สำนักหอสมุดกลาง พระจอมเกล้าลาดกระบัง

หุ่นยนต์ตะขาบ

Centipede Robot

โดย

นายเฉลิมพล ทองเนื้อดี

นายวโรจน์ พงษ์บุพศิริกุล

เลขหมู่.....
เลขทะเบียน..... 42254
วัน, เดือน, ปี..... 16 พ.ค. 2546

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

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรอุตสาหกรรมศาสตรบัณฑิต

สาขาวิชาเทคโนโลยีอิเล็กทรอนิกส์

คณะวิศวกรรมศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2543

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

CENTIPEDE ROBOT

By

Mr. Chalermphol Thongneadee

Mr. Warraot Pongbupasirikul

Bachelor of Industrial Technology

Department of Industrial Technology

Faculty of Engineering

KING MONGKUT'S INSTITUTE OF TECHNOLOGY LADKRABANG

2000

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ชื่อโครงงานภาษาไทย	หุ่นยนต์ตะขาบ
ชื่อโครงงานภาษาอังกฤษ	Centipede Robot
ผู้จัดทำ	นายเฉลิมพล ทองเนื้อดี
	นายวโรจน์ พงษ์บุพศิริกุล
อาจารย์ที่ปรึกษา	อาจารย์พนารัตน์ ระวิวรรณ
	อาจารย์บุณย์ชนะ ภู่ระหงษ์
ภาควิชา	เทคนิคอุตสาหกรรม
ปีการศึกษา	2543

คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง อนุมัติให้รับปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรอุตสาหกรรมศาสตร์บัณฑิต

ลงชื่อ.....อาจารย์ที่ปรึกษา
(อาจารย์พนารัตน์ ระวิวรรณ)

ลงชื่อ.....อาจารย์ที่ปรึกษา
(อาจารย์บุณย์ชนะ ภู่ระหงษ์)

คณะกรรมการการสอบปริญญานิพนธ์

.....กรรมการ
( )

.....กรรมการ
( )

.....กรรมการ
( )

.....กรรมการ
( )

.....กรรมการ
( )

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ปริญญานิพนธ์	หุ่นยนต์ตะขาบ
จัดทำโดย	นายเฉลิมพล ทองเนื้อดี 42015505
	นายวโรจน์ พงษ์บุพศิริกุล 42015524
อาจารย์ที่ปรึกษา	อาจารย์พนารัตน์ ระวิวรรณ
	อาจารย์บุณย์ชนะ ภู่ระหงษ์
ปีการศึกษา	2543

บทคัดย่อ

ปริญญานิพนธ์ฉบับนี้ได้เสนอ การสร้างและควบคุมหุ่นยนต์ตะขาบ โดยนำ MCS51 ไมโครคอนโทรลเลอร์มาควบคุมการทำงาน ของมอเตอร์ไฟตรง (DC Motor) ซึ่งจะทำหน้าที่ควบคุมการก้าวเดินของตัวหุ่นยนต์ให้สามารถเคลื่อนที่และหลบหลีกสิ่งกีดขวางได้ และ ใช้ IC เบอร์ L293D ในการขับมอเตอร์ไฟตรงให้สามารถเพิ่มความสามารถในการขับเคลื่อนของมอเตอร์ไฟตรงได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Thesis title	Centipede Robot
Student	Mr. Chalermphol Thongneadee 42015505
	Mr. Warraot Pongbupasirikul 42015524
Adviser	Miss.Panarat Rawiwan
	Mr. Boonchana Poorahong
Academic	Year 2000

Abstract

This thesis presents building and controlling centipede robot .The DC motor are controlled by MCS51 micro controller for motion obstacle avoidance and increase ability for drive DC motor by used IC#L293D

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กิตติกรรมประกาศ

จากความสำเร็จในการสร้าง หุ่นยนต์ตะขาบ คณะผู้จัดทำ ขอขอบพระคุณท่าน อาจารย์ บุณย์ชนะ ภู่ระหงษ์ (อาจารย์ที่ปรึกษา) และ อาจารย์ พนารัตน์ ระวิวรรณ (อาจารย์ที่ปรึกษา)ที่ได้ให้คำชี้แนะให้การสนับสนุน ให้ความช่วยเหลือในทุก ๆ ด้าน และ ขอขอบคุณ ภาคเทคนิคอุตสาหกรรม คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง ที่ได้ให้การสนับสนุน จนกระทั่งโครงงานนี้สำเร็จลุล่วงไปได้ด้วยดี

ลงชื่อ.....

(นาย เฉลิมพล ทองเนื้อดี)

ลงชื่อ.....

(นาย วโรจน์ พงษ์บุพศิริกุล)

คณะผู้จัดทำ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ

เรื่อง **หน้า**

บทคัดย่อภาษาไทย I

บทคัดย่อภาษาอังกฤษ II

กิตติกรรมประกาศ III

สารบัญ IV

สารบัญรูป VI

สารบัญตาราง IX

บทที่ 1 บทนำ

1.1 ความเป็นมา 1

1.2 ชื่อโครงงาน 1

1.3 วัตถุประสงค์ 1

1.4 ขีดความสามารถของโครงงาน 1

1.5 เนื้อหาโดยสังเขป 1

บทที่ 2 ทฤษฎีและหลักการ

2.1 เฟืองชุดแบบธรรมดา 3

2.2 ไมโครคอนโทรลเลอร์ MCS51 5

2.3 การเขียนโปรแกรมภาษาแอสเซมบลี 10

2.4 IC#L293D (ไอซีขับมอเตอร์) 12

2.5 การทำงานเป็นไทเมอร์ 13

2.6 การสื่อสารข้อมูลแบบอะซิงโครนัส 27

2.7 รูปแบบของการสื่อสารข้อมูลแบบหนึ่งลาย(1– Wire™communication protocol) 48

2.8 ระบบเซอร์โว (Servo Systems) 54

2.9 ดีซีเซอร์โวมอเตอร์ (DC. Servomotor) 55

บทที่ 3 โครงสร้างและการออกแบบ

3.1 การออกวงจรกับความสัมพันธ์ของ cpu 57

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ (ต่อ)

เรื่อง	หน้า
3.2 การออกแบบวงจร SENSOR	58
3.3 การออกแบบวงจรขับมอเตอร์	61
3.4 การออกแบบวงจร Microcontroller	63
3.5 การออกแบบโครงสร้างขา และ ตัวของหุ่นยนต์	65
บทที่ 4 การออกแบบโปรแกรมควบคุม	
4.1 การออกแบบและโปรแกรมควบคุม	74
บทที่ 5 การทดลอง	
5.1 การทดลองรูปแบบของการเดิน	80
5.2 การทดลองความเร็วในการเดิน (หุ่นยนต์ 1 ตัว)	81
5.3 การทดลองความเร็วในการเดิน (หุ่นยนต์ ต่อเชื่อมกันหมด 8 ตัว)	82
บทที่ 6 สรุปโครงงาน ปัญหาและข้อเสนอแนะ	
6.1 สรุปโครงงาน	90
6.2 ปัญหาในการทำโครงงาน	90
6.3 ข้อเสนอแนะในการพัฒนาต่อไป	91
ภาคผนวก ก. โปรแกรมควบคุมและวงจร	
ภาคผนวก ข. รูปถ่ายหุ่นยนต์	
ภาคผนวก ค. รายละเอียดของไมโครคอนโทรลเลอร์	
ภาคผนวก ง. รายละเอียดของ IC ที่ใช้ในโครงงาน	
บรรณานุกรม	

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูป

หน้า

รูปที่ 2.1 เฟืองชุดแบบธรรมดาหมุนทางเดียว 3
รูปที่ 2.2 เฟืองชุดแบบธรรมดาหมุนสวนทางกัน 3
รูปที่ 2.3 การจัดตำแหน่งขาต่างๆ ของไมโครคอนโทรลเลอร์ตระกูล MCS-51 6
รูปที่ 2.4 หน้าที่พิเศษของแต่ละขาของพอร์ต P3 7
รูปที่ 2.5 กระบวนการต่างๆ ที่เกิดขึ้นจากซอร์สโค้ดที่เขียนบนเครื่องพีซีจนเป็นออปเจ็กต์โค้ด ซึ่งถูกโหลดมาเก็บไว้ในหน่วยความจำโปรแกรมบน MCS-51 บอร์ด 11
รูปที่ 2.6 โครงสร้างภายในของ L293D 12
รูปที่ 2.7 โครงสร้างภายนอกของ L293D 12
รูปที่ 2.8 ไดอะแกรมการทำงานในโหมด 0 ของไทเมอร์/เคาน์เตอร์ 1 19
รูปที่ 2.9 ไดอะแกรมการทำงานในโหมด 1 ของไทเมอร์/เคาน์เตอร์ 1 19
รูปที่ 2.10 ไดอะแกรมการทำงานในโหมด 2 ของไทเมอร์/เคาน์เตอร์ 1 20
รูปที่ 2.11 ไดอะแกรมการทำงานในโหมด 3 ของไทเมอร์/เคาน์เตอร์ 1 21
รูปที่ 2.12 ไดอะแกรมการทำงานในโหมดแคปเจอร์ของไทเมอร์ / เคาน์เตอร์ 2 25
รูปที่ 2.13 ไดอะแกรมการทำงานในโหมดตั้งค่าการนับอัตโนมัติของไทเมอร์ / เคาน์เตอร์ 2 26
รูปที่2.14 ไดอะแกรมการทำงานในโหมดกำเนิดอัตราบอดในการสื่อสารข้อมูลผ่านพอร์ตอนุกรมของไทเมอร์ / เคาน์เตอร์ 2 26
รูปที่ 2.15 ไดอะแกรมการทำงานในโหมด 0 ของพอร์ตอนุกรมภายในไมโครคอนโทรลเลอร์ MCS-51 34
รูปที่ 2.16 ไดอะแกรมการทำงานในโหมด 1 ของพอร์ตอนุกรมภายในไมโครคอนโทรลเลอร์ MCS-51 โหมด 2 36
รูปที่ 2.17 ไดอะแกรมการทำงานในโหมด 2 ของพอร์ตอนุกรมภายในไมโครคอนโทรลเลอร์ MCS-51 38
รูปที่ 2.18 ไดอะแกรมการทำงานในโหมด 3 ของพอร์ตอนุกรมภายในไมโครคอนโทรลเลอร์ MCS-51 39
รูปที่ 2.19 การเชื่อมต่อบนระบบบัสหนึ่งสาย 43
รูปที่ 2.20 ไทม์สล็อตการรีเซตและการตอบรับของอุปกรณ์บนระบบบัสหนึ่งสาย 44

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูป (ต่อ)

หน้า

รูปที่ 2.21 ไทม์สล็อตการอ่านข้อมูลของอุปกรณ์มาสเตอร์ ซึ่งตรงกับไทม์สล็อตการเขียนข้อมูลของอุปกรณ์สเลฟ 45
รูปที่ 2.22 ไทม์สล็อตการเขียนข้อมูล "1" ของอุปกรณ์มาสเตอร์ ซึ่งตรงกับไทม์สล็อตการอ่านข้อมูลของอุปกรณ์สเลฟ 46
รูปที่ 2.23 ไทม์สล็อตการเขียนข้อมูล "0" ของอุปกรณ์มาสเตอร์ 46
รูปที่ 2.24 การจัดขาของ DS1820 47
รูปที่ 2.25 โครงสร้างการทำงานภายในของไอซีตรวจจับอุณหภูมิ DS1820 48
รูปที่ 2.26 การจัดสรรพื้นที่ของสแครตช์แพดใน DS1820 48
รูปที่ 2.27 การเชื่อมต่อ DS1820 กับไมโครคอนโทรลเลอร์ MCS-51 49
รูปที่ 2.28 โฟลวชาร์ตและรายละเอียดโปรแกรมย่อยการรีเซต DS1820 50
รูปที่ 2.29 โฟลวชาร์ตและรายละเอียดโปรแกรมย่อยรอการตอบรับจาก DS1820 51
รูปที่ 2.30 โฟลวชาร์ตและรายละเอียดโปรแกรมย่อยการอ่านและเขียนข้อมูลกับ DS1820 52
รูปที่ 2.31 โครงสร้างของดีซีมอเตอร์แบบแม่เหล็กถาวร 55
รูปที่ 3.1 การเชื่อมต่อข้อมูลของหุ่นแต่ละข้อ 57
รูปที่ 3.2 วงจรตรวจจับสิ่งกีดขวาง Collision Avoidance sensor 59
รูปที่ 3.3 การต่อ Micro Switch กับ Mcs51 61
รูปที่ 3.4 block diagram การต่อ Mcs51กับ วงจรขับ motor 62
รูปที่ 3.5 วงจรขับมอเตอร์ dc สองตัว 62
รูปที่ 3.6 แหล่งจ่ายไฟ 63
รูปที่ 3.7 ภาพรวมการเชื่อมต่อทั้ง 3 ส่วน 64
รูปที่ 3.8 ขาที่ใช้งานจริง 65
รูปที่ 3.9 ขนาดของเฟืองที่ใช้งาน 66
รูปที่ 3.10 การจัดตำแหน่งของเฟือง 66
รูปที่ 3.11 การต่อขาที่ใช้งานจริงเข้ากับเฟือง 67
รูปที่ 3.12 ขนาดและลักษณะของตัววาง มอเตอร์ 68
รูปที่ 3.13 ภาพด้านข้าง และ ตำแหน่งเฟืองของตัวหุ่นยนต์ 69
รูปที่ 3.14 ภาพด้านหน้าของตัวหุ่นยนต์ 69

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูป (ต่อ)

หน้า

รูปที่ 3.15 ภาพด้านบนพร้อมติดตั้งเฟืองและ มอเตอร์ของตัวหุ่นยนต์ 70
รูปที่ 3.16 รูปข้อต่อในการเคลื่อนตัวของหุ่น 70
รูปที่ 3.17 รูปข้อต่อระหว่างหุ่นแต่ละข้อ 71
รูปที่ 3.18 รูปด้านบนของข้อต่อระหว่างหุ่นแต่ละข้อ 71
รูปที่ 3.19 ด้านข้างของหุ่นยนต์เมื่อประกอบขา 72
รูปที่ 3.20 ด้านบนของหุ่นยนต์เมื่อประกอบขา 72
รูปที่ 3.21 ด้านบนของหุ่นยนต์เมื่อต่อกัน 2 ข้อ 73
รูปที่ 4.1 การเชื่อมต่อของข้อมูลของกันและกัน 74
รูปที่ 4.2 FLOW CHARTแสดงการควบคุมหุ่นยนต์(ตัวที่ 1) 76
รูปที่ 4.3 FLOW CHARTแสดงการควบคุมหุ่นยนต์(ตัวที่ 2) 78
รูปที่ 4.4 FLOW CHARTแสดงการควบคุมหุ่นยนต์(ตัวที่ 3) 79
รูปที่ 5.1 การวางขา 120 องศา 80
รูปที่ 5.2 การวางขา 90 องศา 81
รูปที่ 5.3 การวางตำแหน่งของSENSOR 84
รูปที่ 5.4 สัญญาณ PWM ที่เกิดขึ้น เมื่อ SENSOR ตรวจไม่พบสิ่งกีดขวาง(SR,SL = OFF) 84
รูปที่ 5.5 การตรวจพบสิ่งกีดขวางทางซ้ายของSENSOR 85
รูปที่ 5.6 สัญญาณ PWM ที่เกิดขึ้น เมื่อ SENSOR ตรวจพบสิ่งกีดขวาง(SL = ON) 85
รูปที่ 5.7 การตรวจพบสิ่งกีดขวางทางขวาของSENSOR 86
รูปที่ 5.8 สัญญาณ PWM ที่เกิดขึ้น เมื่อ SENSOR ตรวจพบสิ่งกีดขวาง(SR = ON) 86
รูปที่ 5.9 การตรวจพบสิ่งกีดขวางทางด้านหน้าของSENSOR 87
รูปที่ 5.10 สัญญาณ PWM ที่เกิดขึ้น เมื่อ SENSOR ตรวจพบสิ่งกีดขวาง(SL,SR = ON) 87
รูปที่ 5.11 การเคลื่อนที่ไปข้างหน้าของหุ่นยนต์ 88
รูปที่ 5.12 ลักษณะการเลี้ยวของหุ่นยนต์ 89

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญตาราง

หน้า

ตารางที่ 2.1 ข้อมูลของรีจิสเตอร์ TMOD เพื่อกำหนดให้ไทเมอร์/เคาน์เตอร์ 0 ทำงานเป็นไทเมอร์ 22

ตารางที่ 2.2 ข้อมูลของรีจิสเตอร์ TMOD เพื่อกำหนดให้ไทเมอร์/เคาน์เตอร์ 0 ทำงานเป็นเคาน์เตอร์ 22

ตารางที่ 2.3 ข้อมูลของรีจิสเตอร์ TMOD เพื่อกำหนดให้ไทเมอร์/เคาน์เตอร์ 1 ทำงานเป็นไทเมอร์ 23

ตารางที่ 2.4 ข้อมูลของรีจิสเตอร์ TMOD เมื่อกำหนดให้ไทเมอร์/เคาน์เตอร์ 1 ทำงานเป็นเคาน์เตอร์ 23

ตารางที่ 2.5 การเลือกโหมดทำงานของไทเมอร์/เคาน์เตอร์ 2 24

ตารางที่ 2.6 ข้อมูลของรีจิสเตอร์ T2CON เพื่อกำหนดให้ไทเมอร์/เคาน์เตอร์ 2 ทำงานเป็นไทเมอร์ 26

ตารางที่ 2.7 ข้อมูลของรีจิสเตอร์ T2CON เพื่อกำหนดให้ไทเมอร์/เคาน์เตอร์ 2 ทำงานเป็นเคาน์เตอร์ 27

ตารางที่ 2.8 การเลือกอัตราบอดของวงจรพอร์ตอนุกรมภายในไมโครคอนโทรลเลอร์ MCS-51 41

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 1
บทนำ

1.1 ความเป็นมา

เนื่องจากเดิมการสร้างหุ่นยนต์ อาจจะต้องใช้อุปกรณ์ราคาแพง ซับซ้อน และการผลิตหุ่นยนต์นั้นจะใช้ ไมโครคอนโทรลเลอร์เป็นหัวใจหลักในการควบคุม การทำงาน

จึงได้มีแนวความคิดที่จะสร้างและพัฒนาหุ่นยนต์ ที่สามารถเคลื่อนไหวเลียนแบบตะขาบ และ มีราคาถูก ซึ่งใช้ไมโครคอนโทรลเลอร์ควบคุมการทำงานของหุ่นยนต์ และในส่วนของการเคลื่อนที่ที่จะใช้ขา แทนล้อซึ่งทำให้ หุ่นยนต์ สามารถเดินไปในที่ขรุขระ และ สามารถปีนป่าย สิ่งกีดขวางขนาดไม่ใหญ่มากนักได้

1.2 ชื่อโครงงาน

หุ่นยนต์ตะขาบ (Centipede Robot)

1.3 วัตถุประสงค์

- เพื่อสร้างหุ่นยนต์ที่สามารถเลียนแบบการเคลื่อนที่ของตะขาบให้ได้มากที่สุด
- เพื่อสามารถสร้างหุ่นยนต์ที่มีประสิทธิภาพและมีราคาถูก
- เพื่อประยุกต์ใช้งาน ไมโครคอนโทรลเลอร์ เพื่อควบคุมการทำงานของหุ่นยนต์ ให้มีประสิทธิภาพมากที่สุด

1.4 ขีดความสามารถของโครงงาน

- ควบคุมให้หุ่นยนต์เคลื่อนที่ และ หลบหลีกสิ่งกีดขวางได้
- สามารถเลียนแบบการเคลื่อนไหวของตะขาบได้

1.5 เนื้อหาโดยสังเขป

บทที่ 2 เป็นทฤษฎีและหลักการที่เกี่ยวข้องในการสร้างหุ่นยนต์และช่วยในการออกแบบระบบ

บทที่ 3 โครงสร้างและการออกแบบ จะกล่าวถึง การออกแบบ ทางด้าน Hardware และ Solfware

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 4 จะกล่าวถึงการออกแบบโปรแกรมควบคุมการทำงานของหุ่นยนต์ ในสภาวะต่างๆ

บทที่ 5 การทดลอง

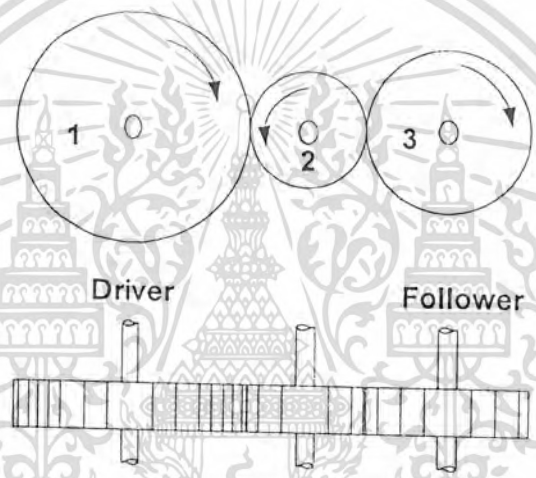
บทที่ 6 สรุปโครงงานและข้อเสนอแนะ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

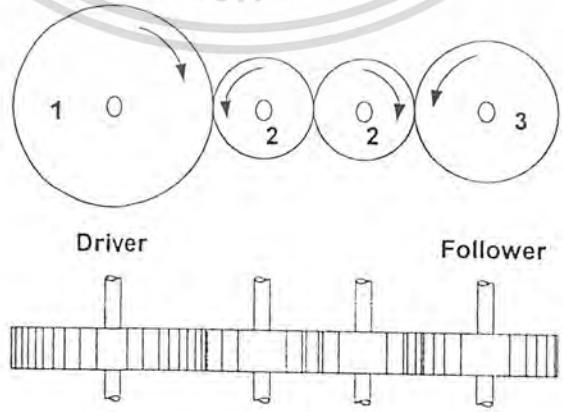
บทที่ 2
ทฤษฎีและหลักการ

2.1 เฟืองชุดแบบธรรมดา

ในกรณีที่เพลาสองอันอย่างห่างกันมาก การถ่ายทอดกำลังงานไปยังเฟืองอีกอันหนึ่งสามารถทำได้ โดยใช้เฟืองร่วมขนาดใหญ่อันเดียวมาเชื่อต่อเข้าจะประหยัดมากกว่าการใช้เฟืองใหญ่เพียงอันเดียว แต่ในการเลือกใช้แบบไหนก็ขึ้นอยู่กับเนื้อที่การใช้งาน และทิศทางการหมุนของเฟืองที่ต้องการนำไปใช้งาน เช่น



รูปที่ 2.1 เฟืองชุดแบบธรรมดาหมุนทางเดียว



รูปที่ 2.2 เฟืองชุดแบบธรรมดาหมุนสวนทางกัน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ในกรณีให้เฟืองขับและเฟืองตามหมุนไปในทิศทางเดียวกัน เฟืองร่วมต้องมีจำนวนเป็นเลขคี่ดังรูปที่ 2.1 และถ้าต้องการให้เฟืองขับและเฟืองตามหมุนไปทิศทางตรงกันข้ามต้องใช้เฟืองร่วมเป็นจำนวนเลขคู่ดังรูปที่ 2.2

พิจารณาการทำงาน

ให้ N1= ความเร็วรอบของเฟืองขับ T1= จำนวนฟันของเฟืองขับ

N2= ความเร็วรอบของเฟืองร่วม T2= จำนวนฟันของเฟืองร่วม

N3= ความเร็วรอบของเฟืองตาม T3= จำนวนฟันของเฟืองตาม

พิจารณาการทำงานร่วมกันของเฟืองขับและเฟืองร่วม จะได้

$$\frac{N2}{N1} = \frac{T1}{T2}$$

ในทำนองเดียวกัน จากการทำงานร่วมกันระหว่างเฟืองร่วมกับเฟืองตาม จะได้

$$\frac{N3}{N2} = \frac{T2}{T3}$$

สรุปได้ว่า

$$\frac{\text{ความเร็วของเฟืองตาม}}{\text{ความเร็วของเฟืองขับ}} = \frac{\text{จำนวนฟันของเฟืองขับ}}{\text{จำนวนฟันของเฟืองตาม}}$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

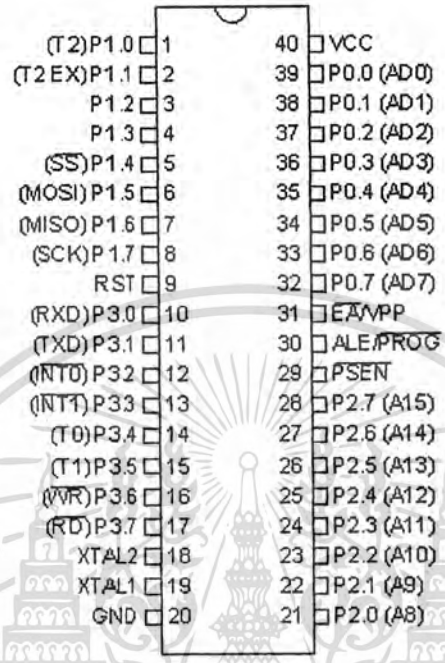
2.2 ไมโครคอนโทรลเลอร์ MCS51

2.2.1 คุณสมบัติทั่วไปของไมโครคอนโทรลเลอร์ MCS-51

- เป็นไมโครคอนโทรลเลอร์ขนาด 8 บิท
- มีวงจรออสซิลเลเตอร์และวงจรผลิตสัญญาณนาฬิกาภายในไอซี
- มีขาสัญญาณอินพุตเอาต์พุตจำนวน 32 บิต
- สามารถเชื่อมต่อหน่วยความจำข้อมูลภายนอก (external data memory) โดยอ้างตำแหน่งแอดเดรสได้ถึง 64 K
- สามารถเชื่อมหน่วยความจำโปรแกรมภายนอก (external program memory) โดยอ้างตำแหน่งแอดเดรสได้ถึง 64 K
- มีหน่วยความจำโปรแกรมภายในตัว (on-chip program memory) ขนาด 4 K โดยเฉพาะเบอร์ 8052 จะมีหน่วยความจำในส่วนนี้ถึง 8 K สำหรับเบอร์ 8031 และ M8032 จะไม่มีความจำในส่วนนี้
- มีหน่วยความจำข้อมูลภายในตัว(on-chip data memory)ขนาด 128 ไบต์ โดยเฉพาะเบอร์ 8032 และ 8052 จะมีหน่วยความจำในส่วนนี้ถึง 256 ไบต์
- หน่วยความจำข้อมูลภายในบางส่วนสามารถเข้าถึงข้อมูลระดับบิตได้ด้วย ทำให้การควบคุมหรือการตรวจสอบสถานะบิตทำได้ง่าย ส่งผลให้การเขียนโปรแกรมทำได้ง่ายมากขึ้น
- มีไทเมอร์/เคาน์เตอร์ (timer/counter) ขนาด 16 บิต จำนวน 2 ตัว โดยเฉพาะเบอร์ 8032 หรือ 8052 จะมีไทเมอร์/เคาน์เตอร์จำนวน 3 ตัว
- การอินเตอร์รัปต์สามารถทำได้จาก 5 แหล่งกำเนิด โดยเฉพาะเบอร์ 8032 และ 8052 จะทำการอินเตอร์รัพต์ได้ 6 แหล่งกำเนิด โดยการอินเตอร์รัปต์ยังสามารถจัดระดับความสำคัญได้เป็น 2 ระดับ
- มีพอร์ตสื่อสารอนุกรมภายในตัวเอง ซึ่งทำงานเป็นแบบฟูลดูเพล็กซ์ (full duplex)
- มีคำสั่งในการคำนวณทางคณิตศาสตร์และทางตรรกศาสตร์
- คำสั่งโดยส่วนใหญ่ใช้เวลาการทำงานเพียง 1 ไมโครวินาที เมื่อใช้คริสตอลความถี่ 12 เมกะเฮิรตซ์
- ต้องการแหล่งจ่ายไฟ 5 โวลต์ เพียงชุดเดียว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.2.2 โครงสร้างภายนอกของ MCS-51



รูปที่ 2.3 การจัดตำแหน่งขาต่างๆ ของไมโครคอนโทรลเลอร์ตระกูล MCS-51

หน้าที่การใช้งานของขาต่างๆมีดังนี้

- ขา Vcc เป็นขาป้อนแรงดันไฟเลี้ยง +5 โวล์ต
- ขา Vss เป็นขากราวด์
- ขาพอร์ต 0 (Port 0) มี 8 ขา ได้แก่ขา P0.0 – P0.7 เป็นขาพอร์ตอินพุตเอาต์พุตแบบ 2 ทิศทางสำหรับใช้งานทั่วไป โดยถ้าใช้งานเป็นอินพุตพอร์ตต้องทำการเขียนค่า 1 ไปยังแต่ละบิตของพอร์ต เพื่อกำหนดให้ขาพอร์ตเหล่านั้นอยู่ในสถานะปล่อยลอย ซึ่งในสถานะนี้เองที่สามารถนำมาใช้เป็นพอร์ตอินพุตอิมพีแดนซ์สูงได้ นอกจากพอร์ตนี้จะใช้งานเป็นพอร์ตอินพุตเอาต์พุตแล้วมันยังถูกใช้งานในการติดต่อกับหน่วยความจำภายนอกด้วย โดยทำหน้าที่ในการกำหนดแอดเดรสไบต์ต่ำ (A0-A7) ซึ่งจะใช้งานเป็นแบบมัลติเพล็กซ์กับการรับส่งข้อมูลขนาด 8 บิต (D0-D7)
- ขาพอร์ต 1 (port 1) มีขา 8 ขา ได้แก่ขา P1.0-P1.7 เป็นขาพอร์ตอินพุตเอาต์พุตแบบ 2 ทิศทาง สำหรับใช้งานทั่วไป โดยถ้าใช้งานเป็นอินพุตพอร์ตต้องทำการเขียนค่า 1 ไปยังแต่ละบิตของพอร์ต เพื่อกำหนดให้เป็นพอร์ตอินพุต นอกจากนี้สำหรับเบอร์ 8032

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

และ 8052 ขาพอร์ต P1.1 และ P1.0 จะถูกนำมาใช้งานเป็นขา T2 และ T2EX ตามลำดับด้วย

- ขาพอร์ต 2 (port 2) มีขา 8 ขา ได้แก่ขา P2.0-P2.7 เป็นขาพอร์ตอินพุตเอาต์พุตแบบ 2 ทิศทาง สำหรับใช้งานทั่วไป โดยถ้าใช้งานเป็นอินพุตพอร์ตต้องทำการเขียนค่า 1 ไปยังแต่ละบิตของพอร์ต เพื่อกำหนดให้เป็นพอร์ตอินพุต นอกจากพอร์ตนี้จะถูกใช้งานเป็นอินพุตเอาต์พุตแล้วมันยังถูกใช้งานในการติดต่อกับหน่วยความจำภายนอกด้วย โดยทำหน้าที่ในการกำหนดตำแหน่งแอดเดรสไบต์สูง (A8-A15)
- ขาพอร์ต 3 (port 3) มีขา 8 ขา ได้แก่ขา P3.0-P3.7 เป็นขาพอร์ตอินพุตเอาต์พุตแบบ 2 ทิศทาง สำหรับใช้งานทั่วไป โดยถ้าใช้งานเป็นอินพุตพอร์ตต้องทำการเขียนค่า 1 ไปยังแต่ละบิตของพอร์ต เพื่อกำหนดให้เป็นพอร์ตอินพุต นอกจากพอร์ตนี้จะถูกใช้งานเป็นอินพุตเอาต์พุตแล้วมันยังถูกใช้งานในหน้าที่พิเศษต่างๆ ดังแสดงในตาราง

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)

รูปที่ 2.4 หน้าที่พิเศษของแต่ละขาของพอร์ต P3

- ขารีเซต (RST) ใช้สำหรับการรีเซตการทำงานของไมโครคอนโทรลเลอร์ โดยการรีเซตต้องคงสถานะเป็น 1 อย่างน้อยนาน 2 แมชชีนไซเกิล ในขณะที่ออสซิลเลเตอร์ยังทำงานอยู่
- ขา $\overline{ALE}$/PROG เป็นขาสัญญาณเพื่อทำหน้าที่ควบคุมการแลตช์ (latch) ค่าตำแหน่งแอดเดรสไบต์ต่ำ (Address Latch Enable) เมื่อต้องการติดต่อกับหน่วยความจำภายนอก นอกจากนี้ขานี้ยังทำหน้าที่เป็นอินพุตรับพัลซ์ในการโปรแกรม (program pulse

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

input) ในส่วนของหน่วยความจำ EPROM สำหรับไมโครคอนโทรลเลอร์ในตระกูล MCS-51 ที่มีหน่วยความจำภายในเป็น EPROM

- ขา $\overline{\text{PSEN}}$ (Program Store Enable) ทำหน้าที่เป็นสัญญาณสโตรบเพื่ออ่านคำสั่งจากหน่วยความจำโปรแกรมภายนอก เมื่อไมโครคอนโทรลเลอร์ประมวลผลคำสั่งจากหน่วยความจำภายนอก ขานี้จะส่งสัญญาณสโตรบจำนวน 2 ครั้งในแต่ละแมชชีนไซเกิล แต่ในขณะที่ติดต่อกับหน่วยความจำข้อมูลภายนอกจะไม่มีการส่งสัญญาณสโตรบแต่อย่างใด
- ขา $\overline{\text{EA}}$ / VPP (External Access enable/VPP) เป็นขาสำหรับเลือกใช้หน่วยความจำโปรแกรมจากภายในหรือจากภายนอก โดยถ้ามีสถานะเป็น 0 จะหมายถึงให้ไมโครคอนโทรลเลอร์รับคำสั่งจากหน่วยความจำภายนอกที่ตำแหน่งแอดเดรส 0-0FFFH (0-1FFFH ถ้าเป็นเบอร์ 8052) อย่างไรก็ตามถ้าบิตป้องกัน (security bit) ในหน่วยความจำ EPROM ถูกโปรแกรมไว้ ไมโครคอนโทรลเลอร์จะไม่รับคำสั่งจากหน่วยความจำภายนอกเลย นอกจากนี้ขานี้ยังทำหน้าที่รับแรงดันไฟสำหรับการโปรแกรม (Vpp) ขนาด 21 โวลต์ เพื่อใช้ในระหว่างการโปรแกรม EPROM

2.2.3 โครงสร้างภายในของ MCS-51

การจัดหน่วยความจำ

ในไมโครคอนโทรลเลอร์ตระกูล MCS-51 แบ่งชนิดหรือหน้าที่ของหน่วยความจำออกเป็น 2 ส่วนคือ หน่วยความจำโปรแกรม (program memory) และหน่วยความจำข้อมูล (data memory)

หน่วยความจำโปรแกรมจะใช้สำหรับเก็บโปรแกรมควบคุมการทำงานของไมโครคอนโทรลเลอร์ซึ่งบางเบอร์จะมีหน่วยความจำส่วนนี้อยู่ภายในตัว โดยอาจจะมีขนาดไม่เท่ากันหรือเป็นหน่วยความจำต่างชนิดกัน เช่น บางเบอร์เป็น ROM และบางเบอร์อาจเป็น EPROM และบางเบอร์อาจไม่หน่วยความจำในส่วนนี้เลย โปรแกรมการทำงานจะถูกเก็บไว้ยังความจำโปรแกมภายนอกทั้งหมด

สำหรับหน่วยความจำข้อมูลจะใช้สำหรับเก็บข้อมูลหรือค่าตัวแปรต่างๆ จากการทำงานของโปรแกรม ซึ่งใน MCS-51 ทุกเบอร์จะมีหน่วยความจำในส่วนนี้อยู่จำนวนหนึ่ง แต่อาจมีขนาดมากน้อยต่างกันไปในแต่ละเบอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หน่วยความจำโปรแกรม

หน่วยความจำโปรแกรมสามารถ แบ่งออกได้เป็น 2 ส่วนคือ หน่วยความจำโปรแกรมภายในและหน่วยความจำโปรแกรมภายนอก หน่วยความจำโปรแกรมภายในจะถูกเลือกใช้งานถ้าขาสัญญาณ EA มีค่าเป็น 1 โดยจะถูกใช้งานในช่วงแอดเดรส 0-0FFFH (หรือช่วงแอดเดรส 0-1FFFH ในเบอร์ 8052) นอกเหนือจากช่วงแอดเดรสนี้จะใช้หน่วยความจำโปรแกรมภายนอกทั้งหมด ในกรณีตรงกันข้ามถ้าขาสัญญาณ EA มีค่าเป็น 0 ในช่วงแอดเดรส 0-0FFFH (หรือช่วงแอดเดรส 0-1FFFH ในเบอร์ 8052) จะถูกใช้จากหน่วยความจำภายนอก หรือกล่าวได้ว่า ถ้าขาสัญญาณ EA มีค่าเป็น 0 จะมีการเลือกใช้หน่วยความจำโปรแกรมภายนอกทั้งหมดตลอดช่วงแอดเดรส

หน่วยความจำข้อมูล

หน่วยความจำข้อมูลสามารถแบ่งออกได้เป็น 2 ส่วน คือ หน่วยความจำข้อมูลภายในและหน่วยความจำข้อมูลภายนอก สำหรับหน่วยความจำข้อมูลภายในยังแบ่งออกได้เป็น 2 ส่วนย่อย คือ ส่วนที่ใช้เก็บข้อมูลทั่วไปและส่วนที่ใช้เป็นรีจิสเตอร์หน้าที่พิเศษหรือ SFR (Special Function Register) โดยส่วนที่ใช้เก็บข้อมูลทั่วไปจะถูกใช้สำหรับเก็บข้อมูลหรือค่าตัวแปรต่างๆ จากการทำงานของโปรแกรม ส่วนรีจิสเตอร์หน้าที่พิเศษจะถูกใช้งานเป็นรีจิสเตอร์ควบคุมการทำงานและบอกสถานะการทำงานของไมโครคอนโทรลเลอร์

ไมโครคอนโทรลเลอร์ตระกูล MCS-51 ทุกเบอร์จะมีหน่วยความจำข้อมูภายในขนาด 128 ไบต์เป็นอย่างน้อย และบางเบอร์อาจจะมีถึง 256 ไบต์

รีจิสเตอร์หน้าที่พิเศษ (SFR)

รีจิสเตอร์หน้าที่พิเศษมีบทบาทอย่างมากในการควบคุมการทำงานของไมโครคอนโทรลเลอร์และทำให้การเขียนโปรแกรมสามารถทำได้สะดวกมากขึ้น รีจิสเตอร์หน้าที่พิเศษทำหน้าที่สำคัญคือควบคุมการทำงานในส่วนต่างๆ ภายในไมโครคอนโทรลเลอร์และทำหน้าที่แสดงสถานะการทำงานซึ่งในรีจิสเตอร์หน้าที่พิเศษบางตัวยังสามารถเข้าถึงได้ในระดับบิต (bit addressable) ด้วย

รีจิสเตอร์ใช้งานทั่วไป

รีจิสเตอร์ใช้งานทั่วไปมีไว้สำหรับให้ผู้เขียนโปรแกรมสามารถนำข้อมูลไปพักไว้ชั่วคราวหรือใช้งานทั่วไปได้ตามต้องการ ซึ่งรีจิสเตอร์ใช้งานทั่วไปนี้มีอยู่ด้วยกัน 8 ตัวคือ รีจิสเตอร์ R0-R7 โดยรีจิสเตอร์ทั้ง 8 ตัวถูกจัดให้อยู่รวมกันและมีให้เลือกใช้ถึง 4 แบงค์ (bank) นั่นคือมีรีจิสเตอร์ใช้งานถึง 32 ตัวให้ใช้งาน เพียงแต่การเลือกใช้รีจิสเตอร์ R0-R7 ในแบงค์ใดแบงค์หนึ่งจะถูกกำหนดจากบิต RS0,RS1 ในรีจิสเตอร์หน้าที่พิเศษ PSW ดังนั้นการเลือกใช้จึงเลือกได้เพียงแบงค์เดียวในขณะ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ใดขณะหนึ่ง อย่างไรก็ตามค่าข้อมูลที่เก็บไว้ในรีจิสเตอร์แบงค์ใดก็ตามที่มีชื่อเดียวกันแต่อยู่คนละแบงค์จะไม่มีผลซึ่งกันและกันเลย ทำให้ผู้เขียนโปรแกรมใช้งานรีจิสเตอร์ทั่วไปนี้ได้ทั้ง 32 ตัว อย่างเต็มที่และไม่ยุ่งยากในการเขียนโปรแกรม

2.3 การเขียนโปรแกรมภาษาแอสเซมบลี

โปรแกรมภาษาแอสเซมบลีเป็นโปรแกรม ที่เขียนขึ้นภายใต้โปรแกรมเวิร์ดโปรเซสเซอร์ซึ่งโปรแกรมนี้เรียกว่าซอร์สโค้ดโปรแกรม Source code program ในแต่ละบรรทัดจะบรรจุซึ่งสาระสำคัญต่างๆ ในรูปของคำสั่ง คำสั่งที่ถูกเขียนขึ้นเรียกว่านิวโมนิก (mnemonic) นอกจากในแต่ละบรรทัดประกอบด้วยคำสั่งไมโครคอนโทรลเลอร์แล้วยังมีส่วนของคำอธิบาย (comment) ด้วย คำอธิบายเหล่านี้มีประโยชน์อย่างยิ่งเพื่อให้ผู้อื่นทำความเข้าใจได้กับแต่ละคำสั่งที่เขียนขึ้นหรือแม้แต่ตัวผู้เขียนโปรแกรมเอง เมื่อต้องการพัฒนาโปรแกรมเพิ่มเติมจะได้ไม่สับสน

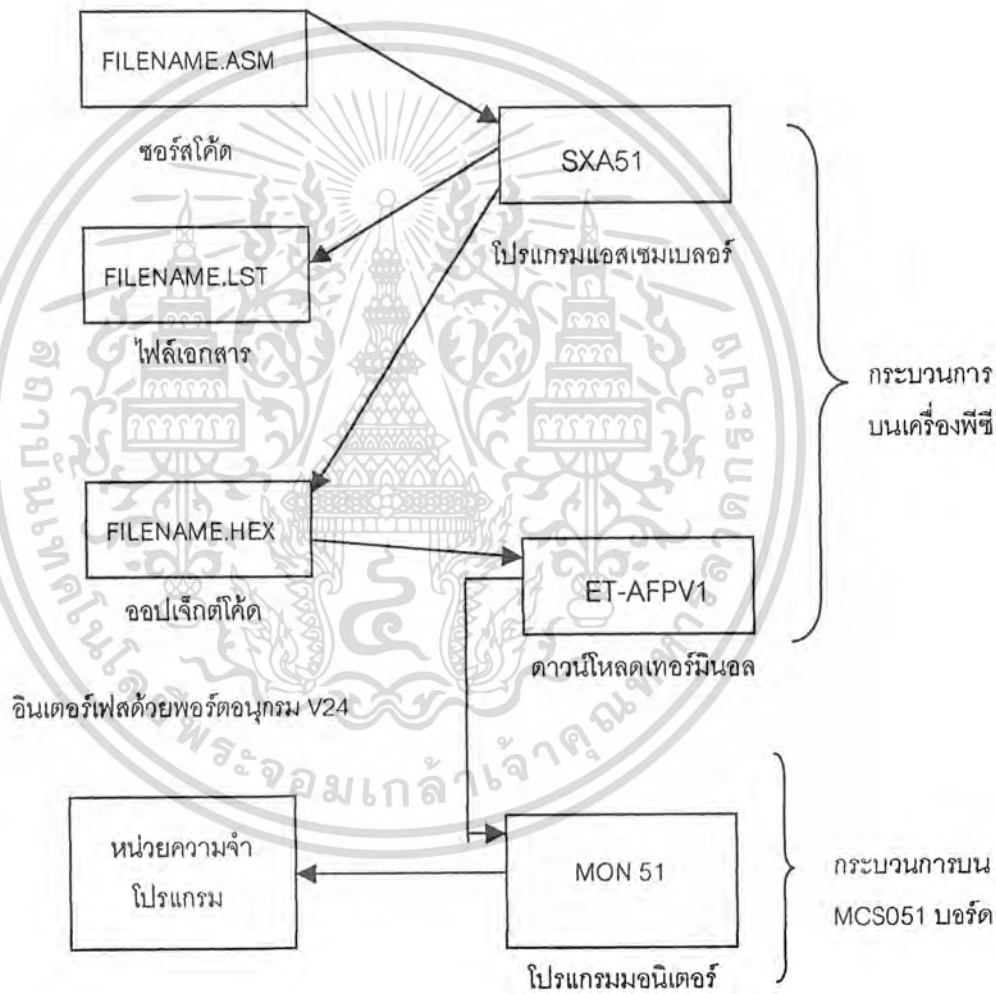
โปรแกรมแอสเซมบลียังไม่สามารถนำมาใช้งานกับไมโครคอนโทรลเลอร์ได้ทันที แต่ต้องนำมาทำการแปลงจากซอร์สโค้ดไฟล์เป็นภาษาเครื่อง (machine code) ทั้งคำสั่งต่าง ๆ ข้อมูลและการอ้างแอดเดรสทั้งหมดถูกแปลงไปเป็นโปรแกรมภาษาเครื่องที่ไมโครคอนโทรลเลอร์สามารถเข้าใจได้ หรือเรียกว่า ออปเจ็กต์โค้ด (object code) โดยใช้โปรแกรมแอสเซมเบลอร์ในที่นี้คือ SXA51.EXE มาใช้งาน หรืออาจกล่าวได้อีกอย่างหนึ่งคือโปรแกรมแอสเซมเบลอร์ทำการสร้างไฟล์ใหม่ขึ้นมาอีกไฟล์หนึ่ง โดยการนำเอาออปเจ็กต์โค้ดมาแทนที่คำสั่งหรือนิวโมนิกที่เขียนขึ้นโดยไม่สนใจคำอธิบายต่างๆ ที่เขียนไว้หรือตัดส่วนนี้ออกไปไม่นำมาใช้งานเลย

เอาท์พุตไฟล์ที่ได้จากการแอสเซมเบลอร์จะมีนามสกุลเป็น .HEX ไฟล์ที่ถูกแปลงเป็นภาษาเครื่องแล้ว จะอยู่ในรูปของเลขฐานสิบหกจำนวน 2 หลัก เรียงตามลำดับคำสั่งต่างๆ ที่เขียนขึ้นไฟล์นี้เองที่สามารถประมวลผลได้ทันที เมื่อมันถูกส่งผ่านไปเก็บไว้ที่หน่วยความจำบน MCS-51 บอร์ด หรือกล่าวได้ว่าเครื่องพีซีทำหน้าที่สร้างออปเจ็กต์โค้ดขึ้นมา และทำการส่งผ่านหรือดาวน์โหลดไปยัง MCS-51 บอร์ด โดยการติดต่อผ่านโปรแกรม ET-AFPV1 เพื่อส่งข้อมูลผ่านพอร์ตสื่อสารอนุกรม

