สำนักหอสมุดกลาง พระจอมเกล้าลาดกระบัง

หุ่นยนต์หนูกล

MOUSE ROBOT

โดย

นายเดช กุลวงค์

นายอรรถพร ฉ่ำพงษ์สันติ

อาจารย์ที่ปรึกษา

ผศ.ดร.อรรถสิทธิ์ หล่าสกุล

เลขหมู่.....
เลขทะเบียน 42240
วัน, เดือน, ปี 15 พ.ค. 2545

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

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาอุตสาหกรรมศาสตรบัณฑิต

สาขาวิชาเทคโนโลยีโทรคมนาคม ภาควิชาเทคนิคอุตสาหกรรม

คณะวิศวกรรมศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2543

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หัวข้อปริญญานิพนธ์ หุ่นยนต์หนูกล

MOUSE ROBOT

นักศึกษา นายเดช กุลวงค์ เลขประจำตัว 42015597

นายอรรถพร ฉ่ำพงษ์สันติ เลขประจำตัว 42015626

อาจารย์ที่ปรึกษา ผศ.ดร.อรรถสิทธิ์ หล่าสกุล

ภาควิชา เทคนิคอุตสาหกรรม

ปีการศึกษา 2543

คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเหล้าเจ้าคุณทหารลาดกระบังอนุมัติให้ปริญญานิพนธ์ฉบับนี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาอุตสาหกรรมศาสตรบัณฑิต

..... อ.ที่ปรึกษา

(ผศ.ดร.อรรถสิทธิ์ หล่าสกุล)

..... กรรมการ

(.....)

..... กรรมการ

(.....)

..... กรรมการ

(.....)

..... กรรมการ

(.....)

..... กรรมการ

(.....)

ลิขสิทธิ์ของคณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หุ่นยนต์หนูกล

นายเดช กุลวงค์
นายอรรถพร ฉ่ำพงษ์สันติ
อาจารย์ที่ปรึกษา
ผศ.ดร.อรรถสิทธิ์ หล่าสกุล
ปีการศึกษา 2543

บทคัดย่อ

ปัจจุบันการใช้หุ่นยนต์ในงานลักษณะต่างๆนั้นมีการใช้งานอย่างกว้างขวางไม่ว่าจะในโรงงานอุตสาหกรรมหรือในงานการทหารและอื่นๆอีกมากมาย ซึ่งหุ่นใช้งานจริงๆนั้นจะมีราคาที่สูงมาก ดังนั้นการศึกษาถึงการควบคุมหุ่นยนต์ในลักษณะพื้นฐาน (ตามเส้น,ตามช่องทาง) ด้วยหุ่นจำลองขนาดเล็ก จึงเป็นจุดที่ผู้เริ่มต้นเรียนรู้ในงานด้านนี้น่าจะได้ศึกษาก่อน เพื่อก้าวสู่ระบบสูงต่อไป งานปริญญานิพนธ์นี้เป็นการสร้างจำลอง (Robot Mouse) และออกแบบโปรแกรมตัวอย่างของการควบคุมหุ่นจำลองนี้ในลักษณะ 2 แบบของเซ็นเซอร์ เป็นปริญญานิพนธ์ที่เน้นในการให้ผู้เรียนรู้ได้ฝึกทักษะของการเขียนโปรแกรมหุ่นยนต์ ตามจินตนาการของตัวเอง หุ่นยนต์ได้ถูกออกแบบด้วยวงจรง่ายๆ เพื่อความง่ายในการควบคุม และยังสามารถนำไปประยุกต์เพื่อสร้างหุ่นยนต์กู้ภัย, หุ่นยนต์สำรวจได้อีกด้วย โดยการเพิ่มตัวเซ็นเซอร์ต่างๆ เช่นกล้องแบบไร้สาย เป็นต้น ด้วยเหตุนี้โครงงานนี้จึงเป็นประโยชน์ต่อการเรียนรู้ขั้นพื้นฐานของการควบคุมหุ่นยนต์เป็นอย่างยิ่ง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

MOUSE ROBOT

Mr.Dach Kunwong

Mr.Atthaporn Champongsanti

Advisor

Asst.Prof.Dr.Attasit Lasakul

Year 2000

Abstract

Many Kinds of Robotics have been applied to factory and military purpose. One of problem to study about controlling the robotics moving is the high-cost of machine. The small robotics such as robot mouse is the most important/ suitable machine for people who want to study /learn that how to control the robot moving. This project is designed to build small robot with 2 fundamental sensors. The example programs are also designed to guide of controlling. Further more, the machine use the wireless system that means it is very easy for training and can be applied to another kind of applications.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กิตติกรรมประกาศ

ปริญญานิพนธ์ฉบับนี้ได้จัดทำขึ้นเป็นผลสำเร็จ ทางคณะผู้จัดทำต้องขอขอบพระคุณท่านบูรพาจารย์ทั้งหลายท่านผู้เขียนเอกสารและตำราอ้างอิงต่างๆ โดยเฉพาะอาจารย์ที่ปรึกษาอาจารย์อรรถสิทธิ์ หล่าสกุล เป็นอย่างสูงที่ช่วยกรุณาถ่ายทอดความรู้ดูแลและเอาใจใส่กระผมเป็นอย่างดี จนทำให้โครงงานชิ้นนี้สำเร็จลุล่วงไปได้ด้วยดี และขอบพระคุณบิดา – มารดา ที่เฝ้าตรากตรำทำงานส่งเสียให้พวกเราเรียนจนสำเร็จ ขอบคุณสำหรับกำลังใจจากเพื่อนๆที่ช่วยเหลือและค่อยให้กำลังใจมาโดยตลอด ทั้งนี้คณะผู้จัดทำต้องขอขอบพระคุณภาควิชาเทคนิคอุตสาหกรรม คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบังที่ให้โอกาสคณะผู้จัดทำได้มีโอกาสเข้ามาศึกษา ณ สถาบันแห่งนี้

สุดท้ายนี้ทางคณะผู้จัดทำ ขอขอบพระคุณท่านอาจารย์ทุกท่านที่กรุณาประสิทธิ์ประสาทวิชาความรู้ รวมถึงแนวทางความคิดและแนวทางปฏิบัติให้แก่คณะผู้จัดทำ จนทำให้ปริญญานิพนธ์ฉบับนี้สำเร็จผลตามเป้าหมายทุกประการ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ

เรื่อง	หน้า
บทคัดย่อภาษาไทย	ก
บทคัดย่อภาษาอังกฤษ	ข
กิตติกรรมประกาศ	ค
บทที่ 1 บทนำ	
1.1 ลักษณะขอบเขต	1
1.2 บล็อกไดอะแกรม	2
1.3 วัตถุประสงค์ และจุดมุ่งหมาย	3
1.4 ส่วนประกอบของโครงงาน	3
1.4.1 ซอร์ฟแวร์	3
1.4.2 ส่วนฮาร์ดแวร์	3
1.5 ประโยชน์ที่คาดว่าจะได้รับ	3
บทที่ 2 ทฤษฎีเบื้องต้นที่ควรทราบ	
2.1 สเต็ปมอเตอร์	4
2.2 สเต็ปมอเตอร์แบบยูนิโพล่าร์	4
2.3 การขับสเต็ปมอเตอร์	5
2.4 โครงสร้างการจัดขาไอซี	7
2.5 คุณสมบัติของ MCS-51	11
2.6 โครงสร้างและการทำงานของพอร์ต	15
2.7 การใช้งานพอร์ตอินพุต	16
2.8 การใช้งานพอร์ตเอาต์พุต	16
2.9 การอ่านค่าลอจิกจากพอร์ต	17
2.10 หน่วยความจำโปรแกรม	18
2.11 รีจีสเตอร์ฟังก์ชันพิเศษ	19
2.12 รีจีสเตอร์แสดงสถานะของไดอะแกรม	20
บทที่ 3 การทำงานของหุ่นยนต์ MOUSE ROBOT	
3.1 โฟโต้ทรานซิสเตอร์	21
3.2 การทำงานของโฟโต้ทรานซิสเตอร์	22

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.2 แผงวงจรตรวจจับแสง 22
3.3 แผงวงจรตรวจจับเส้น 24
3.4 วงจรตรวจจับทิศทางการทำงานตามเส้นและความต่างสี 24
บทที่ 4 สรุปผลการทดลองและแนวทางการแก้ไขปัญหา
บรรณานุกรม
ภาคผนวก
ภาคผนวก ก โปรแกรมการทำงานของหุ่นยนต์
1. อุปกรณ์ที่ใช้ใน โปรแกรม
2. Flowchart ของโปรแกรมการเดินตามเส้นสี
3. Flowchart ของโปรแกรมการเดินออกจากเขาวงกต
4. ส่วนโปรแกรมการทำงานของหุ่น
ภาคผนวก ข รายละเอียดของไอซีที่ใช้ในวงจร

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูป

เรื่อง	หน้า
รูปที่ 1.1 แสดงบล็อกไดอะแกรมควบคุมการทำงานของ Mouse Robot	2
รูปที่ 2.1 โครงสร้างพื้นฐานของสเต็ปเปอร์มอเตอร์แบบยูนิโพล่าร์	4
รูปที่ 2.2 วงจรขับสเต็ปเปอร์มอเตอร์ของไมโครคอลโทรลเลอร์ MCS-51	7
รูปที่ 2.3 แสดงการจัดขาของ AT89S8252	8
รูปที่ 2.4 การติดต่อคริสตอสเพื่อสร้างสัญญาณนาฬิกา	8
รูปที่ 2.5 แสดงผังไดอะแกรมของ MCS-51 พื้นฐาน	13
รูปที่ 2.6 แสดงผังไดอะแกรมของ AT89S8252	14
รูปที่ 3.1 แสดงโฟโต้ทรานซิสเตอร์	21
รูปที่ 3.2 แสดงวงจรไบแอสและกราฟลักษณะสมบัติของโฟโต้ทรานซิสเตอร์	22
รูปที่ 3.3 แสดงการนำโฟโต้ทรานซิสเตอร์ไปเปิด-ปิด ควบคุมการทำงานของรีเลย์	23
รูปที่ 3.3.1 วงจร Light-Interruption Alarm ที่ใช้โฟโต้ทรานซิสเตอร์ควบคุมเอสซีอาร์	23
รูปที่ 3.4 รูปวงจรแสดงการทำงานของตัวตรวจจับการทำงานตามเส้นและความต่างของสี	24

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญตาราง

เรื่อง หน้า

ตารางที่ 2.1 ลำดับการป้อนสัญญาณกระตุ้นของวงจรขับสเต็ปเปอร์มอเตอร์แบบ 1 เฟส 5
ตารางที่ 2.2 ลำดับการป้อนสัญญาณกระตุ้นของวงจรขับสเต็ปเปอร์มอเตอร์แบบ 2 เฟส 5
ตารางที่ 2.3 ลำดับการป้อนสัญญาณกระตุ้นของวงจรขับสเต็ปเปอร์มอเตอร์แบบฮาล์ฟสเต็ป 6
ตารางที่ 2.4 ตารางแสดงค่าพอร์ตต่างๆ 10
ตารางที่ 2.5 แสดงหน้าที่พิเศษของพอร์ต 1 ในไมโครคอลโทรลเลอร์ MCS-51 แบบเฟลช 15
ตารางที่ 2.6 แสดงการจัดพื้นที่ของรีจีเตอร์ SFR แต่ละตัวในหน่วยความจำข้อมูลส่วนบน 19
ตารางที่ 2.7 การเลือกแบงก์ของหน่วยความจำส่วนล่างเพื่อติดต่อกับรีจีสเตอร์แบงก์ R0-R7 20

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 1
บทนำ

ปัจจุบันนี้หุ่นยนต์ได้มีการออกแบบสร้างและพัฒนาขึ้นมาโดยวัตถุประสงค์ของการสร้างและพัฒนาหุ่นยนต์ คือสร้างความสะดวกสะบายให้กับมนุษย์ โดยหุ่นยนต์ที่สร้างขึ้นนั้นจะทำงานแทนหรือทำการรับใช้มนุษย์ โดยเฉพาะอย่างยิ่งในทางระบบอุตสาหกรรมได้นำหุ่นยนต์มาทำงานแทนแรงงานมนุษย์มากขึ้นเรื่อยๆเนื่องจากหุ่นยนต์มีการทำงานได้แม่นยำและแน่นอนมากกว่ามนุษย์ และหุ่นยนต์สามารถทำงานได้ตลอดเวลา ทำให้ต้นทุนการผลิตของระบบอุตสาหกรรมลดลง โดยการทำงานของหุ่นยนต์นั้นจะทำตามโปรแกรมที่กำหนด อุปกรณ์ที่นิยมใช้ควบคุมการทำงานหุ่นยนต์ได้แก่ ไมโครคอนโทรลเลอร์และคอมพิวเตอร์

โดยทั่วไปแล้วหุ่นยนต์แบ่งออกเป็น

1.Fixed Robotic เป็นหุ่นยนต์อยู่กับที่

2.Mouse Robot เป็นหุ่นยนต์สามารถเคลื่อนที่ได้โดยอิสระ โดยมีฐานเป็นล้อ

ในโครงานนี้จะกล่าวถึง Mouse Robot ที่บางทีเรียกว่า Autonomous Guide Vehicle System (AGVS) ซึ่งเป็นระบบจัดการหรือขนส่งพัสดุ ที่ใช้ยานพาหนะขับเคลื่อนด้วยตัวเองไปตามพื้นที่ที่กำหนดโดยรับพลังงานจากแบตเตอร์รี่ภายในตัวเอง

ประเภทของ AGVSแบ่งออกได้เป็น 3 ประเภท คือ

1.Driveless Trains

เปรียบเสมือนหัวรถจักรที่สามารถจูงรถพ่วงได้อีกหลายคัน

2.AGVS PALLET

เปรียบเสมือนตู้โดยสารของรถไฟ

3. AGVS Unit Load Carries

เป็นเครื่องจักรที่ทำหน้าที่ที่รับสำภาระจากพนักงานแล้วส่งขึ้นไป PalleTruck โดยอัตโนมัติ

การเคลื่อนที่ของ AGVS

1.การเคลื่อนที่ตามสายนำสัญญาณ(Signal Carrying Wire) โดย AGVS เคลื่อนที่ตามสายนำสัญญาณที่พื้น

2. การเคลื่อนที่ตามเส้นแถบสีที่พื้น(Tracking line on the floor) โดย AGVS เคลื่อนที่ตามแถบสีที่พื้น

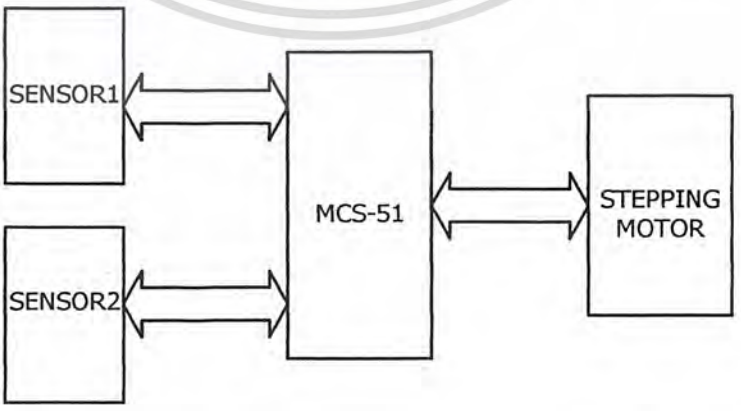
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การเคลื่อนที่ทั้งสองวิธีนั้น AGVS จำเป็นต้องมีความสามารถในการตรวจจับสภาพแวดล้อมหลายๆจุด เพื่อถ้าเจอปัญหาและอุปสรรคสามารถหยุดการเคลื่อนที่ได้ทันที การเคลื่อนที่โดยใช้สายนำสัญญาณ การติดตั้งสายทำได้ลำบากและราคาของสายแพง การเปลี่ยนแปลงโปรแกรมทำได้ยาก ส่วนการเคลื่อนที่ตามเส้นต้องมีการบำรุงรักษาเส้นแถบสีให้สะอาดเสมอ การเปลี่ยนแปลงโปรแกรมทำได้ยากเช่นกัน จากข้อเสียของการเคลื่อนที่ของ AGVS ดังกล่าวมานำไปสู่การคิดและพัฒนาโครงงานการควบคุมหุ่นยนต์เพื่อการขนส่งฉบับนี้

ในโครงงานนี้จะกล่าวถึงการควบคุมเส้นทางการเคลื่อนที่ของ Mouse Robot ให้ไปตามระยะทางและทิศทางที่กำหนด โดยการใช้คำสั่งควบคุมมาจากสถานีควบคุม (Control Station) จะเห็นว่าวิธีนี้ เป็นวิธีที่จะประหยัดและคุ้มค่าเมื่อมีการเปลี่ยนแปลงเส้นทางการเดินหุ่นยนต์บ่อยๆ และไม่ต้องสิ้นเปลืองค่าสายนำสัญญาณหรือเส้นแถบสีบนพื้น แต่การออกแบบทำยากกว่า เพราะต้องใช้ระบบสื่อสารที่ดีติดต่อระหว่างหุ่นยนต์และสถานีควบคุม และหุ่นยนต์ต้องมีข้อผิดพลาดในเคลื่อนที่น้อยมาก

1.1 ลักษณะและขอบเขต

ลักษณะการโปรแกรมการเคลื่อนที่ของ Mouse Robot ในโครงงาน จะใช้แถบเส้น ในการทางการเดินของ Mouse Robot โดยผ่านการควบคุมโดยโปรแกรมการทำงานที่เขียนไว้ในโปรแกรมคำสั่งทำสามารถเปลี่ยนแปลงโปรแกรมการทำงานของ Mouse Robot ได้สะดวกและรวดเร็ว รวมทั้งสามารถบันทึกเส้นทางการเคลื่อนไว้เพื่อเปิดมาใช้งานในครั้งต่อไป ตรวจสอบข้อผิดพลาดการทำงานของหุ่นยนต์สามารถดูได้จากการแสดงตำแหน่งและการทำงานจากการเขียนคำสั่งจากโปรแกรมที่สั่งของคอมพิวเตอร์ เราสามารถเขียนบล็อกไดอะแกรมการทำงานของโครงงานได้ดังรูป



รูปที่ 1.1 แสดงบล็อกไดอะแกรมควบคุมการทำงานของ Mouse Robot

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1.2 โดยบล็อกไดอะแกรม

1. สถานีควบคุม ใช้ไมโครคอมพิวเตอร์
2. ตัวหุ่นยนต์ เป็น Mouse Robot ขนาดเล็ก ใช้ไมโครคอนโทรลเลอร์เป็นตัวควบคุมการทำงาน ระบบการขับเคลื่อนใช้สเต็ปเปอร์มอเตอร์ การติดต่อระหว่างหุ่นยนต์และตัว Sensor ในการควบคุมการทำงานของตัวหุ่น

1.3 วัตถุประสงค์และจุดมุ่งหมาย

1.เพื่อศึกษาการทำงานของ Mouse Robot

2.เพื่อออกแบบโปรแกรมการควบคุมการเคลื่อนที่ของ Mouse Robot

3.เพื่อออกแบบโปรแกรมแสดงผล ตำแหน่งและการทำงานของหุ่นยนต์

1.4 ส่วนประกอบของโครงงาน

โครงงานนี้ประกอบด้วย 2 ส่วนคือ

1.4.1 ซอร์ฟแวร์

ส่วนที่เป็นซอร์ฟแวร์ คือส่วนที่เป็นโปรแกรมใช้ในการควบคุมเส้นทางการเคลื่อนที่และแสดงตำแหน่งของ Mouse Robot

1.4.2 ส่วนฮาร์ดแวร์

ส่วนของฮาร์ดแวร์ คือ Mouse Robot เป็นแบบจำลองการทำงาน

1.5 ประโยชน์ที่คาดว่าจะได้รับ

1.สามารถนำไปใช้ในการขนส่งในโรงงานอุตสาหกรรม สนามบิน ท่าเรือ สถานีรถไฟหรือที่อื่นๆได้จริง

2.สามารถนำ Mouse Robot ไปประยุกต์ใช้งานเป็นหุ่นยนต์ทำความสะอาดตามสถานที่ต่างๆ เช่น โรงยิม ได้

3. สามารถนำ Mouse Robot ไปประยุกต์ใช้งานเป็นหุ่นยนต์ไกด์นำชมสถานที่ต่างๆเช่น พิพิธภัณฑ์ งานแสดงสินค้า

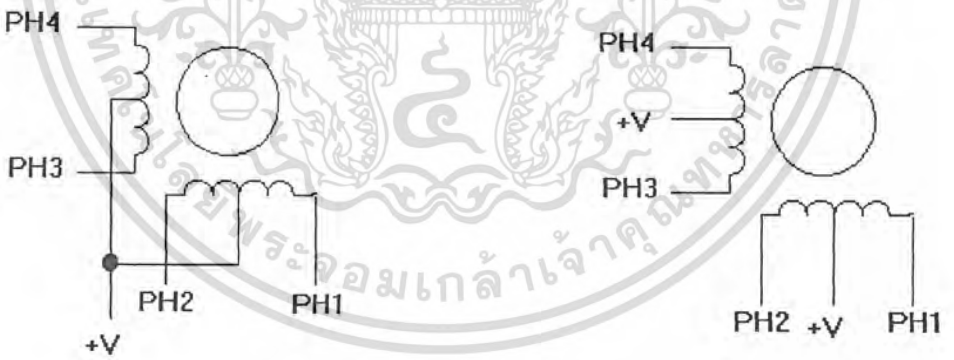
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 2
ทฤษฎีเบื้องต้นที่ควรทราบ

2.1 สเต็ปเปอร์มอเตอร์

สเต็ปเปอร์มอเตอร์ (StepperMotor) เป็นมอเตอร์ที่มีการหมุนเป็นจังหวะหรือเรียกว่าเป็นสเต็ป กล่าวคือ เมื่อจ่ายสัญญาณให้แก่มอเตอร์อย่างถูกต้อง มอเตอร์จะหมุนไปเป็นจังหวะไม่เหมือนมอเตอร์อื่นทั่วไป ที่เมื่อทำการจ่ายไฟแล้วก็จะหมุนไปทันที ส่วนจะหมุนซ้ายหรือขวานั้นขึ้นอยู่กับการจ่ายไฟให้แก่ขดลวดภายในมอเตอร์ ความละเอียดของการหมุนของสเต็ปเปอร์จะขึ้นอยู่กับองค์ประกอบสำคัญ 2 ประการคือ โครงสร้างของสเต็ปเปอร์มอเตอร์และวงจรขับ โดยวงจรขับสามารถที่จะช่วยให้การหมุนของสเต็ปเปอร์มอเตอร์มีความละเอียดมากขึ้นได้

เหตุผลที่สเต็ปเปอร์มอเตอร์ได้รับความนิยมมากก็คือสามารถกำหนดตำแหน่งของการหมุนได้อย่างแม่นยำ โดยไม่ต้องใช้วงจรขับที่มีความซับซ้อนมากนักไม่ต้องสร้างวงจรป้อนกลับเพื่อตรวจสอบตำแหน่งต้นทางหรือปลายทางและเมื่อมอเตอร์ไม่หมุนก็จะไม่มีการใช้พลังงานจึงไม่เกิดความสูญเสียมากมายดังเช่นมอเตอร์ธรรมดา



รูปที่2.1 โครงสร้างพื้นฐานของสเต็ปเปอร์มอเตอร์แบบยูนิโพล่าร์

2.2 สเต็ปเปอร์มอเตอร์แบบยูนิโพล่าร์

สเต็ปเปอร์มอเตอร์ที่มีใช้มีด้วยกัน 2 แบบคือ แบบยูนิโพล่าร์และไบโพล่าร์ ในปัจจุบันสเต็ปเปอร์มอเตอร์แบบไบโพล่าร์แทบจะไม่มีการใช้งาน เนื่องจากวงจรที่ขับมีความยุ่งยากและใช้อุปกรณ์มากโครงสร้างของสเต็ปเปอร์มอเตอร์แบบยูนิโพล่าร์แสดงดังรูป สเต็ปเปอร์มอเตอร์แบบนี้จะมีการพันมอเตอร์สองขดบนแต่ละขั้วแม่เหล็กสเตเตอร์แต่ละขดแบ่งเป็น 2 เฟส รวมมอเตอร์ทั้ง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตัวมีขดลวดทั้งสิ้น 4 เฟส คือเฟส 1,2,3 และ 4 มีการต่อสายออกมาจากขดลวดแต่ละขดเพื่อจ่ายไฟเลี้ยง ทำให้มอเตอร์แบบนี้จึงมีทั้งแบบ 5 และ 6 สาย โดยสายที่ 5 คือสายจ่ายไฟเลี้ยง ส่วนกรณี 6 สายจะต้องนำสายไฟเลี้ยงของขดลวดทั้ง 2 ของมอเตอร์มาต่อรวมกันแล้วจ่ายไฟจึงจะทำให้สเต็ปเปอร์มอเตอร์ทำงานได้

2.3 การขับสเต็ปเปอร์มอเตอร์

การขับสเต็ปเปอร์มอเตอร์หมุนจะต้องป้อนสัญญาณไฟฟ้าไปยังแต่ละเฟสของขดลวดอย่างเหมาะสมและมีรูปแบบที่ถูกต้อง สเต็ปเปอร์มอเตอร์จึงหมุนได้ โดยมีรูปแบบการขับอย่างง่าย 3 รูปแบบคือ แบบ1 เฟส,แบบ 2 เฟสหรือฟูลสเต็ป(full step) และแบบฮาล์ฟสเต็ป(half step)

แบบ 1 เฟสเป็นการขับที่ง่ายที่สุด โดยทำการป้อนสัญญาณกระตุ้นขดลวดครั้งละเฟสในช่วงเวลาหนึ่งไล่เรียงกันไป เช่น เริ่มจากเฟสที่ 1 ต่อด้วยเฟสที่ 2,3 และ 4 แล้ววนกลับมาเฟส 1 ใหม่ หรือจะให้เริ่มเฟสที่ 1 ไปยังเฟสที่ 4,3 และ 2 แล้ววนกลับมาเฟสที่ 1 อีกครั้ง ด้วยลำดับการป้อนสัญญาณกระตุ้นที่ต่างกัน ทำให้ทิศทางการหมุนของสเต็ปเปอร์มอเตอร์สวนทางกันด้วย การขับสเต็ปเปอร์มอเตอร์แบบนี้ จะมีเพียงเฟสเดียวที่ได้รับสัญญาณกระตุ้น ในตารางที่ 2.1 แสดงลำดับการป้อนสัญญาณเพื่อขับสเต็ปเปอร์มอเตอร์แบบ 1 เฟส

สเต็ปที่	PHASE1	PHASE2	PHASE3	PHASE4
1	ทำงาน			
2		ทำงาน		
3			ทำงาน	
4				ทำงาน

ตารางที่ 2.1 ลำดับการป้อนสัญญาณกระตุ้นของวงจรขับสเต็ปเปอร์มอเตอร์แบบ 1 เฟส

สเต็ปที่	PHASE1	PHASE2	PHASE3	PHASE4
1	ทำงาน	ทำงาน		
2		ทำงาน	ทำงาน	ทำงาน
3			ทำงาน	
4	ทำงาน			ทำงาน

ตารางที่ 2.2 ลำดับการป้อนสัญญาณกระตุ้นของวงจรขับสเต็ปเปอร์มอเตอร์แบบ 2 เฟส

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

แบบ 2 เฟสจะมีลักษณะคล้ายกับแบบ 1 เฟส บางทีเรียกการขับแบบนี้ว่า แบบฟูลสเต็ป (Full Step) แต่แทนที่จะส่งสัญญาณกระตุ้นเพียงเฟสเดียว ในการขับแบบนี้จะป้อนสัญญาณกระตุ้นไปยังเฟสของมอเตอร์ที่อยู่ใกล้กันในเวลาเดียวกันและเรียงลำดับกันไปเช่นเดียวกับแบบ 1 เฟส ดังแสดงในตารางที่ 2.2 คือเริ่มด้วยการป้อนสัญญาณกระตุ้นไปยังเฟส 1 และ 2 พร้อมกัน

ในสเต็ปที่ 1 ในสเต็ปที่ 2 จะป้อนสัญญาณไปยังเฟสที่ 2 และ 3 ถัดมาในเฟสที่ 3 จะทำการป้อนสัญญาณกระตุ้นไปที่เฟส 3 และ 4 ในสเต็ปที่ 4 จะป้อนสัญญาณไปยังเฟสที่ 4 และ 1แล้ววนกลับไปที่เฟส 1 และ 2 อีกครั้ง ด้วย การขับแบบนี้ทำให้แรงบิดหรือทอล์ก(toque)มากกว่าแบบหนึ่งเฟส แต่ข้อเสียคือ ใช้พลังงานในการขับเพิ่มมากขึ้น

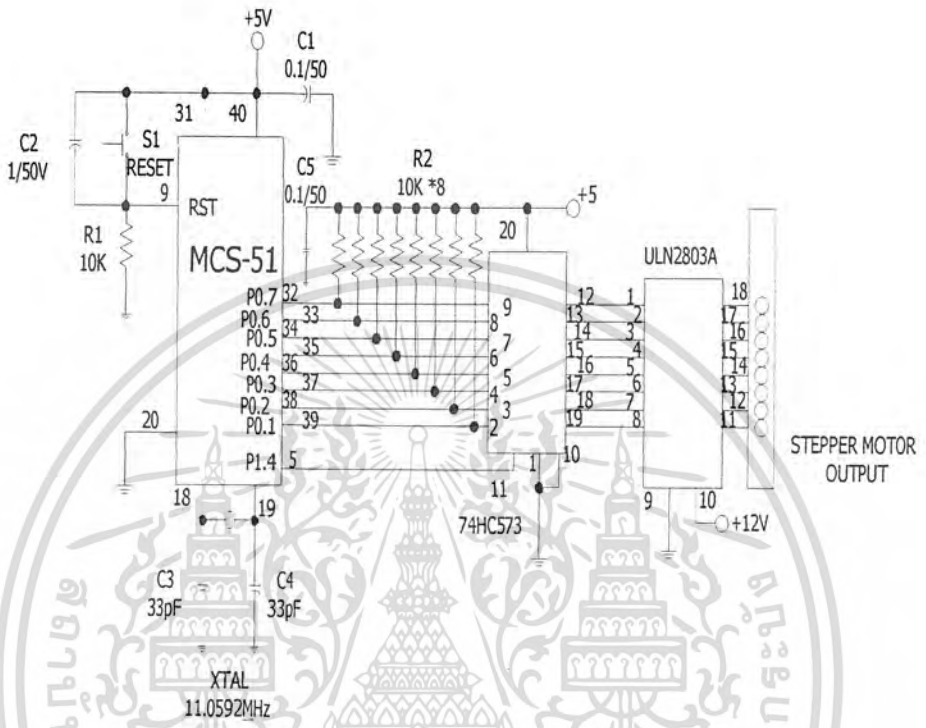
สเต็ปที่	PHASE1	PHASE2	PHASE3	PHASE4
1	ทำงาน			
2	ทำงาน	ทำงาน		
3		ทำงาน		
4		ทำงาน	ทำงาน	
5			ทำงาน	
6			ทำงาน	ทำงาน
7				ทำงาน
8	ทำงาน			ทำงาน

ตารางที่ 2.3 ลำดับการป้อนสัญญาณกระตุ้นของวงจรขับสเต็ปเปอร์มอเตอร์แบบฮาล์ฟสเต็ป

แบบฮาล์ฟสเต็ป การขับแบบนี้ได้รับความนิยมมากที่สุด เนื่องจากสามารถช่วยให้สเต็ปเปอร์มอเตอร์สามารถหมุนได้อย่างละเอียดมากขึ้นเป็นสองเท่าของความละเอียดปกติของสเต็ปเปอร์มอเตอร์ โดยมีรูปแบบการขับให้หมุนแสดงในตารางที่ 2.3 จะเห็นได้ว่า การขับสเต็ปเปอร์มอเตอร์แบบนี้เป็นการผสมผสานระหว่างแบบ 1เฟสและ 2 เฟส กล่าวคือ มีทั้งการป้อนสัญญาณกระตุ้นไปยังขดลวดเพียงเฟสเดียวและพร้อมกับแบบ 2 เฟสในช่วงเวลาหนึ่ง ด้วยการขับแบบนี้ส่งผลให้แรงบิดที่ได้จาการหมุนเพิ่มขึ้น เพราะระยะทางในการหมุนสั้นลง ความถูกต้องของตำแหน่งที่หมุนมีเพิ่มมากขึ้น เพียงแต่ว่าในการขับแต่ละสเต็ปจะให้ผลเพียงครึ่งสเต็ปของการขับปกติดังนั้นหาก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ต้องการให้เคลื่อนที่เป็นไปแบบเต็มสเต็ปจะต้องกำหนดให้ทำการหมุนไป 2 สเต็ปต่อเนื่องกัน

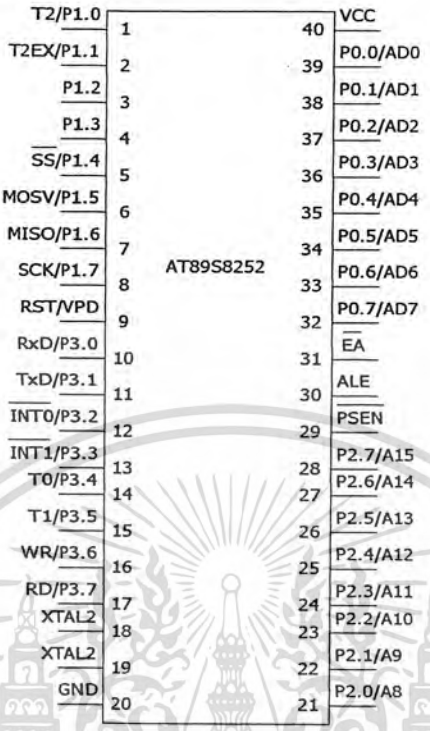


รูปที่ 2.2 วงจรขับสเต็ปเปอร์มอเตอร์ของไมโครคอลโทรเลอร์ MCS-51

2.4 โครงสร้างการจัดขาไอซี

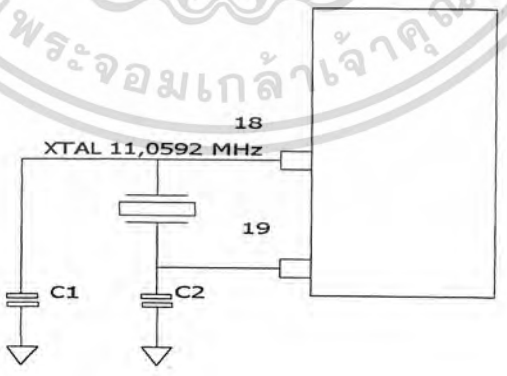
1. ขาที่ 40 หรือ VCC มีหน้าที่สำหรับจ่ายไฟกระแสตรง ปกติแล้วเราจะใช้ไฟฟ้ามีแรงดัน 5V
2. ขาที่ 20 เป็นขาใช้สำหรับต่อลงกราวด์
3. ขาที่ 9 เป็นขาสำหรับสร้างสัญญาณรีเซ็ต (Reset signal) แก่ตัวไมโครคอนโทรเลอร์ ซึ่งมีข้อกำหนดเอาไว้ว่า จะต้องจ่ายแรงดันไปขานี้อย่างน้อยเป็นเวลา 2 แมชชีนไซเคิล แล้วหยุดป้อนแรงดัน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.3 แสดงการจัดขาของ AT89S8252

