เครื่องชั่งน้ำหนักแบบละเอียด

FINE SCALE

จัดทำโดย

38013395	นาย	จักรกริช	ฉิมพลีปักษ์
38013400	นาย	เชาว์	เกษมกุล
38013435	นาย	อมรเทพ	ขันทอง

วัน เดือน ปี -5 ต.ค. 2541
เลขทะเบียน 038510
เลขเรียกหนังสือ T 400 ...

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญา
วิศวกรรมศาสตร์ สาขาวิศวกรรมการวัดและควบคุมอุตสาหกรรม
คณะวิศวกรรมศาสตร์
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
ปีการศึกษา 2540

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

038510

ภาควิชา เทคโนโลยีการวัดและควบคุมอุตสาหกรรม

สาขาวิชา วิศวกรรมการวัดและควบคุมอุตสาหกรรม

คณะวิศวกรรมศาสตร์

สถาบันเทคโนโลยีการวัดและควบคุมอุตสาหกรรม

เรื่อง เครื่องชั่งน้ำหนักแบบละเอียด (FINE SCALE)

จัดทำโดย

38013395	นาย จักรกริช	ฉิมพลีปักษ์
38013400	นาย เชาว์	เกษมกุล
38013435	นาย อมรเทพ	ขันทอง

..... อาจารย์ที่ปรึกษา

(อาจารย์วิริยะ กองรัตน์)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หัวข้อปริญญานิพนธ์	เครื่องชั่งละเอียด (FINE SCALE)
นักศึกษา	นาย จักรกริช ฉิมพลีปักษ์
	นาย เชาว์ เกษมกุล
	นาย อมรเทพ ขันทอง
อาจารย์ที่ปรึกษา	อาจารย์ วิริยะ กองรัตน์
ระดับการศึกษา	วิศวกรรมศาสตร์
ปีการศึกษา	2540

บทคัดย่อ

ในปัจจุบันระบบไมโครคอมพิวเตอร์ เข้ามามีบทบาทมากในชีวิตประจำวัน ในระบบไมโครคอมพิวเตอร์จะสามารถนำมาดัดแปลง ให้ข้อมูลต่าง ๆเป็นลักษณะตามที่ต้องการ เช่น การนำข้อมูลเก็บไว้ในแบบของ Digital การนำข้อมูลมานำเสนอในรูปแบบต่าง ๆ

เครื่องชั่งละเอียดนี้ก็เป็นการนำไมโครคอมพิวเตอร์มาใช้ประโยชน์โดยการนำข้อมูลทางไฟฟ้ามาแสดงผลค่าของน้ำหนักในแบบแสดงเป็นตัวเลข และเก็บข้อมูลนั้นได้รวมทั้งนำข้อมูลมาประมวลผลได้อีกด้วย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

THESIS	FINE SCALE	
STUDENT	Mr. JAKKARIT	CHIMPLEEPUK
	Mr.CHOA	KASAMKUL
	Mr.AMONTHAPE	KANTHONG
ADVISOR	Mr.VIRIYA	KONGRAT
EDUCATION LEVEL	Bachelor of industrial instrumentation engineering	
EDUCATION YEAR	1997	

ABSTRACT

For now microcomputer system is populer. The microcomputer can be data applicate, such as save data, digital display.

The FINE SCALE is touse the microcomputer system, by transduce voltage to digital display and save data for applicate.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ

