโอดเปล่งแสงภายในตัวเชื่อมโยงทางแสงสว่าง จากการทดลองวงจรเชื่อมโยงแสงออก

จากวงจรนำมาเขียนโปรแกรมโดยจะทำการป้อนลอจิกสูงออกที่พอร์ต P1 ดังนี้

```
            ORG       0000H
START:      MOV       A,#01H
            MOV       P1,A
            ACALL     DELAY
            RL        A
            SJMP      START
DELAY:      MOV       7,#010H
DELAY_1:    MOV       6,#0E6H
DELAY_2:    NOP
            DJNZ      R6,DELAY_2
            DJNZ      R7,DELAY_1
            RET
            END
```

รูปที่ 4.10 โปรแกรมทดลองวงจรเชื่อมโยงทางแสงออก

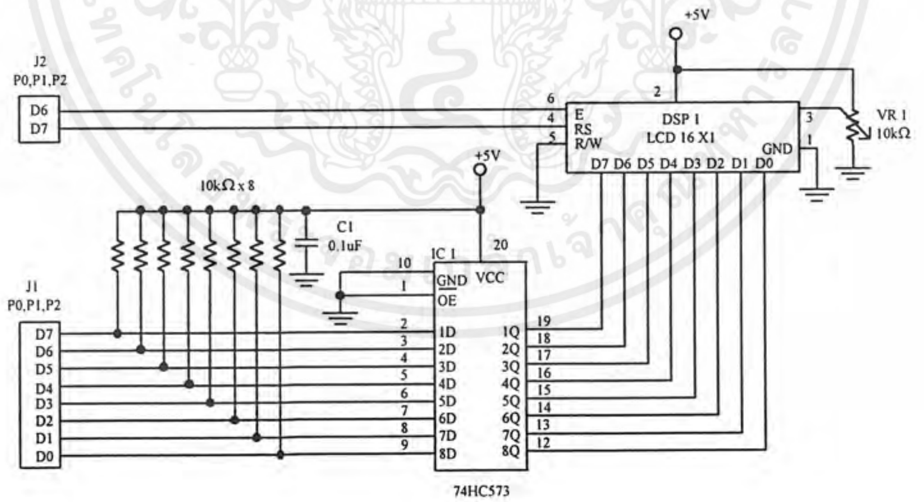
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากโปรแกรมดังกล่าวได้ผลการทดลองดังนี้

ตารางที่ 4.6 ผลการทดลองวงจรเชื่อมโยงทางแสงออก

ข้อมูลพอร์ต	บิตแสดงผล
01	0
02	1
04	2
08	3
10	4
20	5
40	6
80	7

4.3.5 การทดลองวงจรแสดงผลแบบผลึกเหลวขนาด 16 ตัวอักษร



รูปที่ 4.11 วงจรแสดงผลแบบผลึกเหลวขนาด 16 ตัวอักษร

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ในการทดลองวงจรแสดงผลแบบผลึกเหลวขนาด 16 ตัวอักษรนั้นเราสามารถเขียนโปรแกรมเพื่อทำการควบคุมได้โดยจ่ายไฟเลี้ยงให้กับจอแสดงผลแบบผลึกเหลวในการเขียนโปรแกรมจะต้องกำหนดตำแหน่ง 8 บิตแรกที่ตำแหน่ง 00H และ 8 บิตหลังที่ตำแหน่ง 040H

จากวงจรสามารถนำมาเขียนโปรแกรมได้ดังนี้

```
            LCD_EN      BIT     P1.6
            LCD_RS      BIT     P1.7
            LCD_ADDR    EQU     030H
            LCD_DATA    EQU     031H
            ORG         0000H
LOOP:       MOV         LCD_ADDR,#000H
            ACALL       SET_ADDR_LCD
            MOV         DPTR,#DATA1
            ACALL       WRLINE_LCD
            MOV         LCD_ADDR,#040H
            ACALL       SET_ADDR_LCD
            MOV         DPTR,#DATA2
            ACALL       WRLINE_LCD
            ACALL       DELAY_1S
            ACALL       DELAY_1S
            ACALL       LCD_OFF
            ACALL       DELAY_1
            ACALL       LCD_ON
            ACALL       DELAY_1S
            MOV         LCD_ADDR,#000H
            ACALL       SET_ADDR_LCD
            MOV         DPTR,#DATA3
            ACALL       WRLINE_LCD
            MOV         LCD_ADDR,#040H
            ACALL       SET_ADDR_LCD
            MOV         DPTR,#DATA4
            ACALL       WRLINE_LCD
            ACALL       DELAY_1S
            ACALL       DELAY_1S
            ACALL       LCD_OFF
            ACALL       DELAY_1S
            ACALL       LCD_ON
            ACALL       DELAY_1S
            ACALL       LCD_OFF
            ACALL       DELAY_1S
            ACALL       LCD_ON
            ACALL       DELAY_1S
            AJMP        LOOP
LCD_CLR:    CLR         LCD_RS
            MOV         P2,#01H
            ACALL       LCD_CLK
            RET
LCD_OFF:    CLR         LCD_RS
            MOV         P2,#08H
            ACALL       LCD_CLK
            RET
LCD_CLK:    SETB        LCD_EN
            ACALL       LCD_DELAY
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
            CLR          LCD_EN
            ACALL        LCD_DELAY
            RET
LCD_ON:     CLR          LCD_RS
            MOV          P2,#0CH
            ACALL        LCD_CLK
            RET
SET_ADDR_LCD:CLR         LCD_RS
            MOV          A,LCD_ADDR
            SETB         ACC.7
            MOV          P2,A
            ACALL        LCD_CLK
            RET
WRLINE_LCD: MOV          R0,#0
WRLINE_LCD_1:SETB        LCD_RS
            CLR          A
            MOVC         A,@A+DPTR
            MOV          P2,A
            ACALL        LCD_CLK
            INC          DPTR
            INC          R0
            CJNE         R0,#8,WRLINE_LCD_1
            ACALL        LCD_ON
            RET
LCD_DELAY:  MOV          7,#002
LCD_DELAY_1:MOV          6,#0E6H
LCD_DELAY_2:NOP
            NOP
            DJNZ         R6,LCD_DELAY_2
            DJNZ         R7,LCD_DELAY_1
            RET
DELAY_10MS: MOV          7,#010
DELAY_10MS_1:MOV         6,#0E6H
DELAY_10MS_2:NOP
            NOP
            DJNZ         R6,DELAY_10MS_2
            DJNZ         R7,DELAY_10MS_1
            RET
DELAY_1S:   MOV          5,#100
DELAY_1S_1: ACALL        DELAY_10MS
            DJNZ         R5,DELAY_1S_1
            RET
DATA1:      DB            '#  KMITL'
DATA2:      DB            ' ED.21 #'
DATA3:      DB            '   GOOD '
DATA4:      DB            '  BYE   '
            END
```

รูปที่ 4.12 โปรแกรมขับโมดูลแบบผลึกเหลว

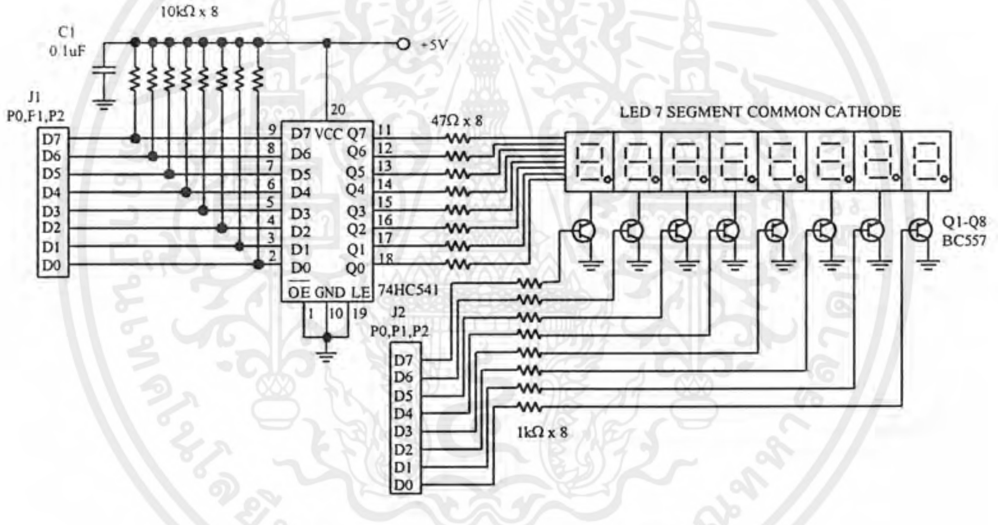
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ผลการทดลองโปรแกรมขับโมดูลแบบผลึกเหลวได้ดังตารางที่ 4.7

ตารางที่ 4.7 ผลการทดลองวงจรขับโมดูลแบบผลึกเหลว

หน้าจอแอลซีดี	ข้อความ
1	# KMITL ED.21 #
2	GOOD BYE

4.3.6 การทดลองวงจรแสดงผลแบบ 7 ส่วน



รูปที่ 4.13 วงจรแสดงผลแบบ 7 ส่วน

จากวงจรรูปที่ 4.13 ส่วนแสดงผลใช้ไอซี 74HC541 เป็นตัวค้างข้อมูลโดยไอซี 74HC541 ทำหน้าที่ค้างข้อมูลทีละพอร์ต ทรานซิสเตอร์ทำหน้าที่แสดงผลหลักของข้อมูลที่แสดงผล การส่งข้อมูลจะส่งผ่านไอซี 74HC541 โดยข้อมูลที่ส่งมาเป็นสัญญาณรหัสของการแสดงผล การแสดงผลแบบ 7 ส่วน แสดงผลออกมาเป็นตัวเลข ภายในประกอบด้วยไดโอดเปล่งแสง 8 ดวง เป็นส่วน a, b, c, d, e, f, g และจุดทศนิยม (dp) การแสดงผลของวงจรจะต้องจ่ายไฟให้แก่ขาร่วมของแอลอีดี ตัวเลขทีละหลักตามลำดับโดยผ่านทรานซิสเตอร์ การแสดงผลของวงจรแสดงผลแบบ 7 ส่วน ต้องอาศัยการเขียนโปรแกรมจากไมโครคอนโทรลเลอร์ MCS-51 ควบคู่ไปกับการเปิดตารางข้อมูลประกอบด้วย

จากวงจรสามารถนำมาเขียนโปรแกรมได้ดังนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        DSP1      BIT   P0.0
        DSP2      BIT   P0.1
                  ORG     0000H
                  CLR      DSP1
                  CLR      DSP2
                  MOV     A,#077H
                  MOV     P2,A
                  ACALL   DELAY_1MS
DELAY_1MS:        MOV     R0,#0E6H
DELAY:            NOP
                  NOP
                  DJNZ    R0,DELAY
                  RET
                  END
```

รูปที่ 4.14 โปรแกรมขับแอลอีดีแบบ 7 ส่วน

ตารางที่ 4.8 ผลการทดลองวงจรแอลอีดีแบบ 7 ส่วน

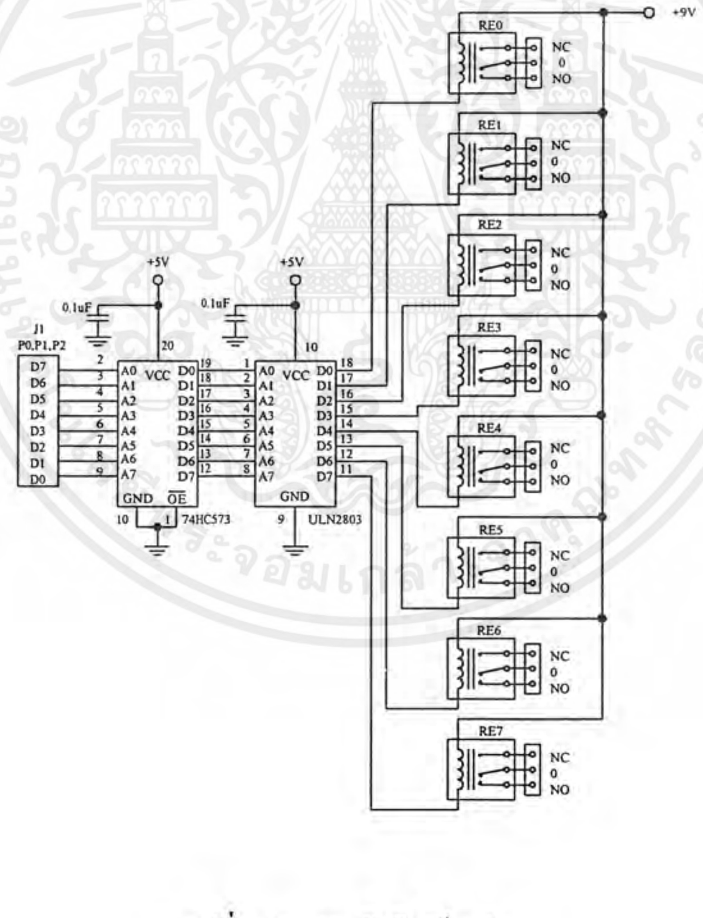
ข้อมูลพอร์ต	ตัวเลข
3FH	0
06H	1
5BH	2
4FH	3
66H	4
6DH	5
7DH	6
07H	7
7FH	8
6FH	9
77H	A
7CH	B
39H	C
5EH	D
79H	E
71H	F

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.3.7 การทดลองวงจรขับรีเลย์ 8 ช่อง

วงจรขับรีเลย์ เป็นวงจรที่ใช้ขับรีเลย์จำนวน 8 ตัวแสดง การทำงานโดยมีการจ่ายไฟให้กับรีเลย์ 9 โวลต์ เมื่อรับอินพุตจากพอร์ตของ MCS-51 โดยจัมพ์เลือกระหว่างพอร์ต P0 , P1 หรือ P2 ซึ่งวงจรขับรีเลย์จะมีบัฟเฟอร์ 74HC573 เป็นตัวค้างสภาวะโดยมีขาที่ 11 เป็นขาเลือกสัญญาณ แล้วส่งผ่านข้อมูลไปยังรีเลย์ ซึ่งสามารถนำไปใช้ในการควบคุมการเปิด-ปิดวงจรอิเล็กทรอนิกส์ได้ ดังรูปที่ 4.15

ทำการทดลองโดยป้อนลอจิกสูงให้กับอุปกรณ์รีเลย์โดยใช้แอลอีดีเป็นอุปกรณ์แสดงผล ซึ่งจากการทดลองได้ผลว่าเมื่อทำการป้อนลอจิก "0" ให้กับอุปกรณ์รีเลย์ทำให้รีเลย์ไม่ทำงาน แต่เมื่อป้อนลอจิก "1" ให้กับอุปกรณ์รีเลย์ทำให้รีเลย์ทำงานแสดงว่าสภาวะการทำงานของรีเลย์จะทำงานที่สภาวะเป็น "1"



รูปที่ 4.15 วงจรขับรีเลย์ 8 ช่อง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากวงจรสามารถนำมาเขียนโปรแกรมได้ดังนี้

```
            ORG         0000H
START:      MOV         A,#01H
            MOV         P2,A
            ACALL       DELAY
            RL          A
            SJMP        START
DELAY:      MOV         7,#010H
DELAY_1:    MOV         6,#0E6H
DELAY_2:    NOP
            NOP
            DJNZ        R6,DELAY_2
            DJNZ        R7,DELAY_1
            RET
            END
```

รูปที่ 4.16 โปรแกรมการควบคุมการขับรีเลย์ 8 ช่อง

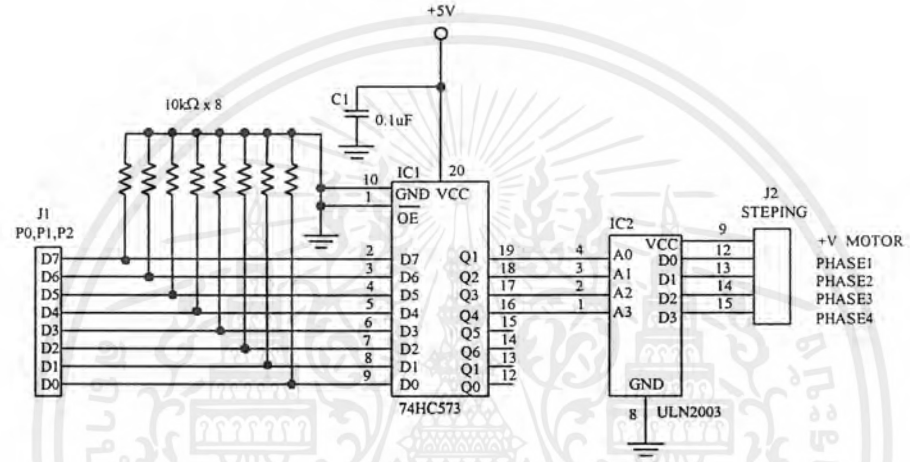
ตารางที่ 4.9 ผลการทดลองวงจรขับรีเลย์

พอร์ต P2	บิตที่แสดงผล
01	0
02	1
04	2
08	3
10	4
20	5
40	6
80	7
03	0,1
07	0,1,2
0F	0,1,2,3
1F	0,1,2,3,4
3F	0,1,2,3,4,5
7F	0,1,2,3,4,5,6
FF	0,1,2,3,4,5,6,7

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.3.8 การทดลองวงจรขับสเต็ปปิ้งมอเตอร์

การทำงานของวงจรขับสเต็ปปิ้งมอเตอร์จะใช้การส่งผ่านข้อมูล ผ่านไอซีแลตช์ 74HC573 แล้วส่งต่อไปยังไอซีไดร์เวอร์กระแสสูง แบบคอลเล็คเตอร์เปิดเบอร์ ULN2003 ด้วยการใช้ไอซีแบบนี้ทำให้สามารถเลือกแรงดันขยับสเต็ปปิ้งมอเตอร์ได้ ตั้งแต่ 5 – 30 โวลต์ โดย ULN2003 มีความสามารถจ่ายกระแสได้สูงสุด 500 มิลลิแอมป์ ต่อขา จากวงจรจะมี 4 เฟสคือ เฟส 1, 2, 3 และ 4 ถ้าต้องการให้สเต็ปปิ้งมอเตอร์ขยับก็ทำการส่งข้อมูลที่เป็น “1” ให้กับวงจร



รูปที่ 4.17 วงจรขับสเต็ปปิ้งมอเตอร์

การทดลองทำการป้อนข้อมูลที่เป็น “1” ให้กับเฟสของสเต็ปปิ้งมอเตอร์ตามเฟส 1, 2, 3, 4 ตามลำดับ ซึ่งเป็นผลทำให้สเต็ปปิ้งมอเตอร์หมุนวนรอบไปเรื่อย ๆ

จากวงจรนำมาเขียนโปรแกรมควบคุมได้ดังนี้

```
        ORG     0000H
START:  MOV     A,#01H
        MOV     P2,A
        ACALL   DELAY
        MOV     A,#02H
        MOV     P2,A
        ACALL   DELAY
        MOV     A,#04H
        MOV     P2,A
        ACALL   DELAY
        MOV     A,#08H
        MOV     P2,A
        ACALL   DELAY
        JMP     START
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
DELAY:      MOV     R3,#0FFH
DELAY_1:    MOV     R4,#0FFH
            DJNZ    R4,$
            DJNZ    R3,DELAY_1
            RET
            END
```

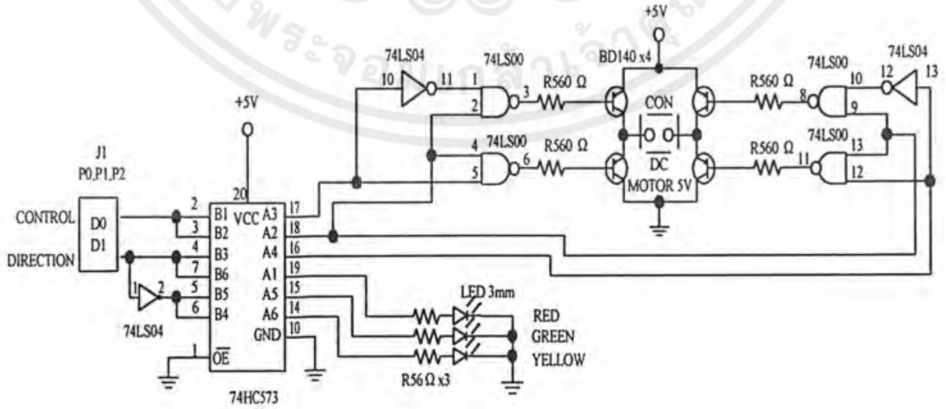
รูปที่ 4.18 โปรแกรมขับสเต็ปปิ้งมอเตอร์

จากโปรแกรมจะได้ผลการทดลองดังตารางที่ 4.10

ตารางที่ 4.10 ผลการทำงานของสเต็ปปิ้งมอเตอร์

ข้อมูลพอร์ต P1	การทำงานของสเต็ปปิ้งมอเตอร์			
	เฟสที่1	เฟสที่ 2	เฟสที่ 3	เฟสที่ 4
01H	0	0	0	1
02H	0	0	1	0
04H	0	1	0	0
08H	1	0	0	0

4.3.9 การทดลองวงจรมอเตอร์ไฟฟ้ากระแสตรง



รูปที่ 4.19 วงจรมอเตอร์ไฟฟ้ากระแสตรง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การทดลองวงจรมอเตอร์ไฟฟ้ากระแสตรงจะใช้ไอซี 74HC573 เป็นไอซีแลตช์ การควบคุมความเร็วของมอเตอร์ให้ความเร็วคงที่ตามที่ต้องการเป็นเรื่องที่ทำได้ค่อนข้างยาก ถ้าใช้การควบคุมแบบธรรมดาทั่วไป คือ การควบคุมแรงดันที่จ่ายให้กับมอเตอร์ การจะควบคุมความเร็วของมอเตอร์ให้ได้ความเร็วคงที่ ต้องควบคุมกระแสที่จ่ายให้กับมอเตอร์ และการควบคุมทิศทางการหมุนของมอเตอร์ จากการทดลองได้ทำการป้อนแรงดันให้กับมอเตอร์ไฟฟ้ากระแสตรงเพื่อควบคุมทิศทาง เมื่อป้อนสภาวะลอจิกสูง ให้มอเตอร์จะทำให้มอเตอร์หมุนขวา และป้อนสภาวะที่เป็นลอจิกต่ำ จะทำให้มอเตอร์หยุดหมุน

จากวงจรนำมาเขียนโปรแกรมได้ดังนี้

```
              ORG       0000H
              MOV       A,#01H
START:        MOV       P2,A
              ACALL     DELAY_1S
              JMP       START
DELAY:        MOV       7,#010H
DELAY_1:      MOV       6,#0E6H
DELAY_2:      NOP
              NOP
              DJNZ      R6,DELAY_2
              DJNZ      R7,DELAY_1
              RET
DELAY_1S:     MOV       5,#1000
DELAY_1S_1:   ACALL     DELAY
              DJNZ      R5,DELAY_1S_1
              RET
              END
```

รูปที่ 4.20 โปรแกรมขับมอเตอร์ไฟฟ้ากระแสตรง

จากโปรแกรมจะได้ผลการทดลองดังตารางที่ 4.11

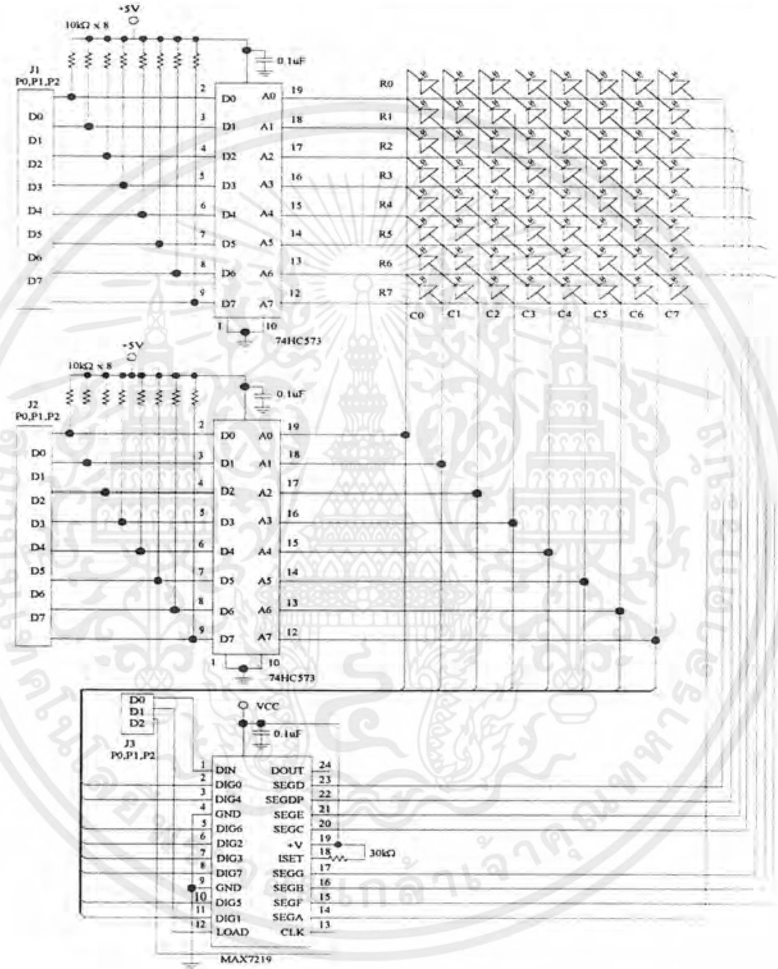
ตารางที่ 4.11 ผลการทดลองวงจรมอเตอร์ไฟฟ้ากระแสตรง

ข้อมูลพอร์ต P1	สภาวะการทำงานของมอเตอร์
01H	หมุนทางขวา
00H	หยุด
03H	หมุนทางซ้าย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.3.10 การทดลองวงจรแสดงผลแอลอีดีแบบเมตริกซ์ 8x8

วงจรแอลอีดีแบบเมตริกซ์ จะใช้แอลอีดี 8x8 จุด รวมเป็น 64 ตัว และใช้ไอซีอีก 2 ตัว เบอร์ 74HC573 เป็นตัวสำหรับเช็คแถวและหลัก และไอซีอีก 1 ตัว เบอร์ MAX7219 ที่สามารถทำงานได้ทั้งแถว และหลักในตัวเดียว ในการควบคุมให้แอลอีดีเมตริกซ์ทำงานได้นั้นจะต้องป้อนลอจิกสูงให้กับแถวเพื่อให้แอลอีดีสว่าง และป้อนลอจิกต่ำให้กับคอลัมน์เพื่อให้แอลอีดีติดเช่นเดียวกัน



รูปที่ 4.21 วงจรแสดงผลแอลอีดีแบบเมตริกซ์ 8x8

การคิดข้อมูลที่จะนำแสดงเป็นตัวอักษร จะพบว่ากำหนดให้พอร์ต 1 ของ AT89S8252 เป็นพอร์ตสำหรับควบคุมแถว และพอร์ตที่ 2 เป็นพอร์ตที่ใช้ควบคุมคอลัมน์ ข้อมูลที่ทำให้แอลอีดีติดได้ คือ ข้อมูลที่ออกมาจากพอร์ต 1 ซึ่งต้องเป็น ลอจิกสูง และข้อมูลพอร์ต 2 ต้องเป็น ลอจิกต่ำ จากหลักการดังกล่าวสามารถนำมาเขียนโปรแกรมได้ดังนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

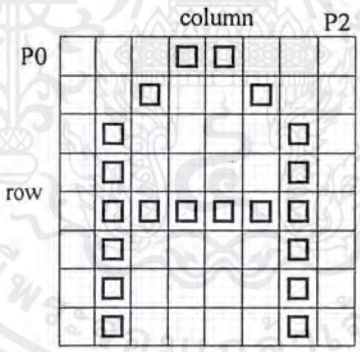
```
            ORG       0000H
START:      MOV       A,#0FCH
            MOV       P0,A
            ACALL     DELAY
            MOV       A,#0BFH
            MOV       P2,A
            ACALL     DELAY
            MOV       A,#00H
            MOV       P0,A
            ACALL     DELAY
            MOV       A,#0FFH
            MOV       P2,A
            ACALL     DELAY
            MOV       A,#12H
            MOV       P0,A
            ACALL     DELAY
            MOV       A,#0DFH
            MOV       P2,A
            ACALL     DELAY
            MOV       A,#00H
            MOV       P0,A
            ACALL     DELAY
            MOV       A,#0FFH
            MOV       P2,A
            ACALL     DELAY
            MOV       A,#11H
            MOV       P0,A
            ACALL     DELAY
            MOV       A,#0EFH
            MOV       P2,A
            ACALL     DELAY
            MOV       A,#00H
            MOV       P0,A
            ACALL     DELAY
            MOV       A,#0FFH
            MOV       P2,A
            ACALL     DELAY
            MOV       A,#11H
            MOV       P0,A
            ACALL     DELAY
            MOV       A,#0F7H
            MOV       P2,A
            ACALL     DELAY
            MOV       A,#00H
            MOV       P0,A
            ACALL     DELAY
            MOV       A,#0FFH
            MOV       P2,A
            ACALL     DELAY
            MOV       A,#12H
            MOV       P0,A
            ACALL     DELAY
            MOV       A,#0FBH
            MOV       P2,A
            ACALL     DELAY
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
            MOV         A,#00H
            MOV         P0,A
            ACALL       DELAY
            MOV         A,#0FFH
            MOV         P2,A
            MOV         A,#0FCH
            MOV         P0,A
            ACALL       DELAY
            MOV         A,#0FDH
            MOV         P2,A
            ACALL       DELAY
            JMP         START
DELAY:      MOV         R3,#01H
DELAY_1:    MOV         R4,#0FFH
            DJNZ        R4,$
            DJNZ        R3,DELAY_1
            RET
            END
```

รูปที่ 4.22 โปรแกรมแสดงผลแอลอีดีเมตริกซ์

จากโปรแกรมการทดลองได้ผลการทดลองดังรูป



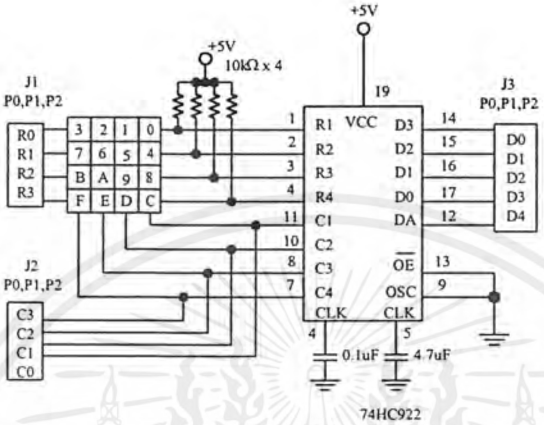
รูปที่ 4.23 ผลการทดลองโปรแกรมแอลอีดีเมตริกซ์

4.3.11 การทดลองวงจรสวิตช์เมตริกซ์

วงจรสวิตซ์เมตริกซ์จะใช้สวิตช์ต่อกันแบบเมตริกซ์ 4x4 โดยสวิตช์จะถูกต่อกันในแนวแกนตั้งและแนวแกนนอน จะเรียกแนวตั้งว่า หลักหรือคอลัมน์ และแนวแกนนอนจะเรียกว่า แถวหรือโรว์ ดังนั้นค่าของสวิตช์จะประกอบด้วย ตำแหน่งในแนวหลักและแถว กระบวนการที่จะทำให้ได้มาซึ่งค่าของสวิตซ์มีขั้นตอนซับซ้อนพอสมควร แต่วงจรสวิตซ์แบบเมตริกซ์มีข้อดี คือ สามารถ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รองรับการเพิ่มของสวิตช์ได้อย่างสะดวก เพียงเพิ่มจำนวนสวิตซ์และแก้ไขซอฟต์แวร์อีกเล็กน้อยเท่านั้น การเชื่อมต่อกับไมโครคอนโทรลเลอร์จะใช้สายหลัก 4 เส้น และแถว 4 เส้นต่อเข้ากับพอร์ตเดียวกันดังรูปที่ 4.21



รูปที่ 4.24 การทดลองวงจรสวิตซ์เมตริกซ์

จากการทดลองสามารถเขียนโปรแกรมเพื่อตรวจสอบค่าคีย์สวิตซ์เมตริกซ์ได้ดังนี้

