

สำนักหอสมุดกลาง พระจอมเกล้าลาดกระบัง

เครื่องชั่งน้ำหนักด้วยไมโครคอมพิวเตอร์

MICROCOMPUTER WEIGHT SCALING MACHINE



โดย

นางสาวพรเพชร มาสมหันต์

นางสาวอัญญาวดี สุริยคุปต์

ปฏิญานี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต

สาขาวิศวกรรมการวัดคุม

คณะวิศวกรรมศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2542

เลขหมึก.....
เลขทะเบียน..... 36793

วัน, เดือน, ปี 29 ต.ค. 2543

ปีการศึกษา 2542

เครื่องชั่งน้ำหนักด้วยไมโครคอมพิวเตอร์

MICROCOMPUTER WEIGHT SCALING MACHINE

โดย

นางสาวพรเพชร มาสมหันต์

นางสาวอัญญาวดี สุริยคุปต์

อาจารย์ที่ปรึกษา

อาจารย์สาท คำมูล

ปริญญาโทชั้นปีการศึกษา 2542

ภาควิชาวิศวกรรมการวัดคุม

คณะวิศวกรรมศาสตร์สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เรื่อง เครื่องชั่งน้ำหนักด้วยไมโครคอมพิวเตอร์

Microcomputer Weight Scaling Machine

ผู้จัดทำ

1. นางสาวพรเพชร มาสมหันต์ 39014353
2. นางสาวอัญญาวดี สุริยคุปต์ 39014668

.....อาจารย์ที่ปรึกษา

(อาจารย์สาท คำมูล)

เครื่องชั่งน้ำหนักด้วยไมโครคอมพิวเตอร์

นางสาวพรเพชร มาสมหันต์ 39014553

นางสาวอัญญาวดี สุริยคุปต์ 39014668

อาจารย์ที่ปรึกษา

อาจารย์สาท คำมูล

ปีการศึกษา 2542

บทคัดย่อ

ปริญญานิพนธ์เล่มนี้นำเสนอโครงการ เครื่องชั่งน้ำหนักด้วยไมโครคอมพิวเตอร์ โดยได้นำเอาไมโครคอนโทรลเลอร์ 8051 มาประยุกต์ใช้ในการควบคุมการทำงานของ โหลดเซลล์และเชื่อมต่อกับระบบคอมพิวเตอร์ ซึ่งโครงการนี้สามารถแสดงผลของน้ำหนัก ทางจอคอมพิวเตอร์ และทางเสียงพูดและนำผลมาประเมินเพื่อบอกรูปร่างของผู้ชั่งได้ เหมาะสำหรับนำไปใช้งานตามสถานที่ต่างๆ เช่น โรงพยาบาล สถานศึกษา ห้างสรรพสินค้า เป็นต้น ซึ่งข้อมูลของผู้ชั่งจะถูกเก็บไว้ในฐานข้อมูล (Database) เพื่อใช้ในการจัดการทางฐานข้อมูลต่อไป

MICROCOMPUTER WEIGHT SCALING MACHINE

PORNPECH MASMAHAN 39014353

AUNYAWADEE SURİYAKUPT 39014608

ADVISER

SART KUMMOON

EDUCATION YEAR 2542

ABSTRACT

The Microcomputer weight scaling machine inside this thesis, It is Weight presented to be weight various detail of the microcomputer weight Scaling machine. This is done by using a microcontroler chip number 8051, And the chip is used to control the operation of this application. The Scaling Machine is able to display result of the object weight on to PC' s monitor And although speak out the weight result. Then, result is baught into the Estimation through roughly describe the phisical size of object, finally the Microcomputer weight scaling machine is suitable for the portable use in Difference specific such as Hospital, School-University , Department store etc. In which data of weighed objects is stored in database to be use in further Data managing.

สารบัญ

	หน้า
บทคัดย่อภาษาไทย	I
Abstract	II
บทที่ 1 บทนำ	
1.1 ความสำคัญของโครงการ	1
1.2 วัตถุประสงค์ของโครงการ	1
1.3 ขอบเขตโครงการ	1
1.4 วิธีการดำเนินงาน	2
1.5 ประโยชน์ที่คาดว่าจะได้รับจากโครงการ	2
บทที่ 2 ทฤษฎี	3
2.1.1 อุปกรณ์ทรานส์ควิเซอร์	3
2.1.2 Operational Amplifiers เบื้องต้น	25
2.1.3 ส่วนแปลงอนาล็อกเป็นดิจิทัล	28
บทที่ 3 โครงสร้างของ MCS - 51	36
3.1.1 คุณสมบัติของไมโครคอนโทรลเลอร์ตระกูล MCS - 51	36
3.1.2 โครงสร้างภายในของ 8051	36
3.1.3 พอร์ทของ 8051	38
3.1.4 การต่อหน่วยความจำ Program Memory และ Data memory	42
3.1.5 การแบ่งประเภทของหน่วยความจำ	43
3.1.6 ชุดคำสั่งของ MCS - 51	44
3.1.7 การสื่อสารข้อมูลแบบอนุกรม	45
บทที่ 4 การประมวลผลและการแสดงผล	50
4.1 เบลไฟล์	50
4.2 เบลไฟล์กับฐานข้อมูล	54
4.3 การบอกนำหน้าด้วยเสียงพูด	58

สารบัญ (ต่อ)

บทที่ 5	วงจรการทำงานและผลการทดลอง	64
5.1.1	วงจร Instrument Amplifiers	64
5.1.2	วงจรอินเตอร์เฟซ (Interface)	65
5.1.3	ผลการทดลองของวงจร Analog to Digital	67
5.1.4	การทำงานของโพลคเซลล์	67
5.1.5	ผลการทดลองโพลคเซลล์	68
5.1.6	ผลการทดลองวงจรรวม	69
บทที่ 6	สรุปผลและแนวทางปฏิบัติ	76
	ภาคผนวก	77
	ภาคผนวก ก Source Code ของโปรแกรม Delphi และไมโครคอนโทรลเลอร์	
	ภาคผนวก ข รายละเอียดข้อมูลของไอซีที่ใช้ในโครงการ	
	กิตติกรรมประกาศ	
	บรรณานุกรม	

สารบัญรูป

	หน้า
รูปที่ 2.1 เกจความเครียดแบบไม่มีขีดขีด	4
รูปที่ 2.2 ชนิดของเกจความเครียด	5
รูปที่ 2.3 การติดเกจความเครียด	5
รูปที่ 2.4 สมดุลย์บริดจ์ของเกจความเครียด	6
รูปที่ 2.5 ไม่สมดุลย์บริดจ์ของเกจความเครียด	6
รูปที่ 2.6 พลัทธิของ E_0'/E_0 เทียบกับ R_0/R_1	8
รูปที่ 2.7 ลิซีบริดจ์	9
รูปที่ 2.8 เอซีบริดจ์	9
รูปที่ 2.9 การชดเชยอุณหภูมิด้วยดัมมิเกจ	10
รูปที่ 2.10 การจัดวางเกจความเครียดโดยใช้เกจ 4 ตัว	11
รูปที่ 2.11 รูปแบบการจัดวางที่เป็นไปได้สำหรับการวัดแรง P	12
รูปที่ 2.12 การจัดวางเกจความเครียดเพื่อการวัดแรง P	13
รูปที่ 2.13 การจัดวางในอีกลักษณะหนึ่ง	14
รูปที่ 2.14 การใช้ Apex Resister เพื่อการสมดุลย์และการชดเชย	15
รูปที่ 2.15 การจัดวางสมดุลย์แบบขนาน	15
รูปที่ 2.16 การสอบเทียบโดยการขนานตัวต้านทาน	16
รูปที่ 2.17 การสอบเทียบเกจความเครียดทางไดนามิกโดยใช้ลูกเบียร์	17
รูปที่ 2.18 การสอบเทียบทางไดนามิกโดยใช้ตัวสั้นสะท้อน	17
รูปที่ 2.19 โหลดเซลล์แบบลิงค์	19
รูปที่ 2.20 โหลดเซลล์แบบคาน	21
รูปที่ 2.21 โหลดเซลล์แบบวงแหวน	23
รูปที่ 2.22 Basic Op- Amp	25
รูปที่ 2.23 Single-Ended Operation	25
รูปที่ 2.24 Inverting Constant-Gain multiplier	26
รูปที่ 2.25 Noninverting Constant-Gain multiplier	26
รูปที่ 2.26 Op-Amp Application Voltage subtraction	27
รูปที่ 2.27 Subtraction Circuit Only one Op-Amp	27
รูปที่ 2.29 ตัวแปลงสัญญาณอนาล็อกเป็นดิจิทัล	28

สารบัญรูป (ต่อ)

รูปที่ 2.30 คุณสมบัติของ A/D converter แบบ 4 บิต	29
รูปที่ 2.31 แสดงการตอบสนองของเวลาแปลงสัญญาณของ A/D converter	30
รูปที่ 2.32 ขบวนการแปลงสัญญาณดิจิทัลด้วย A/D converter	30
รูปที่ 3.33 วงจร Sample and Hold	31
รูปที่ 3.34 การทำงานของวงจรอินทิเกรเตอร์	31
รูปที่ 3.35 ตัวแปลงสัญญาณ A/D คอนเวอร์เตอร์แบบแรมปี	32
รูปที่ 3.36 กราฟเอาต์พุตของส่วน D/A ของ A/D คอนเวอร์เตอร์	33
รูปที่ 3.37 คอนเวอร์เตอร์ซึ่งใช้เทคนิค SAR	34
รูปที่ 3.38 กราฟของส่วน D/A ระหว่างแรงสัญญาณ	34
รูปที่ 3.1 บล็อกไดอะแกรมของ MCS-51	37
รูปที่ 3.2 การจัดวางขาของ 8051	38
รูปที่ 3.3 แสดงโครงสร้างพอร์ต 0 (บิต)	38
รูปที่ 3.4 แสดงโครงสร้างพอร์ต 1 (บิต)	39
รูปที่ 3.5 แสดงโครงสร้างพอร์ต 2 (บิต)	40
รูปที่ 3.6 แสดงโครงสร้างพอร์ต 0 (บิต)	41
รูปที่ 3.7 การต่อขาเรซีตให้กับ 8051	42
รูปที่ 3.8 การต่อหน่วยความจำ	42
รูปที่ 3.9 ผังแสดงหน่วยความจำสำหรับเก็บโปรแกรมเบอร์ 8051	43
รูปที่ 3.10 ผังแสดงหน่วยความจำสำหรับ Data Memory เบอร์ 8051	43
รูปที่ 3.11 แสดงบล็อกไดอะแกรมพอร์ตคอนโทรล	46
รูปที่ 4.1 แสดงการติดต่อกับฐานข้อมูล	57
รูปที่ 4.2 แสดงส่วนประกอบต่างๆของเครื่องซึ่ง	59
รูปที่ 4.3 แสดงไฟว์ชาร์ทการทำงานของเครื่องซึ่งนำหน้า	60
รูปที่ 4.4 แสดงไฟว์ชาร์ทของโปรแกรมประมวลผล	61
รูปที่ 4.5 แสดงไฟว์ชาร์ทของโปรแกรมประมวลผลพิเศษ	62
รูปที่ 4.6 แสดงไฟว์ชาร์ทของโปรแกรมประมวลผลพิเศษเชิง	63
รูปที่ 5.1 วงจร Instrument Amplifier	64
รูปที่ 5.2 แสดงไฟว์ชาร์ทการทำงานของวงจรอินเตอร์เฟซ	65

สารบัญรูป (ต่อ)

รูปที่ 5.3	วงจรอินเตอร์เฟซ	66
รูปที่ 5.4	แสดงหน้าจอก่อนจะทำการชั่งน้ำหนัก	72
รูปที่ 5.5	แสดงหน้าจอผลการทำงานของเครื่องชั่ง	73
รูปที่ 5.6	แสดงหน้าจอการจัดเก็บข้อมูล	74
รูปที่ 5.7	แสดงหน้าจอผลการค้นหาข้อมูล	75

สารบัญตาราง

	หน้า
ตาราง 5.1 ผลการทดลองวงจร Analog to Digital	67
ตาราง 5.2 แสดงค่าเอาต์พุตที่ได้จากโพลีเซตล์	68
ตาราง 5.3 เปรียบเทียบค่าแรงดันเอาต์พุตของโพลีเซตล์กับค่าความละเอียดที่ได้จากวงจรแปลงระดับสัญญาณอนาล็อกเป็นดิจิทัล	69

บทที่ 1

บทนำ

1.1 ความสำคัญของโครงการ

ในปัจจุบันอุตสาหกรรมและการควบคุมต่างๆ ได้พัฒนาไปมากจากเดิมที่ส่วนใหญ่จะเป็นแบบธรรมดาเป็นแบบกึ่งอัตโนมัติและแบบอัตโนมัติ ดังนั้นอุปกรณ์เครื่องมือวัดต่างๆ ที่ใช้ในการประกอบการเหล่านี้จะต้องได้รับการพัฒนาตามไปด้วย จะเห็นได้ว่าในปัจจุบันเครื่องคอมพิวเตอร์ได้เข้ามามีบทบาทอย่างมากไม่ว่าจะเป็นงานด้านเอกสาร หรืองานด้านการควบคุมอุตสาหกรรมที่เรามักพบได้เสมอ ดังนั้นแนวทางการพัฒนาอุปกรณ์ส่วนใหญ่จึงมีทิศทางเดียวกัน นั่นคือการทำทุกอย่างให้ใช้งานร่วมกันได้ โดยการใช้คอมพิวเตอร์เป็นตัวเชื่อมต่อเข้าด้วยกัน

เครื่องชั่งน้ำหนักด้วยไมโครคอมพิวเตอร์นี้มีหลักการทำงาน คือมีส่วนที่รับค่าน้ำหนักเข้ามาและมีส่วนแสดงผล ส่วนที่รับค่าน้ำหนักจะประกอบไปด้วย Load Cell วงจร Pre – Amplifier ทำหน้าที่ในการขยายสัญญาณและกำหนดค่าสัญญาณที่เหมาะสม ให้กับวงจร ADC เพื่อทำการแปลงสัญญาณอนาล็อกเป็นสัญญาณดิจิทัล แล้วส่งไปยังคอมพิวเตอร์ เพื่อทำการประมวลผลและแสดงผลทางจอคอมพิวเตอร์

1.2 วัตถุประสงค์ของโครงการ

1. ทำเครื่องชั่งน้ำหนักโดยใช้ โหลดเซลล์ เป็นตัวเซนเซอร์และประมวลผลด้วยคอมพิวเตอร์
2. แสดงผลของน้ำหนักและประเมินเกณฑ์ได้
3. เพื่อศึกษาคุณสมบัติและวิเคราะห์อุปกรณ์ทรานส์ดิวเซอร์

1.3 ขอบเขตโครงการ

1. เครื่องชั่งน้ำหนักสามารถบอกน้ำหนัก
 - ผ่านทางจอคอมพิวเตอร์
 - ทางเสียงพูด
 - ผ่านเครื่อง Print
2. มีฐานข้อมูลไว้เก็บข้อมูลของผู้ชั่ง

1.4 วิธีการดำเนินงาน (โดยสังเขป)

1. ศึกษาการใช้โปรแกรมเซลล์ไฟล์และค้นคว้าเกี่ยวกับระบบฐานข้อมูล
2. ศึกษาและออกแบบระบบการทำงาน
3. ศึกษาการทำงานของอุปกรณ์ที่ใช้ในโครงการ
4. ทดสอบการทำงานของระบบ
5. ปรับปรุงแก้ไข

1.5 ประโยชน์ที่คาดว่าจะได้รับจากโครงการ

1. เข้าใจการทำงานของระบบฐานข้อมูล
2. สามารถออกแบบวงจรการทำงานทางอิเล็กทรอนิกส์
3. เข้าใจหลักการทำงานของไมโครคอนโทรลเลอร์
4. สามารถใช้โปรแกรมเซลล์ไฟล์มาใช้ในการทำโครงการ
5. สามารถนำโครงการไปพัฒนาในทางธุรกิจต่อไป
6. ได้นำความรู้ทางทฤษฎีมาทดลองและประยุกต์งานใช้ได้จริง
7. ได้ผลิตเครื่องมือทางเทคโนโลยีที่สามารถใช้ประโยชน์ได้ต่อไป
8. ได้ศึกษาถึงปัญหาในการปฏิบัติงานจริงและแก้ปัญหาเหล่านั้นได้

บทที่ 2 ทฤษฎี

2.1 อุปกรณ์ทรานส์ดิวเซอร์

เกจความเครียด (Strain Gages)

ทรานส์ดิวเซอร์ชนิดนี้อยู่บนพื้นฐานที่ว่าถ้าตัวนำไฟฟ้าอันหนึ่งถูกดึงให้ตึง หรือ ถูกอัดตัวค่าความต้านทานของมันจะเปลี่ยนแปลง อันเนื่องจากการเปลี่ยนในความยาวพื้นที่ หรือ พิกัดความต้านทาน (Resistivity) ของมัน ค่าความต้านทาน R ของตัวนำไฟฟ้าอันหนึ่ง มีพื้นที่หน้าตัด A ความยาว L และ พิกัดความต้านทานของวัสดุ ρ คือ

$$R = \frac{\rho l}{\mu A} \quad (2.1)$$

เกจแฟกเตอร์ F ของตัวนำไฟฟ้าถูกนิยามเหมือนกับ

$$F = \frac{\Delta R/R}{\Delta L/L} = \frac{\Delta R/R}{\zeta} \quad (2.2)$$

เมื่อ ΔR เป็นการเปลี่ยนแปลงของค่าความต้านทาน R เนื่องจากเกจความเครียด ζ ซึ่ง

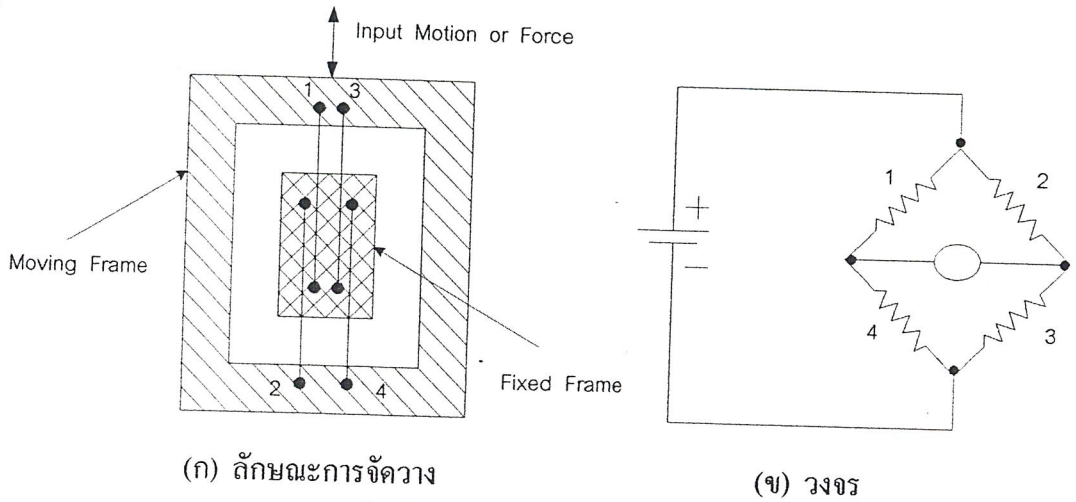
$$\zeta = \frac{\Delta L}{L}$$

ในทางปฏิบัติ ตัวนำไฟฟ้าที่นำมาใช้มักอยู่ในรูปของเส้นลวดเล็ก ๆ หรือ โลหะแผ่นบาง ทรานส์ดิวเซอร์เกจความเครียดสามารถแบ่งออกๆ ได้เป็น 2 ชนิด คือ

1. เกจความเครียดชนิดไม่ยึดติด
2. เกจความเครียดชนิดยึดติด

เกจความเครียดชนิดไม่ยึดติด เกจความเครียดชนิดนี้ ลวดค่าความต้านทานเส้นหนึ่งจะถูกดึงให้ตึงระหว่าง 2 เฟรม เฟรมหนึ่งเป็นเฟรมเคลื่อนที่ ขณะที่อีกเฟรมหนึ่งอยู่กับที่ ดังแสดงในรูปที่ 2.1(ก) ขนาดของเส้นลวดตามปกติแล้วมีความยาว 25 mm. และเส้นผ่านศูนย์กลาง 25 μ m. เพลทที่บิดงอได้จะทำหน้าที่เช่นเดียวกับสปริงระหว่างเฟรมทั้งสอง เส้นลวดจะอยู่ภายใต้การ

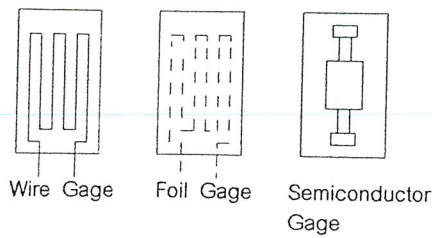
ป้อนโพลดอันใหม่ ซึ่งมากกว่าโพลดที่มีการอัดตัวที่ตั้งเอาไว้ในตอนแรก การเคลื่อนที่ของอินพุตที่แสดงในรูปที่ 2.1 จะจึงเส้นลวดเส้นที่ 1 และ 3 ให้ตึง ขณะเดียวกันก็จะลดความตึงในลวดเส้นที่ 2 และ 4 การเคลื่อนที่ในทิศทางตรงกันข้ามกันก็จะให้ผลที่ตรงกันข้าม เส้นลวดจะถูกนำมาต่อเข้ากับวงจรวิทซ์โทนบริดจ์ แสดงดังในรูปที่ 2.1 (ข)



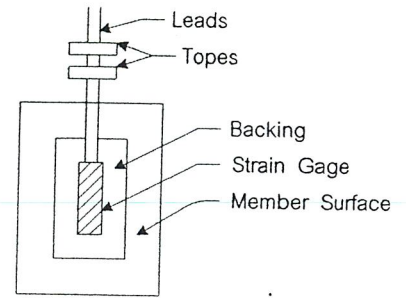
รูปที่ 2.1 เกจความเครียดชนิดไม่ยึดติด

โดยทรานส์ดิวเซอร์ชนิดนี้สามารถวัดการเคลื่อนที่ที่มีขนาดเล็กมาก ๆ ถึง $50 \mu\text{m}$. และแรงที่มีขนาดเล็กมาก ๆ อีกด้วย ทรานส์ดิวเซอร์ชนิดนี้อาจนำไปใช้เพื่อการวัด แรง ความดัน และความเร่งได้

เกจความเครียดชนิดยึดติด ทรานส์ดิวเซอร์เกจความเครียดชนิดยึดติด ถูกใช้อย่างกว้างขวางเพื่อวัดสิ่งที่แปรเปลี่ยนไปทางกายภาพหลาย ๆ อย่าง เช่น ความเครียด แรง แรงบิด ความดัน และการสั่นสะเทือนเป็นต้น เกจเหล่านี้มักจะทำด้วยวัสดุจำพวกโลหะหรือสารกึ่งตัวนำและอยู่ในรูปของเกลลวด (เส้นผ่าศูนย์กลางประมาณ 25 m .) หรือโลหะแผ่นบางๆหรือแท่งเล็กๆ (ในกรณีของสารกึ่งตัวนำ) ดังแสดงในรูปที่ 2.2 เกจเหล่านี้มีกระดาศหรือวัสดุชนิดอื่นห่อหุ้ม เพื่อใช้ยึดติดกับพื้นผิวที่ต้องการวัดความเครียด ดังแสดงในรูปที่ 2.3 เมื่อยึดติดแล้ว เกจจะได้รับความเครียดเหมือนกับความเครียดที่มีอยู่ในพื้นผิวนั้น เกจเหล่านี้มีความไวสูงมากและเมื่อนำไปใช้ร่วมกับอุปกรณ์อิเล็กทรอนิกส์ ความเครียดที่ต่ำประมาณ 10^{-7} อาจวัดได้



รูปที่ 2.2 ชนิดของเกจความเครียด



รูปที่ 2.3 การติดเกจความเครียด

เกจที่ทำด้วยโลหะผสมทองแดง – นิกเกิ้ล มีเกจเฟเตอร์ 2 ถึง 3 ขณะที่ เซมิคอนดักเตอร์ เกจมีเกจเฟเตอร์ 100 ถึง 200 ตารางที่ 1 จะให้ข้อมูลเกี่ยวกับวัสดุห่อหุ้มเกจ ชนิดเส้นลวด และสารยึดติด

ตารางที่ 1

วัสดุห่อหุ้มเกจ	สารยึดติด	ชนิดเส้นลวด	หมายเหตุ
กระดาษหรือไหม	Nitrocellulose	โลหะผสม Cu, Ni	ใช้ได้สูงถึง 60 องศา
เบคาไลท์	Epoxy	โลหะผสม Cu, Ni	ใช้ได้สูงถึง 200 องศา
Glass Weave	Ceramic cement	โลหะผสม Cr, Ni	ใช้ได้สูงถึง 400 องศา

กระแสที่ไหลผ่านเกจปกติถูกจำกัดอยู่ที่ 10 ถึง 30 mA ขึ้นอยู่กับระยะเวลาในการทดสอบ เพื่อป้องกันความเสียหายที่จะเกิดขึ้นกับเส้นลวด เกจที่ห่อหุ้มด้วยเบคาไลท์สามารถต้านทานกระแสที่มีค่าค่อนข้างสูงได้

เราจะต้องให้ความระมัดระวังขณะทำการติดเกจ พื้นผิวที่ต้องการติดเกจจะต้องทำให้สะอาดตลอดเวลา ต่อมาจึงใส่สารยึดติดตามคำแนะนำของโรงงานผู้ผลิต แล้วทำการเชื่อมต่อขาของเกจและยึดติดอย่างหนาแน่นเข้ากับส่วนที่ต้องการทดสอบ จากนั้นตรวจสอบการเชื่อมต่อเกจและฉนวนเป็นลำดับสุดท้าย

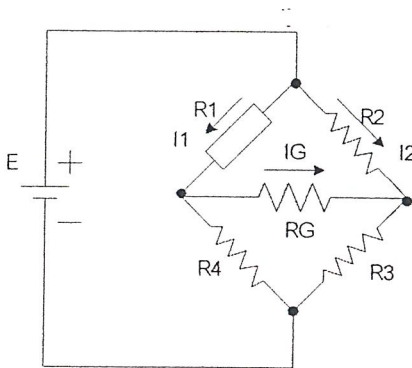
บริดจ์ของเกกความเครียด

เกกความเครียด โดยปกติแล้ว จะถูกค่อเป็นส่วนหนึ่งของวงจรวีทชโตนบริดจ์เพื่อว่าการเปลี่ยนแปลงในค่าความต้านทานของมัน อันเนื่องมาจากความเครียดจะได้วัดได้หรือให้เอาท์พุทที่สามารถนำไปแสดงผลหรือบันทึกผลได้ มีการจัดวางบริดจ์อยู่สองรูปแบบ นั่นคือ

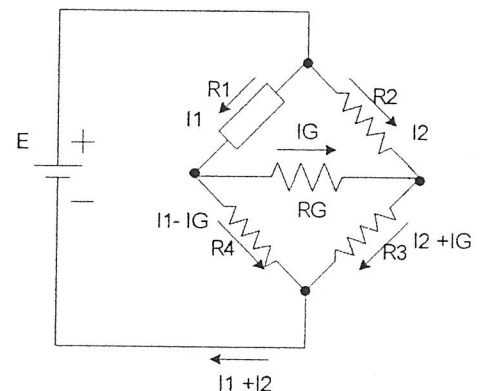
1. สมดุลย์บริดจ์ (Balanced bridge)
2. ไม่สมดุลย์บริดจ์ (Unbalanced bridge)

ในการจัดวางแบบสมดุลย์บริดจ์ รูปที่ 2.4 ค่าความต้านทานของเกกความเครียด R_1 เป็นแขนข้างหนึ่งของวีทชโตนบริดจ์ ขณะที่แขนที่เหลือมีค่าความต้านทาน R_2 R_3 และ R_4 บริดจ์จะถูกกระตุ้นด้วยแหล่งจ่ายคิซีด้วยแรงเคลื่อนไฟฟ้า E และ R_G เป็นค่าความต้านทานของกัลวาโนมิเตอร์ บริดจ์กล่าวได้ว่าสมดุล เมื่อไม่มีกระแสไหลผ่านกัลวาโนมิเตอร์ เงื่อนไขของการสมดุลย์ซึ่งเป็นที่รู้จักกันดีคือ

$$\frac{R_1}{R_4} = \frac{R_2}{R_3}$$



รูปที่ 2.4 สมดุลย์บริดจ์ของเกกความเครียด



รูปที่ 2.5 ไม่สมดุลย์บริดจ์ของเกกความเครียด

ถ้า R_1 มีการเปลี่ยนแปลงเนื่องจากความเครียด บริดจ์ซึ่งสมดุลย์ในตอนต้นก็จะไม่สมดุลย์ อันนี้อาจจะทำให้สมดุลย์ได้อีกครั้งโดยการเปลี่ยนแปลงค่า R_4 หรือ R_2 การเปลี่ยนแปลงอันนี้สามารถวัดได้และถูกใช้เพื่อการแสดงการเปลี่ยนแปลงของ R_1 ซึ่งมีความสัมพันธ์กับความเครียด เทคนิคอันนี้สามารถนำมาใช้เพื่อวัดความเครียดทางสถิติ เพียงอย่างเดียว

การจัดวางแบบไม่สมดุลย์บริดจ์ ที่แสดงในรูปที่ 2.5 กระแสที่ไหลผ่านกัลวานมิเตอร์ หรือแรงเคลื่อนไฟฟ้าที่ตกคร่อมกัลวานมิเตอร์ถูกใช้ เพื่อแสดงความเครียดในแกนความเครียด วิธีนี้สามารถใช้เพื่อวัดความเครียดทางไดนามิกได้ดีเช่นเดียวกับการวัดความเครียดทางสถิตินี้ เพื่อแสดงถึงการได้มาของความสัมพันธ์ระหว่างกระแสเอาท์พุท ที่ไม่สมดุลย์เกี่ยวกับการเปลี่ยนแปลงในค่าความต้านทานของแกน R_1 สามารถเขียนสมการได้ดังนี้โดยใช้กฎของ Kirchhoff

$$I_1 R_1 + R_4 (I_1 - I_G) = E \quad (2.3)$$

$$I_1 R_1 + I_G R_G - I_2 R_2 = 0 \quad (2.4)$$

$$I_G R_G + (I_2 + I_G) R_3 - (R_1 - I_G) R_4 = 0 \quad (2.5)$$

แก้สมการที่ (2.3) (2.4) และ (2.5) เพื่อหาค่า I_G เราจะได้

$$I_G = \frac{E(R_2 R_4 - R_1 R_3)}{[R_2(R_1 + R_4)(R_G + R_4 + R_3) + R_1 R_3 R_4 - R_2 R_4 + R_G R_3(R_1 + R_4)]} \quad (2.6)$$

จากสมการที่ (2.6) จะพบว่า $I_G = 0$ เมื่อ $R_2 R_4 = R_1 R_3$ อันนี้สอดคล้องกับสมการที่ (2.3)

จากสมการที่ (2.6) ค่ากระแสที่ไม่สมดุลย์ I_G สามารถคำนวณได้สำหรับการเปลี่ยนแปลงใดๆ ในค่าความต้านทานของแกนความเครียด R_1 สิ่งที่ได้เมื่อ $R_1 = R_2 = R_3 = R_4$ และถ้า R_1 เปลี่ยนแปลงเป็นค่า $R_1 + \Delta R_1$ เราจะได้สมการที่ (2.6) คือ

$$I_G = \frac{-E \Delta R_1}{4R_1(R_1 + R_G)} \quad (2.7)$$

$$= \frac{E F S_1}{(R_1 + I_G)} \quad (2.8)$$

เมื่อ เป็นความเครียดที่เกิดจาก R_1 และ

$$S_1 = \frac{\Delta R_1}{R_1 F}$$

F เป็นแกนแฟลคเตอร์ของแกนความเครียด

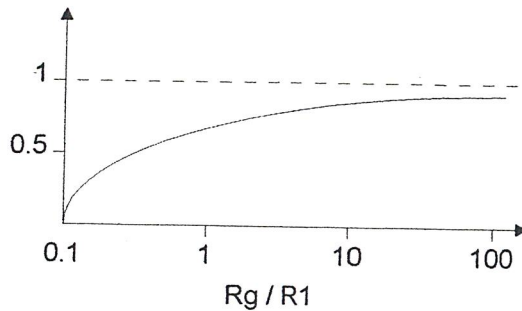
แรงเคลื่อนไฟฟ้าเอาท์พุทตกคร่อม R_G (E_0)

$$\begin{aligned} &= I_G R_G \\ &= \frac{-E S_1 R_G F}{4(R_1 + R_G)} \end{aligned} \quad (2.9)$$

$$\text{แรงคลื่นไฟฟ้าเอทพุทที่เปิดวงจร (E')} = \frac{-EF1}{4} \quad (2.10)$$

อันนี้เกิดขึ้นได้โดยการกำหนดให้ $R_G \rightarrow \infty$ ในสมการที่ (2.9)

อัตราส่วนของ E'/E_0 ถูกพล็อตเทียบกับ R_G / R_1 ในรูปที่ 2.6 เราพบว่า R_G จะต้องมีค่าอย่างน้อยร้อยละเท่าของค่าความต้านทานของเกจความเครียด R_1 เพื่อจะได้ให้เอทพุท E_0 มากที่สุดเท่าที่จะเป็นไปได้



รูปที่ 2.6 พล็อตของ E'/E_0 เทียบกับ R_G/R_1

เมื่อแขนของวิชโตนบริดจ์มากกว่าหนึ่งแขน บรรจุด้วยเกจความเครียดและค่าความต้านทานของมันเปลี่ยนแปลงเนื่องจากความเครียด เอทพุทที่ได้รับก็คือผลรวมของการเปลี่ยนแปลงเหล่านั้น นั่นคือรูปที่ 2.7 ถ้า R_2 เปลี่ยนแปลงเป็น $R_2 + \Delta R_2$ และในตอนต้นกำหนดให้ $R_1 = R_2 = R_3 = R_4$

สมการที่ (2.7) จะกลายเป็น

$$I_G = \frac{-E\Delta R_2}{4R_2(R_1 + R_G)}$$

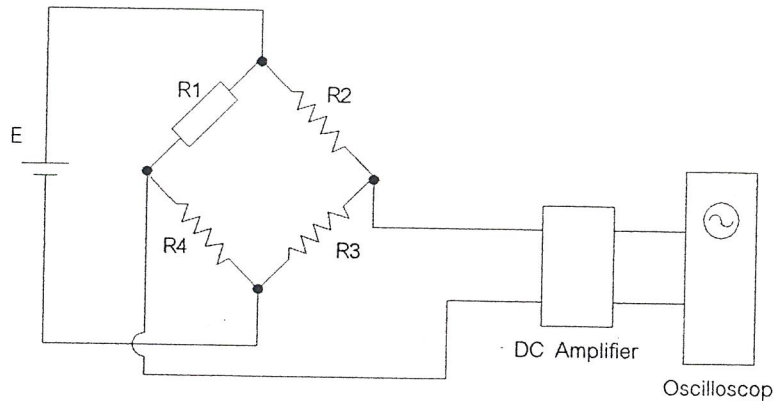
ดังนั้นถ้า R_1 เปลี่ยนแปลงเป็น $R_1 + \Delta R_1$ และ R_2 เปลี่ยนเป็น $R_2 + \Delta R_2$ ผลรวมของการเปลี่ยนแปลงคือ

$$I_G = -\frac{E}{4(R_1 + R_G)} \left[\frac{\Delta R_1}{R_1} - \frac{\Delta R_2}{R_2} \right] \quad (2.11)$$

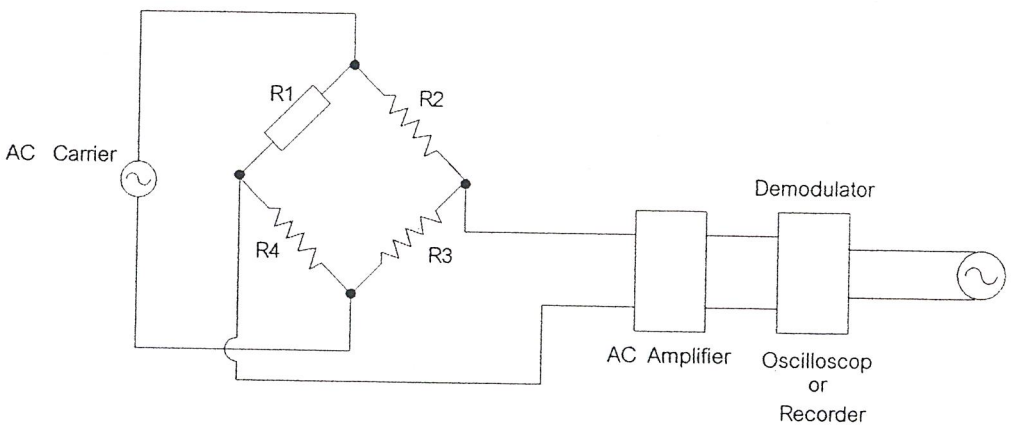
ในทำนองเดียวกัน ถ้าแขนทั้งสี่ของบริดจ์มีเกจความเครียดต่ออยู่ ซึ่งค่าความต้านทานของมันจะเปลี่ยนแปลงเนื่องจากความเครียด เราจะพบว่า

$$I_G = \frac{E}{4(R_2 + R_G)} \left[\frac{-\Delta R_1}{R_1} + \frac{\Delta R_2}{R_2} - \frac{\Delta R_3}{R_3} + \frac{\Delta R_4}{R_4} \right] \quad (2.12)$$

ด้วยเหตุนี้ เราจะพบว่ากระแสเอาต์พุตจะเพิ่มขึ้นถ้าการเปลี่ยนแปลงของค่าความต้านทาน ΔR_2 และ ΔR_4 และมีธรรมชาติที่เหมือนกับ ΔR_3 ดังนั้นแขนที่อยู่ใกล้ชิดกันของบริดจ์จะต้องมีความเครียดในธรรมชาติที่ตรงกันข้ามกัน เพื่อให้เอาต์พุตมีขนาดใหญ่เหนือเพิ่มความไวของบริดจ์ อันนี้จะต้องให้ความสนใจขณะจัดวางเกจความเครียดบนพื้นที่ที่ต้องการวัดในลักษณะที่ว่าถ้า R_1 เป็นความดึงเครียด R_2 ควรจะเป็นความอัดเครียด R_3 เป็นความดึงเครียด และ R_4 ควรจะเป็นความอัดเครียด ดังนั้น มันเป็นไปได้ที่จะให้คำนิยามของคำว่า “แฟกเตอร์สนับสนุนสัญญาณ (Signal enhancement factor) ของบริดจ์” เหมือนกับค่าอัตราส่วนของเอาต์พุตสูงสุด เนื่องจากการเปลี่ยนแปลงในเกจความเครียดหลายๆ ตัวต่อเอาต์พุตสูงสุดที่ได้รับจากการใช้เกจความเครียดเพียงตัวเดียวบนพื้นที่ที่ต้องการวัด หรือกล่าวอีกนัยหนึ่ง เกจที่มีมากกว่าหนึ่งตัว (ภายใต้ความเครียด) ถูกจัดวางไว้อย่างเหมาะสมสามารถทำให้ความไวเพิ่มมากขึ้นหรือทำให้สัญญาณเพิ่มสูงขึ้น



รูปที่ 2.7 ดีซีบริดจ์



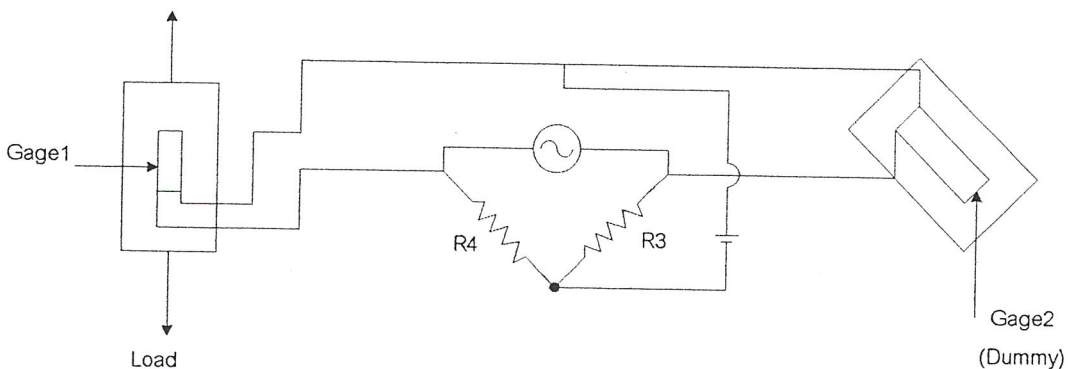
รูปที่ 2.8 เอซีบริดจ์

การกระตุ้นบริดจ์อาจจะทำด้วยแหล่งจ่ายไฟดีซีหรือเอซี ดังวงจรที่ให้ไว้ในรูปที่ 2.7 และ 2.8 ในทั้งสองกรณี ตัวขยายเป็นสิ่งที่ต้องการเพื่อการคัพปลิง ในกรณีของเอซีบริดจ์ สัญญาณเป็นแอมพลิฟายด์สัญญาณเหมือนแสดงในรูปที่ 2.8 ดังนั้นคิมมอดูเลเตอร์จึงเป็นสิ่งที่ต้องการเพื่อการกรองความถี่พาหะเอาท์พุทที่ได้จะเหมือนกับความถี่ที่กำลังถูกวัด ใน จำพวกของบริดจ์ทั้งสองชนิด เอซีบริดจ์จะไม่มีเปลี่ยนแปลงที่เอาท์พุท เนื่องจากอุณหภูมิ และไม่มีสัญญาณรบกวน ซึ่งตามปกติแล้วมักจะพบในดีซีบริดจ์ อย่างไรก็ตาม เอซีบริดจ์มี ราคาแพงและการตอบสนองต่อความถี่สูงของมันถูกจำกัดที่ประมาณ 1 ใน 5 ของความถี่พาหะที่ใช้

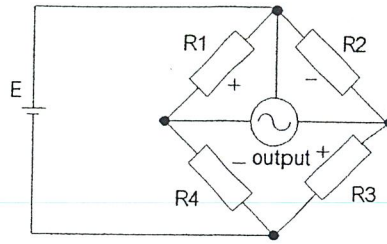
การชดเชยอุณหภูมิ นอกจากความเครียดแล้วการเปลี่ยนแปลงของอุณหภูมิจะมีผลทำให้เกิด การเปลี่ยนแปลงของค่าความต้านทานในเกจความเครียดด้วย แต่มันไม่สะดวกที่จะใช้การ คำนวณเพื่อการแก้ไขอุณหภูมิ การแก้ไขและการชดเชยอุณหภูมิที่สร้างขึ้น โดยตัวของมันเองจะ สะดวกกว่า อันนี้ทำได้โดย

1. ใช้คัมมิเกจ (Dummy gage)
2. ใช้เกจมากกว่าหนึ่งตัวที่มีการจัดวางไว้อย่างเหมาะสม

คัมมิเกจ (Dummy gage) แสดงดังรูปที่ 2.9 เกจค่าความต้านทาน R_2 เท่ากับ R_1 ถูกยึดติดบน แท่งของวัสดุชนิดเดียวกับแท่งวัสดุทดสอบ (Test Specimen) ซึ่งกำลังได้รับโหลด อย่างไรก็ตาม แท่งของวัสดุที่มี R_2 ไม่ได้รับความเครียด แต่อยู่ภายใต้การเปลี่ยนแปลงของอุณหภูมิอัน เดียวกันเหมือนกับแท่งวัสดุทดสอบ ดังนั้น R_1 ก็คือเกจใช้งานภายใต้ความเครียดและอุณหภูมิมี การเปลี่ยนแปลงไปพร้อมๆ กัน ขณะที่ R_2 คือคัมมิเกจภายใต้การเปลี่ยนแปลงของอุณหภูมิอัน เดียวกันด้วยเหตุที่ R_1 และ R_2 เป็นแขนที่ใกล้ชิดกันของบริดจ์เอาท์พุทเนื่องจากอุณหภูมิ เปลี่ยนแปลงจึงเป็นศูนย์ เพราะว่า R_1 และ R_2 มีการเปลี่ยนแปลงเนื่องจากอุณหภูมิเป็นจำนวน เท่ากัน



รูปที่ 2.9 การชดเชยอุณหภูมิด้วยคัมมิเกจ



รูปที่ 2.10 การจัดวางเกจความเครียดโดยใช้เกจ 4 ตัว

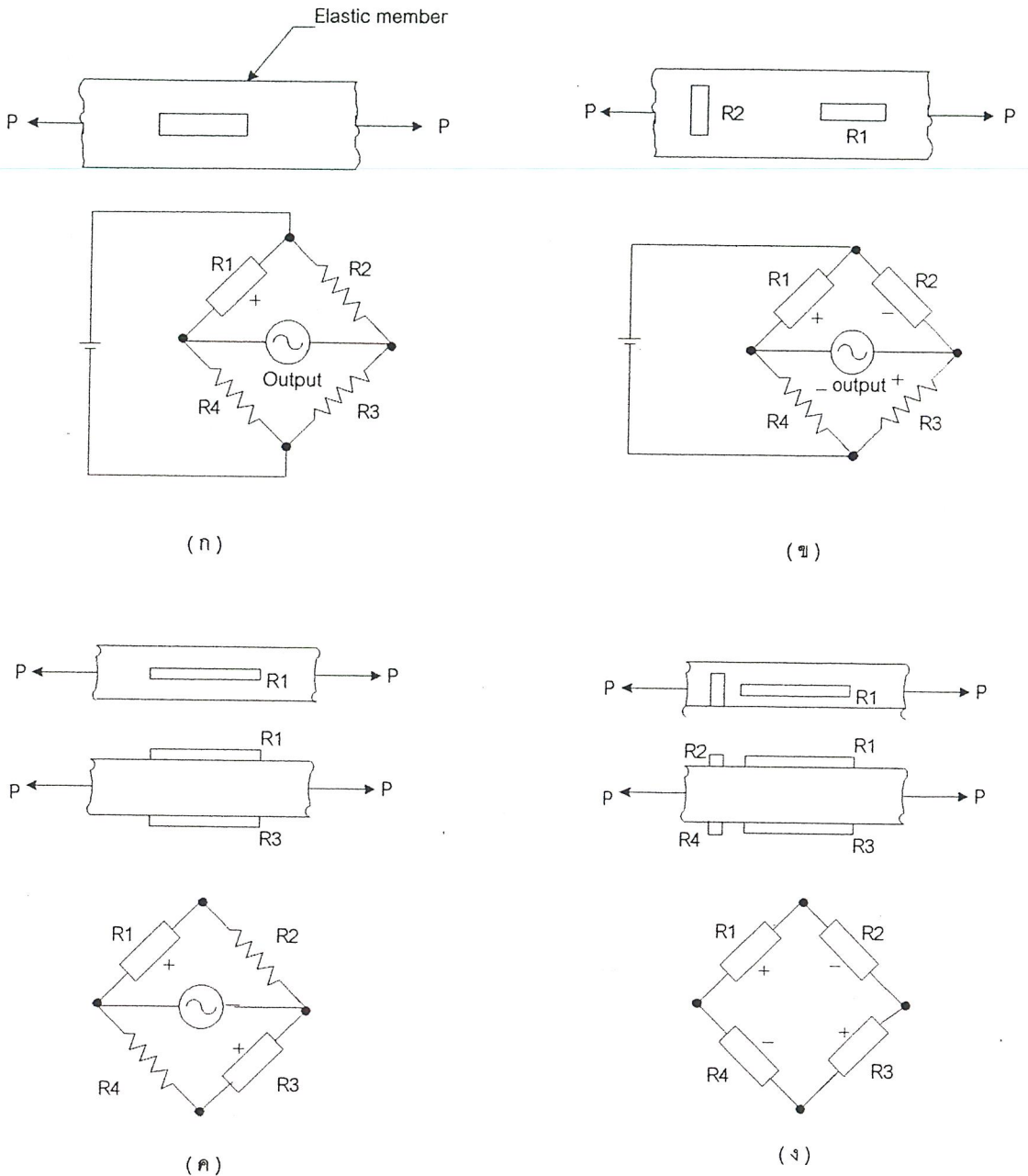
เกจมากกว่าหนึ่งตัวที่มีการจัดวางไว้อย่างเหมาะสม เกจ R_2 ไม่ได้เป็นคัมมิเกจแต่ถูกวางไว้อย่างเหมาะสมบนแท่งวัสดุทดสอบเพื่อว่ามันจะได้มีความเครียดในธรรมชาติที่ตรงกันข้ามกับความเครียดใน R_1 ในกรณีเช่นนี้จะทำให้ระบบมีความไวและมีการชดเชยอุณหภูมิไปพร้อม ๆ กัน แท้จริงแล้วถ้าแขนทั้ง 4 ของบริดจ์เป็นเกจใช้งานเหมือนแสดงในรูปที่ 2.10 โดยแขนที่ใกล้เคียงกันของบริดจ์มีความเครียดในธรรมชาติที่ตรงกันข้าม สัญญาณเอาต์พุตจะมีค่าสูงสุดและมีการชดเชยอุณหภูมิด้วย และเป็นเรื่องปกติที่ค่าความต้านทานเริ่มต้นของทุกๆ แขนของบริดจ์จะต้องมีค่าเท่ากัน

