

สำนักหอสมุดกลาง พระจอมเกล้าลาดกระบัง

หุ่นยนต์แมลง  
ROBOT BUG



โดย  
นาย ฉัฐพงษ์ วนาประเสริฐศักดิ์

ปฏิญานិพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต  
สาขาวิชาอิเล็กทรอนิกส์  
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง  
ปีการศึกษา 2542

เลขหมู่.....  
เลขทะเบียน..... 36901  
เอกสารนี้เป็นเอกสารสงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
วัน, เดือน, ปี 29 ส.ค. 2542  
เมื่อกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หุ่นยนต์แมลง  
ROBOT BUG



ปริญญาานิพนธ์สำหรับปริญญาวิศวกรรมศาสตรบัณฑิต  
สาขาวิชาอิเล็กทรอนิกส์  
คณะวิศวกรรมศาสตร์  
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง  
ปีการศึกษา 2542

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ปริญญาโทปีการศึกษา 2542

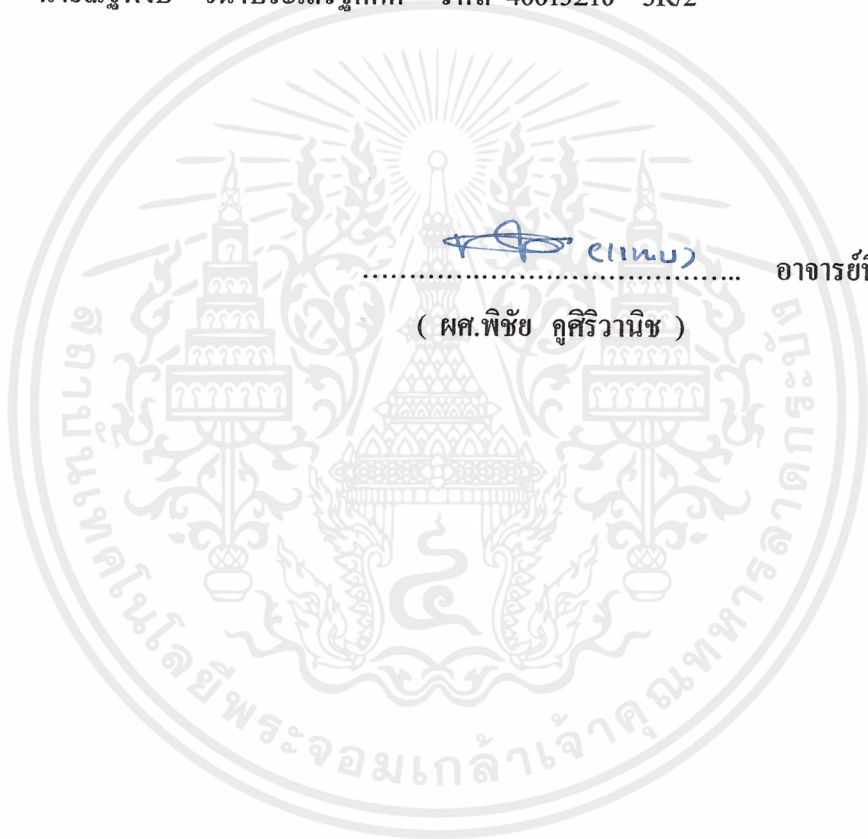
ภาควิชาอิเล็กทรอนิกส์

คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เรื่อง หุ่นยนต์แมลง

ผู้จัดทำ

1. นายณัฐพงษ์ วนาประเสริฐศักดิ์ รหัส 40013210 3R/2



อาจารย์ที่ปรึกษา

( ผศ.พิชัย กุศิริวานิช )

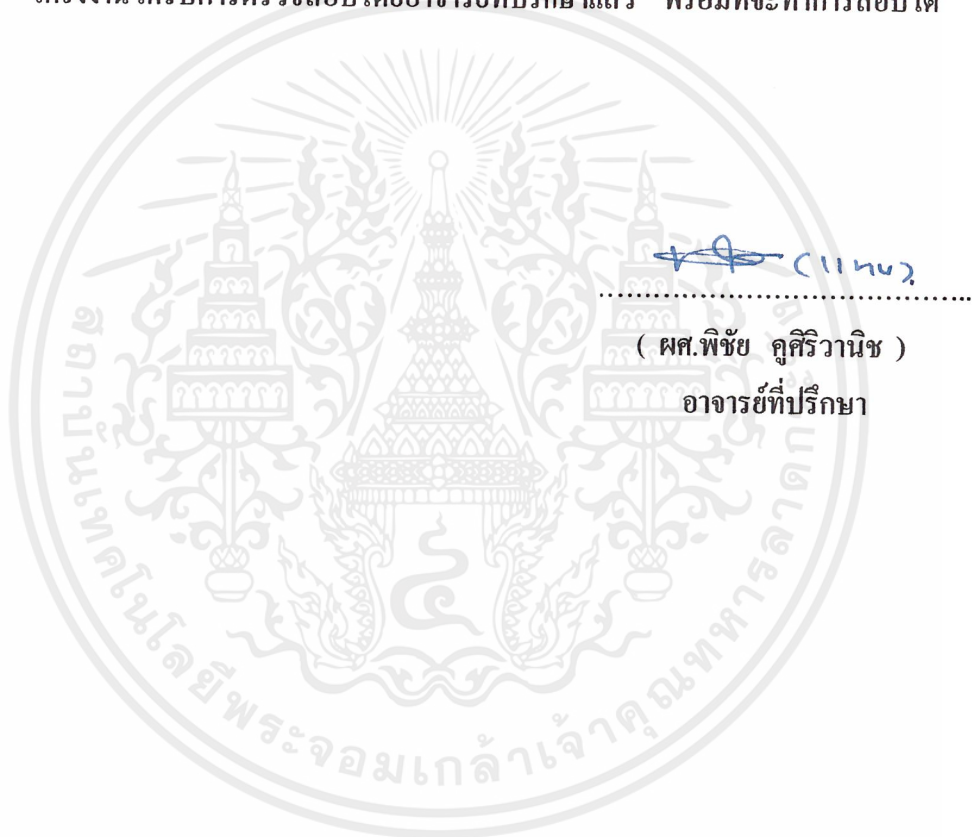
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หุ่นยนต์แมลง

Robot Bug

นายณัฐพงษ์ วนาประเสริฐศักดิ์ รหัส 40013210

โครงการได้รับการตรวจสอบโดยอาจารย์ที่ปรึกษาแล้ว พร้อมทั้งจะทำการสอบได้



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## กิตติกรรมประกาศ

สำหรับการจัดทำปริยฐานิพนธ์ฉบับนี้ ที่สำเร็จลุล่วงมาได้ก็ต้องขอขอบพระคุณหลาย ๆ ฝ่ายที่ช่วยกันให้การสนับสนุนจนประสบความสำเร็จ ขอขอบคุณ ผศ.พิชัย คูศิริวานิช ซึ่งเป็นอาจารย์ที่ปรึกษาได้ให้คำปรึกษา ขอขอบคุณเพื่อน ๆ ที่มีส่วนในงานชิ้นนี้ และขอขอบคุณสำหรับผู้ที่มีส่วนร่วม แต่มิได้เอ่ยนามไว้ ณ. ที่นี้ และที่สำคัญความห่วงใยจาก บิดา มารดา และครอบครัวที่คอยเป็นกำลังใจมาตลอด



(นายณัฐพงษ์ วนาประเสริฐศักดิ์)



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## ROBOT BUG

Mr. Nattapong Wanaprasertsak

Assistant Prof. Pichai Kusiriwannich

Academic year 1999

### Abstract

Microcontroller technology has been dramatically developed especially in the family of MCS-51 which, at present, are extensively utilized in number of industries and others.

The Robot Bug Project is motivated through various applications of the microcontroller 89C2051 under the family MCS-51. In this project, 89C2051 is programmed to control the functions of the robot. There is a sensor to detect light beam from the source. The robot will then move toward along the light source. The robot can automatically track the light source without stumbling. Through the coordination work of programmed hardware and software.

## สารบัญ

เรื่อง	หน้า
กิตติกรรมประกาศ	I
บทคัดย่อ	III
Abstract	III
สารบัญ	IV
สารบัญรูป	VI
<b>บทที่ 1 บทนำ</b>	1
1.1 วัตถุประสงค์	1
1.2 ขอบเขตของโครงการ	1
1.3 วิธีดำเนินงาน	2
1.4 ประโยชน์ที่ได้รับจากการทำโครงการ	3
<b>บทที่ 2 MCS-51</b>	4
2.1 การจัจัดขาลักษณะภายนอกของ MCS-51	4
2.2 การจัดการหน่วยความจำ	5
2.3 ออสซิลเลเตอร์และวงจรมอนิเตอร์	5
<b>บทที่ 3 การออกแบบโครงการและการสร้าง</b>	7
3.1 ภาคตรวจจับสัญญาณแสง	7
3.2 ภาคควบคุมมอเตอร์ล้อเลื่อน	8
3.3 ภาคควบคุมมอเตอร์เซ็นเซอร์	10
3.4 ภาคตรวจจับสิ่งกีดขวาง	10
3.5 โปรแกรมเริ่มต้นเมื่อเปิดเครื่อง	13
3.6 โปรแกรมหลัก	14
3.7 โปรแกรมเคลื่อนที่ไปตามแสงทางซ้าย	15
3.8 โปรแกรมเคลื่อนที่ไปตามแสงทางขวา	16
3.9 โปรแกรมเคลื่อนที่ไปข้างหน้า	17
3.10 โปรแกรมอินเทอร์รัพท์ใช้งานร่วมกับโปรแกรมเคลื่อนที่ไปข้างหน้า	18

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.11 โปรแกรมตรวจจับสิ่งกีดขวาง	19
3.12 โปรแกรมตรวจสอบระดับแสง โดยรอบเก็บไว้เป็นค่าอ้างอิง	20
3.13 โปรแกรมควบคุมความเร็วมอเตอร์	21
<b>บทที่ 4 ผลการทดลอง</b>	22
4.1 ภาคตรวจจับสัญญาณแสง	22
4.2 ภาคควบคุมมอเตอร์สี่ล้อ	22
4.3 ภาคควบคุมมอเตอร์เซ็นเซอร์	23
4.4 สรุปผลการทดลอง	23
<b>หนังสืออ้างอิง</b>	24
<b>ภาคผนวก ก โปรแกรมหุ่นยนต์แมลง</b>	25
<b>ภาคผนวก ข DATA SHEET</b>	45



## สารบัญรูป

รูปที่ 1.1	Block diagram แสดงการทำงานของหุ่นยนต์แมลง	2
รูปที่ 2.1	ลักษณะการจัดขาภายนอกของ MCS-51	5
รูปที่ 2.2	วงจรรอสซิลเลเตอร์ภายใน MCS-51	6
รูปที่ 2.3	การใช้วงจรรอสซิลเลเตอร์บน HMOS ชิป	6
รูปที่ 2.4	การจ่ายสัญญาณนาฬิกาภายนอกในการขับ HMOS MCS-51	7
รูปที่ 3.1	วงจรรภาคตรวจจับสัญญาณแสง	8
รูปที่ 3.2	วงจรรภาคควบคุมมอเตอร์ล้อเลื่อน	9
รูปที่ 3.3	แสดงจุดการควบคุมทิศทางการหมุนของมอเตอร์	10
รูปที่ 3.4	ลักษณะของพัลส์ที่ใช้ควบคุมความเร็วมอเตอร์	10
รูปที่ 3.5	วงจรรภาคควบคุมมอเตอร์เซ็นเซอร์	11
รูปที่ 3.6	วงจรรภาคตรวจจับสิ่งกีดขวาง	12
รูปที่ 3.7	วงจรรหุ่นยนต์แมลง	13

# บทที่ 1

## บทนำ

ในปัจจุบันวิวัฒนาการของไมโครโปรเซสเซอร์ได้มีการเปลี่ยนแปลงไปอย่างสูงและรวดเร็ว ได้มีการเพิ่มความสามารถในการประมวลผลของซีพียู ให้ชุดคำสั่งน้อยลงและสามารถทำงานได้หลายคำสั่งภายในหนึ่งวัฏจักร ซึ่งก็จะเป็นการเพิ่มความเร็วในการประมวลผลตามคำสั่งโปรแกรมด้วย เทคโนโลยีทางไมโครโปรเซสเซอร์ยังมีอีกแขนงหนึ่งที่ได้ออกแบบมาเพื่อจุดมุ่งหมายที่จะให้ชิปไมโครโปรเซสเซอร์มีความสามารถสูงทางด้านการควบคุมนั่นก็คือ ไมโครคอนโทรลเลอร์ (Microcontroller) ไมโครคอนโทรลเลอร์ได้มีการรวมเอาสิ่งที่เป็นของไมโครคอมพิวเตอร์ให้เข้ามาอยู่บนชิปเดียวกัน เพื่อที่จะให้เหมาะกับการใช้งานทางด้านระบบการควบคุมงานอัตโนมัติต่างๆ ได้ และสามารถที่จะทำงานด้วยชิปเพียงตัวเดียวด้วยตัวมันเองได้ โดยใช้อุปกรณ์อื่นๆ ภายนอกมาต่อกับชิปตัวนี้ให้น้อยที่สุด

### 1.1 วัตถุประสงค์

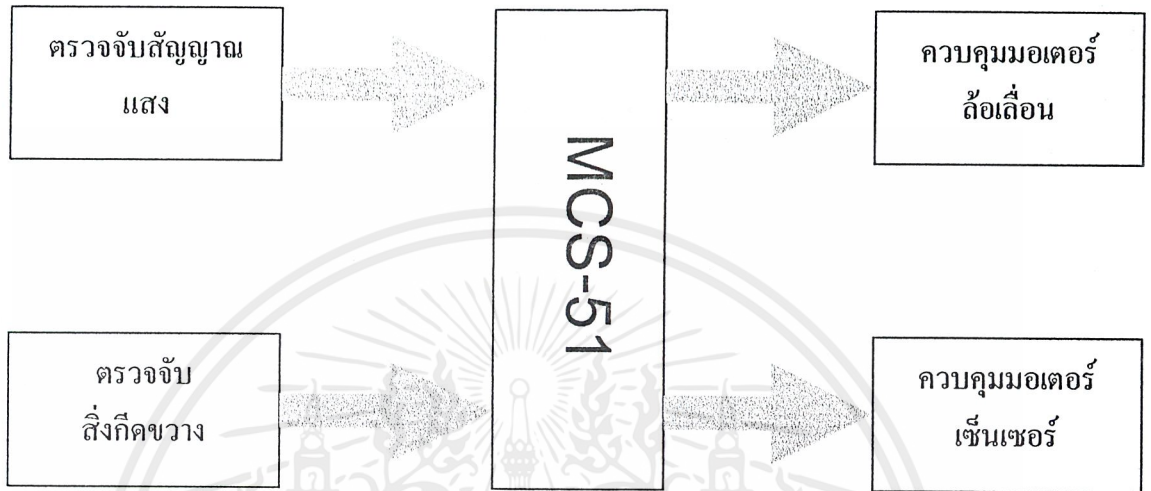
1. เพื่อศึกษาและเข้าใจการทำงานของไมโครคอนโทรลเลอร์
2. เพื่อออกแบบและควบคุมอุปกรณ์อื่นให้สามารถทำงานได้ตามที่ต้องการ
3. สามารถที่จะนำเอามาพัฒนาเพื่อใช้ในงานต่างๆ ได้ต่อไป

### 1.2 ขอบเขตของโครงการ

1. หุ่นยนต์สามารถเคลื่อนที่ไปในทิศทางของจุดกำเนิดแสงได้
2. การเคลื่อนที่ของหุ่นยนต์จะเป็นไปในลักษณะของล้อเลื่อน
3. หุ่นยนต์สามารถที่จะเคลื่อนที่หลบหลีกสิ่งกีดขวางได้อย่างอัตโนมัติ

### 1.3 วิธีการดำเนินงาน

การทำงานของหุ่นยนต์แมลงสามารถแสดงได้ดังรูปที่ 1.1



รูปที่ 1.1 Block diagram แสดงการทำงานของหุ่นยนต์แมลง

จาก Block diagram สามารถที่จะดำเนินงานได้ตามขั้นตอนดังนี้

1. ศึกษารายละเอียดของระบบการทำงานของ MCS-51
2. ออกแบบวงจรและทดลองในรายละเอียดของชุดควบคุมมอเตอร์ไฟตรง ชุดตรวจสอบสัญญาณแสง
3. เขียนโปรแกรมเพื่อตรวจสอบสัญญาณแสง และสั่งงานควบคุมให้มอเตอร์ทำงานได้
4. ออกแบบวงจรชุดควบคุมการตรวจสอบทิศทางของแสง
5. เขียนโปรแกรมเพื่อควบคุมวงจรชุดควบคุมการตรวจสอบทิศทางของแสง
6. ทดสอบการทำงานของหุ่นยนต์ให้สามารถเคลื่อนที่ไปยังทิศทางของจุดกำเนิดแสง
7. ศึกษารายละเอียดของวัตถุหรือสิ่งกีดขวาง
8. ออกแบบวงจรชุดตรวจสอบวัตถุหรือสิ่งกีดขวาง
9. เขียนโปรแกรมเพื่อตรวจสอบวัตถุหรือสิ่งกีดขวาง และควบคุมให้หุ่นยนต์เคลื่อนที่หลบหลีกวัตถุหรือสิ่งกีดขวางนั้นเพื่อไปยังจุดกำเนิดแสง
10. สร้างแผ่นวงจรพิมพ์สมบูรณ์ ลงอุปกรณ์ และทดสอบการทำงาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

11. สรุปการทำโครงการ ข้อเสนอแนะและปัญหาในการทำงาน งานการเขียน พิมพ์ ตรวจสอบแก้ไข และเสนอรายงาน

#### 1.4 ประโยชน์ที่ได้รับจากการทำโครงการ

ประโยชน์โดยตรงที่ได้รับจากโครงการนี้คงจะไม่มี แต่ข้อดีของการคิดทำโครงการนี้ขึ้นมาก็คือ

- ทำให้ผู้สร้างได้แสดงความคิดสร้างสรรค์ออกมาได้อย่างเต็มที่
- อาจใช้เป็นเครื่องค้นแบบเพื่อนำมาพัฒนาให้ได้ประโยชน์อื่นๆ ต่อไป
- สามารถสร้างสรรค์สิ่งแปลกๆ ใหม่ออกมาใช้ประโยชน์กันต่อไป



## บทที่ 2

### MCS-51

ตระกูล MCS-51 ได้ถูกออกแบบมาเพื่อใช้ในอุตสาหกรรมมาตรฐานขนาด 8 บิตไมโครคอนโทรลเลอร์ และให้มีความสามารถในการงานควบคุมประยุกต์ใช้งานในเรื่อง sequential real time control, close loop control และ data control อุปกรณ์ที่เพิ่มขึ้นตามลักษณะหลักๆ ไปต่อไปนี้

1. ใช้ HMOS และ CHMOS เทคโนโลยีในการสร้างและทำงานด้วยแหล่งจ่ายไฟขนาด 5 V. เพียงแหล่งเดียว
2. ซีพียูมีขนาดคำ 8 บิต
3. มีวงจรรอสซิลเลเตอร์ และวงจรรนาฬิกาบนชิป
4. ชุดแบงก์ (BANK) รีจิสเตอร์มี 4 ชุด แต่ละชุดมีรีจิสเตอร์ 8 ตัว
5. มีตัวจับเวลา/ตัวนับ ขนาด 16 บิต 2 ชุด และสำหรับเบอร์ 8032/8052 มี 3 ชุด
6. มีพอร์ตไอโอแบบขนานสองทิศทางจำนวน 4 พอร์ต พอร์ตละ 8 บิตรวมทั้งหมดเป็น 32 เส้น แต่จะเหลือเพียง 16 เส้น สำหรับเบอร์ 8031 อีก 16 เส้นจะใช้ในการเข้าถึงทางแอดเดรสและข้อมูล
7. พอร์ตแบบอนุกรมสามารถที่จะโปรแกรมการรับส่งแบบ Full Duplex ที่ความเร็วสูง
8. หนึ่งวัฏจักรคำสั่งจะกินเวลา 1 ไมโครวินาที ด้วยการใช้นคริตอล 12 เมกะเฮิร์ตซ์
9. แอดเดรสข้อมูลภายนอกได้ 64 กิโลไบต์
10. โครงสร้างอินเตอร์รัพต์จะติดตั้งได้ถึง 5 แห่งและ 6 แห่งสำหรับ 8032/8052 พร้อมด้วยการจัดไพรออริตี้ (Priority) ได้ 2 ระดับ
11. มีคำสั่งคูณและหารทางฮาร์ดแวร์ที่ทำได้ภายใน 4 ไมโครวินาที

ตระกูล MCS-51 จะมีทั้งแบบมี ROM ในตัวหรือไม่มี ROM หรือมี EPROM บนชิป เดียวกันและจะมีตำแหน่งขาที่เหมือนกัน

#### 2.1 การจัดขาลักษณะภายนอกของ MCS-51

รูปที่ 2.1 แสดงการจัดขาลักษณะภายนอกของชิป MCS-51 ซึ่งจะมีการแบ่งกลุ่มการจัดขาลักษณะตามสถาปัตยกรรมของ MCS-51 อยู่ 4 กลุ่มคือ

1. กลุ่มขารับแหล่งจ่ายไฟฟ้า และระบบสัญญาณนาฬิกา
2. กลุ่มขาแอดเดรสและข้อมูล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3. กลุ่มขาควบคุม
4. กลุ่มขาพอร์ตแบบขนานและอนุกรม

1	P1.0	VEE	40
2	P1.1	P0.0	39
3	P1.2	P0.1	38
4	P1.3	P0.2	37
5	P1.4	P0.3	36
6	P1.5	P0.4	35
7	P1.6	P0.5	34
8	P1.7	P0.6	33
9	VPDRST	P0.7	32
10	P3.0/CLKD	/EA	31
11	P3.1/XTXD	PICG/ALF	30
12	P3.2/INT0	FSEN	29
13	P3.3/INT1	P1.7	28
14	P3.4/T0	P1.6	27
15	P3.5/T1	P1.5	26
16	P3.6/WR	P1.4	25
17	P3.7/RS	P1.3	24
18	X1	P1.2	23
19	X2	P1.1	22
20	GND	P1.0	21

MCS-51

รูปที่ 2.1 ลักษณะการจัดขาภายนอกของ MCS-51

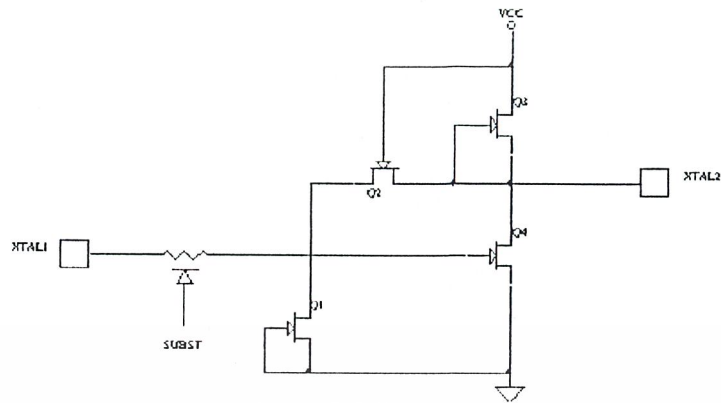
## 2.2 การจัดการหน่วยความจำ

ตัว MCS-51 จะแยกแอดเดรสสำหรับหน่วยความจำของโปรแกรม และหน่วยความจำของข้อมูลออกจากกัน หน่วยความจำของโปรแกรมขยายได้ถึง 64 กิโลไบต์ และจำนวนไบต์ค่า 4 กิโลไบต์จะอยู่ใน 8051 หน่วยความจำของข้อมูลภายในมี 128 ไบต์ (256 ไบต์สำหรับ 8032/8052) บนชิป และอีก 128 ไบต์ใช้สำหรับรีจิสเตอร์ฟังก์ชันพิเศษ (Special Function Register : SFR) และเข้าถึงหน่วยความจำข้อมูลภายนอกได้อีก 64 กิโลไบต์

## 2.3 ออสซิลเลเตอร์และวงจรรักษาฬิกา

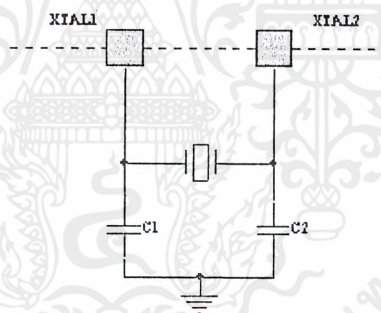
วงจรรออสซิลเลเตอร์ที่อยู่ชิปแบบ HMOS ของตระกูล MCS-51 จะเป็น Single Linear Invertor ดังแสดงในรูปที่ 2.2 ใช้คริสตอลควบคุมเป็นออสซิลเลเตอร์แลตริแอกทีฟบวกดังรูปที่ 2.3 ในการใช้งานคริสตอลนี้จะทำงานที่โหมด Fundamental เสมือนเป็น Inductance โดยต่อขนานอยู่กับตัวคาปาซิเตอร์ภายนอกที่ต่อกับคริสตอล การกำหนดตัวคริสตอลและค่าคาปาซิเตอร์ C1 และ C2 ในรูปที่ 2.3 ไม่ค่อยวิกฤตนักอาจจะมีค่าประมาณ 30 pF สำหรับทุกความถี่ของตัวคริสตอลชนิดนี้ ส่วนการใช้ Ceramic Resonator ค่าคาปาซิเตอร์ที่มาก่อจะมีค่าสูงกว่า โดยมีค่าประมาณ 47 pF การใช้ค่าคาปาซิเตอร์อาจเปลี่ยน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

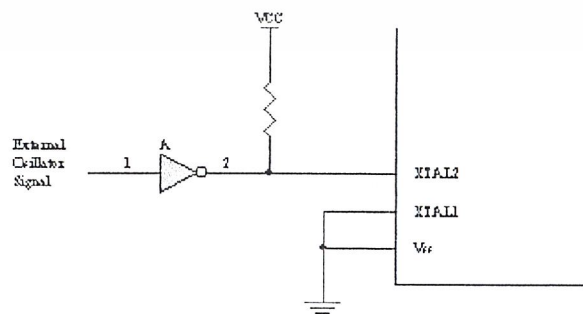


รูปที่ 2.2 วงจรออสซิลเลเตอร์ภายใน MCS-51

แปลงได้ขึ้นอยู่กับตัว Ceramic Resonator นั้น ๆ การจับตัว H-MOS ด้วยสัญญาณนาฬิกา จากภายนอกก็กระทำได้เช่นกันโดยต่อเข้าที่ขา XTAL2 และต่อลงดินที่ขา XTAL1 ดังรูปที่ 2.4 ควรจะใช้ตัวต้านทานพูลอัพเพราะที่ XTAL2 ต้องการสัญญาณในระดับที่ทีแอล



รูปที่ 2.3 การใช้วงจรออสซิลเลเตอร์บน H-MOS ชิป



รูปที่ 2.4 การจ่ายสัญญาณนาฬิกาภายนอกในการจับ H-MOS MCS-51

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

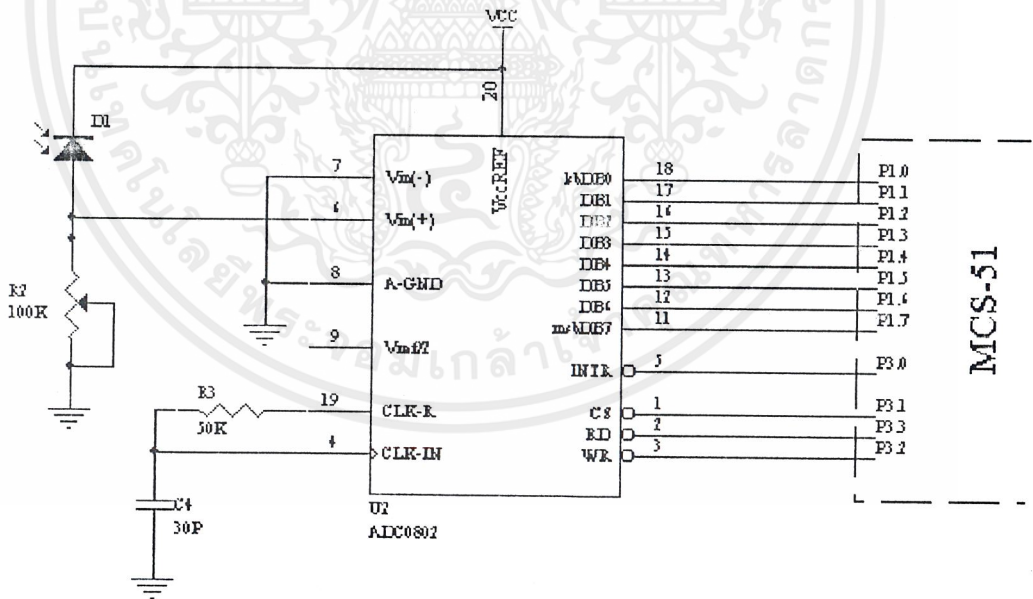
### บทที่ 3

#### การออกแบบโครงงานและการสร้าง

ในการออกแบบโครงงานนี้จะแบ่งออกเป็น 2 ส่วนหลักๆ คือ ส่วนของฮาร์ดแวร์ซึ่งจะประกอบไปด้วย 4 ภาคที่สำคัญ คือ ภาคตรวจจับสัญญาณแสง , ภาคตรวจจับสิ่งกีดขวาง , ภาคควบคุมมอเตอร์ล้อเลื่อน และภาคควบคุมมอเตอร์เซ็นเซอร์ ซึ่งแต่ละภาคจะกล่าวในหัวข้อถัดไป และอีกส่วนก็คือ ส่วนของซอฟต์แวร์ ซึ่งจะใช้ในการควบคุมส่วนของฮาร์ดแวร์แต่ละภาค

#### 3.1 ภาคตรวจจับสัญญาณแสง

วงจรส่วนนี้เป็นวงจรในการตรวจจับสัญญาณแสงโดยจะใช้โฟโตไดโอดในการตรวจจับสัญญาณแสงแล้วเปลี่ยนให้อยู่ในรูปของแรงดันไฟฟ้า จากนั้นใช้ไอซีอนุบาลอกทูดิจิตอลคอนเวอร์เตอร์ (ADC0802) เป็นตัวแปลงระดับของแรงดันไฟฟ้าให้อยู่ในรูปของดิจิตอล เพื่อให้สามารถนำมาใช้งานร่วมกับระบบไมโครคอนโทรลเลอร์ได้ดังรูปที่ 3.1

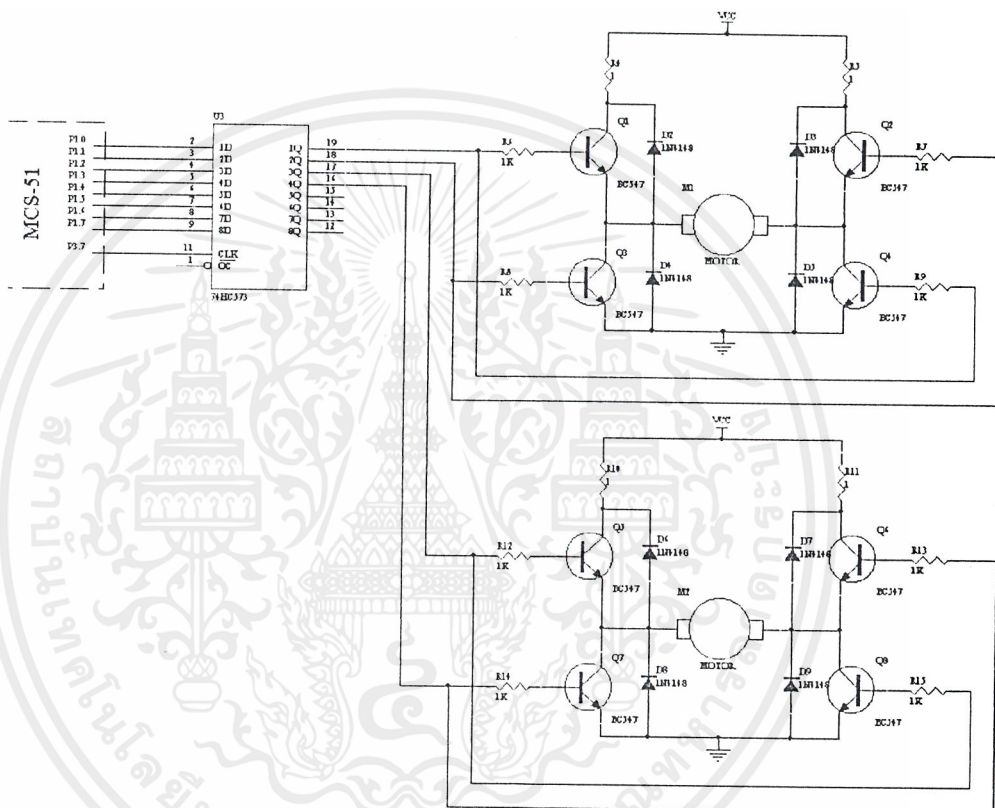


รูปที่ 3.1 วงจรภาคตรวจจับสัญญาณแสง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 3.2 ภาคควบคุมมอเตอร์ล้อเลื่อน

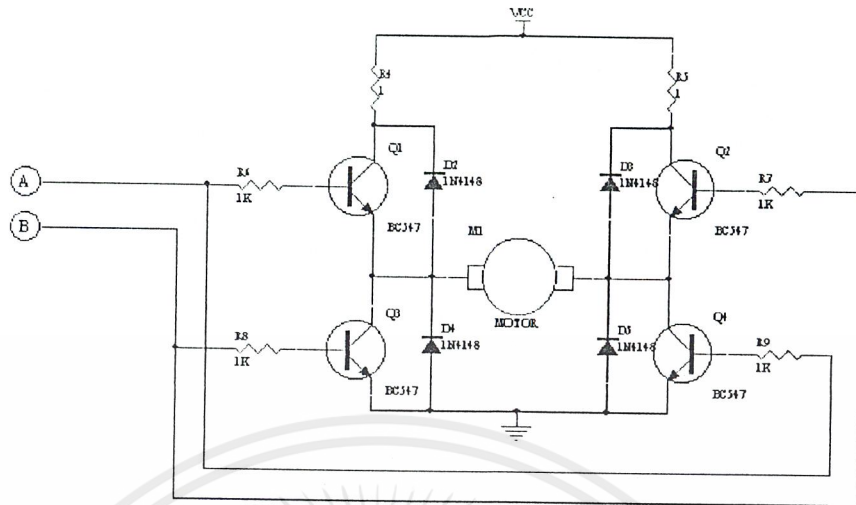
วงจรในภาคนี้เป็นวงจรที่จะใช้ในการควบคุมมอเตอร์ไฟตรงขนาดเล็ก 5 V ให้สามารถทำงานหมุนไปกลับได้ 2 ทิศทางและสามารถที่จะควบคุมให้หมุนช้าหรือเร็วได้ การควบคุมนั้นจะถูกควบคุมโดยไมโครคอนโทรลเลอร์ผ่านไอซีขับเฟอ์ (74HC573) เพื่อใช้เป็นที่พักข้อมูลของไมโครคอนโทรลเลอร์ ดังรูปที่ 3.2