นอกจาก EASM51 จะทำการสร้างออปเจ็กต์โค้ดขึ้นมาไฟล์หนึ่งแล้วมันยังสร้างไฟล์เอกสาร (list file) ขึ้นมาชุดหนึ่งมีนามสกุล .LST ไฟล์เอกสารนี้สร้างขึ้น เพื่อรวบรวมและแสดงออปเจ็กต์โค้ดที่สร้างโดยโปรแกรมแอสเซมเบลอร์จากซอร์สโค้ดโปรแกรมและข้อมูลอื่นๆ ที่สำคัญ ดังนั้นไฟล์เอกสารนี้เป็นประโยชน์อย่างยิ่ง เมื่อต้องการศึกษาการทำงานของโปรแกรมและการทำงานภายในไมโครคอนโทรลเลอร์ ตลอดจนตรวจสอบโปรแกรมที่เขียนขึ้นเพื่อพัฒนาในครั้งต่อ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ไปถึงแม้ว่าไฟล์เอกสารนี้ จะไม่สามารถนำมาใช้งานกับไมโครคอนโทรลเลอร์แต่มันก็มีประโยชน์ดังได้กล่าวมาแล้ว แสดงให้เห็นถึงกระบวนการในการแปลงโปรแกรมภาษาแอสเซมบลี จนกระทั่งเป็นโปรแกรมที่สามารถประมวลผลได้ทันทีกับ MCS-51 บอร์ด



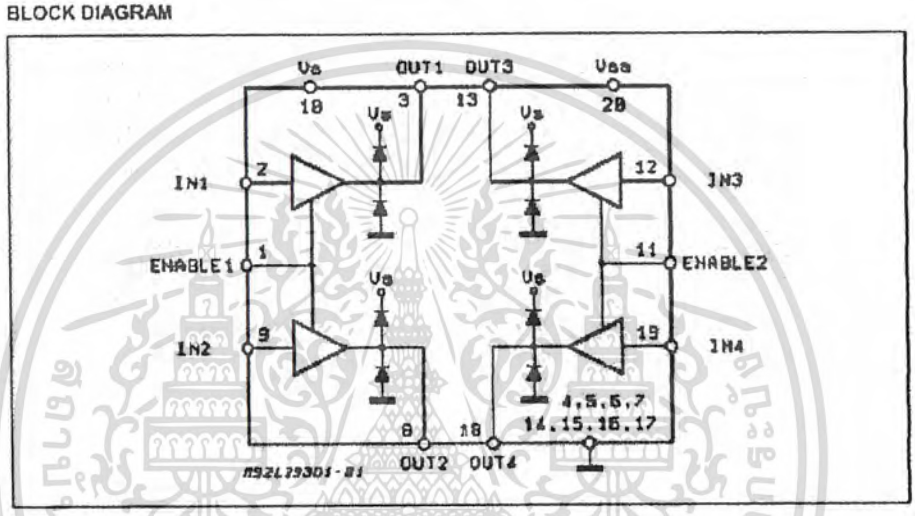
รูปที่ 2.5 กระบวนการต่างๆ ที่เกิดขึ้นจากซอร์สโค้ดที่เขียนบนเครื่องพีซีจนเป็นออปเจ็กต์โค้ด ซึ่งถูกโหลดมาเก็บไว้ในหน่วยความจำโปรแกรมบน MCS-51 บอร์ด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.4 IC#L293D (ไอซีขับมอเตอร์)

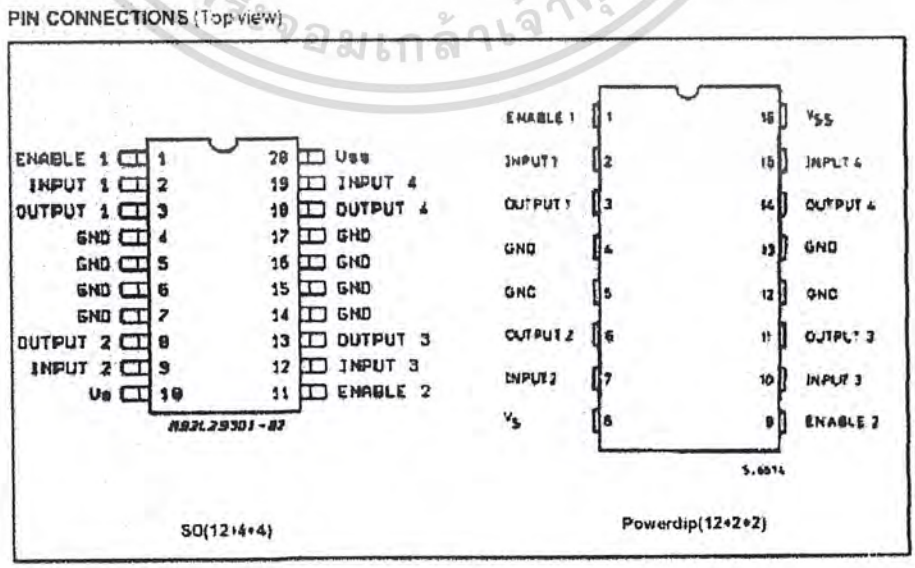
IC#L293D นี้ 1 ตัวสามารถใช้ในการขับมอเตอร์ไฟตรงได้ 2 ตัว โดยอาศัยคุณสมบัติของ L293D ที่สามารถใช้การควบคุมทาง ลอจิก ซึ่งเราสามารถเขียนโปรแกรมควบคุมผ่านทาง MCS51 ได้ทันที

2.4.1โครงสร้างภายในของ L293D



รูปที่ 2.6 โครงสร้างภายในของ L293D

2.4.2 โครงสร้างภายนอก L293D



รูปที่ 2.7 โครงสร้างภายนอกของ L293D

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.5 การทำงานเป็นไทเมอร์

เมื่อกำหนดให้ทำงานเป็นตัวตั้งเวลาหรือไทเมอร์ ค่าของรีจิสเตอร์จะเพิ่มขึ้นในทุก แมชีนไซเกิลดังนั้นเมื่อทำงานเป็นไทเมอร์ รีจิสเตอร์จะทำการนับค่าของแมชีนไซเกิลนั่นเอง และเนื่องจากแมชีนไซเกิลประกอบด้วยคาบเวลาของวงจรกำเนิดสัญญาณนาฬิกา 12 คาบเวลา ดังนั้นอัตราในการนับของรีจิสเตอร์จึงเท่ากับ □ ของความสัญญาณนาฬิกา

การทำงานเป็นเคาน์เตอร์

เมื่อทำงานเป็นตัวนับหรือเคาน์เตอร์ค่าของรีจิสเตอร์จะเพิ่มขึ้นก็ต่อเมื่อมีการเปลี่ยนแปลงของระดับลอจิกจาก "1" เป็น "0" เกิดขึ้นที่ขาอินพุตทางฮาร์ดแวร์ของวงจรไทเมอร์/เคาน์เตอร์ ซึ่งก็คือขา TO (p3.4) และขา T1 (p3.5) สำหรับไมโครคอนโทรลเลอร์ MCS – 51 แบบแฟลชเอบร์ AT89C51 รวมทั้งขา T2 (p1.0) ในไมโครคอนโทรลเลอร์ MCS-51 แบบแฟลชเบอร์ AT89C52 และอนุกรม AT89Sxx โดยจะมีการสุ่มรับสัญญาณจากขาอินพุตในทุก ๆ คาบเวลาที่ 2 ของสเตตที่ 5 (S5P2) ในแต่ละแมชีนไซเกิล

เมื่อสัญญาณอินพุตเปลี่ยนแปลงจาก "1" เป็น "0" เป็นเวลาหนึ่งไซเกิล ในไซเกิลต่อมาค่าของการนับจะเพิ่มขึ้นหนึ่งค่า และจะไปปรากฏในรีจิสเตอร์ภายในคาบเวลาที่ ! ของสเตตที่ 3 (S3P1) ของแมชีนไซเกิลต่อไปหลังจากที่ตรวจจับพบการเปลี่ยนแปลงที่ขาไทเมอร์จะต้องใช้ 2 แมชีนไซเกิล อัตราการนับของเคาน์เตอร์จึงเท่ากับ 1/24 ของความถี่สัญญาณนาฬิกา ดังนั้น ความถี่สูงสุดของสัญญาณอินพุตที่ไทเมอร์ / เคาน์เตอร์ภายในไมโครคอนโทรลเลอร์ MCS -51 แบบแฟลชสามารถตรวจจับได้จึงเท่ากับความถี่ของของสัญญาณนาฬิกาหารด้วย 24 ยกตัวอย่างในไมโครคอนโทรลเลอร์ MCS-51 แบบแฟลชเบอร์ AT89C51 สามารถใช้สัญญาณนาฬิการได้สูงสุด 24MHz ดังนั้นความถี่สูงสุดของสัญญาณอินพุตที่ไทเมอร์/เคาน์เตอร์สามารถตรวจจับได้คือ 1MHz

รีจิสเตอร์ที่เกี่ยวข้องกับการทำงานของไทเมอร์/เคาน์เตอร์ 0 และ 1

ในการทำงานของไทเมอร์/เคาน์เตอร์ 0 และ 1 ในไมโครคอนโทรลเลอร์ MCS-51 แลลแฟลช มีรีจิสเตอร์ที่ต้องเกี่ยวข้องเป็นพื้นฐานอยู่ 6 ตัว ดังมีรายละเอียดต่อไปนี้

รีจิสเตอร์ไทเมอร์

มีด้วยกัน 4 ตัวคือ TLO มีแอดเดรสอยู่ที่ 8AH, THO มีแอดเดรสอยู่ที่ 8CH, TL1 มีแอดเดรสอยู่ที่ 8BH และ TH1 มีแอดเดรสอยู่ที่ 8DH รีจิสเตอร์ทั้ง 4 ตัวจะอยู่ในพื้นที่ของรีจิสเอร์ฟังก์ชั่นพิเศษหรือ SFR รีจิสเตอร์แต่ละตัวมีขนาด 8 บิด แต่ในการใช้งานโดยทั่วไปมักใช้ร่วมกันโดยจัดเป็น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

คู่คือ TLO กับ THO รวมกันเป็นรีจิสเตอร์ Timer 0 ขนาด 16 บิต และ TL1 กับ TH1 รวมกันเป็นรีจิสเตอร์Timer 1 ขนาด 16 บิต โดยใน TLO และ TL1 เก็บข้อมูล 8 บิตล่าง ส่วน THO และ TH1 เก็บข้อมูล 8 บิตบน รีจิสเตอร์ไทเมอร์ทั้ง 2 คู่เมื่อนำมาใช่ร่วมกันจะสามารถเก็บค่าของการนับได้สูงสุด 65,536 หรือ FFFFH เมื่อนับถึงค่านี้แล้วจะวนไปเริ่มนับ 0000H ใหม่และเมื่อเกิดการนับรอบใหม่ บิต TFOหรือ TF1 ในรีจิสเตอร์ ที่ใช้ควบคุมการทำงานของไทเมอร์จะเกิดการเซต เพื่อแจ้งให้ทราบว่า นับเกิดค่าสูงสุดแล้ว การเซตบิต หรือ ขึ้นอยู่กับว่าเลือกใช้งานรีจิสเตอร์ไทเมอร์ตัวใด

รีจิสเตอร์ควบคุมการทำงานของไทเมอร์/เคาน์เตอร์หรือ
TCON (Timer/Counter Control Register)

เป็นรีจิสเตอร์ขนาด 8 บิต มีแอดเดรสอยู่ที่ 88H ในพื้นที่ของรีจิสเตอร์ SFR สามารถเข้าถึงได้ในระดับบิตมีรายละเอียดการทำงานดังนี้

บิต 7	บิต 6	บิต 5	บิต 4	บิต 3	บิต 2	บิต 1	บิต 0
TF1	TR1	TF0	TR0	IE1	IT1	IE0	IT0

TF1 (Timer 1 overflow flag) : เซตด้วยกระบวนการทางฮาร์ดแวร์ เมื่อค่าของรีจิสเตอร์ Timer 1 เกิดการนับเกินหรือเกิดโอเวอร์โฟลว การเคลียร์นี้ทำได้ด้วยกระบวนการทางฮาร์ดแวร์เช่นกัน โดยบิตนี้จะเคลียร์เมื่อมีการอินเตอร์รัปต์เกิดขึ้น

TR1 (Timer 1 run control bit) : ใช้ในการเปิดปิดการทำงานของไทเมอร์ (เอ็นเอเบิลหรือดิสเอเบิล) ทำการเซตและเคลียร์ด้วยกระบวนการทางซอฟต์แวร์ ถ้าต้องการให้ไทเมอร์ 1 ทำงานต้องเซตบิตนี้ให้เป็น "1"

TF0 (Timer 0 overflow flag) : เซตด้วยกระบวนการทางฮาร์ดแวร์ เมื่อค่าของรีจิสเตอร์ Timer 0 เกิดการนับเกินหรือเกิดโอเวอร์โฟลว การเคลียร์บิตนี้ทำได้ด้วยกระบวนการทางฮาร์ดแวร์เช่นกัน โดยบิตนี้จะเคลียร์เมื่อมีการอินเตอร์รัปต์เกิดขึ้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

TR0 (Timer 0 run controol bit) : ใช้ในการเปิดปิดการทำงานของไทเมอร์ 0 (เอ็นเอเบิลหรือดิสเอเบิล) ทำการเซตและเคลียร์ด้วยกระบวนการทางซอฟต์แวร์ ถ้าต้องการให้ไทเมอร์ 0 ทำงานต้องเซตบิตนี้ให้เป็น "1"

IE1 (External Interrupt 1 edge flag) : บิตนี้จะให้ในกระบวนการอินเตอร์รัปต์ สามารถเซตได้ด้วยกระบวนการทางฮาร์ดแวร์ เมื่อสามารถตรวจจับขอบขาของสัญญาณอินเตอร์รัปต์จากภายนอกที่ขาอินพุตอินเตอร์รัปต์ 1 (INT1) ได้ และจะทำการเคลียร์เมื่อมีการบริการอินเตอร์รัปต์เกิดขึ้น

IT1 (Interrupt 1 type control bit) : บิตนี้จะให้ในกระบวนการอินเตอร์รัปต์โดยใช้ในการเลือกลักษณะของสัญญาณอินเตอร์รัปต์จากภายนอกที่ต้องการให้ทำการตอบสนองสำหรับขาอินพุตอินเตอร์รัปต์ 1 (INT!) การเซตและเคลียร์ทำได้ด้วยกระบวนการซอฟต์แวร์

"0" เลือกขอบขาลงของสัญญาณ (falling edge)

"1" เลือกระดับลอจิกต่ำ (low level triggered)

IED (External Interrupt 0 edge flag) : บิตนี้จะใช้ในกระบวนการอินเตอร์รัปต์ สามารถเซตได้ด้วยกระบวนการทางฮาร์ดแวร์ เมื่อสามารถตรวจจับขอบขาของสัญญาณอินเตอร์รัปต์จากภายนอกที่ขาอินพุตอินเตอร์รัปต์ 0 (INT0) ได้ และจะทำการเคลียร์เมื่อมีการบริการอินเตอร์รัปต์เกิดขึ้น

IT0 (Interrupt 0 type control bit): บิตนี้จะใช้ในกระบวนการอินเตอร์รัปต์ โดยใช้ในการเลือกลักษณะของสัญญาณอินเตอร์รัปต์จากภายนอกที่ต้องการให้ทำการตอบสนองสำหรับขาอินพุตอินเตอร์รัปต์ 0 (INT0) การเซตและเคลียร์ทำได้ด้วยกระบวนการซอฟต์แวร์

"0" เลือกขอบขาลงของสัญญาณ (falling edge)

"1" เลือกระดับลอจิกต่ำ (low level triggered)

รีจิสเตอร์เลือกโหมดการทำงานของไทเมอร์/เคาน์เตอร์หรือ

TMOD (Timer/Counter Mode Control Register)

เป็นรีจิสเตอร์ขนาด 8 บิต มีแอดเดรสอยู่ที่ 89H ในพื้นที่ของรีจิสเตอร์ SFR ไม่สามารถเข้าถึงได้ในระดับบิต แบ่งการทำงานออกเป็น 2 ส่วนคือ 4 บิตล่างใช้ในการเลือกโหมดการทำงานของ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ไทเมอร์ 0 และ 4 บิตบนใช้ในการเลือกโหมดการทำงานของไทเมอร์ 1 ดังนั้นในการอธิบายการทำงานจะของอธิบายเพียงส่วนเดียวดังนี้

บิต 7	บิต 6	บิต 5	บิต 4	บิต 3	บิต 2	บิต 1	บิต 0
GATE	C/$\overline{T}$	M1	M0	GATE	C/$\overline{T}$	M1	M0
ไทเมอร์ 1				ไทเมอร์ 0			

GATE : ใช้เลือกลักษณะการควบคุมการทำงานของไทเมอร์/เคาน์เตอร์

"0" ไทเมอร์/เคาน์เตอร์จะทำงานเมื่อบิต TRxในรีจิสเตอร์ TCON เป็น "1" เรียกการควบคุมแบบนี้ว่าการควบคุมทางซอฟต์แวร์

"1" ไทเมอร์/เคาน์เตอร์จะทำงานเมื่อบิต TRxในรีจิสเตอร์ TCON เป็น "1" และสถานะลอจิกที่ขาอินพุตอินเตอร์รัปต์ INT0 และ INT1 เป็น "1" เรียกการควบคุมแบบนี้ว่า การควบคุมทางฮาร์ดแวร์

C/T (Timer or Counter selector): ใช้เลือกลักษณะการทำงานของไทเมอร์/เคาน์เตอร์

"0" เลือกให้ทำงานเป็นไทเมอร์ โดยใช้สัญญาณอินพุตจากสัญญาณนาฬิกาภายในไมโครคอนโทรลเลอร์

"1" เลือกให้ทำงานเป็นเคาน์เตอร์ โดยรับสัญญาณอินพุตทางขา T0 หรือ T1

M1, M0 (Mode selector bit): ใช้เลือกโหมดการทำงานของไทเมอร์/เคาน์เตอร์

"00" เลือกให้ทำงานในโหมดไทเมอร์/เคาน์เตอร์ 13 บิต

"01" เลือกให้ทำงานในโหมดไทเมอร์/เคาน์เตอร์ 16 บิต

"10" เลือกให้ทำงานในโหมดไทเมอร์/เคาน์เตอร์ขนาด 8 บิตแบบตั้งค่าอัตโนมัติ

"11" สำหรับไทเมอร์ 0 เลือกให้ทำงานในโหมดไทเมอร์/เคาน์เตอร์แยกส่วน โดยแยกออกเป็นไทเมอร์/เคาน์เตอร์ 8 บิต 2 ตัว รีจิสเตอร์ TL0 จะได้รับการควบคุมการเปิดปิดจากบิต TR0 ในรีจิสเตอร์ TCON และรีจิสเตอร์ TH0 ซึ่งเป็นไทเมอร์/เคาน์เตอร์ 8 บิตอีกตัวหนึ่ง จะได้รับการควบคุมจากบิต TR1 ในรีจิสเตอร์ TCON ในกรณีของไทเมอร์ 1 เป็นการสั่งให้ไทเมอร์/เคาน์เตอร์ 1 หยุดทำงาน (ดิสเอเบิล)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สำนักหอสมุดกลาง พระจอมเกล้าลาดกระบัง

โหมดการทำงานของไทเมอร์/เคาน์เตอร์ 0 และ 1

ไทเมอร์ 0 และ ไทเมอร์ 0 สามารถเลือกโหมดการทำงานได้ 4 โหมดคือ โหมด 0:ไทเมอร์/เคาน์เตอร์ 13 บิต (13 bit timer/counter), โหมด 1: ไทเมอร์/เคาน์เตอร์ 16 บิต (16 bit timer/counter), โหมด 2:ตั้งค่าอัตโนมัติขนาด 8 บิต (8 bit auto-reload timer/counter) และโหมด 3: ไทเมอร์/เคาน์เตอร์แยกส่วน (split timer/counter)หรืออาจเรียกว่าโหมดไทเมอร์/เคาน์เตอร์ 8 บิต ก็ได้ในขณะที่ไทเมอร์ 2 มีโหมดการทำงาน 3 โหมดคือ โหมดแคปเจอร์ หรือตรวจจับสัญญาณ โหมดตั้งค่าอัตโนมัติ(auto-reload) และโหมดกำเนิดอัตราเร็วในการสื่อสารข้อมูลอนุกรมหรืออัตราบอด (baud rate generator)

การเลือกโหมดการทำงานของไทเมอร์/เคาน์เตอร์ 0และ 1 สามารถทำได้ที่รีจิสเตอร์ TCON และ TMOD ร่วมกัน โดย TCON ใช้ในการเอ็นเอเบิลหรือดิสเอเบิลไทเมอร์/เคาน์เตอร์ ส่วน TMOD ใช้ในการเลือกโหมดและลักษณะการทำงาน ในขณะที่การทำงานของไทเมอร์ 2 จะอธิบายแยกต่างหาก

การทำงานในโหมด 0 : ไทเมอร์/เคาน์เตอร์ 13 บิต

มีไดอะแกรมการทำงานแสดงในรูปที่ 6-1 ในที่นี้จะใช้ไทเมอร์/เคาน์เตอร์ 1 ในการอธิบาย โหมดนี้จะเป็นการกำหนดให้ใช้รีจิสเตอร์ TL1 เพียง 5 บิต และ TH1 ครบ 8 บิต โดยTL0 จะทำหน้าที่คล้ายกับเป็นปรีสเกลเลอร์หาร 32 สัญญาณอินพุตสำหรับการนับจะเลือกจากสัญญาณนาฬิกาภายในหรือภายนอกผ่านทางขา T1 ขึ้นอยู่กับการควบคุมของบิตC/T และ GATE ในรีจิสเตอร์ TMOD บิต TR1 ในรีจิสเตอร์ TCON และสถานะของลอจิกที่ขาอินพุต INT1 เมื่อ TL1 นับครบ 32 คือจาก 0-31 ก็จะส่งสัญญาณไปยัง TH1 เพื่อทำการเพิ่มค่า ดังนั้นในโหมดนี้ค่าของการนับจะมีขนาด 13 บิต เมื่อทำการนับครบรอบ ก็จะทำการเซตบิต TF1 ในรีจิสเตอร์TCON

ส่วนการทำงานในโหมดนี้ของไทเมอร์/เคาน์เตอร์ 0 มีลักษณะเหมอนกันทุกประการ เพียงแต่เปลี่ยนรีจิสเตอร์และขาสัญญาณที่เกี่ยวข้องให้เป็นของไทเมอร์/เคาน์เตอร์ 0

การทำงานในโหมด 1 : ไทเมอร์/เคาน์เตอร์ 16 บิต

มีไดอะแกรมการทำงานแสดงในรูปที่ 6-2 ในที่นี้จะใช้ไทเมอร์ 1 ในการอธิบาย การทำงานในโหมดนี้จะคล้ายกับโหมด 0 แต่จะใช้งานรีจิสเตอร์ TL1 และ TH1 ครบ 8 บิต ดังนั้นในโหมดนี้ค่าของการนับจะมีขนาด 16 บิต คือ 0000H-FFFFH เมื่อทำการนับครบรอบ ค่าของการนับเปลี่ยนจาก FFFFH เป็น 0000H ก็จะเซตบิต TF1 ในรีจิสเตอร์ TCON ส่วนการทำงานในโหมดนี้ของ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

42254

ไทเมอร์ 0 มีลักษณะเหมือนกันทุกประการเพียงแต่เปลี่ยนรีจิสเตอร์และขาสัญญาณที่เกี่ยวข้องให้เป็นของไทเมอร์ 0

การทำงานในโหมด 2 : ไทเมอร์/เคาน์เตอร์ 8 บิตแบบตั้งค่าอัตโนมัติ

มีไดอะแกรมการทำงานแสดงในรูปที่ 6-3 ในที่นี้จะใช้ไทเมอร์/เคาน์เตอร์ 1 ในการอธิบายการทำงานในโหมดนี้จะแยกรีจิสเตอร์ไทเมอร์ออกเป็น 2 ตัว ตัวละ 8 บิต โดยรีจิสเตอร์ TL1 ทำหน้าที่เป็นตัวนับค่า ส่วน TH1 ใช้ในการเก็บค่าเริ่มต้นของการนับ เมื่อเริ่มต้นการทำงาน ค่าของรีจิสเตอร์ TH1 จะถูกส่งไปยังรีจิสเตอร์ TL1 ทำให้เมื่อเริ่มต้นการทำงานค่าของรีจิสเตอร์TL1 และ TH1 จะเหมือนกัน เมื่อ TL1 นับถึง FFH และจะเริ่มต้นการนับรอบใหม่ จะทำการเซตบิต TF1 พร้อม ๆ กับทำการรับค่าการนับเริ่มต้นจาก TH1 ใหม่โดยอัตโนมัติ หรือเรียกกระบวนการนี้ว่ารีโหลด (reloard) เม้ว่าจะมีการส่งค่าเริ่มต้นไปยัง TL1 แล้วก็ตาม ค่าข้องข้อมูลในรีจิสเตอร์ TH1 ก็ยังคงเป็นค่าเดิม ไม่มีการเปลี่ยนแปลง จนกว่าจะมีการกำหนดค่าใหม่ด้วยกระบวนการทางซอฟต์แวร์

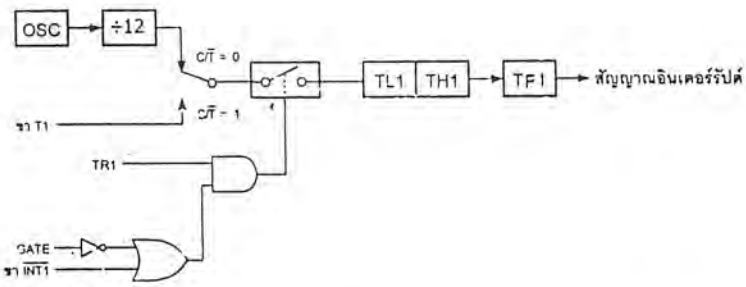
ส่วนการทำงานในโหมดนี้ของไทเมอร์/เคาน์เตอร์ 0 จะมีลักษณะเหมือนกันทุกประการเพียงแต่เปลี่ยนรีจิสเตอร์และขาสัญญาณที่เกี่ยวข้องให้เป็นของไทเมอร์/เคาน์เตอร์ 0

การทำงานในโหมด 3 : ไทเมอร์/เคาน์เตอร์แยกส่วนหรือไทเมอร์/เคาน์เตอร์ 8 บิต

ในโหมดนี้เป็นโหมดเดียวที่การทำงานของไทเมอร์ 0 และไทเมอร์ 1 ไม่เหมือนกัน ขออธิบายในส่วนของไทเมอร์ 1 ก่อน เมื่อเข้าสู่โหมดนี้ จะเป็นการสั่งให้ไทเมอร์/เคานีเตอร์หยุดนับ ค่าของการนับก่อนหน้านี้จะถูกเก็นไว้ในรีจิสเตอร์ไทเมอร์ 1 มีลักษณะการทำงานเหมือนกับการดิสเอเบิลไทเมอร์/เคาน์เตอร์ 1 ด้วยการเคลียร์บิต TR1 ในรีจิสเตอร์ TCON

ส่วนการทำงานของไทเมอร์ 0 ในโหมดนี้มีไดอะแกรมการทำงานแสดงในรูปที่ 6-4 การทำงานในโหมดนี้จะแยกรีจิสเตอร์ไทเมอร์ 0 ออกเป็น 2 ตัว ตัวละ 8 บิต คือรีจิสเตอร์ TL0 และ TH โดยแยกการทำงานออกจากกัน รีจิสเตอร์ TL0 สามารถเลือกการทำงานได้เหมือนกับไทเมอร์/เคาน์เตอร์ตามปกติ ส่วนรีจิสเตอร์ TH0 สามารถทำงานในโหมดไทเมอร์ได้เพียงอย่างเดียว กล่าวคือ สามารถรับสัญญาณอินพุตจากสัญญาณนาฬิกาภายในเพียงทางเดียวเท่านั้น แต่การแจ้งการนับเกินยังคงเหมือนเดิม หากแต่ TL0 แจ้งผ่าบิต TF0 ในขณะที่ จะแจ้งผ่านทางบิต TF1

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.8 ไดอะแกรมการทำงานในโหมด 0 ของไทเมอร์/เคาน์เตอร์ 1

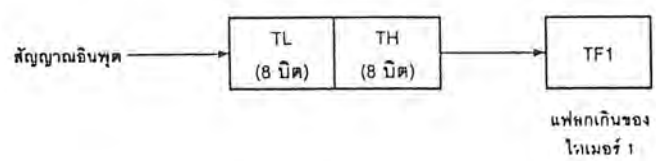
ข้อมูลที่ใช้ในการเลือกโหมดการทำงานของไทเมอร์/เคาน์เตอร์ 0 และ 1

เนื่องจากมีตัวแปรอยู่หลายตัวที่ใช้ในการควบคุมและเลือกโหมดการทำงานของไทเมอร์/เคาน์เตอร์ 0 และ 1 เพื่อให้เกิดความสะดวกในการใช้งานจึงได้ทำการสรุปข้อมูลที่ใช้ในการกำหนดรูปแบบการทำงานของไทเมอร์/เคาน์เตอร์ 0 และ 1 ไว้ในรางที่ 6-1 ถึง 6-4

อย่างไรก็ตาม ข้อมูลที่นำมายกเป็นตัวอย่างขั้นต้นนี้ สามารถเปลี่ยนแปลงได้ตามความต้องการของผู้ใช้งานไม่จำเป็นต้องยึดค่าเหล่านี้ไว้เสมอไป แต่สำหรับผู้เริ่มต้นใช้งานควรใช้ค่าตัวอย่างที่ให้ไว้ในการเขียนโปรรแกรมควบคุมก่อน จนกว่ามีความชำนาญจึงค่อยปรับเปลี่ยนต่อไป

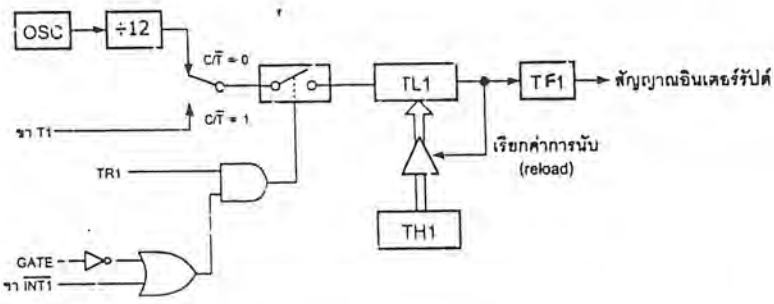
ไทเมอร์/เคาน์เตอร์ 2 ในไมโครคอนโทรลเลอร์ MCS - 51

ในไมโครคอนโทรลเลอร์ MCS-51 แบบแฟลชเบอร์ AT89C52 และในอนุกรม AT89Sxx ยังมีไทเมอร์/เคาน์เตอร์ขนาด 16 บิต ให้ใช้งานเพิ่มเติมอีก 1 ตัวคือ ไทเมอร์/เคาน์เตอร์ 2 (Timer/Counter 2) รีจิสเตอร์ฟังก์ชั่นพิเศษที่เกี่ยวข้องกับไทเมอร์/เคาน์เตอร์ 2 มีเพิ่มเติมอีก 5 ตัว คือ รีจิสเตอร์ไทเมอร์ 2 ซึ่งประกอบด้วยTL2 และ TH2 รีจิสเตอร์ T2CON รีจิสเตอร์ T2MOD มีแอดเดรสอยู่ที่ C9H สุดท้ายคือรีจิสเตอร์แคปเจอร์ ซึ่งประกอบด้วย RCAP2L และ RCAP2H มีแอดเดรสอยู่ที่ CAH และ CBH



รูปที่ 2.9 ไดอะแกรมการทำงานในโหมด 1 ของไทเมอร์/เคาน์เตอร์ 1

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.10 ไดอะแกรมการทำงานในโหมด 2 ของไทเมอร์/เคาน์เตอร์ 1

ไทเมอร์/เคาน์เตอร์ 2 สามารถทำงานเป็นไทเมอร์หรือตัวตั้งเวลาและเคาน์เตอร์หรือตัวนับได้เช่นเดียวกับไทเมอร์/เคาน์เตอร์ 0 และ 1 โดยการกำหนดที่บิตC/T ในรีจิสเตอร์ T2CON เมื่อเทียบกับไทเมอร์/เคาน์เตอร์ 0 และ 1 ก็คือรีจิสเตอร์ TCON สำหรับโหมดการทำงานของไทเมอร์/เคาน์เตอร์ 2 มีด้วยกัน 3 โหมดคือ โหมดแคปเจอร์หรือตรวจจับสัญญาณ (capture), โหมดตั้งค่าอัตโนมัติ (auto – reload) และโหมดกำเนิดอัตราเร็วในการสื่อสารข้อมูลอนุกรมหรืออัตราบอด (baud rate generator)

รีจิสเตอร์ที่เกี่ยวข้องกับการทำงานของไทเมอร์/เคาน์เตอร์ 2

ในการทำงานของไทเมอร์/เคาน์เตอร์ 2 ในไมโครคอนโทรลเลอร์ MCS-51 แบบแฟลช มีรีจิสเตอร์ที่ต้องเกี่ยวข้องเป็นพื้นฐานอยู่ 5 ตัว ดังมีรายละเอียดต่อไปนี้

รีจิสเตอร์ไทเมอร์

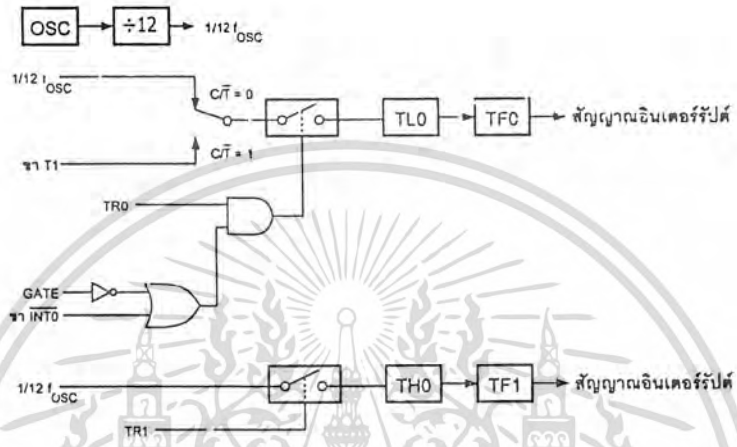
มีด้วยกัน 2 ตัวคือ TL2 มีแอดเดรสอยู่ที่ CCH และ TH2 มีแอดเดรสอยู่ที่ CDH รีจิสเตอร์ทั้ง 2 ตัวจะอยู่ในพื้นที่ของรีจิสเตอร์ฟังก์ชั่นพิเศษหรือ SFR รีจิสเตอร์แต่ละตัวมีขนาด 8 บิต แต่ในการใช้งานโดยทั่วไปมักใช้ รวมกันเป็นรีจิสเตอร์ไทเมอร์ 2 ขนาด 16 บิต โดยใน TL2 จะเก็บข้อมูล 8 บิตล่าง ส่วน TH2 เก็บข้อมูล 8 บิตบน รีจิสเตอร์ไทเมอร์ทั้งคู่เมื่อนำมาใช้งานร่วมกันจะสามารถเก็บค่าของการนับได้สูงสุด 65,536 ค่าหรือ FFFFH เมื่อนับถึงค่านี้แล้วก็จะวนไปเริ่มนับ 0000H ใหม่ และเมื่อเกิดการนับรอบใหม่ จะมีการเซตบิต TF2 ในรีจิสเตอร์ T2CON ที่ใช้ควบคุมการทำงานของไทเมอร์ เพื่อแจ้งให้ทราบว่าเกิดการนับเกินค่าสูงสุด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รีจิสเตอร์ควบคุมการทำงานของไทเมอร์/เคาน์เตอร์ 2 หรือ

T2CON (Timer/counter 2 Control Register)

เป็นรีจิสเตอร์ขนาด 8 บิต มีแอดเดรสอยู่ที่ C8H ในพื้นที่ของรีจิสเตอร์ SFR สามารถเข้าถึงได้ในระดับบิตมีรายละเอียดการทำงานดังนี้



รูปที่ 2.11 ไดอะแกรมการทำงานในโหมด 3 ของไทเมอร์/เคาน์เตอร์ 1

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

TF2 (Timer 2 overflow flag): จะเซตเมื่อค่าของรีจิสเตอร์ Timer 2 เกิดการนับเกินหรือเกิดโอเวอร์โฟลวการเคลียร์บิตนี้ทำได้ด้วยกระบวนการทางซอฟต์แวร์ และจะไม่เกิดการเซตถ้าบิต RCLK หรือบิต TCLK เป็น "1"

EXF2 (Timer 2 external flag): จะเกิดการเซตเมื่อเกิดการแคปแจร์หรือรีโหลดขึ้น ซึ่งเกิดจากการเปลี่ยนแปลงระดับลอจิกจาก "1" เป็น "0" ที่ขาอินพุต T2EX (ขา P.1) และบิต EXEN2 เป็น "1" ในกรณีที่อินเตอร์รัปต์ในไทเมอร์ 2 ได้รับการเอ็นเอเบิน และบิต EXF2 นี้เกิดเซต จะมีผลทำให้ซีพียูไปอ่านค่าเวกเตอร์ของการบริการอินเตอร์รัปต์ บิดนี้สามารถเคลียร์ด้วยกระบวนการทางซอฟต์แวร์เท่านั้น

RCLK (Receive clock flag): ถ้าบิตนี้เกิดการเซต ทำให้วงจรพอร์ดอนุกรมภายในไมโครคอนโทรลเลอรีใช้พัลส์โอเวอร์โฟลวของไทเมอร์ 2 เป็นสัญญาณในการในการรับข้อมูลในโหมด 1 และ 3 ถ้า

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หากบิตนี้เป็น "0" จะใช้พัลส์จากไทเมอร์ 1 (มีรายละเอียดเพิ่มเติมในบทที่ว่าด้วยการสื่อสารผ่านพอร์ตอนุกรม)

TCLK (Transmit clock flag): ถ้าบิตนี้เกิดการเซต ทำให้วงจรพอร์ตอนุกรมภายในไมโครคอนโทรลเลอร์ใช้พัลส์โอเวอร์โฟลวของไทเมอร์ 2 เป็นสัญญาณนาฬิกาในการส่งข้อมูลในโหมด 1 และ 3 ถ้าหากบิตนี้เป็น "0" จะใช้พัลส์จากไทเมอร์ 1 (มีรายละเอียดเพิ่มเติมในบทที่ว่าด้วยการสื่อสารผ่านพอร์ตอนุกรม)

EXEN2 (Timer 2 external enable flag): เมื่อบิตนี้เซตเป็น "1" จะเป็นการเอ็นเอเบิลการแคปเจอรีหรือรีโหลดที่ขา T2EX ให้เกิดขึ้นได้ภายใต้เงื่อนไขว่าไทเมอร์ 2 ต้องไม่ถูกนำไปใช้ในการสื่อสารพอร์ตอนุกรม หรือ บิต RCLK หรือ TCLK เป็น "1" ถ้าบิตนี้เป็น "0" ไทเมอร์ 2 จะไม่สนใจเหตุการณ์ที่ขา T2EX

โหมด	ฟังก์ชั่นของไทเมอร์ 0	TMOD	
		การควบคุมจากภายใน	การควบคุมจากภายนอก
0	ไทเมอร์/เคาน์เตอร์ 13 บิต	C0H	08H
1	ไทเมอร์/เคาน์เตอร์ 16 บิต	01H	09H
2	8 บิตตั้งค่าอัตโนมัติ	02H	0AH
3	ไทเมอร์/เคาน์เตอร์แยกส่วน	03H	0BH

ตารางที่ 2.1 ข้อมูลของรีจิสเตอร์ TMOD เพื่อกำหนดให้ไทเมอร์/เคาน์เตอร์ 0 ทำงานเป็นไทเมอร์

โหมด	ฟังก์ชั่นของไทเมอร์ 0	TMOD	
		การควบคุมจากภายใน	การควบคุมจากภายนอก
0	ไทเมอร์/เคาน์เตอร์ 13 บิต	04H	0CH
1	ไทเมอร์/เคาน์เตอร์ 16 บิต	05H	0DH
2	8 บิตตั้งค่าอัตโนมัติ	06H	0EH
3	ไทเมอร์/เคาน์เตอร์แยกส่วน	07H	0FH

ตารางที่ 2.2 ข้อมูลของรีจิสเตอร์ TMOD เพื่อกำหนดให้ไทเมอร์/เคาน์เตอร์ 0 ทำงานเป็นเคาน์เตอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โหมด	ฟังก์ชั่นของไทเมอร์ 1	TMOD การควบคุมจากภายใน	TMOD การควบคุมจากภายนอก
0	ไทเมอร์/เคาน์เตอร์ 13 บิต	00H	80H
1	ไทเมอร์/เคาน์เตอร์ 16 บิต	10H	90H
2	8 บิตตั้งค่าอัตโนมัติ	20H	A0H
3	หยุดทำงาน	30H	B0H

ตารางที่ 2.3 ข้อมูลของรีจิสเตอร์ TMOD เพื่อกำหนดให้ไทเมอร์/เคาน์เตอร์ 1 ทำงานเป็นไทเมอร์

โหมด	ฟังก์ชั่นของไทเมอร์ 1	TMOD การควบคุมจากภายใน	TMOD การควบคุมจากภายนอก
0	ไทเมอร์/เคาน์เตอร์ 13 บิต	40H	C0H
1	ไทเมอร์/เคาน์เตอร์ 16 บิต	50H	D0H
2	8 บิตตั้งค่าอัตโนมัติ	60H	E0H
3	-	-	-

ตารางที่ 2.4 ข้อมูลของรีจิสเตอร์ TMOD เมื่อกำหนดให้ไทเมอร์/เคาน์เตอร์ 1 ทำงานเป็นเคาน์เตอร์

TR2: บิตนี้จะใช้ในควบคุมการเริ่มต้นและหยุดการทำงานของไทเมอร์ 2 เป็นการควบคุมทางซอฟต์แวร์

"0" หยุดการทำงานของไทเมอร์ 2 (STOP)

"1" เริ่มต้นการทำงานของไทเมอร์ 2 (STAT)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

C/T (Timer or Counter selector): ใช้เลือกลักษณะการทำงานของไทเมอร์/เคาน์เตอร์ 2

"0" เลือกให้ทำงานเป็นไทเมอร์ โดยใช้สัญญาณอินพุตจากสัญญาณนาฬิกาภายในไมโครคอนโทรลเลอร์

"1" เลือกให้ทำงานเป็นเคาน์เตอร์ โดยรับสัญญาณอินพุตทางขา T2(P1.0)

CP/RL2 (Capture/Reload flag): เป็นบิตแสดงสภานะการทำงานของไทเมอร์ 2

"0" มีการรีโหลดค่าของการนับแบบอัตโนมัติเกิดขึ้น เมื่อไทเมอร์ 2 เกิดโอเวอร์โฟลวหรือเกิดการเปลี่ยนระดับลอจิกแบบลบ (เปลี่ยนจาก "1" เป็น "0") ที่ขา T2EX ในกรณีที่บิต EXEN2 เป็น "1" แต่ถ้าหากบิต RCLKหรือ TCLK เป็น "1" จะยอมให้การรีโหลดแบบอัตโนมัติเกิดขึ้นเมื่อไทเมอร์ 2 เกิดโอเวอร์โฟลวเท่านั้น

"1" แสดงว่ามีการแคปเจอร์เกิดขึ้น เมื่อเกิดการเปลี่ยนระดับลอจิกแบบลบ (เปลี่ยนจาก "1" เป็น "0") ที่ขา T2EX ในกรณีที่บิต EXEN2 เป็น "1"

RCLK+TCLK	CP/$\overline{RL2}$	TR2	โหมด
0	0	1	16 บิตแบบตั้งค่าการนับอัตโนมัติหรือรีโหลด
0	1	1	แคปเจอร์หรือตรวจจับสัญญาณ
1	X	1	กำเนิดอัตราบอด
X	X	0	คิสเอเบิล

ตารางที่ 2.5 การเลือกโหมดทำงานของไทเมอร์/เคาน์เตอร์ 2

การทำงานในโหมดแคปเจอร์หรือตรวจจับสัญญาณ (capture)

การทำงานในโหมดนี้จะเกิดขึ้นได้อย่างสมบูรณ์เมื่อกำหนดให้บิต EXEN2 เป็น "1" หลังจากที่เลือกให้ไทเมอร์ 2 ทำงานในโหมดแคปเจอร์แล้ว เมื่อเลือกให้ทำงานในโหมดนี้ขาอินพุตที่ทำการตรวจจับคือขาอินพุต T2EX ซึ่งตรงกับขา P1.1 ที่ขา T2EX จะมีวงจรตรวจจับการเปลี่ยนแปลงของระดับลอจิกจาก "1" เป็น "0" รีจิสเตอร์TL2 และ TH2 จะทำการนับค่าเพิ่มขึ้นไปเรื่อย ๆ จนกว่าที่ขา T2EX จะตรวจจับการเปลี่ยนแปลงของระดับลอจิกจาก "1" เป็น "0" รีจิสเตอร์ TL2 และ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

TH2 จะทำการนับค่าเพิ่มขึ้นไปเรื่อย ๆ จนกว่าที่ขา T2EX จะตรวจจับการเปลี่ยนแปลงได้ เมื่อตรวจจับได้ค่าของรีจิสเตอร์ TL2 และ TH2 จะถูกส่งต่อไปยังรีจิสเตอร์ RCAP21 และ RCAP2H พร้อมกันนั้นยังทำการเซตบิต EXF2 ให้เป็น "1" เพื่อสร้างสัญญาณอินเตอรัปต์ให้เกิดขึ้น ในรูปที่ 6-5 แสดงการทำงานของไทเมอร์ 2 ในโหมดแคปเจอร์นี้

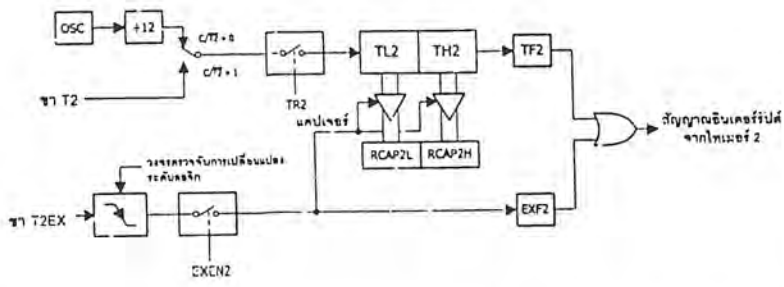
การทำงานในโหมดตั้งค่าการนับอัตโนมัติ

การทำงานในโหมดนี้ การกำหนดสถานะที่บิต EXEN2 จะทำให้เกิดการทำงานใน 2 ลักษณะคือ ถ้าบิต EXEN2 เป็น "0"เมื่อไทเมอร์ 2 ทำการนับค่าเกินรอบแล้ว จะเกิดการเซตบิต TF2 และจะทำการเรียกค่าการนับใหม่จากรีจิสเตอร์ RCAP2L และ RCAP2H โดยอัตโนมัติ การกำหนดค่าของรีจิสเตอร์ RCAP2L และ RACP2H สามารถกระทำได้ด้วยกระบวนการทางซอต์แวร์