```
;*********************************************
;                SCANKEY
;*********************************************
GET_KEYPAD:        MOV     P2,#00001111B
                   MOV     KPAD_DATA,#0
CHK_COL0:          CLR     KPAD_COL0
                   MOV     A,P2
                   ANL     A,00
                   CJNE    A,#00H,COL0_DETECT
                   AJMP    CHK_COL1
COL0_DETECT:       MOV     KPAD_DATA,#01
                   AJMP    GET_ROW
CHK_COL1:          SETB    KPAD_COL0
                   CLR     KPAD_COL1
                   MOV     A,P2
                   ANL     A,#00FH
                   CJNE    A,#00FH,COL1_DETECT
                   AJMP    CHK_COL2
COL1_DETECT:       MOV     KPAD_DATA,#02
                   AJMP    GET_ROW
CHK_COL2:          SETB    KPAD_COL1
                   CLR     KPAD_COL2
                   MOV     A,P2
                   ANL     A,#00FH
                   CJNE    A,#00FH,COL2_DETECT
                   AJMP    CHK_COL3
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
COL2_DETECT:     MOV     KPAD_DATA,#03
                 AJMP    GET_ROW
CHK_COL3:        SETB    KPAD_COL2
                 CLR     KPAD_COL3
                 MOV     A,P2
                 ANL     A,#00FH
                 CJNE    A,#00FH,COL3_DETECT
                 RET
COL3_DETECT:     MOV     KPAD_DATA,#04
GET_ROW:         CLR     KPAD_COL0
                 CLR     KPAD_COL1
                 CLR     KPAD_COL2
                 CLR     KPAD_COL3
                 JB      KPAD_ROW0,CHK_ROW1
                 RET
CHK_ROW1:        JB      KPAD_ROW1,CHK_ROW2
                 MOV     A,KPAD_DATA
                 ADD     A,#4
                 MOV     KPAD_DATA,A
                 RET
CHK_ROW2:        JB      KPAD_ROW2,CHK_ROW3
                 MOV     A,KPAD_DATA
                 ADD     A,#8
                 MOV     KPAD_DATA,A
                 RET
CHK_ROW3:        MOV     A,KPAD_DATA
                 ADD     A,#12
                 MOV     KPAD_DATA,A
                 RET
                 END
```

รูปที่ 4.25 โปรแกรมตรวจสอบค่าคีย์สวิตซ์เมตริกซ์

จากโปรแกรมผลการทดลองเป็นดังนี้

ในสภาวะที่ยังไม่มีการกดสวิตซ์ใดๆ ระดับแรงดันที่สามารถวัดได้ทางเอาต์พุตของคีย์สวิตซ์จะมีระดับแรงดันต่ำ หรือลอจิก “1” เมื่อมีการกดสวิตซ์ ค่าแรงดันที่เอาต์พุตของคีย์สวิตซ์ตัวที่กด จะมีระดับแรงดันต่ำ หรือลอจิก “0”

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 5

บทสรุป ปัญหา แนวทางการแก้ไข และพัฒนา

5.1 บทสรุป

ปริญญานิพนธ์ฉบับนี้เป็นการนำเสนอชุดทดลองการเชื่อมต่อไมโครคอนโทรลเลอร์ MCS-51 กับอุปกรณ์ภายนอกได้หลายชนิดซึ่งสามารถเขียนโปรแกรมควบคุมอุปกรณ์ภายนอกได้หลายชนิด ซึ่งเป็นการรวมวงจรการทดลองหลายๆ วงจรไว้ในชุดเดียวกัน เราสามารถเลือกทำการทดลองวงจรใดก็ได้ โดยใช้ซอฟต์แวร์ควบคุมการเลือกพอร์ตของวงจรที่ทำการทดลอง โดยที่พอร์ตแต่ละพอร์ตจะต้องไม่ซ้อนกัน ทำให้ง่ายและสะดวกต่อการใช้งาน ทางคณะผู้จัดทำหวังว่าชุดทดลองชุดนี้จะเป็นประโยชน์อย่างมาก และเพิ่มประสิทธิภาพในการเรียนการสอน ในรายวิชาที่มีความเกี่ยวข้องกับไมโครโปรเซสเซอร์ รวมถึงการพัฒนาทักษะในการเรียนภาคปฏิบัติของนักศึกษาให้เพิ่มขึ้นมากกว่าเดิม

อย่างไรก็ดีในการจัดทำชุดทดลองการเชื่อมต่อไมโครคอนโทรลเลอร์ MCS-51 กับอุปกรณ์ภายนอกนี้ ประสบปัญหาในการจัดทำหลายประการ จึงขอเสนอแนวทางการแก้ปัญหาและการพัฒนาที่น่าจะเป็นไปได้

5.2 ปัญหา และแนวทางแก้ไข

1. การออกแบบชุดทดลองในบางวงจร รูปแบบการวางอุปกรณ์ยังไม่เหมาะสมทำให้เกิดความผิดพลาดในการทดลองได้

แนวทางแก้ไข วงจรที่ออกแบบการวางอุปกรณ์ไม่เหมาะสม แก้ไขโดยทำการออกแบบการวางอุปกรณ์ใหม่

2. วงจรที่ใช้ในการทดลองบางวงจรไม่สามารถทำงานได้ตามที่ได้ออกแบบไว้ ทำให้เสียเวลาในการแก้ไข

แนวทางแก้ไข แก้ไขโดยการเปลี่ยนแปลงวงจรบางส่วน และอุปกรณ์บางตัว เพื่อให้ได้ผลตามที่ได้ออกแบบไว้

3. วงจรบางวงจรใช้แหล่งจ่ายแรงดันที่มีขนาดแรงดันสูง ทำให้ทำการทดลองอาจเกิดการเสียหายของวงจรง่ายต้องใช้ความระมัดระวังสูง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

แนวทางแก้ไข วงจรที่ต้องการแรงดันไฟฟ้าสูง เช่น วงจรขับรีเลย์ที่ต่อไปควบคุม การตัดต่อแรงดันไฟฟ้า 220 โวลต์ จึงต้องใช้ความระมัดระวังในการทดลอง

5.3 แนวทางการพัฒนา

1.พัฒนาทางด้านการออกแบบชุดทดลองให้มีความเหมาะสม และสะดวกในการใช้งานมากขึ้น

2. พัฒนาการวางอุปกรณ์บนแผ่นวงจรพิมพ์ให้มีความสวยงามมากขึ้น

3. พัฒนาทางด้านการออกแบบลายวงจร ให้มีความสวยงาม และไม่ซับซ้อน

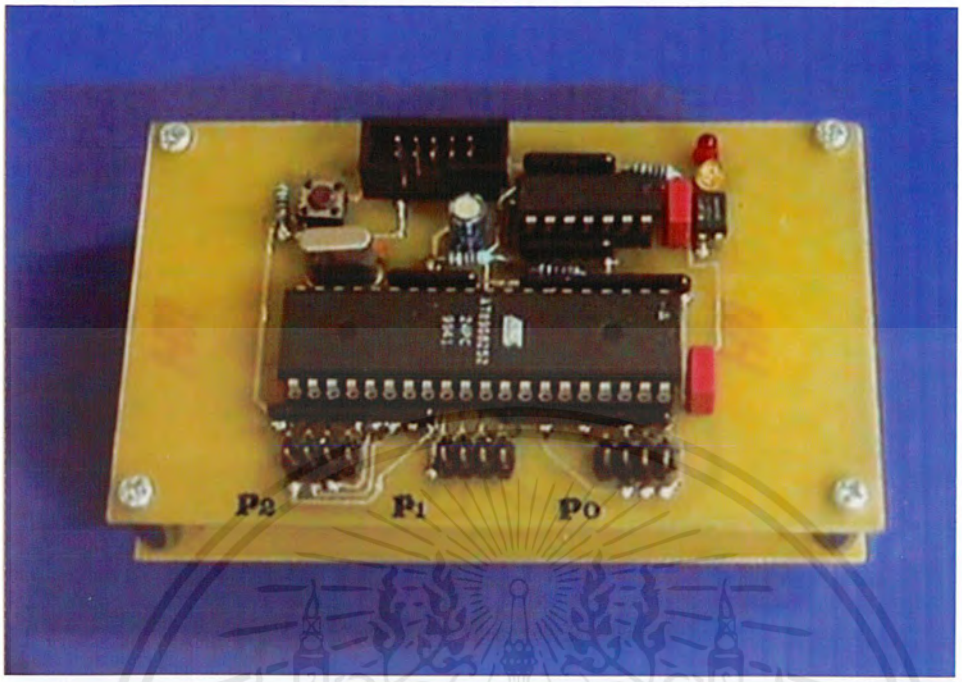
4. พัฒนาการออกแบบลายวงจร ให้มีขนาดเล็กลง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

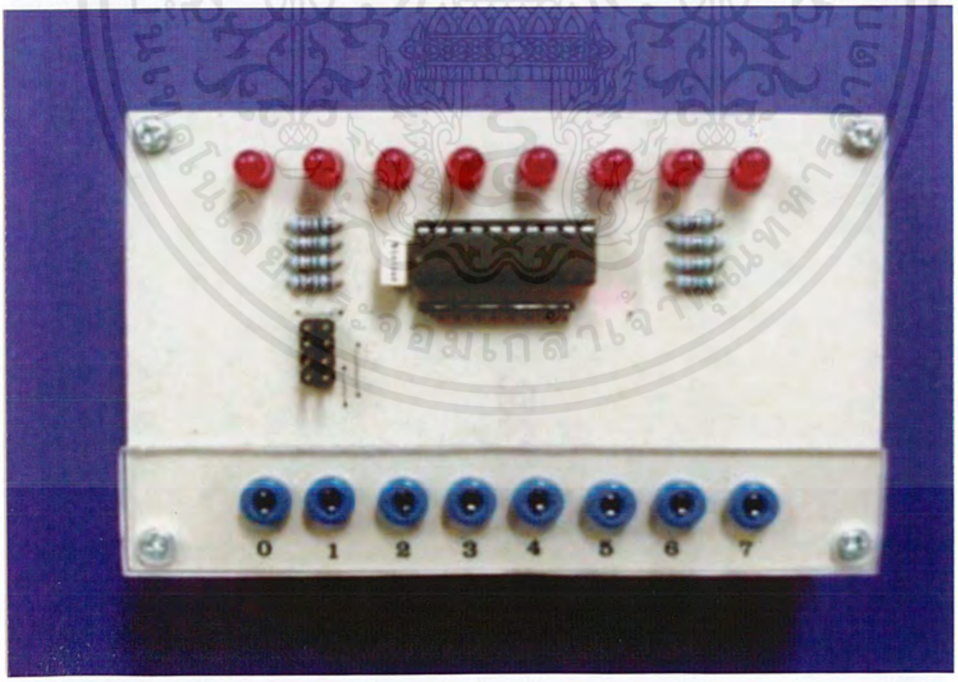
ภาคผนวก ก

เครื่องต้นแบบ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

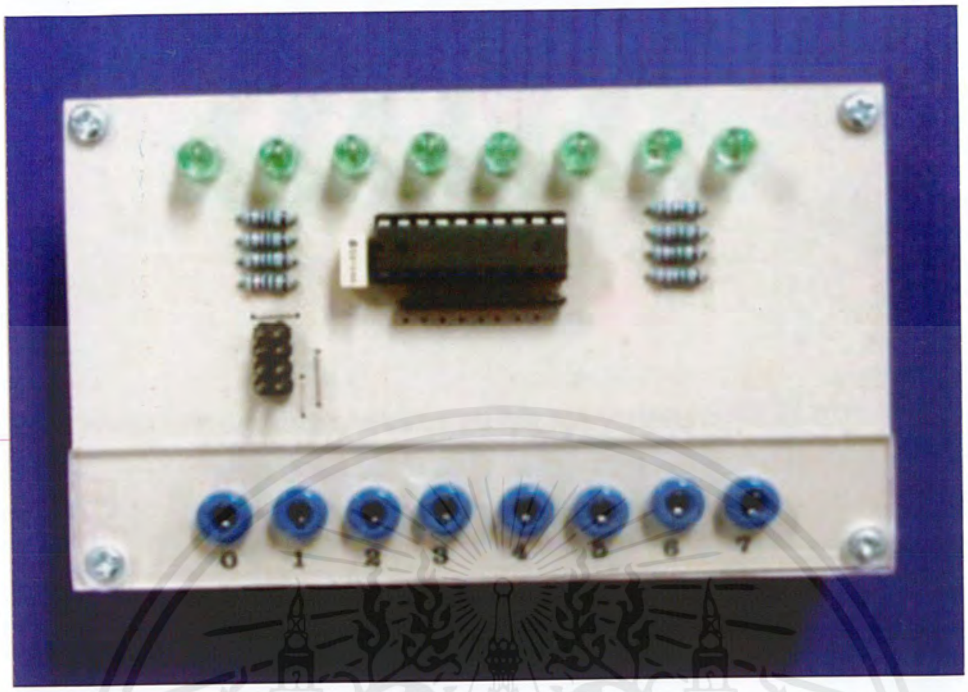


รูปที่ ก.1 วงจรควบคุม

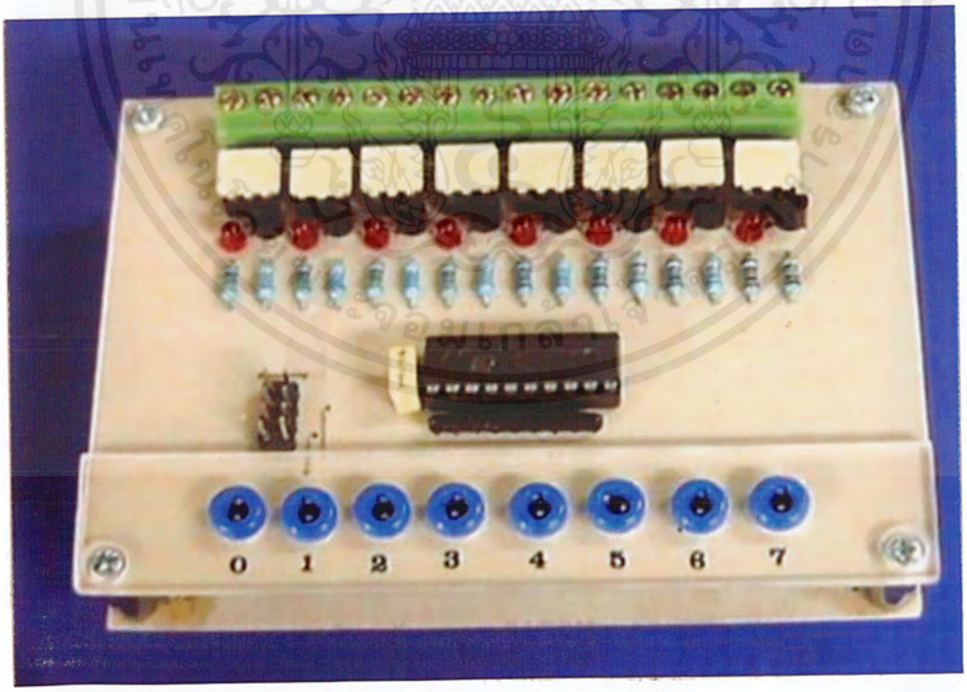


รูปที่ ก.2 วงจรภาคแสดงผลไดโอดเปล่งแสงแบบคาโถดร่วม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

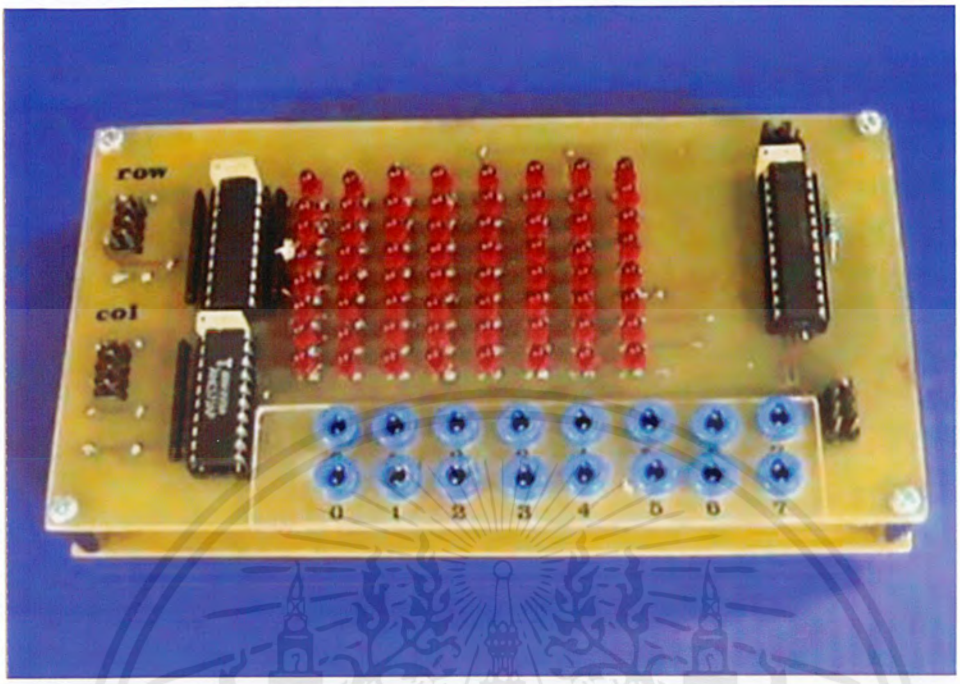


รูปที่ ก.3 วงจรภาคแสดงผลไดโอดเปล่งแสงแบบอาโนดร่วม

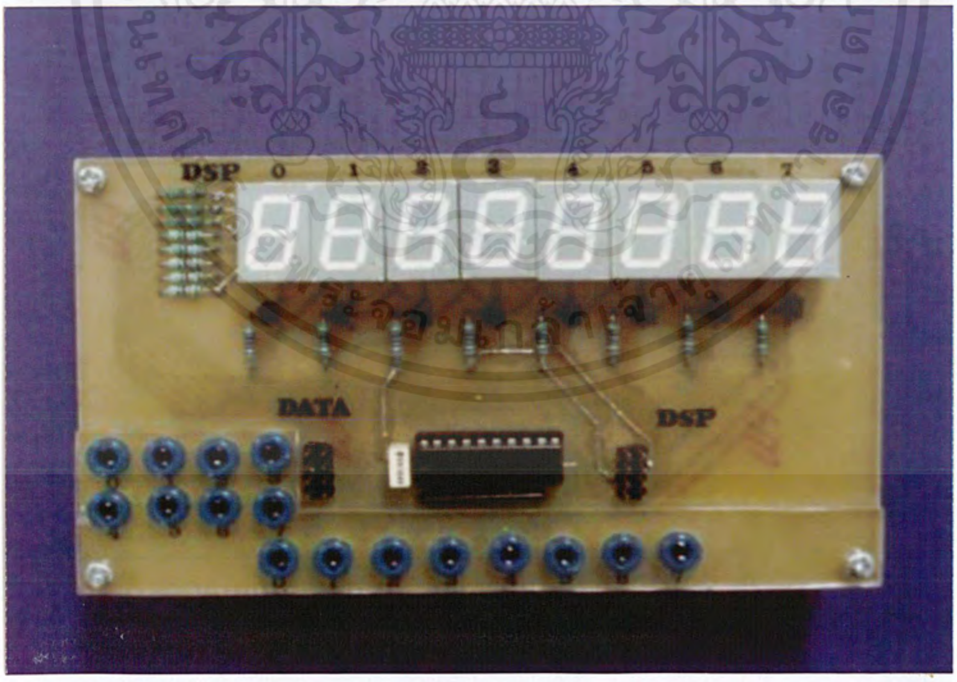


รูปที่ ก.4 วงจรเชื่อมโยงทางแสงออก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

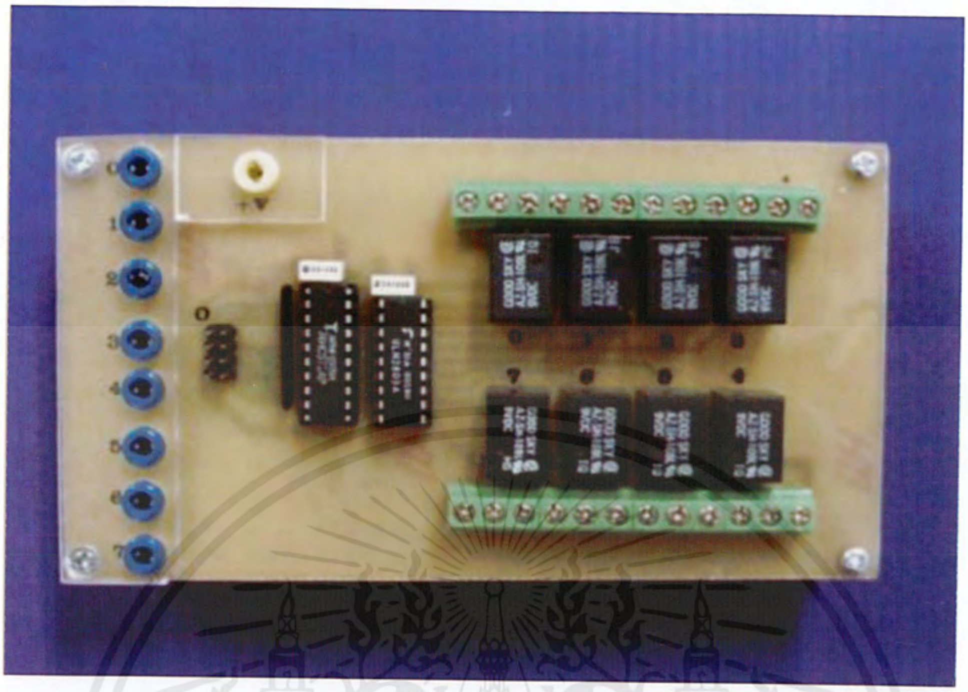


รูปที่ ก.5 วงจรแสดงผลแอลอีดีแบบเมตริกซ์

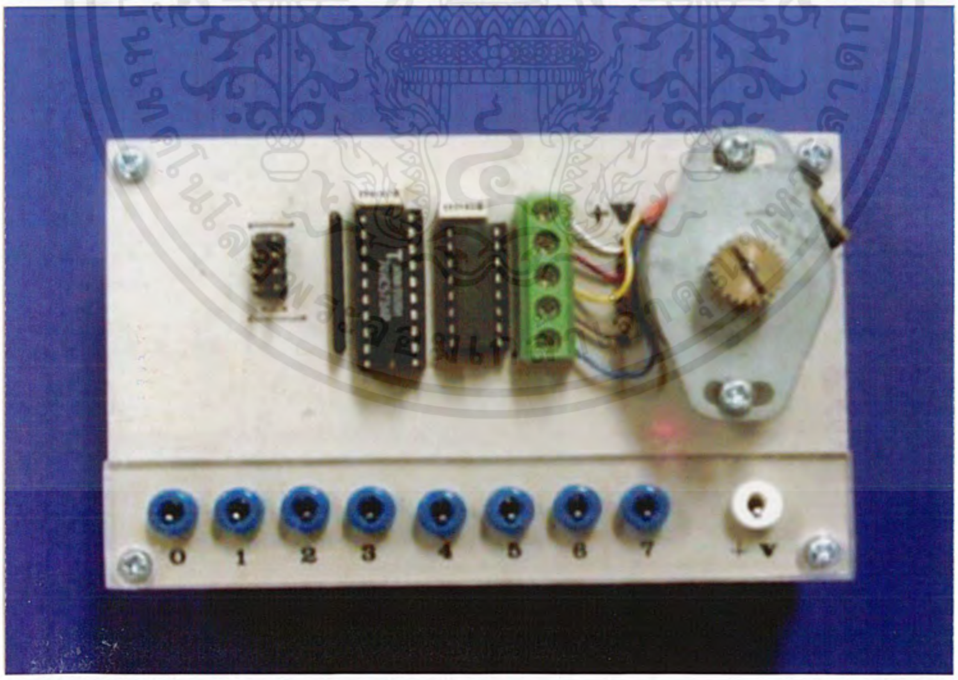


รูปที่ ก.6 วงจรแสดงผลแอลอีดี 7 ส่วน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

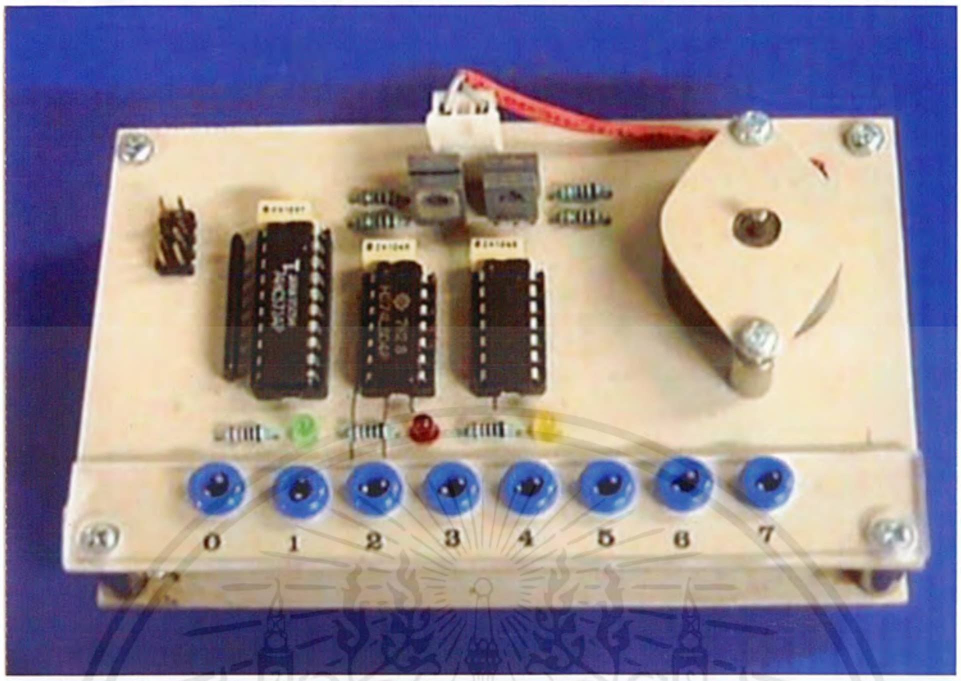


รูปที่ ก.7 วงจรควบคุมรีเลย์

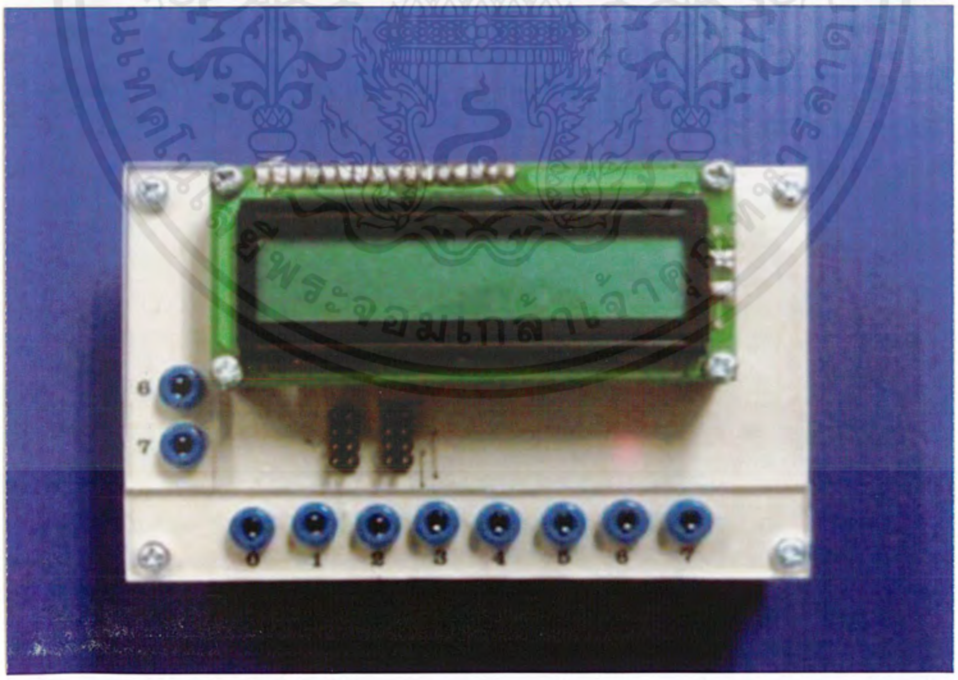


รูปที่ ก.8 วงจรควบคุมสเต็ปปิ้งมอเตอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

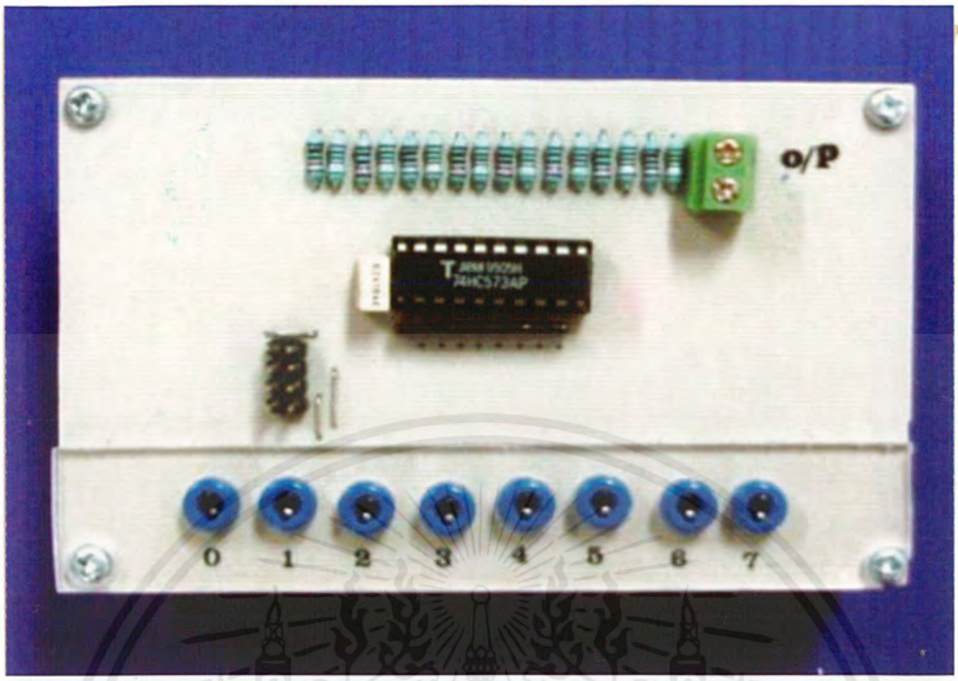


รูปที่ ก.9 วงจรควบคุมมอเตอร์ไฟฟ้ากระแสตรง

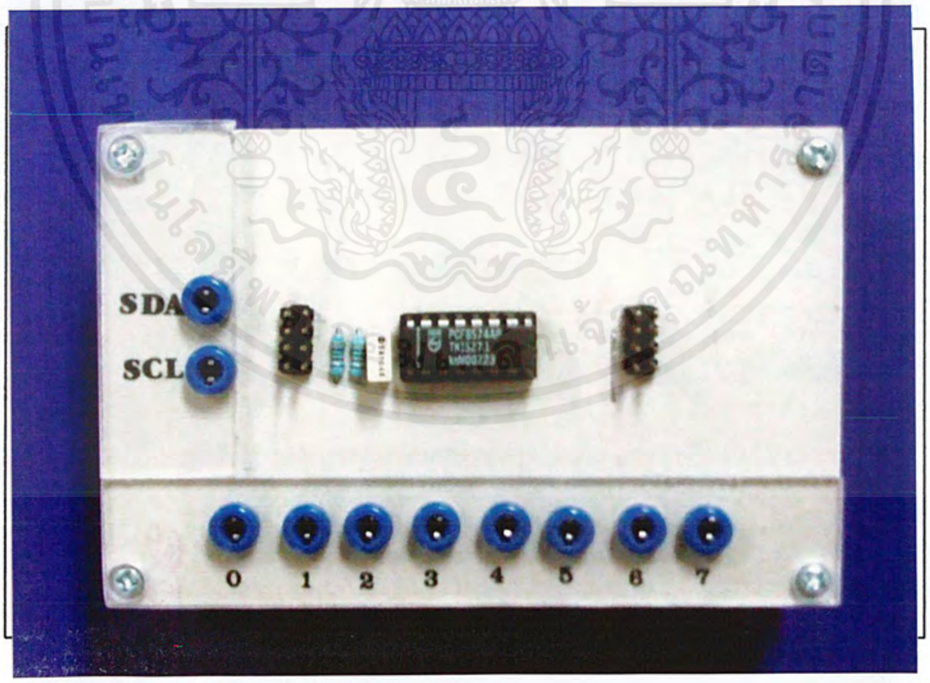


รูปที่ ก.10 วงจรแสดงผลแบบผลึกเหลว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

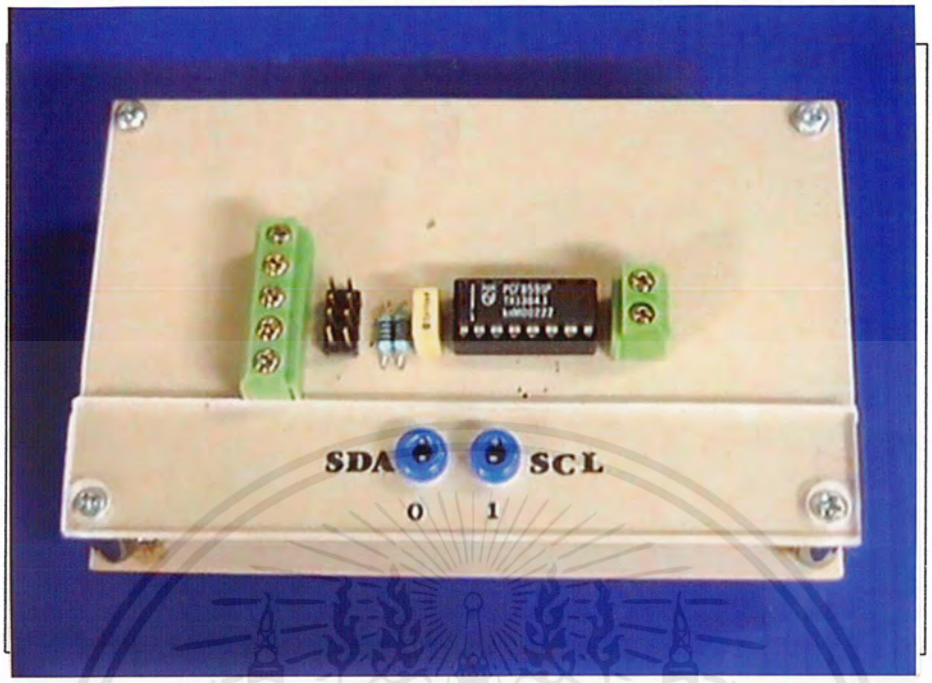


รูปที่ ก.11 วงจรแปลงสัญญาณดิจิตอลเป็นแอนะลอกแบบ R-2R

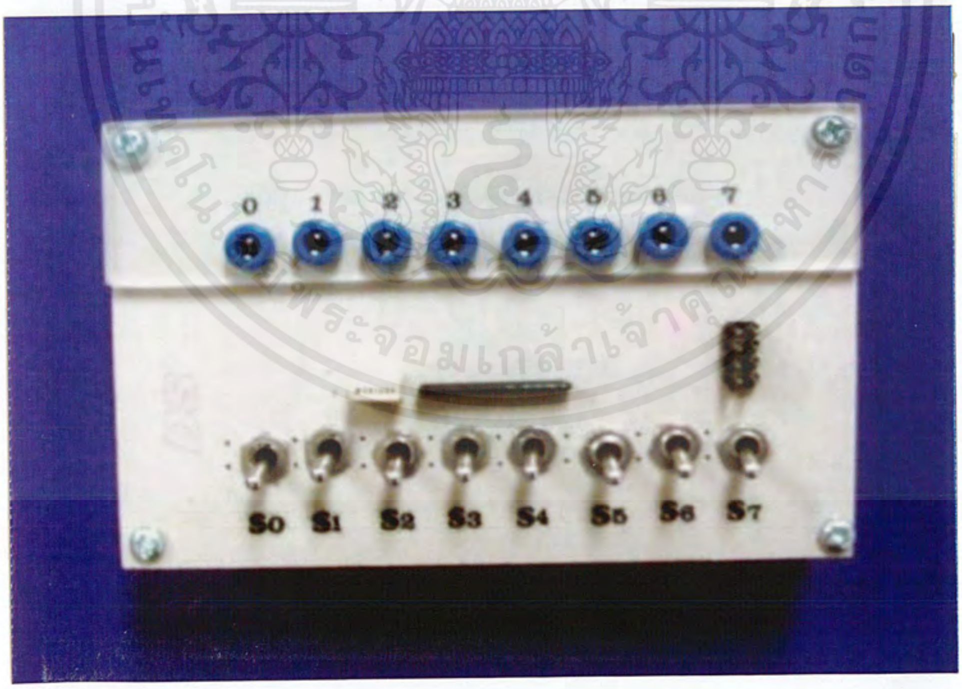


รูปที่ ก.12 วงจรเชื่อมต่ออุปกรณ์บนระบบบัส I^2C

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

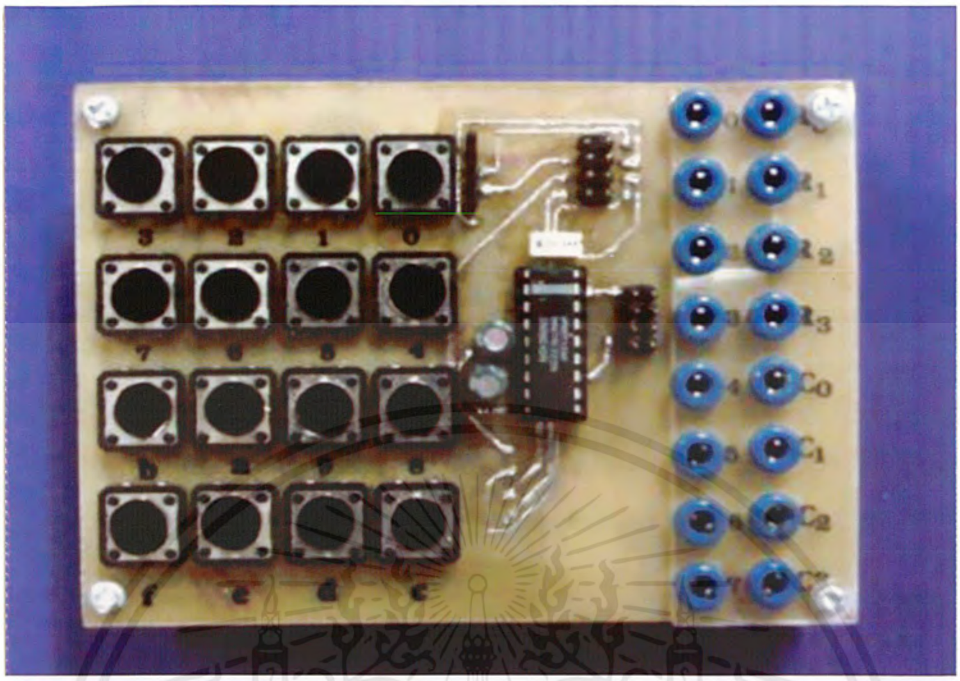


รูปที่ ก.13 วงจรแปลงสัญญาณแอนะลอกเป็นดิจิตอล

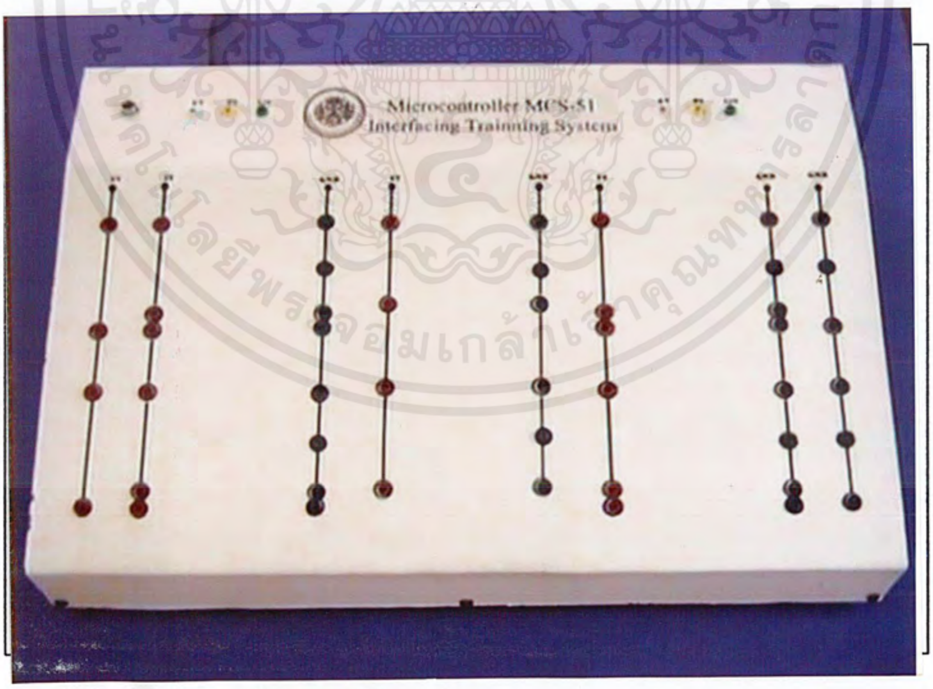


รูปที่ ก.14 วงจรควบคุมแบบสวิตซ์เดี่ยว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

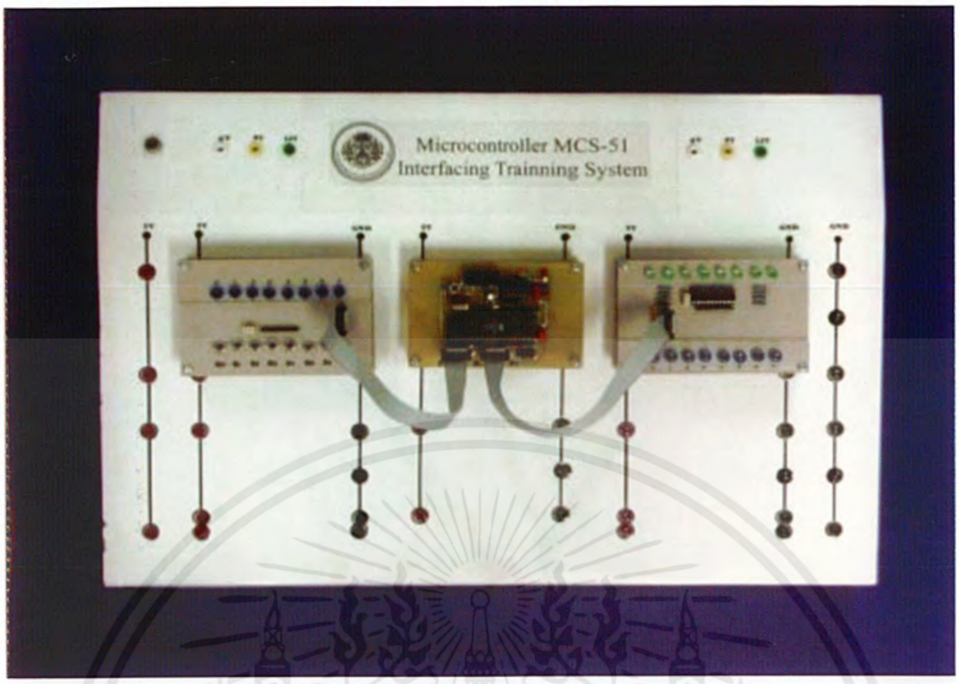


รูปที่ ก.15 วงจรควบคุมแบบคีย์สวิตซ์

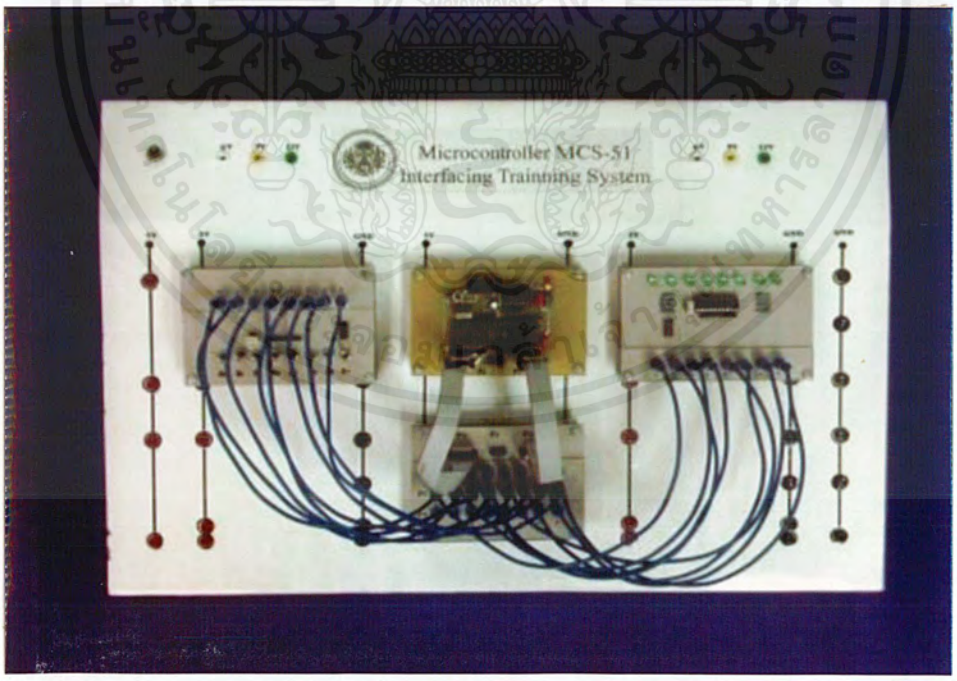


รูปที่ ก.16 ส่วนจ่ายแรงดัน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ ก.17 การเชื่อมต่อโดยใช้สาย FC-10P



รูปที่ ก.18 การเชื่อมต่อโดยใช้สายต่อ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก ข

รายการอุปกรณ์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รายการอุปกรณ์

1. วงจรแหล่งกำเนิดแรงดัน

ตารางที่ ข.1 รายการอุปกรณ์วงจรแหล่งกำเนิดแรงดัน

รายการอุปกรณ์	จำนวน (ตัว)
Pin 2 ขา	8
คอนเน็คเตอร์ 2 ขา	8
ขารองปริ้น 10 มิลลิเมตร	4
น็อตยึด	8
ไอซี 9422	1
ไอซี 7806	1
ไอซี 7809	1
ไอซี 7812	1
ตัวเก็บประจุ 0.1 ไมโครฟารัด WIMA	8
ตัวเก็บประจุ 470 ไมโครฟารัด 50 โวลต์	4
ไดโอด 1N4001	4
ไดโอดบริดจ์ 4 A	4
ฟิวส์ 3 A 250 โวลต์	1

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2. วงจรควบคุม

ตารางที่ ข.2 รายการอุปกรณ์วงจรควบคุม

รายการอุปกรณ์	จำนวน (ตัว)
Pin 10 ขา	4
ขารองปริ้น 15 มิลลิเมตร	4
น็อตยึด	8
ไอซี AT89S8252	1
ไอซี 74LS07	1
ซ็อกเกต 40 ขา	1
ซ็อกเกต 14 ขา	1
ตัวความต้านทาน 10 กิโลโอห์ม 9 ขา	3
ตัวความต้านทาน 10 กิโลโอห์ม 5 ขา	3
ตัวความต้านทาน 220 กิโลโอห์ม 1%	2
ตัวความต้านทาน 330 กิโลโอห์ม 1%	1
ตัวความต้านทาน 11 กิโลโอห์ม 1%	1
ตัวเก็บประจุ 0.1 ไมโครฟารัด	2
ตัวเก็บประจุ 22 pF	2
ตัวเก็บประจุ 4.7 ไมโครฟารัด 50 โวลต์	1
ออปโต้ PC817	1
คริสตอล 18.432 MHz	1
แอลอีดี 0.3 มิลลิเมตร	2
ไมโครสวิตช์ 4 ขา	1

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3. วงจรสวิตซ์เดี่ยว

ตารางที่ ข.3 รายการอุปกรณ์วงจรสวิตช์เดี่ยว

รายการอุปกรณ์	จำนวน (ตัว)
Pin 8 ขา	1
ขารองปริ้น 15 มิลลิเมตร	4
น็อตยึด	8
ตัวความต้านทาน 10 กิโลโอห์ม 9 ขา	1
ตัวเก็บประจุ 0.1 ไมโครฟารัด	1
สวิตช์โยก 3 ขา	8
แจ็คตัวเมีย	8

4. วงจรแสดงผลไดโอดเปล่งแสงแบบคาโถดร่วม

ตารางที่ ข.4 รายการอุปกรณ์วงจรแสดงผลไดโอดเปล่งแสงแบบคาโทดร่วม

รายการอุปกรณ์	จำนวน (ตัว)
Pin 8 ขา	1
ขารองปริ้น 15 มิลลิเมตร	4
น็อตยึด	8
ไอซี 74HC541	1
ซ็อกเกต 20 ขา	1
ตัวความต้านทาน 10 กิโลโอห์ม 9 ขา	1
ตัวความต้านทาน 220 โอห์ม 1%	8
ตัวเก็บประจุ 0.1 ไมโครฟารัด	1
แอลอีดี 5 มิลลิเมตร	8
แจ็คตัวเมีย	8

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5. วงจรแสดงผลไดโอดเปล่งแสงแบบอาโนดร่วม

ตารางที่ ข.5 รายการอุปกรณ์วงจรวงจรแสดงผลไดโอดเปล่งแสงแบบอาโนดร่วม

รายการอุปกรณ์	จำนวน (ตัว)
Pin 8 ขา	1
ขารองปริ้น 15 มิลลิเมตร	4
น็อตยึด	8
ไอซี 74HC541	1
ซ็อกเกต 20 ขา	1
ตัวความต้านทาน 10 กิโลโอห์ม 9 ขา	1
ตัวความต้านทาน 220 โอห์ม 1%	8
ตัวเก็บประจุ 0.1 ไมโครฟารัด	1
แอลอีดี 5 มิลลิเมตร	8
แจ็คตัวเมีย	8

6. วงจรแสดงผลแอลอีดีแบบ 7 ส่วน

ตารางที่ ข.6 รายการอุปกรณ์วงจรแสดงผลแอลอีดีแบบ 7 ส่วน

รายการอุปกรณ์	จำนวน (ตัว)
Pin 8 ขา	2
ขารองปริ้น 15 มิลลิเมตร	4
น็อตยึด	8
ไอซี 74HC541	1
ซ็อกเกต 40 ขา	2
ซ็อกเกต 20 ขา	1
ทรานซิสเตอร์ C557	8

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ ข.6 (ต่อ) รายการอุปกรณ์วงจรแสดงผลแอลอีดีแบบ 7 ส่วน

รายการอุปกรณ์	จำนวน (ตัว)
ตัวความต้านทาน 10 กิโลโอห์ม 9 ขา	1
ตัวความต้านทาน 47 โอห์ม 1%	8
ตัวความต้านทาน 1 กิโลโอห์ม 1%	8
ตัวเก็บประจุ 0.1 ไมโครฟารัด	1
7 SEGMENT คอมมอนคาโถด	8
แจ็คตัวเมีย	8

7.วงจรแสดงผลแบบผลึกเหลว

ตารางที่ ข.7 รายการอุปกรณ์วงจรแสดงผลแบบผลึกเหลว

รายการอุปกรณ์	จำนวน (ตัว)
Pin 8 ขา	2
ขารองปริ้น 15 มิลลิเมตร	4
น็อตยึด	8
ไอซี 74HC541	1
ซ็อกเกต 20 ขา	1
ตัวความต้านทาน 10 กิโลโอห์ม 1%	1
ตัวความต้านทานปรับค่า 10 กิโลโอห์ม	1
ตัวเก็บประจุ 0.1 ไมโครฟารัด	1
จอแสดงผลแบบผลึกเหลว 16 x 1	1
แจ็คตัวเมีย	10

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

8. วงจรควบคุมสเต็ปปิ้งมอเตอร์

ตารางที่ ข. 8 รายการอุปกรณ์วงจรควบคุมสเต็ปปิ้งมอเตอร์

รายการอุปกรณ์	จำนวน (ตัว)
Pin 8 ขา	1
ขารองปริ้น 15 มิลลิเมตร	4
น็อตยึด	8
ไอซี ULN2003	1
ไอซี 74HC573	1
ซ็อกเกต 16 ขา	1
ซ็อกเกต 20 ขา	1
ตัวความต้านทาน 10 กิโลโอห์ม 9 ขา	1
ตัวเก็บประจุ 0.1 ไมโครฟารัด	2
Terminal	2
สเต็ปปิ้งมอเตอร์	1
แจ็คตัวเมีย	9

9. วงจรมอเตอร์ไฟฟ้ากระแสตรง

ตารางที่ ข. 9 รายการอุปกรณ์วงจรควบคุมมอเตอร์ไฟฟ้ากระแสตรง

รายการอุปกรณ์	จำนวน (ตัว)
Pin 8 ขา	1
ขารองปริ้น 15 มิลลิเมตร	4
น็อตยึด	8
ไอซี 74LS00	1

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ ข. 9 (ต่อ) รายการอุปกรณ์วงจรควบคุมมอเตอร์ไฟฟ้ากระแสตรง

รายการอุปกรณ์	จำนวน (ตัว)
ไอซี 74LS04	1
ไอซี 74HC573	1
ซ็อกเกต 14 ขา	2
ซ็อกเกต 20 1ขา	1
ตัวความต้านทาน 10 กิโลโอห์ม 9 ขา	1
ตัวความต้านทาน 560โอห์ม 1%	4
ตัวความต้านทาน 330 โอห์ม 1%	3
ตัวเก็บประจุ 0.1 ไมโครฟารัด	3
แอลอีดี 0.3 มิลลิเมตร เขียว	1
แอลอีดี 0.3 มิลลิเมตร เหลือง	1
แอลอีดี 0.3 มิลลิเมตร แดง	1
ทรานซิสเตอร์ BD140	4
มอเตอร์ไฟฟ้ากระแสตรง 6โวลต์	1
แจ็คตัวเมีย	8

10. วงจรแปลงสัญญาณแอนะลอกเป็นสัญญาณดิจิตอล

ตารางที่ ข. 10 รายการอุปกรณ์วงจรแปลงสัญญาณแอนะลอกเป็นสัญญาณดิจิตอล

รายการอุปกรณ์	จำนวน (ตัว)
Pin 8 ขา	1
ขารองปริ้น 15 มิลลิเมตร	4
น็อตยึด	8
ไอซี PCF8591	1

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ ข.10 รายการอุปกรณ์แปลงสัญญาณแอนะลอกเป็นสัญญาณดิจิตอล

รายการอุปกรณ์	จำนวน (ตัว)
ซ็อกเกต 16 ขา	1
ตัวความต้านทาน 4.7 กิโลโอห์ม 1%	2
ตัวเก็บประจุ 0.1 ไมโครฟารัด	1
Terminal	3
แจ็คตัวเมีย	2

11. วงจรแปลงสัญญาณดิจิตอลเป็นสัญญาณแอนะลอก

ตารางที่ ข. 11 รายการอุปกรณ์วงจรแปลงสัญญาณดิจิตอลเป็นสัญญาณแอนะลอก

รายการอุปกรณ์	จำนวน (ตัว)
Pin 8 ขา	1
ขารองปริ้น 15 มิลลิเมตร	4
น็อตยึด	8
ไอซี 74HC573	1
ซ็อกเกต 20 ขา	1
ตัวความต้านทาน 10 กิโลโอห์ม 9 ขา	1
ตัวความต้านทาน 10 กิโลโอห์ม 1%	8
ตัวความต้านทาน 20 กิโลโอห์ม 1%	8
ตัวเก็บประจุ 0.1 ไมโครฟารัด	1
เทอร์มินอล	1
แจ็คตัวเมีย	8

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

12. วงจรแสดงผลแบบแอลอีดีเมตริกซ์ 8x8

ตารางที่ ข. 12 รายการอุปกรณ์วงจรแสดงผลแบบ แอลอีดีเมตริกซ์ 8x8

รายการอุปกรณ์	จำนวน (ตัว)
Pin 3 ขา	1
Pin 8 ขา	3
ขารองปริ้น 15 มิลลิเมตร	4
น็อตยึด	8
ไอซี 74HC573	1
ไอซี MAX7219	1
ซ็อกแกต 20 ขา	1
ซ็อกเกต 24 ขา	1
ตัวความต้านทาน 10 กิโลโอห์ม 9 ขา	3
ตัวความต้านทาน 30 กิโลโอห์ม 1%	1
ตัวเก็บประจุ 0.1 ไมโครฟารัด	3
แอลอีดี 0.3มิลลิเมตร แดง	64
แจ็คตัวเมีย	16

13. วงจรคีย์สวิตช์

ตารางที่ ข. 13 รายการอุปกรณ์วงจรคีย์สวิตช์

รายการอุปกรณ์	จำนวน (ตัว)
Pin 8 ขา	2
ขารองปริ้น 15 มิลลิเมตร	4
น็อตยึด	8

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ ข. 13 (ต่อ) รายการอุปกรณ์วงจรคีย์สวิตช์

รายการอุปกรณ์	จำนวน (ตัว)
ไอซี 74C922	1
ซ็อกเกต 18 ขา	1
ตัวความต้านทาน 10กิโลโอห์ม 5ขา	1
ตัวเก็บประจุ 0.1 ไมโครฟารัด	1
ตัวเก็บประจุ 0.1 ไมโครฟารัด 50 โวลต์	2
สวิตช์กดติดปล่อยดับ	16
แจ็คตัวเมีย	16

14. วงจรเชื่อมต่อแบบบัส I^2C

ตารางที่ ข. 14 รายการอุปกรณ์วงจรการเชื่อมต่อแบบบัส I^2C

รายการอุปกรณ์	จำนวน (ตัว)
Pin 8 ขา	2
ขารองปริ้น 15 มิลลิเมตร	4
น็อตยึด	8
ไอซี PCF8574	1
ซ็อกเกต 16 ขา	1
ตัวความต้านทาน 4.75 กิโลโอห์ม 1%	2
แจ็คตัวเมีย	16

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

15. วงจรวงจรเชื่อมโยงทางแสง

ตารางที่ ข. 15 รายการอุปกรณ์วงจรเชื่อมโยงทางแสง

รายการอุปกรณ์	จำนวน (ตัว)
Pin 8 ขา	1
ขารองปริ้น 15 มิลลิเมตร	4
น็อตยึด	8
ไอซี 74HC541	1
ซ็อกเกต 20 ขา	1
ซ็อกเกต 6 ขา	8
ออปโต้ 4N26	8
ทรานซิสเตอร์ C 557	8
ตัวความต้านทาน 10 กิโลโอห์ม 9 ขา	1
ตัวความต้านทาน 200 โอห์ม 1%	8
ตัวความต้านทาน 4.7 กิโลโอห์ม 1%	8
ตัวเก็บประเก็บ 0.1 ไมโครฟารัด	1
เทอร์มินอล	8
แอลอีดี 3 มิลลิเมตร แดง	8
แจ็คตัวเมีย	8

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

16. วงจรควบคุมรีเลย์

ตารางที่ ข. 16 รายการอุปกรณ์วงจรควบคุมรีเลย์

รายการอุปกรณ์	จำนวน (ตัว)
Pin 8 ขา	1
ขารองปริ้น 15 มิลลิเมตร	4
น็อตยึด	8
ไอซี ULN2803	1
ไอซี 74HC573	1
ซ็อกเกต 18 ขา	1
ซ็อกเกต 20 ขา	1
ตัวความต้านทาน 10 กิโลโอห์ม 9 ขา	1
ตัวเก็บประจุ 0.1 ไมโครฟารัด	2
รีเลย์ 9 โวลต์	8
เทอร์มินอล	8
แจ็คตัวเมีย เหลือง	1
แจ็คตัวเมีย น้ำเงิน	8

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก ค

ใบงานการทดลอง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ใบงานที่ 1

การใช้งานซอฟต์แวร์สำหรับทดลองฮาร์ดแวร์ (wins8252)

วัตถุประสงค์

1.เพื่อศึกษาโปรแกรมการทดลองฮาร์ดแวร์ชุดทดลองไมโครคอนโทรลเลอร์กับอุปกรณ์ภายนอก

2. เพื่อศึกษาคำสั่งในการใช้โปรแกรม **wins8252**

ทฤษฎีเบื้องต้น

การเขียนโปรแกรมภาษาแอสเซมบลี

การเรียนรู้เพื่อใช้งานไมโครคอนโทรลเลอร์ไม่ว่าจะเป็นเบอร์ใดก็ตาม สิ่งที่สำคัญในลำดับต่อมาหลังจากที่ทำความเข้าใจโครงสร้างทางฮาร์ดแวร์แล้ว ก็คือ การเขียนโปรแกรมเพื่อกำหนดให้ไมโครคอนโทรลเลอร์ทำงาน ข้อมูลของโปรแกรมที่ไมโครคอนโทรลเลอร์ต้องการจะอยู่ในรูปของรหัสเลขฐานสิบหก เรียกโดยทั่วไปว่าภาษาเครื่อง (แมชีนโค้ด) แต่เนื่องจากการเขียนโปรแกรมในลักษณะที่เป็นภาษาเครื่องนี้ ผู้เขียนโปรแกรมต้องทำการเปิดตารางรหัสคำสั่งซึ่งเป็นเรื่องที่ยุ่งยากและทำให้การตรวจสอบโปรแกรมที่เขียนขึ้นกระทำได้ยาก จึงหันมาใช้การเขียนโปรแกรมด้วยภาษาแอสเซมบลี แล้วใช้ซอฟต์แวร์แอสเซมเบลอร์ ทำการแปลภาษาแอสเซมบลีที่เขียนขึ้นเป็นภาษาเครื่อง แล้วเขียนลงในหน่วยความจำโปรแกรมของไมโครคอนโทรลเลอร์

การเขียนโปรแกรมควบคุมการทำงานของไมโครคอนโทรลเลอร์ นิยมใช้ภาษาแอสเซมบลีเนื่องจากสามารถศึกษาและตรวจสอบผลการทำงานได้อย่างชัดเจน

โครงสร้างของโปรแกรมภาษาแอสเซมบลี

ประกอบด้วย 4 ส่วนหลักคือ

1) ลาเบล ใช้ในการอ้างถึงบรรทัดหนึ่งของโปรแกรมที่ทำการเขียนขึ้น

2) รหัสนีโมนิก เป็นส่วนแสดงคำสั่งของไมโครคอนโทรลเลอร์ที่ต้องการให้กระทำ

3)โอเปอร์แรนด์ เป็นส่วนที่แสดงถึงตัวกระทำหรือถูกกระทำและข้อมูลที่ใช้ในการกระทำตามคำสั่งที่กำหนดโดยรหัสนีโมนิกก่อนหน้านี้

4) คอมเมนต์ เป็นส่วนที่ผู้เขียนโปรแอรมเขียนขึ้นเพื่อใช้ในการอธิบายคำสั่งที่กระทำ หรือผลของการกระทำคำสั่งในบรรทัดหรือในโปรแกรมนั้นๆ ทั้งนี้เพื่อช่วยให้ผู้เขียนโปรแกรมสามารถ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตรวจสอบโปรแกรมที่เขียนขึ้นได้ง่ายรวมถึงเป็นประโยชน์ต่อผู้อื่นที่นำโปรแกรมที่เขียนขึ้นนี้ไปศึกษา

คำสั่งเทียม

ในการเขียนโปรแกรมภาษาแอสเซมบลีให้ได้ดีและสามารถตรวจสอบตลอดจนทำความเข้าใจได้ง่ายต้องอาศัยกลุ่มคำสั่งพิเศษอีกกลุ่มหนึ่งที่ไม่มีในชุดคำสั่งของไมโครคอนโทรลเลอร์จะเรียกกลุ่มคำสั่งนี้ว่า คำสั่งเทียม

เนื่องจากในโปรแกรมภาษาแอสเซมบลี จะมีการแบ่งส่วนเป็นโปรแกรมหลัก และโปรแกรมย่อยมากมายการใช้คำสั่งเทียมจะช่วยให้ผู้เขียนโปรแกรมสามารถตรวจสอบโปรแกรมได้อย่างละเอียดขณะเดียวกันยังช่วยให้การกำหนดชื่อหน้าที่ของพอร์ตหรือรีจิสเตอร์ต่างๆ สามารถกระทำได้อย่างชัดเจน และไม่สิ้นเปลืองหน่วยความจำ เนื่องจากการใช้คำสั่งเทียมทุกคำสั่งจะไม่ใช้พื้นที่หน่วยความจำโปรแกรมแต่อย่างใด เนื่องจากคำสั่งเทียมเป็นเพียงเครื่องมือที่ใช้ช่วยให้การอ้างถึงตำแหน่งต่างๆ หรือข้อมูลภายในโปรแกรมภาษาแอสเซมบลีเท่านั้น ซีพียูภายในไมโครคอนโทรลเลอร์มิได้กระทำคำสั่งเทียมนี้โดยตรง ทั้งนี้เนื่องจากคำสั่งเทียมไม่ได้เป็นส่วนหนึ่งของชุดคำสั่งของไมโครคอนโทรลเลอร์นั่นเอง

คำสั่งเทียมของโปรแกรมแอสเซมเบลอร์แต่ละตัวจะแตกต่างกันออกไป อย่างไรก็ตามสามารถสรุปคำสั่งเทียมที่มีการใช้งานกันโดยทั่วไป และโปรแกรมแอสเซมเบอร์ส่วนใหญ่สามารถแปลความหมายได้ ดังนี้

EQU

รูปแบบการใช้งาน

มาจากคำว่า EQUAL หมายถึง เท่ากับ ใช้ในการแทนตำแหน่งหรือค่าคงที่ด้วยชื่อที่สามารถเข้าใจได้ง่าย เช่น