การชดเชยอุณหภูมิเป็นสิ่งจำเป็นเมื่อต้องการวัดความเครียดทางสถิติ เพราะว่าการเปลี่ยนแปลงของอุณหภูมิจะเป็นผลทำให้เกิดการเปลี่ยนแปลงที่สัญญาณเอาต์พุต ซึ่งการเปลี่ยนแปลงนี้จะน้อยมากสำหรับการวัดความเครียดทางไดนามิก ดังนั้น การวัดความเครียดทางไดนามิกจึงไม่จำเป็นต้องมีการชดเชยอุณหภูมิ

การจัดวางเกจความเครียด (Strain gage arrangement) มีปัจจัย 2 อย่างต่อไปนี่ที่ต้องคำนึงถึงขณะตัดสินใจวางเกจความเครียดบนชิ้นส่วนยึดหยุ่นเพื่อวัดการเปลี่ยนแปลงทางกายภาพคือ

1. ความไวสูง
2. การชดเชยอุณหภูมิ

การจัดวางที่แสดงในรูปที่ 2.10 เป็นสิ่งที่สอดคล้องกับการพิจารณาทั้งสองประการข้างต้น

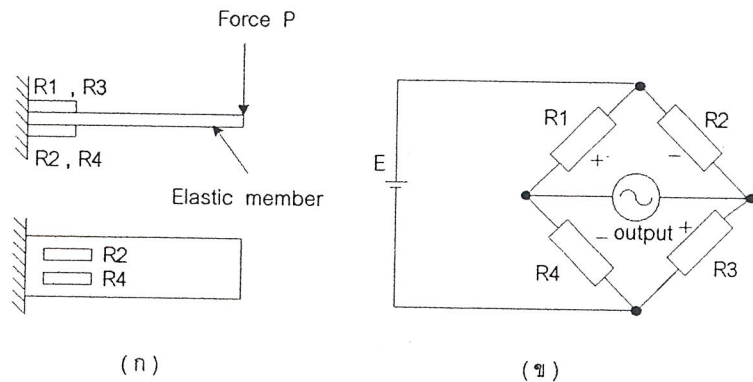


รูปที่ 2.11 รูปแบบการจัดวางที่เป็นไปได้สำหรับการวัดแรง P

รูปแบบของการจัดวางเกจที่เป็นไปได้ของการวัดแรงคั้น P ที่ป้อนเข้าไปที่ชิ้นส่วนยึดหยุ่น แสดงดังรูปที่ 2.11 (ก) (ข) (ค) และ (ง) โดยเกจความเครียดแสดงด้วยรูปสี่เหลี่ยมผืนผ้า ในรูปที่ เกจ R_1 ถูกยึดติดอยู่บนชิ้นส่วนยึดหยุ่น เพื่อวัดความเครียดตามแนวแกนในชิ้นส่วนยึดหยุ่น แต่ระบบไม่มีการชดเชยอุณหภูมิและเอาท์พุทจะขึ้นอยู่กับ การเปลี่ยนแปลงของ R_1 เพียงอย่างเดียว

ในการจัดวางตามรูปที่ 2.11 (จ) เกจสองตัว R_1 และ R_2 ถูกวางไว้เป็นมุมฉาก หรือเรียกว่า "การจัดวางแบบพอยสัน (Poisson's arrangement)" ดังนั้น R_1 จะเปลี่ยนแปลงเนื่องจากความตึงเครียด ตามแนวแกน ขณะที่ R_2 เปลี่ยนเนื่องจากความอัดเครียดตามแนวขวางในชิ้นส่วนยึดหยุ่น โดยความเครียดอันหลังจะเป็น v เท่าของความเครียดอันแรก เมื่อ v หมายถึงอัตราส่วนของพอยสัน (Poisson's ratio) แฟกเตอร์สนับสนุนสัญญาณจะเป็น $(1 + v)$ เพราะว่าคุณค่าความต้านทานของ R_1 และ R_2 มีความเครียดในธรรมชาติที่ตรงกันข้าม และระบบมีการชดเชยอุณหภูมิด้วยเนื่องจาก การเปลี่ยนแปลงของอุณหภูมิจะมีผลกระทบต่อ R_1 และ R_2 เท่าๆกัน จึงไม่มีผลต่อเอาต์พุต การจัดวางของรูปที่ 2.11 (ก) ซึ่งทั้ง R_1 และ R_3 อยู่ภายใต้ความตึงเครียดตามแนวแกนด้วยจำนวนที่เท่ากันและ R_1 และ R_3 ถูกติดไว้บนแขนของบริดจ์ด้านตรงข้ามกัน จึงให้แฟกเตอร์สนับสนุนสัญญาณเป็น 2 เท่า แต่จะไม่มี การชดเชยอุณหภูมิ ดังนั้นการจัดวางแบบนี้จึงต้องการสองคัมมิเกจเพื่อทำให้เกิดความมั่นใจในเรื่องการชดเชยอุณหภูมิ

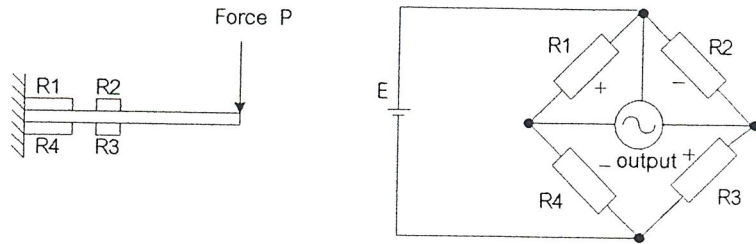
การจัดวางของรูปที่ 2.11 (ง) ซึ่งใช้เกจสี่ตัวโดย R_2 และ R_4 ถูกจัดวางไว้เป็นมุมฉากกับ R_1 และ R_3 จะให้แฟกเตอร์สนับสนุนสัญญาณเป็น $2(1 + v)$ และมีการชดเชยการเปลี่ยนแปลงของอุณหภูมิ



รูปที่ 2.12 การจัดวางเกจความเครียดเพื่อการวัดแรง P

อีกตัวอย่างหนึ่งแรง P สามารถวัดได้โดยใช้คานยึดหยุ่น (Elastic cantilever) ดังแสดงในรูปที่ 2.12 (ก) เกจความเครียดถูกยึดติดอยู่ที่โคนของคาน ตรงที่ความเครียดของการโค้งมากที่สุด R_1 และ R_3 มีความตึงเครียด ขณะที่ R_2 และ R_4 มีความอัดเครียด การจัดวางบริดจ์ของรูปที่ 2.12 (ข) จะให้แฟกเตอร์สนับสนุนสัญญาณเป็น 4 และมีการชดเชยอุณหภูมิด้วย การจัดวางในลักษณะอื่นดังแสดงในรูป 2.13 (ก) ซึ่ง R_1 และ R_2 ถูกยึดติดตามการจัดวางของพอยสัน

และอันนี้ก็ใช้กับ R_1 และ R_2 ด้วย การจัดวางของบริดจ์ดังรูปที่ 2.13 (ข) แขนที่ติดกันจะมีความเครียดในธรรมชาติที่ตรงกันข้ามกัน แม้ว่าจะมีขนาดไม่เท่ากันก็ตาม การจัดวางของรูปที่ 2.13 (ข) จะให้แฟคเตอร์สนับสนุนสัญญาณเป็น $2(1 + \nu)$ เมื่อ ν เป็นอัตราส่วนของพอยสัน และอุณหภูมิก็จะถูกชดเชยด้วย



รูปที่ 2.13 การจัดวางในอีกลักษณะหนึ่ง

การสมดุลย์ของบริดจ์ (Balancing of Bridges)

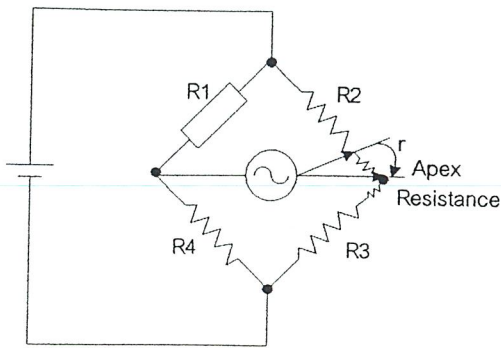
บริดจ์ต้องทำให้สมดุลย์ในตอนเริ่มต้น อย่างไรก็ตามในทางปฏิบัติ ค่าความต้านทานที่แขนทั้งสองอาจจะไม่เท่ากัน วิธีใดวิธีหนึ่งต่อไปนี้จะถูกนำมาเพื่อใช้ในการสมดุลย์บริดจ์

1. วิธีการสมดุลย์แบบอนุกรม (Series balancing method) โดยใช้ Apex resistor แสดงในรูป 2.14 เนื่องจากการเคลื่อนที่ที่จุดสัมผัสของ Apex resistance ถ้า R_2 เพิ่มขึ้นเป็น $r R_2$ ก็จะลดลงเป็น r ด้วย

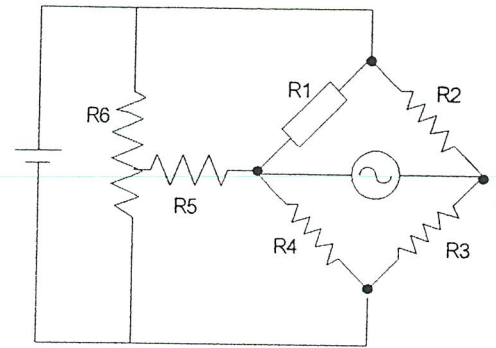
2. วิธีการสมดุลย์แบบขนาน (Parallet balancing method) บริดจ์สามารถทำให้สมดุลย์ได้โดยการเคลื่อนที่ที่จุดสัมผัส P

การสอบเทียบ

การสอบเทียบกลายเป็นสิ่งที่จำเป็นสำหรับการจัดวางวงจรบริดจ์แบบไม่สมดุลย์ซึ่งเอาท์พุทของบริดจ์ได้จากความเครียดของเกจความเครียด การสอบเทียบอาจทำได้ทั้งทางไฟฟ้าและทางกล วิธีการทางไฟฟ้าโดยปกติจะสร้างขึ้นในวงจรของบริดจ์ความเครียด สามารถพบเห็นได้โดยทั่วไป



รูปที่ 2.14 การใช้ Apex resistor เพื่อการ
สมดุลและการสอบเทียบ



รูปที่ 2.15 การจัดวางสมดุลแบบขนาน

วิธีการทางไฟฟ้า

วิธีแรกการเปลี่ยนแปลงของค่าความต้านทานที่แขนอาจเลียนแบบ โดยการใช้ Apex resistance ของรูปที่ 2.14 มันเป็นไปได้ที่จะสอบเทียบ Apex resistance ในเทอมของความเครียด (ξ) ในแกนตัวเดียว (สมมติว่าเป็น R_1) ความสามารถอันนี้แสดงดังนี้
จากรูป 2.14 ถ้าบริดจ์ถูกทำให้สมดุลในตอนแรกนั้นคือ

$$\frac{R_1}{R_4} = \frac{R_2}{R_3} = 1 \quad (2.13)$$

จากนั้น Apex resistance เปลี่ยนแปลงไป นั่นคือ R_2 เพิ่มขึ้นเป็น R_2+r และ R_3 ลดลงเป็น R_3+r บริดจ์เปลี่ยนเป็นไม่สมดุล แต่ค่าของ $\Delta R_1/R_1$ สอดคล้องกับการเปลี่ยนแปลงข้างต้น

$$\frac{R_1 + \Delta R_1}{R_4} = \frac{R_2 + r}{R_3 - r} \quad (2.14)$$

กำหนดให้ $x = r/R_3$ เราจะพบว่า

$$\frac{\Delta R_1}{R_1} = \frac{2x}{1-x} \quad (2.15)$$

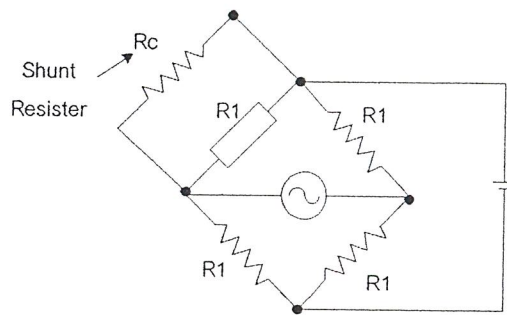
ดังนั้น Apex resistance สามารถสอบเทียบได้ในเทอมของ $\Delta R_1/R_1$ หรือความเครียด ξ ถ้ารู้ค่าแกนแฟกเตอร์

การสอบเทียบทางวงจรไฟฟ้าวิธีที่สองโดยการใช้ค่าความต้านทาน (Shunt resistor) ดังแสดงในรูปที่ 2.16 ถ้าตัวต้านทานขนาน R_c ถูกต่อขนานกับเกจความเครียด R_1 การเปลี่ยนแปลง ΔR_1 คือ

$$\Delta R_1 = R_1 - R_3 // R_c \quad (2.16)$$

หรือเราอาจเขียนสมการใหม่เป็น

$$\frac{\Delta R_1}{R_1} = \frac{\Delta R_1}{R_1 + R_c} \approx \frac{R_1}{R_c} \quad \text{ถ้า } R_c \gg R_1 \quad (2.17)$$



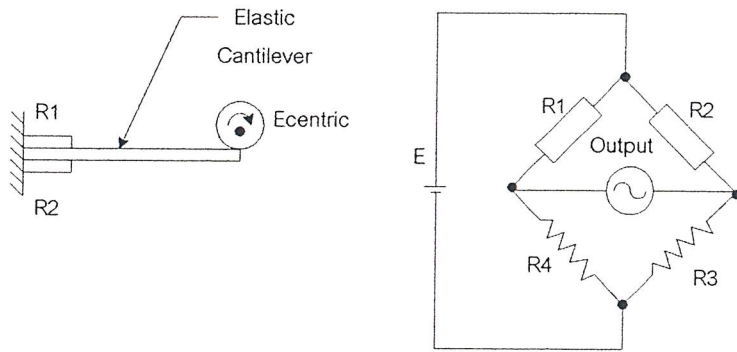
รูปที่ 2.16 การสอบเทียบโดยการขนานตัวต้านทาน

ซึ่งโดยทั่วไป สำหรับ $\Delta R_1/R_1$ มีค่าน้อยมากๆ ดังนั้น การเปลี่ยนแปลงที่เอาท์พุทซึ่งเป็นผลมาจากการต่อ R_c ขนานกับค่าความต้านทาน R_1 สามารถวัดได้หรือมีความสัมพันธ์กับ $\Delta R_1/R_1$ หรือความเครียด ในการทดสอบทางกล สำหรับการสอบเทียบทาง Static dead weights สามารถนำมาใช้ได้โดยตรงและความเครียดสามารถคำนวณได้และมีความสัมพันธ์กับเอาท์พุท

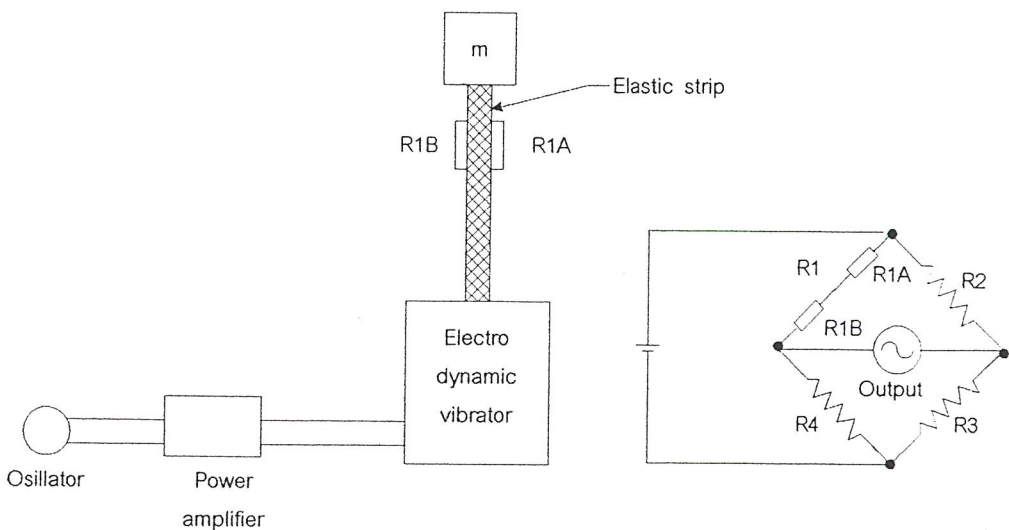
วิธีทางไดนามิก

การสอบเทียบทางไดนามิกก็มีสองวิธี วิธีแรกจะใช้การหมุนของลูกเบี้ยว (Rotating eccentric) ในรูปที่ 2.17 และวิธีหลังจะใช้ตัวสั่นสะเทือน (Vibration) ในรูปที่ 2.18 คานยึดหยุ่นถูกใช้เหมือนกับอุปกรณ์ยึดหยุ่น ความเครียดที่แท้จริงสามารถคำนวณได้และแพคเตอร์ของการสอบเทียบซึ่งมีความสัมพันธ์กับเอาท์พุทและความเครียดสามารถคำนวณได้ดังรูปที่ 2.18 ความถี่แอมพลิจูดของตัวสั่นสะเทือนสามารถคำนวณได้ เพราะว่าแรงอย่าง Sinusoidal โดยมีมวล m ถูกป้อนไปที่แท่งยึดหยุ่น (Elastic strip) เป็น $m\omega^2 x_0$ เมื่อ ω เป็น Circular

frequency ของการสั่น และ x_0 เป็นแอมพลิจูดของมวลที่เคลื่อนที่ซึ่งจำเป็นที่จะต้องถูกวัด โดยทรานส์ดิวเซอร์เคลื่อนที่ใดๆที่อธิบายไปแล้ว ตรงข้ามกับอันนี้ เอาท์พุทของบริดจ์สามารถวัดได้ ในรูปที่ 2.18 ทั้ง R1A และ R1B ถูกจัดวางเป็นแขนข้างหนึ่งของวิทซ์โค่นบริดจ์ การกระทำเช่นนี้เพื่อการขจัดผลของการโค้งงอใดๆของแท่งยึดหยุ่น การโค้งงอในแท่งยึดหยุ่น การโค้งงอจะมีการเปลี่ยนแปลงที่เหมือนกันแต่ในธรรมชาติที่ตรงกันข้ามของ R1A และ R1B ทำให้ค่าความต้านทานรวมของแขนไม่เปลี่ยนแปลง ดังนั้นเกจจะตอบสนองความเครียด อันเนื่องมาจากแรงทางไดนามิกที่ได้จากการเคลื่อนที่ของมวล m เพียงอย่างเดียว



รูปที่ 2.17 การสอบเทียบเกจความเครียดทางไดนามิกโดยใช้ลูกเบี้ยว



รูปที่ 2.18 การสอบเทียบทางไดนามิกโดยใช้ตัวสั่นสะเทือน

ทรานสดิวเซอร์แบบหลักการของแรง (Force Transducers)

ทรานสดิวเซอร์นี้ใช้สำหรับเปลี่ยนแรงทางการกายภาพให้เป็นสัญญาณไฟฟ้าแรงสามารถแบ่งออกเป็น 2 ประเภทใหญ่ ๆ

1. แรงจลน์ เป็นแรงที่ทำให้วัตถุเปลี่ยนแปลงความเร็วในการเคลื่อนที่ เช่น แรงที่กระทำบนข้อมือหุ่นยนต์เนื่องจากความเร่งของมวลของวัตถุในระหว่างการบังคับให้เคลื่อนที่
 2. แรงสถิต เป็นแรงที่กระทำบนวัตถุโดยไม่เกี่ยวข้องกับการเคลื่อนที่ เช่น แรงที่กระทำโดยคริปเปอร์ (Qripper) ของหุ่นยนต์บนผิวของวัตถุในระหว่างการจัดการกับวัตถุ
- การวัดแรงทั่ว ๆ ไป ทรานสดิวส์เซอร์จะเปลี่ยนพลังงานกลเป็นพลังงานไฟฟ้า โดยการวัดการเปลี่ยนแปลงรูปร่างทางกายภาพของทรานสดิวส์เซอร์ ซึ่งแรงเป็นตัวการทำให้เกิดขึ้น

ส่วนประกอบยืดหยุ่นที่ใช้ในการวัดแรง (ElasticElement For Force Measurement)

ส่วนยืดหยุ่นที่นิยมใช้สำหรับการแสดงขนาดของแรงที่กระทำโดยผ่านการวัดการกระจัด เช่น

1. สปริง ในกรณีแรงจะหาได้จาก

$$P = ky \tag{2.18}$$

เมื่อ $k =$ ค่าคงตัวของสปริง
 $y =$ ระยะกระจัดจากจุดสมดุล

2. แท่งวัสดุ แรงที่วัดได้จะเป็นไปตามสมการ

$$P = (AE / L) * y \tag{2.19}$$

เมื่อ $A =$ พื้นที่หน้าตัด
 $E =$ ค่าโมดูลัสของความยืดหยุ่นของวัสดุ
 $L =$ ความยาว

3. คานยื่น (Cantilever Beam) การเบนไปของปลายคาน สัมพันธ์กับแรง P ที่กระทำโดย

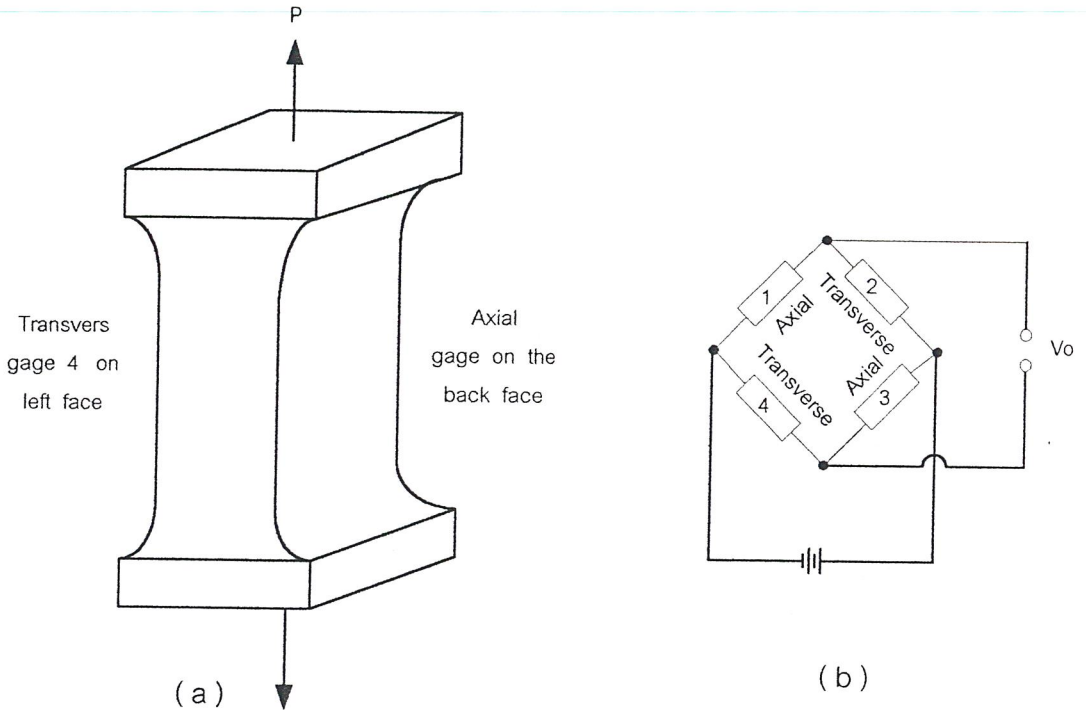
$$P = (3EI / L^3) * y \tag{2.20}$$

เมื่อ $I =$ โมเมนต์ของความเฉื่อยของคานรอบแกนที่ผ่านศูนย์กลางในทิศทางการเบน

นอกจากนี้ยังมี ริงค์ (Ring) และเว็บ (Webs) ซึ่งเป็นส่วนยืดหยุ่นที่ใช้ทำทรานสดิวส์เซอร์และได้มีการแบ่งคโหลดเซลล์เป็นชนิดต่าง ๆ ดังนี้

LOAD CELL

1. โหลดเซลล์แบบลิงค์ (Link-type Load cell)



รูปที่ 2.19

โหลดเซลล์แบบลิงค์อย่างง่าย ประกอบด้วยลิงค์ และเกจความเครียด 4 อัน ดังรูปที่ 2.19 ภาระ P สามารถเป็นไปได้ทั้งแรงดัน (Tensile Load) หรือภาระแรงอัด (Compressive Load) เกจความเครียด 2 อัน ยึดติดกับลิงค์ในแนวแกน และอีก 2 อันในแนวขวางกับแกน

เกจความเครียดทั้ง 4 จะต่อเป็นวงจรวีทสโตนบริดจ์ (Wheatstone Bridge) โดยเกจที่อยู่ในแนวแกนต่ออยู่ในแนวแกนต่อในแกน 1 และ 3 ส่วน เกจที่อยู่ในแนวขวางต่ออยู่ในแนวแกน 2 และ 4 ดังรูปที่ 2.19 เมื่อภาระ P กระทำต่อลิงค์ความเครียดตามแนวแกน และตามแนวขวางจะเกิดขึ้นในลิงค์และสัมพันธ์กับภาระตามสมการ

$$\epsilon_a = P / AE \tag{2.21}$$

$$\epsilon_t = -VP / AE \tag{2.22}$$

เมื่อ A = พื้นที่หน้าตัดของลิงค์

E = โมดูลัสของความยืดหยุ่นของวัสดุที่ใช้ทำลิงค์

V = อัตราส่วนปัวซองของวัสดุที่ใช้ทำลิงค์

การตอบสนองของเกจต่อ โหลด P ที่กระทำเป็นไปตามสมการ

$$\Delta R_1/R_1 = \Delta R_3/R_3 = S_g \epsilon_1 = S_g P/AE \quad (2.23)$$

$$\Delta R_2/R_2 = \Delta R_4/R_4 = S_g \epsilon_2 = -VS_g P/AE \quad (2.24)$$

ถ้าสมมติว่าเกจความเครียดทั้ง 4 ตัวบนลึงค์เหมือนกันทุกประการ แรงดันออก V_0 จาก
วีทสโตนบริดจ์หาได้จากการแทนค่าสมการ (2.24) ลงในสมการ

$$V_0 = \frac{r}{(1+r)^2} \left(\frac{\Delta R_1}{R_1} - \frac{\Delta R_2}{R_2} + \frac{\Delta R_3}{R_3} - \frac{\Delta R_4}{R_4} \right) V_s$$

จะได้

$$V_0 = \frac{S_g P(1+\nu)}{2AE} V_s \quad (2.25)$$

$$P = \frac{2AE}{S_g(1+\nu)} V_0 = C V_0 \quad (2.26)$$

จากสมการ (2.26) แสดงว่าภาระ P เป็นสัดส่วนเชิงเส้นกับแรงดันออก V_0 และค่าคงตัว
สัดส่วนหรือค่าคงตัวเปรียบเทียบคือ

$$C = \frac{2AE}{S_g(1+\nu)} V_s \quad (2.27)$$

เมื่อ C = ค่าคงตัวเปรียบเทียบ

ความไวของโหลดเซลล์วีทสโตนบริดจ์คอมบิเนชัน (Load Cell - Wheatstone Bridge Combination) คือ

$$S = \frac{V_0}{P} = \frac{1}{C} = \frac{S_g P(1+\nu)}{2AE} V_s \quad (2.28)$$

จากสมการ (2.28) แสดงว่าความไวของโหลดเซลล์ขึ้นอยู่กับพื้นที่หน้าตัดของลึงค์ (A) โม
ดูลัสของความยืดหยุ่นของวัสดุที่ใช้ทำลึงค์ (E, V) เกจแฟกเตอร์, S_g และแรงดันที่จ่ายให้กับ
B บริดจ์ (V_s) ช่วง (Range) ของแรงที่ใช้กับโหลดเซลล์แบบลึงค์ หาได้จาก พื้นที่หน้าตัดของ
ลึงค์และความถี่ S_g ของวัสดุที่ใช้ทำลึงค์ โดย

$$P_{max} = S_g A \quad (2.29)$$

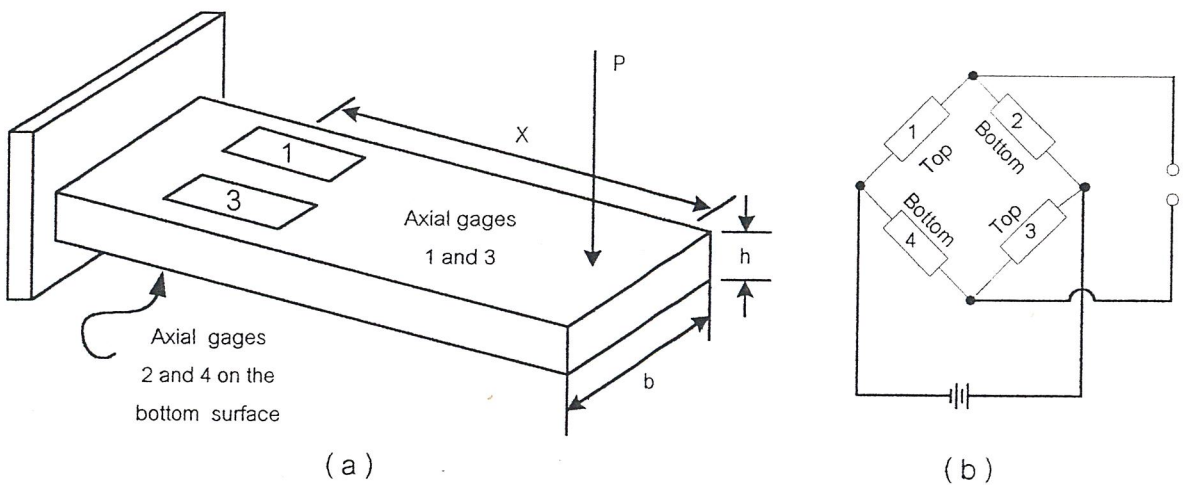
อัตราส่วนแรงดันสูงสุด (Maximum Load) (V_0/V_s)_{max} สำหรับโหลดเซลล์แบบลึงค์หา
ได้จากการแทนสมการข้างบนลงใน (2.25) ได้

$$\left(\frac{V_0}{V_s} \right)_{Max} = \frac{S_g S_g (1+\nu)}{2E} \quad (2.30)$$

โหลดเซลล์แบบลิ้งค์เกือบทั้งหมด กำหนดค่าเต็มสเกลของภาระ ($P = P_{\max}$) ที่ $(V_0/V_s)^* = 3mV/V$ ด้วยค่าเต็มสเกลเฉพาะของอัตราส่วนแรงดัน $(V_0/V_s)^*$ นี้ ภาระ P บนโหลดเซลล์จะหาได้จาก

$$P = \frac{V_0/V_s}{(V_0/V_s)^*} P_{\max} \quad (2.31)$$

2. โหลดเซลล์แบบคาน (BEAN - TYPE LOAD CELL)



รูปที่ 2.20

โหลดเซลล์แบบคานใช้วัดภาระในกรณีที่ใช้โหลดเซลล์แบบลิ้งค์ไม่ได้ รูปที่ 2.20 (a) เป็นคานยื่น (CantiLever Beam) ซึ่งมีเกจความเครียด 2 อัน ติดอยู่ที่ผิวด้านล่าง ซึ่งทำหน้าที่เป็นชิ้นส่วนยึดหยุ่นและเซนเซอร์สำหรับเซลล์ชนิดนี้ คือเกจ ซึ่งต่อกันเป็นวงจรวีทสโตนบริดจ์ ดังรูป ภาระ P ทำให้เกิดโมเมนต์ $M = P_x$ ที่ตำแหน่ง X ทำให้ได้ผลลัพธ์

$$\epsilon_1 = -\epsilon_2 = \epsilon_3 = -\epsilon_4 = \frac{\sigma M}{Ebh^2} = \frac{6P_x}{Ebh^2} \quad (2.32)$$

เมื่อ b คือความกว้างของหน้าตัดของคาน

h คือความหนาของหน้าตัดของคาน

การตอบสนองของเกจความเครียดหาได้จากสมการ $\Delta R/R = S_g \epsilon$ และสมการ (c) ลงในสมการ

$$\Delta R1/R1 = -\Delta R2/R2 = \Delta R3/R3 = \Delta R4/R4 = \frac{\sigma S_g P x}{E b h^2} \quad (2.33)$$

แรงดันออก V_o จากวีทสโตนบริดจ์ซึ่งเป็นผลจากการกระทำของภาระ P หาได้โดยการแทนสมการ ลงในสมการ

$$V_o = \frac{r}{(1+r)^2} \left(\frac{\Delta R1}{R1} - \frac{\Delta R2}{R2} + \frac{\Delta R3}{R3} - \frac{\Delta R4}{R4} \right) V_s$$

และสมมติว่าเกจความเครียดทั้ง 4 เหมือนกันทุกประการ จะได้

$$V_o = \frac{\sigma S_g P x V_s}{E b h^2} \quad (2.34)$$

$$P = \frac{E b h^2}{\sigma S_g X V_s} V_o = C V_o \quad (2.35)$$

จากสมการ (2.35) แสดงว่าภาระ P เป็นสัดส่วนกับแรงดันออก V_o และค่าคงตัวสัดส่วนหรือค่าคงตัวปรับเทียบ (Calibration Constant) C คือ

$$C = (E b h^2) / (\sigma S_g X V_s) \quad (2.36)$$

ความไวของการรวมระหว่างโหนดเซลล์กับวีทสโตนบริดจ์ โดยสมการ $S = V_o / P$ ดังนั้น

$$S = V_o / P = 1/C = (\sigma S_g X V_s) / E b h^2 \quad (2.37)$$

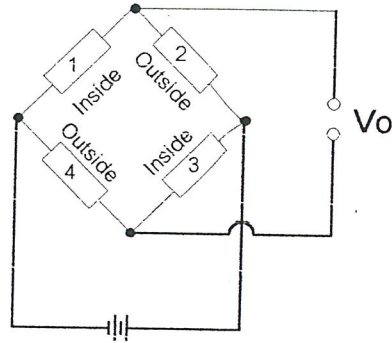
จากสมการ (2.37) ข้างต้นแสดงว่าความไวของโหนดเซลล์แบบคานขึ้นอยู่กับการปรับหน้าตัดของคาน (b, h) โมดูลัสของความยืดหยุ่นของวัสดุที่ใช้ทำคาน (E) ตำแหน่งของภาระเทียบกับเกจ (X) เกจแพกเตอร์ (S_g) และแรงดันที่ป้อนให้กับวงจรวีทสโตนบริดจ์ (V_s) ช่วงของโหนดที่ใช้กับโหนดเซลล์แบบคานขึ้นอยู่กับการปรับหน้าตัดของคานตำแหน่งที่ภาระกระทำ และความถี่ของวัสดุที่ใช้ทำคาน ถ้าสมมติว่าเกจติดที่ตำแหน่งใกล้กับตัวรองรับ (Beam Support) จะได้

$M_{gagc} = M_{max}$ และ $(V_o / V_s)_{max}$ หาได้จากการแทนสมการ P_{max} ลงในสมการ (2.34) ดังนั้น

$$(V_o / V_s)_{max} = S_g S_f / E \quad (2.38)$$

โหนดเซลล์แบบคานที่มีขายในท้องตลาดมีอัตราส่วนแรงดัน (V_o / V_s) ที่ภาระเต็มสเกลระหว่าง 4 และ 5 mV/V

3. โหลดเซลล์แบบวงแหวน (Ring-Type Load Cell)



รูปที่ 2.21

ส่วนประกอบของโหลดเซลล์แบบวงแหวนมี Proving Ring เป็นส่วนยึดหยุ่น และมีเซนเซอร์ซึ่งสามารถใช้ได้ทั้งเกจความเครียดและ LVDT ถ้าใช้ LVDT วัดการกระจัดอันเนื่องมาจากการยืดหรือดึงในแนวเส้นผ่านศูนย์กลางของวงแหวนความสัมพันธ์ระหว่างการกระจัด σ กับภาระ P จะเป็นไปตามสมการ (โดยประมาณ)

$$\delta = (1.79PR^3)/Ewt^3 \quad (2.39)$$

เมื่อ E คือค่าโมดูลัสที่ใช้ทำวงแหวน

w คือความหนา

t คือความหนาของวงแหวน

แรงดันออก V_o ของ LVDT สามารถแสดงโดยสมการ

$$V_o = S \delta V_s \quad (2.40)$$

เมื่อ S คือความไวของ LVDT

V_s คือแรงดันที่ป้อนเข้ากับขดลวดปฐมภูมิของ LVDT

ความสัมพันธ์ระหว่างแรงดันออก V_0 และภาระ P หาได้จากแทนสมการ ลงในสมการได้

$$V_0 = (1.79 \text{ SPR}^3) V_s / \text{Ewt}^3 \quad (2.41)$$

หรือ
$$P = (0.56 \text{ Ewt}^3) V_0 / \text{SR}^3 V_s = C V_0 \quad (2.42)$$

โดย
$$C = (0.56 \text{ Ewt}^3) / \text{SR}^3 V_s \quad (2.43)$$

เมื่อ C คือค่าปรับเทียบ

ความไวของริงค์ LVDT คอมบิเนชัน S_i คือ

$$S_i = V_0 / P = 1/C = (1.79 \text{ SR}^3) V_s / \text{Ewt}^3 \quad (2.44)$$

ช่วงภาระของโพลีแซตต์แบบวงแหวนถูกควบคุมด้วยความแข็งแรงของวัสดุ ถ้าโพลีแซตต์ใช้วัดโพลีที่กระทำเป็นจังหวะ (Cyclic Load) ความล้า (Fatigue) S_f เป็นสิ่งสำคัญที่ต้องพิจารณาความเค้นสูงสุดในชิ้นส่วนวงแหวนอยู่ที่ผิวด้านในของวงแหวน ตรงที่เส้นผ่านศูนย์กลางตั้งฉากกับแนวโพลี ค่าประมาณของความเค้นที่ตำแหน่งนี้คือ

$$\sigma_0 = 1.09 \text{ PR} / \text{Wt}^2 \quad (2.45)$$

จากสมการ สำหรับการวัด โพลีที่กระทำเป็นจังหวะได้

$$P_{\max} = 0.92 \text{ Wt}^2 S_f / R \quad (2.46)$$

อัตราส่วนแรงดันที่โพลีสูงสุด $(V_0/V_s)_{\max}$ จะเป็น

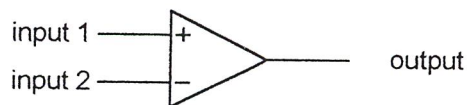
$$(V_0/V_s)_{\max} = 1.64 \text{ SR}^2 S_f / \text{Et} \quad (2.47)$$

ถ้ารู้อัตราส่วนแรงดันเต็มสเกล $(V_0/V_s)^*$ และภาระสูงสุด P_{\max} สำหรับโพลีแซตต์เฉพาะ สามารถกำหนดโพลีที่สอดคล้องกับ V_0 ที่ถูกวัดโดยประมาณ

$$P = (V_0/V_s) P_{\max} / (V_0/V_s)^* \quad (2.48)$$

2.2 Operational Amplifiers เบื้องต้น

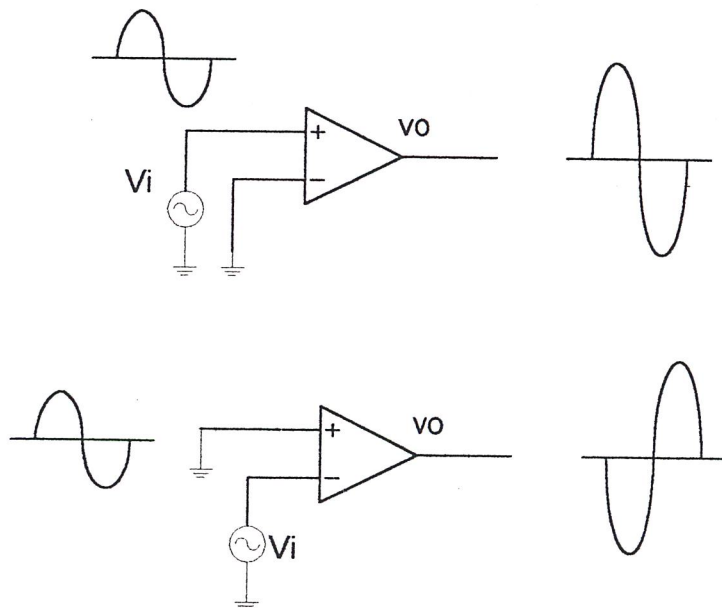
Op-Amp เป็นอุปกรณ์ที่มีอัตราขยายสูงและมี input impedance สูง output impedance ต่ำ การใช้งาน Op-Amp โดยสามารถทำการเปลี่ยนระดับแรงดัน การ Oscillator และการวงจร Filter รูปที่ 2.22 แสดงรูปแบบพื้นฐานของ Op-Amp ซึ่งจะมี 2 input และ 1 output การทำงานจะนำผลต่างของ 2 input นี้มาทำการขยายสัญญาณ



รูปที่ 2.22 Basic Op – Amp

Single – Ended Input

ผลที่ได้ของ Single – Ended Input คือเมื่อทำการต่อสัญญาณเข้าทาง input เดียวแล้วอีก input ต่อดึง ground จะได้สัญญาณดังรูปที่ 2.23

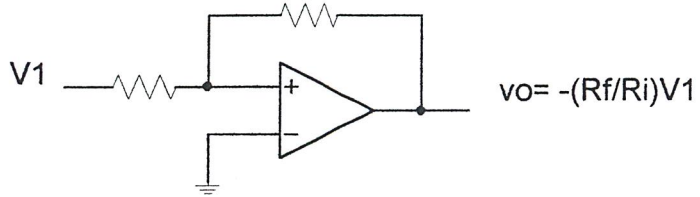


รูปที่ 2.23 Single –ended operation

Inverting Amplifier

เป็นวงจรที่ให้อัตราขยายคงที่ โดยจะให้ output ที่มีลักษณะสัญญาณที่กลับกันกับสัญญาณของ input

$$V_o = - [R_f/R_1]V_1 \tag{2.49}$$



รูปที่ 2.24 Inverting constant-gain multiplier

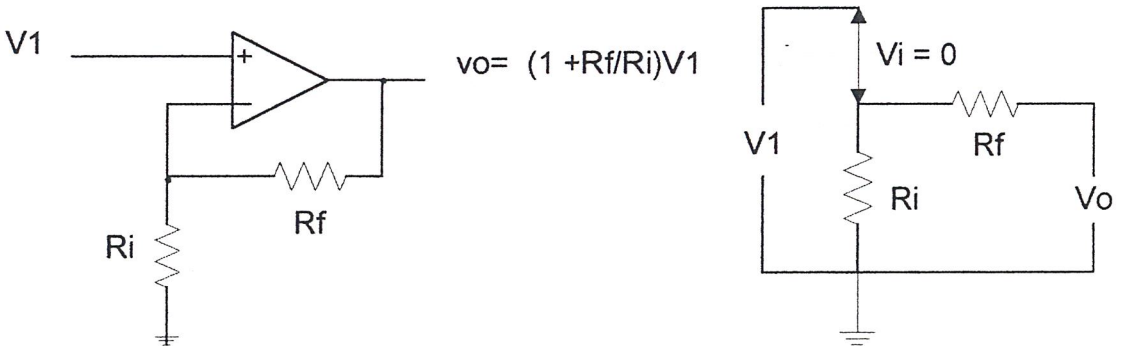
Noninverting Amplifier

เป็นการต่อวงจรตามรูปที่ 2.25 การใช้งานจะใช้ในรูปของ noninverting จะให้ความคงที่ของความถี่ได้ดีกว่าการคำนวณค่าอัตราขยายของวงจรจะสามารถหาได้จากสูตร

$$V_1 = [R_1/(R_1+R_f)]V_0 \tag{2.50}$$

ซึ่งก็คือ

$$\frac{V_o}{V_1} = \frac{R_1 + R_f}{R_1} = 1 + \frac{R_f}{R_1} \tag{2.51}$$



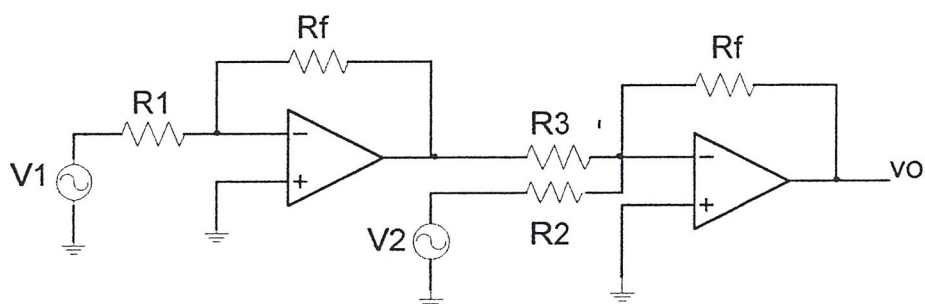
รูปที่ 2.25 Noninverting constant – gain multiplier

Voltage Subtraction

สัญญาณ 2 สัญญาณสามารถมาหักล้างกันได้ โดยสัญญาณหนึ่งมาหักล้างกับอีกสัญญาณหนึ่ง ตามรูปที่ 2.26 โดยใช้ Op-Amp 2 ตัวผลที่ได้ของ output จะเป็น

$$V_0 = -\left[\frac{R_f}{R_3} \left(-\frac{R_f}{R_1} V_1 \right) + \frac{R_f}{R_2} V_2 \right] \quad (2.52)$$

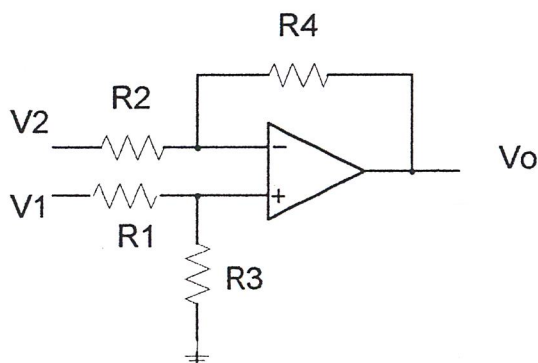
$$V_0 = -\left(\frac{R_f}{R_2} V_2 - \frac{R_f R_f}{R_3 R_1} V_1 \right) \quad (2.53)$$



รูปที่ 2.26 Op - Amp Applications voltage Subtraction

การต่อวงจรในรูปแบบอื่น ๆ ที่ทำการหักล้างสัญญาณ 2 สัญญาณตามรูปที่ 2.27... การต่อแบบนี้จะใช้ Op-Amp เพียงตัวเดียวและ output ที่ได้จะเป็น