ในกรณีที่บิต EXEN2 เป็น "1" หลังจากที่เลือกให้ไทเมอร์ 2 จะทำการนับเหมือนเดิม แต่เงื่อนไขของการตั้งค่าการนับใหม่โดยอัตโนมัติจากรีจิสเตอร์ RCAP2L และ RCAP2H จะให้การเปลี่ยนแปลงของระดับลอจิกจาก "1" เป็น "0" ที่ขา T2EX เป็นตัวกำหนดและทำการเซตบิต EXF2 เพื่อสร้างสัญญาณอินเตอร์รัปต์แทนบิต TF2 ในรูปที่ 6-6 แสดงไดอะแกรมการทำงานของไทเมอร์ 2 ในโหมดตั้งค่าการนับอัตโนมัติ

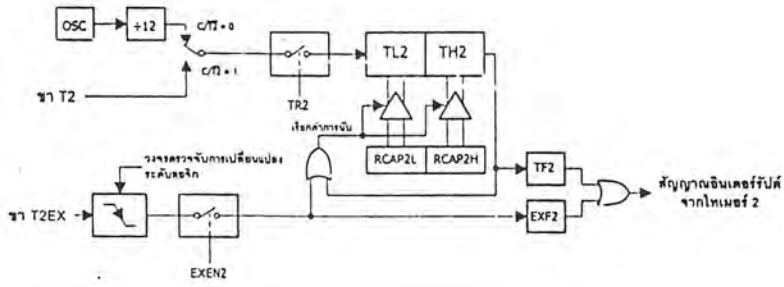
การทำงานในโหมดกำเนิดอัตราเร็วในการสื่อสารข้อมูลอนุกรมหรืออัตราบอด

การทำงานในโหมดนี้จะเกิดขึ้นเมื่อทำการเซตบิต RCLK และ/หรือ TCLK ในรีจิสเตอร์ T2CON ให้เป็น "1" เมื่อเข้าสู่การทำงานในโหมดนี้ บิต TF2 จะไม่เกิดการเซต และไม่มีการสร้างสัญญาณอินเตอร์รัปต์ แต่การอินเตอร์รัปต์ในไทเมอร์ 2 ก็ไม่ได้ถูกดิสเอเบิล เมื่ออยู่ในโหมดนี้ บิต EXEN2 ถูกเซต การเปลี่ยนแปลงระดับลอจิกจาก "1" เป็น "0" ที่ขา T2EX จะส่งผลให้บิต EXF2 เกิดการเซตแต่จะไม่มีการเรียนค่าจากรีจิสเตอร์RCAP2L และ RCAP2H เมื่อเป็นเช่นนี้ขา T2EX จึงสามารถให้เป็นขาอินพุตสำหรับการอินเตอร์รัปต์ได้เป็นกรณีพิเศษ

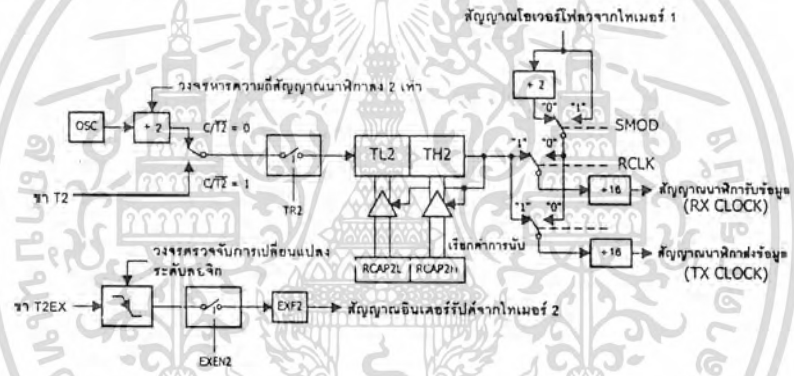


รูปที่ 2.12 ไดอะแกรมการทำงานในโหมดแคปเจอร์ของไทเมอร์ / เคาน์เตอร์ 2

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.13 ไดอะแกรมการทำงานในโหมดตั้งค่าการนับอัตโนมัติของไทเมอร์ / เคาน์เตอร์ 2



รูปที่ 2.14 ไดอะแกรมการทำงานในโหมดกำเนิดอัตราบอดในการสื่อสารข้อมูลผ่านพอร์ตอนุกรม ของไทเมอร์ / เคาน์เตอร์ 2

โหมดการทำงานของ ไทเมอร์/เคาน์เตอร์ 2	T2CON	
	การควบคุม จากภายใน	การควบคุม จากภายนอก
16 บิตแบบตั้งค่าการนับอัตโนมัติหรือรีโหลด	00H	08H
แคปเจอร์หรือตรวจจับสัญญาณ	01H	09H
กำเนิดอัตราบอด โดยอัตรารับและส่งเท่ากัน	34H	36H
รับข้อมูลเท่านั้น	24H	26H
ส่งข้อมูลเท่านั้น	14H	16H

ตารางที่ 2.6 ข้อมูลของรีจิสเตอร์ T2CON เพื่อกำหนดให้ไทเมอร์/เคาน์เตอร์ 2 ทำงานเป็นไทเมอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โหมดการทำงานของ ไทเมอร์/เคาน์เตอร์ 2	T2CON	
	การควบคุม จากภายใน	การควบคุม จากภายนอก
16 บิตแบบตั้งค่าการนับอัตโนมัติหรือรีโหลด	02H	0AH
แคปเจอร์หรือตรวจจับสัญญาณแบบ 16 บิต	03H	0BH

ตารางที่ 2.7 ข้อมูงของรีจิสเตอร์ T2CON เพื่อกำหนดให้ไทเมอร์/เคาน์เตอร์ 2 ทำงานเป็นเคาน์เตอร์

เมื่อไทเมอร์ 2 ทำงาน (บิต TR2 เป็น "1") จะถูกกำหนดให้ทำงานเป็นไทเมอร์ จึงไม่สามารถอ่านหรือเขียนค่ากับรีจิสเตอร์ TL2 และ TH2 ได้ ค่าของรีจิสเตอร์ไทเมอร์จะเพิ่มขึ้นทุก ๆ สเตต (สเตตเท่ากับ 2 คาบเวลาของสัญญาณนาฬิกา) ค่าการนับของรีจิสเตอร์นำมาใช้สร้างสัญญาณนาฬิกาของการสื่อสารข้อมูลผ่านพอร์ตอนุกรมภายในไมโครคอนโทรลเลอร์ ในรูปที่ 2.7 แสดงการทำงานของไทเมอร์ 2 ในการกำเนิดอัตราบอด

ข้อมูลที่ใช้ในการเลือกโหมดการทำงานของไทเมอร์/เคาน์เตอร์2

เนื่องจากมีตัวแปรอยู่หลายตัวที่ใช้ในการควบคุมและเลือกโหมดการทำงานของไทเมอร์/เคาน์เตอร์ 2 เช่นเดียวกับไทเมอร์/เคาน์เตอร์ 0 และ 1 โดยเฉพาะอย่างยิ่งโหมดของการทำงานที่แตกต่างกันมากและมีความซับซ้อนเพิ่มมากขึ้นเพื่อให้เกิดความสะดวกในการใช้งานจึงได้ทำการสรุปข้อมูลที่ใช้ในการกำหนดรูปแบบการทำงานของไทเมือร์/เคาน์เตอร์ 2 ไว้ในตารางที่ 2.6 ถึง 2.7

2.6 การสื่อสารข้อมูลแบบอะซิงโครนัส

การสื่อสารข้อมูลแบบอะซิงโครนัสคือการรับและส่งข้อมูลโดยไม่จำเป็นต้องมีสัญญาณนาฬิการ่วมด้วยแต่จะใช้การกำหนดค่าอัตราเร็วในการรับและส่งข้อมูลให้มีค่าเท่ากัน ซึ่งเรียนกอัตราเร็วนี้ว่าอัตราบอดหรือบอดเรต (baud rate)มีหน่วยเป็น บิตต่อวินาที (bit per second : bps)

รูปแบบของข้อมูลที่ใช้ในการรับส่งแบบอะซิงโครนัสประกอบด้วย 4 ส่วนด้วยกันคือ

1. บิตเริ่มต้น (start bit) มีขนาด 1 บิต

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2. บิตข้อมูลแบบอนุกรม มีขนาด 8 บิต
3. บิตตรวจสอบพาริตี้ (parity bit) มีขนาด 1 บิตหรือไม่มี
4. บิตปิดท้ายหรือบิตหยุด (stop bit) มีขนาด 1 บิต

รูปที่ 2.14 แสดงรูปแบบของข้อมูลอนุกรมแบบอะซิงโครนัสเมื่อไม่มีการส่งข้อมูลขา DATA จะมีสถานะลอจิก "1" เรียกสถานะนี้ว่า สถานะหยุดรอ (waiting sstage) การเริ่มต้นส่งข้อมูลจะเริ่มจากการให้ขา DATA มีลอจิก "0" ด้วยช่วงระยะเวลา 1 บิต เรียกบิตนี้ว่า บิตเริ่มต้น (start bit) จากนั้นบิตข้อมูลจะถูกส่งออกไป โดยเริ่มจากบิตที่มีนัยสำคัญต่ำสุดหรือบิต LSB ก่อน ซึ่งข้อมูลที่ต้องการส่งมีจำนวน 8 บิต จากนั้นตามด้วย บิตพาริตี้(parity bit) ซึ่งใช้ในการตรวจสอบความผิดพลาดที่เกิดขึ้นจากการส่งข้อมูล บิดสุดท้ายที่จะส่งคือ บิตปิดท้าย หรือ บิตหยุด (stop bit) โดยจะเป็นการทำให้ขา DATA มีสถานะลอจิก "1" อีกครั้งด้วยระยะเวลาอย่างน้อย 1 บิต , 1.5 บิต หรือ 2 บิต เพื่อเป็นการแสดงว่าสิ้นสุดข้อมูลแล้ว อัตราความเร็วในการรับและส่งข้อมมูลของการรับส่งข้อมูลแบบอะซิงโครนัสหรืออัตราบอดหรือบอดเรตที่ใช้สำหรับพอร์ตอนุกรม RS -232 มีด้วยกันหลายค่า ตั้งแต่ 110 ถึง 19.200 บิตต่อวินาที โดยมีค่าเพิ่มมากขึ้นตามเทคโนโลยีของคอมพิวเตอร์ เนื่องจากอัตราบอดคือค่าของจำนวนบิตที่สามารถส่งได้ใน 1 วินาที สมมติว่า ข้อมูลอนุกรมมีขนาด 8 บิต ไม่มีการตรวจสอบพาริตี้ มีบิตเริ่มต้น 1 บิต และบิตปิดท้าย 1 บิต ความยาวของข้อมูล 1 ไบต์ จะมีความยาวเท่ากับ 10 บิต ถ้าใช้บอดเรตในการส่งข้อมูลเท่ากับ 9,600 บิตต่อวินาที ก็จะสามารถรับส่งข้อมูบได้ด้วยความเร็ว 960 ไบต์ต่อวินาที

การตรวจสอบพาริตี้สามารถกำหนดให้เป็นแบบคี่ (odd) แบบคู่ (even) หรือไม่มีการตรวจสอบพาริตี้ก็ได้พาริตี้คี่หรือพาริตี้คู่แสดงถึงจำนวนลอจิก "1" ทั้งหมดภายในข้อมูลที่ส่งไป 1 ไบต์รวมบิตพาริตี้ว่ามีจำนวนเป็นเลขคู่หรือเลขคี่ ยกตัวอย่า ข้อมูลที่จะทำการส่งมีขนาด 8 บิต มีค่าเท่ากับ 99H หรือ 10011001B จะห็นว่าข้อมูลในไบต์นี้มีจำนวนลอจิก "1" จำนวน 4 ตัวซึ่งเป็นเลขคู่ ดังนั้นถ้ากำหนดค่าพาริตี้เป็นคู่ ค่าของบิตพาริตี้จะต้องมีลอจิกเป็น "0" แต่ถ้ากำหนดพาริตี้เป็นคี่ าของบิตพาริตี้จะต้องเป็น "1" เพื่อให้ข้อมูล 1 เพื่อให้ข้อมูล 1 ไบต์รวมทั้งบิตพาริตี้เป็นคี่

บิตพาริตี้ถูกสร้างขึ้นจากภาคส่งข้อมูลของ UART (Universal Asynchronous Receiver Transmitter : เป็นอุปกรณ์ที่ใช้ในการรับและส่งข้อมูลอนุกรม) ซึ่งทางภาครับต้องกำหนดการตรวจสอบพาริตี้ที่ตรงกันเอาไว้ว่าจะตรวจสอบพาริตี้คี่หรือคู่ จากนั้นภาครับของ UART จะทำการตรวจสอบค่าพาริตี้ที่เกิดขึ้นว่าเป็นคู่หรือเป็นคี่ โดยการนับ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จำนวนลอจิก "1" ทั้งหมดรวมทั้งบิตพาริตี้ด้วย ถ้ากำหนดพาริตี้ไว้เป็นคู่แต่อ่านค่าตัวเลขในการนับออกมาได้ตัวเลขเป็นคี่ ทางภาครับจะแสดงข้อผิดพลาดออกมาให้ผู้ใช้ทราบ กระบวนการดังกล่าวเป็นวิธีการตรวจสอบความผิดพลาดที่เกิดขึ้นในการรับส่งข้อมูลที่ง่ายที่สุด แต่มันสามารถตรวจสอบได้เมื่อมีบิตข้อมูลที่ทำการรับส่งผิดพลาดเพียงบิตเดียวเท่านั้น ถ้าข้อมูลที่ทำการส่งมีบิตที่ผิดพลาดมากกว่า 1 บิต การตรวจสอบด้วยวิธีนี้จะไม่ได้ผล สำหรับการตั้งพาริตี้เป็น NONE นั้นทั้งภาครับและส่งจะไม่มีการตรวจสอบพาริตี้

รีจิสเตอร์ที่เกี่ยวข้องกับการทำงานของพอร์ตอนุกรมใน MCS-51

ในการทำงานของพอร์ตอนุกรมในไมโครคอนโทรลเลอร์ MCS-51 มีรีจิสเตอร์ที่เกี่ยวข้องอยู่ 2 ตัวดังนี้

รีจิสเตอร์บัฟเฟอร์ของพอร์ตอนุกรมหรือ SBUF (Serial data buffer register)

มีแอดเดรสอยู่ที่ 99H ในพื้นที่ของรีจิสเตอร์ฟังก์ชั่นพิเศษหรือ SFR มีขนาด 8 บิต แบ่งเป็น 2 ส่วนคือรีจิสเตอร์บัฟเฟอร์สำหรับส่งข้อมูล (transmit buffer register) และรับข้อมูล (receive buffer register) เมื่อมีการเขียนข้อมูลมายังรีจิสเตอร์ SBUF ข้อมูลนั้นจะถูกส่งต่อไปยังบัฟเฟอร์สำหรับส่งข้อมูลเพื่อส่งออกจากไมโครคอนโทรลเลอร์ผ่านทางขา TxD หรือขา P3.1 ในกรีที่มีการอ่านข้อมูลจากรีจิสเตอร์ SBUF ข้อมูลจะถูกส่งผ่านไปยังรีจิสเตอร์บัฟเฟอร์สำหรับรับข้อมูลเพื่อส่งต่อไปยังไมโครคอนโทรลเลอร์ต่อไป สำหรับการรับข้อมูลอนุกรมจากภายนอกนั้นจะผ่านมาทางขา RxD หรือ P3.0 ของไมโครคอนโทรลเลอร์ MCS-51 แบบแฟลช

รีจิสเตอร์ควบคุมการทำงานของพอร์ตอนุกรมหรือ

SCON (Serial port Control Register)

เป็นรีจิสเตอร์ขนาด 8 บิต มีแอดเดรสอยู่ที่ 98H ในพื้นที่ของรีจิสเตอร์ SFR สามารถเข้าถึงได้ในระดับบิต มีรายละเอียดการทำงานดังนี้

บิต 7	บิต 6	บิต 5	บิต 4	บิต 3	บิต 2	บิต 1	บิต 0
SM0	SM1	SM2	REN	TB8	RB8	TI	RI

SM0-SM1 (Serial port mode bit 0-1): ใช้ในการเลือกโหมดการทำงานของพอร์ตอนุกรมภายในไมโครคอนโทรเลอร์ MCS-51 ดังมีรายละเอียดต่อไปนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

SM2: ใช้ในการเอ็นเอเบิลการสื่อสารในแบบมัลติโปรเซสเซอร์ (multiprocessor) ในการทำงานของโหมด 2 และ3 ของพอร์ตอนุกรมในไมโครคอนโทรลเลอร์ MCS-51 ถ้าบิตนี้เป็น "1" บิต จะไม่แอกตีฟถ้าบิตที่ 9 ที่รับเข้ามาเป็น "0" (ข้อมูลบิตที่ 9 เก็บไว้ที่บิตRB8) ในการทำงานโหมด 1 ถ้าบิต R1 นี้เซต บิต จะไม่แอกตีฟถ้ายังไม่ได้รับบิตหยุด ส่วนในโหมด 0 บิตนี้ไม่มีการใช้งาน

REN (Enable serial reception): ใช้ในการเอ็นเอเบิลการรับข้อมูลของพอร์ตอนุกรม ทำการเซตและเคลียร์ด้วยกระบวนการทางซอฟต์แวร์ ถ้าต้องการให้มีการรับข้อมูลต้องเซตบิตนี้ให้เป็น "1"

TB8: ใช้สำหรับเก็บข้อมูลบิตที่ 9 ที่ต้องการส่งออกไปในการทำงานโหมด 2 และ 3 ของพอร์ตอนุกรมในไมโครคอนโทรลเลอร์ MCS-51 เซตและเคลียร์ด้วยกระบสนการทางซอฟแวร์

RB8: ใช้สำหรับรับข้อมูลบิตที่ 9 ที่เข้ามาในการทำงานโหมด 2 และ 3 ของพอร์ตอนุกรมในไมโคคอนโทรลเลอร์MCS-51 แต่ถ้าหากพอร์ตอนุกรมทำงานอยู่ในโหมด 1 และ บิต SM1 เป็น "0" ข้อมูลที่บิต RB8 คือข้อมูลของบิตหยุด (STOP bit) สำหรับในการทำงานโหมด 0 บิตนี้จะไม่ใช้งาน บิต RB8 นี้สามารถเซตและเคลียร์ด้วยกระบวนการทางซอฟต์แวร์

TI (Transmit Interrupt flag): ใช้ในการแสดงการเกิดอินเตอร์รัปต์เมื่อมีการส่งข้อมูลออกจากพอร์ตอนุกรมของไมโครคอนโทรลเลอร์ MCS-51 สามารถเซตได้ด้วยกระบวนการทางฮาร์ดแวร์ เมื่อทำการส่งข้อมูลบิตที่ 8 ไปเรียบร้อยแล้วในการทำงานโหมด 0 ส่วนในการทำงานโหมดอื่น บิตนี้จะเซตเมื่อมีการเริ่มต้นส่งบิตหยุดออกไป การเคลียร์บิตนี้ต้องใช้กระบวนการทางซอฟต์แวร์เท่านั้น

RI (Receive Interrupt flag): ใช้แสดงการเกิดอินเตอร์รัปต์เมื่อมีการรับข้อมูลเข้าสู่พอร์ตอนุกรม สามารถเซตได้ด้วยกระบวนการทางฮาร์ดแวร์ เมื่อทำการรับข้อมูลบิตที่ 8 เรียบ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ร้อยแล้วในการทำงานโหมด 0 ส่วนในการทำงานโหมดอื่น บิตนี้จะเซตเมื่อมีสามารถรับบิตหยุดของข้อมูลอนุกรมไปได้ครึ่งทางแล้ว ยกเว้นในกรณีที่บิต SM2 มีการเซต บิตนี้จะเซตได้ก็ต่อเมื่อการรับบิตอยุดหรือบิตที่ 9 เกิดขึ้นอย่างสมบูรณ์แล้วการเคลียร์บิตนี้ต้องใช้กระบวนการทางซอฟต์แวร์เท่านั้น

โหมดการทำงานของพอร์ตอนุกรมใน MCS-51

พอร์ตอนุกรมในไมโครคอนโทรลเลอร์ MCS-51 สามารถเลือกการทำงานได้ถึง 4 โหมดคือ

1. โหมด 0 เป็นการกำหนดให้พอร์ตอนุกรมทำงานในลักษณะชิฟต์รีจิสเตอร์
2. โหมด 1 เป็นการกำหนดให้เป็น UARTขนาด 8 บิต สามารถเลือกอัตราบอดได้
3. โหมด 2 เป็นการกำหนดให้เป็น UART ขนาด 9 บิต โดยมีอัตราบอดคงที่
4. โหมด 3 เป็นการกำหนดให้เป็น UART ขนาด 9 บิตสามารถเลือกอัตราบอดได้

การเลือกโหมดทำได้ด้วยการกำหนดข้อมูลให้แก่บิต SM0 และ SM1 ในรีจิสเตอร์ SCON

การทำงานในโหมด 0 ของวงจรพอร์ตอนุกรม

มีไดอะแกรมการทำงานและไดอะแกรมเวลาแสดงในรูปที่ 2.15 ข้อมูลอนุกรมจะผ่านเข้าและออกทางขา RxD ส่วนขา TxD ทำหน้าที่เป็นสัญญาณนาฬิกาของการเลื่อนข้อมูล (shift clock) ในโหมดนี้มีจำนวนข้อมูล 8 บิต โดยทำการรับและส่งข้อมูนในบิต LSB อัตราในการรับส่งข้อมูลหรืออัตราบอดถูกกำหนดไว้คงที่ที่ 1/12 ของความถี่สัญญาณนาฬิกา

เริ่มต้นการส่งข้อมูลด้วยการเขียนข้อมูลที่ต้องการส่งมายังรีจิสเตอร์ SBUF สัญญาณเขียนข้อมูล SBUFแอกตีฟเป็น "1" ที่สเตต 6 เฟส2 (S6P2) ของแมชีนไซเกิล ส่งมายังวงจรควบคุมการส่ง (TX control) ทำให้วงจรควบคุมเริ่มต้นส่งข้อมูล สัญญาณ SEND จะแอกตีฟเป็น "1" ตลอดการส่งข้อมูล

ข้อมูลจากรีจิสเตอร์ SBUF จะถูกเลื่อนออกที่ขา P.30 หรือขา RxD ครั้งละบิต ตามจังหวะของสัญญาณนาฬิกาที่ส่งออกมาทางขา P3.1 หรือ TxD โดยสัญญาณนาฬิกาของการเลื่อนข้อมูลจะมีขอบขาลงของสัญญาณที่สเตต 3 เฟส และมีขอบขาขึ้นของสัญญาณที่สเตต 6 เฟส 1 ของแต่ละแมชีนไซเกิลในกระบวนการส่งข้อมูล จนกระทั่งเมื่อส่งข้อมูลครบ 8 บิตแล้ว บิต TI ในรีจิสเตอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

SCONจะเกิดการเซตเป็นการแจ้งให้ทราบว่าส่งข้อมูลครบแล้วหากการอินเตอร์รัปต์จากพอร์ตอนุกรมได้รับการเอ็นเอเบิลไว้ก็จะเกิดการอินเตอร์รัปต์ขึ้นในระบบ เมื่อเสร็จสิ้นกระบวนการรับข้อมูล สัญญาณ SEND จะกลายเป็น "0" จนกว่าจะเริ่มต้นกระบวนการรับข้อมูลใหม่

ในกระบวนการรับข้อมูล เริ่มต้นด้วยการเซต REN ให้เป็น "1" และเคลียร์บิต RIในรีจิสเตอร์ SCONก่อนที่สเตต 6 เฟส 2 ของแซีนไซเกิลถัดไป วงจรควบคุมการรับ (RX control) จะทำการเขียนข้อมูล 11111110 ไปยังชิฟต์รีจิสเตอร์สำหรับรับข้อมูลและทำการแอกตีฟสัญญาณ RECEIVE ให้เป็น "1"ในสัญญาณนาฬิกาลูกถัดไป

เมื่อสัญญาณ RECEIVE แอกตีฟก็จะเกิดการส่งสัญญาณนาฬิกาของการเลื่อนข้อมูล (Shift clock) ขึ้นผ่านทางขา P3.1หรือ TxD เพื่อทำการกำหนดจังหวะการรับข้อมูลครั้งละบิต โดยสัญญาณนาฬิการนี้จะเกิดขึ้นในช่วงสเตต 3 เฟส 1 ถึงสเตต 6 เฟส 1 ของแต่ละแมชีนไซเกิล การรับข้อมูลเข้ามาทางขา P3.0หรือ RxD จะเกิดขึ้นที่สเตต 5 เฟส 2 ในแมชีนไซเกิลเดียวกับสัญญาณนาฬิกาของการเลื่อนข้อมูล จนกระทั่งรับข้อมูลครบทั้ง 8 บิต บิต จะได้รับการเซตเพื่อแจ้งการเสร็จสิ้นกระบวนการรับข้อมูล หากการอินเตอร์รัปต์จากพอร์ตอนุกรมได้รับการเอ็นเอเบิลไว้ก็จะเกิดการอินเตอร์รัปต์ขึ้นในระบบ เมื่อเสร็จสิ้นกระบวนการรับข้อมูล สัญญาณ RECEIVE จะกลายเป็น "0" จนกว่าจะเริ่มต้นกระบวนการรับข้อมูลใหม่

การทำงานในโหมดนี้ของพอร์ตอนุกรมในไมโครคอนโทรเลอร์ MCS-51 จะให้ในการเชื่อมต่อกับไอซีรีจิสเตอร์ภายนอกเพื่อทำการขยายจำนวนพอร์ตอินพุตหรือเอาต์พุต แต่ไม่เป็นที่นิยมใช้งานมากนัก เนื่องจากในไมโครคอมโทรลเลอร์ MCS-51 เองมีพอร์ตอยู่ค่อนข้างมาก และติดต่อกับพอร์ตเหล่านั้นได้ง่ายและเร็วกว่ามาก

การทำงานในโหมด 1 ของวงจรพอร์ตอนุกรม

มีไดอะแกรมแสดงในรูปที่ 2.16 ในโหมดนี้ใช้ในการรับส่งข้อมูลรวม 10 บิต โดยส่งข้อมูลออกทางขา P3.1หรือ TxD และรับข้อมูลเข้าทางขา P3.0 หรือ RxDข้อมูลทั้ง 10 บิตประกอบด้วยบิตเริ่มต้น (มีค่าเป็น "0") 1 บิต บิตข้อมูล 8 บิต โดยรับหรือส่งข้อมูลในบิต LSB ก่อน และบิตหยุดหรือบิตปิดท้าย (มีค่าเป็น "1") ในการรับข้อมูลบิตหยุดจะถูกเก็บไว้ในบิต RB8 ในรีจิสเตอร์ SCON อัตราบอดในโหมดนี้ได้รับการกำหนดโดยอัตราการเกิดโอเวอร์โฟลว์ของไทเมอร์ 1 ใน AT89C51 ส่วนในไมโครคอนโทรลเลอร์เบอร์ AT89C52 และในอนุกรมAT89Sxxสามารถเลือกใช้อัตราการเกิดโอเวอร์โฟลวของไทเมอร์ 1 หรือไทเมอร์ 2 ในการกำหนดอัตราบอดได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

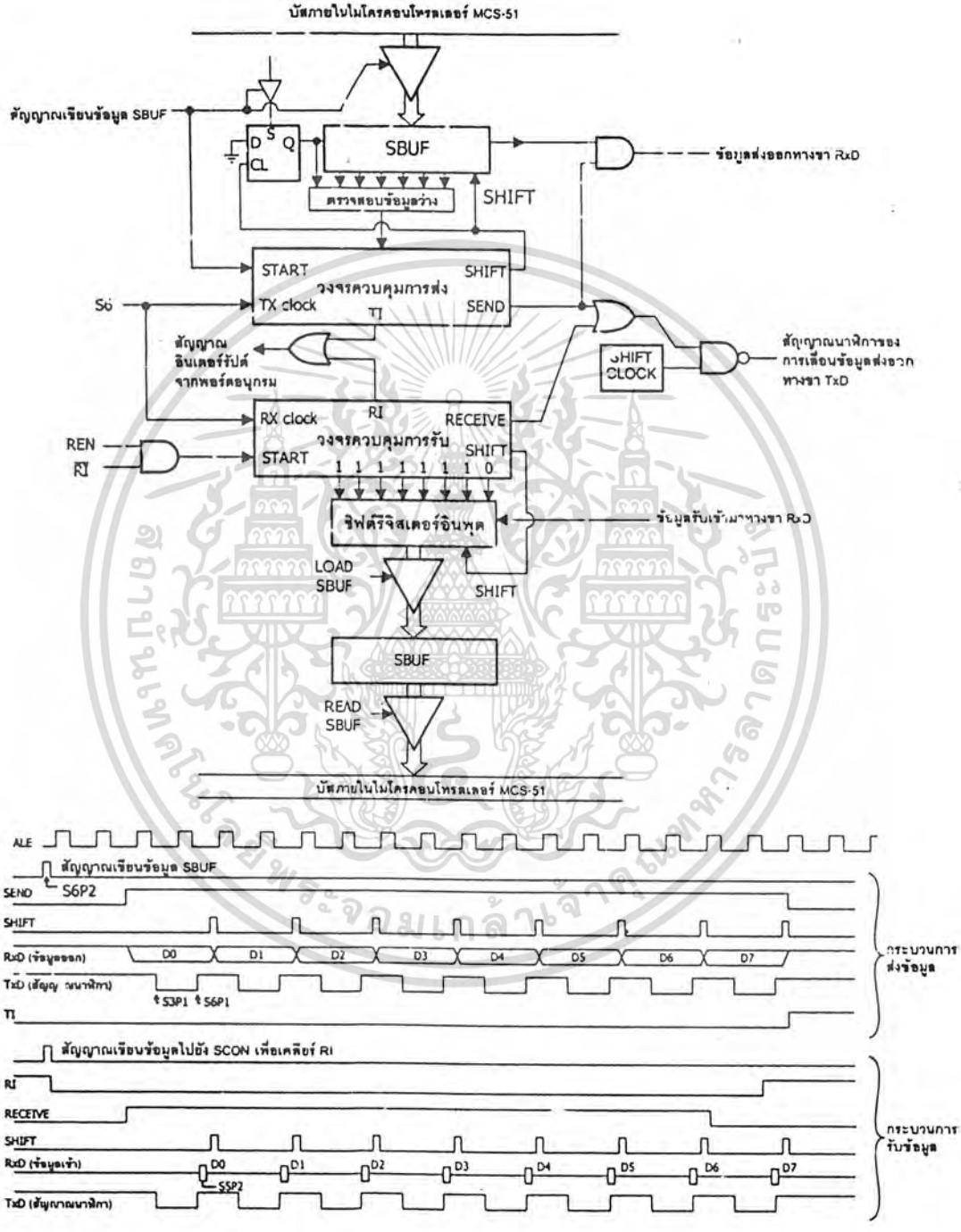
กระบวนการส่งข้อมูลเริ่มต้นด้วยการแอกตีฟสัญญาณเขียนข้อมูลมายังรีจิสเตอร์ SBUF ส่งมายังวงจรควบคุมการส่ง(TX control) จากนั้นวงจนควบคุมจะทำการแอกตีฟสัญญาณ SEND ที่สเตต 1 เฟส 1 ขอแมชีนไซเกิลต่อมา โดยสัญญาณ SEND จะเป็น "0" ตลอดการส่งข้อมูลเมื่อสัญญาณ SEND แอกตีฟจะทำการส่งบิตเริ่มต้นก่อนเป็นบิตแรก โดยมีคาบเวลาของบิตเริ่มต้นเท่ากับ 1 แมชีนไซเกิล จากนั้นตามด้วยการส่งยิตข้อมูล 8 บิต เรียงลำดับจากบิต LSB โดยข้อมูลที่ทำการส่งถูกเรียกออกมาจากรีจิสเตอร์บัฟเฟอร์สำหรับการส่งข้อมูล ในทุก ๆ บิตข้อมูลที่ทำการส่งออกไป จะเกิดสัญญาณพัลส์ SHIFT ขึ้น เพื่อให้เรียกข้อมูลในแต่ละบิตจากรีจิสเตอร์บัฟเฟลอร์การกำหนดจังหวะการส่งข้อมูลใช้สัญญาณนาฬิกาการส่ง(TX clock) เป็นตัวกำหนด โดยสัญญาณ TCLK นาฬิกานี้ได้มาจากการหารสัญญาณ จากไทเมอร์ 1 ด้วย 16 หลังจากการส่งบิตข้อมูลก็จะทำการส่งบิตหยุดหรือบิตปิดท้าย 1 บิต ดังนั้นการส่งข้อมูลจะใช้สัญญาณนาฬิกาทั้งหมด 10 ลูก เมื่อทำการส่งข้อมูลครบเรียบร้อยแล้ว จะทำการเซตบิต ในรีจิสเตอร์ SCON หากการอินเอตร์รัปต์จากพอร์ตอนุกรมได้รับการเอ็นเอเบิลไว้ ก็จะเกิดการอินเตอร์รัปต์ขึ้นในระบบ หลังจากที่ทำการบริการอินเตอร์รัปต์หรือส่งข้อมูลเรียบร้อยแล้ว ต้องทำการเคลียร์บิต TI ก่อนเป็นอันดับแรก เพื่อให้การรับส่งข้อมูลทางพอร์ตอนุกรมดำเนินต่อไปได้

ด้านการรับข้อมูล จะทำการตรวจจับการเปลี่ยนแปลงระดับลอจิกจาก "1" เป็น "0" ที่ขา RxD โดยซ้อัตรา การสุ่มเท่ากับ 1/16 เท่าของอัตราบอด เมื่อตรวจจับพบไทเมอร์/เคาน์เตอร์ที่ใช้ในการกำหนดอัตราบอดจะรีเซตและทำการเขียนข้อมูล 1FFH ไปยังชิฟต์รีจิสเตอร์ข้อมูลจะเริ่มเดินทางเข้าสู่พอร์ตอนุกรมของไมโครคอนโทรลเลอร์ผ่านทางขา RxDในการตีความว่าบิตที่เข้ามาเป็น "0" หรือ "1" จะใช้ผลการสุ่มข้างมาก โดยบิตของข้อมูลที่เข้ามาได้รับการแบ่งออกเป็น 16 สเตต การสุ่มข้อมูลจะทำการสุ่มสเตตที่ 7,8 และ 9 หาก 2 ใน 3 ของการสุ่มพบว่าข้อมูลเป็นลอจิกใด จะตีความข้อมูลในบิตนั้นเป็นตามเสียงข้างมาก ยกตัวอย่าง สุ่มพบลอจิก "1" 2 ใน 3 ครั้ง จะตีความว่าบิตของข้อมูลที่รับได้นั้นเป็น "1"

ลำดับของการรับข้อมูลมีลักษณะเดียวกับการส่งข้อมูลคือ เริ่มด้วยบิตเริ่มต้นก่อน ตามด้วยบิตข้อมูลและบิตปิดท้าย ในทุก ๆ การรับข้อมูลได้ 1 บิตจะมีพัลส์ SHIFT เกิดขึ้นเพื่อทำการเลื่อนข้อมูลเข้าสู่รีจิสเตอร์บัฟเฟอร์การรับข้อมูล การกำหนดจังหวะการรับข้อมูลใช้สัญญาณนาฬิกาการรับข้อมูล (RX clock) หลังจากสัญญาณนาฬิกาลูกสุดท้าย อันหมายถึงสามารถรับข้อมูลได้ครบแล้ว วงจรควบคุมการรับข้อมูลจะทำการส่งข้อมูลจากรีจิสเตอร์บัฟเฟอร์ไปยังรีจิสเตอร์ SBUF และบิด RB8 ในรีจิสเตอร์ SCON โดยข้อมูลในบิต RB8 ก็คือข้อมุลของบิต หยุดนั่นเอง พร้อมกันนั้นยังทำการเซตบิต ในรีจิสเตอร์ ด้วย หากการอินเตอร์รัปต์จากพอร์ตอนุกรมได้รับการ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การทำงานในโหมดนี้ได้รับความนิยมสูงสุดเนื่องจากมีกระบวนการที่ไม่ซับซ้อนและสามารถทำการรับส่งข้อมูลกับคอมพิวเตอร์ได้อย่างมีประสิทธิภาพ



รูปที่ 2.15 ไดอะแกรมการทำงานในโหมด 0 ของพอร์ตอนุกรมภายในไมโครคอนโทรลเลอร์ MCS-51

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การทำงานในโหมด 2 และ 3 ของวงจรพอร์ตอนุกรม

ในทั้งสองโหมดนี้จะใช้รูปแบบข้อมูลรวม 11 บิต ประกอบด้วยบิตเริ่มต้น มีค่าเป็น "0" จำนวน 1 บิต บิตข้อมูล 8 บิตโดยทำการรับและส่งบิต LSB ก่อน บิตข้อมูลบิตที่ 9 และบิตปิดท้าย มีค่าเป็น "1" จำนวน 1 บิต ในการส่งข้อมูล ข้อมูลบิตที่ 9 จะได้รับการเก็บไว้ในบิต TB8 ในรีจิสเตอร์ SCONและในการรับข้อมูล ข้อมูลบิตที่ 9 จะนำไปเก็บไว้ในบิต TB8 ในรีจิสเตอร์ SCON สำหรับอัตราบอดในโหมด 2 จะคงที่โดยเลือกได้ 2 ค่าคือ 1/32 หรือ 1/64 ของความถี่สัญญาณนาฬิกา สำหรับในโหมด 3 อัตราบอดสามารถปรับได้เหมือนกับในโหมด 1

ในรูปที่ 2.19 และ 2.20 เป็นไดอะแกรมการทำงานและไดออะแกรมของการทำงานในโหมด 2 และ 3 ของพอร์ตอนุกรม การทำงานโดยรวมจะคล้ายกับการทำงานในโหมด 1 ส่วนที่แตกต่างกันคือจำนวนบิตของข้อมูลที่ในโหมด

อัตราบอดของพอร์ตอนุกรมในไมโครคอนโทรลเลอร์ MCS-51

โหมด 0

อัตราบอดของโหมดนี้มีค่าคงที่ โดยสามารถคำนวณได้จากสูตร

อัตราบอดในโหมด 0 = ความถี่ของสัญญาณนาฬิกา/12 หน่วยเป็นบิตต่อวินาที

โหมด 1 และ 3

เนื่องจากทั้งสองโหมดนี้สามารถเลือกแหล่งกำเนิดอัตราบอดได้ 2 แหล่งคือ จากอัตราโอเวอร์โฟลวของไทเมอร์ 1 และ 2 สำหรับอัตราบอดเมื่อใช้การโอเวอร์โฟลวของไทเมอร์ 1 จะต้องใช้ค่าของบิต SMOD ในรีจิสเตอร์ PCON (จะกล่าวถึงรายละเอียดของรีจิสเตอร์ตัวนี้ในบทที่ว่าด้วยการทำงานในโหมดประหยัดพลังงาน) มาพิจารราประกอบด้วย สามารถคำนวณค่าอัตราบอดได้จาก

อัตราบอด = (2 ค่าของบิต SMOD /32) x อัตราโอเวอร์โฟลวของไทเมอร์ 1

ถ้าหากในไทเมอร์ 1 ไม่ได้เอ็นเอเบิลการอินเตอร์รัปต์ไว้ สามารถคำนวณค่าอัตราบอดได้จาก

อัตราบอด = (2 ค่าในรีจิสเตอร์ SMOD /32) x (ความถี่สัญญาณนาฬิกา/{12x{256-(THI)}}}

ในตารางที่ 2.8 แสดงการกำหนดอัตราบอดโดยใช้ไทเมอร์ 1

ในกรณีที่ใช้ไทเมอร์ 2 ในการกำหนดอัตราบอดโดยกำหนดให้ไทเมอร 2 ทำงานในโหมดกำเนิดอัตราบอด (baud rate generator) สามารถคำนวณหาค่าอัตราบอดได้จาก

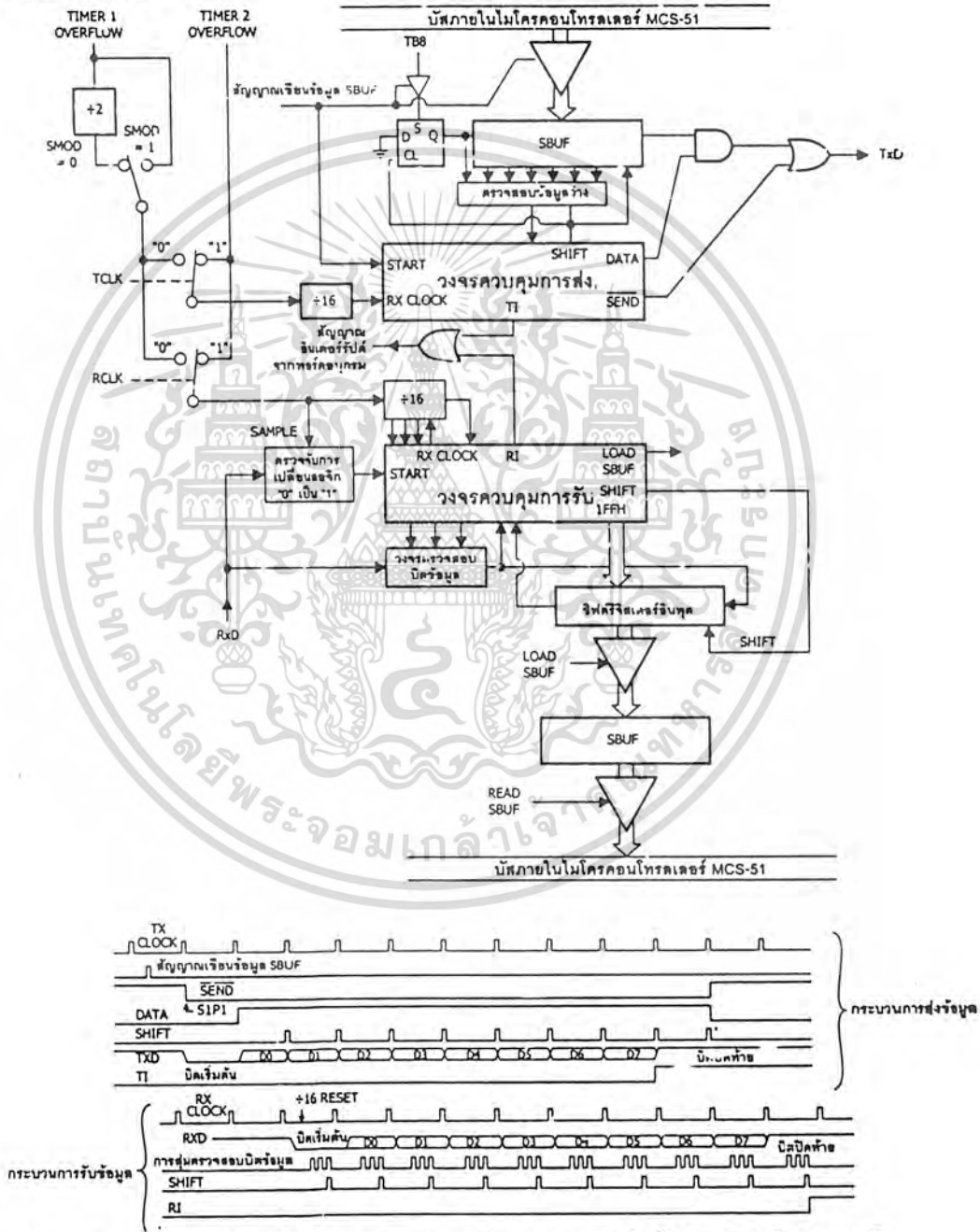
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

อัตราบอด = อัตราโอเวอร์โฟลวของไทเมอร์ 2/16 หน่วยเป็น บิตต่อวินาที

ถ้าหากกำหนดให้ไทเมอร์ 2 ทำงานในโหมดปกติ สามารถคำนวณหาค่าอัตราบอดได้จาก

อัตราบอด = ความถี่ของสัญญาณนาฬิกา (32 x (65536-(RCAP2H,RCAP2L)))

โดยที่ (RCAP2H, RCAP2L เป็นค่าของรีจิสเตอร์ RCAP2H และ RCAP2L มีขนาด 16 บิตไม่คิดเครื่องหมาย



รูปที่ 2.16 ไดอะแกรมการทำงานในโหมด 1 ของพอร์ตอนุกรมภายในไมโครคอนโทรลเลอร์ MCS-51 โหมด 2

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

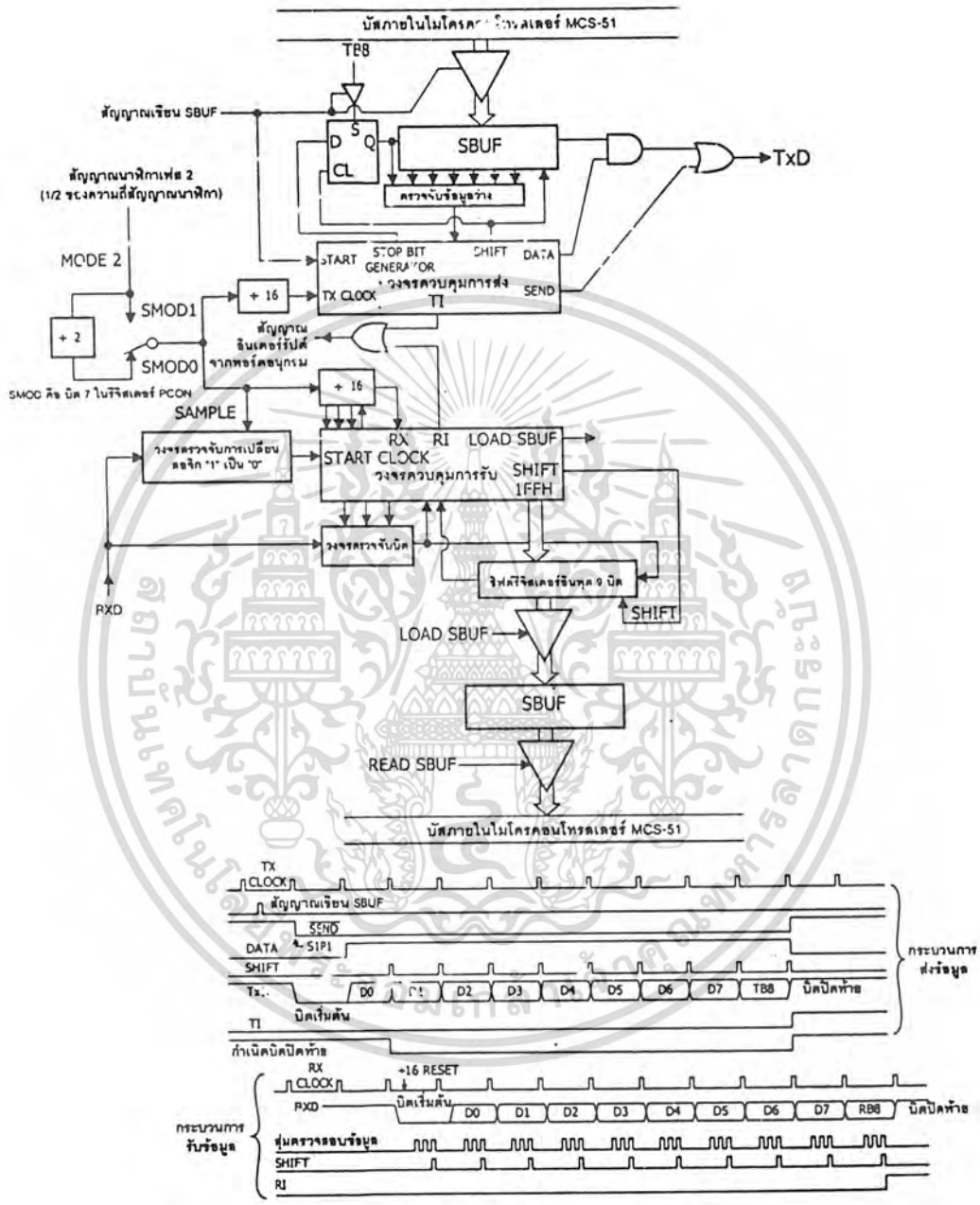
ในโหมดนี้อัตราบอดจะขึ้นอยู่กับค่าของบิต SMOD ในรีจิสเตอร์ PCON ถ้า PCON เป็น "0" อัตราบอดจะเท่ากับ 1/64 ของความถี่สัญญาณนาฬิกาในกรณีที่ SMODเป็น "1" อัตราบอดจะเท่ากับ 1/32 ของความถี่สัญญาณนาฬิกา สามารถแสดงเป็นสูตรคำนวณทางคณิศาสตร์ได้ดังนี้