รูปที่ 3.2 วงจรภาคควบคุมมอเตอร์ล้อเลื่อน

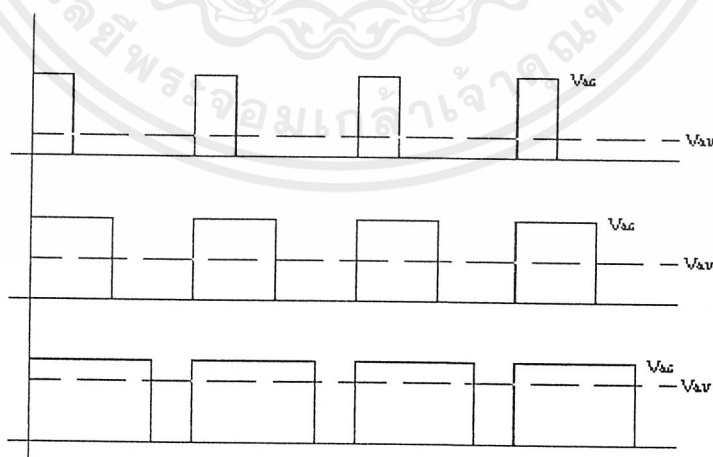
วงจรที่ใช้ควบคุมมอเตอร์จะเป็นแบบ H-Bridge ซึ่งสามารถควบคุมให้กระแสไหลผ่านมอเตอร์ได้ 2 ทิศทาง โดยการควบคุมนั้นจะใช้ค่าระดับแรงดันไฟฟ้า +5 โวลต์จ่ายให้กับแต่ละจุด ก็จะทำให้กระแสไหลผ่านตัวต้านทาน 1 กิโลโอห์มทั้งสองตัวทรานซิสเตอร์ทั้งสองตัวก็จะทำงานจ่ายกระแสให้กับมอเตอร์หมุนในทิศทางหนึ่ง หากต้องการให้มอเตอร์หมุนกลับทิศทางก็สามารถทำได้โดยการจ่ายแรงดันไฟฟ้า +5 โวลต์ ให้กับอีกจุดหนึ่งโดยจะต้องหยุดจ่ายแรงดันไฟฟ้าให้กับอีกจุดหนึ่งก่อนเสมอ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.3 แสดงจุดการควบคุมทิศทางการหมุนของมอเตอร์

ส่วนการควบคุมความเร็วการหมุนของมอเตอร์นั้นก็สามารถทำได้โดยการควบคุมกระแสที่จะจ่ายให้มอเตอร์ หากต้องการให้มอเตอร์หมุนเร็วขึ้นก็ทำการจ่ายกระแสให้มอเตอร์เพิ่มขึ้น แต่จากวงจรจะเห็นว่าเราไม่สามารถควบคุมการจ่ายกระแสได้โดยตรง ดังนั้นจึงทำการจ่ายกระแสในลักษณะของพัลส์และควบคุมความกว้างของพัลส์ในช่วงระดับแรงดัน +5 โวลต์ หากความกว้างของระดับแรงดันมีค่ามาก ค่ากระแสเฉลี่ยก็จะมีค่าสูงขึ้นทำให้มอเตอร์หมุนเร็วขึ้น แต่ถ้าความกว้างของระดับแรงดันลดลงค่ากระแสเฉลี่ยก็จะมีค่าลดลงทำให้มอเตอร์หมุนช้าลง

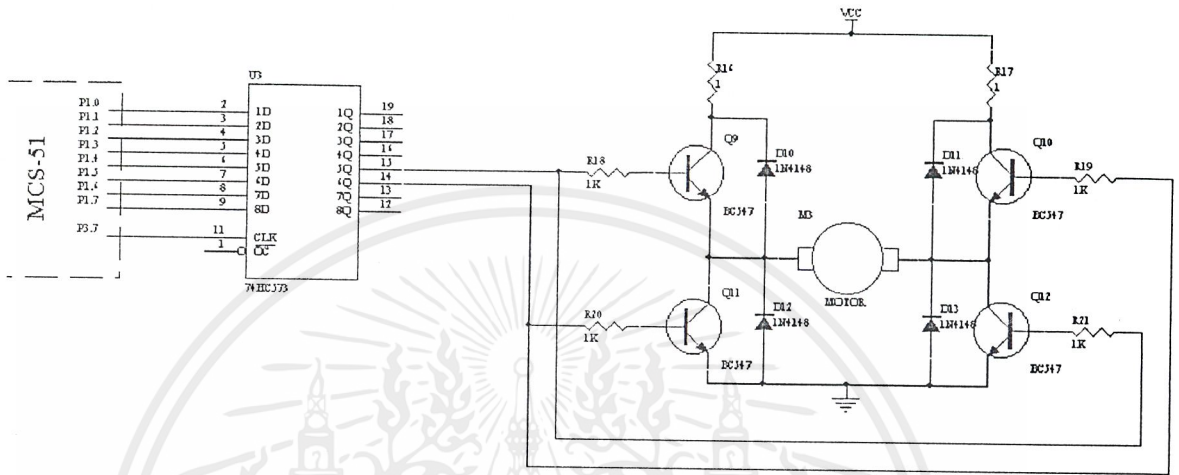


รูปที่ 3.4 ลักษณะของพัลส์ที่ใช้ควบคุมความเร็วมอเตอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 3.3 ภาคควบคุมมอเตอร์เซ็นเซอร์

วงจรในการควบคุมมอเตอร์เซ็นเซอร์จะเหมือนกับภาคควบคุมมอเตอร์ล้อเลื่อนจะแตกต่างกันก็แต่เพียงจุดต่อควบคุมจากไมโครคอนโทรลเลอร์ดังรูปที่ 3.3

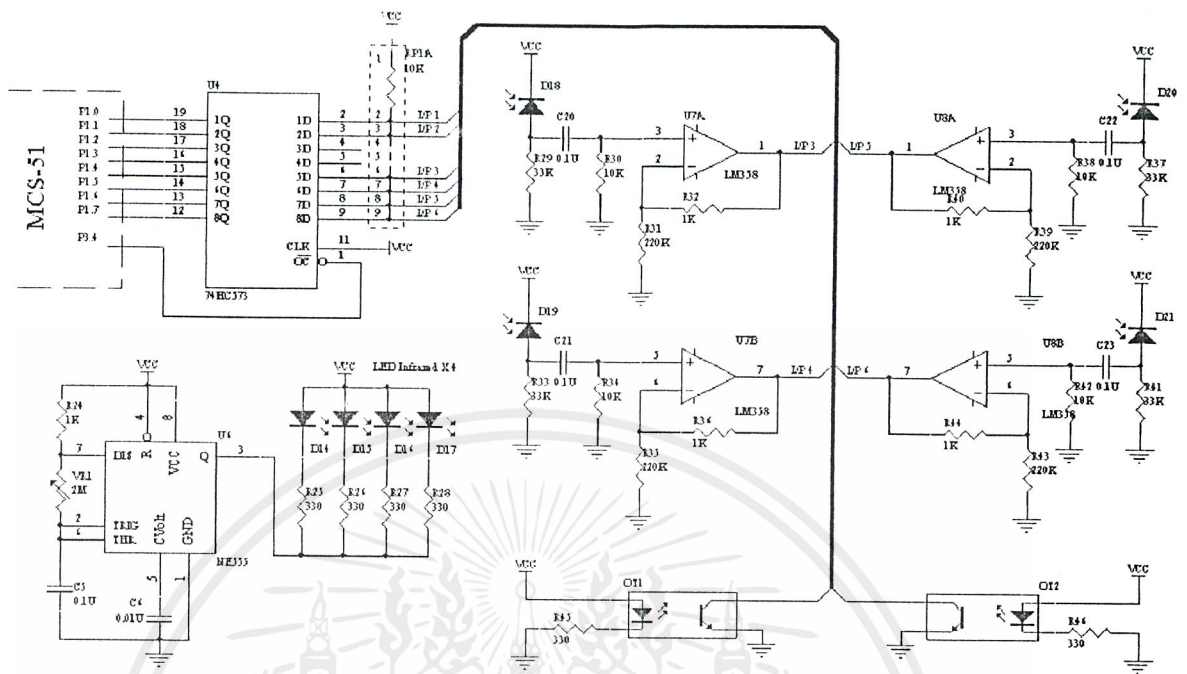


รูปที่ 3.5 วงจรภาคควบคุมมอเตอร์เซ็นเซอร์

### 3.4 ภาคตรวจจับสิ่งกีดขวาง

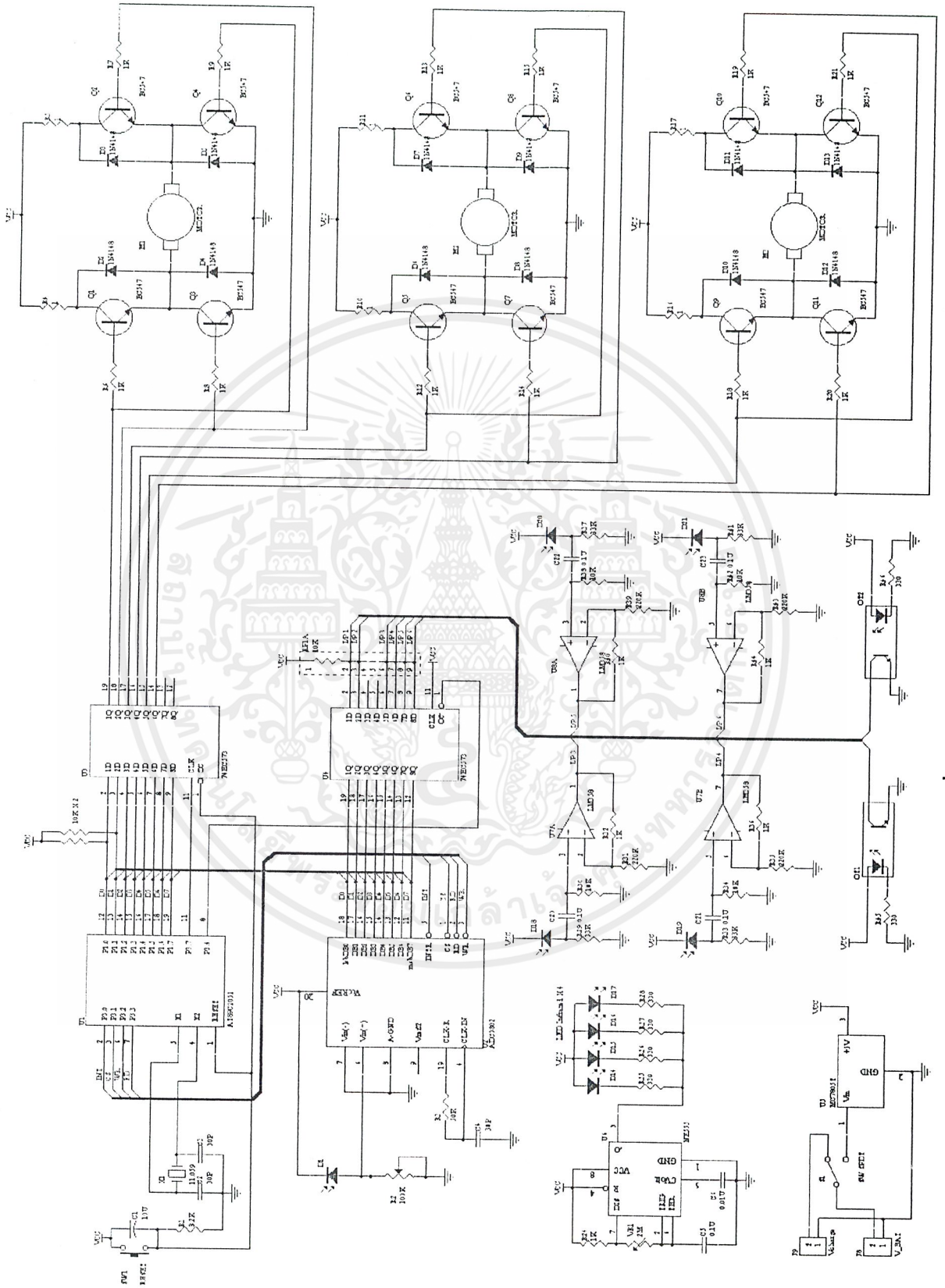
วงจรภาคตรวจจับสิ่งกีดขวางจะใช้วิธีการตรวจจับโดยใช้อินฟราเรดส่งสัญญาณออกไปและรับสัญญาณที่สะท้อนกลับมาหากพบสิ่งกีดขวาง

จากรูปจะใช้ไอซี 555 กำเนิดสัญญาณพัลส์ความถี่ประมาณ 5 กิโลเฮิร์ตซ์ ส่งไปยังตัวส่งอินฟราเรดแต่ละตัว ตัวส่งอินฟราเรดก็จะส่งสัญญาณอินฟราเรดออกไปเพื่อตรวจหาสิ่งกีดขวางข้างหน้า หากมีสิ่งกีดขวางอยู่ข้างหน้าก็จะเกิดการสะท้อนกลับของสัญญาณที่ส่งออกไปกลับมายังตัวรับอินฟราเรด สัญญาณที่ได้นี้จะถูกนำมาขยายให้สูงขึ้น โดยใช้ไอซีออปแอมป์เพื่อให้ระดับสัญญาณมีค่าสูงพอที่จะนำไปใช้กับระบบไมโครคอนโทรลเลอร์ สัญญาณที่ขยายแล้วนั้นจะถูกส่งไปยังไมโครคอนโทรลเลอร์ เพื่อนำไปคำนวณ เปรียบเทียบและตรวจสอบความถูกต้อง



รูปที่ 3.6 วงจรภาคตรวจจับสิ่งกีดขวาง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.4 วงจรหาคอนตัมแมต

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้拿去ไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

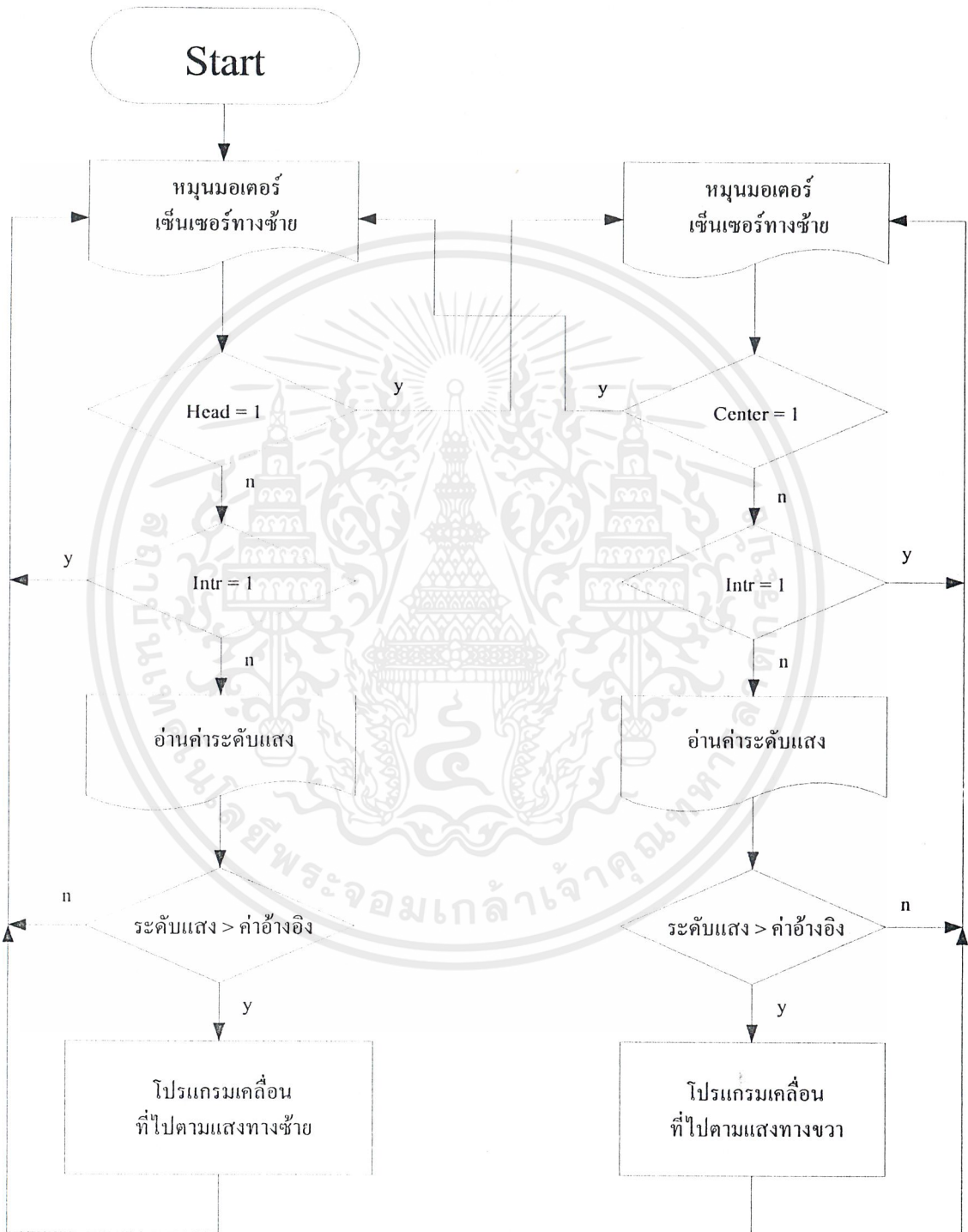
Flow chart แสดงการทำงานของโปรแกรมในส่วนต่าง ๆ

### 3.5 โปรแกรมเริ่มต้นเมื่อเปิดเครื่อง



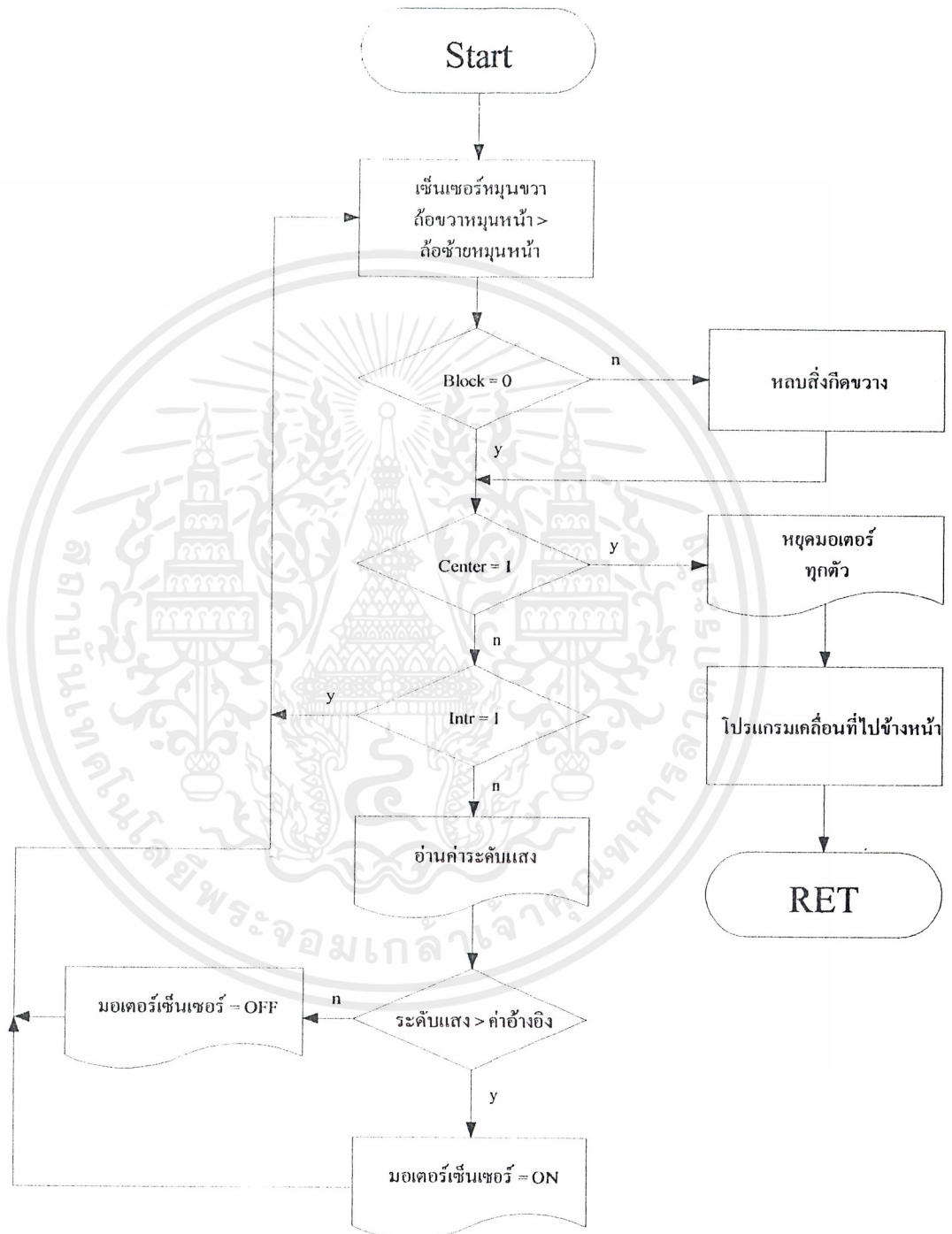
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## 3.6 โปรแกรมหลัก



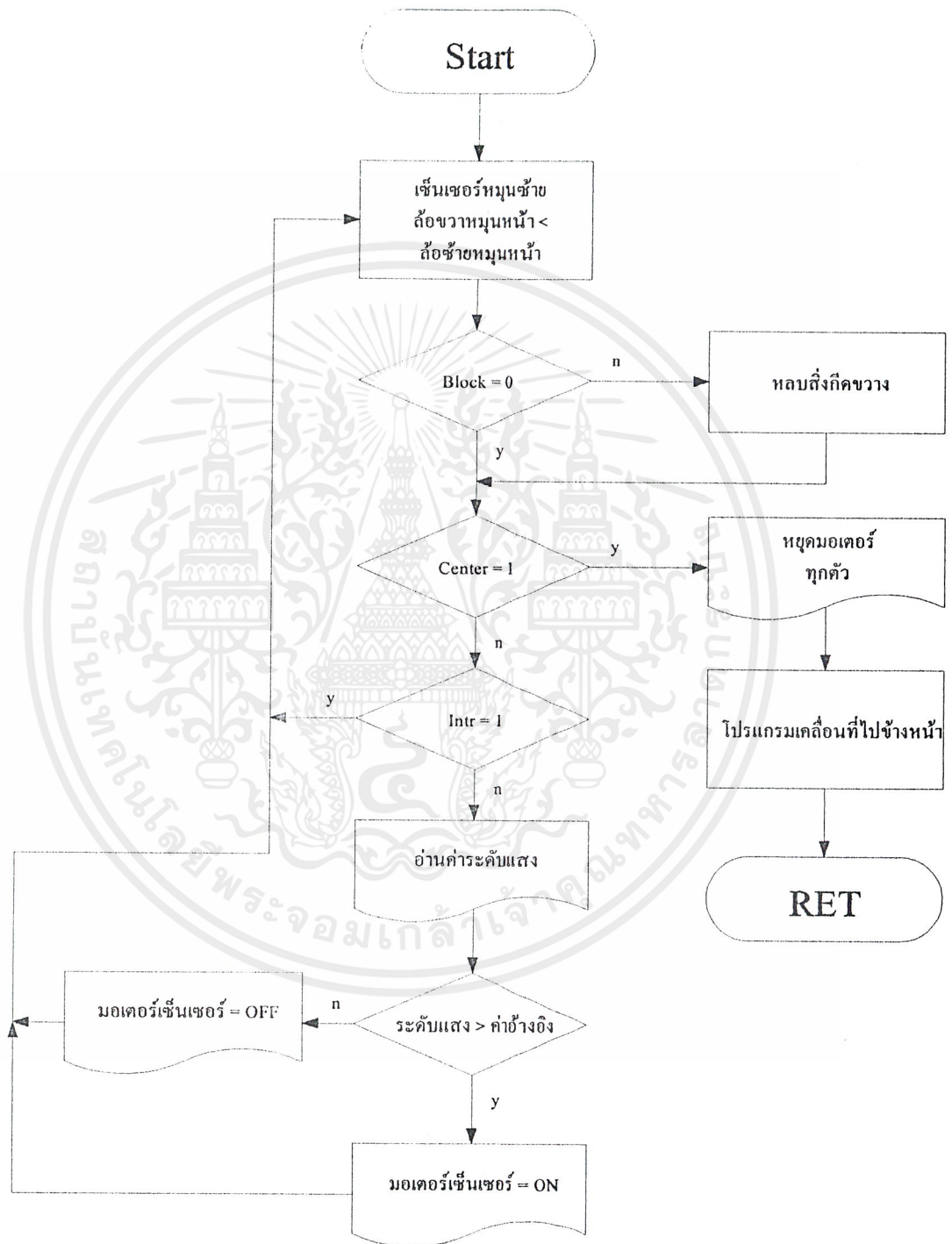
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 3.7 โปรแกรมเคลื่อนที่ไปตามแสงทางซ้าย



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

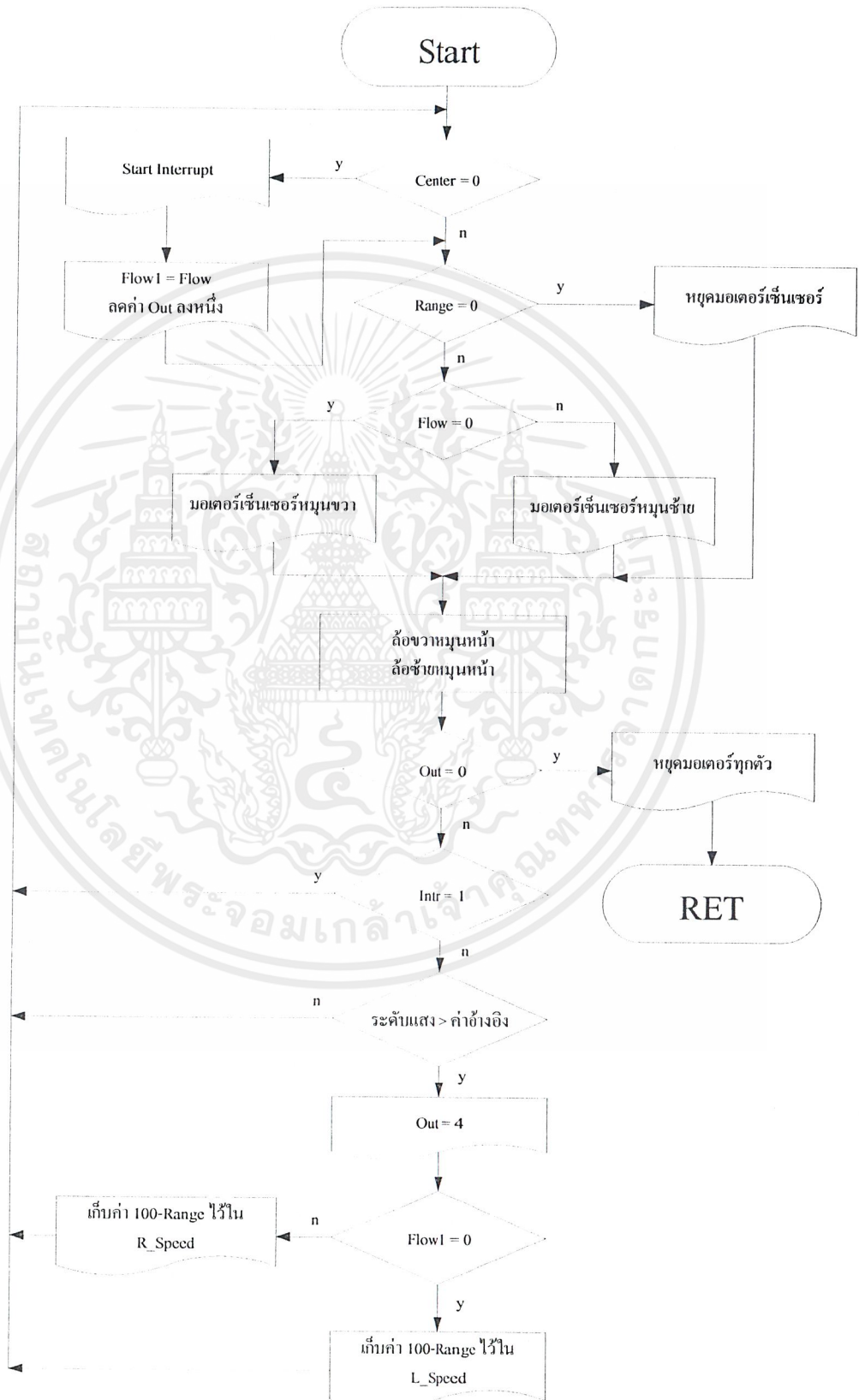
### 3.8 โปรแกรมเคลื่อนที่ไปตามแสงทางขวา



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

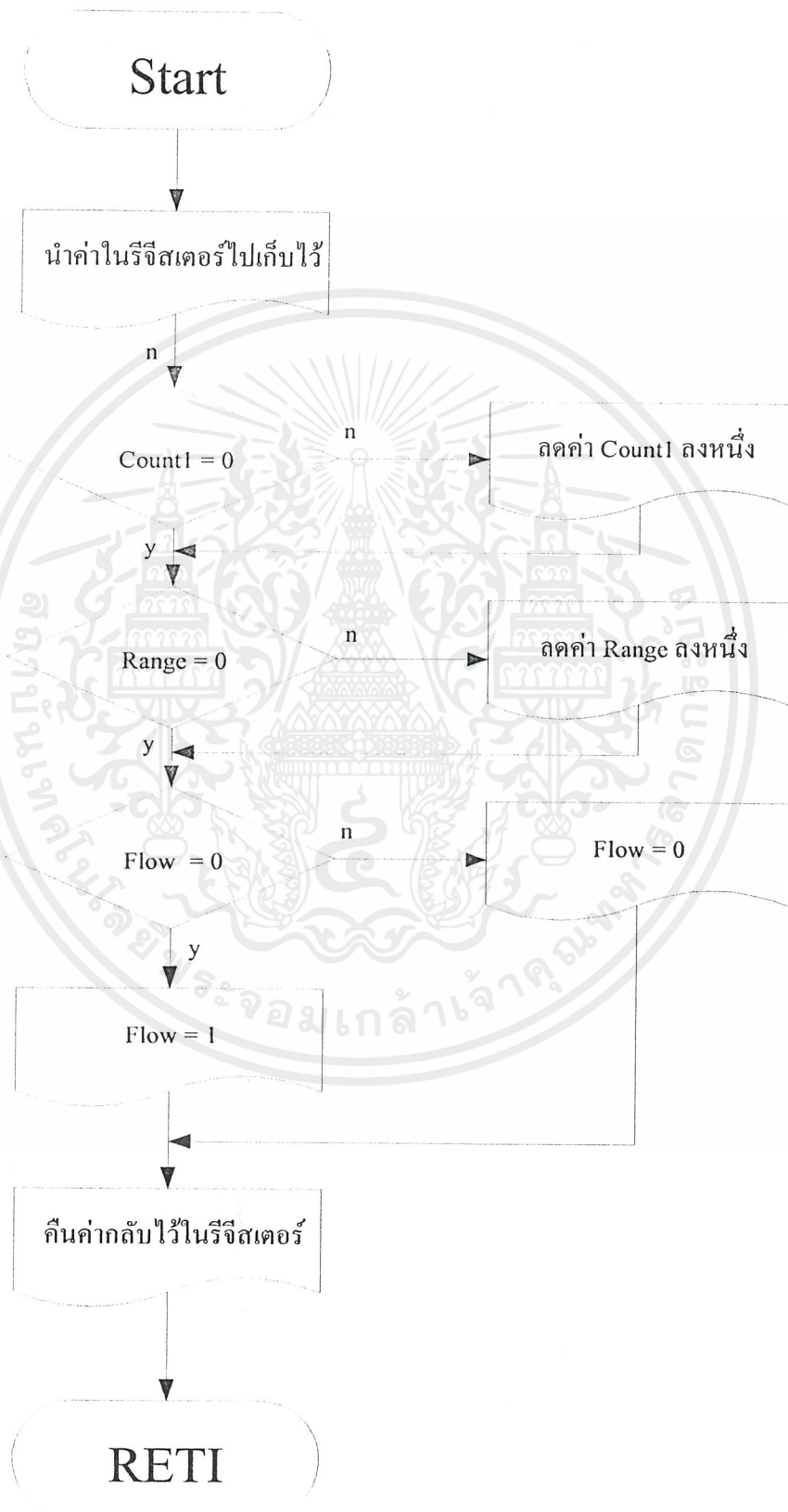
## สำนักหอสมุดกลาง พระจอมเกล้าลาดกระบัง

### 3.9 โปรแกรมเคลื่อนที่ไปข้างหน้า



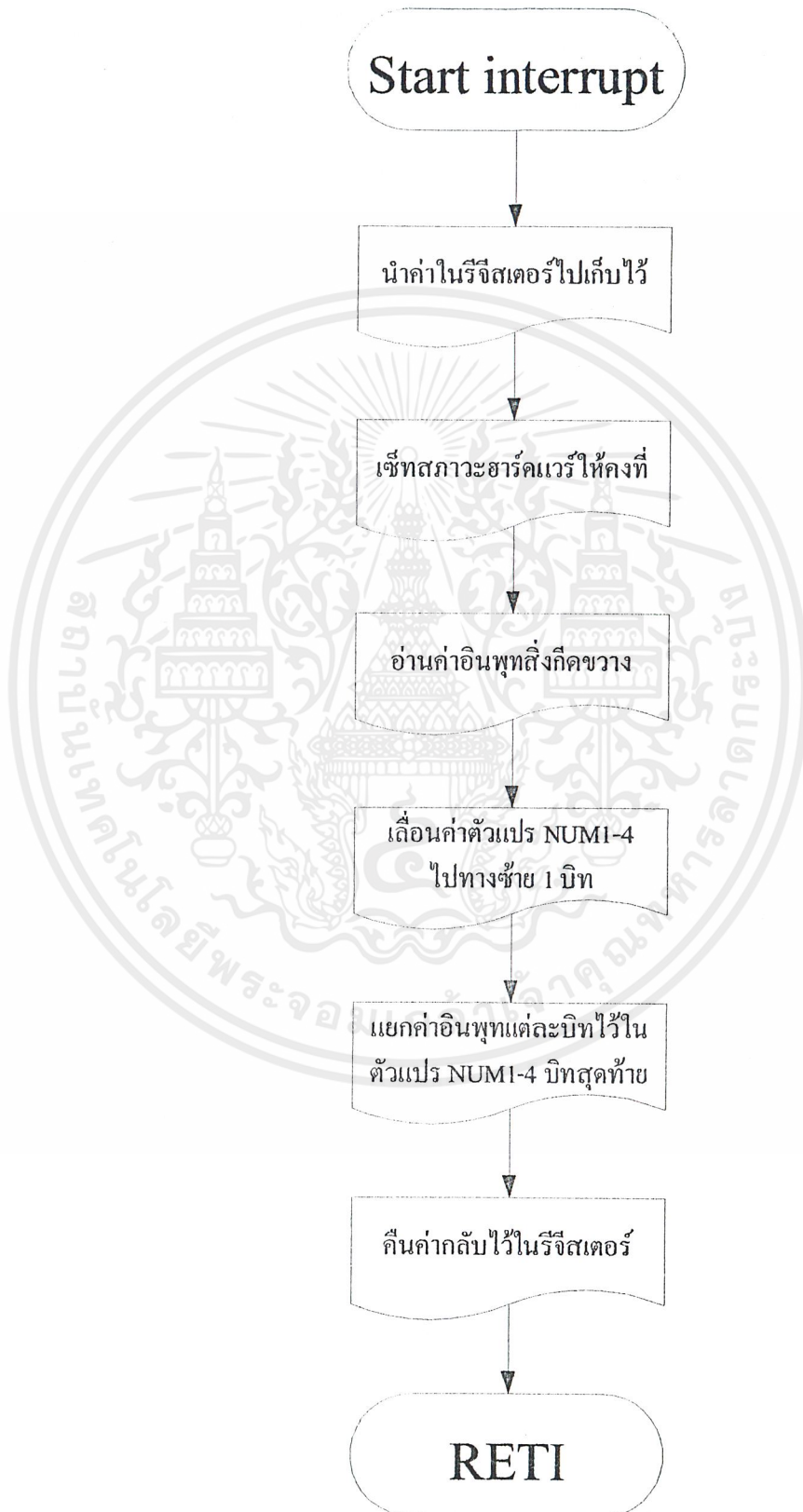
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 3.10 โปรแกรมอินเทอร์รัพท์ใช้งานร่วมกับโปรแกรมเคลื่อนที่ไปข้างหน้า