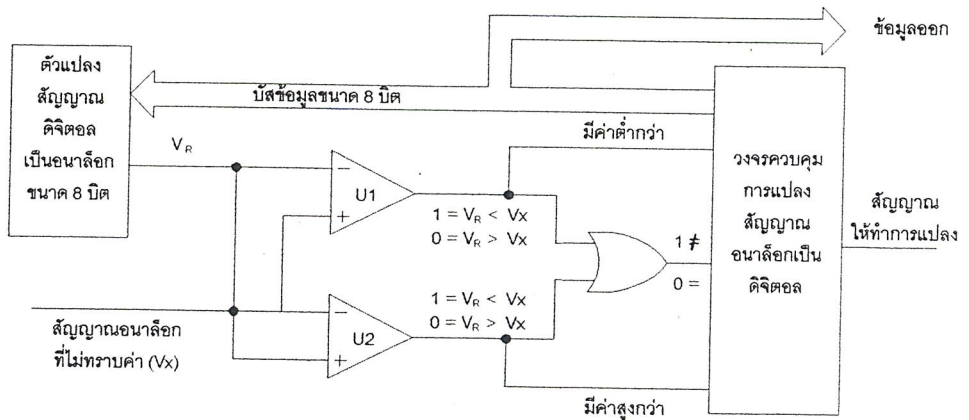
$$V_0 = \frac{R_3}{R_1 + R_3} \frac{R_2 + R_4}{R_2} V_1 - \frac{R_4}{R_2} V_2 \quad (2.54)$$



รูปที่ 2.27 Subtraction circuit only one Op - Amp

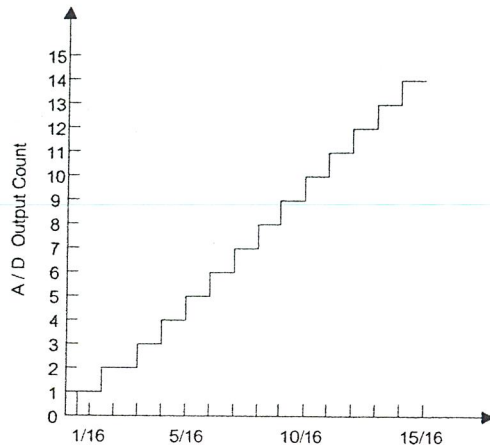
2.3 ส่วนแปลงอนาล็อกเป็นดิจิทัล (A/D)

ในการวัดสัญญาณต่างๆ ด้วยคอมพิวเตอร์นั้น โดยทั่วไปจะทำการประมวลผลข้อมูลด้วยซอฟต์แวร์ซึ่งต้องรับข้อมูลเป็นดิจิทัล ดังนั้นในการติดต่อกับอุปกรณ์ภายนอกที่เป็นสัญญาณอนาล็อกจึงจำเป็นต้องมีการเปลี่ยนแปลงสัญญาณอนาล็อกดังกล่าวให้เป็นสัญญาณดิจิทัลก่อนจะนำไปประมวลผล



รูปที่ 2.29 ตัวแปลงสัญญาณอนาล็อกเป็นดิจิทัล

จากรูปที่ 2.29 แสดงตัวแปลงสัญญาณอนาล็อกเป็นดิจิทัลที่ประกอบด้วย ตัวแปลงสัญญาณดิจิทัลเป็นอนาล็อก ตัวเปรียบเทียบระดับสัญญาณไฟฟ้า 2 ตัวและวงจรควบคุมในวงจรนี้ตัวแปลงสัญญาณดิจิทัลเป็นอนาล็อกจะสร้างสัญญาณอนาล็อกให้มีระดับศักดาเท่ากับสัญญาณอนาล็อกที่ไม่ทราบค่า (ที่ป้อนเข้า)และเมื่อระดับศักดาไฟฟ้าทั้ง 2 มีค่าเท่ากัน ค่าเอาต์พุตที่ออกมาจะเป็น 0 นั้นหมายความว่าค่าเลขฐานสองที่เราป้อนเข้าตัวแปลงสัญญาณดิจิทัลเป็นอนาล็อกจะทำให้มันสร้างสัญญาณอนาล็อกที่มีศักดาไฟฟ้าเท่ากับสัญญาณอนาล็อกที่ไม่ทราบค่า แต่ถ้าสัญญาณอนาล็อกที่ถูกสร้างจากตัวแปลงดิจิทัลเป็นสัญญาณอนาล็อกนี้ มีระดับขนาดของศักดาไฟฟ้ามากกว่าหรือน้อยกว่าสัญญาณอนาล็อกที่ไม่ทราบค่า 2 ตัวเปรียบเทียบระดับศักดาทั้ง 2 ตัวก็จะส่งสัญญาณไปบอกหน่วยควบคุมการแปลงสัญญาณอนาล็อกเป็นดิจิทัล ให้ทำการปรับค่าเลขฐานสอง ที่ป้อนเข้าตัวแปลงสัญญาณดิจิทัลเป็นอนาล็อกจนกว่าตัวแปลงดิจิทัลเป็นอนาล็อกนี้ จะสร้างสัญญาณอนาล็อกที่มีระดับศักดาไฟฟ้าเท่ากับสัญญาณอนาล็อกที่ไม่ทราบค่า



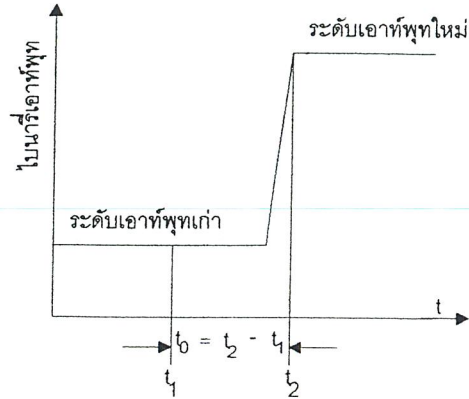
รูปที่ 2.30 คุณสมบัติของ A/D คอนเวอร์เตอร์แบบ 4 บิต

ส่วนไอซีที่ใช้ในการแปลงอนาล็อกเป็นดิจิทัลนั้น เลือกใช้ ADC 804 ซึ่งเป็นไอซีสำหรับการแปลงอนาล็อกเป็นดิจิทัล ที่ใช้หลักการ ชั้คเซตซีฟ แอปพริอิกซิเมชั่น ส่วนวงจรเชื่อมต่อกับส่วนประมวลผลนั้นใช้ Port 1 ของไมโครคอนโทรลเลอร์เป็นทางเข้าของข้อมูล

หลักการของตัวแปลงสัญญาณอนาล็อกเป็นสัญญาณดิจิทัล (A/D Converter)

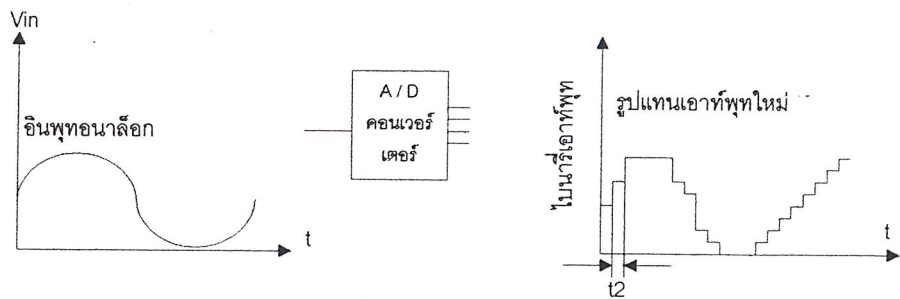
A/D คอนเวอร์เตอร์ มีสัญญาณอินพุท เป็นสัญญาณอนาล็อกและมีสัญญาณดิจิทัล เป็นสัญญาณเอาต์พุทจำนวน n บิต อินพุทของ A/D คอนเวอร์เตอร์เป็นสัญญาณอนาล็อกที่ผ่านออกมาจากวงจร sample and Hold ที่ทำหน้าที่รับสัญญาณอนาล็อกที่ต้องการแปลงเป็นสัญญาณดิจิทัลเข้ามาในวงจรแล้วนำเอาต์พุทไปต่อเชื่อมกับอินพุทของ A/D คอนเวอร์เตอร์ โดยการแปลงสัญญาณจาก A/D คอนเวอร์เตอร์ แต่ละครั้งวงจรนี้รับสัญญาณอนาล็อกแล้วส่งเข้าขาอินพุทของ A/D คอนเวอร์เตอร์ ในช่วงเวลาที่เท่ากับเวลา Sample ที่กำหนดจากความถี่ Sample ที่กำหนดในวงจร ดังนั้นในการควบคุมการทำงานของ A/D คอนเวอร์เตอร์ ในการแปลงสัญญาณจะมีค่าจำนวนข้อมูลเท่าใด

พิจารณาได้จากความถี่ Sample ภายในชุดวงจร Sample and Hold การทำงานของ A/D คอนเวอร์เตอร์นี้ ยังขึ้นอยู่กับผลของเวลาในการแปลงสัญญาณ (Conversion Time, T_c) ภายใน A/D คอนเวอร์เตอร์ซึ่งก็คือเวลาที่ใช้ในระหว่างที่อินพุทเข้ามาจนถึงการแสดงค่าระดับเอาต์พุทใหม่



รูปที่ 2.31 แสดงการตอบสนองของเวลาแปลงสัญญาณของ A / D คอนเวอร์เตอร์

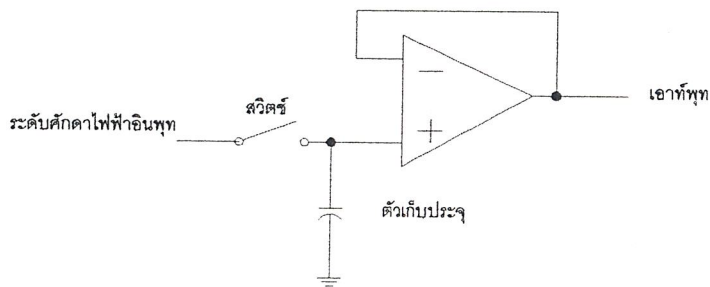
พิจารณาตามรูปที่ 2.31 อินพุทจากสัญญาณอนาล็อกที่เข้าใน A / D คอนเวอร์เตอร์จะอยู่ ณ เวลา t_1 และสัญญาณตอบสนองของอินพุทจะเกิดขึ้นจากผลต่างของเวลาทั้ง 2 คือเวลาการเปลี่ยนแปลงสัญญาณซึ่งเป็นเวลาที่ใช้จริงของ A / D คอนเวอร์เตอร์ ดังนั้นการกำหนดเวลา Sample จากความถี่ Sample ควรมีความมากกว่าเวลาในการแปลงสัญญาณ จึงจะได้ค่าดิจิทัลเอาท์พุทที่ถูกต้องและเพื่อให้ทราบถึงผลของการ Sample จนสัญญาณอินพุทอนาล็อกแปลงเป็นดิจิทัลเอาท์พุท พิจารณาจากกราฟ รูปที่ 2.32



รูปที่ 2.32 ขบวนการแปลงสัญญาณดิจิทัลด้วย A / D คอนเวอร์เตอร์

ทั้งตัวแปลงสัญญาณที่ปรับค่าอย่างต่อเนื่องและตัวแปลงสัญญาณที่ประมาณค่าผลสำเร็จจะเปลี่ยนค่าในแต่ละครั้งเมื่อมันได้รับคำสั่งจากไมโครโพรเซสเซอร์เท่านั้น ซึ่งก็คือตัวแปลงสัญญาณอนาล็อกเป็นดิจิทัลที่ประมาณค่าผลสำเร็จขนาด 8 บิต (ที่ใช้เวลา 1 ไมโครวินาทีในการทำงานแต่ละขั้นตอน จะใช้เวลาทั้งหมด 8 ไมโครวินาที ในการแปลงอนาล็อกเป็นดิจิทัล ในช่วงเวลา 8 ไมโครวินาทีนี้สัญญาณอนาล็อกที่ไม่ทราบค่าที่ป้อนเข้าตัวแปลง

สัญญาณนั้นจะต้องมีระดับศักดาไฟฟ้าที่คงที่ มิฉะนั้นเอาท์พุทที่ได้อาจไม่ถูกต้องเพื่อป้องกันการผิดพลาดที่อาจเกิดขึ้นจึงได้มีการนำวงจร Sample and Hold มาใช้ โดยวงจรมีระดับศักดาไฟฟ้าของสัญญาณที่รับเข้ามาในตัวเก็บประจุก่อนที่ตัวแปลงสัญญาณจะเริ่มทำงานเมื่อสวิตช์ถูกปิด ตัวเก็บประจุจะได้รับประจุนมีระดับไฟฟ้าตกรวมเท่ากับศักดาไฟฟ้าที่ป้อนเข้ามา และเมื่อสวิตช์ถูกเปิดระดับศักดาไฟฟ้าในตัวเก็บประจุจะคงอยู่ที่ค่านี้นานกว่าสวิตช์จะถูกปิดอีกครั้ง โดยปกติสวิตช์แบบ FET (Field effect transistor) ที่ทำงานได้ด้วยความเร็วสูงจะถูกนำมาใช้ในวงจรมี

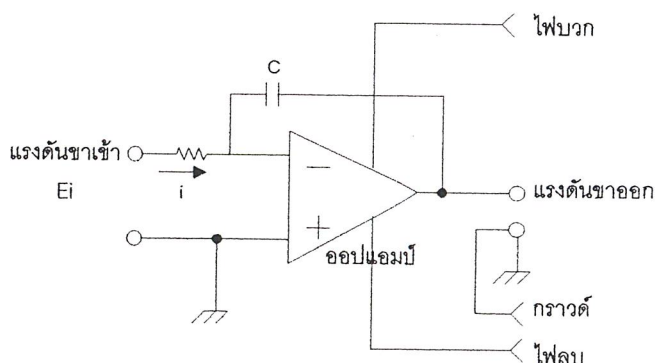


รูปที่ 2.33

วงจร A/D คอนเวอร์เตอร์ (Analog To Digital Converter) ที่นิยมมี 3 แบบคือ

1. แบบสโลปคู่ (Dual Slop)
2. แบบแปลงสัญญาณเป็นความถี่ (V To F Converter)
3. แบบประมาณทีละบิต (Successive Approximation)

สโลปคู่ (Dual Slop) เป็นแบบที่ง่ายที่สุด ไม่จำเป็นต้องใช้อุปกรณ์ที่มีคุณภาพมากนักก็สามารถแปลงสัญญาณได้อย่างแม่นยำ แต่มีข้อเสียตรงที่ใช้เวลาในการแปลงสัญญาณมากเกินไป จึงไม่เหมาะในการวัดแรงดันในช่วงเวลาสั้น ๆ

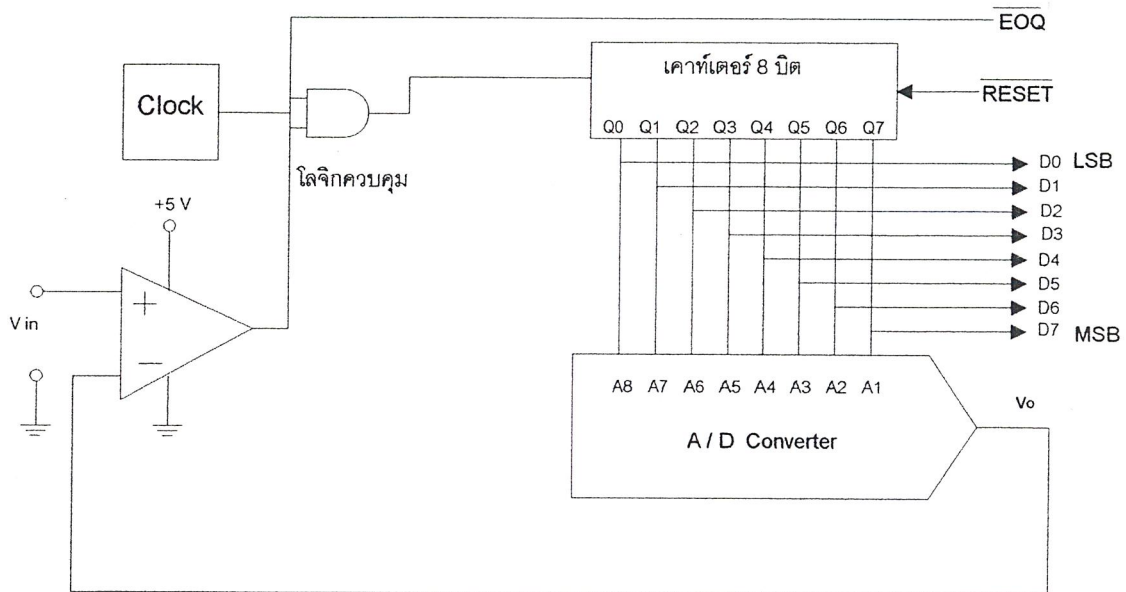


รูปที่ 2.34 การทำงานของวงจรอินทิเกรเตอร์

พิจารณาจากรูปที่ 2.34 ซึ่งเป็นวงจรอินทิเคเตอร์แบบพื้นฐาน ประกอบด้วยอปแอมป์ ทำหน้าที่เป็นวงจรขยายสัญญาณแตกต่าง

แรมป์เอาต์พุต (Ramp Output) ของ A / Dคอนเวอร์เตอร์

เป็น A / D คอนเวอร์เตอร์ที่เข้าใจการทำงานง่ายที่สุด โดยแสดงโครงสร้างการทำงานดังรูปที่ 2.35

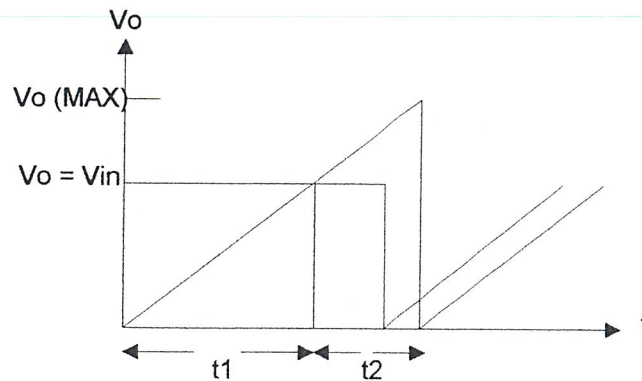


รูปที่ 2.35 ตัวแปลงสัญญาณ A / D คอนเวอร์เตอร์แบบแรมป์

สามารถอธิบายการทำงานหลักการได้ดังนี้

1. คอมพิวเตอร์ส่งสัญญาณรีเซ็ตไปที่เคาท์เตอร์ เพื่อให้สามารถรับสัญญาณอินพุตได้โดยขณะนั้นผลเอาต์พุตของ A / D คอนเวอร์เตอร์ (V_0) มีค่าอยู่ที่ระดับต่ำสุด
2. การทำงานของวงจรมีการเปลี่ยนแปลงความถี่ของสัญญาณนาฬิกาซึ่งสัญญาณอินพุตถูกนำมา And กับสัญญาณนาฬิกาแล้วจึงส่งไปยังเคาท์เตอร์เป็นผลทำให้ V_0 แสดงระดับแรงดันที่สูงขึ้นทีละ 1 LSB
3. ณ จุดเดียวกันนั้นตัวเคาท์เตอร์จะนับเพิ่มขึ้นจนกระทั่งเอาต์พุตของ A / D คอนเวอร์เตอร์สูงกว่าค่าแรงดันอินพุต V_{in} เมื่อถึงจุดนี้ของคอมพิวเตอร์จะแสดงค่าไปจนกระทั่งเป็น 0 โวลต์เมื่อถึง 0 โวลต์ จะมีสัญญาณนาฬิกาหยุดการนับที่จุดซึ่ง V_0 เริ่มมากกว่า V_{in} ขา EOC จะลดระดับเป็นระดับต่ำ และส่งสัญญาณไปให้คอมพิวเตอร์วางข้อมูล ขณะนี้พร้อมที่จะอ่านได้แล้ว ซึ่งข้อมูลที่ส่งไปยังคอมพิวเตอร์เป็นสัญญาณดิจิทัลโดยผ่านทางอินพุตพอร์ท

4. หลังจากคอมพิวเตอร์อ่านข้อมูลเสร็จ คอมพิวเตอร์ก็จะส่งสัญญาณรีเซ็ตมาที่ A/D คอนเวอร์เตอร์ หลังจากนั้นก็เริ่มทำกระบวนการเบบเดิมอีกครั้ง



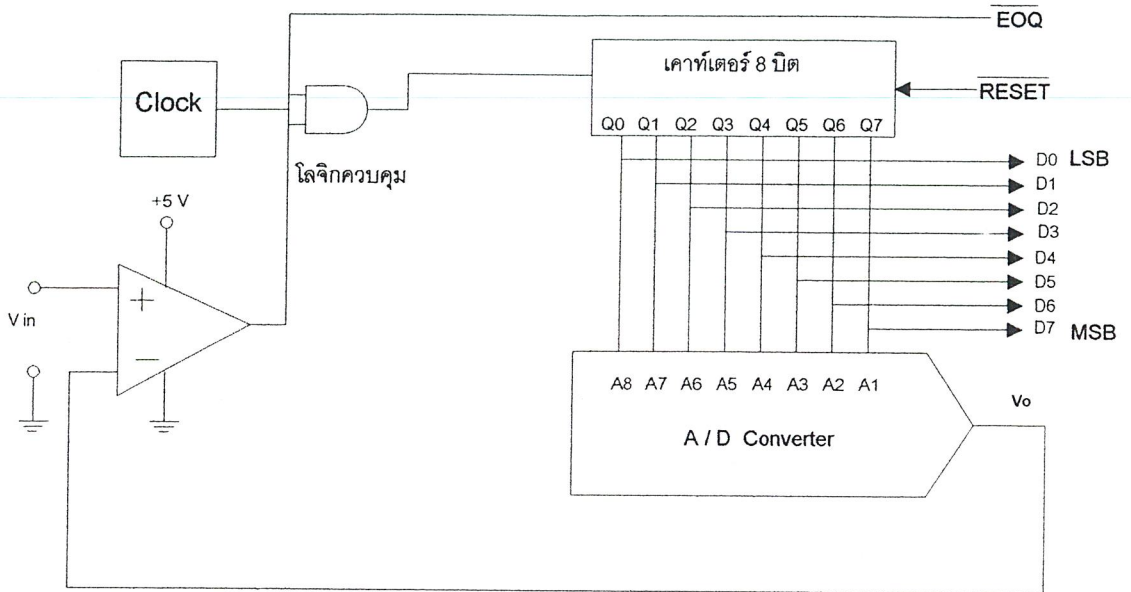
รูปที่ 2.36 กราฟเอาต์พุตของส่วน D/A ของ A/D คอนเวอร์เตอร์

กราฟนี้แสดงการทำงานของแรมป์ โดย t_1 แทนเวลาที่นับใช้เพื่อแรมป์เอาต์พุตของ D/A คอนเวอร์เตอร์ที่เริ่มจะเลย V_{in} และ t_2 แทนเวลาระหว่างที่คอมพิวเตอร์อ่านข้อมูลและส่งสัญญาณรีเซ็ตกลับมาให้ A/D คอนเวอร์เตอร์ เริ่มต้นกระบวนการเดิมอีกครั้งหนึ่ง ดังนั้นในการเก็บข้อมูล 1 ค่าจะต้องใช้เวลาในการแปลงสัญญาณ 2 ช่วงด้วยกัน คือ เวลา t_1 ถูกกำหนดโดยอ่านข้อมูลและส่งสัญญาณรีเซ็ตกลับ

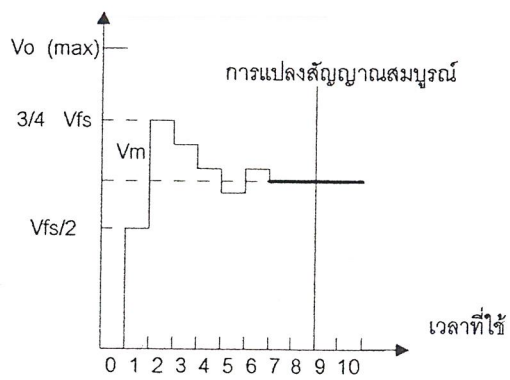
ซัสเซสซีฟแอสปรัอกซิเมชัน (Successive Approximation) A/D คอนเวอร์เตอร์

SAR เป็นตัวเลขหลักของ A/D คอนเวอร์เตอร์ ซึ่งลักษณะใกล้เคียงกับแบบแรมป์ โดยโครงสร้างการทำงานดังรูป 2.37 ซึ่งเป็นการแสดงบล็อกไดอะแกรมสำหรับ A/D คอนเวอร์เตอร์ แบบ SAR

SAR คอนเวอร์เตอร์ ต้องใช้ D/A คอนเวอร์เตอร์ และคอมพาราเตอร์ในการทำงานเหมือนกัน ส่วนที่แตกต่างได้แก่ ส่วนของเคาท์เตอร์ วิธีแรมป์ทำงานโดยอาศัยการนับขึ้นในลำดับไบนารีจนกระทั่ง V_{in} การทำงาน SAR คอนเวอร์เตอร์ จะอธิบายโดยใช้รูป 2.34 และ 2.35 อธิบาย



รูปที่ 2.37 คอนเวอร์เตอร์ซึ่งใช้เทคนิค SAR



รูปที่ 2.38 กราฟของส่วน D/A ระหว่างการแปลงสัญญาณ

1. การเริ่มต้นของการเปลี่ยนแปลงจากอนาล็อกเป็นดิจิทัล พิจารณาจากสัญญาณนาฬิกาที่ส่งไปยังระบบควบคุมลอจิก โดยเอาต์พุตของระบบควบคุมลอจิกจะส่งพัลส์ลูกหนึ่งให้กับ SAR เพื่อเซตค่าของ MSB และลบค่าเอาต์พุตที่ยังเหลืออยู่ ซึ่งจะทำให้ค่าของ V_0 เท่ากับค่า $V_{fs}/2$ ซึ่งจะน้อยกว่าค่า V_{in} และเอาต์พุตของคอมพาราเตอร์จะยังคงอยู่ในระดับ “1” ระบบควบคุมลอจิกจะตรวจสอบว่า ถ้าคอมพาราเตอร์ยังคงเป็น “1” อยู่จะส่งสัญญาณไปที่ SAR ซึ่งจะล้างค่าของ MSB เป็นลอจิก “1” ที่ Q7
2. เมื่อมีสัญญาณนาฬิกาถูกตัดไป MSB ตัวใหม่ ซึ่งอยู่ที่ Q6 ของ SAR จะถูกเซตค่าเป็น “1” ดังรูปที่ 2.37 เอาต์พุตของ D/A จะเกินค่าของ V_{in} ทำให้เอาต์พุตของคอมพาราเตอร์เปลี่ยนระดับ ระบบควบคุมลอจิกตรวจสอบทราบก็จะไม่ทำการล้างค่า Q6 นั้น คือค่าที่ล้างที่ตำแหน่ง Q6 มีค่าเท่ากับ “0”
3. สัญญาณนาฬิกาถูกถัดมา Q6 จะถูกเคลียร์และ Q5 จะถูกเซต จากการเปรียบเทียบได้ เอาต์พุตของคอมพาราเตอร์เป็น “1” ส่วนควบคุมลอจิกก็จะไม่ทำการล้างค่า Q5 ไว้เพราะยังสูงกว่าค่าของ v_{in} โดยปกติแล้ววงจรจะทดสอบเอาต์พุต โดยเริ่มต้นที่ MSB ของ SAR ถ้าเอาต์พุตของ D/A คอนเวอร์เตอร์ เกินค่า V_{in} แล้วบิตนั้นจะล้างค่าลอจิก “1” ไว้หลังจากการตรวจสอบหมดแล้ว ไบนารีเอาต์พุตจะเป็นอัตราส่วนกับ V_{in} จะเห็นว่า A/D แบบนี้ดีกว่าแบบแรมปี คือเวลาในการแปลงสัญญาณเป็นสัดส่วนกับจำนวนบิตของเคาท์เตอร์แบบ SAR แต่ SAR A/D คอนเวอร์เตอร์มีข้อเสียแบบเดียวกับแรมปี คือข้อผิดพลาดทางอัตราขยายออฟเซต และความไม่เป็นเชิงเส้นที่เกิดขึ้นในส่วน D/A

บทที่ 3

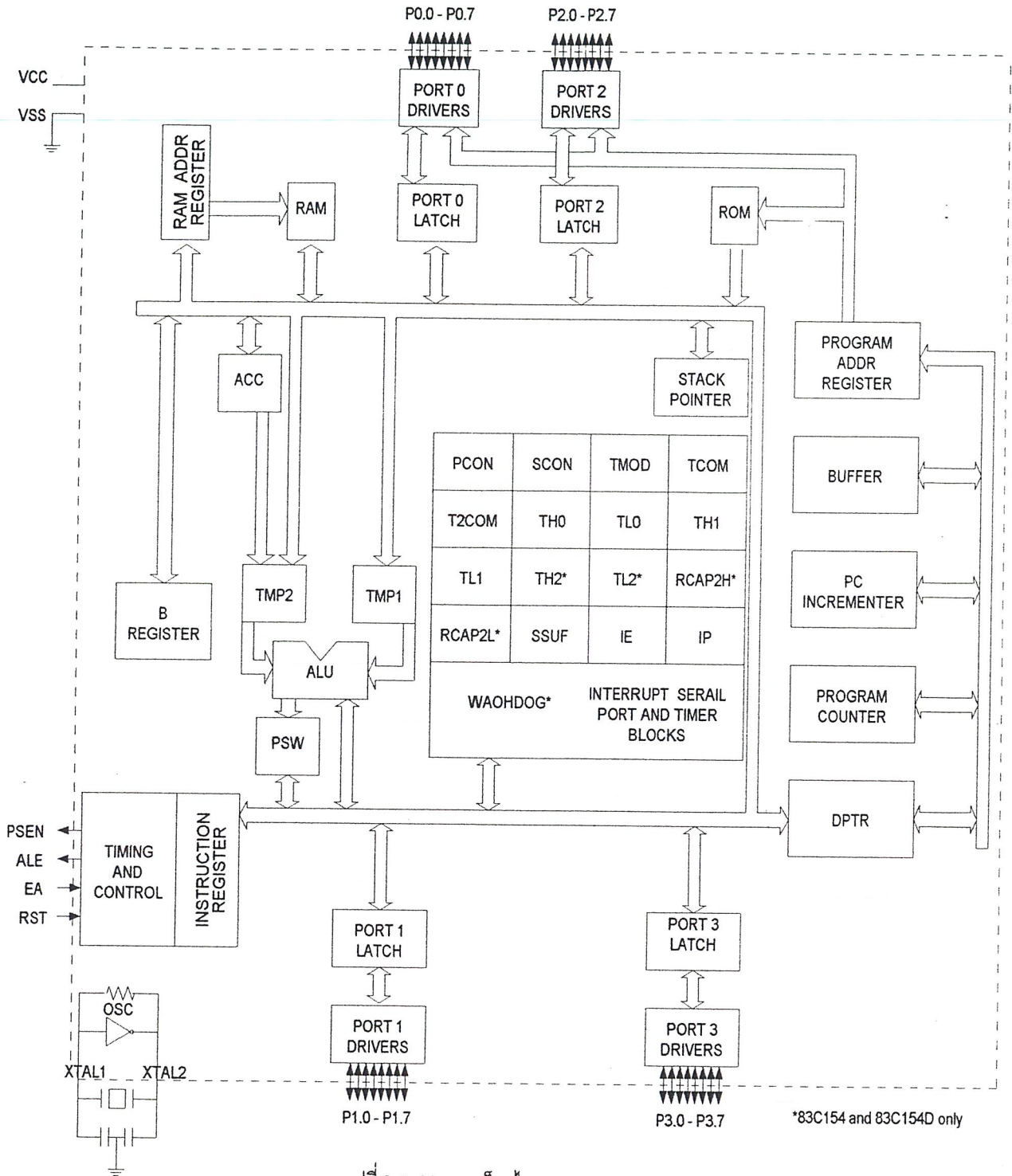
โครงสร้างของ MCS-51

3.1 คุณสมบัติของไมโครคอนโทรลเลอร์ตระกูล MCS-51

- ต้องการแหล่งจ่ายไฟ + 5 V ชุดเดียว
- มีหน่วยความจำโปรแกรม (Program Memory) ขนาด 4 กิโลไบต์สำหรับเบอร์ 8051 และ 8031 สำหรับเบอร์ 8052 มีหน่วยความจำถึง 8 กิโลไบต์
- มีหน่วยความจำสำหรับเก็บข้อมูล (Data Memory) ขนาด 128 ไบต์ สำหรับเบอร์ 8052 ขึ้นไปมี 256 ไบต์
- หน่วยความจำสำหรับเก็บโปรแกรมและข้อมูลแยกจากกันอย่างละ 64 กิโลไบต์
- มีไทม์เมอร์ เคนต์เตอร์ ขนาด 16 บิต 2 ชุด (สำหรับ 8052 มี 3 ชุด) ทำงานได้ 4 โหมค
- รับอินเตอร์รัพท์ได้ 6 แหล่ง 5 เวกเตอร์ สำหรับเบอร์ 8052 ขึ้นไปมี 8 แหล่ง 6 เวกเตอร์
- มีพอร์ตรับ-ส่ง ข้อมูลอนุกรม (UART) 2 พอร์ต แบบ Full Duplex เลือกgrupได้ 4 โหมค
- มีคำสั่งในการทำ AND , OR หรือ Complement ได้ทั้ง 8 บิตและ 1 บิต

3.2 โครงสร้างภายในของ 8051

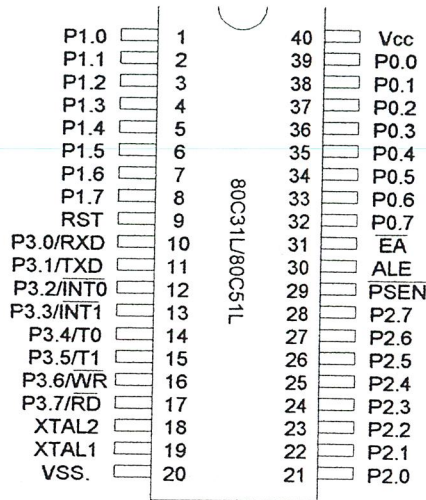
MCS-51 ใช้เทคโนโลยีในการผลิตเป็นแบบ NMOS และ CMOS เบอร์ 8032 และ 8052 จะมี ROM BASIC อยู่ภายใน จึงสะดวกสำหรับโปรแกรมเมอร์ ที่จะเขียนโปรแกรมด้วยภาษาเบสิก โครงสร้างภายในสำหรับเบอร์ 8051 แสดงดังรูป 3.1



*83C154 and 83C154D only

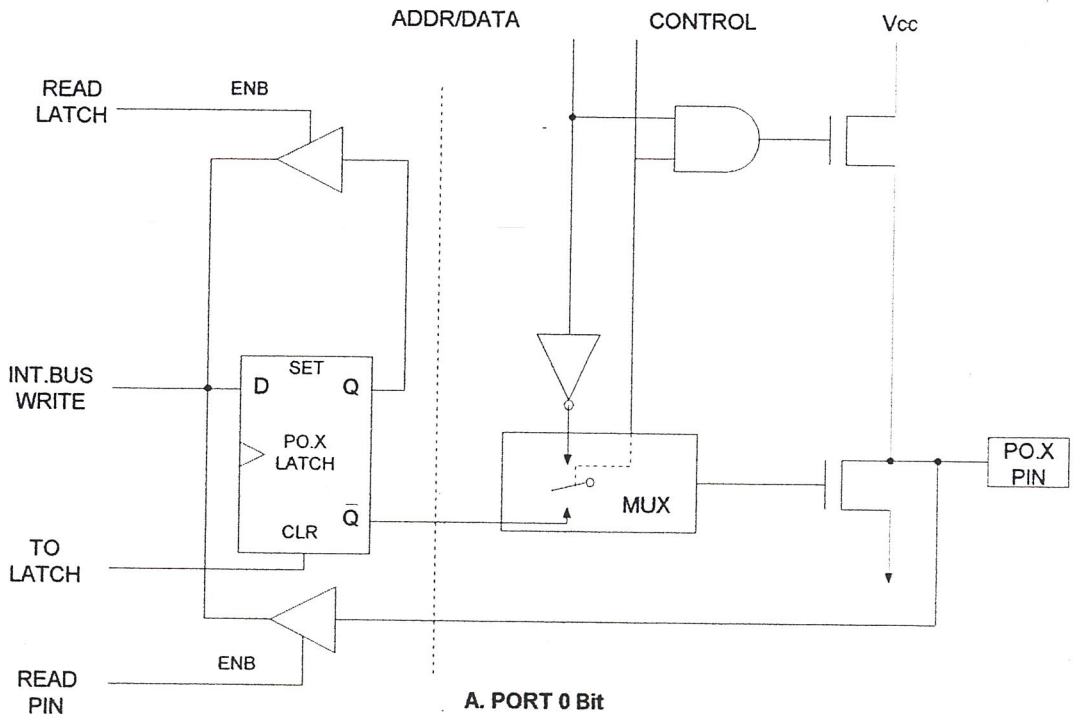
รูปที่ 3.1 8051 บล็อกไดอะแกรมของ MCS - 51

3.3 พอร์ตของ 8051



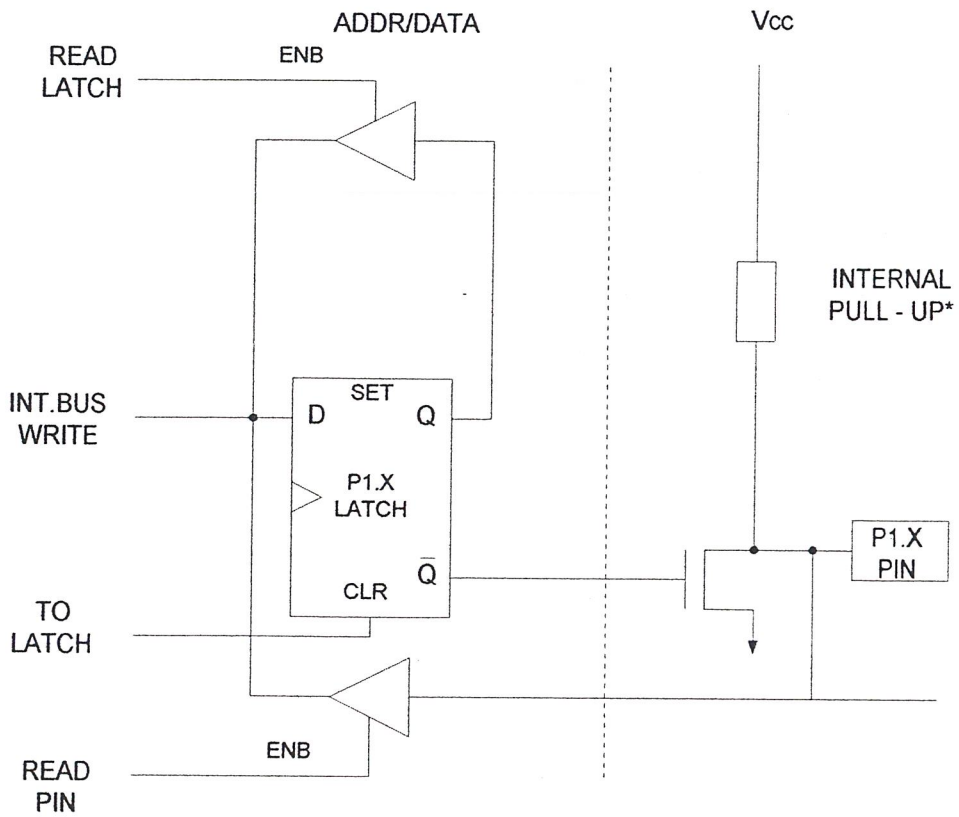
รูปที่ 3.2 การจัดวางขาของ 8051

- Vcc (ขา 40) ต่อกับ + 5 V
- Vss (ขา 20) เป็นขา GND
- พอร์ต 0 (ขา 32-39) มีทั้งหมด 8 บิต คือ (P0.7-0.0) มีโครงสร้างแบบ Open-Drain ,
- Bi-directional แสดงดังรูป 3.3



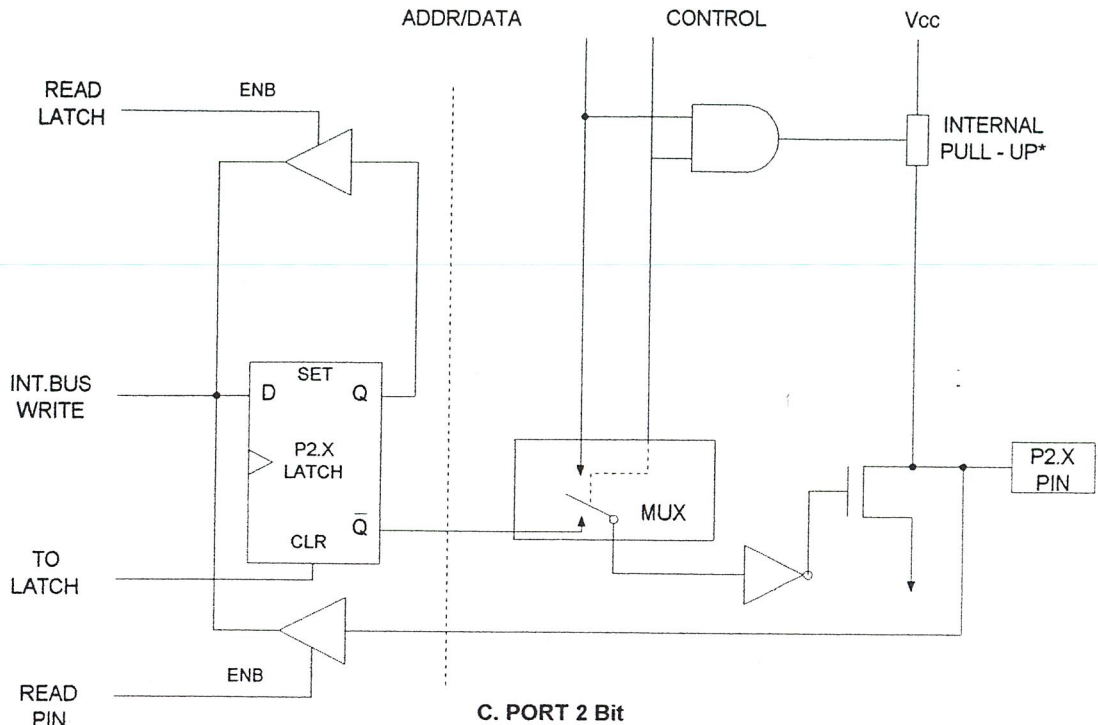
รูป 3.3 แสดงโครงสร้างพอร์ต 0 (บิต)

- พอร์ต 0 (ขา 32-39) มีทั้งหมด 8 บิต คือ (P0.7-P0.0) ใช้งานได้ 2 หน้าที่คือแอดเดรส บัส และค่าตัวบัส เมื่อต้องการติดต่อกับหน่วยความจำภายนอก หรือเป็นไอโอพอร์ต ถ้า ต้องการให้ทำงานเป็นอินพุทพอร์ต ต้องส่งลอจิก “1” ไปยังพอร์ตนี้ จะมีผลให้ \bar{Q} ของ D-FF เป็น “0” ทำให้ FET ตัวล่างมีสถานะ OFF สัญญาณที่ใช้อ่านอินพุทพอร์ตแลตช์ โดยส่งสัญญาณ READ LATCH ไปกระตุ้นที่ TRI-State Buffer ตัวบนและการอ่าน Port (pin) จะใช้สัญญาณ Read (pin)
- พอร์ต 1 (ขา 1-8) มีทั้งหมด 8 บิต คือ (P1.0-P1.7) มีโครงสร้างคล้ายพอร์ต 0 แต่จะใช้ ความต้านทานภายในพูลอัพแทน Internal Pull up Register มีโครงสร้างดังรูป 3.4
- พอร์ต 2 (ขา 21-28) มีทั้งหมด 8 บิต คือขา (P2.7-P2.0) มีโครงสร้างคล้าย พอร์ต 0 โดย มี FET ตัวล่างตัวเดียว ส่วนด้านบนใช้ความต้านทานพูลอัพแทน (Internal Pull up) พอร์ตนี้ทำงาน 2 หน้าที่ คือสามารถใช้เป็นแอดเดรสบัสขนาด 8 บิต (A15-A8) และเป็น ไอโอพอร์ตใช้งานทั่วไป เมื่อจะใช้งานเป็นอินพุทพอร์ตต้องส่งลอจิก “1” มาที่พอร์ตนี้ก่อน เพื่อบังคับให้ FET อยู่ในสถานะ Off ดังแสดงในรูป 3.5



B. PORT 1Bit

รูปที่ 3.4 โครงสร้างของพอร์ต 1 (บิต)



รูป 3.5 โครงสร้างของพอร์ต 2 (บิต)

- พอร์ต 3 (ขา 10-17) มีทั้งหมด 8 บิต คือขา (P3.7-P3.0) มีโครงสร้างคล้ายพอร์ต 1 ทำงานได้ 2 หน้าที่ คือเป็นไอโอพอร์ต ถ้าจะโปรแกรมให้เป็นอินพุตพอร์ตต้องส่งลอจิก “ 1 “ มาที่พอร์ตนี้ก่อน และอีกหน้าที่หนึ่งก็คือใช้ส่งสัญญาณควบคุมออกมา และรับสัญญาณเข้าไป สัญญาณต่างๆมีดังนี้

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

P3.1/TXD (Serial Output Port) เป็นขาที่ใช้ส่งข้อมูลแบบอนุกรม (UART)

P3.2/INT0 (External Interrupt 0) ใช้รับสัญญาณการขัดจังหวะจากภายนอก เบอร์ 0

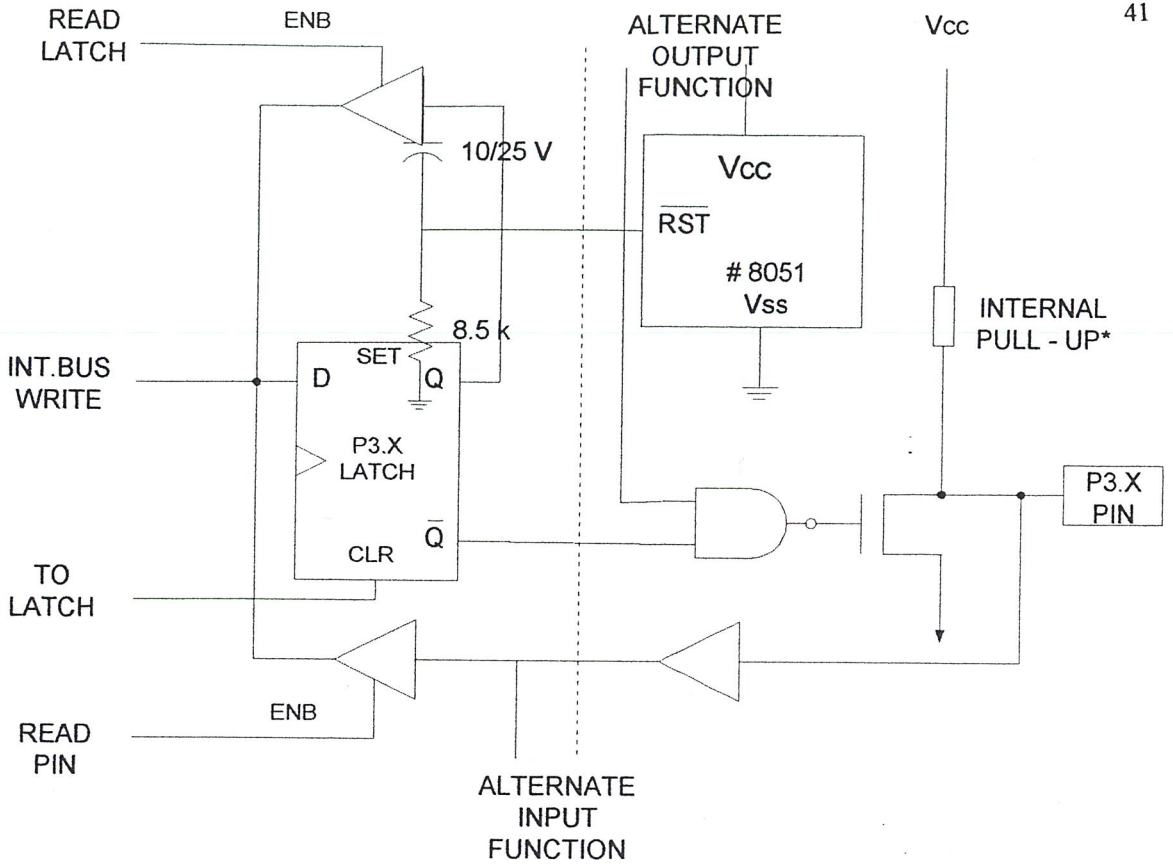
P3.3/INT1 (External Interrupt 1) ใช้รับสัญญาณการขัดจังหวะจากภายนอก เบอร์ 1