```
OUT_PORT    EQU    PORT2
```

คือ กำหนดให้พอร์ต 2 มีชื่อว่า OUT_PORT ในกรณีผู้เขียนโปรแกรมแสดงให้เห็นว่าพอร์ต 2 ของไมโครคอนโทรลเลอร์ได้รับการกำหนดให้เป็นพอร์ตเอาต์พุต

คำสั่ง EQU นี้ควรแทรกหรือกำหนดไว้ตอนต้นของโปรแกรมเสมอ ซึ่งอาจต้องใช้หลายบรรทัด เมื่อใช้คำสั่ง EQU ในการแทนค่าตำแหน่งหรือข้อมูลแล้ว หากมีความต้องการเปลี่ยนแปลงค่าตำแหน่งหรือข้อมูลนั้น แต่ยังคงต้องการใช้ชื่อตัวแปรเดิมอยู่ สามารถทำได้ด้วยการแก้ไขที่คำสั่ง EQU เพียงตำแหน่งเดียว เมื่อแก้ไขแล้ว ในโปรแกรมบรรทัดที่มีการอ้างอิงถึงชื่อตัวแปรนั้นก็ได้รับการแก้ไขค่าทั้งหมดโดยอัตโนมัติ เป็นการเพิ่มความสะดวกในการแก้ไขโปรแกรมให้กับผู้เขียนโปรแกรม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ORG

รูปแบบการใช้งาน

มาจากคำว่า ORIGIN หมายถึงจุดเริ่มต้น คำสั่งนี้ใช้ในการกำหนดจุดเริ่มต้นหรือตำแหน่งเริ่มต้นของโปรแกรมหลัก, โปรแกรมย่อยเริ่มต้นการทำงาน และโปรแกรมย่อยบริการอินเตอร์รัปต์หน่วยความจำโปรแกรม เนื่องโปรแกรมเหล่านั้นจะมีตำแหน่งเริ่มต้นที่แน่นอน เช่น โปรแกรมหลักมักจะมีจุดเริ่มต้นที่ตำแหน่ง 0000H เสมอในขณะที่โปรแกรมย่อยบริการอินเตอร์รัปต์เนื่องจากพอร์ตอนุกรมมีการรับส่งข้อมูลจะมีตำแหน่งเริ่มต้นที่ 0023H

ในกรณีที่ใช้ชื่อลาเบลในการกำหนดตำแหน่งเริ่มต้นการทำงานต้องมีการกำหนดให้ทราบก่อนว่า ลาเบลนั้นใช้อ้างถึงตำแหน่งใดด้วยคำสั่ง EQU ก่อนที่จะใช้คำสั่ง ORG

DB (Define Byte)

รูปแบบการใช้งาน

ใช้ในการกำหนดค่าให้แก่หน่วยความจำข้อมูลในตำแหน่งที่อ้างอิงด้วยชื่อของลาเบล มักจะใช้ประโยชน์ในรูปของตารางข้อมูล เช่น ตารางข้อมูลการแสดงผลของแอลอีดี 7 ส่วน, ตารางข้อมูลของโน้ตดนตรี

คำสั่ง DB สามารถใช้กับข้อมูลขนาด 8 บิต และมักจะบรรจุอยู่ตอนท้ายของโปรแกรมเสมอ เช่น

```
LED_TABL : DB       3FH, 06H, 5BH, 4FH, 66H, 6DH, 7DH, 7FH, 6FH
```

จากตัวอย่างเป็นการกำหนดข้อมูลการแสดงผลของ แอลอีดี 7 ส่วน โดยที่ลาเบล LED_TABL บรรจุข้อมูลสำหรับแสดงตัวเลข 0-9 เริ่มจากค่า 3FH เป็นข้อมูลสำหรับแสดงเลข 0 เรียงไปตามลำดับ ค่าสุดท้าย 6FH ก็จะเป็นข้อมูลสำหรับการแสดงเลข 9 ของแอลอีดี 7 ส่วนแบบคาโถดร่วม ถ้าหากข้อมูลที่เรียกใช้มีเครื่องหมาย '…' (… หมายถึงข้อมูล) ข้อมูลดังกล่าวจะเป็นข้อมูลตัวอักษรซึ่งแทนด้วยรหัสแอสกี้ของตัวอักษรตัวนั้นๆ

DW (Define Word)

รูปแบบการใช้งาน

ใช้ในการกำหนดค่าให้แก่หน่วยความจำข้อมูลในตำแหน่งที่อ้างอิงถึงด้วยชื่อของลาเบล มีการใช้งานเหมือนคำสั่ง DB ต่างกันเพียงขนาดของข้อมูล โดยคำสั่ง DW จะใช้ขนาดของข้อมูลเป็น 16 บิต หรือเรียกใช้ข้อมูลครั้งละ 2 ตัวอักษร

BIT

รูปแบบการใช้งาน

เป็นคำสั่งที่มีลักษณะคล้ายกับคำสั่ง EQU หากแต่จะใช้เรียกแทนข้อมูลในระดับบิต เช่น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

PORT_ENB BIT P1.0

เป็นการกำหนดให้บิต 0 ของรีจิสเตอร์ P1 หรือในรีจิสเตอร์พอร์ต 1 มีชื่อ PORT_ENB

ข้อควรระวังของคำสั่งเทียม BIT คือ ในโปรแกรมแอสเซมเบลอร์บางตัวอาจไม่รู้จักคำสั่งนี้

$ (string หรือ Dollar sign)

สัญลักษณ์นี้ใช้ในส่วนของโอเปอร์แรนด์เป็นแทนค่าในปัจจุบันของตำแหน่งที่บรรจุอยู่ใน รีจิสเตอร์ PC หรือโปรแกรมเคาน์เตอร์ มักใช้ร่วมกับคำสั่งการกระโดด เช่น

SJMP $

จากคำสั่งนี้ คือการให้ซีพียูกระโดดไปยังตำแหน่งปัจจุบันที่เก็บอยู่ในโปรแกรมเคาน์เตอร์ นั่นคือซีพียูจะทำงานวนอยู่ที่ตำแหน่งนี้ไม่รู้จบ ในโปรแกรมแอสเซมเบลอร์บางตัวก็ไม่รู้จักเครื่องหมายนี้ จึงอาจใช้แทนเครื่องหมาย $ ด้วยการเขียนโปรแกรม เช่น

END_LOOP : SJMP END_LOOP

END

เป็นคำสั่งที่ใช้แจ้งให้โปรแกรมแอสเซมเบลอร์ ทราบถึงบรรทัดสุดท้ายของโปรแกรมที่ต้องทำการแปลคำสั่ง ในโปรแกรมภาษาแอสเซมบลีทุกโปรแกรมต้องใส่คำสั่ง END นี้ไว้เป็นบรรทัดสุดท้ายเสมอ หากมีการเขียนโปรแกรมที่ต่อท้ายคำสั่ง END โปรแกรมแอสเซมเบลอร์จะไม่ทำการแปลโปรแกรมที่ต่อท้ายคำสั่ง END อย่างเด็ดขาด แม้ว่าจะเขียนถูกต้องไม่ผิดหลักไวยกรณ์หรือรูปแบบก็ตาม ในทางตรงกันข้ามก็จะไม่มีการแจ้งเตือนว่ามีการเขียนโปรแกรมผิดพลาดขึ้นในส่วนของโปรแกรมที่ต่อท้ายคำสั่ง END

ข้อกำหนดที่สำคัญของการเขียนโปรแกรมภาษาแอสเซมบลีของไมโครคอนโทรลเลอร์

มีด้วยกัน 4 ประการคือ

1) การตีความระหว่างค่าของตัวเลข และตัวอักษร โดยปกติหากมีการเขียนตัวเลขลงในโปรแกรมความหมายของตัวเลขนั้นคือค่าของตัวเลขฐานสิบ แต่ถ้าหากต้องการกำหนดให้เป็นเลขฐานอื่นต้องมีการเขียนตัวอักษรต่อท้ายดังนี้

B สำหรับเลขฐาน 2 ค่าของตัวเลขมีได้เพียง 2 ตัวคือ 0 และ 1 เช่น 11001001B

O หรือ Q สำหรับเลขฐานแปด ค่าตัวเลขมีตั้งแต่ 0-7 เช่น 1267Q หรือ 1267O

D สำหรับเลขฐานสิบ เช่น 4598

H สำหรับเลขฐานสิบหก ค่าของตัวเลขมีตั้งแต่ 0-F เช่น 123AH, 0FE7H ในกรณีที่เขียนข้อมูลที่ปรากฏเป็นข้อมูลตัวเลขฐานสิบหก หรือตัวอักษรชื่อลาเบลและชื่อหรือตัวแปร

2) ในการกำหนดชื่อหรือลาเบล ต้อต่อด้วยเครื่องหมาย : (โคลอน)เสมอ ในขณะที่หากใช้คำสั่ง EQU ที่ชื่อตัวแปรต้องมีเครื่องหมาย : ต่อท้าย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3) การเว้นช่องระหว่างชื่อลาเบล รหัสคำสั่งนีโมนิก โอเปอร์แรนด์ และคอมเมนต์ ควรใช้การกดปุ่ม TAB

4) ในส่วนของคอมเมนต์ หรือส่วนของโปรแกรมที่ไม่ต้องการให้มีการแอสเซมเบลอร์ต้องใส่เครื่องหมาย : หรือเซมิโคลอนนำหน้าส่วนหรือข้อความนั้นเสมอ

ในการทดลองฮาร์ดแวร์ต้องใช้โปรแกรม 2 โปรแกรม คือ **SXA51** ซึ่งเป็นโปรแกรมครอสแอสแซมเบลอร์ **(cross assembler)** สำหรับแปลโปรแกรมภาษาแอสเซมบลี **source.Asm** เป็น **object.Hex** เพื่อเขียนข้อมูลลงในหน่วยความจำโปรแกรมไมโครคอนโทรลเลอร์ AT89S8252 และโปรแกรม **wins8252** ซึ่งใช้สำหรับเขียนข้อมูลลงในหน่วยความจำโปรแกรม
ไมโครคอนโทรลเลอร์ AT89S8252

โปรแกรม CP-S8252

เป็นซอฟแวร์ที่ใช้ในการโปรแกรมข้อมูลลงในหน่วยความจำโปรแกรมไมโครคอนโทรลเลอร์ AT89S8252 บนบอร์ดคอนโทรลเลอร์ทางพอร์ต SPI ด้วยการใช้งานแบบนี้ทำให้ผู้ใช้งานไม่จำเป็นต้องถอดไมโครคอนโทรลเลอร์ออกมาจากบอร์ดการทดลองมายังเครื่องโปรแกรมภายนอก โปรแกรมนี้สามารถ Down load โปรแกรมแบบ SPI

คุณสมบัติของโปรแกรม wins8252

1) สามารถทำงานบนระบบปฏิบัติการวินโดว์

2) ติดต่อข้อมูลจาก PC ผ่านทาง Printer Port ลงในหน่วยความจำไมโครคอนโทรลเลอร์

3) โปรแกรมที่ทำงานบนระบบดอส จะทำงานแบบ Command Line ซึ่งมีการใช้งานง่าย

4)โปรแกรมจะลบอัตโนมัติทุกครั้งที่มีการโปรแกรมลงในหน่วยความจำ

5) รับข้อมูล Hex ในรูปแบบ Intel Hex

เครื่องมือและอุปกรณ์

1) บอร์ดไมโครคอนโทรลเลอร์ MCS51-S8252

2) คอมพิวเตอร์ที่ติดตั้งวินโดว์ 95 หรือ 98

3) โปรแกรม wins8252

ขั้นตอนการใช้งาน

1) ต่อสาย Down load เข้ากับ Connector 10 Pin โดยต่อสายให้ถูกต้องตามตำแหน่งขา Connector

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2) ต่อสาย Down load อีกด้านหนึ่งเข้ากับ Printer Port ของ PC

3) คัดลอกไฟล์ **wins8252** ลงในไดร์ฟที่ต้องการ (ส่วนโปรแกรมที่ใช้บนระบบปฏิบัติการวินโดว์ ชื่อว่า **wins8252**)

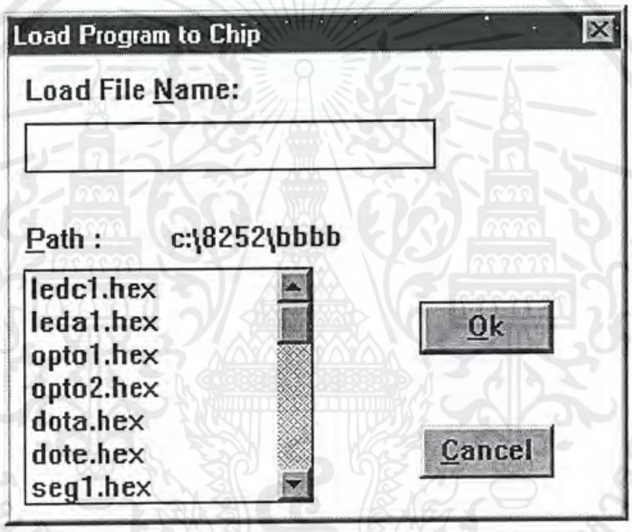
4) เตรียมระบบโดยการต่อบอร์ดไมโครคอนโทรลเลอร์กับ Power Supply

5) เรียก **wins8252.Exe** ในวินโดว์

6) เลือกโปรแกรมจากเมนู

7) ใส่ชื่อ **File.Hex** ที่ต้องการ Down load โดย FILE จะต้องมี รูปแบบเป็น Intel Hex

8) กดปุ่ม Ok



รูปที่ 1.1 โปรแกรม wins8252 ในวินโดว์

Menu Erase : เป็นเมนูที่ใช้ลบข้อมูลในไมโครคอนโทรลเลอร์โดยไม่มีเขียนข้อมูลใดๆลงในหน่วยความจำ

Menu Security : เป็นเมนูที่ใช้ป้องกันการอ่านข้อมูลออกมาอีกในโปรแกรม

Menu Option : เป็นเมนูที่ใช้เลือก LPT (LPT1 or LPT2) ที่ใช้ดาวน์โหลดข้อมูลและใช้เลือกค่า Delay ในกรณีเครื่องPCมีความเร็วมากเกินไปจนไม่สามารถดาวน์โหลดโปรแกรมลงในหน่วยความจำไมโครคอนโทรลเลอร์ได้

โปรแกรมจะทำการ Autoerase ทุกครั้งที่มีการเขียนข้อมูลลงในไมโครคอนโทรลเลอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การสร้างไฟล์เป็นนามสกุล .Asm โดยใช้โปรแกรม Edit

1) ทำการเรียกระบบปฏิบัติการ MS-DOS Prompt สร้างโฟล์เดอร์พร้อมตั้งชื่อไฟล์

C:\>CD S8252 <enter>

C:\s8252>Edit Lab1.Asm <enter>

2) จากนั้นทำการเขียนโปรแกรมที่ต้องการ

3) ทำการบันทึกไฟล์ดังกล่าวเป็นนามสกุล .Asm

4) จากนั้นก็ Exit ออกจากหน้าจอ Edit

5) นำไฟล์สร้างไว้ไปทำการแอสแซมเบลอร์ไฟล์ จาก File.Asm เป็น File.Hex

การแอสแซมเบลอร์ไฟล์ จาก File.Asm เป็น File.Hex

1) เรียกระบบปฏิบัติการ MS-DOS Prompt

2) เรียกไฟล์ Sxa51 ตามด้วยชื่อไฟล์ที่ตั้งชื่อไว้ (File.Asm)

3) กดปุ่ม Enter ต่อจากนั้น File.Hex ก็จะไปปรากฏอยู่ในโปรแกรม Wins8252.Exe