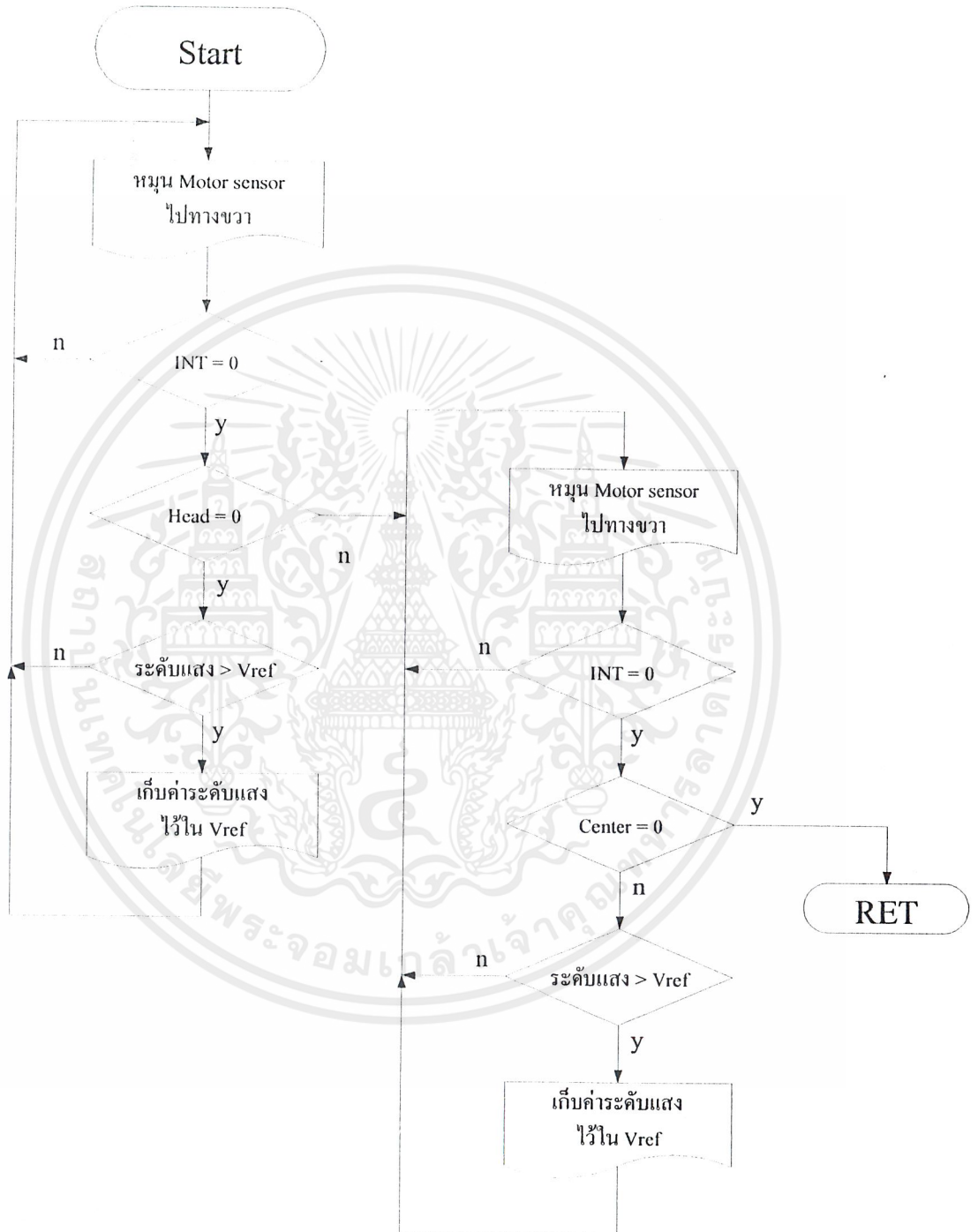
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 3.11 โปรแกรมตรวจจับสิ่งกีดขวาง



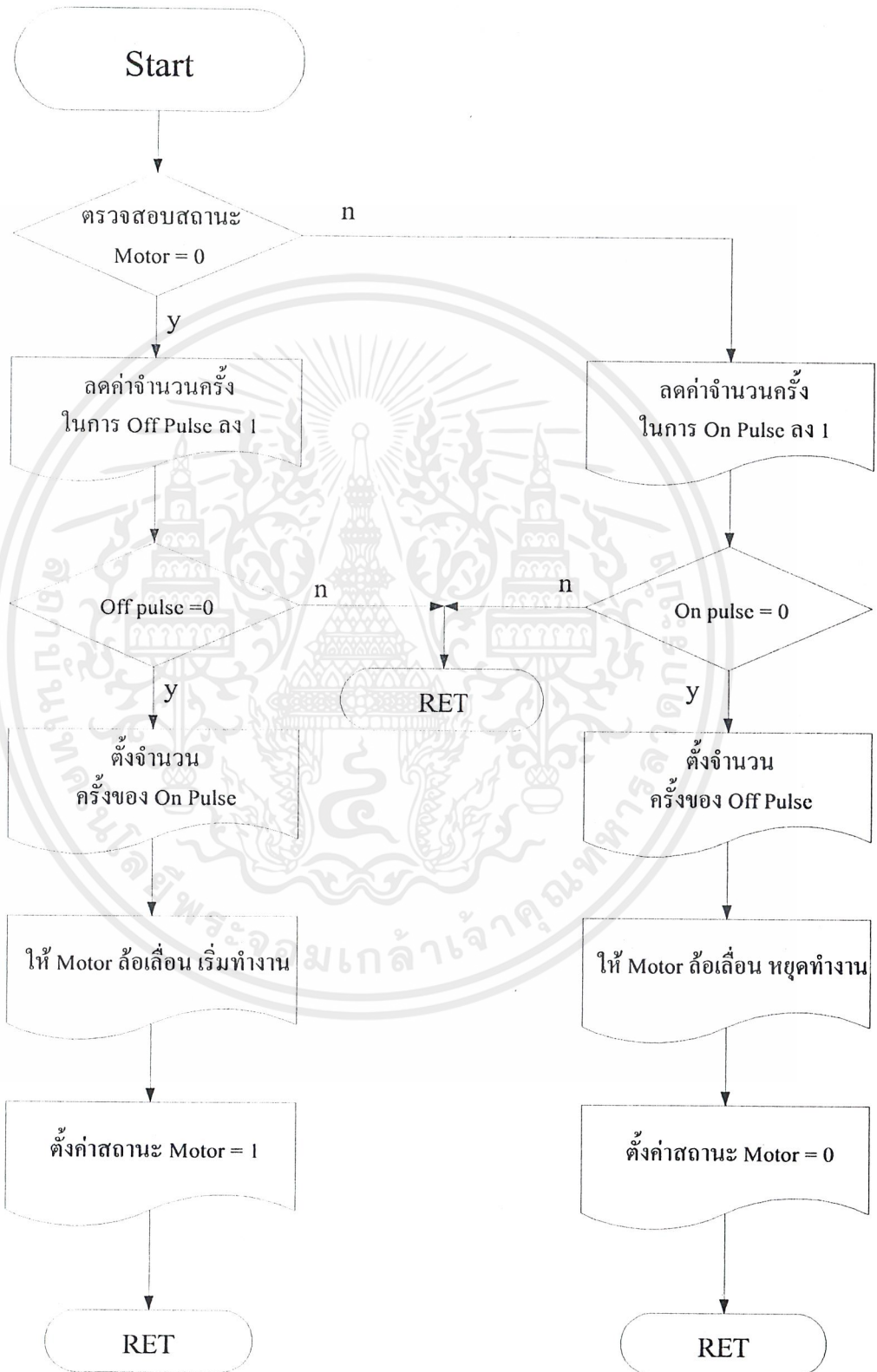
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 3.12 โปรแกรมตรวจสอบระดับแสงโดยรอบเก็บไว้เป็นค่าอ้างอิง



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## 3.13 โปรแกรมควบคุมความเร็วมอเตอร์



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## บทที่ 4

### ผลการทดลอง

ในบทนี้จะกล่าวถึงผลการทดลองของการทำงานแต่ละส่วน เพื่อเป็นแนวทางการปรับปรุงแก้ไขโครงการซึ่งพอจะสรุปได้เป็นภาค ๆ ดังนี้

#### 4.1 ภาคตรวจจับสัญญาณแสง

การทำงานทางด้านฮาร์ดแวร์สามารถที่จะทำงานได้ แต่ว่าระดับแรงดันที่ได้จากตัวโฟโตไดโอดจะมีการเปลี่ยนแปลงที่ไม่ค่อยสม่ำเสมอคือ ในช่วงต้นๆ เมื่อได้รับแสงเล็กน้อยระดับแรงดันที่ได้จากตัวโฟโตไดโอดจะมีการเปลี่ยนแปลงค่อนข้างมาก และเมื่อได้รับแสงมากขึ้นระดับแรงดันที่ได้ก็จะมีการเปลี่ยนแปลงเพียงเล็กน้อย ดังตารางที่ 4.1

ระยะทาง	5 เมตร	4 เมตร	3 เมตร	2 เมตร	1 เมตร	0.5 เมตร	0.25 เมตร
ระดับแรงดัน	2.3 V	3.1 V	3.8 V	4.4 V	4.8 V	5 V	5 V

ตารางที่ 4.1 แสดงค่าระดับแรงดันกับระยะห่างของจุดกำเนิดแสง

ส่วนของซอฟต์แวร์ที่ใช้ในการควบคุมการทำงานของฮาร์ดแวร์สามารถทำงานได้ถูกต้องตามความต้องการและทำงานได้เป็นอย่างดี คือสามารถที่จะควบคุมให้ไอซีทำการอ่านค่าระดับแรงดันจากโฟโตไดโอดแล้วแปลงเป็นค่าดิจิทัล จากนั้นก็ทำการอ่านค่าดิจิทัลจากตัวไอซีมาไว้ในตัวไมโครคอนโทรลเลอร์เพื่อที่จะนำไปใช้งานต่อ ๆ ไป

#### 4.2 ภาคควบคุมมอเตอร์ล้อเลื่อน

มอเตอร์ที่ใช้งานนั้นเป็นมอเตอร์กระแสตรงขนาดเล็กที่มีการทลเฟืองภายในเพื่อช่วยในเรื่องของแรงบิดให้มากยิ่งขึ้น ดังนั้นความเร็วรอบของมอเตอร์ที่ได้จึงต่ำ ส่วนวงจรที่ใช้ในการควบคุมทิศทางและความเร็วในการหมุนของมอเตอร์กระแสตรงสามารถควบคุมทิศทางได้ดี แต่ในส่วนของกำลังที่ได้นั้นจะต่ำกว่าการควบคุมโดยควบคุมระดับของกระแสไฟโดยตรง

% duty	10	20	30	40	50	60	70	80	90	100
กระแส (mA)	3	6	10	13	16	20	24	27	31	35

ตารางที่ 4.2 แสดงค่าของ % duty กับกระแสของมอเตอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 4.2 เป็นการแสดงให้เห็นถึงการควบคุมความเร็วมอเตอร์กระแสตรงโดยซอร์ฟแวร์ ซึ่งใช้ซอร์ฟแวร์ในการสร้างพัลส์ขึ้นและทำการควบคุมความกว้างของพัลส์ในช่วงบวกลให้สามารถเปลี่ยนแปลงได้ตามความต้องการ โดยกำหนดเป็น % duty ขึ้นเพื่อที่จะได้สะดวกแก่การตรวจสอบและการนำไปใช้งาน ซึ่งจะเห็นได้ว่าค่ากระแสของมอเตอร์มีการเปลี่ยนแปลงตามค่า % duty หรือก็คือความเร็วการหมุนของมอเตอร์มีการเปลี่ยนแปลงตามค่า % duty

#### 4.3 ภาคควบคุมมอเตอร์เซ็นเซอร์

มอเตอร์ที่ใช้งานในภาคนี้เป็นมอเตอร์กระแสตรงชนิดเดียวกับภาคควบคุมมอเตอร์ล้อเลื่อน และใช้วงจรแบบเดียวกันในการควบคุมทิศทางและความเร็ว ดังนั้นผลของการทำงานจึงเหมือนกันกับภาคควบคุมมอเตอร์ล้อเลื่อน จะต่างกันก็แค่เพียงจุดประสงค์ในการทำงานคือใช้ในการควบคุมเซ็นเซอร์แสงหรือโฟโต ไดโอดในการตรวจจับสัญญาณแสงโดยรอบของตัวหุ่นยนต์ และตรวจหาจุดกำเนิดแสงปลายทางเพื่อที่จะใช้ในการกำหนดทิศทางในการเคลื่อนที่ไปยังจุดกำเนิดแสงได้อย่างถูกต้อง

#### 4.4 ภาคตรวจจับสิ่งกีดขวาง

วงจรในภาคนี้สามารถทำการตรวจจับสิ่งกีดขวางได้ แต่ระยะห่างที่ได้นั้นยังต่ำและความแม่นยำในการหลบหลีกก็ยังไม่หลากหลายเป็นเพียงรูปแบบการหลบหลีกง่าย ๆ ไม่ซับซ้อน โดยจะทำการหลบหลีกสิ่งกีดขวางเฉพาะด้านหน้าของตัวหุ่นยนต์เท่านั้นคือ เมื่อมีสิ่งกีดขวางอยู่ด้านหน้าก็จะทำการเลี้ยวไปทางขวาแล้วจึงเคลื่อนที่ไปจุดกำเนิดแสงต่อไป

ลักษณะสีของวัตถุที่นำมาใช้เป็นสิ่งกีดขวางก็มีผลต่อระยะห่างในการตรวจจับเช่นกัน หากใช้วัตถุที่มีสีดำด้านจะทำให้ไม่สามารถทำการตรวจจับได้หากเป็นสีดำนการตรวจจับยังสามารถทำได้แต่ระยะทางจะไม่ห่างมากนัก สีขาวเป็นสีที่วงจรสามารถตรวจจับได้ระยะห่างมากที่สุด

#### 4.5 สรุปผลการทดลอง

ในส่วนของฮาร์ดแวร์ทุกส่วนของตัวหุ่นยนต์สามารถทำงานได้ดังที่ได้กล่าวไว้ในข้างต้น ในส่วนของซอร์ฟแวร์นั้นก็ยังสามารถทำการควบคุมให้หุ่นยนต์แมลงสามารถตรวจหาแสงและสามารถควบคุมทิศทางให้เคลื่อนที่ไปตามแสงได้

### หนังสืออ้างอิง

1. ธีรวัฒน์ ประกอบผล, “การประยุกต์ใช้งานไมโครคอนโทรลเลอร์”, สมาคมส่งเสริมเทคโนโลยี (ไทย-ญี่ปุ่น), 1999. (19-35)
2. Triebel, Walter A, 1997, The 8088 and Microprocessors, Englewood Cliffs, NJ.:Prentice-all, pp. 723-727,810-818.



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## ภาคผนวก ก



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

1          ;***** USER RAM *****
0000      2          ORG          00H
0000      3          DS          8
0008      4          SYSSTK: DS    30
0026      5          ST_M:   DS    1
0027      6          R_TURN: DS    1
0028      7          L_TURN: DS    1
0029      8          BR_TURN: DS    1
002A      9          BL_TURN: DS    1
002B     10          SHIFT:  DS    1
002C     11          B_SHIFT: DS    1
002D     12          V_REF:   DS    1
002E     13          BV_REF:  DS    1
002F     14          ST_SCAN: DS    1
0030     15          OUT:     DS    1
0031     16          EXIT:    DS    2
0033     17          R_SPEED: DS    1
0034     18          L_SPEED: DS    1
0035     19          NUM1:    DS    1
0036     20          NUM2:    DS    1
0037     21          NUM3:    DS    1
0038     22          NUM4:    DS    1
0039     23          ST_NUM:  DS    1
003A     24          WALL:    DS    1
003B     25          COUNT:   DS    1
003C     26          COUNT1:  DS    1
003D     27          FLOW:    DS    1
003E     28          FLOW1:   DS    1
003F     29          RANGE:   DS    1
0040     30          PASS:    DS    1
          31
00B0=    32          INTR     BIT    P3.0
00B1=    33          CS       BIT    P3.1
00B3=    34          READ    BIT    P3.3
00B4=    35          INPUT   BIT    P3.4
00B5=    36          START   BIT    P3.5
00B7=    37          MOTOR   BIT    P3.7
00E0=    38          CENTER  BIT    ACC.0
00E1=    39          HEAD    BIT    ACC.1
00E4=    40          CH1     BIT    ACC.4

```

8051 Cross-Assembler (1.3)  
robot64.asm

(C) 1987, 1989 Binary Technology

```
00E5=          41          CH2          BIT          ACC.5
00E6=          42          CH3          BIT          ACC.6
00E7=          43          CH4          BIT          ACC.7
               44
               45          ;***** START PROGRAM *****
               46
0000           47          ORG          0000H
               48
0000 011D      49          AJMP         START_P
               50
0003           51          ORG          0003H
0003 819D      52          AJMP         INTE0
               53
000B           54          ORG          000BH
000B 81A6      55          AJMP         INTT0
               56
001B           57          ORG          001BH
001B A105      58          AJMP         INTT1
               59
               60
001D 7800      61          START_P: MOV      R0,#0
001F D8FE      62          DJNZ      R0,$
0021 758107    63          MOV      SP,#(SYSSTK-1)
0024 7590FF    64          MOV      P1,#0FFH
0027 75B03F    65          MOV      P3,#00111111B
002A 7400      66          MOV      A,#0
002C F526      67          MOV      ST_M,A
002E D2B7      68          SETB    MOTOR
0030 C2B7      69          CLR     MOTOR
0032 F529      70          MOV      BR_TURN,A
0034 F52A      71          MOV      BL_TURN,A
0036 F52C      72          MOV      B_SHIFT,A
0038 F533      73          MOV      R_SPEED,A
003A F534      74          MOV      L_SPEED,A
003C F52D      75          MOV      V_REF,A
003E F52E      76          MOV      BV_REF,A
0040 F52F      77          MOV      ST_SCAN,A
0042 F530      78          MOV      OUT,A
0044 F531      79          MOV      EXIT,A
0046 F535      80          MOV      NUM1,A
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
0048 F536          81          MOV          NUM2,A
004A F537          82          MOV          NUM3,A
004C F538          83          MOV          NUM4,A
004E F53D          84          MOV          FLOW,A
0050 F53E          85          MOV          FLOW1,A
0052 7401          86          MOV          A,#1
0054 F527          87          MOV          R_TURN,A
0056 F528          88          MOV          L_TURN,A
0058 F52B          89          MOV          SHIFT,A
005A F540          90          MOV          PASS,A
005C 758CC0        91          MOV          TH0,#256-64
005F 758AC0        92          MOV          TL0,#256-64
0062 758D38        93          MOV          TH1,#256-200
0065 758B38        94          MOV          TL1,#256-200
0068 758922        95          MOV          TMOD,#022H
006B D288          96          SETB         TCON.0
006D D2A8          97          SETB         IE.0
006F D2A9          98          SETB         IE.1
0071 D2AB          99          SETB         IE.3
0073 D2AF          100         SETB         IE.7
0075 752C64        101         MOV          B_SHIFT,#100
0078 519C          102         ACALL        CARIB
007A E52E          103         MOV          A,BV_REF
007C 2414          104         ADD          A,#20
007E F52D          105         MOV          V_REF,A
0080 518F          106         ACALL        SET_ADC
0082 753005        107         MOV          OUT,#5
0085 753105        108         MOV          EXIT,#5
0088 753300        109         MOV          R_SPEED,#0
008B 753400        110         MOV          L_SPEED,#0
111
112                ;***** GO TO LIGHT *****
008E 7590FF        113         RUN:      MOV          P1,#0FFH
0091 C2B4          114         CLR          INPUT
0093 E590          115         MOV          A,P1
0095 D2B4          116         SETB         INPUT
0097 30E116        117         JNB         HEAD,R_1
009A E540          118         MOV          A,PASS
009C 7012          119         JNZ         R_1
009E 754001        120         MOV          PASS,#1
```

00A1	C28E	121	CLR	TCON.6
00A3	753F64	122	MOV	RANGE,#100
00A6	E53D	123	MOV	A, FLOW
00A8	7004	124	JNZ	\$+6
00AA	053D	125	INC	FLOW
00AC	01B0	126	AJMP	R_1
00AE	153D	127	DEC	FLOW
00B0	E53D	128	MOV	A, FLOW
00B2	7010	129	JNZ	R_2
00B4	852690	130	MOV	P1, ST_M
00B7	C294	131	CLR	P1.4
00B9	859026	132	MOV	ST_M, P1
00BC	D2B7	133	SETB	MOTOR
00BE	C2B7	134	CLR	MOTOR
00C0	714E	135	ACALL	SHIFT_L
00C2	01D2	136	AJMP	R_3
00C4	852690	137	MOV	P1, ST_M
00C7	C295	138	CLR	P1.5
00C9	859026	139	MOV	ST_M, P1
00CC	D2B7	140	SETB	MOTOR
00CE	C2B7	141	CLR	MOTOR
00D0	710B	142	ACALL	SHIFT_R
00D2	7191	143	ACALL	F_RIGHT
00D4	71D4	144	ACALL	F_LEFT
00D6	5162	145	ACALL	S_ROL
00D8	E539	146	MOV	A, ST_NUM
00DA	6002	147	JZ	\$+4
00DC	31CA	148	ACALL	M_SIDE
00DE	E530	149	MOV	A, OUT
00E0	7002	150	JNZ	\$+4
00E2	21A4	151	AJMP	E_RUN
00E4	E53D	152	MOV	A, FLOW
00E6	B53E5F	153	CJNE	A, FLOW1, RUN2
00E9	7590FF	154	MOV	P1, #0FFH
00EC	C2B4	155	CLR	INPUT
00EE	E590	156	MOV	A, P1
00F0	D2B4	157	SETB	INPUT
00F2	30E014	158	JNB	CENTER, R1_3
00F5	E540	159	MOV	A, PASS
00F7	6010	160	JZ	R1_3

00F9	753C32	161		MOV	COUNT1,#50
00FC	753F64	162		MOV	RANGE,#100
00FF	754000	163		MOV	PASS,#0
0102	753105	164		MOV	EXIT,#5
0105	1530	165		DEC	OUT
0107	D28E	166		SETB	TCON.6
0109	20B082	167	R1_3:	JB	INTR,RUN
010C	51F7	168		ACALL	CHK_IP
010E	C3	169		CLR	C
010F	B52D00	170		CJNE	A,V_REF,\$+3
0112	401E	171		JC	E_R1
0114	D5311B	172		DJNZ	EXIT,E_R1
0117	C28E	173		CLR	TCON.6
0119	753005	174		MOV	OUT,#5
011C	E53E	175		MOV	A,FLOW1
011E	7014	176		JNZ	R1_4
0120	74AA	177		MOV	A,#100+70
0122	953F	178		SUBB	A,RANGE
0124	F533	179		MOV	R_SPEED,A
0126	7464	180		MOV	A,#100
0128	953F	181		SUBB	A,RANGE
012A	F53F	182		MOV	RANGE,A
012C	7446	183		MOV	A,#70
012E	953F	184		SUBB	A,RANGE
0130	F534	185		MOV	L_SPEED,A
0132	018E	186	E_R1:	AJMP	RUN
0134	74AA	187	R1_4:	MOV	A,#100+70
0136	953F	188		SUBB	A,RANGE
0138	F534	189		MOV	L_SPEED,A
013A	7464	190		MOV	A,#100
013C	953F	191		SUBB	A,RANGE
013E	F53F	192		MOV	RANGE,A
0140	7446	193		MOV	A,#70
0142	953F	194		SUBB	A,RANGE
0144	F533	195		MOV	R_SPEED,A
0146	018E	196		AJMP	RUN
		197			
0148	7590FF	198	RUN2:	MOV	P1,#0FFH
014B	C2B4	199		CLR	INPUT
014D	E590	200		MOV	A,P1

014F	D2B4	201		SETB	INPUT
0151	30E035	202		JNB	CENTER, R2_3
0154	853D3E	203		MOV	FLOW1, FLOW
0157	308E2F	204		JNB	TCON.6, R2_3
015A	C28E	205		CLR	TCON.6
015C	753105	206		MOV	EXIT, #5
015F	E53E	207		MOV	A, FLOW1
0161	6014	208		JZ	R2_4
0163	74AA	209		MOV	A, #100+70
0165	953F	210		SUBB	A, RANGE
0167	F533	211		MOV	R_SPEED, A
0169	7464	212		MOV	A, #100
016B	953F	213		SUBB	A, RANGE
016D	F53F	214		MOV	RANGE, A
016F	7446	215		MOV	A, #70
0171	953F	216		SUBB	A, RANGE
0173	F534	217		MOV	L_SPEED, A
0175	2189	218		AJMP	R2_3
0177	74AA	219	R2_4:	MOV	A, #100+70
0179	953F	220		SUBB	A, RANGE
017B	F534	221		MOV	L_SPEED, A
017D	7464	222		MOV	A, #100
017F	953F	223		SUBB	A, RANGE
0181	F53F	224		MOV	RANGE, A
0183	7446	225		MOV	A, #70
0185	953F	226		SUBB	A, RANGE
0187	F533	227		MOV	R_SPEED, A
0189	20B016	228	R2_3:	JB	INTR, E_R2
018C	51F7	229		ACALL	CHK_IP
018E	C3	230		CLR	C
018F	B52D00	231		CJNE	A, V_REF, \$+3
0192	400E	232		JC	E_R2
0194	D5310B	233		DJNZ	EXIT, E_R2
0197	753005	234		MOV	OUT, #5
019A	753C32	235		MOV	COUNT1, #50
019D	753F64	236		MOV	RANGE, #100
01A0	D28E	237		SETB	TCON.6
01A2	018E	238	E_R2:	AJMP	RUN
		239			
		240			

```

241          ;***** RESTART *****
01A4 759000 242          E_RUN:  MOV     P1,#0
01A7 859026 243          MOV     ST_M,P1
01AA D2B7    244          SETB   MOTOR
01AC C2B7    245          CLR    MOTOR
01AE 752900 246          MOV     BR_TURN,#0
01B1 752A00 247          MOV     BL_TURN,#0
01B4 752B01 248          MOV     SHIFT,#01
01B7 752701 249          MOV     R_TURN,#01
01BA 752801 250          MOV     L_TURN,#01
01BD 753300 251          MOV     R_SPEED,#0
01C0 753400 252          MOV     L_SPEED,#0
01C3 753005 253          MOV     OUT,#5
                254
01C6 B122    255          ACALL  DELAY2
01C8 018E    256          AJMP   RUN
                257
                258          ;*****
01CA B40503 259          M_SIDE: CJNE   A,#00000101B,$+6
01CD 31DC    260          ACALL  T_RIGHT
01CF 22      261          RET
01D0 B40103 262          CJNE   A,#00000001B,$+6
01D3 31DC    263          ACALL  T_RIGHT
01D5 22      264          RET
01D6 B40402 265          CJNE   A,#00000100B,$+5
01D9 511F    266          ACALL  T_LEFT
01DB 22      267          RET
                268
                269          ;***** TURN RIGHT *****
01DC 759000 270          T_RIGHT: MOV     P1,#0
01DF 859026 271          MOV     ST_M,P1
01E2 D2B7    272          SETB   MOTOR
01E4 C2B7    273          CLR    MOTOR
01E6 752B01 274          MOV     SHIFT,#01
01E9 E539    275          MOV     A,ST_NUM
01EB 753300 276          MOV     R_SPEED,#0
01EE 75343C 277          MOV     L_SPEED,#60
01F1 7191    278          T_R_1: ACALL  F_RIGHT
01F3 71D4    279          ACALL  F_LEFT
01F5 5162    280          ACALL  S_ROL

```

```
01F7 E539          281          MOV          A,ST_NUM
01F9 30E1F5       282          JNB          ACC.1,T_R_1
01FC 759000       283          MOV          P1,#0
01FF 859026       284          MOV          ST_M,P1
0202 D2B7         285          SETB        MOTOR
0204 C2B7         286          CLR         MOTOR
0206 752900       287          MOV          BR_TURN,#0
0209 752A00       288          MOV          BL_TURN,#0
020C 752B01       289          MOV          SHIFT,#01
020F 752701       290          MOV          R_TURN,#01
0212 752801       291          MOV          L_TURN,#01
0215 753300       292          MOV          R_SPEED,#0
0218 753400       293          MOV          L_SPEED,#0
021B 753005       294          MOV          OUT,#5
021E 22           295          RET
                296
                297
                ;***** TURN LEFT *****
021F 759000       298          T_LEFT:    MOV          P1,#0
0222 859026       299          MOV          ST_M,P1
0225 D2B7         300          SETB        MOTOR
0227 C2B7         301          CLR         MOTOR
0229 752B01       302          MOV          SHIFT,#01
022C E539         303          MOV          A,ST_NUM
022E 75333C       304          MOV          R_SPEED,#60
0231 753400       305          MOV          L_SPEED,#0
0234 7191         306          T_L_1:    ACALL       F_RIGHT
0236 71D4         307          ACALL       F_LEFT
0238 5162         308          ACALL       S_ROL
023A E539         309          MOV          A,ST_NUM
023C 30E3F5       310          JNB          ACC.3,T_L_1
023F 759000       311          MOV          P1,#0
0242 859026       312          MOV          ST_M,P1
0245 D2B7         313          SETB        MOTOR
0247 C2B7         314          CLR         MOTOR
0249 752900       315          MOV          BR_TURN,#0
024C 752A00       316          MOV          BL_TURN,#0
024F 752B01       317          MOV          SHIFT,#01
0252 752701       318          MOV          R_TURN,#01
0255 752801       319          MOV          L_TURN,#01
0258 753300       320          MOV          R_SPEED,#0
```

8051 Cross-Assembler (1.3)  
robot64.asm

(C) 1987, 1989 Binary Technology

```
025B 753400      321          MOV      L_SPEED,#0
025E 753005      322          MOV      OUT,#5
0261 22          323          RET
                 324
                 325          ;***** START ROLER *****
0262 E529        326 S_ROL:     MOV      A,BR_TURN
0264 6005        327          JZ       $+7
0266 C3          328          CLR      C
0267 B5270E      329          CJNE    A,R_TURN,$+17
026A C3          330          CLR      C
026B B53302      331          CJNE    A,R_SPEED,$+5
026E 4178        332          AJMP    $+10
0270 5004        333          JNC     $+6
0272 0529        334          INC     BR_TURN
0274 4178        335          AJMP    $+4
0276 1529        336          DEC     BR_TURN
0278 E52A        337          MOV     A,BL_TURN
027A 6005        338          JZ       $+7
027C C3          339          CLR      C
027D B5280E      340          CJNE    A,L_TURN,$+17
0280 C3          341          CLR      C
0281 B53402      342          CJNE    A,L_SPEED,$+5
0284 418E        343          AJMP    $+10
0286 5004        344          JNC     $+6
0288 052A        345          INC     BL_TURN
028A 418E        346          AJMP    $+4
028C 152A        347          DEC     BL_TURN
028E 22          348          RET
                 349
                 350          ;***** START ADC *****
028F C2B1        351 SET_ADC:  CLR     CS
0291 C2B5        352          CLR     START
0293 00          353          NOP
0294 D2B5        354          SETB   START
0296 D2B1        355          SETB   CS
0298 752B01      356          MOV     SHIFT,#1
029B 22          357          RET
                 358
                 359          ;***** CARIB *****
029C 714E        360 CARIB:    ACALL  SHIFT_L
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