อัตราบอด = (2 ค่าของบิต SMOD/64) x ความถี่สัญญาณนาฬิกา

การกำหนดค่าของไทเมอร์เพื่อเลือกอัตราบอด

ในการใช้งานพอร์ตอนุกรมของไมโครคอนโทรลเลอร์ MCS-51 สิ่งที่ต้องให้ความสนใจมากที่สุดประการหนึ่งคือ อัตราการถ่ายทอดข้อมูล หรือ อัตราบอด ซึ่งการกำหนดอัตราบอดนั้นจะขึ้นอยู่กับค่าความถี่ของสัญญาณนาฬิการเป็นหลัก สำหรับโหมดการทำงานของพอร์ตอนุกรมที่สามารถเลือกอัตราบอดได้อย่างอิสระคือในโหมด 1 และ 3 โดยกำหนดได้จากอัตราการเกิดโอเวอร์โฟลวของไทเมอร์ 1 ถ้าหากไทเมอร์ 1 มีการเกิดโอเวอร์โฟลวในอัตราที่สูงมากเท่าใด อัตราบอดก็จะมีค่าสูงมากขึ้นตาม นั่นหมายความว่า อัตราในการถ่ายทอดข้อมูลจะสูงมาก สามารถถ่ายทอดข้อมูลได้อย่างรวดเร็ว

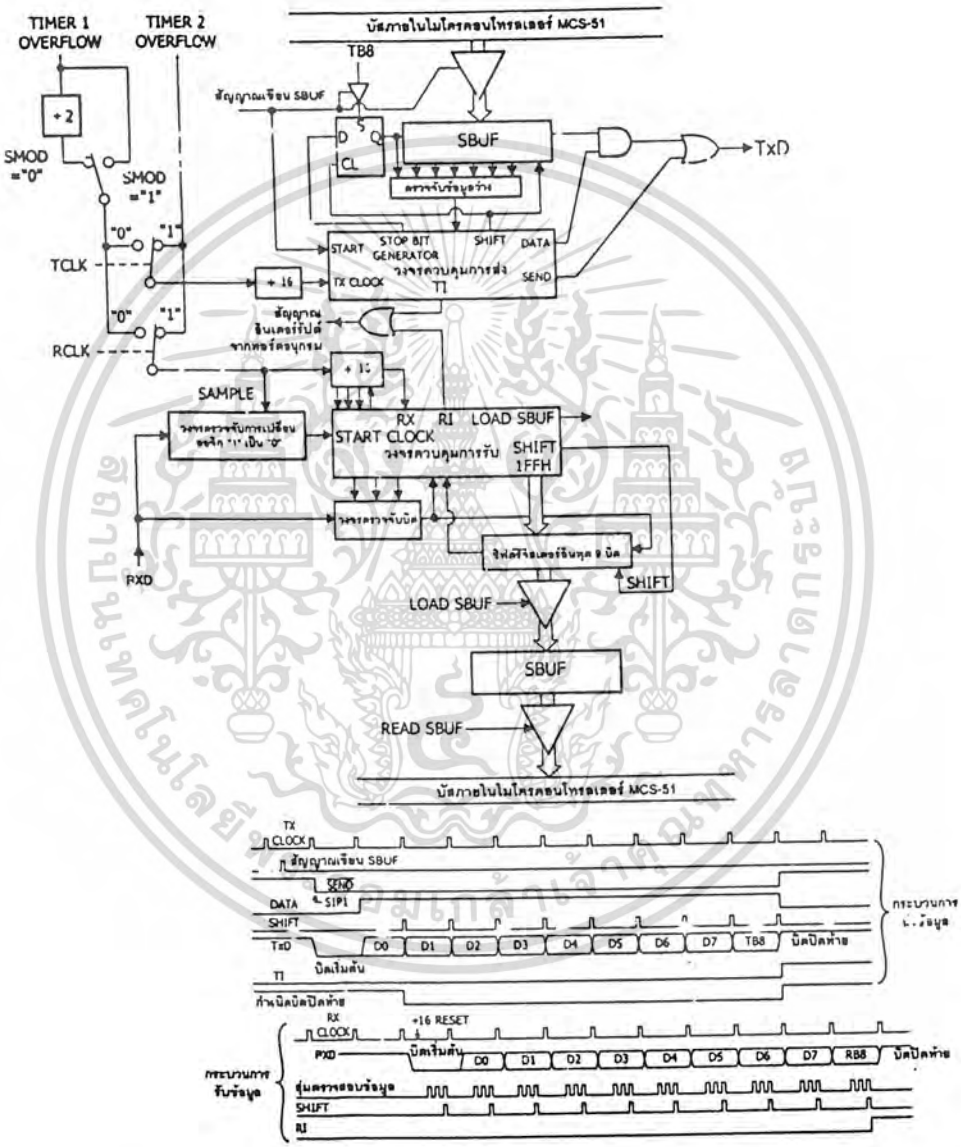
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.17 ไดอะแกรมการทำงานในโหมด 2 ของพอร์ตอนุกรมภายในไมโครคอนโทรลเลอร์ MCS-51

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ในการใช้ไทเมอร์ 1 เพื่อกำหนดอัตราบอดในโหมด 1 และ 3 ของพอร์ตอนุกรมจะต้องกำหนดให้ไทเมอร์ 1 ทำงานในโหมด 2 หรือ โหมด 8 บิตแบบตั้งค่าการนับอัตโนมัติ และการกำหนดค่ารีโหลดให้แก่รีจิสเตอร์ TH1 จึงเป็นตัวแปรหลักที่ใช้ในการกำหนดอัตราบอดให้แก่พอรีตอนุกรมของไมโครคอนดทรลเลอร์MCS-51



รูปที่ 2.18 ไดอะแกรมการทำงานในโหมด 3 ของพอร์ตอนุกรมภายในไมโครคอนโทรลเลอร์ MCS-51

เริ่มต้นด้วยการเคลียร์บิต SMOD ซึ่งเป็นบิต 7 ของรีจิสเตอร์ PCON ให้เป็น "0" ค่าของการรีโหลดให้แก่ TH1 สามารถคำนวณได้จาก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

TH1 = 256 (ค่าความถี่ของคริสตอล /384/อัตราบอด)

แต่ถ้าบิต SMOD เกิดการเซต จะเป็นการเอ็นเอเบิลการทวีคูณของอัตราบอด ดังนั้นการกำหนดค่าให้แก่ TH1 จึงต้องคำนวณจาก

TH1 = (ค่าความถี่ของกริสตอล/192)/อัตราบอด)

ยกตัวอย่าง ถ้าหากไมโครคอนโทรลเลอร์ AT89C51 ใช้คริสตอล 11.0592 MHz ต้องการกำหนดอัตราบอดของพอร์ตอนุกรมของไมโครคอนโทรลเลอร์ไว้ที่ 19,200 บิตต่อวินาที ในกรณีที่ไม่เอ็นเอเบิลการทวีคูณของอัตราบอด ค่ารีโหลดของไมโครคอนโทรเลอร์จะเท่ากับ

TH1 = 256 – (ความถี่ของกริสตอล/384/อัตราบอด)

= 256 –(11059200/384/19200)

= 256-(28800/19200)

= 256-1.5 =254.5

เนื่องจากผลลัพธ์ที่ได้เป็นค่าที่ไม่ใช่จำนวนเต็ม ถ้าหากกำหนดค่าของ TH1 เป็น 254 เมื่อทำการแทนค่าเพื่อคำนวณหาค่าอัตราบอด จะได้อัตราบอดเท่ากับ 14,400 บิตต่อวินาที และถ้าหากกำหนดค่าของ TH1 เป็น 255 อัตราบอดจะมีค่าเท่ากับ 28,800 บิตต่อวินาทีดังนั้นจะเห็นได้ค่าของ TH1 ไม่เป็นจำนวนเต็มจะไม่สามารถทำให้เกิดอัตราบอดตามที่ต้องการได้

ทางแก้ไขคือ ให้ทำการเอ็นเอเบิลการทวีคูณอัตราบอด โดยการเซตบิต SMOD ในรีจิสเตอร์ PCONได้เป็น "1" จากนั้นแทนค่า TH1 ลงในสมการหาค่า เมื่อมีการเซตบิต SMOD ได้ผลดังนี้

TH1 = 256- ((ค่าความถี่ของคริสตอล/192/อัตราบอด)

= 256-((11059200/192/19200)

= 256-(57600/19200)

= 256-3 = 253

นำค่าของ TH1 ที่ได้ทำการแทนค่าคำนวณหาค่าอัตราบอดจะได้เท่ากับ 19,200 บิตต่อวินาที

สามารถสรุปขั้นตอนในการเลือกอัตราบอดโดยการกำหนดค่าของไทเมอร์ 1 ได้ดังนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

อัตราบอด (บิตต่อวินาที : bps)	ความถี่สัญญาณนาฬิกา	SMOD	ไทเมอร์ 1		
			C/T	โหมด	ค่ารีโหลด
โหมด 0 : สูงสุด 1 MHz	12 MHz	X	X	X	X
โหมด 2 : สูงสุด 375K	12 MHz	1	X	X	X
โหมด 1,3 : 62.5K	12 MHz	1	0	2	FFH
19.2K (19,200)	11.0592 MHz	1	0	2	FDH
9.6K (9,600)	11.0592 MHz	0	0	2	FDH
4.8K (4,800)	11.0592 MHz	0	0	2	FAH
2.4K (2,400)	11.0592 MHz	0	0	2	F4H
1.2K (1,200)	11.0592 MHz	0	0	2	E8H
137.5	11.0592 MHz	0	0	2	1DH
110	6 MHz	0	0	2	72H
110	12 MHz	0	0	1	FEEBH

ตารางที่ 2.8 การเลือกอัตราบอดของวงจรพอร์ตอนุกรมภายในไมโครคอนโทรลเลอร์ MCS-51

1. กำหนดให้พอร์ตอนกรมของไมโครคอนโทรลเลอร์ MCS-51 ทำงานในโหมด 1 หรือ 3
2. กำหนดให้ไทเมอร์ 1 ทำงานในโหมด 2 หรือโหมด 8 บิตตั้งค่าอัตโนมัติ
3. กำหนดข้อมูลให้แก่ TH1 เท่ากับ 253 เพื่อให้สามารถกำเนิดอัตราบอดได้ 19,200 บิตตอวินาทีตามที่ต้องการ
4. ทำการเซตบิต SMOD ซึ่งเป็นบิต 7 ของรีจิสเตอร์ PCON เพื่อเอ็นเอเบิลการทวีคูณอัตราบอด

การเขียนหรือส่งข้อมูลออกจากพอร์ตอนุกรม

ขอ้มูลที่ต้องการส่งออกทุกค่าต้องนำไปเก็บไว้รีจิสเตอร์บัฟเฟอร์ของพอร์ตอนุกรม ซึ่งก็คือรีจิสเตอร์ SBUF ดังตัวอย่าง

```
MOV SBUF,#'A"
```

จากคำสั่งข้างต้นเป็นการส่งข้อมูลของตัวอักษร A ออกไปยังพอร์ตอนุกรมของไมโครคอนโทรลเลอร์ อย่างไรก็ตามก่อนทำการส่งข้อมูลทุกครั้งต้องแน่ใจว่าบิต TI เคลียร์รหือมีค่าเป็น "0"และเมื่อทำการส่งข้อมูลเรียบร้อยแล้ว ก็จะเกิดการเซตบิต TI เพื่อแจ้งให้ทราบดังตัวอย่างโปรแกรมต่อไปนี้

```
CLR  TI          ; เคลียร์บิต TI เพื่อเตรียมการส่งข้อมูลออก
MOV  SBUF,#'A'   ; ส่งข้อมูลของตัวอักษรA ไปยังพอร์ตอนุกรม
JNB  TI,$        ; รอการเซตของบิต เพื่อแจ้งการส่งข้อมูลที่เสร็จสมบูรณ์
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การอ่านหรือรับข้อมูลจากพอร์ตอนุกรม

การรับข้อมูลจากพอร์ตอนุกรมสามารถกระทำได้ง่ายมาก เพียงทำการตรวจสอบว่าบิตเกิดการเซตขึ้นหรือไม่ถ้าพบว่ามีการเซตเกิดขึ้นแล้วให้ทำการอ่านค่าจากรีจิสเตอร์ SBUFโดยต้องทำการโอนย้ายข้อมูลผ่านทางแอกคิวมูเลเตอร์ หรือรีจิสเตอร์ A ดังตัวอย่าง

```
CLR  RI          ; เคลียร์บิต TI เพื่อเตรียมการส่งข้อมูลออก
JNB  RI,$        ;รอคอยการเซตของบิต RI อันเป็นการแจ้งให้ทราบว่า การรับข้อมูลเสร็จสมบูรณ์และมีข้อมูลเกิดขึ้นที่รีจิสเตอร์ SBUF
MOV  A,SBUF      ; อ่านค่าจากรีจิสเตอร์ โดยการโอนย้ายข้อมูลผ่านทางรีจิสเตอร์ A
CLR  RI          ; หลังจากทำการอ่านข้อมูลเรียบร้อยแล้ว ต้องทำการเคลียร์บิต RI เสมอ
```

ระบบสื่อสารข้อมูลอนุกรมแบบหนึ่งสาย (1 – Wire™ Serial Bus)

ระบบการสื่อสารข้อมูลแบบนี้ผู้ค้นคิดคือ ดัลลัสเซมิคอนดักเตอร์ ดังนั้นในบางครั้งจึงเรียกระบบสื่อสารข้อมูลแบบนี้ว่า ระบบสื่อสารข้อมูลดัลลัสหนึ่งสาย (The Dallas 1-Wire Bus) ระบบสื่อสารข้อมูลแบบนี้เป็นระบบที่มีความชาญฉลาด และใช้จำนวนสายสัญญาณเพียง 1 เส้นเท่านั้น โดยไม่ต้องมีสายสัญญาณนาฬิกามาคอยควบคุมจังหวะการถ่ายทอดข้อมูลเหมือนกับระบบสื่อสารข้อมูลอนุกรมในแบบอื่น ๆ เนื่องจากสายข้อมูลนั้นจะทำหน้าที่เสมือนหนึ่งเป็นสายสัญญาณนาฬิกาในตัว ส่วนค่าของข้อมูลจะพิจารณาจากลักษณะของรูปสัญญาณที่ปรากฏบนสายสัญญาณในแต่ละช่องของเวลาหรือต่อไปนี้จะของเรียกว่า ไทม์สล็อต (time-slot) โดยคาบเวลาต่ำสุดและสูงสุดของสถานะต่าง ๆ ในการสื่อสารข้อมูลในแต่ละไทม์สล็อต มีการกำหนดของบเขตไว้อย่างชัดเจน การถ่ายทอดข้อมูลจะเกิดขึ้นในแจ่ละไทม์สล็อตนั้น รูปแบบการถ่ายทอดข้อมูลจะเป็นแบบอะซิงโครนัสในระดับบิต ไม่มีการกำหนดความยาวของข้อมูลเป็นระดับไบต์ ระบบสื่อสารแบบนี้เหมาะที่จะใช้ในการสื่อสารข้อมูลระหว่างไอซีบนแผงวงจรเดียวกัน หรือสร้างเป็นโครงข่ายสื่อสารแบบทวิสต์แพร์ก็ได้

คุณสมบัติทางเทคนิคของระบบบัสหนึ่งสาย

สายสัญญาณบนระบบบัสแบบหนึ่งสายนี้จะเป็นสายสัญญาณแบบสองทิศทาง แต่ข้องมูลจะสามารถเดินทางได้ในทิศทางเดียวภายในช่วงเวลาหนึ่ง ๆ นั่นคือ มีลักษณะคล้ายกับระบบสื่อสารแบบฮาล์ฟดูเพล็กซ์ (halfduplex) ตัวอย่างที่เห็นได้ชัดคือ การใช้งานวิทยุสื่อสารหรือวิทยุสมัครเล่น อุปกรณ์บนระบบบัสต้องมีการระบุอย่างชัดเจนว่าตัวใดเป็นอุปกรณ์มาเตอร์ ตัวใดเป็น

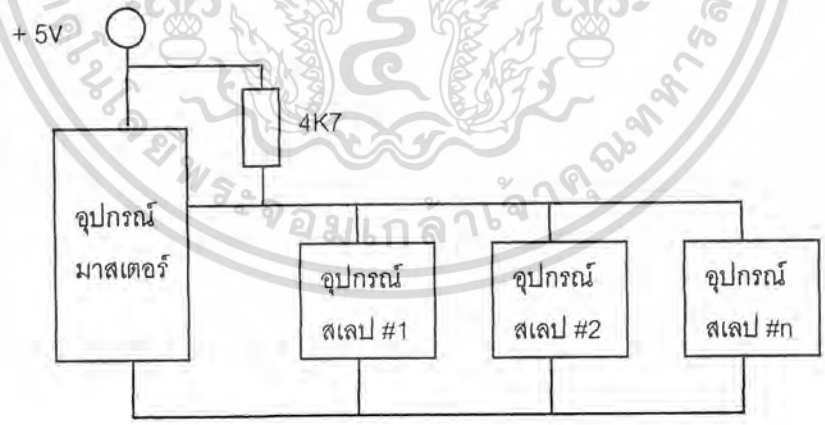
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ในระหว่างการทำงานอุปกรณ์มาสเตอร์และสเลฟสามารถเป็นได้ทั้งตัวส่งและตัวรับ ขึ้นอยู่กับเงื่อนไขของการทำงานในขณะนั้น ยกตัวอย่าง ถ้าหากมีการเขียนข้อมูลจากอุกรณ์มาสเตอร์ไปยังอุปกรณ์สเลฟ ตัวส่งคืออุปกรณ์มาสเตอร์ ตัวรับคืออุปกรณ์สเลฟ ในทางตรงข้าม หากเป็นการอ่านข้อมูลจากอุปกรณ์สเลฟ ตัวส่งจะกลายเป็นอุปกรณ์สเลฟ และตัวรับคืออุปกรณ์มาสเตอร์ ในระบบบัส 1 ระบบต้องมีอุปกรณ์มาสเตอร์เพียงตัวเดียวเท่านั้น

สายสัญญาณของระบบบัสนี้ต้องกำหนดสภาวะปกติไว้ที่ลอจิกสูง สามารถทำได้โดยการต่อตัวต้านทานค่าประมาณ 4.7k พูลอัปกับไฟเลี้ยง +5vΩ ดังนั้นอุปกรณ์ที่นำเข้าต่อบนระบบบัสนี้จึงต้องออกแบบให้ภาคเอาต์พุตที่ต่องต่อกับสายสัญญาณมีลักษณะเป็นคอลเล็กเตอร์เปิดหรือเดรนเปิด ในรูปที่ 2.19 แสดงไดอะแกรม การสื่อสารข้อมูลอนุกรมแบบหนึ่งสายเบื้องต้น

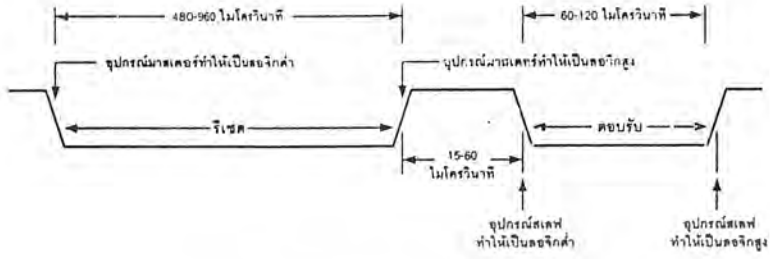
คุณสมบัติของไทม์สล็อต

อุปกรณ์มาสเตอร์จะเป็นอุปกรณ์เพียงตัวเดียวบนระบบบัสหนึ่งสายนี้ที่สามารถทำการอินนิเชียลสายสัญญาณได้ โดยอุปกรณ์มาสเตอร์จะกำเนิดจุดเริ่มต้นของไทม์สล็อตด้วยการทำให้สายสัญญาณเป็นลอจิกต่ำในช่วงเวลาหนึ่ง จากนั้นก็จะทำให้กลับมาเป็นลอจิกสูง ถ้าหากอุปกรณ์สเลฟต้องการส่งข้อมูลมายังอุปกรณ์มาสเตอร์อุปกรณ์สเลฟจะเป็นตัวควบคุมสภาวะของสายสัญญาณต่อไป จนเสร็จสิ้นกระบวนการ แต่ถ้าหากอุปกรณ์มาสเตอร์ต้องการส่งข้อมูลก็จะสามารถดำเนินการต่อไปได้เลย



รูปที่ 2.19 การเชื่อมต่อบนระบบบัสหนึ่งสาย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.20 ไทม์สล็อตการรีเซตและการตอบรับของอุปกรณ์บนระบบบัสหนึ่งสาย

ฟังก์ชั่นของไทม์สล็อตที่กำหนดโดยอุปกรณ์มาสเตอร์มีด้วยกัน 4 ฟังก์ชั่นคือ ไทม์สล็อตของการรีเซต (RESET), การอ่านข้อมูล (READ DATA) การเขียนข้อมูล "1" (WAITE ONE) และการเขียนข้อมูล "0" (WRITE ZERO) ไทม์สล็อตรีเซตใช้ในการเริ่มต้นติดต่อกับอุปกรณ์สเลฟ ในขณะที่ไทม์สล็อตการอ่านจะสำหรับอ่านข้อมูลที่ส่งมาจากอุปกรณ์สเลฟ ส่วนไทม์สล็อตการเขียนข้อมูล "1"และ "0" ใช้สำหรับเขียนข้อมูลไปยังอุปกรณ์สเลฟผ่านสายสัญญาณของระบบ

ทางด้านอุปกรณ์สเลฟมีฟังก์ชั่นของไทม์สล็อตอยู่ทั้งสิ้น 3 ฟังก์ชั่น คือ ไทม์สล็อตของการตอบสนอง (PRESENCE), การเขียนข้อมูล "1" (WRITE ONE) และการเขียนข้อมูล "0" WRITE ZERO) ไทม์สล็อตของการตอบสนองให้สำหรับตอบสนองการติดต่อจากอุปกรณ์มาสเตอร์ โดยอุปกรณ์สเลฟตัวที่ถูกเลือกจากอุปกรณ์มาสเตอร์จะต้องส่งสัญญาณตอบสนองลงบนสายสัญญาณเพื่อแจ้งให้อุปกรณ์มาสเตอร์ทราบว่า ขณะนี้สามารถติดต่อกันได้แล้ว ส่วนไทม์สล็อตการเขียนข้อมูล "1" และ "0" ใช้สำหรับส่งขอ้มูลไปยังอุปกรณ์มาสเตอร์ผ่ายสายสัญญาณของระบบซึ่งจะสัมพันธ์กับไทม์สล็อตการอ่านข้อมูลของอุปกรณ์มาสเตอร์

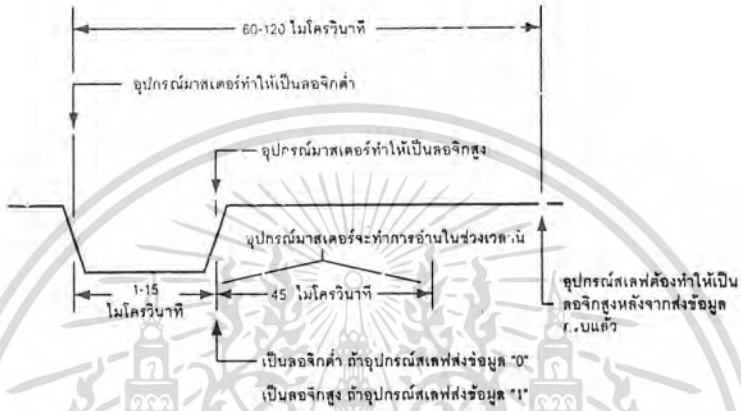
การแยกแยะฟังก์ชั่นของแต่ละไทม์สล็อตจะใช้ความยาวของคาบเวลาและลักษณะของรูปสัญญาณเป็นตัวกำหนดและทุกครั้งที่มีการเปลี่ยนแผลงฟังก์ชั่นต้องทำให้สายสัญญาณอยู่ในสภาวะว่างเสมอ ซึ่งก็คือการทำให้สายสัญญาณเป็นลอจิกสูงอย่างน้อยเป็นเวลา 1 ไมโครวินาที

ไทม์สล็อตการรีเซตและตอบสนอง

อุปกรณ์มาสเตอร์ทำให้เกิดการรีเซตบนสายสัญญาณเพื่อแจ้งแก่อุปกรณ์สเลฟ โดยการทำให้สายสัญญาณเป็นลอจิกต่ำต่อเนื่องอย่างน้อย 480 ไมโครวินาที และจะต้องทำให้สายสัญญาณกลับมาเป็นลอจิกสูงภายใน 480 ไมโครวินาทีหลังจากนั้น ถ้าหากมีอุปกรณ์สเลฟต่ออยู่บนสาย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สัญญาณ จะมีการตอบสนองสัญญาณรีเซตนั้นด้วยสัญญาณตอบสนอง (PRESENCE) โดยการทำให้สายสัญญาณเป็นลอจิกต่ำต่อเนื่องนานประมาณ 60 – 240 ไมโครวินาที หลังจากสัญญาณรีเซตปรากฏประมาณ 15 – 60 ไมโครวินาที ในรูปที่ 2.20 แสดงไทม์สล็อตของการรีเซตและการตอบสนอง



รูปที่ 2.21 ไทม์สล็อตการอ่านข้อมูลของอุปกรณ์มาสเตอร์ ซึ่งตรงกับไทม์สล็อตการเขียนข้อมูลของอุปกรณ์สเลฟ

ไทม์สล็อตการอ่านข้อมูลของอุปกรณ์มาสเตอร์และการเขียนข้อมูลของอุปกรณ์สเลฟ

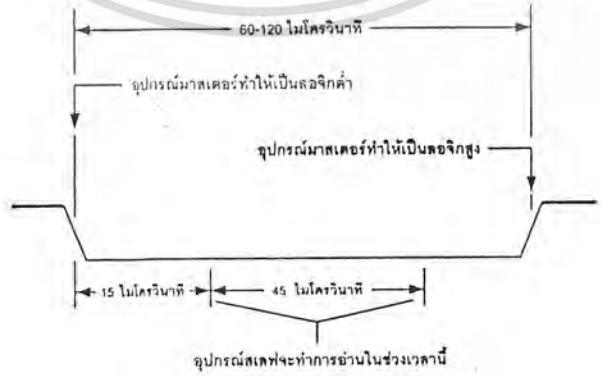
เมื่อต้องการอ่านข้อมูลจากอุปกรณ์สเลฟ อุปกรณ์มาสเตอร์จะทำให้สายสัญญาณเป็นลอจิก ต่ำประมาณ 1-15 ไมโครวินาที จากนั้นต้องทำให้สถานะของสายกลับมาเป็นลอจิกสูง อุปกรณ์สเลฟจะส่งข้อมูลมาให้อุปกรณ์มาสเตอร์โดยถ้าข้อมูลเป็น "0" อุปกรณ์สเลฟจะทำให้สายสัญญาณเป็นลอจิกต่ำนานประมาณ 45 ไมโครวินาทีแล้วทำให้สายสัญญาณกลับมาสู่สภาวะลอจิกสูงอีกครั้ง แต่ถ้าเป็นข้อมูล "1" อุปกรณ์สเลฟจะทำให้สายสัญญาณเป็นลอจิกสูงต่อเนื่องไปอีก 45 ไมโครวินาที รวมเวลาทั้งหมดในไทม์สล็อตนี้ประมาณ 60-120 ไมโครวินาที นั่นคือในไทม์สล็อตนี้จะต้องใช้เวลารวมไม่เกิน 120 ไมโครวินาที ในขณะที่อุปกรณ์มาสเตอร์จะใช้เวลาในการอ่านข้อมูลอยู่ระหว่าง 15 และ 60 ไมโครวินาทีหลังจากเริ่มต้นไทม์สล็อตนี้ ในรูปที่ 2.22 แสดงรูปสัญญาณของไทม์สล็อตการอ่านข้อมูลของอุปกรณ์มาสเตอร์ซึ่งก็จะมีลักษณะเหมือนกับการเขียนข้อมูลของอุปกรณ์สเลฟ และไทม์สล็อตทั้งสองจะเกิดขึ้นในช่วงเวลาเดียวกัน กล่าวคือ เมื่ออุปกรณ์มาสเตอร์อ่าน อุปกรณ์สเลฟก็ต้องทำการเขียน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ไทม์สล็อตการเขียนข้อมูลของอุปกรณ์มาสเตอร์

เมื่ออุปกรณ์มาสเตอร์ต้องการเขียนข้อมูล อุปกรณ์มาสเตอร์จะทำให้สายสัญญาณเป็นลอจิก ต่ำประมาณ 1 – 15 ไมโครวินาที จากนั้นต้องทำให้สถานะของสายกลับมาเป็นลอจิกสูง แล้วดำเนินการเขียนข้อมูลได้ในทันทีถ้าข้อมูลที่ต้องการเขียนไปยังอุกรณ์สเลฟเป็น "0" อุปกรณ์มาสเตอร์จะทำให้สายสัญญาณเป็นลอจิกต่ำนานประมาณ 45 ไมโครวินาที แล้วทำให้สายสัญญาณกลับมาสู่สภาวะลอจิกสูงอีกครั้ง แต่ถ้าต้องการเขียนข้อมูล "1" อุปกรณ์มาสเตอร์จะทำให้สายสัญญาณเป็นลอจิกสูงต่อเนื่องไปอีก 45 ไมโครวินาที รวมเวลาทั้งหมดในไทม์สล็อตนี้ประมาณ 60 – 120 ไมโครวินาที ในรูปที่ 2.22 แสดงรูปสัญญาณของไทม์สล็อตการเขียนข้อมูลของอุปกรณ์มาสเตอร์ซึ่งก็จะมีลักษณะเหมือนกับการอ่านข้อมูลของอุปรกรณ์สเลฟ และไทม์สล็อตทั้งสองจะเกิดขึ้นในช่วงเวลาเดียวกัน กล่าวคือ เมื่ออุปกรณ์มาสเตอร์เขียน อุปกรณ์ สเลฟก็ต้องทำการอ่านข้อมูล

รูปที่ 2.22 ไทม์สล็อตการเขียนข้อมูล "1" ของอุปกรณ์มาสเตอร์ ซึ่งตรงกับไทม์สล็อตการอ่านข้อมูลของอุปกรณ์สเลฟ



รูปที่ 2.23 ไทม์สล็อตการเขียนข้อมูล "0" ของอุปกรณ์มาสเตอร์

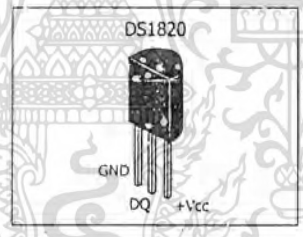
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.7 รูปแบบของการสื่อสารข้อมูลแบบหนึ่งลาย (1 – Wire™ communication protocol)

ในการติดต่อสื่อสารข้อมูลในระบบบัสหนึ่งสายอุปกรณ์มาสเตอร์จะสามารถติดต่อกับอุปกรณ์สเลฟได้ครั้งละ 1 ตัวเท่านั้น ดังนั้นอุปกรณ์สเลฟแต่ละตัวต้องมีข้อมูลกำหนดแอดเดรสเฉพาะตัว โดยจะเก็บไว้ในหน่วยความจำรอมภายในอุปกรณ์สเลฟตัวนั้น ๆ โดยปกติอุปกรณ์สเลฟในระบบบัสหนึ่งสายของดัลลัสนี้จะมีหน่วยความจำขนาด 64 บิตหรือ 8 ไบต์ สำหรับเก็บข้อมูลต่าง ๆ ที่สำคัญของอุปกรณ์แต่ละตัว ซึ่งประกอบด้วย

1. รหัสของตระกูล จำนวน 8 บิต
2. เลขหมายประจำตัว (Serial number) จำนวน 48 บิต
3. รหัสตรวจสอบความผิดพลาด (CRC : Cyclical Redundancy Check) จำนวน 8 บิต

ผู้ใช้งานสามารถอ่านข้อมูลประจำตัวของอุปกรณ์สเลฟได้ด้วยการใช้คำสั่งอ่านหน่วยความจำรอม (Read ROM) ในกรณีที่บนสายสัญญาณมีอุกรณ์สเลฟเยงตัวเดียวไม่จำเป็นต้องอ้างแอดเดรสในการติดต่อ



รูปที่ 2.24 การจัดขาของ DS1820

รูปแบบการติดต่อบนระบบบัสหนึ่งสายจะเริ่มต้นขึ้นเมื่ออุปกรณ์มาสเตอร์ทำการรีเซตและกำหนดแอดเดรสของอุปกรณีที่ทำการติดต่อ ถ้าหากมีอุปกรณ์สเลฟเยงตัวเดียวสามารถข้ามขั้นตอนการติดต่อกับหน่วยความจำรอมในอุปกรณ์สเลฟได้ จะเรียกวิธีการดังกล่าว การไม่ติต่อหน่วยความจำรอม หรือ สคิปรอม (Skip ROM) จากนั้นรอการตอบรับจากอุปกรณ์สเลฟ เมื่อการตอบรับสมบูรณ์ก็จะสามารถเริ่มต้นขั้นตอนการอ่านหรือเขียนขอ้มูลได้ต่อไป

ไอซีตรวจจับอุณหภูมิ DS1820

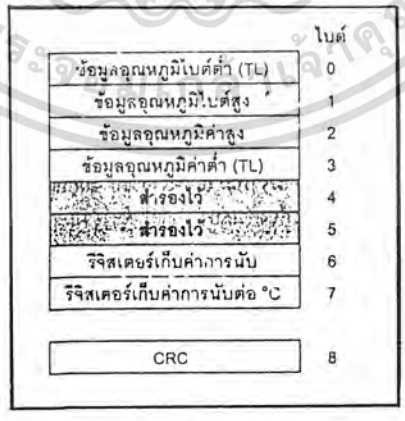
เป็นไอซีตรวจจับอุณหภูมิที่ใช้การติดต่อแบบระบบบัสหนึ่งสาย มีขาต่อใช้งานเพียง 3 ขา คือ DQ ซึ่งเป็นขาเชื่อมต่อกับระบบบัส, ขาต่อไฟเลี้ยงภายนอก และขากราวด์ และมีโครงสร้างการทำงานภายในแสดงในรูปที่ 2.24

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หัวใจสำคัญของ DS1820 อยู่ที่ตัวตรวจจับอุณหภูมิและหน่วยความจำความเร็วสูงที่เรียกว่า สแครตช์(scratchpad) แพด ซึ่งมีขนาด 9 ไบต์ มีการจัดสรรหน่วยความจำส่วนนี้แสดงในรูปที่ 2.25

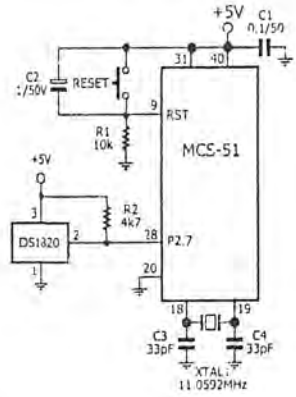
เมื่อวัดอุณหภูมิได้ก็จะนำค่าที่วัดได้นี้มาเก็บไว้ในสแรตช์แพตที่ไบต์ 0 และ 1 ทั้งนี้เนื่องจากไอซี DS1820 สามารถให้ข้อมูลของอุณหภูมิได้ละเอียดถึง 16 บิต เมื่อนำมาแปลงเป็นข้อมูลเลขฐานสิบจึงสามารถแสดงความละเอียดของค่าอุณหภูมิได้ถึง 0.5 องศาเซลเซียสและ 0.9 องศาฟาเรนไฮต์ โดยมีย่านวัดอุณหภูมิ –55 ถึง +125
องศาเซลเซียสหรือ –67 ถึง +257 องศาฟาเรนไฮต์ โดยค่าขององศาฟาเรนไฮต์ต้องใช้การแปลงหน่วยเข้ามาช่วย

รูปที่ 2.25 โครงสร้างการทำงานภายในของไอซีตรวจจับอุณหภูมิ DS1820



รูปที่ 2.26 การจัดสรรพื้นที่ของสแครตช์แพตใน DS1820

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูป 2.27 การเชื่อมต่อ DS1820 กับไมโครคอนโทรลเลอร์ MCS-51

ใช้เวลาในการแปลงค่าอุณหภูมิเป็นข้อมูลดิจิตอลประมาณ 200 มิลลิวินาที สามารถกำหนดขอบเขตของอุณหภูมิที่ทำการวัดได้ และให้แจ้งเตือนเมื่อค่าของอุณหภูมิสูงขึ้นหรือลดต่ำลงถึงค่าที่กำหนด โดยค่าอุณหภูมิที่ทำการวัดได้ และให้แจ้งเตือนเมื่อค่าของอุณหภูมิสูงขึ้นหรือลดต่ำลงถึงค่าที่กำหนด โดยค่าอุณหภูมิที่กำหนดนี้จะเก็บไว้ที่สแครตช์แพตในไบต์ 2 และ 3

คำสั่งเพื่อควบคุมการทำงานของ DS1820

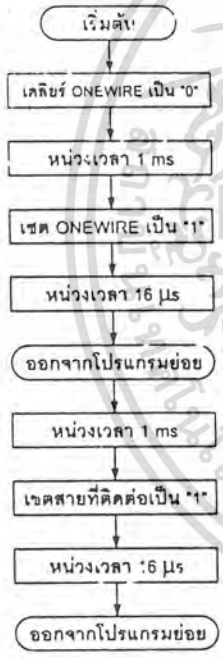
ในการติดต่อกับไอซี DS1820 จะมีคำสั่งที่ต้องส่งให้แก่ DS1820 เพื่อกำหนดรูปแบบการทำงาน คำสั่งที่ใช้มากที่สุดมีด้วยกัน 3 คำสั่งดังนี้

1. คำสั่งไม่ติดต่อกับหน่วยความจำรอมหรือสคิปรอม (Skip ROM) เนื่องจากในการใช้งาน DS1820โดยปรกติแล้วจะมี DS1820 อยู่บนสายสัญญาณเพียงตัวเดียว จึงไม่จำเป็นต้องใช้ข้อมูลกำหนดแอดเดรส ดังนั้นจึงไม่ติองติดต่อกับหน่วยความจำรอมเพื่ออ่านข้อมูล ข้อมูลของคำสั่งสคิปรอมที่ต้องส่งให้ DS1820 คือ 0ccH
2. คำสั่งแปลงอุณหภูมิ (Convert T) มีค่าเท่ากับ 44H เมื่อส่งคำสั่งนี้ให้ DS1820 จะต้องทำการวนลูปรออย่างน้อย 200 มิลลิวินาที เพื่อให้ DS1820 ได้ใช้เวลานี้ในการแปลงค่าอุณหภูมิเป็นข้อมูลดิจิตอลมาเก็บไว้ในสแครตช์แพต
3. คำสั่งอ่านข้อมูลจากสแครตช์แพต (Read Scratchpad) มีค่าเท่ากับ 0BEH เมื่อส่งคำสั่งนี้ จะทยอยส่งข้อมูลค่าอุณหภูมิออกมาทั้งหมด 9 ไบต์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การเชื่อมต่อกับไมโครคอนโทรลเลอร์ MCS-51

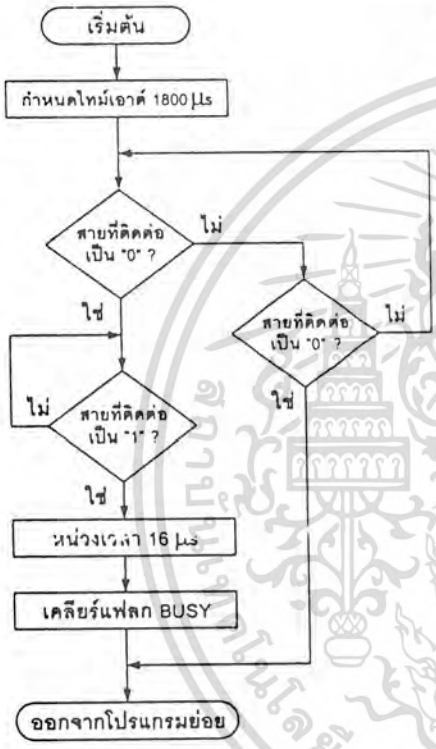
แสดงวงจรการเชื่อมต่อในรูปที่ 2.28 ใช้ขาพอร์ตเพียง 1 ขาเท่านั้นสำหรับการเชื่อมต่อกับ DS1820 โดยต้องมีตัวต้านทานค่า 4.7kΩ ต่อพูลอัปกับไฟเลี้ยง +5v จากนั้นจึงทำการเขียนโปรแกรมเพื่อติดต่อกัน โดยใช้รูปแบบการติดต่อตามมาตรฐานระบบบัสหนึ่งสายของดัลลัส




```
DS1820_RST:   CLR     ONEWIRE        ; Clear ONEWIRE
              ACALL   DELAY_1ms      ; Delay
              SETB    ONEWIRE        ; Set ONEWIRE
              MOV     R4,#6          ; Delay
              DJNZ    R4,$           ;
              RET                    ; Return

DELAY_1ms:    MOV     R6,#0E6H       ; Each loop = 1 ms
DELAY_1ms_1:  NOP
              NOP
              DJNZ    R6,DELAY_1ms_1
              RET
```

รูปที่ 2.28 โฟลวชาร์ตและรายละเอียดโปรแกรมย่อยการรีเซต DS1820

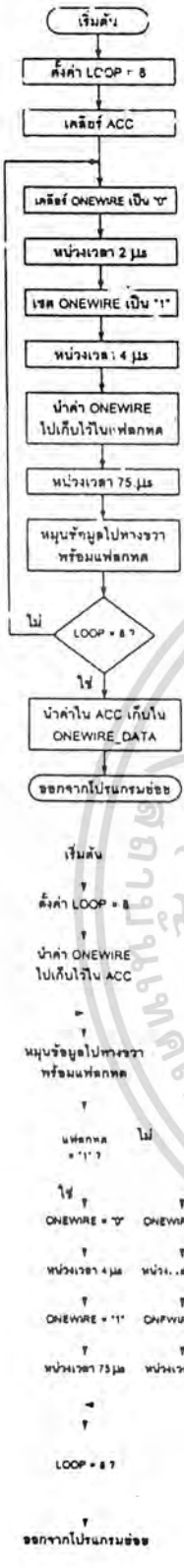
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้




```
DS1820_PRES:    MOV   R4,#8     ; Set Loop wait 1
DS1820_PRES_1:  MOV   R3,#0     ; Set Loop wait 2
DS1820_PRES_2:  JNB   ONEWIRE,DS1820_PRES_3
                      ; Check ONEWIRE was clear?
                DJNZ  R3,DS1820_PRES_2
                      ; Wait loop check 2
                DJNZ  R4,DS1820_PRES_1
                      ; Wait loop check 1
                RET             ; Return
DS1820_PRES_3:  JNB   ONEWIRE,$
                      ; Wait until ONEWIRE set
                MOV   R4,#8     ; Delay
                DJNZ  R4,$      ;
                CLR   BUSY      ; Clear BUSY Flag
                RET             ; Return
```

รูปที่ 2.29 โฟลวชาร์ตและรายละเอียดโปรแกรมย่อยรอการตอบรับจาก DS1820

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้




```
DS1820_RD:          MOV   R4,#8       ; Set loop 8 times
                    CLR   A           ; Clear ACC.
DS1820_RD_LOOP:     CLR   ONEWIRE     ; Clear ONEWIRE
                    NOP               ; Delay
                    NOP
                    SETB  ONEWIRE     ; Set ONEWIRE
                    NOP               ; Delay
                    NOP
                    NOP
                    NOP
                    MOV   C,ONEWIRE ; Get ONEWIRE to Carry
                    ACALL ONEWIRE_DELAY  ; Delay 75 us
                    RRC   A
                    DJNZ  R4,DS1920_RD_LOOP
                                      ; Do until 8 times
                    MOV   ONEWIRE_DATA,A
                                      ; Move ACC. to
                                      ; ONEWIRE_DATA
                    RET               ; Return

ONEWIRE_DELAY:      MOV   6,#012H     ; Each loop = 75 us
ONEWIRE_DELAY_1:    NOP
                    NOP
                    DJNZ  R6,ONEWIRE_DELAY_1
                    RET