4. ขาที่ 18 และ 19 เป็นขาที่ใช้สำหรับต่อกับวงจรกำเนิดสัญญาณ ซึ่งจะมีอยู่ด้วยกันหลายรูปแบบ แต่ที่นิยมใช้กันจะเป็นวงจรความถี่ที่เกิดจากคริสตอล



รูปที่ 2.4 การติดต่อคริสตอลเพื่อสร้างสัญญาณนาฬิกา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หมายเหตุ ค่าของ C1 และ C2 จะเป็นค่าอะไรก็ได้ที่อยู่ในช่วง 20 pF ถึง 30 pF และการทำงานของไมโครคอนโทรเลอร์ MCS-51 มีข้อกำหนดเอาไว้ว่า การทำงานใน 1 แมชีนไซเคิล (Machine cycle) หรือ ครบวงรอบการทำงาน ซึ่งก็คือ กระบวนการอ่านคำสั่ง (Fetch) และประมวลผลคำสั่ง (Execute) จะใช้สัญญาณนาฬิกา 12 ลูก ดังนั้น ถ้าใช้คริสตอลความถี่ 12 MHz ก็จะได้ว่า ไมโครคอนโทรลเลอร์จะสามารถประมวลผลได้ 1000000 แมชีนไซเคิลต่อวินาทีนั่นเอง แต่ปกติแล้วบอร์ดไมโครคอนโทรลเลอร์ MCS-51 จะใช้คริสตอลที่ความถี่ 11.0592 MHz ทั้งนี้เนื่องจากเพื่อนำเอาค่าความถี่นี้ไปสร้างสัญญาณนาฬิกาสำหรับการสื่อสารแบบอนุกรม

5. ขาที่ 31 หรือ EA เป็นขาที่ทำหน้าทีใช้ในการกำหนดว่า ในการทำงานนั้นให้อ่านข้อมูลโปรแกรมจากในตัวชิพไมโครคอนโทรเลอร์เองหรือจากหน่วยความจำโปรแกรมภายนอก โดยถ้าขานี้มีสถานะเป็น 0 (ขาใดมีเครื่องหมายขีดบนชื่อแสดงว่าขานั้นทำงานที่สถานะเป็น 0 หรือต่อลงกราวด์)
6. ขาที่ 29 หรือ PSEN ขานี้เมื่อใดมีสถานะเป็น 0 นั่นหมายความว่า ไมโครคอนโทรเลอร์กำลังอ่านข้อมูลโปรแกรมจากหน่วยความจำโปรแกรมภายนอก
7. ขาที่ 30 หรือ ALE เป็นขาที่ทำหน้าทีบอกให้รู้ว่ามีการส่งข้อมูลที่เป็นตำแหน่งออกมาที่พอร์ต 0 ของไมโครคอนโทรเลอร์ ทั้งนี้เพราะ พอร์ต 0 ของไมโครคอนโทรเลอร์นั้นสามารถที่จะส่งได้ทั้งข้อมูลที่เป็นตำแหน่งของหน่วยความจำ และข้อมูล
8. ขาที่ 1 ถึง 8 ทำหน้าที่เป็นพอร์ตอินพุต/เอ้าท์พุต ซึ่งเรียกกันว่าพอร์ต 1 หรือ PO พอร์ตนี้สามารถใช้งานได้ทั่วไป แต่ในบางรุ่นเช่น 8052/8032 ที่ขา P1.0 และ P1.1 จะเป็นขาที่ใช้กับไทเมอร์ 2 ด้วย
9. ขาที่ 32 ถึง 39 เป็นพอร์ต 0 หรือ P0 ของไมโครคอนโทรเลอร์ เป็นได้ทั้งพอร์ตอินพุตและเอ้าท์พุต(แต่ต้องทำการ Pullup ด้วย เนื่องจากพอร์ตนี้ไม่มีวงจรนี้) นอกจากเป็นพอร์ตอินพุตและเอ้าท์พุตแล้ว พอร์ต 0 ยังทำหน้าที่พิเศษอีกคือ ส่งบิตของแอดเดรสที่เป็นไบต์ต่ำ และเป็นพอร์ตรับ/ส่งข้อมูลหรือเรียกอีกอย่างหนึ่งว่าเป็นทั้งแอดเดรสบัส และดาต้าบัสหรือบัสข้อมูลในตัวเดียว
10. ขาที่ 21 ถึง 28 พอร์ต 2 หรือ P2 ของไมโครคอนโทรลเลอร์ สามารถใช้ได้เป็นทั้งพอร์ตอินพุตและเอาท์พุต นอกจากนี้ยังทำหน้าที่พิเศษอีกอย่างคือ เป็นแอดเดรสบัสที่ส่งไบต์สูงออกจากตัวไมโครคอนโทรเลอร์อีกด้วย (ใช้คู่กับพอร์ต 0) ตรงนี้จะเห็นว่า เมื่อเราทำการต่อแบบมีหน่วยความจำภายนอกชิพไมโครคอนโทรลเลอร์ จะทำให้เราสูญเสียพอร์ตไป 2 พอร์ต สำหรับเป็นแอดเดรสบัส (พอร์ต 0 และพอร์ต 2) และบัสข้อมูล(พอร์ต 0) จึงทำให้เรามีพอร์ตใช้งานน้อยลงไปด้วย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

11. ขาที่ 10 ถึง 13 หรือ P3 ของไมโครคอนโทรลเลอร์ เป็นได้ทั้งพอร์ตอินพุตและเอ้าท์พุตของไมโครคอนโทรลเลอร์ นอกจากนี้แต่ละขายังทำหน้าที่พิเศษอีกหลายอย่างดังนี้

ชื่อขา	หน้าที่
P3.0	RxD หรือใช้ในการรับข้อมูลจากการสื่อสารแบบอนุกรม
P3.1	TxD หรือใช้ในการส่งข้อมูลจากการสื่อสารแบบอนุกรม
P3.2	INT0 หรือรับสัญญารอินเทอร์รัพท์จากอุปกรณ์ภายนอกจากตัวไมโครคอนโทรลเลอร์
P3.3	INT1หรือรับสัญญารอินเทอร์รัพท์จากอุปกรณ์ภายนอกจากตัวไมโครคอนโทรลเลอร์
P3.4	TO หรือใช้งานกับ TIMER COUNTER 0
P3.5	T1 หรือใช้งานกับ TIMER COUNTER 0
P3.6	WR หรือทำหน้าที่ส่งสัญญาณเพื่อบอกแก่หน่วยความจำข้อมูลว่ากำลังส่งข้อมูลจากบัสข้อมูลไปเก็บไว้ในหน่วยความจำ ณ ตำแหน่งที่ชี้โดยแคสเดรสบัส
P3.7	RD หรือทำหน้าที่ส่งสัญญาณเพื่อบอกแก่หน่วยความจำข้อมูลว่าต้องการอ่านข้อมูลจากหน่วยความจำใด ณ ตำแหน่งที่ชี้โดยแอดเดรสบัส ผ่านทางบัสข้อมูล

ตารางที่ 2.4 ตารางแสดงค่าพอร์ตต่าง

หมายเหตุ พอร์ตต่างๆของไมโครคอลโทรลเลอร์ MCS-51 มีข้อกำหนดทางกระแสไฟฟ้าดังนี้

พอร์ต 0 แต่ละขาสามารถส่งกระแสซอร์ส (กระแสที่ส่งออกจากขาของพอร์ต) ได้ขาละ 10 mA และทุกขาส่งค่า 1 พร้อมกันได้ 26 mA

พอร์ต 1 ถึงพอร์ต 3 สามรถสร้างกระแสซอร์สได้ขาละ 15 mA และทุกขาของทั้ง 3 พอร์ตรวมกันจะได้กระแส 71 mA

กรณีที่ต้องการใช้พอร์ตเป็นอินพุต เราจะต้องส่งบิต 1 ไปที่ทุกขาของพอร์ตนั้นก่อน เพราะจะทำให้พอ์ตนั้นมีค่าอิมพีแดนซ์ (impedance : ค่าความต้านทานกระแส) สูง จึงทำงานเป็นพอร์ตอินพุตได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.5 คุณสมบัติของ MCS-51

MCS-51 นั้น เป็นไมโครคอนโทรลเลอร์ตระกูลหนึ่งที่ได้รับความนิยมมาช้านาน และได้รับการตอบรับอย่างดีจากผู้ผลิตไอซี ซึ่งสังเกตได้จากการที่โครงสร้างหลักของไมโครคอนโทเลอร์ตระกูลนี้ยังคงเหมือนเดิม โค้ดโปรแกรมเดิมนั้นยังคงนำมาใช้ได้กับไมโครคอนโทรเลอร์รุ่นใหม่ๆ ที่ออกสู่ท้องตลาด ทั้งที่ไมโครคอนโทรเลอร์รุ่นใหม่ๆ ที่ออกมานั้น มีประสิทธิภาพและความเร็สูงกว่าเดิมมากขึ้นเรื่อยๆ ดังนั้น คุณสมบัติต่างๆ จากหัวข้อนี้จึงเป็นพื้นฐานที่ดีสำหรับการนำไปใช้กับไมโครคอนโทรลเลอร์รุ่นใหม่ๆ ในปัจจุบัน ซึ่งคุณสมบัติของ MCS-51 แบบคร่าวๆ เป็นดังนี้

1. เป็นไมโครคอนโทรเลอร์ที่ทำงานแบบ 8 บิต
2. มีแอดเดรสบัสกว้าง 16 บิต
3. รีจิสเตอร์ขนาด 8 บิต จำนวน 34 ตัว
4. รีจิสเตอร์ขนาด 16 บิต จำนวน 2 ตัว
5. หน่วยความจำแบบสแต็ก มีขนาด 128 ไบต์
6. มีหน่วยความจำแรม (Ram) ภายใน 128 ไบต์ (8051/8031) หรือ 256 ไบต์ (8052/8032)
7. อ้างอิงหน่วยความจำภายนอกสูงสุด 128 KB (ข้อมูล 64 KB , โปรแกรม 64 KB)
8. มีแฟล็กเก็บสถานะการทำงาน 4 ตัว
9. มีไทเมอร์ 2 ตัว (ใน 8051/8031) และอีก 3 ตัว (ใน 8052/8032)
10. มีพอร์ตแบบขนาน 4 พอร์ต
11. มีพอร์ตแบบอนุกรม 1 พอร์ต ทำงานแบบฟูลดูเพล็กซ์ (Full Duplex)
12. บางรุ่นมีหน่วยความจำ ROM , EPROM , Flash ROM , EEROM ในตัวเอง
13. มีพอร์ตอินพุต/เอาท์พุต (I/O port) จำนวน 32 บิต
14. มีวงจรออสซิลเลเตอร์ (Oscillator) และวงจรผลิตสัญญาณนาฬิกาภายในไอซี
15. มีแหล่งเกิดอินเทอร์รัพท์ (Interrupt) 5 แหล่ง (ใน 8051/8031) และ 6 แหล่ง (ใน 8052/8032)
16. มีคำสั่งคำนวณทางคณิตศาสตร์และตรรกศาสตร์ (Boolean processor)

นอกจากนี้คุณสมบัติพื้นฐานดังที่กล่าวมานี้ ในไอซีไมโครคอนโทรเลอร์บางเบอร์จะมีความสามารถที่สูงกว่าที่กำหนดเอาไว้อีกหลายอย่าง ซึ่งก็แล้วแต่ผู้ผลิตไอซีไมโครคอนโทรเลอร์ออกแบบเอาไว้อย่างไร แต่จุดหนึ่งที่ไอซีทุกเบอร์มีเหมือนกันก็คือ สามารถใช้คำสั่งพื้นฐาน และมีความสามารถพื้นฐานเหมือนกัน ดังนั้น การเขียนโปรแกรมตามมาตรฐานของไมโครคอนโทรเลอร์ MCS-51 จะทำให้ผู้พัฒนาสามารถนำไปใช้กับไอซีไมโครคอนโทรเลอร์รุ่นอื่นๆ ได้อีกด้วย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

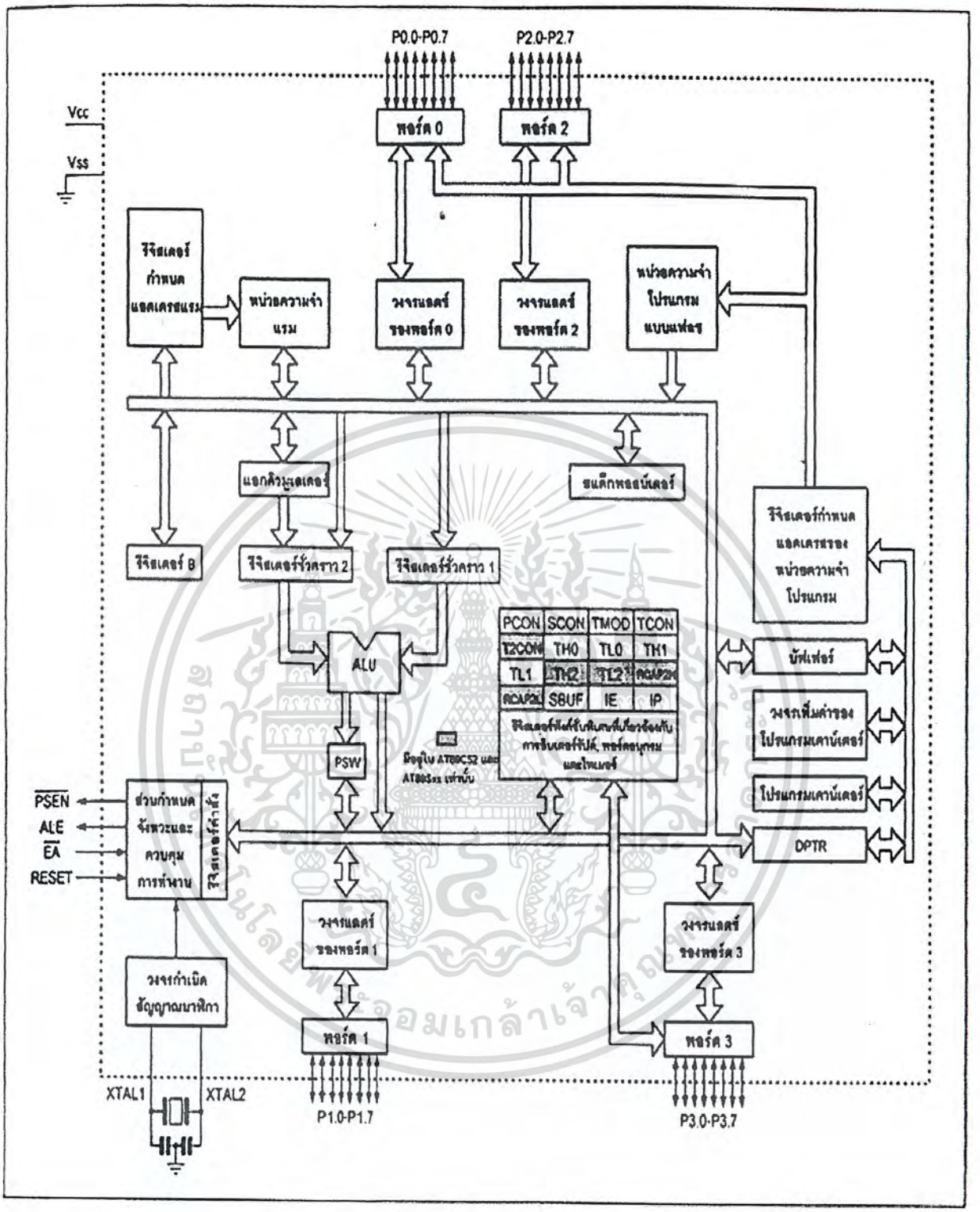
ปัจจุบันเทคโนโลยีทางด้านการสร้างไมโครคอนโทรเลอร์แบบชิพเดี่ยว (Single Chip) ก้าวหน้าไปกว่าเดิมมาก คือ ในไอซีเบอร์ 80C51/80C52 เป็นไมโครคอนโทรเลอร์ที่สามารถโปรแกรมเก็บเอาไว้ในตัวได้เลย แต่การโปรแกรมนั้นต้องอาศัยเครื่องมือในการเขียนจากโรงงานผลิต ซึ่งไม่เหมาะสำหรับนักพัฒนาทั่วๆ ไป ทางผู้ผลิตเลยปรับเปลี่ยนรูปแบบการโปรแกรมไอซีมาเป็นแบบ EPROM แต่ก็มีปัญหาในเรื่องของการลบข้อมูลในตัวชิพ เพราะจะต้องอาศัยเครื่องลบแบบแสง UV

หลังจากนั้นก็เลยมีผู้ผลิตชิพที่สามารถโปรแกรมและลบได้ด้วยการใช้กระแสไฟฟ้า ทำให้ต้นทุนในการเขียนและลบข้อมูลในตัวชิพถูกลงมากทำให้ผู้ใช้นิยมที่จะเลือกไมโครคอนโทรเลอร์ที่สามารถเขียนและลบด้วยสัญญาณทางไฟฟ้ากันมากขึ้น หนึ่งในหลานบริษัทที่ออกแบบชิพลักษณะนี้คือบริษัท Atmel ตัวอย่างไมโครคอนโทรเลอร์ที่ Atmel ผลิตก็ได้แก่ไอซีเบอร์ AT89C51 , AT89C52 , AT89C1051 , AT89C2051 , AT89C4051 , AT89S8252 , AT89C55 ฯลฯ ซึ่งไมโครคอนโทรเลอร์ของ Atmel จะใช้หน่วยความจำแบบแฟลซ (Flash Memory) ทำให้สามารถเขียนและลบได้กว่า 1,000 ครั้ง และสามารถเก็บโปรแกรมได้นับ 10 ปี และโดยเฉพาะรุ่น AT89Sxxxx นั้นจะสามารถลบและ เขียนโปรแกรมได้โดยไม่ต้องนำตัวชิพไปโปรแกรมด้วยเครื่องโปรแกรม ทั้งนี้เพราะไมโครคอนโทรเลอร์รุ่นนี้ถูกออกแบบให้สามารถทำการโปรแกรมชิพด้วยระบบ In-System reprogrammable โดยอาศัยวงจร SPI (Serial Peripheral Interface) ในการเขียนและลบ(ดังรูป2.6)

จากความง่ายในการเขียนข้อมูลและการลบข้อมูลออกจากตัวชิพ จึงมีผู้พัฒนาระบบมากมายเลือกใช้ไมโครคอนโทรเลอร์ตระกูลนี้ ดังนั้น ในหนังสือเล่มนี้จึงอ้างอิงถึงความสามารถของไมโครคอนโทรเลอร์ตระกูล MCS-51 เอาไว้ด้วย และในหัวข้อสุดท้ายของบทนี้จะกล่าวถึงตัวไมโครคอนโทรเลอร์เบอร์ AT89C1051 , AT89C2051 และ AT89C4051 ทั้งนี้เพราะ ไมโครคอนโทรเลอร์กลุ่มนี้เป็นไมโครคอนโทรเลอร์ที่มีขนาดเล็กมาก คือ มีเพียง 20 ขา แต่สามารถทำงานได้ในระดับเดียวกับไมโครคอนโทรเลอร์ตัวใหญ่ โดยมีขีดจำกัดด้อยกว่าในบางเรื่องเท่านั้น

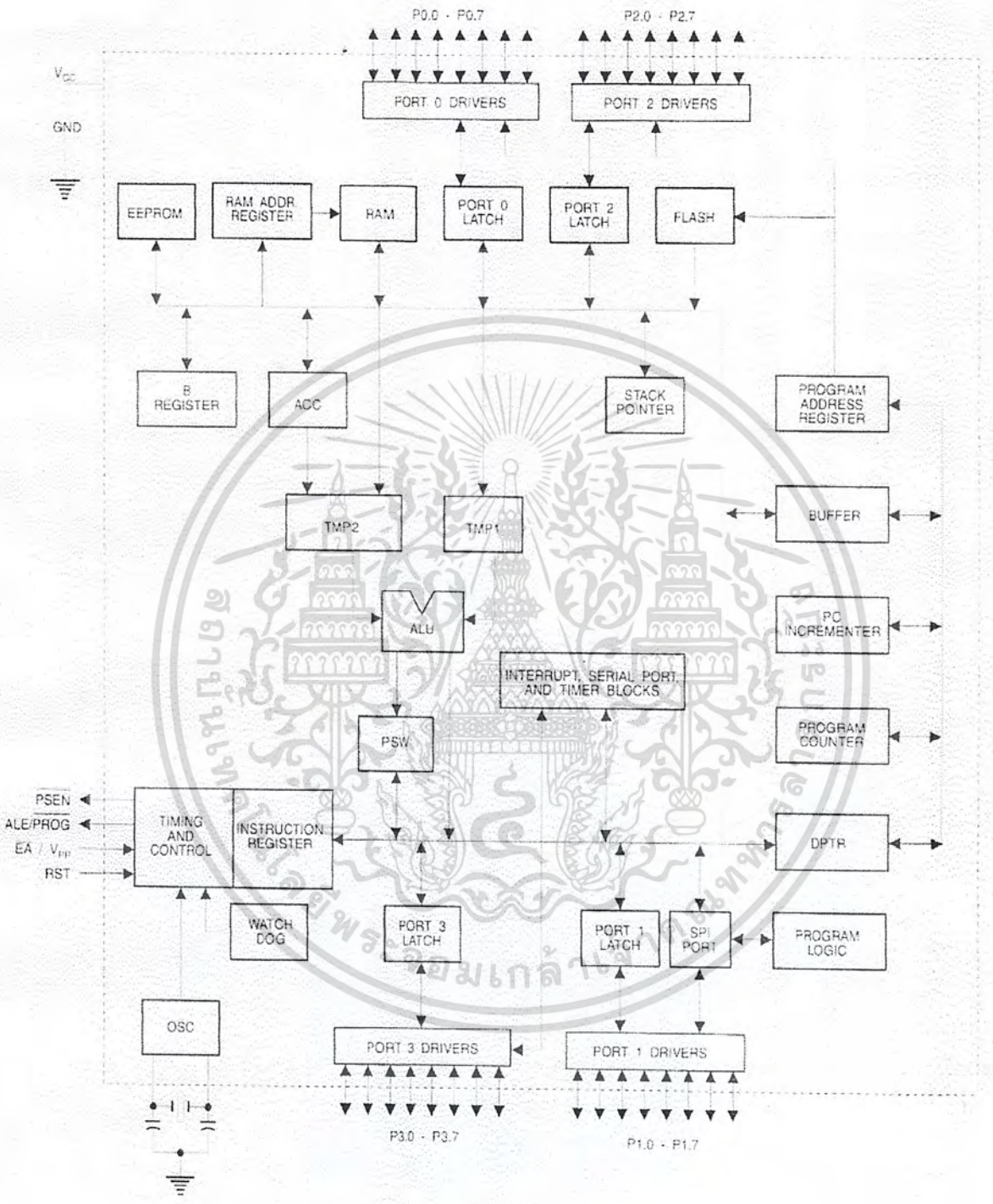
จากผังไดอะแกรมรูปที่ 2.5 และ 2.6 จะเห็นว่าปกติในตัวไมโครคอนโทรลเลอร์ตระกูล MCS-51 จะมีหน่วยความจำประเภทรอม (ROM) หรือแฟลช (Flash) ในตัว แต่อย่างใดก็ตามยังมีไมโครคอนโทรลเลอร์บางรุ่นที่ไม่มีหน่วยความจำนี้ อันได้แก่ไมโครคอนโทรลเลอร์เบอร์ 80C31, 80C32 แต่คุณสมบัติอื่นๆก็ยังเหมือนเดิมกับเบอร์อื่นทุกประการ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.5 แสดงผังไดอะแกรมของ MCS-51 พื้นฐาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.6 แสดงผังไดอะแกรมของ AT89S8252

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.6 โครงสร้างและการทำงานของพอร์ต

ไมโครคอนโทรลเลอร์ MCS-51 แบบแฟลชมีพอร์ตให้ใช้งานทั้งสิ้น 4 พอร์ต คือ พอร์ต 0 ถึง พอร์ต 3 แต่ละพอร์ตมีขนาด 8 บิต เป็นพอร์ตแบบ 2 ทิศทาง กล่าวคือ สามารถเป็นได้ทั้งอินพุต สำหรับรับสัญญาณข้อมูลเข้าและเอาต์พุตสำหรับส่งสัญญาณข้อมูลออก ทุกพอร์ตของไมโครคอนโทรลเลอร์ MCS-51 แบบแฟลชมีวงจรแลตช์และวงจรขับตลอดจนบัฟเฟอร์อินพุต ดังแสดงให้เห็นในตาราง

ขา	เบอร์ของไมโครคอนโทรลเลอร์	หน้าที่พิเศษ
P1.0	AT89C52/AT89Sxx	ขา T2 เป็นขาอินพุตนับค่าของไทเมอร์/เคาร์เตอร์ 2 และเป็นขาเอาต์พุตของการกำเนิดสัญญาณนาฬิกาโดยไทเมอร์ 2 (clock out)
P1.1	AT89C52/AT89Sxx	ขา T2EX เป็นขาอินพุตทริกเอร์สำหรับการแคปเจอร์/รีโหลดและควบคุมทิศทางของสัญญาณ
P1.4	AT89Sxx	ขา $\overline{SS}$ (Slave Select) เป็นขาเลือกการติดต่อในกรณีที่ไมโครคอนโทรลเลอร์เป็นอุปกรณ์สเลฟ ในระบบการติดต่อแบบ SPI
P1.5	AT89Sxx	ขา MOSI (Master data output, Slave date data input) ใช้ในการติดต่อกับพอร์ต SPI
P1.6	AT89Sxx	ขา MISO (Master data input, Slave date data input) ใช้ในการติดต่อกับพอร์ต SPI
P1.7	AT89Sxx	ขา SCK (Master clock output) เป็นขาสัญญาณนาฬิกาของการติดต่อกับพอร์ต SPI

ตารางที่ 2.5 แสดงหน้าที่พิเศษของพอร์ต 1 ในไมโครคอนโทรลเลอร์ MCS-51 แบบเฟลช

ที่พอร์ต 0 และพอร์ต 2 จะใช้งานเป็นพอร์ตอินพุตและเอาต์พุตสำหรับงานทั่วไป และใช้ในการติดต่อกับหน่วยความจำภายนอก สำหรับพอร์ต 3 ทั้งพอร์ตและพอร์ต 1 บางขานอกจากจะใช้เป็นขาพอร์ตอินพุตเอาต์พุตตามปกติแล้ว ยังสามารถใช้งานในหน้าที่พิเศษได้อีก ขึ้นอยู่กับว่าเป็นไมโครคอนโทรลเลอร์ MCS-51 แบบเฟลชเบอร์ใด ดังสรุปได้ในตารางที่ 2.5

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

วงจรภายในของแต่ละพอร์ตของไมโครคอนโทรลเลอร์ MCS-51 แบบเฟลชเป็นวงจรของพอร์ต 0 วงจรแลตช์ของแต่ละบิตในแต่ละพอร์ตก็คือ วงจรดีฟลอป นั่นเอง การอ่านค่าสถานะของพอร์ตและสถานะของวงจรแลตช์สามารถกระทำได้อย่างอิสระต่อกันด้วยสัญญาณที่แยกจากกัน นั่นคือ สัญญาณอ่านข้อมูลจากขาพอร์ตและสัญญาณอ่านข้อมูลจากวงจรแลตช์ส่วนการเขียนข้อมูลมายังพอร์ตต้องส่งสัญญาณมายังขา CLK ของดีฟลิฟลอปในขณะที่ข้อมูลจะส่งผ่านมาทางขาบัสข้อมูลภายในเข้าสู่ขา D ของดีฟลิฟลอบ

ที่พอร์ตนี้มีวงวจรมัลติเพล็กซ์สำหรับกำหนดลักษณะการทำงานของพอร์ตว่าต้องการใช้งานเป็นขาพอร์ตอินพุตเอาต์พุตปกติหรือใช้ในการติดต่อกับหน่วยความจำภายนอกไมโครคอนโทรลเลอร์

เนื่องจากที่ขาพอร์ต 0 ไม่มีวงจรพูลอัปภายในหากมีการนำพอร์ต 0 ไปใช้งานเป็นพอร์ตอินพุตจะต้องต่อตัวต้านทานพลูอัปภายนอกเข้าที่ขาพอร์ต 0 ทุกขาด้วย

2.7 การใช้งานเป็นพอร์ตอินพุต

เนื่องจากพอร์ตทั้งหมดของไมโครคอนโทรลเลอร์ MCS-51 แบบแฟลชสามารถเป็นได้ทั้งอินพุตและเอาต์พุต ดังนั้นจึงมีความจำเป็นอย่างยิ่งต้องทำความเข้าใจถึงการกำหนดลักษระการทำงานให้แก่พอร์ตของไมโครคอนโทรลเลอร์ MCS-51 แบบแฟลช

ในการกำหนดให้เป็นพอร์ตอินพุตต้องเริ่มต้นด้วยการเขียนข้อมูล “1” มาที่แต่ละบิตของพอร์ตที่ต้องการใช้งานเป็นอินพุต เพื่อหยุดการทำงานของเฟตที่ใช้ในการขับสัญญาณเอาต์พุตของบิตนั้นๆ ทำให้ขาสัญญาณของพอร์ตเชื่อมต่อเข้ากับวงจรพูลอัปภายในโดยตรง ส่งผลให้ขาพอร์ตนั้นมีลอจิกเป็น “1” สามารถรับสัญญาณลอจิก “0” จากอุปกรณ์ภายนอกได้ง่าย สัญญาณข้อมูลจากอุปกรณ์ภายนอกจะถูกส่งเข้ามาแล้วเก็บไว้ในวงจรบัฟเฟอร์ภายในพอร์ต แล้วรอให้ซีพียูมาอ่านค่าเข้าไป เมื่อเป็นเช่นนี้อุปกรณ์ภายนอกที่เชื่อมต่อกับพอร์ตอินพุตของไมโครคอนโทรลเลอร์ MCS-51 แบบแฟลชควรกำหนดให้ทำงานในสภาวะลอจิก “0” จะดีและสะดวกที่สุด (ซึ่งในปัจจุบันอุปกรณ์อินพุตที่เชื่อมต่อไมโครคอนโทรลเลอร์แทบทั้งหมดทำงานที่ลอจิก “0” แล้ว)

2.8 การใช้งานเป็นพอร์ตเอาต์พุต

โดยปกติแล้ว ขาพอร์ตจะกำหนดให้มีลักษณะเป็นเอาต์พุตอยู่แล้ว ดังนั้นจึงสามารถส่งข้อมูลออกไปได้อย่างง่ายดายและตรงไปตรงมากล่าวคือเมื่อต้องการส่งข้อมูล “0” ออกไปทางเอาต์พุตก็ให้เขียนข้อมูล “0” ไปยังวงจรแลตช์ ซึ่งก็จะส่งต่อไปขับเฟตทำงาน ที่ขาพอร์ตที่กำหนดให้ทำงานก็จะเกิดลอจิก “0” ขึ้น ในทางตรงข้ามหากต้องการส่งข้อมูล “1” ออกไป ก็ให้เขียนข้อมูล “1” ไปยัง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สำนักหอสมุดกลาง พระจอมเกล้าลาดกระบัง

วงจรแลตช์ วงจรขับก็จะหยุดทำงาน ทำให้ที่ขาพอร์ตเชื่อมต่อกับวงจรพูลอัปภายในเกิดเป็นลอจิก "1" ที่ขาพอร์ตนั้น ซึ่งจะคล้ายกับการกำหนดให้เป็นขาอินพุตมากเพียงแต่แตกต่างกันที่กระบวนการในการเคลื่อนย้ายข้อมูล โดยถ้าเป็นอินพุตจะมีสัญญาณมาอ่านข้อมูลที่บัฟเฟอร์ แต่ถ้าเป็นเอาต์พุตจะไม่มีการอ่านข้อมูลที่บัฟเฟอร์แต่อย่างใด เว้นแต่ในกรณีที่ต้องการตรวจสอบข้อมูลที่ส่งออกมาทางเอาต์พุต

เมื่อใช้งานพอร์ตของไมโครคอนโทรลเลอร์ MCS-51 แบบแฟลชเป็นพอร์ตเอาต์พุตแต่ละขา (หรือแต่ละบิต) ของแต่ละพอร์ตมีความสามารถในการจ่ายกระแสหรือที่เรียกว่า กระแสซอร์ส (Source current) ได้สูงสุด 10 mA และทุกขารวมกันในแต่ละพอร์ต (ทั้ง 8 บิต) สูงสุด 26 mA สำหรับพอร์ต 0 และ 15 mA สำหรับพอร์ต 1-3 ในกรณีที่ใช้งานทุกพอร์ตเอาต์พุตจะสามารถจ่ายกระแสได้รวมกันสูงสุด 71 mA ดังนั้นในการใช้งานเป็นพอร์ตเอาต์พุตเพื่อไม่ให้เกิดปัญหาเกี่ยวกับความสามารถในการจ่ายกระแสจึงควรต่อวงจรบัฟเฟอร์ทางเอาต์พุตเพื่อช่วยในการขับกระแสอีกทางหนึ่ง

2.9 การอ่านค่าลอจิกจากพอร์ต

ในไมโครคอนโทรลเลอร์ MCS-51 แบบแฟลชสามารถอ่านค่าลอจิกจากพอร์ตได้ 2ลักษณะ คือ อ่านจากขาพอร์ตโดยตรง และอ่านจากวงจรแลตซ์ของแต่ละพอร์ต

ในกรณีที่พอร์ตต่อกับขาเบสทรานซิสเตอร์ชนิด NPN และขาอิมิตเตอร์ของทรานซิสเตอร์ตัวนั้นต่อลงกราวด์ หากมีการส่งข้อมูล "1" ไปยังทรานซิสเตอร์ จะทำให้ทรานซิสเตอร์ทำงานสถานะลอจิกที่ขาพอร์ตจะเป็น "0" เนื่องจากเมื่อทรานซิสเตอร์ทำงาน จะเสมือนว่าขาพอร์ตนั้นถูกต่อลงกราวด์ทำให้หากอ่านค่าลอจิกที่ขาพอร์ตจะได้ผลตรงข้ามกับที่ส่งออกมา แต่ถ้าหากทำงานอ่านค่าลอจิกที่วงจรแลตช์จะได้ค่าที่ตรงกับค่าที่ต้องการส่งจริง ดังนั้นในการอ่านค่าลอจิกพอร์ตจึงต้องเลือกวิธีการให้เหมาะสมกับอุปกรณ์ที่นำมาต่อด้วย