029E	7590FF	361	MOV	P1,#0FFH
02A1	C2B4	362	CLR	INPUT
02A3	E590	363	MOV	A,P1
02A5	D2B4	364	SETB	INPUT
02A7	30E0F2	365	JNB	CENTER,CARIB
02AA	518F	366	ACALL	SET_ADC
02AC	714E	367	CARIB0: ACALL	SHIFT_L
02AE	7590FF	368	MOV	P1,#0FFH
02B1	C2B4	369	CLR	INPUT
02B3	E590	370	MOV	A,P1
02B5	D2B4	371	SETB	INPUT
02B7	20E111	372	JB	HEAD,CARIB1
02BA	20B0EF	373	JB	INTR,CARIB0
02BD	51F7	374	ACALL	CHK_IP
02BF	C3	375	CLR	C
02C0	B52E02	376	CJNE	A,BV_REF,N_CARIB0
02C3	41AC	377	AJMP	CARIB0
02C5	40E5	378	N_CARIB0:JC	CARIB0
02C7	F52E	379	MOV	BV_REF,A
02C9	41AC	380	AJMP	CARIB0
02CB	714E	381	CARIB1: ACALL	SHIFT_L
02CD	7590FF	382	MOV	P1,#0FFH
02D0	C2B4	383	CLR	INPUT
02D2	E590	384	MOV	A,P1
02D4	D2B4	385	SETB	INPUT
02D6	20E011	386	JB	CENTER,E_CARIB
02D9	20B0EF	387	JB	INTR,CARIB1
02DC	51F7	388	ACALL	CHK_IP
02DE	C3	389	CLR	C
02DF	B52E02	390	CJNE	A,BV_REF,N_CARIB1
02E2	41CB	391	AJMP	CARIB1
02E4	40E5	392	N_CARIB1:JC	CARIB1
02E6	F52E	393	MOV	BV_REF,A
02E8	41CB	394	AJMP	CARIB1
02EA	852690	395	E_CARIB: MOV	P1,ST_M
02ED	C295	396	CLR	P1.5
02EF	859026	397	MOV	ST_M,P1
02F2	D2B7	398	SETB	MOTOR
02F4	C2B7	399	CLR	MOTOR
02F6	22	400	RET	

```

401
402 ;***** READ LIGHT INPUT *****
02F7 7590FF 403 CHK_IP: MOV P1,#0FFH
02FA C2B1 404 CLR CS
02FC C2B3 405 CLR READ
02FE 00 406 NOP
02FF E590 407 MOV A,P1
0301 D2B3 408 SETB READ
0303 C2B5 409 CLR START
0305 00 410 NOP
0306 D2B5 411 SETB START
0308 D2B1 412 SETB CS
030A 22 413 RET
414
415 ;***** SHIFT MOTOR SENSOR *****
030B E526 416 SHIFT_R: MOV A,ST_M
030D 20E416 417 JB ACC.4,OFF_SR
0310 D52B12 418 DJNZ SHIFT,E_ON_SR
0313 E52C 419 MOV A,B_SHIFT
0315 600F 420 JZ OFF_SR
0317 F52B 421 MOV SHIFT,A
0319 852690 422 MOV P1,ST_M
031C D294 423 SETB P1.4
031E 859026 424 MOV ST_M,P1
0321 D2B7 425 SETB MOTOR
0323 C2B7 426 CLR MOTOR
0325 22 427 E_ON_SR: RET
428
0326 D52B24 429 OFF_SR: DJNZ SHIFT,E_OFF_SR
0329 7464 430 OFF_SR0: MOV A,#100
032B C3 431 CLR C
032C B52C0C 432 CJNE A,B_SHIFT,OFF_SR2
032F 752B01 433 OFF_SR1: MOV SHIFT,#1
0332 852690 434 MOV P1,ST_M
0335 C294 435 CLR P1.4
0337 859026 436 MOV ST_M,P1
033A 22 437 RET
033B 40F2 438 OFF_SR2: JC OFF_SR1
033D 952C 439 SUBB A,B_SHIFT
033F F52B 440 MOV SHIFT,A

```

```
0341 852690      441          MOV      P1, ST_M
0344 C294        442          CLR      P1.4
0346 859026     443          MOV      ST_M, P1
0349 D2B7        444          SETB    MOTOR
034B C2B7        445          CLR      MOTOR
034D 22          446          E_OFF_SR:RET
                                447
034E E526        448          SHIFT_L: MOV   A, ST_M
0350 20E516     449          JB      ACC.5, OFF_SL
0353 D52B12     450          DJNZ   SHIFT, E_ON_SL
0356 E52C        451          MOV      A, B_SHIFT
0358 600F        452          JZ      OFF_SL
035A F52B        453          MOV      SHIFT, A
035C 852690     454          MOV      P1, ST_M
035F D295        455          SETB    P1.5
0361 859026     456          MOV      ST_M, P1
0364 D2B7        457          SETB    MOTOR
0366 C2B7        458          CLR      MOTOR
0368 22          459          E_ON_SL: RET
                                460
0369 D52B24     461          OFF_SL:  DJNZ   SHIFT, E_OFF_SL
036C 7464        462          OFF_SL0: MOV   A, #100
036E C3          463          CLR      C
036F B52C0C     464          CJNE   A, B_SHIFT, OFF_SL2
0372 752B01     465          OFF_SL1: MOV   SHIFT, #1
0375 852690     466          MOV      P1, ST_M
0378 C295        467          CLR      P1.5
037A 859026     468          MOV      ST_M, P1
037D 22          469          RET
037E 40F2        470          OFF_SL2: JC    OFF_SL1
0380 952C        471          SUBB   A, B_SHIFT
0382 F52B        472          MOV      SHIFT, A
0384 852690     473          MOV      P1, ST_M
0387 C295        474          CLR      P1.5
0389 859026     475          MOV      ST_M, P1
038C D2B7        476          SETB    MOTOR
038E C2B7        477          CLR      MOTOR
0390 22          478          E_OFF_SL:RET
                                479
```

8051 Cross-Assembler (1.3)  
robot64.asm

(C) 1987, 1989 Binary Technology

```

      480      ;***** CONTROL MOTOR (FORWORD) **
0391 E526      481      F_RIGHT: MOV      A,ST_M
0393 20E016    482      JB          ACC.0,OFF_FR
0396 D52712    483      DJNZ     R_TURN,E_ON_FR
0399 E529      484      MOV      A,BR_TURN
039B 6012      485      JZ          OFF_FR0
039D F527      486      MOV      R_TURN,A
039F 852690    487      MOV      P1,ST_M
03A2 D290      488      SETB     P1.0
03A4 859026    489      MOV      ST_M,P1
03A7 D2B7      490      SETB     MOTOR
03A9 C2B7      491      CLR      MOTOR
03AB 22        492      E_ON_FR: RET
      493
03AC D52724    494      OFF_FR:  DJNZ     R_TURN,E_OFF_FR
03AF 7464      495      OFF_FR0: MOV     A,#100
03B1 C3        496      CLR      C
03B2 B5290C    497      CJNE     A,BR_TURN,OFF_FR2
03B5 752701    498      OFF_FR1: MOV     R_TURN,#1
03B8 852690    499      MOV      P1,ST_M
03BB C290      500      CLR      P1.0
03BD 859026    501      MOV      ST_M,P1
03C0 22        502      RET
03C1 40F2      503      OFF_FR2: JC      OFF_FR1
03C3 9529      504      SUBB     A,BR_TURN
03C5 F527      505      MOV      R_TURN,A
03C7 852690    506      MOV      P1,ST_M
03CA C290      507      CLR      P1.0
03CC 859026    508      MOV      ST_M,P1
03CF D2B7      509      SETB     MOTOR
03D1 C2B7      510      CLR      MOTOR
03D3 22        511      E_OFF_FR:RET
      512
03D4 E526      513      F_LEFT:  MOV     A,ST_M
03D6 20E216    514      JB          ACC.2,OFF_FL
03D9 D52812    515      DJNZ     L_TURN,E_ON_FL
03DC E52A      516      MOV      A,BL_TURN
03DE 6012      517      JZ          OFF_FLO
03E0 F528      518      MOV      L_TURN,A
03E2 852690    519      MOV      P1,ST_M
03E5 D292      520      SETB     P1.2
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

03E7 859026      521                MOV          ST_M,P1
03EA D2B7        522                SETB        MOTOR
03EC C2B7        523                CLR         MOTOR
03EE 22          524                E_ON_FL:   RET
                    525
03EF D52824      526                OFF_FL:    DJNZ        L_TURN,E_OFF_FL
03F2 7464        527                OFF_FL0:   MOV          A,#100
03F4 C3           528                CLR         C
03F5 B52A0C      529                CJNE        A,BL_TURN,OFF_FL2
03F8 752801      530                OFF_FL1:   MOV          L_TURN,#1
03FB 852690      531                MOV          P1,ST_M
03FE C292        532                CLR         P1.2
0400 859026      533                MOV          ST_M,P1
0403 22          534                RET
0404 40F2        535                OFF_FL2:   JC          OFF_FL1
0406 952A        536                SUBB        A,BL_TURN
0408 F528        537                MOV          L_TURN,A
040A 852690      538                MOV          P1,ST_M
040D C292        539                CLR         P1.2
040F 859026      540                MOV          ST_M,P1
0412 D2B7        541                SETB        MOTOR
0414 C2B7        542                CLR         MOTOR
0416 22          543                E_OFF_FL:  RET
                    544
                    545                ;***** CONTROL MOTOR (REVERSE) **
0417 E526        546                R_RIGHT:   MOV          A,ST_M
0419 20E116      547                JB          ACC.1,OFF_RR
041C D52712      548                DJNZ        R_TURN,E_ON_RR
041F E529        549                MOV          A,BR_TURN
0421 6012        550                JZ          OFF_RR0
0423 F527        551                MOV          R_TURN,A
0425 852690      552                MOV          P1,ST_M
0428 D291        553                SETB        P1.1
042A 859026      554                MOV          ST_M,P1
042D D2B7        555                SETB        MOTOR
042F C2B7        556                CLR         MOTOR
0431 22          557                E_ON_RR:   RET
                    558
0432 D52724      559                OFF_RR:    DJNZ        R_TURN,E_OFF_RR
0435 7464        560                OFF_RR0:   MOV          A,#100

```

```

0437 C3          561          CLR          C
0438 B5290C     562          CJNE         A, BR_TURN, OFF_RR2
043B 752701     563          OFF_RR1: MOV  R_TURN, #1
043E 852690     564          MOV         P1, ST_M
0441 C291       565          CLR         P1.1
0443 859026     566          MOV         ST_M, P1
0446 22         567          RET
0447 40F2       568          OFF_RR2: JC  OFF_RR1
0449 9529       569          SUBB        A, BR_TURN
044B F527       570          MOV         R_TURN, A
044D 852690     571          MOV         P1, ST_M
0450 C291       572          CLR         P1.1
0452 859026     573          MOV         ST_M, P1
0455 D2B7       574          SETB        MOTOR
0457 C2B7       575          CLR         MOTOR
0459 22         576          E_OFF_RR: RET
                   577
045A E526       578          R_LEFT: MOV  A, ST_M
045C 20E316     579          JB          ACC.3, OFF_RL
045F D52812     580          DJNZ        L_TURN, E_ON_RL
0462 E52A       581          MOV         A, BL_TURN
0464 6012       582          JZ          OFF_RL0
0466 F528       583          MOV         L_TURN, A
0468 852690     584          MOV         P1, ST_M
046B D293       585          SETB        P1.3
046D 859026     586          MOV         ST_M, P1
0470 D2B7       587          SETB        MOTOR
0472 C2B7       588          CLR         MOTOR
0474 22         589          E_ON_RL: RET
                   590
0475 D52824     591          OFF_RL: DJNZ L_TURN, E_OFF_RL
0478 7464       592          OFF_RL0: MOV A, #100
047A C3         593          CLR         C
047B B52A0C     594          CJNE         A, BL_TURN, OFF_RL2
047E 752801     595          OFF_RL1: MOV  L_TURN, #1
0481 852690     596          MOV         P1, ST_M
0484 C293       597          CLR         P1.3
0486 859026     598          MOV         ST_M, P1
0489 22         599          RET
048A 40F2       600          OFF_RL2: JC  OFF_RL1

```

```

048C 952A          601          SUBB      A,BL_TURN
048E F528          602          MOV       L_TURN,A
0490 852690        603          MOV       P1,ST_M
0493 C293          604          CLR      P1.3
0495 859026        605          MOV     ST_M,P1
0498 D2B7          606          SETB    MOTOR
049A C2B7          607          CLR      MOTOR
049C 22            608          E_OFF_RL:RET
                    609
                    610
049D 753B06        611          ;***** EXTERNAL INTERRUPT ****
INTE0:             611          MOV       COUNT,#06
04A0 753A00        612          MOV       WALL,#0
04A3 D28C          613          SETB    TCON.4
04A5 32            614          RETI
                    615
                    616
04A6 C0E0          617          ;***** INTERNAL INTERRUPT ****
INTT0:            617          PUSH     ACC
04A8 C0D0          618          PUSH     PSW
04AA E5B0          619          MOV      A,P3
04AC 4405          620          ORL     A,#00000101B
04AE C0E0          621          PUSH     ACC
04B0 75B03F        622          MOV     P3,#00111111B
04B3 C090          623          PUSH     P1
04B5 7590FF        624          MOV     P1,#0FFH
04B8 C2B4          625          CLR     INPUT
04BA E590          626          MOV     A,P1
04BC D2B4          627          SETB   INPUT
04BE 423A          628          ORL     WALL,A
04C0 D53B39        629          DJNZ   COUNT,E_INTT0
04C3 C28C          630          CLR     TCON.4
04C5 E53A          631          MOV     A,WALL
04C7 33            632          RLC    A
04C8 C538          633          XCH   A,NUM4
04CA 33            634          RLC    A
04CB C538          635          XCH   A,NUM4
04CD 33            636          RLC    A
04CE C537          637          XCH   A,NUM3
04D0 33            638          RLC    A
04D1 C537          639          XCH   A,NUM3
04D3 33            640          RLC    A

```

```
04D4 C536      641      XCH      A, NUM2
04D6 33        642      RLC      A
04D7 C536      643      XCH      A, NUM2
04D9 33        644      RLC      A
04DA C535      645      XCH      A, NUM1
04DC 33        646      RLC      A
04DD C535      647      XCH      A, NUM1
04DF 753900    648      MOV      ST_NUM, #0
04E2 74FF      649      MOV      A, #0FFH
04E4 B53503    650      CJNE     A, NUM1, $+6
04E7 433901    651      ORL      ST_NUM, #00000001B
04EA B53703    652      CJNE     A, NUM3, $+6
04ED 433904    653      ORL      ST_NUM, #00000100B
04F0 B53603    654      CJNE     A, NUM2, $+6
04F3 433902    655      ORL      ST_NUM, #00000010B
04F6 B53803    656      CJNE     A, NUM4, E_INTT0
04F9 433908    657      ORL      ST_NUM, #00001000B
04FC D090      658      E_INTT0: POP      P1
04FE D0B0      659      POP      P3
0500 D0D0      660      POP      PSW
0502 D0E0      661      POP      ACC
0504 32        662      RETI
                663
0505 C0D0      664      INTT1:   PUSH     PSW
0507 C0E0      665      PUSH     ACC
0509 D53C08    666      DJNZ     COUNT1, E_INTT1
050C 753C32    667      MOV      COUNT1, #50
050F D53F02    668      DJNZ     RANGE, E_INTT1
0512 C28E      669      CLR      TCON.6
0514 D0E0      670      E_INTT1: POP      ACC
0516 D0D0      671      POP      PSW
0518 32        672      RETI
                673
                674 ;***** DELAY TIME *****
0519 7E50      675      DELAY:  MOV      R6, #50H
051B 7F00      676      DLY1:   MOV      R7, #0
051D DFFE      677      DJNZ     R7, $
051F DEFA      678      DJNZ     R6, DLY1
0521 22        679      RET
                680
```

8051 Cross-Assembler (1.3) (C) 1987, 1989 Binary Technology  
robot64.asm

```
0522 7D0A      681      DELAY2:  MOV      R5,#10
0524 7E00      682      DLY2_1:  MOV      R6,#0
0526 7F00      683      DLY2_2:  MOV      R7,#0
0528 DFFE      684              DJNZ     R7,$
052A DEFA      685              DJNZ     R6,DLY2_2
052C DDF6      686              DJNZ     R5,DLY2_1
052E 22        687      RET
          688
          689
0000=        690      END
```



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## ภาคผนวก ข



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

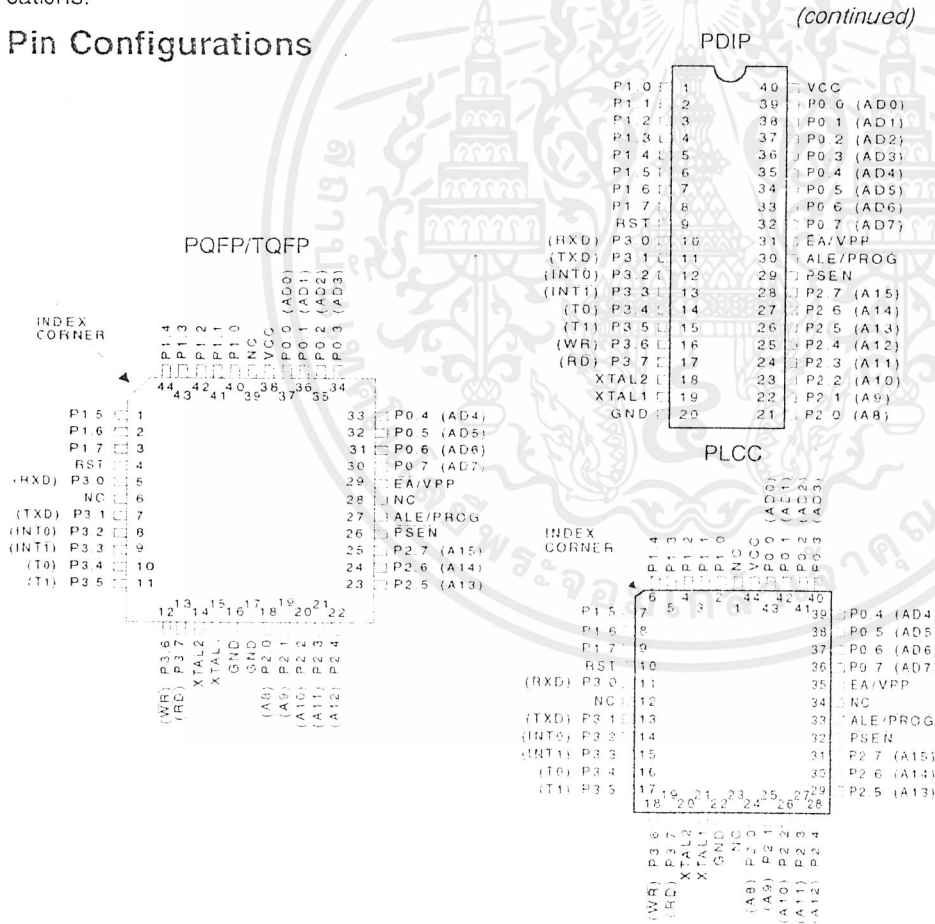
## Features

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

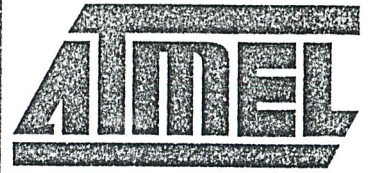
## Description

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

## Pin Configurations



0265F-A-12/97

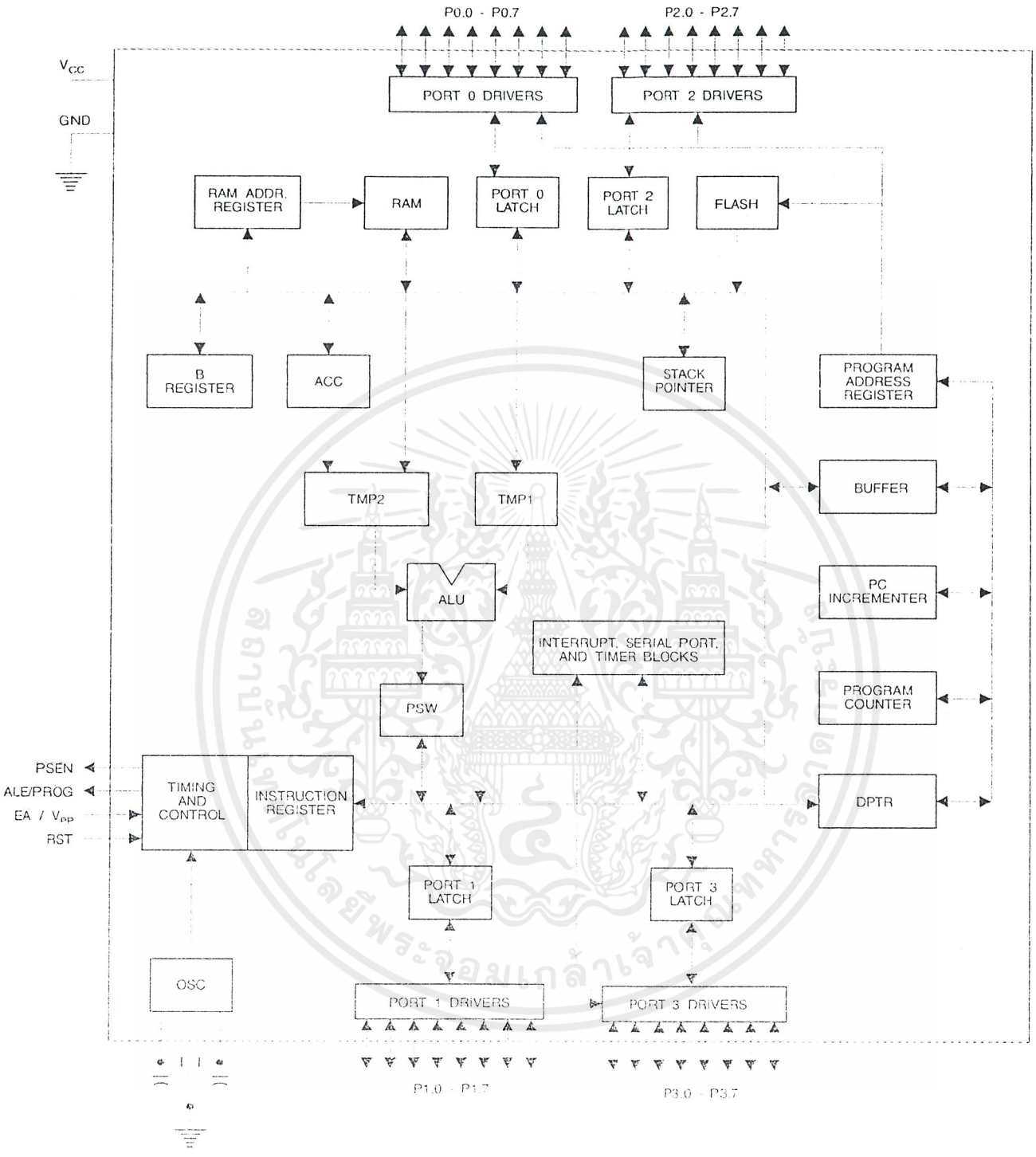


## 8-Bit Microcontroller with 4K Bytes Flash

## AT89C51

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

# Block Diagram



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

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

## Pin Description

**V<sub>CC</sub>**  
Supply voltage.

**GND**  
Ground.

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

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

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

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

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<sub>IL</sub>) because of the internal pullups.

Port 2 emits the high-order address byte during fetches from external program memory and during accesses to external data memory that use 16-bit addresses (MOVX @ DPTR). In this application it uses strong internal pullups

when emitting 1s. During accesses to external data memory that use 8-bit addresses (MOVX @ RI), Port 2 emits the contents of the P2 Special Function Register.

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

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

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

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

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

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

**ALE/PROG**  
Address Latch Enable output pulse for latching the low byte of the address during accesses to external memory. This pin is also the program pulse input (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.

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



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

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

External Access Enable.  $\overline{\text{EA}}$  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,  $\overline{\text{EA}}$  will be internally latched on reset.

$\overline{\text{EA}}$  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, for parts that require 12-volt  $V_{PP}$ .

#### XTAL1

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

#### XTAL2

Output from the inverting oscillator amplifier.

### Oscillator Characteristics

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

### Idle Mode

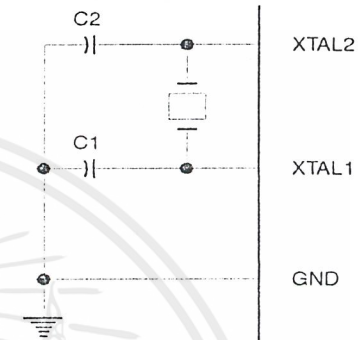
In idle mode, the CPU puts itself to sleep while all the on-chip peripherals remain active. The mode is invoked by software. The content of the on-chip RAM and all the special functions registers remain unchanged during this mode. The idle mode can be terminated by any enabled interrupt or by a hardware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

### Status of External Pins During Idle and Power Down Modes

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

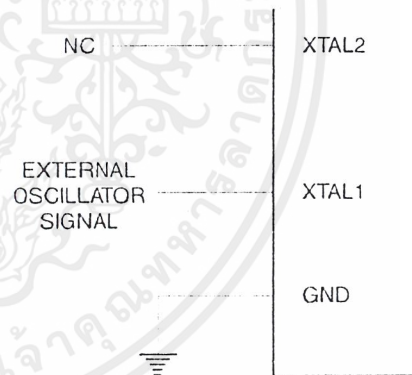
It should be noted that when idle is terminated by a hardware reset, the device normally resumes program execution, from where it left off, up to two machine cycles before the internal reset algorithm takes control. On-chip hardware inhibits access to internal RAM in this event, but access to the port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

Figure 1. Oscillator Connections



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

Figure 2. External Clock Drive Configuration



## Power Down Mode

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

## Lock Bit Protection Modes

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

## Programming the Flash

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

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

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

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

## Program Memory Lock Bits

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

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

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

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

**Data Polling:** The AT89C51 features  $\overline{Data}$  Polling to indicate the end of a write cycle. During a write cycle, an attempted read of the last byte written will result in the complement of the written datum on PO.7. Once the write cycle has been completed, true data are valid on all outputs, and the next cycle may begin.  $\overline{Data}$  Polling may begin any time after a write cycle has been initiated.

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





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

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

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

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

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

## Programming Interface

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

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

## Flash Programming Modes

Mode	RST	PSEN	ALE/PROG	EAV <sub>PP</sub>	P2.6	P2.7	P3.6	P3.7
Write Code Data	H	L		H/12V	L	H	H	H
Read Code Data	H	L	H	H	L	L	H	H
Write Lock	Bit - 1	H	L		H/12V	H	H	H
	Bit - 2	H	L		H/12V	H	H	L
	Bit - 3	H	L		H/12V	H	L	H
Chip Erase	H	L	(1)	H/12V	H	L	L	L
Read Signature Byte	H	L	H	H	L	L	L	L

Note: 1. Chip Erase requires a 10-ms PROG pulse.

Figure 3. Programming the Flash

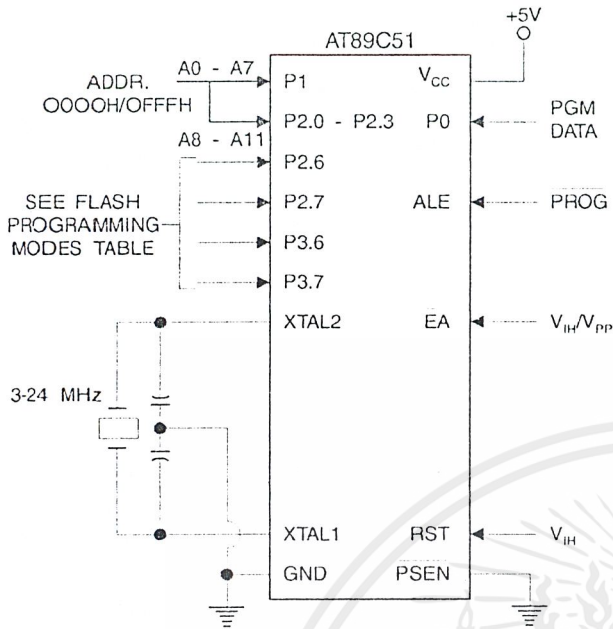
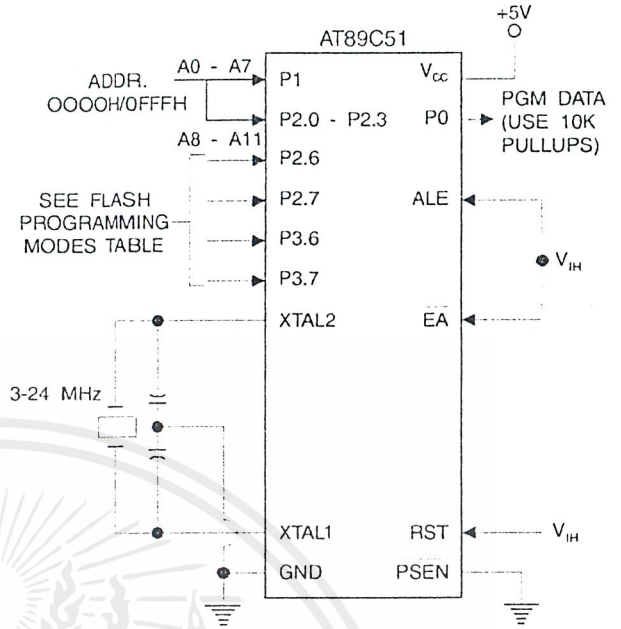


Figure 4. Verifying the Flash



## Flash Programming and Verification Characteristics

$T_A = 0^\circ\text{C to } 70^\circ\text{C}$ ,  $V_{CC} = 5.0 \pm 10\%$

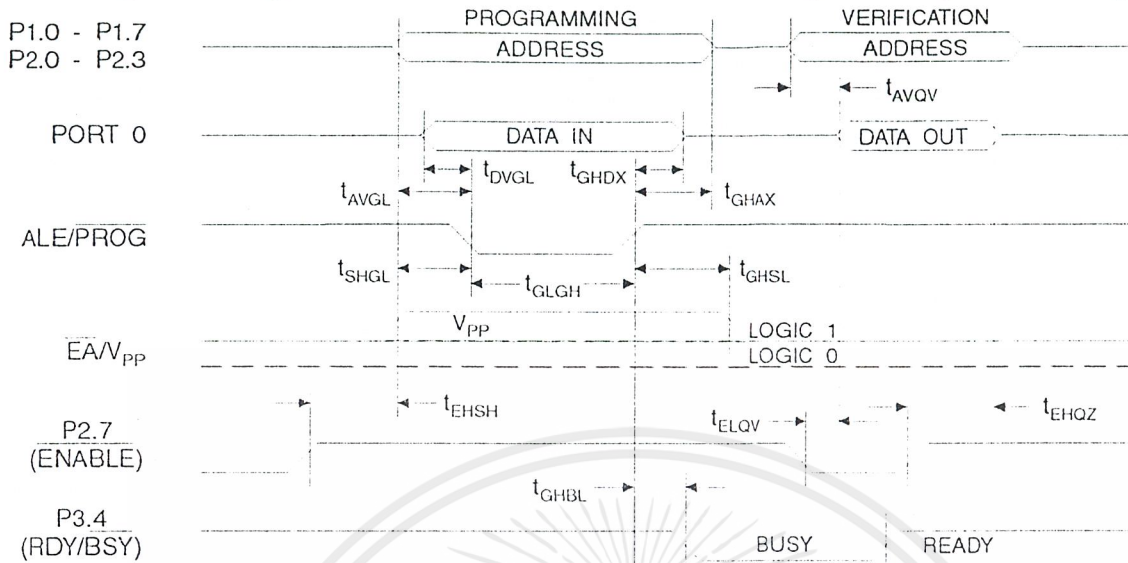
Symbol	Parameter	Min	Max	Units
$V_{PP}^{(1)}$	Programming Enable Voltage	11.5	12.5	V
$I_{PP}^{(1)}$	Programming Enable Current		1.0	mA
$1/t_{CLCL}$	Oscillator Frequency	3	24	MHz
$t_{AVGL}$	Address Setup to $\overline{PROG}$ Low	$48t_{CLCL}$		
$t_{GHAX}$	Address Hold After $\overline{PROG}$	$48t_{CLCL}$		
$t_{DVGL}$	Data Setup to $\overline{PROG}$ Low	$48t_{CLCL}$		
$t_{GHDX}$	Data Hold After $\overline{PROG}$	$48t_{CLCL}$		
$t_{EHS}$	P2.7 ( $\overline{ENABLE}$ ) High to $V_{PP}$	$48t_{CLCL}$		
$t_{SHGL}$	$V_{PP}$ Setup to $\overline{PROG}$ Low	10		$\mu\text{s}$
$t_{GHSL}^{(1)}$	$V_{PP}$ Hold After $\overline{PROG}$	10		$\mu\text{s}$
$t_{GLGH}$	$\overline{PROG}$ Width	1	110	$\mu\text{s}$
$t_{AVOV}$	Address to Data Valid		$48t_{CLCL}$	
$t_{ELQV}$	$\overline{ENABLE}$ Low to Data Valid		$48t_{CLCL}$	
$t_{EHOZ}$	Data Float After $\overline{ENABLE}$	0	$48t_{CLCL}$	
$t_{GHBL}$	$\overline{PROG}$ High to $\overline{BUSY}$ Low		1.0	$\mu\text{s}$
$t_{WC}$	Byte Write Cycle Time		2.0	ms

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

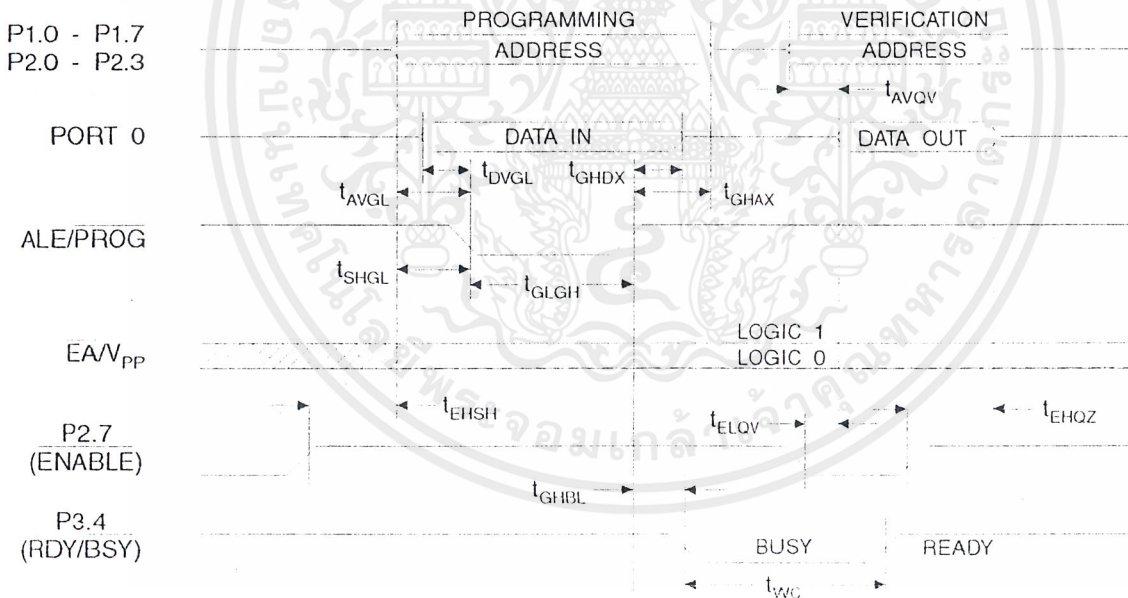


เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## Flash Programming and Verification Waveforms - High Voltage Mode ( $V_{PP} = 12V$ )



## Flash Programming and Verification Waveforms - Low Voltage Mode ( $V_{PP} = 5V$ )



**Absolute Maximum Ratings\***

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

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

**DC Characteristics**

T<sub>A</sub> = -40°C to 85°C, V<sub>CC</sub> = 5.0V ± 20% (unless otherwise noted)

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

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



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
 ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