DS1820_WR:          MOV   R4,#8       ; Set loop 8 times
                    MOV   A,ONEWIRE_DATA      ; Get DATA
DS1820_WR_LOOP:     RRC   A
                    JNC   DS1820_WR_L ; Carry Flag was set ?
                    CLR   ONEWIRE     ; Set => TX high
                    NOP               ; Delay
                    NOP
                    NOP
                    NOP
                    SETB  ONEWIRE     ; Set ONEWIRE
                    ACALL ONEWIRE_DELAY  ; Delay 75 us
                    AJMP  DS1820_WR_NX   ; Jump to next write
DS1820_WR_L:        CLR   ONEWIRE     ; Clear => TX Low
                    ACALL ONEWIRE_DELAY ; Delay 75 us
                    SETB  ONEWIRE     ; Set ONEWIRE
                    NOP               ; Delay
                    NOP
                    NOP
                    NOP
DS1820_WR_NX:       DJNZ  R4,DS1820_WR_LOOP
                                      ; Do until 8 times
                    RET               ; Return
ONEWIRE_DELAY:      MOV   R6,#012H    ; Each loop = 75 us
ONEWIRE_DELAY_1:    NOP
                    NOP
                    DJNZ  R6,ONEWIRE_DELAY_1
                    RET
```

รูปที่ 2.30 โฟลวชาร์ตและรายละเอียดโปรแกรมย่อยการอ่านและเขียนข้อมูลกับ DS1820

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การเขียนโปรแกรมเพื่อติดต่อกับ DS1820

จากรายละเอียดของรูปแบบการสื่อสารในระบบบัสหนึ่งสายที่กล่าวมาตั้งแต่ต้น สามารถนำมาใช้เพื่อเป็นข้อมูลในการเขียนโปรแกรมติดต่อ โดยจะต้องเขียนโปรแกรมย่อยเพื่อสร้างไทม์สล็อตของฟังก์ชั่นต่าง ๆ ดังแสดงรายละเอียดของโฟลวชาร์ตและโปรแกรมย่อยในรูปที่ 2.27 ถึง 2.29 เริ่มจากโปรแกรมย่อยการรีเซต(RESET) โปรแกรมย่อยรอการตอบรับจาก DS1820 (PRESENCE), โปรแกรมย่อยการอ่านและเขียนข้อมูลกับDS1820

2.8 ระบบเซอร์โว (Servo Systems)

คือระบบการ คอนโทรลเครื่องจักรกล ที่มีการทำงานเป็น ไดนามิค (Dynamic) หมายถึง ระบบการคอนโทรล ที่มีตัวแปรอินพุต r(t) เปลี่ยนแปลงไปตามเวลา ดังนั้นตามหลักการของระบบคอนโทรลที่มีการป้อนกลับแบบลูปปิดแล้ว การทำงานของระบบเซอร์โวตัวแปรเอาต์พุตของระบบจะติดตามการเปลี่ยนแปลงของตัวแปรอินพุตไปในเวลาเดียวกัน ซึ่งเราอาจจะเรียกระบบดังกล่าวได้อย่างหนึ่งว่าเป็นระบบแทรคกิ้งคอนโทรล (Tracking Control System) หรือระบบโฟลโลอิ้งคอนโทรล (Following Control System)

สรุปได้ว่าระบบเซอร์โว คือ ระบบการคอนโทรลเครื่องจักรกลที่ไปขับโหลดให้มีการเคลื่อนไหวในลักษณะเป็นตำแหน่งมุม ความเร็วรอบ อัตราเร่ง และแรงบิด หรือ ระบบเซอร์โวก็คือ ระบบการคอนโทรลมอเตอร์นั่นเองซึ่งจำแนกออกได้เป็น 4 แบบดังนี้

1) ระบบการคอนโทรลตำแหน่งเพลาของมอเตอร์ , เมื่อ $r(t) = \theta(t)$

2) ระบบการคอนโทรลความเร็วรอบของเพลามอเตอร์ , เมื่อ

$r(t) = d\theta(t) / d(t) = \omega(t)$

3) ระบบการคอนโทรลอัตราเร่งของเพลามอเตอร์ , เมื่อ $r(t) = d\omega(t) / dt = \propto(t)$

4) ระบบการคอนโทรลแรงบิดของเพลามอเตอร์ , เมื่อ $r(t) = Jd\omega / dt = T(t)$

ระบบเซอร์โวที่นำมาใช้ในโครงการนี้จะใช้แบบการคอนโทรลตำแหน่งมุม ซึ่งระบบเซอร์โวประกอบด้วยเซอร์โวมอเตอร์ และวงจรส่วนควบคุมการทำงานของเซอร์โวมอเตอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ดีซีเซอร์โวมอเตอร์ (DC.motor)

ดีซีเซอร์โวมอเตอร์เป็นตัวทรานสดิวเซอร์ (transducer) แรงบิดซึ่งมีการออกแบบให้มีคุณลักษณะพิเศษ คือ แรงบิดของเพลาของดีซีมอเตอร์จะเป็นสัดส่วนโดยตรงกับกระแสอาร์มาเจอร์ (Armature) แรงบิดของเพลาของดีซีมอเตอร์จะได้จากผลสนามแม่เหล็กและขดลวดตัวนำ ในที่นี้กระแสที่ไหลในขดลวดตัวนำจะสร้างฟิวส์ (Field) ที่ประกอบด้วยเส้นแรงแม่เหล็ก Φ และขดลวดตัวนำเหล่านั้นอยู่ห่างจากศูนย์กลางการหมุนเท่ากับ r ความสัมพันธ์ระหว่างแรงบิดของเพลาและกระแสเท่ากับ

$$T = K\Phi I$$

เมื่อ T คือแรงบิดของเพลามีหน่วยเป็นนิวตัน-เมตร

Φ คือเส้นแรงแม่เหล็กมีหน่วยเป็นเวเบอร์ (wb)

I คือกระแสเป็นแอมแปร์ (A)

K คือค่าคงที่

ดังนั้นแรงบิดของเพลาจะเป็นสัดส่วนโดยตรงกับผลคูณของเส้นแรงแม่เหล็ก และกระแสเมื่อขดลวดตัวนำเคลื่อนที่ในสนามแม่เหล็ก ก็จะทำให้เกิดแรงดันตกคร่อมตัวมันเอง แรงดันตกคร่อมนี้จะเป็นสัดส่วนกับความเร็วของเพลาของมอเตอร์ และต้านการไหลของกระแส ความสัมพันธ์ระหว่างโวลท์เตจย้อนกลับ (back emf) นี้และความเร็วของเพลามอเตอร์ คือ

$$E = K\Phi\omega$$

เมื่อ E คือ แรงดันย้อนกลับมีหน่วยเป็นโวลท์ (V)

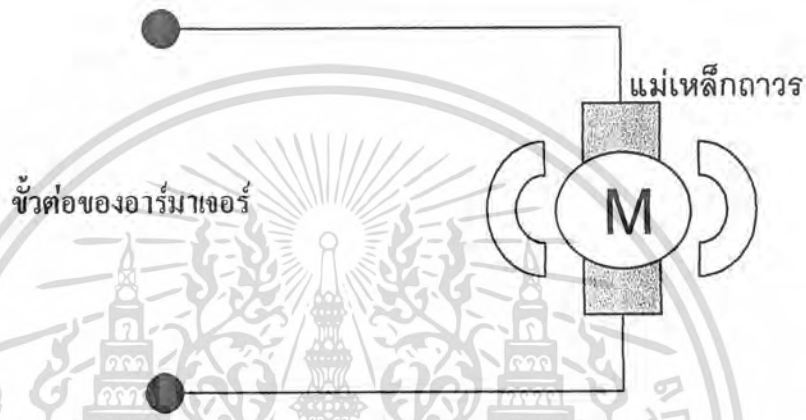
Φ คือ เส้นแรงแม่เหล็กมีหน่วยเป็นเวเบอร์ (wb)คือ

ω ความเร็วของมอเตอร์มีหน่วยเป็นเรเดียนต่อวินาที (radian/sec)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ดีซีมอเตอร์นั้นสามารถจำแนกออกไปได้หลายประเภท ซึ่งจะขึ้นอยู่กับวิธีในการสร้าง จึงขอกล่าวรวมๆ เฉพาะที่รู้จักกันเป็นส่วนใหญ่ในปัจจุบันนั่นคือ ดีซีมอเตอร์แบบขนาน (Shunt DC.motor) แบบอนุกรม (Series) , แบบผสม (compound) และแบบแม่เหล็กถาวร (permanent magnet motor)

ในโครงงานนี้ดีซีมอเตอร์แบบแม่เหล็กถาวรได้ถูกนำมาใช้ ซึ่งมีลักษณะดังรูปต่อไปนี้



รูปที่ 2.31 โครงสร้างของดีซีมอเตอร์แบบแม่เหล็กถาวร

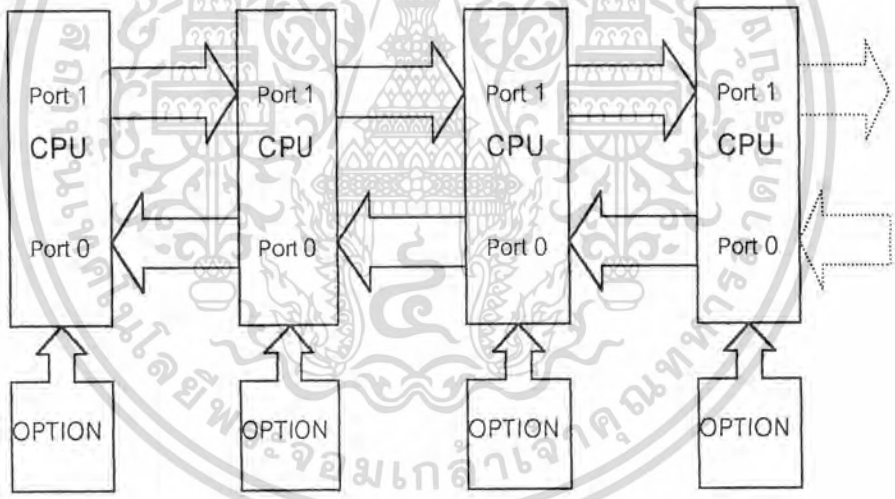
มอเตอร์แบบนี้จะใช้การกระตุ้นฟิลด์ของมอเตอร์เป็นแม่เหล็กถาวรซึ่งต่างจากแบบอื่นที่ใช้ขดลวด ซึ่งแบบนี้จะให้เส้นแรงของฟิลด์มีค่าคงที่ ดังนั้นอัตราส่วนระหว่างกระแสอาร์มาเจอร์และแรงบิดจะมีค่าคงที่ด้วย ซึ่งจะมีข้อดีคือ ไม่มีการสูญเสียในฟิลด์ ทำให้มีประสิทธิภาพสูงกว่าและมีขนาดเล็กกว่าเมื่อเทียบกับมอเตอร์แบบใช้ขดลวดในการกระตุ้น ที่มีขนาดกำลังม้าเท่ากันจึงเหมาะที่จะนำไปใช้งานที่ต้องการแรงบิดของโหลดสูงๆ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 3
โครงสร้างและการออกแบบ

3.1 การออกวงจรกับความสัมพันธ์ของ cpu

ในโครงงานนี้มีการออกแบบหุ่นยนต์ให้มีการใช้หน่วยความจำและโปรแกรมอย่างอิสระ โดยหุ่นแต่ละข้อจะใช้ Microcontroller (mcs51) แยกกัน ข้อล่ะตัวแต่มีการเชื่อมต่อกันโดยผ่าน PORT INPUT และ OUTPUT ของแต่ละ Microcontroller เข้าด้วยกันฉะนั้นการเดินและการควบคุมจะแยกกันโดยเป็นแบบกึ่งอิสระโดย ในบางเงื่อนไขจะต้องเชื่อมถึงกันเพื่อประโยชน์ในการเชื่อมต่อกันเป็นหุ่นตะขาบ และจะมีส่วนที่เพิ่มเติมเข้ามาในหุ่นได้คือส่วนของ option เราสามารถนำวงจรต่างๆที่จะนำมาติดตั้งในหุ่นโดยเราสามารถแยกวงจรออกให้เล็กลงได้โดยการแยกไปอยู่ในหุ่นแต่ละตัวและต่อเชื่อมถึงกันได้ผ่าน Microcontroller ดังรูปที่ 3.1



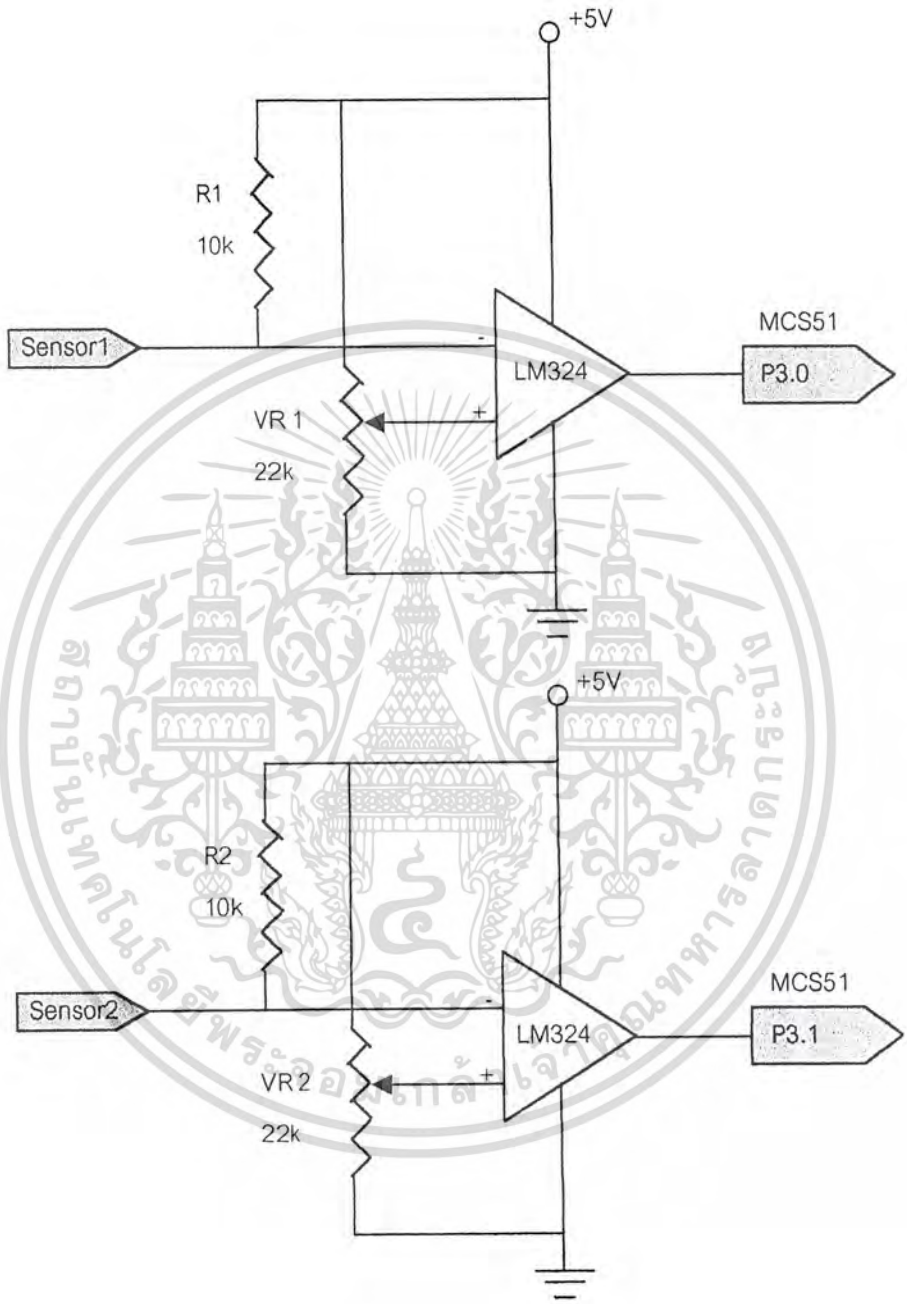
รูปที่ 3.1 การเชื่อมต่อข้อมูลของหุ่นแต่ล่ะข้อ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.2 การออกแบบวงจร SENSOR

ในการควบคุมหุ่นยนต์นั้นภายในหุ่นทั้งหมดจะมี 8 ข้อ โดยข้อแรก จะใช้ SENSOR ตรวจจับสิ่งกีดขวางเพื่อสะดวกในการหลบหลีกในระยะที่เหมาะสมได้ การออกแบบนั้นเราจะใช้ SENSOR ทั้งหมด 2 ตัวชนิดของ SENSOR เป็นแบบ Opto Sensor ของ OMRON ที่สามารถเป็นภาครับส่งภายในตัว และสามารถปรับ Vref ของตัวเองได้ ซึ่งนั้นก็หมายถึงว่าเราสามารถปรับระยะของ SENSOR ได้ 2 วิธี คือปรับที่ ภาคขยายสัญญาณ หรือ ปรับที่ ตัวSENSOR ก็ได้ และที่เราต้องนำ SENSOR มาต่อเข้ากับภาคขยายสัญญาณ นั้นเนื่องจากสัญญาณที่ออกมาจากตัวSENSOR เองมีระดับของสัญญาณที่ค่อยข้างต่ำมากๆจึงไม่สามารถจะนำสัญญาณที่ได้จาก SENSOR มาต่อใช้โดยตรงร่วมกับ Microcontroller ได้ในการขยายสัญญาณนั้น จะใช้วงจร Opamp Comparator โดยมีลักษณะดังนี้ จะมีด้วยกัน 2 ชุด โดย ออปแอมป์ ที่ใช้ เป็น IC เบอร์ LM 324 (ใน LM 324 หนึ่งตัวจะมี ออปแอมป์ อยู่ทั้งหมด 4 ตัว)เมื่อเราได้ทำการขยายสัญญาณที่ได้จาก SENSOR ก็สามารถนำแรงดัน เอาท์พุทจากออปแอมป์ที่ได้ส่งไปยัง Microcontroller เพื่อประมวลผลต่อไปได้ ดังรูปที่ 3.2

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.2 วงจรตรวจจับสิ่งกีดขวาง Collision Avoidance sensor

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากวงจรดังรูป SENSOR ที่ใช้ คือ OMRON นั้นจะช่วยได้ตัดปัญหาเรื่องแสงรบกวนได้เป็นอย่างดี เนื่องจากSENSOR ที่ใช้มีการป้องกันแสงเป็นอย่างดีและยังมีการส่งสัญญาณออกไปกระทบวัตถุด้วยวิธีการ MOD สัญญาณออกไปเพื่อตัดปัญหาเรื่องแสงได้โดยที่ไม่ต้องใช้การเขียนโปรแกรม MOD สัญญาณจาก Microcontroller เพื่อช่วยในการตัดแสงเลยเป็นการประหยัดหน่วยความจำและพื้นที่ของการเขียนโปรแกรมและลดความยุ่งไปได้มากทีเดียว จากวงจรเมื่อ ออปแอมป์ได้อินพุทจาก ตัว SENSOR มาแล้ว จะทำการเปรียบเทียบกับ Vref ของตัวออปแอมป์เองเมื่อแรงดันของ SENSOR มีค่ามากกว่าระดับแรงดันอ้างอิง และเมื่อนำมาเทียบกันกับแรงดันอ้างอิงแล้วออปแอมป์จะทำการขยายสัญญาณออกไปโดยที่จะขยายออกไป โดยที่เราจะทำการป้อน ไฟ +5 โวลท์ เป็นไฟ vcc ของออปแอมป์ฉะนั้นการขยายที่ออปแอมป์จะมีค่าแรงดันเอาท์พุท เป็น บวก เท่ากับแรงดันที่ป้อนให้ออปแอมป์ (vcc)ก็จะสามารถนำไปใช้กับ Microcontroller ได้เพราะมี อินเตอร์เฟสเดียวกันที่ +5 โวลท์

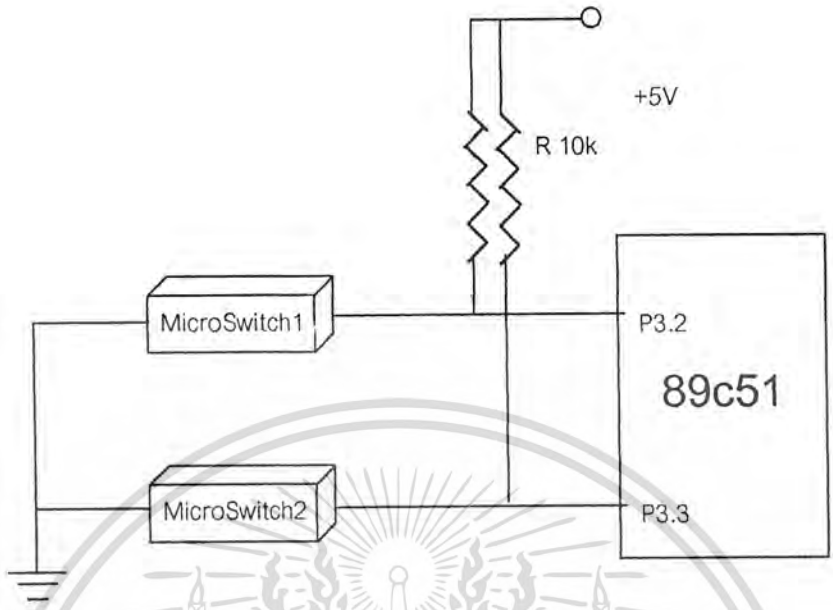
เราจะทำการปรับค่าของความต้านทาน (VR 22 K)ให้แรงดันอ้างอิงสูงกว่าแรงดันจาก SENSOR และภายในตัวSENSOR จะต้องปรับระยะของ SENSOR DETECTION ให้เท่ากันทั้งสองตัว

ปรับที่ตัวต้านทานปรับค่าได้(VR 22 K) ให้แรงดันอ้างอิงเท่ากับ 2.2 โวลท์ (ระดับสัญญาณอินพุทที่ได้มีค่าต่ำ)

แรงดันจาก SENSOR 2 ตัวเท่ากันคือ 2.0 โวลท์

ส่วนของ SENSOR อีกแบบที่ใช้ในการตรวจสอบการเลี้ยวของตัวหน้าเราจะใช้ Micro Switchในการตรวจสอบสัมผัสว่ามีตัวหน้ามีทิศทางการเลี้ยวไปในทางใดในการทำงานของ Micro Switch ในการออกแบบไม่ยุ่งยากมากมายเท่าไรโดยจะนำ Micro Switch มาต่อกับกราวด์และอีกขั้วต่อกับไฟ +5 โวลท์โดยที่มีการต่อ ตัวตานทานเป็นการ pull upไว้ด้วยกันไม่ให้ Microcontroller เกิดการเสียหายโดยเลือกที่ Micro Switch ให้ปกติเป็น open circuit เมื่อมีการสัมผัสก็กลายเป็นการต่อไฟ 0 โวลท์เข้าที่ อินพุทของ Microcontroller และนำค่าที่ได้ไปประมวลผล ดังรูปที่ 3.3

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

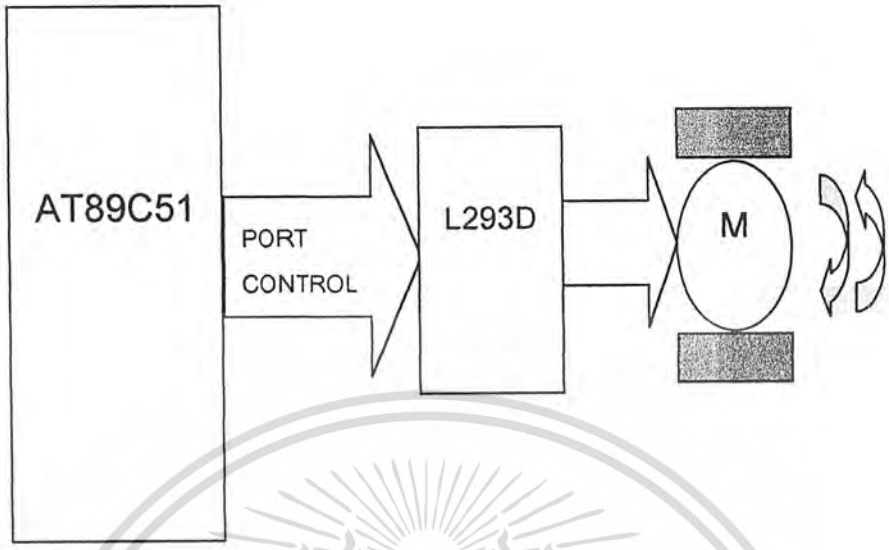


รูปที่ 3.3 การต่อ Micro Switch กับ Mcs51

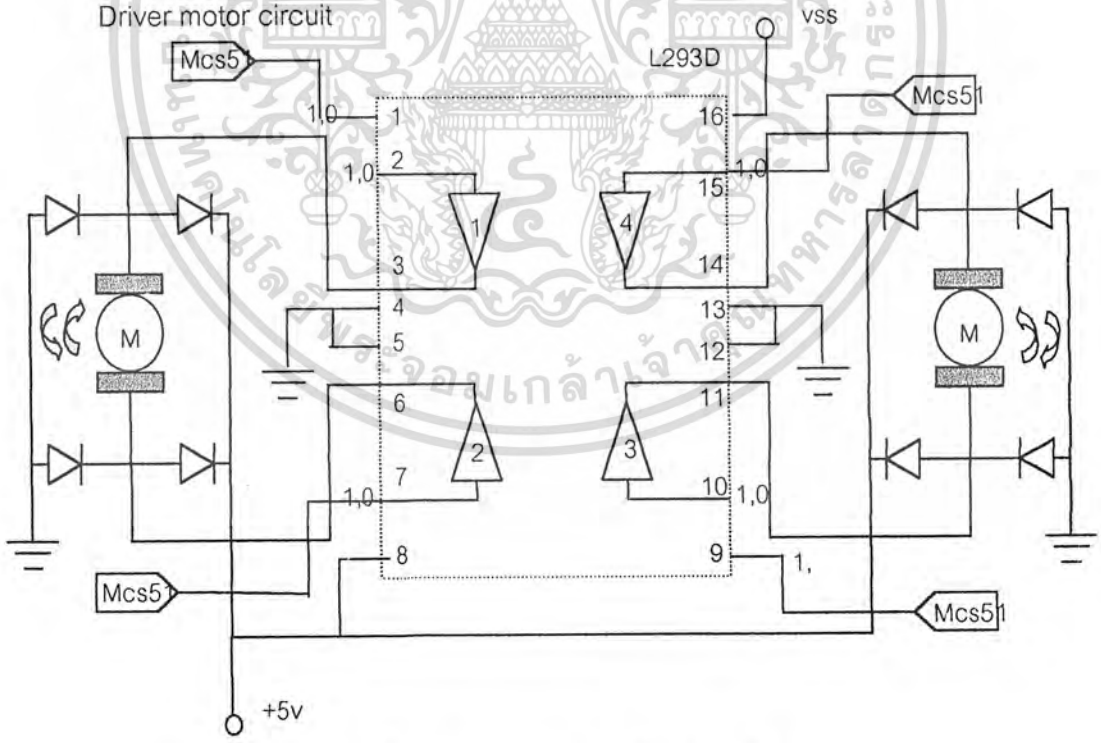
3.3 การออกแบบวงจรขับมอเตอร์

การขับเคลื่อนของหุ่นเราจะใช้ มอเตอร์ เป็นแกนหลักในที่นี้เราเลือกที่จะใช้ DC MOTOR เพราะมีราคาถูกและยังควบคุมได้สะดวกกว่า SERVO หรือ AC MOTOR และยังมี แรงบิดภายในตัว มอเตอร์ เองสูงด้วย และยังจะเพิ่มแรงได้โดยการทดรอบ มอเตอร์ ชุดเฟืองดังได้กล่าวไว้ในบทที่ 2 แต่เดิมจะใช้ TRในการขยายและควบคุมทิศทางของมอเตอร์ แต่เนื่องจากได้มีการผลิต IC เพื่อช่วยในการควบคุม DC MOTOR และมีประสิทธิภาพที่ดีด้วย มีแรงขับกระแสสูงที่เพียงพอในการใช้ในแต่ช่องของ IC จึงเปลี่ยนจากเดิมมาใช้ IC แทน นั้นคือ IC L293D โดยมีการต่อตามรูปที่ 3.4 และรูปที่ 3.5 วิธีการทำงานคือจะนำ IC L293D ต่อเข้ากับ พอรต์ เอาท์พุท ของ Microcontroller โดยตรงต่อไฟ vcc +5 โวลท์ให้กับ IC ส่วน VSSคือไฟที่ต้องการใช้ของมอเตอร์ ในที่นี้ใช้ไฟ 9 โวลท์ ในตัว IC หนึ่งตัวสามารถที่จะควบคุมมอเตอร์ได้ถึง 2 ตัวทีเดียวโดยใช้วิธีการต่อมอเตอร์กับวงจรบริดท์เรดติไฟท์เพื่อจัดกระแสโดยวิธีการควบคุมคือป้อนไฟ 0,1 หรือ 1,0 ให้กับขาอินพุท1และ2โดยจะทำการต่อมอเตอร์เข้ากับขาเอาท์พุท 1 และ 2 เช่นกันเมื่อเราป้อนไฟ 0,1ก็จะทำให้มอเตอร์หมุนไปในทิศทางหนึ่งและถ้าป้อน 1,0ก็จะทำให้มอเตอร์หมุนไปในทิศทางตรงกันข้ามโดยการควบคุมเราจะใช้ Microcontroller ในการควบคุมส่วนขาEnableนั้นเราจะเซตไว้ให้เป็น "1" เพื่อให้ภาคขับของแต่ละช่องนั้นๆทำงานโดยถ้าเรา เซตส่วนขาEnable ให้เป็น 0 ช่องขับนั้นจะถูก disable ทันที ดังรูปที่ 3.5

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.4 block diagram การต่อ Mcs51กับ วงจรขับ motor



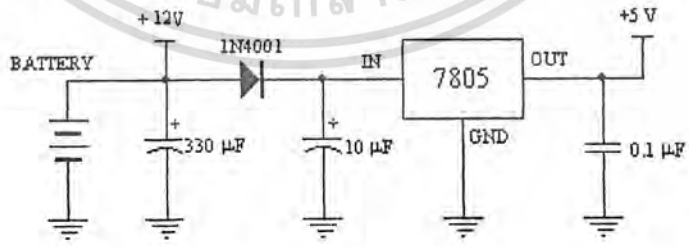
รูปที่ 3.5 วงจรขับมอเตอร์ dc สองตัว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.4 การออกแบบวงจร Microcontroller

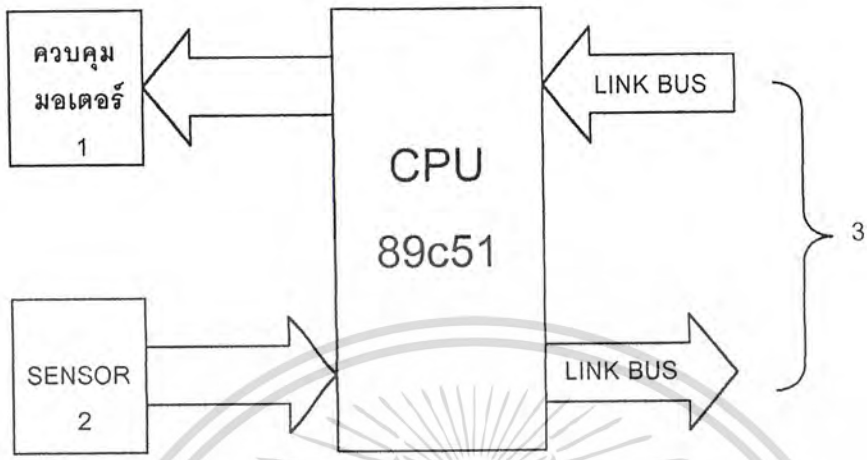
การเชื่อมต่อ Microcontroller Mcs51 เพื่อทำงานควบคุมการเคลื่อนที่ของหุ่นยนต์จะแบ่งส่วนสำคัญที่เชื่อมต่อกันเป็น 3 ส่วนด้วยกันคือ 1 ส่วน ของภาค Sensor, 2 ส่วนของภาคควบคุมการเคลื่อนที่ 3 คือส่วนของการแลกเปลี่ยนข้อมูลของกันและกัน โดยจะออกแบบให้วงจรออกมาในรูปแบบของ พอรต์เทเบิล สามารถนำมาต่อเข้าแยกออกได้ง่ายและเล็กกระทัดรัด และ Mcs51 ไม่จำเป็นที่จะต้อง ใช้หน่วยความจำภายนอกจึงทำให้ประหยัดค่าใช้จ่ายได้มาก Mcs51จะใช้เพียงแต่หน่วยความจำภายในและใช้พอร์ตที่มีอยู่ในการรับส่งค่าเท่านั้น

การใช้งาน Microcontroller (สามารถหาความรู้เพิ่มเติมได้ที่ภาคผนวกและบทที่ 2) การเซตควบคุมเบื้องต้น โดยโครงสร้างของ Microcontroller Mcs51 จะมี port I/P ทั้งหมดอยู่ 4 port คือ P0.0-.7, P1.0-.7, P2.0-.7, P3.0-.7.โดยที่ขา P0.0-.7สามารถเป็นได้ทั้ง อินพุต และ เอาต์พุต ในที่นี้ทางผู้จัดทำได้กำหนดให้เป็น อินพุต โดย เซตค่า port ให้มีค่าเป็น 1 เพื่อทำหน้าที่รับอินพุต ของcontrol ต่างๆจากหุ่นอื่น ต่อมา ขา P1.0-.7 สามารถเป็นได้ทั้ง อินพุต และ เอาต์พุต ในที่นี้ทางผู้จัดทำได้กำหนดให้เป็น เอาต์พุต โดย เซตค่า port ให้มีค่าเป็น 0 เพื่อทำหน้าที่ส่ง control ต่างๆ ต่อมา P2.0-.7สามารถเป็นได้ทั้ง อินพุต และ เอาต์พุต ในที่นี้ทางผู้จัดทำได้กำหนดให้เป็น เอาต์พุต โดย เซตค่า port ให้มีค่าเป็น 0 เพื่อทำหน้าที่ส่งข้อมูลจากการที่ Microcontroller ทำการประมวลและส่งค่าการเคลื่อนที่หรือข้อมูลการควบคุม มอเตอร์ และ P3.0-.7 สามารถเป็นได้ทั้ง อินพุต และ เอาต์พุต ในที่นี้ทางผู้จัดทำได้กำหนดให้เป็น อินพุต โดย เซตค่า port ให้มีค่าเป็น 1 เพื่อทำหน้าที่รับอินพุต ของ Sensor ทั้งซ้ายและขวาเพื่อนำไปประมวลผลต่อไป



รูปที่3.6 แหล่งจ่ายไฟ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.7 ภาพรวมการเชื่อมต่อทั้ง 3 ส่วน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.5 การออกแบบโครงสร้างขา และ ตัวของหุ่นยนต์

วัสดุที่ใช้สร้าง ประกอบด้วย แผ่น พลาสติก ในส่วนของ พลาสติก จะนำมาสร้างเป็น ขา และ ตัวหุ่น เนื่องจากมีคุณสมบัติ ทนทาน น้ำหนักเบา ตกแต่งง่าย และ ราคาถูก ส่วนของ

หุ่นยนต์ที่สร้างมีขนาด 104 mm X 86 mm X 40 mm : 1 ข้อ

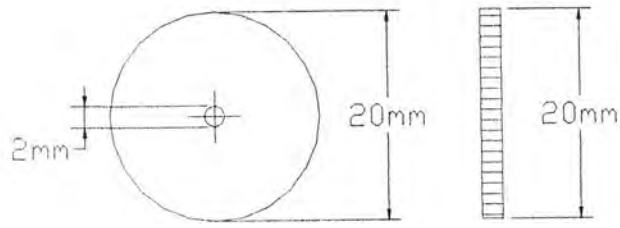
หุ่นยนต์มีน้ำหนัก 500 กรัม : 1 ข้อ

ขาหุ่นยนต์ มีเส้นผ่านศูนย์กลาง 90 mm

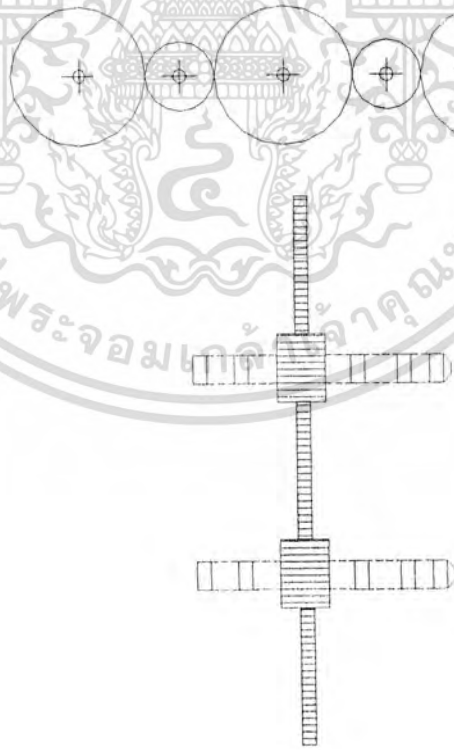
รูปที่ 3.8 ขาที่ใช้งานจริง

จากรูปที่ 3.8 อุปกรณ์ที่ใช้ทำขาคือ อะคริลิค ลักษณะรูปทรงของขานี้ ใช้สำหรับในการปีนสิ่งกีดขวาง และ การพยุงตัวที่ดี

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

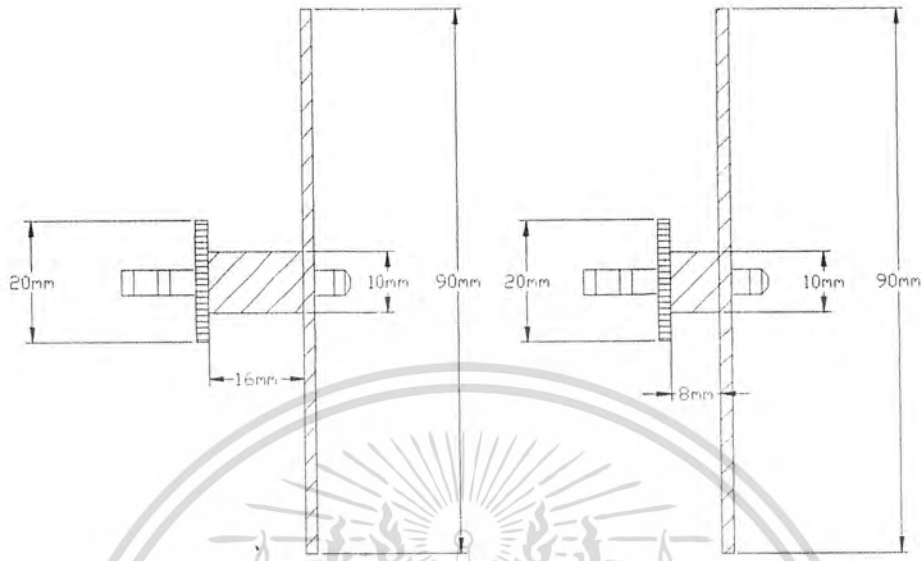


รูปที่ 3.9 ขนาดของเฟืองที่ใช้งาน



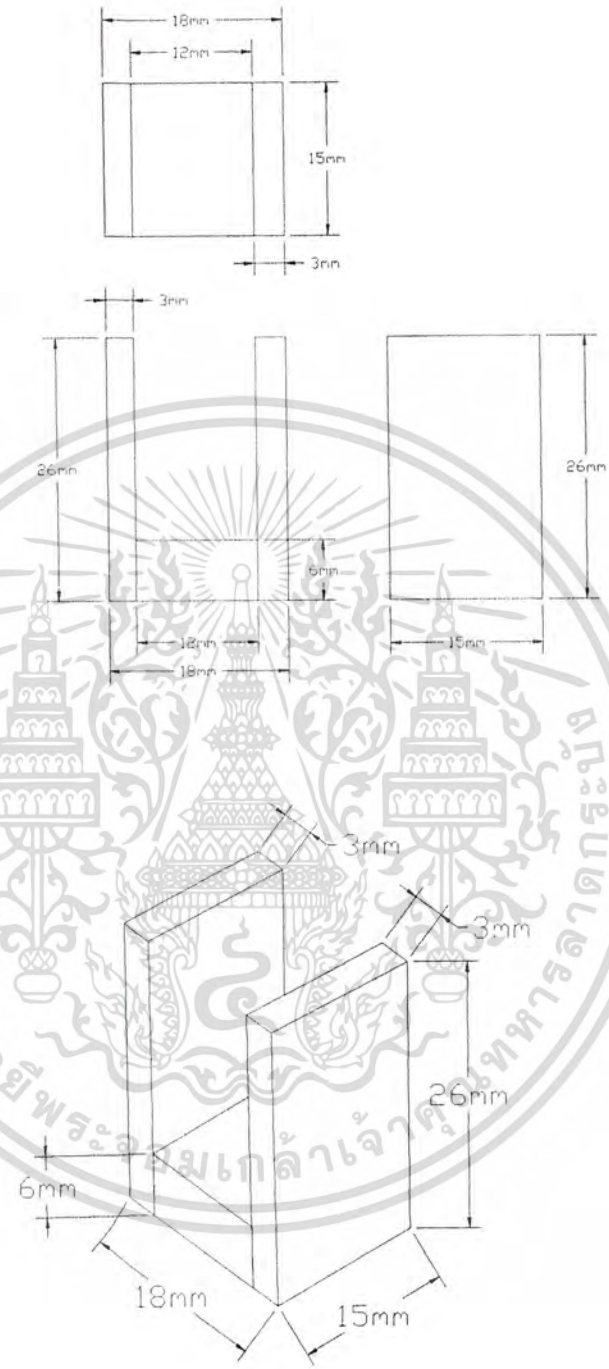
รูปที่ 3.10 การจัดตำแหน่งของเฟือง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



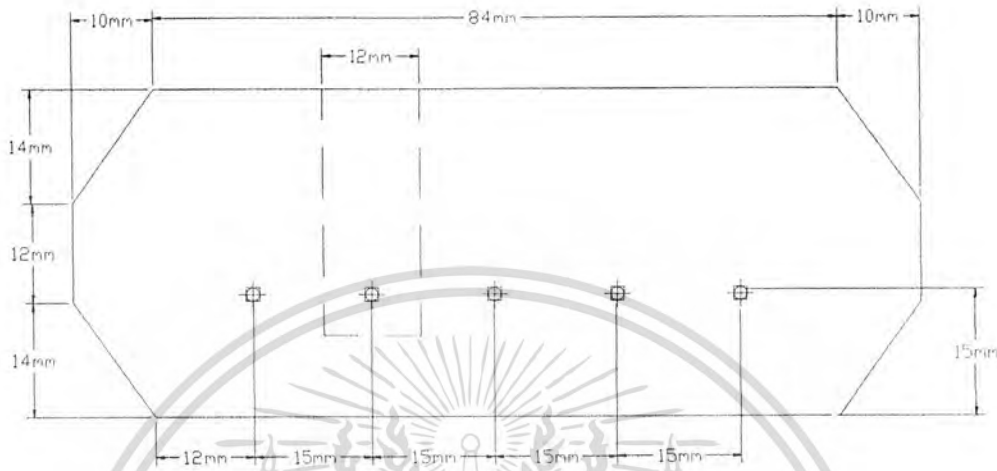
รูปที่ 3.11 การต่อขาที่ใช้งานจริงเข้ากับเฟือง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

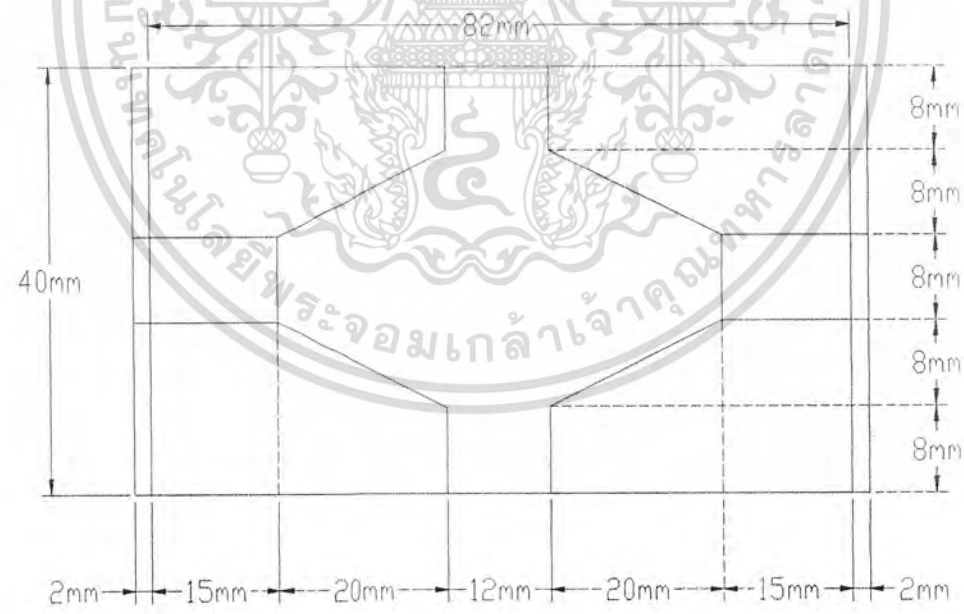


รูปที่ 3.12 ขนาดและลักษณะของตัววาง มอเตอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

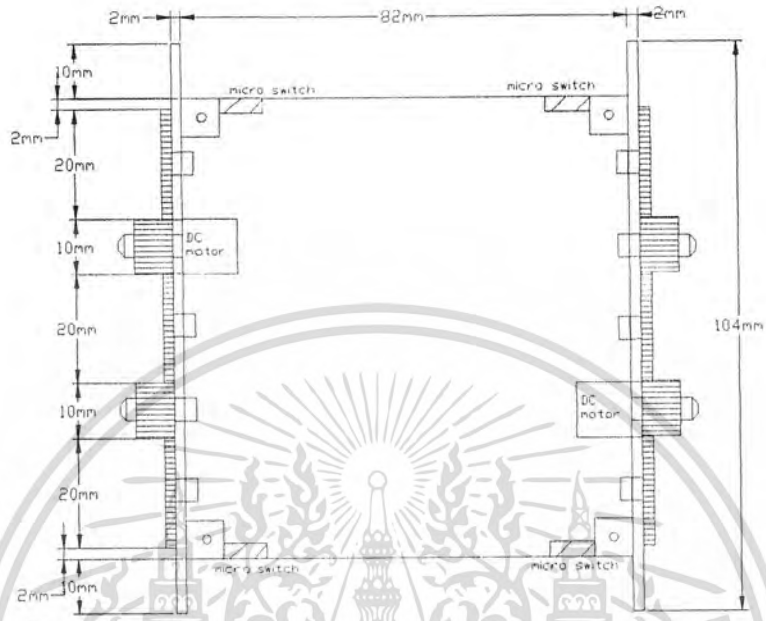


รูปที่ 3.13 ภาพด้านข้าง และ ตำแหน่งเฟืองของตัวหุ่นยนต์

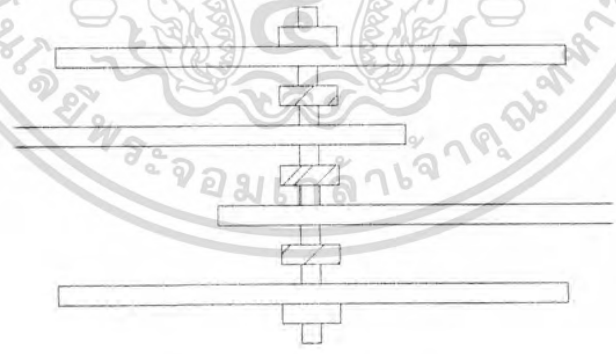


รูปที่ 3.14 ภาพด้านหน้าของตัวหุ่นยนต์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



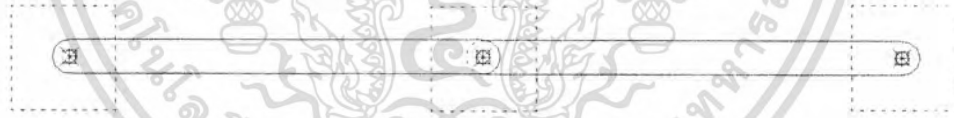
รูปที่ 3.15 ภาพด้านบนพร้อมติดตั้งเฟืองและ มอเตอร์ของตัวหุ่นยนต์



รูปที่ 3.16 ข้อต่อในการเคลื่อนตัวของหุ่น

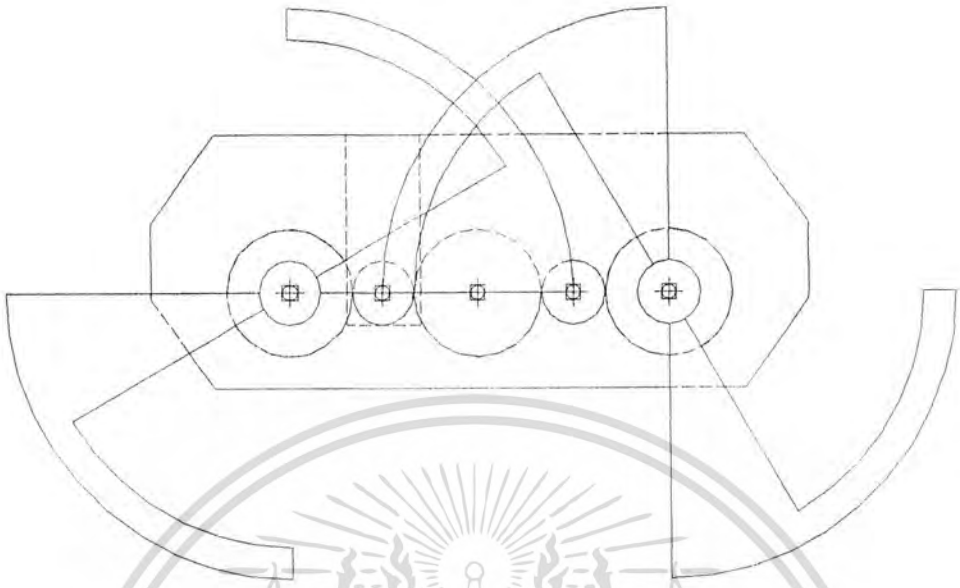
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 3.17 ข้อต่อระหว่างหุ่นแต่ละข้อ

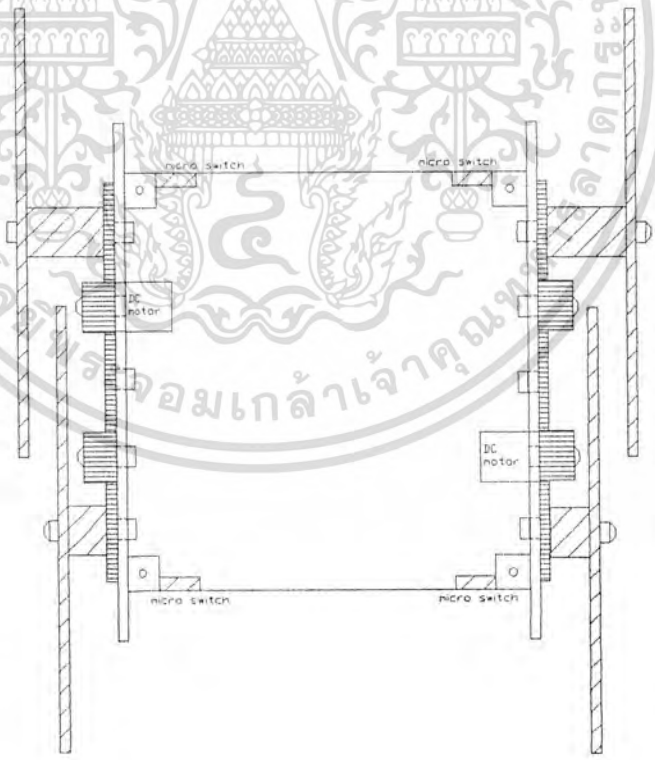


รูปที่ 3.18 ด้านบนของข้อต่อระหว่างหุ่นแต่ละข้อ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

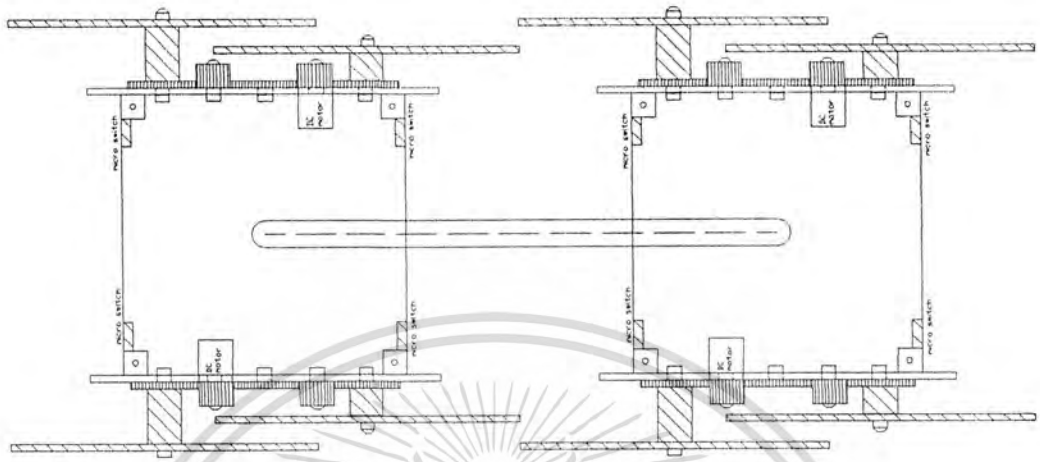


รูปที่ 3.19 ด้านข้างของหุ่นยนต์เมื่อประกอบขา



รูปที่ 3.20 ด้านบนของหุ่นยนต์เมื่อประกอบขา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



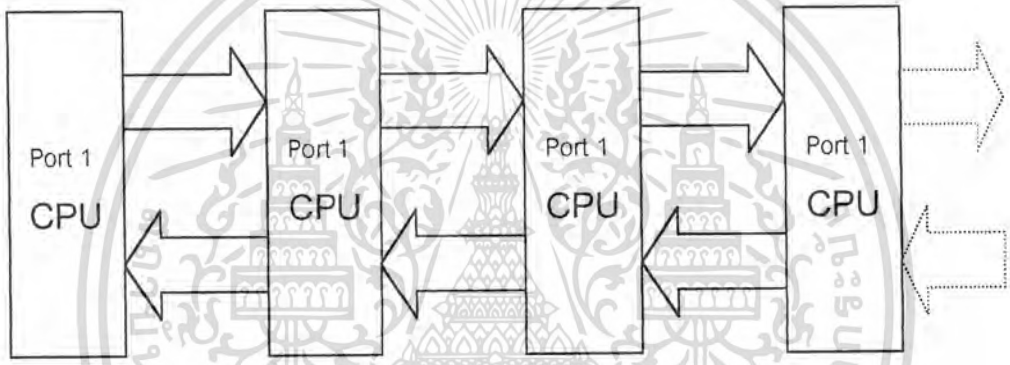
รูปที่ 3.21 ด้านบนของหุ่นยนต์เมื่อต่อกัน 2 ข้อ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 4
การออกแบบโปรแกรมควบคุม

4.1 การออกแบบและโปรแกรมควบคุม

หลักการออกแบบหุ่นตะขาบนี้จะแบ่งเป็นข้อต่อหลายๆข้อรวมกัน แต่ล่ะข้อมีความสามารถแตกต่างกันไปตามแต่ที่เราจะใส่optionเข้าไปแต่จะมีหลักการเดินและการควบคุมที่คล้ายกันมีทิศทางของตัวโปรแกรมไปในทางเดียวกันดังรูปที่4.1,4.2,4.3และ4.4ตามลำดับ



รูปที่ 4.1 การเชื่อมต่อของข้อมูลของกันและกัน

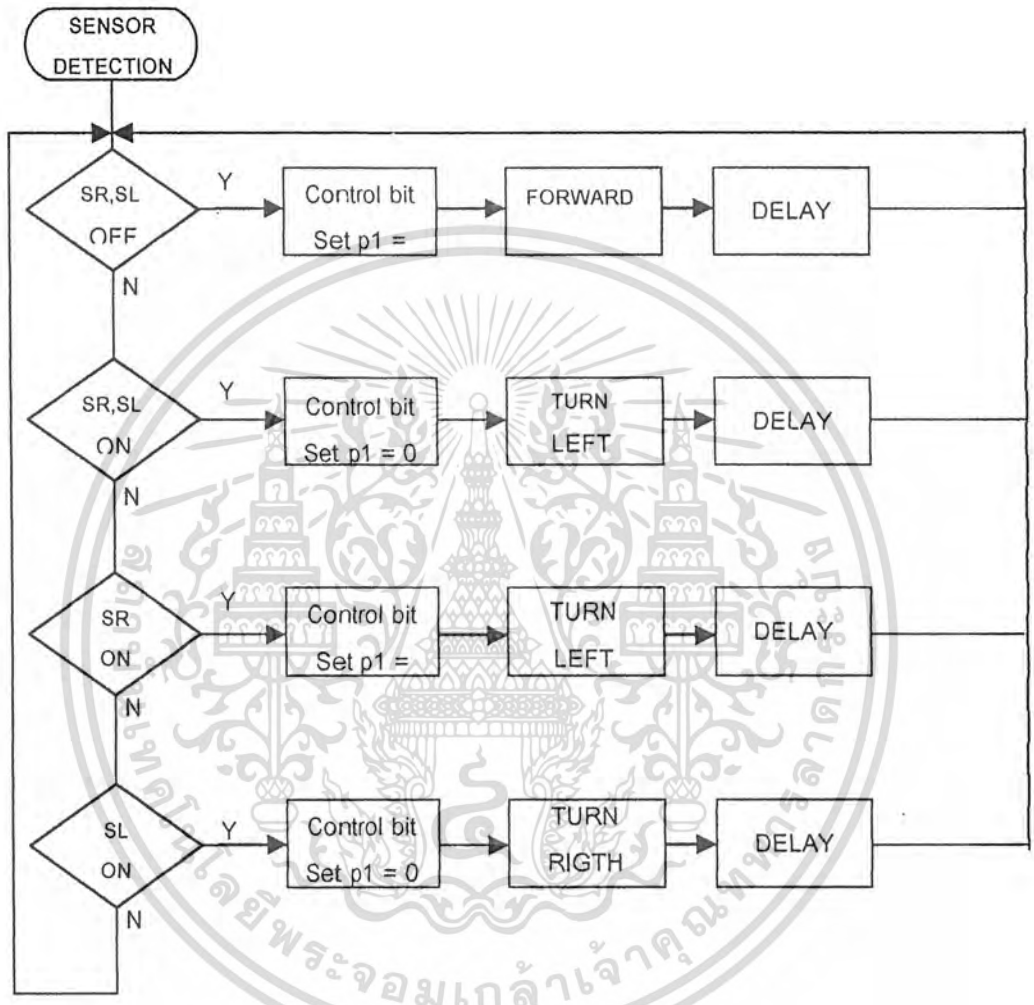
การทำงานของข้อที่ 1

การทำงานของหุ่นยนต์จะแบ่งกันเป็นข้อๆแยกกันโดยใช้mcs51 แยกกันโดยอิสระแต่จะมีการเชื่อมให้มีข้อมูลถึงกันสามารถแลกเปลี่ยนข้อมูลกันได้โดย ใช้การ SET CONTROL BIT (P1) ในการเชื่อมต่อทางด้าน OUTPUT ด้วย PORT 1และทางด้าน INPUTเชื่อมต่อโดย ใช้ PORT 0 รับค่าของ CONTROL BIT (P1) จึงทำให้สามารถแลกเปลี่ยนข้อมูลกันได้ การทำงานของระบบเริ่มที่ภาคของSENSORตรวจสอบสิ่งกีดขวางว่าตรวจสอบพบหรือไม่โดยเมื่อมีการตรวจสอบแล้วจะทำการส่งข้อมูลมายัง microcontroller เมื่อ microcontroller ได้รับข้อจาก SENSOR แล้วจะทำการประมวลผลแล้วทำการสั่งให้หุ่นยนต์ขับเคลื่อนโดยสั่งไปทางมอเตอร์ จาก flow chart ตัว microcontrollerจะเริ่มตรวจสอบข้อมูลที่ได้กับเงื่อนไขที่ได้กำหนดไว้ คือเริ่มแรกจะทำการเปรียบเทียบเงื่อนไขที่ได้จาก SENSOR ทางด้านหน้า คือ SENSORทางซ้าย (ในที่นี้ขอใช้คำแทนด้วยSL)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