```
Microsoft(R) Windows 98
(C)Copyright Microsoft Corp 1981-
1998.
C:\WINDOWS>CD\
C:\>CD S8252
C:\s8252>SXA51 FILE.ASM
8051 Cross-Assembler (1.3)
Copyright (c) 1987, 1989
Binary Technology, Inc. Meriden, NH
No errors detected
Object file size: 51 bytes
Program entry address: 0000 (Hex)
C:\s8252>
```

สรุปลำดับขั้นการทดลอง

1) เขียนโปรแกรมภาษาแอสเซมบลี ด้วย Notepad หรือ Edit ในระบบปฏิบัติการดอสและบันทึกไฟล์เป็นนามสกุล Hex

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2) ทำการแอสเซมเบลอร์ไฟล์นามสกุล .Asm เป็นนามสกุล .Hex ถ้าหากมีข้อผิดพลาดให้แก้ไขจนกว่าการแอสเซมเบลอร์ไม่แจ้งว่าผิดพลาด

3) ทำการเชื่อมต่อบอร์ดคอนโทรลเลอร์กับบอร์ดเชื่อมต่อที่ต้องการทดลองกับชุด Power Supply

4) เปิดโปรแกรม Wins8252

5) เรียกไฟล์โปรแกรมที่เป็น .Hex แล้วกดปุ่มใน Ok

6) รันโปรแกรมโดยการกดปุ่มสวิตช์รีเซตที่บอร์ดคอนโทรลเลอร์

การต่อสาย DB-25 กับ FC-10P

การต่อสาย DB-25 กับ FC-10P มีดังต่อไปนี้

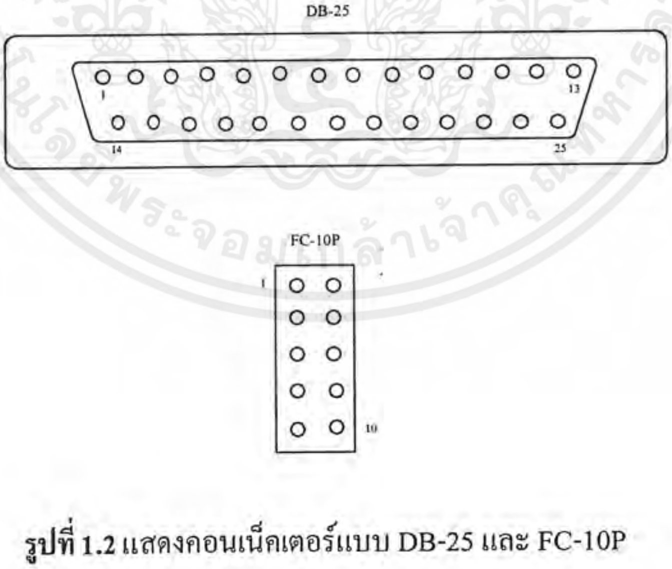
1) ขา 1 ของ DB-25 ต่อกับขา 8 ของ FC-10P (MOSI)

2) ขา 2 ของ DB-25 ต่อกับขา 10 ของ FC-10P (SCK)

3) ขา 3 ของ DB-25 ต่อกับขา 7 ของ FC-10P (RES)

4) ขา 10 ของ DB-25 ต่อกับขา 9 ของ FC-10P (MISO)

5) ขา 18 ของ DB-25 ต่อกับขา 1 ของ FC-10P (GND)



รูปที่ 1.2 แสดงคอนเน็คเตอร์แบบ DB-25 และ FC-10P

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ใบงานที่ 2
การเชื่อมต่อแอลอีดีแบบอาโนดร่วม และคาโถดร่วม

วัตถุประสงค์

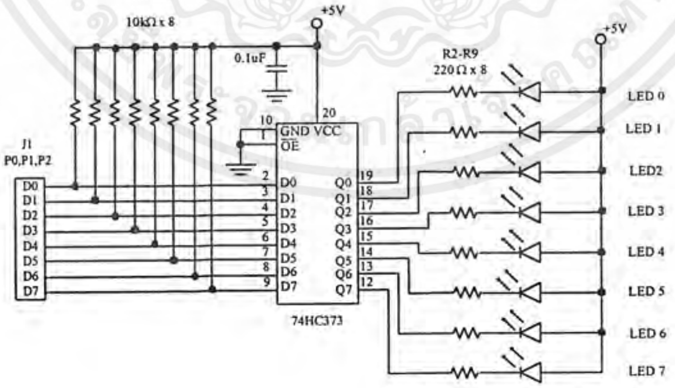
1. สามารถเขียนโปรแกรมเพื่อส่งข้อมูลออกทางพอร์ตของไมโครคอนโทรลเลอร์ได้
2. สามารถเขียนโปรแกรมเพื่อควบคุมการทำงานของแอลอีดีแบบอาโนดร่วมได้
3. สามารถเขียนโปรแกรมเพื่อควบคุมการทำงานของแอลอีดีแบบคาโถดร่วมได้

ทฤษฎีเบื้องต้น

ไดโอดเปล่งแสง (Light Emitting Diode : LED) เป็นไดโอดที่สามารถเปล่งแสงสว่างออกมาด้วยคลื่นความถี่เดียวและมีเฟสต่อเนื่องได้ ซึ่งต่างจากแสงธรรมดากับที่คนเรามองเห็น ประกอบไปด้วยคลื่นที่มีความถี่และเฟสต่างๆ มารวมกัน

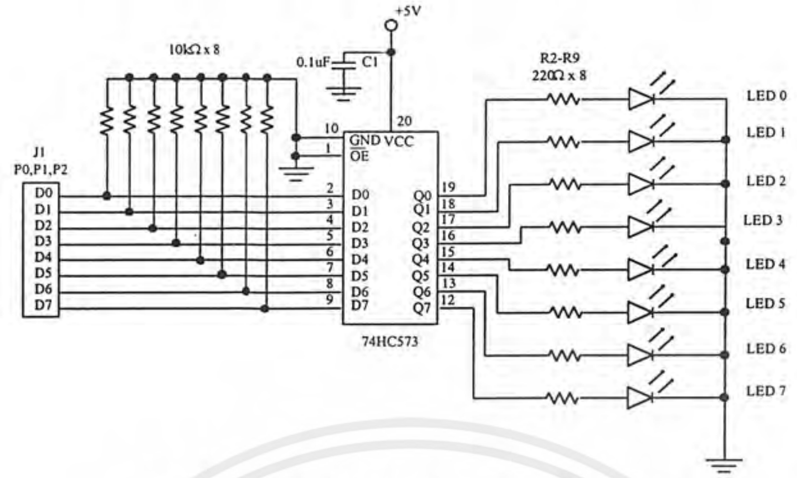
ไดโอดเปล่งแสงมีการต่อใช้งานอยู่ 2 ชนิด

1) ชนิดอาโนดร่วม (Common Anode) คือการนำขาอาโนด (A) ของไดโอดเปล่งแสงมาต่อร่วมกันดังรูปที่ 2.1 ในการที่จะทำให้วงจรทำงานได้นั้น ในการทดลองต้องป้อนลอจิก "0" ให้จึงเกิดการเปล่งแสง ถ้าป้อนลอจิก "1" จะทำให้ไดโอดเปล่งแสงดับ



รูปที่ 2.1 วงจรไดโอดเปล่งแสงชนิดอาโนดร่วม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.2 วงจรไดโอดเปล่งแสงชนิดคาโถดร่วม

2) ชนิดคาโถดร่วม (Common Cathode) คือการนำขาคาโถด (K) ของไดโอดเปล่งแสงมาต่อร่วมกันดังรูปที่ 2.2 ในการที่จะทำให้วงจรทำงานได้นั้น ในการทดลองต้องป้อนลอจิก "1" ให้จึงเกิดการเปล่งแสง ถ้าป้อนลอจิก "0" จะทำให้ไดโอดเปล่งแสงดับ

เครื่องมือและอุปกรณ์

1) บอร์ดไมโครคอนโทรลเลอร์ MCS51-S8252 MC-01

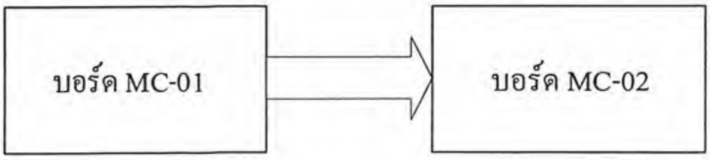
2) บอร์ดวงจรแอลอีดีชนิดอาโนดร่วม MC-02

3) บอร์ดวงจรแอลอีดีชนิดคาโถดร่วม MC-03

4) คอมพิวเตอร์ที่ติดตั้งวินโดว์ 95 หรือ 98 พร้อมโปรแกรม wins8252

ลำดับขั้นการทดลอง

1) ทำการเชื่อมต่อบอร์ดไมโครคอนโทรลเลอร์ MCS51-S8252 MC-01 กับบอร์ดเชื่อมต่อวงจรแอลอีดีชนิดคาโนดร่วม MC-02 โดยต่อที่พอร์ต P.1 ดังรูปที่ 2.3 และกดปุ่ม Power on



รูปที่ 2.3 การเชื่อมต่อบอร์ดวงจรแอลอีดีชนิดคาโนดร่วม กับบอร์ดควบคุม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2) เปิดโปรแกรม wins8252 ทำการตั้งชื่อและบันทึกโดยใช้ไฟล์ชื่อ Lab2.Asm

3) เขียนโปรแกรมดังต่อไปนี้

```
            ORG    0000H
            MOV    A,#01H
START :     MOV    P1,A
            ACALL  DELAY
            RL     A
            JMP    START
DELAY :     MOV    R3,#0FFH
DELAY_1 :   MOV    R4,#0FFH
            DJNZ   R4,$
            DJNZ   R3,DELAY_1
            RET
            END
```

รูปที่ 2.4 โปรแกรมไฟวิ่ง

4) ทำการบันทึกและแอสเซมเบลอร์ไฟล์จาก Lab2.Asm เป็น Lab2.Hex

5) ทำการโปรแกรมลงบนไมโครคอนโทรลเลอร์

6) กดสวิตช์รีเซตแล้วสังเกตผลการทดลอง

____

____

____

____

7) กดปุ่ม Power off และทำการเปลี่ยนบอร์ดวงจรแอลอีดีชนิดอาโนดร่วม และกดปุ่ม Power on

8) เขียนโปรแกรมจากโปรแกรม 2.1 โดยแก้ไข

`MOV A, #01H` เป็น `MOV A,#0FEH`

9) ทำการบันทึกและแอสเซมเบลอร์ไฟล์จาก Lab2.Asm เป็น Lab2.Hex

10) ทำการโปรแกรมลงบนไมโครคอนโทรลเลอร์

11) กดสวิตช์รีเซตแล้วสังเกตผลการทดลอง

____

____

____

____

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สรุปผลการทดลอง

คำถามท้ายการทดลอง

1. จากโปรแกรมการทดลองที่ 3 ให้แก้ไขโปรแกรมโดยให้แอลอีดีสว่างจากดวงที่ 1 ถึงดวงที่ 8 โดยให้สว่างจากซ้ายไปขวา

2. จากโปรแกรมการทดลองที่ 8 ให้ทำการแก้ไขโปรแกรมโดยให้แอลอีดีดับนาน 500 มิลลิวินาที แล้วเริ่มติดจากดวงที่ 1 ถึงดวงที่ 8 โดยหน่วงเวลานาน 500 มิลลิวินาที

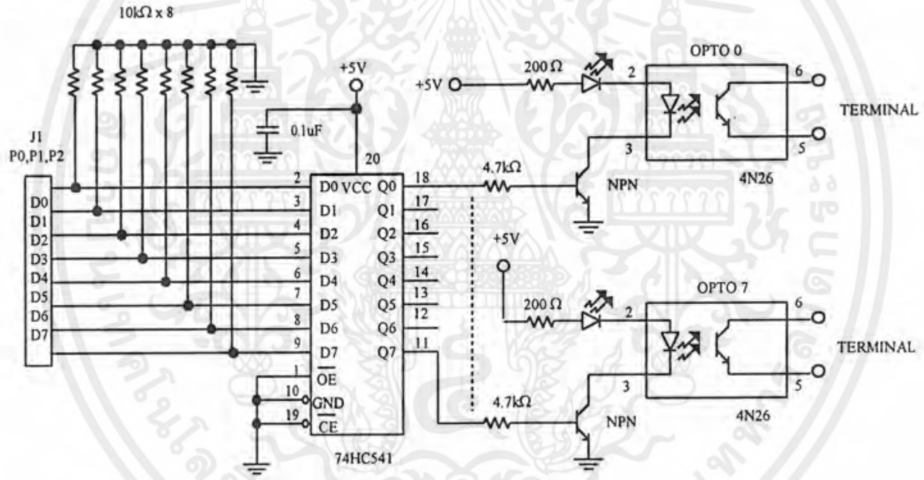
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ใบงานที่ 3
วงจรเชื่อมโยงแสงออก

วัตถุประสงค์

1. สามารถเขียนโปรแกรมเพื่อควบคุมการทำงานผ่านอุปกรณ์เชื่อมโยงแสงออกได้
2. สามารถนำการทำงานไปประยุกต์ใช้กับบอร์ดอื่นๆได้

ทฤษฎีเบื้องต้น



รูปที่ 3.1 วงจรเชื่อมโยงแสงแสงออก

จากวงจรรูปที่ 3.1 การทำงานจะใช้การเชื่อมต่อทางแสงโดยใช้ไดโอดเปล่งแสงภายใน ซึ่งจะควบคุมโดยทรานซิสเตอร์ 2N3904 แบบเอ็นพีเอ็น ถ้าข้อมูลที่ส่งออกมาทางพอร์ตมีค่าเป็นลอจิกสูงจะทำให้ไดโอดเปล่งแสงภายในตัวเชื่อมโยงทางแสงจะสว่าง ซึ่งสามารถนำไปใช้ในการควบคุมการเปิด-ปิดวงจรอิเล็กทรอนิกส์ได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เครื่องมือและอุปกรณ์

1) บอร์ดไมโครคอนโทรลเลอร์ MCS51-S8252 MC-01

2) บอร์ดอุปกรณ์เชื่อมโยงทางแสง MC-04

3) บอร์ดอินพุตสวิตช์เดี่ยว MC-14

4) คอมพิวเตอร์ที่ติดตั้งวินโดว์ 95 หรือ 98 พร้อมโปรแกรม wins8252

ลำดับขั้นการทดลอง

1) ทำการเชื่อมต่อบอร์ดไมโครคอนโทรลเลอร์ MCS51-S8252 MC-01 กับบอร์ดขับอุปกรณ์เชื่อมโยงแสง MC-04 ที่พอร์ต P.1 และบอร์ดอินพุตสวิตช์เดี่ยว MC-14 ที่พอร์ต P.0 (สังเกตสวิตช์ให้อยู่ที่ตำแหน่ง OFF ทั้งหมด)

รูปที่ 3.2 การเชื่อมต่ออุปกรณ์เชื่อมโยงแสงออกกับสวิตซ์เดี่ยว

2) เปิดสวิตช์ Power On

3) เปิดโปรแกรม wins8252 ทำการตั้งชื่อและบันทึกโดยใช้ไฟล์ชื่อ Lab3.Asm

4) เขียนโปรแกรมดังต่อไปนี้

```
            ORG         0000H
MAIN :      MOV         P0,#0FFH
            MOV         P2,#000H
SHOW1 :     JNB         P0.0,SHOW2
            ACALL       OPTO1
SHOW2 :     JNB         P0.1,SHOW3
            ACALL       OPTO2
SHOW3 :     JNB         P0.2,SHOW4
            ACALL       OPTO3
SHOW4 :     JNB         P0.3,SHOW5
            ACALL       OPTO4
SHOW5 :     JNB         P0.4,SHOW6
            ACALL       OPTO5
SHOW6 :     JNB         P0.5,SHOW7
            ACALL       OPTO6
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
SHOW7 :        JNB       P0.6,SHOW8
               ACALL     CPTO7
SHOW8 :        JNB       P0.7,MAIN
               ACALL     OPTO8
               JMP        MAIN
;*********************************************
OPTO1 :        MOV       P2,#01H
               RET
OPTO2:         MOV       P2,#02H
               RET
OPTO3:         MOV       P2,#04H
               RET
OPTO4:         MOV       P2,#08H
               RET
OPTO5:         MOV       P2,#10H
               RET
OPTO6:         MOV       P2,#20H
               RET
OPTO7:         MOV       P2,#40H
               RET
OPTO8:         MOV       P2,#80H
               RET
;**********************************************
DELAY:         MOV       7,#010H
DELAY_1:       MOV       6,#0E6H
DELAY_2:       NOP
               NOP
               DJNZ      R6,DELAY_2
               DJNZ      R7,DELAY_1
               RET
               END
```

รูปที่ 3.3 โปรแกรมการอ่านค่าจากสวิตซ์ด้วยวงจรเชื่อมโยงแสงออก

5) ทำการบันทึกและแอสเซมเบลอร์ไฟล์จาก Lab3.Asm เป็น Lab3.Hex

6) ทำการ โปรแกรมลงบนไมโครคอนโทรลเลอร์

7) กดสวิตช์รีเซตแล้วสังเกตผลการทดลอง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สรุปผลการทดลอง

คำถามท้ายการทดลอง

จงเขียนโปรแกรมควบคุมอุปกรณ์ผ่านอุปกรณ์เชื่อมโยงแสงโดยเมื่อ

-กดสวิตช์ 1 ให้ ออปโต้คัปเปอร์ตัวที่ 1, 2, 3, 4 ทำงาน

-กดสวิตช์ 2 ให้ออปโต้คัปเปอร์ ตัวที่ 5, 6, 7, 8 ทำงาน

-เมื่อกดสวิตช์อื่นๆ ออปโต้คัปเปอร์ ไม่ทำงาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ใบงานที่ 4

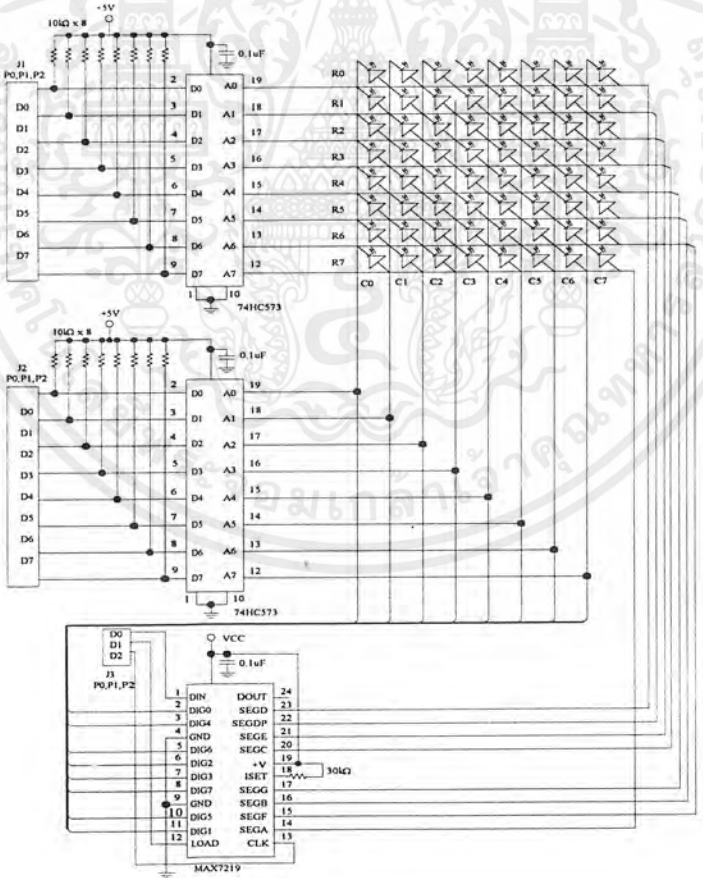
วงจรแอลอีดีแบบเมตริกซ์

วัตถุประสงค์

1. สามารถเขียนโปรแกรมเพื่อส่งข้อมูลไปแสดงผลที่แอลอีแบบเมตริกซ์ได้
2. สามารถนำไปประยุกต์ใช้กับบอร์ดอื่นๆได้

ทฤษฎีเบื้องต้น

วงจรแอลอีดีแบบเมตริกซ์ จะใช้ แอลอีดี 8x8 รวมเป็น 64 ตัว และใช้ไอซีอีก 2 ตัว เบอร์ 74HC573 เป็นตัวสำหรับเช็คแถวและหลัก และไอซีอีก 1 ตัว เบอร์ MAX7219 ที่สามารถทำงานได้ทั้งแถว และหลักในตัวเดียว



รูปที่ 4.1 วงจรแอลอีดีแบบเมตริกซ์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เครื่องมือและอุปกรณ์

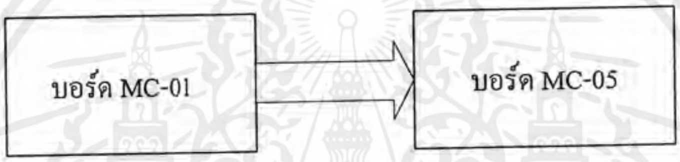
1) บอร์ดไมโครคอนโทรลเลอร์ MCS51-S8252 MC-01

2) บอร์ดวงจรแอลอีดีแบบเมตริกซ์ MC-05

3) คอมพิวเตอร์ที่ติดตั้งวินโดว์ 95 หรือ 98 พร้อมโปรแกรม wins8252

ลำดับขั้นการทดลอง

1) ทำการเชื่อมต่อบอร์ดไมโครคอนโทรลเลอร์ MCS51-S8252 -01 กับ บอร์ดวงจรแอลอีดีแบบเมตริกซ์ MC-05 โดยต่อที่พอร์ต P.1 เพื่อใช้ควบคุมแถว และต่อพอร์ต P.2 เพื่อใช้ควบคุมคอลัมน์ ตามรูปที่ 4.2 กดปุ่ม Power on



รูปที่ 4.2 การเชื่อมต่อบอร์ดแอลอีดีแบบเมตริกซ์ MC-05

2) เปิดโปรแกรม wins8252 ทำการตั้งชื่อและบันทึกโดยใช้ไฟล์ชื่อ Lab4.Asm

3) เขียนโปรแกรมดังต่อไปนี้

```
        ORG     0000H
START:  MOV     A,#0FCH
        MOV     P0,A
        ACALL   DELAY
        MOV     A,#0BFH
        MOV     P2,A
        ACALL   DELAY

        MOV     A,#00H
        MOV     P0,A
        ACALL   DELAY
        MOV     A,#0FFH
        MOV     P2,A
        ACALL   DELAY

        MOV     A,#12H
        MOV     P0,A
        ACALL   DELAY
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        MOV         A,#0DFH
        MOV         P2,A
        ACALL       DELAY

        MOV         A,#00H
        MOV         P0,A
        ACALL       DELAY

        MOV         A,#0FFH
        MOV         P2,A
        ACALL       DELAY

        MOV         A,#11H
        MOV         P0,A
        ACALL       DELAY
        MOV         A,#0EFH
        MOV         P2,A
        ACALL       DELAY

        MOV         A,#00H
        MOV         P0,A
        ACALL       DELAY
        MOV         A,#0FFH
        MOV         P2,A
        ACALL       DELAY

        MOV         A,#11H
        MOV         P0,A
        ACALL       DELAY
        MOV         A,#0F7H
        MOV         P2,A
        ACALL       DELAY

        MOV         A,#00H
        MOV         P0,A
        ACALL       DELAY
        MOV         A,#0FFH
        MOV         P2,A
        ACALL       DELAY

        MOV         A,#12H
        MOV         P0,A
        ACALL       DELAY
        MOV         A,#0FBH
        MOV         P2,A
        ACALL       DELAY

        MOV         A,#00H
        MOV         P0,A
        ACALL       DELAY
        MOV         A,#0FFH
        MOV         P2,A
        MOV         A,#0FCH
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        MOV         P0,A
        ACALL       DELAY
        MOV         A,#0FDH
        MOV         P2,A
        ACALL       DELAY

        JMP         START
DELAY:  MOV         R3,#01H
DELAY_1: MOV        R4,#0FFH
        DJNZ        R4,$
        DJNZ        R3,DELAY_1
        RET

        END
```

รูปที่ 4.3 โปรแกรมแสดงตัวอักษร “A”

4) ทำการบันทึกและแอสเซมเบลอร์ไฟล์จาก Lab4.Asm เป็น Lab4.Hex

5) ทำการโปรแกรมลงบนไมโครคอนโทรลเลอร์

6) กดสวิตช์รีเซตแล้วสังเกตผลการทดลอง

สรุปผลการทดลอง

คำถามท้ายการทดลอง

จากการทดลองให้เขียนโปรแกรมแสดงตัวอักษร “E” ติดอยู่ตลอดเวลาโดยใช้พอร์ต P.1 ควบคุมแถวและพอร์ต P.2 เพื่อใช้ควบคุมคอลัมน์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

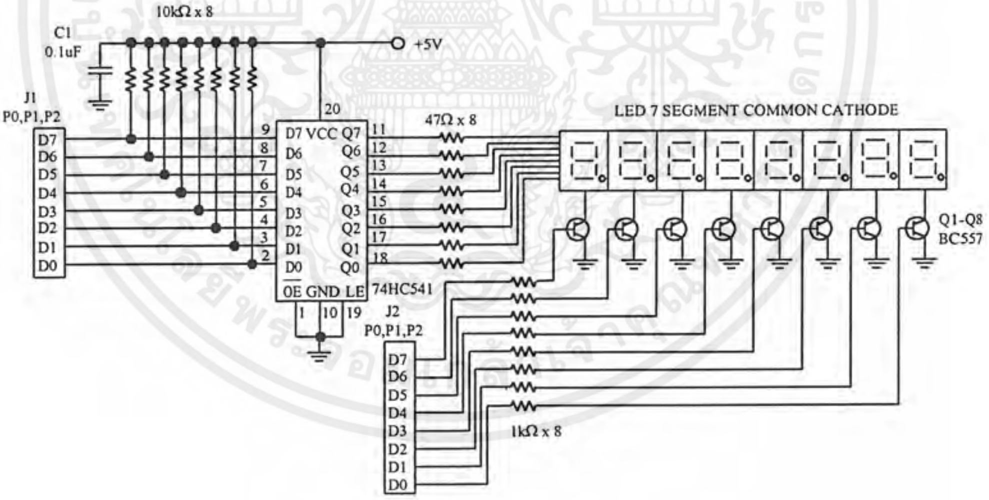
ใบงานที่ 5
การเชื่อมต่อกับแอลอีดีแบบ 7 ส่วน

วัตถุประสงค์

1. สามารถเขียนโปรแกรมเพื่อขับแอลอีดีแบบ 7 ส่วนแบบตัวเดี่ยวและมัลติเพล็กซ์ได้
2. สามารถนำการทำงานไปประยุกต์ใช้กับบอร์ดอื่นๆ ได้

ทฤษฎีเบื้องต้น

การแสดงผลแบบ 7 ส่วน แสดงผลออกมาเป็นตัวเลข ภายในประกอบด้วยแอลอีดี 8 ดวง เป็นส่วน a, b, c, d, e, f, g และจุดทศนิยม (dp) การแสดงผลของวงจรแอลอีดีแบบ 7 ส่วน จำเป็นต้องอาศัยการเขียนโปรแกรมจากไมโครคอนโทรลเลอร์ MCS-51 จ่ายไฟให้แก่ขาร่วมของแอลอีดีตัวเลขทีละหลักตามลำดับโดยผ่านทรานซิสเตอร์ดังรูปที่ 1



รูปที่ 5.1 วงจรแอลอีดีแบบ 7 ส่วน

ในการเขียนโปรแกรมในการควบคุมต้องเขียนควบคู่ไปกับการเปิดตารางข้อมูลช่วยแล้วใช้โปรแกรมหน่วงเวลาเพื่อให้แอลอีดีในเซกเมนต์ที่ถูกสั่งให้ทำงานนั้นติดสว่างนานพอให้ผู้ใช้งานเห็นข้อมูลที่นำมาแสดงผล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เครื่องมือและอุปกรณ์

1) บอร์ดไมโครคอนโทรลเลอร์ MCS51-S8252 MC-01

2) บอร์ดวงจรแอลอีดีแบบ 7 ส่วน MC-06

3) คอมพิวเตอร์ที่ติดตั้งวินโดว์ 95 หรือ 98 พร้อมโปรแกรม wins8252

ลำดับขั้นการทดลอง

1) ทำการเชื่อมต่อบอร์ดไมโครคอนโทรลเลอร์ MCS51-S8252 MC-01 กับบอร์ดวงจรแอลอีดีแบบ 7 ส่วน MC-06 แสดงค่าที่พอร์ต P.2 และ DSP 1 ที่ P1.0 และ DSP 2 ที่ P1.0 โดยต่อตามรูปที่ 5.2

รูปที่ 5.2 บอร์ดวงจรแอลอีดีแบบ 7 ส่วน กับบอร์ดควบคุม

2) กดสวิตช์ Power On

3) เปิดโปรแกรม wins8252 ทำการตั้งชื่อและบันทึกโดยใช้ไฟล์ชื่อ Lab5.Asm

4) เขียนโปรแกรมดังต่อไปนี้

```
DSP1        BIT     P1.0
DSP2        BIT     P1.1
DSP_NUM0    EQU     3FH
DSP_NUM1    EQU     06H
DSP_NUM2    EQU     5BH
DSP_NUM3    EQU     4FH
DSP_NUM4    EQU     66H
DSP_NUM5    EQU     6DH
DSP_NUM6    EQU     7DH
DSP_NUM7    EQU     07H
DSP_NUM8    EQU     7FH
DSP_NUM9    EQU     6FH
DSP_NUMA    EQU     77H
DSP_NUMB    EQU     7CH
DSP_NUMC    EQU     39H
DSP_NUMD    EQU     5EH
DSP_NUME    EQU     79H
DSP_NUMF    EQU     71H
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                ORG       0000
                MOV       P2,#00H
MAIN:           CLR       DSP1
                SETB      DSP2
LOOP:           MOV       P2,#DSP_NUM0
                ACALL     DELAY_500MS
                MOV       P2,#DSP_NUM1
                ACALL     DELAY_500MS
                MOV       P2,#DSP_NUM2
                ACALL     DELAY_500MS
                MOV       P2,#DSP_NUM3
                ACALL     DELAY_500MS
                MOV       P2,#DSP_NUM4
                ACALL     DELAY_500MS
                MOV       P2,#DSP_NUM5
                ACALL     DELAY_500MS
                MOV       P2,#DSP_NUM6
                ACALL     DELAY_500MS
                MOV       P2,#DSP_NUM7
                ACALL     DELAY_500MS
                MOV       P2,#DSP_NUM8
                ACALL     DELAY_500MS
                MOV       P2,#DSP_NUM9
                ACALL     DELAY_500MS
                MOV       P2,#DSP_NUMA
                ACALL     DELAY_500MS
                MOV       P2,#DSP_NUMB
                ACALL     DELAY_500MS
                MOV       P2,#DSP_NUMC
                ACALL     DELAY_500MS
                MOV       P2,#DSP_NUMD
                ACALL     DELAY_500MS
                MOV       P2,#DSP_NUME
                ACALL     DELAY_500MS
                MOV       P2,#DSP_NUMF
                ACALL     DELAY_500MS
                MOV       R4,#2

BLINK_DOT:      SETB      P2.7
                ACALL     DELAY_250MS
                CLR       P2.7
                ACALL     DELAY_250MS
                DJNZ      R4,BLINK_DOT
                CPL       DSP1
                CPL       DSP2
                AJMP      LOOP
DELAY_250MS:    MOV       7,#250
DELAY_250MS_1:  MOV       6,#0E6H
DELAY_250MS_2:  NOP
                NOP
                DJNZ      R6,DELAY_250MS_2
                DJNZ      R7,DELAY_250MS_1
                RET
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
DELAY_500MS:      MOV         5,#2
DELAY_500MS_1:    ACALL       DELAY_250MS
                  DJNZ        R5,DELAY_500MS_1
                  RET
                  END
```

รูปที่ 5.3 โปรแกรมแสดงค่าตัวเลข

5) ทำการบันทึกและแอสเซมเบลอร์ไฟล์จาก ab5.Asm เป็น Lab5.Hex

6) ทำการโปรแกรมลงบนไมโครคอนโทรลเลอร์

7) กดสวิตช์รีเซตแล้วสังเกตผลการทดลอง

สรุปผลการทดลอง

คำถามท้ายการทดลอง

จากการทดลองจงเขียนโปรแกรมแสดงการนับเลขตั้งแต่ 0-F โดยกำหนด DSP_NUM0 ถึง DSP_NUMF โดยให้แอลอีดีแบบ 7 ส่วนส่วนติดครั้งละ 4 หลัก สลับกันไปไม่รู้จบ โดยกำหนดให้