2.10 หน่วยความจำโปรแกรม (Program memory)

การจัดหน่วยความจำโปรแกรมของไมโครคอนโทรลเลอร์ MCS-51 แบบแฟลชในเบอร์ต่างๆ ที่นิยมใช้งาน อันประกอบด้วยเบอร์ AT89X51 และ AT89C52 จะเห็นได้ว่าทั้งสองเบอร์สามารถติดต่อหน่วยความจำโปรแกรมได้สูงสุด 64 กิโลไบต์ โดยสามารถเลือกใช้หน่วยความจำโปรแกรมภายในอย่างเดียวหรือรวมกับภายนอกหรือเลือกใช้หน่วยความจำภายนอกอย่างเดียวก็ได้ โดยภายใน AT89C51 จะมีหน่วยความจำโปรแกรมภายใน 4 กิโลไบต์ ในขณะที่ AT89C52 จะมีขนาด 8 กิโลไบต์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

42240

ในกรณีที่ใช้หน่วยความจำภายในและภายนอกรวมกัน หากใช้ AT89C51 ก็สามารถติดต่อกับหน่วยความจำภายนอกได้ 60 กิโลไบต์ และถ้าใช้เบอร์ AT89C52 จะสามารถติดต่อกับหน่วยความจำโปรแกรมภายนอกได้ 56 กิโลไบต์

หน่วยความจำโปรแกรมมีแอดเดรสเริ่มต้นที่ 0000H เมื่อซีพียูได้รับการรีเซตให้เริ่มต้นการทำงานจะต้องมาเริ่มต้นที่แอดเดรส 0000H นี้เสมอ อย่างไรก็ตามในพื้นที่ของหน่วยความโปรแกรมไม่ว่าจะใช้งานจากภายในหรือภายนอกก็ตาม ต้องมีการสงวนพื้นที่บางตำแหน่งเอาไว้สำหรับการบริการอินเตอร์รัปต์ 6 ประเภท ประเภทละ 8 ไบต์ ประกอบด้วย

พื้นที่สำหรับบริการอินเตอร์รัปต์ 0 จากภายนอก กำหนดไว้ที่แอดเดรส 0003H
พื้นที่สำหรับบริการอินเตอร์รัปต์จากไทเมอร์ 0 กำหนดไว้ที่แอดเดรส 000BH
พื้นที่สำหรับบริการอินเตอร์รัปต์ 1 จากภายนอก กำหนดไว้ที่แอดเดรส 0013H
พื้นที่สำหรับบริการอินเตอร์รัปต์จากไทเมอร์ 1 กำหนดไว้ที่แอดเดรส 001BH
พื้นที่สำหรับบริการอินเตอร์รัปต์ของการสื่อสารอนุกรม กำหนดไว้ที่แอดเดรส 000BH
พื้นที่สำหรับบริการอินเตอร์รัปต์จากไทเมอร์ 2 กำหนดไว้ที่แอดเดรส 002BH

2.11 หน่วยความจำข้อมูล (Data memory)

มีด้วยกัน 2 แบบ คือ หน่วยความจำข้อมูลภายนอกและภายใน โดยไมโครคอนโทรลเลอร์ MCS-51 แบบแฟลชในอนุกรม AT89 สามารถติดต่อกับหน่วยความจำข้อมูลภายนอกได้สูงสุด 64 กิโลไบต์ โดยการใช้คำสั่ง MOVX ในการติดต่อกับหน่วยความจำข้อมูลภายนอก การติดต่อกับหน่วยความจำข้อมูลภายนอกของไมโครคอนโทรลเลอร์ MCS-51 แบบแฟลช จะเห็นได้ว่า มีลักษณะคล้ายกับการติดต่อกับหน่วยความโปรแกรมภายนอกแตกต่างกันที่มีสัญญาณที่ใช้สำหรับการอ่านและเขียนหน่วยความจำข้อมูลภายนอก นั่นคือขา $\overline{RD}$ และ $\overline{WR}$

สำหรับ ไมโครคอนโทรลเลอร์ MCS-51 แบบแฟลชในอนุกรม AT89 ทุกเบอร์จะมีหน่วยความจำข้อมูลภายในเป็นแบบแรม (RAM : Random Access Memory) โดยแต่ละเบอร์จะมีขนาดแตกต่างกันไป ในเบอร์ AT89C51 มีหน่วยความจำข้อมูลภายในขนาด 128 ไบต์ ในขณะที่เบอร์ AT89C-52 มีขนาด 256 ไบต์ สำหรับการจัดสรรความจำข้อมูลภายในแบ่งเป็น 3 ส่วน คือ หน่วยความจำข้อมูลส่วนล่าง (lower), ส่วนบน (upper) และรีจิสเตอร์ฟังก์ชั่นพิเศษ (SFR : Special Function Register) แต่ละส่วนมีขนาด 128 ไบต์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.12 รีจิสเตอร์ฟังก์ชั่นพิเศษ (Special Function Register : SFR)

เป็นรีจิสเตอร์ที่ใช้ควบคุมการทำงานของไมโครคอนโทรลเลอร์ MCS-51 แบบแฟชลทั้งหมดมีด้วยกัน 22 ตัว สำหรับในไมโครคอนโทรลเลอร์ MCS-51 แบบแฟลชเบอร์ AT89C51 และ 28 ตัวในไมโครคอนโทรลเลอร์ MCS-51 แบบแฟลชเบอร์ AT89C92 และอนุกรม AT98Sxx ทั้งนี้เนื่องจากใน AT89C52 และ AT89Sxx มีจำนวนไทเมอร์เคาน์เตอร์มากกว่า AT89C51

รีจิสเตอร์ SFR มีแอดเดรสอยู่ระหว่าง 80H-FFH ในพื้นที่ของหน่วยความจำข้อมูลส่วนบน สามารถเข้าถึงได้โดยตรง (Direct addressing) แสดงการจัดสรรพื้นที่ของรีจิสเตอร์ SFR แต่ละตัวในหน่วยความจำข้อมูลส่วนบน สำหรับรายละเอียดเบื้องต้นของรีจิสเตอร์ SFR มีดังนี้

บิต7	บิต6	บิต5	บิต4	บิต3	บิต2	บิต1	บิต0
CY	AC	F0	RS1	RS0	OV	-	P

ตารางที่ 2.6 แสดงการจัดพื้นที่ของรีจีสเตอร์ SFR แต่ละตัวในหน่วยความจำข้อมูลส่วนบน

CY : แฟลกทด (Carry flag) เป็น "1" เมื่อมีการกระทำคำสั่งทางคณิตศาสตร์และลอจิก แล้วค่าของแอกคิวมูลเลเตอร์เกิน 255 (ฐานสิบ) หรือ FFH

AC : แฟลกทดเสริมเป็น (Auxiliary Carry flay) "1" เมื่อมีการกระทำคำสั่งทางคณิตศาสตร์แล้วทำให้เกิดการทดข้ามจากบิต 3 มายังบิต 4 มักใช้ในการ แปลงค่าเป็นเลขฐานสิบ (BCD operation)

FO : แฟลกใช้งานทั่วไป เมื่อผู้เขียนโปรแกรมกำหนดค่าที่บิตนี้แล้ว มไว่าจะกระทำคำสั่งใดๆ ที่บิตนี้จะไม่มีการเปลี่ยนแปลง

RS1 : บิตเลือกรีจิสเตอร์แบงก์ (Register Select1) ใช้งานร่วมกับบิต RS0 เพื่อเลือกแบงก์ของรีจิสเตอร์ R0-R7

RS0 : บิตเลือกรีจิสเตอร์แบงก์ (Register Select0) ใช้งานร่วมกับบิต RS1 เพื่อเลือกแบงก์ของรีจิสเตอร์ R0-R7

OV : บิตเกิน (Overflow) เป็น "1" เมื่อมีการกระทำคำสั่งทางคณิตศาสตร์และลอจิกแล้ว ทำให้เกิดการทดข้ามจากบิต 7 ของแอกคิวมูเลเตอร์ หรือแอกคิวมูเลเตอร์มีค่าเกิน 127 (ฐานสิบ)นอกจากนั้นยังใช้เป็นการแสดงค่าลบด้วย

- : บิตนี้ผู้ใช้งานสามารถกำหนดใช้งานได้อย่างอิสระ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

P : บิตพาริตี้ (Parity) ใช้ในการตรวจสอบจำนวนค่า "1" ภายในแอกคิวมูเลเตอร์ ถ้าหากในแอกคิวมูเลเตอร์มีจำนวนบิตที่เป็น "1" รวมกันเป็นเลขคู่ บิตนี้จะเป็น "1"

RS1	RS0	แบงก์ของรีจิสเตอร์	ช่วงแอดเดรส
0	0	แบงก์ 0	00H-07H
0	1	แบงก์ 1	08H-0FH
1	0	แบงก์ 2	10H-17H
1	0	แบงก์ 3	18H-1FH

ตารางที่ 2.7 แสดงการเลือกแบงก์ของหน่วยความจำส่วนล่างเพื่อติดต่อกับรีจิสเตอร์แบงก์ R0-R7

2.13 รีจิสเตอร์แสดงสถานะของโปรแกรม (Program Status Word : PSW)

เป็นรีจิสเตอร์ขนาด 8 บิต สามารถเข้าถึงได้ในระดับบิต นั่นหมายความว่า สามารถกระทำคำสั่งหรือกำหนดค่าในแต่ละบิตของรีจิสเตอร์ตัวนี้ได้อย่างอิสระ มีแอดเดรสอยู่ที่ D0H เป็นรีจิสเตอร์ที่เก็บสถานะของการทำงานของโปรแกรมในขณะนั้น จะเรียกสถานะต่างๆ ของโปรแกรมว่า แฟลก (flag) เมื่อซีพียูกระทำคำสั่งทางคณิตศาสตร์ และลอจิกแล้วเกิดการเปลี่ยนแปลงสถานะขึ้น ผลของการเปลี่ยนแปลงนั้นจะมาปรากฏที่บิตต่างๆ ของรีจิสเตอร์ PSW รายละเอียดของแต่ละบิตในรีจิสเตอร์ PSW

จะเห็นได้ว่านอกจากรีจิสเตอร์ PSW ถูกใช้ในการเก็บสถานะของโปรแกรมแล้วที่บิต RS0 และ RS1 ยังใช้ในการเลือกแบงก์ของหน่วยความจำส่วนล่าง ซึ่งเป็นพื้นที่ของรีจิสเตอร์ R0-R7 ด้วย ดังมีรายละเอียดแสดงในตาราง โดยปกติแล้วในการใช้งานรีจิสเตอร์ R0-R7 มักนิยมเลือกใช้แบงก์ 0 เป็นลำดับแรก หากไม่เพียงพอจึงเลือกในแบงก์อื่นๆ มาใช่แต่ต้องระมัดระวังในการกำหนดค่าและลำดับการติดต่อให้ดี มิเช่นนั้นอาจทำให้การเขียนโปรแกรมเกิดความสับสน ดังนั้นสำหรับผู้เริ่มต้นใช้งานไมโครคอนโทรลเลอร์ MCS-51 จึงควรเลือกใช้รีจิสเตอร์ R0-R7 ในแบงก์ 0 เพียงแบงก์เดียวให้ชำนาญเสียก่อน

การกำหนดค่าของรีจิสเตอร์ PWS เพื่อเลือกใช้งานรีจิสเตอร์ R0-R7 ควรกำหนดไว้ที่ตอนต้นของโปรแกรมเสมอ เพื่อจะได้เขียนโปรแกรมติดต่อกับรีจิสเตอร์ R0-R7 ได้อย่างสะดวกและไม่เกิดความผิดพลาด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 3

การทำงานของหุ่นยนต์ MOUSE ROBOT

ส่วนประกอบทางด้านอิเล็กทรอนิกส์ของหุ่นยนต์ MOUSE ROBOT มี 4 ส่วน ดังนี้

1. ส่วนประกอบตรวจจับประกอบไปด้วยแผงวงจรตรวจจับ 2 รูปแบบคือ แผงวงจรตรวจจับแสง, แผงวงจรตรวจจับความต่างของสีสำหรับตรวจจับเส้น
2. ส่วนขับเอาท์พุต ประกอบไปด้วยวงจรขับสเต็ปเปอร์มอเตอร์
3. ส่วนภาคควบคุม ซึ่งถือว่าเป็นหัวใจหลักในการทำงานของหุ่นยนต์ใช้ไมโครคอนโทรลเลอร์ในการควบคุมการทำงาน
4. ภาคจ่ายไฟ

3.1 โฟโต้ทรานซิสเตอร์

โฟโต้ทรานซิสเตอร์ (Photo Transistor) มีโครงสร้างแตกต่างจากโฟโต้ไดโอด คือบริเวณที่รับแสงคือรอยต่อพี-เอ็นระหว่างเบสกับคอลเลกเตอร์ (B-C) และกระแสไหลผ่านทรานซิสเตอร์ระหว่างขั้วอิมิเตอร์กับคอลเลกเตอร์ โฟโต้ทรานซิสเตอร์นั้นเปรียบเสมือนสวิตช์แสง (Light Switch) เพราะเมื่อมีแสงมาตกกระทบรอยต่อระหว่าง B-C จะทำให้ทรานซิสเตอร์ทำงานได้ มีกระแสคอลแลกเตอร์ (I_C) ตามสมการ

$$I_C = \beta_{dc} \times I_\lambda$$

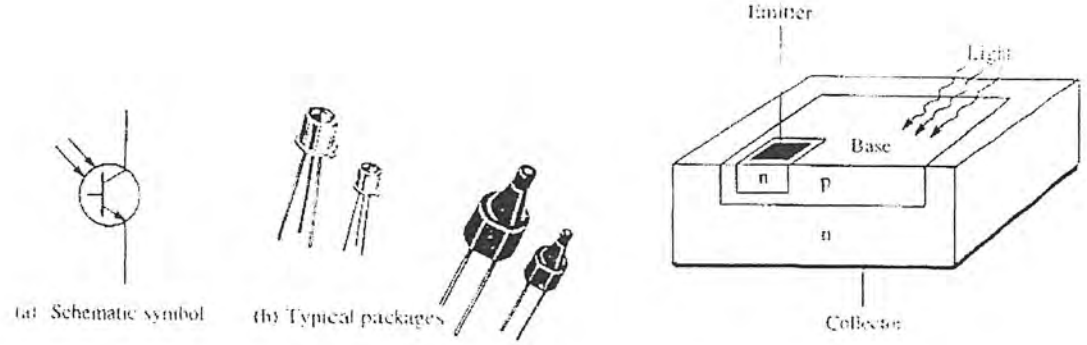
เมื่อ I_C = กระแสคอลเลกเตอร์

β_{dc} = อัตราการขยายกระแสของโฟโต้ทรานซิสเตอร์

I_λ = กระแสเบสที่เกิดจากแสงสว่างตกกระทบรอยต่อเบส-คอลเลกเตอร์

ลักษณะโครงสร้างและสัญลักษณ์ของโฟโต้ทรานซิสเตอร์ แสดงในรูปที่ 3.1 (a),(b), (c)

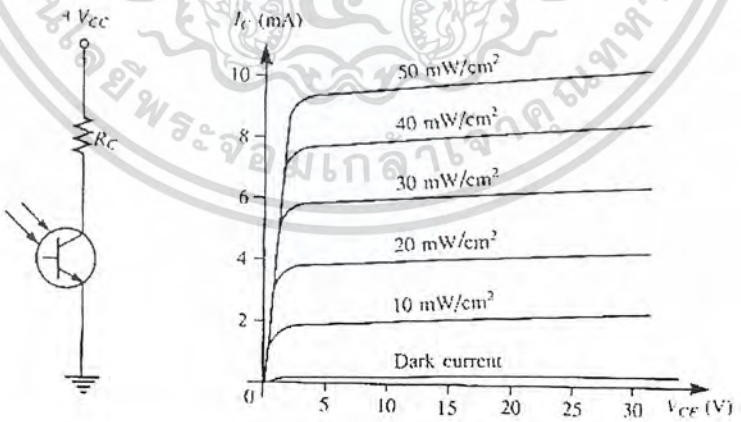
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.1 แสดงโฟโต้ทรานซิสเตอร์

3.2 การทำงานของโฟโต้ทรานซิสเตอร์ (Photo Transistor Operation) เมื่อไบแอสโฟโต้ทรานซิสเตอร์ให้ถูกต้อง โดยให้ไบแอสแรงดันระหว่างขั้วอิมิเตอร์และคอลเลกเตอร์ สำหรับเบสไม่ต้องไบแอส ดังรูปที่ 3.2 ขณะนี้ถ้าไม่มีแสงตกกระทบรอยต่อของเบส-คอลเลกเตอร์ ทรานซิสเตอร์จะไม่ทำงานและจะไม่มีกระแสไหลผ่าน Pc แต่จะมีกระแสรั่วไหลระหว่างรอยต่อคอลเลกเตอร์และอิมิเตอร์จำนวนหนึ่ง(uA)เรียกว่า Dark Current

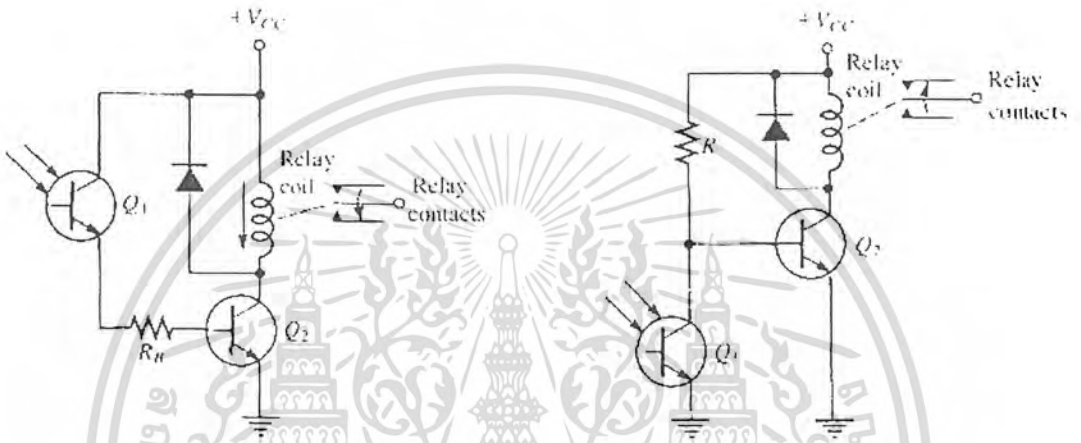
เมื่อให้แสงที่มีความสว่างเล็กน้อยที่รอยต่อเบส-คอลเลกเตอร์จะทำให้กระแส Ic ไหลได้ตามสมการและถ้าเพิ่มความสว่างของแสงให้มากขึ้น ค่ากระแส Ic จะสูงขึ้นตามไปด้วยดังกราฟลักษณะสมบัติของโฟโต้ทรานซิสเตอร์ในรูปที่ 3.2



รูปที่ 3.2 แสดงวงจรไบแอสและกราฟลักษณะสมบัติของโฟโต้ทรานซิสเตอร์

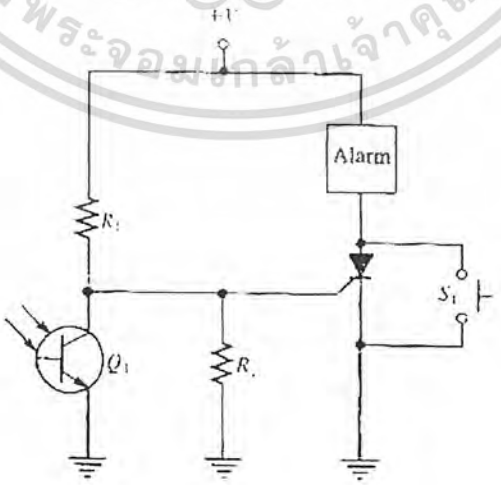
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.3 การประยุกต์ใช้งานโฟโต้ทรานซิสเตอร์(Photo Transistor Application)นิยมนำโฟโต้ทรานซิสเตอร์ไปใช้เป็นสวิตช์แสงหรือตัวตรวจจับแสง(Lighting Sensor) เมื่อแสงมาตกกระทบที่ตัวโฟโต้ทรานซิสเตอร์บริเวณเลนส์ด้านบน จะทำให้ทรานซิสเตอร์ทำงานได้ในลักษณะเปิด-ปิด (On-Off Control) สามารถควบคุมการเปิด-ปิดของรีเลย์หรืออุปกรณ์ อื่นๆได้ ดังแสดงในรูปที่ 3.3 (a) ,(b)



รูปที่ 3.3 แสดงการนำโฟโต้ทรานซิสเตอร์ไปเปิด-ปิด ควบคุมการทำงานของรีเลย์

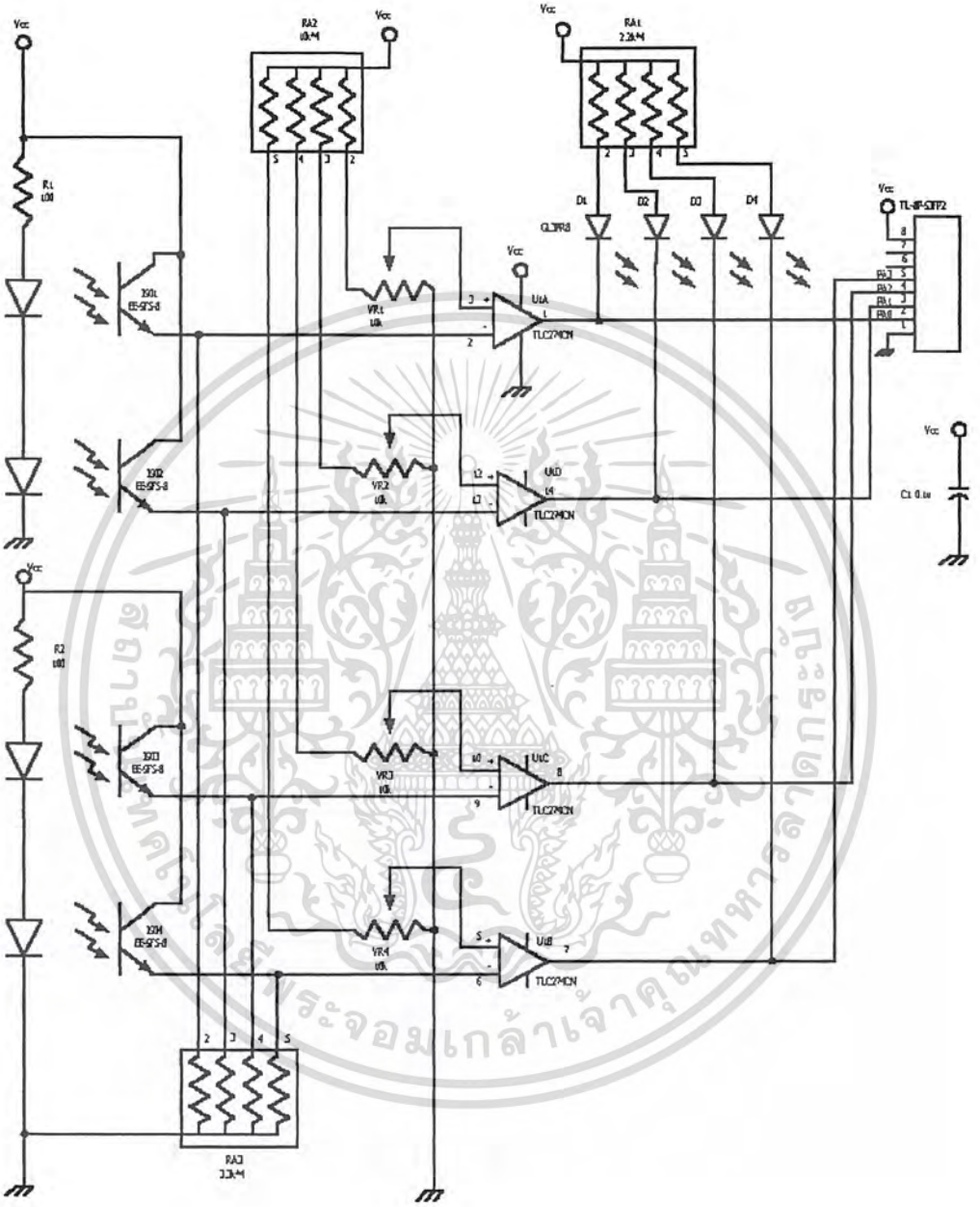
หรือวงจรที่แสดงในรูปที่ 3.3.1 คือวงจรส่งสัญญาณเตือนภัย(Alarm) เมื่อมีแสงมาตกกระทบทรานซิสเตอร์ Q1 จะทำให้ออดไฟฟ้าดัง โดยใช้เอสซีอาร์เป็นตัวควบคุมออดไฟฟ้า เป็นต้น



รูปวงจรที่ 3.3.1 วงจร Light-Interruption Alarm ที่ใช้โฟโต้ทรานซิสเตอร์ควบคุมเอสซีอาร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.4 วงจร ตรวจจับทิศทางการทำงานตามเส้นและความต่างสี



รูปที่ 3.4 รูปวงจรแสดงการทำงานของตัวตรวจจับการทำงานตามเส้นและความต่างของเส้นสี

จากรูปเป็นวงจรตรวจจับทิศทางการทำงานของตัว Sensor ตามเส้นสีและตามช่องทาง ในลักษณะการตรวจจับของวงจรคือเมื่อเราทำการจ่ายไฟบวก 5 โวลต์ เมื่อมีกระแสไหลในวงจรทำให้โฟโต้ทรานซิสเตอร์ติดสว่างพร้อมที่จะทำการ Sensor เมื่อ Sensor อยู่ในตำแหน่งเส้นที่จะทำการ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Sensor ได้คือเมื่อมีแสงอินฟาเรทของโฟโต้ทรานซิสเตอร์ส่องไปยังพื้นเส้นที่เรากำหนดไว้ ในการทำโครงงานนี้ได้กำหนดให้ Sensor ทำงานตามพื้นสีขาว Sensor จะมีการทำงานโดย เมื่อแสงอินฟาเรทของโฟโต้ทรานซิสเตอร์ไปตกกระทบที่พื้นแล้วสะท้อนกลับมายัง Sensor ก้จะทำการตรวจจับทิศทางการเคลื่อนที่ของหุ่นว่าจะเคลื่อนที่ในลักษณะใด ที่ Sensor จะทำงานจะส่งลอจิก 0 ไปเข้าที่ขา IC TLC274CN ซึ่งเป็น IC CN เข้าที่ขาลบ IC ตัวนี้เป็น IC Op-amp ทำหน้าที่เปรียบเทียบแรงดัน ซึ่งได้ตั้งค่าแรงดันอ้างอิงไว้ที่ขาบวกของ Op-amp จึงเกิดการทำงานเปรียบเทียบแรงดันออกมาเป็นลอจิก 1 เป็น Output ไปเข้า Input ของ Controller เพื่อทำการประมวลผลทางโปรแกรมสั่งงานต่อไป ซึ่งในการทำงานข้างต้นจะมีตัว Sensor ของการทำงานตามเส้น 1ชุด และการทำงานตามช่องทางอีก 2 ชุดโดยแต่ละชุดจะมี Sensor อย่างละ4 ตัว โดยผลการทำงานจะแสดงออกให้รู้ว่า Sensor ที่ทำงานโดย LED ทั้ง 4 ตัวในวงจรข้างต้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 4

สรุปผลการทดลองและแนวทางแก้ไขปัญหา

สรุปผลการทำงานของหุ่นยนต์และโปรแกรมควบคุมการทำงาน

1.การเลี้ยวของหุ่นยนต์ซึ่งออกแบบให้มีความละเอียดในการเลี้ยวโดย Sensors จะเป็นตัวกำหนดทิศทางการเดินทาง ยังมีข้อผิดพลาด (error) เล็กน้อย ซึ่งสาเหตุเกิดจากระยะห่างระหว่างล้อไม่ถูกต้องเนื่องจากขาดเครื่องมือวัดที่มีความเที่ยงตรงและแน่นอนในการสร้าง

2.ระยะทางที่หุ่นยนต์เคลื่อนที่ยังเกิด error ขึ้นเล็กน้อย เนื่องจากโปรแกรมซึ่งคำนวนระยะการเคลื่อนที่ของหุ่นยนต์ซึ่งนำค่าระยะทางจริงมาหารกับระยะทางที่หุ่นยนต์เคลื่อนที่ได้ 1 สเต็ป มีการปัดทศนิยมเป็นจำนวนเต็มและการคำนวนหามุมเลี้ยวก็มีการปัดขึ้นเป็นจำนวนเต็มเช่นเดียวกัน ถึงแม้ว่าเราจะใช้สเต็ปเปอร์มอเตอร์ที่มีจำนวนสเต็ปมาก ซึ่งจะทำให้ระยะทางการเคลื่อนที่ต่อ1 สเต็ปน้อยลง แต่เมื่อหุ่นยนต์เคลื่อนที่ซ้ำๆกันหลายรอบ จึงมี error เกิดขึ้น

การแก้ไขข้อผิดพลาดนี้จะทำโดยนำวงจรตรวจจับความต่างสี(Tracking line on the floor) มาเช็คจุดสีที่พื้นที่จุดเริ่มต้นการเคลื่อนที่ของหุ่นยนต์ นำมาเป็นจุดอ้างอิง (reference) ถ้าไม่ตรงจุดจะทำให้ทราบว่ามีการเกิด error ขึ้น

3.การส่งข้อมูลเพื่อแสดงพิกัดตำแหน่ง ซึ่งทำการส่งทุก 1 สเต็ปการเคลื่อนที่ของหุ่นยนต์จะทำให้การเกิด error ของการเคลื่อนที่ลดลงแต่การรับส่งข้อมูลทุกๆ 1 สเต็ป จะทำไม่สามารถเพิ่มความเร็วในการเคลื่อนที่เพราะต้องเสียเวลาในการรับส่งข้อมูล ดังนั้นถ้าต้องการความเร็วในการเคลื่อนที่สูงต้องเพิ่มอัตราการรับส่งข้อมูล(baud rate)ให้สูงขึ้น

4.เมื่อหุ่นยนต์หยุดทำงานจำเป็นต้องจ่ายกระแสให้สเต็ปเปอร์มอเตอร์ตลอดเพื่อรักษาตำแหน่งของหุ่นยนต์ให้คงที่ เพราะว่าถ้าไม่มีกระแสให้สเต็ปเปอร์มอเตอร์เมื่อหุ่นยนต์เริ่มเคลื่อนที่ทิศทางของหุ่นยนต์จะเปลี่ยนไป ซึ่งการจ่ายกระแสให้สเต็ปเปอร์มอเตอร์ตอนหุ่นยนต์หยุดทำงานจะกินกระแสมากกว่าตอนทำงาน

5.สเต็ปเปอร์มอเตอร์สามารถกำหนดตำแหน่งการหมุนได้อย่างแม่นยำโดยไม่ต้องใช้วงจรป้อนกลับ แต่ข้อเสียของมอเตอร์นี้คือ มีขนานใหญ่ น้ำหนักมาก แรงบิดน้อย จะใช้พลังงานมากกรณีต้องการแรงบิดสูง

6. ค่า Turn error เกิดจากการสร้างหุ่นยนต์ทางแมคคานิกส์ไม่ได้ตามที่ออกแบบไว้ ซึ่งทำให้มุมเลี้ยวผิดพลาดไม่ได้ตามที่กำหนดซึ่งเป็นค่าที่เราไม่ต้องการ ค่า Turn error หาได้จากจำนวนสเต็ปในการหมุนของหุ่นยนต์จริงหารด้วยจำนวนสเต็มที่โปรแกรมคำนวนออกมา ซึ่งเราต้องการให้ค่าทั้งสองนี้เท่ากัน คือ ค่า Turn error เท่ากับ 1 จะทำให้การเลี้ยวไม่มีข้อผิดพลาด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

7.จากการ Sensor แบบตามเส้นจะเกิดการ Error เมื่อ Sensorไม่สามารถหาค่าระยะทางที่กำหนดได้เนื่องจากการทำงานของสเต็ปมอเตอร์ในการเดินหาทิศทางเกิดการผิดพลาดและยังเกิดการผิดพลาดเนื่องจากการผิดพลาดเนื่องจาก Sensor ที่ ทำงานไม่ละเอียดพอ

8.เกิดการผิดพลาดเนื่องจากพื้น ของรถวิ่งไม่เท่ากันเนื่องจากเกิดการผิดพลาดของการวัดในการทำพื้นวิ่ง

แนวทางการพัฒนาโครงงาน

แนวทางการพัฒนาโปรแกรมควบคุม

1.ควรพัฒนาให้โปรแกรมสามารถควบคุมให้หุ่นยนต์ได้อย่างน้อย 2 ตัวขึ้นไป เพราะในระบบการขนส่งจริงจะใช้หุ่นยนต์หลายตัวทำงานร่วมกันซึ่งจะทำให้ระบบมีประสิทธิภาพและทำให้เกิดความรวดเร็ว

2. ควรพัฒนาโปรแกรมให้หาระยะทางการเคลื่อนที่ของหุ่นยนต์จากแผนที่ของระบบขนส่งจริง โดยมีการกำหนดมาตราส่วนและสัญลักษณ์ เช่น รูปร่าง ลักษณะ สี เป็นต้น ทางแผนที่

แนวทางการพัฒนาหุ่นยนต์

1.ควรพัฒนาให้หุ่นยนต์หาค่าพิกัดตำแหน่งและทิศทางด้วยตนเองเพื่อให้สถานีควบคุมสามารถทำงานได้รวดเร็วขึ้น

2. ควรพัฒนาหุ่นยนต์ให้สามารถเคลื่อนที่ในทางขรุขระได้

3. ควรพัฒนาให้หุ่นยนต์รับน้ำหนักของสิ่งของได้

4.ควรพัฒนาให้หุ่นยนต์หาจุดเริ่มต้นซึ่งใช้เป็นจุด Reference โดยอาศัยวงจรตรวจจับสีที่พื้น หรือกล้องมอนิเตอร์

5.การพัฒนาขั้นต่อไปควรเป็นแบบไร้สายเพื่อง่ายต่อการควบคุมในระยะที่ต้องการได้ดี

ปัญหาและอุปสรรค์

1.วัสดุ อุปกรณ์ที่ใช้สร้างหุ่นยนต์ เช่น ล้อของหุ่นยนต์ หาซื้อยากและราคาแพง

2.ขาดเครื่องมือที่มีประสิทธิภาพสำหรับการสร้างหุ่นยนต์ เช่น เครื่องเจาะ เครื่องกลึง เครื่องมือวัดต่างๆ

3.การปรับแต่งระยะห่างระหว่างล้อของหุ่นยนต์เพื่อให้ได้มุมเลี้ยวรอบตัว 360 องศา ทำได้ยากและต้องทำขั้นตอนนี้หลายครั้ง