และ SENSORทางขวา (ในที่นี้ขอใช้คำแทนด้วย SR)ว่ามีสภาวะ OFFหรือไม่ ถ้าเป็นไปตามเงื่อนไขก็จะทำการ SET CONTROL BIT (P1) ให้มีค่าเป็น 1 (เพื่อนำไปเป็นเงื่อนไขการของการทำงานของหุ่นข้อต่อไป) แล้วจะทำการสั่งการให้หุ่นเดินไปเป็นระยะเวลาด้วย DELAY ค่าหนึ่ง แล้วทำการ วนลูปไปที่การตรวจที่ตำแหน่งเริ่มต้น แต่ถ้าทำการเปรียบแล้วไม่ตรงตามเงื่อนไขก็จะเลื่อนไปตรวจสอบที่เงื่อนไขถัดไป โดยเงื่อนไขถัดไปตาม flow chart จะเป็นการเปรียบเทียบว่ามีการตรวจพบสิ่งกีดขวางทางด้านหน้าใช่หรือไม่ (SR,SL= ON) ถ้าใช่ก็จะทำการ SET CONTROL BIT (P1) ให้มีค่าเป็น 0 (เพื่อนำไปเป็นเงื่อนไขการของการทำงานของหุ่นข้อต่อไป ใน การ set p1 = 0 มีความหมายเพื่อให้ ข้อต่อไปทราบว่ามีการตรวจพบสิ่งกีดขวางและทำการหยุดตัวเองเพื่อให้ตัวแรกหลบสิ่งกีดขวาง) แล้วเมื่อ SET แล้วจะสั่งให้หุ่นหยุดและหมุนตัวหลบไปทางซ้ายด้วยระยะเวลา DELAY ค่าหนึ่งและทำการวนลูปกลับไปที่ตำแหน่งเริ่มต้นของโปรแกรม แต่ถ้าทำการเปรียบแล้วไม่ตรงตามเงื่อนไขก็จะเลื่อนไปตรวจสอบที่เงื่อนไขถัดไป โดยเงื่อนไขถัดไปตาม flow chart จะเป็นการเปรียบเทียบว่ามีการตรวจพบสิ่งกีดขวางทางด้านขวาใช่หรือไม่ (SR= ON) ถ้าใช่ก็จะทำการ SET CONTROL BIT (P1) ให้มีค่าเป็น 0 แล้วจะสั่งให้หุ่นหยุดและหมุนตัวหลบไปทางซ้ายด้วยระยะเวลา DELAY ค่าหนึ่งและทำการวนลูปกลับไปที่ตำแหน่งเริ่มต้นของโปรแกรม แต่ถ้าทำการเปรียบแล้วไม่ตรงตามเงื่อนไขก็จะเลื่อนไปตรวจสอบที่เงื่อนไขถัดไป โดยเงื่อนไขถัดไปจะเป็นการเปรียบเทียบว่ามีการตรวจพบสิ่งกีดขวางทางด้านซ้ายใช่หรือไม่ (SL= ON) ถ้าใช่ก็จะทำการ SET CONTROL BIT (P1) ให้มีค่าเป็น 0 แล้วจะสั่งให้หุ่นหยุดและหมุนตัวหลบไปทางขวาด้วยระยะเวลา DELAY ค่าหนึ่งและทำการวนลูปกลับไปที่ตำแหน่งเริ่มต้นของโปรแกรม แต่ถ้าทำการเปรียบแล้วไม่ตรงตามเงื่อนไขก็จะทำการวนลูปกลับไปที่ตำแหน่งเริ่มต้นของโปรแกรมเลย เพราะเป็นสิ้นสุดการเปรียบเงื่อนไขแล้วและจะทำงานตามflow chart แบบนี้ไปเรื่อย ตาม (รูปที่ 4.2) โดยที่ค่าของCONTROL BIT (P1) จะถูกนำไปเป็นเงื่อนไขการทำงานของหุ่นตัวต่อไป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.2 FLOW CHARTแสดงการควบคุมหุ่นยนต์(ตัวที่ 1)

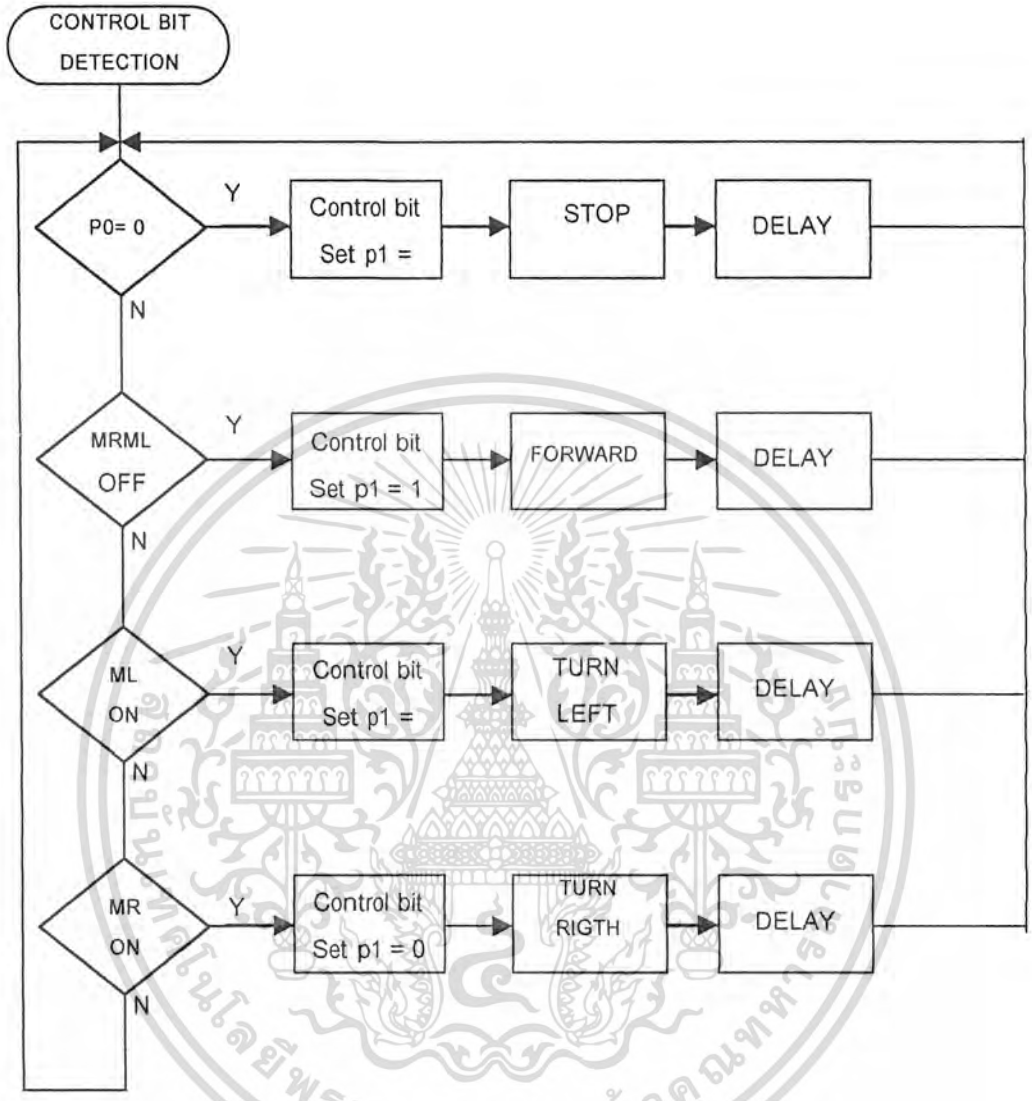
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การทำงานของข้อที่ 2 และข้อต่อๆไป

การทำงานของหุ่นตัวที่สองจะมีลักษณะคล้ายคลึงกันกับตัวแรกแต่จะตรวจสอบเงื่อนไขจากข้อมูลที่แตกต่างกันคือตรวจสอบจาก CONTROL BIT (P1) และ Micro switchที่จะตรวจสอบจะหน้าสัมผัสกับแกนหมุนของข้อต่อการทำงานเริ่มต้นด้วย การตรวจสอบค่าจาก CONTROL BIT (P1)ว่ามีค่าเป็น 0 หรือเป็น 1โดยถ้า เป็น 0จะทำการหยุดตัวเองอยู่กับที่เป็นค่าDEAYค่าหนึ่งที่มีค่าเท่ากันกับ DELAYของ ตัวแรก และจะทำการ SET CONTROL BIT (P1) ของตัวเองให้มีค่าเป็น 0 (เพื่อนำไปเป็นเงื่อนไขการของการทำงานของหุ่นข้อต่อไป) และ วนลูปกลับไปที่ตำแหน่งเริ่มต้นเพื่อตรวจสอบว่าในขณะนั้นตัวแรกได้หลบสิ่งกีดขวางพ้นแล้วหรือไม่โดยการตรวจที่ค่าของ CONTROL BIT (P1)และจะทำไปเรื่อยๆจนกว่าค่าของ CONTROL BIT (P1)จะมีค่าเป็น 1 แสดงว่าพ้นแล้วและจะทำการเลื่อนไปเปรียบเงื่อนไขตัวต่อไป โดยเปรียบเทียบเงื่อนไขจากข้อมูลของ Micro switchที่จะตรวจสอบจะหน้าสัมผัสกับแกนหมุนของข้อต่อโดยเงื่อนไขแรกคือตรวจดู ว่า Micro switch ที่หน้าสัมผัสซ้าย(ในที่นี้ขอแทนด้วย ML)และว่า Micro switch ที่หน้าสัมผัสขวา(ในที่นี้ขอแทนด้วย MR) โดยเปรียบเทียบว่า MR และ ML มีค่าเป็น OFF ใช่หรือไม่ ถ้าใช่ ก็จะสั่งให้หุ่นเดินตรงตามตัวหน้าปกติ และจะทำการ SET CONTROL BIT (P1) ของตัวเองให้มีค่าเป็น 1 และจะทำการวนลูปกลับไปที่ตำแหน่งเริ่มต้น(เพื่อตรวจสอบว่าในขณะนั้นตัวแรกได้ส่งค่า CONTROL BIT (P1) มาเป็นอะไร พบสิ่งกีดขวางหรือไม่)แต่ถ้าทำการเปรียบแล้วไม่ตรงตามเงื่อนไขก็จะเลื่อนไปตรวจสอบที่เงื่อนไขถัดไป โดยเงื่อนไขถัดไปจะเป็นการเปรียบเทียบว่า Micro switch ที่หน้าสัมผัสขวา(MR)อยู่ในสภาวะ ON ใช่หรือไม่

ถ้าใช่ ทำการ SET CONTROL BIT (P1) ของตัวเองให้มีค่าเป็น 0 และจะสั่งให้หุ่นหยุดเดินและหมุนตัวไปตามหน้าสัมผัสคือ หมุนไปทางขวา และจะทำการวนลูปกลับไปที่ตำแหน่งเริ่มต้น(เพื่อตรวจสอบว่าในขณะนั้นตัวแรกได้ส่งค่า CONTROL BIT (P1) มาเป็นอะไร) แต่ถ้าทำการเปรียบแล้วไม่ตรงตามเงื่อนไขก็จะเลื่อนไปตรวจสอบที่เงื่อนไขถัดไป โดยเงื่อนไขถัดไปจะเป็นการเปรียบเทียบว่า Micro switch ที่หน้าสัมผัสขวา(ML)อยู่ในสภาวะ ON ใช่หรือไม่ ถ้าใช่ ทำการ SET CONTROL BIT (P1) ของตัวเองให้มีค่าเป็น 0 และจะสั่งให้หุ่นหยุดเดินและหมุนตัวไปตามหน้าสัมผัสคือ หมุนไปทางซ้าย และจะทำการวนลูปกลับไปที่ตำแหน่งเริ่มต้น เพื่อตรวจสอบ ค่า CONTROL BIT (P1) เป็นอันสิ้นสุดเงื่อนไข(รูปที่ 4.3)

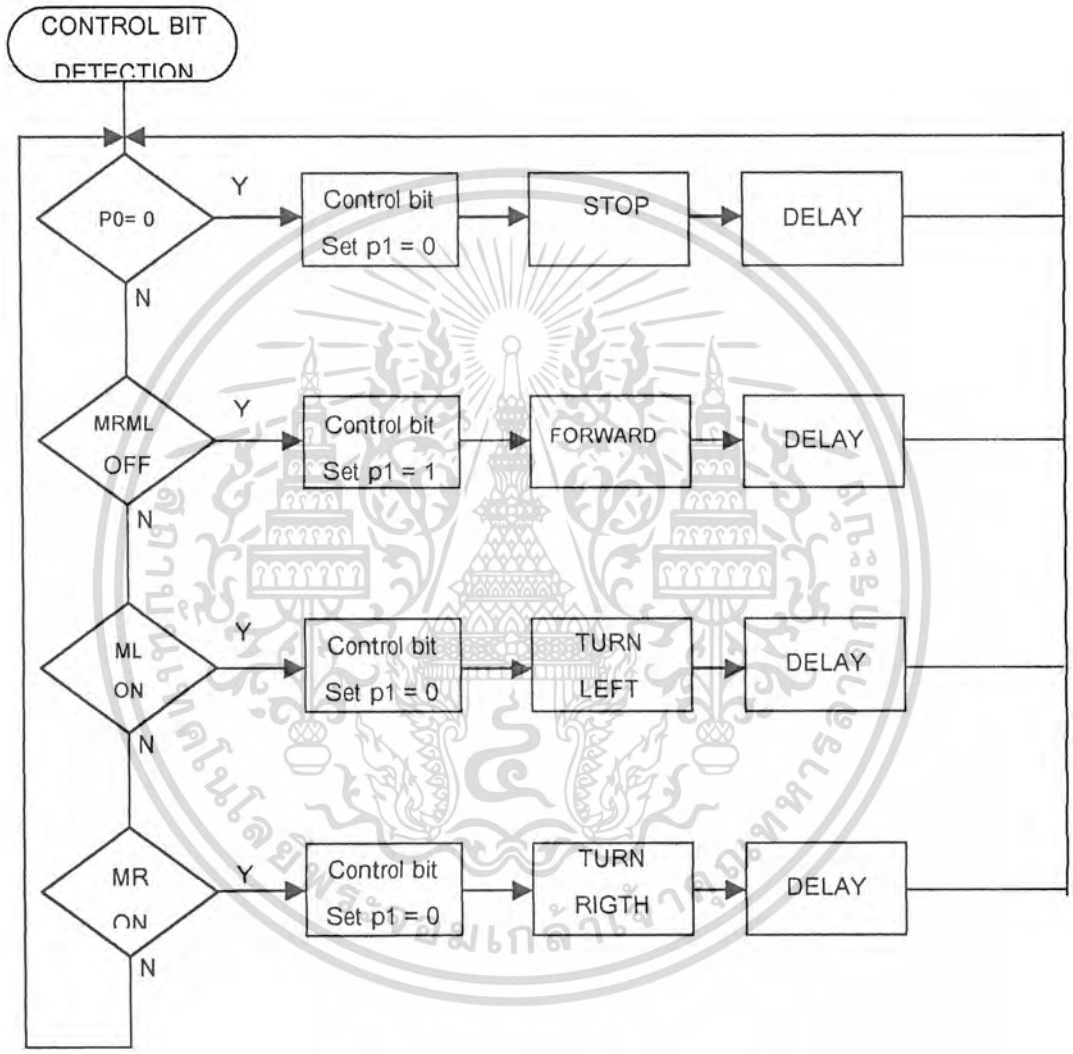
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.3 FLOW CHARTแสดงการควบคุมหุ่นยนต์(ตัวที่ 2)

โดยการทำงานของข้อต่อๆไปจะเหมือนกันกับตัวที่สอง(รูปที่ 4.4)จึงทำให้การทำงานของข้อต่อแต่ละข้อมีความสัมพันธ์กัน โดยข้อที่สำคัญคือการใช้DELAYนั้นจะต้องมีค่าที่เท่ากันจึงจะทำให้เกิดสัมพันธ์กันทางกายภาพจะทำไม่เกิดการนำหน้าหรือล่าหลังกันจนเกิดแรงเสียดทานทำให้ข้อต่อเสียหายหรือ MOTOR ชำรุดได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



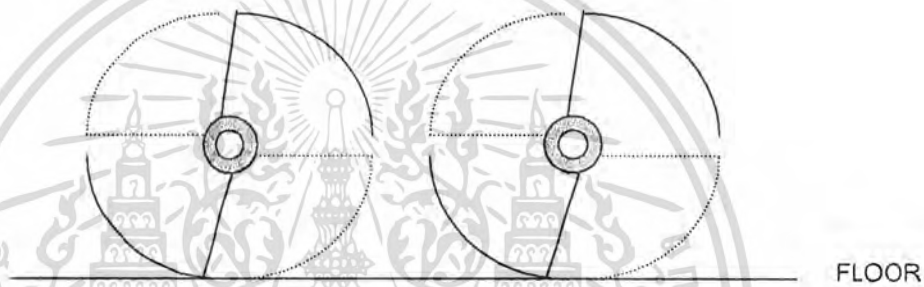
รูปที่ 4.4 FLOW CHARTแสดงการควบคุมหุ่นยนต์(ตัวที่ 3)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 5
การทดลอง

5.1 การทดลองรูปแบบของการเดิน

จากการทดลองให้หุ่นยนต์เคลื่อนที่ โดยทางผู้จัดทำได้ทดลองวางตำแหน่งขาแต่ละขามีมุมที่ต่างกันอยู่ 120 องศา ดังรูปที่ 5.1

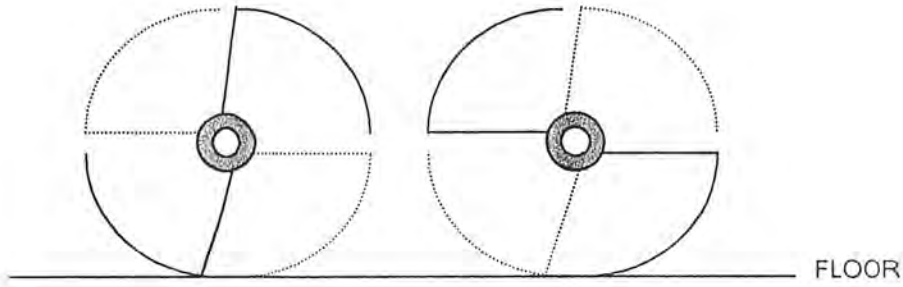


รูปที่ 5.1 การวางขา 120 องศา

จากการทดลองให้หุ่นยนต์เคลื่อนที่ โดยการทดลองจะทำการทดลองการเคลื่อนที่เพียงข้อเดียวเพื่อความสะดวกในการทดสอบ จะเป็นการทดลองการเดินว่ามีสถานะภาพการเดินเป็นอย่างไร โดยจากการทดลอง จะพบว่าหุ่นยนต์เดินได้ไม่ดีเท่าที่ควรเพราะ การทดแทนกันของแต่ละจุดที่สัมผัสกับพื้นไม่สม่ำเสมอจึงเกิดการเสียสมดุลย์ขณะที่หุ่นยนต์เคลื่อนที่ และในขณะการหมุนตัวยิ่งไม่มีความสมดุลย์เลย

ต่อมาได้ทำการทดลองปรับเปลี่ยนตำแหน่งของขาใหม่ โดยที่เปลี่ยนจากมุมที่ต่างกันอยู่ 120 องศาเป็นต่างกันอยู่ 90 องศา ทั้งหน้าและหลัง ดังรูปที่ 5.2 นั้น จะให้ที่ดีกว่าเดิมที่ใช้ 180 องศา จะให้การทรงตัวที่ดีในขณะที่หุ่นยนต์เดินตรงและเลี้ยวเพราะทุกสแต็ปของการเดินจะมีจุดที่ขาของหุ่นยนต์สัมผัสกับพื้นอยู่ทั้งหมด 4 จุดและขณะที่หุ่นยนต์ทำการหมุนตัวนั้นจะใช้พื้นที่ในการหมุนตัวเองของหุ่นแต่ละข้อน้อยมาก คือขณะที่หุ่นยนต์หมุนตัวนั้น ตัวหุ่นยนต์จะหมุนตัวอยู่กับที่

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 5.2 การวางขา 90 องศา

5.2 การทดลองความเร็วในการเดิน (หุ่นยนต์ 1 ตัว)

การทดลองการจับเวลาในการจับเวลาในการเคลื่อนที่ของหุ่นยนต์ ทางผู้จัดทำได้ทำการทดลองโดยการป้อนแรงดันที่ระดับต่างๆดังนี้

ป้อนไฟ Vss ให้ มอเตอร์ 6 โวลท์

- เดินหน้า ระยะทาง 0.5 เมตร ใช้เวลาประมาณ 22 วินาที
- หมุนตัวไปทางซ้าย 45 องศา ใช้เวลาประมาณ 6 วินาที
- หมุนตัวไปทางขวา 45 องศา ใช้เวลาประมาณ 6 วินาที

ป้อนไฟ Vss ให้ มอเตอร์ 9 โวลท์

- เดินหน้า ระยะทาง 0.5 เมตร ใช้เวลาประมาณ 12 วินาที
- หมุนตัวไปทางซ้าย 45 องศา ใช้เวลาประมาณ 2.5 วินาที
- หมุนตัวไปทางขวา 45 องศา ใช้เวลาประมาณ 2.5 วินาที

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ป้อนไฟ Vss ให้ มอเตอร์ 12 โวลท์

- เดินหน้า ระยะทาง 0.5 เมตร ใช้เวลาประมาณ 6 วินาที
- หมุนตัวไปทางซ้าย 45 องศา ใช้เวลาประมาณ 2 วินาที
- หมุนตัวไปทางขวา 45 องศา ใช้เวลาประมาณ 2 วินาที

5.3 การทดลองความเร็วในการเดิน (หุ่นยนต์ ต่อเชื่อมกันหมด 8 ตัว)

การทดลองการจับเวลาในการจับเวลาในการเคลื่อนที่ของหุ่นยนต์ ทางผู้จัดทำได้ทำการทดลองโดยการป้อนแรงดันเหมือนเดิมแต่จะทำการเชื่อมต่อ 8 ตัวเข้าด้วยกัน โดยใน step การเดินจะมีการบวกเวลาของดีเลย์การรอคอยให้มีการหมุนหลบหลีกสิ่งกีดขวางโดยเมื่อมีการหมุนที่ข้อใด ตัวที่เหลือก็จะ ทำการ หยุดรอคอยจนกว่าจะไม่มีการหมุนของข้อใดเลย

ป้อนไฟ Vss ให้ มอเตอร์ 6 โวลท์ (8 step)

- เดินหน้า ระยะทาง 0.5 เมตร ใช้เวลาประมาณ 22 วินาที
- หมุนตัวไปทางซ้าย 45 องศา ใช้เวลาประมาณ 6 วินาที
  หุ่นตัวอื่น (7 ตัว) หยุดรอ ใช้เวลาประมาณ 6 วินาที
  ทั้งหมด 8 step เท่ากับ หยุดรอ 8 ครั้ง หมุนตัว 8 ครั้ง
  เวลารวมจะใช้เวลาประมาณ 1 นาที 36 วินาที
- หมุนตัวไปทางขวา 45 องศา ใช้เวลาประมาณ 6 วินาที
  หุ่นตัวอื่น (7 ตัว) หยุดรอ ใช้เวลาประมาณ 6 วินาที
  ทั้งหมด 8 step เท่ากับ หยุดรอ 8 ครั้ง หมุนตัว 8 ครั้ง
  เวลารวมจะใช้เวลาประมาณ 1 นาที 36 วินาที

ป้อนไฟ Vss ให้ มอเตอร์ 9 โวลท์ (8 step)

- เดินหน้า ระยะทาง 0.5 เมตร ใช้เวลาประมาณ 12 วินาที
- หมุนตัวไปทางซ้าย 45 องศา ใช้เวลาประมาณ 2.5 วินาที
  หุ่นตัวอื่น (7 ตัว) หยุดรอ ใช้เวลาประมาณ 2.5 วินาที
  ทั้งหมด 8 step เท่ากับ หยุดรอ 8 ครั้ง หมุนตัว 8 ครั้ง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เวลารวมจะใช้เวลาประมาณ 40 วินาที

- หมุนตัวไปทางขวา 45 องศา ใช้เวลาประมาณ 2.5 วินาที

หุ่นตัวอื่น (7 ตัว) หยุดรอ ใช้เวลาประมาณ 2.5 วินาที

ทั้งหมด 8 step เท่ากับ หยุดรอ 8 ครั้ง หมุนตัว 8 ครั้ง

เวลารวมจะใช้เวลาประมาณ 40 วินาที

ป้อนไฟ Vss ให้ มอเตอร์ 12 โวลท์ (8 step)

- เดินหน้า ระยะทาง 0.5 เมตร ใช้เวลาประมาณ 6 วินาที
- หมุนตัวไปทางซ้าย 45 องศา ใช้เวลาประมาณ 2 วินาที

หุ่นตัวอื่น (7 ตัว) หยุดรอ ใช้เวลาประมาณ 2 วินาที

ทั้งหมด 8 step เท่ากับ หยุดรอ 8 ครั้ง หมุนตัว 8 ครั้ง

เวลารวมจะใช้เวลาประมาณ 32 วินาที

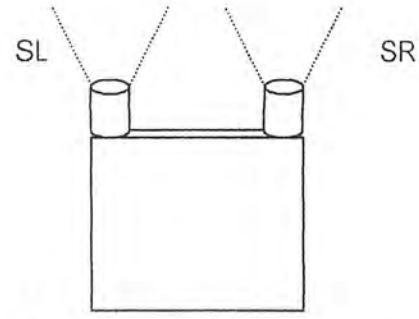
- หมุนตัวไปทางขวา 45 องศา ใช้เวลาประมาณ 2 วินาที

หุ่นตัวอื่น (7 ตัว) หยุดรอ ใช้เวลาประมาณ 2 วินาที

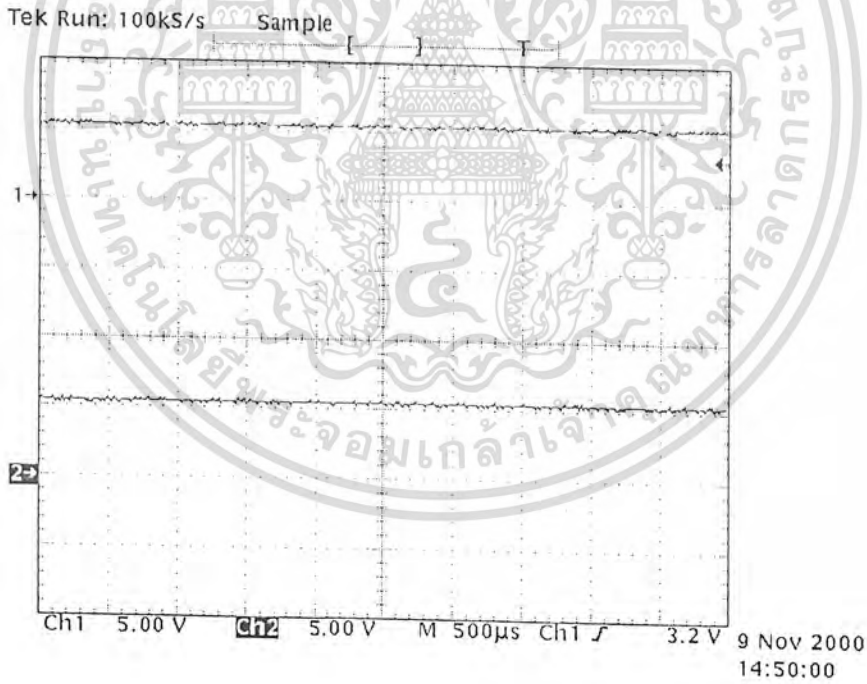
ทั้งหมด 8 step เท่ากับ หยุดรอ 8 ครั้ง หมุนตัว 8 ครั้ง

เวลารวมจะใช้เวลาประมาณ 32 วินาที

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

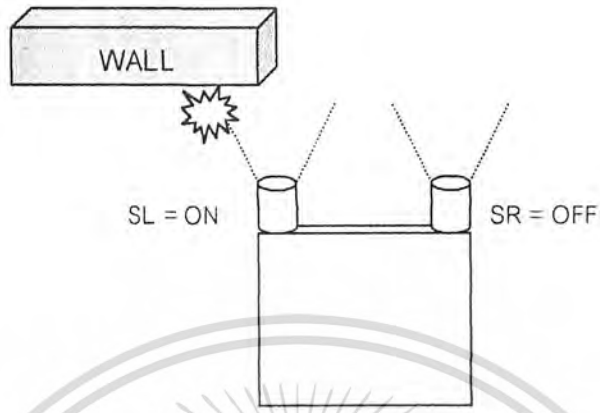


รูปที่ 5.3 การวางตำแหน่งของSENSOR

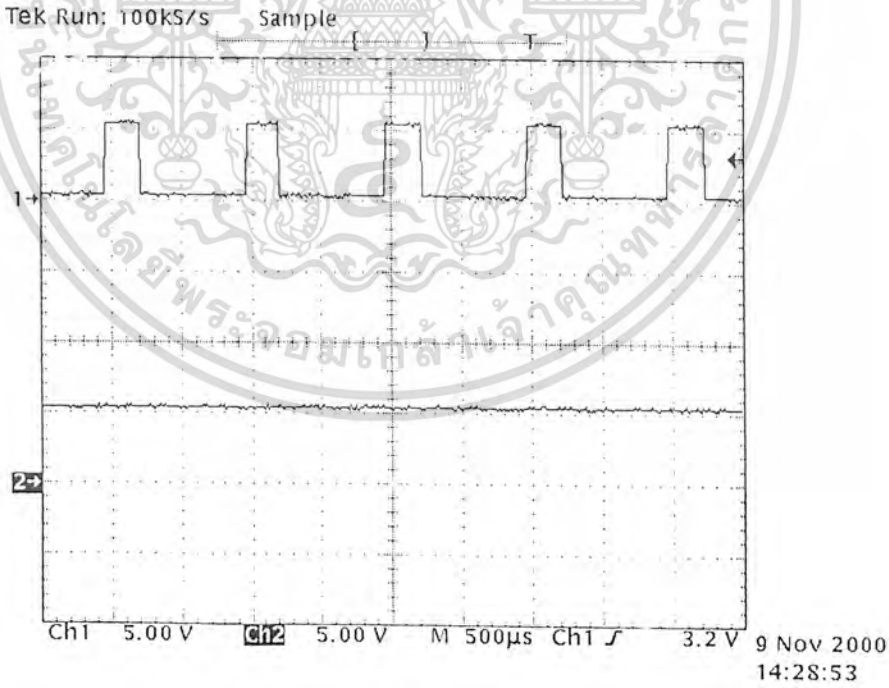


รูปที่ 5.4 สัญญาณ PWM ที่เกิดขึ้น เมื่อ SENSOR ตรวจไม่พบสิ่งกีดขวาง(SR,SL = OFF)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

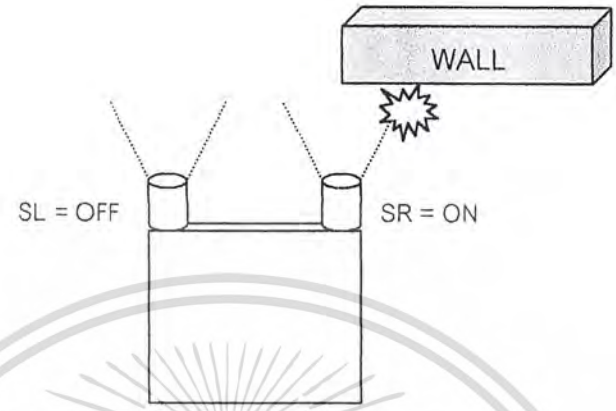


รูปที่ 5.5 การตรวจพบสิ่งกีดขวางทางซ้ายของSENSOR

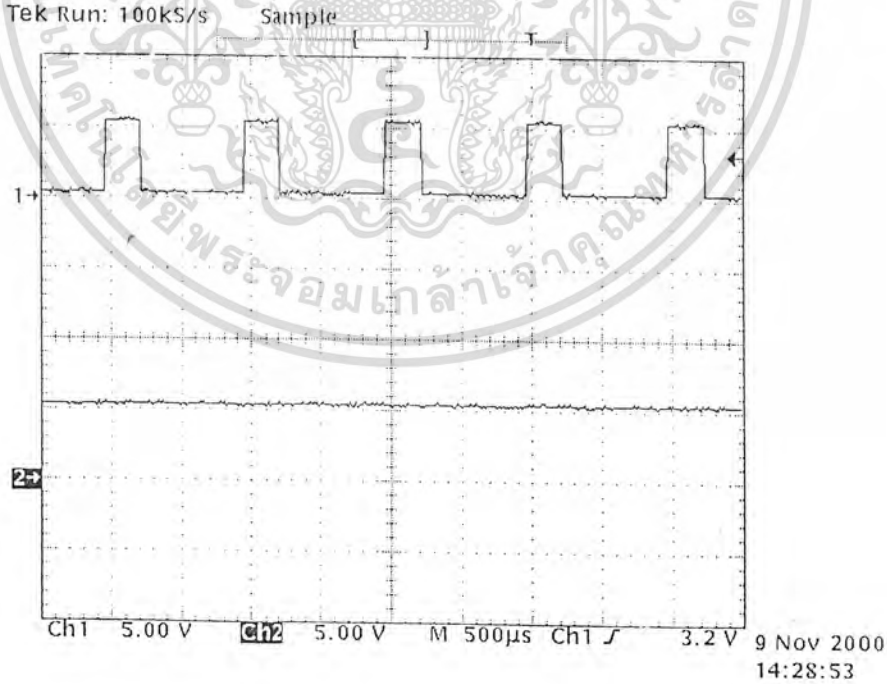


รูปที่ 5.6 สัญญาณ PWM ที่เกิดขึ้น เมื่อ SENSOR ตรวจพบสิ่งกีดขวาง(SL = ON)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

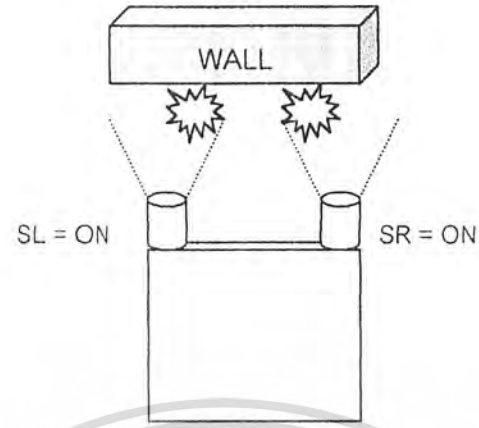


รูปที่ 5.7 การตรวจพบสิ่งกีดขวางทางขวาของSENSOR

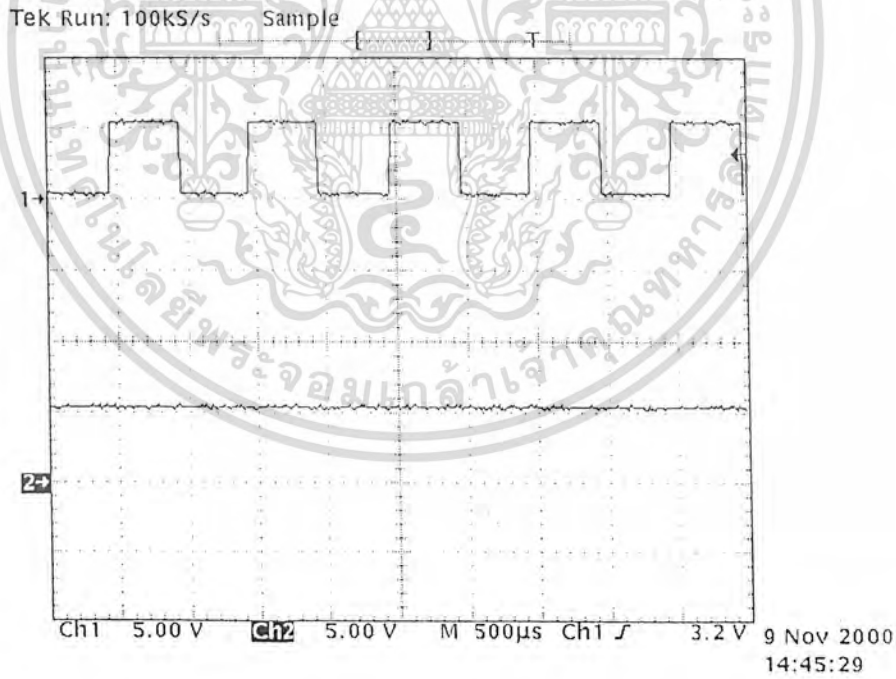


รูปที่ 5.8 สัญญาณ PWM ที่เกิดขึ้น เมื่อ SENSOR ตรวจพบสิ่งกีดขวาง(SR = ON)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

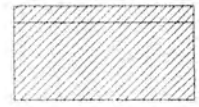


รูปที่ 5.9 การตรวจพบสิ่งกีดขวางทางด้านหน้าของSENSOR



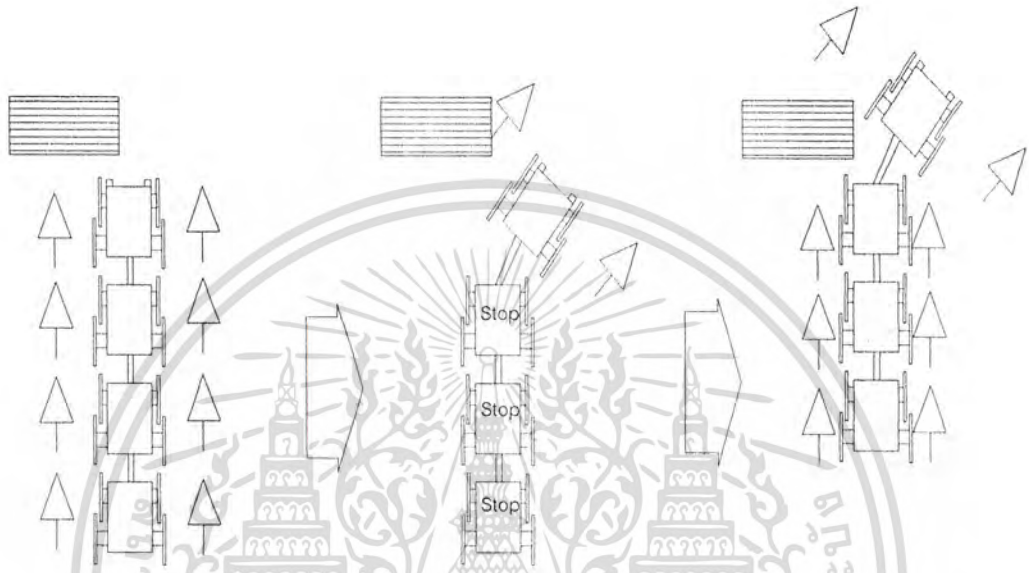
รูปที่ 5.10 สัญญาณ PWM ที่เกิดขึ้น เมื่อ SENSOR ตรวจพบสิ่งกีดขวาง(SL,SR = ON)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 5.11 การเคลื่อนที่ไปข้างหน้าของหุ่นยนต์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 5.12 ลักษณะการเลี้ยวของหุ่นยนต์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 6
สรุปโครงงานและข้อเสนอแนะ

6.1 สรุปโครงงาน

จากความต้องการเบื้องต้นของผู้จัดทำนั้น ต้องการสร้างหุ่นยนต์ที่สามารถที่ได้และมีลักษณะคล้ายตะขาบโดยมีการเชื่อมต่อกันเป็นข้อๆคล้ายตะขาบและพยายามที่จะสร้างให้มีขนาดเล็กที่สุด แต่เนื่องจากวัสดุอุปกรณ์ที่ต้องการนั้นไม่สามารถจัดหาได้หรือไม่ก็มีราคาที่แพงมากไม่ตรงกับจุดประสงค์ของการจัดทำหุ่นยนต์นี้ขึ้น โดยเจตนาของคณะผู้จัดทำที่ต้องการคือเป็นหุ่นยนต์ที่มีราคาถูกและสามารถหาซื้อวัสดุมาสร้างโดยง่าย

หุ่นยนต์ตะขาบที่สร้างขึ้นมีลักษณะที่ต่อการไปทั้งหมด 8 ตัว โดยที่สามารถแลกเปลี่ยนข้อมูลซึ่งกันและกันการสร้างขาจะเป็นแบบขาคร่ึงวงกลมสลับทิศกัน หุ่นหนึ่งตัวจะใช้ ดีซีมอเตอร์ 2 ตัว วาง ไขว้กันเพื่อความสมดุลย์ของหุ่นยนต์ และ มอเตอร์ทั้งหมด ใช้ 16 ตัว ทั้งหมดนี้ใช้ Microcontroller Mcs51 ควบคุมทั้งหมดมีการเคลื่อนที่โดยมีการเคลื่อนเป็นเงื่อนไขของกันและกันจึงเกิดความสัมพันธ์ทางกายภาพ ใช้แหล่งพลังงานสองแหล่ง คือ แบตเตอรี่ 9 โวลท์ และ 12 โวลท์ มีระบบตรวจจับสิ่งกีดขวางเป็นชนิด อินฟาเรด โดยมีระยะตรวจจับ 5 นิ้ว การเชื่อมกันใช้แกนหมุนที่ทำมาจาก PVC และ ยาง

จากการทดสอบการทำงานของโปรแกรมกับตัวฮาร์ดแวร์ แล้วผลออกมาเป็นที่น่าพอใจในระดับหนึ่งคือสามารถเดินหน้าได้หลบหลีกสิ่งกีดขวางได้ในพื้นที่ราบและขรุขระ สามารถนำวงจรต่างๆมาแยกวางลงบนหุ่นเพื่อนำไปใช้ทำประโยชน์ต่างๆได้

6.2 ปัญหาในการทำโครงงาน

6.2.1 ปัญหาในการสร้างตัวหุ่นยนต์

- ทางผู้จัดทำขาดความรู้ความเข้าใจทางด้านเครื่องกล จึงไม่สามารถแก้ปัญหา ที่เกิดขึ้นได้ทั้งหมดและไม่มีความชำนาญในการสร้างชิ้นส่วนต่างๆของหุ่นยนต์จึงทำให้ใช้เวลานานในการสร้าง
- ของที่หาส่วนมากเป็นที่ใช้แล้วหรือของมือสองจึงไม่สามารถสั่งหรือซื้อได้ตามต้องการจึงต้องมีการรอของมาขายทำให้เสียเวลามากขึ้น
- เกิดการบิดไปมาของข้อต่อจึงทำให้เกิดการเสียหายและการใช้ไฟในการทดสอบ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

6.2.2 ปัญหาในการควบคุม

- ความเร็วของมอเตอร์ไม่เท่ากันเป็นการยากในการควบคุมให้เดินโดยไม่บิดไปมาได้
- มีการตรวจสอบเงื่อนไขที่มากมายทำให้ยากต่อการออกแบบโปรแกรมและมีการเคลื่อนที่ที่ช้าในเวลาหลบสิ่งกีดขวาง
- มีการใช้ พลังงานจาก แบตอรี่ค่อนข้างสูง

6.3 ข้อเสนอแนะในการพัฒนาต่อไป

ข้อเสนอแนะนี้จัดทำขึ้นเพื่อเป็นแนวทางในการพัฒนา หุ่นยนต์ตะขาบให้สามารถนำไปพัฒนาให้มีประสิทธิภาพ โดยจะนำเสนอเป็นข้อๆดังนี้

- เพิ่มระบบต่างๆเพื่อนำหลักการของ line process มาใช้กับ หุ่นยนต์ตะขาบ
- หาแหล่งพลังงานใหม่ที่ดีกว่าและทนทานต่อความร้อน
- พัฒนาโครงสร้างให้มีความเสถียรภาพมากกว่าที่เป็นอยู่
- มีการควบคุมระยะใกล้ ไกล
- ทำให้มีขนาดเล็กลงกว่าเดิมที่เป็นอยู่
- เพิ่มความสามารถในการติดต่อสื่อสารกับคอมพิวเตอร์โดยตรง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก ก.

โปรแกรมควบคุมและวงจร

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        ORG     00H
        LJMP    START

        ORG     03H
        LJMP    INT_0

        ORG     13H
        LJMP    INT_1

        ORG     40H

START:
M_GO    EQU     11001010B
M_STOP  EQU     11100000B
M_R     EQU     11000110B
M_L     EQU     11001001B

ON      EQU     11111111B
OFF     EQU     11110011B

        MOV     P0,#0FFH
        MOV     P1,#00H
        MOV     P2,#00H
        MOV     P3,#0FFH

MAIN:   SETB    EA
        CLR     IT0
        CLR     IT1
        SETB    EX0
        SETB    EX1
        MOV     A,P0
        MOV     B,A
        ANL     A,#00000011B
        CJNE    A,#00000011B,SW_R1
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        MOV     P1,#ON
        LCALL   GO
        LJMP    MAIN

SW_R1:  MOV     A,P0
        ANL     A,#00000011B
        CJNE    A,#00000010B,SW_L1
        MOV     P1,#OFF
        LCALL   STOP
        LCALL   LL
        LJMP    MAIN

SW_L1:  MOV     A,B
        ANL     A,#00000011B
        CJNE    A,#00000001B,SW_R2
        MOV     P1,#OFF
        LCALL   STOP
        LCALL   RR
        LJMP    MAIN

SW_L2:  MOV     A,B
        ANL     A,#00110000B
        CJNE    A,#00010000B,SW_L2
        MOV     P1,#OFF
        LCALL   STOP
        MOV     P2,#M_R
        LCALL   DELAY
        LJMP    MAIN

SW_R2:  MOV     A,B
        ANL     A,#00110000B
        CJNE    A,#00100000B,MAIN
        MOV     P1,#OFF
        LCALL   STOP
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        MOV     P2,#M_L
        LCALL   DELAY
        LJMP    MAIN

;*******************TIME DEAY*****************************
DELAY:  MOV     R6,#0FFH        ; ***TEST***
D_LAY:  MOV     R7,#0FFH
D:      DJNZ    R7,D
        DJNZ    R6,D_LAY
        RET
;==================MOTOR MOVING================
GO:     MOV     P2,#M_GO
        RET

STOP:   MOV     P2,#M_STOP
        RET

LL:     MOV     P2,#M_L
        MOV     A,P0
        ANL     A,#00110000B
        CJNE    A,#00010000B,LL
        LCALL   STOP
        MOV     P1,#ON
        RET

RR:     MOV     P2,#M_R
        MOV     A,P0
        ANL     A,#00110000B
        CJNE    A,#00100000B,RR
        LCALL   STOP
        MOV     P1,#ON
        RET

INT_1:  PUSH    ACC
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        MOV     A,P3
        ANL     A,#00110000B
        CJNE    A,#00010000B,LL
        LCALL   STOP
        MOV     P1,#ON
        RET

RR:     MOV     P2,#M_R
        MOV     A,P3
        ANL     A,#00110000B
        CJNE    A,#00100000B,RR
        LCALL   STOP
        MOV     P1,#ON
        RET
END

ORG     0000H

M_GO        EQU     11000101B
M_STOP      EQU     11100000B
M_R         EQU     11000110B
M_L         EQU     11001001B
ON          EQU     00110011B
OFF         EQU     11001100B


        MOV     P0,#0FFH
        MOV     P1,#00H
        MOV     P2,#00H
        MOV     P3,#0FFH

        MOV     P1,#ON
        LCALL   STOP
        LCALL   STOP
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

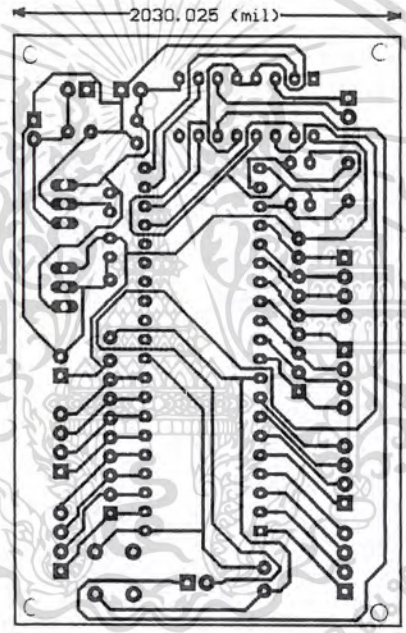
```
SEN_L: MOV    A,B
       ANL    A,#00000011B
       CJNE   A,#00000001B,SW_1
       MOV    P1,#OFF
       LCALL  STOP
       LCALL  RR
       LJMP   MAIN