```
- DSP 1     BIT    P1.0
- DSP 2     BIT    P1.1
- DSP 3     BIT    P1.2
- DSP 4     BIT    P1.3
- DSP 5     BIT    P1.4
- DSP 6     BIT    P1.5
- DSP 7     BIT    P1.6
- DSP 8     BIT    P1.7
```

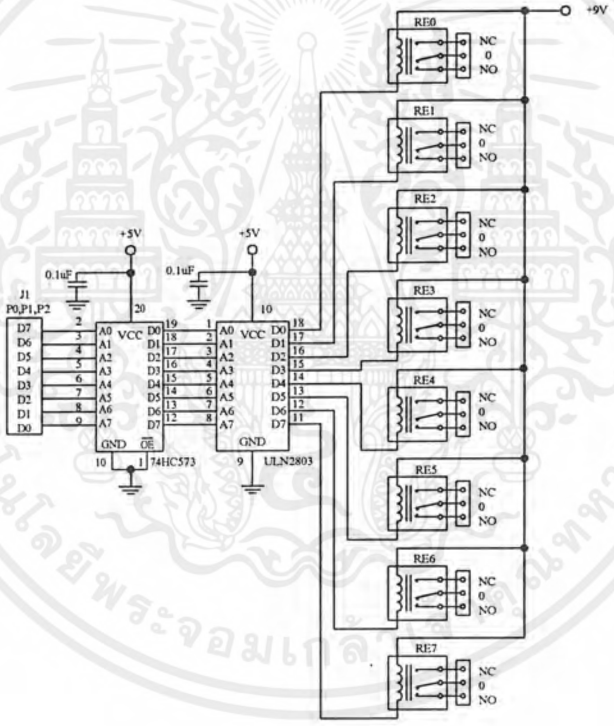
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ใบงานที่ 6
การควบคุมอุปกรณ์รีเลย์

วัตถุประสงค์

1. สามารถเขียนโปรแกรมเพื่อควบคุมการทำงานของอุปกรณ์รีเลย์ร่วมกับสวิตช์เดี่ยวได้
2. สามารถนำการทำงานไปประยุกต์ใช้กับบอร์ดอื่นๆได้

ทฤษฎีเบื้องต้น



รูปที่ 6.1 วงจรควบคุมการขับรีเลย์

วงจรขับรีเลย์ เป็นวงจรที่ใช้ขับรีเลย์จำนวน 8 ตัว แสดง การทำงานโดยมีการจ่ายแรงดันให้กับรีเลย์ 9 โวลต์เมื่อรับอินพุตจากพอร์ตของ MCS-51 โดยจัมพ์เลือกระหว่างพอร์ต P0, P1 หรือ P2 ซึ่งวงจรขับรีเลย์จะมีบัฟเฟอร์ 74HC573 โดยมีขาที่ 11 เป็นตัวเลือก แล้วส่งผ่านข้อมูลไปยังรีเลย์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เครื่องมือและอุปกรณ์

1) บอร์ดไมโครคอนโทรลเลอร์ MCS51-S8252 MC-01

2) บอร์ดขับอุปกรณ์รีเลย์ MC-07

3) บอร์ดอินพุตสวิตช์เดี่ยว MC-14

4) คอมพิวเตอร์ที่ติดตั้งวินโดว์ 95 หรือ 98 พร้อมโปรแกรม wins8252

ลำดับขั้นการทดลอง

1) ทำการเชื่อมต่อบอร์ดไมโครคอนโทรลเลอร์ MCS51-S8252 MC-01 กับบอร์ดขับอุปกรณ์รีเลย์ MC-07 ที่พอร์ต P.2 และบอร์ดอินพุตสวิตช์เดี่ยว MC-14 ที่พอร์ต P.0 (สังเกตสวิตช์ทุกตัวให้อยู่ที่ตำแหน่ง Off ทั้งหมด) ตามรูปที่ 6.2

รูปที่ 6.2 การเชื่อมต่อบอร์ดขับรีเลย์กับบอร์ดควบคุม

2) เปิดสวิตช์ Power on

3) เปิดโปรแกรม wins8252 ทำการตั้งชื่อและบันทึกโดยใช้ไฟล์ชื่อ Lab6.Asm

4) เขียนโปรแกรมดังต่อไปนี้

```
          ORG       0000H
MAIN:     MOV       P0,#0FFH
          MOV       P2,#000H
SW1:      JNB       P0.0,SW2
          ACALL     RELAY1
SW2:      JNB       P0.1,SW3
          ACALL     RELAY2
SW3:      JNB       P0.2,SW4
          ACALL     RELAY3
SW4:      JNB       P0.3,SW5
          ACALL     RELAY4
SW5:      JNB       P0.4,SW6
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
              ACALL        RELAY5
SW6:          JNB          P0.5,SW7
              ACALL        RELAY6
SW7:          JNB          P0.6,SW8
              ACALL        RELAY7
SW8:          JNB          P0.7,MAIN
              ACALL        RELAY8
              JMP          MAIN
;*******************************************
RELAY1:       MOV          P2,#01H
              RET
RELAY2:       MOV          P2,#02H
              RET
RELAY3:       MOV          P2,#04H
              RET
RELAY4:       MOV          P2,#08H
              RET
RELAY5:       MOV          P2,#10H
              RET
RELAY6:       MOV          P2,#20H
              RET
RELAY7:       MOV          P2,#40H
              RET
RELAY8:       MOV          P2,#80H
              RET
;*******************************************
DELAY:        MOV          7,#010H
DELAY_1:      MOV          6,#0E6H
DELAY_2:      NOP
              NOP
              DJNZ         R6,DELAY_2
              DJNZ         R7,DELAY_1
              RET
              END
```

รูปที่ 6.3 โปรแกรม การอ่านค่าจากคีย์สวิตช์กับอุปกรณ์รีเลย์

5) ทำการบันทึกและแปลงไฟล์จาก Lab6.Asm เป็น Lab6.Hex

6) ทำการ โปรแกรมลงบนไมโครคอนโทรลเลอร์

7) กดสวิตช์รีเซตแล้วสังเกตผลการทดลอง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สรุปผลการทดลอง

คำถามท้ายการทดลอง

จงเขียนโปรแกรมควบคุมอุปกรณ์รีเลย์โดยเมื่อ

- กดสวิตช์ 0 ให้อุปกรณ์รีเลย์ตัวที่ 1 และตัวที่ 2 ทำงาน
- กดสวิตช์ 1 ให้อุปกรณ์รีเลย์ตัวที่ 3 และตัวที่ 4 ทำงาน
- กดสวิตช์ 2 ให้อุปกรณ์รีเลย์ตัวที่ 5 และตัวที่ 6 ทำงาน
- กดสวิตช์ 3ให้อุปกรณ์รีเลย์ตัวที่ 7 และตัวที่ 8 ทำงาน
- กดสวิตช์ 4 ให้อุปกรณ์รีเลย์ทุกตัวไม่ทำงาน
- เมื่อกดสวิตช์ 5, 6, 7 อุปกรณ์รีเลย์ทุกตัวทำงาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ใบงานที่ 7
การขับสเต็ปปิ้งมอเตอร์

วัตถุประสงค์

1. สามารถเขียนโปรแกรมเพื่อควบคุมการทำงานของ สเต็ปปิ้งมอเตอร์ได้
2. สามารถนำการทำงานไปประยุกต์ใช้กับบอร์ดอื่นๆ ได้

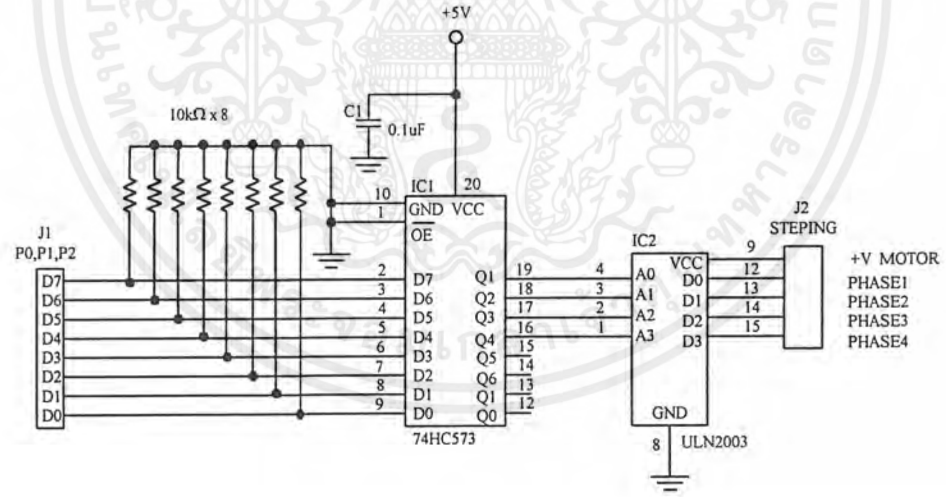
ทฤษฎีเบื้องต้น

สเต็ปปิ้งมอเตอร์ถือว่าเป็นอุปกรณ์เอาต์พุตอย่างหนึ่งซึ่งสามารถควบคุมได้ด้วยคอมพิวเตอร์ ลักษณะการทำงานจะเคลื่อนที่เป็นขั้น (step)

สเต็ปปิ้งมอเตอร์ มีส่วนประกอบสำคัญ 2 ส่วน คือ

1)โรเตอร์ คือส่วนที่หมุนได้ จะเป็นแม่เหล็กถาวรหรืออื่น ๆ

2) สเตเตอร์คือส่วนที่ติดอยู่กับที่ เป็นขดลวดหลาย ๆ ขด



รูปที่ 7.1 วงจรขับสเต็ปปิ้งมอเตอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การหมุนของสเต็ปปิ้งมีอยู่ 2 แบบ คือ

1. แบบครึ่งขั้น

เป็นการนำเอาวิธีขับแบบ Wave Unipolar ผสมกับ Two Phase Unipolar หมุนจะหมุนครั้งละครึ่งขั้น และองศาที่ได้จากการหมุนจะเท่ากับครึ่งหนึ่งของการหมุนแบบเต็มขั้นการป้อนกระแสจะเป็นไปตามตาราง

ขั้น	เฟส			
	1	2	3	4
1	1	1	0	0
2	0	1	0	0
3	0	1	1	0
4	0	0	1	1
5	0	0	1	1
6	0	0	0	1
7	1	0	0	1
8	1	0	0	0

รูปที่ 7.2 การจ่ายกระแสให้กับขดลวดแบบครึ่งขั้น

2. แบบเต็มขั้น

แบ่งการขับออกเป็น 2 แบบคือ

1) การขับแบบ Wave Unipolar เป็นการป้อนกระแสให้กับขดลวดแต่ละขดของสเต็ปปิ้งมอเตอร์ทีละขดเรียงกันไป ลักษณะการขับแบบนี้จะให้แรงบิดน้อย

2) การขับแบบ 2 เฟส เป็นการป้อนกระแสให้แก่ขดลวด 2 ขดพร้อมกัน ดังตาราง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ขั้น	เฟส			
	1	2	3	4
1	1	1	0	0
2	0	1	1	0
3	0	0	1	1
4	1	0	0	1

รูปที่ 7.3 ตารางการจ่ายกระแสให้กับขวดลวดแต่ละเฟสของสเต็ปปิ้งมอเตอร์

เครื่องมือและอุปกรณ์

1) บอร์ดไมโครคอนโทรลเลอร์ MCS51-S8252 MC-01

2) บอร์ดสเต็ปปิ้งมอเตอร์ MC-08

3) คอมพิวเตอร์ที่ติดตั้งวินโดว์ 95 หรือ 98 พร้อมโปรแกรม wins8252

ลำดับขั้นการทดลอง

1) ทำการเชื่อมต่อบอร์ดไมโครคอนโทรลเลอร์ MCS51-S8252 MC-01 กับบอร์ดสเต็ปปิ้งมอเตอร์ MC-08 และแหล่งจ่ายไฟโดยสังเกตเฟสมอเตอร์ให้ถูกต้องด้วยโดยต่อที่พอร์ต P.1 ตามรูปที่ 7.4 และกดปุ่ม Power on

รูปที่ 7.4 การเชื่อมต่อบอร์ดสเต็ปปิ้งมอเตอร์กับบอร์ดควบคุม

2) เปิดโปรแกรม wins8252 ทำการตั้งชื่อและบันทึกโดยใช้ไฟล์ชื่อ Lab7.Asm

3) เขียนโปรแกรมดังต่อไปนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
            ORG          0000H
START:      MOV          A,#01H
            MOV          P1,A
            ACALL        DELAY
            MOV          A,#02H
            MOV          P1,A
            ACALL        DELAY
            MOV          A,#04H
            MOV          P1,A
            ACALL        DELAY
            MOV          A,#08H
            MOV          P1,A
            ACALL DELAY
DELAY:      MOV          R3,#0FFH
DELAY_1:    MOV          R4,#0FFH
            DJNZ         R4,$
            DJNZ         R3,DELAY_1
            RET
            END
```

รูปที่ 7.5 โปรแกรมขับสเต็ปปิ้งมอเตอร์

4) ทำการบันทึกและแอสเซมเบลอร์ไฟล์จาก Lab7.Asm เป็น Lab7.Hex

5) ทำการโปรแกรมลงบนไมโครคอนโทรลเลอร์

6) กดสวิตช์รีเซตแล้วสังเกตผลการทดลอง

____

____

____

____

สรุปผลการทดลอง

____

____

____

____

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

คำถามท้ายการทดลอง

จากโปรแกรมที่ 7.1 จงแก้ไขโปรแกรมให้สเต็ปปิ้งมอเตอร์หมุนทางขวาครบทั้ง 5 รอบ แล้วหมุนทางซ้ายอีก 2 รอบ แล้วกลับมาหมุนขวาอีก 5 รอบ วนรอบเช่นนี้ไม่รู้จบสิ้น โดยการหมุนมีการหน่วงเวลา 1 วินาที โดยใช้การควบคุมจากพอร์ต P.0

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ใบงานที่ 8
การขับมอเตอร์ไฟฟ้ากระแสตรง

วัตถุประสงค์

1. สามารถเขียนโปรแกรมสั่งงานมอเตอร์ไฟฟ้ากระแสตรงได้
2. สามารถเขียนโปรแกรมเพื่อควบคุมทิศทางของมอเตอร์ไฟฟ้ากระแสตรงได้

ทฤษฎีเบื้องต้น

มอเตอร์ไฟฟ้ากระแสตรงสามารถจำแนกได้หลายประเภทแต่ที่ใช้ในปัจจุบันกันอย่างมากก็คือ มอเตอร์ไฟฟ้ากระแสตรงแบบขนาน, แบบอนุกรม, แบบผสม และแบบแม่เหล็กถาวร

การควบคุมความเร็วของมอเตอร์สามารถทำได้ 2 วิธี คือ

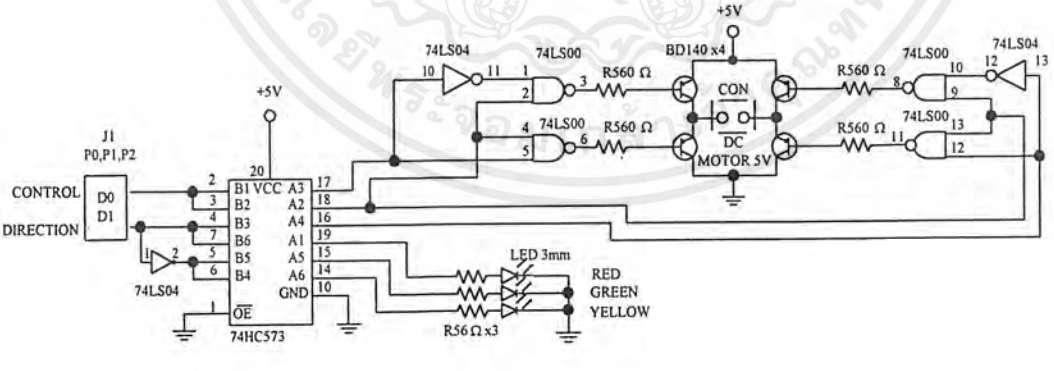
1) การควบคุมแรงดันอาร์เมเจอร์

2) การควบคุมความเข้มของสนามแม่เหล็ก

การควบคุมทิศทางสามารถควบคุมได้ 2 ทิศทาง คือ

1) ควบคุมให้มอเตอร์ไฟฟ้ากระแสตรงหมุนขวา

2) ควบคุมให้มอเตอร์ไฟฟ้ากระแสตรงหมุนซ้าย



รูปที่ 8.1 วงจรควบคุมมอเตอร์ไฟฟ้ากระแสตรง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เครื่องมือและอุปกรณ์

1) บอร์ดไมโครคอนโทรลเลอร์ MCS51-S8252 MC-01

2) บอร์ดมอเตอร์ไฟฟ้ากระแสตรง MC-09

3) บอร์ดอินพุตสวิตช์เดี่ยว MC-14

4) คอมพิวเตอร์ที่ติดตั้งวินโดว์ 95 หรือ 98 พร้อมโปรแกรม wins8252

ลำดับขั้นการทดลอง

1) ทำการเชื่อมต่อบอร์ดไมโครคอนโทรลเลอร์ MCS51-S8252 MC-01 กับบอร์ดมอเตอร์ไฟฟ้ากระแสตรง MC-09 โดยต่อที่พอร์ต P.2 และบอร์ดอินพุตสวิตช์เดี่ยว MC-14 โดยต่อที่พอร์ต P.0 ตามรูปที่ 8.2 และกดปุ่ม Power on

รูปที่ 8.2 การเชื่อมต่อบอร์ดมอเตอร์ไฟฟ้ากระแสตรงกับบอร์ดสวิตซ์

2) เปิดโปรแกรม wins8252 ทำการตั้งชื่อและบันทึกโดยใช้ไฟล์ชื่อ Lab8.Asm

3) เขียนโปรแกรมดังต่อไปนี้

```
                ORG         0000H
MAIN:           MOV         P0,#0FFH
                MOV         P2,#000H
                MOV         P1,#0EFH
;*********************************************
;     scan  key
;*********************************************
KEY1:           JNB         P0.0,KEY2
                ACALL       ROTATE_R
                JB          P0.0,$
KEY2:           JNB         P0.1,KEY3
                ACALL       ROTATE_L
                JB          P0.1,$
KEY3:           JNB         P0.2,MAIN
                ACALL       STOP
                AJMP        MAIN
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
ROTATE_R:           MOV         P2,#01H
                    RET
ROTATE_L:           MOV         P2,#03H
                    RET
STOP:               MOV         P2,#00H
                    RET
;******************************************
DELAY:              MOV         7,#0FFH
DELAY_1:            MOV         6,#0E6H
DELAY_2:            NOP
                    NOP
                    DJNZ        R6,DELAY_2
                    DJNZ        R7,DELAY_1
                    RET
                    END
```

รูปที่ 8.3 โปรแกรมขับมอเตอร์ไฟฟ้ากระแสตรง

4) ทำการบันทึกและแปลงไฟล์จาก Lab8.Asm เป็น Lab8.Hex

5) ทำการโปรแกรมลงบนไมโครคอนโทรลเลอร์

6) กดสวิตช์รีเซตแล้วสังเกตผลการทดลองบันทึกค่าลงในตาราง

ตารางที่ 1 ตารางบันทึกผลการทดลอง

INPUT		MOTOR	LED		
P0.1	P0.0		GREEN	RED	YELLOW
0	0				
0	1				
1	0				
1	1				

สรุปผลการทดลอง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ใบงานที่ 9
วงจรแสดงผลแบบผลึกเหลว

วัตถุประสงค์

1. สามารถเขียนโปรแกรมเพื่อส่งข้อมูลไปแสดงผลที่วงจรแสดงผลแบบผลึกเหลวแบบ 16 ตัวอักษรได้

2. สามารถนำไปประยุกต์ใช้กับบอร์ดอื่นๆ ได้

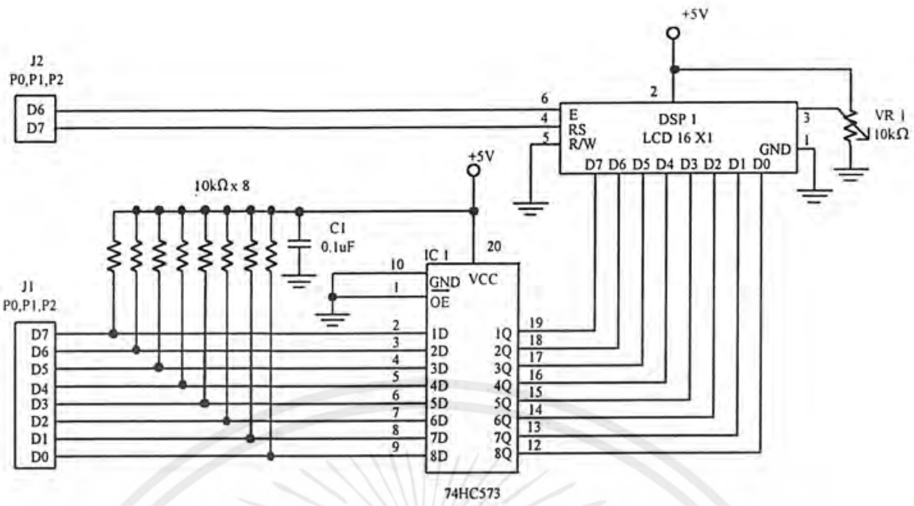
ทฤษฎีเบื้องต้น

ในการติดต่อกับวงจรแสดงผลแบบผลึกเหลวจะต้องมีการหน่วงเวลาหลังจากที่ทำการส่งรหัสคำสั่งหรือข้อมูลเนื่องจากต้องรอให้คอนโทรลเลอร์ภายในแอลซีดีแปลความหมายของรหัสคำสั่งและทำงานตามคำสั่งให้เรียบร้อยก่อน จากนั้นจึงจะรับข้อมูลหรือดำเนินการต่อไป

ดังนั้น ในการใช้งานแอลซีดี ผู้เขียนโปรแกรมต้องมีโปรแกรมเพื่อหน่วงเวลารอให้แอลซีดีพร้อมทำงานด้วย โดยเมื่อเริ่มจ่ายไฟให้แก่แอลซีดีต้องรอประมาณ 10 มิลลิวินาที เพื่อให้แอลซีดีทำการเตรียมความพร้อมหรือ อินิเชียล (initial) หลังจากนั้นจะกำหนดลอจิกให้กับขา RS ของแอลซีดี แล้วต้องหน่วงเวลาประมาณ 2 มิลลิวินาที เพื่อให้คอนโทรลเลอร์ในแอลซีดีแปลความหมายของลอจิกที่ขา RS ว่า ข้อมูลต่อไปที่จะได้รับนั้นเป็นรหัสคำสั่ง หรือเป็นข้อมูลที่ต้องการแสดงผล จากนั้นจะเป็นการส่งข้อมูลมารอที่บัสข้อมูล D0 – D7 (กรณีทำงานในโหมด 8 บิต) ขั้นตอนต่อไป จะเป็นการส่งสัญญาณพัลซ์ไปที่ขา E เพื่ออีนาเบิลแอลซีดีให้รับข้อมูลจากบัสข้อมูลเข้าไป โดยพัลซ์ที่ป้อนเข้าที่ขา E ของ แอลซีดีต้องเป็นพัลซ์ขอบขาขึ้น จากนั้นทำการหน่วงเวลา 2 มิลลิวินาที

วงจรใช้การเขียนโปรแกรมในการสร้างสัญญาณ ให้แสดงผลออกทางหน้าจอแสดงผลแบบผลึกเหลว ซึ่งสามารถแสดงผลได้ ครั้งละ 1 บรรทัด จำนวน 16 ตัวอักษร

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 9.1 วงจรแสดงผลแบบผลึกเหลวขนาด 16 ตัวอักษร

การเขียนคำสั่งและข้อมูลให้แก่แอลซีดี

ในการเขียนข้อมูลเพื่อควบคุมให้แอลซีดี แสดงผลตามที่ใช้งานต้องการ ต้องส่งคำสั่ง (instruction) แล้วกำหนดโหมดการทำงานให้แก่แอลซีดี ก่อน จากนั้นจึงเริ่มส่งข้อมูล (data) ที่ต้องการแสดงผล เนื่องจากบัสข้อมูลของแอลซีดี มี 8 เส้นคือ D0-D7 และใช้เป็นทางผ่านของทั้งคำสั่งและข้อมูล ดังนั้นในการส่งคำสั่งและข้อมูลจึงต้องอาศัยการกำหนดสัญญาณลอจิกที่ขา RS ถ้าหากขา RS ได้ลอจิก “0” หมายความว่า ข้อมูลที่ป้อนให้แก่แอลซีดี ขณะนั้นเป็นคำสั่งในทางตรงข้าม หากเขา RS ได้รับลอจิก “1” ข้อมูลที่ป้อนให้ขณะนั้นเป็นข้อมูลที่ใช้ในการแสดงผล

เมื่อต้องการเขียนหรือข้อมูลใน CGRAM และ DDRAM เริ่มต้นต้องกำหนดแอดเดรสที่ต้องการอ่านหรือเขียนก่อน โดยใช้คำสั่งเลือกแอดเดรส จากนั้นกำหนดให้ขา RS เป็น “1” เพื่อแจ้งให้ตัวควบคุมภายในแอลซีดี ทราบว่าข้อมูลที่ปรากฏต่อไปนี้เป็นข้อมูลปกติไม่ใช่คำสั่ง

ในกรณีที่ต้องการอ่านข้อมูลต้องกำหนดให้ขา $R/\overline{W}$ เป็น “1” ข้อมูลขนาด 8 บิต (หรือ 4 บิต) ก็จะปรากฏบนบัสข้อมูล โดยข้อมูลที่อ่านออกมาได้จะเป็นข้อมูลจากแอดเดรสของ CGRAM

ในกรณีที่ต้องการเขียนข้อมูล เมื่อกำหนดแอดเดรสและป้อนลอจิก “1” ให้ขา RS แล้วต้องกำหนดให้ขา $R/\overline{W}$ เป็น “0” ข้อมูลที่อยู่บนบัสข้อมูลจะถูกเขียนลงในรีจิสเตอร์ DR จากนั้นจึงถ่ายทอดลงไป DDRAM ต่อไป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จังหวะการทำงานของ แอลซีดี

ในการติดต่อกับแอลซีดีจะต้องมีการหน่วงเวลาหลังเวลาจากที่ทำการส่งรหัสคำสั่งหรือข้อมูลเนื่องจากต้องรอให้คอนโทรลเลอร์ภายในแอลซีดีแปลความหมายของรหัสคำสั่งและทำงานตามคำสั่งให้เรียบร้อยก่อน จากนั้นจึงจะรับข้อมูลหรือดำเนินการต่อไป

ดังนั้น ในการใช้แอลซีดีผู้เขียนโปรแกรมต้องมีโปรแกรมเพื่อหน่วงเวลารอให้แอลซีดีพร้อมทำงานด้วยโดยเมื่อเริ่มจ่ายไฟให้แก่แอลซีดีต้องรอประมาณ 10 มิลลิวินาที เพื่อให้แอลซีดีทำการเตรียมความพร้อมหรืออินิเชียล (initial) หลังจากนั้นก็จะกำหนดลอจิกให้แก่ขา RS ของแอลซีดีแล้วต้องหน่วงเวลาอีกประมาณ 2 มิลลิวินาทีเพื่อให้คอนโทรลเลอร์ในแอลซีดีแปลความหมายของลอจิกที่ขา RS ว่า ข้อมูลต่อไปที่จะได้รับนั้นเป็นรหัสคำสั่งหรือเป็นข้อมูลที่ต้องการแสดงผล จากนั้นจะเป็นการส่งข้อมูลมารอที่บัสข้อมูล D0-D7 (กรณีทำงานในโหมด 8 บิต) ขั้นตอนต่อไปจะเป็นการส่งสัญญาณพัลซ์ไปที่ขา E เพื่ออีนาเบิลแอลซีดีให้รับข้อมูลจากบัสข้อมูลเข้าไป โดยพัลซ์ที่ป้อนเข้าที่ขา E ของแอลซีดีต้องเป็นพัลซ์ขอบขาขึ้น จากนั้นทำการหน่วงเวลา 2 มิลลิวินาที

เครื่องมือและอุปกรณ์

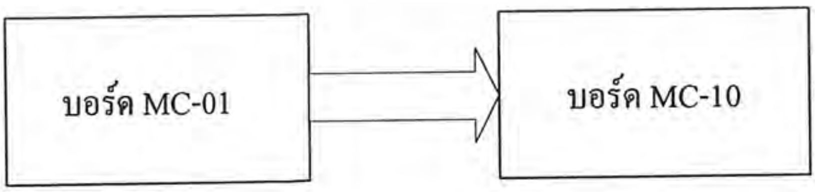
1) บอร์ดไมโครคอนโทรลเลอร์ MCS51-S8252 MC-01

2) บอร์ดแสดงผลแบบผลึกเหลว MC-10

3) คอมพิวเตอร์ที่ติดตั้งวินโดว์ 95 หรือ 98 พร้อมโปรแกรม wins8252

ลำดับขั้นการทดลอง

1) ทำการเชื่อมต่อบอร์ดไมโครคอนโทรลเลอร์ MCS51-S8252 MC-01 กับบอร์ดขับโมดูลแบบผลึกเหลว MC-10 ที่พอร์ต P2 โดยต่อขา E ที่ P1.7 และต่อขา RS ที่ P1.6 โดยต่อตามรูปที่ 9.2



รูปที่ 9.2 การเชื่อมต่อบอร์ดโมดูลแบบผลึกเหลวกับบอร์ดควบคุม

2.)เปิดโปรแกรม wins8252 ทำการตั้งชื่อและบันทึกโดยใช้ไฟล์ชื่อ Lab9.Asm

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3) เขียนโปรแกรมดังต่อไปนี้

```
            LCD_EN        BIT    P1.7
            LCD_RS        BIT    P1.6
            LCD_ADDR      EQU    030H
            LCD_DATA      EQU    031H

                  ORG            0000H
LOOP:             MOV            LCD_ADDR,#000H
                  ACALL          SET_ADDR_LCD
                  MOV            DPTR,#DATA1
                  ACALL          WRLINE_LCD
                  MOV            LCD_ADDR,#040H
                  ACALL          SET_ADDR_LCD
                  MOV            DPTR,#DATA2
                  ACALL          WRLINE_LCD
                  ACALL          DELAY_1S
                  ACALL          DELAY_1S
                  ACALL          LCD_OFF
                  ACALL          DELAY_1S
                  ACALL          LCD_ON
                  ACALL          DELAY_1S
                  ACALL          LCD_OFF
                  ACALL          DELAY_1S
                  ACALL          LCD_ON
                  ACALL          DELAY_1S
                  MOV            LCD_ADDR,#000H
                  ACALL          SET_ADDR_LCD
                  MOV            DPTR,#DATA3
                  ACALL          WRLINE_LCD
                  MOV            LCD_ADDR,#040H
                  ACALL          SET_ADDR_LCD
                  MOV            DPTR,#DATA4
                  ACALL          WRLINE_LCD
                  ACALL          DELAY_1S
                  ACALL          DELAY_1S
                  ACALL          LCD_OFF
                  ACALL          DELAY_1S
                  ACALL          LCD_ON
                  ACALL          DELAY_1S
                  ACALL          LCD_OFF
                  ACALL          DELAY_1S
                  ACALL          LCD_ON
                  ACALL          DELAY_1S
                  AJMP           LOOP
LCD_CLR:          CLR            LCD_RS
                  MOV            P2,#01H
                  ACALL          LCD_CLK
                  RET
LCD_OFF:          CLR            LCD_RS
                  MOV            P2,#08H
                  ACALL          LCD_CLK
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                    RET
LCD_CLK:            SETB        LCD_EN
                    ACALL       LCD_DELAY
                    CLR         LCD_EN
                    ACALL       LCD_DELAY
                    RET
LCD_ON:             CLR         LCD_RS
                    MOV         P2,#0CH
                    RET
SET_ADDR_LCD:       CLR         LCD_RS
                    MOV         A,LCD_ADDR
                    SETB        ACC.7
                    MOV         P2,A
                    ACALL       LCD_CLK
                    RET
WRLINE_LCD :        MOV         R0,#0
WRLINE_LCD_1 :      SETB        LCD_RS
                    CLR         A
                    MOVC        A,@A+DPTR
                    MOV         P2,A
                    ACALL       LCD_CLK
                    INC         DPTR
                    INC         R0
                    CJNE        R0,#8,WRLINE_LCD_1
                    ACALL       LCD_ON
                    RET
LCD_DELAY :         MOV         7,#002
LCD_DELAY_1 :       MOV         6,#0E6H
LCD_DELAY_2 :       NOP
                    NOP
                    DJNZ        R6,LCD_DELAY_2
                    DJNZ        R7,LCD_DELAY_1
                    RET
DELAY_10MS:         MOV         7,#010
DELAY_10MS_1:       MOV         6,#0E6H
DELAY_10MS_2:       NOP
                    NOP
                    DJNZ        R6,DELAY_10MS_2
                    DJNZ        R7,DELAY_10MS_1
                    RET
DELAY_1S:           MOV         5,#100
DELAY_1S_1:         ACALL       DELAY_10MS
                    DJNZ        R5,DELAY_1S_1
                    RET
DATA1 :       DB   '      #   KMI'
DATA2 :       DB   'TL   #       '
DATA3 :       DB   'ED..ENGI     '
DATA4 :       DB   'NEER..21     '
                    END
```

รูปที่ 9.3 โปรแกรมขับโมดูลแบบผลึกเหลว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4) ทำการบันทึกและแอสเซมเบลอร์ไฟล์จาก Lab9.Asm เป็น Lab9.Hex

5) ทำการโปรแกรมลงบนไมโครคอนโทรลเลอร์

6) กดสวิตช์รีเซตแล้วสังเกตผลการทดลอง

สรุปผลการทดลอง

คำถามท้ายการทดลอง

จากการทดลองให้เขียนโปรแกรมแสดงชื่อนักศึกษาแสดงออกที่พอร์ต P.1 โดยให้กระพริบจำนวน 2 ครั้ง หน่วงเวลาการกระพริบครั้งละ 1 วินาที วนรอบการทำงานไม่รู้จบ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

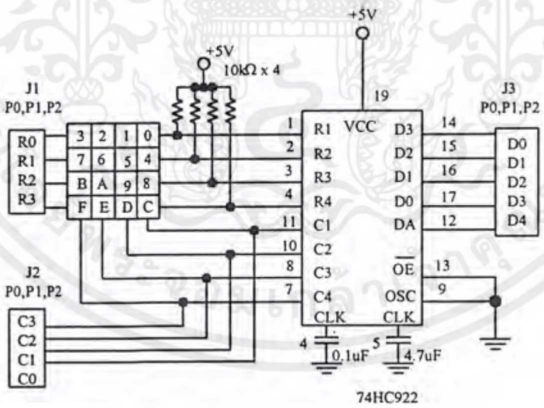
ใบงานที่ 10
การเชื่อมต่อคีย์สวิตช์กับแอลอีดีแบบ 7 ส่วน

วัตถุประสงค์

1. สามารถเขียนโปรแกรมเพื่อขับแอลอีดีตัวเลข 7 ส่วนแบบตัวเดี่ยวและมัลติเพล็กซ์ได้
2. สามารถเขียนโปรแกรมเพื่ออ่านค่าหรือรับค่าการกดคีย์สวิตช์ได้
3. สามารถนำคีย์สวิตช์ไปประยุกต์ใช้งานกับบอร์ดอื่นๆ ได้

ทฤษฎีเบื้องต้น

วงจรคีย์สวิตช์จะใช้สวิตช์ 4X4 จุด รวมเป็น 16 ตัว จะใช้การถอดรหัสโดยใช้ต่อเข้าโดยตรงกับพอร์ตของ MCS-51 โดยแบ่งเป็นคอลัมน์ 4 บิต และแถวอีก 4 บิต การต่อกับวงจรสามารถทำได้โดยใช้สัญญาณ 8 เส้น โดย 4 เส้นเป็นสายสัญญาณคอลัมน์ต่อเข้ากับพอร์ตของ MCS-51 และอีก 4 เส้นเป็นสายแถวต่อเข้ากับพอร์ตของ MCS-51 โดยใช้พอร์ตเดียวกัน โดยแบ่งเป็น 4 บิตบน และ 4 บิตล่าง