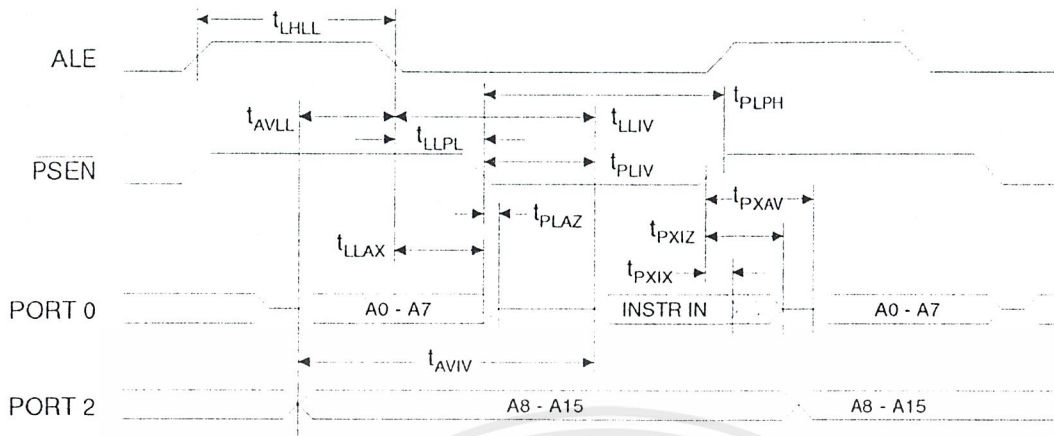
## AC Characteristics

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

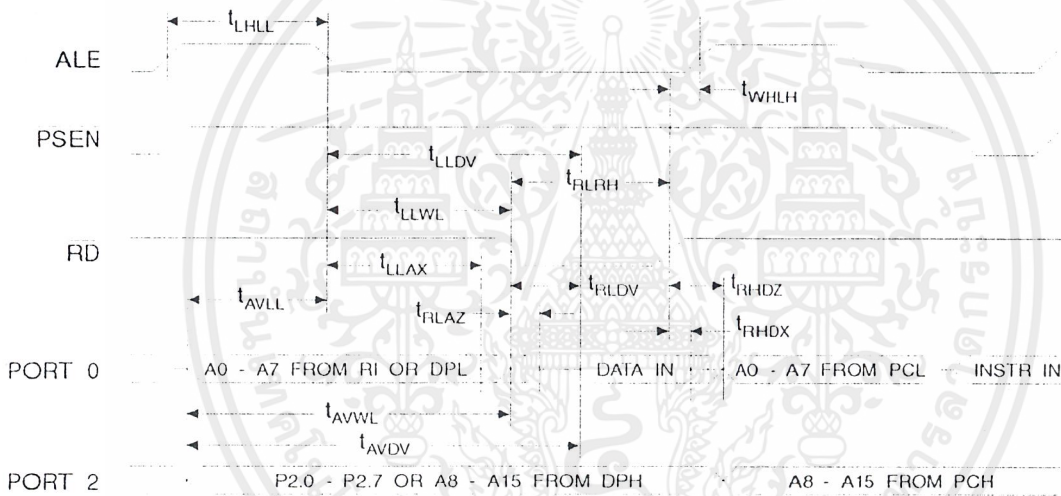
### External Program and Data Memory Characteristics

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

External Program Memory Read Cycle

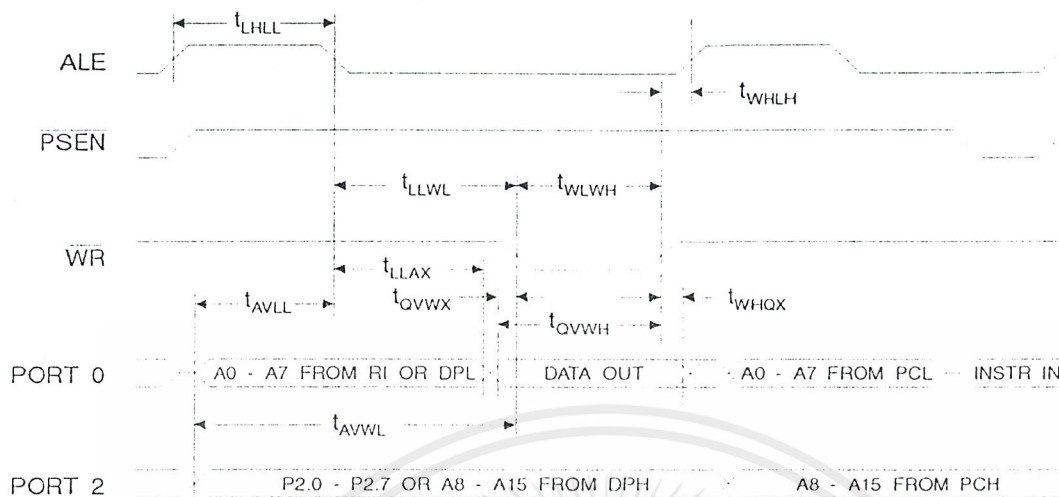


External Data Memory Read Cycle

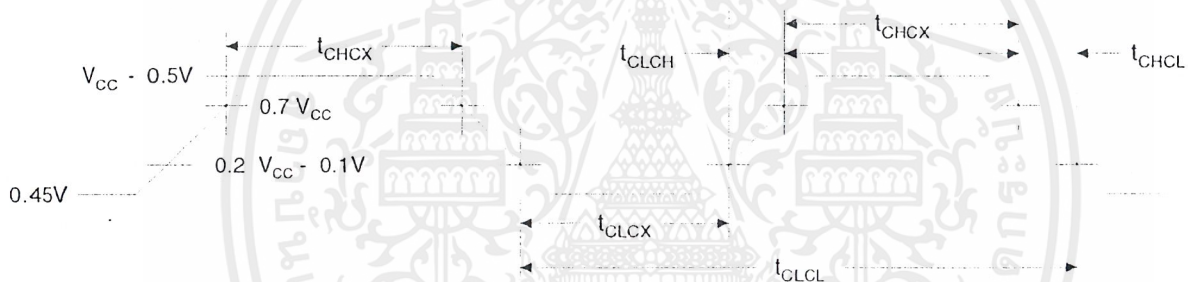


เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## External Data Memory Write Cycle



## External Clock Drive Waveforms



## External Clock Drive

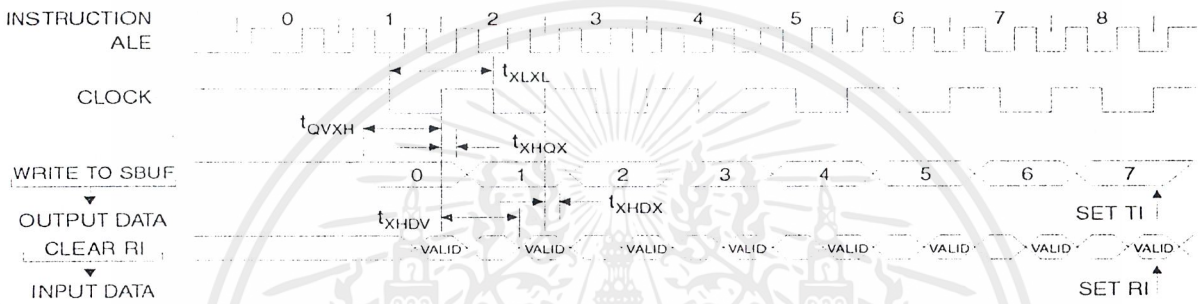
Symbol	Parameter	Min	Max	Units
1/t <sub>CLCL</sub>	Oscillator Frequency	0	24	MHz
t <sub>CLCL</sub>	Clock Period	41.6		ns
t <sub>CHCX</sub>	High Time	15		ns
t <sub>CLCX</sub>	Low Time	15		ns
t <sub>CLCH</sub>	Rise Time		20	ns
t <sub>CHCL</sub>	Fall Time		20	ns

Serial Port Timing: Shift Register Mode Test Conditions

(V<sub>CC</sub> = 5.0 V ± 20%; Load Capacitance = 80 pF)

Symbol	Parameter	12 MHz Osc		Variable Oscillator		Units
		Min	Max	Min	Max	
t <sub>XLXL</sub>	Serial Port Clock Cycle Time	1.0		12t <sub>CLCL</sub>		μs
t <sub>QVXH</sub>	Output Data Setup to Clock Rising Edge	700		10t <sub>CLCL</sub> -133		ns
t <sub>XHOX</sub>	Output Data Hold After Clock Rising Edge	50		2t <sub>CLCL</sub> -117		ns
t <sub>XHDX</sub>	Input Data Hold After Clock Rising Edge	0		0		ns
t <sub>XHDV</sub>	Clock Rising Edge to Input Data Valid		700		10t <sub>CLCL</sub> -133	ns

Shift Register Mode Timing Waveforms



AC Testing Input/Output Waveforms<sup>(1)</sup>

Float Waveforms<sup>(1)</sup>



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

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



## Ordering Information

Speed (MHz)	Power Supply	Ordering Code	Package	Operation Range	
12	5V ± 20%	AT89C51-12AC	44A	Commercial (0°C to 70°C)	
		AT89C51-12JC	44J		
		AT89C51-12PC	40P6		
		AT89C51-12QC	44Q		
			AT89C51-12AI	44A	Industrial (-40°C to 85°C)
			AT89C51-12JI	44J	
			AT89C51-12PI	40P6	
			AT89C51-12QI	44Q	
			AT89C51-12AA	44A	Automotive (-40°C to 105°C)
			AT89C51-12JA	44J	
			AT89C51-12PA	40P6	
			AT89C51-12QA	44Q	
16	5V ± 20%	AT89C51-16AC	44A	Commercial (0°C to 70°C)	
		AT89C51-16JC	44J		
		AT89C51-16PC	40P6		
		AT89C51-16QC	44Q		
			AT89C51-16AI	44A	Industrial (-40°C to 85°C)
			AT89C51-16JI	44J	
			AT89C51-16PI	40P6	
			AT89C51-16QI	44Q	
			AT89C51-16AA	44A	Automotive (-40°C to 105°C)
			AT89C51-16JA	44J	
			AT89C51-16PA	40P6	
			AT89C51-16QA	44Q	
20	5V ± 20%	AT89C51-20AC	44A	Commercial (0°C to 70°C)	
		AT89C51-20JC	44J		
		AT89C51-20PC	40P6		
		AT89C51-20QC	44Q		
			AT89C51-20AI	44A	Industrial (-40°C to 85°C)
			AT89C51-20JI	44J	
			AT89C51-20PI	40P6	
			AT89C51-20QI	44Q	

## ADC0801/ADC0802/ADC0803/ADC0804/ADC0805 8-Bit $\mu$ P Compatible A/D Converters

### General Description

The ADC0801, ADC0802, ADC0803, ADC0804 and ADC0805 are CMOS 8-bit successive approximation A/D converters that use a differential potentiometric ladder—similar to the 256R products. These converters are designed to allow operation with the NSC800 and INS8080A derivative control bus with TRI-STATE® output latches directly driving the data bus. These A/Ds appear like memory locations or I/O ports to the microprocessor and no interfacing logic is needed.

Differential analog voltage inputs allow increasing the common-mode rejection and offsetting the analog zero input voltage value. In addition, the voltage reference input can be adjusted to allow encoding any smaller analog voltage span to the full 8 bits of resolution.

### Features

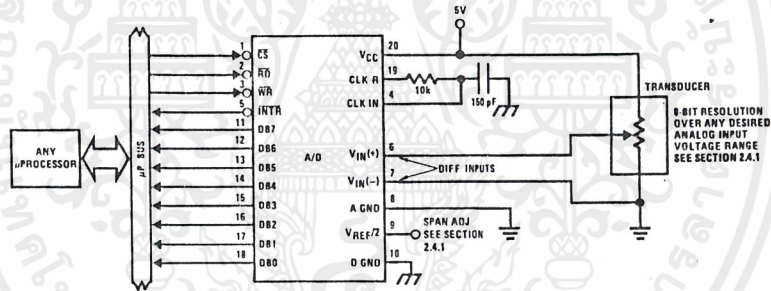
- Compatible with 8080  $\mu$ P derivatives—no interfacing logic needed - access time - 135 ns
- Easy interface to all microprocessors, or operates "stand alone"

- Differential analog voltage inputs
- Logic inputs and outputs meet both MOS and TTL voltage level specifications
- Works with 2.5V (LM336) voltage reference
- On-chip clock generator
- 0V to 5V analog input voltage range with single 5V supply
- No zero adjust required
- 0.3" standard width 20-pin DIP package
- 20-pin molded chip carrier or small outline package
- Operates ratiometrically or with 5  $V_{DC}$ , 2.5  $V_{DC}$ , or analog span adjusted voltage reference

### Key Specifications

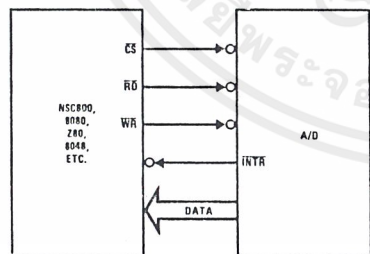
- Resolution 8 bits
- Total error  $\pm 1/4$  LSB,  $\pm 1/2$  LSB and  $\pm 1$  LSB
- Conversion time 100  $\mu$ s

### Typical Applications



TL/H/5671-1

### 8080 Interface



TL/H/5671-31

### Error Specification (Includes Full-Scale, Zero Error, and Non-Linearity)

Part Number	Full-Scale Adjusted	$V_{REF}/2 = 2.500 V_{DC}$ (No Adjustments)	$V_{REF}/2 = \text{No Connection}$ (No Adjustments)
ADC0801	$\pm 1/4$ LSB		
ADC0802		$\pm 1/2$ LSB	
ADC0803	$\pm 1/2$ LSB		
ADC0804		$\pm 1$ LSB	
ADC0805			$\pm 1$ LSB

### Absolute Maximum Ratings (Notes 1 & 2)

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

Supply Voltage ( $V_{CC}$ ) (Note 3)	6.5V
Voltage	
Logic Control Inputs	-0.3V to +18V
At Other Input and Outputs	-0.3V to ( $V_{CC} + 0.3V$ )
Lead Temp. (Soldering, 10 seconds)	
Dual-In-Line Package (plastic)	260°C
Dual-In-Line Package (ceramic)	300°C
Surface Mount Package	
Vapor Phase (60 seconds)	215°C
Infrared (15 seconds)	220°C

Storage Temperature Range	-65°C to +150°C
Package Dissipation at $T_A = 25^\circ\text{C}$	875 mW
ESD Susceptibility (Note 10)	800V

### Operating Ratings (Notes 1 & 2)

Temperature Range	$T_{MIN} \leq T_A \leq T_{MAX}$
ADC0801/02LJ, ADC0802LJ/883	$-55^\circ\text{C} \leq T_A \leq +125^\circ\text{C}$
ADC0801/02/03/04LCJ	$-40^\circ\text{C} \leq T_A \leq +85^\circ\text{C}$
ADC0801/02/03/05LCN	$-40^\circ\text{C} \leq T_A \leq +85^\circ\text{C}$
ADC0804LCN	$0^\circ\text{C} \leq T_A \leq +70^\circ\text{C}$
ADC0802/03/04LCV	$0^\circ\text{C} \leq T_A \leq +70^\circ\text{C}$
ADC0802/03/04LCWM	$0^\circ\text{C} \leq T_A \leq +70^\circ\text{C}$
Range of $V_{CC}$	$4.5 V_{DC}$ to $6.3 V_{DC}$

### Electrical Characteristics

The following specifications apply for  $V_{CC} = 5 V_{DC}$ ,  $T_{MIN} \leq T_A \leq T_{MAX}$  and  $f_{CLK} = 640$  kHz unless otherwise specified.

Parameter	Conditions	Min	Typ	Max	Units
ADC0801: Total Adjusted Error (Note 8)	With Full-Scale Adj. (See Section 2.5.2)			$\pm 1/4$	LSB
ADC0802: Total Unadjusted Error (Note 8)	$V_{REF}/2 = 2.500 V_{DC}$			$\pm 1/2$	LSB
ADC0803: Total Adjusted Error (Note 8)	With Full-Scale Adj. (See Section 2.5.2)			$\pm 1/2$	LSB
ADC0804: Total Unadjusted Error (Note 8)	$V_{REF}/2 = 2.500 V_{DC}$			$\pm 1$	LSB
ADC0805: Total Unadjusted Error (Note 8)	$V_{REF}/2$ -No Connection			$\pm 1$	LSB
$V_{REF}/2$ Input Resistance (Pin 9)	ADC0801/02/03/05 ADC0804 (Note 9)	2.5 0.75	8.0 1.1		k $\Omega$ k $\Omega$
Analog Input Voltage Range	(Note 4) $V(+)$ or $V(-)$	Gnd-0.05		$V_{CC} + 0.05$	$V_{DC}$
DC Common-Mode Error	Over Analog Input Voltage Range		$\pm 1/16$	$\pm 1/8$	LSB
Power Supply Sensitivity	$V_{CC} = 5 V_{DC} \pm 10\%$ Over Allowed $V_{IN}(+)$ and $V_{IN}(-)$ Voltage Range (Note 4)		$\pm 1/16$	$\pm 1/8$	LSB

### AC Electrical Characteristics

The following specifications apply for  $V_{CC} = 5 V_{DC}$  and  $T_A = 25^\circ\text{C}$  unless otherwise specified.

Symbol	Parameter	Conditions	Min	Typ	Max	Units
$T_C$	Conversion Time	$f_{CLK} = 640$ kHz (Note 6)	103		114	$\mu\text{s}$
$T_C$	Conversion Time	(Note 5, 6)	66		73	$1/f_{CLK}$
$f_{CLK}$	Clock Frequency Clock Duty Cycle	$V_{CC} = 5V$ , (Note 5) (Note 5)	100 40	640	1460 60	kHz %
CR	Conversion Rate in Free-Running Mode	INTR tied to WR with $\overline{CS} = 0 V_{DC}$ , $f_{CLK} = 640$ kHz	8770		9708	conv/s
$t_{W(WR)L}$	Width of WR Input (Start Pulse Width)	$\overline{CS} = 0 V_{DC}$ (Note 7)	100			ns
$t_{ACC}$	Access Time (Delay from Falling Edge of RD to Output Data Valid)	$C_L = 100$ pF		135	200	ns
$t_{1H}, t_{0H}$	TRI-STATE Control (Delay from Rising Edge of RD to Hi-Z State)	$C_L = 10$ pF, $R_L = 10k$ (See TRI-STATE Test Circuits)		125	200	ns
$t_{WI}, t_{RI}$	Delay from Falling Edge of WR or RD to Reset of INTR			300	450	ns
$C_{IN}$	Input Capacitance of Logic Control Inputs			5	7.5	pF
$C_{OUT}$	TRI-STATE Output Capacitance (Data Buffers)			5	7.5	pF

**CONTROL INPUTS** (Note: CLK IN (Pin 4) is the input of a Schmitt trigger circuit and is therefore specified separately)

$V_{IN}(1)$	Logical "1" Input Voltage (Except Pin 4 CLK IN)	$V_{CC} = 5.25 V_{DC}$	2.0		15	$V_{DC}$
-------------	---	------------------------	-----	--	----	----------

**AC Electrical Characteristics** (Continued)The following specifications apply for  $V_{CC} = 5V_{DC}$  and  $T_{MIN} \leq T_A \leq T_{MAX}$ , unless otherwise specified.

Symbol	Parameter	Conditions	Min	Typ	Max	Units
<b>CONTROL INPUTS</b> (Note: CLK IN (Pin 4) is the input of a Schmitt trigger circuit and is therefore specified separately)						
$V_{IN(0)}$	Logical "0" Input Voltage (Except Pin 4 CLK IN)	$V_{CC} = 4.75 V_{DC}$			0.8	$V_{DC}$
$I_{IN(1)}$	Logical "1" Input Current (All Inputs)	$V_{IN} = 5 V_{DC}$		0.005	1	$\mu A_{DC}$
$I_{IN(0)}$	Logical "0" Input Current (All Inputs)	$V_{IN} = 0 V_{DC}$	-1	-0.005		$\mu A_{DC}$
<b>CLOCK IN AND CLOCK R</b>						
$V_{T+}$	CLK IN (Pin 4) Positive Going Threshold Voltage		2.7	3.1	3.5	$V_{DC}$
$V_{T-}$	CLK IN (Pin 4) Negative Going Threshold Voltage		1.5	1.8	2.1	$V_{DC}$
$V_H$	CLK IN (Pin 4) Hysteresis ( $V_{T+} - V_{T-}$ )		0.6	1.3	2.0	$V_{DC}$
$V_{OUT(0)}$	Logical "0" CLK R Output Voltage	$I_O = 360 \mu A$ $V_{CC} = 4.75 V_{DC}$			0.4	$V_{DC}$
$V_{OUT(1)}$	Logical "1" CLK R Output Voltage	$I_O = -360 \mu A$ $V_{CC} = 4.75 V_{DC}$	2.4			$V_{DC}$
<b>DATA OUTPUTS AND INTR</b>						
$V_{OUT(0)}$	Logical "0" Output Voltage Data Outputs INTR Output	$I_{OUT} = 1.6 mA, V_{CC} = 4.75 V_{DC}$ $I_{OUT} = 1.0 mA, V_{CC} = 4.75 V_{DC}$			0.4 0.4	$V_{DC}$ $V_{DC}$
$V_{OUT(1)}$	Logical "1" Output Voltage	$I_O = -360 \mu A, V_{CC} = 4.75 V_{DC}$	2.4			$V_{DC}$
$V_{OUT(1)}$	Logical "1" Output Voltage	$I_O = -10 \mu A, V_{CC} = 4.75 V_{DC}$	4.5			$V_{DC}$
$I_{OUT}$	TRI-STATE Disabled Output Leakage (All Data Buffers)	$V_{OUT} = 0 V_{DC}$ $V_{OUT} = 5 V_{DC}$	-3		3	$\mu A_{DC}$ $\mu A_{DC}$
$I_{SOURCE}$		$V_{OUT}$ Short to Gnd, $T_A = 25^\circ C$	4.5	6		$mA_{DC}$
$I_{SINK}$		$V_{OUT}$ Short to $V_{CC}$ , $T_A = 25^\circ C$	9.0	16		$mA_{DC}$
<b>POWER SUPPLY</b>						
$I_{CC}$	Supply Current (Includes Ladder Current)  ADC0801/02/03/04LCJ/05 ADC0804LCN/LCV/LCWM	$f_{CLK} = 640 kHz$ , $V_{REF/2} = NC$ , $T_A = 25^\circ C$ and $CS = 5V$		1.1 1.9	1.8 2.5	$mA$ $mA$

Note 1: Absolute Maximum Ratings indicate limits beyond which damage to the device may occur. DC and AC electrical specifications do not apply when operating the device beyond its specified operating conditions.

Note 2: All voltages are measured with respect to Gnd, unless otherwise specified. The separate A Gnd point should always be wired to the D Gnd.

Note 3: A zener diode exists, internally, from  $V_{CC}$  to Gnd and has a typical breakdown voltage of 7  $V_{DC}$ .

Note 4: For  $V_{IN(-)} \geq V_{IN(+)}$  the digital output code will be 0000 0000. Two on-chip diodes are tied to each analog input (see block diagram) which will forward conduct for analog input voltages one diode drop below ground or one diode drop greater than the  $V_{CC}$  supply. Be careful, during testing at low  $V_{CC}$  levels (4.5V), as high level analog inputs (5V) can cause this input diode to conduct—especially at elevated temperatures, and cause errors for analog inputs near full-scale. The spec allows 50 mV forward bias of either diode. This means that as long as the analog  $V_{IN}$  does not exceed the supply voltage by more than 50 mV, the output code will be correct. To achieve an absolute 0  $V_{DC}$  to 5  $V_{DC}$  input voltage range will therefore require a minimum supply voltage of 4.950  $V_{DC}$  over temperature variations, initial tolerance and loading.

Note 5: Accuracy is guaranteed at  $f_{CLK} = 640 kHz$ . At higher clock frequencies accuracy can degrade. For lower clock frequencies, the duty cycle limits can be extended so long as the minimum clock high time interval or minimum clock low time interval is no less than 275 ns.

Note 6: With an asynchronous start pulse, up to 8 clock periods may be required before the internal clock phases are proper to start the conversion process. The start request is internally latched, see Figure 2 and section 2.0.

Note 7: The CS input is assumed to bracket the WR strobe input and therefore timing is dependent on the WR pulse width. An arbitrarily wide pulse width will hold the converter in a reset mode and the start of conversion is initiated by the low to high transition of the WR pulse (see timing diagrams).

Note 8: None of these A/Ds requires a zero adjust (see section 2.5.1). To obtain zero code at other analog input voltages see section 2.5 and Figure 5.

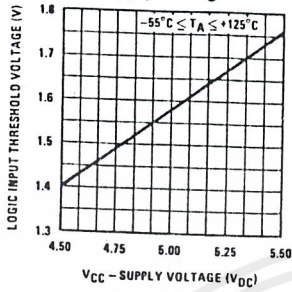
Note 9: The  $V_{REF/2}$  pin is the center point of a two-resistor divider connected from  $V_{CC}$  to ground. In all versions of the ADC0801, ADC0802, ADC0803, and ADC0805, and in the ADC0804LCJ, each resistor is typically 16 k $\Omega$ . In all versions of the ADC0804 except the ADC0804LCJ, each resistor is typically 2.2 k $\Omega$ .

Note 10: Human body model, 100 pF discharged through a 1.5 k $\Omega$  resistor.

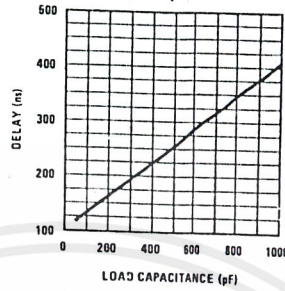
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## Typical Performance Characteristics

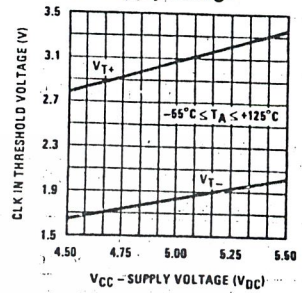
Logic Input Threshold Voltage vs. Supply Voltage



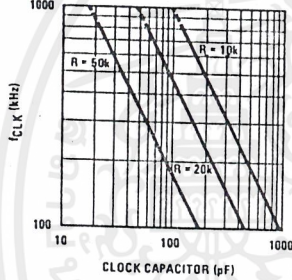
Delay From Falling Edge of RD to Output Data Valid vs. Load Capacitance



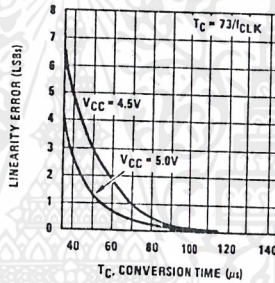
CLK IN Schmitt Trip Levels vs. Supply Voltage



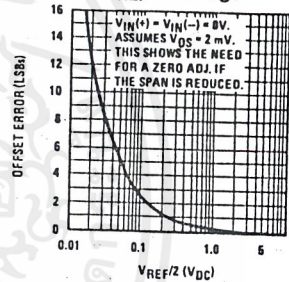
fCLK vs. Clock Capacitor



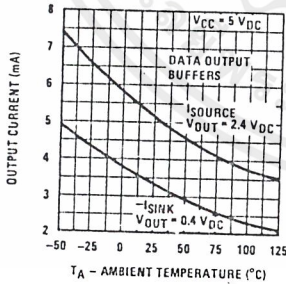
Full-Scale Error vs Conversion Time



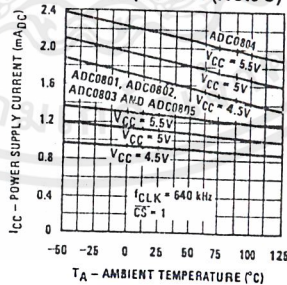
Effect of Unadjusted Offset Error vs. VREF/2 Voltage



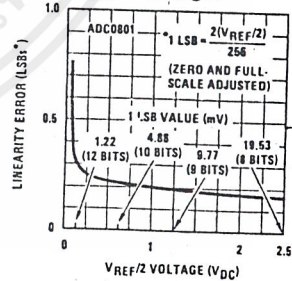
Output Current vs Temperature



Power Supply Current vs Temperature (Note 9)

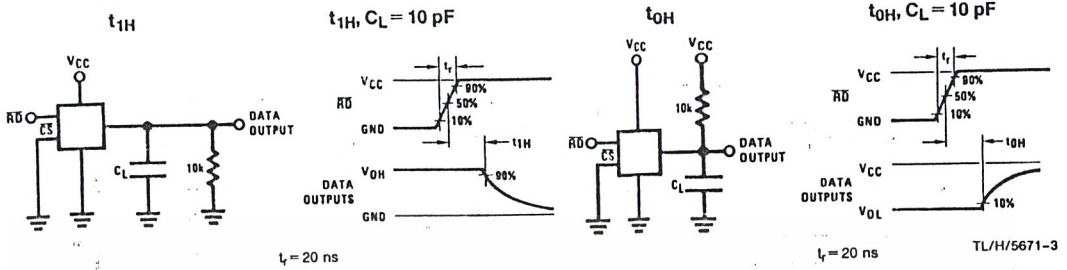


Linearity Error at Low VREF/2 Voltages

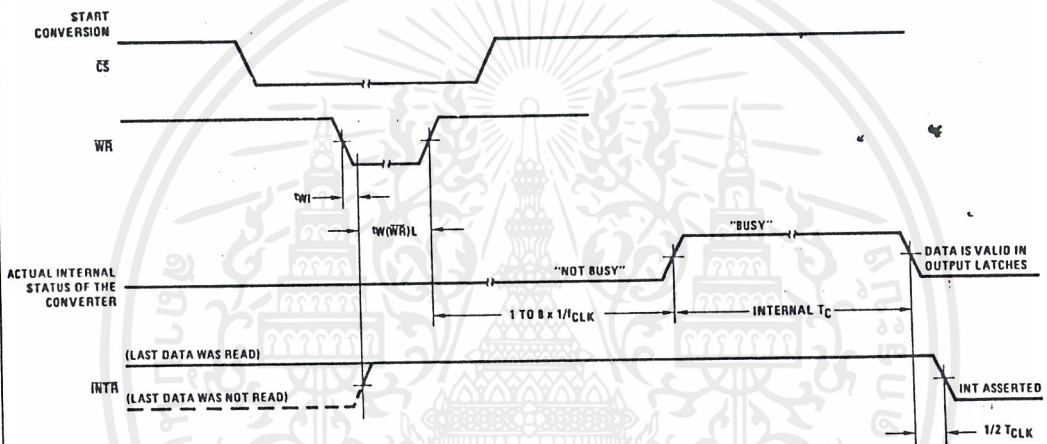


เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

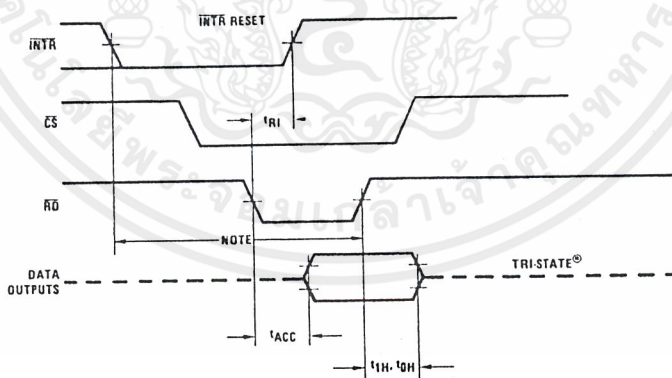
### TRI-STATE Test Circuits and Waveforms



### Timing Diagrams (All timing is measured from the 50% voltage points)



### Output Enable and Reset $\overline{INTR}$



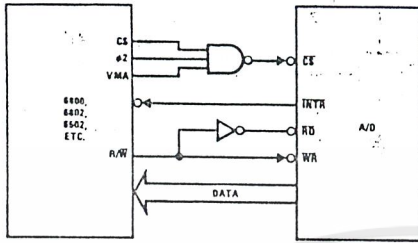
Note: Read strobe must occur 8 clock periods ( $8/1/\text{CLK}$ ) after assertion of interrupt to guarantee reset of  $\overline{INTR}$ .

TL/H/5671-4

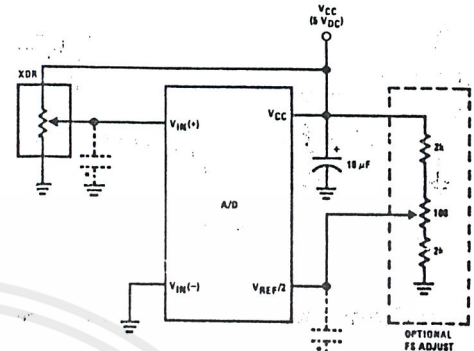
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Typical Applications (Continued)

6800 Interface

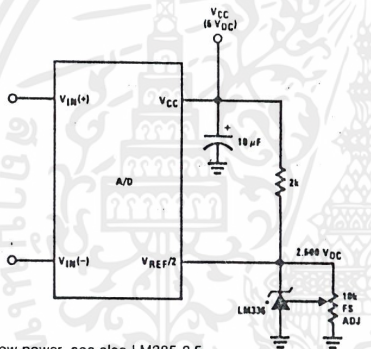


Ratiometric with Full-Scale Adjust



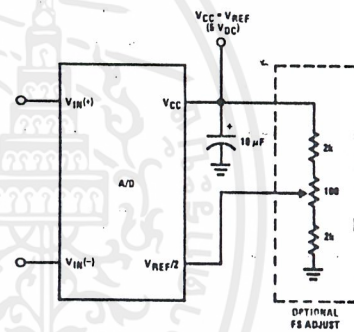
Note: before using caps at  $V_{IN}$  or  $V_{REF/2}$ , see section 2.3.2 Input Bypass Capacitors.