SW_1:  MOV    A,B
       ANL    A,#00110011B
       CJNE   A,#00010011B,SW_2
       MOV    P1,#OFF
       LCALL  STOP
       MOV    P2,#M_R
       LJMP   SW_1

SW_2:  MOV    A,B
       ANL    A,#00110011B
       CJNE   A,#00100011B,MAIN
       MOV    P1,#OFF
       LCALL  STOP
       MOV    P2,#M_L
       LJMP   SW_2

;**********TIME DEAY**********
DELAY: MOV    R6,#0FFH        ;
D_LAY: MOV    R7,#0FFH
D:     DJNZ   R7,D
       DJNZ   R6,D_LAY
       RET

;================ CHECK LINK BUS ==============
LINK:  MOV    A,P0
       ANL    A,#11110000B
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

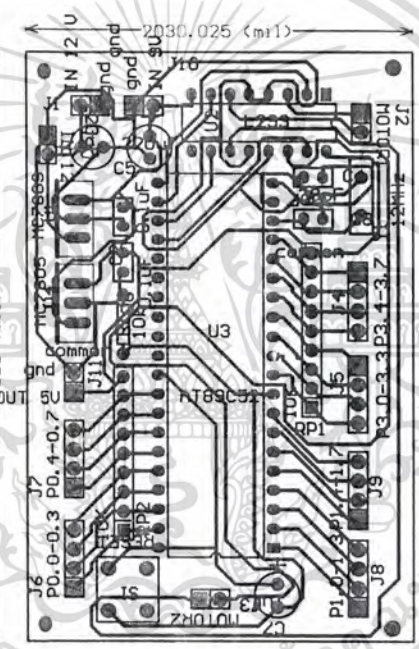
```
        CJNE   A,#11000000B,E_LINK
        MOV    P2,#M_STOP
        LJMP   LINK
E_LINK: RET

;==================MOTOR MOVING==============
GO:     MOV    P2,#M_GO
        RET

STOP:   MOV    P2,#M_STOP
        LCALL  DELAY
        LCALL  DELAY
        LCALL  DELAY
        LCALL  DELAY
        RET

LL:     MOV    P2,#M_L
        MOV    A,P3
        ANL    A,#00110000B
        CJNE   A,#00010000B,LL
        LCALL  STOP
        MOV    P1,#ON
        RET

RR:     MOV    P2,#M_R
        MOV    A,P3
        ANL    A,#00110000B
        CJNE   A,#00100000B,RR
        LCALL  STOP
        MOV    P1,#ON
        RET
END
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

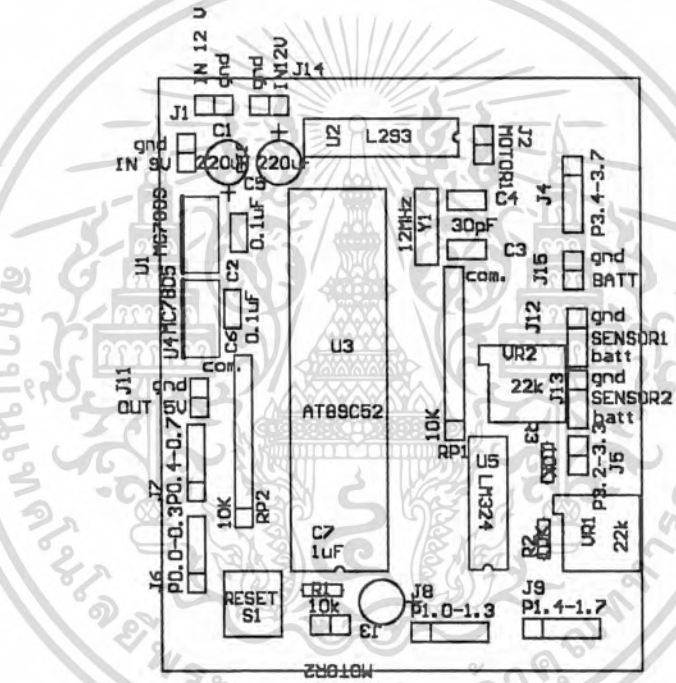
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



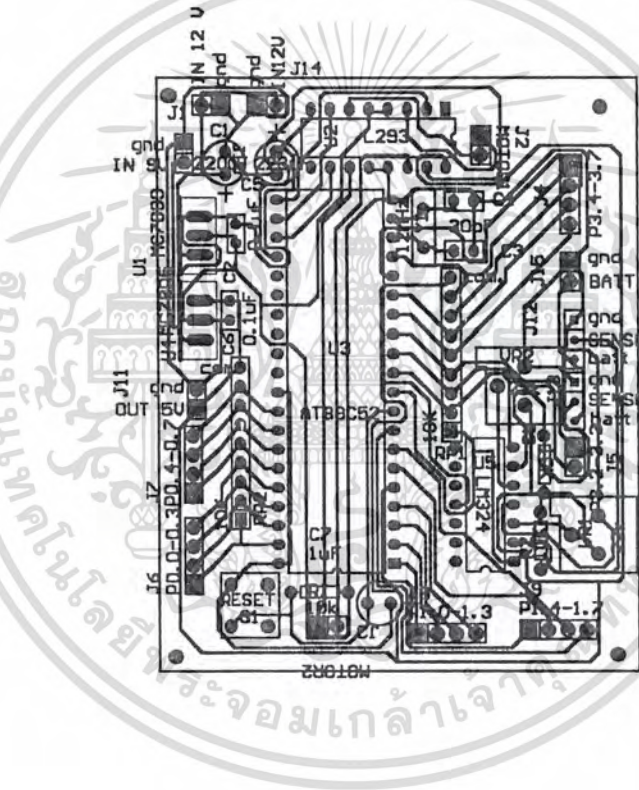
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



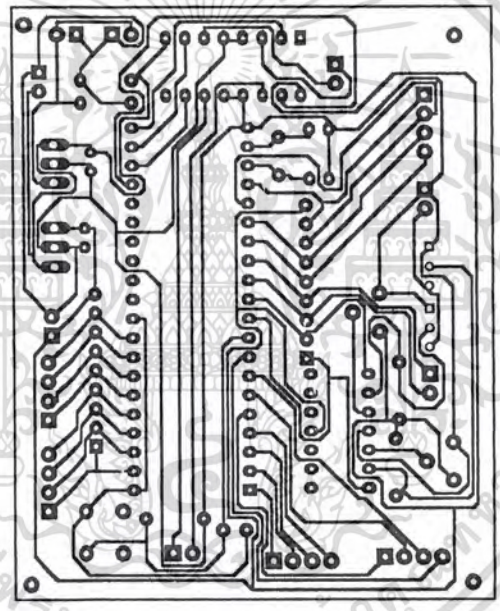
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



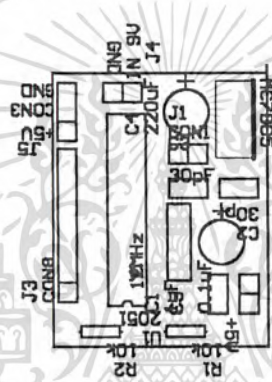
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



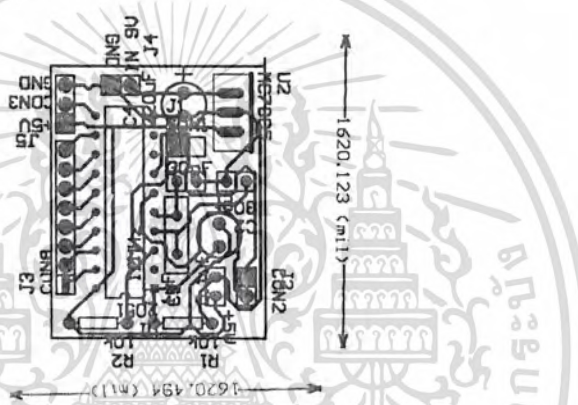
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



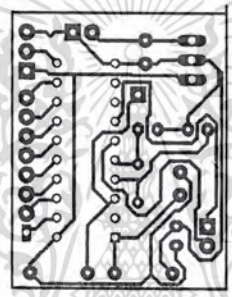
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



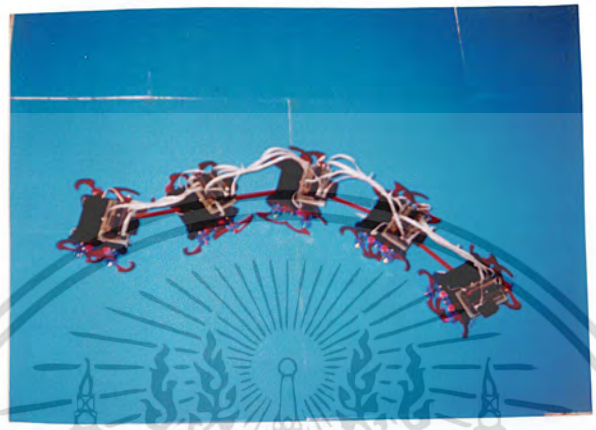
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



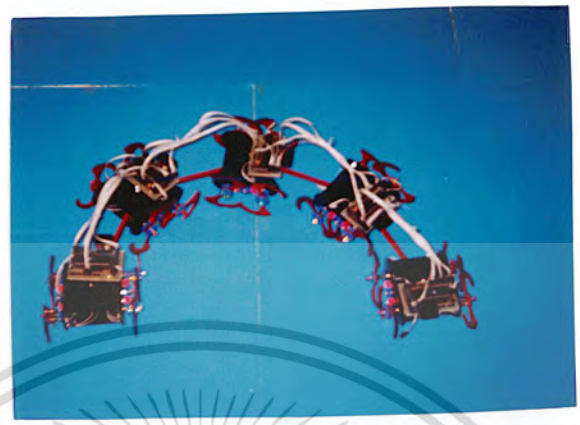
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก ข.
รูปถ่ายหุ่นยนต์

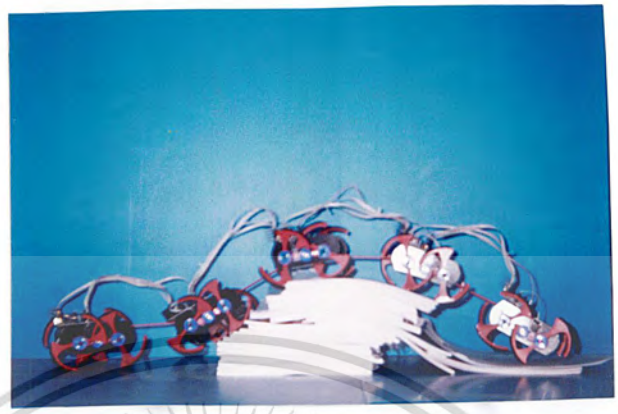
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก ค.
รายละเอียดของไมโครคอนโทรลเลอร์

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
- 4.0V 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.

8-Bit Microcontroller with 8K Bytes Flash

AT89S8252

0401D-A-12/97

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

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

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

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

Port 2

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

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

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

Port 3

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

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

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

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

RST

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

ALE/$\overline{PROG}$

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

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

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

$\overline{PSEN}$

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

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

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

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

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

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

XTAL1
Input to the inverting oscillator amplifier and input to the internal clock operating circuit.

XTAL2
Output from the inverting oscillator amplifier.

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.

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

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.

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.

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.

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

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.

Table 3. WMCON—Watchdog and Memory Control Register

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

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

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.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Table 4. SPCR—SPI Control Register

SPCR Address = D5H — Reset Value = 0000 01XXB

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

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	SPIF	WCOL	—	—	—	—	—	—
	7	6	5	4	3	2	1	0

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.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Table 6. SPDR—SPI Data Register

SPDR Address = 86H — Reset Value = unchanged

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

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			Period (nominal)
PS2	PS1	PS0	
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

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

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 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

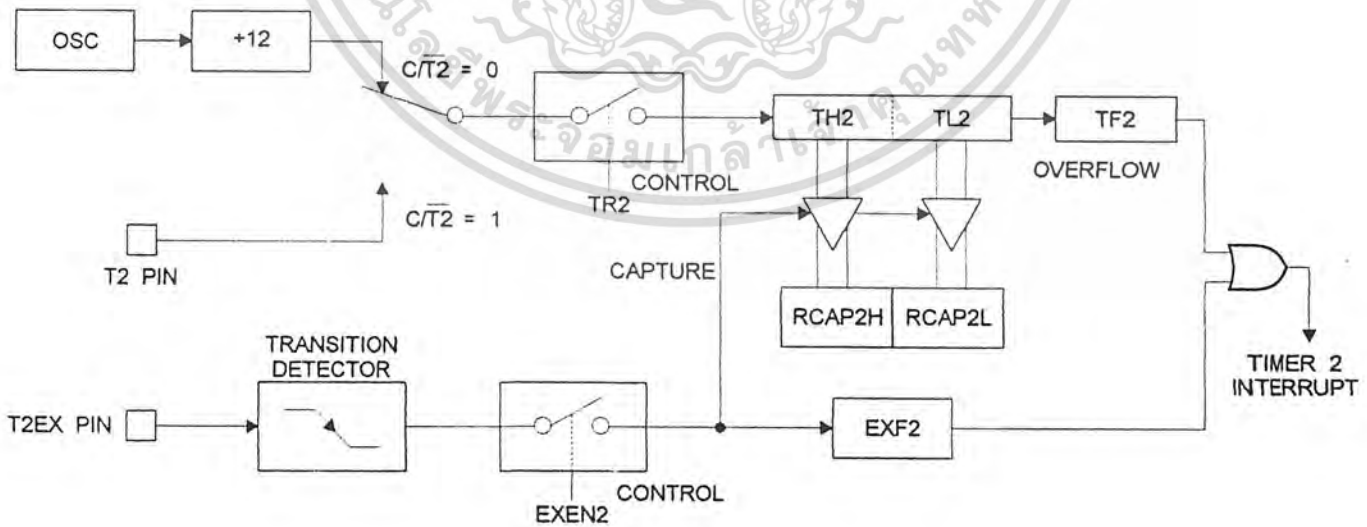
In the capture mode, two options are selected by bit EXEN2 in T2CON. If EXEN2 = 0, Timer 2 is a 16 bit timer or counter which upon overflow sets bit TF2 in T2CON. This bit can then be used to generate an interrupt. If EXEN2 = 1, Timer 2 performs the same operation, but a 1-to-0 transition at external 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.

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

Figure 1. Timer 2 in Capture Mode



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

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)

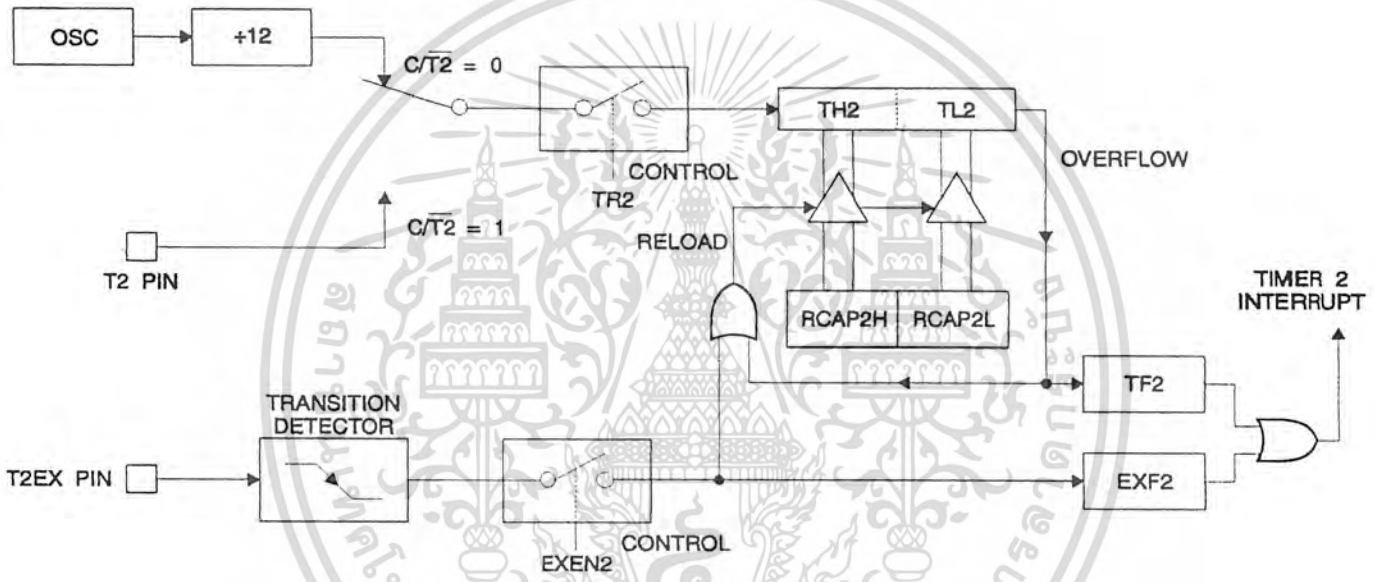


Table 9. T2MOD—Timer 2 Mode Control Register

T2MOD Address = 0C9H — Reset Value = XXXX XX00B

Not Bit Addressable

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

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)

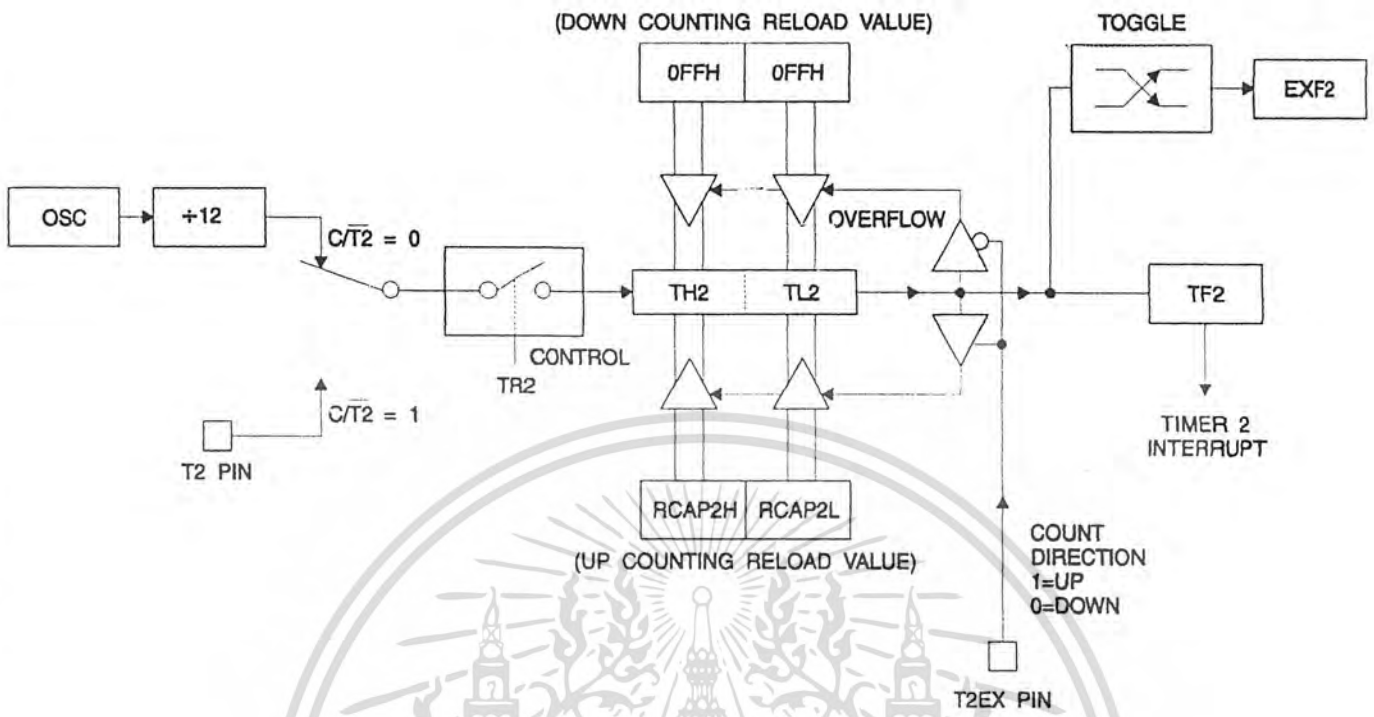
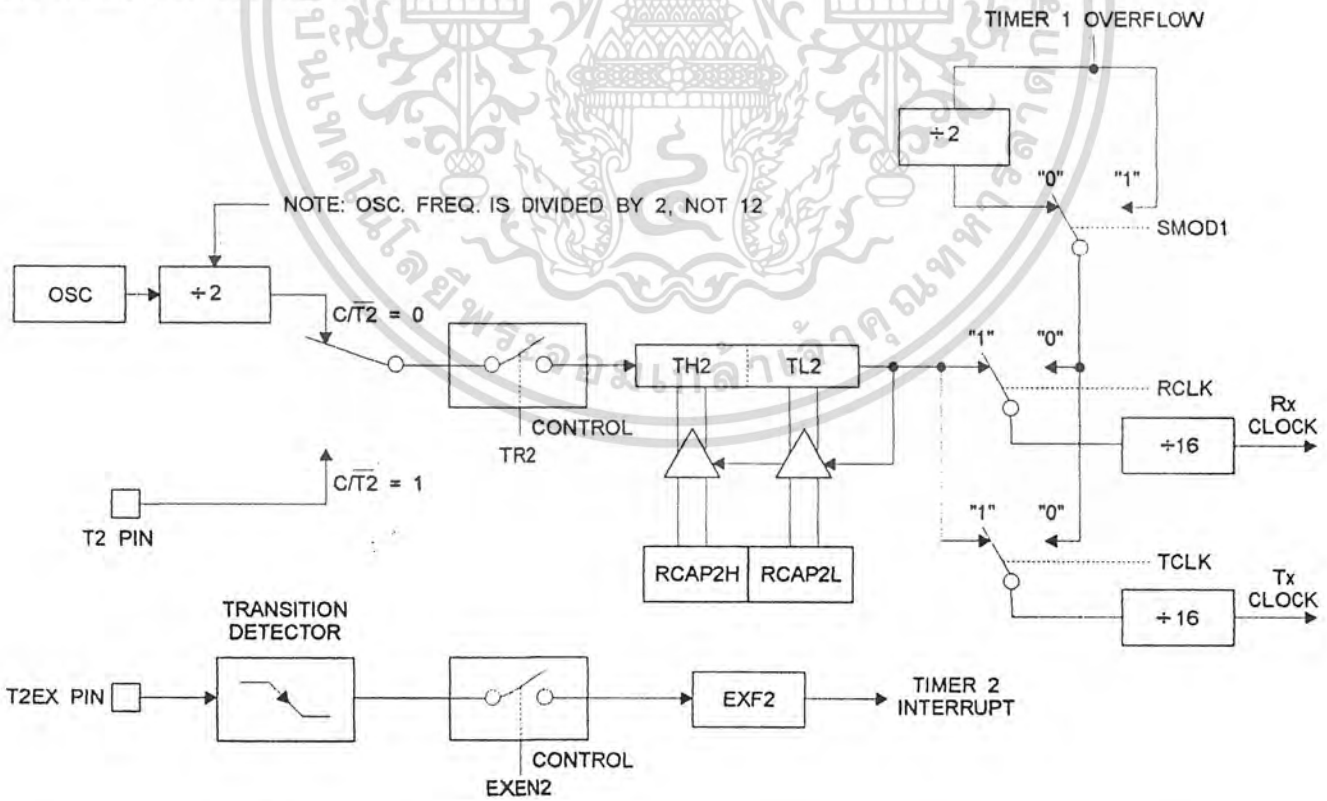


Figure 4. Timer 2 in Baud Rate Generator Mode



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Figure 14. Programming the Flash/EEPROM Memory

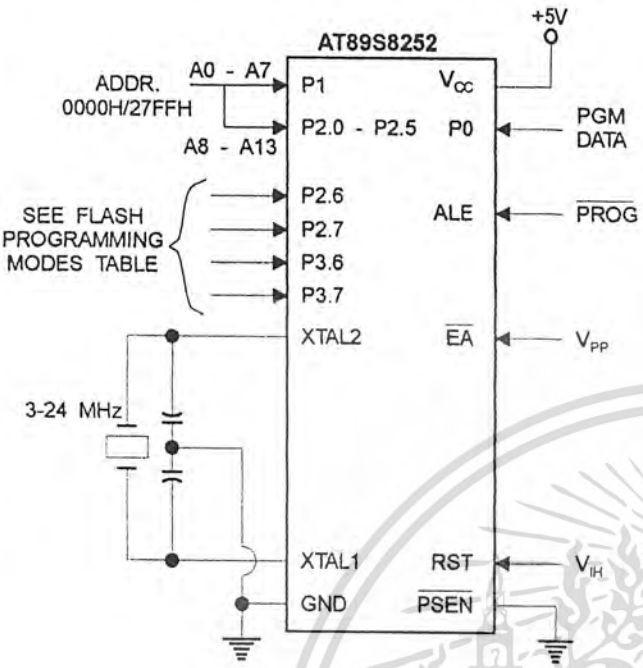


Figure 15. Flash/EEPROM Serial Downloading

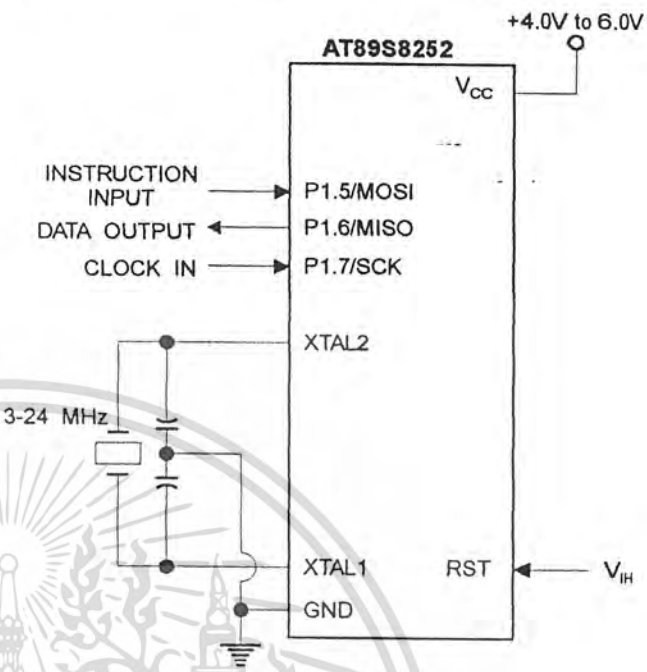
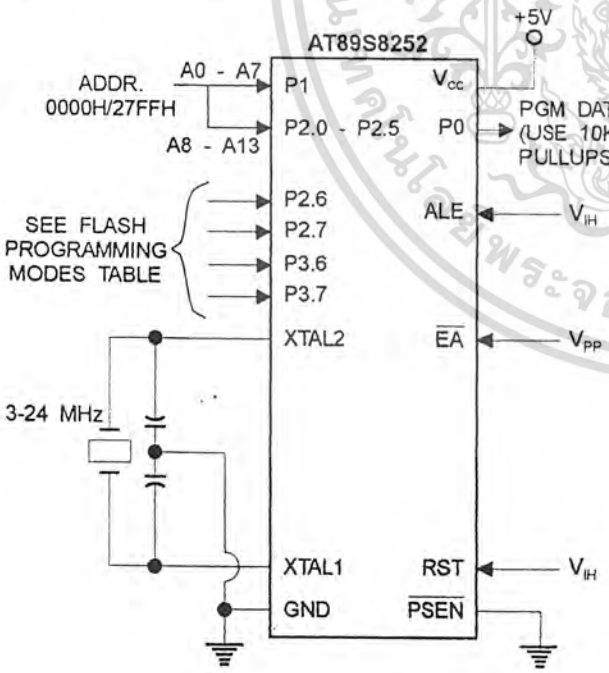


Figure 16. Verifying the Flash/EEPROM Memory



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

ภาคผนวก ง.
รายละเอียดของ IC ที่ใช้ในโครงงาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

SLVS055F – APRIL 1981 – REVISED DECEMBER 1999

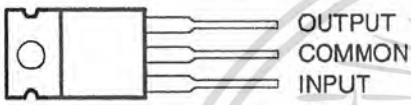
- ±1% Output Tolerance at 25°C
- ±2% Output Tolerance Over Full Operating Range
- Thermal Shutdown
- Internal Short-Circuit Current Limiting
- Pinout Identical to μA7800 Series
- Improved Version of μA7800 Series

description

Each fixed-voltage precision regulator in the TL780 series is capable of supplying 1.5 A of load current. A unique temperature-compensation technique, coupled with an internally trimmed band-gap reference, has resulted in improved accuracy when compared to other three-terminal regulators. Advanced layout techniques provide excellent line, load, and thermal regulation. The internal current-limiting and thermal-shutdown features make the devices essentially immune to overload.

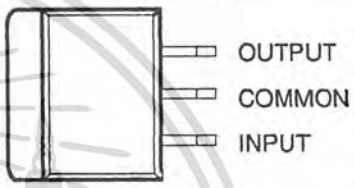
The TL780-*xx*C series regulators are characterized for operation over the virtual junction temperature range of 0°C to 125°C.

KC PACKAGE
(TOP VIEW)



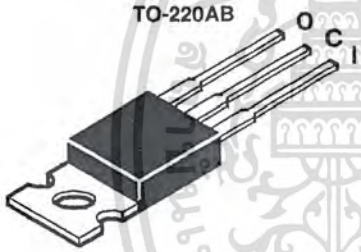
The COMMON terminal is in electrical contact with the mounting base.

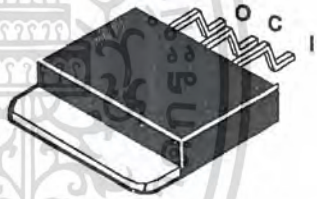
KTE PACKAGE
(TOP VIEW)



The COMMON terminal is in electrical contact with the mounting base.

TO-220AB





AVAILABLE OPTIONS

T_J	V_O TYP (V)	PACKAGED DEVICES		CHIP FORM (Y)
		HEAT-SINK MOUNTED (KC)	PLASTIC FLANGE MOUNTED (KTE)	
0°C to 125°C	5	TL780-05CKC	TL780-05CKTE	TL780-05Y
	12	TL780-12CKC	TL780-12CKTE	TL780-12Y
	15	TL780-15CKC	TL780-15CKTE	TL780-15Y

The KTE package is available taped and reeled. Add the suffix R to the device type (e.g., TL780-05CKTER). Chip forms are tested at 25°C.

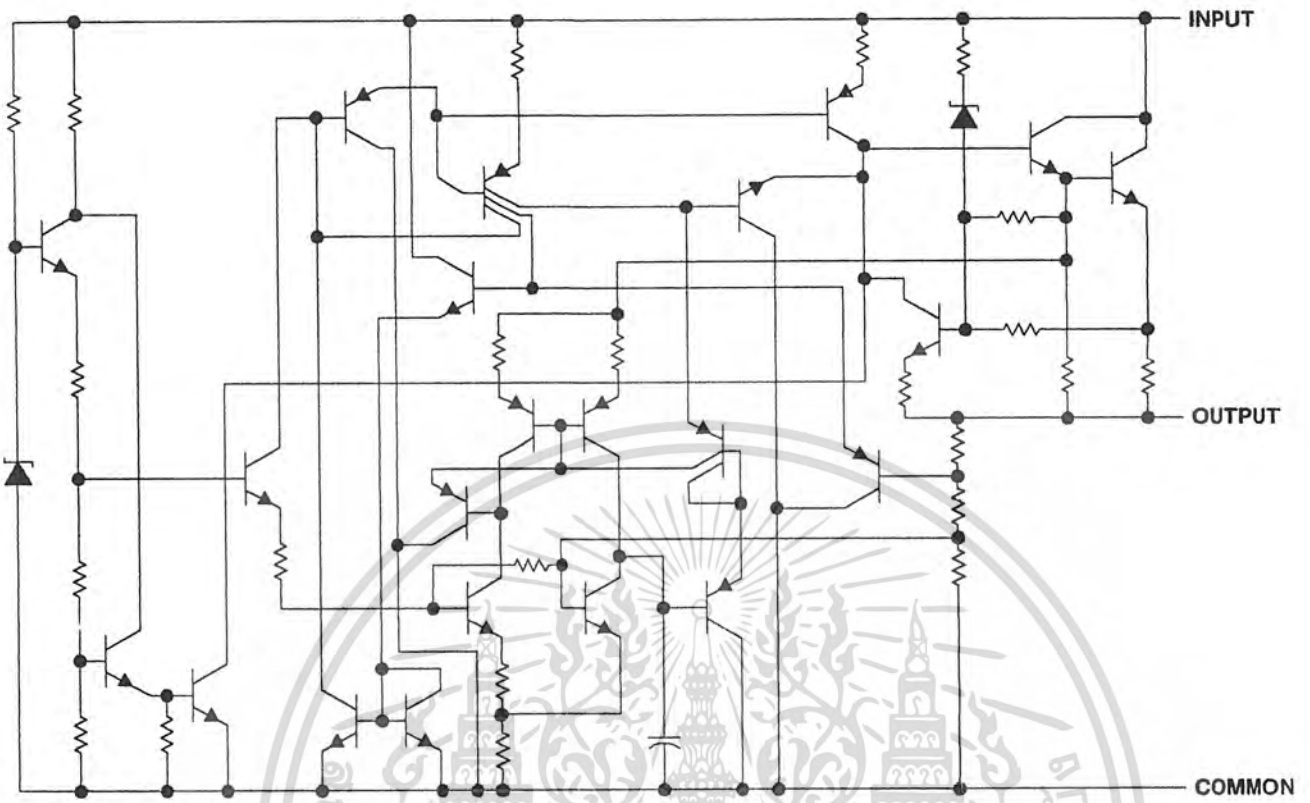
Please be aware that an important notice concerning availability, standard warranty, and use in critical applications of Texas Instruments semiconductor products and disclaimers thereto appears at the end of this data sheet.

PRODUCTION DATA information is current as of publication date. Products conform to specifications per the terms of Texas Instruments standard warranty. Production processing does not necessarily include testing of all parameters.

Copyright © 1999, Texas Instruments Incorporated

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

schematic



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

absolute maximum ratings over operating temperature range (unless otherwise noted)†

Input voltage, V_I ..... 35 V
Package thermal impedance, θ_{JA} (see Notes 1 and 2): KC package ..... 22°C/W
KTE package ..... 23°C/W
Operating free-air, T_A; case, T_C; or virtual junction, T_J, temperature range ..... 0°C to 150°C
Lead temperature 1,6 mm (1/16 inch) from case for 10 seconds ..... 260°C
Storage temperature range, T_{stg} ..... –65°C to 150°C

† Stresses beyond those listed under "absolute maximum ratings" may cause permanent damage to the device. These are stress ratings only, and functional operation of the device at these or any other conditions beyond those indicated under "recommended operating conditions" is not implied. Exposure to absolute-maximum-rated conditions for extended periods may affect device reliability.

NOTES: 1. Maximum power dissipation is a function of $T_J(max)$, θ_{JA}, and T_A. The maximum allowable power dissipation at any allowable ambient temperature is $P_D = (T_J(max) - T_A)/\theta_{JA}$. Operating at the absolute maximum T_J of 150°C can impact reliability. Due to variations in individual device electrical characteristics and thermal resistance, the built-in thermal overload protection may be activated at power levels slightly above or below the rated dissipation.
2. The package thermal impedance is calculated in accordance with JESD 51.

recommended operating conditions

		MIN	MAX	UNIT
Input voltage, V_I	TL780-05C	7	25	V
	TL780-12C	14.5	30	
	TL780-15C	17.5	30	
Output current, I_O			1.5	A
Operating virtual junction temperature, T_J		0	125	°C

electrical characteristics at specified virtual junction temperature, V_I = 10 V, I_O = 500 mA (unless otherwise noted)

PARAMETER	TEST CONDITIONS	T_J‡	TL780-05C			UNIT
			MIN	TYP	MAX	
Output voltage	I_O = 5 mA to 1 A, P ≤ 15 W, V_I = 7 V to 20 V	25°C	4.95	5	5.05	V
		0°C to 125°C	4.9		5.1	
Input voltage regulation	V_I = 7 V to 25 V	25°C		0.5	5	mV
	V_I = 8 V to 12 V			0.5	5	
Ripple rejection	V_I = 8 V to 18 V, f = 120 Hz	0°C to 125°C	70	85		dB
Output voltage regulation	I_O = 5 mA to 1.5 A	25°C		4	25	mV
	I_O = 250 mA to 750 mA			1.5	15	
Output resistance	f = 1 kHz	0°C to 125°C		0.0035		W
Temperature coefficient of output voltage	I_O = 5 mA	0°C to 125°C		0.25		mV/°C
Output noise voltage	f = 10 Hz to 100 kHz	25°C		75		μV
Dropout voltage	I_O = 1 A	25°C		2		V
Input bias current		25°C		5	8	mA
Input bias-current change	V_I = 7 V to 25 V	0°C to 125°C		0.7	1.3	mA
	I_O = 5 mA to 1 A			0.003	0.5	
Short-circuit output current		25°C		750		mA
Peak output current		25°C		2.2		A

‡ Pulse-testing techniques maintain the junction temperature as close to the ambient temperature as possible. Thermal effects must be taken into account separately. All characteristics are measured with a 0.33-μF capacitor across the input and a 0.22-μF capacitor across the output.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

electrical characteristics at specified virtual junction temperature, V_I = 19 V, I_O = 500 mA (unless otherwise noted)

PARAMETER	TEST CONDITIONS	T_J†	TL780-12C			UNIT
			MIN	TYP	MAX	
Output voltage	I_O = 5 mA to 1 A, P ≤ 15 W, V_I = 14.5 V to 27 V	25°C	11.88	12	12.12	V
		0°C to 125°C	11.76		12.24	
Input voltage regulation	V_I = 14.5 V to 30 V	25°C		1.2	12	mV
	V_I = 16 V to 22 V			1.2	12	
Ripple rejection	V_I = 15 V to 25 V, f = 120 Hz	0°C to 125°C	65	80		dB
Output voltage regulation	I_O = 5 mA to 1.5 A	25°C		6.5	60	mV
	I_O = 250 mA to 750 mA			2.5	36	
Output resistance	f = 1 kHz	0°C to 125°C		0.0035		W
Temperature coefficient of output voltage	I_O = 5 mA	0°C to 125°C		0.6		mV/°C
Output noise voltage	f = 10 Hz to 100 kHz	25°C		180		µV
Dropout voltage	I_O = 1 A	25°C		2		V
Input bias current		25°C		5.5	8	mA
Input bias-current change	V_I = 14.5 V to 30 V	0°C to 125°C		0.4	1.3	mA
	I_O = 5 mA to 1 A			0.03	0.5	
Short-circuit output current		25°C		350		mA
Peak output current		25°C		2.2		A

† Pulse-testing techniques maintain the junction temperature as close to the ambient temperature as possible. Thermal effects must be taken into account separately. All characteristics are measured with a 0.33-µF capacitor across the input and a 0.22-µF capacitor across the output.

electrical characteristics at specified virtual junction temperature, V_I = 23 V, I_O = 500 mA (unless otherwise noted)

PARAMETER	TEST CONDITIONS	T_J†	TL780-15C			UNIT
			MIN	TYP	MAX	
Output voltage	I_O = 5 mA to 1 A, P ≤ 15 W, V_I = 17.5 V to 30 V	25°C	14.85	15	15.15	V
		0°C to 125°C	14.7		15.3	
Input voltage regulation	V_I = 17.5 V to 30 V	25°C		1.5	15	mV
	V_I = 20 V to 26 V			1.5	15	
Ripple rejection	V_I = 18.5 V to 28.5 V, f = 120 Hz	0°C to 125°C	60	75		dB
Output voltage regulation	I_O = 5 mA to 1.5 A	25°C		7	75	mV
	I_O = 250 mA to 750 mA			2.5	45	
Output resistance	f = 1 kHz	0°C to 125°C		0.0035		W
Temperature coefficient of output voltage	I_O = 5 mA	0°C to 125°C		0.62		mV/°C
Output noise voltage	f = 10 Hz to 100 kHz	25°C		225		µV
Dropout voltage	I_O = 1 A	25°C		2		V
Input bias current		25°C		5.5	8	mA
Input bias-current change	V_I = 17.5 V to 30 V	0°C to 125°C		0.4	1.3	mA
	I_O = 5 mA to 1 A			0.02	0.5	
Short-circuit output current		25°C		230		mA
Peak output current		25°C		2.2		A

† Pulse-testing techniques maintain the junction temperature as close to the ambient temperature as possible. Thermal effects must be taken into account separately. All characteristics are measured with a 0.33-µF capacitor across the input and a 0.22-µF capacitor across the output.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

DALLAS SEMICONDUCTOR

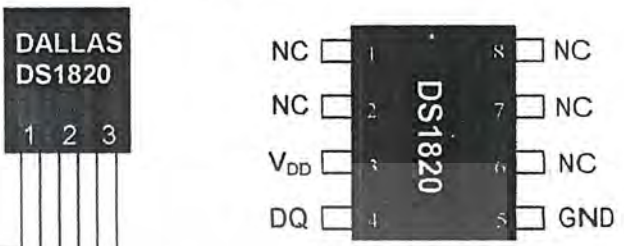
DS18S20
High Precision 1-Wire® Digital Thermometer

www.dalsemi.com

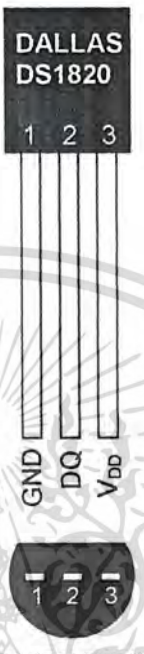
FEATURES

- Unique 1-wire interface requires only one port pin for communication
- Each device has a unique 64-bit serial code stored in an on-board ROM
- Multi-drop capability simplifies distributed temperature sensing applications
- Requires no external components
- Can be powered from data line. Power supply range is 3.0V to 5.5V
- Measures temperatures from –55°C to +125°C (–67°F to +257°F)
- ±0.5°C accuracy from –10°C to +85°C
- 9-bit thermometer resolution
- Converts temperature in 750 ms (max.)
- User-definable nonvolatile alarm settings
- Alarm search command identifies and addresses devices whose temperature is outside of programmed limits (temperature alarm condition)
- Applications include thermostatic controls, industrial systems, consumer products, thermometers, or any thermally sensitive system

PIN ASSIGNMENT



8-pin 150-mil SOIC
(DS18S20Z)



(BOTTOM VIEW)

TO-92
(DS18S20)

PIN DESCRIPTION

GND - Ground
DQ - Data In/Out
V_{DD} - Power Supply Voltage
NC - No Connect

DESCRIPTION

The DS18S20 Digital Thermometer provides 9–bit centigrade temperature measurements and has an alarm function with nonvolatile user-programmable upper and lower trigger points. The DS18S20 communicates over a 1-wire bus that by definition requires only one data line (and ground) for communication with a central microprocessor. It has an operating temperature range of –55°C to +125°C and is accurate to ±0.5°C over the range of –10°C to +85°C. In addition, the DS18S20 can derive power directly from the data line ("parasite power"), eliminating the need for an external power supply.

Each DS18S20 has a unique 64-bit serial code, which allows multiple DS18S20s to function on the same 1–wire bus; thus, it is simple to use one microprocessor to control many DS18S20s distributed over a large area. Applications that can benefit from this feature include HVAC environmental controls, temperature monitoring systems inside buildings, equipment or machinery, and process monitoring and control systems.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

DETAILED PIN DESCRIPTIONS Table 1

8-PIN SOIC*	TO-92	SYMBOL	DESCRIPTION
5	1	GND	**Ground.**
4	2	DQ	**Data Input/Output pin.** Open-drain 1-wire interface pin. Also provides power to the device when used in parasite power mode (see "Parasite Power" section.)
3	3	V_{DD}	**Optional V_{DD} pin.** V_{DD} must be grounded for operation in parasite power mode.

*All pins not specified in this table are "No Connect" pins.

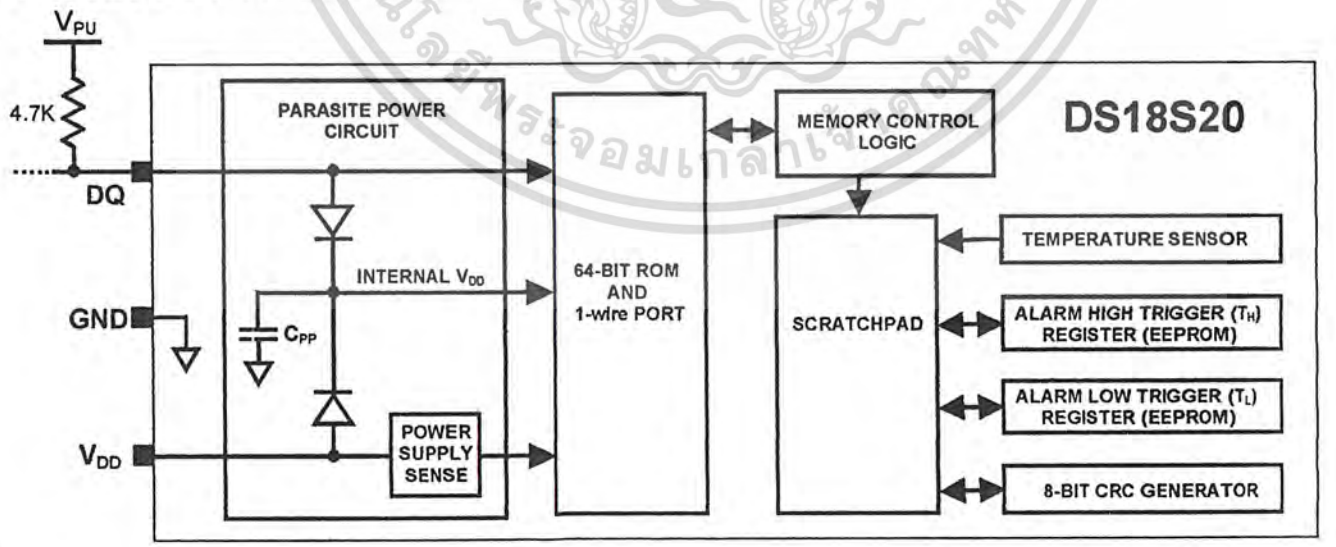
OVERVIEW

Figure 1 shows a block diagram of the DS18S20, and pin descriptions are given in Table 1. The 64-bit ROM stores the device's unique serial code. The scratchpad memory contains the 2-byte temperature register that stores the digital output from the temperature sensor. In addition, the scratchpad provides access to the 1-byte upper and lower alarm trigger registers (T_H and T_L). The T_H and T_L registers are nonvolatile (EEPROM), so they will retain data when the device is powered down.

The DS18S20 uses Dallas' exclusive 1-wire bus protocol that implements bus communication using one control signal. The control line requires a weak pullup resistor since all devices are linked to the bus via a 3-state or open-drain port (the DQ pin in the case of the DS18S20). In this bus system, the microprocessor (the master device) identifies and addresses devices on the bus using each device's unique 64-bit code. Because each device has a unique code, the number of devices that can be addressed on one bus is virtually unlimited. The 1-wire bus protocol, including detailed explanations of the commands and "time slots," is covered in the 1-WIRE BUS SYSTEM section of this datasheet.

Another feature of the DS18S20 is the ability to operate without an external power supply. Power is instead supplied through the 1-wire pullup resistor via the DQ pin when the bus is high. The high bus signal also charges an internal capacitor (C_{PP}), which then supplies power to the device when the bus is low. This method of deriving power from the 1-wire bus is referred to as "parasite power." As an alternative, the DS18S20 may also be powered by an external supply on V_{DD}.

DS18S20 BLOCK DIAGRAM Figure 1



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

OPERATION – MEASURING TEMPERATURE