รูปที่ 10.1วงจรคีย์สวิตช์

การเชื่อมต่อคีย์สวิตซ์เข้ากับไมโครคอนโทรลเลอร์ ใช้ 2 พอร์ต ของไมโครคอนโทรลเลอร์ต่อเข้ากับคีย์สวิตซ์ทั้ง 8 เส้นคือ สายของคอลัมน์ 4 สาย C0-C3 และทางสายแถวอีก 4 เส้น คือ R0-R3 โดยเฉพาะแถวต้องต่อตัวต้านทานพูลอัพไว้ เพื่อกำหนดสภาวะเริ่มต้นที่ไม่มีการกดคีย์สวิตช์ โดยไมโครคอนโทรลเลอร์จะทำการส่งข้อมูล “0” ไปยังคอลัมน์ ในทุกครั้งที่มีการส่งข้อมูลไปยังสายคอลัมน์ไมโครคอนโทรลเลอร์จะทำการอ่านค่าที่แถวเข้ามาด้วย หากไม่มีการกดคีย์จะเป็น “1”

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

"1" ถ้ามีการกดคีย์ค่าของแถวจะเปลี่ยนไปเป็นการแจ้งให้ทราบว่ามีการกดคีย์แล้ว จากนั้นไมโครคอนโทรลเลอร์จะทำการค้นหาตำแหน่งต่อไป โดยการค้นหาตำแหน่งที่ได้มาอย่างแรกคือตำแหน่งของคีย์นั้น จากนั้นก็จะนำค่าตำแหน่งนั้นไปเปิดตารางข้อมูล เพื่อจะได้หมายเลขของคีย์อย่างแท้จริง

เครื่องมือและอุปกรณ์

1) บอร์ดไมโครคอนโทรลเลอร์ MCS51-S8252

2) บอร์ดแอลอีดีแบบ 7 ส่วน MC-06

3) บอร์ดคีย์สวิตช์ MC-15

4) คอมพิวเตอร์ที่ติดตั้งวินโดว์ 95 หรือ 98 พร้อมโปรแกรม wins8252

ลำดับขั้นการทดลอง

1) ทำการเชื่อมต่อบอร์ดไมโครคอนโทรลเลอร์ MCS51-S8252 MC-01 กับบอร์ดแอลอีดีแบบ 7 ส่วน MC-06 แสดงค่าที่พอร์ต P.0 และ DSP 1 ที่ P1.2 และ DSP 2 ที่ P1.3 และเชื่อมต่อกับบอร์ดบอร์ดคีย์สวิตช์ MC-15 ที่พอร์ต P.2 ดังรูปที่ 10.2

รูปที่ 10.2 การเชื่อมต่อบอร์ดคีย์สวิตซ์กับบอร์ดแอลอีดีแบบ 7 ส่วน

2) กดสวิตช์ Power On

3) เปิดโปรแกรม wins8252 ทำการตั้งชื่อและบันทึกโดยใช้ไฟล์ชื่อ Lab10.Asm

4) เขียนโปรแกรมดังต่อไปนี้

```
DSP1            BIT     P1.2
DSP2            BIT     P1.3
DRIVER_LE       BIT     P1.4
KPAD_ROW0       BIT     P2.0
KPAD_ROW1       BIT     P2.1
KPAD_ROW2       BIT     P2.2
KPAD_ROW3       BIT     P2.3
KPAD_COL3       BIT     P2.4
KPAD_COL2       BIT     P2.5
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
          KPAD_COL1           BIT      P2.6
          KPAD_COL0           BIT      P2.7
          LEFT_SHF            BIT      P3.2
          RIGHT_SHF           BIT      P3.3
     ;***************************
          FLAG                EQU      02FH
          KEYPRESS            BIT      FLAG.0
          R_SHF               BIT      FLAG.1
          DSP1_BUFFER         EQU      030H
          DSP2_BUFFER         EQU      031H
          KPAD_DATA           EQU      032H
     ;****************************
               ORG     000H
               MCV     P0,#00H
               MOV     P1,#0EFH
               MOV     P2,#0FFH
               MOV     P3,#1FH
MAIN:          MOV     DSP1_BUFFER,#0
               MOV     DSP2_BUFFER,#0
               MOV     FLAG,#0
               MOV     P0,#080H
               ACALL   DRIVE_LE
LOOP:          ACALL   GET_KEYPAD
               MOV     A,KPAD_DATA
               JZ      NEXT
               JB      KEYPRESS,SHOW
               SETB    KEYPRESS
               JB      R_SHF,RIGHT_SHIFT
LEFT_SHIFT:    MOV     DSP2_BUFFER,DSP1_BUFFER
               MOV     DSP1_BUFFER,KPAD_DATA
               AJMP    SHOW
RIGHT_SHIFT:   MOV     DSP1_BUFFER,DSP2_BUFFER
               MOV     DSP2_BUFFER,KPAD_DATA
               AJMP    SHOW
NEXT:          CLR     KEYPRESS
SHOW:          ACALL   SHOW_DSP
               JB      RIGHT_SHF,NEXT_1
               SETB    R_SHF
               MOV     P0,#001H
               ACALL   DRIVE_LE
NEXT_1:        JB      LEFT_SHF,LOOP
               CLR     R_SHF
               MOV     P0,#080H
               ACALL   DRIVE_LE
               JB      RIGHT_SHF,LOOP
               MOV     DSP1_BUFFER,#0
               MOV     DSP2_BUFFER,#0
               AJMP    LOOP
*******************************************
;         SCANKEY
;******************************************
GET_KEYPAD:    MOV     P2,#0FFH
               MOV     KPAD_DATA,#00H
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
CHK_COL0:       CLR     KPAD_COL0
                MOV     A,P2
                ANL     A,#00FH
                CJNE    A,#00FH,COL0_DETECT
                AJMP    CHK_COL1
COL0_DETECT:    MOV     KPAD_DATA,#01H
                AJMP    GET_ROW
CHK_COL1:       SETB    KPAD_COL0
                CLR     KPAD_COL1
                MOV     A,P2
                ANL     A,#00FH
                CJNE    A,#00FH,COL1_DETECT
                AJMP    CHK_COL2
COL1_DETECT:    MOV     KPAD_DATA,#02H
                AJMP    GET_ROW
CHK_COL2:       SETB    KPAD_COL1
                CLR     KPAD_COL2
                MOV     A,P2
                ANL     A,#00FH
                CJNE    A,#00FH,COL2_DETECT
                AJMP    CHK_COL3
COL2_DETECT:    MOV     KPAD_DATA,#03H
                AJMP    GET_ROW
CHK_COL3:       SETB    KPAD_COL2
                CLR     KPAD_COL3
                MOV     A,P2
                ANL     A,#00FH
                CJNE    A,#00FH,COL3_DETECT
                RET
COL3_DETECT:    MOV     KPAD_DATA,#04H
GET_ROW:        CLR     KPAD_COL0
                CLR     KPAD_COL1
                CLR     KPAD_COL2
                CLR     KPAD_COL3
                JB      KPAD_ROW0,CHK_ROW1
                RET
CHK_ROW1:       JB      KPAD_ROW1,CHK_ROW2
                MOV     A,KPAD_DATA
                ADD     A,#3
                MOV     KPAD_DATA,A
                RET
CHK_ROW2:       JB      KPAD_ROW2,CHK_ROW3
                MOV     A,KPAD_DATA
                ADD     A,#6
                MOV     KPAD_DATA,A
                RET
CHK_ROW3:       MOV     A,KPAD_DATA
                ADD     A,#9
                MOV     KPAD_DATA,A
                RET
;***********************************************
;     show 7 segment
;***********************************************
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
SHOW_DSP:           MOV     R4,#5
SCAN_DSP_LOOP:      MOV     A,DSP1_BUFFER
                    MOV     DPTR,#DSP_BLANK
                    MOVC    A,@A+DPTR
                    MOV     P0,A
                    CLR     DSP1
                    ACALL   DELAY_1MS
                    SETB    DSP1
                    MOV     A,DSP2_BUFFER
                    MOV     DPTR,#DSP_BLANK
                    MOVC    A,@A+DPTR
                    MOV     P0,A
                    ACALL   DELAY_1MS
                    SETB    DSP2
                    DJNZ    R4,SCAN_DSP_LOOP
                    RET
DRIVE_LE_CLK:       SETB    DRIVER_LE
                    NOP
                    CLR     DRIVER_LE
                    NOP
                    RET
DELAY_1MS:          MOV     6,#0E6H
DELAY_1MS_1:        NOP
                    NOP
                    DJNZ    R6,DELAY_1MS_1
                    RET
;****************************************
DSP_BLANK:          DB          08H
DSP_NUM0:           DB          3FH
DSP_NUM1:           DB          06H
DSP_NUM2:           DB          5BH
DSP_NUM3:           DB          4FH
DSP_NUM4:           DB          66H
DSP_NUM5:           DB          6DH
DSP_NUM6:           DB          7DH
DSP_NUM7:           DB          07H
DSP_NUM8:           DB          7FH
DSP_NUM9:           DB          6FH
DSP_STAA:           DB          77H
DSP_NUMB:           DB          7CH
DSP_NUMC:           DB          39H
DSP_STAD:           DB          5EH
DSP_NUME:           DB          79H
DSP_NUMF:           DB          71H

                    END
```

รูปที่ 10.3 โปรแกรมแสดงค่าตัวเลขตามค่าคีย์สวิตช์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5) ทำการบันทึกและแอสเซมเบลอร์ไฟล์จาก Lab10.Asm เป็น Lab10.Hex

6) ทำการโปรแกรมลงบนไมโครคอนโทรลเลอร์

7) กดสวิตช์รีเซตแล้วสังเกตผลการทดลอง

สรุปผลการทดลอง

คำถามท้ายการทดลอง

จากโปรแกรมการทดลองจงเขียนโปรแกรมการแสดงค่าตัวเลขตามค่าคีย์ที่ทำการกดโดยแสดงผลด้วยแอลอีดีแบบ 7 ส่วน โดยกดค่าคีย์ใดให้แสดงค่าคีย์นั้นทั้ง 8 หลัก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ใบงานที่ 11
การเชื่อมต่อกับวงจรแปลงสัญญาณดิจิตอลเป็นสัญญาณแอนะลอกแบบ R-2R

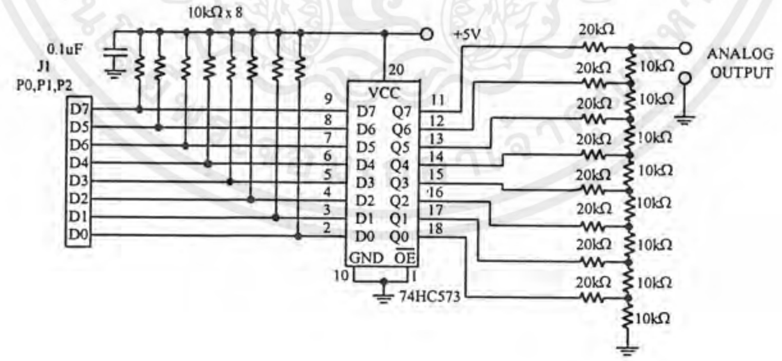
วัตถุประสงค์

1. สามารถเขียนโปรแกรมเพื่อใช้งานวงจรแปลงสัญญาณดิจิตอลเป็นสัญญาณแอนะลอกแบบ R-2Rได้

2. สามารถเขียนโปรแกรมเพื่อประยุกต์ใช้กับบอร์ดอื่นๆ ได้

ทฤษฎีเบื้องต้น

วงจรแปลงสัญญาณดิจิตอลเป็นแอนะลอกอีกแบบหนึ่งที่ได้รับความนิยม คือ แบบ R-2R แลดซ์เดอร์ จะใช้ตัวต้านทานต่อขนานกันหลายสาขา กระแสในวงจรจะถูกเฉลี่ยกันในวงจรด้วยอัตราส่วนของตัวต้านทานคงที่คือ R และ 2R ในการใช้งานจะต่อเข้ากับพอร์ตใดก็ได้ โดยผ่านวงจรแลตช์ 74HC574 เพื่อได้ข้อมูลดิจิตอลออกไปยังวงจรแปลงสัญญาณดิจิตอลเป็นสัญญาณแอนะลอกจะแปลงสัญญาณเป็นสัญญาณแอนะลอกให้เอง ได้ดังรูปที่ 11.1



รูปที่ 11.1 วงจรแปลงสัญญาณดิจิตอลเป็นแอนะลอกแบบ R-2R

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เครื่องมือและอุปกรณ์

1) บอร์ดไมโครคอนโทรลเลอร์ MCS51-S8252 MC-01
2) บอร์ดวงจรแปลงสัญญาณดิจิตอลเป็นแอนะลอกแบบ R-2R MC-11
3) ออสซิลโลสโคป พร้อมสายวัด
4) คอมพิวเตอร์ที่ติดตั้งวินโดว์ 95 หรือ 98 พร้อมโปรแกรม wins8252

ลำดับขั้นการทดลอง

1) ทำการเชื่อมต่อบอร์ดไมโครคอนโทรลเลอร์MCS51-S8252 MC-01 กับบอร์ดวงจรแปลงสัญญาณดิจิตอลเป็นแอนะลอกแบบ R-2R MC-11 โดยต่อที่พอร์ต P.0 โดยต่อตามรูปที่ 11.2

รูปที่ 11.2 การเชื่อมต่อบอร์ดแปลงสัญญาณดิจิตอลเป็นแอนะลอกกับบอร์ดควบคุม

2) เปิดโปรแกรม wins8252 ทำการตั้งชื่อและบันทึกโดยใช้ไฟล์ชื่อ Lab11.Asm

3) เขียนโปรแกรมดังต่อไปนี้

```
        ORG     0000H
START:  CLR     A
        MOV     A,#00H
        MOV     P0,A
        ACALL   DELAY
        CLR     A
        MOV     A,#01H
        MOV     P0,A
        ACALL   DELAY
        CLR     A
        MOV     A,#02H
        MOV     P0,A
        ACALL   DELAY
        MOV     A,#03H
        MOV     P0,A
        ACALL   DELAY
        CLR     A
        MOV     A,#04H
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        MOV       P0,A
        ACALL     DELAY
        CLR       A
        MOV       A,#05H
        MOV       P0,A
        ACALL     DELAY
        CLR       A
        MOV       A,#06H
        MOV       P0,A
        ACALL     DELAY
        CLR       A
        MOV       A,#07H
        MOV       P0,A
        ACALL     DELAY
        CLR       A
        MOV       A,#08H
        MOV       P0,A
        ACALL     DELAY
        CLR       A
        MOV       A,#09H
        MOV       P0,A
        ACALL     DELAY
        CLR       A
        MOV       A,#0AH
        MOV       P0,A
        ACALL     DELAY
        CLR       A
        MOV       A,#0BH
        MOV       P0,A
        ACALL     DELAY
        CLR       A
        MOV       A,#0CH
        MOV       P0,A
        ACALL     DELAY
        CLR       A
        MOV       A,#0EH
        MOV       P0,A
        ACALL     DELAY
        CLR       A
        MOV       A,#0FH
        MOV       P0,A
        ACALL     DELAY
        CLR       A
        MOV       A,#1FH
        MOV       P0,A
        ACALL     DELAY
        CLR       A
        MOV       A,#2FH
        MOV       P0,A
        ACALL     DELAY
        CLR       A
        MOV       A,#3FH
        MOV       P0,A
        ACALL     DELAY
        CLR       A
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
MOV     A,#4FH
MOV     P0,A
ACALL   DELAY
CLR     A
MOV     A,#5FH
MOV     P0,A
ACALL   DELAY
CLR     A
MOV     A,#6FH
MOV     P0,A
ACALL   DELAY
CLR     A
MOV     A,#7FH
MOV     P0,A
ACALL   DELAY
CLR     A
MOV     A,#8FH
MOV     P0,A
ACALL   DELAY
MOV     A,#9FH
MOV     P0,A
ACALL   DELAY
CLR     A
MOV     A,#0AFH
MOV     P0,A
ACALL   DELAY
CLR     A
MOV     A,#0BFH
MOV     P0,A
ACALL   DELAY
CLR     A
MOV     A,#0CFH
MOV     P0,A
ACALL   DELAY
CLR     A
MOV     A,#0DFH
MOV     P0,A
ACALL   DELAY
CLR     A
MOV     A,#0EFH
MOV     P0,A
ACALL   DELAY
CLR     A
MOV     A,#0FFH
MOV     P0,A
ACALL   DELAY
CLR     A
MOV     A,#0EFH
MOV     P0,A
ACALL   DELAY
CLR     A
MOV     A,#0DFH
MOV     P0,A
ACALL   DELAY
CLR     A
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
MOV     A,#0CFH
MOV     P0,A
ACALL   DELAY
CLR     A
MOV     A,#0BFH
MOV     P0,A
ACALL   DELAY
CLR     A
MOV     A,#0AFH
MOV     P0,A
ACALL   DELAY
CLR     A
MOV     A,#9FH
MOV     P0,A
ACALL   DELAY
CLR     A
MOV     A,#8FH
MOV     P0,A
ACALL   DELAY
CLR     A
MOV     A,#7FH
MOV     P0,A
ACALL   DELAY
CLR     A
MOV     A,#6FH
MOV     P0,A
ACALL   DELAY
CLR     A
MOV     A,#5FH
MOV     P0,A
ACALL   DELAY
CLR     A
MOV     A,#4FH
MOV     P0,A
ACALL   DELAY
CLR     A
MOV     A,#3FH
MOV     P0,A
ACALL   DELAY
CLR     A
MOV     A,#2FH
MOV     P0,A
ACALL   DELAY
CLR     A
MOV     A,#1FH
MOV     P0,A
ACALL   DELAY
CLR     A
MOV     A,#0FH
MOV     P0,A
ACALL   DELAY
CLR     A
MOV     A,#0EH
MOV     P0,A
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
ACALL     DELAY
CLR       A
MOV       A,#0DH
MOV       P0,A
ACALL     DELAY
CLR       A
MOV       A,#0CH
MOV       P0,A
ACALL     DELAY
CLR       A
MOV       A,#0BH
MOV       P0,A
ACALL     DELAY
CLR       A
MOV       A,#0AH
MOV       P0,A
ACALL     DELAY
CLR       A
MOV       A,#09H
MOV       P0,A
ACALL     DELAY
CLR       A
MOV       A,#08H
MOV       P0,A
ACALL     DELAY
CLR       A
MOV       A,#07H
MOV       P0,A
ACALL     DELAY
MOV       A,#06H
MOV       P0,A
ACALL     DELAY
CLR       A
MOV       A,#05H
MOV       P0,A
ACALL     DELAY
CLR       A
MOV       A,#04H
MOV       P0,A
ACALL     DELAY
CLR       A
MOV       A,#04H
MOV       P0,A
ACALL     DELAY
CLR       A
MOV       A,#03H
MOV       P0,A
ACALL     DELAY
CLR       A
MOV       A,#02H
MOV       P0,A
CLR       A
MOV       A,#01H
MOV       P0,A
ACALL     DELAY
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
     CLR        A
     MOV        A,#00H
     MOV        P0,A
     ACALL      DELAY
     JMP        START
DELAY:   MOV    R3,#20H
DELAY_1:MOV     R4,#50H
      DJNZ      R4,$
      DJNZ      R3,DELAY_1
      RET
      END
```

รูปที่ 11.3 โปรแกรมแปลงสัญญาณดิจิตอลเป็นสัญญาณแอนะลอก

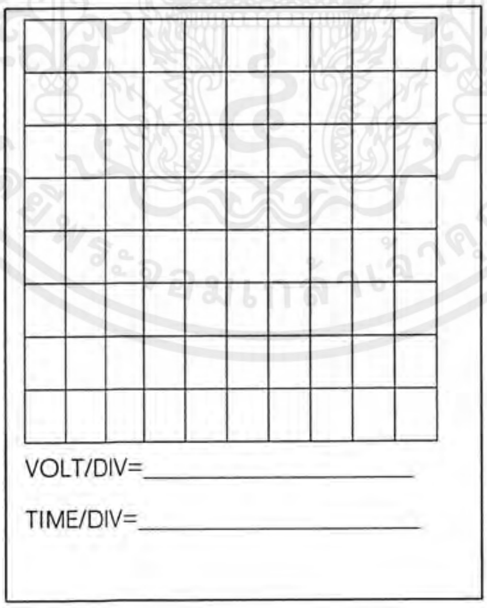
4) ทำการบันทึกและแอสเซมเบลอร์ไฟล์จาก Lab11.Asm เป็น Lab11.Hex

5) ทำการโปรแกรมลงบนไมโครคอนโทรลเลอร์

6) กดสวิตช์รีเซตแล้วสังเกตผลการทดลอง

นำออสซิลโลสโคปมาต่อวัดสัญญาณเอาต์พุต บันทึกรูปสัญญาณที่เกิดขึ้นในกราฟ

ตารางที่ 1 บันทึกผลการทดลอง



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สรุปผลการทดลอง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ใบงานที่ 12
การขยายพอร์ตอินพุตเอาต์พุตด้วยอุปกรณ์ระบบบัส I^2C

วัตถุประสงค์

1. สามารถเขียนโปรแกรมเพื่อติดต่อกับอุปกรณ์ที่ใช้ในการเชื่อมต่อบนระบบบัส I^2C ได้
2. สามารถเขียนโปรแกรมเพื่อประยุกต์ใช้กับบอร์ดอื่นๆ ได้

ทฤษฎีเบื้องต้น

I^2C ย่อมาจาก Inter IC Communication หมายถึงการติดต่อสื่อสารระหว่างไอซีบัส I^2C ได้ สามารถติดต่อ สั่งงาน และควบคุมภายใต้สายสัญญาณเพียง 2 เส้น เส้นหนึ่งคือ สายข้อมูลอนุกรม หรือ SDA ส่วนสายสัญญาณอีกสายหนึ่งคือสายสัญญาณนาฬิกาหรือ SCL

สาย SDA และ สาย SCL เป็นสายสัญญาณ 2 ทิศทาง ต้องมีการต่อตัวความต้านทานพูลอัป กับแรงดัน 5 โวลต์อยู่ตลอดเวลา เพื่อให้สายมีสถานะเป็นลอจิกสูง ในขณะที่มีการติดต่อใช้งาน

การกำหนดการติดต่อต้องมีรูปแบบการติดต่อที่เป็นข้อตกลงพื้นฐานดังนี้

- อุปกรณ์ที่สร้างข้อมูลหรือส่งข้อมูลเรียกว่า ตัวส่ง
- อุปกรณ์ที่เป็นผู้รับข้อมูลเรียกว่าตัวรับ
- อุปกรณ์ที่ทำหน้าที่ควบคุมจังหวะการติดต่อบนบัส I^2C เรียกว่ามาสเตอร์
- อุปกรณ์ที่ถูกควบคุมหรืออุปกรณ์ต่อพ่วงเข้าบนบัส I^2C เรียกว่า สเลฟ

ข้อกำหนด 2 ประการสำคัญของการติดต่อบนบัส I^2C สาย SDA และ สาย SCL คือ

(1) การถ่ายทอดข้อมูลจะเกิดขึ้นได้เมื่อบัสว่างเท่านั้น

(2) ในระหว่างการถ่ายทอดข้อมูลเมื่อใดก็ตามที่สาย SCL มีสถานะเป็นลอจิกสูง

การต่ออุปกรณ์ระบบบัส I^2C กับไมโครคอนโทรลเลอร์

สามารถทำได้โดยใช้ขาเพียง 2 ขา โดยกำหนดให้ขาหนึ่งเป็น SDA และ อีกขาหนึ่งเป็น SCL และต่อตัวความต้านทาน ประมาณ 4.7 กิโลโอห์ม ที่ขาทั้ง 2 ขาเพียงเท่านี้ก็สามารถติดต่อกับอุปกรณ์ระบบบัส I^2C ได้แล้ว

การเขียนโปรแกรมติดต่อบัส I^2C

เริ่มต้นด้วยการสร้างสภาวะมาตรฐานของบัส อันประกอบด้วย สภาวะเริ่มต้น, สภาวะสิ้นสุดการส่งข้อมูล, สภาวะหยุด, และสัญญาณนาฬิกาบนขา SCL

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การสร้างสภาวะเริ่มต้น

1) ทำให้ขา SDA และ ขา SCL มีลอจิกเป็น “1”

2) ทำให้ขา SDA เป็น “0” โดย ขา SCL เป็น “1”

3) กำหนดให้ขาทั้งสองขาเป็น “0” ทั้งคู่ ซึ่งพร้อมในการติดต่อ

การสร้างสภาวะหยุด

1) กำหนดให้ขาทั้งสองขาเป็น “0” ทั้งคู่

2) ทำให้ขา SCL เป็น “0” โดยที่ ขา SDA ยังเป็น “1”

3) ทำให้ขา SDA เป็น “1” ซึ่งทำให้บัสกลับสู่สภาวะว่าง

การส่งข้อมูลลอจิก “0”

4) ทำให้ขา SDA เป็น “0” สำหรับส่งข้อมูลลอจิก “0”

5) ทำให้ขา SCL เป็น “1” สำหรับป้อนสัญญาณนาฬิกาโดยที่ ขา SDA ยังเป็น “0” อยู่

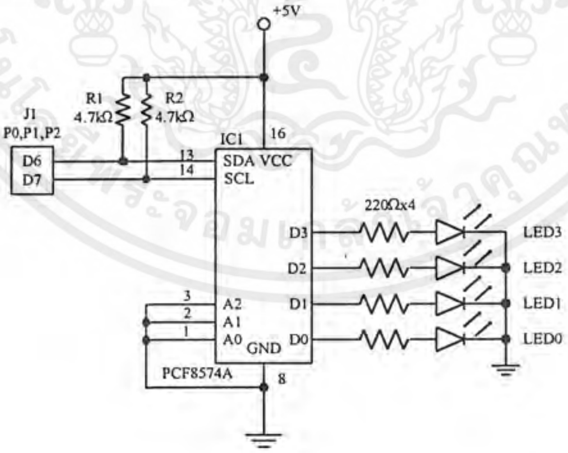
6) ทำให้ขา SCL กลับมาเป็น “0”

การส่งข้อมูลลอจิก “1”

7) ทำให้ขา SDA เป็น “1” สำหรับส่งข้อมูลลอจิก “1”

8) ทำให้ขา SCL เป็น “1” สำหรับป้อนสัญญาณนาฬิกาโดยที่ ขา SDA ยังเป็น “1” อยู่

9) ทำให้ขา SCL กลับมาเป็น “0”



รูปที่ 12.1 การขยายพอร์ตอินพุตเอาต์พุตด้วยอุปกรณ์ระบบบัส I^2C

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เครื่องมือและอุปกรณ์

1) บอร์ดไมโครคอนโทรลเลอร์ MCS51-S8252 MC-01

2) บอร์ดแสดงผลแบบผลึกเหลว MC-10

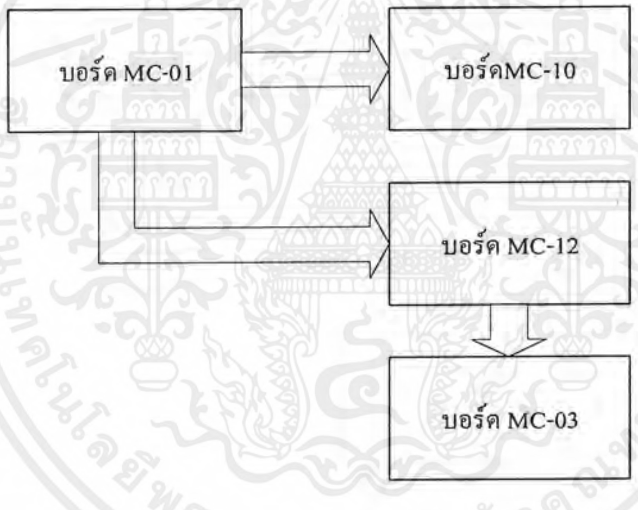
3) บอร์ดขับไดโอดเปล่งแสงชนิดคอมมอนอาโนด MC-03

4) บอร์ดขยายพอร์ตบนระบบ I^2C MC-12

5) คอมพิวเตอร์ที่ติดตั้งวินโดว์ 95 หรือ 98 พร้อมโปรแกรม wins8252

ลำดับขั้นการทดลอง

1) ทำการเชื่อมต่อบอร์ดไมโครคอนโทรลเลอร์ MCS51-S8252 MC-01 กับบอร์ดแสดงผลแบบผลึกเหลว MC-10 ที่พอร์ต P0 โดยต่อขา E ที่ P2.7 และต่อขา RS ที่ P2.6 ตามรูปที่ 12.2



รูปที่ 12.2 การเชื่อมต่อบอร์ดขยายพอร์ตกับบอร์ดควบคุม

2) บอร์ดขยายพอร์ตบนระบบ I^2C MC-12 ที่พอร์ต P.1 โดยที่ต่อสาย SDA ที่ P1.0 และ SCL ที่ P1.1

3) ทำการเชื่อมต่อบอร์ดขับไดโอดเปล่งแสงแบบแอาโนดร่วม MC-03 ที่เอาต์พุตของบอร์ดขยายพอร์ตบนระบบ I^2C ที่บิตที่ 0 ถึงบิตที่ 3