Absolute with a 2.500V Reference

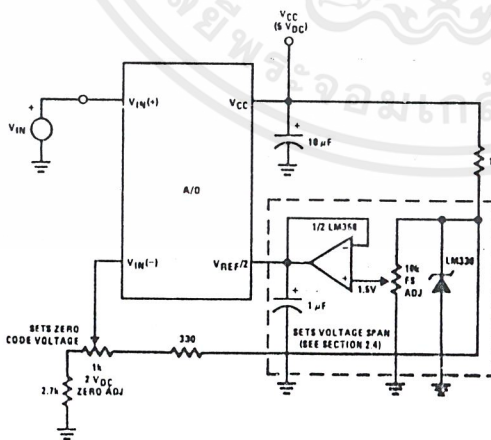


\*For low power, see also LM385-2.5

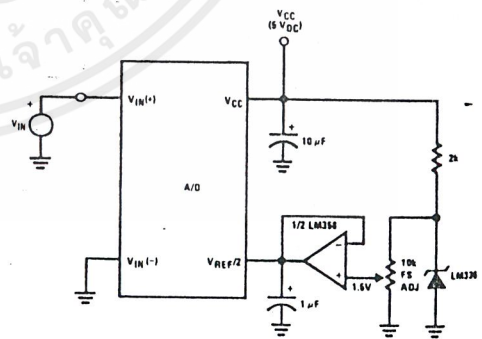
Absolute with a 5V Reference



Zero-Shift and Span Adjust:  $2V \leq V_{IN} \leq 5V$



Span Adjust:  $0V \leq V_{IN} \leq 3V$

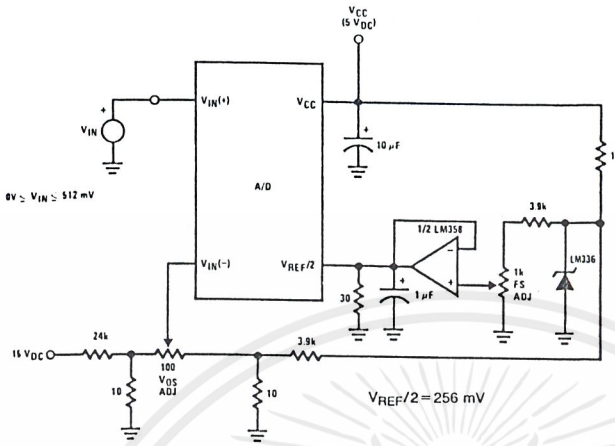


TL/H/5671-5

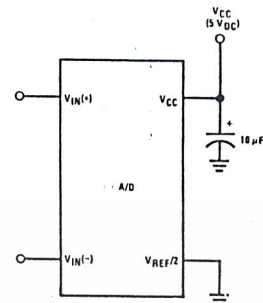
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Typical Applications (Continued)

Directly Converting a Low-Level Signal

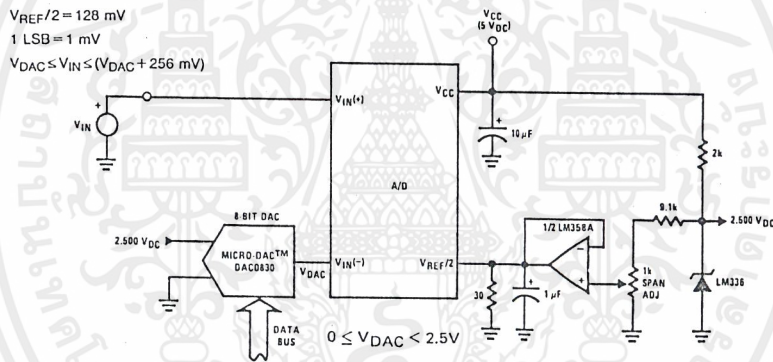


A  $\mu\text{P}$  Interfaced Comparator

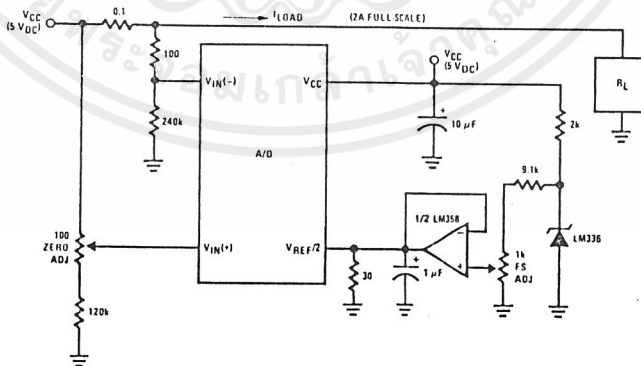


For:  $V_{IN}(+) > V_{IN}(-)$   
Output = FF<sub>HEX</sub>  
For:  $V_{IN}(+) < V_{IN}(-)$   
Output = 00<sub>HEX</sub>

1 mV Resolution with  $\mu\text{P}$  Controlled Range



Digitizing a Current Flow

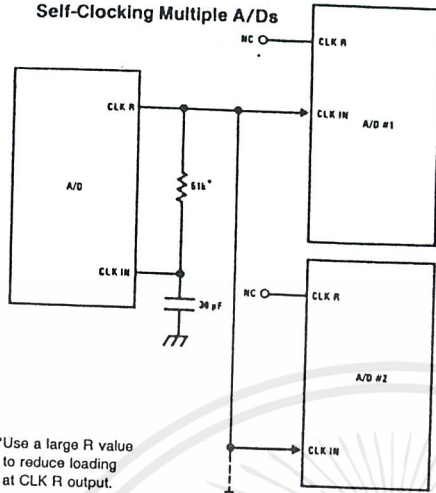


TL/H/5671-6

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### Typical Applications (Continued)

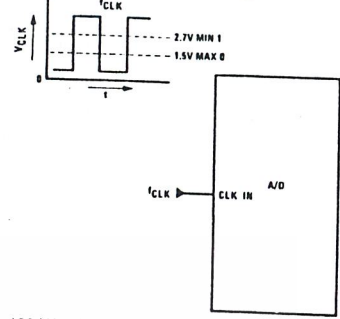
#### Self-Clocking Multiple A/Ds



\*Use a large R value to reduce loading at CLK R output.

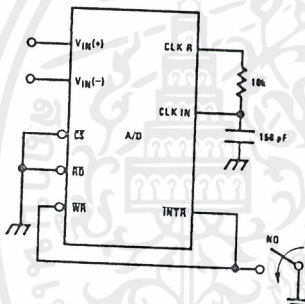
IF MORE THAN 6 ADDITIONAL A/Ds, USE A CMOS BUFFER (NOT TTL)

#### External Clocking



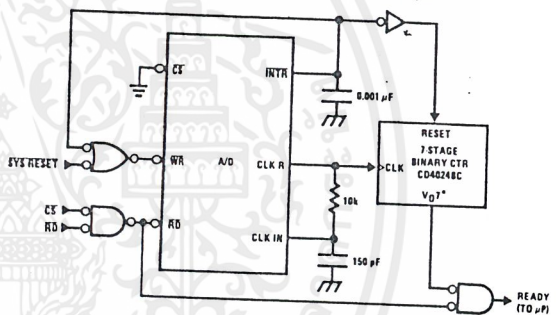
$100 \text{ kHz} \leq f_{\text{CLK}} \leq 1460 \text{ kHz}$

#### Self-Clocking in Free-Running Mode



\*After power-up, a momentary grounding of the WR input is needed to guarantee operation.

#### μP Interface for Free-Running A/D

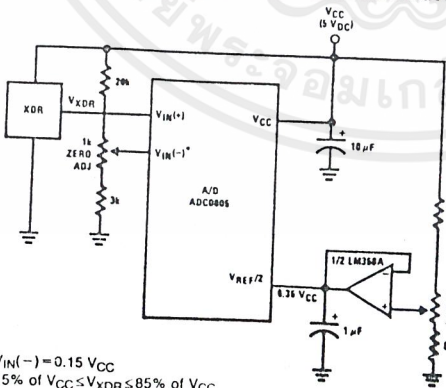


\*  $V_{07}^*$  PREVENTS RD DURING A/D DATA UPDATE

RESET  $(2 \times 1/f_{\text{CLK}})$

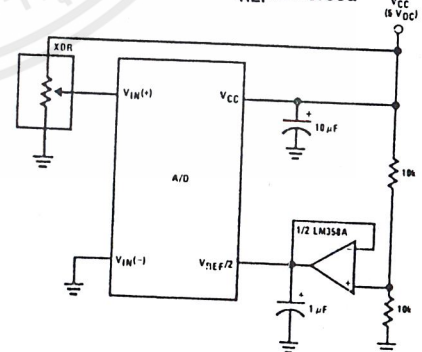
RESET  $(8 \times 1/f_{\text{CLK}})$

#### Operating with "Automotive" Resistometric Transducers



\* $V_{\text{IN}}(-) = 0.15 V_{\text{CC}}$   
 $15\% \text{ of } V_{\text{CC}} \leq V_{\text{XDR}} \leq 85\% \text{ of } V_{\text{CC}}$

#### Resistometric with $V_{\text{REF}}/2$ Forced

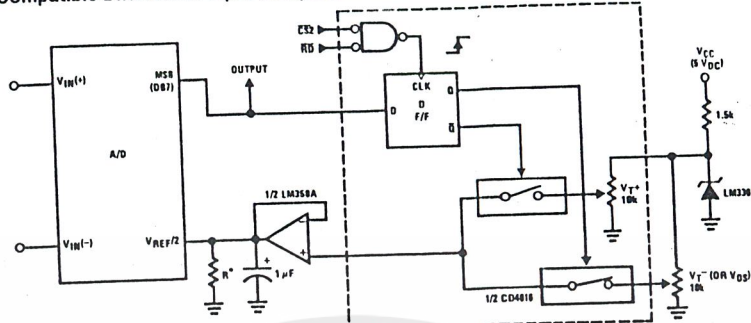


TL/H/5671-7

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
 ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

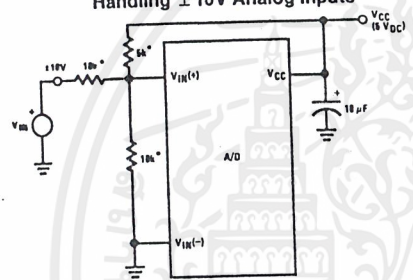
### Typical Applications (Continued)

$\mu$ P Compatible Differential-Input Comparator with Pre-Set  $V_{OS}$  (with or without Hysteresis)



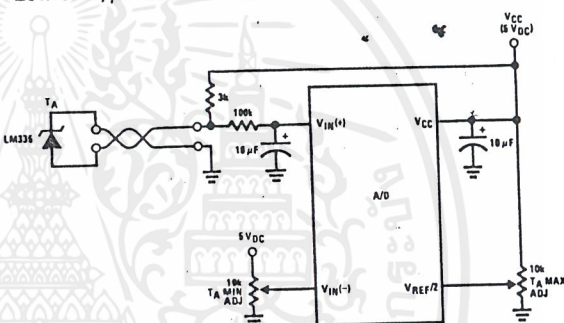
\*See Figure 5 to select R value  
 DB7 = "1" for  $V_{IN}(+) > V_{IN}(-) + (V_{REF}/2)$   
 Omit circuitry within the dotted area if hysteresis is not needed

Handling  $\pm 10V$  Analog Inputs

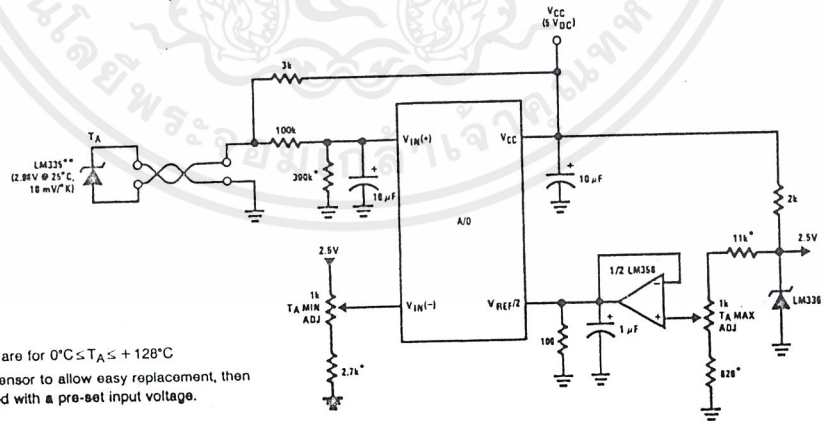


\*Bockman Instruments # 694-3-R10K resistor array

Low-Cost,  $\mu$ P Interfaced, Temperature-to-Digital Converter



$\mu$ P Interfaced Temperature-to-Digital Converter



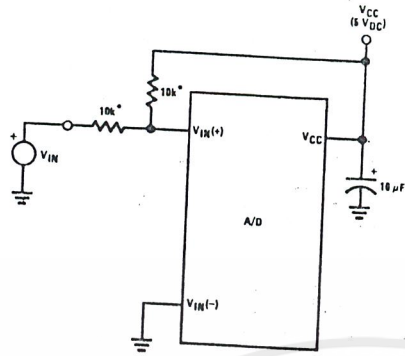
\*Circuit values shown are for  $0^{\circ}C \leq T_A \leq +128^{\circ}C$   
 \*\*Can calibrate each sensor to allow easy replacement, then A/D can be calibrated with a pre-set input voltage.

TL/H/5671-B

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
 ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## Typical Applications (Continued)

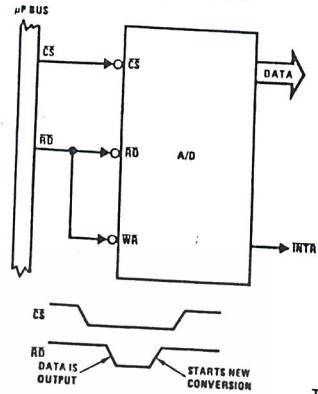
### Handling $\pm 5V$ Analog Inputs



TL/H/5671-33

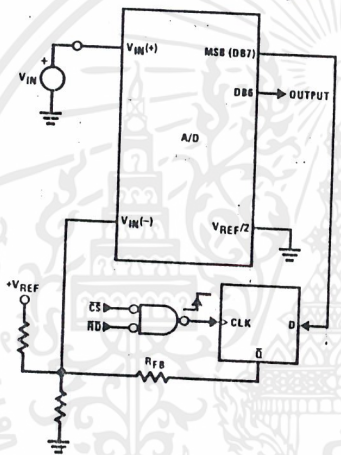
\*Beckman Instruments # 694-3-R10K resistor array

### Read-Only Interface



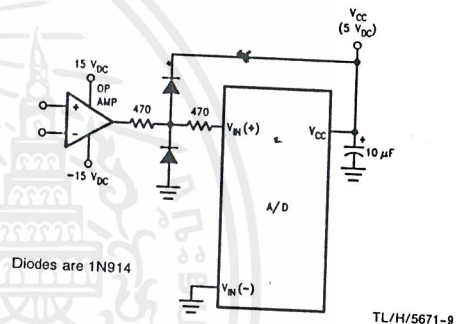
TL/H/5671-34

### µP Interfaced Comparator with Hysteresis



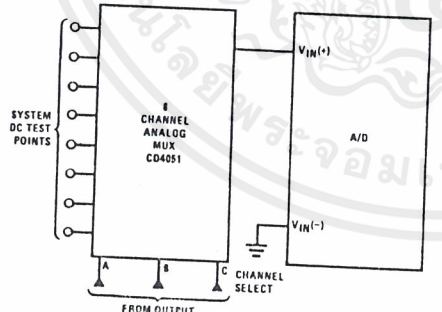
TL/H/5671-35

### Protecting the Input



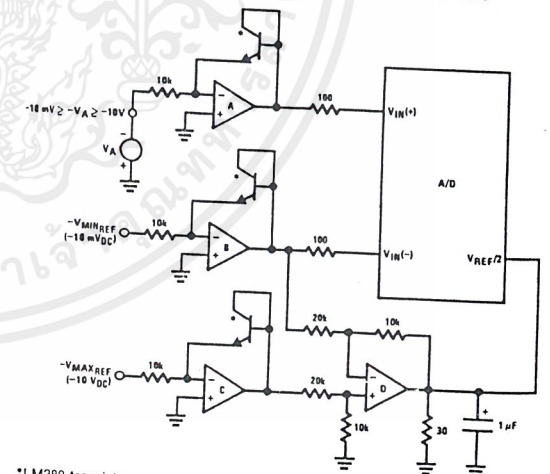
TL/H/5671-9

### Analog Self-Test for a System



TL/H/5671-36

### A Low-Cost, 3-Decade Logarithmic Converter



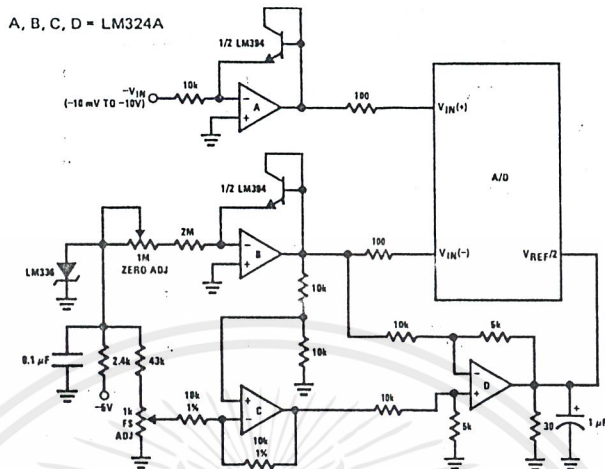
TL/H/5671-37

\*LM369 transistors

A, B, C, D = LM324A quad op amp

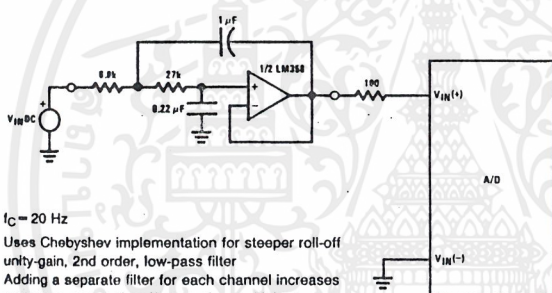
Typical Applications (Continued)

3-Decade Logarithmic A/D Converter



5671-34

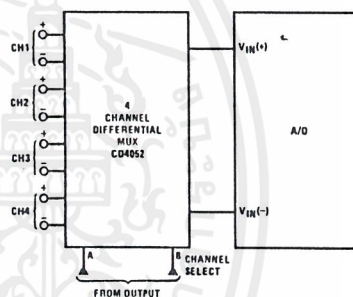
Noise Filtering the Analog Input



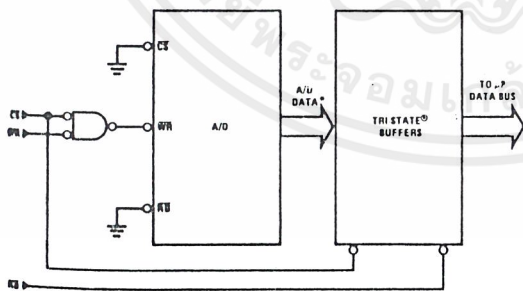
$f_c = 20 \text{ Hz}$   
 Uses Chebyshev implementation for steeper roll-off unity-gain, 2nd order, low-pass filter  
 Adding a separate filter for each channel increases system response time if an analog multiplexer is used

5671-9

Multiplexing Differential Inputs

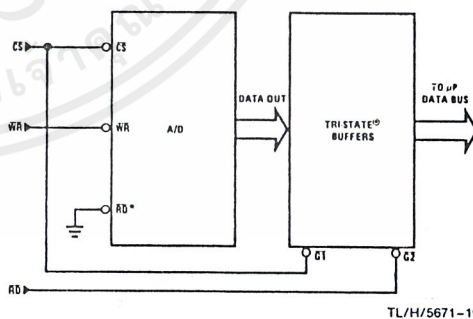


Output Buffers with A/D Data Enabled



\*A/D output data is updated 1 CLK period prior to assertion of INTR

Increasing Bus Drive and/or Reducing Time on Bus



\*Allows output data to set-up at falling edge of CS

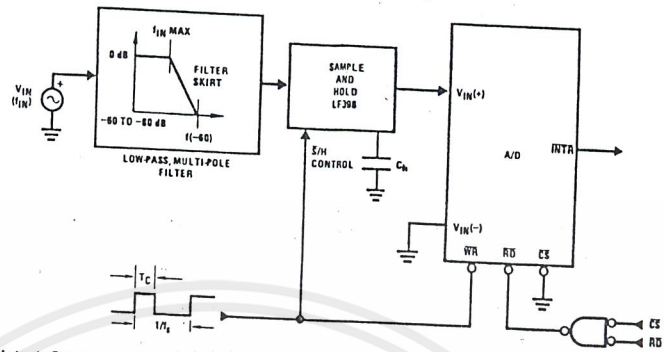
TL/H/5671-10

1-37

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
 ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

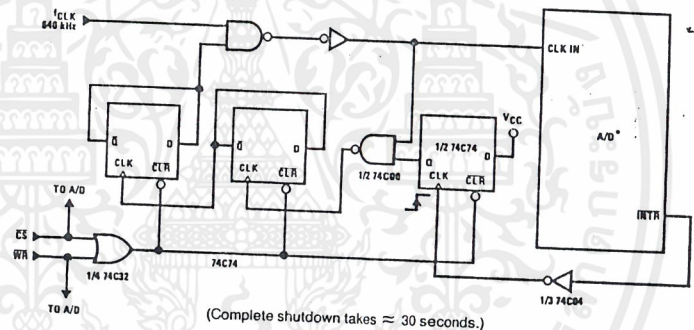
Typical Applications (Continued)

Sampling an AC Input Signal



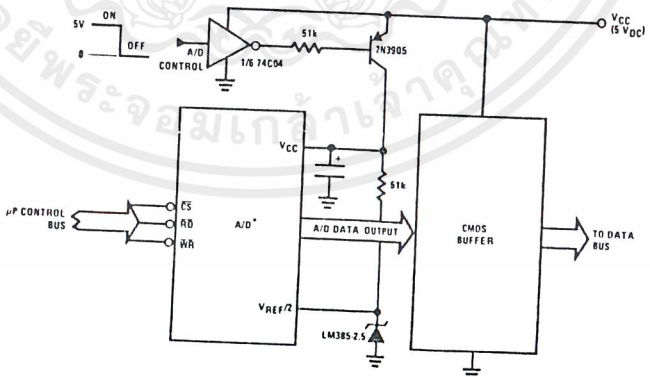
Note 1: Oversample whenever possible [keep  $f_s > 2f(-60)$ ] to eliminate input frequency folding (aliasing) and to allow for the skirt response of the filter.  
 Note 2: Consider the amplitude errors which are introduced within the passband of the filter.

70% Power Savings by Clock Gating



(Complete shutdown takes  $\approx 30$  seconds.)

Power Savings by A/D and V<sub>REF</sub> Shutdown



\*Use ADC0801, 02, 03 or 05 for lowest power consumption.  
 Note: Logic inputs can be driven to V<sub>CC</sub> with A/D supply at zero volts.  
 Buffer prevents data bus from overdriving output of A/D when in shutdown mode.

TL/H/5671-11

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
 ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## Functional Description

### 1.0 UNDERSTANDING A/D ERROR SPECS

A perfect A/D transfer characteristic (staircase waveform) is shown in *Figure 1a*. The horizontal scale is analog input voltage and the particular points labeled are in steps of 1 LSB (19.53 mV with 2.5V tied to the  $V_{REF}/2$  pin). The digital output codes that correspond to these inputs are shown as  $D-1$ ,  $D$ , and  $D+1$ . For the perfect A/D, not only will center-value ( $A-1$ ,  $A$ ,  $A+1$ , . . . ) analog inputs produce the correct output digital codes, but also each riser (the transitions between adjacent output codes) will be located  $\pm 1/2$  LSB away from each center-value. As shown, the risers are ideal and have no width. Correct digital output codes will be provided for a range of analog input voltages that extend  $\pm 1/2$  LSB from the ideal center-values. Each tread (the range of analog input voltage that provides the same digital output code) is therefore 1 LSB wide.

*Figure 1b* shows a worst case error plot for the ADC0801. All center-valued inputs are guaranteed to produce the correct output codes and the adjacent risers are guaranteed to be no closer to the center-value points than  $\pm 1/4$  LSB. In

other words, if we apply an analog input equal to the center-value  $\pm 1/4$  LSB, we guarantee that the A/D will produce the correct digital code. The maximum range of the position of the code transition is indicated by the horizontal arrow and it is guaranteed to be no more than  $1/2$  LSB.

The error curve of *Figure 1c* shows a worst case error plot for the ADC0802. Here we guarantee that if we apply an analog input equal to the LSB analog voltage center-value the A/D will produce the correct digital code.

Next to each transfer function is shown the corresponding error plot. Many people may be more familiar with error plots than transfer functions. The analog input voltage to the A/D is provided by either a linear ramp or by the discrete output steps of a high resolution DAC. Notice that the error is continuously displayed and includes the quantization uncertainty of the A/D. For example the error at point 1 of *Figure 1a* is  $+1/2$  LSB because the digital code appeared  $1/2$  LSB in advance of the center-value of the tread. The error plots always have a constant negative slope and the abrupt up-side steps are always 1 LSB in magnitude.

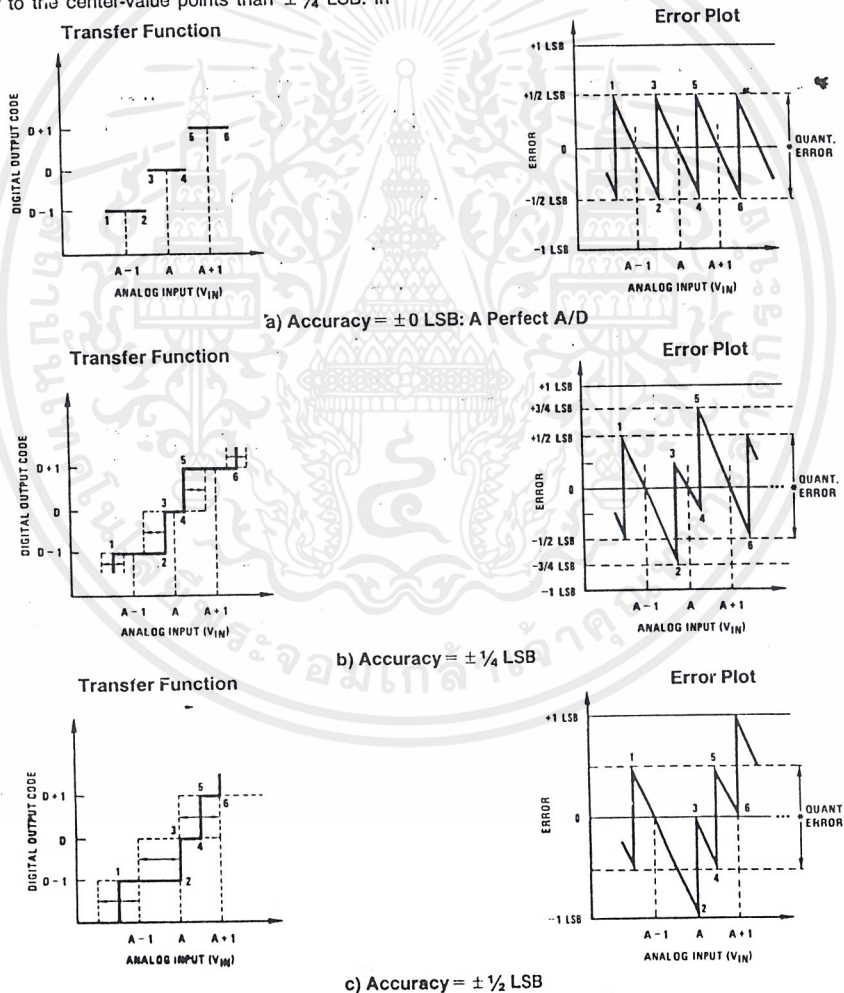


FIGURE 1. Clarifying the Error Specs of an A/D Converter

TL/H/5671-12

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## Functional Description (Continued)

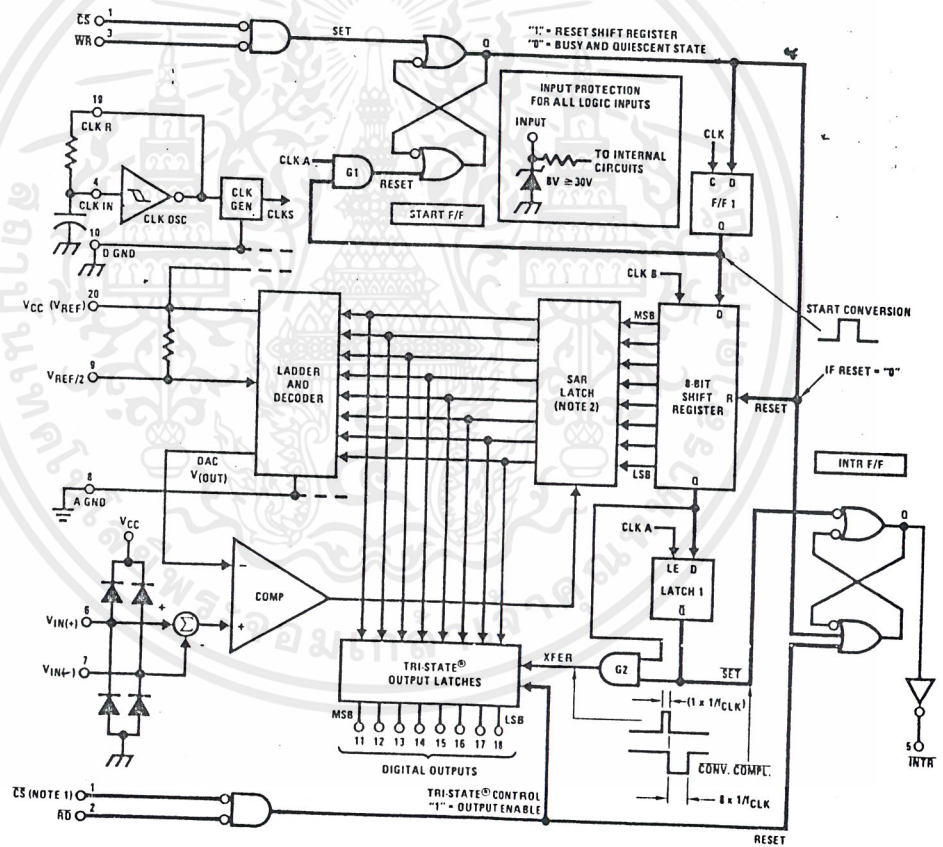
### 2.0 FUNCTIONAL DESCRIPTION

The ADC0801 series contains a circuit equivalent of the 256R network. Analog switches are sequenced by successive approximation logic to match the analog difference input voltage  $[V_{IN(+)} - V_{IN(-)}]$  to a corresponding tap on the R network. The most significant bit is tested first and after 8 comparisons (64 clock cycles) a digital 8-bit binary code (1111 1111 = full-scale) is transferred to an output latch and then an interrupt is asserted ( $\overline{INTR}$  makes a high-to-low transition). A conversion in process can be interrupted by issuing a second start command. The device may be operated in the free-running mode by connecting  $\overline{INTR}$  to the  $\overline{WR}$  input with  $\overline{CS} = 0$ . To ensure start-up under all possible conditions, an external  $\overline{WR}$  pulse is required during the first power-up cycle.

On the high-to-low transition of the  $\overline{WR}$  input the internal SAR latches and the shift register stages are reset. As long as the  $\overline{CS}$  input and  $\overline{WR}$  input remain low, the A/D will remain in a reset state. Conversion will start from 1 to 8 clock periods after at least one of these inputs makes a low-to-high transition.

A functional diagram of the A/D converter is shown in Figure 2. All of the package pinouts are shown and the major logic control paths are drawn in heavier weight lines.

The converter is started by having  $\overline{CS}$  and  $\overline{WR}$  simultaneously low. This sets the start flip-flop (F/F) and the resulting "1" level resets the 8-bit shift register, resets the Interrupt ( $\overline{INTR}$ ) F/F and inputs a "1" to the D flop, F/F1, which is at the input end of the 8-bit shift register. Internal clock signals then transfer this "1" to the Q output of F/F1. The AND gate, G1, combines this "1" output with a clock signal to provide a reset signal to the start F/F. If the set signal is no longer present (either  $\overline{WR}$  or  $\overline{CS}$  is a "1") the start F/F is reset and the 8-bit shift register then can have the "1" clocked in, which starts the conversion process. If the set signal were to still be present, this reset pulse would have no effect (both outputs of the start F/F would momentarily be at a "1" level) and the 8-bit shift register would continue to be held in the reset mode. This logic therefore allows for wide  $\overline{CS}$  and  $\overline{WR}$  signals and the converter will start after at least one of these signals returns high and the internal clocks again provide a reset signal for the start F/F.



Note 1:  $\overline{CS}$  shown twice for clarity.

Note 2: SAR = Successive Approximation Register.

FIGURE 2. Block Diagram

TL/H/5671-13

**Functional Description (Continued)**

After the "1" is clocked through the 8-bit shift register (which completes the SAR search) it appears as the input to the D-type latch, LATCH 1. As soon as this "1" is output from the shift register, the AND gate, G2, causes the new digital word to transfer to the TRI-STATE output latches. When LATCH 1 is subsequently enabled, the Q output makes a high-to-low transition which causes the INTR F/F to set. An inverting buffer then supplies the  $\overline{\text{INTR}}$  input signal.

Note that this  $\overline{\text{SET}}$  control of the INTR F/F remains low for 8 of the external clock periods (as the internal clocks run at  $\frac{1}{2}$  of the frequency of the external clock). If the data output is continuously enabled ( $\overline{\text{CS}}$  and  $\overline{\text{RD}}$  both held low), the  $\overline{\text{INTR}}$  output will still signal the end of conversion (by a high-to-low transition), because the  $\overline{\text{SET}}$  input can control the Q output of the INTR F/F even though the RESET input is constantly at a "1" level in this operating mode. This INTR output will therefore stay low for the duration of the  $\overline{\text{SET}}$  signal, which is 8 periods of the external clock frequency (assuming the A/D is not started during this interval).

When operating in the free-running or continuous conversion mode ( $\overline{\text{INTR}}$  pin tied to  $\overline{\text{WR}}$  and  $\overline{\text{CS}}$  wired low—see also section 2.8), the START F/F is SET by the high-to-low transition of the  $\overline{\text{INTR}}$  signal. This resets the SHIFT REGISTER which causes the input to the D-type latch, LATCH 1, to go low. As the latch enable input is still present, the Q output will go high, which then allows the INTR F/F to be RESET. This reduces the width of the resulting  $\overline{\text{INTR}}$  output pulse to only a few propagation delays (approximately 300 ns).

When data is to be read, the combination of both  $\overline{\text{CS}}$  and  $\overline{\text{RD}}$  being low will cause the INTR F/F to be reset and the TRI-STATE output latches will be enabled to provide the 8-bit digital outputs.

**2.1 Digital Control Inputs**

The digital control inputs ( $\overline{\text{CS}}$ ,  $\overline{\text{RD}}$ , and  $\overline{\text{WR}}$ ) meet standard TTL logic voltage levels. These signals have been renamed when compared to the standard A/D Start and Output Enable labels. In addition, these inputs are active low to allow an easy interface to microprocessor control busses. For non-microprocessor based applications, the  $\overline{\text{CS}}$  input (pin 1) can be grounded and the standard A/D Start function is obtained by an active low pulse applied at the  $\overline{\text{WR}}$  input (pin 3) and the Output Enable function is caused by an active low pulse at the  $\overline{\text{RD}}$  input (pin 2).