4.การซื้อไม้ในการทำหุ่นควรเป็นไม้ที่หนาพอสมควรเพื่อไม่ให้พื้นโค้งงอได้ง่าย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บรรณานุกรม

ชัยวัฒน์ ลิ้มพรจิตรวิไล และคณะ.2542. เรียนรู้และปฏิบัติไมโครคอนโทรเลอร์ MCS-51. กรุงเทพ: บริษัท อิโนเวตีฟ เอ็กเพอริเมนต์ จำกัด

ศุภชัย บุศราทิจ.2543. คู่มือภาษาซีสำหรับไมโครคอนโทรเลอร์ MCS-51. กรุงเทพ: บริบัท อีทีที จำกัด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ผนวก ก. โปรแกรมการทำงานของหุ่นยนต์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

อุปกรณ์ที่ใช้ในโปรแกรม

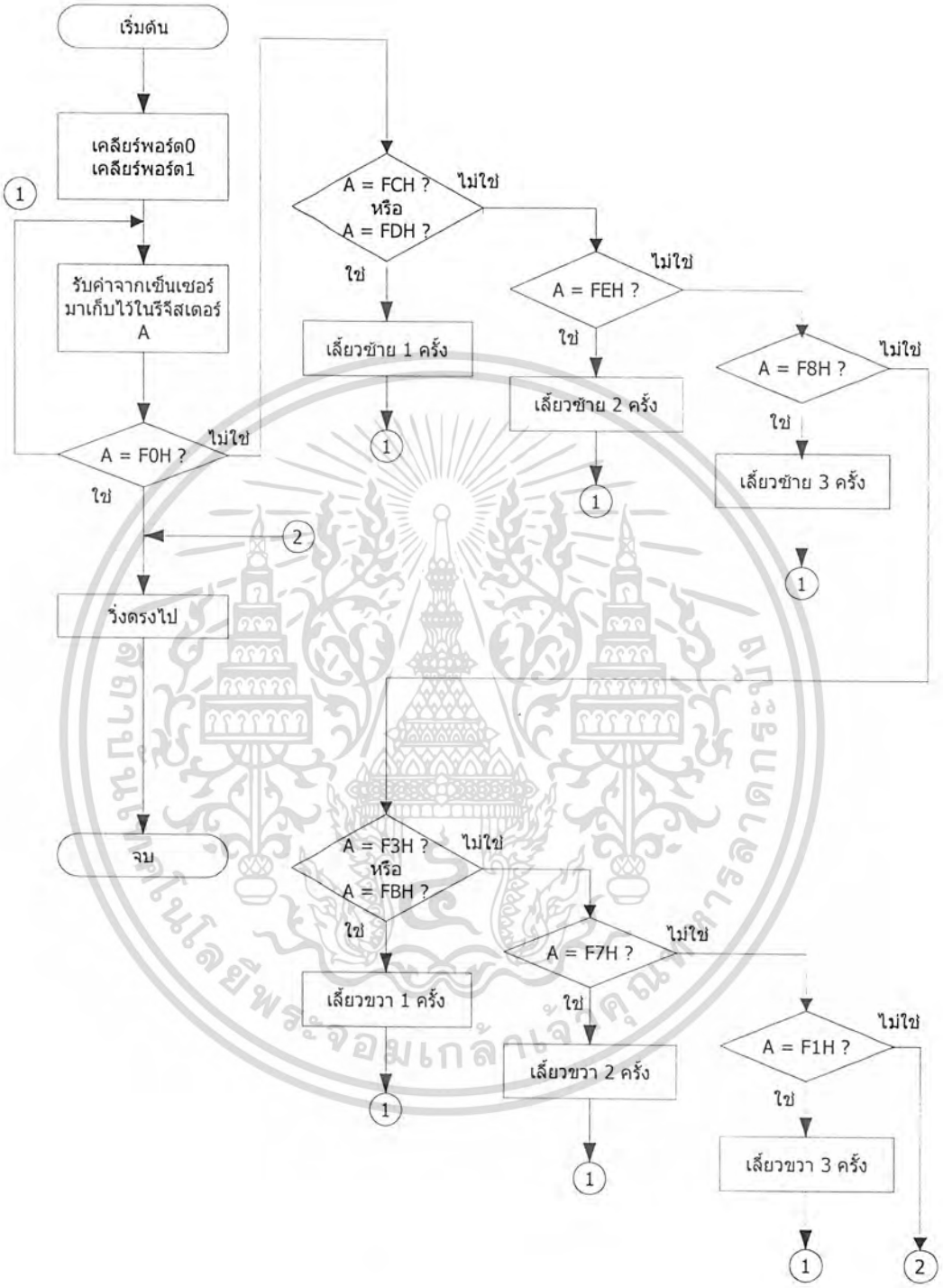
ส่วนของตัวหุ่นยนต์

1. Microcontroller with 8K Bytes Flash เบอร์ AT89S8252 CMOS 8 Bit 40 ขา	1 ตัว
2. ULN2803A 18 ขา	1 ตัว
3. Stepping Motor 12 Volt กระแส 0.16 แอมป์ Step Angle 1.8 DEG.	1 ตัว
4. Battery SL6-1.3 (6 V 1.3 AH/20HR)	2 ตัว
5. LED 3ชุด ชุดละ	4 ตัว
6. ความต้านทานปรับค่าได้ 10 KB 4ชุดละ	4 ตัว
7. RPACK 10K*4 จำนวน 3ชุด ชุดละ	3 ตัว
8. IC TLC274CN 14 ขา	1 ตัว
9. Socket 12 ขา	1 ตัว
10. Socket 6 ขา	1 ตัว
11. Socket 5 ขา	1 ตัว
13. Socket 2 ขา	1 ตัว
14. Switch 6 ขา	2 ตัว
15. Sensor 4 ชุด ชุดละ	4 ตัว
16. แผงวงจรการเดินของรถ 120*120 นิ้ว	1 ชุด

ส่วนของประกอบของ Sensor

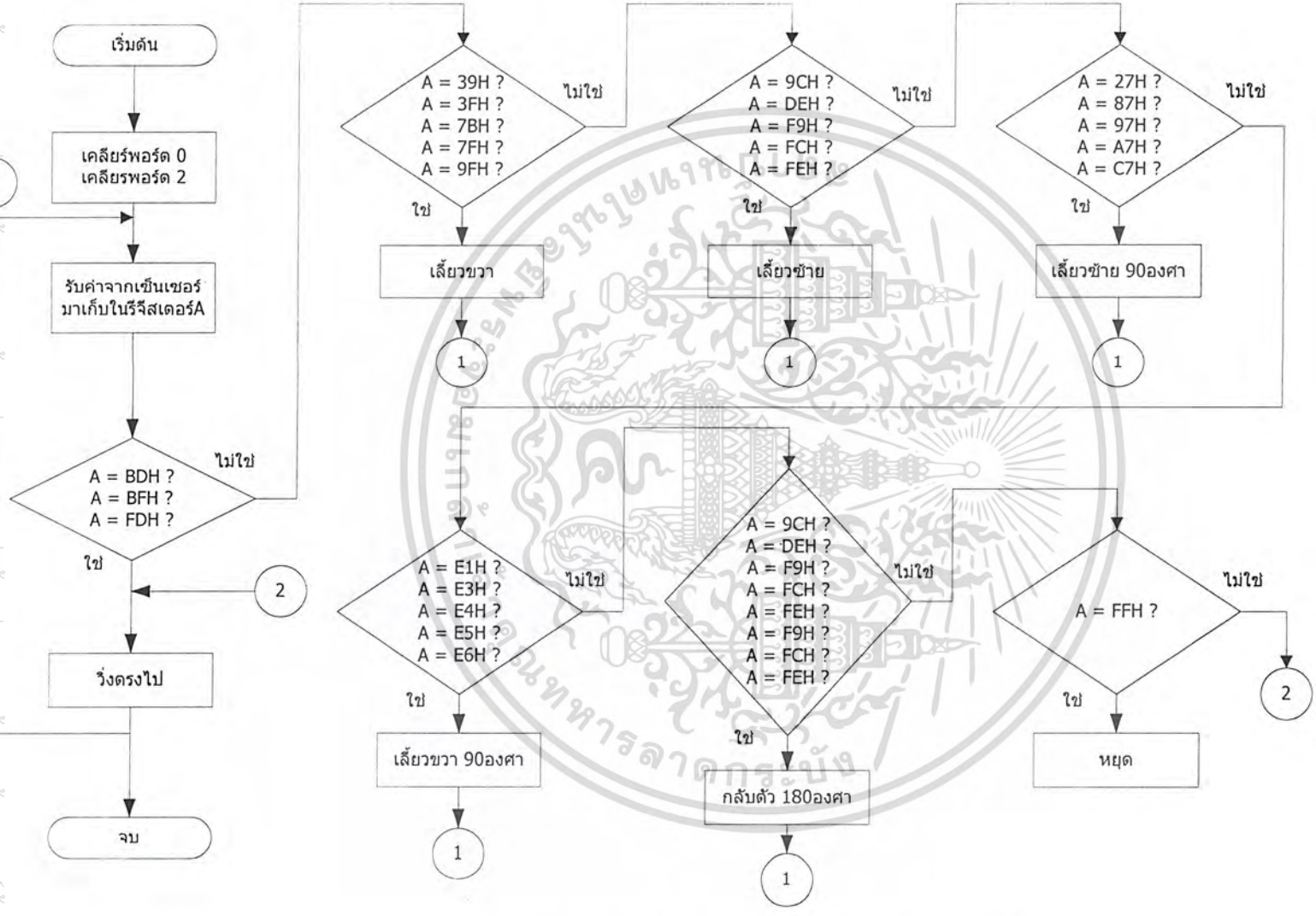
1. Sensor เบอร์ EE-SF5-B จำนวน 3ชุด ชุดละ	4 ตัว
2. RPACK 10K*4 จำนวน 3ชุด ชุดละ	3 ตัว
3. ความต้านทานปรับค่าได้ 10 KB 4ชุดละ	4 ตัว
4. IC TLC274CN 14 ขา	1 ตัว
5. LED 3ชุด ชุดละ	4 ตัว
6. ความต้านทาน 100 โอมห์ 3 ชุด ชุดละ	2 ตัว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