P3.4/TO (Counter 0 External Input) ขารับสัญญาณพัลส์อินพุตเข้าไปยังวงจร Counter 0 (เป็นอินพุตโหมดเคาน์เตอร์)

P3.5/T1 (Counter 1 External Input) ขารับสัญญาณพัลส์อินพุตเข้าไปยังวงจร Counter 1 (เป็นอินพุตโหมดเคาน์เตอร์)

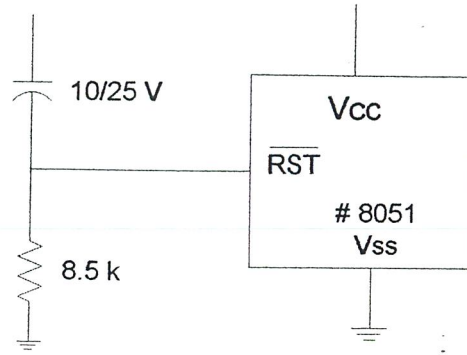
P3.6/WR (External Data Memory Write Strobe) ขาสัญญาณควบคุมการเขียนข้อมูลลงหน่วยความจำข้อมูลภายนอก

P3.6/RD (External Data Memory Read Strobe) ขาสัญญาณควบคุมการอ่านข้อมูลจากหน่วยความจำภายนอก



รูป 3.6 โครงสร้างของพอร์ต 3 (บิต)

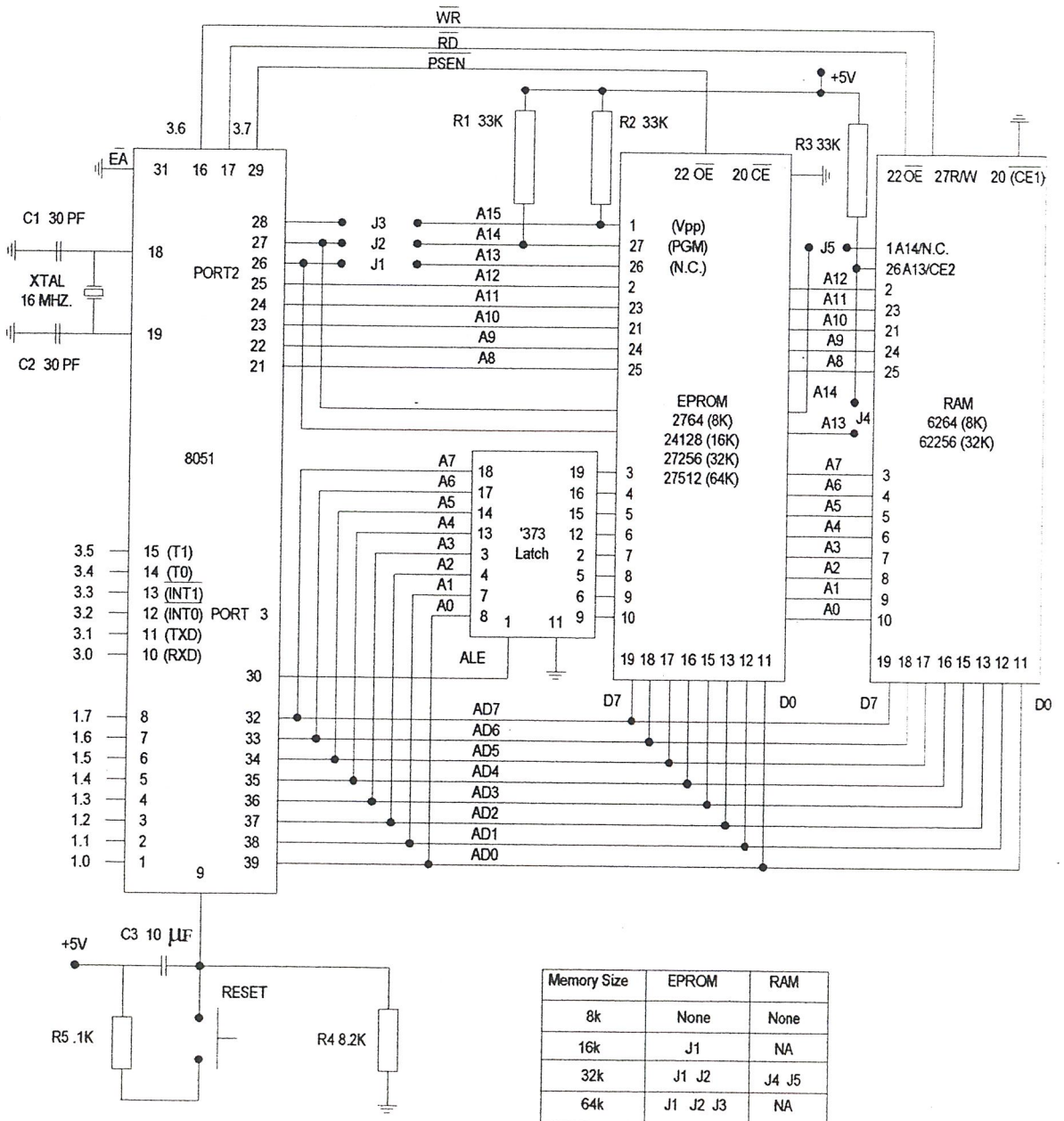
- ALE (ขา 30) เป็นขาส่งสไตรบสำหรับใช้ในการแลตช์แอดเดรสไบต์ต่ำ (A7-A0) ที่ส่งออก
มาจากพอร์ต 0 สัญญาณนี้จะแอดทิฟทุกๆ 2 ครั้ง ใน 1 แมกซ์ซินไซเคิล
- PSEN (ขา 29) เป็นขาสไตรบที่ใช้สำหรับอ่านข้อมูลจาก Program Memory ภายนอก สัญญาณนี้จะส่งออกมา 2 ครั้งในแต่ละแมกซ์ซินไซเคิล แต่ถ้าเป็นการอ่าน Internal Program Memory จะไม่มีสัญญาณออกที่ขานี้
- EA (ขา 30) ใช้เลือกหน่วยความจำโปรแกรมภายนอก
บิต "0" จะอ่านโปรแกรมจากภายนอกชิพ
บิต "1" จะอ่านโปรแกรมจากภายในชิพ
- RST (ขา 9) ขารีเซต จะรีเซ็ตได้ก็ต่อเมื่อลอจิก '1' เข้าที่ขานี้ นานอย่างน้อย 2 แมกซ์ซินไซเคิล
- XTAL1 (ขา 19) ใช้ต่อคริสตอลภายนอกโดยเป็นอินพุตเข้าสู่วงจรรอสซิทเลเตอร์ภายใน
- XTAL2 (ขา 18) ใช้ต่อคริสตอลภายนอกโดยเป็นเอาต์พุตของวงจรรอสซิทเลเตอร์ภายใน



รูป 3.7 การต่อขารี่เซ็ทให้กับ 8051

3.4 การต่อหน่วยความจำ Program Memory และ Data Memory

การต่อหน่วยความจำแสดงคังรูป 3.8



Memory Size	EPROM	RAM
8k	None	None
16k	J1	NA
32k	J1 J2	J4 J5
64k	J1 J2 J3	NA

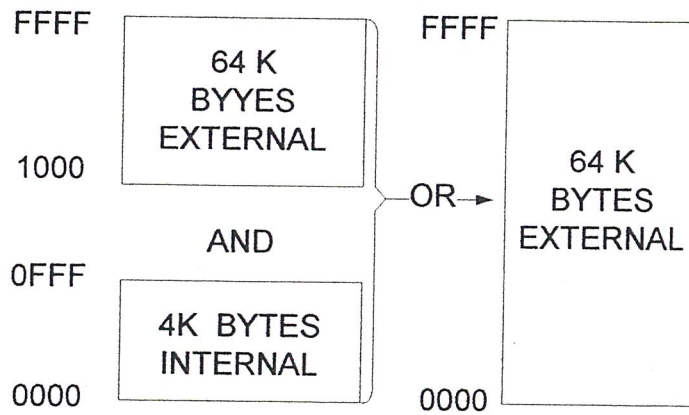
3.5 การแบ่งประเภทของหน่วยความจำ

หน่วยความจำที่ใช้กับ MCS-51 มีอยู่ด้วยกัน 2 ชนิด

หน่วยความจำสำหรับเก็บโปรแกรม (Program Memory)

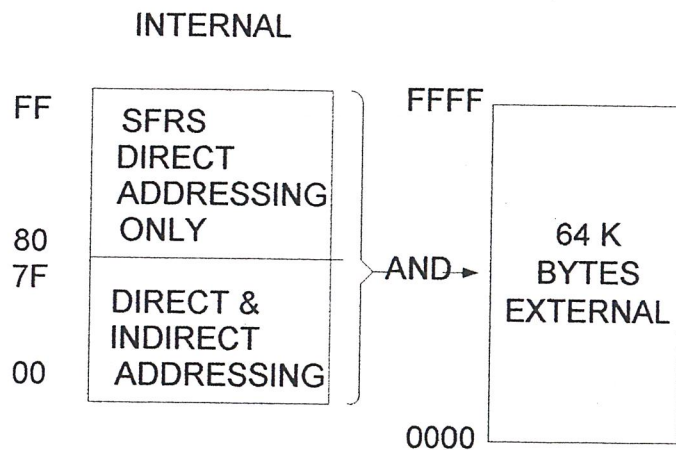
-หน่วยความจำสำหรับเก็บข้อมูล (Data Memory)

หน่วยความจำสำหรับเก็บโปรแกรม เป็นหน่วยความจำที่ใช้เก็บ โปรแกรมสั่งงาน
 บรรจุอยู่ในชิพ 8051 ส่วนที่เป็น Program Memory ก็คือ ROM ขนาด 4 กิโลไบต์นั่นเอง แสดง
 ดังรูป 3.9



รูป 3.9 ผังแสดงหน่วยความจำสำหรับเก็บโปรแกรมสำหรับเบอร์ 8051

Data Memory (RAM) แบ่งเป็น 2 ส่วนคือ หน่วยความจำข้อมูลภายในชิพมีเพียง
 128 ไบต์ และหน่วยความจำข้อมูลภายนอกชิพมีความจุ 64 กิโลไบต์ แสดงดังรูป 3.10



รูป 3.10 ผังแสดงหน่วยความจำสำหรับ Data Memory เบอร์ 8051

3.6 ชุดคำสั่งของ MCS-51

ชุดคำสั่งของ 8051 มีทั้งหมด 256 คำสั่ง วิธีการแปลงภาษาแอสเซมบลีให้เป็นภาษาเครื่อง สามารถดำเนินการได้ 2 แบบ คือ การทำด้วยมือและการโปรแกรมแอสเซมบลี เช่น SXA-51 , Cross-16 ฯลฯ โดยจะต้องเขียนโปรแกรมแล้วเก็บไว้ใน file โดยตั้งชื่อให้มีนามสกุลเป็นจุด ASM โปรแกรมที่ใช้พิมพ์อาจจะใช้โปรแกรม Editor การเขียนโปรแกรมภาษาแอสเซมบลีจะใช้ รหัสช่วยจำ (Memory Code) ซึ่งประกอบด้วย Operation Code , Operand และ Comment

1. Operation Code (ตัวดำเนินการ) ประกอบด้วย 2 ถึง 4 อักขร เช่น MOV, MOVX, ADD ฯลฯ

2. Operand (ตัวดำเนินการ) เป็นชุดอักขรที่บอกการดำเนินงาน อาจจะเป็นชุดเดียว หรือ 2 ชุดก็ได้ ถ้าเป็น 2 ชุด จะคั่นด้วยเครื่องหมายคอมมา (,) เช่น A,#20h เมื่อเขียน

ให้เต็มรูปแบบของภาษาแอสเซมบลีจะได้ดังนี้ MOV A, #20h หมายความว่าให้นำค่า 20 h เข้า รีจิสเตอร์ A โดยที่ A หมายถึง Distination

#20h หมายถึง Source

โดยที่ Source จะต้องอยู่ทางขวามือ และ Distination (ข้อมูลปลายทาง) จะอยู่ทางซ้ายมือ

3. Comment เป็นส่วนขยายความหมาย ของชุดรหัสคำสั่ง จะนำหน้าด้วยเครื่องหมาย “ ; ” ในส่วนของ Comment จะมีหรือไม่มีก็ได้

3.7 การสื่อสารข้อมูลแบบอนุกรม

การสื่อสารข้อมูลแบบอนุกรม

การส่งข้อมูลจะส่งแบบอนุกรม คือข้อมูลจะถูกทำการถ่ายโอนหรือส่งออกทีละบิต ระหว่างจุดส่งกับจุดรับ ตัวกลางการสื่อสารต้องการเพียงช่องเดียว หรือสายเพียงคู่เดียวในการส่งข้อมูล ส่วนสายสัญญาณที่เหลือจะเป็นสายส่งสัญญาณควบคุม และสายกราวด์ โดยลักษณะของการส่งแบบอนุกรมนั้น จะสามารถส่งได้ในระยะสั้นๆ จนถึงระยะทางเป็นไมล์ อัตราความเร็วของข้อมูลที่ใช้นั้นขึ้นอยู่กับระยะทางตั้งแต่ 0 ถึง 2 ล้าน บิตต่อวินาที มีการใช้มาตรฐานของ EIA RS-232C คือมีระดับสัญญาณไฟฟ้าขนาด 12 โวลต์ หรืออาจจะใช้มาตรฐาน 20 มิลลิแอมป์เคอร์เรนต์รูป หรืออาจจะใช้ระดับสัญญาณของ TTL ก็ได้ ซึ่งค่าใช้จ่ายก็น้อยกว่าการส่งข้อมูลแบบขนาน รูปแบบของการติดต่อสื่อสารแบบอนุกรมอาจเป็นไปได้ทั้ง 3 รูปแบบ คือแบบซิมเพล็กซ์ (simplex) คือส่งข้อมูลได้ทางเดียวเท่านั้น บางครั้งก็เรียกว่า การส่งทิศทางเดียว (Unidirection Data Bus) แบบที่สองคือการส่งแบบฮาร์ฟดูเพล็กซ์ (Half Duplex) ซึ่งข้อมูลสามารถส่งได้ทั้งสองสถานี แต่จะต้องผลัดกันส่ง และผลัดกันรับ จะทำการส่งและรับพร้อมกันไม่ได้ แบบที่สามคือแบบฟูลดูเพล็กซ์ (Full Duplex) ทั้งสองสถานีจะสามารถทำการรับ และส่งได้ในเวลาเดียวกัน ความเร็วในการถ่ายโอนข้อมูลแบบอนุกรมนี้นี้ จะมีหน่วยวัดเป็นบิตต่อวินาที (BPS : Bit Per Second) หน่วยที่แสดงถึงการเปลี่ยนแปลงของสัญญาณใน 1 วินาที หรือเรียกว่า บอดเรต (Baud Rate) หรืออัตราบอด โดยที่อัตราบิตจะเท่ากับอัตราบอดคูณกับบิต ในหนึ่งบอดการส่งข้อมูลจะส่งแบบอะซิงโครนัส ซึ่งประกอบด้วยบิตเริ่มต้นหรือบิตสตาร์ท (Start Bit) และบิตสิ้นสุด (Stop Bit) ข้อมูลถูกส่งออกไปอย่างไม่มีกำหนดเวลาแน่นอน

พอร์ตสื่อสารแบบอนุกรม

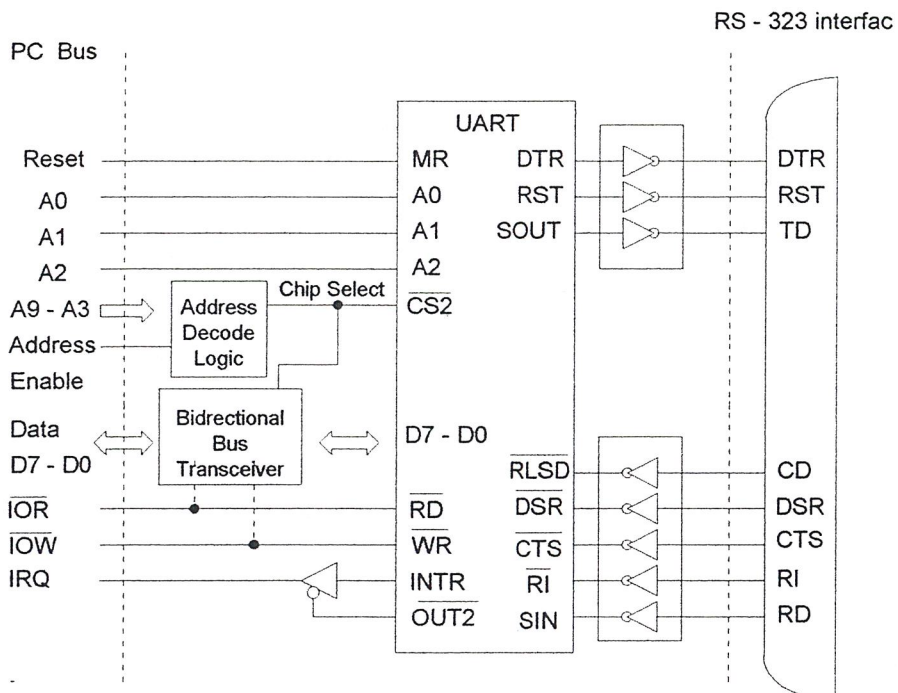
ในเครื่องคอมพิวเตอร์ทั่วไป จะมีอะแดปเตอร์สื่อสารแบบอะซิงโครนัส (Asynchronous) เป็นการ์ดอะแดปเตอร์แบบอนุกรมและขนานในการ์ดอันเดียวกัน

สำหรับอะแดปเตอร์แบบอนุกรมจะแบ่งได้เป็น 3 ประเภทดังนี้

- ประเภทแรก เป็นอะแดปเตอร์ที่ใช้ชิพ (Chip) เบอร์ 8250 หรือ 16450 UART เป็นอะแดปเตอร์ที่พื้นฐานที่สุดสำหรับการสื่อสาร โดยทั่วไป

- ประเภทที่สอง คล้ายๆกับประเภทแรก แต่จะใช้ชิพเบอร์ 16550 และ FIFO (First In First Out) บัฟเฟอร์ ซึ่งมีประสิทธิภาพมากกว่า ใช้สำหรับการสื่อสารโมเด็มความเร็วสูงและสภาวะแวดล้อมการทำงานแบบหลายๆงาน (Multitasking Environment) แต่จะมีราคาแพงกว่าประเภทแรก

- ประเภทที่สาม ประกอบด้วยอะแดปเตอร์ แบบพิเศษ คือ ตัวควบคุมพอร์ตอนุกรม (Serial Port Controller) และเฮสท์เอ็นฮานซ์ซีเรียลอินเตอร์เฟส (Hayes Enhanced Serial Interface : HESI) สำหรับประเภทที่สามนี้จะสนับสนุนการ การถ่ายเทข้อมูลแบบ DMA (Direct Memory Access) และมีโฟลว์คอนโทรล (Flow Control) ของตัวเอง HESI เป็นโคโพรเซสเซอร์ (Coprocessor) ทางกาสื่อสารที่ สมบูรณ์ สามารถใช้ไมโครโพรเซสเซอร์ ของตัวเองในการจัดการสื่อสาร โดยไม่ขึ้น อยู่กับประเภทของคอมพิวเตอร์



รูปที่ 3.11 แสดงบล็อกไดอะแกรมพอร์ตอนุกรม

1. ไอโอแอดเดรส (I/O Address) ในคอมพิวเตอร์จะมีไอโอแอดเดรสสำหรับใช้อย่างถึงอะแดปเตอร์ แบบอนุกรมอยู่ 4 ช่วง ดังรูปที่ 3.2 ไอโอแอดเดรสดังกล่าวเชื่อมต่อกับ UART (Universal Asynchronous Receiver Transmitter) ซึ่งทำหน้าที่ในการติดต่อสื่อสารแบบอะซิงโครนัสและอินเตอร์เฟสแบบ RS-232 เมื่อเราทำการติดตั้งพอร์ตอนุกรมอันแรก (COM1) จะมีฐานแอดเดรสอยู่ที่ 03F8H โดยเราสามารถอ้างอิงรีจิสเตอร์ใน UART จากฐานของแอดเดรสนี้ เช่น รีจิสเตอร์ตัวที่ 4 ของ COM1 จะมีแอดเดรสตรงกับพีซี I/O แอดเดรสที่ $03F8H + 4 = 03FCH$

พอร์ตนุกรม	ฐานไอโอแอดเดรส	IRQ มาตรฐาน
COM1	03F8H-03FFH	4
COM2	02F8H-02FFH	3
COM3	03E8H-03EFH	2
COM4	02E8H-02EFH	1

ตาราง แสดงความสัมพันธ์ระหว่างเบอร์พอร์ตนุกรมและ ไอ โอแอดเดรส

หมายเหตุ COM1, COM2, COM3 และ COM4 เป็นชื่อของอุปกรณ์ (Device) ที่ตั้ง โดยระบบปฏิบัติการ (MS-Dos) สำหรับพอร์ตนุกรมอันที่ 1 ถึงอันที่ 4 ตามลำดับที่ติดตั้งอยู่ในระบบ เพราะ โปรแกรมส่วนใหญ่จะติดต่อกับฮาร์ดแวร์โดยตรง ดังนั้นชื่อเหล่านี้จึงเป็นที่รู้จักโดยทั่วไป เมื่ออ้างถึงการสื่อสารแบบอะซิงโครนัส

2. การเชื่อมต่อกับพีซีบัส (Bus Interface) การอ่านและการเขียนข้อมูลผ่านรีจิสเตอร์ของ UART ไอโอแอดเดสของรีจิสเตอร์นั้น จะวางอยู่บนพีซีแอดเดรสบัส พีซีสโตรบ (strobe) สัญญาณ IOR (I/O read) หรือ (I/O write) เมื่อส่วนวงจรถอดรหัสแอดเดรส (Address Decoding Logic) ที่อยู่บนการ์ดคอนโทรลตรวจสอบพบว่าเป็นแอดเดรสของตัวเอง มันจะทำการกระตุ้นสัญญาณเลือกชิป (Chip Select) ของ UART (CS2) การเข้าถึงรีจิสเตอร์ภายในของ UART สามารถทำได้โดยการถอดรหัส (Decode Address) จากเส้นแอดเดรส A2-A0 ซึ่งเป็น 3 บิตต่ำของไอโอแอดเดรส

สัญญาณเลือกชิปจะทำให้สัญญาณข้อมูลของ UART เชื่อมต่อกับบัสข้อมูลของพีซี โดยการควบคุมแบบรับส่งสองทาง (Bidirection Bus Transceiver) ทิศทางการไหลของข้อมูลที่ผ่านมาตัวรับส่ง (Transceiver) จะได้จากการตรวจสอบสัญญาณ IOR และ IOW ของพีซีซึ่งสัญญาณเหล่านี้จะต่อกับสัญญาณ RD และ WR ของ UART

3. การเชื่อมต่ออินเทอร์รัพต์ (Interrupt Interface) สำหรับแต่ละพอร์ตนุกรมจะมีสัญญาณอินเทอร์รัพต์โดยต่อสัญญาณ INTR เข้ากับบัสของพีซี โดยได้กำหนดให้ IRQ4 ใช้ร่วมกันระหว่าง COM1 และ COM3 ส่วน IRQ3 ใช้ร่วมกันระหว่าง COM2 และ COM4 หมายความว่าเราจะไม่สามารถที่จะใช้ COM1, COM3 โดยใช้แบบอินเทอร์รัพต์ได้ในเวลาเดียวกัน

จากบล็อกไคอะแกรม โครงสร้างพอร์ตอนุกรมข้างต้น จะเห็นว่าสัญญาณ INTR ไม่ได้ต่อโดยตรงกับสัญญาณ IRQ ที่อยู่บนบัสของพีซี แต่จะมีลอจิกเกตคั่นกลางซึ่งควบคุมการเปิดปิดของเกตนี้โดยใช้สัญญาณ OUT2 ของ UART ดังนั้นในการที่จะทำให้เกิดการที่จะทำให้เกิดการอินเตอร์รัพต์ขึ้นจึงต้องมีการโปรแกรมให้เซต บิต OUT2 ที่อยู่ในรีจิสเตอร์ควบคุมโมเด็ม (Modem Control Register) เป็น 1 เสียก่อน

4. อินเทอร์เฟซ RS-232 ขา DTR, RTS และ SOUT ของ UART จะต่อกับอินเทอร์เฟซแบบ RS-232 โดยผ่านตัวขับสัญญาณ (Line Drivers) สำหรับตัวขับสัญญาณนี้จะแปลงสัญญาณ 0 โวลต์ และ 5 โวลต์ของ UART ไปเป็น -12 โวลต์ และ +12 โวลต์ ตามมาตรฐานของ RS-232 เช่นเดียวกัน กับขาสัญญาณอินพุต CD, DSR, CTS, RI และ RD ของ RS-232 ก็จะต่อกับขาของ UART ที่สอดคล้องกัน ในการออกแบบตัวขับสัญญาณของ RS-232 นั้นจะใช้ตัวขับสัญญาณที่มีลอจิกตรงข้าม คือมัน จะทำการลบค่าลอจิกอินพุตที่มีมาจาก UART

5. รายละเอียดของรีจิสเตอร์ของ UART ในรายละเอียดของรีจิสเตอร์จะเป็นการกล่าวถึงหน้าที่ของแต่ละบิตที่อยู่ภายในรีจิสเตอร์ ซึ่งจะกล่าวถึงโดยอ้างจากฐานแอดเดรสของพอร์ต COM1

มาตรฐาน RS-232

มาตรฐาน RS-232 ได้จัดพิมพ์ขึ้นเมื่อปี ค.ศ. 1969 เป็นมาตรฐานที่กำหนดขึ้นโดยสมาคมของโรงงานอุตสาหกรรมผู้ผลิตอุปกรณ์อิเล็กทรอนิกส์แห่งอเมริกา (EIA : The Electronics Industries Association) ซึ่งใช้กันแพร่หลายในระบบการสื่อสารข้อมูลคอมพิวเตอร์ RS ย่อมาจาก Recommend Standard ส่วน 232 เป็นหมายเลขบังคับของมาตรฐานตัว C นี้ เป็นหมายเลขของฉบับท้ายสุดของมาตรฐาน โดยจะกล่าวถึงมาตรฐานของลักษณะของสัญญาณไฟฟ้าในการอินเทอร์เฟซเทอร์มินอลเข้ากับไมโครคอมพิวเตอร์ หรืออินเทอร์เฟซเครื่องพิมพ์เข้ากับคอมพิวเตอร์เป็นต้น โดยจะทำการส่งข้อมูล แบบอนุกรม จุดประสงค์ของมาตรฐานนี้ก็เพื่อบรรยายคุณลักษณะของการเชื่อมต่ออุปกรณ์รับ-ส่งข้อมูลปลายทาง (Data terminal Equipment : DTE) กับอุปกรณ์สื่อสารข้อมูล (Data Communication Equipment : DCE) สำหรับผู้ใช้ไมโครคอมพิวเตอร์ DTE หมายถึงตัวไมโครคอมพิวเตอร์ และ DCE หมายถึงโมเด็ม อุปกรณ์อื่นๆ เช่นเครื่องพิมพ์ที่รับสัญญาณ แบบอนุกรมอาจจะเป็นได้ทั้ง DTE และ DCE ขึ้นอยู่กับผู้ผลิต ซึ่งความเร็วและระยะทางของการเชื่อมต่อของ RS-232 สามารถเชื่อมต่อการถ่ายโอนข้อมูลได้จาก 0-20,000 บิตต่อวินาที ซึ่งเพียงพอสำหรับไมโครคอมพิวเตอร์ที่มีอัตราบอด 110 ถึง 9600 ความยาวของสายเชื่อมต่อโดยสัญญาณตามมาตรฐานของ RS-232 จำกัดอยู่ประมาณ 50 ฟุต

ลักษณะของสัญญาณ RS-232

ถ้าแรงดันไฟฟ้าเป็นบวกสภาพลอจิกจะเป็นศูนย์ สถานภาพของสัญญาณจะเป็นสเปซและฟังก์ชันในการควบคุมจะเป็น ON แต่เมื่อแรงดันไฟฟ้าเป็นลบ สภาพลอจิกจะเป็นหนึ่ง ส่วนสถานภาพของสัญญาณจะเป็นมาร์คและฟังก์ชันในการควบคุมจะเป็น OFF ลักษณะของสัญญาณที่ขาต่างๆ จะเป็นดังนี้

- สัญญาณ Transmit Data เป็นสัญญาณที่ส่งออกจาก DTE หรือ ไมโครคอมพิวเตอร์ไปยังโมเด็ม หรือต่อเข้ากับไมโครคอมพิวเตอร์ตัวอื่น หรือเครื่องพิมพ์ เมื่อไม่มีสัญญาณส่งออกสถานภาพของลอจิกที่ขาี้จะมีค่าเท่ากับหนึ่ง

- สัญญาณ Receive Data เป็นทางเข้าของสัญญาณ ไปยัง DTE หรือ ไมโครคอมพิวเตอร์ เมื่อไม่มีสัญญาณรับเข้ามา ขานี้จะมีสถานภาพของลอจิกเท่ากับหนึ่ง

- สัญญาณ Request To Send ใช้สัญญาณส่งสัญญาณ ไปยังโมเด็มหรือเครื่องพิมพ์เป็นการเรียกร้องที่ส่งสัญญาณมาทางขา 2 สัญญาณนี้ใช้คู่กับ CTS หรือ Clear To Send อุปกรณ์รับ หากได้รับสัญญาณ RTS ก็จะทราบว่าจะพร้อมจะรับสัญญาณได้หรือยัง หากพร้อมที่จะรับก็จะส่งออกไปที่สาย CTS

- สัญญาณ Clear To Send เมื่ออยู่ในสถานะ OFF หรือลอจิกหนึ่ง หมายความว่าพร้อมจะรับข้อมูลแล้ว

- สัญญาณ Data Set Ready เมื่อสัญญาณนี้อยู่ในสถานะ ON หรือลอจิกศูนย์ เป็นการบอกไมโครคอมพิวเตอร์ หรือฝ่ายส่งว่า โมเด็มต่อเข้ากับสายโทรศัพท์เรียบร้อยแล้ว และพร้อมที่จะส่งแล้ว โมเด็มที่มีการหมุนหมายเลขอัตโนมัติจะส่งสัญญาณสายนี้ไปบอกให้คอมพิวเตอร์รู้ว่าต่อโทรศัพท์สำเร็จแล้ว

บทที่ 4

การประมวลผลและการแสดงผล

4.1 เคลไฟล์

เคลไฟล์เป็นเครื่องมือสำหรับการพัฒนาแอปพลิเคชันบน Windows ภาษาพื้นฐานที่ใช้คือ ภาษา Pascal (Object Pascal) ในการเขียนโปรแกรม ดังนั้นผู้ที่เคยเขียนภาษา Pascal มาก่อนจะรู้สึกง่ายในการใช้งาน นอกจากนี้เคลไฟล์ยังสนับสนุนการพัฒนาโปรแกรมแบบ Visual Programming ที่ผู้พัฒนาสามารถเลือกคอมโพเนนต์ที่ต้องการมาวางบนฟอร์ม แล้วกำหนดคุณสมบัติบางอย่างของคอมโพเนนต์เหล่านั้น รวมทั้งอาจมีการเขียนโปรแกรมควบคุมการทำงานของคอมโพเนนต์เหล่านั้นด้วย

ความสามารถหลัก ๆ ของ Delphi มีดังนี้

1. พัฒนาแอปพลิเคชันทั่วไปที่รันบนระบบปฏิบัติการวินโดวส์โดยเราสามารถสร้างโปรแกรมด้านกราฟฟิก โปรแกรมคำนวณต่างๆได้อย่างมากมายให้ตรงกับความต้องการได้

2. พัฒนาแอปพลิเคชันด้านฐานข้อมูล ซึ่ง Delphi มีจุดเด่นเป็นอย่างมากในการสร้างแอปพลิเคชันประเภทนี้ ไม่ว่าจะเป็ความง่ายในการพัฒนา ความเร็วของแอปพลิเคชันที่พัฒนาหรือความพร้อมในของเครื่องมือที่ใช้ในการพัฒนาแอปพลิเคชัน

เราสามารถติดต่อกับไฟล์ข้อมูลทั่วไป เช่น dBASE, Paradox, และ Access หรือติดต่อกับฐานข้อมูลบนเครื่อง Server ไม่ว่าจะเป็น Oracle, Interbase, SQL Server หรือฐานข้อมูลประเภทอื่นๆ ได้

นอกจากนี้ Delphi ยังมีเครื่องมือในการสร้างรายงานชื่อ QuickReport ซึ่งเป็นเครื่องมือที่ช่วยให้เราสามารถสร้างรายงานได้อย่างง่าย

1. ActiveX แบบต่างๆ ได้แก่ ActiveX Control และ ActiveForm Control ก็คือเครื่องมือที่ช่วยให้เราสามารถนำส่วนของโปรแกรมที่เราได้สร้างเอาไว้ใน Delphi ไปใช้ในโปรแกรมอื่น ได้ เช่น ใน Visual Basic หรือ Microsoft Excel เป็นต้น

2. คอมโพเนนต์ของเคลไฟล์

ในการสร้างแอปพลิเคชันด้วยเคลไฟล์นั้น เรานำเอาสิ่งที่เรียกว่าคอมโพเนนต์ (component) ต่างๆมาประกอบกันเป็นแอปพลิเคชัน โดยจะเขียนโค้ดในภาษาปาสคาลควบคุมไปเพื่อควบคุมการทำงานด้วย การนำเอาคอมโพเนนต์ต่างๆมาสร้างเป็นแอปพลิเคชันนั้นช่วยลดเวลาในการสร้างแอปพลิเคชันมาก

คอมโพเนนต์ของเคลฟี่ สามารถแบ่งออกเป็น 2 ประเภทคือ

1. คอมโพเนนต์แบบ Visual (Visual Component)

เป็นคอมโพเนนต์ที่เราสามารถมองเห็นได้ทั้งขณะออกแบบ และในขณะรัน เช่น คอมโพเนนต์ปุ่ม (Button) คอมโพเนนต์รูปภาพ (Image) เป็นต้น

2. คอมโพเนนต์แบบ Non – Visual (Non – Visual Component)

เป็นคอมโพเนนต์ที่ขณะออกแบบจะเห็นเป็นไอคอน (Icon) แต่ขณะรันจะมองไม่เห็นคอมโพเนนต์นั้น เช่น คอมโพเนนต์ตาราง (Table) คอมโพเนนต์แหล่งข้อมูล(DataSource) เป็นต้น ในเคลฟี่ การใช้คอมโพเนนต์เป็นส่วนที่สำคัญที่สุดในการพัฒนาแอปพลิเคชัน นอกจากนี้จะใช้ทำเป็นส่วนที่ติดต่อกับผู้ใช้ (User Interface) แล้วยังใช้ในการควบคุมการทำงานของแอปพลิเคชันอีกด้วย

เราสามารถที่จะควบคุมการทำงานของคอมโพเนนต์ได้ 3 ทาง คือทางคุณสมบัติ

(Prpperties) ,เมธอด (Methods) และอีเวนต์ (Event) โดยใช้คุณสมบัติในการกำหนดค่าต่างๆ ให้คอมโพเนนต์และการใช้ Methods ควบคุมการทำงานของคอมโพเนนต์ขณะเกิด Event ต่างๆ

คุณสมบัติ (Properties)

เป็นส่วนที่ใช้ในการกำหนดลักษณะของคอมโพเนนต์ที่ต้องการเช่นคุณสมบัติของ Font ในคอมโพเนนต์ Label จะใช้กำหนดว่าข้อความที่จะปรากฏบนคอมโพเนนต์ Label นั้นจะมีรูปแบบ Font อย่างไร ซึ่งคุณสมบัตินี้จะช่วยให้เราสามารถพัฒนาแอปพลิเคชันได้อย่างสะดวกขึ้น โดยสามารถแก้ไขคุณสมบัติของคอมโพเนนต์ได้โดยไม่ต้องเขียนโปรแกรมและเห็นผลของการกำหนดค่าทันที ไม่ต้องรอผลในการรัน ซึ่งจะช่วยป้องกันการเกิด Error ได้อีกวิธีหนึ่ง การกำหนดค่าให้กับพรอพเพอร์ตี้ สามารถทำได้ 2 วิธี

1. กำหนดผ่านออบเจกต์อินสเปกเตอร์ (Object Inspector) ในขณะออกแบบ
2. เขียนเข้าไปในโปรแกรมเพื่อกำหนดค่าพรอพเพอร์ตี้ในขณะรัน

เมธอด (Methods)

เป็นลักษณะของฟังก์ชันแบบหนึ่งที่ใช้ประกอบการทำงานของคอมโพเนนต์นั้นๆ ในการเรียกใช้ Methods ก็เหมือนกับการเรียกใช้ฟังก์ชัน คือสามารถส่งตัวแปรไปและคอยรับค่าที่ได้จากการคำนวณกลับมา โดยที่ Methods จะสามารถเรียกใช้ได้เฉพาะขณะรันแอปพลิเคชันเท่านั้น

อีเวนต์ (Event)

เป็นส่วนของคอมโพเนนต์ที่ใช้ในการรับการตอบสนองต่อการกระทำอย่างใดอย่างหนึ่ง ที่เกิดขึ้นกับคอมโพเนนต์ เช่น เมื่อทำการ Click ที่ คอมโพเนนต์ Button1 จะเกิด Events OnClick ของคอมโพเนนต์ Button1 ขึ้น ซึ่งในบาง Events จะมีการส่งค่าพารามิเตอร์ที่เป็นข้อมูลที่เกี่ยวข้องกับ Events ที่เกิดขึ้นมาใหม่ และเราสามารถตรวจสอบได้ว่าคอมโพเนนต์นั้น ๆ มีความสามารถในการเกิด Events ได้บ้างโดยให้ดูที่แท็บ Events ของหน้าต่าง Object Inspector

ในเคลไฟล์แบ่งอีเวนต์ออกเป็น 2 ชนิดคือ

1. อีเวนต์ที่เกิดจากผู้ใช้ เช่น ผู้ใช้คลิกที่ปุ่มบนฟอร์ม หรือเลือกข้อมูลจากรายการของข้อมูล เป็นต้น
2. อีเวนต์ที่เกิดจากระบบ เช่น เมื่อแอปพลิเคชันเริ่มทำงาน , แอปพลิเคชันจะจบการทำงาน เป็นต้น

แต่ละอีเวนต์ที่เกิดขึ้นนั้นจะมีเมธอดที่ใช้ตอบสนองต่ออีเวนต์นั้น ซึ่งเรียกว่า อีเวนต์แฮนด์เลอร์ (Event Handler) สำหรับอีเวนต์ที่จะเกิดกับแต่ละคอมโพเนนต์นั้น เราดูได้จากเพจอีเวนต์ของออปเจกต์อินสเปกเตอร์

Form Designer

Form Designer ใช้สำหรับออกแบบส่วนที่ติดต่อกับผู้ใช้ ซึ่งเราสามารถนำคอมโพเนนต์ต่างๆ จาก คอมโพเนนต์เพลตเล็ต มาวางลงบน form Designer โดยที่เราสามารถแก้ไขขนาด ย้ายคอมโพเนนต์ไปมา รวมทั้งเพิ่มหรือลบคอมโพเนนต์ออกจากฟอร์มได้ โดยเราสามารถเปรียบเทียบ Form Designer ได้กับกระดาษวาดเขียนที่เราสามารถวาดหรือลบส่วนที่เราวาดได้

Code Editor

Code Editor เป็น Text Editor สำหรับการเขียนโปรแกรมในเคลไฟล์ ซึ่งประกอบไปด้วยคุณสมบัติต่าง ๆ ที่อำนวยความสะดวกในการเขียนโปรแกรม

เมื่อเปิดเคลไฟล์ขึ้นมาครั้งแรก ภายใน Code Editor จะแสดง Code เริ่มต้นที่เคลไฟล์สร้างเอาไว้ให้ชื่อ Unit 1 ภายใน code Editor เราสามารถเปิดไฟล์ได้พร้อมกันหลายไฟล์ โดยเราสามารถเลือกไฟล์ที่ต้องการแก้ไขได้จากแท็บทางด้านบนของ Code Editor โดยคลิก บนแท็บที่แสดงชื่อไฟล์ที่ต้องการ

Code Editor ช่วยเพิ่มความสะดวกในการเขียนโปรแกรมเป็นอย่างมาก เราสามารถปรับแต่งคุณสมบัติต่าง ๆ ของ Code Editor ได้โดยเลือกเมนู Tool >Environment Options

คอมโพเนนต์ที่ใช้ในการแสดงผลทางหน้าจอ

ชื่อ	หน้าที่และการใช้งาน
Label	ใช้สำหรับแสดงข้อความ
Edit	ใช้แสดงข้อความที่ละบรรทัด ที่ผู้ใช้สามารถแก้ไขข้อความนั้นได้
Button	ใช้สร้างปุ่มพื้นฐานทั่วไป
Check Box	ใช้สำหรับงานที่ต้องการ Option ให้เลือก ซึ่งสามารถเลือกได้หลายๆOption พร้อมกัน
Radio Button	เหมือน Check Box แต่สามารถเลือกได้เพียง 1 ตัวเลือกในกลุ่มตัวเลข เดียวกัน
List Box	ใช้สำหรับแสดงข้อมูลที่เป็น list ให้เลือก
Combo Box	เป็นคอมโพเนนต์ที่ผสมระหว่าง Edit กับ List Box
DBGrid	เป็นเหมือนตารางสำหรับแสดง และแก้ไขข้อมูลในฐานข้อมูลซึ่ง DBGrid จะครอบคลุมการทำงานเกือบทั้งหมดในการจัดการกับข้อมูล การแสดงข้อมูล การแก้ไขข้อมูล และการลบข้อมูล
Table	เป็นคอมโพเนนต์ที่ใช้ในการเข้าถึงตารางฐานข้อมูลโดยตรง โดยสามารถแก้ไข ค้นหา คัดลอก และลบข้อมูลในตาราง สามารถกำหนดให้มีการกรองเอาเฉพาะข้อมูลที่ต้องการ และสามารถกำหนดความสัมพันธ์แบบ Master / Detail กับตารางอื่นได้
DataSource	เป็นตัวกลางในการเชื่อมต่อระหว่าง DataSet (Table) กับData Control (DBGrid) ที่ใช้สำหรับการแสดง / แก้ไขข้อมูล

การใช้คอมโพเนนต์เหล่านี้ สามารถเลือกใช้ได้โดย Double Click ที่คอมโพเนนต์ที่ต้องการที่ Component Palette ก็จะมีคอมโพเนนต์ที่ต้องการทันที

4.2 เคลฟล์กับฐานข้อมูล

ฐานข้อมูลเป็นสิ่งที่จำเป็นสำหรับระบบงานทางด้านคอมพิวเตอร์ ซึ่งแอปพลิเคชันส่วนใหญ่ที่มีการพัฒนาขึ้นมักจะเป็นแอปพลิเคชันที่ทำงานด้านฐานข้อมูลด้วยกันทั้งนั้น ซึ่งงานทางด้านนี้เป็นงานที่มีความละเอียดค่อนข้างมาก และมักมีการเปลี่ยนแปลงเสมอ ดังนั้นการมีเครื่องมือในการพัฒนาแอปพลิเคชันที่สามารถสร้างและปรับปรุงเปลี่ยนแปลงสิ่งต่างๆ ได้ง่ายจะเป็นประโยชน์อย่างมากในการพัฒนาระบบงาน

เราใช้เคลฟล์เป็นโปรแกรมช่วยในการจัดเก็บข้อมูลในส่วนของฐานข้อมูล โดยเคลฟล์ได้แบ่งส่วนต่างๆ ของระบบงานด้านฐานข้อมูลอย่างเป็นระเบียบตั้งแต่การเชื่อมต่อกับฐานข้อมูล การเข้าถึงฐานข้อมูล การติดต่อกับผู้ใช้ และการอำนวยความสะดวกต่างๆ

ฐานข้อมูลมาตรฐานที่เคลฟล์ใช้เป็นตัวเก็บข้อมูลคือ Paradox และ dBase แต่เคลฟล์สามารถทำงานกับฐานข้อมูลภายนอกอื่นๆ ได้ไม่ว่าจะเป็นไฟล์ฐานข้อมูลทั่วไปเช่น Microsoft Access , Microsoft FoxPro และในชุด Client / Server ยังสามารถใช้กับระบบฐานข้อมูลที่อยู่ในเครื่อง Server เช่น Oracle , MS SQL Server , Interbase

เคลฟล์มีการเชื่อมต่อกับฐานข้อมูลผ่านระบบที่ชื่อว่า Borland Database Engin (BDE) ซึ่งเป็นตัวจัดการระบบทางด้านฐานข้อมูลทั้งหมด เคลฟล์มีโปรแกรมช่วยอำนวยความสะดวกในการพัฒนางานด้านฐานข้อมูลให้อย่างเปรียบเทียบพร้อม โปรแกรมต่างๆ เหล่านี้จะช่วยให้การพัฒนางานสะดวกและง่ายขึ้นเป็นอย่างมากตั้งแต่การสร้าง และปรับปรุงโครงสร้างฐานข้อมูล การโอนย้ายข้อมูล การแก้ไขข้อมูล และการแสดงข้อมูล

โปรแกรมสำหรับช่วยงานทางด้านฐานข้อมูลในเคลฟล์มีดังนี้

- BDE Administrator
- Database Desktop
- Database Explorer
- Data Migration Expert
- SQL Monitor

ศัพท์ทางด้านฐานข้อมูลที่ควรรู้

ฐานข้อมูล (Database) คือกลุ่มของข้อมูลที่มีความสัมพันธ์กัน ซึ่งเราสามารถบันทึก เปลี่ยนแปลง ลบและสืบค้นกลับมาใช้งาน ได้อย่างรวดเร็วและเป็นระบบ

ตาราง (Table) คือที่สำหรับเก็บข้อมูล ซึ่งในฐานข้อมูลอาจมีตารางได้หลายตาราง โดยแต่ละตารางจะด้วยคอลัมน์ เรียกว่าฟิลด์ และแถวเรียกว่า เรคคอร์ด

อินเด็กซ์ (Index) คือดัชนีของตาราง เราอาจมองเห็นเหมือนกับดัชนีที่อยู่หน้าท้ายของหนังสือ ซึ่งจะเป็นตัวช่วยในการค้นหาข้อมูลภายในตาราง

SQL (Structure Query Language) คือภาษาที่ใช้ในการจัดการกับข้อมูลในฐานข้อมูลเช่น การเรียกข้อมูลคิวรี ออกมาแสดง การเพิ่มข้อมูล การแก้ไขข้อมูล และการลบข้อมูล เป็นต้น

Query คือการสอบถามข้อมูลจากฐานข้อมูล ปกติเราจะเขียน คิวรี โดยใช้คำสั่ง SQL

BDE (Borland Database Engine) เป็นระบบที่ใช้ในการเชื่อมต่อกับฐานข้อมูล ซึ่งงานด้านฐานข้อมูลต่างๆ ของเดลไฟล์ จะทำผ่าน BDE ทั้งหมด

Driver เป็นตัวกลางที่ช่วยในการเชื่อมต่อจาก BDE ไปยังฐานข้อมูล เราสามารถเชื่อมต่อไปยังฐานข้อมูลทุกประเภทที่เรามี Driver ในการเชื่อมต่อ ซึ่งใน BDE มี Driver ในการเชื่อมต่อกับฐานข้อมูลที่เป็นที่นิยมในปัจจุบันเกือบทุกประเภท เราอาจจะต้องกำหนดคุณสมบัติบางตัวของ Driver ให้ถูกต้องเพื่อให้การเชื่อมต่อกับฐานข้อมูลเป็นไปอย่างสมบูรณ์

Alias เป็นตัวเก็บคุณสมบัติต่างๆ ในการเชื่อมต่อไปยังฐานข้อมูล เช่นประเภทฐานข้อมูล , ตำแหน่งที่เก็บฐานข้อมูล , ชื่อฐานข้อมูล เป็นต้น เช่น มีฐานข้อมูล Paradox อยู่ที่ โฟลเดอร์ C:\Address เราสามารถสร้าง เอเลียสที่เชื่อมต่อไปยังฐานข้อมูลนี้ โดยกำหนดคุณสมบัติการเชื่อมต่อดังนี้