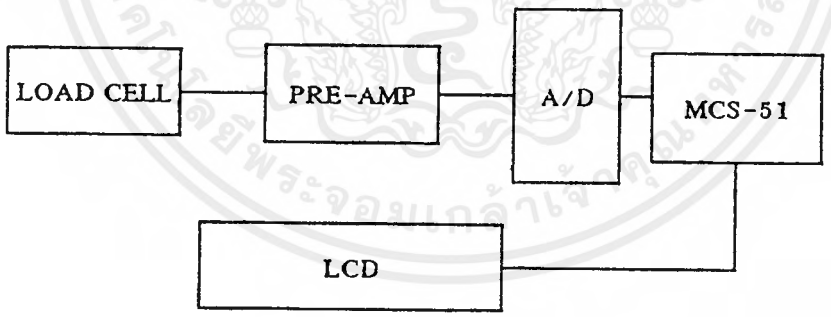
หัวเรื่อง	หน้า
บทนำ	1
ทฤษฎีเบื้องต้น	4
เกจวัดความเครียด	4
การติดต่อกับไมโคลคอนโทลเลอร์	19
การควบคุม ADC	20
Z80 ไมโครโปรเซสเซอร์	25
Operational Amplifier	34
ความรู้พื้นฐานของ DOT MATRIX LCD MODULE	38
การต่อสวิชท์จำนวนมาก (Key Board)	42
รายละเอียดส่วนต่าง ๆของเครื่องชั่งแบบละเอียด	46
- FLOW CHART	47
- ส่วนของ SOFT WARE	50
- Pre-Amplier	80
- วิธีการใช้เครื่องชั่งแบบละเอียด	83
กิตติกรรมประกาศ	
บรรณานุกรม	

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทนำ

ในปัจจุบันอุตสาหกรรมและการควบคุมต่างๆ ได้พัฒนาไปมากจากเดิมที่ส่วนใหญ่จะเป็นแบบธรรมดามาเป็นแบบกึ่งอัตโนมัติและเป็นแบบอัตโนมัติ ดังนั้นอุปกรณ์เครื่องมือวัดต่างๆ ที่ใช้ในการประกอบการเหล่านี้ จะต้องได้รับการพัฒนาตามไปด้วย จะเห็นได้ว่าในปัจจุบัน เครื่องคอมพิวเตอร์ได้เข้ามามีบทบาทอย่างมากไม่ว่า งานด้านเอกสาร หรืองานด้านการควบคุมอุตสาหกรรมที่เรามักพบได้เสมอ ดังนั้นแนวทางการพัฒนาอุปกรณ์ส่วนใหญ่จึงมีทิศทางเดียวกัน นั่นคือการทำทุกอย่างให้ใช้งานร่วมกันได้ โดยการใช้คอมพิวเตอร์เป็นตัวเชื่อมเข้าด้วยกัน

เครื่องชั่งน้ำหนักแบบละเอียดนี้มีหลักการทำงาน เช่นเดียวกับเครื่องชั่งน้ำหนักแบบทั่วๆไป คือมีส่วนที่รับค่าน้ำหนักเข้ามาและมีส่วนแสดงผล ส่วนที่รับค่าน้ำหนักเข้ามาจะประกอบไปด้วย LOAD CELL วงจร Pre-Amplifier ทำหน้าที่ในการขยายสัญญาณและกำหนดค่าสัญญาณที่เหมาะสม ให้กับวงจร ADC เพื่อทำการแปลงสัญญาณอนาลอกเป็นสัญญาณดิจิตอล แล้วส่งไปยังส่วนแสดงผลด้วยการส่งข้อมูลแบบอนุกรม แล้วทำการแสดงผลด้วยจอภาพ LCD การแสดงผลจะใช้ MCS-51 เป็นตัวทำการประมวลผลที่ได้ออกมาเป็นตัวเลข และการพัฒนาโปรแกรมสามารถทำได้จากการโปรแกรมของ MCS-51



บล็อกไดอะแกรมแสดงส่วนประกอบของเครื่องชั่ง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