Flowchart ของโปรแกรมการเดินตามเส้นสี

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



Flowchart ของโปรแกรมการเดินออกจากเขาวงกต

```
;*************************************************************************
;   Program          :   Move in Magic Mounten
;   For              :   Mouse Robot
;   Filename         :   KWK.ASM
;   Assembler        :   SXA51
;*************************************************************************
;-------------------------------------------------
;   Main Program
;-------------------------------------------------
            ORG         0000H           ;start
;-------------------------------------------------
;   Clear Port
;-------------------------------------------------
            MOV         P0,#00H         ;clear port0
            MOV         P2,#00H         ;clear port2
;-------------------------------------------------
;   Receive input data
;-------------------------------------------------
INPUT:      MOV         P2,#0FFH        ;receive input from senser
            MOV         A,0A0H          ;move data to register A
;-------------------------------------------------
;   Compare data with register A to move forward
;-------------------------------------------------
STEP1:      CJNE        A,#0BDH,STEP2   ;compare A with input data
            AJMP        GOGO            ;move forward
STEP2:      CJNE        A,#0BFH,STEP3   ;compare A with input data
            AJMP        GOGO            ;move forward
STEP3:      CJNE        A,#0FDH,STEP4   ;compare A with input data
            AJMP        GOGO            ;move forward
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
;-----------------------------------------------------
;       Compare data with register A to turn right
;-----------------------------------------------------
STEP4:      CJNE    A,#039H,STEP5       ;compare A with input data
            AJMP    RIGHT               ;turn right
STEP5:      CJNE    A,#03FH,STEP6       ;compare A with input data
            AJMP    RIGHT               ;turn right
STEP6:      CJNE    A,#07BH,STEP7       ;compare A with input data
            AJMP    RIGHT               ;turn right
STEP7:      CJNE    A,#07FH,STEP8       ;compare A with input data
            AJMP    RIGHT               ;turn right
STEP8:      CJNE    A,#09FH,STEP9       ;compare A with input data
            AJMP    RIGHT               ;turn right
;-----------------------------------------------------
;       Compare data with register A to turn left
;-----------------------------------------------------
STEP9:      CJNE    A,#0DEH,STEP10      ;compare A with input data
            AJMP    LEFT                ;turn left
STEP10:     CJNE    A,#0FEH,STEP11      ;compare A with input data
            AJMP    LEFT                ;turn left
STEP11:     CJNE    A,#09CH,STEP12      ;compare A with input data
            AJMP    LEFT                ;turn left
STEP12:     CJNE    A,#0FCH,STEP13      ;compare A with input data
            AJMP    LEFT                ;turn left
STEP13:     CJNE    A,#0F9H,STEP14      ;compare A with input data
            AJMP    LEFT                ;turn left
;-----------------------------------------------------
;       Compare data with register A to turn left 90 degree
;-----------------------------------------------------
STEP14:     CJNE    A,#027H,STEP15      ;compare A with input data
            AJMP    LLEFT               ;turn left 90 degree
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
STEP15:   CJNE    A,#087H,STEP16    ;compare A with input data
          AJMP    LLEFT             ;turn left 90 degree
STEP16:   CJNE    A,#097H,STEP17    ;compare A with input data
          AJMP    LLEFT             ;turn left 90 degree
STEP17:   CJNE    A,#0A7H,STEP18    ;compare A with input data
          AJMP    LLEFT             ;turn left 90 degree
STEP18:   CJNE    A,#0C7H,STEP19    ;compare A with input data
          AJMP    LLEFT             ;turn left 90 degree
;-------------------------------------------------------
;     Compare data with register A to turn right 90 degree
;-------------------------------------------------------
STEP19:   CJNE    A,#0E1H,STEP20    ;compare A with input data
          AJMP    LRIGHT            ;turn right 90 degree
STEP20:   CJNE    A,#0E3H,STEP21    ;compare A with input data
          AJMP    LRIGHT            turn right 90 degree
STEP21:   CJNE    A,#0E4H,STEP22    ;compare A with input data
          AJMP    LRIGHT            ;turn right 90 degree
STEP22:   CJNE    A,#0E5H,STEP23    ;compare A with input data
          AJMP    LRIGHT            ;turn right 90 degree
STEP23:   CJNE    A,#0E6H,STEP24    ;compare A with input data
          AJMP    LRIGHT            ;turn right 90 degree
;-------------------------------------------------------
;     Compare data with register A to u-turn
;-------------------------------------------------------
STEP24:   CJNE    A,#0A5H,STEP25    ;compare A with input data
          AJMP    UTURN             ;u-turn
STEP25:   CJNE    A,#021H,STEP26    ;compare A with input data
          AJMP    UTURN             ;u-turn
STEP26:   CJNE    A,#023H,STEP27    ;compare A with input data
          AJMP    UTURN              ;u-turn
STEP27:   CJNE    A,#063H,STEP28    ;compare A with input data
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                AJMP        UTURN                   ;u-turn
STEP28:         CJNE        A,#085H,STEP29          ;compare A with input data
                AJMP        UTURN                   ;u-turn
STEP29:         CJNE        A,#084H,STEP30          ;compare A with input data
                AJMP        UTURN                   ;u-turn
STEP30:         CJNE        A,#0C4H,STEP31          ;compare A with input data
                AJMP        UTURN                   ;u-turn
STEP31:         CJNE        A,#0C6H,STEP32          ;compare A with input data
                AJMP        UTURN                   ;u-turn
;-------------------------------------------------
;       Compare data with register A to stop
;-------------------------------------------------
STEP32:         CJNE        A,#0FFH,STEP33          ;compare A with input data
                AJMP        STOP                    ;stop
;-------------------------------------------------
;       Any data move forward
;-------------------------------------------------
STEP33:         ACALL       GOGO                    ;move forward
;-------------------------------------------------
;       Move forward
;-------------------------------------------------
GOGO:           MOV         P0,#0CCH                ;drive stepping motor
                ACALL       DLAY                    ;delay time
                MOV         P0,#066H                ;drive stepping motor
                ACALL       DLAY                    ;delay time
                MOV         P0,#033H                ;drive stepping motor
                ACALL       DLAY                    ;delay time
                MOV         P0,#099H                ;drive stepping motor
                ACALL       DLAY                    ;delay time
                AJMP        INPUT
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
;-------------------------------------------------------
;     Turn right
;-------------------------------------------------------
RIGHT:    MOV     P0,#0C0H        ;drive stepping motor
          ACALL   DLAY            ;delay time
          MOV     P0,#060H        ;drive stepping motor
          ACALL   DLAY            ;delay time
          MOV     P0,#030H        ;drive stepping motor
          ACALL   DLAY            ;delay time
          MOV     P0,#090H        ;drive stepping motor
          ACALL   DLAY            ;delay time
          AJMP    NPUT

;-------------------------------------------------------
;     Turn right 90 degree
;-------------------------------------------------------
LRIGHT:   MOV     B,#05H          ;time of doing
RR:       MOV     P0,#0CCH        ;drive stepping motor
          ACALL   DLAY            ;delay time
          MOV     P0,#069H        ;drive stepping motor
          ACALL   DLAY            ;delay time
          MOV     P0,#033H        ;drive stepping motor
          ACALL   DLAY            ;delay time
          MOV     P0,#096H        ;drive stepping motor
          ACALL   DLAY            ;delay time
          MOV     P0,#0CCH        ;drive stepping motor
          ACALL   DLAY            ;delay time
          MOV     P0,#069H        ;drive stepping motor
          ACALL   DLAY            ;delay time
          MOV     P0,#033H        ;drive stepping motor
          ACALL   DLAY            ;delay time
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
MOV     P0,#096H        ;drive stepping motor
ACALL   DLAY            ;delay time
MOV     P0,#0CCH        ;drive stepping motor
ACALL   DLAY            ;delay time
MOV     P0,#069H        ;drive stepping motor
ACALL   DLAY            ;delay time
MOV     P0,#033H        ;drive stepping motor
ACALL   DLAY            ;delay time
MOV     P0,#096H        ;drive stepping motor
ACALL   DLAY            ;delay time
MOV     P0,#0CCH        ;drive stepping motor
ACALL   DLAY            ;delay time
MOV     P0,#069H        ;drive stepping motor
ACALL   DLAY            ;delay time
MOV     P0,#033H        ;drive stepping motor
ACALL   DLAY            ;delay time
MOV     P0,#096H        ;drive stepping motor
ACALL   DLAY            ;delay time
MOV     P0,#0CCH        ;drive stepping motor
ACALL   DLAY            ;delay time
MOV     P0,#069H        ;drive stepping motor
ACALL   DLAY            ;delay time
MOV     P0,#033H        ;drive stepping motor
ACALL   DLAY            ;delay time
MOV     P0,#096H        ;drive stepping motor
ACALL   DLAY            ;delay time
DJNZ    B,RR            ;do B time?
AJMP    INPUT
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
;-------------------------------------------------------
;     Turn left
;-------------------------------------------------------
LEFT:   MOV     P0,#00CH        ;drive stepping motor
        ACALL   DLAY            ;delay time
        MOV     P0,#006H        ;drive stepping motor
        ACALL   DLAY            ;delay time
        MOV     P0,#003H        ;drive stepping motor
        ACALL   DLAY            ;delay time
        MOV     P0,#009H        ;drive stepping motor
        ACALL   DLAY            ;delay time
        AJMP    INPUT
;-------------------------------------------------------
;     Turn left 90 degree
;-------------------------------------------------------
LLEFT:  MOV     B,#05H          ;time of doing
LL:     MOV     P0,#0CCH        ;drive stepping motor
        ACALL   DLAY            ;delay time
        MOV     P0,#096H        ;drive stepping motor
        ACALL   DLAY            ;delay time
        MOV     P0,#033H        ;drive stepping motor
        ACALL   DLAY            ;delay time
        MOV     P0,#069H        ;drive stepping motor
        ACALL   DLAY            ;delay time
        MOV     P0,#0CCH        ;drive stepping motor
        ACALL   DLAY            ;delay time
        MOV     P0,#096H        ;drive stepping motor
        ACALL   DLAY            ;delay time
        MOV     P0,#033H        ;drive stepping motor
        ACALL   DLAY            ;delay time
        MOV     P0,#069H        ;drive stepping motor
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        ACALL   DLAY            ;delay time
        MOV     P0,#0CCH        ;drive stepping motor
        ACALL   DLAY            ;delay time
        MOV     P0,#096H        ;drive stepping motor
        ACALL   DLAY            ;delay time
        MOV     P0,#033H        ;drive stepping motor
        ACALL   DLAY            ;delay time
        MOV     P0,#069H        ;drive stepping motor
        ACALL   DLAY            ;delay time
        MOV     P0,#0CCH        ;drive stepping motor
        ACALL   DLAY            ;delay time
        MOV     P0,#096H        ;drive stepping motor
        ACALL   DLAY            ;delay time
        MOV     P0,#033H        ;drive stepping motor
        ACALL   DLAY            ;delay time
        MOV     P0,#069H        ;drive stepping motor
        ACALL   DLAY            ;delay time
        MOV     P0,#0CCH        ;drive stepping motor
        ACALL   DLAY            ;delay time
        MOV     P0,#096H        ;drive stepping motor
        ACALL   DLAY            ;delay time
        MOV     P0,#033H        ;drive stepping motor
        ACALL   DLAY            ;delay time
        MOV     P0,#069H        ;drive stepping motor
        ACALL   DLAY            ;delay time
        DJNZ    B,LL            ;do B time?
        AJMP    INPUT
;-------------------------------------------------------
;   U-Turn
;-------------------------------------------------------
UTURN:  MOV     B,#0AH          ;time of doing
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
UU:     MOV     P0,#0CCH        ;drive stepping motor
        ACALL   DLAY            ;delay time
        MOV     P0,#069H        ;drive stepping motor
        ACALL   DLAY            ;delay time
        MOV     P0,#033H        ;drive stepping motor
        ACALL   DLAY            ;delay time
        MOV     P0,#096H        ;drive stepping motor
        ACALL   DLAY            ;delay time
        MOV     P0,#0CCH        ;drive stepping motor
        ACALL   DLAY            ;delay time
        MOV     P0,#069H        ;drive stepping motor
        ACALL   DLAY            ;delay time
        MOV     P0,#033H        ;drive stepping motor
        ACALL   DLAY            ;delay time
        MOV     P0,#096H        ;drive stepping motor
        ACALL   DLAY            ;delay time
        MOV     P0,#0CCH        ;drive stepping motor
        ACALL   DLAY            ;delay time
        MOV     P0,#069H        ;drive stepping motor
        ACALL   DLAY            ;delay time
        MOV     P0,#033H        ;drive stepping motor
        ACALL   DLAY            ;delay time
        MOV     P0,#096H        ;drive stepping motor
        ACALL   DLAY            ;delay time
        MOV     P0,#0CCH        ;drive stepping motor
        ACALL   DLAY            ;delay time
        MOV     P0,#069H        ;drive stepping motor
        ACALL   DLAY            ;delay time
        MOV     P0,#033H        ;drive stepping motor
        ACALL   DLAY            ;delay time
        MOV     P0,#096H        ;drive stepping motor
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                ACALL   DLAY            ;delay time
                MOV     P0,#0CCH        ;drive stepping motor
                ACALL   DLAY            ;delay time
                MOV     P0,#069H        ;drive stepping motor
                ACALL   DLAY            ;delay time
                MOV     P0,#033H        ;drive stepping motor
                ACALL   DLAY            ;delay time
                MOV     P0,#096H        ;drive stepping motor
                ACALL   DLAY            ;delay time
                DJNZ    B,UU            ;do B time?
                MOV     P0,#0CCH        ;drive stepping motor
                ACALL   DLAY            ;delay time
                MOV     P0,#069H        ;drive stepping motor
                ACALL   DLAY            ;delay time
                MOV     P0,#033H        ;drive stepping motor
                ACALL   DLAY            ;delay time
                MOV     P0,#096H        ;drive stepping motor
                ACALL   DLAY            ;delay time
                AJMP    INPUT
;-------------------------------------------------
;       Stop
;-------------------------------------------------
STOP:           MOV     1,#050
ST1:            MOV     2,#0100
ST2:            MOV     3,#0E6H
ST3:            NOP
                NOP
                DJNZ    R3,ST3
                DJNZ    R2,ST2
                DJNZ    R1,ST1
                AJMP    INPUT
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
;-------------------------------------------------------
;     Delay time 50 ms
;-------------------------------------------------------
DLAY:   MOV     1,#050          ;speed adjust for drive stepping motor
L1:     MOV     2,#0E6H
L2:     NOP
        NOP
        DJNZ    R2,L2
        DJNZ    R1,L1
        RET                     return address
        END                     ;end of program
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
;*********************************************************************
;Program        :    Move forward white line
;For            :    Mouse Robot
;Filename       :    WLINE.ASM
;Assembler      :    SXA51
;*********************************************************************
;-------------------------------------------
;  Main Program
;-------------------------------------------
                ORG         0000H               ;start in address 0000h
;-------------------------------------------
;  Clear Port0 and Port1
;-------------------------------------------
                MOV         P0,#00H             ;clear port0
                MOV         P1,#00H             ;clear port1
;-------------------------------------------
;  Receive input data
;-------------------------------------------
INPUT:          MOV         P1,#0FFH            ;set input port
                MOV         A,90H               ;move data to register A from senser
;-------------------------------------------
;  Compare data with register A to move forward
;-------------------------------------------
STEP0:          CJNE        A,#0F0H,STEP1       ;compare A with F0h
                ACALL       GO                  ;move forward
                ACALL       GO                  ;move forward
                AJMP        INPUT               ;recieve input data and move to A
STEP1:          CJNE        A,#0F9H,STEP2       ;compare A with F9h
                ACALL       GO                  ;move forward
                ACALL       GO                  ;move forward
                ACALL       GO                  ;move forward
                ACALL       GO                  ;move forward
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                AJMP        INPUT               ;recieve input data and move to A
;-----------------------------------------
;   Compare data to turn left
;-----------------------------------------
STEP2:          CJNE        A,#0FCH,STEP3       ;compare A with FCh
                ACALL       LEFT                ;turn left
                AJMP        INPUT               ;recieve input data and move to A
STEP3:          CJNE        A,#0FEH,STEP4       ;compare A with FEh
                ACALL       LEFT                ;turn left
                ACALL       LEFT                ;turn left
                AJMP        INPUT               ;recieve input data and move to A
STEP4:          CJNE        A,#0FDH,STEP5       ;compare A with FDh
                ACALL       LEFT                ;turn left
                AJMP        INPUT               ;recieve input data and move to A
STEP5:          CJNE        A,#0F8H,STEP6       ;compare A with F8h
                ACALL       LEFT                ;turn left
                ACALL       LEFT                ;turn left
                ACALL       LEFT                ;turn left
                AJMP        INPUT               ;recieve input data and move to A
;-----------------------------------------
;   Compare data to turn right
;-----------------------------------------
STEP6:          CJNE        A,#0F3H,STEP7       ;compare A with F3h
                ACALL       RIGHT               ;turn right
                AJMP        INPUT               ;recieve input data and move to A
STEP7:          CJNE        A,#0F7H,STEP8       ;compare A with F7h
                ACALL       RIGHT               ;turn right
                ACALL       RIGHT               ;turn right
                AJMP        INPUT               ;recieve input data and move to A
STEP8:          CJNE        A,#0FBH,STEP9       ;compare A with FBh
                ACALL       RIGHT               ;turn right
                AJMP        INPUT               ;recieve input data and move to A
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
STEP9:      CJNE    A,#0F1H,STEP10      ;compare A with F1h
            ACALL   RIGHT               ;turn right
            ACALL   RIGHT               ;turn right
            ACALL   RIGHT               ;turn right
            AJMP    INPUT               ;recieve input data and move to A
;---------------------------------------------
;   Compare data to move circle
;---------------------------------------------
STEP10:     CJNE    A,#0FFH,STEP11
            ACALL   CCLE                ;move circle
            AJMP    INPUT               ;recieve input data and move to A
;---------------------------------------------
;   Any data move forward
;---------------------------------------------
STEP11:     ACALL   GO                  ;move forward
            AJMP    INPUT               ;recieve input data and move to A
;---------------------------------------------
;   Move forward
;---------------------------------------------
GO:         MOV     P0,#088H            ;drive steping motor
            ACALL   DLAY                ;delay time 20 ms
            MOV     P0,#0CCH
            ACALL   DLAY
            MOV     P0,#044H