4) เปิดโปรแกรม wins8252 ทำการตั้งชื่อและบันทึกโดยใช้ไฟล์ชื่อ Lab12.Asm

5) เขียนโปรแกรมดังต่อไปนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
LCD_EN          BIT     P2.7
LCD_RS          BIT     P2.6
FLAG            EQU     02FH
I2C_ACK         BIT     FLAG.0
LCD_ADDR        EQU     030H
LCD_DATA        EQU     031H
LCD_PTR         EQU     032H
I2C_ADDR        EQU     033H
I2C_DATA        EQU     034H
IO_DATA         EQU     035H
OUT_DATA        EQU     036H
PCF8574_ID      EQU     070H
;************************************
;MAIN PROGRAM
;************************************
                ORG     0000H
                MOV         P0,#00H
                MOV         P1,#0EFH
                MOV         P2,#0FFH
                MOV         P3,#01FH
MAIN:           ACALL       INIT_LCD
                MOV         LCD_ADDR,#000H
                ACALL       SET_ADDR_LCD
                MOV         DPTR,#TITLE_1
                ACALL       DELAY_1S
                ACALL       DELAY_1S
                MOV         OUT_DATA,#11H
                MOV         LCD_ADDR,#000H
                ACALL       SET_ADDR_LCD
                MOV         DPTR,#SCR_IO_STATUS
                ACALL       WRLINE_LCD
LOOP:           MOV         A,OUT_DATA
                MOV         B,A
                RL          A
                MOV         OUT_DATA,A
                MOV         A,B
                ANL         A,#0FH
                MOV         R1,A
                ORL         A,#0F0H
                MOV         IO_DATA,A
                ACALL       PCF8574_WR
                MOV         LCD_ADDR,#40H
                ACALL       SET_ADDR_LCD
                ACALL       PCF8574_RD
                MOV         A,IO_DATA
                ANL         A,#0F0H
                ADD         A,R1
                MOV         LCD_DATA,A
                ACALL       IO2LCD
                ACALL       DELAY_1S
                AJMP        LOOP
;****************************************
;IO2_DATA
;****************************************
IO2LCD:         MOV         A,LCD_DATA
                MOV         R4,#8
IO2LCD_LOOP:    RLC         A
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                JC          IO_BIT_HIGH
                MOV         LCD_DATA,#'0'
                AJMP        IO2LCD_WRCHAR
IO_BIT_HIGH:    MOV         LCD_DATA,#'1'
IO2LCD_WRCHAR:  ACALL       WRCHAR_LCD
                DJNZ        R4,IO2LCD_LOOP
                RET
;LCD INITIAL*************************
INIT_LCD:       ACALL       DELAY_100MS
                CLR         LCD_RS
                MOV         P0,#38H
                ACALL       LCD_CLK
                ACALL       DELAY_10MS
                MOV         P0,#38H
                ACALL       LCD_CLK
                MOV         P0,#06H
                ACALL       LCD_CLK
                ACALL       LCD_HOME
LCD_CLR:        CLR         LCD_RS
                MOV         P0,#01H
                ACALL       LCD_CLK
                RET
LCD_HOME:       CLR         LCD_RS
                MOV         P0,#02H
                ACALL       LCD_CLK
                RET
LCD_OFF:        CLR         LCD_RS
                MOV         P0,#08H
                ACALL       LCD_CLK
                RET
LCD_CLK:        SETB        LCD_EN
                ACALL       LCD_DELAY
                CLR         LCD_EN
                ACALL       LCD_DELAY
                RET
LCD_ON:         CLR         LCD_RS
                MOV         P0,#00CH
                ACALL       LCD_CLK
                RET
SET_ADDR_LCD:   CLR         LCD_RS
                MOV         A,LCD_ADDR
                SETB        ACC.7
                MOV         P0,A
                ACALL       LCD_CLK
                RET
WRCHAR_LCD:     SETB        LCD_RS
                MOV         P0,LCD_DATA
                ACALL       LCD_CLK
                ACALL       LCD_ON
                RET
WRLINE_LCD:     MOV         R0,0
WRLINE_LCD_1:   SETB        LCD_RS
                CLR         A
                MOVC        A,@A+DPTR
                MOV         P0,A
                ACALL       LCD_CLK
                INC         DPTR
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                INC         R0
                CJNE        R0,#8,WRLINE_LCD_1
                MOV         LCD_ADDR,#040H
                ACALL       SET_ADDR_LCD
WRLINE_LCD_2:   SETB        LCD_RS
                CLR         A
                MOVC        A,@A+DPTR
                MOV         P0,A
                INC         DPTR
                INC         R0
                CJNE        R0,#16,WRLINE_LCD_2
                ACALL       LCD_ON
                RET
PCF8574_RD:     MOV         I2C_ADDR,#PCF8574_ID+1
                ACALL       I2C_SLAVE
                MOV         I2C_DATA,RD
                MOV         IO_DATA,I2C_DATA
                ACALL       I2C_NACK_BIT
                ACALL       I2C_STOP
                RET
PCF8574_WR:     MOV         I2C_ADDR,#PCF8574_ID
                ACALL       I2C_SLAVE
                MOV         I2C_DATA,IO_DATA
                ACALL       I2C_DATA_WR
                ACALL       I2C_STOP
                RET
I2C_DATA_WR:    PUSH        ACC
                SETB        I2C_ACK
                MOV         A,I2C_DATA
                MOV         R5,#008
I2C_DATA_WR_1:  RLC         A
                MOV         SDA,C
                ACALL       I2C_CLK
                DJNZ        R5,I2C_DATA_WR_1
                SETB        SDA
                ACALL       I2C_DELAY
                SETB        SCL
                ACALL       I2C_DELAY
                JB          SDA,I2C_DATA_WR_2
                CLR         I2C_ACK
I2C_DATA_WR_2:  CLR         SCL
                POP         ACC
                RET
I2C_DATA_RD:    PUSH        ACC
                CLR         A
                MOV         R5,008
I2C_DATA_RD_1:  ACALL       I2C_DELAY
                SETB        SCL
                ACALL       I2C_DELAY
                MOV         C,SDA
                RLC         A
                DJNZ        R5,I2C_DATA_RD_1
                MOV         I2C_DATA,A
                POP         ACC
                RET
I2C_SLAVE:      PUSH        ACC
                SETB        I2C_ACK
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                MOV         A,I2C_ADDR
                ACALL       I2C_START
                MOV         R5,#008
I2C_SLAVE_1:    RLC         A
                MOV         SDA,C
                ACALL       I2C_CLK
                DJNZ        R5,I2C_SLAVE_1
                SETB        SDA
                ACALL       I2C_DELAY
                SETB        SCL
                ACALL       I2C_DELAY
                JB          SDA,I2C_SLAVE_2
                CLR         I2C_ACK
I2C_SLAVE_2:    CLR         SCL
                POP         ACC
                RET
I2C_START:      SETB        SCL
                SETB        SDA
                ACALL       I2C_DELAY
                CLR         SDA
                ACALL       I2C_DELAY
                RET
I2C_STOP:       CLR         SDA
                ACALL       I2C_DELAY
                SETB        SCL
                ACALL       I2C_DELAY
                SETB        SDA
                RET
I2C_CLK:        ACALL       I2C_DELAY
                SETB        SCL
                ACALL       I2C_DELAY
                CLR         SCL
                RET
I2C_NACK_BIT:   SETB        SDA
                ACALL       I2C_DELAY
                ACALL       I2C_CLK
                RET
I2C_DELAY:      MOV         6,#00CH
I2C_DELAY_1:    NOP
                NOP
                DJNZ        R6,I2C_DELAY_1
                RET
LCD_DELAY:      MOV         7,#002
LCD_DELAY_1:    MOV         6,#0E6H
LCD_DELAY_2:    NOP
                NOP
                DJNZ        R6,LCD_DELAY_2
                DJNZ        R7,LCD_DELAY_1
                RET
DELAY_10MS:     MOV         7,#010
DELAY_10MS_1:   MOV         6,#0E6H
DELAY_10MS_2:   NOP
                NOP
                DJNZ        R6,DELAY_10MS_2
                DJNZ        R7,DELAY_10MS_1
                RET
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
DELAY_100MS:      MOV           7,#100
DELAY_100MS_1:    MOV           6,#0E6H
DELAY_100MS_2:    NOP
                  NOP
                  DJNZ          R6,DELAY_100MS_2
                  DJNZ          R7,DELAY_100MS_1
                  RET
DELAY_1S:         MOV           5,#100
DELAY_1S_1:       ACALL         DELAY_10MS
                  DJNZ          R5,DELAY_1S_1
                  RET
TITLE_1:          DB            'I2C PCF8574  I/O'
SCR_IO_STATUS:    DB            'STATUS:         '
                  END
```

รูปที่ 12.3 โปรแกรมการขยายพอร์ตอินพุตเอาต์พุตด้วยอุปกรณ์ระบบบัส I^2C

6) ทำการบันทึกและแอสเซมเบลอร์ไฟล์จาก Lab12.Asm เป็น Lab12.Hex

7) ทำการโปรแกรมลงบนไมโครคอนโทรลเลอร์

8) กดสวิตช์รีเซตแล้วสังเกตผลการทดลอง

____

____

____

____

สรุปผลการทดลอง

____

____

____

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ใบงานที่ 13

วงจรแปลงสัญญาณแอนะลอกเป็นสัญญาณดิจิตอล

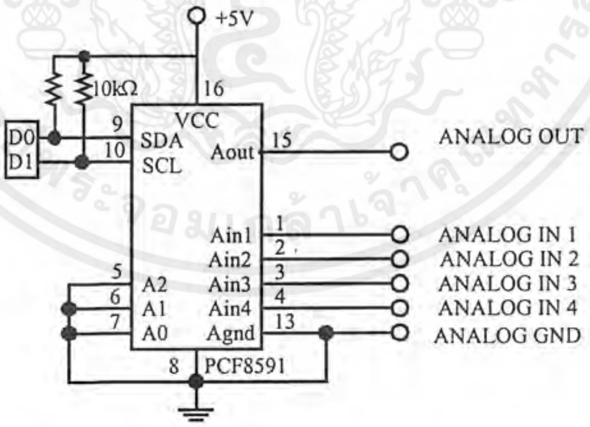
วัตถุประสงค์

1. สามารถเขียนโปรแกรมเพื่อแปลงสัญญาณแอนะลอกเป็นสัญญาณดิจิตอลได้
2. สามารถเขียนโปรแกรมเพื่อประยุกต์ใช้กับบอร์ดอื่นๆ ได้

ทฤษฎีเบื้องต้น

คุณสมบัติของวงจรแปลงสัญญาณแอนะลอกเป็นสัญญาณดิจิตอลจะมีลักษณะเหมือนกับวงจรแปลงสัญญาณดิจิตอลเป็นสัญญาณแอนะลอกจะมีคุณสมบัติพิเศษอีก 1 อย่าง คือ การเปลี่ยนแปลงเวลา (Conversion Time) หมายถึงช่วงเวลาที่วงจรแปลงสัญญาณแอนะลอกเป็นสัญญาณดิจิตอล ใช้ในการแปลงสัญญาณแอนะลอกเป็นสัญญาณดิจิตอลได้ 1 ค่า

การทดลองวงจรแปลงสัญญาณแอนะลอกเป็นสัญญาณดิจิตอลนี้ จะใช้การส่งข้อมูลผ่านไอซีเบอร์ PCF8591



รูปที่ 13.1 วงจรแปลงสัญญาณแอนะลอกเป็นสัญญาณดิจิตอล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การอ่านค่าข้อมูลอินพุตแอนะลอกของ PCF 8591

1) เตรียมข้อมูลกำหนดแอดเดรสโดยต่อไว้ที่ 000

2) เรียกโปรแกรมย่อยติดต่อกับอุปกรณ์สเลฟ

3) ส่งข้อมูลควบคุมไปยัง PCF 8591

4) ส่งสัญญาณ STOP

5) เรียกโปรแกรมย่อยติดต่อกับอุปกรณ์สเลฟ

6) ส่งข้อมูลกำหนดแอดเดรสอีกครั้งเพื่อเริ่มต้นอ่านข้อมูลโดยส่ง “1”ให้แก่ขา R/W

7) อ่านค่าจากขาอินพุตของวงจรแปลงสัญญาณแอนะลอกเป็นดิจิตอลช่องที่1

8) หากต้องการอ่านค่าในช่องต่อไป ก็ให้เริ่มการติดต่อใหม่ ดังนั้นในการเขียนโปรแกรมเพื่ออ่านค่า ต่อเนื่องทั้ง 4 ช่องหรือมากกว่าจึงต้องเขียนโปรแกรมลูปเพื่อกำหนดรอบการทำงาน 4 รอบ หรือมากกว่า ก็สามารถอ่านค่าได้ครบทุกช่อง

การเขียนข้อมูลไปยังวงจรแปลงสัญญาณดิจิตอลเป็นสัญญาณแอนะลอกของ PCF 8591

การเขียนข้อมูลไปยังขาแอนะลอกเอาต์พุตมีข้อแตกต่างจากการอ่านดังนี้

1) เรียกโปรแกรมย่อยติดต่อกับอุปกรณ์สเลฟ

2) ส่งข้อมูลกำหนดตำแหน่งโดยให้ทำงานในโหมดเขียนข้อมูล

3) ส่งข้อมูลควบคุม 40H ไปยัง PCF 8591 เพื่ออีนาเบิลแอนะลอกเอาต์พุต

4) ส่งข้อมูลไปยังเอาต์พุตแอนะลอก โดยค่าที่ส่งออกไปจะต้องมีค่าอยู่ระหว่าง 0-255

5) ส่งสภาวะหยุด

การเชื่อมต่อ PCF 8591กับไมโครคอนโทรลเลอร์ต้องต่อ SDA และ สาย SCL เข้ากับพอร์ตของไมโครคอนโทรลเลอร์

เครื่องมือและอุปกรณ์

1) บอร์ดไมโครคอนโทรลเลอร์ MCS51-S8252 MC-01

2) บอร์ดแปลงสัญญาณแอนะล็อกเป็นดิจิตอล MC-13

3) บอร์ดแสดงผลแบบผลึกเหลว MC-10

4) คอมพิวเตอร์ที่ติดตั้งวินโดว์ 95 หรือ 98 พร้อมโปรแกรม wins8252

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ลำดับขั้นการทดลอง

1) บอร์ดไมโครคอนโทรลเลอร์ MCS51-S8252 MC-01 กับบอร์ดแปลงสัญญาณแอนะลอกเป็นดิจิตอล MC-13 โดยต่อ SDA ที่พอร์ต P1.0 และต่อSCL ที่พอร์ต P1.1

2) ทำการต่อบอร์ดแสดงผลแบบผลึกเหลว MC-10 เข้ากับพอร์ต P2 และต่อขา E ที่พอร์ต P0.7 และต่อขา RS ที่พอร์ต P0.6

3) เปิดโปรแกรม wins8252 ทำการตั้งชื่อและบันทึกโดยใช้ไฟล์ชื่อ Lab13.Asm

4) เขียนโปรแกรมดังต่อไปนี้

```
;***************************
;PROGRAM     PCF8591A/D
;***************************
                    SDA           BIT    P1.0
                    SCL           BIT    P1.1
                    LCD_EN        BIT    P0.7
                    LCD_RS        BIT    P0.6
                    FLAG          EQU    02FH
                    I2C_ACK       BIT    FLAG.0
                    LCD_ADDR      EQU    030H
                    LCD_DATA      EQU    031H
                    LCD_PTR       EQU    032H
                    I2C_ADDR      EQU    033H
                    I2C_DATA      EQU    034H
                    CONTROL       EQU    035H
                    DA_DATA       EQU    036H
                    CHANNEL       EQU    037H
                    AD_DATA       EQU    038H
                    BUFFER        EQU    039H
                    PCF8591_ID    EQU    090H

;************************************
;MAIN PROGRAM
;************************************
                    ORG           0000H
                    MOV           P0,#0FF
                    MOV           P1,#0EFH
                    MOV           P2,#000H
                    MOV           P3,#01FH
MAIN:               ACALL         INIT_LCD
                    MOV           LCD_ADDR,#000H
                    ACALL         SET_ADDR_LCD
                    MOV           DPTR,#TITLE_1
                    ACALL         DELAY_1S
                    ACALL         DELAY_1S
                    MOV           DA_DATA,#0
MAIN_LOOP:          MOV           CHANNEL,#0
                    MOV           R1,#BUFFER
                    MOV           R4,#4
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
CON_LOOP:       MOV     A,CHANNEL
                ADD     A,#40H
                MOV     CONTROL,A
                ACALL   PCF8591_WR
                ACALL   PCF8591_RD
                ACALL   PCF8591_RD
                MOV     @R1,AD_DATA
                INC     R1
                INC     CHANNEL
                DJNZ    R4,CON_LOOP

                MOV     LCD_ADDR,#000H
                ACALL   SET_ADDR_LCD
                MOV     DPTR,#SCR_DA
                ACALL   WRLINE_LCD
                MOV     LCD_ADDR,#045H
                ACALL   SET_ADDR_LCD
                MOV     LCD_DATA,DA_DATA
                ACALL   HEX2LCD
                INC     DA_DATA
                ACALL   DELAY_1S
                MOV     R1,#BUFFER
                MOV     LCD_ADDR,#000H
                ACALL   SET_ADDR_LCD
                MOV     DPTR,#SCR_AD01
                ACALL   WRLINE_LCD
                MOV     LCD_ADDR,#004H
                ACALL   SET_ADDR_LCD
                MOV     LCD_DATA,@R1
                ACALL   HEX2LCD
                INC     R1
                MOV     LCD_ADDR,#044H
                ACALL   SET_ADDR_LCD
                MOV     LCD_DATA,@R1
                ACALL   HEX2LCD
                ACALL   DELAY_1S
                INC     R1
                MOV     LCD_ADDR,#000H
                ACALL   SET_ADDR_LCD
                MOV     DPTR,#SCR_AD23
                ACALL   WRLINE_LCD
                MOV     LCD_ADDR,#004H
                ACALL   SET_ADDR_LCD
                MOV     LCD_DATA,@R1
                ACALL   HEX2LCD
                INC     R1
                MOV     LCD_ADDR,#044H
                ACALL   SET_ADDR_LCD
                MOV     LCD_DATA,@R1
                ACALL   HEX2LCD
                ACALL   DELAY_1S
                AJMP    MAIN_LOOP
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
;****************************************
;HEX  CODE
;****************************************