The core functionality of the DS18S20 is its direct-to-digital temperature sensor. The temperature sensor output has 9-bit resolution, which corresponds to 0.5°C steps. The DS18S20 powers-up in a low-power idle state; to initiate a temperature measurement and A-to-D conversion, the master must issue a Convert T [44h] command. Following the conversion, the resulting thermal data is stored in the 2-byte temperature register in the scratchpad memory and the DS18S20 returns to its idle state. If the DS18S20 is powered by an external supply, the master can issue "read time slots" (see the 1-WIRE BUS SYSTEM section) after the Convert T command and the DS18S20 will respond by transmitting 0 while the temperature conversion is in progress and 1 when the conversion is done. If the DS18S20 is powered with parasite power, this notification technique cannot be used since the bus must be pulled high by a strong pullup during the entire temperature conversion. The bus requirements for parasite power are explained in detail in the POWERING THE DS18S20 section of this datasheet.

The DS18S20 output data is calibrated in degrees centigrade; for Fahrenheit applications, a lookup table or conversion routine must be used. The temperature data is stored as a 16-bit sign-extended two's complement number in the temperature register (see Figure 2). The sign bits (S) indicate if the temperature is positive or negative: for positive numbers S = 0 and for negative numbers S = 1. Table 2 gives examples of digital output data and the corresponding temperature reading.

Resolutions greater than 9 bits can be calculated using the data from the temperature, COUNT REMAIN and COUNT PER °C registers in the scratchpad. Note that the COUNT PER °C register is hard-wired to 16 (10h). After reading the scratchpad, the TEMP_READ value is obtained by truncating the 0.5°C bit (bit 0) from the temperature data (see Figure 2). The extended resolution temperature can then be calculated using the following equation:

$$TEMPERATURE = TEMP_READ - 0.25 + \frac{COUNT_PER_C - COUNT_REMAIN}{COUNT_PER_C}$$

Additional information about high-resolution temperature calculations can be found in Application Note 105: "High Resolution Temperature Measurement with Dallas Direct-to-Digital Temperature Sensors".

TEMPERATURE REGISTER FORMAT Figure 2

	bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0
LS Byte	2^6	2^5	2^4	2^3	2^2	2^1	2^0	2^{-1}
	bit 15	bit 14	bit 13	bit 12	bit 11	bit 10	bit 9	bit 8
MS Byte	S	S	S	S	S	S	S	S

TEMPERATURE/DATA RELATIONSHIP Table 2

TEMPERATURE	DIGITAL OUTPUT (Binary)	DIGITAL OUTPUT (Hex)
+85.0°C*	0000 0000 1010 1010	00AAh
+25.0°C	0000 0000 0011 0010	0032h
+0.5°C	0000 0000 0000 0001	0001h
0°C	0000 0000 0000 0000	0000h
-0.5°C	1111 1111 1111 1111	FFFFh
-25.0°C	1111 1111 1100 1110	FFCEh
-55.0°C	1111 1111 1001 0010	FF92h

*The power-on reset value of the temperature register is +85°C

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Push-Pull Four Channel Driver

FEATURES

- Output Current 1A Per Channel (600mA for L293D)
- Peak Output Current 2A Per Channel (1.2A for L293D)
- Inhibit Facility
- High Noise Immunity
- Separate Logic Supply
- Over-Temperature Protection

DESCRIPTION

The L293 and L293D are quad push-pull drivers capable of delivering output currents to 1A or 600mA per channel respectively. Each channel is controlled by a TTL-compatible logic input and each pair of drivers (a full bridge) is equipped with an inhibit input which turns off all four transistors. A separate supply input is provided for the logic so that it may be run off a lower voltage to reduce dissipation.

Additionally the L293D includes the output clamping diodes within the IC for complete interfacing with inductive loads.

Both devices are available in 16-pin Batwing DIP packages. They are also available in Power SOIC and Hermetic DIL packages.

TRUTH TABLE

V_I (each channel)	V_{INH}*	V_O
H	H	H
L	H	L
H	L	X**
L	L	X**

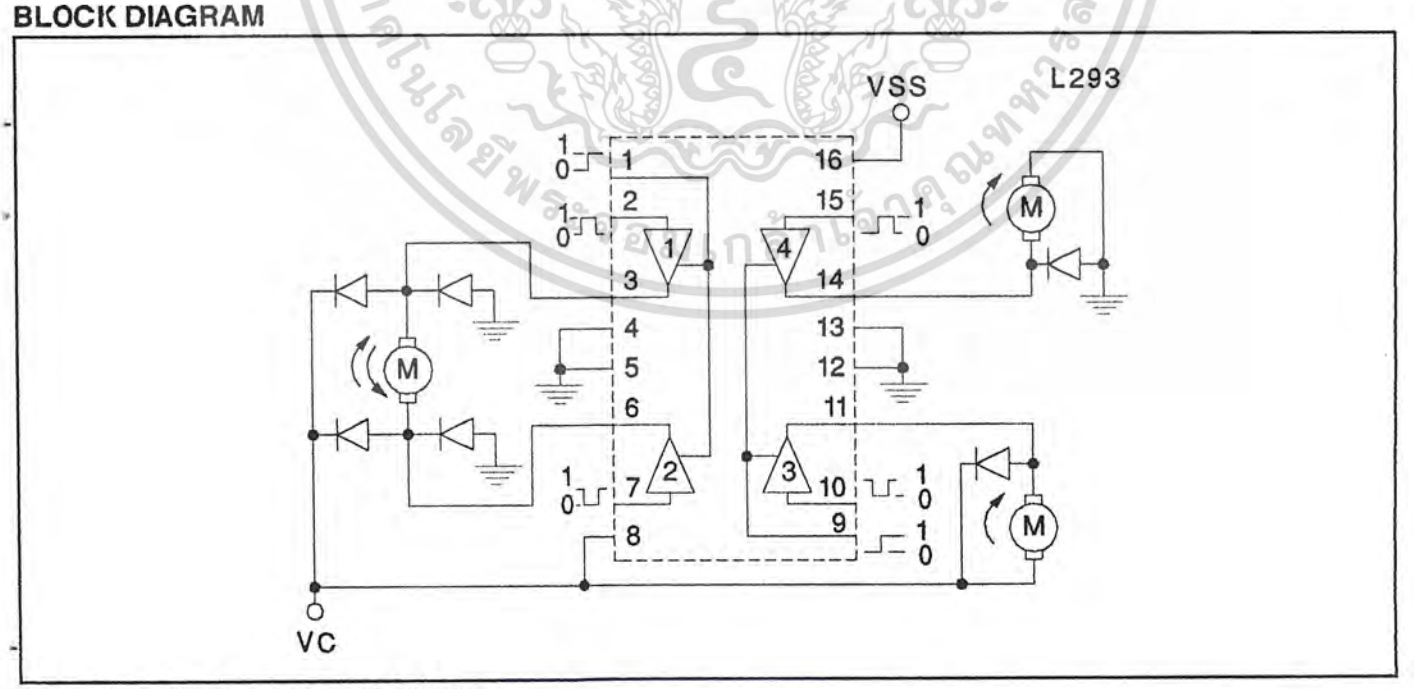
**Relative to the considered channel*
***High output impedence*

ABSOLUTE MAXIMUM RATINGS

Collector Supply Voltage, V_C	36V
Logic Supply Voltage, V_{SS}	36V
Input Voltage, V_I	7V
Inhibit Voltage, V_{INH}	7V
Peak Output Current (Non-Repetitive), I_{OUT} (L293)	2A
I_{OUT} (L293D)	1.2A
Total Power Dissipation at Tground-pins = 80°C, N Batwing pkg, (Note)	5W
Storage and Junction Temperature, Tstg, TJ	-40 to +150°C

Note:Consult packaging section of Databook for thermal limitations and considerations of packages.

BLOCK DIAGRAM



Note: Output diodes are internal in L293D.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

MOUNTING INSTRUCTIONS

The Rthj-amp of the L293 can be reduced by soldering the GND pins to a suitable copper area of the printed circuit board or to an external heatsink.

The diagram of Figure 13 shows the maximum package power Ptot and the θJA as a function of the side "l" of two equal square copper areas having a thickness of 35μ (see Figure 10). In addition, it is possible to use an external heatsink (see Figure 11).

During soldering the pins' temperature must not exceed 260°C and the soldering time must not be longer than 12 seconds.

The external heatsink or printed circuit copper area must be connected to electrical ground.

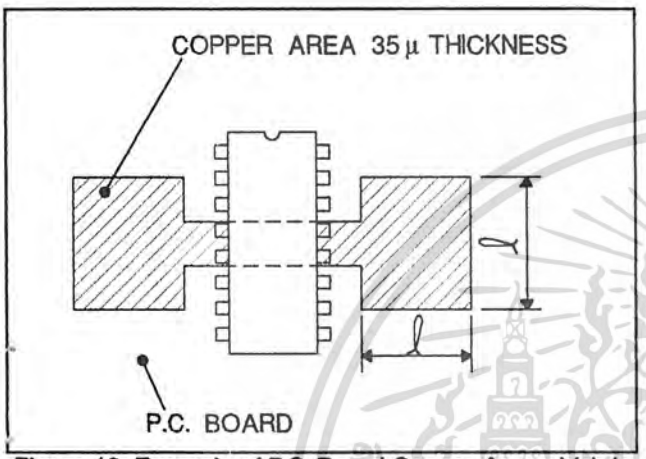


Figure 10: Example of P.C. Board Copper Area which is used as Heatsink

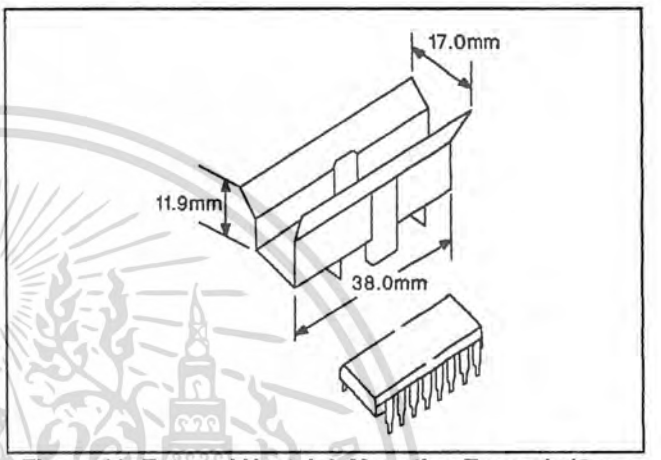


Figure 11: External Heatsink Mounting Example (θJA = 25°C/W)

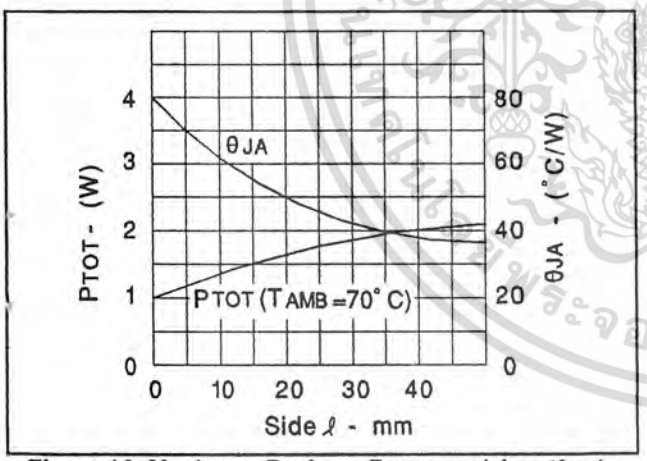


Figure 12: Maximum Package Power and Junction to Ambient Thermal Resistance

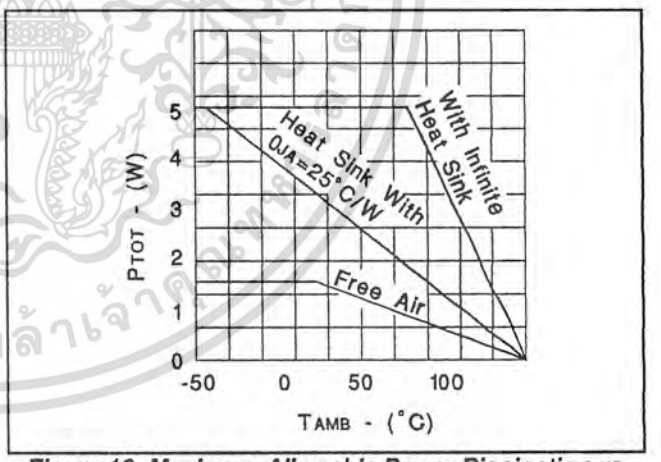


Figure 13: Maximum Allowable Power Dissipation vs Ambient Temperature

UNITRODE INTEGRATED CIRCUITS
7 CONTINENTAL BLVD. • MERRIMACK, NH 03054
TEL. (603) 424-2410 • FAX (603) 424-3460

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

National Semiconductor

August 2000

LM124/LM224/LM324/LM2902
Low Power Quad Operational Amplifiers

General Description

The LM124 series consists of four independent, high gain, internally frequency compensated operational amplifiers which were designed specifically to operate from a single power supply over a wide range of voltages. Operation from split power supplies is also possible and the low power supply current drain is independent of the magnitude of the power supply voltage.

Application areas include transducer amplifiers, DC gain blocks and all the conventional op amp circuits which now can be more easily implemented in single power supply systems. For example, the LM124 series can be directly operated off of the standard +5V power supply voltage which is used in digital systems and will easily provide the required interface electronics without requiring the additional ±15V power supplies.

Unique Characteristics

- In the linear mode the input common-mode voltage range includes ground and the output voltage can also swing to ground, even though operated from only a single power supply voltage
- The unity gain cross frequency is temperature compensated
- The input bias current is also temperature compensated

Advantages

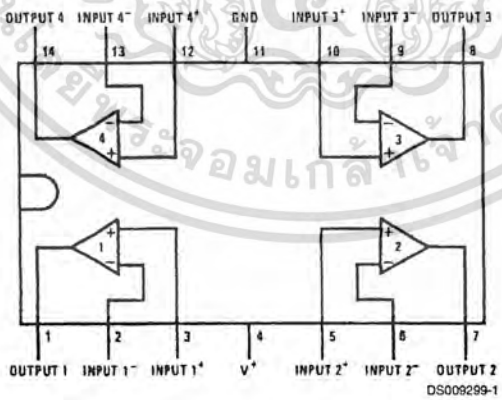
- Eliminates need for dual supplies
- Four internally compensated op amps in a single package
- Allows directly sensing near GND and V_{OUT} also goes to GND
- Compatible with all forms of logic
- Power drain suitable for battery operation

Features

- Internally frequency compensated for unity gain
- Large DC voltage gain 100 dB
- Wide bandwidth (unity gain) 1 MHz (temperature compensated)
- Wide power supply range:
  Single supply 3V to 32V
  or dual supplies ±1.5V to ±16V
- Very low supply current drain (700 µA)—essentially independent of supply voltage
- Low input biasing current 45 nA (temperature compensated)
- Low input offset voltage 2 mV and offset current: 5 nA
- Input common-mode voltage range includes ground
- Differential input voltage range equal to the power supply voltage
- Large output voltage swing 0V to V^+ – 1.5V

Connection Diagram

Dual-In-Line Package



Top View

Order Number LM124J, LM124AJ, LM124J/883 (Note 2), **LM124AJ/883** (Note 1), **LM224J, LM224AJ, LM324J, LM324M, LM324MX, LM324AM, LM324AMX, LM2902M, LM2902MX, LM324N, LM324AN, LM324MT, LM324MTX or LM2902N LM124AJRQML and LM124AJRQMLV**(Note 3)
See NS Package Number J14A, M14A or N14A

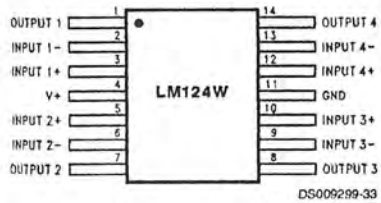
Note 1: LM124A available per JM38510/11006
Note 2: LM124 available per JM38510/11005

© 2000 National Semiconductor Corporation

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Connection Diagram (Continued)

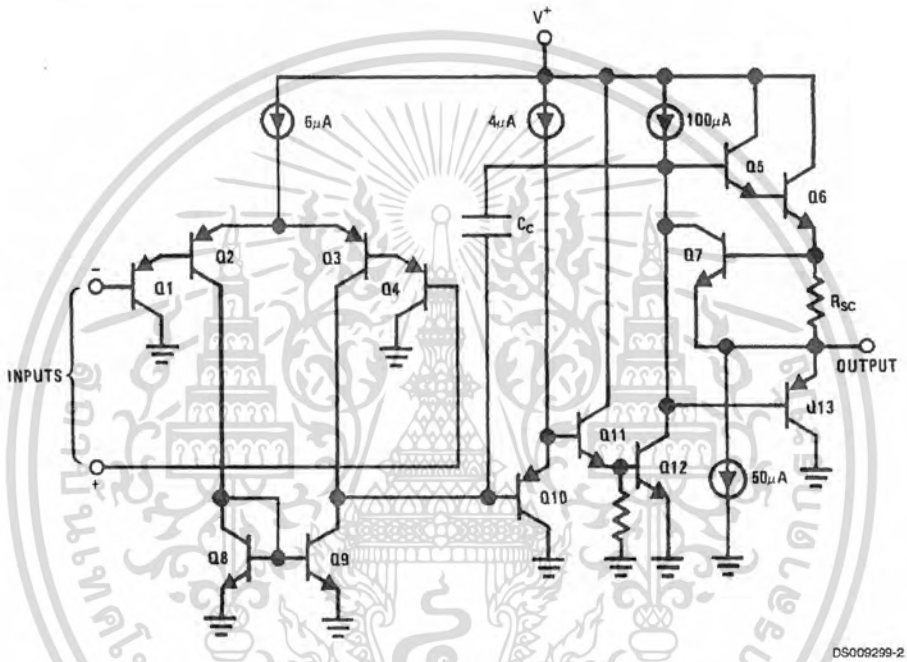
Note 3: See STD Mil DWG 5962R99504 for Radiation Tolerant Device



DS009299-33

Order Number LM124AW/883, LM124AWG/883, LM124W/883 or LM124WG/883
LM124AWRQML and LM124AWRQMLV(Note 3)
See NS Package Number W14B
LM124AWGRQML and LM124AWGRQMLV(Note 3)
See NS Package Number WG14A

Schematic Diagram (Each Amplifier)



DS009299-2

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Absolute Maximum Ratings (Note 12)

If Military/Aerospace specified devices are required, please contact the National Semiconductor Sales Office/ Distributors for availability and specifications.

	LM124/LM224/LM324 LM124A/LM224A/LM324A	LM2902
Supply Voltage, V^+	32V	26V
Differential Input Voltage	32V	26V
Input Voltage	−0.3V to +32V	−0.3V to +26V
Input Current (V_{IN} < −0.3V) (Note 6)	50 mA	50 mA
Power Dissipation (Note 4)		
Molded DIP	1130 mW	1130 mW
Cavity DIP	1260 mW	1260 mW
Small Outline Package	800 mW	800 mW
Output Short-Circuit to GND (One Amplifier) (Note 5) $V^+ \leq 15V$ and $T_A = 25°C$	Continuous	Continuous
Operating Temperature Range		−40°C to +85°C
LM324/LM324A	0°C to +70°C	
LM224/LM224A	−25°C to +85°C	
LM124/LM124A	−55°C to +125°C	
Storage Temperature Range	−65°C to +150°C	−65°C to +150°C
Lead Temperature (Soldering, 10 seconds)	260°C	260°C
Soldering Information		
Dual-In-Line Package		
Soldering (10 seconds)	260°C	260°C
Small Outline Package		
Vapor Phase (60 seconds)	215°C	215°C
Infrared (15 seconds)	220°C	220°C
See AN-450 "Surface Mounting Methods and Their Effect on Product Reliability" for other methods of soldering surface mount devices.		
ESD Tolerance (Note 13)	250V	250V

Electrical Characteristics

$V^+ = +5.0V$, (Note 7), unless otherwise stated

Parameter	Conditions	LM124A			LM224A			LM324A			Units
		Min	Typ	Max	Min	Typ	Max	Min	Typ	Max	
Input Offset Voltage	(Note 8) $T_A = 25°C$		1	2		1	3		2	3	mV
Input Bias Current (Note 9)	$I_{IN(+)}$ or $I_{IN(-)}$, $V_{CM} = 0V$, $T_A = 25°C$		20	50		40	80		45	100	nA
Input Offset Current	$I_{IN(+)}$ or $I_{IN(-)}$, $V_{CM} = 0V$, $T_A = 25°C$		2	10		2	15		5	30	nA
Input Common-Mode Voltage Range (Note 10)	$V^+ = 30V$, (LM2902, $V^+ = 26V$), $T_A = 25°C$	0		$V^+-1.5$	0		$V^+-1.5$	0		$V^+-1.5$	V
Supply Current	Over Full Temperature Range $R_L = \infty$ On All Op Amps										mA
	$V^+ = 30V$ (LM2902 $V^+ = 26V$)		1.5	3		1.5	3		1.5	3	
	$V^+ = 5V$		0.7	1.2		0.7	1.2		0.7	1.2	
Large Signal Voltage Gain	$V^+ = 15V$, $R_L \geq 2k\Omega$, ($V_O = 1V$ to $11V$), $T_A = 25°C$	50	100		50	100		25	100		V/mV
Common-Mode Rejection Ratio	DC, $V_{CM} = 0V$ to $V^+ - 1.5V$, $T_A = 25°C$	70	85		70	85		65	85		dB

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Electrical Characteristics (Continued)

V^+ = +5.0V, (Note 7), unless otherwise stated

Parameter		Conditions		LM124A			LM224A			LM324A			Units
				Min	Typ	Max	Min	Typ	Max	Min	Typ	Max	
Power Supply Rejection Ratio		V^+ = 5V to 30V (LM2902, V^+ = 5V to 26V), T_A = 25°C		65	100		65	100		65	100		dB
Amplifier-to-Amplifier Coupling (Note 11)		f = 1 kHz to 20 kHz, T_A = 25°C (Input Referred)			−120			−120			−120		dB
Output Current	Source	V_{IN}^+ = 1V, V_{IN}^- = 0V, V^+ = 15V, V_O = 2V, T_A = 25°C		20	40		20	40		20	40		mA
	Sink	V_{IN}^- = 1V, V_{IN}^+ = 0V, V^+ = 15V, V_O = 2V, T_A = 25°C		10	20		10	20		10	20		mA
		V_{IN}^- = 1V, V_{IN}^+ = 0V, V^+ = 15V, V_O = 200 mV, T_A = 25°C		12	50		12	50		12	50		µA
Short Circuit to Ground		(Note 5) V^+ = 15V, T_A = 25°C			40	60		40	60		40	60	mA
Input Offset Voltage		(Note 8)				4			4			5	mV
V_{OS} Drift		R_S = 0Ω			7	20		7	20		7	30	µV/°C
Input Offset Current		$I_{IN(+)} - I_{IN(-)}$, V_{CM} = 0V				30			30			75	nA
I_{OS} Drift		R_S = 0Ω			10	200		10	200		10	300	pA/°C
Input Bias Current		$I_{IN(+)}$ or $I_{IN(-)}$			40	100		40	100		40	200	nA
Input Common-Mode Voltage Range (Note 10)		V^+ = +30V (LM2902, V^+ = 26V)		0		V^+−2	0		V^+−2	0		V^+−2	V
Large Signal Voltage Gain		V^+ = +15V (V_OSwing = 1V to 11V) $R_L \geq 2$ kΩ		25			25			15			V/mV
Output Voltage Swing	V_{OH}	V^+ = 30V (LM2902, V^+ = 26V)	R_L = 2 kΩ	26			26			26			V
			R_L = 10 kΩ	27	28		27	28		27	28		
	V_{OL}	V^+ = 5V, R_L = 10 kΩ			5	20		5	20		5	20	mV
Output Current	Source	V_O = 2V	V_{IN}^+ = +1V, V_{IN}^- = 0V, V^+ = 15V	10	20		10	20		10	20		mA
	Sink		V_{IN}^- = +1V, V_{IN}^+ = 0V, V^+ = 15V	10	15		5	8		5	8		mA

Electrical Characteristics

V^+ = +5.0V, (Note 7), unless otherwise stated

Parameter	Conditions	LM124/LM224			LM324			LM2902			Units
		Min	Typ	Max	Min	Typ	Max	Min	Typ	Max	
Input Offset Voltage	(Note 8) T_A = 25°C		2	5		2	7		2	7	mV
Input Bias Current (Note 9)	$I_{IN(+)}$ or $I_{IN(-)}$, V_{CM} = 0V, T_A = 25°C		45	150		45	250		45	250	nA
Input Offset Current	$I_{IN(+)}$ or $I_{IN(-)}$, V_{CM} = 0V, T_A = 25°C		3	30		5	50		5	50	nA
Input Common-Mode Voltage Range (Note 10)	V^+ = 30V, (LM2902, V^+ = 26V), T_A = 25°C	0		V^+−1.5	0		V^+−1.5	0		V^+−1.5	V
Supply Current	Over Full Temperature Range R_L = ∞ On All Op Amps										mA
	V^+ = 30V (LM2902 V^+ = 26V)		1.5	3		1.5	3		1.5	3	
	V^+ = 5V		0.7	1.2		0.7	1.2		0.7	1.2	
Large Signal Voltage Gain	V^+ = 15V, $R_L \geq$ 2kΩ, (V_O = 1V to 11V), T_A = 25°C	50	100		25	100		25	100		V/mV
Common-Mode Rejection Ratio	DC, V_{CM} = 0V to V^+ − 1.5V, T_A = 25°C	70	85		65	85		50	70		dB
Power Supply Rejection Ratio	V^+ = 5V to 30V (LM2902, V^+ = 5V to 26V),	65	100		65	100		50	100		dB

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Electrical Characteristics (Continued)

V^+ = +5.0V, (Note 7), unless otherwise stated

Parameter		Conditions		LM124/LM224			LM324			LM2902			Units
				Min	Typ	Max	Min	Typ	Max	Min	Typ	Max	
		T_A = 25°C											
Amplifier-to-Amplifier Coupling (Note 11)		f = 1 kHz to 20 kHz, T_A = 25°C (Input Referred)			−120			−120			−120		dB
Output Current	Source	V_{IN}^+ = 1V, V_{IN}^- = 0V, V^+ = 15V, V_O = 2V, T_A = 25°C		20	40		20	40		20	40		mA
	Sink	V_{IN}^- = 1V, V_{IN}^+ = 0V, V^+ = 15V, V_O = 2V, T_A = 25°C		10	20		10	20		10	20		mA
		V_{IN}^- = 1V, V_{IN}^+ = 0V, V^+ = 15V, V_O = 200 mV, T_A = 25°C		12	50		12	50		12	50		µA
Short Circuit to Ground		(Note 5) V^+ = 15V, T_A = 25°C			40	60		40	60		40	60	mA
Input Offset Voltage		(Note 8)				7			9			10	mV
V_{OS} Drift		R_S = 0Ω			7			7			7		µV/°C
Input Offset Current		$I_{IN(+)} - I_{IN(-)}$, V_{CM} = 0V				100			150		45	200	nA
I_{OS} Drift		R_S = 0Ω			10			10			10		pA/°C
Input Bias Current		$I_{IN(+)}$ or $I_{IN(-)}$			40	300		40	500		40	500	nA
Input Common-Mode Voltage Range (Note 10)		V^+ = +30V (LM2902, V^+ = 26V)		0		V^+−2	0		V^+−2	0		V^+−2	V
Large Signal Voltage Gain		V^+ = +15V (V_OSwing = 1V to 11V) $R_L \geq 2$ kΩ		25			15			15			V/mV
Output Voltage Swing	V_{OH}	V^+ = 30V (LM2902, V^+ = 26V)	R_L = 2 kΩ	26			26			22			V
			R_L = 10 kΩ	27	28		27	28		23	24		
	V_{OL}	V^+ = 5V, R_L = 10 kΩ			5	20		5	20		5	100	mV
Output Current	Source	V_O = 2V	V_{IN}^+ = +1V, V_{IN}^- = 0V, V^+ = 15V	10	20		10	20		10	20		mA
	Sink		V_{IN}^- = +1V, V_{IN}^+ = 0V, V^+ = 15V	5	8		5	8		5	8		

Note 4: For operating at high temperatures, the LM324/LM324A/LM2902 must be derated based on a +125°C maximum junction temperature and a thermal resistance of 88°C/W which applies for the device soldered in a printed circuit board, operating in a still air ambient. The LM224/LM224A and LM124/LM124A can be derated based on a +150°C maximum junction temperature. The dissipation is the total of all four amplifiers — use external resistors, where possible, to allow the amplifier to saturate of to reduce the power which is dissipated in the integrated circuit.

Note 5: Short circuits from the output to V^+ can cause excessive heating and eventual destruction. When considering short circuits to ground, the maximum output current is approximately 40 mA independent of the magnitude of V^+. At values of supply voltage in excess of +15V, continuous short-circuits can exceed the power dissipation ratings and cause eventual destruction. Destructive dissipation can result from simultaneous shorts on all amplifiers.

Note 6: This input current will only exist when the voltage at any of the input leads is driven negative. It is due to the collector-base junction of the input PNP transistors becoming forward biased and thereby acting as input diode clamps. In addition to this diode action, there is also lateral NPN parasitic transistor action on the IC chip. This transistor action can cause the output voltages of the op amps to go to the V^+voltage level (or to ground for a large overdrive) for the time duration that an input is driven negative. This is not destructive and normal output states will re-establish when the input voltage, which was negative, again returns to a value greater than −0.3V (at 25°C).

Note 7: These specifications are limited to −55°C ≤ T_A ≤ +125°C for the LM124/LM124A. With the LM224/LM224A, all temperature specifications are limited to −25°C ≤ T_A ≤ +85°C, the LM324/LM324A temperature specifications are limited to 0°C ≤ T_A ≤ +70°C, and the LM2902 specifications are limited to −40°C ≤ T_A ≤ +85°C.

Note 8: V_O ≈ 1.4V, R_S = 0Ω with V^+ from 5V to 30V; and over the full input common-mode range (0V to V^+ − 1.5V) for LM2902, V^+ from 5V to 26V.

Note 9: The direction of the input current is out of the IC due to the PNP input stage. This current is essentially constant, independent of the state of the output so no loading change exists on the input lines.

Note 10: The input common-mode voltage of either input signal voltage should not be allowed to go negative by more than 0.3V (at 25°C). The upper end of the common-mode voltage range is V^+ − 1.5V (at 25°C), but either or both inputs can go to +32V without damage (+26V for LM2902), independent of the magnitude of V^+.

Note 11: Due to proximity of external components, insure that coupling is not originating via stray capacitance between these external parts. This typically can be detected as this type of capacitance increases at higher frequencies.

Note 12: Refer to RETS124AX for LM124A military specifications and refer to RETS124X for LM124 military specifications.

Note 13: Human body model, 1.5 kΩ in series with 100 pF.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บรรณานุกรม

1. รศ.สมยศ จุณณะปิยะ. 2541 . การประยุกต์ใช้งานไมโครคอนโทรลเลอร์ ตระกูล Mcs51 คณะวิศวกรรมศาสตร์, สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง,พิมพ์ครั้งที่ 2.

2. ผศ.พิพัฒน์ เลาหสงคราม. 2532 . "ไมโครคอนโทรเลอร์ Mcs51" . วารสารวิศวสารลาดกระบัง . ปีที่ 9 หน้า 60 – 68

3. วรพจน์ กรแก้ววัฒนกุล ชัยวัฒน์ ลิ้มพรจิตรวิไล . เรียนรู้และปฏิบัติการ ไมโครคอนโทรเลอร์ Mcs-51 แบบแฟลช ฉบับ AT89c5x ของ Atmel . , (c) Innovative Experiment Co.,Ltd

4. Joseph L. Jones and Anita M. Flynn . MOBILE ROBOTS Inspiration to Implementation ., A K Peters Wellesley, Massachusetts

5. GORDON MCCOMB . THE ROBOT BUILDER ' S BONAZA 99 INEXPENSIVE ROBOTICS PROJECT . ,McGraw-HiLL ,Int , 1987

6. John D. Lent . McGraw-will Circuit Encyclopedia and Troubleshooting Guied vol. 1 . 1993

7.คู่มือ AT89S8252,AT89S53 ATMEL FLASH MICROCONTROLLERS "Robobug Assembly" , Florida , 2000 , www.mekatronic.com

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้