            ACALL   DLAY
            MOV     P0,#066H
            ACALL   DLAY
            MOV     P0,#022H
            ACALL   DLAY
            MOV     P0,#033H
            ACALL   DLAY
            MOV     P0,#011H
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                ACALL   DLAY
                MOV     P0,#099H
                ACALL   DLAY
                RET                             ;return address
;-----------------------------------------------
;   Turn right
;-----------------------------------------------
RIGHT:          MOV     P0,#080H                ;drive steping motor
                ACALL   DLAY                    ;delay time 20 ms
                MOV     P0,#0C0H
                ACALL   DLAY
                MOV     P0,#040H
                ACALL   DLAY
                MOV     P0,#060H
                ACALL   DLAY
                MOV     P0,#020H
                ACALL   DLAY
                MOV     P0,#030H
                ACALL   DLAY
                MOV     P0,#010H
                ACALL   DLAY
                MOV     P0,#090H
                ACALL   DLAY
                RET                             ;return address
;-----------------------------------------------
;   Turn left
;-----------------------------------------------
LEFT:           MOV     P0,#008H                ;drive steping motor
                ACALL   DLAY                    ;delay time 20 ms
                MOV     P0,#00CH
                ACALL   DLAY
                MOV     P0,#004H
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        ACALL   DLAY
        MOV     P0,#006H
        ACALL   DLAY
        MOV     P0,#002H
        ACALL   DLAY
        MOV     P0,#003H
        ACALL   DLAY
        MOV     P0,#001H
        ACALL   DLAY
        MOV     P0,#009H
        ACALL   DLAY
        RET                         ;return address
;-----------------------------------------
;   Move circle
;-----------------------------------------
CCLE:   MOV     P0,#088H            ;drive steping motor
        ACALL   DLAY                ;delay time 20 ms
        MOV     P0,#0C9H
        ACALL   DLAY
        MOV     P0,#041H
        ACALL   DLAY
        MOV     P0,#063H
        ACALL   DLAY
        MOV     P0,#022H
        ACALL   DLAY
        MOV     P0,#036H
        ACALL   DLAY
        MOV     P0,#014H
        ACALL   DLAY
        MOV     P0,#09CH
        ACALL   DLAY
        RET                         ;return address
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
;-----------------------------------------------
;delay time 20 ms
;-----------------------------------------------
DLAY:   MOV     1,#020          ;adjust speed at R1 [ now use 20 ms ]
L1:     MOV     2,#0E6H         ;delay time 1 ms
L2:     NOP
        NOP
        DJNZ    R2,L2
        DJNZ    R1,L1
        RET                     ;return address
        END                     ;end of program
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ผนวก ข. รายละเอียดของไอซีที่ใช้ในวงจร

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Features

- Compatible with MCS-51™ Products
- 8K Bytes of In-System Reprogrammable Downloadable Flash Memory
  - SPI Serial Interface for Program Downloading
  - Endurance: 1,000 Write/Erase Cycles
- 2K Bytes EEPROM
  - Endurance: 100,000 Write/Erase Cycles
- 4V to 6V Operating Range
- Fully Static Operation: 0 Hz to 24 MHz
- Three-level Program Memory Lock
- 256 x 8-bit Internal RAM
- 32 Programmable I/O Lines
- Three 16-bit Timer/Counters
- Nine Interrupt Sources
- Programmable UART Serial Channel
- SPI Serial Interface
- Low-power Idle and Power-down Modes
- Interrupt Recovery From Power-down
- Programmable Watchdog Timer
- Dual Data Pointer
- Power-off Flag

Description

The AT89S8252 is a low-power, high-performance CMOS 8-bit microcomputer with 8K bytes of downloadable Flash programmable and erasable read only memory and 2K bytes of EEPROM. The device is manufactured using Atmel's high-density nonvolatile memory technology and is compatible with the industry-standard 80C51 instruction set and pinout. The on-chip downloadable Flash allows the program memory to be reprogrammed in-system through an SPI serial interface or by a conventional nonvolatile memory programmer. By combining a versatile 8-bit CPU with downloadable Flash on a monolithic chip, the Atmel AT89S8252 is a powerful microcomputer which provides a highly-flexible and cost-effective solution to many embedded control applications.

The AT89S8252 provides the following standard features: 8K bytes of downloadable Flash, 2K bytes of EEPROM, 256 bytes of RAM, 32 I/O lines, programmable watchdog timer, two data pointers, three 16-bit timer/counters, a six-vector two-level interrupt architecture, a full duplex serial port, on-chip oscillator, and clock circuitry. In addition, the AT89S8252 is designed with static logic for operation down to zero frequency and supports two software selectable power saving modes. The Idle Mode stops the CPU while allowing the RAM, timer/counters, serial port, and interrupt system to continue functioning. The Power-down mode saves the RAM contents but freezes the oscillator, disabling all other chip functions until the next interrupt or hardware reset.

The downloadable Flash can be changed a single byte at a time and is accessible through the SPI serial interface. Holding RESET active forces the SPI bus into a serial programming interface and allows the program memory to be written to or read from unless Lock Bit 2 has been activated.

ATMEL®

8-bit Microcontroller with 8K Bytes Flash

AT89S8252

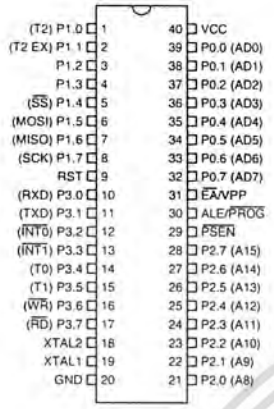
Rev. 0401E–02/00

ATMEL

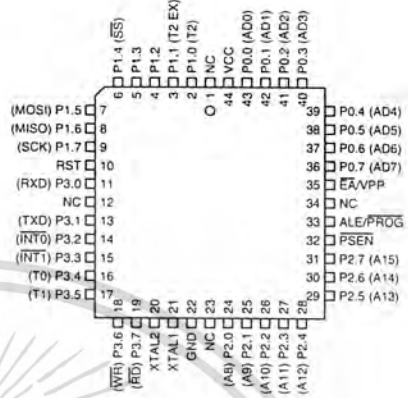
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Pin Configurations

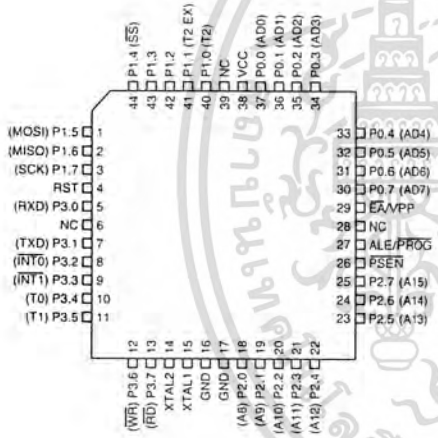
PDIP



PLCC



PQFP/TQFP



Pin Description

VCC

Supply voltage.

GND

Ground.

Port 0

Port 0 is an 8-bit open drain bbi-didirectional 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.

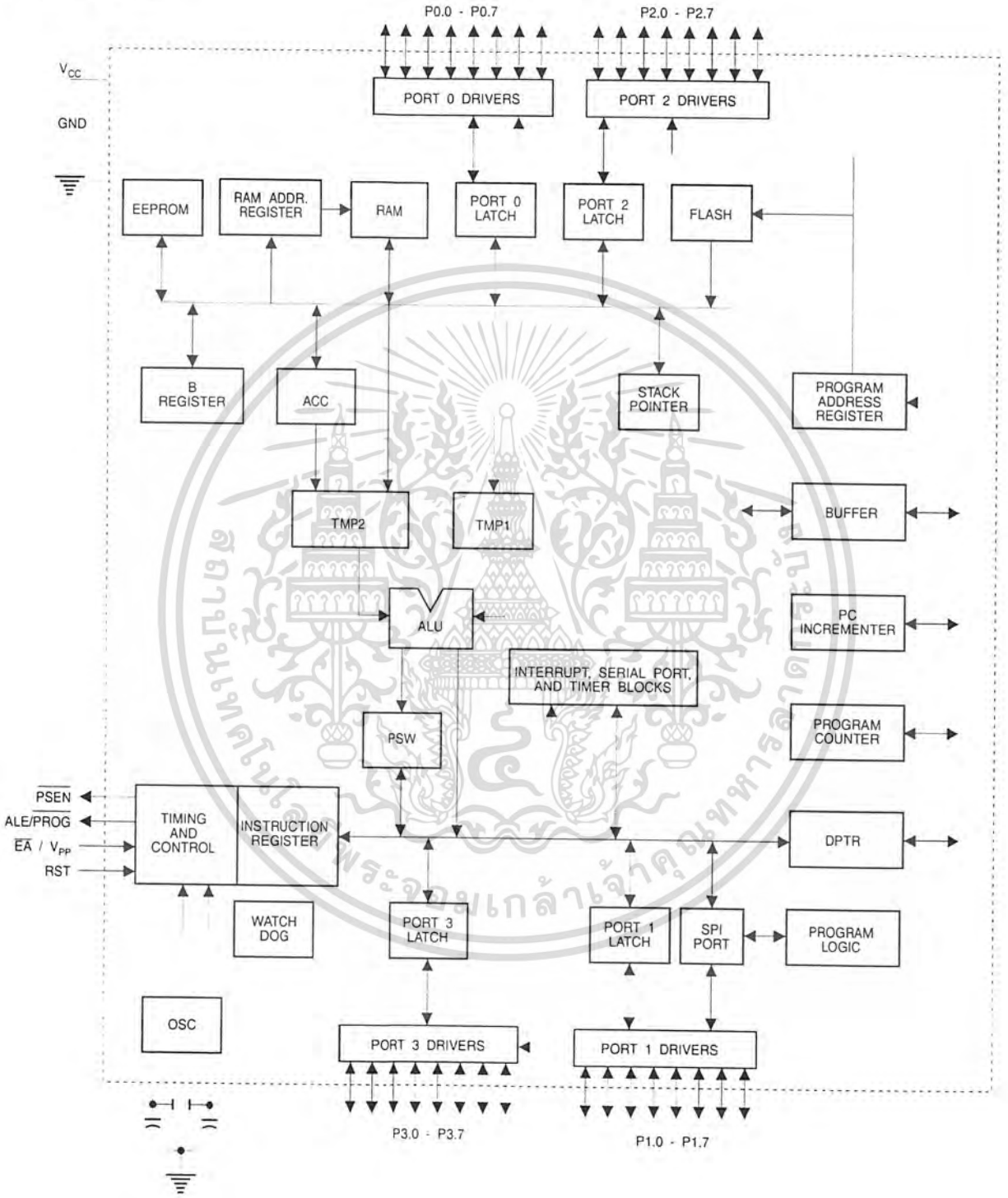
Port 0 also receives the code bytes during Flash programming and outputs the code bytes during program verification. External pullups are required during program verification.

Port 1

Port 1 is an 8-bit bi-directional 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.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Block Diagram



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

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.

Pin Description

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 bi-directional 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 bi-directional 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.

$\overline{EA}$/VPP

External Access Enable. $\overline{EA}$ must be strapped to GND in order to enable the device to fetch code from external pro-

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

gram 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.

XTAL1

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

XTAL2

Output from the inverting oscillator amplifier.

Table 1. AT89S8252 SFR Map and Reset Values

0F8H									0FFH
0F0H	B 00000000								0F7H
0E8H									0EFH
0E0H	ACC 00000000								0E7H
0D8H									0DFH
0D0H	PSW 00000000					SPCR 000001XX			0D7H
0C8H	T2CON 00000000	T2MOD XXXXXX00	RCAP2L 00000000	RCAP2H 00000000	TL2 00000000	TH2 00000000			0CFH
0C0H									0C7H
0B8H	IP XX000000								0BFH
0B0H	P3 11111111								0B7H
0A8H	IE 0X000000		SPSR 00XXXXXX						0AFH
0A0H	P2 11111111								0A7H
98H	SCON 00000000	SBUF XXXXXXXX							9FH
90H	P1 11111111						WMCON 00000010		97H
88H	TCON 00000000	TMOD 00000000	TL0 00000000	TL1 00000000	TH0 00000000	TH1 00000000			8FH
80H	P0 11111111	SP 00000111	DP0L 00000000	DP0H 00000000	DP1L 00000000	DP1H 00000000	SPDR XXXXXXXX	PCON 0XXX0000	87H

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Special Function Registers

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

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

User software should not write 1s to these unlisted locations, since they may be used in future products to invoke new features. In that case, the reset or inactive values of the new bits will always be 0.

Timer 2 Registers Control and status bits are contained in registers T2CON (shown in Table 2) and T2MOD (shown in Table 9) for Timer 2. The register pair (RCAP2H, RCAP2L) are the Capture/Reload registers for Timer 2 in 16 bit capture mode or 16-bit auto-reload mode.

Table 2. T2CON—Timer/Counter 2 Control Register

T2CON Address = 0C8H — Reset Value = 0000 0000B

Bit Addressable

Bit	7	6	5	4	3	2	1	0
	TF2	EXF2	RCLK	TCLK	EXEN2	TR2	C/$\overline{T2}$	CP/$\overline{RL2}$

Symbol	Function
TF2	Timer 2 overflow flag set by a Timer 2 overflow and must be cleared by software. TF2 will not be set when either RCLK = 1 or TCLK = 1.
EXF2	Timer 2 external flag set when either a capture or reload is caused by a negative transition on T2EX and EXEN2 = 1. When Timer 2 interrupt is enabled, EXF2 = 1 will cause the CPU to vector to the Timer 2 interrupt routine. EXF2 must be cleared by software. EXF2 does not cause an interrupt in up/down counter mode (DCEN = 1).
RCLK	Receive clock enable. When set, causes the serial port to use Timer 2 overflow pulses for its receive clock in serial port Modes 1 and 3. RCLK = 0 causes Timer 1 overflows to be used for the receive clock.
TCLK	Transmit clock enable. When set, causes the serial port to use Timer 2 overflow pulses for its transmit clock in serial port Modes 1 and 3. TCLK = 0 causes Timer 1 overflows to be used for the transmit clock.
EXEN2	Timer 2 external enable. When set, allows a capture or reload to occur as a result of a negative transition on T2EX if Timer 2 is not being used to clock the serial port. EXEN2 = 0 causes Timer 2 to ignore events at T2EX.
TR2	Start/Stop control for Timer 2. TR2 = 1 starts the timer.
C/$\overline{T2}$	Timer or counter select for Timer 2. C/$\overline{T2}$ = 0 for timer function. C/$\overline{T2}$ = 1 for external event counter (falling edge triggered).
CP/$\overline{RL2}$	Capture/Reload select. CP/$\overline{RL2}$ = 1 causes captures to occur on negative transitions at T2EX if EXEN2 = 1. CP/$\overline{RL2}$ = 0 causes automatic reloads to occur when Timer 2 overflows or negative transitions occur at T2EX when EXEN2 = 1. When either RCLK or TCLK = 1, this bit is ignored and the timer is forced to auto-reload on Timer 2 overflow.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Watchdog and Memory Control Register The WMCON register contains control bits for the Watchdog Timer (shown in Table 3). The EEMEN and EEMWE bits are used to select the 2K bytes on-chip EEPROM, and to enable byte-write. The DPS bit selects one of two DPTR registers available.

Table 3. WMCON—Watchdog and Memory Control Register

WMCON Address = 96H — Reset Value = 0000 0010B

Bit	PS2	PS1	PS0	EEMWE	EEMEN	DPS	WDTRST	WDTEN
	7	6	5	4	3	2	1	0

Symbol	Function
PS2 PS1 PS0	Prescaler Bits for the Watchdog Timer. When all three bits are set to "0", the watchdog timer has a nominal period of 16 ms. When all three bits are set to "1", the nominal period is 2048 ms.
EEMWE	EEPROM Data Memory Write Enable Bit. Set this bit to "1" before initiating byte write to on-chip EEPROM with the MOVX instruction. User software should set this bit to "0" after EEPROM write is completed.
EEMEN	Internal EEPROM Access Enable. When EEMEN = 1, the MOVX instruction with DPTR will access on-chip EEPROM instead of external data memory. When EEMEN = 0, MOVX with DPTR accesses external data memory.
DPS	Data Pointer Register Select. DPS = 0 selects the first bank of Data Pointer Register, DP0, and DPS = 1 selects the second bank, DP1
WDTRST RDY/$\overline{BSY}$	Watchdog Timer Reset and EEPROM Ready/$\overline{Busy}$ Flag. Each time this bit is set to "1" by user software, a pulse is generated to reset the watchdog timer. The WDTRST bit is then automatically reset to "0" in the next instruction cycle. The WDTRST bit is Write-Only. This bit also serves as the RDY/$\overline{BSY}$ flag in a Read-Only mode during EEPROM write. RDY/$\overline{BSY}$ = 1 means that the EEPROM is ready to be programmed. While programming operations are being executed, the RDY/$\overline{BSY}$ bit equals "0" and is automatically reset to "1" when programming is completed.
WDTEN	Watchdog Timer Enable Bit. WDTEN = 1 enables the watchdog timer and WDTEN = 0 disables the watchdog timer.

SPI Registers Control and status bits for the Serial Peripheral Interface are contained in registers SPCR (shown in Table 4) and SPSR (shown in Table 5). The SPI data bits are contained in the SPDR register. Writing the SPI data register during serial data transfer sets the Write Collision bit, WCOL, in the SPSR register. The SPDR is double buffered for writing and the values in SPDR are not changed by Reset.

Interrupt Registers The global interrupt enable bit and the individual interrupt enable bits are in the IE register. In addition, the individual interrupt enable bit for the SPI is in the SPCR register. Two priorities can be set for each of the six interrupt sources in the IP register.

Dual Data Pointer Registers To facilitate accessing both internal EEPROM and external data memory, two banks of 16 bit Data Pointer Registers are provided: DP0 at SFR address locations 82H-83H and DP1 at 84H-85H. Bit DPS = 0 in SFR WMCON selects DP0 and DPS = 1 selects DP1. The user should always initialize the DPS bit to the appropriate value before accessing the respective Data Pointer Register.

Power Off Flag The Power Off Flag (POF) is located at bit_4 (PCON.4) in the PCON SFR. POF is set to "1" during power up. It can be set and reset under software control and is not affected by RESET.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Table 4. SPCR—SPI Control Register

SPCR Address = D5H — Reset Value = 0000 01XXB

Bit	7	6	5	4	3	2	1	0
	SPIE	SPE	DORD	MSTR	CPOL	CPHA	SPR1	SPR0

Symbol	Function
SPIE	SPI Interrupt Enable. This bit, in conjunction with the ES bit in the IE register, enables SPI interrupts: SPIE = 1 and ES = 1 enable SPI interrupts. SPIE = 0 disables SPI interrupts.
SPE	SPI Enable. SPI = 1 enables the SPI channel and connects $\overline{SS}$, MOSI, MISO and SCK to pins P1.4, P1.5, P1.6, and P1.7. SPI = 0 disables the SPI channel.
DORD	Data Order. DORD = 1 selects LSB first data transmission. DORD = 0 selects MSB first data transmission.
MSTR	Master/Slave Select. MSTR = 1 selects Master SPI mode. MSTR = 0 selects Slave SPI mode.
CPOL	Clock Polarity. When CPOL = 1, SCK is high when idle. When CPOL = 0, SCK of the master device is low when not transmitting. Please refer to figure on SPI Clock Phase and Polarity Control.
CPHA	Clock Phase. The CPHA bit together with the CPOL bit controls the clock and data relationship between master and slave. Please refer to figure on SPI Clock Phase and Polarity Control.
SPR0 SPR1	SPI Clock Rate Select. These two bits control the SCK rate of the device configured as master. SPR1 and SPR0 have no effect on the slave. The relationship between SCK and the oscillator frequency, F_{osc}, is as follows: SPR1 SPR0 SCK = F_{osc} divided by 0 0 4 0 1 16 1 0 64 1 1 128

Table 5. SPSR – SPI Status Register

SPSR Address = AAH — Reset Value = 00XX XXXXB

Bit	7	6	5	4	3	2	1	0
	SPIF	WCOL	–	–	–	–	–	–

Symbol	Function
SPIF	SPI Interrupt Flag. When a serial transfer is complete, the SPIF bit is set and an interrupt is generated if SPIE = 1 and ES = 1. The SPIF bit is cleared by reading the SPI status register with SPIF and WCOL bits set, and then accessing the SPI data register.
WCOL	Write Collision Flag. The WCOL bit is set if the SPI data register is written during a data transfer. During data transfer, the result of reading the SPDR register may be incorrect, and writing to it has no effect. The WCOL bit (and the SPIF bit) are cleared by reading the SPI status register with SPIF and WCOL set, and then accessing the SPI data register.

Table 6. SPDR – SPI Data Register

SPDR Address = 86H — Reset Value = unchanged

Bit	7	6	5	4	3	2	1	0
	SPD7	SPD6	SPD5	SPD4	SPD3	SPD2	SPD1	SPD0

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Data Memory – EEPROM and RAM

The AT89S8252 implements 2K bytes of on-chip EEPROM for data storage and 256 bytes of RAM. The upper 128 bytes of RAM occupy a parallel space to the Special Function Registers. That means the upper 128 bytes have the same addresses as the SFR space but are physically separate from SFR space.

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

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

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

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

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

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

The on-chip EEPROM data memory is selected by setting the EEMEN bit in the WMCON register at SFR address location 96H. The EEPROM address range is from 000H to 7FFH. The MOVX instructions are used to access the EEPROM. To access off-chip data memory with the MOVX instructions, the EEMEN bit needs to be set to "0".

The EEMWE bit in the WMCON register needs to be set to "1" before any byte location in the EEPROM can be written. User software should reset EEMWE bit to "0" if no further EEPROM write is required. EEPROM write cycles in the serial programming mode are self-timed and typically take 2.5 ms. The progress of EEPROM write can be monitored by reading the RDY/$\overline{BSY}$ bit (read-only) in SFR WMCON. RDY/$\overline{BSY}$ = 0 means programming is still in progress and RDY/$\overline{BSY}$ = 1 means EEPROM write cycle is completed and another write cycle can be initiated.

In addition, during EEPROM programming, an attempted read from the EEPROM will fetch the byte being written with the MSB complemented. Once the write cycle is completed, true data are valid at all bit locations.

Programmable Watchdog Timer

The programmable Watchdog Timer (WDT) operates from an independent oscillator. The prescaler bits, PS0, PS1 and PS2 in SFR WMCON are used to set the period of the Watchdog Timer from 16 ms to 2048 ms. The available timer periods are shown in the following table and the actual timer periods (at V_{CC} = 5V) are within ±30% of the nominal.

The WDT is disabled by Power-on Reset and during Power-down. It is enabled by setting the WDTEN bit in SFR WMCON (address = 96H). The WDT is reset by setting the WDTRST bit in WMCON. When the WDT times out without being reset or disabled, an internal RST pulse is generated to reset the CPU.

Table 7. Watchdog Timer Period Selection

WDT Prescaler Bits			
PS2	PS1	PS0	Period (nominal)
0	0	0	16 ms
0	0	1	32 ms
0	1	0	64 ms
0	1	1	128 ms
1	0	0	256 ms
1	0	1	512 ms
1	1	0	1024 ms
1	1	1	2048 ms

Timer 0 and 1

Timer 0 and Timer 1 in the AT89S8252 operate the same way as Timer 0 and Timer 1 in the AT89C51, AT89C52 and AT89C55. For further information, see the October 1995 Microcontroller Data Book, page 2-45, section titled, "Timer/Counters."

Timer 2

Timer 2 is a 16 bit Timer/Counter that can operate as either a timer or an event counter. The type of operation is selected by bit C/$\overline{T2}$ in the SFR T2CON (shown in Table 2). Timer 2 has three operating modes: capture, auto-reload (up or down counting), and baud rate generator. The modes are selected by bits in T2CON, as shown in Table 8.

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

In the Counter function, the register is incremented in response to a 1-to-0 transition at its corresponding external input pin, T2. In this function, the external input is sampled during S5P2 of every machine cycle. When the samples show a high in one cycle and a low in the next cycle, the count is incremented. The new count value appears in the register during S3P1 of the cycle following the one in which

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

the transition was detected. Since two machine cycles (24 oscillator periods) are required to recognize a 1-to-0 transition, the maximum count rate is 1/24 of the oscillator frequency. To ensure that a given level is sampled at least once before it changes, the level should be held for at least one full machine cycle.

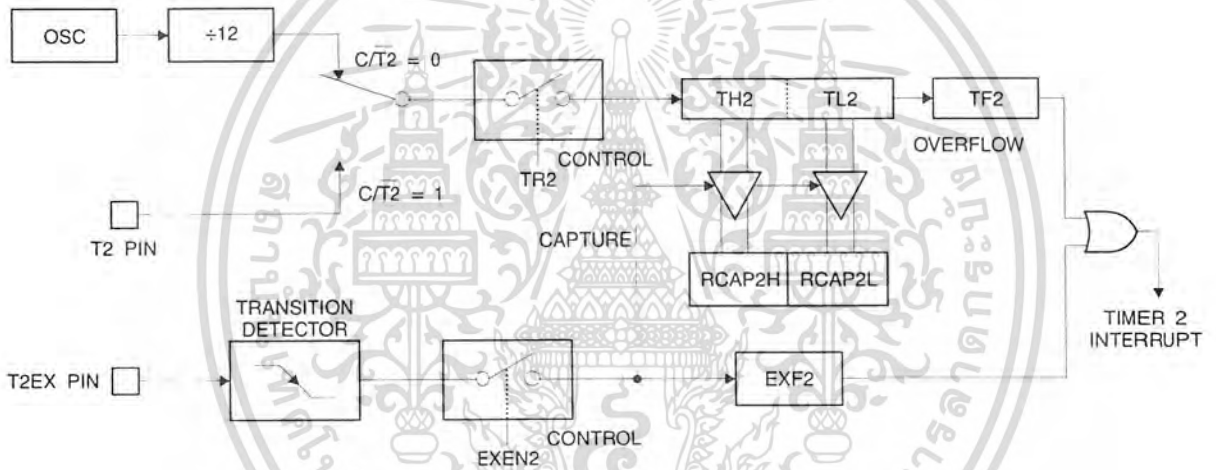
Table 8. Timer 2 Operating Modes

RCLK + TCLK	CP/$\overline{RL2}$	TR2	MODE
0	0	1	16-bit Auto-reload
0	1	1	16-bit Capture
1	X	1	Baud Rate Generator
X	X	0	(Off)

Capture Mode

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

Figure 1. Timer 2 in Capture Mode



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Auto-reload (Up or Down Counter)

Timer 2 can be programmed to count up or down when configured in its 16 bit auto-reload mode. This feature is invoked by the DCEN (Down Counter Enable) bit located in the SFR T2MOD (see Table 9). Upon reset, the DCEN bit is set to 0 so that timer 2 will default to count up. When DCEN is set, Timer 2 can count up or down, depending on the value of the T2EX pin.

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

Setting the DCEN bit enables Timer 2 to count up or down, as shown in Figure 3. In this mode, the T2EX pin controls the direction of the count. A logic 1 at T2EX makes Timer 2 count up. The timer will overflow at 0FFFFH and set the TF2 bit. This overflow also causes the 16 bit value in RCAP2H and RCAP2L to be reloaded into the timer registers, TH2 and TL2, respectively.