HEX2LCD:            PUSH        ACC
                    MOV         A,LCD_DATA
                    MOV         B,#16
                    DIV         AB
                    ADD         A,#030H
                    MOV         LCD_DATA,A
                    ACALL       HEX_CHK
                    ACALL       WRCHAR_LCD
                    MOV         A,B
                    ADD         A,#030H
                    MOV         LCD_DATA,A
                    ACALL       HEX_CHK
                    ACALL       WRCHAR_LCD
                    POP         ACC
                    RET
HEX_CHK:            MOV         A,LCD_DATA
                    CJNE        A,#030H,CHK_OTHER
CHK_OTHER:          JNC         CONV_2_ALPHA
                    RET
CONV_2_ALPHA:       ADD         A,#7
                    MOV         LCD_DATA,A
                    RET
INIT_LCD:           ACALL       DELAY_100MS
                    CLR         LCD_RS
                    MOV         P2,#38H
                    ACALL       LCD_CLK
                    ACALL       LCD_OFF
                    ACALL       LCD_CLR
                    MOV         P2,#06H
                    ACALL       LCD_CLK
                    ACALL       LCD_HOME
LCD_CLR:            CLR         LCD_RS
                    MOV         P2,#01H
                    ACALL       LCD_CLK
                    RET
LCD_HOME:           CLR         LCD_RS
                    MOV         P2,#02H
                    ACALL       LCD_CLK
                    RET
LCD_OFF:            CLR         LCD_RS
                    MOV         P2,#08H
                    ACALL       LCD_CLK
                    RET
LCD_CLK:            SETB        LCD_EN
                    ACALL       LCD_DELAY
                    CLR         LCD_EN
                    ACALL       LCD_DELAY
                    RET
LCD_ON:             CLR         LCD_RS
                    MOV         P2,#00CH
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                ACALL       LCD_CLK
                RET
SET_ADDR_LCD:   CLR         LCD_RS
                MOV         A,LCD_ADDR
                MOV         P2,A
                ACALL       LCD_CLK
                RET
WRCHAR_LCD:     SETB              LCD_RS
                MOV         P2,LCD_DATA
                ACALL       LCD_CLK
                ACALL       LCD_ON
                RET
WRLINE_LCD:     MOV         R0,0
WRLINE_LCD_1:   SETB        LCD_RS
                CLR         A
                MOVC        A,@A+DPTR
                MOV         P2,A
                ACALL       LCD_CLK
                INC         DPTR
                INC         R0
                CJNE        R0,#8,WRLINE_LCD_1
                MOV         LCD_ADDR,#040H
                ACALL       SET_ADDR_LCD
WRLINE_LCD_2:   SETB        LCD_RS
                CLR         A
                MOVC        A,@A+DPTR
                MOV         P2,A
                ACALL       LCD_CLK
                INC         DPTR
                INC         R0
                CJNE        R0,#16,WRLINE_LCD_2
                ACALL       LCD_ON
                RET
PCF8591_RD:     MOV         I2C_ADDR,#PCF8591_ID+1
                ACALL       I2C_SLAVE
                MOV         I2C_DATA,RD
                MOV         AD_DATA,I2C_DATA
                ACALL       I2C_NACK_BIT
                ACALL       I2C_STOP
                RET
PCF8591_WR:     MOV         I2C_ADDR,#PCF8591_ID
                ACALL       I2C_SLAVE
                MOV         I2C_DATA,CONTROL
                ACALL       I2C_DATA_WR
                MOV         I2C_DATA,DA_DATA
                ACALL       I2C_DATA_WR
                ACALL       I2C_STOP
                RET
I2C_DATA_WR:    PUSH        ACC
                SETB        2C_ACK
                MOV         A,I2C_DATA
                MOV         R5,#008
I2C_DATA_WR_1:  RLC         A
                MOV         SDA,C
                ACALL       I2C_CLK
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                DJNZ        R5,I2C_DATA_WR_1
                SETB        SDA
                ACALL       I2C_DELAY
                SETB        SCL
                ACALL       I2C_DELAY
                JB          SDA,I2C_DATA_WR_2
                CLR         I2C_ACK
I2C_DATA_WR_2:  CLR         SCL
                POP         ACC
                RET
I2C_DATA_RD:    PUSH        ACC
                CLR         A
                MOV         R5,008
I2C_DATA_RD_1:  ACALL       I2C_DELAY
                SETB        SCL
                ACALL       I2C_DELAY
                MOV         C,SDA
                RLC         A
                DJNZ        R5,I2C_DATA_RD_1
                MOV         I2C_DATA,A
                POP         ACC
                RET
I2C_SLAVE:      PUSH        ACC
                SETB        I2C_ACK
                MOV         A,I2C_ADDR
                ACALL       I2C_START
                MOV         R5,#008
I2C_SLAVE_1:    RLC         A
                MOV         SDA,C
                ACALL       I2C_CLK
                DJNZ        R5,I2C_SLAVE_1
                SETB        SDA
                ACALL       I2C_DELAY
                SETB        SCL
                ACALL       I2C_DELAY
                JB          SDA,I2C_SLAVE_2
                CLR         I2C_ACK
I2C_SLAVE_2:    CLR         SCL
                POP         ACC
                RET
I2C_START:      SETB        SCL
                SETB        SDA
                ACALL       I2C_DELAY
                CLR         SDA
                ACALL       I2C_DELAY
                CLR         SCL
                RET
I2C_STOP:       CLR         SDA
                ACALL       I2C_DELAY
                SETB        SCL
                ACALL       I2C_DELAY
                SETB        SDA
                RET
I2C_CLK:        ACALL       I2C_DELAY
                SETB        SCL
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                CLR         SCL
                RET
I2C_STOP:       CLR         SDA
                ACALL       I2C_DELAY
                SETB        SCL
                ACALL       I2C_DELAY
                SETB        SDA
                RET
I2C_CLK:        ACALL       I2C_DELAY
                SETB        SCL
                ACALL       I2C_DELAY
                RET
I2C_NACK_BIT:   SETB        SDA
                ACALL       I2C_DELAY
                ACALL       I2C_CLK
                RET
I2C_DELAY:      MOV         6,#00CH
I2C_DELAY_1:    NOP
                NOP
                DJNZ        R6,I2C_DELAY_1
                RET
LCD_DELAY:      MOV         7,#002
LCD_DELAY_1:    MOV         6,#0E6H
LCD_DELAY_2:    NOP
                NOP
                DJNZ        R6,LCD_DELAY_2
                DJNZ        R7,LCD_DELAY_1
                RET
DELAY_10MS:     MOV         7,#010
DELAY_10MS_1:   MOV         6,#0E6H
DELAY_10MS_2:   NOP
                NOP
                DJNZ        R6,DELAY_10MS_2
                DJNZ        R7,DELAY_10MS_1
                RET
DELAY_100MS:    MOV         7,#100
DELAY_100MS_1:  MOV         6,#0E6H
DELAY_100MS_2:  NOP
                NOP
                DJNZ        R6,DELAY_100MS_2
                DJNZ        R7,DELAY_100MS_1
                RET
DELAY_1S:       MOV         5,#100
DELAY_1S_1:     ACALL       DELAY_10MS
                DJNZ        R5,DELAY_1S_1
                RET
TITLE_1:        DB     'I2C 8591 A/D D/A'
SCR_AD01:       DB     'Ch0:  H  Ch1:  H'
SCR_AD23:       DB     'Ch2:  H  Ch3:  H'
SCR_DA:         DB     'Analog Out   :  H'
                END
```

รูปที่ 13.3 โปรแกรมแปลงสัญญาณแอนะลอกเป็นสัญญาณดิจิตอล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4) ทำการบันทึกและแอสเซมเบลอร์ไฟล์จาก Lab13.Asm เป็น Lab13.Hex

5) ทำการโปรแกรมลงบนไมโครคอนโทรลเลอร์

6) กดสวิตช์รีเซตแล้วสังเกตผลการทดลอง

สรุปผลการทดลอง

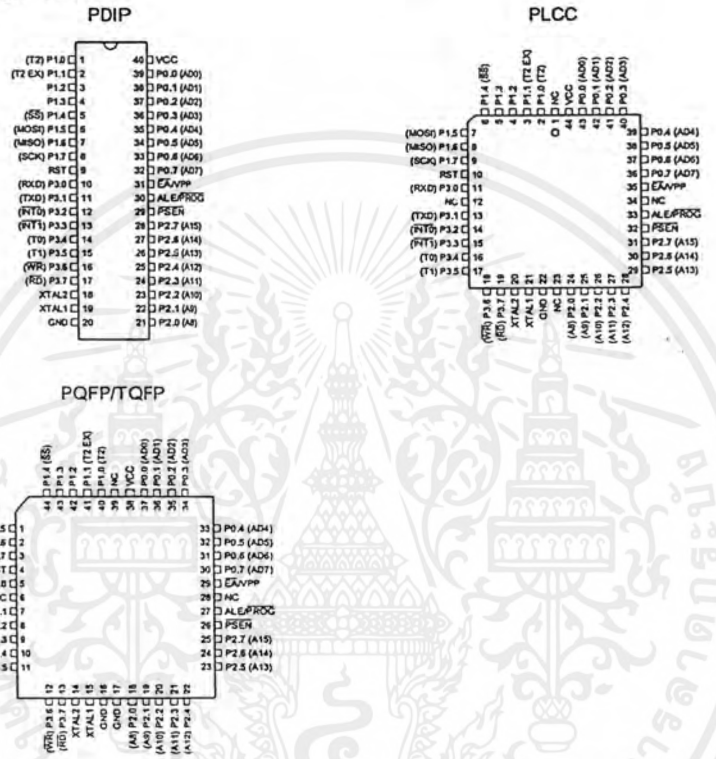
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก ง

รายละเอียดข้อมูล และคุณสมบัติของอุปกรณ์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Pin Configurations



Pin Description

V_{CC}
Supply voltage.

GND
Ground.

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

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

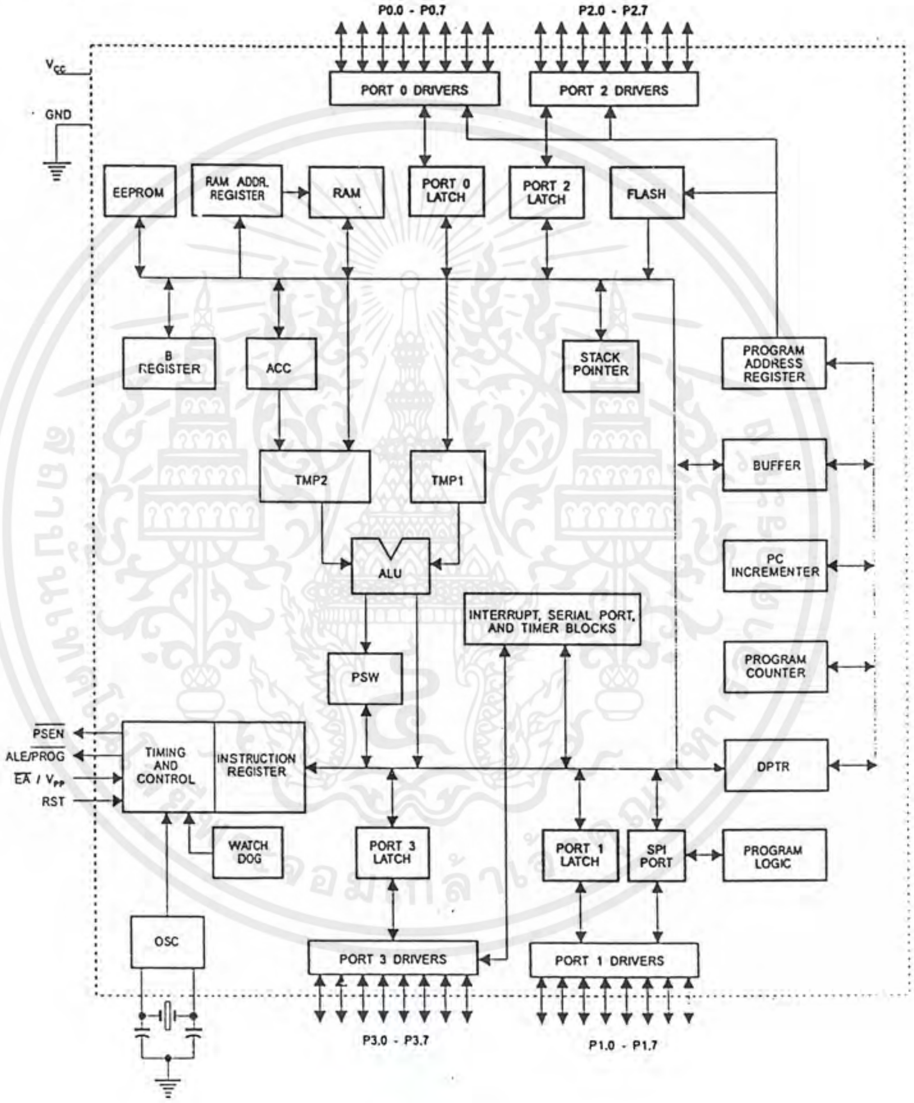
Port 0 also receives the code bytes during Flash programming and outputs the code bytes during program verification. External pullups are required during program verification.

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

Some Port 1 pins provide additional functions. P1.0 and P1.1 can be configured to be the timer/counter 2 external count input (P1.0/T2) and the timer/counter 2 trigger input (P1.1/T2EX), respectively.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Block Diagram



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Furthermore, P1.4, P1.5, P1.6, and P1.7 can be configured as the SPI slave port select, data input/output and shift clock input/output pins as shown in the following table.

Port Pin	Alternate Functions
P1.0	T2 (external count input to Timer/Counter 2), clock-out
P1.1	T2EX (Timer/Counter 2 capture/reload trigger and direction control)
P1.4	$\overline{SS}$ (Slave port select input)
P1.5	MOSI (Master data output, slave data input pin for SPI channel)
P1.6	MISO (Master data input, slave data output pin for SPI channel)
P1.7	SCK (Master clock output, slave clock input pin for SPI channel)

Port 1 also receives the low-order address bytes during Flash programming and 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, Port 2 uses strong internal pullups when emitting 1s. During accesses to external data memory that use 8-bit addresses (MOVX @ RI), Port 2 emits the contents of the P2 Special Function Register.

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

Port 3

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

Port 3 also serves the functions of various special features of the AT89S8252, as shown in the following table.

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

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

RST

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

ALE/$\overline{PROG}$

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

$\overline{PSEN}$

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

When the AT89S8252 is executing code from external program memory, $\overline{PSEN}$ is activated twice each machine cycle, except that two $\overline{PSEN}$ activations are skipped during each access to external data memory.

$\overline{EA}$/V_{PP}

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

$\overline{EA}$ should be strapped to V_{CC} for internal program executions. This pin also receives the 12-volt programming enable voltage (V_{PP}) during Flash programming when 12-volt programming is selected.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Serial Programming Algorithm

To program and verify the AT89S8252 in the serial programming mode, the following sequence is recommended:

1. Power-up sequence:

   Apply power between V_{CC} and GND pins.

   Set RST pin to "H".

   If a crystal is not connected across pins XTAL1 and XTAL2, apply a 3 MHz to 24 MHz clock to XTAL1 pin and wait for at least 10 milliseconds.
2. Enable serial programming by sending the Programming Enable serial instruction to pin MOSI/P1.5. The frequency of the shift clock supplied at pin SCK/P1.7 needs to be less than the CPU clock at XTAL1 divided by 40.
3. The Code or Data array is programmed one byte at a time by supplying the address and data together with the appropriate Write instruction. The selected memory location is first automatically erased before new data is written. The write cycle is self-timed and typically takes less than 2.5 ms at 5V.
4. Any memory location can be verified by using the Read instruction which returns the content at the selected address at serial output MISO/P1.6.
5. At the end of a programming session, RST can be set low to commence normal operation.

Power-off sequence (if needed):

Set XTAL1 to "L" (if a crystal is not used).

Set RST to "L".

Turn V_{CC} power off.

Serial Programming Instruction

The Instruction Set for Serial Programming follows a 3-byte protocol and is shown in the following table:

Instruction Set

Instruction	Input Format			Operation
	Byte 1	Byte 2	Byte 3	
Programming Enable	1010 1100	0101 0011	xxxx xxxx	Enable serial programming interface after RST goes high.
Chip Erase	1010 1100	xxxx x100	xxxx xxxx	Chip erase both 8K & 2K memory arrays.
Read Code Memory	aaaa a001	low addr	xxxx xxxx	Read data from Code memory array at the selected address. The 5 MSBs of the first byte are the high order address bits. The low order address bits are in the second byte. Data are available at pin MISO during the third byte.
Write Code Memory	aaaa a010	low addr	data in	Write data to Code memory location at selected address. The address bits are the 5 MSBs of the first byte together with the second byte.
Read Data Memory	00aa a101	low addr	xxxx xxxx	Read data from Data memory array at selected address. Data are available at pin MISO during the third byte.
Write Data Memory	00aa a110	low addr	data in	Write data to Data memory location at selected address.
Write Lock Bits	1010 1100	LB1 LB2 LB3 x x111	xxxx xxxx	Write lock bits. Set LB1, LB2 or LB3 = "0" to program lock bits.

Notes: 1. $\overline{\text{DATA}}$ polling is used to indicate the end of a write cycle which typically takes less than 2.5 ms at 5V.

2. "aaaaa" = high order address.

3. "x" = don't care.

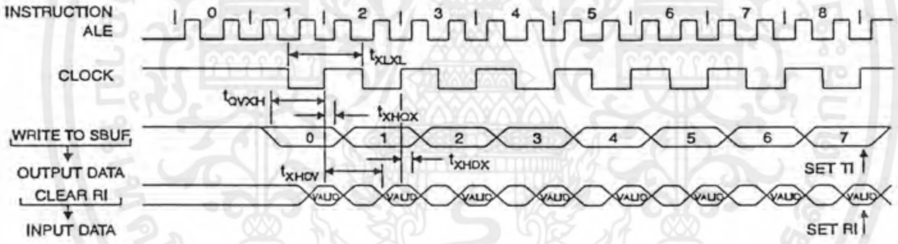
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Serial Port Timing: Shift Register Mode Test Conditions

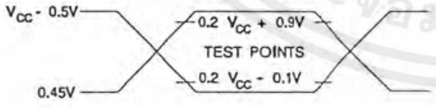
The values in this table are valid for V_{CC} = 4.0V to 6V and Load Capacitance = 80 pF.

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

Shift Register Mode Timing Waveforms

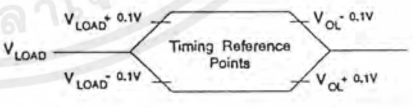


AC Testing Input/Output Waveforms(1)



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

Float Waveforms(1)



Notes: 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 อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Flash and EEPROM Parallel Programming Modes

Mode	RST	$\overline{PSEN}$	ALE/$\overline{PROG}$	$\overline{EA}/V_{PP}$	P2.6	P2.7	P3.6	P3.7	Data I/O P0.7:0	Address P2.5:0 P1.7:0
Serial Prog. Modes	H	h(1)	h(1)	x						
Chip Erase	H	L	‾_/‾ (2)	12V	H	L	L	L	X	X
Write (10K bytes) Memory	H	L	‾_/‾	12V	L	H	H	H	DIN	ADDR
Read (10K bytes) Memory	H	L	H	12V	L	L	H	H	DOUT	ADDR
Write Lock Bits:	H	L	‾_/‾	12V	H	L	H	L	DIN	X
Bit - 1									P0.7 = 0	X
Bit - 2									P0.6 = 0	X
Bit - 3									P0.5 = 0	X
Read Lock Bits:	H	L	H	12V	H	H	L	L	DOUT	X
Bit - 1									@P0.2	X
Bit - 2									@P0.1	X
Bit - 3									@P0.0	X
Read Atmel Code	H	L	H	12V	L	L	L	L	DOUT	30H
Read Device Code	H	L	H	12V	L	L	L	L	DOUT	31H
Serial Prog. Enable	H	L	‾_/‾ (2)	12V	L	H	L	H	P0.0 = 0	X
Serial Prog. Disable	H	L	‾_/‾ (2)	12V	L	H	L	H	P0.0 = 1	X
Read Serial Prog. Fuse	H	L	H	12V	H	H	L	H	@P0.0	X

Notes:
1. "h" = weakly pulled "High" internally.
2. Chip Erase and Serial Programming Fuse require a 10-ms $\overline{PROG}$ pulse. Chip Erase needs to be performed first before reprogramming any byte with a content other than FFH.
3. P3.4 is pulled Low during programming to indicate RDY/$\overline{BSY}$.
4. "X" = don't care

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Flash Programming and Verification Characteristics-Parallel Mode

T_A = 0°C to 70°C, V_{CC} = 5.0V ± 10%

Symbol	Parameter	Min	Max	Units
V_{PP}	Programming Enable Voltage	11.5	12.5	V
I_{PP}	Programming Enable Current		1.0	mA
$1/t_{CLCL}$	Oscillator Frequency	3	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}	$48t_{CLCL}$		
t_{SHGL}	V_{PP} Setup to $\overline{PROG}$ Low	10		µs
t_{GLGH}	$\overline{PROG}$ Width	1	110	µs
t_{AVQV}	Address to Data Valid		$48t_{CLCL}$	
t_{ELQV}	$\overline{ENABLE}$ Low to Data Valid		$48t_{CLCL}$	
t_{EHQZ}	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

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Absolute Maximum Ratings*

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

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

DC Characteristics

The values shown in this table are valid for T_A = -40°C to 85°C and V_{CC} = 5.0V ± 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		0.5	V
V_{OL1}	Output Low Voltage (1) (Port 0, ALE, $\overline{PSEN}$)	I_{OL} = 3.2 mA		0.5	V
V_{OH}	Output High Voltage (Ports 1,2,3, ALE, $\overline{PSEN}$)	I_{OH} = -60 μA, V_{CC} = 5V ± 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} = 5V ± 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} = 0.45V		-50	μA
I_{TL}	Logical 1 to 0 Transition Current (Ports 1,2,3)	V_{IN} = 2V, V_{CC} = 5V ± 10%		-650	μA
I_{LI}	Input Leakage Current (Port 0, $\overline{EA}$)	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		25	mA
		Idle Mode, 12 MHz		6.5	mA
	Power Down Mode (2)	V_{CC} = 6V		100	μA
		V_{CC} = 3V		40	μA

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

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

AC Characteristics

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

External Program and Data Memory Characteristics

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

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

PIN DESCRIPTION

PIN NO.	SYMBOL	NAME AND FUNCTION
2, 3, 4, 5, 6, 7, 8, 9	D_0 to D_7	data inputs
11	LE	latch enable input (active HIGH)
1	$\overline{OE}$	3-state output enable input (active LOW)
10	GND	ground (0 V)
19, 18, 17, 16, 15, 14, 13, 12	Q_0 to Q_7	3-state latch outputs
20	V_{CC}	positive supply voltage

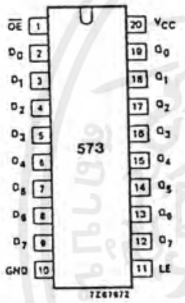


Fig.1 Pin configuration.

Fig.2 Logic symbol.

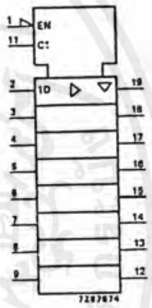


Fig.3 IEC logic symbol.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

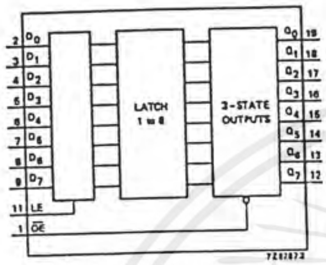


Fig.4 Functional diagram.

FUNCTION TABLE

OPERATING MODES	INPUTS			INTERNAL LATCHES	OUTPUTS
	$\overline{OE}$	LE	D_N		Q_0 to Q_7
enable and read register (transparent mode)	L L	H H	L H	L H	L H
latch and read register	L L	L L	l h	L H	L H
latch register and disable outputs	H H	L L	l h	L H	Z Z

Notes

1. H = HIGH voltage level
   h = HIGH voltage level one set-up time prior to the HIGH-to-LOW LE transition
   L = LOW voltage level
   l = LOW voltage level one set-up time prior to the HIGH-to-LOW LE transition
   Z = high impedance OFF-state

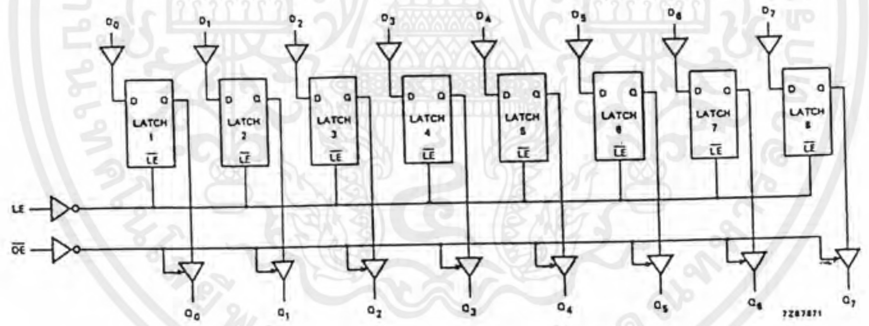


Fig.5 Logic diagram.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

DC CHARACTERISTICS FOR 74HC

For the DC characteristics see *"74HC/HCT/HCU/HCMOS Logic Family Specifications"*.

Output capability: bus driver
I_{CC} category: MSI

AC CHARACTERISTICS FOR 74HC

GND = 0 V; $t_r = t_f = 6$ ns; $C_L = 50$ pF

SYMBOL	PARAMETER	T_{amb} (°C) 74HC +25 min.	typ.	max.	−40 to +85 min.	max.	−40 to +125 min.	max.	UNIT	TEST CONDITIONS V_{CC} (V)	WAVEFORMS
t_{PHL}/t_{PLH}	propagation delay D_n to Q_n		47 17 14	150 30 26		190 38 33		225 45 38	ns	2.0 4.5 6.0	Fig.6
t_{PHL}/t_{PLH}	propagation delay LE to Q_n		50 18 14	150 30 26		190 38 33		225 45 38	ns	2.0 4.5 6.0	Fig.7
t_{PZH}/t_{PZL}	3-state output enable time $\overline{OE}$ to Q_n		44 16 13	140 28 24		175 35 30		210 42 36	ns	2.0 4.5 6.0	Fig.8
t_{PHZ}/t_{PLZ}	3-state output disable time $\overline{OE}$ to Q_n		55 20 16	150 30 26		190 38 33		225 45 38	ns	2.0 4.5 6.0	Fig.8
t_{THL}/t_{TLH}	output transition time		14 5 4	60 12 10		75 15 13		90 18 15	ns	2.0 4.5 6.0	Fig.6
t_W	enable pulse width HIGH	80 16 14	14 5 4		100 20 17		120 24 20		ns	2.0 4.5 6.0	Fig.7
t_{su}	set-up time D_n to LE	50 10 9	11 4 3		65 13 11		75 15 13		ns	2.0 4.5 6.0	Fig.9
t_h	hold time D_n to LE	5 5 5	3 1 1		5 5 5		5 5 5		ns	2.0 4.5 6.0	Fig.9

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

DC CHARACTERISTICS FOR 74HCT

For the DC characteristics see *"74HC/HCT/HCU/HCMOS Logic Family Specifications"*.

Output capability: bus driver
I_{CC} category: MSI

Note to HCT types

The value of additional quiescent supply current (ΔI_{CC}) for a unit load of 1 is given in the family specifications.
To determine ΔI_{CC} per input, multiply this value by the unit load coefficient shown in the table below.

INPUT	UNIT LOAD COEFFICIENT
D_n	0.35
LE	0.65
$\overline{OE}$	1.25

AC CHARACTERISTICS FOR 74HCT

GND = 0 V; $t_r = t_f = 6$ ns; $C_L = 50$ pF

SYMBOL	PARAMETER	T_{amb} (°C) 74HCT +25 min.	typ.	max.	−40 to +85 min.	max.	−40 to +125 min.	max.	UNIT	TEST CONDITIONS V_{CC} (V)	WAVEFORMS
t_{PHL}/t_{PLH}	propagation delay D_n to Q_n		20	35		44		53	ns	4.5	Fig.6
t_{PHL}/t_{PLH}	propagation delay LE to Q_n		18	35		44		53	ns	4.5	Fig.7
t_{PZH}/t_{PZL}	3-state output enable time $\overline{OE}$ to Q_n		17	30		38		45	ns	4.5	Fig.8
t_{PHZ}/t_{PLZ}	3-state output disable time $\overline{OE}$ to Q_n		18	30		38		45	ns	4.5	Fig.8
t_{THL}/t_{TLH}	output transition time		5	12		15		18	ns	4.5	Fig.6
t_W	enable pulse width HIGH	16	5		20		24		ns	4.5	Fig.7
t_{su}	set-up time D_n to LE	13	7		16		20		ns	4.5	Fig.9
t_h	hold time D_n to LE	9	4		11		14		ns	4.5	Fig.9

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

19-4452; Rev 3; 7/97

MAXIM

Serially Interfaced, 8-Digit LED Display Drivers

General Description

The MAX7219/MAX7221 are compact, serial input/output common-cathode display drivers that interface microprocessors (µPs) to 7-segment numeric LED displays of up to 8 digits, bar-graph displays, or 64 individual LEDs. Included on-chip are a BCD code-B decoder, multiplex scan circuitry, segment and digit drivers, and an 8x8 static RAM that stores each digit. Only one external resistor is required to set the segment current for all LEDs. The MAX7221 is compatible with SPI™, QSPI™, and Microwire™, and has slew-rate-limited segment drivers to reduce EMI.

A convenient 3-wire serial interface connects to all common µPs. Individual digits may be addressed and updated without rewriting the entire display. The MAX7219/MAX7221 also allow the user to select code-B decoding or no-decode for each digit.

The devices include a 150µA low-power shutdown mode, analog and digital brightness control, a scan-limit register that allows the user to display from 1 to 8 digits, and a test mode that forces all LEDs on.

Applications

Bar-Graph Displays
7-Segment Displays
Industrial Controllers
Panel Meters
LED Matrix Displays

Features

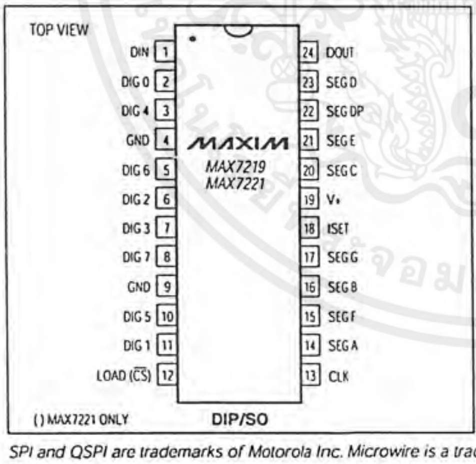
- ♦ 10MHz Serial Interface
- ♦ Individual LED Segment Control
- ♦ Decode/No-Decode Digit Selection
- ♦ 150µA Low-Power Shutdown (Data Retained)
- ♦ Digital and Analog Brightness Control
- ♦ Display Blanked on Power-Up
- ♦ Drive Common-Cathode LED Display
- ♦ Slew-Rate Limited Segment Drivers for Lower EMI (MAX7221)
- ♦ SPI, QSPI, Microwire Serial Interface (MAX7221)
- ♦ 24-Pin DIP and SO Packages

Ordering Information

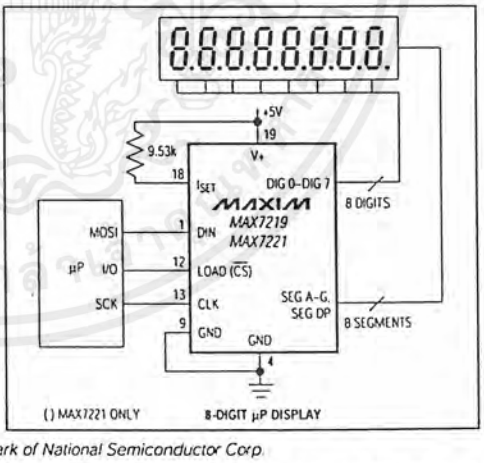
PART	TEMP. RANGE	PIN-PACKAGE
MAX7219CNG	0℃ to +70℃	24 Narrow Plastic DIP
MAX7219CWG	0℃ to +70℃	24 Wide SO
MAX7219C/D	0℃ to +70℃	Dice*
MAX7219ENG	-40℃ to +85℃	24 Narrow Plastic DIP
MAX7219EWG	-40℃ to +85℃	24 Wide SO
MAX7219ERG	-40℃ to +85℃	24 Narrow CERDIP

Ordering Information continued at end of data sheet.
**Dice are specified at T_A = +25℃.*

Pin Configuration



Typical Application Circuit



SPI and QSPI are trademarks of Motorola Inc. Microwire is a trademark of National Semiconductor Corp.

For free samples & the latest literature: http://www.maxim-ic.com, or phone 1-800-998-8800.
For small orders, phone 408-737-7600 ext. 3468.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Serially Interfaced, 8-Digit LED Display Drivers

ABSOLUTE MAXIMUM RATINGS

Voltage (with respect to GND)
- V+ ....-0.3V to 6V
- DIN, CLK, LOAD, $\overline{CS}$ ....-0.3V to 6V
- All Other Pins ....-0.3V to (V+ + 0.3V)

Current
- DIG0–DIG7 Sink Current 500mA
- SEGA–G, DP Source Current 100mA

Continuous Power Dissipation (T_A = +85 ℃)
- Narrow Plastic DIP 0.87W
- Wide SO 0.76W
- Narrow CERDIP 1.1W

Operating Temperature Ranges
- MAX7219C_G/MAX7221C_G ....0 ℃ to +70 ℃
- MAX7219E_G/MAX7221E_G ....-40 ℃ to +85 ℃

Storage Temperature Range ....-65 ℃ to +160 ℃

Lead Temperature (soldering, 10sec) ....+300 ℃

Stresses beyond those listed under "Absolute Maximum Ratings" may cause permanent damage to the device. These are stress ratings only, and functional operation of the device at these or any other conditions beyond those indicated in the operational sections of the specifications is not implied. Exposure to absolute maximum rating conditions for extended periods may affect device reliability.

ELECTRICAL CHARACTERISTICS

(V+ = 5V ±10%, R_{SET} = 9.53kΩ ±1%, T_A = T_{MIN} to T_{MAX}, unless otherwise noted.)

PARAMETER	SYMBOL	CONDITIONS	MIN	TYP	MAX	UNITS
Operating Supply Voltage	V+		4.0		5.5	V
Shutdown Supply Current	I+	All digital inputs at V+ or GND, T_A = +25 ℃			150	µA
Operating Supply Current	I+	R_{SET} = open circuit			8	mA
		All segments and decimal point on, $I_{SEG_}$ = -40mA		330		
Display Scan Rate	f_{OSC}	8 digits scanned	500	800	1300	Hz
Digit Drive Sink Current	I_{DIGIT}	V+ = 5V, V_{OUT} = 0.65V	320			mA
Segment Drive Source Current	I_{SEG}	T_A = +25 ℃, V+ = 5V, V_{OUT} = (V+ - 1V)	-30	-40	-45	mA
Segment Current Slew Rate (MAX7221 only)	$\Delta I_{SEG}/\Delta t$	T_A = +25 ℃, V+ = 5V, V_{OUT} = (V+ - 1V)	10	20	50	mA/µs
Segment Drive Current Matching	ΔI_{SEG}			3.0		%
Digit Drive Leakage (MAX7221 only)	I_{DIGIT}	Digit off, V_{DIGIT} = V+			-10	µA
Segment Drive Leakage (MAX7221 only)	I_{SEG}	Segment off, V_{SEG} = 0V			1	µA
Digit Drive Source Current (MAX7219 only)	I_{DIGIT}	Digit off, V_{DIGIT} = (V+ - 0.3V)	-2			mA
Segment Drive Sink Current (MAX7219 only)	I_{SEG}	Segment off, V_{SEG} = 0.3V	5			mA

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Serially Interfaced, 8-Digit LED Display Drivers

ELECTRICAL CHARACTERISTICS (continued)

($V+ = 5V \pm 10\%$, $R_{SET} = 9.53k\Omega \pm 1\%$, $T_A = T_{MIN}$ to T_{MAX}, unless otherwise noted.)

PARAMETER	SYMBOL	CONDITIONS	MIN	TYP	MAX	UNITS
LOGIC INPUTS						
Input Current DIN, CLK, LOAD, $\overline{CS}$	I_{IH}, I_{IL}	V_{IN} = 0V or V+	-1		1	µA
Logic High Input Voltage	V_{IH}		3.5			V
Logic Low Input Voltage	V_{IL}				0.8	V
Output High Voltage	V_{OH}	DOUT, I_{SOURCE} = -1mA	V+ - 1			V
Output Low Voltage	V_{OL}	DOUT, I_{SINK} = 1.6mA			0.4	V
Hysteresis Voltage	ΔV_I	DIN, CLK, LOAD, $\overline{CS}$		1		V
TIMING CHARACTERISTICS						
CLK Clock Period	t_{CP}		100			ns
CLK Pulse Width High	t_{CH}		50			ns
CLK Pulse Width Low	t_{CL}		50			ns
$\overline{CS}$ Fall to SCLK Rise Setup Time (MAX7221 only)	t_{CSS}		25			ns
CLK Rise to $\overline{CS}$ or LOAD Rise Hold Time	t_{CSH}		0			ns
DIN Setup Time	t_{DS}		25			ns
DIN Hold Time	t_{DH}		0			ns
Output Data Propagation Delay	t_{DO}	C_{LOAD} = 50pF			25	ns
Load-Rising Edge to Next Clock Rising Edge (MAX7219 only)	t_{LDCK}		50			ns
Minimum $\overline{CS}$ or LOAD Pulse High	t_{CSW}		50			ns
Data-to-Segment Delay	t_{DSPD}				2.25	ms

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Serially Interfaced, 8-Digit LED Display Drivers

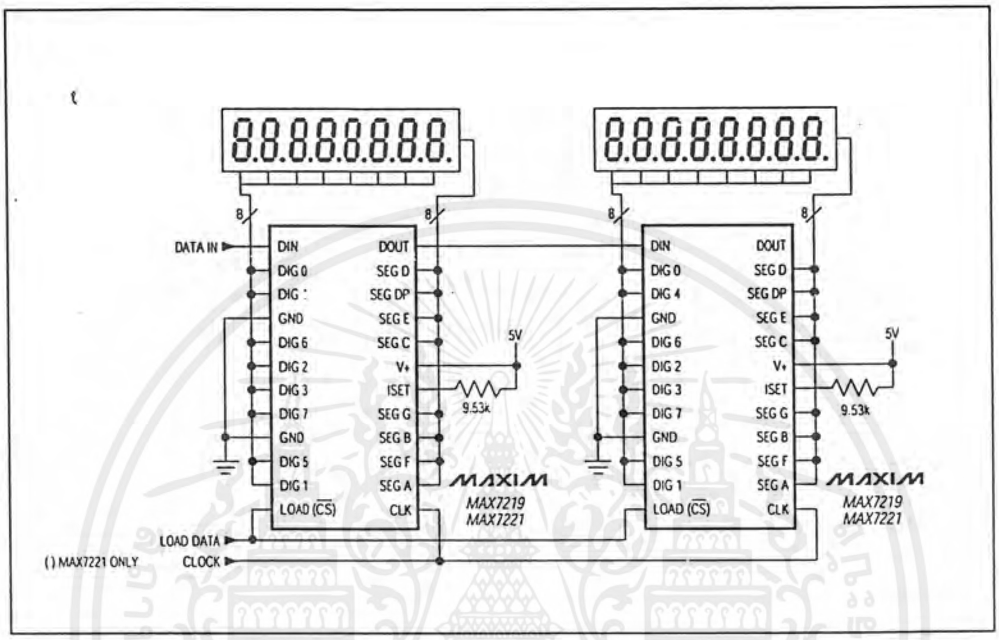


Figure 3. Cascading MAX7219/MAX7221s to Drive 16 7-Segment LED Digits

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4 BLOCK DIAGRAM

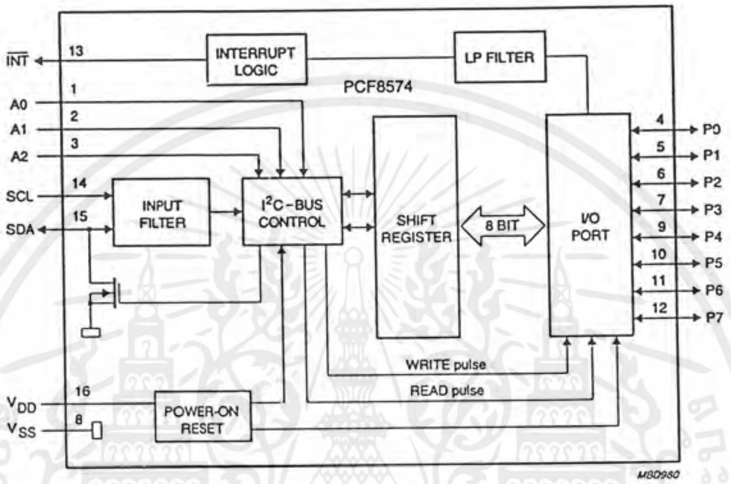


Fig.1 Block diagram (SOT38-1 and SOT162-1).

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5 PINNING

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

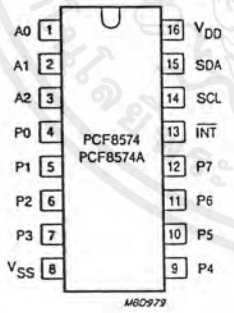


Fig.2 Pin configuration (DIP16; SO16).

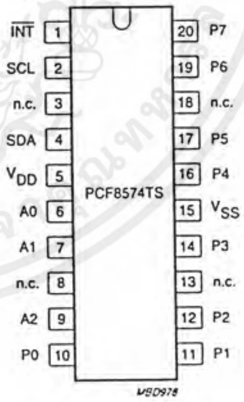


Fig.3 Pin configuration (SSOP20).

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

8 LIMITING VALUES

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

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

9 HANDLING

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

10 DC CHARACTERISTICS

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

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

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

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

Note

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

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

11 I²C-BUS TIMING CHARACTERISTICS

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

Note

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

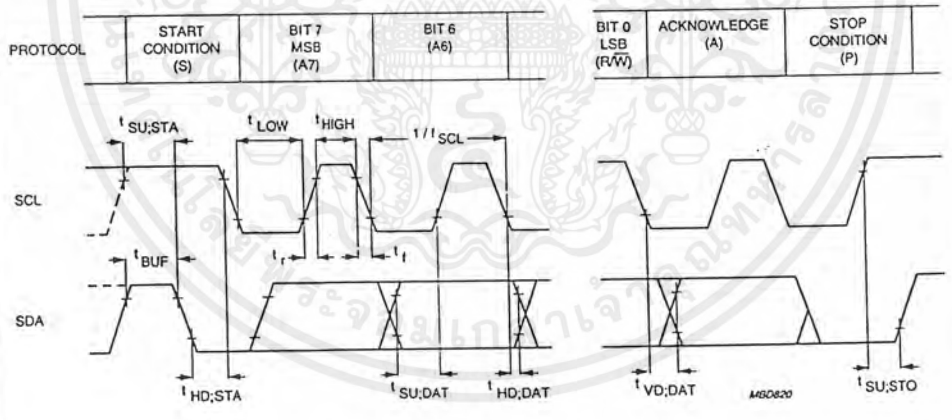


Fig.15 I²C-bus timing diagram.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5 BLOCK DIAGRAM

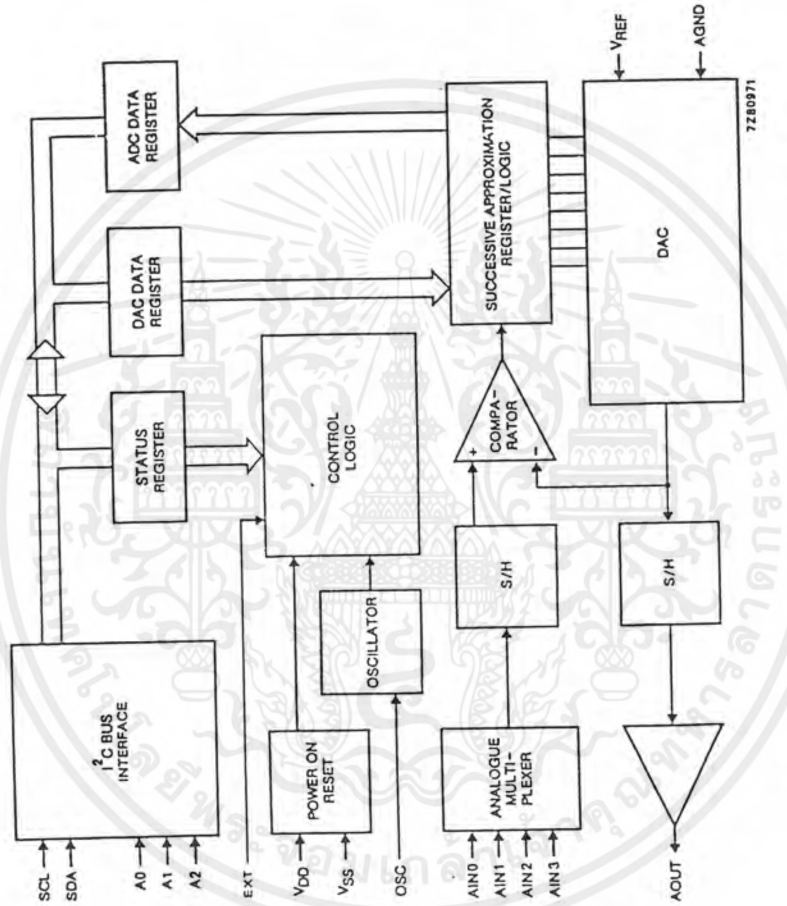


Fig.1 Block diagram.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

6 PINNING

SYMBOL	PIN	DESCRIPTION
AIN0	1	analog inputs (A/D converter)
AIN1	2	
AIN2	3	
AIN3	4	
A0	5	hardware address
A1	6	
A2	7	
V_{SS}	8	negative supply voltage
SDA	9	I^2C-bus data input/output
SCL	10	I^2C-bus clock input
OSC	11	oscillator input/output
EXT	12	external/internal switch for oscillator input
AGND	13	analog ground
V_{REF}	14	voltage reference input
AOUT	15	analog output (D/A converter)
V_{DD}	16	positive supply voltage

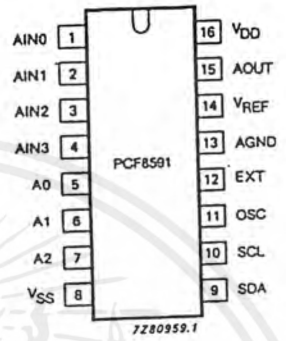


Fig.2 Pinning diagram.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

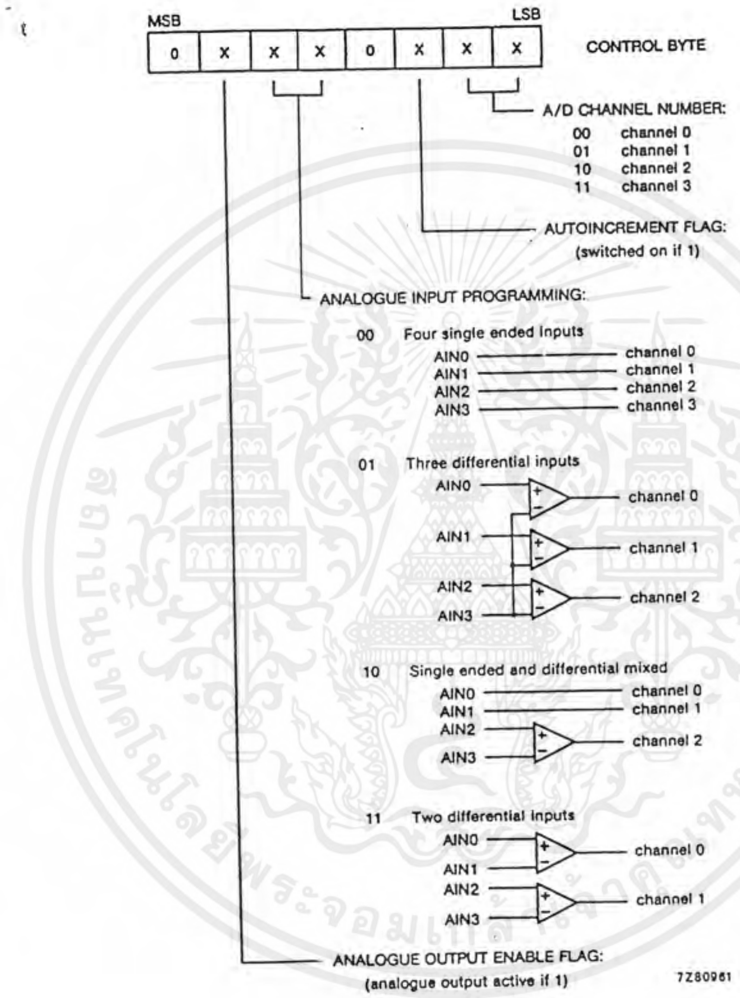


Fig.4 Control byte.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

7.4 A/D conversion

The A/D converter makes use of the successive approximation conversion technique. The on-chip D/A converter and a high-gain comparator are used temporarily during an A/D conversion cycle.

An A/D conversion cycle is always started after sending a valid read mode address to a PCF8591 device. The A/D conversion cycle is triggered at the trailing edge of the acknowledge clock pulse and is executed while transmitting the result of the previous conversion (see Fig.8).

Once a conversion cycle is triggered an input voltage sample of the selected channel is stored on the chip and is converted to the corresponding 8-bit binary code. Samples picked up from differential inputs are converted to an 8-bit two's complement code (see Figs 9 and 10).

The conversion result is stored in the ADC data register and awaits transmission. If the auto-increment flag is set the next channel is selected.

The first byte transmitted in a read cycle contains the conversion result code of the previous read cycle. After a Power-on reset condition the first byte read is a hexadecimal 80. The protocol of an I²C-bus read cycle is shown in Chapter 8, Figs 15 and 16.

The maximum A/D conversion rate is given by the actual speed of the I²C-bus.

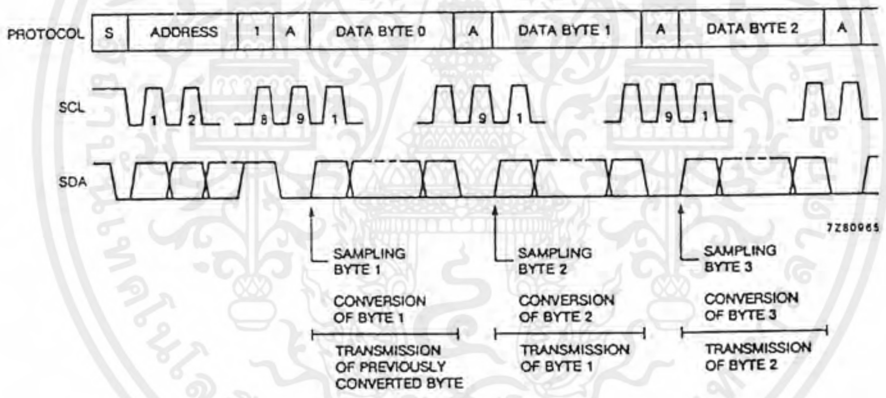


Fig.8 A/D conversion sequence.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

11 DC CHARACTERISTICS

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

SYMBOL	PARAMETER	CONDITIONS	MIN.	TYP.	MAX.	UNIT
Supply						
V_{DD}	supply voltage (operating)		2.5	–	6.0	V
I_{DD}	supply current					
	standby	$V_I = V_{SS}$ or V_{DD}; no load	–	1	15	μA
	operating, AOUT off	f_{SCL} = 100 kHz	–	125	250	μA
	operating, AOUT active	f_{SCL} = 100 kHz	–	0.45	1.0	mA
V_{POR}	Power-on reset level	note 1	0.8	–	2.0	V
Digital inputs/output: SCL, SDA, A0, A1, A2						
V_{IL}	LOW level input voltage		0	–	$0.3 \times V_{DD}$	V
V_{IH}	HIGH level input voltage		$0.7 \times V_{DD}$	–	V_{DD}	V
I_L	leakage current					
	A0, A1, A2	$V_I = V_{SS}$ to V_{DD}	−250	–	+250	nA
	SCL, SDA	$V_I = V_{SS}$ to V_{DD}	−1	–	+1	μA
C_i	input capacitance		–	–	5	pF
I_{OL}	LOW level SDA output current	V_{OL} = 0.4 V	3.0	–	–	mA
Reference voltage inputs						
V_{REF}	reference voltage	$V_{REF} > V_{AGND}$; note 2	V_{SS} + 1.6	–	V_{DD}	V
V_{AGND}	analog ground voltage	$V_{REF} > V_{AGND}$; note 2	V_{SS}	–	V_{DD} − 0.8	V
I_{LI}	input leakage current		−250	–	+250	nA
R_{REF}	input resistance	pins V_{REF} and AGND	–	100	–	kΩ
Oscillator: OSC, EXT						
I_{LI}	input leakage current		–	–	250	nA
f_{OSC}	oscillator frequency		0.75	–	1.25	MHz

Notes

1. The power on reset circuit resets the I²C-bus logic when V_{DD} is less than V_{POR}.
2. A further extension of the range is possible, if the following conditions are fulfilled:

$$\frac{V_{REF} + V_{AGND}}{2} \geq 0.8V, V_{DD} - \frac{V_{REF} + V_{AGND}}{2} \geq 0.4V$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

12 D/A CHARACTERISTICS

V_{DD} = 5.0 V; V_{SS} = 0 V; V_{REF} = 5.0 V; V_{AGND} = 0 V; R_L = 10 kΩ; C_L = 100 pF; T_{amb} = −40 °C to +85 °C unless otherwise specified.

SYMBOL	PARAMETER	CONDITIONS	MIN.	TYP.	MAX.	UNIT
Analog output						
V_{OA}	output voltage	no resistive load	V_{SS}	–	V_{DD}	V
		R_L = 10 kΩ	V_{SS}	–	0.9 × V_{DD}	V
I_{LO}	output leakage current	AOUT disabled	–	–	250	nA
Accuracy						
OS_e	offset error	T_{amb} = 25 °C	–	–	50	mV
L_e	linearity error		–	–	±1.5	LSB
G_e	gain error	no resistive load	–	–	1	%
t_{DAC}	settling time	to ½ LSB full scale step	–	–	90	µs
f_{DAC}	conversion rate		–	–	11.1	kHz
SNRR	supply noise rejection ratio	f = 100 Hz; V_{DDN} = 0.1 × V_{PP}	–	40	–	dB

13 A/D CHARACTERISTICS

V_{DD} = 5.0 V; V_{SS} = 0 V; V_{REF} = 5.0 V; V_{AGND} = 0 V; R_S = 10 kΩ; T_{amb} = −40 °C to +85 °C unless otherwise specified.

SYMBOL	PARAMETER	CONDITIONS	MIN.	TYP.	MAX.	UNIT
Analog inputs						
V_{IA}	analog input voltage		V_{SS}	–	V_{DD}	V
I_{LIA}	analog input leakage current		–	–	100	nA
C_{IA}	analog input capacitance		–	10	–	pF
C_{ID}	differential input capacitance		–	10	–	pF
V_{IS}	single-ended voltage	measuring range	V_{AGND}	–	V_{REF}	V
V_{ID}	differential voltage	measuring range; $V_{FS} = V_{REF} - V_{AGND}$	$\frac{-V_{FS}}{2}$	–	$\frac{+V_{FS}}{2}$	V
Accuracy						
OS_e	offset error	T_{amb} = 25 °C	–	–	20	mV
L_e	linearity error		–	–	±1.5	LSB
G_e	gain error		–	–	1	%
GS_e	small-signal gain error	ΔV_i = 16 LSB	–	–	5	%
CMRR	common-mode rejection ratio		–	60	–	dB
SNRR	supply noise rejection ratio	f = 100 Hz; V_{DDN} = 0.1 × V_{PP}	–	40	–	dB
t_{ADC}	conversion time		–	–	90	µs
f_{ADC}	sampling/conversion rate		–	–	11.1	kHz

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บรรณานุกรม

ชัยวัฒน์ ลิ้มพรจิตรวิไลและวรพจน์ กรแก้ววัฒนกุล.**เรียนรู้และปฏิบัติการไมโครคอนโทรลเลอร์ MCS 51**.กรุงเทพ:อินโนเวตีฟเอ็กเพอริเมนต์,2542

ปรเมษฐ์ ประนยานันท์และปิยพงศ์ เผ่าวนิช.**คู่มือและการประยุกต์ใช้งานไมโครคอนโทรลเลอร์ MCS 51**.กรุงเทพ:ห้างหุ้นส่วนจำกัดเอช.เอ็น.การพิมพ์,2536

สมยศ จุณะปิยะ.**การใช้งานไมโครคอนโทรลเลอร์ตระกูล MCS 51**.กรุงเทพ:มทป.,2537

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ประวัติผู้แต่ง

ชื่อผู้ทำปริญญานิพนธ์	นายพีระ แจ้งศิริกุล
วันเดือนปีเกิด	26 ธันวาคม 2521
สถานที่เกิด	จังหวัดเชียงใหม่
ภูมิลำเนาเดิม	17/6 ถ.ราชมรรคา ซ.3 ต.พระสิงห์ อ.เมือง จ.เชียงใหม่ 50200
ที่อยู่ปัจจุบัน	300/90 หมู่ 10 หมู่บ้านรุ่งอรุณ 1 ถ.ฉลองกรุง แขวงลำปลาทิว เขตลาดกระบัง กรุงเทพฯ 10520
โทรศัพท์	01-4240255

ประวัติการศึกษา

ประถมศึกษา	โรงเรียนอนุบาลเชียงใหม่
มัธยมศึกษาตอนต้น	โรงเรียนยุพราชวิทยาลัย
ประกาศนียบัตรวิชาชีพ (ปวช.)	วิทยาลัยเทคนิคเชียงใหม่
ประกาศนียบัตรวิชาชีพชั้นสูง (ปวส.)	วิทยาลัยเทคนิคเชียงใหม่
ปริญญาตรี	สาขาวิชาวิศวกรรมโทรคมนาคม ภาควิชาครุศาสตร์วิศวกรรม คณะครุศาสตร์อุตสาหกรรม
ผลงานที่ได้รับรางวัล	-
ทุนการศึกษา	-
คติพจน์	อย่าเชื่อจนกว่าจะได้เห็นด้วยตา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ประวัติผู้แต่ง

ชื่อผู้ทำปริญญานิพนธ์	นายเพ็ชรพรชัย หน่ายคอน
วันเดือนปีเกิด	1 มิถุนายน 2521
สถานที่เกิด	จังหวัดชลบุรี
ภูมิลำเนาเดิม	274 หมู่ 10 ต.คลองตะเกรา อ.ท่าตะเกียบ จ.ฉะเชิงเทรา 24160
ที่อยู่ปัจจุบัน	300/125 หมู่ 10 หมู่บ้านรุ่งอรุณ 1 ถ.ฉลองกรุง แขวงลำปลาทิว เขตลาดกระบัง กรุงเทพฯ 10520
โทรศัพท์	7374452-3 ห้อง 127

ประวัติการศึกษา

ประถมศึกษา	โรงเรียนบ้านศรีเจริญทอง
มัธยมศึกษาตอนต้น	โรงเรียนมัธยมวัชรวีร์
ประกาศนียบัตรวิชาชีพ (ปวช.)	วิทยาลัยเทคนิคฉะเชิงเทรา
ประกาศนียบัตรวิชาชีพชั้นสูง (ปวส.)	วิทยาลัยเทคนิคฉะเชิงเทรา
ปริญญาตรี	สาขาวิชาวิศวกรรมโทรคมนาคม ภาควิชาครุศาสตร์วิศวกรรม คณะครุศาสตร์อุตสาหกรรม
ผลงานที่ได้รับรางวัล	-
ทุนการศึกษา	-
คติพจน์	ไก่ยังมีขน คนต้องมีดี

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ประวัติผู้แต่ง

ชื่อผู้ทำปริญญานิพนธ์	นายอรรณพ อุยโต
วันเดือนปีเกิด	18 กุมภาพันธ์ 2522
สถานที่เกิด	จังหวัดฉะเชิงเทรา
ภูมิลำเนาเดิม	5 หมู่ 4 ต.คลองเขื่อน กิ่งอำเภอคลองเขื่อน จ.ฉะเชิงเทรา 24000
ที่อยู่ปัจจุบัน	111/12 ซ.ไปรษณีย์เก่า แขวงลาดกระบัง เขตลาดกระบัง กรุงเทพฯ 10520
โทรศัพท์	038-509118

ประวัติการศึกษา

ประถมศึกษา	โรงเรียนวัดคลองเขื่อนมงคลบุตรประชาสรรค์
มัธยมศึกษาตอนต้น	โรงเรียนเบญจมราชรังสฤษฎิ์
ประกาศนียบัตรวิชาชีพ (ปวช.)	วิทยาลัยเทคนิคฉะเชิงเทรา
ประกาศนียบัตรวิชาชีพชั้นสูง (ปวส.)	วิทยาลัยเทคนิคฉะเชิงเทรา
ปริญญาตรี	สาขาวิชาวิศวกรรมโทรคมนาคม ภาควิชาครุศาสตร์วิศวกรรม คณะครุศาสตร์อุตสาหกรรม
ผลงานที่ได้รับรางวัล	-
ทุนการศึกษา	-
คติพจน์	โลกนี้ไม่มีอะไรแน่นอน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้