Type = STANDARD

Default Driver = Paradox

Path = C:\Address

Enable BCD = False

การสร้างแอปพลิเคชันงานฐานข้อมูล โดยเคลไฟล์จะอาศัย

- เครื่องมือพัฒนางานด้านฐานข้อมูลของเดลไฟล์ (Delphi Database Development Tool)
- **Delphi Data Access Component** จะเป็นคอมโพเนนต์ที่ใช้ในการเข้าถึงข้อมูลในลักษณะต่างๆ ซึ่งลักษณะข้อมูลที่มีจะถูกเข้าถึงบ่อยคือการใช้คอมโพเนนต์ Table ซึ่งเราสามารถเข้าไปกำหนดค่าของ Fields ใน Records และสามารถจัดการข้อมูลได้เช่น กำหนด Table , List Table ภายใน Data base ซึ่งเครื่องมือเหล่านี้จะอยู่ใน Data Access Page ของ Delphi 's Component Palette ซึ่งประกอบด้วย Ttable , TdataSource , Tquery และอื่นๆ

Delphi Data Aware GUI Component เป็น Visual Component ที่ใช้แสดงข้อมูลของ Data Field ของTable ได้เช่น DBGrid , DBListBox และอื่นๆ ซึ่งจะเกี่ยวข้องกับ Database แต่ Tool เหล่านี้จะต้องใช้ร่วมกับ DataSource , DataAccess Component โดยใช้ Component ในการติดต่อกับผู้ใช้ หรือ BDE ซึ่งจะทำหน้าที่ติดต่อกับฐานข้อมูลอีกที

Delphi Data Control เป็นคอมโพเนนต์ที่ใช้ในการควบคุมการติดต่อกับผู้ใช้ในรูปแบบต่างๆ เช่น ช่องสำหรับกรอกข้อมูล , Combo Box , Check Box เป็นต้น

ขั้นตอนในการสร้างแอปพลิเคชันด้านฐานข้อมูล

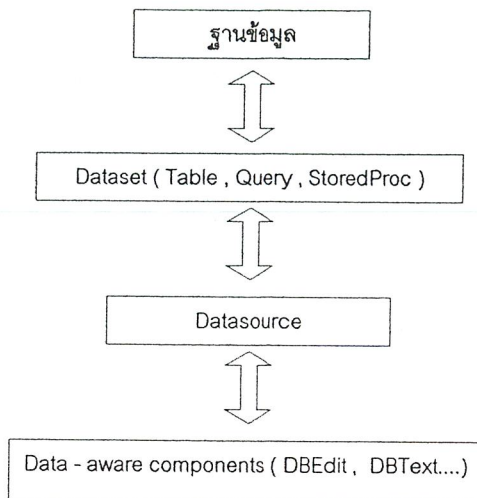
1. สร้าง Alias ที่เชื่อมต่อไปยังฐานข้อมูลที่ต้องการ(ใช้โปรแกรม BDE Administrator)
2. สร้างตารางฐานข้อมูล และ Index (ใช้โปรแกรม Database Desktop)
3. สร้างแอปพลิเคชันใหม่ในเดลไฟล์
4. กำหนดการเข้าถึงข้อมูลโดยใช้คอมโพเนนต์ประเภท Data Access
5. ออกแบบและสร้างหน้าจอส่วนติดต่อกับผู้ใช้ โดยใช้คอมโพเนนต์ประเภท Data Control

คอมโพเนนต์สำหรับสร้างแอปพลิเคชันฐานข้อมูล

คอมโพเนนต์ที่ใช้ในงานฐานข้อมูล มี 2 ส่วนคือ

1. คอมโพเนนต์ในเพจ Data Access
2. คอมโพเนนต์ในเพจ Data Control

เราสามารถอธิบายความสัมพันธ์ระหว่างคอมโพเนนต์จากทั้ง 2 เพจ และการติดต่อกับฐานข้อมูลได้ดังรูป 3.1



รูปที่ 4.1 แสดงการติดต่อกับฐานข้อมูล

จากรูปจะเห็นว่าส่วนที่ติดต่อกับฐานข้อมูลโดยตรงคือส่วนของ Dataset ซึ่งเป็นคอมโพเนนต์ที่ทำหน้าที่ดึงกลุ่มของข้อมูลจากตารางที่เราสนใจในฐานข้อมูล โดยคอมโพเนนต์ที่เป็น dataset คือ

คอมโพเนนต์ตาราง (Table)

คอมโพเนนต์คิวรี(Query)

ซึ่งคอมโพเนนต์ทั้ง 2 ตัวนี้จะอยู่ในเพจ Data Access แต่ Dataset จะแสดงข้อมูลจากฐานข้อมูลไม่ได้เนื่องจากเป็นคอมโพเนนต์แบบ Non - Visual

แต่การสร้างแอปพลิเคชันฐานข้อมูลส่วนมากจำเป็นจะต้องมี ส่วนที่ใช้แสดงข้อมูล ดังนั้นเคลไฟล์จึงมีคอมโพเนนต์ที่เรียกว่า Data-aware ซึ่งก็คือคอมโพเนนต์ในเพจ Data Control เป็นคอมโพเนนต์ที่สามารถนำข้อมูลจาก Dataset มาแสดงได้ แต่การนำข้อมูลมาแสดงนั้นต้องมีตัวกลางในการเชื่อมต่อระหว่างคอมโพเนนต์ของทั้ง 2 กลุ่ม ซึ่งเราจะเรียกว่า คอมโพเนนต์แหล่งข้อมูล (Datasource)

ฐานข้อมูลเป็นที่รวมของข้อมูลที่มีความสัมพันธ์กัน โดยข้อมูลเหล่านี้จะถูกเก็บอยู่ในตารางที่ประกอบด้วยฟิลด์และ เร็คคอร์ด

ฟิลด์ก็คือ หัวข้อของข้อมูลที่จัดกลุ่มอยู่ในตาราง เช่น ชื่อ ส่วนสูง น้ำหนัก เป็นต้น ถ้าเราใส่ข้อมูลต่าง ๆ จนครบทุกฟิลด์ในตารางเราก็จะได้เร็คคอร์ดขึ้นมา ซึ่งฟิลด์และเร็คคอร์ดก็คือคอลัมน์ และแถวของตารางนั่นเอง

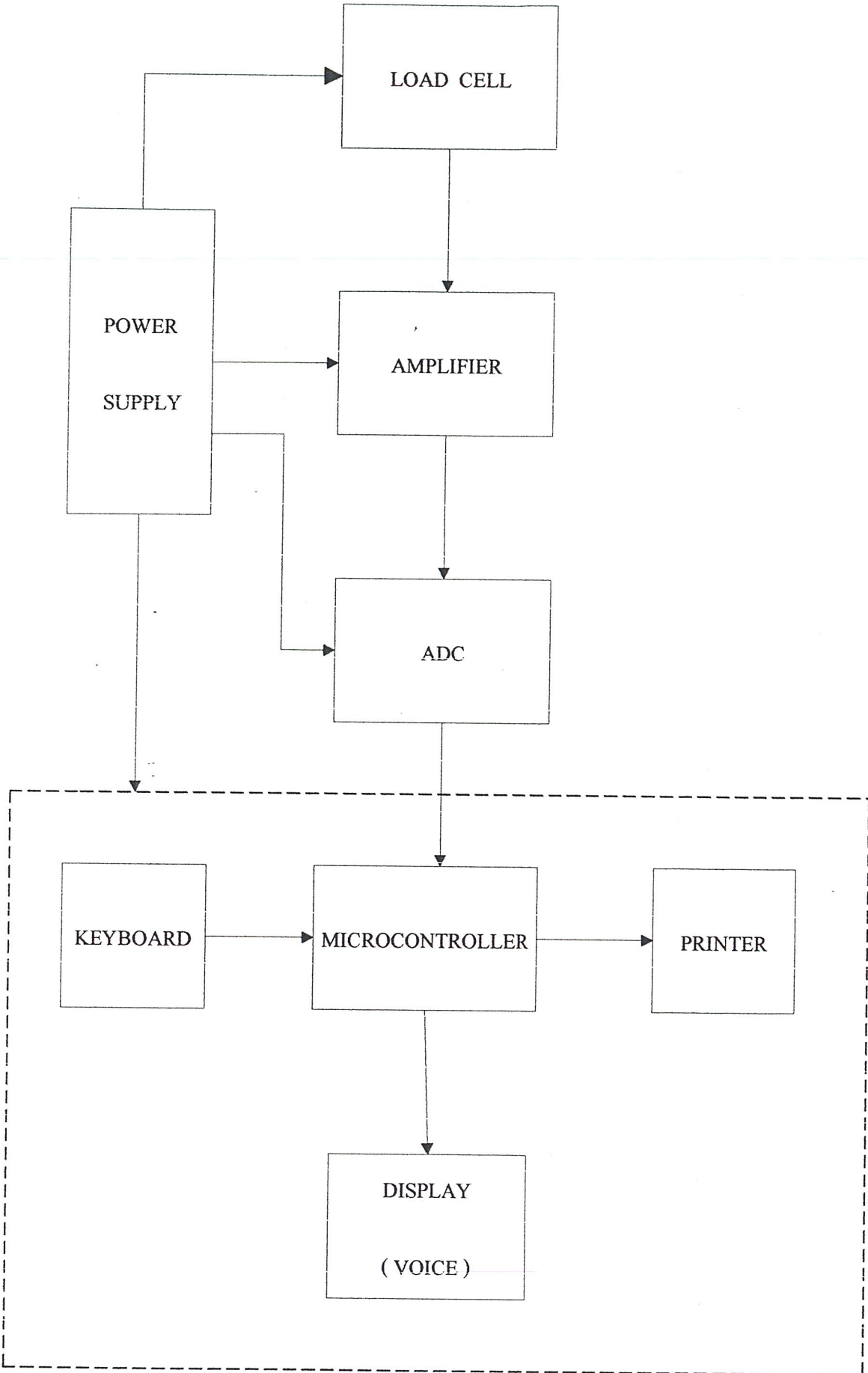
4.3 การบอกรำนำหน้าด้วยเสียงพูด

การใส่เสียงลงในแอปพลิเคชัน สามารถทำได้โดยใช้คอมพิวเตอร์ MediaPlayer ซึ่งฟังก์ชันการทำงานของคอมพิวเตอร์นี้คล้ายกับการเล่นเครื่องเสียงธรรมดา เช่น เปิด ปิด หยุดชั่วคราว การบันทึกเสียง โดยไฟล์ที่สามารถเล่นได้มีหลายรูปแบบ เช่น ไฟล์ WAV , ไฟล์ MIDI ไฟล์จาก CD

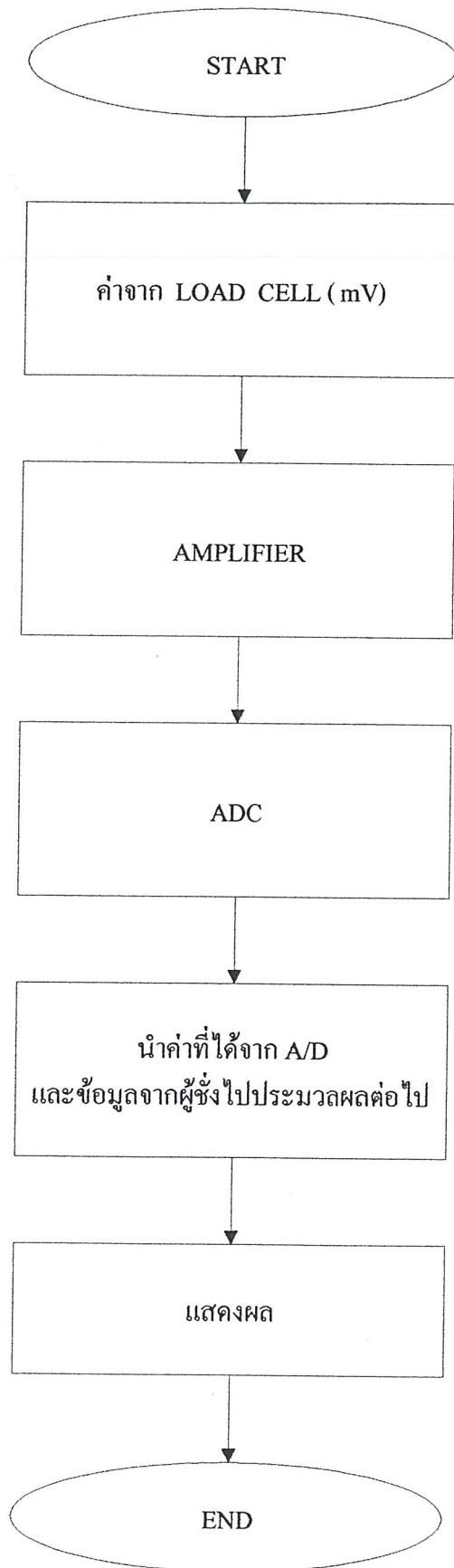
ขั้นตอนในการบอกรำนำหน้าด้วยเสียงพูด

1. อัดเสียงพูดลงในโปรแกรมอัดเสียงของ Windows โดยจะอัดเสียงคำพูดแต่ละคำไป เช่น คำว่า ศูนย์ หนึ่ง สอง สาม สี่ ห้า หก เจ็ด แปด เก้า สิบ เอ็ด ยี่ จุด กิโลกรัมแล้วบันทึกไว้เป็นไฟล์ Wav

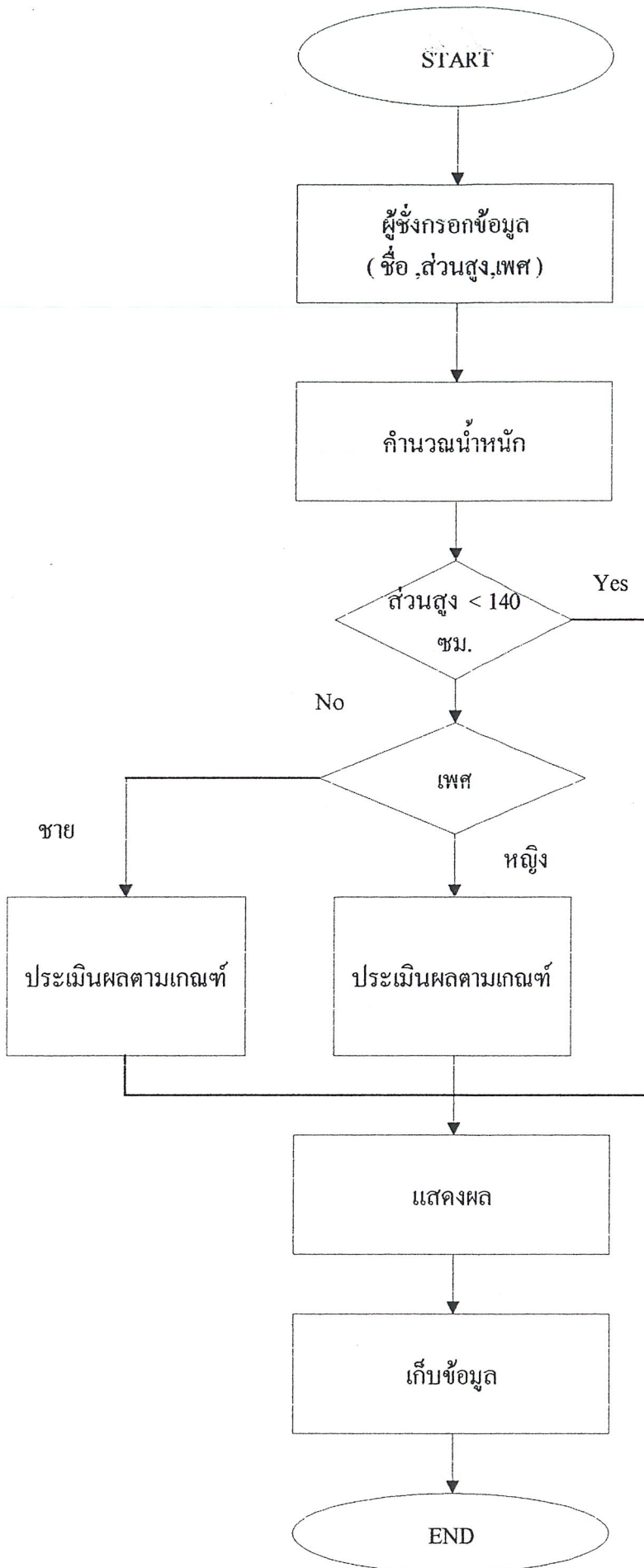
2. ใช้คอมพิวเตอร์ MediaPlayer ช่วยในการเล่นเสียง โดยเขียนโปรแกรมให้กับคอมพิวเตอร์ MediaPlayer นี้ให้บอกรำนำหน้าออกมาตามตัวเลขที่ขึ้นบนหน้าจอคอมพิวเตอร์



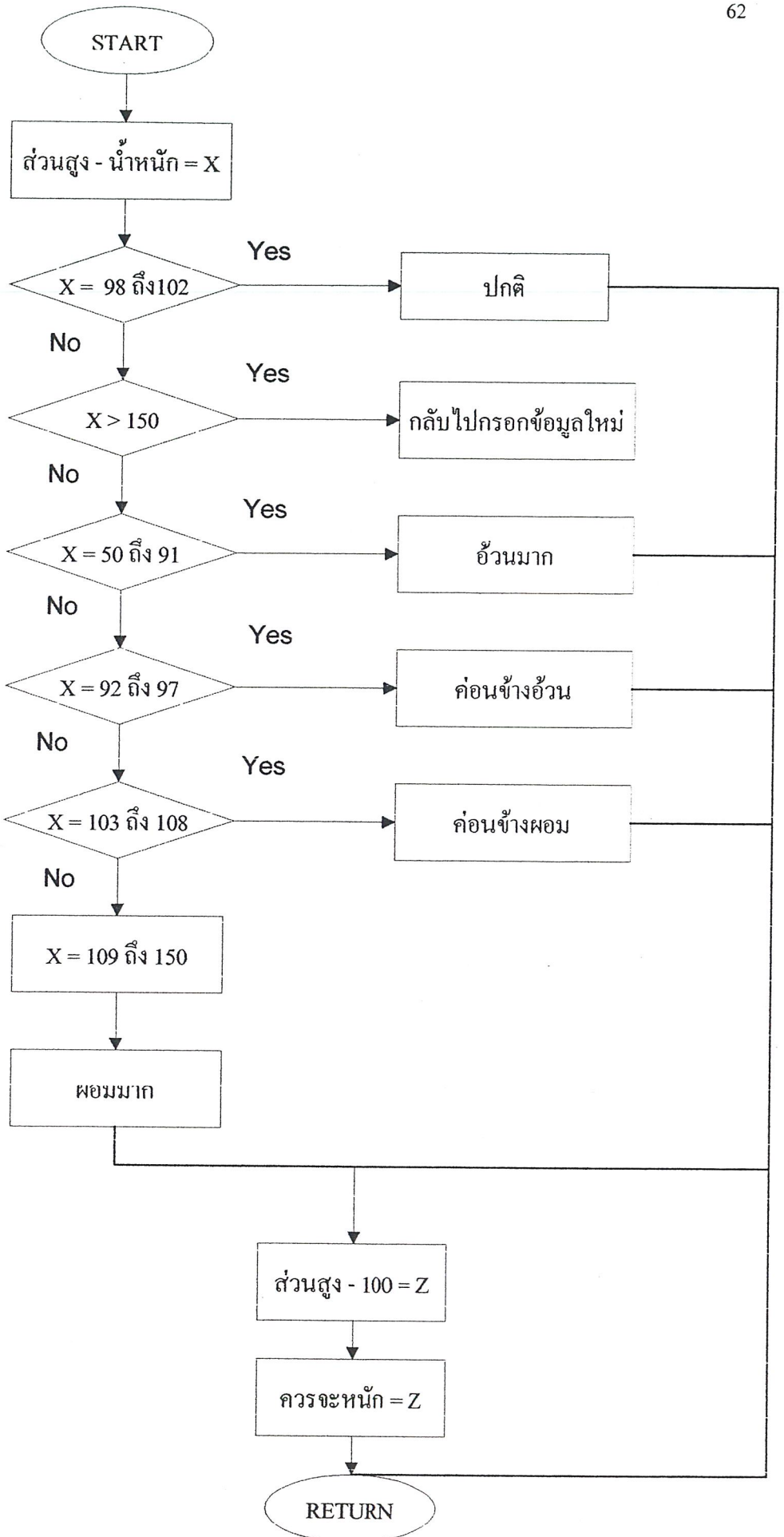
รูปที่ 4.2 แสดงส่วนประกอบต่างๆของเครื่องชั่ง



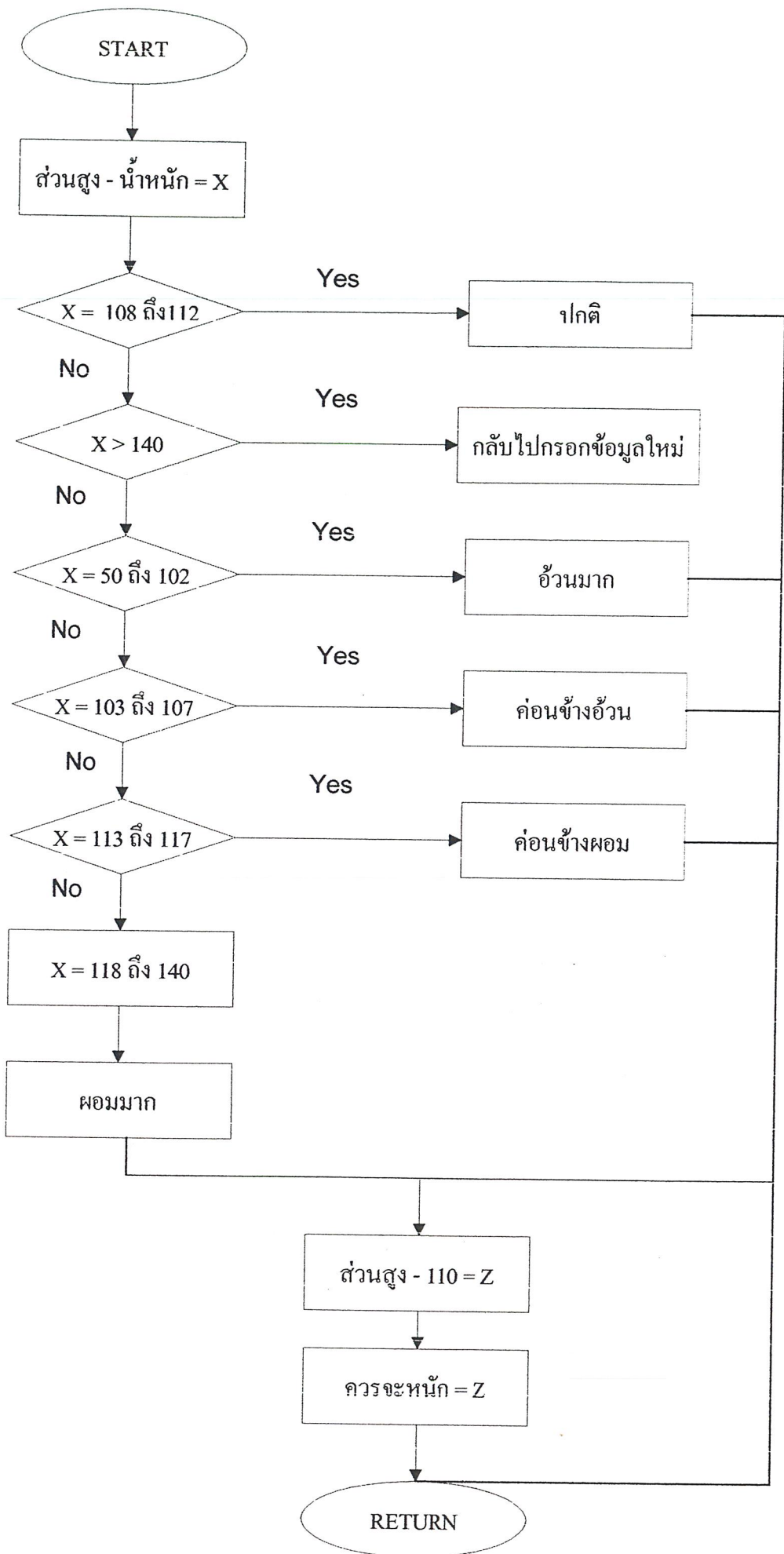
รูปที่ 4.3 แสดงโฟลว์ชาร์ทการทำงานของเครื่องชั่งน้ำหนัก



รูปที่ 4.4 แสดงโฟลว์ชาร์ทของโปรแกรมประมวลผล



รูปที่ 4.5 แสดงโฟลว์ชาร์ทของโปรแกรมประมวลผลเพศชาย



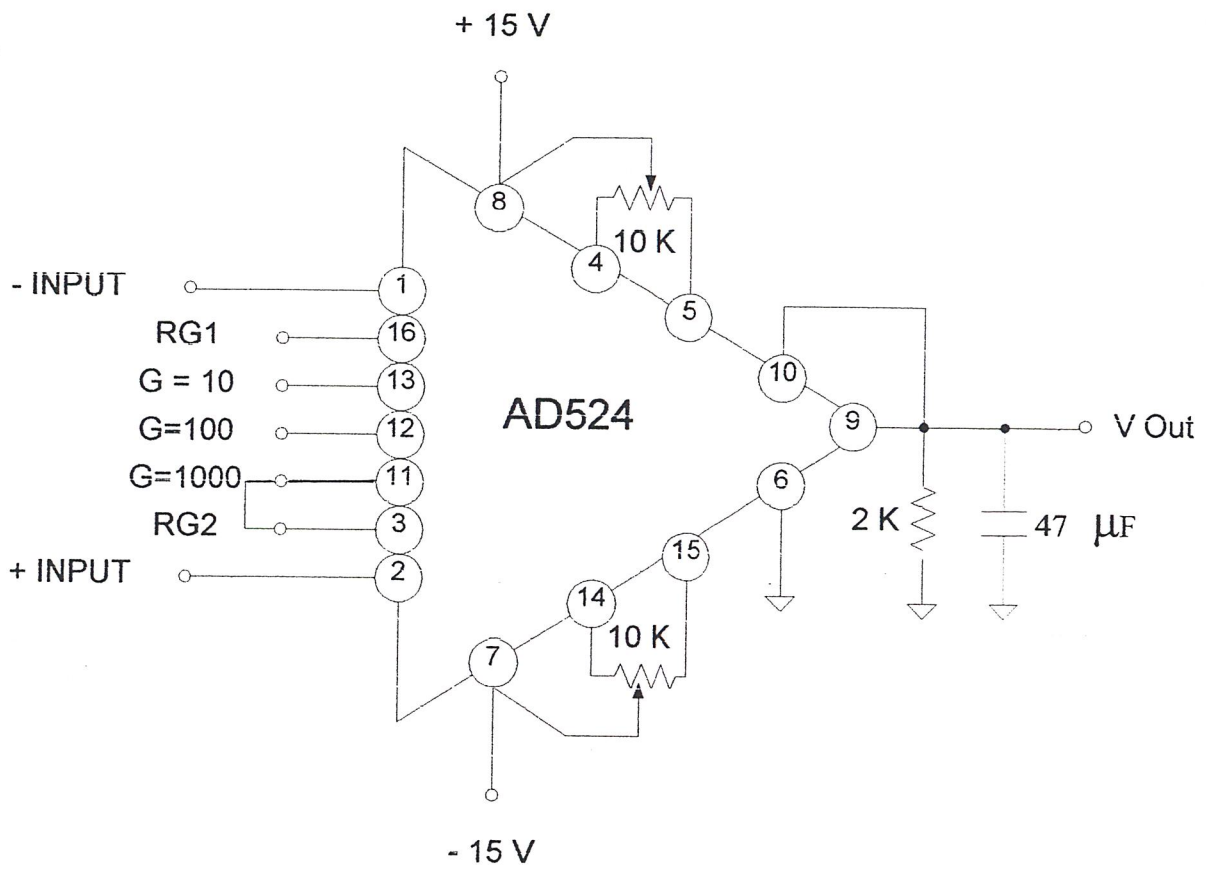
รูปที่ 4.6 แสดงโฟลว์ชาร์ทของ โปรแกรมประมวลผลเพศหญิง

บทที่ 5

วงจรการทำงานและผลการทดลอง

5.1 วงจร Instrument Amplifier

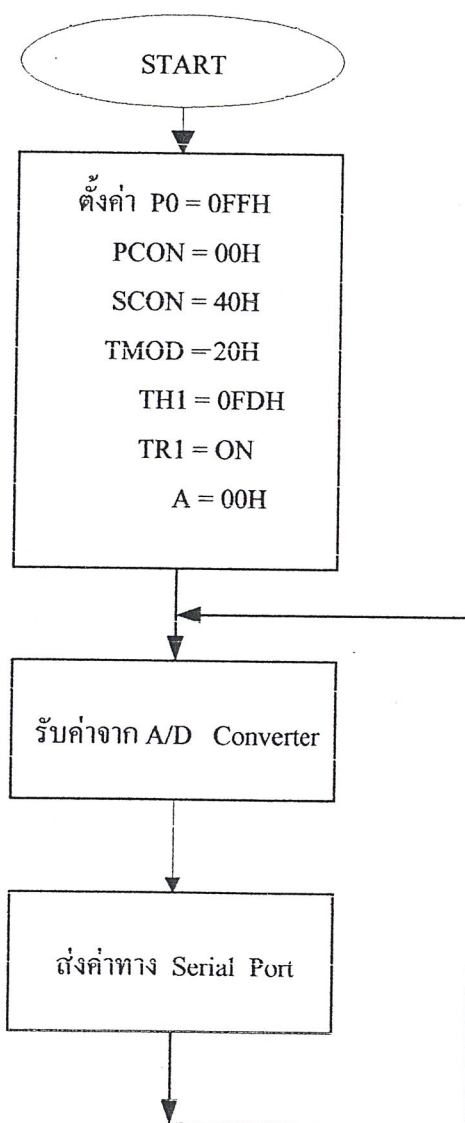
ใช้ในการขยายสัญญาณจากโหนดเซลล์ ที่มีค่าน้อยมาก ให้มีค่าประมาณ 0-5 โวลต์ เพื่อให้วงจรอินเทอร์เฟซสามารถทำงานได้



รูปที่ 5.1 วงจร Instrument Amplifier

5.2 วงจรอินเทอร์เฟซ (Interface)

แรงดันไฟฟ้าที่ได้รับขยายแรงดันจาก Instrument Amplifier ถูกส่งเข้าวงจรอินเทอร์เฟซ โดยผ่าน A/D คอนเวอร์เตอร์ขนาด 8 บิต ให้เป็นสัญญาณดิจิทัล เข้าสู่ไมโครคอนโทรลเลอร์ และเข้า MAX 232 ซีเรียลพอร์ต เพื่อส่งข้อมูลให้กับคอมพิวเตอร์ทำการประมวลผลต่อไป



รูปที่ 5.2 แสดงไฟลว์ชาร์ตการทำงานของวงจรอินเทอร์เฟซ

5.3 ผลการทดลองของวงจร Analog to Digital

ทดลองโดยการป้อนอินพุตโวลเตจให้กับ A/D ค่าต่างๆ

ค่าแรงดันอินพุต (V)	ค่าดิจิทัลที่ได้จากการวัด	ค่าดิจิทัลที่ได้จากการคำนวณ
0.0	00000000b	00000000b
0.5	00011011b	00011010b
1.0	00110110b	00110010b
1.5	01001100b	01001101b
2.0	01100110b	01100110b
2.5	10000010b	10000000b
3.0	10011100b	10011001b
3.5	10110011b	10110011b
4.0	11001110b	11001100b
4.5	11100111b	11100110b
5.0	11111110b	11111111b

ตาราง 5.1 ผลการทดลองวงจร Analog to Digital

5.4 การทำงานของโหนดเซลล์

การติดตั้งโหนดเซลล์

Supply	Green (input+)	, +5 V
	Black (input-)	, 0 V
signal	Red (output+)	ขา 2 (Instrument Amp)
	Write (output-)	ขา 1 (Instrument Amp)

5.5 ผลการทดลองโพลีเซลล์

ทำการวัดค่าเอาต์พุตจาก โพลีเซลล์ ในขณะที่มีการชั่งน้ำหนักต่างๆ โดยทำการป้อนแรงดันไฟฟ้าให้กับโพลีเซลล์ 5 โวลต์

น้ำหนัก (Kg.)	แรงดันไฟฟ้า (mV)
0	0.4
10	0.9
20	1.4
30	1.9
40	2.4
50	2.9
60	3.4
70	3.9
80	4.4
90	4.9
100	5.4

ตาราง 5.2 แสดงค่าเอาต์พุตที่ได้จากโพลีเซลล์

น้ำหนักของโพลีเซลล์ที่วัดได้จากโพลีเซลล์จะอยู่ในรูปของแรงดันไฟฟ้า (DC Voltage) มีค่าน้อยมากมีหน่วยเป็นมิลลิโวลต์ (mV) ซึ่งไม่พอที่จะนำเข้าสู่ A/D คอนเวอร์เตอร์ จึงต้องผ่าน Instrument Amplifier เพื่อทำการขยายแรงดัน จากการทดสอบเมื่อไม่มีโพลีเซลล์และไม่มีการปรับค่าความต้านทาน RL จะมีแรงดันเอาต์พุตจาก Instrument Amplifier อยู่ค่าหนึ่ง เมื่อต้องการหาค่าแรงดันที่ได้จากโพลีเซลล์จึงปรับค่าความต้านทาน RL เพื่อให้แรงดันขณะไม่มีโพลีเซลล์เท่ากับศูนย์และสามารถเปรียบเทียบค่าความแตกต่างของแรงดันที่วัดได้จากการชั่งน้ำหนักโพลีเซลล์ที่ค่าน้ำหนักต่างๆ ให้เห็นอย่างชัดเจน

5.6 ผลการทดลองวงจรรวม

ทำการวัดค่าเอาต์พุตจาก Instrument Amplifier ในขณะที่มีการชั่งน้ำหนักต่างๆ โดยทำการป้อนแรงดันไฟฟ้าให้กับโพลีเซลล์ 5 โวลต์

ครั้งที่ 1

น้ำหนักที่ทดสอบ (Kg.)	O/P จาก Ins.Amp(V)	ค่าความละเอียด	น้ำหนักที่วัดได้ (Kg.)
0	-0.031	0	0.00
10	0.471	25	9.80
20	0.975	50	19.60
30	1.475	76	29.80
40	1.977	101	39.60
50	2.474	127	49.80
60	2.976	153	60.00
70	3.470	179	70.10
80	3.983	204	80.00
90	4.501	231	90.50
100	4.950	255	100.00

ครั้งที่ 2

น้ำหนักที่ทดสอบ (Kg.)	O/P จาก Ins.Amp(V)	ค่าความละเอียด	น้ำหนักที่วัดได้ (Kg.)
0	-0.042	0	0.00
10	0.458	25	9.80
20	1.002	51	20.00
30	1.481	77	30.10
40	2.010	102	40.00
50	2.492	129	50.50
60	2.967	153	60.00

น้ำหนักที่ทดสอบ (Kg.)	O/P จาก Ins.Amp(V)	ค่าความละเอียด	น้ำหนักที่วัดได้ (Kg.)
70	3.465	179	70.10
80	3.970	204	80.00
90	4.47	230	90.10
100	4.97	255	100.00

ตารางที่ 5.3 เปรียบเทียบค่าแรงดันเอาพุทของโหลดเซลล์กับค่าความละเอียดที่ได้จากวงจรแปลงระดับสัญญาณอนาล็อกเป็นดิจิทัล

ขั้นตอนการใช้งาน

1. ก่อนชั่งน้ำหนักเครื่องคอมพิวเตอร์จะแสดงหน้าจอคั่งรูปที่ 5.4
2. กรอกชื่อลงในช่องว่างนี้ เช่น
ชื่อ :
3. กรอกส่วนสูงลงในช่องว่างนี้ เช่น
ส่วนสูง : cm.
4. เลือกเพศ เช่น
เพศ : ชาย หญิง
5. ชั่งน้ำหนัก แล้วกด Enter
6. เครื่องคอมพิวเตอร์จะแสดงน้ำหนักและเกณฑ์ของผู้ชั่ง คั่งรูป 5.5
7. ปุ่ม Print เมื่อกดปุ่มนี้เครื่องจะทำการพิมพ์ข้อมูลของผู้ชั่งออกมา

8. ปุ่ม Clear เมื่อกดปุ่มนี้จะเป็นการ Clear หน้าจอพร้อมที่จะทำการชั่งน้ำหนักต่อไป

9. ปุ่ม Exit เมื่อกดปุ่มนี้จะเป็นการออกจากโปรแกรมเครื่องชั่งน้ำหนักนี้



10. ปุ่ม ข้อมูล เมื่อกดปุ่มนี้จะเป็นการเข้าไปดูรายละเอียดข้อมูลของผู้ชั่งทุกคนที่ชั่งน้ำหนักที่เครื่องนี้ ดังรูปที่ 5.6



11. เมื่อต้องการค้นหาข้อมูลของผู้ชั่งสามารถทำได้ดังนี้

11.1 เลือกรายการที่ต้องการจะค้นหาในช่อง Field เช่น ชื่อ น้ำหนัก ส่วนสูง เพศ ใดๆอย่างหนึ่ง เช่น

Field :

11.2 เลือกและกรอกรายการที่ต้องการค้นหาลงในช่อง Value เช่น ต้องการ ดูข้อมูลของคนที่สูงมากกว่าหรือเท่ากับ 165 cm. ทำได้ดังนี้

Value :

11.3 กดปุ่มค้นหา เครื่องจะทำการค้นหาข้อมูลและแสดงผลการค้นหา ดังรูปที่ 5.7



เครื่องชั่งน้ำหนักด้วยไมโครคอมพิวเตอร์

ภาควิชาเทคโนโลยีการวัดคุมทางอุตสาหกรรม

คณะวิศวกรรมศาสตร์สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

20/4/00

16:23:18

กรุณากรอกข้อมูล แล้วขึ้นชั่ง สักครั้ง กด Enter

ชื่อ :

ส่วนสูง : cm.

เพศ : ชาย หญิง



รูปที่ 5.4 แสดงหน้าจอก่อนจะทำการชั่งน้ำหนัก

โปรแกรม



เครื่องชั่งน้ำหนักด้วยไมโครคอมพิวเตอร์

ภาควิชาเทคโนโลยีการวัดคุมทางอุตสาหกรรม

คณะวิศวกรรมศาสตร์สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

20/4/00

16:24:39

กรุณารอกข้อมูล แล้วขึ้นชั่ง สักตุ้กด Enter

ชื่อ : ส่วนสูง : cm.

เพศ : ชาย หญิง

น้ำหนักของคุณ = 60.70 Kg.

อยู่ในเกณฑ์ ผอมมาก คุณควรมีน้ำหนัก = 76 Kg.



รูปที่ 5.5 แสดงหน้าจอผลการทำงานของเครื่องชั่ง

ข้อมูล

Field : 

Value : =

ค้นหา

จำนวนข้อมูล :

Date	Time	Name	Sex	High	Weight	Status
20/4/00	15:39:31	Boy	ชาย	100	20.00	
20/4/00	15:40:10	Jum	หญิง	114	30.50	
20/4/00	15:41:08	วรรณ จันทรัมย์	หญิง	145	40.30	อ้วนมาก
20/4/00	15:41:49	Pop	ชาย	150	49.80	ผอมมาก
20/4/00	15:42:45	สมชาย	ชาย	156	80.00	อ้วนมาก
20/4/00	15:43:54	สมปอง	ชาย	175	89.80	อ้วนมาก
20/4/00	15:44:53	จริยา เด็กดี	หญิง	174	70.10	ค่อนข้างอ้วน
20/4/00	15:47:33	ลูกเกด	หญิง	165	55.20	ปกติ
20/4/00	15:50:09	นาวิ วงศ์กระจ่าง	ชาย	176	74.50	ปกติ
20/4/00	15:51:12	นพ พรธานี	ชาย	178	84.70	ปกติ
20/4/00	16:11:08	t	ชาย	156	50.90	ค่อนข้างผอม
20/4/00	16:12:51	p	หญิง	152	50.50	อ้วนมาก

Delete

กลับหน้าหลัก



รูปที่ 5.6 แสดงหน้าจอการจับเก็บข้อมูล

ข้อมูล

Field : High

Value : >= 165

High >='165'

ค้นหา

จำนวนข้อมูล : 7

Date	Time	Name	Sex	High	Weight	Status
20/4/00	15:43:54	สมปอง	ชาย	175	89.80	อ้วนมาก
20/4/00	15:44:53	จรรยา เด็กดี	หญิง	174	70.10	ค่อนข้างอ้วน
20/4/00	15:47:33	อุกเกศ	หญิง	165	55.20	ปกติ
20/4/00	15:50:09	นาวิ วงศ์กระจำ	ชาย	176	74.50	ปกติ
20/4/00	15:51:12	นพ พรธานี	ชาย	178	84.70	ปกติ
20/4/00	16:15:41	อรรมพ ทวีทรัพย์	ชาย	175	61.10	ผอมมาก
20/4/00	16:24:20	อรรมพ ทวีทรัพย์	ชาย	176	60.70	ผอมมาก

Delete

กลับหน้าหลัก

รูปที่ 5.7 แสดงหน้าจอผลการค้นหาข้อมูล

บทที่ 6

สรุปผลและแนวทางปฏิบัติ

สรุปผลการทดลอง

จากการทดลองชั่งน้ำหนัก ค่าน้ำหนักที่ได้นั้นเกิดความคลาดเคลื่อนขึ้นเนื่องมาจากมีสัญญาณรบกวนภายในวงจรซึ่งสามารถลดสัญญาณรบกวนนี้ได้ระดับหนึ่งแต่ก็ยังมีสัญญาณรบกวนเหลือเพียงเล็กน้อย

1. เครื่องชั่งนี้สามารถชั่งน้ำหนักได้ ตั้งแต่ 0 - 100 กิโลกรัม
2. สามารถบอกลักษณะรูปร่างของผู้ชั่งซึ่งเทียบจาก ส่วนสูงและน้ำหนัก
3. สามารถแสดงผลทางจอคอมพิวเตอร์ ทางเสียงและPrintข้อมูลออกมาได้
4. สามารถเก็บข้อมูลของผู้ชั่งได้

ปัญหาที่พบในการทำงาน

1. เกิดความล่าช้าในการทำงาน เนื่องจาก Load Cell จัดหามาช้ามาก
2. ส่วนแปลงสัญญาณอนาล็อกเป็นดิจิตอล ใช้ตัวแปลงสัญญาณ A/D ขนาด 8 บิต ซึ่งให้ค่าความละเอียดเพียง 256 ค่า มีผลให้ค่าน้ำหนักที่ได้ไม่ละเอียดพอ และเกิดค่าผิดพลาดได้ง่าย
3. ระยะเวลาในการทำงานของส่วนขยายสัญญาณมีน้อยมาก เพราะต้องทำภายหลังจากทราบข้อมูล และรายละเอียดของ Load Cell
4. เนื่องจากค่าเอาต์พุตที่ถูกส่งมาจากโพลคเซลล์มีค่าน้อยมาก จึงทำให้เกิดสัญญาณรบกวน

แนวทางในการพัฒนา

โครงการนี้สามารถพัฒนาเพิ่มเติม ได้ในส่วนของเซ็นเซอร์วัดความสูง เพื่อความสมบูรณ์และความสะดวกในการใช้งานของเครื่องชั่งน้ำหนัก

ภาคผนวก

ภาคผนวก ก

Source Code ของโปรแกรม Delphi และไมโครคอนโทรลเลอร์

Source Code ของโปรแกรม Delphi

```
unit UnitProject;
```

```
interface
```

```
uses
```

```
Windows, Messages, SysUtils, Classes, Graphics, Controls, Forms, Dialogs,  
StdCtrls, Buttons, ComCtrls, Db, DBTables, ExtCtrls, MPlayer, Comms;
```

```
const
```

```
    You=0;      endYou=2100;  
    first=30000; endfirst=31000;  
    one=2700;   endone=3300;  
    two=4200;   endtwo=5000;  
    three=5900; endthree=6600;  
    four=7500;  endfour=8600;  
    five=9100;  endfive=10000;  
    six=10500;  endsix=11500;  
    seven=11800; endseven=12800;  
    eight=13500; endeight=14300;  
    nine=14900; endnine=16100;  
    ten=16500;  endten=17500;  
    Ed=18100;   endEd=19200;  
    Yee=19700;  endYee=20600;  
    Jud=21000;  endJud=22000;  
    Zero=22650; endZero=23500;  
    Roi=23600;  endRoi=24700;  
    Kg=24800;   endKg=26500;  
    khun=26700; endkhun=28400;  
    veryfat=28600; endveryfat=29900;  
    fat=30000;  endfat=31400;  
    normal=31500; endnormal=32600;  
    slim=32650; endslim=34400;
```

```
veryslim=35000; endveryslim=36500;
```

```
empty=37000; endempty=38000;
```

```
type
```

```
TFrmProject = class(TForm)
```

```
Label1: TLabel;
```

```
Label2: TLabel;
```

```
Label3: TLabel;
```

```
Label6: TLabel;
```

```
lblName: TLabel;
```

```
edtName: TEdit;
```

```
lblHigh: TLabel;
```

```
edtHigh: TEdit;
```

```
lblCm: TLabel;
```

```
lblSex: TLabel;
```

```
btnCalc: TButton;
```

```
btnData: TButton;
```

```
btnClear: TButton;
```

```
btnExit: TButton;
```

```
lblShowDetail1: TLabel;
```

```
lblShowWeight1: TLabel;
```

```
RbtnMale: TRadioButton;
```

```
RbtnFemale: TRadioButton;
```

```
lblShowDetail2: TLabel;
```

```
lblShowWeight2: TLabel;
```

```
lblShowDetail3: TLabel;
```

```
DataSource1: TDataSource;
```

```
Table1: TTable;
```

```
lblShowWeight3: TLabel;
```

```
lblSHowDetail4: TLabel;
```

```
Label4: TLabel;
```

```
Label5: TLabel;
```

```
Timer1: TTimer;
```

```
Label7: TLabel;  
btnPrint: TButton;  
MediaPlayer1: TMediaPlayer;  
DataSource2: TDataSource;  
Table2: TTable;  
ComPort1: TComPort;  
Label8: TLabel;  
procedure btnCalcClick(Sender: TObject);  
procedure btnClearClick(Sender: TObject);  
procedure btnExitClick(Sender: TObject);  
Procedure CallMale;  
procedure CallFemale;  
procedure RbtnMaleClick(Sender: TObject);  
procedure RbtnFemaleClick(Sender: TObject);  
procedure btnDataClick(Sender: TObject);  
procedure Timer1Timer(Sender: TObject);  
procedure FormClose(Sender: TObject; var Action: TCloseAction);  
procedure btnPrintClick(Sender: TObject);  
procedure ComPort1RxChar(Sender: TObject; InQue: Integer);  
procedure FormActivate(Sender: TObject);
```

```
private
```

```
    procedure translator(number:real;var a1,b1,c1,d1:integer);  
    procedure sYou;  
    procedure sfirst;  
    procedure sone;  
    procedure stwo;  
    procedure sthree;  
    procedure sfour;  
    procedure sfive;  
    procedure ssix;  
    procedure sseven;
```

```
    procedure seight;
    procedure snine;
    procedure sten;
    procedure sEd;
    procedure sYee;
    procedure sJud;
    procedure sZero;
    procedure sKg;
    procedure sRoi;
    procedure AA(a1:integer);
    procedure BB(b1:integer);
    procedure CC(c1:integer);
    procedure DD(d1:integer);
    procedure Female;
    procedure Male;
    procedure sveryfat;
    procedure sfat;
    procedure snormal;
    procedure sslim;
    procedure sveryslim;
    procedure sempty;
    { Private declarations }
public
    { Public declarations }
end;

var
    FrmProject: TFrmProject;

implementation

uses Unit1, Unit2;
```

```
{SR *.DFM}

procedure TfrmProject.empty;
begin
    with MediaPlayer1 do
        begin
            Position:= empty;
            EndPos:= endempty;
            Play;
            end;
        end;
    procedure TfrmProject.sYou;
    begin
        with MediaPlayer1 do
            begin
                Position:= You;
                EndPos:= endYou;
                Play;
                end;
            end;
        procedure TfrmProject.sRoi;
        begin
            with MediaPlayer1 do
                begin
                    Position:= Roi;
                    EndPos:= endRoi;
                    Play;
                    end;
                end;
            procedure TfrmProject.sKg;
            begin
                with MediaPlayer1 do
```

```
        begin
            Position:= Kg;
            EndPos:= endKg;
            Play;
            end;
end;
procedure TfrmProject.sfirst;
begin
    with MediaPlayer1 do
        begin
            Position:= first;
            EndPos:= endfirst;
            Play;
            end;
end;

procedure TfrmProject.sone;
begin
    with MediaPlayer1 do
        begin
            Position:= one;
            EndPos:= endone;
            Play;
            end;
end;

procedure TfrmProject.stwo;
begin
    with MediaPlayer1 do
        begin
            Position:= two;
            EndPos:= endtwo;
            Play;
```

```
        end;
    end;
    procedure TfrmProject.sthree;
    begin
    with MediaPlayer1 do
        begin
            Position:= three;
            EndPos:= endthree;
            Play;
        end;
    end;
    procedure TfrmProject.sfour;
    begin
    with MediaPlayer1 do
        begin
            Position:= four;
            EndPos:= endfour;
            Play;
        end;
    end;
    procedure TfrmProject.sfive;
    begin
    with MediaPlayer1 do
        begin
            Position:= five;
            EndPos:= endfive;
            Play;
        end;
    end;
    procedure TfrmProject.ssix;
    begin
    with MediaPlayer1 do
```

```
begin
    Position:= six;
    EndPos:= endsix;
    Play;
end;

end;

procedure TfrmProject.sseven;
begin
with MediaPlayer1 do
    begin
        Position:= seven;
        EndPos:= endseven;
        Play;
    end;
end;

procedure TfrmProject.seight;
begin
with MediaPlayer1 do
    begin
        Position:= eight;
        EndPos:= endeight;
        Play;
    end;
end;

procedure TfrmProject.snine;
begin
with MediaPlayer1 do
    begin
        Position:= nine;
        EndPos:= endnine;
        Play;
    end;
end;
```

```
end;
procedure TfrmProject.sten;
begin
with MediaPlayer1 do
begin
Position:= ten;
EndPos:= endten;
Play;
end;
end;
procedure TfrmProject.sEd;
begin
with MediaPlayer1 do
begin
Position:= Ed;
EndPos:= endEd;
Play;
end;
end;
procedure TfrmProject.sYee;
begin
with MediaPlayer1 do
begin
Position:= Yee;
EndPos:= endYee;
Play;
end;
end;
procedure TfrmProject.sJud;
begin
with MediaPlayer1 do
begin
```

```
        Position:= Jud;
        EndPos:= endJud;
        Play;
    end;
end;

procedure TfrmProject.sZero;
begin
with MediaPlayer1 do
    begin
        Position:= Zero;
        EndPos:= endZero;
        Play;
    end;
end;

procedure TfrmProject.sveryfat;
begin
with MediaPlayer1 do
    begin
        Position:= veryfat;
        EndPos:= endveryfat;
        Play;
    end;
end;

procedure TfrmProject.sfat;
begin
with MediaPlayer1 do
    begin
        Position:= fat;
        EndPos:= endfat;
        Play;
    end;
end;
```

```
end;

procedure TfrmProject.snormal;
begin
with MediaPlayer1 do
begin
Position:= normal;
EndPos:= endnormal;
Play;
end;
end;

procedure TfrmProject.sslim;
begin
with MediaPlayer1 do
begin
Position:= slim ;
EndPos:= endslim;
Play;
end;
end;

procedure TfrmProject.sveryslim;
begin
with MediaPlayer1 do
begin
Position:= veryslim ;
EndPos:= endveryslim;
Play;
end;
end;

procedure TfrmProject.AA(a1:integer);
begin
if a1 =1 then sfirst else
```

```

    if a1 =2 then sYee else
    if a1 =3 then sthree else
    if a1=4 then sfour else
    if a1=5 then sfive else
    if a1=6 then ssix else
    if a1=7 then sseven else
    if a1=8 then seight else
    if a1=9 then snine;
end;
procedure TfrmProject.BB(b1:integer);
begin
    if b1=1 then sEd else
    if b1 =2 then stwo else
    if b1 =3 then sthree else
    if b1=4 then sfour else
    if b1=5 then sfive else
    if b1=6 then ssix else
    if b1=7 then sseven else
    if b1=8 then seight else
    if b1=9 then snine;
end;
procedure TfrmProject.CC(c1:integer);
begin
    if c1=0 then sZero else
    if c1=1 then sone else
    if c1 =2 then stwo else
    if c1 =3 then sthree else
    if c1=4 then sfour else
    if c1=5 then sfive else
    if c1=6 then ssix else
    if c1=7 then sseven else
    if c1=8 then seight else