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

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

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

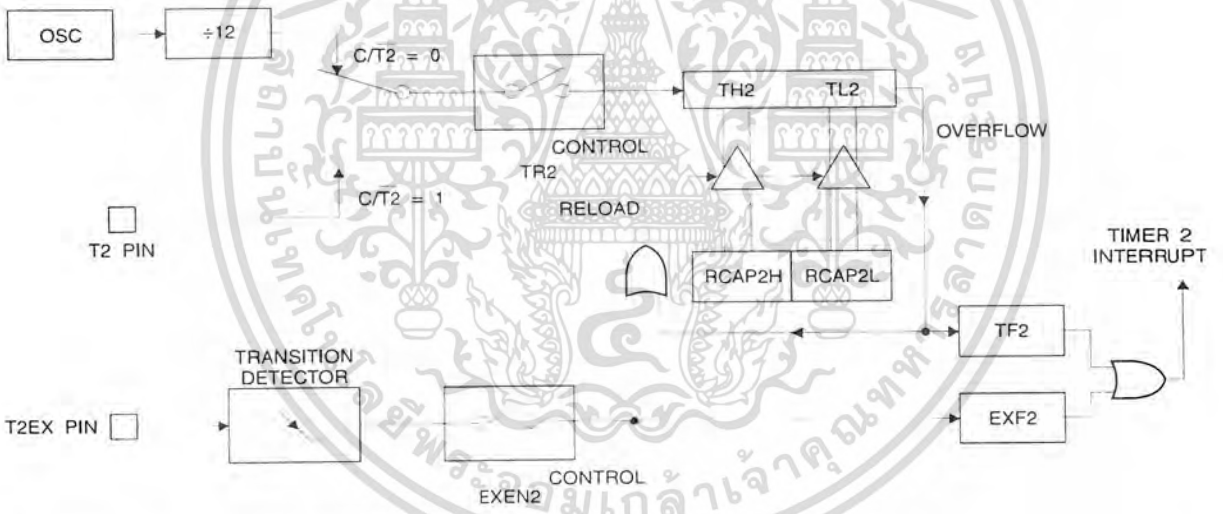


Table 9. T2MOD – Timer 2 Mode Control Register

T2MOD Address = 0C9H — Reset Value = XXXX XX00B

Not Bit Addressable

	–	–	–	–	–	–	T2OE	DCEN
Bit	7	6	5	4	3	2	1	0

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

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

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

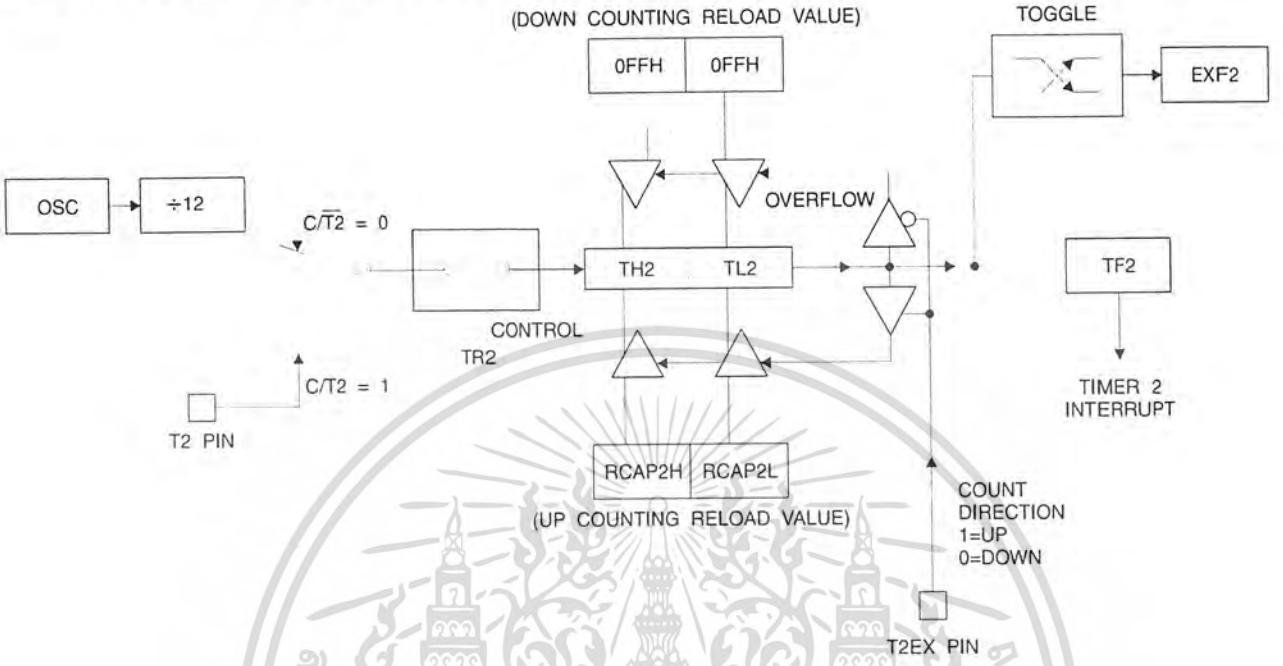
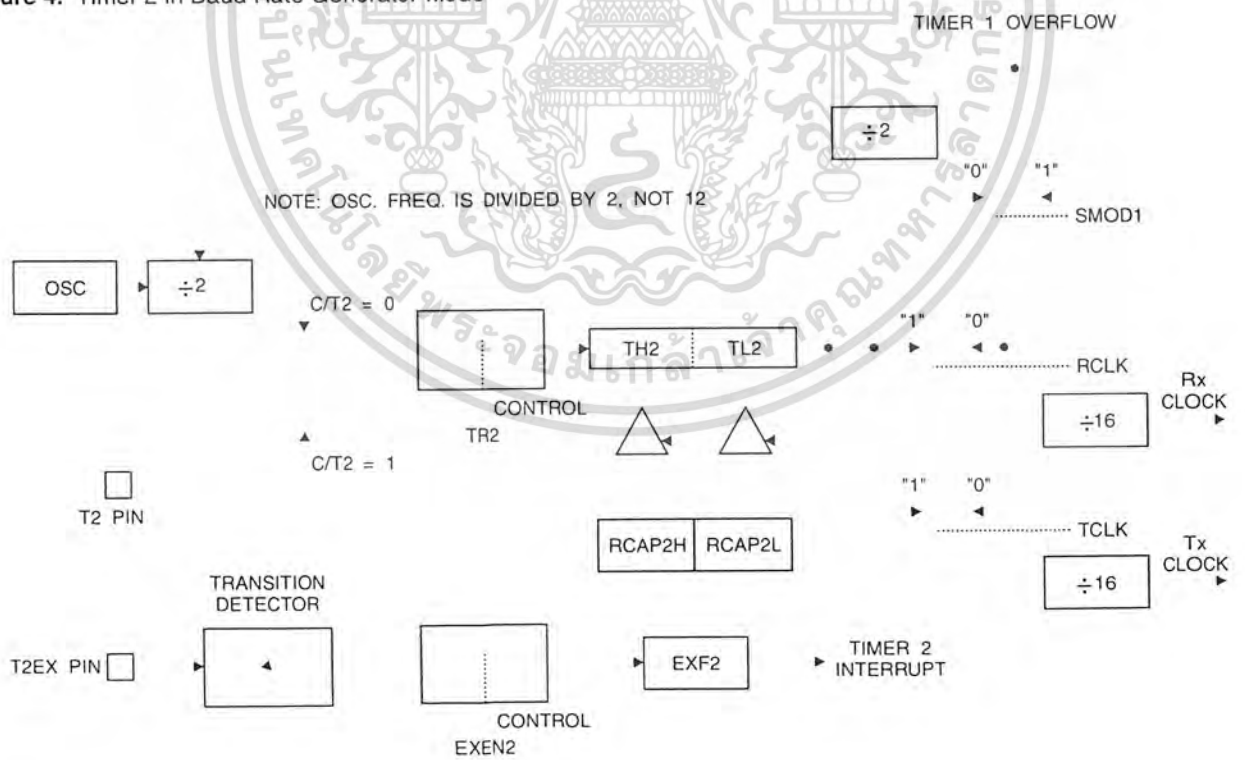


Figure 4. Timer 2 in Baud Rate Generator Mode



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Baud Rate Generator

Timer 2 is selected as the baud rate generator by setting TCLK and/or RCLK in T2CON (Table 2). Note that the baud rates for transmit and receive can be different if Timer 2 is used for the receiver or transmitter and Timer 1 is used for the other function. Setting RCLK and/or TCLK puts Timer 2 into its baud rate generator mode, as shown in Figure 4.

The baud rate generator mode is similar to the auto-reload mode, in that a rollover in TH2 causes the Timer 2 registers to be reloaded with the 16 bit value in registers RCAP2H and RCAP2L, which are preset by software.

The baud rates in Modes 1 and 3 are determined by Timer 2's overflow rate according to the following equation.

$$\text{Modes 1 and 3 Baud Rates} = \frac{\text{Timer 2 Overflow Rate}}{16}$$

The Timer can be configured for either timer or counter operation. In most applications, it is configured for timer operation ($C/\overline{T2} = 0$). The timer operation is different for Timer 2 when it is used as a baud rate generator. Normally, as a timer, it increments every machine cycle (at 1/12 the oscillator frequency). As a baud rate generator, however, it increments every state time (at 1/2 the oscillator frequency). The baud rate formula is given below.

$$\frac{\text{Modes 1 and 3}}{\text{Baud Rate}} = \frac{\text{Oscillator Frequency}}{32 \times [65536 - (\text{RCAP2H,RCAP2L})]}$$

where (RCAP2H, RCAP2L) is the content of RCAP2H and RCAP2L taken as a 16 bit unsigned integer.

Timer 2 as a baud rate generator is shown in Figure 4. This figure is valid only if RCLK or TCLK = 1 in T2CON. Note that a rollover in TH2 does not set TF2 and will not generate an interrupt. Note too, that if EXEN2 is set, a 1-to-0 transition in T2EX will set EXF2 but will not cause a reload from (RCAP2H, RCAP2L) to (TH2, TL2). Thus when Timer 2 is in use as a baud rate generator, T2EX can be used as an extra external interrupt.

Note that when Timer 2 is running (TR2 = 1) as a timer in the baud rate generator mode, TH2 or TL2 should not be read from or written to. Under these conditions, the Timer is incremented every state time, and the results of a read or write may not be accurate. The RCAP2 registers may be read but should not be written to, because a write might overlap a reload and cause write and/or reload errors. The timer should be turned off (clear TR2) before accessing the Timer 2 or RCAP2 registers.

Programmable Clock Out

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

To configure the Timer/Counter 2 as a clock generator, bit $C/\overline{T2}$ (T2CON.1) must be cleared and bit T2OE (T2MOD.1) must be set. Bit TR2 (T2CON.2) starts and stops the timer.

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

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

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

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Figure 5. Timer 2 in Clock-out Mode

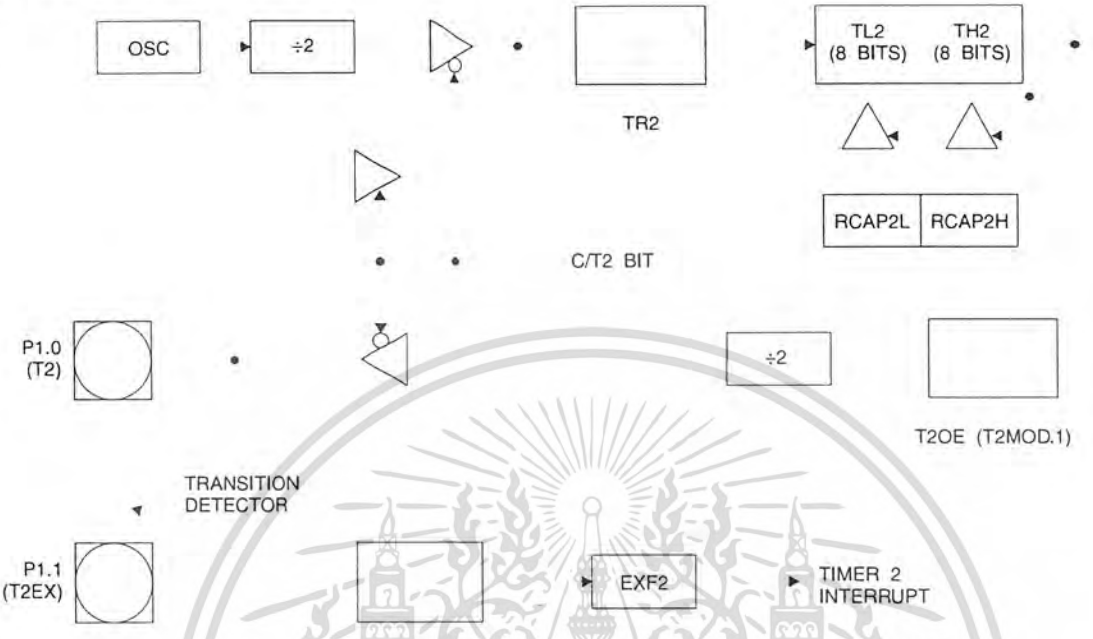
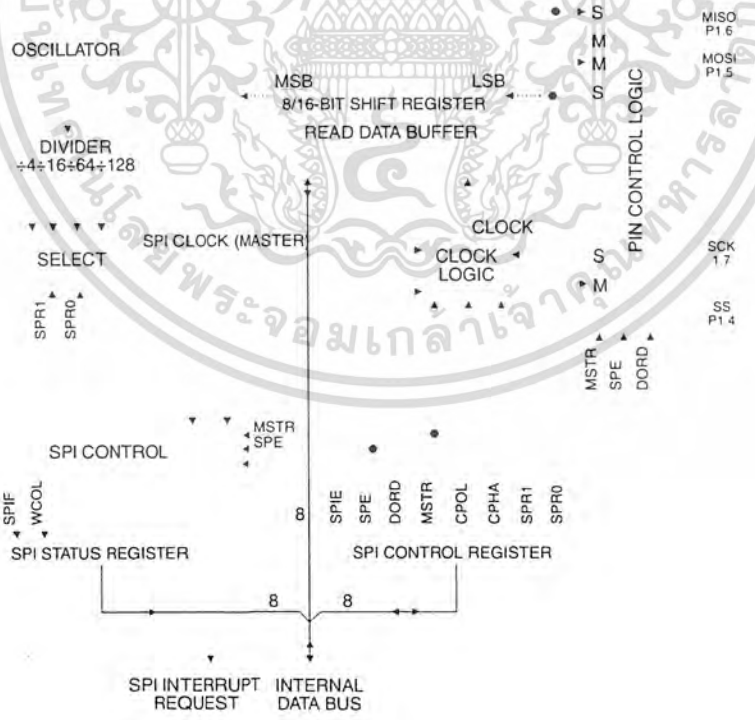


Figure 6. SPI Block Diagram



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

UART

The UART in the AT89S8252 operates the same way as the UART in the AT89C51, AT89C52 and AT89C55. For further information, see the October 1995 Microcontroller Data Book, page 2-49, section titled, "Serial Interface."

Serial Peripheral Interface

The serial peripheral interface (SPI) allows high-speed synchronous data transfer between the AT89S8252 and peripheral devices or between several AT89S8252 devices. The AT89S8252 SPI features include the following:

- Full-Duplex, 3-Wire Synchronous Data Transfer
- Master or Slave Operation
- 1.5 MHz Bit Frequency (max.)
- LSB First or MSB First Data Transfer
- Four Programmable Bit Rates
- End of Transmission Interrupt Flag
- Write Collision Flag Protection
- Wakeup from Idle Mode (Slave Mode Only)

The interconnection between master and slave CPUs with SPI is shown in the following figure. The SCK pin is the clock output in the master mode but is the clock input in the slave mode. Writing to the SPI data register of the master CPU starts the SPI clock generator, and the data written shifts out of the MOSI pin and into the MOSI pin of the slave CPU. After shifting one byte, the SPI clock generator stops, setting the end of transmission flag (SPIF). If both the SPI interrupt enable bit (SPIE) and the serial port interrupt enable bit (ES) are set, an interrupt is requested.

The Slave Select input, $\overline{SS}$/P1.4, is set low to select an individual SPI device as a slave. When $\overline{SS}$/P1.4 is set high, the SPI port is deactivated and the MOSI/P1.5 pin can be used as an input.

There are four combinations of SCK phase and polarity with respect to serial data, which are determined by control bits CPHA and CPOL. The SPI data transfer formats are shown in Figure 8 and Figure 9.

Figure 7. SPI Master-slave Interconnection

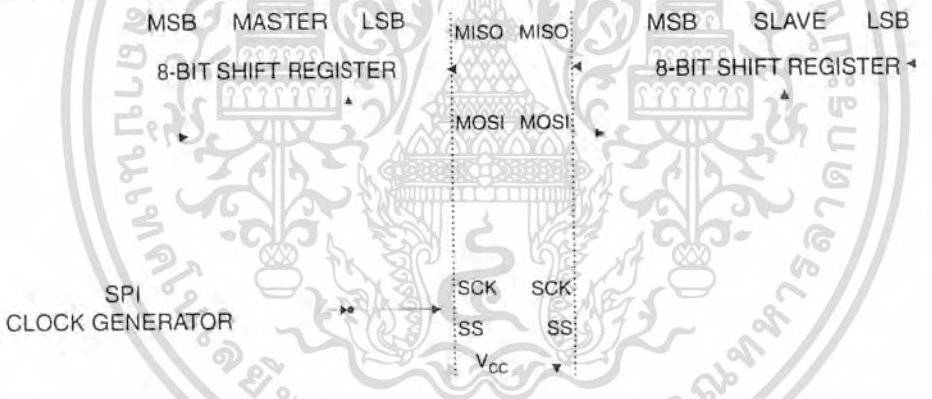
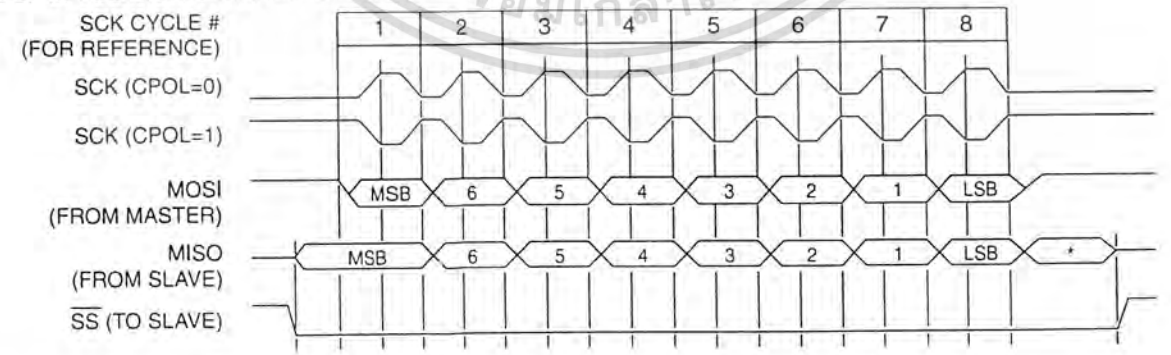


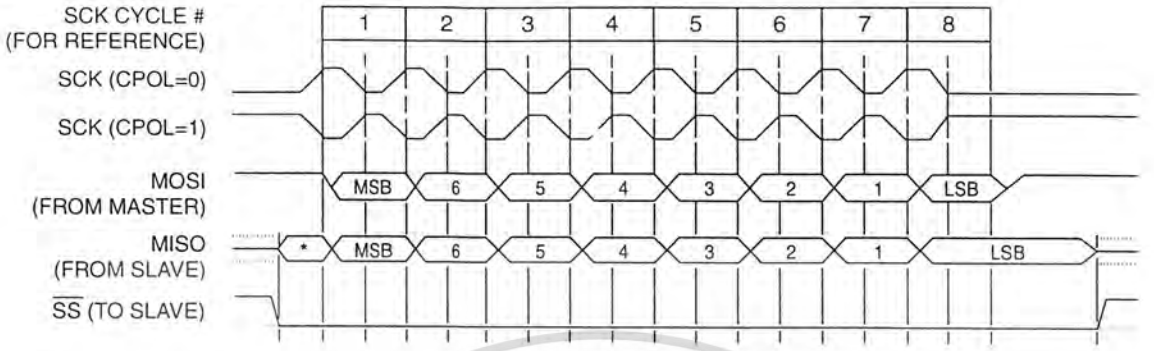
Figure 8. SPI transfer Format with CPHA = 0



*Not defined but normally MSB of character just received

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Figure 9. SPI Transfer Format with CPHA = 1



*Not defined but normally LSB of previously transmitted character

Interrupts

The AT89S8252 has a total of six interrupt vectors: two external interrupts ($\overline{INT0}$ and $\overline{INT1}$), three timer interrupts (Timers 0, 1, and 2), and the serial port interrupt. These interrupts are all shown in Figure 10.

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

Note that Table 10 shows that bit position IE.6 is unimplemented. In the AT89C51, bit position IE.5 is also unimplemented. User software should not write 1s to these bit positions, since they may be used in future AT89 products.

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

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

Table 10. Interrupt Enable (IE) Register

(MSB)(LSB)

EA	–	ET2	ES	ET1	EX1	ET0	EX0

Enable Bit = 1 enables the interrupt.

Enable Bit = 0 disables the interrupt.

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

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

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Figure 10. Interrupt Sources

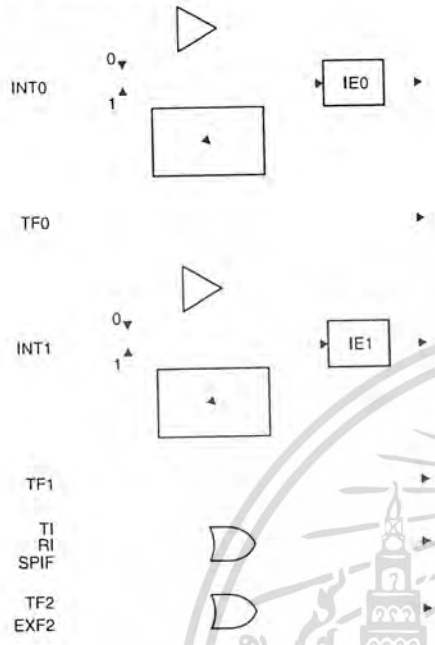
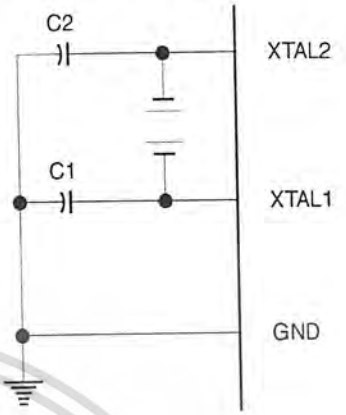
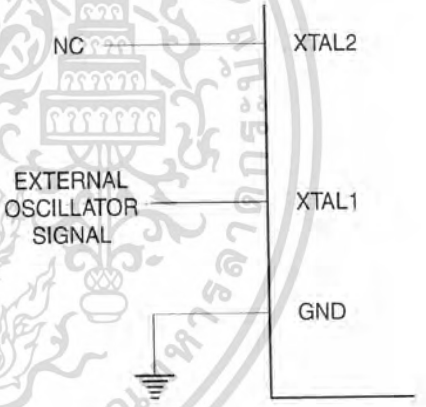


Figure 11. Oscillator Connections



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

Figure 12. External Clock Drive Configuration



Oscillator Characteristics

XTAL1 and XTAL2 are the input and output, respectively, of an inverting amplifier that can be configured for use as an on-chip oscillator, as shown in Figure 11. Either a quartz crystal or ceramic resonator may be used. To drive the device from an external clock source, XTAL2 should be left unconnected while XTAL1 is driven, as shown in Figure 12. There are no requirements on the duty cycle of the external clock signal, since the input to the internal clocking circuitry is through a divide-by-two flip-flop, but minimum and maximum voltage high and low time specifications must be observed.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Idle Mode

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

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

Status of External Pins During Idle and Power-down Modes

Mode	Program Memory	ALE	$\overline{PSEN}$	PORT0	PORT1	PORT2	PORT3
Idle	Internal	1	1	Data	Data	Data	Data
Idle	External	1	1	Float	Data	Address	Data
Power-down	Internal	0	0	Data	Data	Data	Data
Power-down	External	0	0	Float	Data	Data	Data

Power-down Mode

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

To exit power-down via an interrupt, the external interrupt must be enabled as level sensitive before entering power-down. The interrupt service routine starts at 16 ms (nominal) after the enabled interrupt pin is activated.

Program Memory Lock Bits

The AT89S8252 has three lock bits that can be left unprogrammed (U) or can be programmed (P) to obtain the additional features listed in the following table.

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

Once programmed, the lock bits can only be unprogrammed with the Chip Erase operations in either the parallel or serial modes.

Lock Bit Protection Modes[1][2]

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

Notes: 1. U = Unprogrammed
2. P = Programmed

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Programming the Flash and EEPROM

Atmel's AT89S8252 Flash Microcontroller offers 8K bytes of in-system reprogrammable Flash Code memory and 2K bytes of EEPROM Data memory.

The AT89S8252 is normally shipped with the on-chip Flash Code and EEPROM Data memory arrays in the erased state (i.e. contents = FFH) and ready to be programmed. This device supports a High-voltage (12V) Parallel programming mode and a Low-voltage (5V) Serial programming mode. The serial programming mode provides a convenient way to download the AT89S8252 inside the user's system. The parallel programming mode is compatible with conventional third party Flash or EPROM programmers.

The Code and Data memory arrays are mapped via separate address spaces in the serial programming mode. In the parallel programming mode, the two arrays occupy one contiguous address space: 0000H to 1FFFH for the Code array and 2000H to 27FFH for the Data array.

The Code and Data memory arrays on the AT89S8252 are programmed byte-by-byte in either programming mode. An auto-erase cycle is provided with the self-timed programming operation in the serial programming mode. There is no need to perform the Chip Erase operation to reprogram any memory location in the serial programming mode unless any of the lock bits have been programmed.

In the parallel programming mode, there is no auto-erase cycle. To reprogram any non-blank byte, the user needs to use the Chip Erase operation first to erase both arrays.

Parallel Programming Algorithm: To program and verify the AT89S8252 in the parallel programming mode, the following sequence is recommended:

1. Power-up sequence:

   Apply power between V_{CC} and GND pins.

   Set RST pin to "H".

   Apply a 3 MHz to 24 MHz clock to XTAL1 pin and wait for at least 10 milliseconds.
2. Set $\overline{PSEN}$ pin to "L"

   ALE pin to "H"

   $\overline{EA}$ pin to "H" and all other pins to "H".
3. Apply the appropriate combination of "H" or "L" logic levels to pins P2.6, P2.7, P3.6, P3.7 to select one of the programming operations shown in the Flash Programming Modes table.
4. Apply the desired byte address to pins P1.0 to P1.7 and P2.0 to P2.5.

   Apply data to pins P0.0 to P0.7 for Write Code operation.
5. Raise $\overline{EA}/V_{PP}$ to 12V to enable Flash programming, erase or verification.
6. Pulse ALE/$\overline{PROG}$ once to program a byte in the Code memory array, the Data memory array or the lock bits. The byte-write cycle is self-timed and typically takes 1.5 ms.
7. To verify the byte just programmed, bring pin P2.7 to "L" and read the programmed data at pins P0.0 to P0.7.
8. Repeat steps 3 through 7 changing the address and data for the entire 2K or 8K bytes array or until the end of the object file is reached.
9. Power-off sequence:

   Set XTAL1 to "L".

   Set RST and $\overline{EA}$ pins to "L".

   Turn V_{CC} power off.

In the parallel programming mode, there is no auto-erase cycle and to reprogram any non-blank byte, the user needs to use the Chip Erase operation first to erase both arrays.

Data Polling: The AT89S8252 features $\overline{DATA}$ Polling to indicate the end of a write cycle. During a write cycle in the parallel or serial programming mode, an attempted read of the last byte written will result in the complement of the written datum on P0.7 (parallel mode), and on the MSB of the serial output byte on MISO (serial mode). Once the write cycle has been completed, true data are valid on all outputs, and the next cycle may begin. $\overline{DATA}$ Polling may begin any time after a write cycle has been initiated.

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

Program Verify: If lock bits LB1 and LB2 have not been programmed, the programmed Code or Data byte can be read back via the address and data lines for verification. The state of the lock bits can also be verified directly in the parallel programming mode. In the serial programming mode, the state of the lock bits can only be verified indirectly by observing that the lock bit features are enabled.

Chip Erase: Both Flash and EEPROM arrays are erased electrically at the same time. In the parallel programming mode, chip erase is initiated by using the proper combination of control signals and by holding ALE/$\overline{PROG}$ low for 10 ms. The Code and Data arrays are written with all "1"s in the Chip Erase operation.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

In the serial programming mode, a chip erase operation is initiated by issuing the Chip Erase instruction. In this mode, chip erase is self-timed and takes about 16 ms.

During chip erase, a serial read from any address location will return 00H at the data outputs.

Serial Programming Fuse: A programmable fuse is available to disable Serial Programming if the user needs maximum system security. The Serial Programming Fuse can only be programmed or erased in the Parallel Programming Mode.

The AT89S8252 is shipped with the Serial Programming Mode enabled.

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

(030H) = 1EH indicates manufactured by Atmel
(031H) = 72H indicates 89S8252

Programming Interface

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

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

Serial Downloading

Both the Code and Data memory arrays can be programmed using the serial SPI bus while RST is pulled to V_{CC}. The serial interface consists of pins SCK, MOSI (input) and MISO (output). After RST is set high, the Programming Enable instruction needs to be executed first before program/erase operations can be executed.

An auto-erase cycle is built into the self-timed programming operation (in the serial mode ONLY) and there is no need to first execute the Chip Erase instruction unless any of the lock bits have been programmed. The Chip Erase operation turns the content of every memory location in both the Code and Data arrays into FFH.

The Code and Data memory arrays have separate address spaces:

0000H to 1FFFH for Code memory and 000H to 7FFH for Data memory.

Either an external system clock is supplied at pin XTAL1 or a crystal needs to be connected across pins XTAL1 and XTAL2. The maximum serial clock (SCK) frequency should be less than 1/40 of the crystal frequency. With a 24 MHz oscillator clock, the maximum SCK frequency is 600 kHz.

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 VCC 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:

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

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	x x111	xxxx xxxx	Write lock bits. Set LB1, LB2 or LB3 = "0" to program lock bits.

Note: 1. 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 อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

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 $\overline{RDY/BSY}$.
4. "X" = don't care

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Figure 13. Programming the Flash/EEPROM Memory

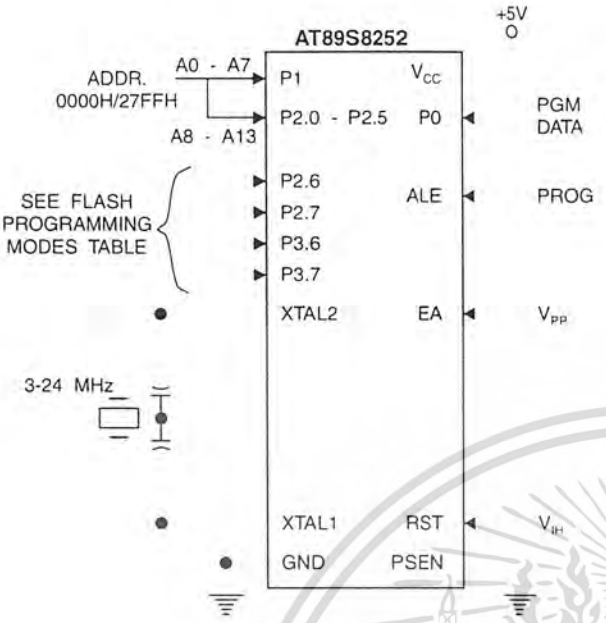


Figure 14. Verifying the Flash/EEPROM Memory

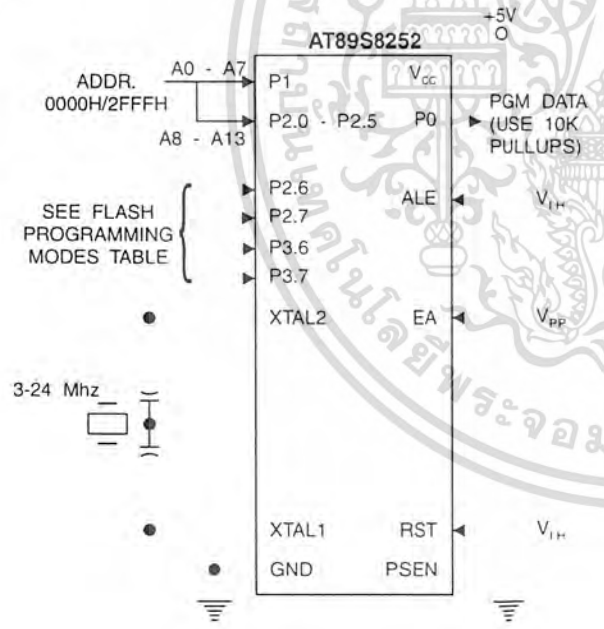
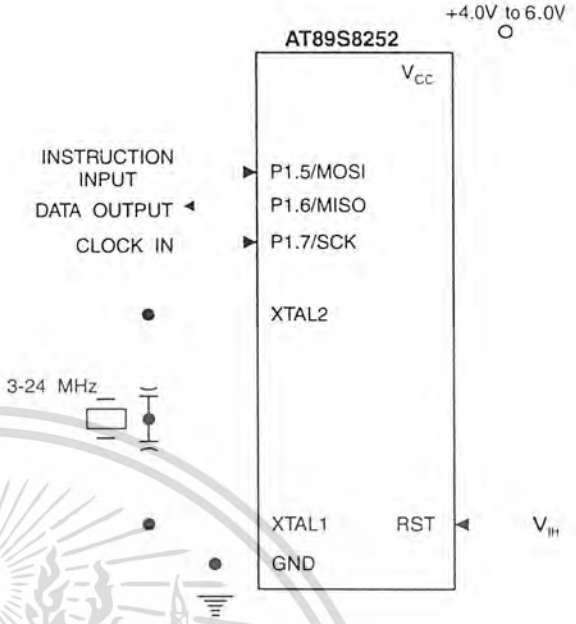


Figure 15. Flash/EEPROM Serial Downloading



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

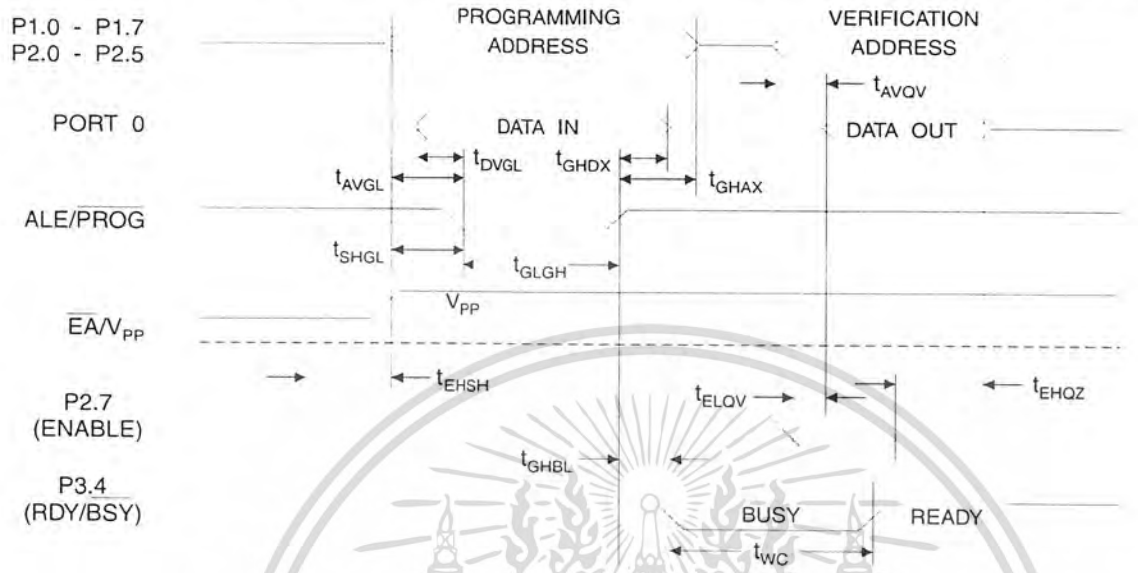
Flash Programming and Verification Characteristics – Parallel Mode

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

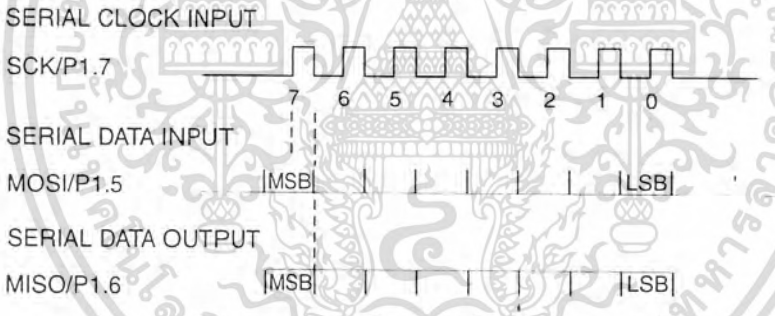
Symbol	Parameter	Min	Max	Units
V_{PP}	Programming Enable Voltage	11.5	12.5	V
I_{PP}	Programming Enable Current		1.0	mA
$1/t_{CLCL}$	Oscillator Frequency	3	24	MHz
t_{AVGL}	Address Setup to $\overline{PROG}$ Low	$48t_{CLCL}$		
t_{GHAX}	Address Hold after $\overline{PROG}$	$48t_{CLCL}$		
t_{DVGL}	Data Setup 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 อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Flash/EEPROM Programming and Verification Waveforms – Parallel Mode



Serial Downloading Waveforms



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Absolute Maximum Ratings*

Operating Temperature	-55°C to +125°C
Storage Temperature	-65°C to +150°C
Voltage on Any Pin with Respect to Ground	-1.0V 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 Hifh-voltage	(Except XTAL1, RST)	0.2 V_{CC} + 0.9	V_{CC} + 0.5	V
V_{IH1}	Input Hifh-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 Hifh-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 Hifh-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 Pull-down 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 อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

AC Characteristics

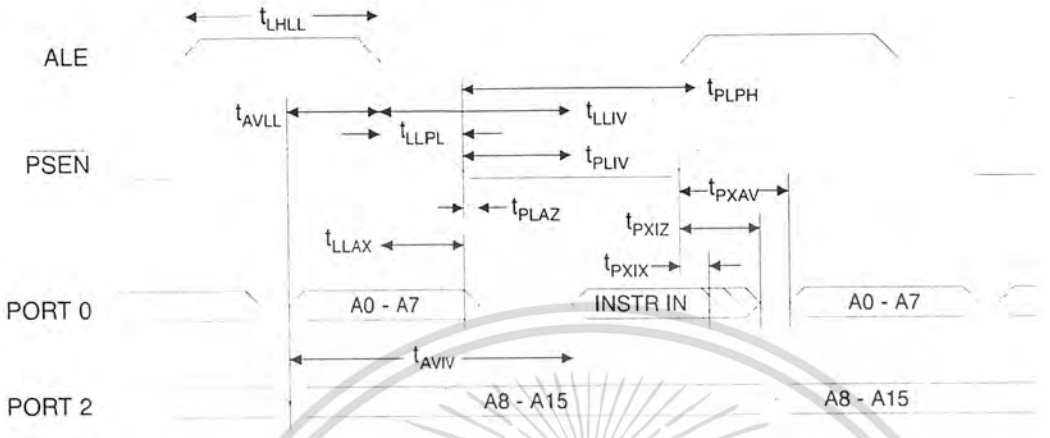
Under operating conditions, load capacitance for Port 0, ALE/$\overline{PROG}$, and $\overline{PSEN}$ = 100 pF; load capacitance for all other outputs = 80 pF.

External Program and Data Memory Characteristics

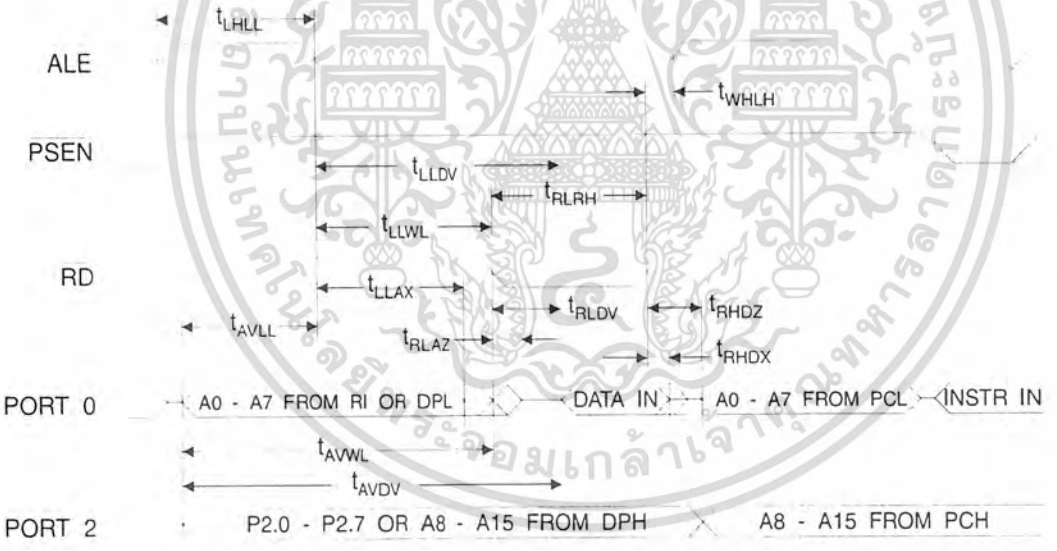
Symbol	Parameter	Variable Oscillator		Units
		Min	Max	
$1/t_{CLCL}$	Oscillator Frequency	0	24	MHz
t_{LHLL}	ALE Pulse Width	$2t_{CLCL}$ - 40		ns
t_{AVLL}	Address Valid to ALE Low	t_{CLCL} - 13		ns
t_{LLAX}	Address Hold after ALE Low	t_{CLCL} - 20		ns
t_{LLIV}	ALE Low to Valid Instruction In		$4t_{CLCL}$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

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

External Program Memory Read Cycle

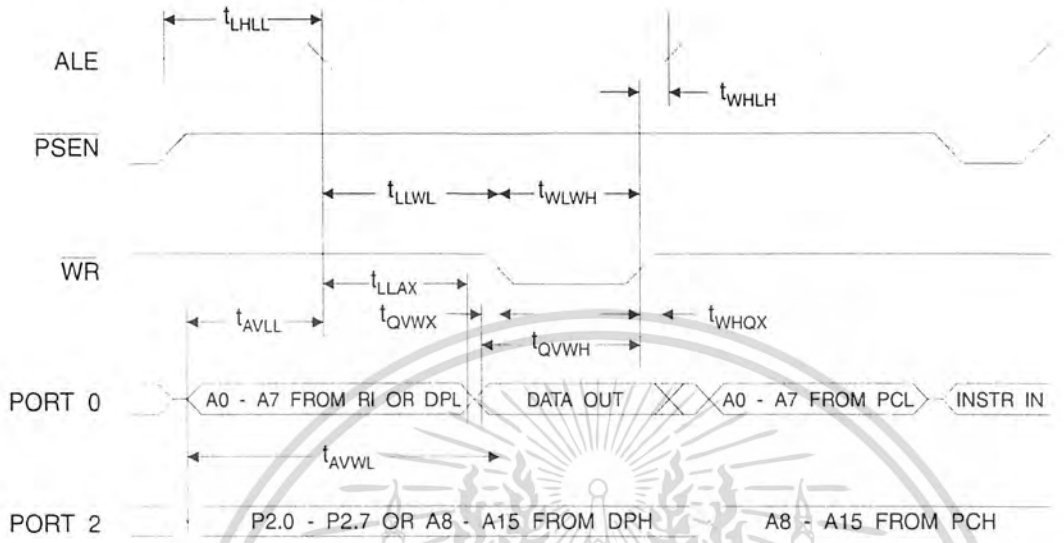


External Data Memory Read Cycle

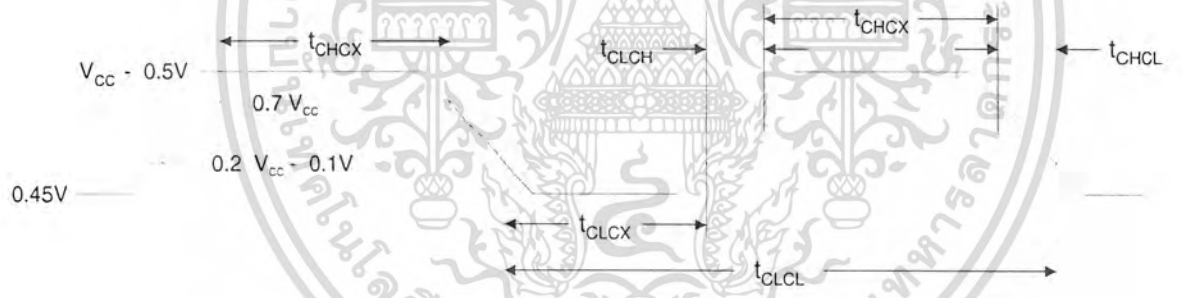


เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

External Data Memory Write Cycle



External Clock Drive Waveforms



External Clock Drive

Symbol	Parameter	V_{CC} = 4.0V to 6.0V		Units
		Min	Max	
$1/t_{CLCL}$	Oscillator Frequency	0	24	MHz
t_{CLCL}	Clock Period	41.6		ns
t_{CHCX}	High Time	15		ns
t_{CLCX}	Low Time	15		ns
t_{CLCH}	Rise Time		20	ns
t_{CHCL}	Fall Time		20	ns

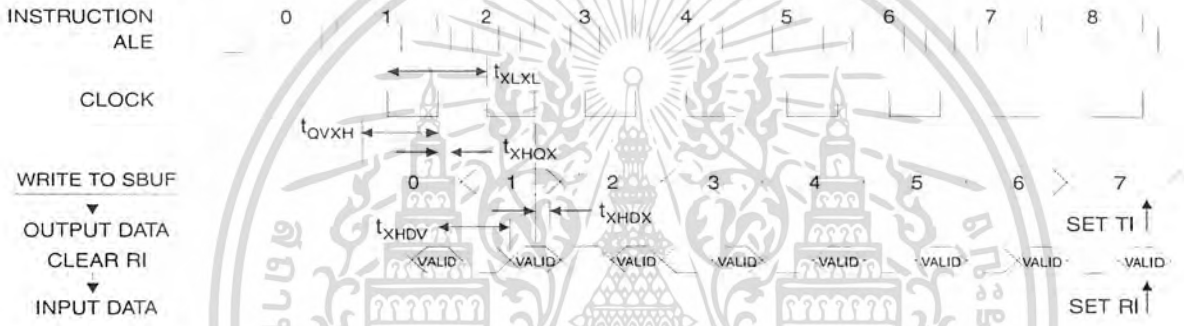
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

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

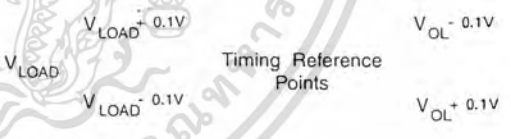
Shift Register Mode Timing Waveforms



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]

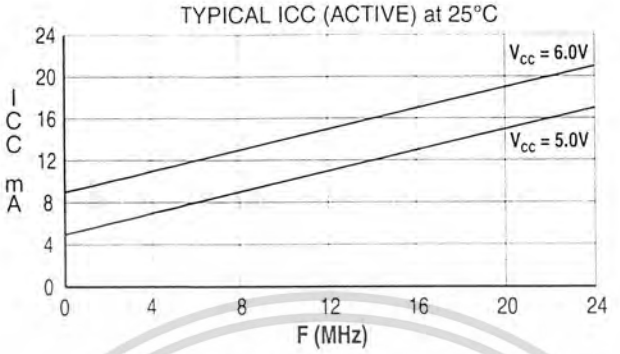


Notes: 1. For timing purposes, a port pin is no longer floating when a 100 mV change from load voltage occurs. A port pin begins to float when a 100 mV change from the loaded V_{OH}/V_{OL} level occurs.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

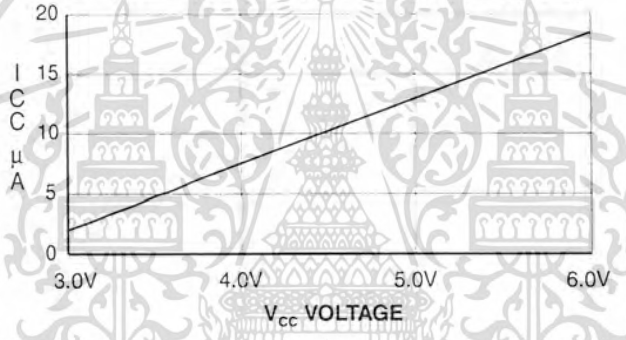
AT89S8252

TYPICAL ICC (ACTIVE) at 25°C



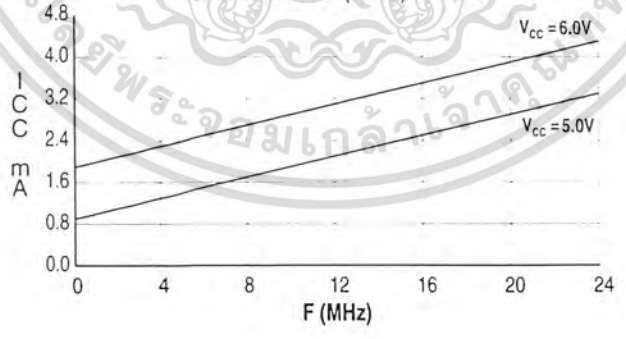
AT89S8252

TYPICAL ICC vs. VOLTAGE - POWER DOWN (85°C)



AT89S8252

TYPICAL ICC (IDLE) at 25°C



Notes: 1. XTAL1 tied to GND for Icc (power-down)
2. Lock bits programmed

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Ordering Information

Speed (MHz)	Power Supply	Ordering Code	Package	Operation Range
24	4.0V to 6.0V	AT89S8252-24AC AT89S8252-24JC AT89S8252-24PC AT89S8252-24QC	44A 44J 40P6 44Q	Commercial (0°C to 70°C)
	4.0V to 6.0V	AT89S8252-24AI AT89S8252-24JI AT89S8252-24PI AT89S8252-24QI	44A 44J 40P6 44Q	Industrial (-40°C to 85°C)
33	4.5V to 5.5V	AT89S8252-33AC AT89S8252-33JC AT89S8252-33PC AT89S8252-33QC	44A 44J 40P6 44Q	Commercial (0°C to 70°C)

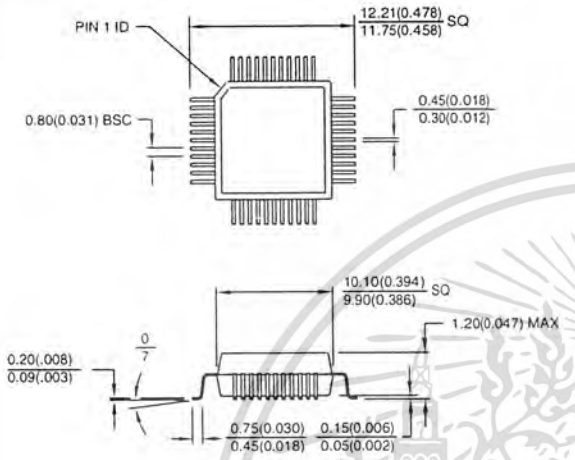
= Preliminary Information

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

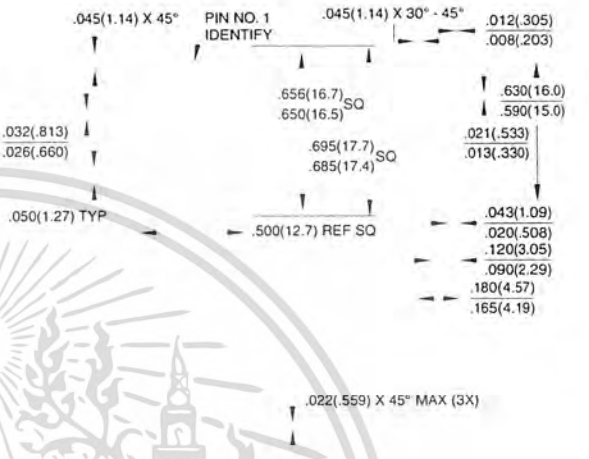
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Packaging Information

44A, 44-lead, Thin (1.0 mm) Plastic Gull Wing Quad Flatpack (TQFP)
Dimensions in Millimeters and (Inches)*
JEDEC STANDARD MS-026 ACB

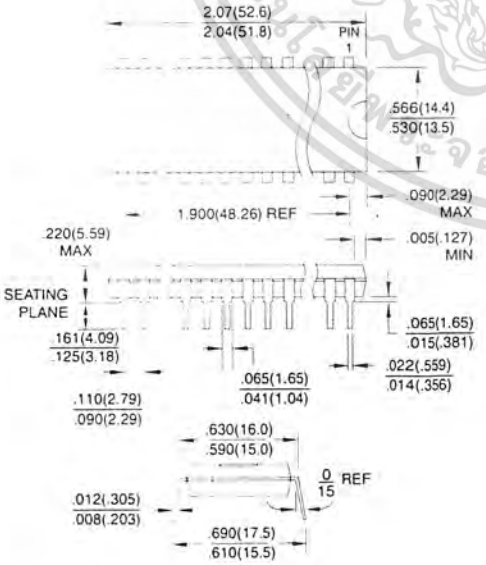


44J, 44-lead, Plastic J-leaded Chip Carrier (PLCC)
Dimensions in Inches and (Millimeters)
JEDEC STANDARD MS-018 AC

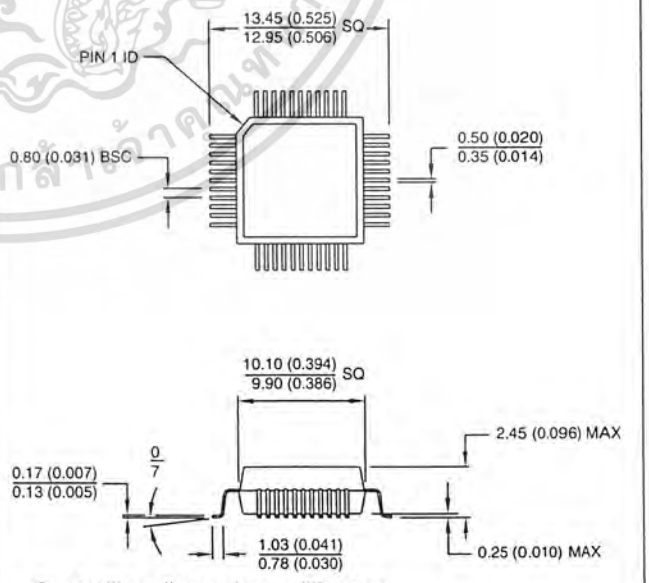


Controlling dimension: millimeters

40P6, 40-lead, 0.600" Wide, Plastic Dual Inline Package (PDIP)
Dimensions in Inches and (Millimeters)



44Q, 44-lead, Plastic Quad Flat Package (PQFP)
Dimensions in Millimeters and (Inches)*
JEDEC STANDARD MS-022 AB



Controlling dimension: millimeters