**2.2 Analog Differential Voltage Inputs and Common-Mode Rejection**

The A/D has additional applications flexibility due to the analog differential voltage input. The  $V_{\text{IN}}(-)$  input (pin 7) can be used to automatically subtract a fixed voltage value from the input reading (tare correction). This is also useful in 4-20 mA-20 mA current loop conversion. In addition, common-mode noise can be reduced by use of the differential input. The time interval between sampling  $V_{\text{IN}}(+)$  and  $V_{\text{IN}}(-)$  is  $\frac{1}{2}$  clock periods. The maximum error voltage due to this

slight time difference between the input voltage samples is given by:

$$\Delta V_e(\text{MAX}) = (V_p)(2\pi f_{\text{cm}}) \left( \frac{4.5}{f_{\text{CLK}}} \right),$$

where:

- $\Delta V_e$  is the error voltage due to sampling delay
- $V_p$  is the peak value of the common-mode voltage
- $f_{\text{cm}}$  is the common-mode frequency

As an example, to keep this error to  $\frac{1}{4}$  LSB ( $\sim 5$  mV) when operating with a 60 Hz common-mode frequency,  $f_{\text{cm}}$ , and using a 640 kHz A/D clock,  $f_{\text{CLK}}$ , would allow a peak value of the common-mode voltage,  $V_p$ , which is given by:

$$V_p = \frac{[\Delta V_e(\text{MAX}) (f_{\text{CLK}})]}{(2\pi f_{\text{cm}}) (4.5)}$$

or

$$V_p = \frac{(5 \times 10^{-3}) (640 \times 10^3)}{(6.28) (60) (4.5)}$$

which gives

$$V_p \approx 1.9 \text{ V.}$$

The allowed range of analog input voltages usually places more severe restrictions on input common-mode noise levels.

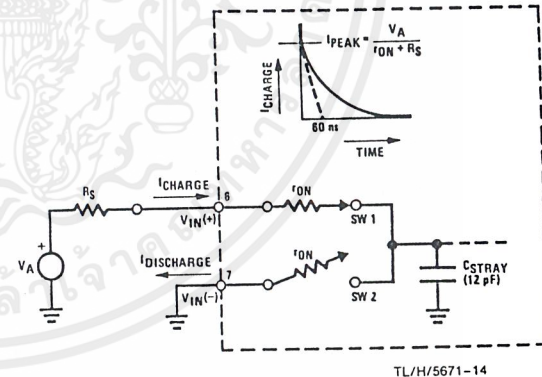
An analog input voltage with a reduced span and a relatively large zero offset can be handled easily by making use of the differential input (see section 2.4 Reference Voltage).

**2.3 Analog Inputs**

**2.3.1 Input Current**

**Normal Mode**

Due to the internal switching action, displacement currents will flow at the analog inputs. This is due to on-chip stray capacitance to ground as shown in Figure 3.



$r_{\text{ON}}$  of SW 1 and SW 2  $\approx 5 \text{ k}\Omega$   
 $t = r_{\text{ON}} C_{\text{STRAY}} \approx 5 \text{ k}\Omega \times 12 \text{ pF} = 60 \text{ ns}$

**FIGURE 3. Analog Input Impedance**

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
 ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## Functional Description (Continued)

The voltage on this capacitance is switched and will result in currents entering the  $V_{IN}(+)$  input pin and leaving the  $V_{IN}(-)$  input which will depend on the analog differential input voltage levels. These current transients occur at the leading edge of the internal clocks. They rapidly decay and *do not cause errors* as the on-chip comparator is strobed at the end of the clock period.

### Fault Mode

If the voltage source applied to the  $V_{IN}(+)$  or  $V_{IN}(-)$  pin exceeds the allowed operating range of  $V_{CC} + 50$  mV, large input currents can flow through a parasitic diode to the  $V_{CC}$  pin. If these currents can exceed the 1 mA max allowed spec, an external diode (1N914) should be added to bypass this current to the  $V_{CC}$  pin (with the current bypassed with this diode, the voltage at the  $V_{IN}(+)$  pin can exceed the  $V_{CC}$  voltage by the forward voltage of this diode).

### 2.3.2 Input Bypass Capacitors

Bypass capacitors at the inputs will average these charges and cause a DC current to flow through the output resistances of the analog signal sources. This charge pumping action is worse for continuous conversions with the  $V_{IN}(+)$  input voltage at full-scale. For continuous conversions with a 640 kHz clock frequency with the  $V_{IN}(+)$  input at 5V, this DC current is at a maximum of approximately 5  $\mu$ A. Therefore, *bypass capacitors should not be used at the analog inputs or the  $V_{REF}/2$  pin for high resistance sources ( $> 1$  k $\Omega$ )*. If input bypass capacitors are necessary for noise filtering and high source resistance is desirable to minimize capacitor size, the detrimental effects of the voltage drop across this input resistance, which is due to the average value of the input current, can be eliminated with a full-scale adjustment while the given source resistor and input bypass capacitor are both in place. This is possible because the average value of the input current is a precise linear function of the differential input voltage.

### 2.3.3 Input Source Resistance

Large values of source resistance where an input bypass capacitor is not used, *will not cause errors* as the input currents settle out prior to the comparison time. If a low pass filter is required in the system, use a low valued series resistor ( $\leq 1$  k $\Omega$ ) for a passive RC section or add an op amp RC active low pass filter. For low source resistance applications, ( $\leq 1$  k $\Omega$ ), a 0.1  $\mu$ F bypass capacitor at the inputs will prevent noise pickup due to series lead inductance of a long wire. A 100 $\Omega$  series resistor can be used to isolate this capacitor—both the R and C are placed outside the feedback loop—from the output of an op amp, if used.

### 2.3.4 Noise

The leads to the analog inputs (pin 6 and 7) should be kept as short as possible to minimize input noise coupling. Both noise and undesired digital clock coupling to these inputs can cause system errors. The source resistance for these inputs should, in general, be kept below 5 k $\Omega$ . Larger values of source resistance can cause undesired system noise pickup. Input bypass capacitors, placed from the analog inputs to ground, will eliminate system noise pickup but can create analog scale errors as these capacitors will average the transient input switching currents of the A/D (see section 2.3.1.). This scale error depends on both a large source

resistance and the use of an input bypass capacitor. This error can be eliminated by doing a full-scale adjustment of the A/D (adjust  $V_{REF}/2$  for a proper full-scale reading—see section 2.5.2 on Full-Scale Adjustment) with the source resistance and input bypass capacitor in place.

## 2.4 Reference Voltage

### 2.4.1 Span Adjust

For maximum applications flexibility, these A/Ds have been designed to accommodate a 5  $V_{DC}$ , 2.5  $V_{DC}$  or an adjusted voltage reference. This has been achieved in the design of the IC as shown in Figure 4.

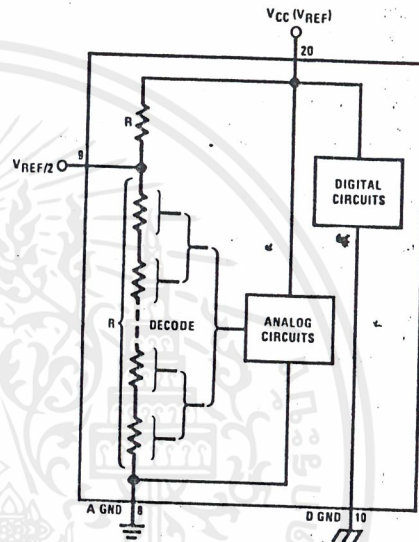


FIGURE 4. The  $V_{REFERENCE}$  Design on the IC

Notice that the reference voltage for the IC is either  $1/2$  of the voltage applied to the  $V_{CC}$  supply pin, or is equal to the voltage that is externally forced at the  $V_{REF}/2$  pin. This allows for a ratiometric voltage reference using the  $V_{CC}$  supply, a 5  $V_{DC}$  reference voltage can be used for the  $V_{CC}$  supply or a voltage less than 2.5  $V_{DC}$  can be applied to the  $V_{REF}/2$  input for increased application flexibility. The internal gain to the  $V_{REF}/2$  input is 2, making the full-scale differential input voltage twice the voltage at pin 9.

An example of the use of an adjusted reference voltage is to accommodate a reduced span—or dynamic voltage range of the analog input voltage. If the analog input voltage were to range from 0.5  $V_{DC}$  to 3.5  $V_{DC}$ , instead of 0V to 5  $V_{DC}$ , the span would be 3V as shown in Figure 5. With 0.5  $V_{DC}$  applied to the  $V_{IN}(-)$  pin to absorb the offset, the reference voltage can be made equal to  $1/2$  of the 3V span or 1.5  $V_{DC}$ . The A/D now will encode the  $V_{IN}(+)$  signal from 0.5V to 3.5 V with the 0.5V input corresponding to zero and the 3.5  $V_{DC}$  input corresponding to full-scale. The full 8 bits of resolution are therefore applied over this reduced analog input voltage range.

### 2.4.2 I

The c-  
absolt  
magni  
output  
conver  
code.  
metric  
lute cc  
temper  
factors  
voltage  
mV<sub>DC</sub>  
gain of  
the initi  
becom  
reduces  
respons  
1 LSB  
this rec  
voltage  
with ter  
2.5V pl  
and stal  
In gene  
quire an  
of refere  
transfer  
ferences  
sive. Th  
Semicon  
(6 mV  
range pe

Functional Description (Continued)

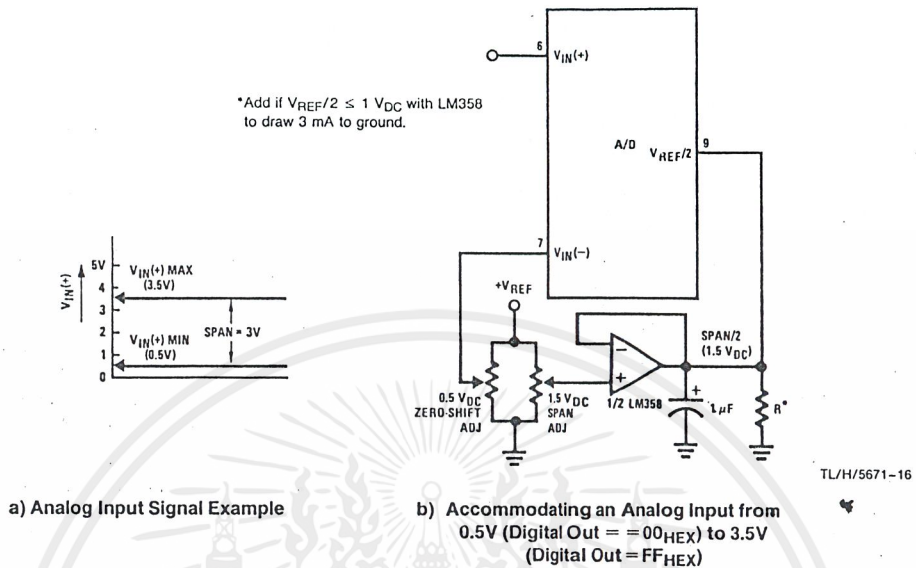


FIGURE 5. Adapting the A/D Analog Input Voltages to Match an Arbitrary Input Signal Range

2.4.2 Reference Accuracy Requirements

The converter can be operated in a ratiometric mode or an absolute mode. In ratiometric converter applications, the magnitude of the reference voltage is a factor in both the output of the source transducer and the output of the A/D converter and therefore cancels out in the final digital output code. The ADC0805 is specified particularly for use in ratiometric applications with no adjustments required. In absolute conversion applications, both the initial value and the temperature stability of the reference voltage are important factors in the accuracy of the A/D converter. For  $V_{REF/2}$  voltages of 2.4  $V_{DC}$  nominal value, initial errors of  $\pm 10$  mV<sub>DC</sub> will cause conversion errors of  $\pm 1$  LSB due to the gain of 2 of the  $V_{REF/2}$  input. In reduced span applications, the initial value and the stability of the  $V_{REF/2}$  input voltage become even more important. For example, if the span is reduced to 2.5V, the analog input LSB voltage value is correspondingly reduced from 20 mV (5V span) to 10 mV and 1 LSB at the  $V_{REF/2}$  input becomes 5 mV. As can be seen, this reduces the allowed initial tolerance of the reference voltage and requires correspondingly less absolute change with temperature variations. Note that spans smaller than 2.5V place even tighter requirements on the initial accuracy and stability of the reference source.

In general, the magnitude of the reference voltage will require an initial adjustment. Errors due to an improper value of reference voltage appear as full-scale errors in the A/D transfer function. IC voltage regulators may be used for references if the ambient temperature changes are not excessive. The LM336B 2.5V IC reference diode (from National Semiconductor) has a temperature stability of 1.3 mV typ (6 mV max) over  $0^{\circ}C \leq T_A \leq +70^{\circ}C$ . Other temperature range parts are also available.

2.5 Errors and Reference Voltage Adjustments

2.5.1 Zero Error

The zero of the A/D does not require adjustment. If the minimum analog input voltage value,  $V_{IN(MIN)}$ , is not ground, a zero offset can be done. The converter can be made to output 0000 0000 digital code for this minimum input voltage by biasing the A/D  $V_{IN(-)}$  input at this  $V_{IN(MIN)}$  value (see Applications section). This utilizes the differential mode operation of the A/D.

The zero error of the A/D converter relates to the location of the first riser of the transfer function and can be measured by grounding the  $V_{IN(-)}$  input and applying a small magnitude positive voltage to the  $V_{IN(+)}$  input. Zero error is the difference between the actual DC input voltage that is necessary to just cause an output digital code transition from 0000 0000 to 0000 0001 and the ideal  $1/2$  LSB value ( $1/2$  LSB = 9.8 mV for  $V_{REF/2} = 2.500 V_{DC}$ ).

2.5.2 Full-Scale

The full-scale adjustment can be made by applying a differential input voltage that is  $1 1/2$  LSB less than the desired analog full-scale voltage range and then adjusting the magnitude of the  $V_{REF/2}$  input (pin 9 or the  $V_{CC}$  supply if pin 9 is not used) for a digital output code that is just changing from 1111 1110 to 1111 1111.



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## Functional Description (Continued)

### 2.5.3 Adjusting for an Arbitrary Analog Input Voltage Range

If the analog zero voltage of the A/D is shifted away from ground (for example, to accommodate an analog input signal that does not go to ground) this new zero reference should be properly adjusted first. A  $V_{IN}(+)$  voltage that equals this desired zero reference plus  $\frac{1}{2}$  LSB (where the LSB is calculated for the desired analog span,  $1 \text{ LSB} = \text{analog span}/256$ ) is applied to pin 6 and the zero reference voltage at pin 7 should then be adjusted to just obtain the  $00_{\text{HEX}}$  to  $01_{\text{HEX}}$  code transition.

The full-scale adjustment should then be made (with the proper  $V_{IN}(-)$  voltage applied) by forcing a voltage to the  $V_{IN}(+)$  input which is given by:

$$V_{IN}(+) \text{ fs adj} = V_{\text{MAX}} - 1.5 \left[ \frac{(V_{\text{MAX}} - V_{\text{MIN}})}{256} \right]$$

where:

$V_{\text{MAX}}$  = The high end of the analog input range and

$V_{\text{MIN}}$  = the low end (the offset zero) of the analog range. (Both are ground referenced.)

The  $V_{\text{REF}}/2$  (or  $V_{\text{CC}}$ ) voltage is then adjusted to provide a code change from  $FE_{\text{HEX}}$  to  $FF_{\text{HEX}}$ . This completes the adjustment procedure.

### 2.6 Clocking Option

The clock for the A/D can be derived from the CPU clock or an external RC can be added to provide self-clocking. The CLK IN (pin 4) makes use of a Schmitt trigger as shown in Figure 6.

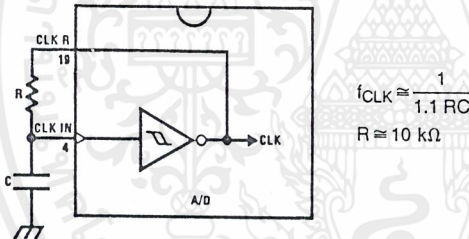


FIGURE 6. Self-Clocking the A/D

TL/H/5671-17

Heavy capacitive or DC loading of the clock R pin should be avoided as this will disturb normal converter operation. Loads less than 50 pF, such as driving up to 7 A/D converter clock inputs from a single clock R pin of 1 converter, are allowed. For larger clock line loading, a CMOS or low power TTL buffer or PNP input logic should be used to minimize the loading on the clock R pin (do not use a standard TTL buffer).

### 2.7 Restart During a Conversion

If the A/D is restarted ( $\overline{\text{CS}}$  and  $\overline{\text{WR}}$  go low and return high) during a conversion, the converter is reset and a new conversion is started. The output data latch is not updated if the

conversion in process is not allowed to be completed, therefore the data of the previous conversion remains in this latch. The  $\overline{\text{INTR}}$  output simply remains at the "1" level.

### 2.8 Continuous Conversions

For operation in the free-running mode an initializing pulse should be used, following power-up, to ensure circuit operation. In this application, the  $\overline{\text{CS}}$  input is grounded and the  $\overline{\text{WR}}$  input is tied to the  $\overline{\text{INTR}}$  output. This  $\overline{\text{WR}}$  and  $\overline{\text{INTR}}$  node should be momentarily forced to logic low following a power-up cycle to guarantee operation.

### 2.9 Driving the Data Bus

This MOS A/D, like MOS microprocessors and memories, will require a bus driver when the total capacitance of the data bus gets large. Other circuitry, which is tied to the data bus, will add to the total capacitive loading, even in TRI-STATE (high impedance mode). Backplane bussing also greatly adds to the stray capacitance of the data bus.

There are some alternatives available to the designer to handle this problem. Basically, the capacitive loading of the data bus slows down the response time, even though DC specifications are still met. For systems operating with a relatively slow CPU clock frequency, more time is available in which to establish proper logic levels on the bus and therefore higher capacitive loads can be driven (see typical characteristics curves).

At higher CPU clock frequencies time can be extended for I/O reads (and/or writes) by inserting wait states (8080) or using clock extending circuits (6800).

Finally, if time is short and capacitive loading is high, external bus drivers must be used. These can be TRI-STATE buffers (low power Schottky such as the DM74LS240 series is recommended) or special higher drive current products which are designed as bus drivers. High current bipolar bus drivers with PNP inputs are recommended.

### 2.10 Power Supplies

Noise spikes on the  $V_{\text{CC}}$  supply line can cause conversion errors as the comparator will respond to this noise. A low inductance tantalum filter capacitor should be used close to the converter  $V_{\text{CC}}$  pin and values of 1  $\mu\text{F}$  or greater are recommended. If an unregulated voltage is available in the system, a separate LM340LAZ-5.0, TO-92, 5V voltage regulator for the converter (and other analog circuitry) will greatly reduce digital noise on the  $V_{\text{CC}}$  supply.

### 2.11 Wiring and Hook-Up Precautions

Standard digital wire wrap sockets are not satisfactory for breadboarding this A/D converter. Sockets on PC boards can be used and all logic signal wires and leads should be grouped and kept as far away as possible from the analog signal leads. Exposed leads to the analog inputs can cause undesired digital noise and hum pickup, therefore shielded leads may be necessary in many applications.

### Functional Description (Continued)

A single point analog ground that is separate from the logic ground points should be used. The power supply bypass capacitor and the self-clocking capacitor (if used) should both be returned to digital ground. Any  $V_{REF}/2$  bypass capacitors, analog input filter capacitors, or input signal shielding should be returned to the analog ground point. A test for proper grounding is to measure the zero error of the A/D converter. Zero errors in excess of  $1/4$  LSB can usually be traced to improper board layout and wiring (see section 2.5.1 for measuring the zero error).

### 3.0 TESTING THE A/D CONVERTER

There are many degrees of complexity associated with testing an A/D converter. One of the simplest tests is to apply a known analog input voltage to the converter and use LEDs to display the resulting digital output code as shown in Figure 7.

For ease of testing, the  $V_{REF}/2$  (pin 9) should be supplied with 2.560 V<sub>DC</sub> and a  $V_{CC}$  supply voltage of 5.12 V<sub>DC</sub> should be used. This provides an LSB value of 20 mV.

If a full-scale adjustment is to be made, an analog input voltage of 5.090 V<sub>DC</sub> ( $5.120 - 1/2$  LSB) should be applied to the  $V_{IN}(+)$  pin with the  $V_{IN}(-)$  pin grounded. The value of the  $V_{REF}/2$  input voltage should then be adjusted until the digital output code is just changing from 1111 1110 to 1111 1111. This value of  $V_{REF}/2$  should then be used for all the tests.

The digital output LED display can be decoded by dividing the 8 bits into 2 hex characters, the 4 most significant (MS) and the 4 least significant (LS). Table 1 shows the fractional binary equivalent of these two 4-bit groups. By adding the voltages obtained from the "VMS" and "VLS" columns in Table 1, the nominal value of the digital display (when

$V_{REF}/2 = 2.560V$ ) can be determined. For example, for an output LED display of 1011 0110 or B6 (in hex), the voltage values from the table are  $3.520 + 0.120$  or 3.640 V<sub>DC</sub>. These voltage values represent the center-values of a perfect A/D converter. The effects of quantization error have to be accounted for in the interpretation of the test results.

For a higher speed test system, or to obtain plotted data, a digital-to-analog converter is needed for the test set-up. An accurate 10-bit DAC can serve as the precision voltage source for the A/D. Errors of the A/D under test can be expressed as either analog voltages or differences in 2 digital words.

A basic A/D tester that uses a DAC and provides the error as an analog output voltage is shown in Figure 8. The 2 op amps can be eliminated if a lab DVM with a numerical subtraction feature is available to read the difference voltage, "A-C", directly. The analog input voltage can be supplied by a low frequency ramp generator and an X-Y plotter can be used to provide analog error (Y axis) versus analog input (X axis).

For operation with a microprocessor or a computer-based test system, it is more convenient to present the errors digitally. This can be done with the circuit of Figure 9, where the output code transitions can be detected as the 10-bit DAC is incremented. This provides  $1/4$  LSB steps for the 8-bit A/D under test. If the results of this test are automatically plotted with the analog input on the X axis and the error (in LSB's) as the Y axis, a useful transfer function of the A/D under test results. For acceptance testing, the plot is not necessary and the testing speed can be increased by establishing internal limits on the allowed error for each code.

### 4.0 MICROPROCESSOR INTERFACING

To discuss the interface with 8080A and 6800 microprocessors, a common sample subroutine structure is used. The microprocessor starts the A/D, reads and stores the results of 16 successive conversions, then returns to the user's program. The 16 data bytes are stored in 16 successive memory locations. All Data and Addresses will be given in hexadecimal form. Software and hardware details are provided separately for each type of microprocessor.

#### 4.1 Interfacing 8080 Microprocessor Derivatives (8048, 8085)

This converter has been designed to directly interface with derivatives of the 8080 microprocessor. The A/D can be mapped into memory space (using standard memory address decoding for  $\overline{CS}$  and the MEMR and MEMW strobes) or it can be controlled as an I/O device by using the  $\overline{I/O}$  R and  $\overline{I/O}$  W strobes and decoding the address bits A0  $\rightarrow$  A7 (or address bits A8  $\rightarrow$  A15 as they will contain the same 8-bit address information) to obtain the  $\overline{CS}$  input. Using the I/O space provides 256 additional addresses and may allow a simpler 8-bit address decoder but the data can only be input to the accumulator. To make use of the additional memory reference instructions, the A/D should be mapped into memory space. An example of an A/D in I/O space is shown in Figure 10.

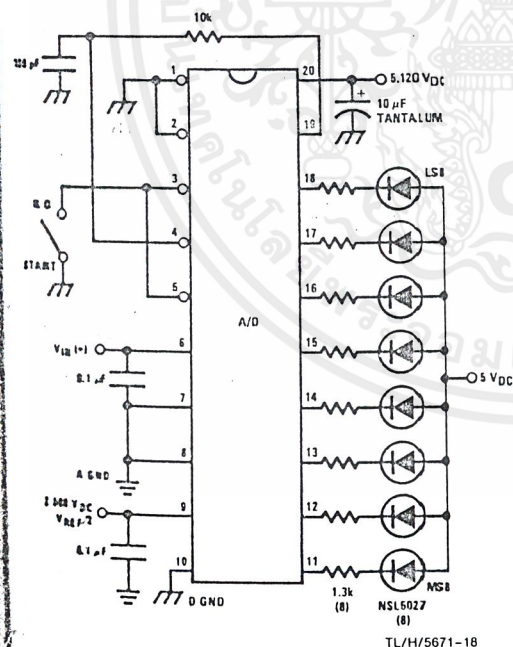


FIGURE 7. Basic A/D Tester

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Functional Description (Continued)

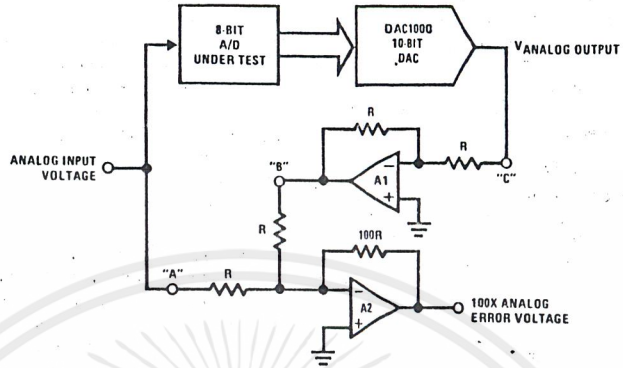


FIGURE 8. A/D Tester with Analog Error Output



FIGURE 9. Basic "Digital" A/D Tester

TL/H/5671-19

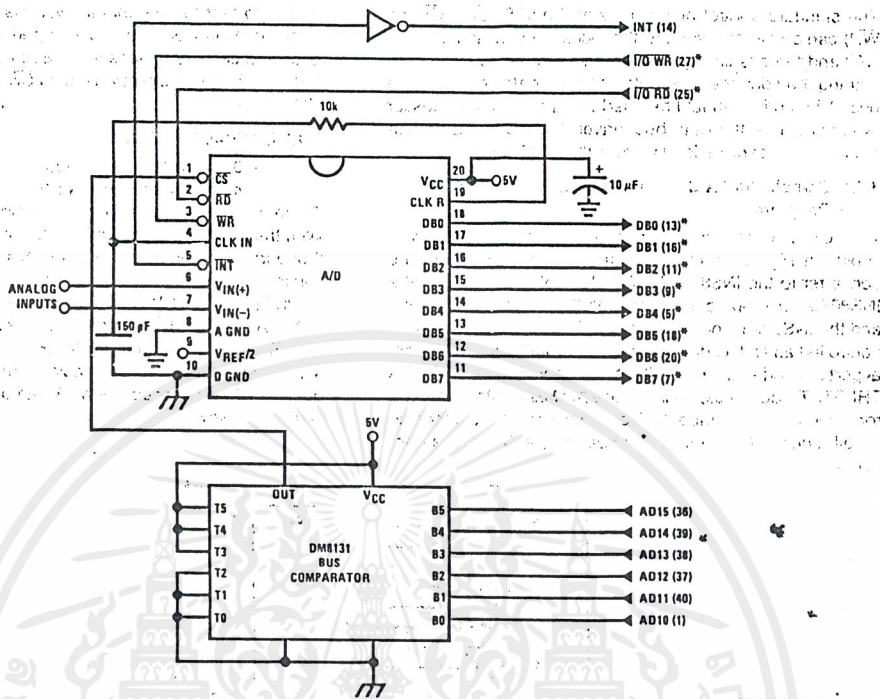
TABLE I. DECODING THE DIGITAL OUTPUT LEDS

HEX	BINARY	FRACTIONAL BINARY VALUE FOR		OUTPUT VOLTAGE CENTER VALUES WITH $V_{REF}/2 = 2.560 V_{DC}$	
		MS GROUP	LS GROUP	VMS GROUP*	VLS GROUP*
		F	1 1 1 1	15/16	15/256
E	1 1 1 0	7/8	7/128	4.480	0.280
D	1 1 0 1	13/16	13/256	4.160	0.260
C	1 1 0 0	3/4	3/64	3.840	0.240
B	1 0 1 1	11/16	11/256	3.520	0.220
A	1 0 1 0	5/8	5/128	3.200	0.200
9	1 0 0 1	9/16	9/256	2/880	0.180
8	1 0 0 0	1/2	1/32	2/560	0.160
7	0 1 1 1	7/16	7/256	2.240	0.140
6	0 1 1 0	3/8	3/128	1.920	0.120
5	0 1 0 1	5/16	2/256	1.600	0.100
4	0 1 0 0	1/4	1/64	1/280	0.080
3	0 0 1 1	3/16	3/256	0.960	0.060
2	0 0 1 0	1/8	1/128	0.640	0.040
1	0 0 0 1	1/16	1/256	0.320	0.020
0	0 0 0 0			0	0

\*Display Output = VMS Group + VLS Group

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Functional Description (Continued)



Note 1: \*Pin numbers for the DP8228 system controller, others are INS8080A.  
 Note 2: Pin 23 of the INS8228 must be tied to +12V through a 1 kΩ resistor to generate the RST 7 instruction when an interrupt is acknowledged as required by the accompanying sample program.

FIGURE 10. ADC0801-INS8080A CPU Interface

SAMPLE PROGRAM FOR FIGURE 10 ADC0801-INS8080A CPU INTERFACE

```

0038 C3 00 03 RST 7: JMP LD DATA
. . .
0100 21 00 02 START: LXI H 0200H ; HL pair will point to
. . . ; data storage locations
0103 31 00 04 RETURN: LXI SP 0400H ; Initialize stack pointer (Note 1)
0106 7D CPI OF H ; Test # of bytes entered
0107 FE 0F JZ CONT ; If # = 16. JMP to
0109 CA 13 01 JZ CONT ; user program
010C D3 E0 OUT E0 H ; Start A/D
010E FB EI ; Enable interrupt
010F 00 NOP ; Loop until end of
0110 C3 0F 01 JMP LOOP ; conversion
0113 . CONT: .
. . .
. . . (User program to
. . . process data)
. . .
0300 DB E0 LD DATA: IN E0 H ; Load data into accumulator
0302 77 MOV M, A ; Store data
0303 23 INX H ; Increment storage pointer
0304 C3 03 01 JMP RETURN
    
```

Note 1: The stack pointer must be dimensioned because a RST 7 instruction pushes the PC onto the stack.  
 Note 2: All address used were arbitrarily chosen.

## Functional Description (Continued)

The standard control bus signals of the 8080  $\overline{CS}$ ,  $\overline{RD}$  and  $\overline{WR}$  can be directly wired to the digital control inputs of the A/D and the bus timing requirements are met to allow both starting the converter and outputting the data onto the data bus. A bus driver should be used for larger microprocessor systems where the data bus leaves the PC board and/or must drive capacitive loads larger than 100 pF.

### 4.1.1 Sample 8080A CPU Interfacing Circuitry and Program

The following sample program and associated hardware shown in Figure 10 may be used to input data from the converter to the INS8080A CPU chip set (comprised of the INS8080A microprocessor, the INS8228 system controller and the INS8224 clock generator). For simplicity, the A/D is controlled as an I/O device, specifically an 8-bit bi-directional port located at an arbitrarily chosen port address, E0. The TRI-STATE output capability of the A/D eliminates the need for a peripheral interface device, however address decoding is still required to generate the appropriate  $\overline{CS}$  for the converter.

It is important to note that in systems where the A/D converter is 1-of-8 or less I/O mapped devices, no address decoding circuitry is necessary. Each of the 8 address bits (A0 to A7) can be directly used as  $\overline{CS}$  inputs—one for each I/O device.

### 4.1.2 INS8048 Interface

The INS8048 interface technique with the ADC0801 series (see Figure 11) is simpler than the 8080A CPU interface. There are 24 I/O lines and three test input lines in the 8048. With these extra I/O lines available, one of the I/O lines (bit 0 of port 1) is used as the chip select signal to the A/D, thus eliminating the use of an external address decoder. Bus control signals  $\overline{RD}$ ,  $\overline{WR}$  and  $\overline{INT}$  of the 8048 are tied directly to the A/D. The 16 converted data words are stored at on-chip RAM locations from 20 to 2F (Hex). The  $\overline{RD}$  and  $\overline{WR}$  signals are generated by reading from and writing into a dummy address, respectively. A sample interface program is shown below.

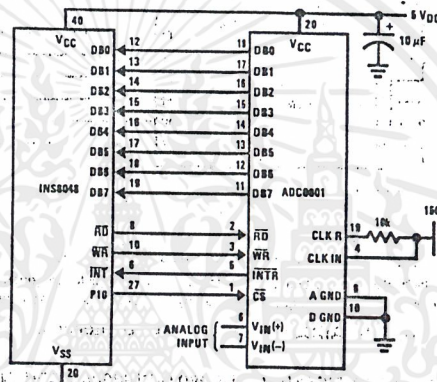


FIGURE 11. INS8048 Interface

### SAMPLE PROGRAM FOR FIGURE 11 INS8048 INTERFACE

```

04 10          JMP          10H          ; Program starts at addr 10
          ORG          3H
04 50          JMP          50H          ; Interrupt jump vector
          ORG          10H          ; Main program
99 FE          ANL          P1, #0FEH    ; Chip select
81             MOVX         A, @R1      ; Read in the 1st data
          ; to reset the intr
89 01          START:      ORL          P1, #1      ; Set port pin high
B8 20          MOV          R0, #20H     ; Data address
B9 FF          MOV          R1, #0FFH    ; Dummy address
BA 10          MOV          R2, #10H     ; Counter for 16 bytes
23 FF          AGAIN:     MOV          A, #0FFH  ; Set ACC for intr loop
99 FE          ANL          P1, #0FEH    ; Send CS (bit 0 of P1)
91             MOVX         @R1, A      ; Send WR out
05             EN          I            ; Enable interrupt
96 21          LOOP:      JNZ          LOOP    ; Wait for interrupt
EA 1B          DJNZ        R2, AGAIN     ; If 16 bytes are read
00             NOP
00             NOP
81             INDATA:    ORG          50H
AC             MOVX         A, @R1      ; Input data, CS still low
18             MOV          @R0, A      ; Store in memory
89 01          INC          R0          ; Increment storage counter
27             ORL          P1, #1      ; Reset CS signal
93             CLR          A          ; Clear ACC to get out of
          RETR          ; the interrupt loop
    
```

## Functional Description (Continued)

### 4.2 Interfacing the Z-80

The Z-80 control bus is slightly different from that of the 8080. General  $\overline{RD}$  and  $\overline{WR}$  strobes are provided and separate memory request,  $\overline{MREQ}$ , and I/O request,  $\overline{IORQ}$ , signals are used which have to be combined with the generalized strobes to provide the equivalent 8080 signals. An advantage of operating the A/D in I/O space with the Z-80 is that the CPU will automatically insert one wait state (the  $\overline{RD}$  and  $\overline{WR}$  strobes are extended one clock period) to allow more time for the I/O devices to respond. Logic to map the A/D in I/O space is shown in Figure 13.

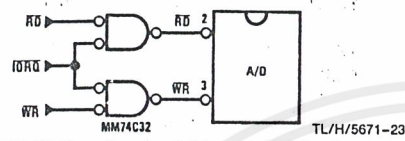


FIGURE 13. Mapping the A/D as an I/O Device for Use with the Z-80 CPU

Additional I/O advantages exist as software DMA routines are available and use can be made of the output data transfer which exists on the upper 8 address lines (A8 to A15) during I/O input instructions. For example, MUX channel selection for the A/D can be accomplished with this operating mode.