```

```

        if c1=9 then snine;
end;
procedure TfrmProject.DD(d1:integer);
begin
    if d1=0 then sZero else
    if d1=1 then sone else
    if d1 =2 then stwo else
    if d1 =3 then sthree else
    if d1=4 then sfour else
    if d1=5 then sfive else
    if d1=6 then ssix else
    if d1=7 then sseven else
    if d1=8 then seight else
    if d1=9 then snine;
end;
procedure TfrmProject.Male;
Var
    High,Weight,L,Score:integer;
begin
    Label7.Caption:=Rbtnmale.Caption;
    L:=StrToInt(label8.caption)*100;
    Weight:=L div 255;
    High:=StrToInt(edtHigh.text);
    if High < 140 then lblShowDetail2.caption:="
else
begin
Score:=High - Weight;
case Score of
    50..91:sveryfat;
    92..97:sfat;
    98..102:snormal;
    103..108:sslim;

```

```

        109..150:sveryslim;
    else
    end;
end;
end;
procedure TfrmProject.Female;
Var
    High,Weight,L,Score:integer;
begin
    Label7.Caption:=RbtnFemale.Caption;
    L:=StrToInt(label8.caption)*100;
    Weight:=L div 255;
    High:=StrToInt(edtHigh.text);
    if High < 140 then lblShowDetail2.caption:="
    else
    begin
        Score:=High - Weight;
        case Score of
            50..102:sveryfat;
            103..107:sfat;
            108..112:snormal;
            113..117:sslim;
            118..140:sveryslim;
        else
        end;
    end;
end;
end;
procedure TfrmProject.translator(number:real;var a1,b1,c1,d1:integer);
var number2,number3:integer;number1:real;
begin
    number2:=trunc(number);
    a1:=number2 div 10;

```

```
b1:=number2 mod 10;
number1:=(number - number2)*100;
number3:=trunc(number1+1);
c1:=number3 div 10;
d1:=number3 mod 10;
a1:=number2 div 10;
end;
```

```
Procedure TFrmProject.CallFemale;
```

```
Var
```

```
High,Weight,L,Score:integer;
```

```
begin
```

```
Label7.Caption:=RbtnFemale.Caption;
```

```
L:=StrToInt(label8.caption)*100;
```

```
Weight:=L div 255;
```

```
High:=StrToInt(edtHigh.text);
```

```
if High < 140 then lblShowDetail2.caption:="
```

```
else
```

```
begin
```

```
Score:=High - Weight;
```

```
case Score of
```

```
50..102:lblShowDetail2.Caption:=( ' อ้วนมาก');
```

```
103..107:lblShowDetail2.Caption:=( ' ค่อนข้างอ้วน');
```

```
108..112:lblShowDetail2.Caption:=( ' ปกติ');
```

```
113..117:lblShowDetail2.Caption:=( ' ค่อนข้างผอม');
```

```
118..140:lblShowDetail2.Caption:=( ' ผอมมาก');
```

```
else
```

```
MessageDlg('ตรวจสอบข้อมูลส่วนสูงใหม่กดปุ่ม CLEAR เพื่อกรอกข้อมูลอีก  
ครั้ง',mtWarning,[mbOK],0);
```

```
end;
```

```
begin
```

```
if Score >140 then begin
```

```

        lblShowDetail1.Caption:="";
        lblShowDetail3.Caption:="";
        lblShowDetail4.Caption:="";
    end
    else begin
        lblShowDetail1.Caption:=(อยู่ในเกณฑ์      '+' '+'คุณควรจะมีน้ำ
หน้าก = '      ');
        lblShowDetail3.Caption:=(IntToStr(High-110));
        lblShowDetail4.Caption:='Kg.'
    end;
end;
end;
end;

```

```

Procedure TFrmProject.CallMale;

```

```

Var

```

```

    High,Weight,L,Score:integer;

```

```

begin

```

```

    Label7.Caption:=Rbtnmale.Caption;

```

```

    L:=StrToInt(label8.caption)*100;

```

```

    Weight:=L div 255;

```

```

    High:=StrToInt(edtHigh.text);

```

```

    if High < 140 then lblShowDetail2.caption:="

```

```

else

```

```

begin

```

```

    Score:=High - Weight;

```

```

    case Score of

```

```

        50..91:lblShowDetail2.Caption:=( ' อ้วนมาก');

```

```

        92..97:lblShowDetail2.Caption:=( ' ค่อนข้างอ้วน');

```

```

        98..102:lblShowDetail2.Caption:=( ' ปกติ');

```

```

        103..108:lblShowDetail2.Caption:=( ' ค่อนข้างผอม');

```

```

        109..150:lblShowDetail2.Caption:=( ' ผอมมาก');

```

```

else
    MessageDlg('ตรวจสอบข้อมูลส่วนสูงใหม่กดปุ่ม CLEAR เพื่อกรอกข้อมูลอีก
ครั้ง',mtWarning,[mbOK],0);
end;

begin
    if Score >150 then begin
        lblShowDetail1.Caption:="";
        lblShowDetail3.Caption:="";
        lblShowDetail4.Caption:="";
    end
    else begin
        lblShowDetail1.Caption:=(อยู่ในเกณฑ์      '+' '+'คุณควรจะมีน้ำ
หนัก = '      ');
        lblShowDetail3.Caption:=(IntToStr(High-100));
        lblShowDetail4.Caption:='Kg.'
    end;
end;
end;
end;

```

```

procedure TFrmProject.btnCalcClick(Sender: TObject);
var
    A,B,C,D,F,L,Weight:integer;
    i,a1,b1,c1,d1: integer;xxx:real;
begin
    L:=StrToInt(label8.caption)*100;
    Weight:=l div 255;
    A:=l mod 255;
    A:=A*10;
    A:=A div 255;
    D:=A;
    B:=L mod 255;

```

```

B:=B*10;
C:=B;
F:=D*255;
C:=C-F;
C:=C*10;
C:=c div 255;
lblShowWeight1.Caption:=(น้ำหนักของคุณ =);
lblShowWeight2.Caption:=(IntToStr(Weight)+'.'+IntToStr(A)+IntToStr(C));
lblShowWeight3.Caption:= 'Kg.';
xxx:=strtfloat(lblShowWeight2.caption);
translator(xxx,a1,b1,c1,d1);
sYou;
while MediaPlayer1.Mode=mpPlaying do
begin i:= i+1; if i=100 then i:=1; end;
if xxx=0.00 then begin sZero;
while MediaPlayer1.Mode=mpPlaying do
begin i:= i+1; if i=100 then i:=1; end;
SkG;
end else begin
if xxx=100.00 then begin sone;
while MediaPlayer1.Mode=mpPlaying do
begin i:= i+1; if i=100 then i:=1; end;
sRoi ;
while MediaPlayer1.Mode=mpPlaying do
begin i:= i+1; if i=100 then i:=1; end;
skg;
while MediaPlayer1.Mode=mpPlaying do
begin i:= i+1; if i=100 then i:=1; end;
if RbtnfeMale.Checked=true then Female
else Male;
end else begin
AA(a1);

```

```

while MediaPlayer1.Mode=mpPlaying do
begin i:= i+1; if i=100 then i:=1; end;
sten;
while MediaPlayer1.Mode=mpPlaying do
begin i:= i+1; if i=100 then i:=1; end;
BB(b1);
while MediaPlayer1.Mode=mpPlaying do
begin i:= i+1; if i=100 then i:=1; end;
sJud;
while MediaPlayer1.Mode=mpPlaying do
begin i:= i+1; if i=100 then i:=1; end;
CC(c1);
while MediaPlayer1.Mode=mpPlaying do
begin i:= i+1; if i=100 then i:=1; end;
DD(d1);
while MediaPlayer1.Mode=mpPlaying do
begin i:= i+1; if i=100 then i:=1; end;
sKg;
while MediaPlayer1.Mode=mpPlaying do
begin i:= i+1; if i=100 then i:=1; end;
if RbtnfeMale.Checked=true then Female
else Male;
end;
end;
begin
btnExit.Enabled:=false;
btnPrint.Enabled:=true;
btnClear.Enabled:=true;
btnCalc.Enabled:=false;
if RbtnMale.Checked=true then CallMale
else CallFemale;
begin

```

```

    Table1.Append;
    Table1['Date']:=Date;
    Table1['Time']:=Time;
    Table1['Name']:=edtName.Text;
    Table1['Sex']:=Label7.Caption;
    Table1['High']:=edtHigh.Text;
    Table1['Weight']:=lblShowWeight2.Caption;
    Table1['Status']:=lblShowDetail2.Caption;
    Table1.Post;
end;
begin
    Table2.Append;
    Table2['Date']:=Date;
    Table2['Time']:=Time;
    Table2['Name']:=edtName.Text;
    Table2['Sex']:=Label7.Caption;
    Table2['High']:=edtHigh.Text;
    Table2['Weight']:=lblShowWeight2.Caption;
    Table2['Status']:=lblShowDetail2.Caption;
    Table2.Post;
end;
end;
procedure TFrmProject.btnClearClick(Sender: TObject);
begin
    edtName.clear;
    edtHigh.clear;
    btnCLear.Enabled:=false;
    lblShowWeight1.caption:="";
    lblShowWeight2.Caption:="";
    lblShowWeight3.Caption:="";

```

```
    lblShowDetail1.Caption:="";
    lblShowDetail2.Caption:="";
    lblShowDetail3.Caption:="";
    lblShowDetail4.Caption:="";
    btnCalc.Enabled:=false;
    RbtnMale.Checked:=false;
    RbtnFemale.Checked:=false;
    btnExit.Enabled:=true;
    btnPrint.Enabled:=false;
    Table2.Delete;
end;
procedure TFrmProject.btnExitClick(Sender: TObject);
begin
    if MessageDlg('Are you sure you want to quit this project?',mtConfirmation,
    [mbYes,mbNo],0)=mryes then
        close;
end;
procedure TFrmProject.RbtnMaleClick(Sender: TObject);
begin
    btnCalc.Enabled:=true;
end;
procedure TFrmProject.RbtnFemaleClick(Sender: TObject);
begin
    btnCalc.Enabled:=true;
end;
procedure TFrmProject.btnDataClick(Sender: TObject);
begin
    frmProject.Hide;
    frmData.Show;
    comport1.open;
    btnExit.Enabled:=true;
end;
```

```

procedure TFrmProject.Timer1Timer(Sender: TObject);
begin
    Label4.Caption:=DateToStr(Date);
    Label5.Caption:=TimeToStr(Time);
end;

procedure TFrmProject.FormClose(Sender: TObject; var Action: TCloseAction);
begin
    MediaPlayer1.Destroy;
    comport1.Close;
end;

procedure TFrmProject.btnPrintClick(Sender: TObject);
begin

    frmPrint1.QuickRep1.Print;
end;

procedure TFrmProject.ComPort1RxChar(Sender: TObject; InQue: Integer);
var i,j:integer;str:string;
begin
    if comport1.connected then
        begin
            while inque>0 do
                begin
                    i:=comport1.read(j,1,True);
                    label8.caption:= inttostr(j);
                    inque:=inque-1;
                end;
            end;
        end;
end;
end;

```

```
procedure TFrmProject.FormActivate(Sender: TObject);  
begin  
    comport1.Open;  
end;  
end.
```

```
unit Unit1;
```

```
interface
```

```
uses
```

```
Windows, Messages, SysUtils, Classes, Graphics, Controls, Forms, Dialogs,  
StdCtrls, Grids, DBGrids, Db, DBTables, ExtCtrls, DBCtrls, ComCtrls;
```

```
type
```

```
TfrmData = class(TForm)
```

```
  DataSource1: TDataSource;
```

```
  Table1: TTable;
```

```
  DBGrid1: TDBGrid;
```

```
  Button1: TButton;
```

```
  Label1: TLabel;
```

```
  Label2: TLabel;
```

```
  Edit1: TEdit;
```

```
  btnFind: TButton;
```

```
  ComboBox1: TComboBox;
```

```
  btnDelete: TButton;
```

```
  Animate1: TAnimate;
```

```
  ComboBox2: TComboBox;
```

```
  Edit2: TEdit;
```

```
  Edit3: TEdit;
```

```
  Label3: TLabel;
```

```
  DataSource2: TDataSource;
```

```
  Table2: TTable;
```

```
  DBGrid2: TDBGrid;
```

```
  procedure Button1Click(Sender: TObject);
```

```
  procedure btnFindClick(Sender: TObject);
```

```
  procedure FormCreate(Sender: TObject);
```

```
  procedure btnDeleteClick(Sender: TObject);
```

```
  procedure FormActivate(Sender: TObject);
```

```
  procedure ComboBox1Change(Sender: TObject);
```

```
private
    { Private declarations }
public
    { Public declarations }
end;

var
    frmData: TfrmData;

implementation

uses UnitProject;

{$R *.DFM}

procedure TfrmData.Button1Click(Sender: TObject);
begin
    frmProject.Show;
    Edit2.Visible:=false;
    Edit1.Clear;
end;

procedure TfrmData.btnFindClick(Sender: TObject);
begin
    Animate1.Active:=True;
    DBGrid1.Visible:=false;
    DBGrid2.Visible:=true;
    Table2.Filter:=ComboBox1.Text+ComboBox2.Text+""+Edit1.Text+"";
    Edit2.Text:=ComboBox1.Text+ComboBox2.Text+""+Edit1.Text+"";
    Edit2.Visible:=true;
    btnDelete.Enabled:=false;
    if DBGrid1.Visible=true then Edit3.Text:=IntToStr(Table1.RecordCount)
    else Edit3.Text:=IntToStr(Table2.RecordCount)
end;
```

```
procedure TfrmData.FormCreate(Sender: TObject);
begin
    ComboBox1.ItemIndex:=1;
    ComboBox2.ItemIndex:=0;
end;
```

```
procedure TfrmData.btnDeleteClick(Sender: TObject);
begin
    if MessageDlg('Delete record',mtConfirmation,[mbOK,mbCancel],0)=mrOK then
        Table1.Delete;
    if DBGrid1.Visible=true then Edit3.Text:=inttostr(Table1.RecordCount)
    else Edit3.Text:=inttostr(Table2.RecordCount);
end;
```

```
procedure TfrmData.FormActivate(Sender: TObject);
begin
    ComboBox1.ItemIndex:=1;
    ComboBox2.ItemIndex:=0;
    Table1.Refresh;
    DBGrid1.Visible:=true;
    DBGrid2.Visible:=false;
    btnDelete.Enabled:=true;
    if DBGrid1.Visible=true then Edit3.Text:=inttostr(Table1.RecordCount)
    else Edit3.Text:=inttostr(Table2.RecordCount);
end;
```

```
procedure TfrmData.ComboBox1Change(Sender: TObject);
begin
    if ComboBox1.ItemIndex = 0 then
        Edit1.Text:= ' / / ' else Edit1.Clear;
end;
```

```
end.
```

unit Unit2;

interface

uses

Windows, Messages, SysUtils, Classes, Graphics, Controls, Forms, Dialogs,
StdCtrls, ExtCtrls, QuickRpt, Db, DBTables, Qrctrls;

type

TfmPrint1 = class(TForm)

Table1: TTable;

QuickRep1: TQuickRep;

PageHeaderBand1: TQRBand;

TitleBand1: TQRBand;

ColumnHeaderBand1: TQRBand;

DetailBand1: TQRBand;

PageFooterBand1: TQRBand;

QRLabel1: TQRLabel;

QRLabel2: TQRLabel;

QRLabel3: TQRLabel;

QRLabel4: TQRLabel;

QRLabel5: TQRLabel;

QRLabel6: TQRLabel;

QRLabel7: TQRLabel;

QRLabel8: TQRLabel;

QRDBText1: TQRDBText;

QRDBText2: TQRDBText;

QRDBText3: TQRDBText;

QRDBText4: TQRDBText;

QRDBText5: TQRDBText;

QRDBText6: TQRDBText;

QRDBText7: TQRDBText;

QRSysData1: TQRSysData;

private

```
    { Private declarations }  
public  
    { Public declarations }  
end;  
  
var  
    frmPrint1: TfrmPrint1;  
  
implementation  
  
uses UnitProject;  
  
{$R *.DFM}  
  
end.
```

```
program NewMyProject;
```

```
uses
```

```
  Forms,
```

```
  UnitProject in 'UnitProject.pas' {FrmProject},
```

```
  Unit1 in 'Unit1.pas' {frmData},
```

```
  Unit2 in 'Unit2.pas' {frmPrint1};
```

```
{$R *.RES}
```

```
begin
```

```
  Application.Initialize;
```

```
  Application.CreateForm(TFrmProject, FrmProject);
```

```
  Application.CreateForm(TfrmData, frmData);
```

```
  Application.CreateForm(TfrmPrint1, frmPrint1);
```

```
  Application.Run;
```

```
end.
```

โปรแกรมภาษาแอสเซมบลี (Assembly) สำหรับ AT89C51 ไมโครคอนโทรลเลอร์

.*****
,

;*START PROGRAM *

.*****
,

ORG 0000H

;START ADDRESS 0000H

MOV PCON,#00

MOV SCON,#40H

MOV TMOD,#20H

MOV TH1,#0FDH

SETB TR1

MOV A,#0

.....

LOOP: CALL ADC

MOV SBUF,A

JNB TI,\$

CLR TI

JMP LOOP

.....

ADC: CLR P2.0

CLR P2.2

NOP

NOP

SETB P2.2

JB P2.3,\$

CLR P2.1

NOP

MOV A,P0

NOP

MOV P1,A

NOP

SETB P2.1

RET

END

ภาคผนวก ข

รายละเอียดข้อมูลของไอซีทีที่ใช้ในโครงการ

PT1241

...better load cells by design

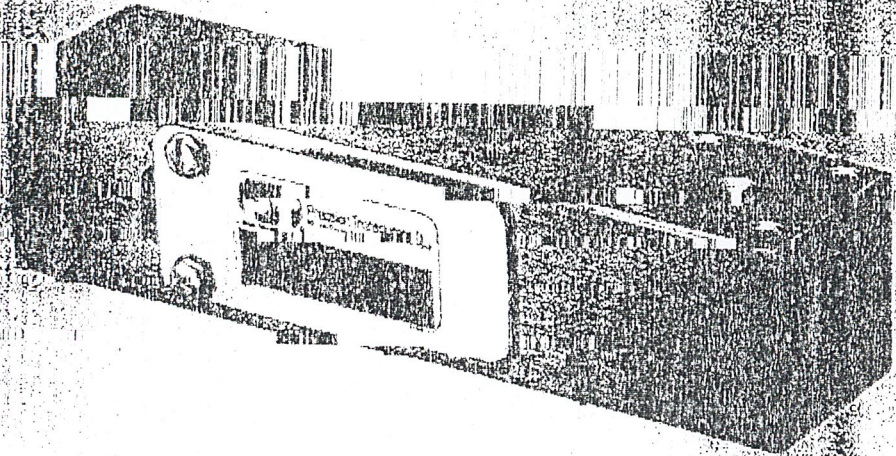
model PT1241 is a low profile, three beam, single point load cell ideally

is limited. Typical applications include platforms, hanging scales and personal weighers.

This high accuracy load cell is Factory Mutual and OIML R60 class C3 approved.

A special humidity resistant protective coating assures long term stability over the entire compensated temperature range. For washdown protection an optional IP67 encapsulation protection is available.

The two additional sense wires feed back the voltage reaching the load cell. Complete compensation of changes in lead resistance due to temperature change is achieved by feeding this voltage into the appropriate electronics.



Precision Transducers model PT1241 low cost medium capacity single point load cell

- Capacities: 50-250 kg.
- Aluminium construction.
- Single point 400x400mm platform.
- 6 wire (sense) circuit.
- IP54 protection (IP67 optional).
- OIML R60 and Factory Mutual Approved

PT1241 SPECIFICATION

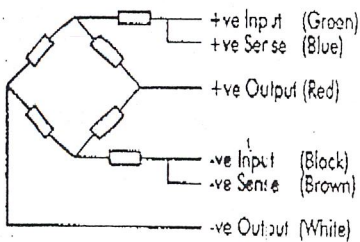
ATTN: K. BROWN Date: F. 887 2167

GRADE	Z	E	F	C3**	UNITS
Rated Capacities	50,100,150,200,250				kg
Rated Output	2.0 ± 10%				mV/V
Total Error*	0.075	0.030	0.020	0.020	±% of Applied Load
Zero Return after 30 mins	0.065	0.050	0.025	0.017	±% of Applied Load
Temperature Effect : On Output	0.0070	0.0030	0.0014	0.0010	±% of Applied Load / °C
: On Zero	0.025	0.0100	0.0060	0.0040	±% of Rated Load Output / °C
Eccentric Loading	0.015	0.0074	0.0074	0.0049	±% of Applied Load / cm.
Zero Balance	10.0				±% of Rated Output
Temperature range : Operating	-30 to +70				°C
: Compensated	-10 to +40				°C
Safe Overload	150				% of Rated Capacity
Ultimate Overload	300				% of Rated Capacity
Excitation : Recommended	10				Volts AC or DC
: Maximum	15				
Input Impedance	415 ± 15				Ohms
Output Impedance	350 ± 3				Ohms
Insulation Impedance	>2000				MegaOhms
Deflection at Rated Capacity	<0.4				mm
Weight : All Types	1.8				kg
Construction	Standard : Anodised Aluminium, Low Cost : Aluminium				
Environmental Protection	Standard : IP54 (IP67 Optional), Low Cost : IP55				
Cable	1 Metre (Standard), 6 Wire, Polyurethane Jacket, Dual Floating Screen				

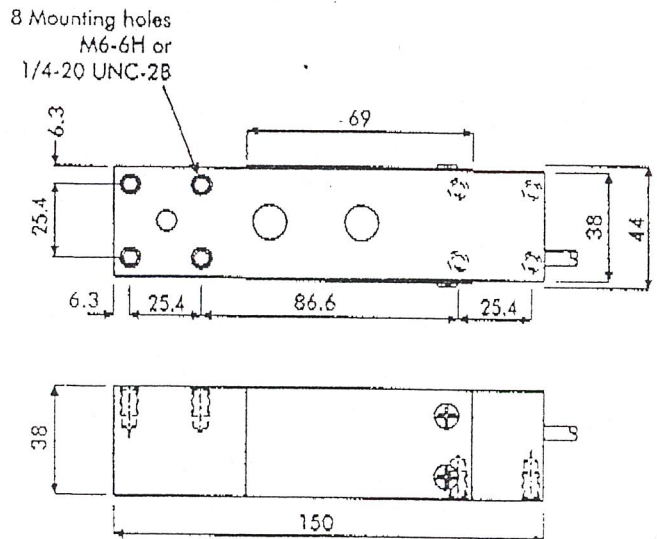
*Total Error - According to OIML F60

** Standard 85% Utilisation - Other utilisation factors available on request.

Wiring Schematic Diagram



Outline Dimensions All Capacities



Also available from Precision Transducers

A bolt hole compatible stainless steel version of the PT1241, designated the PT1240, is also available, enquire for details.

Sealed as standard to IP54, an IP67 version is also available for more demanding environmental conditions.

For low cost applications the model PT1241 can be supplied without covers (IP55), unanodised and in an unbalanced bridge configuration.

For further details please contact the factory or your local distributor.

All dimensions in mm



Precision Transducers Ltd.

Corporate Office and Manufacturing Facility
7 Marken Place, Glenfield,
P.O. Box 34009, Birkenhead,
Auckland, New Zealand.
Telephone +64 9 444-4443
Facsimile +64 9 444-9548
E-mail PreTran@ibm.net

International Sales Office
Suite 4, 31 Windsor Rd., Kellyville,
Private Bag 2, Kellyville,
NSW 2155, Australia.
Telephone +61 2 629-4018
Facsimile +61 2 629-4019
E-mail PTsales@ibm.net

Due to Precision Transducer's policy of continuous development, these specifications are subject to change without notice.

FEATURES

Low Noise: $0.3 \mu\text{V p-p}$ 0.1 Hz to 10 Hz
Low Nonlinearity: 0.003% ($G = 1$)
High CMRR: 120 dB ($G = 1000$)
Low Offset Voltage: $50 \mu\text{V}$
Low Offset Voltage Drift: $0.5 \mu\text{V}/^\circ\text{C}$
Gain Bandwidth Product: 25 MHz
Pin Programmable Gains of 1, 10, 100, 1000
Input Protection, Power On-Power Off
No External Components Required
Internally Compensated
MIL-STD-883B and Chips Available
16-Pin Ceramic DIP and SOIC Packages and
20-Terminal Leadless Chip Carriers Available
Available in Tape and Reel in Accordance
with EIA-481A Standard
Standard Military Drawing Also Available

PRODUCT DESCRIPTION

The AD524 is a precision monolithic instrumentation amplifier designed for data acquisition applications requiring high accuracy under worst-case operating conditions. An outstanding combination of high linearity, high common mode rejection, low offset voltage drift, and low noise makes the AD524 suitable for use in many data acquisition systems.

The AD524 has an output offset voltage drift of less than $25 \mu\text{V}/^\circ\text{C}$, input offset voltage drift of less than $0.5 \mu\text{V}/^\circ\text{C}$, CMR above 90 dB at unity gain (120 dB at $G = 1000$) and maximum nonlinearity of 0.003% at $G = 1$. In addition to the outstanding dc specifications the AD524 also has a 25 MHz gain bandwidth product ($G = 100$). To make it suitable for high speed data acquisition systems the AD524 has an output slew rate of $5 \text{ V}/\mu\text{s}$ and settles in $15 \mu\text{s}$ to 0.01% for gains of 1 to 100.

As a complete amplifier the AD524 does not require any external components for fixed gains of 1, 10, 100 and 1,000. For other gain settings between 1 and 1000 only a single resistor is required. The AD524 input is fully protected for both power on and power off fault conditions.

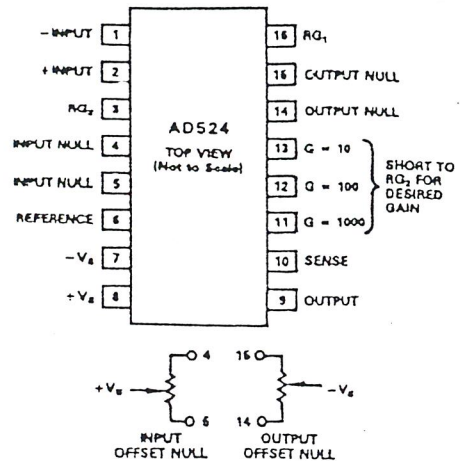
The AD524 IC instrumentation amplifier is available in four different versions of accuracy and operating temperature range. The economical "A" grade, the low drift "B" grade and lower drift, higher linearity "C" grade are specified from -25°C to $+85^\circ\text{C}$. The "S" grade guarantees performance to specification over the extended temperature range -55°C to $+125^\circ\text{C}$. Devices are available in 16-pin ceramic DIP and SOIC packages and a 20-terminal leadless chip carrier.

REV. C

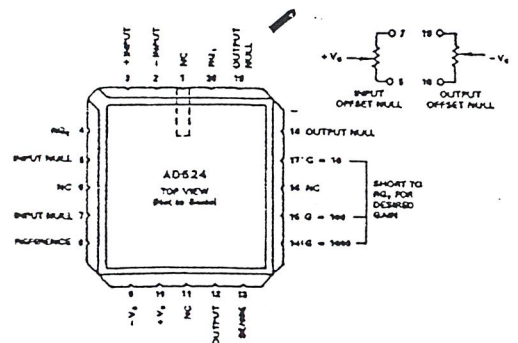
Information furnished by Analog Devices is believed to be accurate and reliable. However, no responsibility is assumed by Analog Devices for its use, nor for any infringements of patents or other rights of third parties which may result from its use. No license is granted by implication or otherwise under any patent or patent rights of Analog Devices.

CONNECTION DIAGRAMS

Ceramic (D) and
SOIC (R) Packages



Leadless Chip Carrier (E) Package



PRODUCT HIGHLIGHTS

1. The AD524 has guaranteed low offset voltage, offset voltage drift and low noise for precision high gain applications.
2. The AD524 is functionally complete with pin programmable gains of 1, 10, 100 and 1000, and single resistor programmable for any gain.
3. Input and output offset nulling terminals are provided for very high precision applications and to minimize offset voltage changes in gain ranging applications.
4. The AD524 is input protected for both power on and power off fault conditions.
5. The AD524 offers superior dynamic performance with a gain bandwidth product of 25 MHz, full power response of 75 kHz and a settling time of $15 \mu\text{s}$ to 0.01% of a 20 V step ($G = 100$)

AD524—SPECIFICATIONS (@ $V_S = \pm 15\text{ V}$, $R_L = 2\text{ k}\Omega$ and $T_A = +25^\circ\text{C}$ unless otherwise noted)

Model	AD524A			AD524B			AD524C			AD524S			Units
	Min	Typ	Max	Min	Typ	Max	Min	Typ	Max	Min	Typ	Max	
AIN													
Gain Equation (External Resistor Gain Programming)	$\left[\frac{40,000}{R_C} + 1\right] \pm 20\%$			$\left[\frac{40,000}{R_C} + 1\right] \pm 20\%$			$\left[\frac{40,000}{R_C} + 1\right] \pm 20\%$			$\left[\frac{40,000}{R_C} + 1\right] \pm 20\%$			
Gain Range (P1n Programmable)	1 to 1000			1 to 1000			1 to 1000			1 to 1000			
Gain Error ¹													
G = 1			±0.05			±0.03			±0.02			±0.05	%
G = 10			±0.25			±0.15			±0.1			±0.25	%
G = 100			±0.5			±0.35			±0.25			±0.5	%
G = 1000			±2.0			±1.0			±0.5			±2.0	%
Nonlinearity													
G = 1			±0.01			±0.005			±0.003			±0.01	%
G = 10, 100			±0.01			±0.005			±0.003			±0.01	%
G = 1000			±0.01			±0.01			±0.01			±0.01	%
Gain vs. Temperature													
G = 1			5			5			5			5	ppm/°C
G = 10			15			10			10			10	ppm/°C
G = 100			35			25			25			25	ppm/°C
G = 1000			100			50			50			50	ppm/°C
VOLTAGE OFFSET (May be Nulled)													
Input Offset Voltage vs. Temperature			250			100			50			100	μV
Output Offset Voltage vs. Temperature			2			0.75			0.5			2.0	μV/°C
Offset Referred to the Input vs. Supply			5			3			2.0			3.0	mV
G = 1			100			50			25			50	μV/°C
G = 10	70			75			80			75			dB
G = 100	85			95			100			95			dB
G = 1000	95			105			110			105			dB
G = 10000	150			110			115			110			dB
INPUT CURRENT													
Input Bias Current vs. Temperature			±50			±25			±15			±50	nA
Input Offset Current vs. Temperature		±100		±100		±15		±100		±10		±100	pA/°C
G = 1			±35			±15			±10			±35	nA
G = 1000		±100		±100		±15		±100		±10		±100	pA/°C
INPUT													
Input Impedance													
Differential Resistance		10 ⁹		10 ⁹		10 ⁹		10 ⁹		10 ⁹		10 ⁹	Ω
Differential Capacitance		10		10		10		10		10		10	pF
Common-Mode Resistance		10 ⁹		10 ⁹		10 ⁹		10 ⁹		10 ⁹		10 ⁹	Ω
Common-Mode Capacitance		10		10		10		10		10		10	pF
Input Voltage Range													
Max Differ. Input Linear (V_{D1}) ²	±10			±10			±10			±10			V
Max Common-Mode Linear (V_{CM})		$12\text{ V} - \left(\frac{G}{2} \times V_D\right)$				$12\text{ V} - \left(\frac{G}{2} \times V_D\right)$				$12\text{ V} - \left(\frac{G}{2} \times V_D\right)$			V
Common-Mode Rejection dc to 60 Hz with 1 kΩ Source Imbalance													
G = 1			70			75			80			70	dB
G = 10			90			95			100			90	dB
G = 100			100			105			110			100	dB
G = 1000			110			115			120			110	dB
OUTPUT RATING													
V_{OUT} , $R_L = 2\text{ k}\Omega$		±10		±10		±10		±10		±10		±10	V
DYNAMIC RESPONSE													
Small Signal - 3 dB													
G = 1		1		1		1		1		1		1	MHz
G = 10		400		400		400		400		400		400	kHz
G = 100		150		150		150		150		150		150	kHz
G = 1000		25		25		25		25		25		25	kHz
Slew Rate		5.0		5.0		5.0		5.0		5.0		5.0	V/μs
Settling Time to 0.01%, 20 V Step													
G = 1 to 100		15		15		15		15		15		15	μs
G = 1000		75		75		75		75		75		75	μs
NOISE													
Voltage Noise, 1 kHz													
R.T.I.		7		7		7		7		7		7	nV/√Hz
R.T.O.		90		90		90		90		90		90	nV/√Hz
R.T.I., 0.1 Hz to 10 Hz													
G = 1		15		15		15		15		15		15	μV p-p
G = 10		2		2		2		2		2		2	μV p-p
G = 100, 1000		0.3		0.3		0.3		0.3		0.3		0.3	μV p-p
Current Noise													
0.1 Hz to 10 Hz		60		60		60		60		60		60	pA p-p

Model	AD524A			AD524B			AD524C			AD524S			Units
	Min	Typ	Max	Min	Typ	Max	Min	Typ	Max	Min	Typ	Max	
SENSE INPUT													
R_{IN}		20			20			20			20		$k\Omega \pm 20\%$
I_{IN}		15			15			15			15		μA
Voltage Range	± 10			± 10			± 10			± 10			V
Gain to Output		1			1			1			1		%
REFERENCE INPUT													
R_{IN}		40			40			40			40		$k\Omega \pm 20\%$
I_{IN}		15			15			15			15		μA
Voltage Range	± 10			± 10			10			10			V
Gain to Output		1			1			1			1		%
TEMPERATURE RANGE													
Specified Performance	-25		+85	-25		+85	-25		+85	-55		+125	$^{\circ}C$
Storage	-65		+150	-65		+150	-65		+150	-65		+150	$^{\circ}C$
POWER SUPPLY													
Power Supply Range	± 6	± 15	± 18	± 6	± 15	± 18	± 6	± 15	± 18	± 6	± 15	± 18	V
Quiescent Current		3.5	5.0		3.5	5.0		3.5	5.0		3.5	5.0	mA

NOTES

- ¹Does not include effects of external resistor R_G .
- ² V_{OL} is the maximum differential input voltage at $G = 1$ for specified nonlinearity. V_{DL} at the maximum = $10V/G$.
- V_D = Actual differential input voltage.
- Example: $G = 10, V_D = 0.50$.
- $V_{CM} = 12V - (10/2 \times 0.50V) = 9.5V$.

Specification subject to change without notice.

All min and max specifications are guaranteed. Specifications shown in boldface are tested on all production units at final electrical test. Results from those tests are used to calculate outgoing quality levels.

ABSOLUTE MAXIMUM RATINGS¹

- Supply Voltage $\pm 18V$
- Internal Power Dissipation 450 mW
- Input Voltage,²
(Either Input Simultaneously) $|V_{IN}| + |V_S| < 36V$
- Output Short Circuit Duration Indefinite
- Storage Temperature Range
(R) $-65^{\circ}C$ to $+125^{\circ}C$
(D, E) $-65^{\circ}C$ to $+150^{\circ}C$
- Operating Temperature Range
AD524A/B/C $-25^{\circ}C$ to $+85^{\circ}C$
AD524S $-55^{\circ}C$ to $+125^{\circ}C$
- Lead Temperature (Soldering 60 secs) $+300^{\circ}C$

NOTES

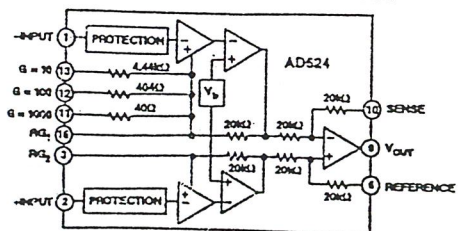
- ¹Stresses above those listed under "Absolute Maximum Ratings" may cause permanent damage to the device. This is a stress rating only and functional operation of the device at these or any other conditions above those indicated in the operational section of this specification is not implied. Exposure to absolute maximum rating conditions for extended periods may affect device reliability.
- ²Max input voltage specification refers to maximum voltage to which either input terminal may be raised with or without device power applied. For example, with $\pm 18V$ supplies max V_{IN} is $\pm 18V$, with zero supply voltage max V_{IN} is $\pm 36V$.

ORDERING GUIDE

Model	Temperature Range	Package Description	Package Option
AD524AD	$-40^{\circ}C$ to $+85^{\circ}C$	16-Pin Ceramic DIP	D-16
AD524AE	$-40^{\circ}C$ to $+85^{\circ}C$	20-Pin Leadless Chip Carrier	E-20A
AD524AR-16	$-40^{\circ}C$ to $+85^{\circ}C$	16-Pin Gull-Wing SOIC	R-16
AD524AR-16-REEL	$-40^{\circ}C$ to $+85^{\circ}C$	Tape & Reel Packaging	
AD524BD	$-40^{\circ}C$ to $+85^{\circ}C$	16-Pin Ceramic DIP	D-16
AD524BE	$-40^{\circ}C$ to $+85^{\circ}C$	20-Pin Leadless Chip Carrier	E-20A
AD524CD	$-40^{\circ}C$ to $+85^{\circ}C$	16-Pin Ceramic DIP	D-16
AD524SD	$-55^{\circ}C$ to $+125^{\circ}C$	16-Pin Ceramic DIP	D-16
AD524SD/883B	$-55^{\circ}C$ to $+125^{\circ}C$	16-Pin Ceramic DIP	D-16
5962-8853901EA ¹	$-55^{\circ}C$ to $+125^{\circ}C$	16-Pin Ceramic DIP	D-16
AD524SE/883B	$-55^{\circ}C$ to $+125^{\circ}C$	20-Pin Leadless Chip Carrier	E-20A
AD524ACHIPS	$-40^{\circ}C$ to $+85^{\circ}C$	Die	
AD524SCHIPS	$-55^{\circ}C$ to $+125^{\circ}C$	Die	

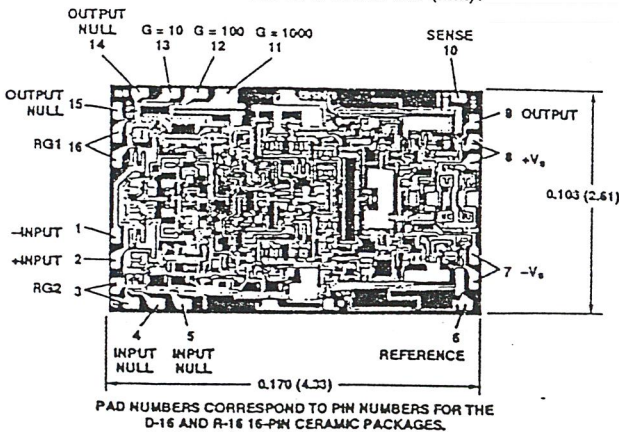
¹Refer to official DESC drawing for tested specifications.

FUNCTIONAL BLOCK DIAGRAM



METALIZATION PHOTOGRAPH

Contact factory for latest dimensions. Dimensions shown in inches and (mm).



PAD NUMBERS CORRESPOND TO PIN NUMBERS FOR THE D-16 AND R-16 16-PIN CERAMIC PACKAGES.

CAUTION

ESD (electrostatic discharge) sensitive device. Electrostatic charges as high as 4000 V readily accumulate on the human body and test equipment and can discharge without detection. Although the AD524 features proprietary ESD protection circuitry, permanent damage may occur on devices subjected to high energy electrostatic discharges. Therefore, proper ESD precautions are recommended to avoid performance degradation or loss of functionality.



INPUT OFFSET AND OUTPUT OFFSET

Voltage offset specifications are often considered a figure of merit for instrumentation amplifiers. While initial offset may be adjusted to zero, shifts in offset voltage due to temperature variations will cause errors. Intelligent systems can often correct for this factor with an auto-zero cycle, but there are many small-signal high-gain applications that don't have this capability.

Voltage offset and drift comprise two components each; input and output offset and offset drift. Input offset is that component of offset that is directly proportional to gain i.e., input offset as measured at the output at $G = 100$ is 100 times greater than at $G = 1$. Output offset is independent of gain. At low gains, output offset drift is dominant, while at high gains input offset drift dominates. Therefore, the output offset voltage drift is normally specified as drift at $G = 1$ (where input effects are insignificant), while input offset voltage drift is given by drift specification at a high gain (where output offset effects are negligible). All input-related numbers are referred to the input (RTI) which is to say that the effect on the output is "G" times larger. Voltage offset vs. power supply is also specified at one or more gain settings and is also RTI.

By separating these errors, one can evaluate the total error independent of the gain setting used. In a given gain configuration both errors can be combined to give a total error referred to the input (R.T.I.) or output (R.T.O.) by the following formula:

Total Error R.T.I. = input error + (output error/gain)

Total Error R.T.O. = (Gain × input error) + output error

As an illustration, a typical AD524 might have a +250 μ V output offset and a -50 μ V input offset. In a unity gain configuration, the total output offset would be 200 μ V or the sum of the two. At a gain of 100, the output offset would be -4.75 mV or: +250 μ V + 100(-50 μ V) = -4.75 mV.

The AD524 provides for both input and output offset adjustment. This simplifies very high precision applications and minimize offset voltage changes in switched gain applications. In such applications the input offset is adjusted first at the highest programmed gain, then the output offset is adjusted at $G = 1$.

GAIN

The AD524 has internal high accuracy pretrimmed resistors for pin programmable gain of 1, 10, 100 and 1000. One of the preset gains can be selected by pin strapping the appropriate gain terminal and RG_2 together (for $G = 1$ RG_2 is not connected).

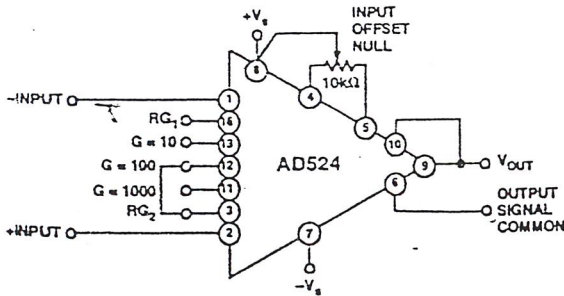


Figure 30. Operating Connections for $G = 100$.

The AD524 can be configured for gains other than those that are internally preset; there are two methods to do this. The first method uses just an external resistor connected between pins 3

and 16 which programs the gain according to the formula

$$R_C = \frac{40k}{G-1}$$

(see Figure 31). For best results R_C should be a

precision resistor with a low temperature coefficient. An external R_C affects both gain accuracy and gain drift due to the mismatch between it and the internal thin-film resistors. Gain accuracy is determined by the tolerance of the external R_C and the absolute accuracy of the internal resistors ($\pm 20\%$). Gain drift is determined by the mismatch of the temperature coefficient of R_C and the temperature coefficient of the internal resistors ($-50 \text{ ppm}/^\circ\text{C}$ typ

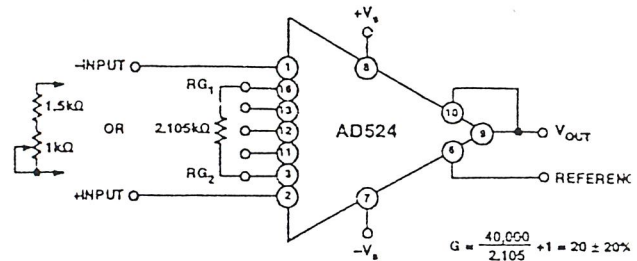


Figure 31. Operating Connections for $G = 20$

The second technique uses the internal resistors in parallel with an external resistor (Figure 32). This technique minimizes the gain adjustment range and reduces the effects of temperature coefficient sensitivity.