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Atmel Headquarters

Corporate Headquarters
2325 Orchard Parkway
San Jose, CA 95131
TEL (408) 441-0311
FAX (408) 487-2600

Europe
Atmel U.K., Ltd.
Coliseum Business Centre
Riverside Way
Camberley, Surrey GU15 3YL
England
TEL (44) 1276-686-677
FAX (44) 1276-686-697

Asia
Atmel Asia, Ltd.
Room 1219
Chinachem Golden Plaza
77 Mody Road Tsimhatsui
East Kowloon
Hong Kong
TEL (852) 2721-9778
FAX (852) 2722-1369

Japan
Atmel Japan K.K.
9F, Tonetsu Shinkawa Bldg.
1-24-8 Shinkawa
Chuo-ku, Tokyo 104-0033
Japan
TEL (81) 3-3523-3551
FAX (81) 3-3523-7581

Atmel Operations

Atmel Colorado Springs
1150 E. Cheyenne Mtn. Blvd.
Colorado Springs, CO 80906
TEL (719) 576-3300
FAX (719) 540-1759

Atmel Rousset
Zone Industrielle
13106 Rousset Cedex
France
TEL (33) 4-4253-6000
FAX (33) 4-4253-6001

Fax-on-Demand
North America:
1-(800) 292-8635
International:
1-(408) 441-0732

e-mail
literature@atmel.com

Web Site
http://www.atmel.com

BBS
1-(408) 436-4309

© Atmel Corporation 2000.
Atmel Corporation makes no warranty for the use of its products, other than those expressly contained in the Company's standard warranty which is detailed in Atmel's Terms and Conditions located on the Company's web site. The Company assumes no responsibility for any errors which may appear in this document, reserves the right to change devices or specifications detailed herein at any time without notice, and does not make any commitment to update the information contained herein. No licenses to patents or other intellectual property of Atmel are granted by the Company in connection with the sale of Atmel products, expressly or by implication. Atmel's products are not authorized for use as critical components in life support devices or systems.

Marks bearing ® and/or ™ are registered trademarks and trademarks of Atmel Corporation.

Terms and product names in this document may be trademarks of others.

Printed on recycled paper.

0401E–02/00/xM

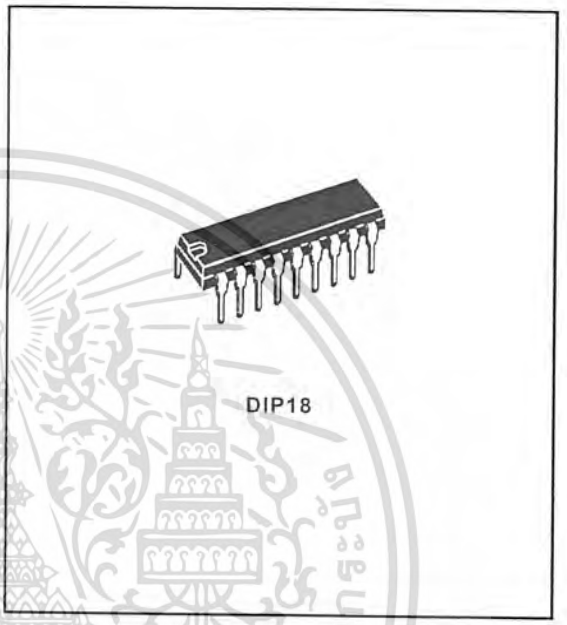
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

SGS-THOMSON MICROELECTRONICS

ULN2801A
ULN2802A - ULN2803A
ULN2804A - ULN2805A

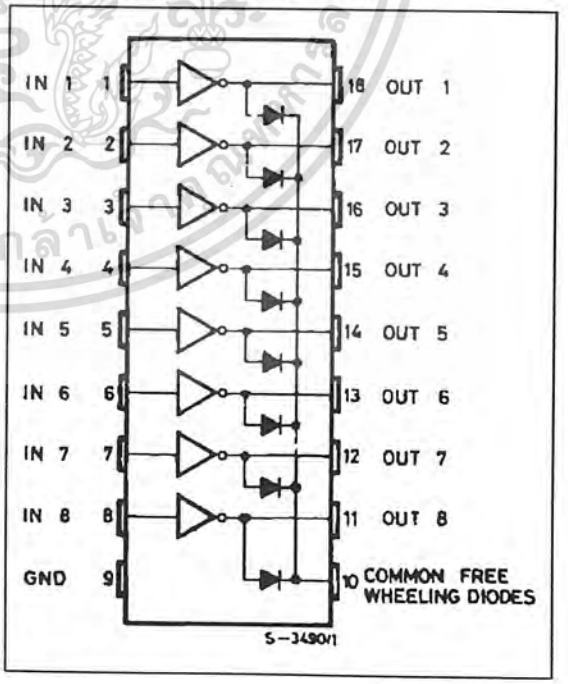
EIGHT DARLINGTON ARRAYS

- EIGHT DARLINGTONS WITH COMMON EMITTERS
- OUTPUT CURRENT TO 500 mA
- OUTPUT VOLTAGE TO 50 V
- INTEGRAL SUPPRESSION DIODES
- VERSIONS FOR ALL POPULAR LOGIC FAMILIES
- OUTPUT CAN BE PARALLELED
- INPUTS PINNED OPPOSITE OUTPUTS TO SIMPLIFY BOARD LAYOUT



DIP18

PIN CONNECTION (top view)



DESCRIPTION

The ULN2801A-ULN2805A each contain eight darlington transistors with common emitters and integral suppression diodes for inductive loads. Each darlington features a peak load current rating of 600mA (500mA continuous) and can withstand at least 50V in the off state. Outputs may be paralleled for higher current capability.

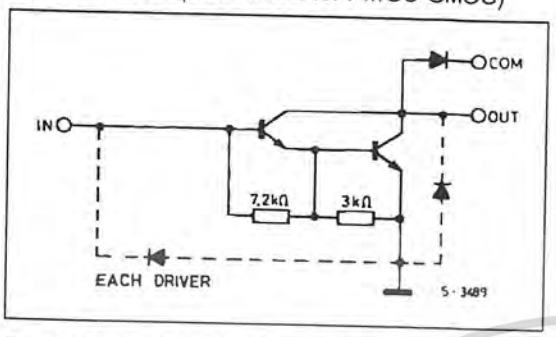
Five versions are available to simplify interfacing to standard logic families : the ULN2801A is designed for general purpose applications with a current limit resistor ; the ULN2802A has a 10.5kΩ input resistor and zener for 14-25V PMOS ; the ULN2803A has a 2.7kΩ input resistor for 5V TTL and CMOS ; the ULN2804A has a 10.5kΩ input resistor for 6-15V CMOS and the ULN2805A is designed to sink a minimum of 350mA for standard and Schottky TTL where higher output current is required.

All types are supplied in a 18-lead plastic DIP with a copper lead from and feature the convenient input-opposite-output pinout to simplify board layout.

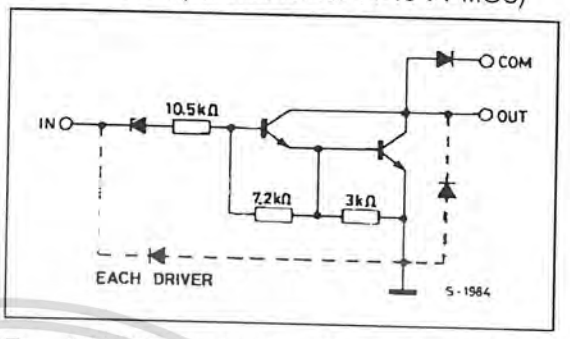
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

SCHEMATIC DIAGRAM AND ORDER CODES

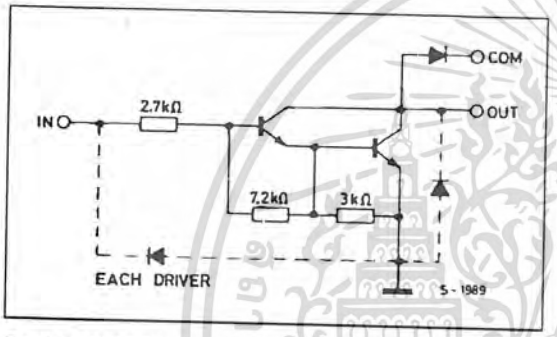
For ULN2801A (each driver for PMOS-CMOS)



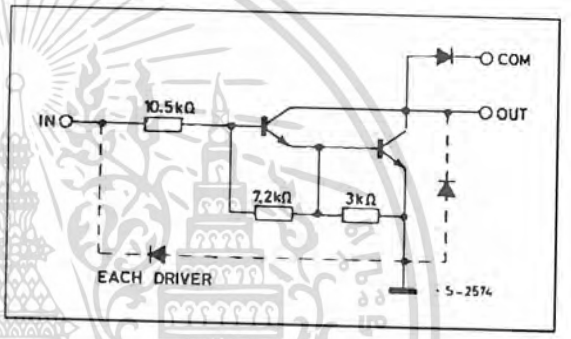
For ULN2802A (each driver for 14-15 V PMOS)



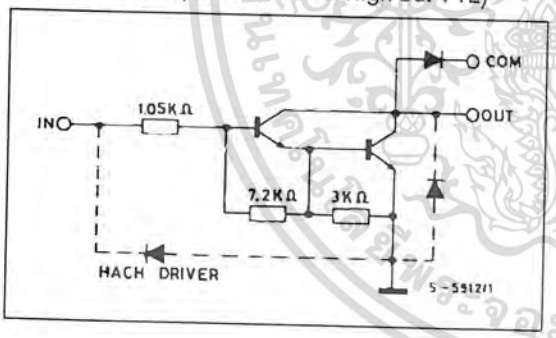
For ULN2803A (each driver for 5 V, TTL/CMOS)



For ULN2804A (each driver for 6-15 V CMOS/PMOS



For ULN2805A (each driver for high out TTL)



SGS-THOMSON MICROELECTRONICS

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ABSOLUTE MAXIMUM RATINGS

Symbol	Parameter	Value	Unit
V_o	Output Voltage	50	V
V_i	Input Voltage for ULN2802A, UL2803A, ULN2804A for ULN2805A	 30 15	V
I_C	Continuous Collector Current	500	mA
I_B	Continuous Base Current	25	mA
P_{tot}	Power Dissipation (one Darlington pair) (total package)	 1.0 2.25	W
T_{amb}	Operating Ambient Temperature Range	– 20 to 85	°C
T_{stg}	Storage Temperature Range	– 55 to 150	°C
Tj	Junction Temperature Range	– 20 to 150	°C

THERMAL DATA

Symbol	Parameter		Value	Unit
$R_{th\ j\text{-}amb}$	Thermal Resistance Junction-ambient	Max.	55	°C/W

ELECTRICAL CHARACTERISTICS (T_{amb} = 25°C unless otherwise specified)

Symbol	Parameter	Test Conditions	Min.	Typ.	Max.	Unit	Fig.
I_{CEX}	Output Leakage Current	V_{CE} = 50V			50	µA	1a
		T_{amb} = 70°C, V_{CE} = 50V			100	µA	1a
		T_{amb} = 70°C for ULN2802A V_{CE} = 50V, V_i = 6V			500	µA	1b
		for ULN2804A V_{CE} = 50V, V_i = 1V			500	µA	1b
$V_{CE(sat)}$	Collector-emitter Saturation Voltage	I_C = 100mA, I_B = 250µA		0.9	1.1	V	2
		I_C = 200mA, I_B = 350µA		1.1	1.3	V	
		I_C = 350mA, I_B = 500µA		1.3	1.6	V	
$I_{i(on)}$	Input Current	for ULN2802A V_i = 17V		0.82	1.25	mA	3
		for ULN2803A V_i = 3.85V		0.93	1.35	mA	
		for ULN2804A V_i = 5V		0.35	0.5	mA	
		V_i = 12V		1	1.45	mA	
		for ULN2805A V_i = 3V		1.5	2.4	mA	
$I_{i(off)}$	Input Current	T_{amb} = 70°C, I_C = 500µA	50	65		µA	4
$V_{i(on)}$	Input Voltage	V_{CE} = 2 V for ULN2802A I_C = 300mA			13	V	5
		for ULN2803A I_C = 200mA			2.4	V	
		I_C = 250mA			2.7	V	
		I_C = 300mA			3	V	
		for ULN2804A I_C = 125mA			5	V	
		I_C = 200mA			6	V	
		I_C = 275mA			7	V	
		I_C = 350mA			8	V	
		for ULN2805A I_C = 350mA			2.4	V	
h_{FE}	DC Forward Current Gain	for ULN2801A V_{CE} = 2V I_C = 350mA	1000			–	2
C_i	Input Capacitance			15	25	pF	–
t_{PLH}	Turn-on Delay Time	0.5 V_i to 0.5 V_o		0.25	1	µs	–
t_{PHL}	Turn-off Delay Time	0.5 V_i to 0.5 V_o		0.25	1	µs	–
I_R	Clamp Diode Leakage Current	V_R = 50V			50	µA	6
		T_{amb} = 70°C, V_R = 50V			100	µA	6
V_F	Clamp Diode Forward Voltage	I_F = 350mA		1.7	2	V	7

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

TEST CIRCUITS

Figure 1a.

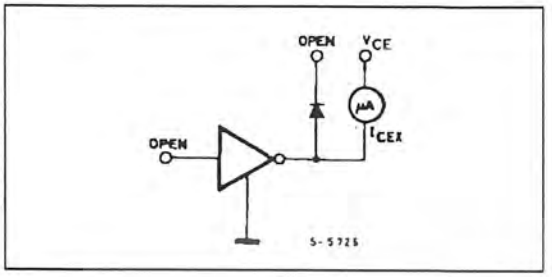


Figure 1b.

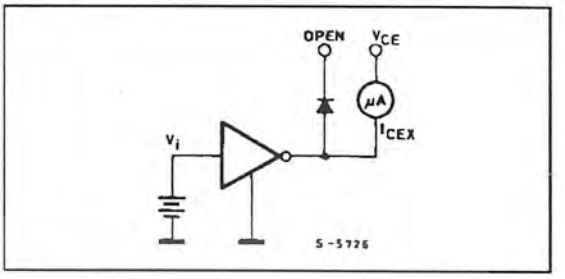


Figure 2.

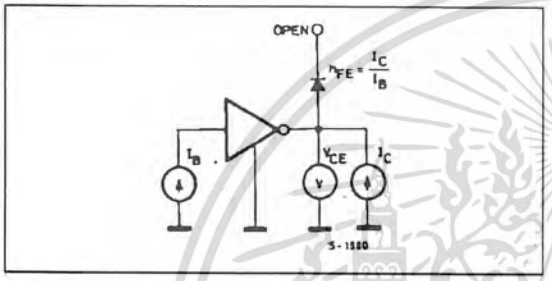


Figure 3.

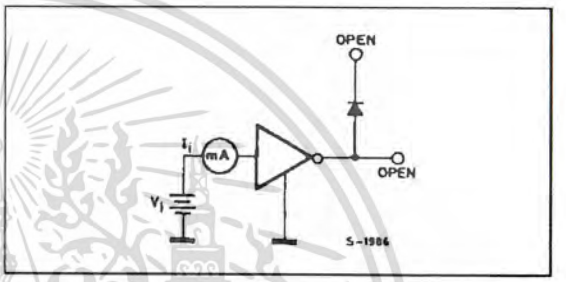


Figure 4.

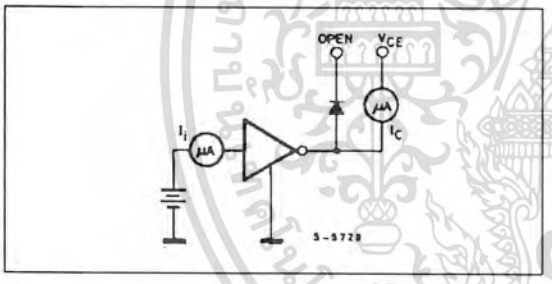


Figure 5.

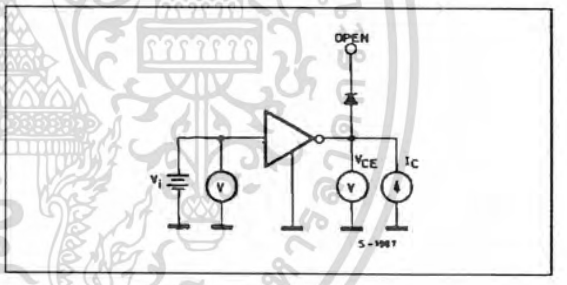


Figure 6.

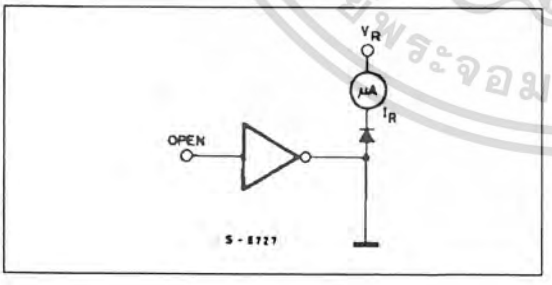
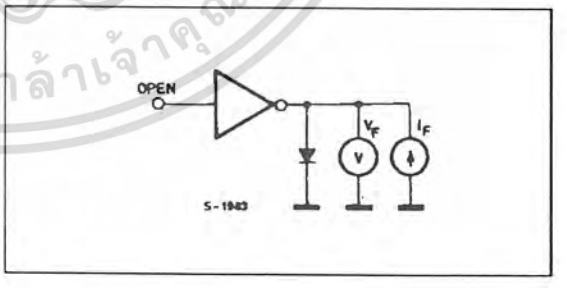


Figure 7.



SGS-THOMSON MICROELECTRONICS

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Figure 8 : Collector Current as a Function of Saturation Voltage.

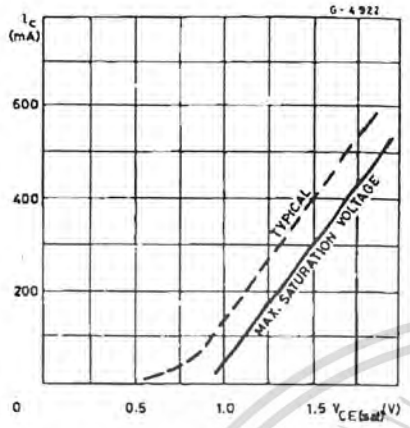


Figure 9 : Collector Current as a Function of Input Current.

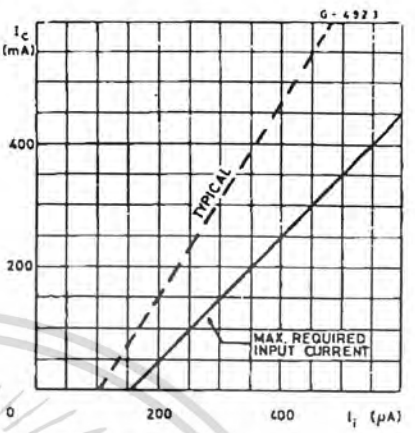


Figure 10 : Allowable Average Power Dissipation as a Function of Ambient Temperature.

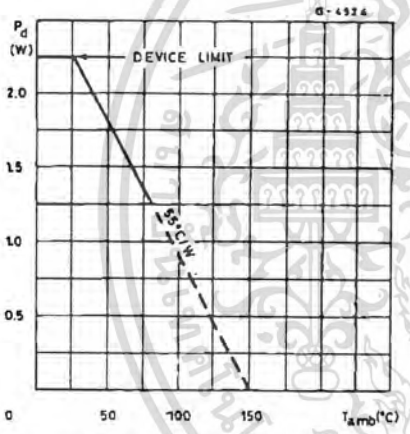


Figure 11 : Peak Collector Current as a Function of Duty Cycle.

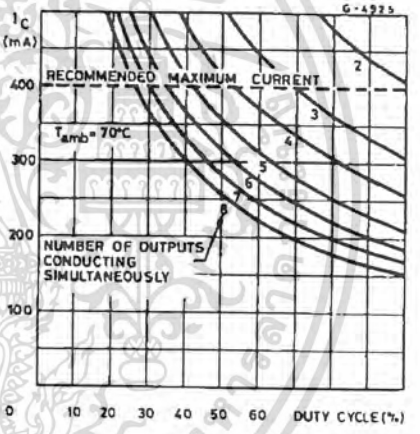


Figure 12 : Peak Collector Current as a Function of Duty.

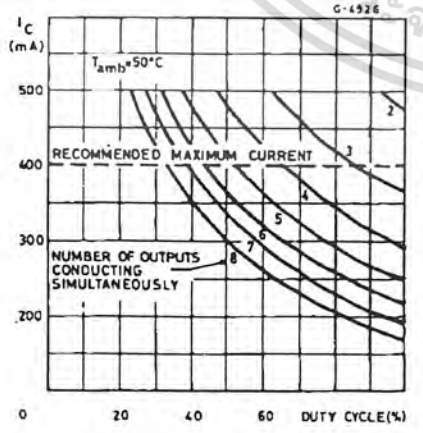
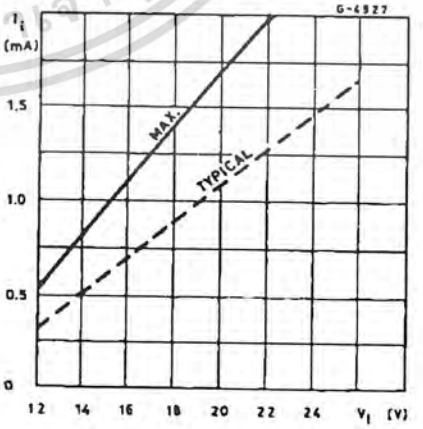


Figure 13 : Input Current as a Function of Input Voltage (for ULN2802A).



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Figure 14 : Input Current as a Function of Input Voltage (for ULN2804A)

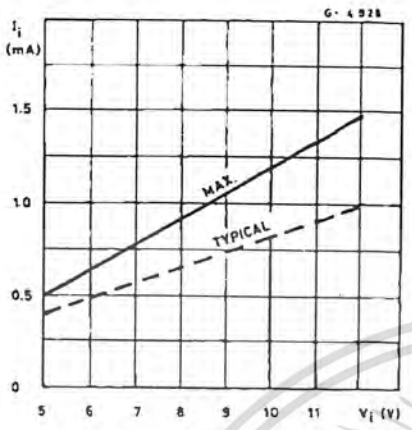


Figure 15 : Input Current as a Function of Input Voltage (for ULN2803A)

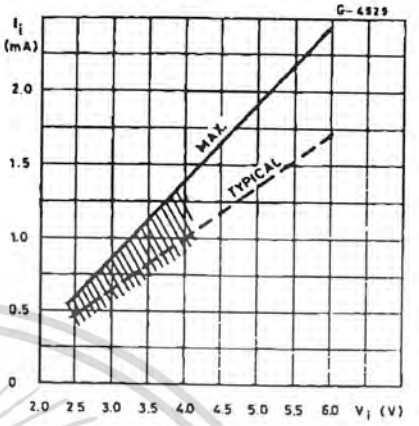
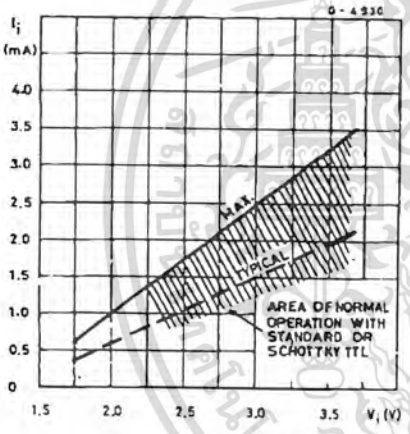


Figure 16 : Input Current as a Function of Input Voltage (for ULN2805A)

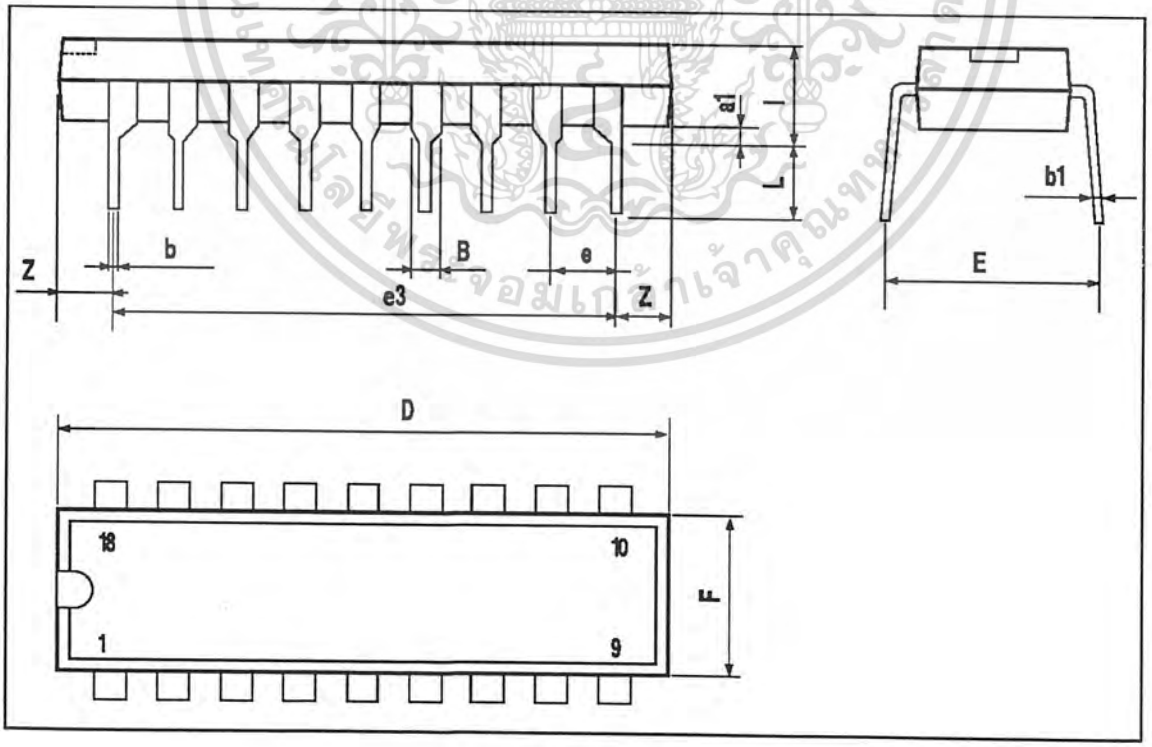


SGS-THOMSON MICROELECTRONICS

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

DIP18 PACKAGE MECHANICAL DATA

DIM.	mm			inch		
	MIN.	TYP.	MAX.	MIN.	TYP.	MAX.
a1	0.254			0.010		
B	1.39		1.65	0.055		0.065
b		0.46			0.018	
b1		0.25			0.010	
D			23.24			0.915
E		8.5			0.335	
e		2.54			0.100	
e3		20.32			0.800	
F			7.1			0.280
I			3.93			0.155
L		3.3			0.130	
Z		1.27	1.59		0.050	0.063



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Information furnished is believed to be accurate and reliable. However, SGS-THOMSON Microelectronics assumes no responsibility for the consequences of use of such information nor for any infringement of patents or other rights of third parties which may result from its use. No license is granted by implication or otherwise under any patent or patent rights of SGS-THOMSON Microelectronics. Specification mentioned in this publication are subject to change without notice. This publication supersedes and replaces all information previously supplied. SGS-THOMSON Microelectronics products are not authorized for use as critical components in life support devices or systems without express written approval of SGS-THOMSON Microelectronics.

© 1997 SGS-THOMSON Microelectronics – Printed in Italy – All Rights Reserved
SGS-THOMSON Microelectronics GROUP OF COMPANIES
Australia - Brazil - Canada - China - France - Germany - Hong Kong - Italy - Japan - Korea - Malaysia - Malta - Morocco - The Netherlands - Singapore - Spain - Sweden - Switzerland - Taiwan - Thailand - United Kingdom - U.S.A.

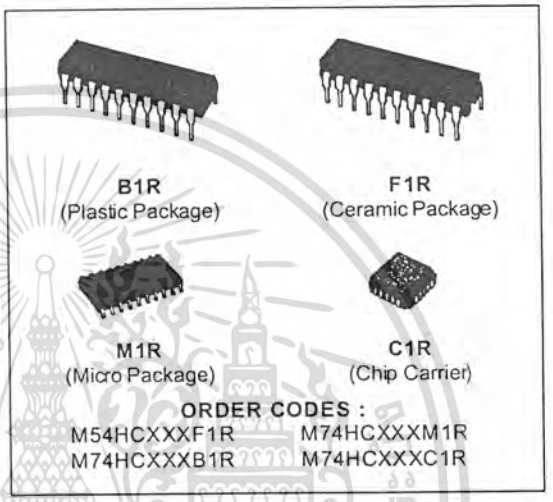
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

SGS-THOMSON MICROELECTRONICS

M54/74HC563
M54/74HC573

OCTAL D-TYPE LATCH WITH 3 STATE OUTPUT
HC563 INVERTING - HC573 NON INVERTING

- HIGH SPEED
  t_{PD} = 13 ns (TYP.) AT V_{CC} = 5 V
- LOW POWER DISSIPATION
  I_{CC} = 4 µA (MAX.) AT T_A = 25 °C
- HIGH NOISE IMMUNITY
  $V_{NIH} = V_{NIL}$ = 28 % V_{CC} (MIN.)
- OUTPUT DRIVE CAPABILITY
  15 LSTTL LOADS
- SYMMETRICAL OUTPUT IMPEDANCE
  $I_{OL} = |I_{OH}|$ = 6 mA (MIN.)
- BALANCED PROPAGATION DELAYS
  $t_{PLH} = t_{PHL}$
- WIDE OPERATING VOLTAGE RANGE
  V_{CC} (OPR) = 2 V TO 6 V
- PIN AND FUNCTION COMPATIBLE
  WITH 54/74LS563/573



ORDER CODES :

M54HCXXXF1R	M74HCXXXM1R
M74HCXXXB1R	M74HCXXXC1R

DESCRIPTION

The M54/74HC563 and M54HC573 are high speed CMOS OCTAL LATCH WITH 3-STATE OUTPUTS fabricated with in silicon gate C^2MOS technology.

These ICs archive the high speed operation similar to equivalent LSTTL while maintaning the CMOS low power dissipation.

These 8 bit D-Type latches are controlled by a latch enable input (LE) and a output enable input ($\overline{OE}$).

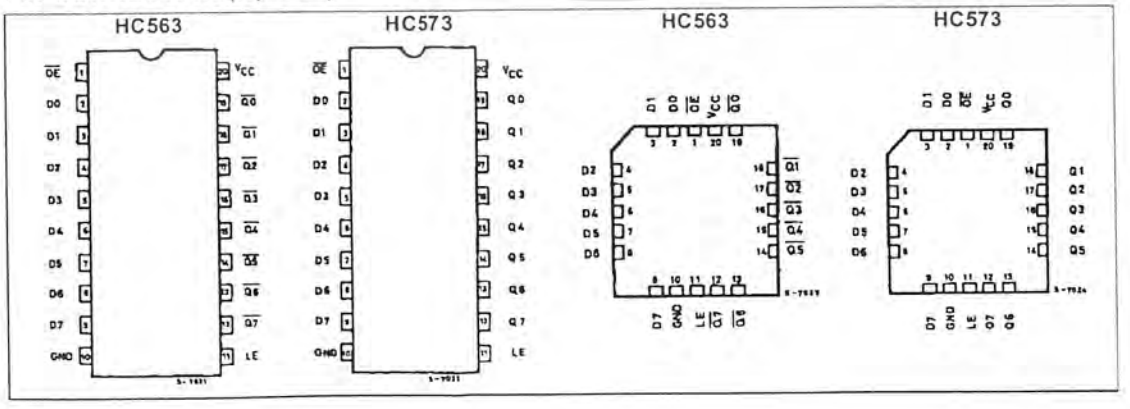
While the LE input is held at a high level, the Q outputs will follow the data input precisely or inversely. When the LE is taken low, the Q outputs will be latched precisely or inversely at the logic level of D input data. While the $\overline{OE}$ input is at low level, the eight outputs will be in a normal logic state (high or low logic level) and while high level the outpts will be in a high impedance state.

The application designer has a choise of combination of inverting and non inverting outputs.

The three state output configuration and the wide choise of outline make bus organized system simple.

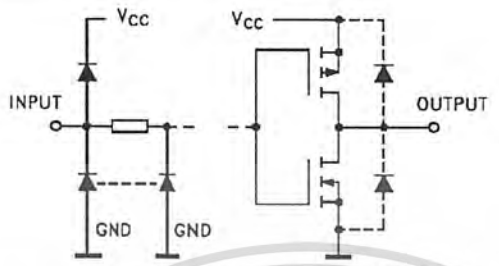
All inputs are equipped with protection circuits against discharge and transient excess voltage.

PIN CONNECTION (top view)



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

INPUT AND OUTPUT EQUIVALENT CIRCUIT



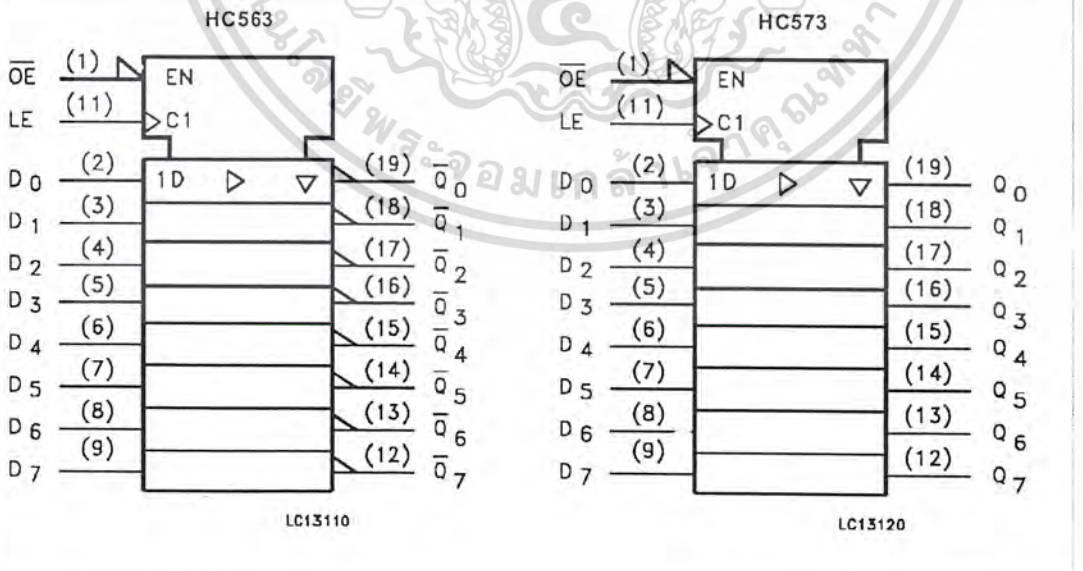
PIN DESCRIPTION (HC563)

PIN No	SYMBOL	NAME AND FUNCTION
1	$\overline{OE}$	3 State output Enable Input (Active LOW)
2, 3, 4, 5, 6, 7, 8, 9	D0 to D7	Data Inputs
12, 13, 14, 15, 16, 17, 18, 19	$\overline{Q0}$ to $\overline{Q7}$	3 State Latch Outputs
11	LE	Latch Enable Input
10	GND	Ground (0V)
20	V_{CC}	Positive Supply Voltage

PIN DESCRIPTION (HC573)

PIN No	SYMBOL	NAME AND FUNCTION
1	$\overline{OE}$	3 State output Enable Input (Active LOW)
2, 3, 4, 5, 6, 7, 8, 9	D0 to D7	Data Inputs
12, 13, 14, 15, 16, 17, 18, 19	Q0 to Q7	3 State Latch Outputs
11	LE	Latch Enable Input
10	GND	Ground (0V)
20	V_{CC}	Positive Supply Voltage

IEC LOGIC SYMBOLS



SGS-THOMSON MICROELECTRONICS

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

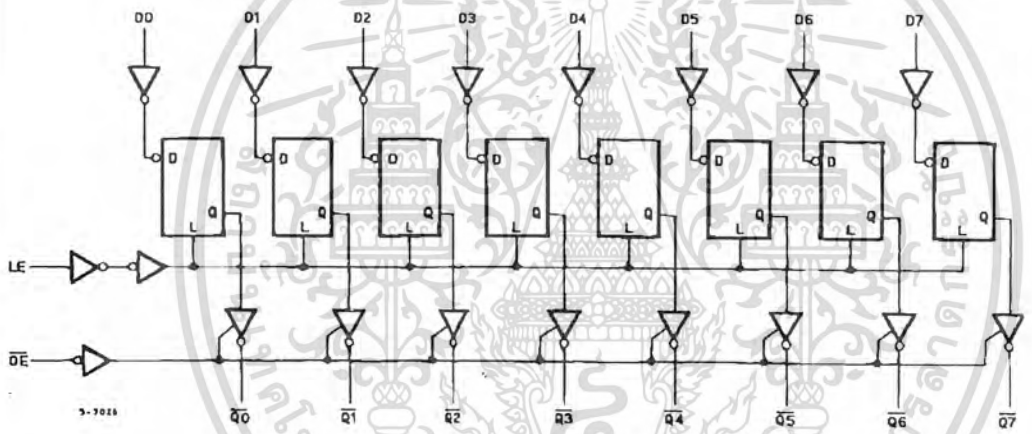
TRUTH TABLE

INPUTS			OUTPUTS	
$\overline{OE}$	LE	D	Q (HC573)	$\overline{Q}$ (HC563)
H	X	X	Z	Z
L	L	X	NO CHANGE *	NO CHANGE *
L	H	L	L	H
L	H	H	H	L

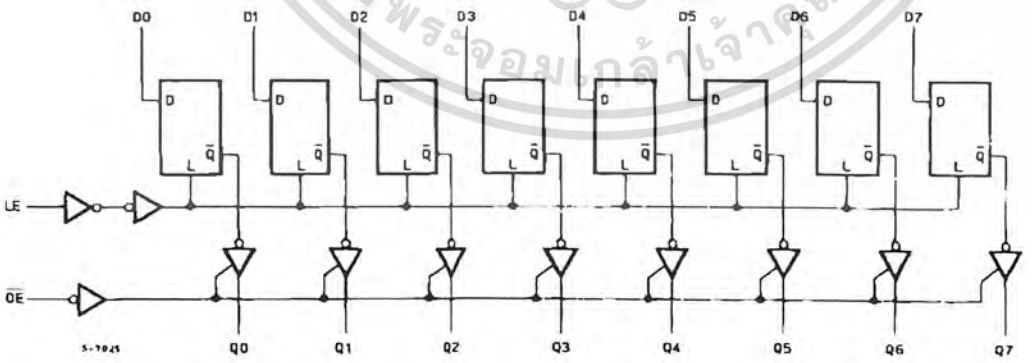
X: DON'T CARE
Z: HIGH IMPEDANCE
*: Q/$\overline{Q}$ OUTPUTS ARE LATCHED AT THE TIME WHEN THE LE INPUT IS TAKEN LOW LOGIC LEVEL.

LOGIC DIAGRAMS

HC563



HC573



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ABSOLUTE MAXIMUM RATINGS

Symbol	Parameter	Value	Unit
V_{CC}	Supply Voltage	-0.5 to +7	V
V_I	DC Input Voltage	-0.5 to V_{CC} + 0.5	V
V_O	DC Output Voltage	-0.5 to V_{CC} + 0.5	V
I_{IK}	DC Input Diode Current	± 20	mA
I_{OK}	DC Output Diode Current	± 20	mA
I_O	DC Output Source Sink Current Per Output Pin	± 35	mA
I_{CC} or I_{GND}	DC V_{CC} or Ground Current	± 70	mA
P_D	Power Dissipation	500 (*)	mW
T_{stg}	Storage Temperature	-65 to +150	°C
T_L	Lead Temperature (10 sec)	300	°C

Absolute Maximum Ratings are those values beyond which damage to the device may occur. Functional operation under these conditions is not implied.
(*) 500 mW: ≡ 65 °C derate to 300 mW by 10mW/°C: 65 °C to 85 °C

RECOMMENDED OPERATING CONDITIONS

Symbol	Parameter		Value	Unit
V_{CC}	Supply Voltage		2 to 6	V
V_I	Input Voltage		0 to V_{CC}	V
V_O	Output Voltage		0 to V_{CC}	V
T_{op}	Operating Temperature: **M54HC** Series		-55 to +125	°C
	M74HC Series		-40 to +85	°C
t_r, t_f	Input Rise and Fall Time	V_{CC} = 2 V	0 to 1000	ns
		V_{CC} = 4.5 V	0 to 500	
		V_{CC} = 6 V	0 to 400	

SGS-THOMSON MICROELECTRONICS
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

DC SPECIFICATIONS

Symbol	Parameter	Test Conditions			Value							Unit
		V_{CC} (V)			T_A = 25 °C 54HC and 74HC			-40 to 85 °C 74HC		-55 to 125 °C 54HC		
					Min.	Typ.	Max.	Min.	Max.	Min.	Max.	
V_{IH}	High Level Input Voltage	2.0			1.5			1.5		1.5		V
		4.5			3.15			3.15		3.15		
		6.0			4.2			4.2		4.2		
V_{IL}	Low Level Input Voltage	2.0					0.5		0.5		0.5	V
		4.5					1.35		1.35		1.35	
		6.0					1.8		1.8		1.8	
V_{OH}	High Level Output Voltage	2.0	V_I = V_{IH} or V_{IL}	I_O=-20 µA	1.9	2.0		1.9		1.9		V
		4.5			4.4	4.5		4.4		4.4		
		6.0			5.9	6.0		5.9		5.9		
		4.5		I_O=-6.0 mA	4.18	4.31		4.13		4.10		
		6.0		I_O=-7.8 mA	5.68	5.8		5.63		5.60		
V_{OL}	Low Level Output Voltage	2.0	V_I = V_{IH} or V_{IL}	I_O= 20 µA		0.0	0.1		0.1		0.1	V
		4.5				0.0	0.1		0.1		0.1	
		6.0				0.0	0.1		0.1		0.1	
		4.5		I_O= 6.0 mA		0.17	0.26		0.33		0.40	
		6.0		I_O= 7.8 mA		0.18	0.26		0.33		0.40	
I_I	Input Leakage Current	6.0	V_I = V_{CC} or GND				±0.1		±1		±1	µA
I_{OZ}	3 State Output Off State Current	6.0	V_I = V_{IH} or V_{IL} V_O = V_{CC} or GND				±0.5		±5.0		±10	µA
I_{CC}	Quiescent Supply Current	6.0	V_I = V_{CC} or GND				4		40		80	µA

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

AC ELECTRICAL CHARACTERISTICS (C_L = 50 pF, Input $t_r = t_f$ = 6 ns)

Symbol	Parameter	Test Conditions			Value							Unit
					T_A = 25 °C 54HC and 74HC			-40 to 85 °C 74HC		-55 to 125 °C 54HC		
		V_{CC} (V)	C_L (pF)		Min.	Typ.	Max.	Min.	Max.	Min.	Max.	
t_{TLH} t_{THL}	Output Transition Time	2.0	50			25	60		75		90	ns
		4.5				7	12		15		18	
		6.0				6	10		13		15	
t_{PLH} t_{PHL}	Propagation Delay Time (LE - Q, $\overline{Q}$)	2.0	50			50	115		145		175	ns
		4.5				15	23		29		35	
		6.0				13	20		25		30	
		2.0	150			60	155		195		235	ns
		4.5				20	31		39		47	
		6.0				17	26		33		40	
t_{PLH} t_{PHL}	Propagation Delay Time (D - Q, $\overline{Q}$)	2.0	50			42	110		140		165	ns
		4.5				14	22		28		33	
		6.0				12	19		24		28	
		2.0	150			57	150		190		225	ns
		4.5				19	30		38		45	
		6.0				16	26		32		38	
t_{PZL} t_{PZH}	3 State Output Enable Time	2.0	50	R_L = 1 KΩ		55	140		175		210	ns
		4.5				17	28		35		42	
		6.0				14	24		30		36	
		2.0	150	R_L = 1 KΩ		66	180		225		270	ns
		4.5				22	36		45		54	
		6.0				19	31		38		46	
t_{PLZ} t_{PHZ}	3 State Output Disable Time	2.0	50	R_L = 1 KΩ		40	125		155		190	ns
		4.5				17	25		31		38	
		6.0				15	21		26		32	
$t_{W(L)}$ $t_{W(H)}$	Minimum Pulse Width	2.0	50			40	75		95		110	ns
		4.5				8	15		19		22	
		6.0				7	13		16		19	
t_s	Minimum Set-up Time	2.0	50			16	50		65		75	ns
		4.5				5	10		13		15	
		6.0				3	9		11		13	
t_h	Minimum Hold Time	2.0	50				5		5		5	ns
		4.5					5		5		5	
		6.0					5		5		5	
C_{IN}	Input Capacitance					5	10		10		10	pF
C_{OUT}	Output Capacitance					10						pF
C_{PD} (*)	Power Dissipation Capacitance		for HC563 for HC573			49 51						pF

(*) C_{PD} is defined as the value of the IC's internal equivalent capacitance which is calculated from the operating current consumption without load. (Refer to Test Circuit). Average operting current can be obtained by the following equation. $I_{CC}(opr) = C_{PD} \bullet V_{CC} \bullet f_{IN} + I_{CC}/8$ (per Gate)

The CPD when n pcs of FLIP-FLOP operate, can be gained by following equations:

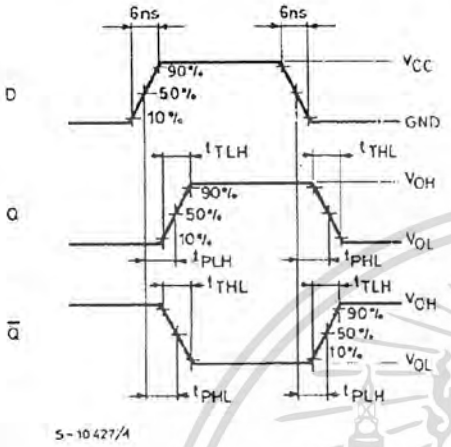
for HC563 CPD (TOTAL) = 33 + 16 x n [pF]; for HC573 CPD (TOTAL) = 33 + 18 x n [pF]

SGS-THOMSON MICROELECTRONICS

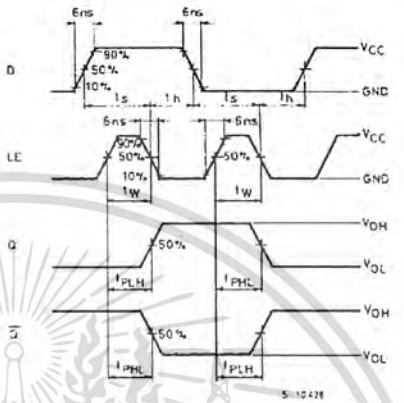
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

SWITCHING CHARACTERISTICS TEST WAVEFORM

t_{PLH}, t_{PHL}, (D - Q, $\overline{Q}$)



t_{PLH}, t_{PHL}, (LE - Q, $\overline{Q}$) t_s, t_h, t_w

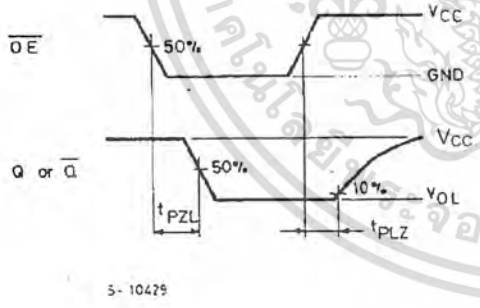


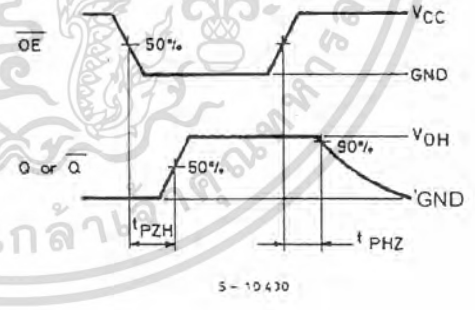
t_{PLZ}, t_{PZL}

The 1KΩ load resistors should be connected between outputs and V_{CC} line and the 50pF load capacitors should be connected between outputsand GND line.
All inputs except $\overline{OE}$ input should be connected to V_{CC} line or GND line such that outputs will be in low logic level while $\overline{OE}$ input is held low.

t_{PHZ}, t_{PZH}

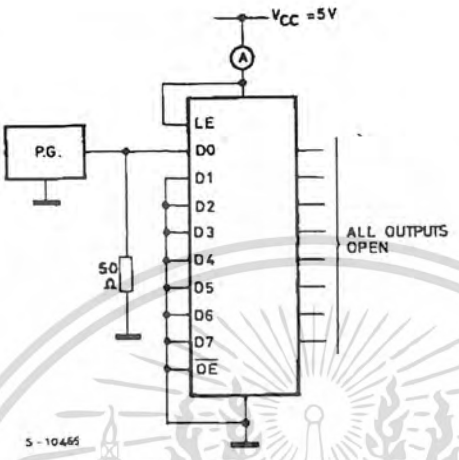
The 1KΩ load resistors and the 50pF load capacitors should be connected between each output and GND line.
All inputs except $\overline{OE}$ input should be connected to V_{CC} or GND line such that output will be in high logic level while $\overline{OE}$ input is held low.





เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

TEST CIRCUIT I_{CC} (Opr.)



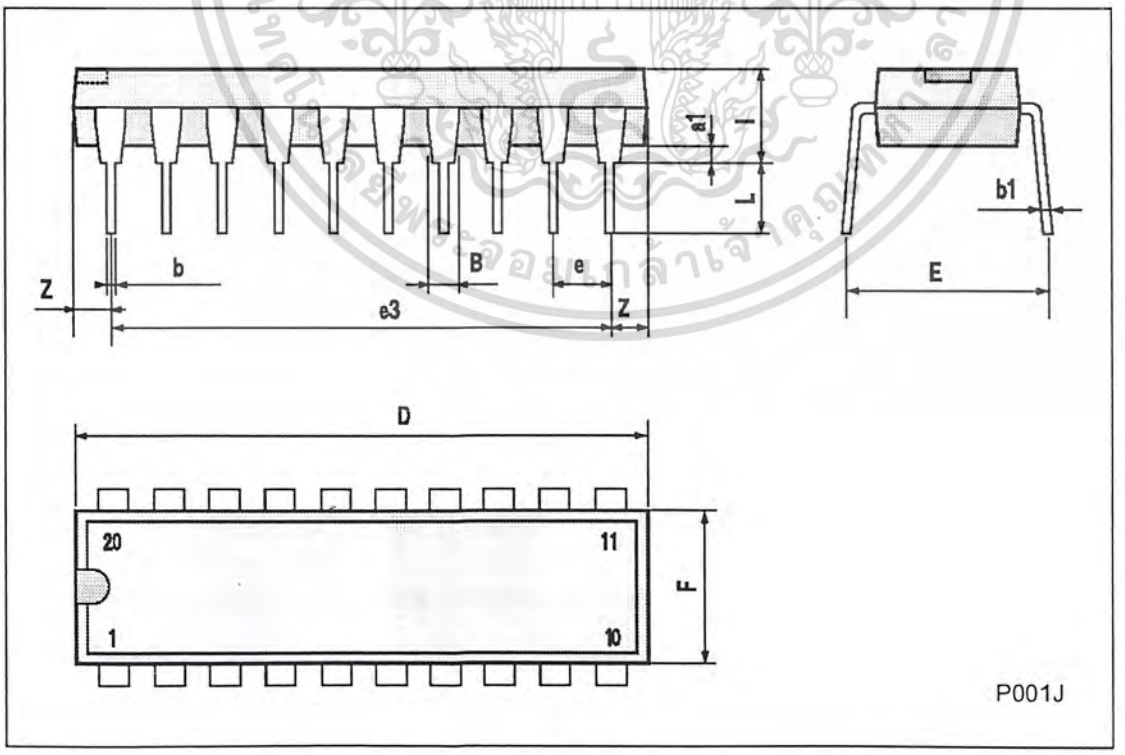
INPUT WAVEFORM IS THE SAME AS THAT IN CASE OF SWITCHING CHARACTERISTICS TEST.

SGS-THOMSON MICROELECTRONICS

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Plastic DIP20 (0.25) MECHANICAL DATA

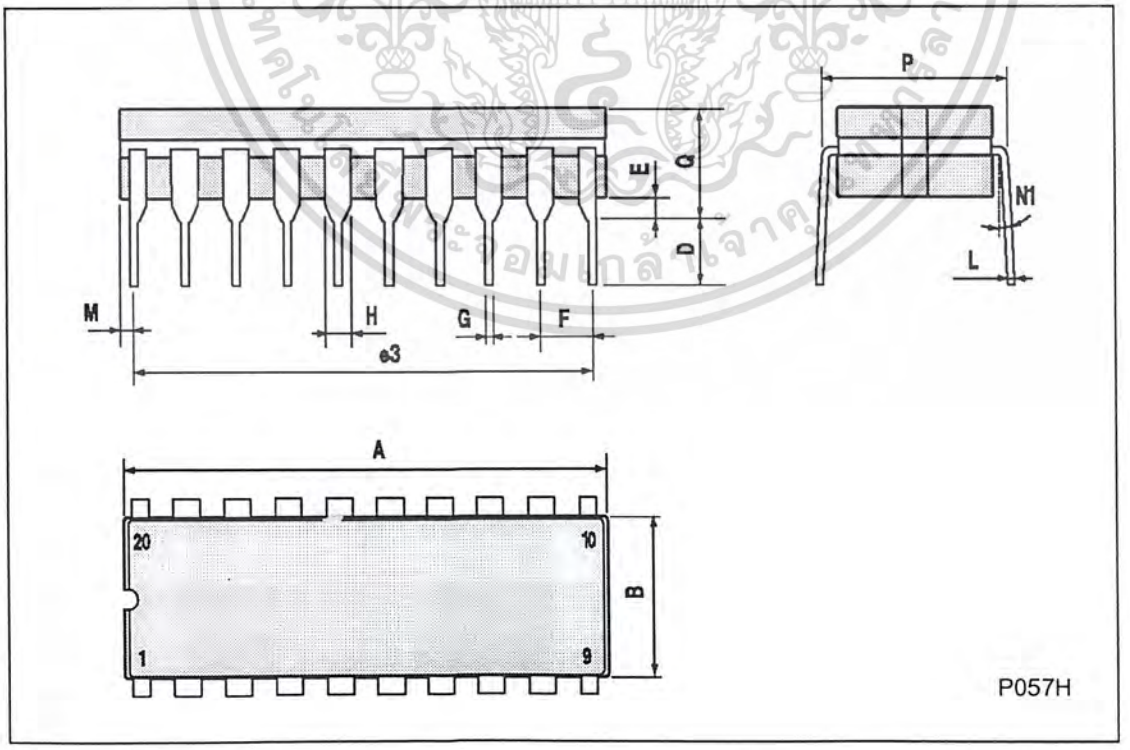
DIM.	mm			inch		
	MIN.	TYP.	MAX.	MIN.	TYP.	MAX.
a1	0.254			0.010		
B	1.39		1.65	0.055		0.065
b		0.45			0.018	
b1		0.25			0.010	
D			25.4			1.000
E		8.5			0.335	
e		2.54			0.100	
e3		22.86			0.900	
F			7.1			0.280
I			3.93			0.155
L		3.3			0.130	
Z			1.34			0.053



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Ceramic DIP20 MECHANICAL DATA

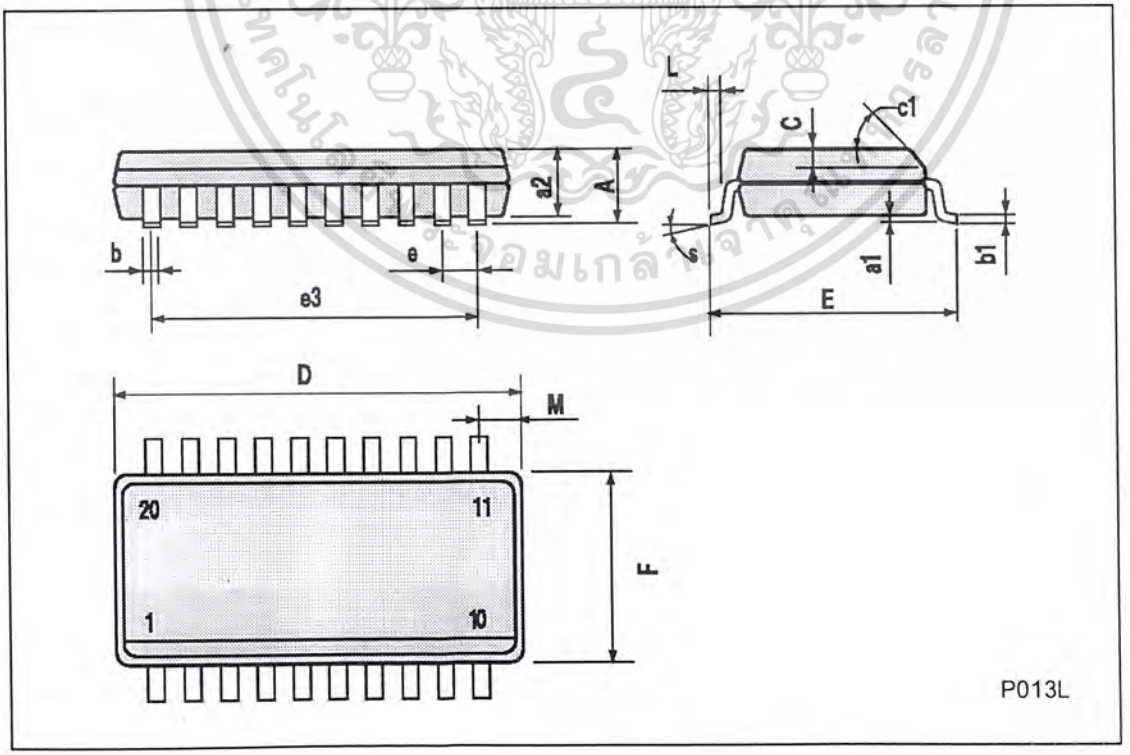
DIM.	mm			inch		
	MIN.	TYP.	MAX.	MIN.	TYP.	MAX.
A			25			0.984
B			7.8			0.307
D		3.3			0.130	
E	0.5		1.78	0.020		0.070
e3		22.86			0.900	
F	2.29		2.79	0.090		0.110
G	0.4		0.55	0.016		0.022
I	1.27		1.52	0.050		0.060
L	0.22		0.31	0.009		0.012
M	0.51		1.27	0.020		0.050
N1	4° (min.), 15° (max.)					
P	7.9		8.13	0.311		0.320
Q			5.71			0.225



SGS-THOMSON MICROELECTRONICS
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

SO20 MECHANICAL DATA

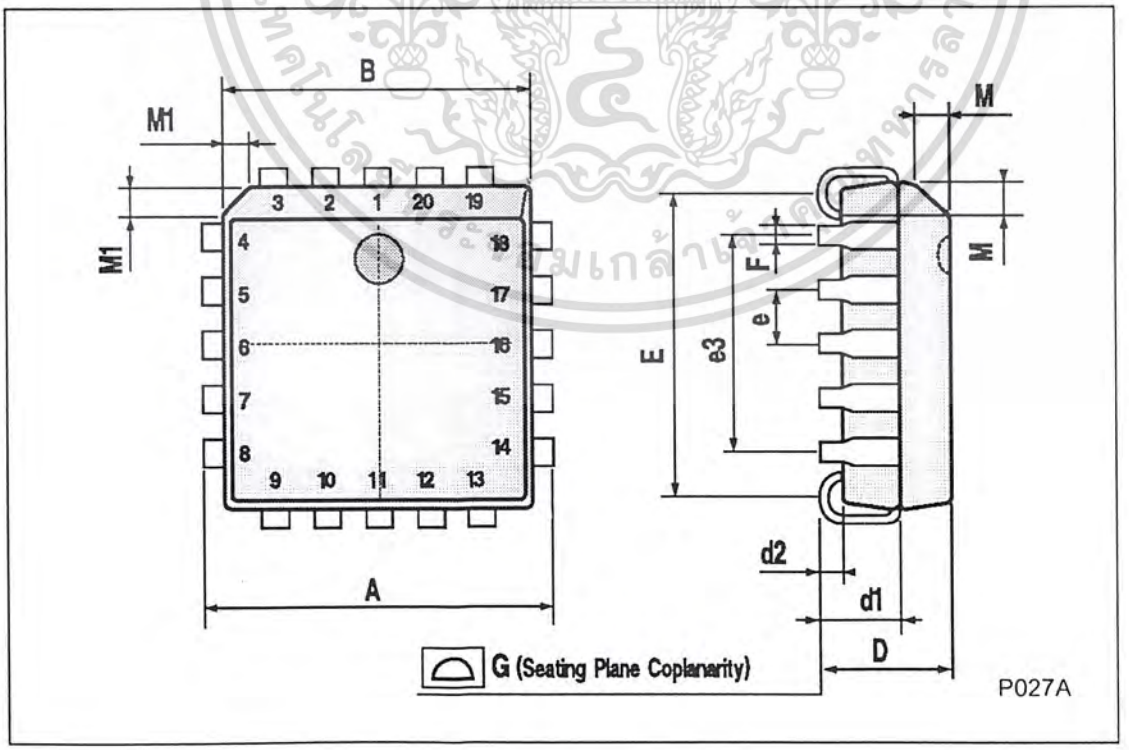
DIM.	mm			inch		
	MIN.	TYP.	MAX.	MIN.	TYP.	MAX.
A			2.65			0.104
a1	0.10		0.20	0.004		0.007
a2			2.45			0.096
b	0.35		0.49	0.013		0.019
b1	0.23		0.32	0.009		0.012
C		0.50			0.020	
c1	45° (typ.)					
D	12.60		13.00	0.496		0.512
E	10.00		10.65	0.393		0.419
e		1.27			0.050	
e3		11.43			0.450	
F	7.40		7.60	0.291		0.299
L	0.50		1.27	0.19		0.050
M			0.75			0.029
S	8° (max.)					



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

PLCC20 MECHANICAL DATA

DIM.	mm			inch		
	MIN.	TYP.	MAX.	MIN.	TYP.	MAX.
A	9.78		10.03	0.385		0.395
B	8.89		9.04	0.350		0.356
D	4.2		4.57	0.165		0.180
d1		2.54			0.100	
d2		0.56			0.022	
E	7.37		8.38	0.290		0.330
e		1.27			0.050	
e3		5.08			0.200	
F		0.38			0.015	
G			0.101			0.004
M		1.27			0.050	
M1		1.14			0.045	



SGS-THOMSON MICROELECTRONICS
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้ [illegible] เท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Information furnished is believed to be accurate and reliable. However, SGS-THOMSON Microelectronics assumes no responsability for the consequences of use of such information nor for any infringement of patents or other rights of third parties which may results from its use. No license is granted by implication or otherwise under any patent or patent rights of SGS-THOMSON Microelectronics. Specifications mentioned in this publication are subject to change without notice. This publication supersedes and replaces all information previously supplied.
SGS-THOMSON Microelectronics products are not authorized for use as critical components in life support devices or systems without express written approval of SGS-THOMSON Microelectonics.

© 1994 SGS-THOMSON Microelectronics - All Rights Reserved

SGS-THOMSON Microelectronics GROUP OF COMPANIES
Australia - Brazil - France - Germany - Hong Kong - Italy - Japan - Korea - Malaysia - Malta - Morocco - The Netherlands - Singapore - Spain - Sweden - Switzerland - Taiwan - Thailand - United Kingdom - U.S.A

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้