### 4.3 Interfacing 6800 Microprocessor Derivatives (6502, etc.)

The control bus for the 6800 microprocessor derivatives does not use the  $\overline{RD}$  and  $\overline{WR}$  strobe signals. Instead it employs a single R/W line and additional timing, if needed, can be derived from the  $\phi 2$  clock. All I/O devices are memory mapped in the 6800 system, and a special signal, VMA, indicates that the current address is valid. Figure 14 shows an interface schematic where the A/D is memory mapped in the 6800 system. For simplicity, the  $\overline{CS}$  decoding is shown using 1/2 DM8092. Note that in many 6800 systems, an al-

ready decoded  $\overline{A75}$  line is brought out to the common bus at pin 21. This can be tied directly to the  $\overline{CS}$  pin of the A/D, provided that no other devices are addressed at HX ADDR: 4XXX or 5XXX.

The following subroutine performs essentially the same function as in the case of the 8080A interface and it can be called from anywhere in the user's program.

In Figure 15 the ADC0801 series is interfaced to the M6800 microprocessor through (the arbitrarily chosen) Port B of the MC6820 or MC6821 Peripheral Interface Adapter (PIA). Here the  $\overline{CS}$  pin of the A/D is grounded since the PIA is already memory mapped in the M6800 system and no  $\overline{CS}$  decoding is necessary. Also notice that the A/D output data lines are connected to the microprocessor bus under program control through the PIA and therefore the A/D  $\overline{RD}$  pin can be grounded.

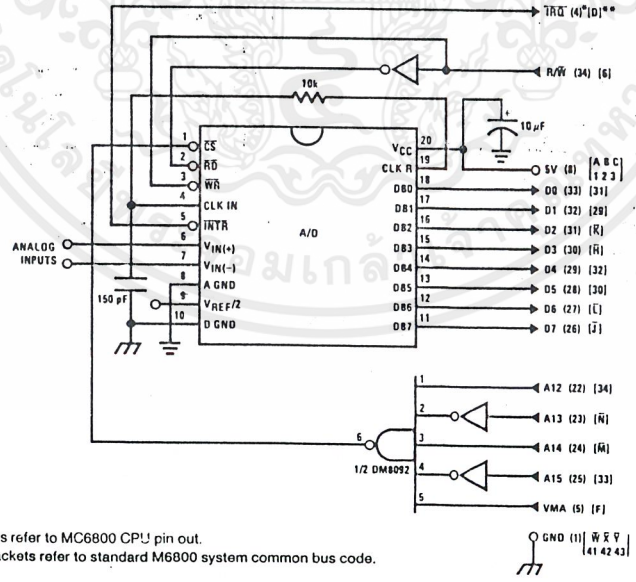
A sample interface program equivalent to the previous one is shown below Figure 15. The PIA Data and Control Registers of Port B are located at HEX addresses 8006 and 8007, respectively.

## 5.0 GENERAL APPLICATIONS

The following applications show some interesting uses for the A/D. The fact that one particular microprocessor is used is not meant to be restrictive. Each of these application circuits would have its counterpart using any microprocessor that is desired.

### 5.1 Multiple ADC0801 Series to MC6800 CPU Interface

To transfer analog data from several channels to a single microprocessor system, a multiple converter scheme presents several advantages over the conventional multiplexer single-converter approach. With the ADC0801 series, the differential inputs allow individual span adjustment for each channel. Furthermore, all analog input channels are sensed simultaneously, which essentially divides the microprocessor's total system servicing time by the number of channels, since all conversions occur simultaneously. This scheme is shown in Figure 16.



Note 1: Numbers in parentheses refer to MC6800 CPU pin out.  
 Note 2: Number or letters in brackets refer to standard M6800 system common bus code.

FIGURE 14. ADC0801-MC6800 CPU Interface

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
 ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Functional Description (Continued)

SAMPLE PROGRAM FOR FIGURE 14 ADC0801-MC6800 CPU INTERFACE

```

0010 DF 36      DATAIN STX      TEMP2      ; Save contents of X
0012 CE 00 2C      LDX      #$002C      ; Upon IRQ low CPU
0015 FF FF F8      STX      $FFFF      ; jumps to 002C
0018 B7 50 00      STAA     $5000      ; Start ADC0801
001B 0E          CLI          ;
001C 3E          CONVRT WAI          ; Wait for interrupt
001D DE 34      LDX      TEMP1      ;
001F 8C 02 0F      CPX      #$020F      ; Is final data stored?
0022 27 14      BEQ      ENDP      ;
0024 B7 50 00      STAA     $5000      ; Restarts ADC0801
0027 08          INX          ;
0028 DF 34      STX      TEMP1      ;
002A 20 F0      BRA      CONVRT     ;
002C DE 34      INTRPT  LDX      TEMP1      ;
002E B6 50 00      LDAA     $5000      ; Read data
0031 A7 00      STAA     X          ; Store it at X
0033 3B          RTI          ;
0034 02 00      TEMP1  FDB      $0200      ; Starting address for
; data storage
0036 00 00      TEMP2  FDB      $0000      ;
0038 CE 02 00      ENDP      LDX      #$0200      ; Reinitialize TEMP1
003B DF 34      STX      TEMP1      ;
003D DE 36      LDX      TEMP2      ;
003F 39          RTS          ; Return from subroutine
; To user's program
    
```

Note 1: In order for the microprocessor to service subroutines and interrupts, the stack pointer must be dimensioned in the user's program.

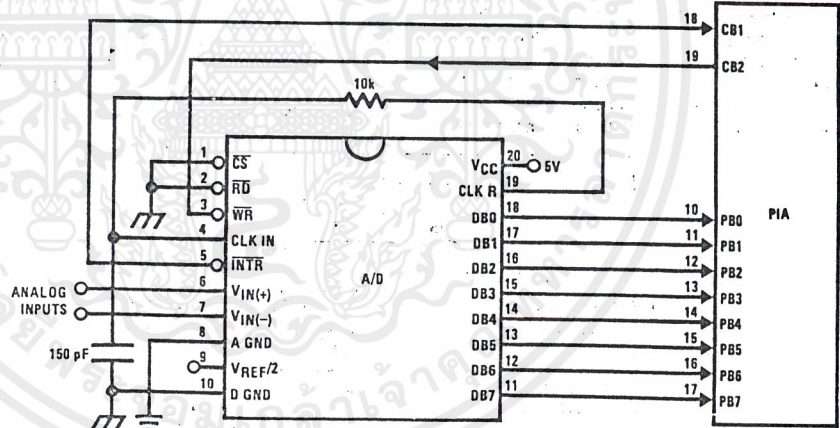


FIGURE 15. ADC0801-MC6820 PIA Interface

TL/H/56

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Functional Description (Continued)

SAMPLE PROGRAM FOR FIGURE 15 ADC0801-MC6820 PIA INTERFACE

```

0010 CE 00 38      DATAIN   LDX      #$0038      ; Upon IRQ low CPU
0013 FF FF F8      STX      $FFF8      ; jumps to 0038
0016 B6 80 06      LDAA     PIAORB    ; Clear possible IRQ flags
0019 4F             CLRA
001A B7 80 07      STAA     PIACRB
001D B7 80 06      STAA     PIAORB    ; Set Port B as input
0020 0E             CLI
0021 C6 34          LDAB     #$34
0023 86 3D          LDAA     #$3D
0025 F7 80 07      CONVRT   STAB     PIACRB    ; Starts ADC0801
0028 B7 80 07      STAA     PIACRB
002B 3E             WAI
002C DE 40          LDX      TEMP1
002E 8C 02 0F      CPX      #$020F    ; Is final data stored?
0031 27 0F          BEQ     ENDP
0033 08             INX
0034 DF 40          STX      TEMP1
0036 20 ED          BRA     CONVRT
0038 DE 40          INTRPT  LDX      TEMP1
003A B6 80 06      LDAA     PIAORB    ; Read data in
003D A7 00          STAA     X          ; Store it at X
003F 3B             RTI
0040 02 00          TEMP1  FDB     $0200 ; Starting address for
                                ; data storage
0042 CE 02 00      ENDP    LDX     #$0200 ; Reinitialize TEMP1
0045 DF 40          STX     TEMP1
0047 39             RTS
                                ; Return from subroutine
                                PIAORB  EQU     $8006 ; To user's program
                                PIACRB  EQU     $8007
    
```

The following schematic and sample subroutine (DATA IN) may be used to interface (up to) 8 ADC0801's directly to the MC6800 CPU. This scheme can easily be extended to allow the interface of more converters. In this configuration the converters are (arbitrarily) located at HEX address 5000 in the MC6800 memory space. To save components, the clock signal is derived from just one RC pair on the first converter. This output drives the other A/Ds.

All the converters are started simultaneously with a STORE instruction at HEX address 5000. Note that any other HEX address of the form 5XXX will be decoded by the circuit, pulling all the CS inputs low. This can easily be avoided by using a more definitive address decoding scheme. All the interrupts are ORed together to insure that all A/Ds have completed their conversion before the microprocessor is interrupted.

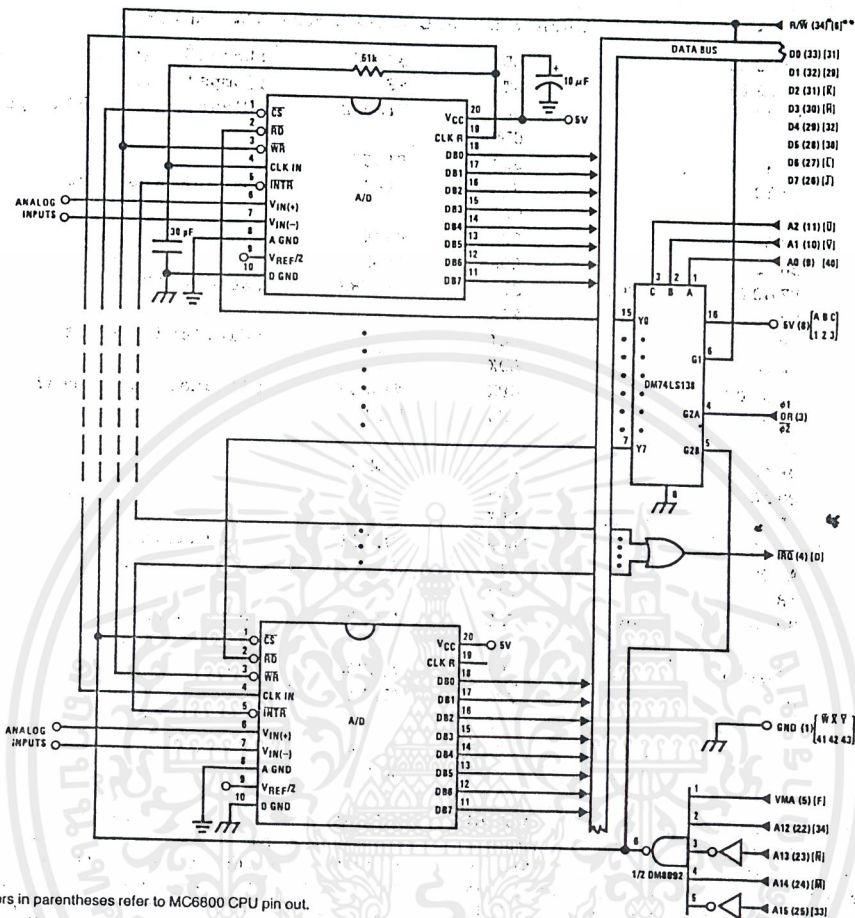
The subroutine, DATA IN, may be called from anywhere in the user's program. Once called, this routine initializes the

CPU, starts all the converters simultaneously and waits for the interrupt signal. Upon receiving the interrupt, it reads the converters (from HEX addresses 5000 through 5007) and stores the data successively at (arbitrarily chosen) HEX addresses 0200 to 0207, before returning to the user's program. All CPU registers then recover the original data they had before servicing DATA IN.

5.2 Auto-Zeroed Differential Transducer Amplifier and A/D Converter

The differential inputs of the ADC0801 series eliminate the need to perform a differential to single ended conversion for a differential transducer. Thus, one op amp can be eliminated since the differential to single ended conversion is provided by the differential input of the ADC0801 series. In general, a transducer preamp is required to take advantage of the full A/D converter input dynamic range.

Functional Description (Continued)



Note 1: Numbers in parentheses refer to MC6800 CPU pin out.

Note 2: Numbers of letters in brackets refer to standard M6800 system common bus code.

TL/H/5671-26

FIGURE 16. Interfacing Multiple A/Ds in an MC6800 System  
SAMPLE PROGRAM FOR FIGURE 16 INTERFACING MULTIPLE A/Ds IN AN MC6800 SYSTEM

ADDRESS	HEX CODE	MNEMONICS	COMMENTS
0010	DF 44	DATAIN STX TEMP	; Save Contents of X
0012	CE 00 2A	LDX #002A	; Upon $\overline{IRQ}$ LOW CPU
0015	FF FF F8	STX \$FFF8	; Jumps to 002A
0018	B7 50 00	STAA \$5000	; Starts all A/D's
001B	0E	CLI	
001C	3E	WAI	; Wait for interrupt
001D	CE 50 00	LDX \$5000	
0020	DF 40	STX INDEXT1	; Reset both INDEXT
0022	CE 02 00	LDX #0200	; 1 and 2 to starting
0025	DF 42	STX INDEXT2	; addresses
0027	DE 44	LDX TEMP	
0029	39	RTS	
002A	DE 40	INTRPT LDX INDEXT1	; Return from subroutine
002C	A6 00	LDAA X	; INDEXT1 $\rightarrow$ X
002E	08	INX	; Read data in from A/D at X
002F	DF 40	STX INDEXT1	; Increment X by one
0031	DE 42	LDX INDEXT2	; X $\rightarrow$ INDEXT1
			; INDEXT2 $\rightarrow$ X

Fu

Note  
For the pre  
whc  
subs  
for  
obv  
the

wh  
erro  
Vos  
sch  
The  
gran  
ing  
The  
with  
outp  
or c

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

**Functional Description** (Continued)

**SAMPLE PROGRAM FOR FIGURE 16 INTERFACING MULTIPLE A/Ds IN AN MC6800 SYSTEM**

ADDRESS	HEX CODE	MNEMONICS	COMMENTS
0033	A7 00	STAA X	; Store data at X
0035	8C 02 07	CPX #0207	; Have all A/D's been read?
0038	27 05	BEQ RETURN	; Yes: branch to RETURN
003A	08	INX	; No: increment X by one
003B	DF 42	STX INDEX2	; X → INDEX2
003D	20 EB	BRA INTRPT	; Branch to 002A
003F	3B	RETURN RTI	
0040	50 00	INDEX1 FDB \$5000	; Starting address for A/D
0042	02 00	INDEX2 FDB \$0200	; Starting address for data storage
0044	00 00	TEMP FDB \$0000	

**Note 1:** In order for the microprocessor to service subroutines and interrupts, the stack pointer must be dimensioned in the user's program.

For amplification of DC input signals, a major system error is the input offset voltage of the amplifiers used for the preamp. Figure 17 is a gain of 100 differential preamp whose offset voltage errors will be cancelled by a zeroing subroutine which is performed by the INS8080A microprocessor system. The total allowable input offset voltage error for this preamp is only 50 μV for 1/4 LSB error. This would obviously require very precise amplifiers. The expression for the differential output voltage of the preamp is:

$$V_O = \underbrace{[V_{IN(+)} - V_{IN(-)}]}_{\text{SIGNAL}} \underbrace{\left[1 + \frac{2R_2}{R_1}\right]}_{\text{GAIN}} + \underbrace{(V_{OS2} - V_{OS1} - V_{OS3} \pm I_X R_X)}_{\text{DC ERROR TERM}} \underbrace{\left(1 + \frac{2R_2}{R_1}\right)}_{\text{GAIN}}$$

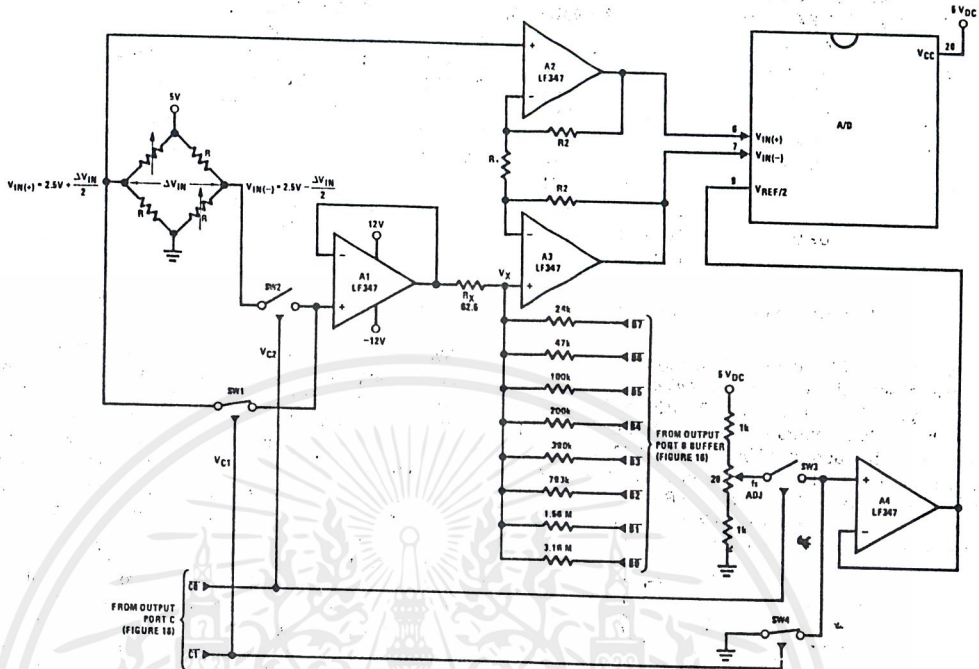
where  $I_X$  is the current through resistor  $R_X$ . All of the offset error terms can be cancelled by making  $\pm I_X R_X = V_{OS1} + V_{OS3} - V_{OS2}$ . This is the principle of this auto-zeroing scheme.

The INS8080A uses the 3 I/O ports of an INS8255 Programmable Peripheral Interface (PPI) to control the auto zeroing and input data from the ADC0801 as shown in Figure 18. The PPI is programmed for basic I/O operation (mode 0) with Port A being an input port and Ports B and C being output ports. Two bits of Port C are used to alternately open or close the 2 switches at the input of the preamp. Switch

SW1 is closed to force the preamp's differential input to be zero during the zeroing subroutine and then opened and SW2 is then closed for conversion of the actual differential input signal. Using 2 switches in this manner eliminates concern for the ON resistance of the switches as they must conduct only the input bias current of the input amplifiers.

Output Port B is used as a successive approximation register by the 8080 and the binary scaled resistors in series with each output bit create a D/A converter. During the zeroing subroutine, the voltage at  $V_X$  increases or decreases as required to make the differential output voltage equal to zero. This is accomplished by ensuring that the voltage at the output of A1 is approximately 2.5V so that a logic "1" (5V) on any output of Port B will source current into node  $V_X$  thus raising the voltage at  $V_X$  and making the output differential more negative. Conversely, a logic "0" (0V) will pull current out of node  $V_X$  and decrease the voltage, causing the differential output to become more positive. For the resistor values shown,  $V_X$  can move  $\pm 12$  mV with a resolution of 50 μV, which will null the offset error term to 1/4 LSB of full-scale for the ADC0801. It is important that the voltage levels that drive the auto-zero resistors be constant. Also, for symmetry, a logic swing of 0V to 5V is convenient. To achieve this, a CMOS buffer is used for the logic output signals of Port B and this CMOS package is powered with a stable 5V source. Buffer amplifier A1 is necessary so that it can source or sink the D/A output current.

### Functional Description (Continued)



- Note 1:  $R_2 = 49.5 R_1$
- Note 2: Switches are LMC13334 CMOS analog switches.
- Note 3: The 9 resistors used in the auto-zero section can be  $\pm 5\%$  tolerance.

FIGURE 17. Gain of 100 Differential Transducer Preamp

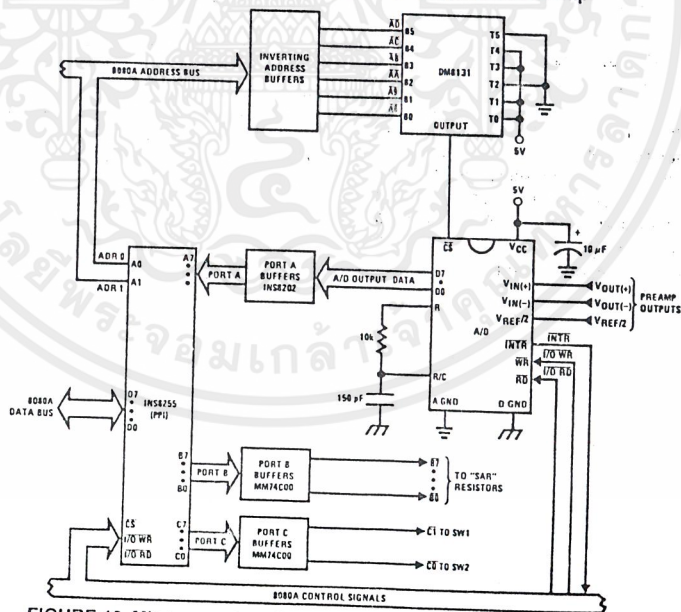


FIGURE 18. Microprocessor Interface Circuitry for Differential Preamp

TL/H/5671-27

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

A flow chart for the zeroing subroutine is shown in Figure 19. It must be noted that the ADC0801 series will output an all zero code when it converts a negative input [ $V_{IN(-)} \geq V_{IN(+)}$ ]. Also, a logic inversion exists as all of the I/O ports are buffered with inverting gates.

Basically, if the data read is zero, the differential output voltage is negative, so a bit in Port B is cleared to pull  $V_X$  more negative which will make the output more positive for the next conversion. If the data read is not zero, the output voltage is positive so a bit in Port B is set to make  $V_X$  more positive and the output more negative. This continues for 8 approximations and the differential output eventually converges to within 5 mV of zero.

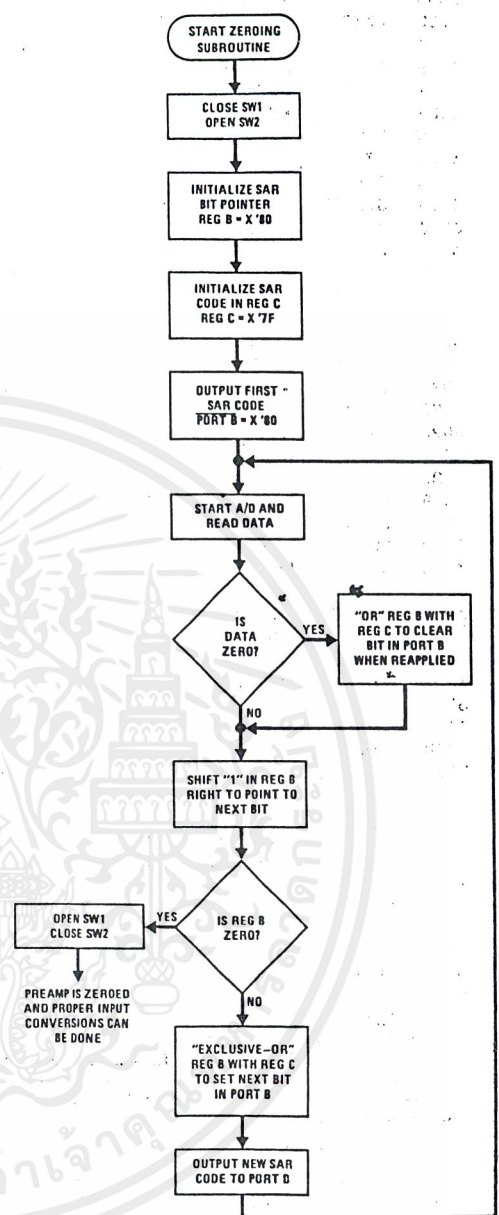
The actual program is given in Figure 20. All addresses used are compatible with the BLC 80/10 microcomputer system. In particular:

- Port A and the ADC0801 are at port address E4
- Port B is at port address E5
- Port C is at port address E6
- PPI control word port is at port address E7
- Program Counter automatically goes to ADDR:3C3D upon acknowledgement of an interrupt from the ADC0801

**5.3 Multiple A/D Converters In a Z-80 Interrupt Driven Mode**

In data acquisition systems where more than one A/D converter (or other peripheral device) will be interrupting program execution of a microprocessor, there is obviously a need for the CPU to determine which device requires servicing. Figure 21 and the accompanying software is a method of determining which of 7 ADC0801 converters has completed a conversion (INTR asserted) and is requesting an interrupt. This circuit allows starting the A/D converters in any sequence, but will input and store valid data from the converters with a priority sequence of A/D 1 being read first, A/D 2 second, etc., through A/D 7 which would have the lowest priority for data being read. Only the converters whose INT is asserted will be read.

The key to decoding circuitry is the DM74LS373, 8-bit D type flip-flop. When the Z-80 acknowledges the interrupt, the program is vectored to a data input Z-80 subroutine. This subroutine will read a peripheral status word from the DM74LS373 which contains the logic state of the INTR outputs of all the converters. Each converter which initiates an interrupt will place a logic "0" in a unique bit position in the status word and the subroutine will determine the identity of the converter and execute a data read. An identifier word (which indicates which A/D the data came from) is stored in the next sequential memory location above the location of the data so the program can keep track of the identity of the data entered.



TL/H/5671-28

FIGURE 19. Flow Chart for Auto-Zero Routine

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

3D00 3E90 MVI 90 ; Program PPI
3D02 D3E7 Out Control Port
3D04 2601 MVI H 01 Auto-Zero Subroutine
3D06 7C MOV A, H
3D07 D3E6 OUT C ; Close SW1 open SW2
3D09 0680 MVI B 80 ; Initialize SAR bit pointer
3D0B 3E7F MVI A 7F ; Initialize SAR code
3D0D 4F MOV C, A Return
3D0E D3E5 OUT B ; Port B = SAR code
3D10 31AA3D LXI SP 3DAA Start ; Dimension stack pointer
3D13 D3E4 OUT A ; Start A/D
3D15 FB IE
3D16 00 NOP Loop ; Loop until INT asserted
3D17 C3163D JMP Loop
3D1A 7A MOV A, D Auto-Zero
3D1B C600 ADI 00
3D1D CA2D3D JZ Set C ; Test A/D output data for zero
3D20 78 MOV A, B Shift B
3D21 F600 ORI 00 ; Clear carry
3D23 1F RAR ; Shift "1" in B right one place
3D24 FE00 CPI 00 ; Is B zero? If yes last
3D26 CA373D JZ Done ; approximation has been made
3D29 47 MOV B, A
3D2A C3333D JMP New C
3D2D 79 MOV A, C Set C
3D2E B0 ORA B ; Set bit in C that is in same
3D2F 4F MOV C, A ; position as "1" in B
3D30 C3203D JMP Shift B
3D33 A9 XRA C New C ; Clear bit in C that is in
3D34 C30D3D JMP Return ; same position as "1" in B
3D37 47 MOV B, A Done ; then output new SAR code.
3D38 7C MOV A, H ; Open SW1, close SW2 then
3D39 EE03 XRI 03 ; proceed with program. Preamp
3D3B D3E6 OUT C ; is now zeroed.
3D3D . Normal
.
.
.
Program for processing
proper data values
3C3D DBE4 IN A Read A/D Subroutine ; Read A/D data
3C3F EEFF XRI FF ; Invert data
3C41 57 MOV D, A
3C42 78 MOV A, B ; Is B Reg = 0? If not stay
3C43 E6FF ANI FF ; in auto zero subroutine
3C45 C21A3D JNZ Auto-Zero
3C48 C33D3D JMP Normal

```

Note: All numerical values are hexadecimal representations.

FIGURE 20. Software for Auto-Zeroed Differential A/D

**5.3 Multiple A/D Converters in a Z-80® Interrupt Driven Mode (Continued)**

The following notes apply:

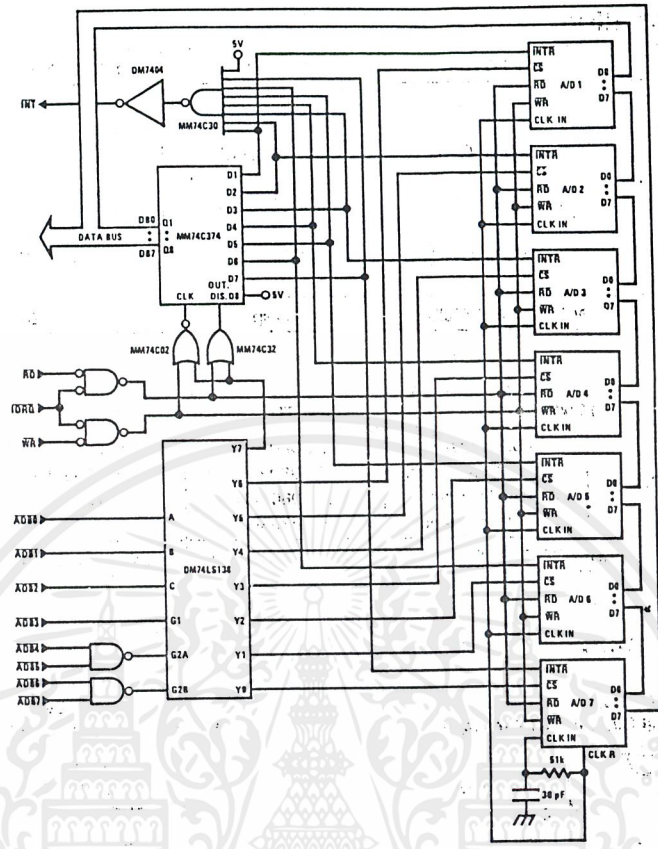
- 1) It is assumed that the CPU automatically performs a RST 7 instruction when a valid interrupt is acknowledged (CPU is in interrupt mode 1). Hence, the subroutine starting address of X0038.
- 2) The address bus from the Z-80 and the data bus to the Z-80 are assumed to be inverted by bus drivers.
- 3) A/D data and identifying words will be stored in sequential memory locations starting at the arbitrarily chosen address X 3E00.
- 4) The stack pointer must be dimensioned in the main program as the RST 7 instruction automatically pushes the PC onto the stack and the subroutine uses an additional 6 stack addresses.

- 5) The peripherals of concern are mapped into I/O space with the following port assignments:

HEX PORT ADDRESS	PERIPHERAL
00	MM74C374 8-bit flip-flop
01	A/D 1
02	A/D 2
03	A/D 3
04	A/D 4
05	A/D 5
06	A/D 6
07	A/D 7

This port address also serves as the A/D identifying word in the program.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



TL/H/5671-29

FIGURE 21. Multiple A/Ds with Z-80 Type Microprocessor

INTERRUPT SERVICING SUBROUTINE

LOC	OBJ CODE	SOURCE STATEMENT	COMMENT
0038	E5	PUSH HL	; Save contents of all registers affected by
0039	C5	PUSH BC	; this subroutine.
003A	F5	PUSH AF	; Assumed INT mode 1 earlier set.
003B	21 00 3E	LD (HL), X3E00	; Initialize memory pointer where data will be stored.
003E	0E 01	LD C, X01	; C register will be port ADDR of A/D converters.
0040	D300	OUT X00, A	; Load peripheral status word into 8-bit latch.
0042	DB00	IN A, X00	; Load status word into accumulator.
0044	47	LD B, A	; Save the status word.
0045	79	TEST LD A, C	; Test to see if the status of all A/D's have
0046	FE 08	CP, X08	; been checked. If so, exit subroutine
0048	CA 60 00	JPZ, DONE	
004B	7E	LD A, B	; Test a single bit in status word by looking for
004C	1F	RRA	; a "1" to be rotated into the CARRY (an INT
004D	47	LD B, A	; is loaded as a "1"). If CARRY is set then load
004E	DA 5500	JPC, LOAD	; contents of A/D at port ADDR in C register.
0051	0C	NEXT INC C	; If CARRY is not set, increment C register to point
0052	C3 4500	JP, TEST	; to next A/D, then test next bit in status word.
0055	ED 78	LOAD IN A, (C)	; Read data from interrupting A/D and invert
0057	EE FF	XOR FF	; the data.
0059	77	LD (HL), A	; Store the data
005A	2C	INC L	
005B	71	LD (HL), C	; Store A/D identifier (A/D port ADDR).
005C	2C	INC L	
005D	C3 51 00	JP, NEXT	; Test next bit in status word.
C360	F1	DONE POP AF	; Re-establish all registers as they were
0061	C1	POP BC	; before the interrupt.
0062	E1	POP HL	
0063	C9	RET	; Return to original program

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

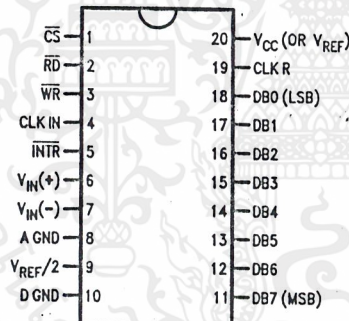
### Ordering Information

TEMP RANGE		0°C TO 70°C	0°C TO 70°C	0°C TO 70°C	-40°C TO +85°C
ERROR	± 1/4 Bit Adjusted	ADC0802LCWM	ADC0802LCV	ADC0804LCN	ADC0801LCN
	± 1/2 Bit Unadjusted				ADC0802LCN
	± 1/2 Bit Adjusted	ADC0803LCWM	ADC0803LCV		ADC0803LCN
	± 1 Bit Unadjusted	ADC0804LCWM	ADC0804LCV		ADC0805LCN
PACKAGE OUTLINE		M20B—Small Outline	V20A—Chip Carrier	N20A—Molded DIP	

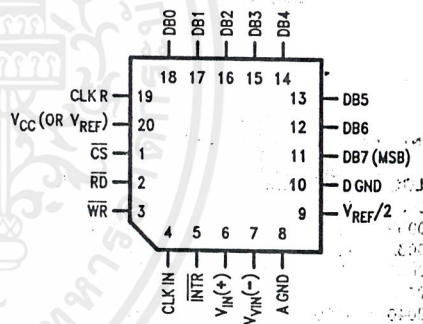
TEMP RANGE		-40°C TO +85°C	-55°C TO +125°C
ERROR	± 1/4 Bit Adjusted	ADC0801LCJ	ADC0801LJ
	± 1/2 Bit Unadjusted	ADC0802LCJ	ADC0802LJ,
	± 1/2 Bit Adjusted	ADC0803LCJ	ADC0802LJ/883
	± 1 Bit Unadjusted	ADC0804LCJ	
PACKAGE OUTLINE		J20A—Cavity DIP	J20A—Cavity DIP

### Connection Diagrams

ADC080X  
Dual-In-Line and Small Outline (SO) Packages



ADC080X  
Molded Chip Carrier (PCC) Package



TL/H/5671-30

See Ordering Information

TL/H/5671-30

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้