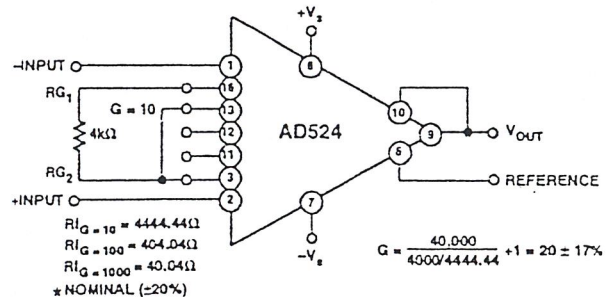


Figure 32. Operating Connections for $G = 20$, Low Gain T.C. Technique

The AD524 may also be configured to provide gain in the output stage. Figure 33 shows an H pad attenuator connected to the reference and sense lines of the AD524. R_1 , R_2 and R_3 should be made as low as possible to minimize the gain variation and reduction of CMRR. Varying R_2 will precisely set the gain without affecting CMRR. CMRR is determined by the match R_1 and R_3 .

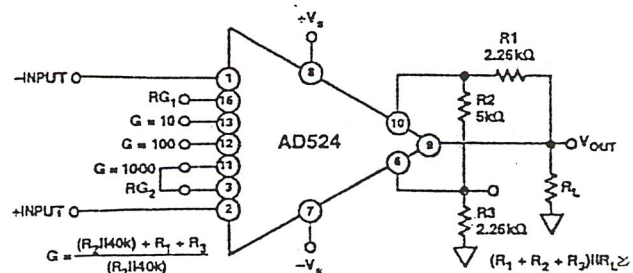


Figure 33. Gain of 2000

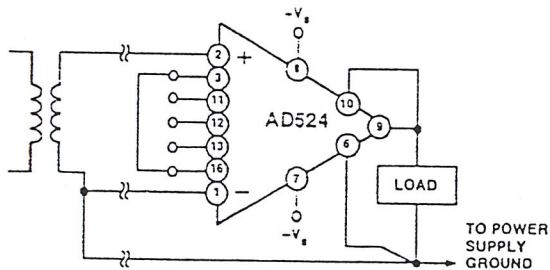
Table I. Output Gain Resistor Values

Output Gain	R2	R1, R3	Nominal Gain
2	5 kΩ	2.26 kΩ	2.02
5	1.05 kΩ	2.05 kΩ	5.01
10	1 kΩ	4.42 kΩ	10.1

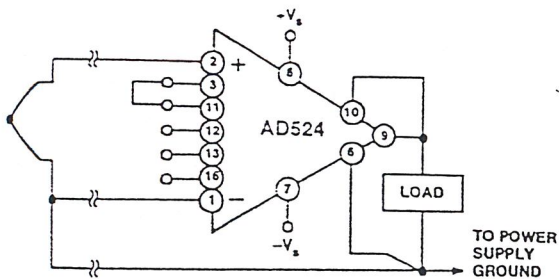
INPUT BIAS CURRENTS

Input bias currents are those currents necessary to bias the input transistors of a dc amplifier. Bias currents are an additional source of input error and must be considered in a total error budget. The bias currents when multiplied by the source resistance appear as an offset voltage. What is of concern in calculating bias current errors is the change in bias current with respect to signal voltage and temperature. Input offset current is the difference between the two input bias currents. The effect of offset current is an input offset voltage whose magnitude is the offset current times the source impedance imbalance.

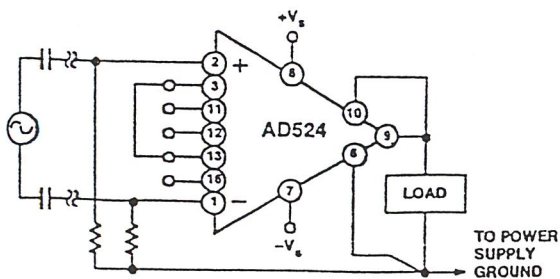
Although instrumentation amplifiers have differential inputs, there must be a return path for the bias currents. If this is not provided, those currents will charge stray capacitances, causing the output to drift uncontrollably or to saturate. Therefore, when amplifying "floating" input sources such as transformers and thermocouples, as well as ac-coupled sources, there must still be a dc path from each input to ground.



a. Transformer Coupled



b. Thermocouple



c. AC Coupled

Figure 34. Indirect Ground Returns for Bias Currents

COMMON-MODE REJECTION

Common-mode rejection is a measure of the change in output voltage when both inputs are changed equal amounts. These specifications are usually given for a full-range input voltage change and a specified source imbalance. "Common-Mode Rejection Ratio" (CMRR) is a ratio expression while "Common-Mode Rejection" (CMR) is the logarithm of that ratio. For example, a CMRR of 10,000 corresponds to a CMR of 80 dB.

In an instrumentation amplifier, ac common-mode rejection is only as good as the differential phase shift. Degradation of ac common-mode rejection is caused by unequal drops across differing track resistances and a differential phase shift due to varied stray capacitances or cable capacitances. In many applications shielded cables are used to minimize noise. This technique can create common mode rejection errors unless the shield is properly driven. Figures 35 and 36 shows active data guards which are configured to improve ac common mode rejection by "bootstrapping" the capacitances of the input cabling, thus minimizing differential phase shift.

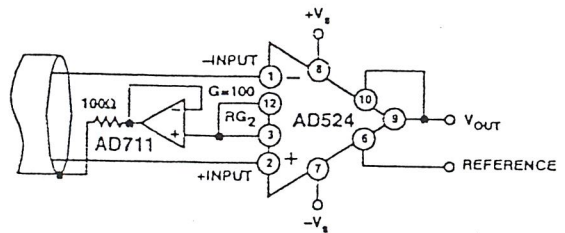


Figure 35. Shield Driver, $G \geq 100$

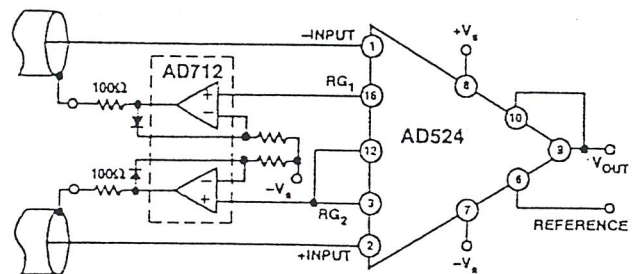


Figure 36. Differential Shield Driver

GROUNDING

Many data-acquisition components have two or more ground pins which are not connected together within the device. These grounds must be tied together at one point, usually at the system power-supply ground. Ideally, a single solid ground would be desirable. However, since current flows through the ground

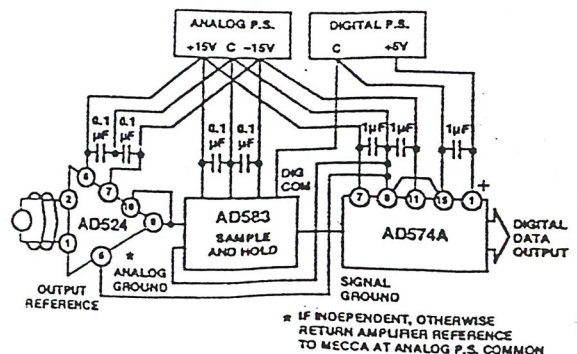


Figure 37. Basic Grounding Practice

MAXIM

+5V-Powered, Multi-Channel RS-232 Drivers/Receivers

General Description

The MAX220-MAX249 family of line drivers/receivers is intended for all EIA/TIA-232E and V.28/V.24 communications interfaces, and in particular, for those applications where $\pm 12V$ is not available.

These parts are particularly useful in battery-powered systems since their low-power shutdown mode reduces power dissipation to less than $5\mu W$. The MAX225, MAX233, MAX235, and MAX245-MAX247 use no external components and are recommended for applications where printed circuit board space is critical.

Applications

Portable Computers
Low-Power Modems
Interface Translation
Battery-Powered RS-232 Systems
Multi-Drop RS-232 Networks

Features

Superior to Bipolar

- ♦ Operate from Single +5V Power Supply (+5V and +12V—MAX231 and MAX239)
- ♦ Low-Power Receive Mode In Shutdown (MAX223/MAX242)
- ♦ Meet All EIA/TIA-232E and V.28 Specifications
- ♦ Multiple Drivers and Receivers
- ♦ 3-State Driver and Receiver Outputs
- ♦ Open-Line Detection (MAX243)

Ordering Information

PART	TEMP. RANGE	PIN-PACKAGE
MAX220CPE	0°C to +70°C	16 Plastic DIP
MAX220CSE	0°C to +70°C	16 Narrow SO
MAX220CWE	0°C to +70°C	16 Wide SO
MAX220C/D	0°C to +70°C	Dice*
MAX220EPE	-40°C to +85°C	16 Plastic DIP
MAX220ESE	-40°C to +85°C	16 Narrow SO
MAX220EWE	-40°C to +85°C	16 Wide SO
MAX220EJE	-40°C to +85°C	16 CERDIP
MAX220MJE	-55°C to +125°C	16 CERDIP

Ordering Information continued at end of data sheet.
*Contact factory for dice specifications.

Selection Table

Part Number	Power Supply (V)	No. of RS-232 Drivers/Rx	No. of Ext. Caps	Nominal Cap. Value (μF)	SHDN. & Three-State	Rx Active in SHDN	Data Rate (kbps)	Features
MAX220	+5	2/2	4	4.7/10	No		120	Ultra low-power, industry-standard pinout
MAX222	+5	2/2	4	0.1	Yes		200	+5V IBM PC serial port with receivers active in shutdown
MAX223 (MAX213)	+5	4/5	4	1.0 (0.1)	Yes	✓	120	MAX241 + receivers active in shutdown
MAX225	+5	5/5	0	-	Yes	✓	120	Available in SO
MAX230 (MAX200)	+5	5/0	4	1.0 (0.1)	Yes		120	5 drivers with shutdown
MAX231 (MAX201)	+5 and +7.5 to +13.2	2/2	2	1.0 (0.1)	No		120	Standard +5/+12V or battery supplies; same functions as MAX232
MAX232 (MAX202)	+5	2/2	4	1.0 (0.1)	No		120 (64)	Industry standard
MAX232A	+5	2/2	4	0.1	No		200	Higher slew rate, small caps
MAX233 (MAX203)	+5	2/2	0	-	No		120	No external caps
MAX233A	+5	2/2	0	-	No		200	No external caps, high slew rate
MAX234 (MAX204)	+5	4/0	4	1.0 (0.1)	No		120	Replaces 1488
MAX235 (MAX205)	+5	5/5	0	-	Yes		120	No external caps
MAX236 (MAX206)	+5	4/3	4	1.0 (0.1)	Yes		120	Shutdown, three-state
MAX237 (MAX207)	+5	5/3	4	1.0 (0.1)	No		120	Complements IBM PC serial port
MAX238 (MAX208)	+5	4/4	4	1.0 (0.1)	No		120	Replaces 1488 and 1489
MAX239 (MAX209)	+5 and +7.5 to +13.2	3/5	2	1.0 (0.1)	No		120	Standard +5/+12V or battery supplies; single package solution for IBM PC serial port
MAX240	+5	5/5	4	1.0	Yes		120	DIP or flatpack package
MAX241 (MAX211)	+5	4/5	4	1.0 (0.1)	Yes		120	Complete IBM PC serial port
MAX242	+5	2/2	4	0.1	Yes	✓	200	Separate shutdown and enable
MAX243	+5	2/2	4	0.1	No		200	Open-line detection simplifies cabling
MAX244	+5	3/10	4	1.0	No		120	High slew rate
MAX245	+5	3/10	0	-	Yes	✓	120	High slew rate, int. caps, two shutdown modes
MAX246	+5	3/10	0	-	Yes	✓	120	High slew rate, int. caps, three shutdown modes
MAX247	+5	3/9	0	-	Yes	✓	120	High slew rate, int. caps, nine operating modes
MAX248	+5	5/8	4	1.0	Yes	✓	120	High slew rate, selective half-chip enables
MAX249	+5	6/10	4	1.0	Yes	✓	120	Available in quad flatpack package

MAXIM

Maxim Integrated Products 15

Call toll free 1-800-998-8800 for free samples or literature.

MAX220-MAX249



+5V-Powered, Multi-Channel RS-232 Drivers/Receivers

ABSOLUTE MAXIMUM RATINGS—MAX220/222/232A/233A/242/243

Supply Voltage (V_{CC})	-0.3V to +6V	16-Pin Narrow SO (derate 8.70mW/°C above +70°C)	696mW
Input Voltages		16-Pin Wide SO (derate 9.52mW/°C above +70°C)	762mW
T_{IN}	-0.3V to ($V_{CC} - 0.3V$)	18-Pin Wide SO (derate 9.52mW/°C above +70°C)	762mW
R_{IN}	$\pm 30V$	20-Pin Wide SO (derate 10.00mW/°C above +70°C)	800mW
T_{OUT} (Note 1)	$\pm 15V$	16-Pin CERDIP (derate 10.00mW/°C above +70°C)	800mW
Output Voltages		18-Pin CERDIP (derate 10.53mW/°C above +70°C)	842mW
T_{OUT}	$\pm 15V$	Operating Temperature Ranges	
R_{OUT}	-0.3V to ($V_{CC} + 0.3V$)	MAX2__AC__, MAX2__C__	0°C to +70°C
Driver/Receiver Output Short Circuited to GND	Continuous	MAX2__AE__, MAX2__E__	-40°C to +85°C
Continuous Power Dissipation ($T_A = +70^\circ C$)		MAX2__AM__, MAX2__M__	-55°C to +125°C
16-Pin Plastic DIP (derate 10.53mW/°C above +70°C)	842mW	Storage Temperature Range	-65°C to +160°C
18-Pin Plastic DIP (derate 11.11mW/°C above +70°C)	889mW	Lead Temperature (soldering, 10 sec)	+300°
20-Pin Plastic DIP (derate 8.00mW/°C above +70°C)	440mW		

Note 1: Input voltage measured with T_{OUT} in high-impedance state, \overline{SHDN} or $V_{CC} = 0V$.

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

ELECTRICAL CHARACTERISTICS—MAX220/222/232A/233A/242/243

($V_{CC} = -5V \pm 10\%$, C1-C4 = 0.1 μF , $T_A = T_{MIN}$ to T_{MAX} , unless otherwise noted.)

PARAMETER	CONDITIONS		MIN	TYP	MAX	UNITS
RS-232 TRANSMITTERS						
Output Voltage Swing	All transmitter outputs loaded with 3k Ω to GND		± 5	± 8		V
Input Logic Threshold Low				1.4	0.8	V
Input Logic Threshold High			2	1.4		V
Logic Pull-Up/Input Current	Normal operation			5	40	μA
	$\overline{SHDN} = 0V$, MAX222/242, shutdown		± 0.01		± 1	
Output Leakage Current	$V_{CC} = 5.5V$, $\overline{SHDN} = 0V$, $V_{OUT} = \pm 15V$, MAX222/242		± 0.01		± 10	μA
	$V_{CC} = \overline{SHDN} = 0V$, $V_{OUT} = \pm 15V$		± 0.01		± 10	
Data Rate	Except MAX220, normal operation			200	116	kbits/sec
	MAX220			22	20	
Transmitter Output Resistance	$V_{CC} = V+V- = 0V$, $V_{OUT} = \pm 2V$		300	10M		Ω
Output Short-Circuit Current	$V_{OUT} = 0V$		± 7	± 22		mA
RS-232 RECEIVERS						
RS-232 Input Voltage Operating Range					± 30	V
RS-232 Input Threshold Low	$V_{CC} = 5V$	Except MAX243 R_{2IN}	0.8	1.3		V
		MAX243 R_{2IN} (Note 2)	-3			
RS-232 Input Threshold High	$V_{CC} = 5V$	Except MAX243 R_{2IN}		1.8	2.4	V
		MAX243 R_{2IN} (Note 2)		-0.5	-0.1	
RS-232 Input Hysteresis	Except MAX243, $V_{CC} = 5V$, no hyst. in shdn.		0.2	0.5	1	V
	MAX243			1		
RS-232 Input Resistance			3	5	7	k Ω
TTL/CMOS Output Voltage Low	$I_{OUT} = 3.2mA$			0.2	0.4	V
TTL/CMOS Output Voltage High	$I_{OUT} = -1.0mA$		3.5	$V_{CC} - 0.2$		V
TTL/CMOS Output Short-Circuit Current	Sourcing $V_{OUT} = GND$		-2	-10		mA
	Sinking $V_{OUT} = V_{CC}$		10	30		
TTL/CMOS Output Leakage Current	$\overline{SHDN} = V_{CC}$ or $\overline{EN} = V_{CC}$, $0V \leq V_{OUT} \leq V_{CC}$			± 0.05	± 10	μA

+5V-Powered, Multi-Channel RS-232 Drivers/Receivers

ELECTRICAL CHARACTERISTICS—MAX220/222/232A/233A/242/243 (continued)
($V_{CC} = +5V \pm 10\%$, $C_1-C_4 = 0.1\mu F$, $T_A = T_{MIN}$ to T_{MAX} , unless otherwise noted.)

PARAMETER	CONDITIONS		MIN	TYP	MAX	UNITS
EN Input Threshold Low	MAX242			1.4	0.8	V
EN Input Threshold High	MAX242		2.5	1.4		V
POWER SUPPLY						
Operating Supply Voltage			4.5		5.5	V
VCC Supply Current ($\overline{SHDN} = V_{CC}$), Figures 5, 6, 9, 19	No load	MAX220		0.5	2	mA
		MAX222/232A/233A/242/243		4	10	
	3k Ω load both outputs	MAX220		12		
		MAX222/232A/233A/242/243		15		
Shutdown Supply Current:	MAX222/242	$T_A = +25^\circ C$		0.1	10	μA
		$T_A = 0^\circ$ to $+70^\circ C$		2	50	
		$T_A = -40^\circ$ to $+85^\circ C$		2	50	
		$T_A = -55^\circ$ to $+125^\circ C$		35	100	
SHDN Input Leakage Current	MAX222/242				± 1	μA
SHDN Threshold Low	MAX222/242			1.4	0.8	V
SHDN Threshold High	MAX222/242		2.5	1.4		V
AC CHARACTERISTICS						
Transition Slew Rate	$C_L = 50pF$ to $2500pF$, $R_L = 3k\Omega$ to $7k\Omega$, $V_{CC} = 5V$, $T_A = +25^\circ C$, measured from $+3V$ to $-3V$ or $-3V$ to $+3V$	MAX222/232A/233A/242/243		12	30	V/ μs
		MAX220		3	30	
Transmitter Propagation Delay TLL to RS-232 (Normal Operation), Figure 1	tPHLT	MAX222/232A/233A/242/243		1.3	3.5	μs
		MAX220		4	10	
	tPLHT	MAX222/232A/233A/242/243		1.5	3.5	
		MAX220		5	10	
Receiver Propagation Delay RS-232 to TLL (Normal Operation), Figure 2	tPHLR	MAX222/232A/233A/242/243		0.5	1	μs
		MAX220		0.6	3	
	tPLHR	MAX222/232A/233A/242/243		0.6	1	
		MAX220		0.8	3	
Receiver Propagation Delay RS-232 to TLL (Shutdown), Figure 2	tPHLS	MAX242		0.5	10	μs
	tPLHS	MAX242		2.5	10	
Receiver-Output Enable Time, Figure 3	tER	MAX242		125	500	ns
Receiver-Output Disable Time, Figure 3	tDR	MAX242		160	500	ns
Transmitter-Output Enable Time (SHDN goes High), Figure 4	tET	MAX222/242, 0.1 μF caps (includes charge-pump start-up)		250		μs
Transmitter-Output Disable Time (SHDN goes Low), Figure 4	tDT	MAX222/242, 0.1 μF caps		600		ns
Transmitter + to - Propagation Delay Difference (Normal Operation)	tPHLT - tPLHT	MAX222/232A/233A/242/243		300		ns
		MAX220		2000		
Receiver + to - Propagation Delay Difference (Normal Operation)	tPHLR - tPLHR	MAX222/232A/233A/242/243		100		ns
		MAX220		225		

Note 2: MAX243 R_{OUT} is guaranteed to be low when R_{IN} is $\geq 0V$ or is floating.

+5V-Powered, Multi-Channel RS-232 Drivers/Receivers

MAX220-MAX249

ELECTRICAL CHARACTERISTICS — MAX223/MAX230-MAX241 (continued)
 (MAX223/230/232/234/236/237/238/240/241 $V_{CC} = +5V \pm 10\%$; MAX233/MAX235 $V_{CC} = 5V \pm 5\%$. C1-C4 = 1.0 μ F, MAX231/MAX239 $V_{CC} = 5V \pm 10\%$, $V+ = 7.5V$ to 13.2V, $T_A = T_{MIN}$ to T_{MAX} , unless otherwise noted.)

PARAMETER	CONDITIONS		MIN	TYP	MAX	UNITS
RS-232 Input Threshold Low	$T_A = +25^\circ\text{C}$ $V_{CC} = 5V$	Normal operation SHDN = 5V (MAX223) SHDN = 0V (MAX235-MAX241)	0.8	1.2		V
		Shutdown (MAX223) SHDN = 0V, EN = 5V (R4, R5)	0.6	1.5		
RS-232 Input Threshold High	$T_A = +25^\circ\text{C}$ $V_{CC} = 5V$	Normal operation SHDN = 5V (MAX223) SHDN = 0V (MAX235-MAX241)		1.7	2.4	V
		Shutdown (MAX223) SHDN = 0V, EN = 5V (R4, R5)		1.5	2.4	
RS-232 Input Hysteresis	$V_{CC} = 5V$; no hysteresis in shutdown		0.2	0.5	1.0	V
RS-232 Input Resistance	$T_A = +25^\circ\text{C}$, $V_{CC} = 5V$		3	5	7	k Ω
TTL/CMOS Output Voltage Low	$I_{OUT} = 1.6\text{mA}$ (MAX231-233 $I_{OUT} = 3.2\text{mA}$)				0.4	V
TTL/CMOS Output Voltage High	$I_{OUT} = 1.0\text{mA}$		3.5	$V_{CC} - 0.4$		V
TTL/CMOS Output Leakage Current	$0V \leq R_{OUT} \leq V_{CC}$; EN = 0V (MAX223); $\overline{\text{EN}} = V_{CC}$ (MAX235-241)			0.05	± 10	μA
Receiver Output Enable Time	Normal operation	MAX223		600		ns
		MAX235-MAX241		400		
Receiver Output Disable Time	Normal operation	MAX223		900		ns
		MAX235-MAX241		250		
Propagation Delay	RS-232 IN to TTL/CMOS OUT, $C_L = 150\text{pF}$	Normal operation		0.5	10	μs
		SHDN = 0V (MAX223)	t_{PHLS}	4	40	
			t_{PLHS}	6	40	
Transition Region Slew Rate	MAX223/MAX230/MAX234-241 $T_A = +25^\circ\text{C}$, $V_{CC} = 5V$ $R_L = 3\text{k}\Omega$ to 7k Ω , $C_L = 50\text{pF}$ to 2500pF, measured from +3V to -3V or -3V to +3V		3	5.1	30	V/ μs
	MAX231/MAX232/MAX233 $T_A = +25^\circ\text{C}$, $V_{CC} = 5V$ $R_L = 3\text{k}\Omega$ to 7k Ω , $C_L = 50\text{pF}$ to 2500pF, measured from +3V to -3V or -3V to +3V			4	30	
Transmitter Output Resistance	$V_{CC} = V+ = V- = 0V$, $V_{OUT} = \pm 2V$		300			Ω
Receiver Out Short-Circuit Current				± 10		mA

+5V-Powered, Multi-Channel RS-232 Drivers/Receivers

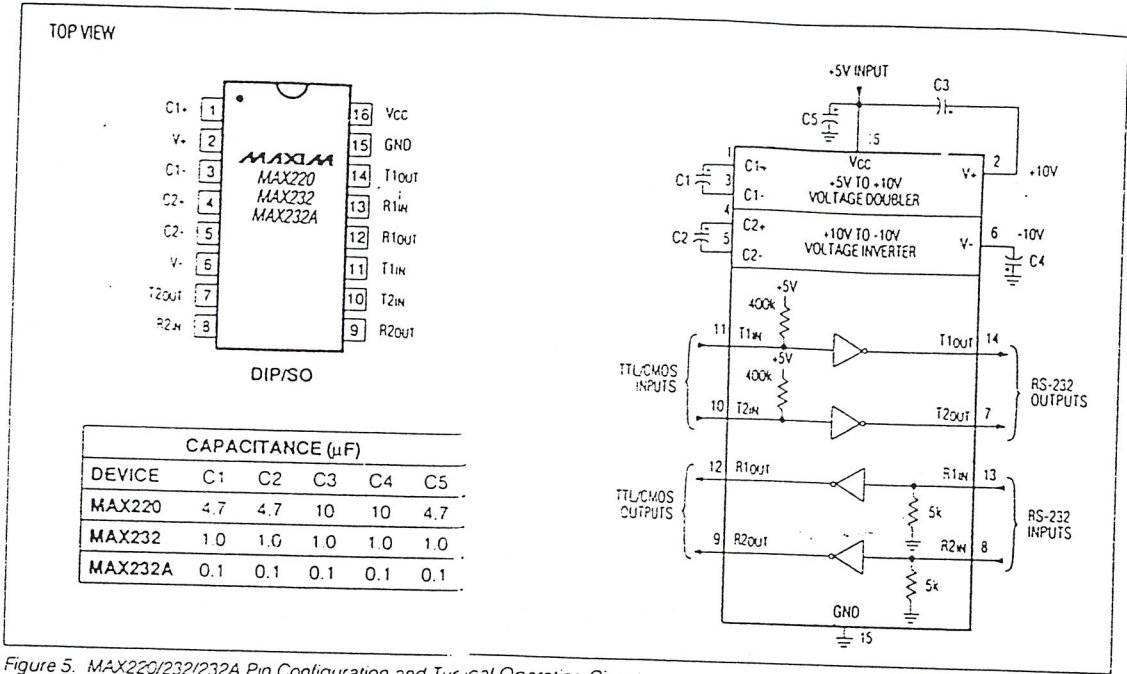


Figure 5. MAX220/232/232A Pin Configuration and Typical Operating Circuit

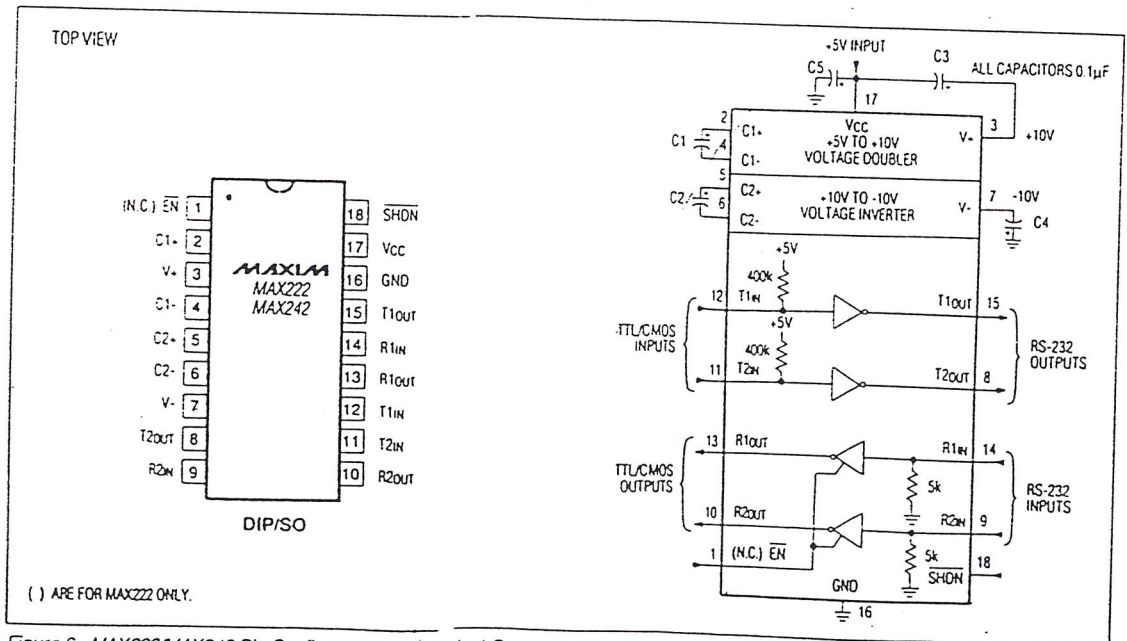


Figure 6. MAX222/MAX242 Pin Configuration and Typical Operating Circuit

+5V-Powered, Multi-Channel RS-232 Drivers/Receivers

Ordering Information (continued)

PART	TEMP. RANGE	PIN-PACKAGE
MAX222CPN	0°C to +70°C	18 Plastic DIP
MAX222CWN	0°C to +70°C	18 Wide SO
MAX222C/D	0°C to +70°C	Dice*
MAX222EPN	-40°C to +85°C	18 Plastic DIP
MAX222EWN	-40°C to +85°C	18 Wide SO
MAX222EJN	-40°C to +85°C	18 CERDIP
MAX222MJN	-55°C to +125°C	18 CERDIP
MAX223CAI	0°C to +70°C	28 SSOP
MAX223CWI	0°C to +70°C	28 Wide SO
MAX223C/D	0°C to +70°C	Dice*
MAX223EAI	-40°C to +85°C	28 SSOP
MAX223EWI	-40°C to +85°C	28 Wide SO
MAX225CWI	0°C to +70°C	28 Wide SO
MAX225EWI	-40°C to +85°C	28 Wide SO
MAX230CPP	0°C to +70°C	20 Plastic DIP
MAX230CWP	0°C to +70°C	20 Wide SO
MAX230C/D	0°C to +70°C	Dice*
MAX230EPP	-40°C to +85°C	20 Wide SO
MAX230EWP	-40°C to +85°C	20 Wide SO
MAX230EJP	-40°C to +85°C	20 CERDIP
MAX230MJP	-55°C to +125°C	20 CERDIP
MAX231CPD	0°C to +70°C	14 Plastic DIP
MAX231CWE	0°C to +70°C	16 Wide SO
MAX231CJD	0°C to +70°C	14 CERDIP
MAX231C/D	0°C to +70°C	Dice*
MAX231EPD	-40°C to +85°C	14 Plastic DIP
MAX231EWE	-40°C to +85°C	16 Wide SO
MAX231EJD	-40°C to +85°C	14 CERDIP
MAX231MJD	-55°C to +125°C	14 CERDIP
MAX232CPE	0°C to +70°C	16 Plastic DIP
MAX232CSE	0°C to +70°C	16 Narrow SO
MAX232CWE	0°C to +70°C	16 Wide SO
MAX232C/D	0°C to +70°C	Dice*
MAX232EPE	-40°C to +85°C	16 Plastic DIP
MAX232ESE	-40°C to +85°C	16 Narrow SO
MAX232EWE	-40°C to +85°C	18 Wide SO
MAX232EJE	-40°C to +85°C	16 CERDIP
MAX232MJE	-55°C to +125°C	16 CERDIP
MAX232MLP	-55°C to +125°C	20 LCC
MAX232ACPE	0°C to +70°C	16 Plastic DIP
MAX232ACSE	0°C to +70°C	16 Narrow SO
MAX232ACWE	0°C to +70°C	16 Wide SO

MAX232AC/D	0°C to +70°C	Dice*
MAX232AEPE	-40°C to +85°C	16 Plastic DIP
MAX232AESE	-40°C to +85°C	16 Plastic SO
MAX232AEWE	-40°C to +85°C	18 Wide SO
MAX232AEJE	-40°C to +85°C	16 CERDIP
MAX232AMJE	-55°C to +125°C	16 CERDIP
MAX232AML	-55°C to +125°C	20 LCC
MAX233CPP	0°C to +70°C	20 Plastic DIP
MAX233EPP	-40°C to +85°C	20 Plastic DIP
MAX233ACPP	0°C to +70°C	20 Plastic DIP
MAX233ACWP	0°C to +70°C	20 Wide SO
MAX233AEPP	-40°C to +85°C	20 Plastic DIP
MAX233AERP	-40°C to +85°C	20 Wide SO
MAX234CPE	0°C to +70°C	16 Plastic DIP
MAX234CWE	0°C to +70°C	16 Wide SO
MAX234C/D	0°C to +70°C	Dice*
MAX234EPE	-40°C to +85°C	16 Plastic DIP
MAX234EWE	-40°C to +85°C	16 Wide SO
MAX234EJE	-40°C to +85°C	16 CERDIP
MAX234MJP	-55°C to +125°C	16 CERDIP
MAX235CPG	0°C to +70°C	24 Wide Plastic DIP
MAX235EPG	-40°C to +85°C	24 Wide Plastic DIP
MAX235EDG	-40°C to +85°C	24 Ceramic SB
MAX235MDG	-55°C to +125°C	24 Ceramic SB
MAX236CNG	0°C to +70°C	24 Narrow Plastic DIP
MAX236CEG	0°C to +70°C	24 Wide SO
MAX236C/D	0°C to +70°C	Dice*
MAX236ENG	-40°C to +85°C	24 Narrow Plastic DIP
MAX236EWG	-40°C to +85°C	24 Wide SO
MAX236ERG	-40°C to +85°C	24 Narrow CERDIP
MAX236MRG	-55°C to +125°C	24 CERDIP
MAX237CNG	0°C to +70°C	24 Narrow Plastic DIP
MAX237CEG	0°C to +70°C	24 Wide SO
MAX237C/D	0°C to +70°C	Dice*
MAX237ENG	-40°C to +85°C	24 Narrow Plastic DIP
MAX237EWG	-40°C to +85°C	24 Wide SO
MAX237ERG	-40°C to +85°C	24 Narrow CERDIP
MAX237MRG	-55°C to +125°C	24 CERDIP
MAX238CNG	0°C to +70°C	24 Narrow Plastic DIP
MAX238CEG	0°C to +70°C	24 Wide SO
MAX238C/D	0°C to +70°C	Dice*
MAX238ENG	-40°C to +85°C	24 Narrow Plastic DIP

*Contact factory for dice specifications.

ADC0802, ADC0803 ADC0804

8-Bit, Microprocessor-
Compatible, A/D Converters

August 1997

Features

- 80C48 and 80C80/85 Bus Compatible - No Interfacing Logic Required
- Conversion Time < 100 μ s
- Easy Interface to Most Microprocessors
- Will Operate In a "Stand Alone" Mode
- Differential Analog Voltage Inputs
- Works with Bandgap Voltage References
- TTL Compatible Inputs and Outputs
- On-Chip Clock Generator
- 0V to 5V Analog Voltage Input Range (Single +5V Supply)
- No Zero-Adjust Required

Description

The ADC0802 family are CMOS 8-Bit, successive-approximation A/D converters which use a modified potentiometric ladder and are designed to operate with the 8080A control bus via three-state outputs. These converters appear to the processor as memory locations or I/O ports, and hence no interfacing logic is required.

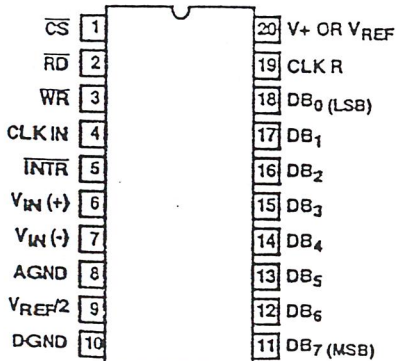
The differential analog voltage input has good common-mode-rejection and permits offsetting the analog zero-input-voltage value. In addition, the voltage reference input can be adjusted to allow encoding any smaller analog voltage span to the full 8 bits of resolution.

Ordering Information

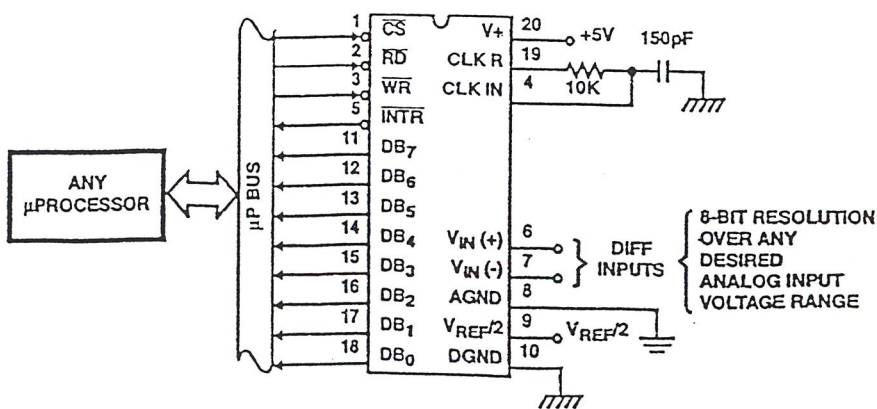
PART NUMBER	ERROR	EXTERNAL CONDITIONS	TEMP. RANGE (°C)	PACKAGE	PKG. NO
ADC0802LCN	$\pm 1/2$ LSB	$V_{REF/2} = 2.500V_{DC}$ (No Adjustments)	0 to 70	20 Ld PDIP	E20.3
ADC0802LCD	$\pm 3/4$ LSB		-40 to 85	20 Ld Cerdip	F20.3
ADC0802LD	± 1 LSB		-55 to 125	20 Ld Cerdip	F20.3
ADC0803LCN	$\pm 1/2$ LSB	$V_{REF/2}$ Adjusted for Correct Full Scale Reading	0 to 70	20 Ld PDIP	E20.3
ADC0803LCD	$\pm 3/4$ LSB		-40 to 85	20 Ld Cerdip	F20.3
ADC0803LCWM	± 1 LSB		-40 to 85	20 Ld SOIC	M20.3
ADC0803LD	± 1 LSB		-55 to 125	20 Ld Cerdip	F20.3
ADC0804LCN	± 1 LSB	$V_{REF/2} = 2.500V_{DC}$ (No Adjustments)	0 to 70	20 Ld PDIP	E20.3
ADC0804LCD	± 1 LSB		-40 to 85	20 Ld Cerdip	F20.3
ADC0804LCWM	± 1 LSB		-40 to 85	20 Ld SOIC	M20.3

Pinout

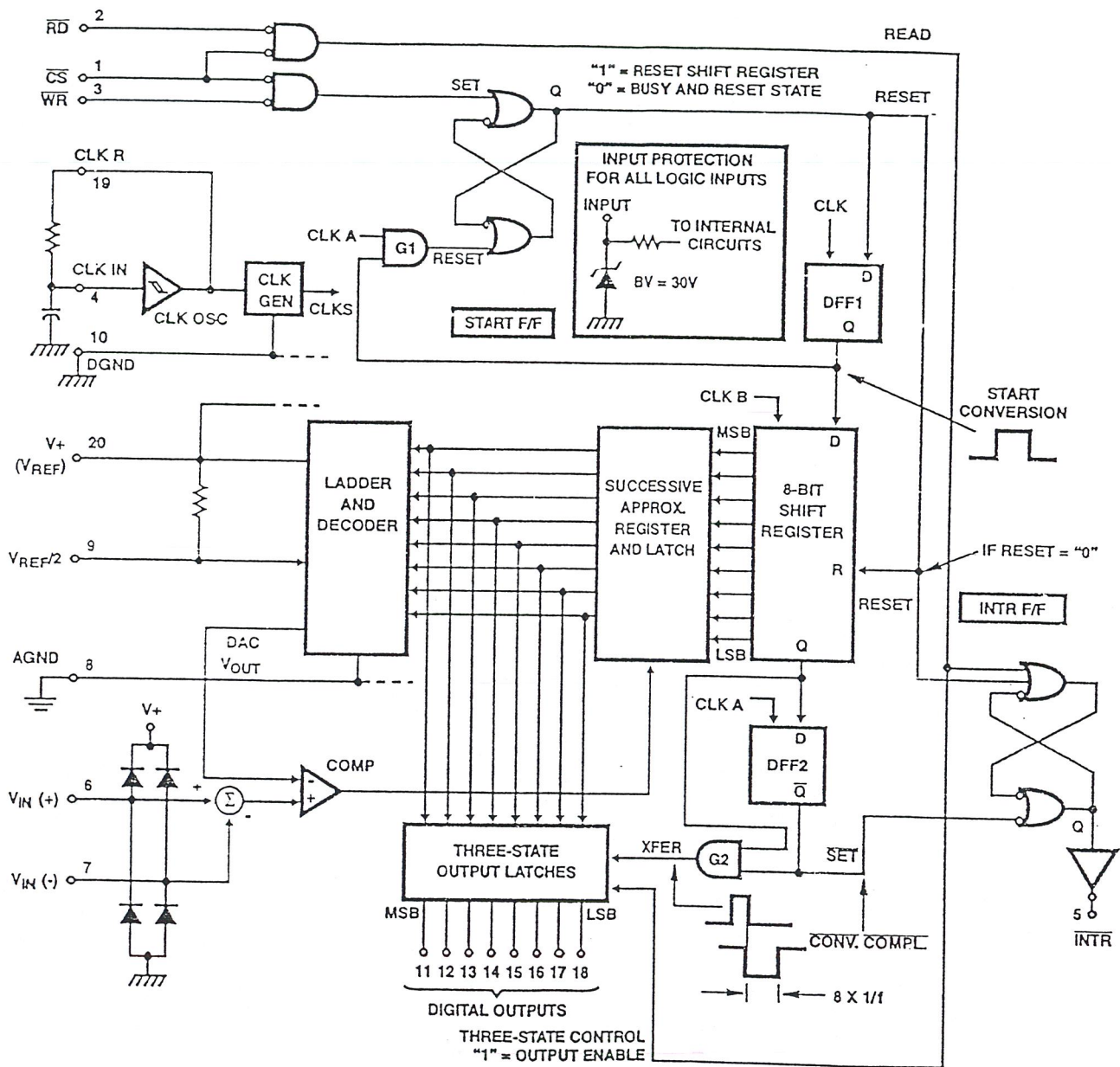
ADC0802, ADC0803, ADC0804
(PDIP, Cerdip)
TOP VIEW



Typical Application Schematic



Functional Diagram



ADC0802, ADC0803, ADC0804

Absolute Maximum Ratings

Supply Voltage	6.5V
Voltage at Any Input	-0.3V to (V ⁺ +0.3V)

Operating Conditions

Temperature Range	
ADC0802/03LD	-55°C to 125°C
ADC0802/03/04LCD	-40°C to 85°C
ADC0802/03/04LCN	0°C to 70°C
ADC0803/04LCWM	-40°C to 85°C

Thermal Information

Thermal Resistance (Typical, Note 1)	θ_{JA} (°C/W)	θ_{JC} (°C/W)
PDIP Package	125	N/A
CERDIP Package	80	20
SOIC Package	120	N/A
Maximum Junction Temperature		
Hermetic Package	175°C	
Plastic Package	150°C	
Maximum Storage Temperature Range	-65°C to 150°C	
Maximum Lead Temperature (Soldering, 10s)	300°C (SOIC - Lead Tips Only)	

CAUTION: Stresses above those listed in "Absolute Maximum Ratings" may cause permanent damage to the device. This is a stress only rating and operation of the device at these or any other conditions above those indicated in the operational sections of this specification is not implied.

NOTE:

- θ_{JA} is measured with the component mounted on an evaluation PC board in free air.

Electrical Specifications (Notes 1, 7)

PARAMETER	TEST CONDITIONS	MIN	TYP	MAX	UNITS
CONVERTER SPECIFICATIONS V ⁺ = 5V, T _A = 25°C and f _{CLK} = 640kHz, Unless Otherwise Specified					
Total Unadjusted Error					
ADC0802	V _{REF/2} = 2.500V	-	-	±1/2	LSB
ADC0803	V _{REF/2} Adjusted for Correct Full Scale Reading	-	-	±1/2	LSB
ADC0804	V _{REF/2} = 2.500V	-	-	±1	LSB
V _{REF/2} Input Resistance	Input Resistance at Pin 9	1.0	1.3	-	kΩ
Analog Input Voltage Range	(Note 2)	GND-0.05	-	(V ⁺) + 0.05	V
DC Common-Mode Rejection	Over Analog Input Voltage Range	-	±1/16	±1/8	LSB
Power Supply Sensitivity	V ⁺ = 5V ±10% Over Allowed Input Voltage Range	-	±1/16	±1/8	LSB
CONVERTER SPECIFICATIONS V ⁺ = 5V, 0°C to 70°C and f _{CLK} = 640kHz, Unless Otherwise Specified					
Total Unadjusted Error					
ADC0802	V _{REF/2} = 2.500V	-	-	±1/2	LSB
ADC0803	V _{REF/2} Adjusted for Correct Full Scale Reading	-	-	±1/2	LSB
ADC0804	V _{REF/2} = 2.500V	-	-	±1	LSB
V _{REF/2} Input Resistance	Input Resistance at Pin 9	1.0	1.3	-	kΩ
Analog Input Voltage Range	(Note 2)	GND-0.05	-	(V ⁺) + 0.05	V
DC Common-Mode Rejection	Over Analog Input Voltage Range	-	±1/8	±1/4	LSB
Power Supply Sensitivity	V ⁺ = 5V ±10% Over Allowed Input Voltage Range	-	±1/16	±1/8	LSB
CONVERTER SPECIFICATIONS V ⁺ = 5V, -25°C to 85°C and f _{CLK} = 640kHz, Unless Otherwise Specified					
Total Unadjusted Error					
ADC0802	V _{REF/2} = 2.500V	-	-	±3/4	LSB
ADC0803	V _{REF/2} Adjusted for Correct Full Scale Reading	-	-	±3/4	LSB
ADC0804	V _{REF/2} = 2.500V	-	-	±1	LSB
V _{REF/2} Input Resistance	Input Resistance at Pin 9	1.0	1.3	-	kΩ
Analog Input Voltage Range	(Note 2)	GND-0.05	-	(V ⁺) + 0.05	V
DC Common-Mode Rejection	Over Analog Input Voltage Range	-	±1/8	±1/4	LSB
Power Supply Sensitivity	V ⁺ = 5V ±10% Over Allowed Input Voltage Range	-	±1/16	±1/8	LSB

ADC0802, ADC0803, ADC0804

Electrical Specifications (Notes 1, 7) (Continued)

PARAMETER	TEST CONDITIONS	MIN	TYP	MAX	UNITS
CONVERTER SPECIFICATIONS $V_+ = 5V$, $-55^\circ C$ to $125^\circ C$ and $f_{CLK} = 640kHz$, Unless Otherwise Specified					
Total Unadjusted Error					
ADC0802	$V_{REF/2} = 2.500V$	-	-	± 1	LSB
ADC0803	$V_{REF/2}$ Adjusted for Correct Full Scale Reading	-	-	± 1	LSB
$V_{REF/2}$ Input Resistance	Input Resistance at Pin 9	1.0	1.3	-	k Ω
Analog Input Voltage Range	(Note 2)	GND-0.05	-	$(V_+) + 0.05$	V
DC Common-Mode Rejection	Over Analog Input Voltage Range	-	$\pm 1/8$	$\pm 1/4$	LSB
Power Supply Sensitivity	$V_+ = 5V \pm 10\%$ Over Allowed Input Voltage Range	-	$\pm 1/8$	$\pm 1/4$	LSB
AC TIMING SPECIFICATIONS $V_+ = 5V$, and $T_A = 25^\circ C$, Unless Otherwise Specified					
Clock Frequency, f_{CLK}	$V_+ = 6V$ (Note 3)	100	640	1280	kHz
	$V_+ = 5V$	100	640	800	kHz
Clock Periods per Conversion (Note 4), t_{CONV}		62	-	73	Clocks/Conv
Conversion Rate In Free-Running Mode, CR	\overline{INTR} tied to \overline{WR} with $\overline{CS} = 0V$, $f_{CLK} = 640kHz$	-	-	8888	Conv/s
Width of \overline{WR} Input (Start Pulse Width), $t_{W(WR)}$	$\overline{CS} = 0V$ (Note 5)	100	-	-	ns
Access Time (Delay from Falling Edge of \overline{RD} to Output Data Valid), t_{ACC}	$C_L = 100pF$ (Use Bus Driver IC for Larger C_L)	-	135	200	ns
Three-State Control (Delay from Rising Edge of \overline{RD} to HI-Z State), t_{1H} , t_{0H}	$C_L = 10pF$, $R_L = 10K$ (See Three-State Test Circuits)	-	125	250	ns
Delay from Falling Edge of \overline{WR} to Reset of \overline{INTR} , t_{WI} , t_{RI}		-	300	450	ns
Input Capacitance of Logic Control Inputs, C_{IN}		-	5	-	pF
Three-State Output Capacitance (Data Buffers), C_{OUT}		-	5	-	pF
DC DIGITAL LEVELS AND DC SPECIFICATIONS $V_+ = 5V$, and T_{MIN} to T_{MAX} , Unless Otherwise Specified					
CONTROL INPUTS (Note 6)					
Logic "1" Input Voltage (Except Pin 4 CLK IN), V_{INH}	$V_+ = 5.25V$	2.0	-	V_+	V
Logic "0" Input Voltage (Except Pin 4 CLK IN), V_{INL}	$V_+ = 4.75V$	-	-	0.8	V
CLK IN (Pin 4) Positive Going Threshold Voltage, V_{+CLK}		2.7	3.1	3.5	V
CLK IN (Pin 4) Negative Going Threshold Voltage, V_{-CLK}		1.5	1.8	2.1	V
CLK IN (Pin 4) Hysteresis, V_H		0.6	1.3	2.0	V
Logic "1" Input Current (All Inputs), I_{INH1}	$V_{IN} = 5V$	-	0.005	1	μA
Logic "0" Input Current (All Inputs), I_{INL0}	$V_{IN} = 0V$	-1	-0.005	-	μA
Supply Current (Includes Ladder Current), I_+	$f_{CLK} = 640kHz$, $T_A = 25^\circ C$ and $\overline{CS} = HI$	-	1.3	2.5	mA
DATA OUTPUTS AND \overline{INTR}					
Logic "0" Output Voltage, V_{OL}	$I_O = 1.6mA$, $V_+ = 4.75V$	-	-	0.4	V

Electrical Specifications (Notes 1, 7) (Continued)

PARAMETER	TEST CONDITIONS	MIN	TYP	MAX	UNITS
Logic "1" Output Voltage, V_{OH}	$I_O = -360\mu A$, $V_+ = 4.75V$	2.4	-	-	V
Three-State Disabled Output Leakage (All Data Buffers), I_{LO}	$V_{OUT} = 0V$	-3	-	-	μA
	$V_{OUT} = 5V$	-	-	3	μA
Output Short Circuit Current, I_{SOURCE}	V_{OUT} Short to Gnd $T_A = 25^\circ C$	4.5	6	-	mA
Output Short Circuit Current, I_{SINK}	V_{OUT} Short to V_+ $T_A = 25^\circ C$	9.0	16	-	mA

NOTES:

- All voltages are measured with respect to GND, unless otherwise specified. The separate AGND point should always be wired to the DGND, being careful to avoid ground loops.
- 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_+ supply. Be careful, during testing at low V_+ 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. As long as the analog V_{IN} does not exceed the supply voltage by more than 50mV, the output code will be correct. To achieve an absolute 0V to 5V input voltage range will therefore require a minimum supply voltage of 4.950V over temperature variations, initial tolerance and loading.
- With $V_+ = 6V$, the digital logic interfaces are no longer TTL compatible.
- 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 \overline{CS} input is assumed to bracket the \overline{WR} strobe input so that timing is dependent on the \overline{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 \overline{WR} pulse (see Timing Diagrams).
- CLK IN (pin 4) is the input of a Schmitt trigger circuit and is therefore specified separately.
- None of these A/Ds requires a zero-adjust. However, if an all zero code is desired for an analog input other than 0V, or if a narrow full scale span exists (for example: 0.5V to 4V full scale) the $V_{IN(-)}$ input can be adjusted to achieve this. See the Zero Error description in this data sheet.

Timing Waveforms

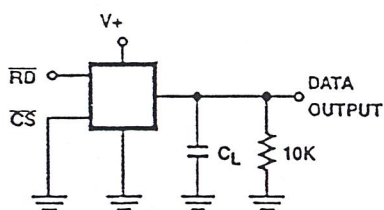


FIGURE 1A. t_{1H}

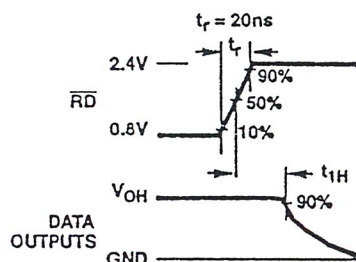


FIGURE 1B. t_{1H} , $C_L = 10pF$

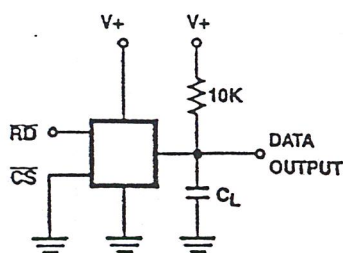


FIGURE 1C. t_{0H}

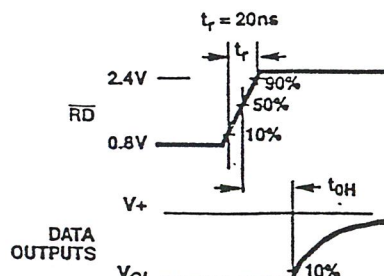


FIGURE 1D. t_{0H} , $C_L = 10pF$

FIGURE 1. THREE-STATE CIRCUITS AND WAVEFORMS

MEMORY ORGANIZATION

PROGRAM MEMORY

The MHS C51 Microcontroller Family has separate address spaces for program Memory and Data Memory. The program memory can be up to 64 K bytes long. The lower 4 K for the 80C51 (8 K for the 80C52, 16 K for the 83C154 and 32 K for the 83C154D) may reside on chip.

Figure 1 to 4 show a map of 80C51, 80C52, 83C154 and 83C154D program memory.

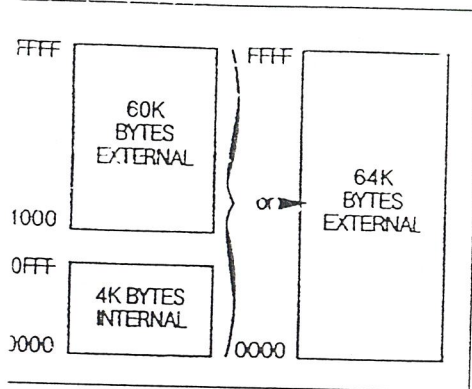


Figure 1 : The 80C51 Program Memory.

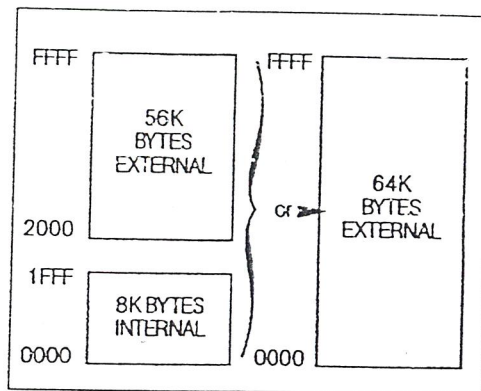


Figure 2 : The 80C52 Program Memory.

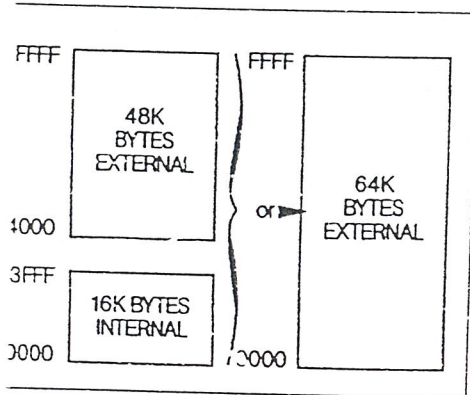


Figure 3 : The 83C154 Program Memory.

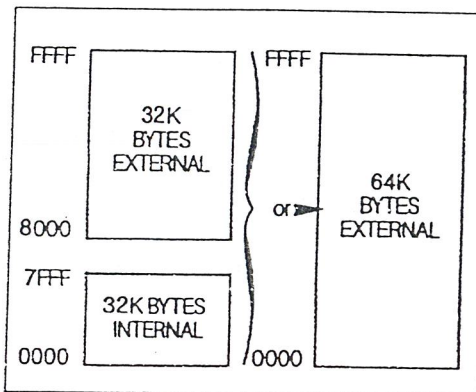


Figure 4 : The 83C154D Program Memory.

DATA MEMORY

The MHS C51 Microcontroller Family can address up to 64 K bytes of Data Memory to the chip. The * MOVX * instruction is used to access the external data memory (refer to the MHS C51 instruction set, in this chapter, for detailed description of instructions).

The 80C51 has 128 bytes of on-chip-RAM (256 bytes in the 80C52, 83C154 and 83C154D) plus a number of Special Function Registers (SFR). The lower 128 bytes of RAM can be accessed either by direct addressing (MOV data addr), or by indirect addressing (MOV @Ri). Figure 5 and 6 show the 80C51, 80C52, 83C154 and 83C154D Data Memory organization.

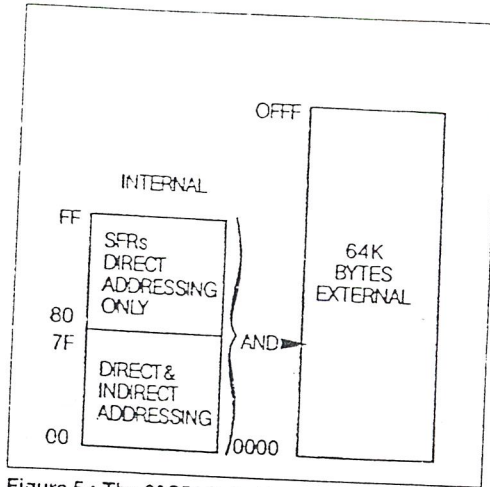


Figure 5 : The 80C51 Data Memory Organisation.

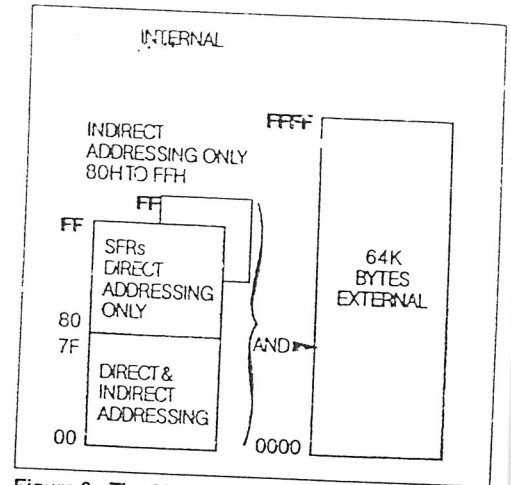


Figure 6 : The 80C52, 83C154 and 83C154D Data Memory Organisation.

INDIRECT ADDRESS AREA

Note that in Figure 6 - the SFRs and the indirect address RAM have the same addresses (80H-OFFH). Nevertheless, they are two separate areas and are accessed in two different ways. For example the instruction

```
MOV 80H, #0AAH
MOV R0, # 80H
MOV @ R0, # 0BBH
```

writes 0AAH to Port 0 which is one of the SFRs and the instruction writes 0BBH in location 80H of the data RAM. Thus, after execution of both of the above instructions Port 0 will contain 0AAH and location 80 of the RAM will contain 0BBH.

Note that the stack operations are examples of indirect addressing, so the upper 128 bytes of data RAM are available as stack space in those devices which implement 256 bytes of internal RAM.

DIRECT AND INDIRECT ADDRESS AREA

The 128 bytes of RAM which can be accessed by both direct and indirect addressing can be divided into 3 segments as listed below and shown in figure 7.

1. **Register Banks 0.3** : Locations 0 through 1FH (32 bytes). ASM-51 and the device after reset default to register bank 0. To use the other register banks the user must select them in the software. Each register bank contains 8 one-byte registers, 0 through 7. Reset initializes the Stack Pointer to location 07H and it is incremented once to start from location 08H which is the first register (R0) of the second register bank. Thus, in order to use more than one register bank, the SP should be initialized to a different location of the RAM where it is not used for data storage (ie, higher part of the RAM).
2. **Bit Addressable Area** : 16 bytes have been assigned for this segment, 20H-2FH. Each one of the 128 bits of this segment can be directly addressed (0-7FH). The bits can be referred to in two ways both of which are acceptable by the ASM-51. One way is to refer to their addresses, ie, 0 to 7FH. The other way is with reference to bytes 20H to 2FH. Thus, bits 0-7 can also be referred to as bits 20.0-20.7, and bits 8-15 are the same as 21.0-21.7 and so on. Each of the 16 bytes in this segment can also be addresses as a byte..

3. **Scratch Pad Area** : Bytes 30H through 7FH are available to user as data RAM. However, if the stack pointer has been initialized to this area, enough number of bytes should be left aside to prevent SP data destruction.

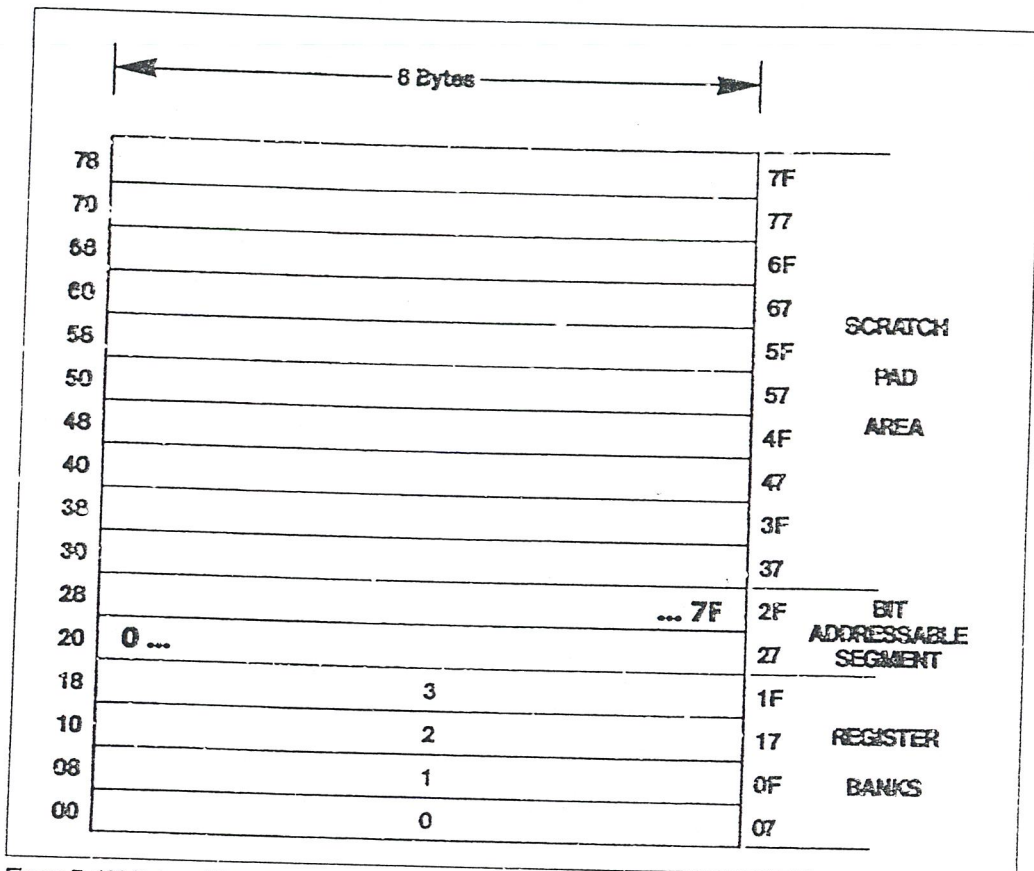


Figure 7. 128 Bytes of RAM Direct and Indirect Addressable.

These SFRs that have their bits assigned for various functions are listed in this section. A brief description of each bit is provided for quick reference. For more detailed information refer to the Architecture chapter of this book.

PSW : PROGRAM STATUS WORD (BIT ADDRESSABLE)

CY	AC	F0	RS1	RS0	OV	F1	P
----	----	----	-----	-----	----	----	---

- CY PSW.7 Carry Flag.
- AC PSW.6 Auxiliary Carry Flag.
- F0 PSW.5 Flag 0 available to the user for general purpose.
- RS1 PSW.4 Register Bank selector bit 1 (SEE NOTE).
- RS0 PSW.3 Register Bank selector bit 0 (SEE NOTE).
- OV PSW.2 Overflow Flag.
- F1 PSW.1 Flag F1 available to the user for general purpose.
- P PSW.0 Parity flag. Set/cleared by hardware each instruction cycle to indicate an odd/even number of '1' bits in the accumulator.

Note :

The value presented by RS0 and RS1 selects the corresponding register bank.

RS1	RS0	REGISTER BANK	ADDRESS
0	0	0	00H-07H
0	1	1	08H-0FH
1	0	2	10H-17H
1	1	3	18H-1FH

* User software should not write 1s to reserved bits. These bits may be used in future MHS C51 products to invoke new features. In that case, the reset or inactive value of the new bit will be 0, and its active value will be 1.

PCON : POWER CONTROL REGISTER (NOT BIT ADDRESSABLE)

SMOD	HPD	RPD	-	GF1	GF0	PD	IDL
------	-----	-----	---	-----	-----	----	-----

- SMOD PCON.7 Double baud rate bit. If SMOD = 1, the baud rate is doubled when the serial port is used in mode 1, 2 and 3.
- HPD PCON.6 Hard Power Down. (83C154 and 83C154D only). The falling/rising edge of a signal connected on pin P3.5 Starts/Stops the Power-Down mode. A reset can also stop this mode.
- RPD PCON.5 Recover Power Down bit. (83C154 and 83C154D only). It's used to cancel a Power-Down/IDLE mode. If it's set, an interrupt (enable or disable) can cancel this mode. A reset can also stop this mode (see Note 1).
- PCON.4 Not implemented, reserved for futur used*
- GF1 PCON.3 General purpose bit.
- GF0 PCON.2 General purpose bit.
- PD PCON.1 Power Down bit. If set, the oscillator is stopped. A reset or an interrupt (83C154 and 83C154D only) can cancel this mode (Note 1).
- IDL PCON.0 IDLE bit. If set the activity CPU is stopped. A reset or an interrupt can cancel this mode (See Note 1).

* User software should not write 1s to reserved bits. These bits may be used in future MHS C51 products to invoke new features. In that case, the reset or inactive value of the new bit will be 0, and its active value will be 1.

Note 1 (83C154 and 83C154D only) :

- if RPD = 0 and if an interrupt cancels the mode Power-Down/IDLE, the next instruction to execute is a LCALL at the interrupt routine.
- RPD = 1 - if interrupt request is enable the next instruction to execute is a LCALL at the interrupt routine.
 - if interrupt request is disable, the program continue with the instruction immediately after the Power-Down/Idle instruction.

INTERRUPTS

In order to use any of the interrupts in the MHS C51, the following three steps must be taken.

1. Set the EA (enable all) bit in the IE register to 1.
2. Set the corresponding individual interrupt enable bit in the IE register to 1.
3. Begin the Interrupt service routine at the corresponding Vector Address of that interrupt. See Table below.

INTERRUPT SOURCE	VECTOR ADDRESS
IE0	0003H
TF0	000BH
IE1	0013H
TF1	001BH
Ri & TI	0023H
TF2 & EXF2	002BH

In addition, for external interrupts, pins $\overline{INT0}$ and $\overline{INT1}$ (P3.2 and P3.3) must be set to 1, and depending on whether the interrupt is to be level or transition activated, bits IT0 or IT1 in the TCON register may need to be set to 1.

ITX = 0 level activated

ITX = 1 transition activated

IE : INTERRUPT ENABLE REGISTER (BIT ADDRESSABLE)

If the bit is 0, the corresponding interrupt is disabled. If the bit is 1, the corresponding interrupt is enabled.

EA	-	ET2	ES	ET1	EX1	ET0	EX0
----	---	-----	----	-----	-----	-----	-----

- EA IE.7 Disables all interrupts. If EA = 0, no interrupt will be acknowledged. If EA = 1, interrupt source is individually enable or disabled by setting or clearing its enable bit.
- IE.6 Not implemented, reserved for future use*.
- ET2 IE.5 Enable or disable the Timer2 overflow or capture interrupt (80C52, 83C154 and 83C154D only).
- ES IE.4 Enable or disable the Serial port interrupt.
- ET1 IE.3 Enable or disable the Timer 1 overflow interrupt.
- EX1 IE.2 Enable or disable External interrupt 1.
- ET0 IE.1 Enable or disable the Timer 0 overflow interrupt.
- EX0 IE.0 Enable or disable External Interrupt 0.

* User software should not write 1s to reserved bits. These bits may be used in future MHS C51 products to invoke new features. In that case, the reset or inactive value of the new bit will be 0, and its active value will be 1.

ASSIGNING HIGHER PRIORITY TO ONE MORE INTERRUPTS

In order to assign higher priority to an interrupt the corresponding bit in the IP register must be set to 1. Remember that while an interrupt service is in progress, it cannot be interrupted by a lower or same level interrupt.

PRIORITY WITHIN LEVEL

Priority within level is only to resolve simultaneous requests of the same priority level. From high to low, interrupt sources are listed below :

- IE0
- TF0
- IE1
- TF1
- R1 or T1
- TF2 or EXF2

IP : INTERRUPT PRIORITY REGISTER (BIT ADDRESSABLE)

If the bit is 0, the corresponding interrupt has a lower priority and if the bit is the corresponding interrupt has a higher priority.

PCT	-	PT2	PS	PT1	PX1	PT0	PX0
-----	---	-----	----	-----	-----	-----	-----

- PCT IP.7 Defines the same priority level for all the source interrupt (83C154 and 83C154D only).
- IP.6 Not implemented, reserved for future use*.
- PT2 IP.5 Defines the Timer 2 interrupt priority level (80C52, 83C154 and 83C154D only).
- PS IP.4 Defines the Serial Port interrupt priority level.
- PT1 IP.3 Defines the Timer 1 Interrupt priority level.
- PX1 IP.2 Defines External Interrupt priority level.
- PT0 IP.1 Defines the Timer 0 interrupt priority level.
- PX0 IP.0 Defines the External Interrupt 0 priority level.

* User software should not write 1s to reserved bits. These bits may be used in future MHS C51 products to invoke new features. In that case, the reset or inactive value of the now bit will be 0, and its active value will be 1.

IOCON : INPUT/OUTPUT CONTROL REGISTER (83C154 and 83C154D only)

WDT	T32	SERR	IZC	P3HZ	P2HZ	P1HZ	ALF
-----	-----	------	-----	------	------	------	-----

- WDT IOCON.7 Watch Dog Timer bit. Set when Timer 1 is overflow (TF = 1). The CPU is reset and the program is executed from address 0.
- T32 IOCON.6 Timer 32 bits. The Timer 1 and Timer 0 are connected together to form a 32 bits Timer/Counter. If C/TO = 0, it's a Timer. if C/TO = 1, it's a counter.
- SERR IOCON.5 Serial Port Reception Error flag. Set when an overrun on frame error is received.
- IZC IOCON.4 Set/Cleared by software to select 100/10 K pull up resistance for Port 1, 2 and 3.
- P3HZ IOCON.3 When Set, Port 3 becomes a tri-state input. When cleared, the pull-up resistance value is selected by IZC.
- P2HZ IOCON.2 When Set, Port 2 becomes a tri-state input. When cleared, the pull-up resistance value is selected by IZC.
- P1HZ IOCON.1 When Set, Port 1 becomes a tri-state input. When cleared, the pull-up resistance value is selected by IZC.
- ALF IOCON.0 All Port tri-state. When Set and CPU in Power-Down mode, port 1, 2 and 3 are tri-state.

TCON : TIMER/COUNTER CONTROL REGISTER (BIT ADDRESSABLE)

TF1	TR1	TF0	TR0	IE1	IT1	IE0	IT0
-----	-----	-----	-----	-----	-----	-----	-----

- TF1 TCON.7 Timer 1 overflow flag. Set by hardware when the Timer/Counter 1 overflows. Cleared by hardware as processor vectors to the interrupt service routine.
- TR1 TCON.6 Timer 1 run control bit. Set/cleared by software to turn Timer/Counter ON/OFF.
- TF0 TCON.5 Timer 0 overflow flag. Set by hardware when the Timer/Counter 0 overflows. Cleared by hardware as processor vectors to the service routine.
- TR0 TCON.4 Timer 0 run control bit. Set/cleared by software to turn Timer/Counter 0 ON/OFF.
- IE1 TCON.3 External Interrupt 1 edge flag. Set by hardware when External interrupt edge is detected. Cleared by hardware when interrupt is processed.
- IT1 TCON.2 Interrupt 1 type control bit. Set/cleared by software to specify falling edge/low level triggered External Interrupt.
- IE0 TCON.1 External Interrupt 0 edge flag. Set by hardware when External Interrupt edge detected. Cleared by hardware when interrupt is processed.
- IT0 TCON.0 Interrupt 0 type control bit. Set/cleared by software to specify falling edge/low level triggered External Interrupt.

TMOD : TIMER/COUNTER MODE CONTROL REGISTER (NOT BIT ADDRESSABLE)

GATE	C/T	M1	M0	GATE	C/T	M1	M0
TIMER 1				TIMER 0			

- GATE When TRx (in TCON) is set and GATE = 1, TIMER/COUNTERx will run only while INTx pin is high (hardware control). When GATE = 0, TIMER/COUNTERx will run only while TRx = 1 (software control).
- C/T Timer or Counter selector. Cleared for Timer operation (input from internal system clock). Set for Counter operation (input from Tx input pin).
- M1 Mode selector bit (NOTE 1).
- M0 Mode selector bit (NOTE 1).

NOTE 1 :

M1	M0	OPERATING MODE
0	0	0 13-bit Timer
0	1	1 16-bit Timer/Counter
1	0	2 8-bit Auto-Reload Timer/Counter
1	1	3 (Timer 0) TL0 is an 8-bit Timer/Counter controlled by the standard Timer 0 control bits, TH0 is an 8-bit Timer and is controlled by Timer 1 control bits.
1	1	3 (Timer 1) Timer/Counter 1 stopped.

TIMER SET-UP

Tables 3 through 6 give some values for TMOD which can be used to set up Timer 0 in different modes. It is assumed that only one timer is being used at a time. It is desired to run Timers 0 and 1 simultaneously, in any mode, the value that in TMOD for Timer 0 must be ORed with the value shown for Timer 1 (Tables 5 and 6). For example, if it is desired to run Timer 0 in mode 1 GATE (external control) and Timer 1 in mode 2 COUNTER, then the value must be loaded into TMOD is 69H (09H from Table 3 ORed with 60H from Table 6). Moreover, it is assumed that the user, at this point, is not ready to turn the timers on and will do that a different point in the program by setting bit TRx (in TCON) to 1.

TIMER/COUNTER 0

MODE	TIMER 0 FUNCTION	TMOD	
		INTERNAL CONTROL (NOTE 1)	EXTERNAL CONTROL (NOTE 2)
0	13-bit Timer	00H	03H
1	16-bit Timer	01H	09H
2	8-bit Auto-Reload	02H	0AH
3	Two 8-bit Timers	03H	0BH

As a Timer : Table 3

MODE	TIMER 0 FUNCTION	TMOD	
		INTERNAL CONTROL (NOTE 1)	EXTERNAL CONTROL (NOTE 2)
0	13-bit Timer	04H	0CH
1	16-bit Timer	05H	0DH
2	8-bit Auto-Reload	06H	0EH
3	one 8-bit counter	07H	0FH

As a Counter : Table 4

- Notes : 1. The Timer is turned ON/OFF by setting/clearing bit TR0 in the software.
 2. The Timer is turned ON/OFF by the 1 to 0 transition on INT0 (P3.2) when TR0 = 1 (hardware control).

TIMER/COUNTER 1

MODE	TIMER 1 FUNCTION	TMOD	
		INTERNAL CONTROL (NOTE 1)	EXTERNAL CONTROL (NOTE 2)
0	13-bit Timer	00H	80H
1	16-bit Timer	10H	90H
2	8-bit Auto-Reload	20H	A0H
3	does not run	30H	B0H

As a Timer : Table 5

MODE	COUNTER 1 FUNCTION	TMOD	
		INTERNAL CONTROL (NOTE 1)	EXTERNAL CONTROL (NOTE 2)
0	13-bit Timer	40H	C0H
1	16-bit Timer	50H	D0H
2	8-bit Auto-Reload	60H	E0H
3	not available	-	-

As a Counter : Table 6

- Notes : 1. The Timer is turned ON/OFF by setting/clearing bit TR1 in the software.
 2. The Timer is turned ON/OFF by the 1 to 0 transition on INT1 (P3.2) when TR1 = 1 (hardware control).

T2CON : TIMER/COUNTER 2 CONTROL REGISTER (BIT ADDRESSABLE)
(80C52, 83C154 and 83C154D only)

TF2	EXF2	RCLK	TCLK	EXEN2	TR2	C/T2	CP/RL2
-----	------	------	------	-------	-----	------	--------

- TF2 T2CON.7 Timer 2 overflow flag set by hardware and cleared by software. TF2 cannot be set when either RCLK = 1 or CLK = 1
- EXF2 T2CON.6 Timer 2 external flag set when either a capture or reload is caused by a negative transition on T2EX, and EXEN2 = 1. When Timer 2 interrupt is enabled, EXF2 = 1 will cause the CPU to vector to the Timer 2 interrupt routine. EXF2 must be cleared by software.
- RCLK T2CON.5 Receive clock flag. When set, causes the Serial Port to use Timer 2 overflow pulses for its receive clock in modes 1 & 3. RCLK = 0 causes Timer 1 overflow to be used for the receive clock.
- TCLK T2CON.4 Transmit clock flag. When set, causes the Serial Port use Timer 2 overflow pulses for its transmit clock in modes 1 & 3, TCLK = 0 causes Timer 1 overflows to be used for the transmit clock.
- EXEN2 T2CON.3 Timer 2 external enable flag. When set, allows a capture or reload to occur as a result of negative transition on T2EX if Timer 2 is not being used to clock the Serial Port. EXEN2 = 0 causes Timer 2 to ignore events as T2EX.
- TR2 T2CON.2 Software START/STOP control for Timer 2. A logic 1 starts the Timer.
- C/T2 T2CON.1 Timer or Counter select.
- CP/RL2 T2CON.0 Capture/Reload flag. When set, captures will occur on negative transitions at T2EX if EXEN2 = 1. When cleared, Auto-Reloads will occur either with Timer2 overflows or negative transitions at T2EX when EXEN2 = 1. When either RCLK = 1 or TCLK = 1, this bit is ignored and the Timer is forced to Auto-Reload on Timer 2 overflow.

TIMER/COUNTER 2 SET-UP

Except for the baud rate generator mode, the values given for T2CON do not include the setting of the TR2 bit. Therefore, bit TR2 must be set, separately, to turn the Timer on.

MODE	T2CON	
	INTERNAL CONTROL (NOTE 1)	EXTERNAL CONTROL (NOTE 2)
16-bit Auto-Reload	00H	08H
16-bit Capture	01H	09H
BAUD rate generator receive & transmit same baud rate	34H	36H
receive only	24H	26H
transmit only	14H	16H

As a Timer : Table 7

MODE	TMOD	
	INTERNAL CONTROL (NOTE 1)	EXTERNAL CONTROL (NOTE 2)
16-bit Auto-Reload	02H	0AH
16-bit Capture	03H	0BH

As a Counter : Table 8

- Notes :
1. Capture/Reload occurs only Timer/Counter overflow.
 2. Capture/Reload occurs on Timer/Counter overflow and a 1 to 0 transition on T2EX (P1.1) pin except when Timer 2 is used in the baud rate generating mode.

SCON : SERIAL PORT CONTROL REGISTER (BIT ADDRESSABLE)

	SM0	SM1	SM2	REN	TN8	RB8	TI	RI
SM0	SCON.7 Serial Port mode specifier (NOTE 1).							
SM1	SCON.6 Serial Port mode specifier (NOTE 1).							
SM2	SCON.5 Enables the multiprocessor communication feature in mode 2 & 3. In mode 2 or 3, if SM2 is set to 1 then RI will not be activated if the received 9th data bit (RB8) is 0. In mode 1, if SM2 = 1 then RI will not be activated if a valid stop bit was not received. In mode 0, SM2 should be 0 (See table 9).							
REN	SCON.4 Set/Cleared by software to Enable/Disable reception.							
TB8	SCON.3 The 9th bit that will be transmitted in modes 2 & 3. Set/Cleared by software.							
RB8	SCON.2 In modes 2 & 3, is the 9th data bit that was received. In mode 1, if SM2 = 0, RB8 is the stop bit that was received. In mode 0, RB8 is not used.							
TI	SCON.1 Transmit interrupt flag. Set by hardware at the end of the 8th bit time in mode 0, or at the beginning of the stop bit in the other modes. Must be cleared by software.							
RI	SCON.0 Receive interrupt flag. Set by hardware at the end of the 8th bit time in mode 0, or half way through the stop bit time in the other modes (except see SM2). Must be cleared by software.							

NOTE 1 :

SM0	SM1	MODE	DESCRIPTION	BAUD RATE
0	0	0	SHIFT REGISTER	Fosc./12
0	1	1	8 bit UART	Variable
1	0	2	8 bit UART	Fosc./64 OR Fosc./32
1	1	3	8 bit UART	Variable

SERIAL PORT SET-UP

MODE	SCON	SM2 VARIATION
0	10H	Single Processor Environment (SM2 = 0)
1	50H	
2	90H	
3	D0H	
0	NA	Multiprocessor Environment (SM2 = 1)
1	70H	
2	B0H	
3	FOH	

Table 9

GENERATING BAUD RATES

Serial Port in Mode 0 :

Mode 0 has a fixed baud rate which is 1/12 of oscillator frequency. To run serial port in this mode none of the Timer/Counters need to be set up. Only the SCON register needs to be defined.

$$\text{Baud Rate} = \frac{\text{Osc Freq}}{12}$$

Serial Port in Mode 1 :

Mode 1 has a variable baud rate. The baud rate can be generated by either Timer 1 or Timer 2 (80C52, 83C154 and 83C154D only).

USING TIMER/COUNTER 1 TO GENERATE BAUD RATES

For this purpose, Timer 1 is used in mode 2 (Auto-Reload). Refer to Timer Setup section of this chapter.

$$\text{Baud Rate} = \frac{K \times \text{Oscillator Freq.}}{32 \times 12 \times [256 - (\text{TH1})]}$$

if SMOD = 0, then K = 1.

If SMOD = 1, then K = 2. (SMOD is the PCON register).

Most of the time the user knows the baud rate and needs to know the reload value for TH1. Therefore, the equation to calculate TH1 can be written as :

$$\text{TH1} = 256 - \frac{K \times \text{Osc Freq.}}{384 \times \text{baud rate}}$$

TH1 must be integer value. Rounding off TH1 to the nearest integer may not produce the desired baud rate. In this case, the user may have to choose another crystal frequency.

Since the PCON register is not bit addressable, one way to set the bit is logical ORing the PCON register (ie, ORL PCON, #80H). The address of PCON is 87H.

USING TIMER/COUNTER 2 TO GENERATE BAUD RATES

For this purpose, Timer 2 must be used in the baud rate generating mode. Refer to Timer 2 Setup Table in this chapter. If Timer 2 is being clocked through pin T2 (P1.0) the baud rate is :

$$\text{Baud Rate} = \frac{\text{Timer 2 Overflow Rate}}{16}$$

And if it being clocked internally the baud rate is :

$$\text{Baud Rate} = \frac{\text{Osc Freq.}}{32 \times [65536 - (\text{RCAP2H}, \text{RCAP2L})]}$$

To obtain the reload value for RCAP2H and RCAP2L the above equation can be written as :

$$\text{RCAP2H}, \text{RCAP2L} = 65536 - \frac{\text{Osc Freq.}}{32 \times \text{Baud Rate}}$$

SERIAL PORT IN MODE 2

The baud rate is fixed in this mode and 1/32 or 1/64 of the oscillator frequency depending on the value of the SMOD bit in the PCON register.

In this mode none of the Timers are used and the clock comes from the internal phase 2 clock.

SMOD = 1, Baud Rate = 1/32 Osc Freq.

SMOD = 0, Baud Rate = 1/64 Osc Freq.

To set the SMOD bit : ORL PCON, #80H. The address of PCON is 87H.

SERIAL PORT IN MODE 3

The baud rate in mode 3 is variable and sets up exactly the same as in mode 1.

MKS C51 INSTRUCTION SET

Interrupt Response time : Refer to Hardware Description Chapter.

Instructions that Affect Flag Settings (1)

INSTRUC.	FLAG			INSTRUC.	FLAG		
	C	OV	AC		C	OV	AC
ADD	X	X	X	CLRC			O
ADDC	X	X	X	CPL C			X
SUBB	X	X	X	ANL C, bit			X
MUL	0	X		ANL C, /bit			X
DIV	0	X		ORL C, bit			X
DA	X			ORL C, bit			X
RRC	X			MOV C, bit			X
RLC	X			CJNE			X
SETB C	1						

(1) note that operations on SFR byte address 208 or bit addresses 209-215 (i.e., the PSW or bits in the PSW) will also affect flag settings.

Note on instruction set and addressing modes :

Rn - Register R7-R0 of the currently selected Register Bank

direct - 8-bit internal data location's address. This could be an Internal Data RAM location (0-127) or a SFR (i.e., I/O port, control register, status register, etc. (128-255)).

@ Ri - 8-bit internal data RAM location (0-255) addresses indirectly through register R1 or R0.

= data - 8-bit constant included in instruction.

= data 16 - 16-bit constant included in instruction.

addr 16 - 16-bit destination address. Used by LCALL & LJMP. Abranch can be anywhere within the 64K-byte Program memory address space

addr 11 - 11-bit destination address. Used by ACALL & AJMP. The branch will be within the same 2K-byte page of program memory as the first byte of the following instruction

rel - Signed (two's complement) 8-bit offset byte. Used by SJMP and all conditionnal jumps. Range is -128 to + 127 bytes relative to first byte of the following instruction.

bit - Direct Addressed bit in internal Data RAM or special Function Register.

Table 10 : 80C51 Instruction Set Summary.

MNEMONIC	DESCRIPTION	BYTE	OSCIL. PERIOD
ARITHMETIC OPERATIONS			
ADD A, Rn	Add register to Accumulator	1	12
ADD A, direct	Add direct byte to Accumulator	2	12
ADD A, @Ri	Add indirect RAM to Accumulator	1	12
ADD A, #data	Add immediate data to Accumulator	2	12
ADDC A, Rn	Add register to Accumulator with Carry	1	12
ADDC A, direct	Add direct byte to Accumulator with Carry	2	12
ADDC A, @Ri	Add indirect RAM to Accumulator with Carry	1	12
ADDC A, #data	Add immediate data to Acc with Carry	2	12
SUBB A, Rn	Subtract Register from Acc with borrow	1	12
SUBB A, direct	Subtract direct byte from Acc with borrow	2	12
SUBB A, @Ri	Subtract indirect RAM from ACC with borrow	1	12
SUBB A, #data	Subtract immediate data from Acc with borrow	2	12
INC A	Increment Accumulator	1	12
INC Rn	Increment register	1	12
INC direct	Increment direct byte	2	12
INC @Ri	Increment direct RAM	1	12
DEC A	Decrement Accumulator	1	12
DEC Rn	Decrement Register	1	12
DEC direct	Decrement direct byte	2	12
DEC @Ri	Decrement indirect RAM	1	12

MHS C51

MNEMONIC	DESCRIPTION	BYTE	OSCIL. PERIOD
ARITHMETIC OPERATIONS (continued)			
INC DPTR	Increment Data Pointer	1	24
MUL AB	Multiply A & B	1	48
DIV AB	Divide A by B	1	48
DA A	Decimal Adjust Accumulator	1	12
LOGICAL OPERATIONS			
ANL A, Rn	AND Register to Accumulator	1	12
ANL A, direct	AND direct byte to Accumulator	2	12
ANL A, @Ri	AND indirect RAM to Accumulator	1	12
ANL A, #data	AND immediate data to Accumulator	2	12
ANL direct, A	AND Accumulator to direct byte	2	12
ANL direct, #data	AND immediate data to direct byte	3	24
ORL A, Rn	OR register to Accumulator	1	12
ORL A, direct	OR direct byte to Accumulator	2	12
ORL A, @Ri	OR indirect RAM to Accumulator	1	12
ORL A, #data	OR immediate data to Accumulator	2	12
ORL direct, A	OR Accumulator to direct byte	2	12
ORL direct, #data	OR immediate data to direct byte	3	24
XRL A, Rn	Exclusive-OR register to Accumulator	1	12
XRL A, direct	Exclusive-OR direct byte to accumulator	2	12
XRL A, @Ri	Exclusive-OR indirect RAM to Accumulator	1	12
XRL A, #data	Exclusive-OR immediate data to Accumulator	2	12
XRL direct, A	Exclusive-OR Accumulator to direct byte	2	12
XRL direct, #data	Exclusive-OR immediate data to direct byte	3	24
CLR A	Clear Accumulator	1	12
CPL A	Complement Accumulator	1	12

MNEMONIC	DESCRIPTION	BYTE	OSCIL. PERIOD
LOGICAL OPERATIONS (continued)			
RL A	Rotate Accumulator Left	1	12
RLC A	Rotate Accumulator Left through the Carry	1	12
RR A	Rotate Accumulator Right	1	12
RRC A	Rotate Accumulator Right through the Carry	1	12
SWAP A	Swap nibbles within the Accumulator	1	12
DATA TRANSFER			
MOV A, Rn	Move Register to Accumulator	1	12
MOV A, direct	Move direct byte to Accumulator	2	12
MOV A, @Ri	Move indirect RAM to Accumulator	1	12
MOV A, #data	Move immediate data to Accumulator	2	12
MOV Rn, A	Move Accumulator to register	1	12
MOV Rn, direct	Move direct byte to register	2	24
MOV Rn, #data	Move immediate data to register	2	12
MOV direct, A	Move Accumulator to direct byte	2	12
MOV direct, Rn	Move register to direct byte	2	24
MOV direct, direct	Move direct byte to direct	3	24
MOV direct, @Ri	Move indirect RAM to direct byte	2	24
MOV direct, #data	Move immediate data to direct byte	3	24
MOV @Ri, A	Move Accumulator to indirect RAM	1	12

MNEMONIC	DESCRIPTION	BYTE	OSCIL. PERIOD
DATA TRANSFER (continued)			
MOV @Ri, direct	Move direct by to indirect RAM	2	24
MOV @Ri, #data	Move immediate data to indirect RAM	2	12
MOVDPTR, #data16	Load Data Pointer with a 16-bit constant	3	24
MOVCA, @A+DPTR	Move Code byte relative to DPTR to Acc	1	24
MOVCA, @A+PC	Move Code byte relative to PC to Acc	1	24
MOVXA, @Ri	Move External RAM (8-bit addr) to Acc	1	24
MOVX A, @DPTR	Move External RAM (16-bit addr) to Acc	1	24
MOVX @Ri, A	Move Acc to External RAM (8-bit addr)	1	24
MOVX @DPTR, A	Move Acc to External RAM (16-bit addr)	1	24
PUSH direct	Push direct byte only stack	2	24
POP direct	Pop direct byte from stack	2	24
XCH A, Rn	Exchange register with Accumulator	1	12
XCH A, direct	Exchange direct byte with Accumulator	2	12
XCH A, @Ri	Exchange indirect RAM with Accumulator	1	12
XCHD A, @Ri	Exchange loworder Digit indirect RAM with Acc	1	12
BOOLEAN VARIABLE MANIPULATION			
CLR C	Clear Carry	1	12
CLR bit	Clear direct bit	2	12
SETB C	Set Carry	1	12
SETB bit	Set direct bit	2	12
CPL C	Complement Carry	1	12
CPL bit	Complement direct bit	2	12
ANL C, bit	AND direct bit to Carry	2	24
ANL C, /bit	AND complement of direct bit to Carry	2	24
ORL C, bit	OR direct bit to Carry	2	24
ORL C, /bit	OR complement of direct bit to Carry	2	24
MOV C, bit	Move direct bit to Carry	2	12
MOV bit, C	Move Carry to direct bit	2	24
JC rel	Jump if Carry is set	2	24
JNC rel	Jump if Carry not set	2	24
JB bit, rel	Jump if direct Bit is set	3	24
JNB bit, rel	Jump if direct Bit is Not set	3	24
JBC bit, rel	Jump if direct Bit is set & clear bit	3	24
PROGRAM BRANCHING			
ACALLK addr11	Absolute Subroutine Call	2	24
LCALL addr16	Long Subroutine Call	3	24
RET	Return from Subroutine	1	24
RETI	Return from interrupt	1	24
AJMPaddr11	Absolute Jump	2	24
LJMPaddr16	Long Jump	3	24
SJMP rel	Short Jump (relative addr)	2	24

MHS C51

MNEMONIC	DESCRIPTION	BYTE	OSCIL. PERIOD
PROGRAM BRANCHING (continued)			
JMP @A+DPTR	Jump direct relative to the DPTR	1	24
JZ rel	Jump if Accumulator is zero	2	24
JNZ rel	Jump if Accumulator is not Zero	2	24
CNJE A, direct, rel	Compare direct byte to Acc and Jump if Not Equal	3	24
CJNE A, #data, rel	Compare immediate to Acc and Jump if Not Equal	3	24

MNEMONIC	DESCRIPTION	BYTE	OSCIL. PERIOD
PROGRAM BRANCHING (continued)			
CJNE Rn, #data, rel	Compare immediate to register and Jump if Not Equal	3	24
CJNE @Ri, #data, rel	Compare immediate to indirect and Jump if Not Equal	3	24
DJNZ Rn, rel	Decrement register and Jump if Not Zero	2	24
DJNZ direct, rel	Decrement direct byte and Jump if Not Zero	3	24
NOP	No Operation	1	12

กิตติกรรมประกาศ

การจัดทำปริญญานิพนธ์นี้สามารถสำเร็จได้ด้วยดี เนื่องจากได้รับความช่วยเหลือจากอาจารย์หลายๆท่าน คณะผู้จัดทำโครงการผู้ศึกษาซึ่งและขอกราบขอบพระคุณ อาจารย์สาท คำมูล และอาจารย์ทุกๆท่านในภาควิชาเทคโนโลยีการควบคุมทางอุตสาหกรรมไว้เป็นอย่างสูง

ขอขอบพระคุณภาควิชาเทคโนโลยีการควบคุมทางอุตสาหกรรม ที่เอื้อเฟื้ออุปกรณ์และเครื่องมือทั้งหมด ตลอดจนสถานที่ใช้ในการทำงาน

ขอขอบพระคุณเพื่อนๆ ทุกคน (ออย, แวน ,ศุน , คต , คัม ฯลฯ) ซึ่งเป็นที่ปรึกษา เป็นกำลังใจที่ดี และเป็นเรี่ยวแรงสำคัญในการทำโครงการครั้งนี้

และกราบขอบพระคุณ คุณพ่อ คุณแม่ ที่เคารพรักยิ่ง ที่ให้การสนับสนุน เป็นกำลังใจที่ยิ่งใหญ่ และกำลังทุนทรัพย์ที่สำคัญตลอดการทำปริญญานิพนธ์นี้

สุดท้ายนี้ หากปริญญานิพนธ์ฉบับนี้มีข้อผิดพลาดและข้อบกพร่องประการใด คณะผู้จัดทำขออภัยไว้ ณ ที่นี้ด้วย

คณะผู้จัดทำ

เมษายน 2543

บรรณานุกรม

1. รศ.สมยศ จุณณะปิยะ, การประยุกต์ใช้งานไมโครคอนโทรลเลอร์ตระกูล MCS-51.สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
2. จักรพงษ์ สุขประเสริฐ, กนกพร ภาวสุทธิกุล, สัจจะ จรัสรุ่งรวิวรร, Delphi4.0.กรุงเทพฯ: บริษัทซีเอ็ดยูเคชั่น จำกัด (มหาชน), 2542
3. กนก กุศลมาลย์นุกูล, ไกรวุฒิ มั่นเสถียรสิน, Delphi 4 .กรุงเทพฯ: บริษัทซัคเซสมิเดีย จำกัด, 2542
4. กฤษดา วิศวธีรานนท์, ไอซีดีจิตอล.กรุงเทพฯ: บริษัทซีเอ็ดยูเคชั่น จำกัด(มหาชน), 2521
5. ทวีชัย ภูริทิพย์, ไขปัญหา RS-232.กรุงเทพฯ: บริษัทซีเอ็ดยูเคชั่น จำกัด(มหาชน), 2538
6. สุนทร วิทูสรพจน์, การใช้งานไมโครคอนโทรลเลอร์ ตระกูล 8051.กรุงเทพฯ: บริษัทซีเอ็ดยูเคชั่น จำกัด(มหาชน), 2537