<u>วัตถุประสงค์</u>

1.เพื่อนำความรู้มาประยุกต์ได้

2.เพื่อเพิ่มทักษะทางด้านต่างๆ

3.เพื่อการศึกษาคุณสมบัติและวิเคราะห์อุปกรณ์ทรานสดิ้วเซอร์

4.เพื่อการศึกษาและพัฒนาการโปรแกรมไมโครคอนโทรลเลอร์

<u>ผลที่คาดว่าจะได้รับ</u>

1.ได้นำความรู้ทางทฤษฎีมาทดลองและประยุกต์ใช้ได้จริง

2.ได้ศึกษาถึงปัญหาในการปฏิบัติงานจริง และแก้ไขปัญหาเหล่านั้นได้

3.ได้ผลิตเครื่องมือทางเทคโนโลยีที่สามารถใช้ประโยชน์ได้ต่อไป

4.ได้ศึกษาและคุ้นเคยในการทำงานเป็นกลุ่ม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ขอบเขตของ Project

1.สามารถชั่งน้ำหนักที่ความละเอียดเป็นมิลลิกรัม(0.00-800.00)

2.สามารถแสดงผลการชั่งน้ำหนักเป็นตัวเลขขอ Display

3.สามารถรองรับการทำงานของโปรแกรมของไมโครคอนโทลเลอร์ได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ทฤษฎีเบื้องต้น

1. เกจความเครียด (Strain Gages)

ทรานส์ดิวเซอร์ชนิดนี้อยู่บนพื้นฐานที่ว่าถ้าตัวนำไฟฟ้าอันหนึ่งถูกขึงให้ตึง หรือถูกอัดตัวค่าความต้านทานของมันจะเปลี่ยนแปลง อันเนื่องจากมีการเปลี่ยนใน ความยาว พื้นที่หรือพิกัดความต้านทาน (Resistivity) ของมัน ค่าความต้านทาน R ของตัวนำไฟฟ้าอันหนึ่งมีพื้นที่หน้าตัด A ความยาว L และพิกัดความต้านทานของวัสดุ p คือ

$$R = \frac{Pl}{A} , \quad \frac{l}{\mu A} \qquad (1.0)$$

เกจแฟกเตอร์ F ของตัวนำไฟฟ้าถูกนิยามเหมือนกับ

$$F = \frac{\Delta R/R}{\Delta L/L} = \frac{\Delta R/R}{5} \qquad (1.1)$$

เมื่อ ΔR เป็นการเปลี่ยนแปลงของค่าความต้านทาน R เนื่องจากความเครียด 5 ซึ่ง

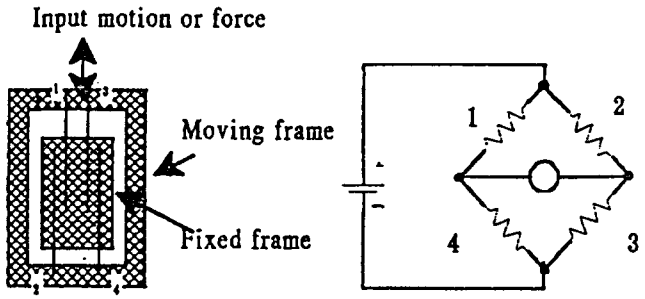
$$5 = \frac{\Delta L}{L}$$

ในทางปฏิบัติ ตัวนำไฟฟ้าที่ใช้มักอยู่ในรูปของเส้นลวดเล็กๆ หรือโลหะแผ่นบางทรานส์ดิวเซอร์เกจความเครียดสามารถแบ่งออกได้เป็นสองชนิด คือ

1. เกจความเครียดชนิดไม่ยึดติด
2. เกจความเครียดชนิดยึดติด

เกจความเครียดชนิดไม่ยึดติด เกจความเครียดชนิดนี้ ลวดค่าความต้านทานเส้นหนึ่งจะถูกขึงให้ตึงระหว่างสองเฟรม เฟรมหนึ่งเป็นเฟรมเคลื่อนที่ ขณะที่อีกเฟรมหนึ่งอยู่กับที่ ดังแสดงในรูปที่ 1 (ก) ขนาดของเส้นลวดตามปกติแล้วมีความยาว 25 มม. และเส้นผ่าศูนย์กลาง 25 μm. เพลทที่บิดงอได้จะ

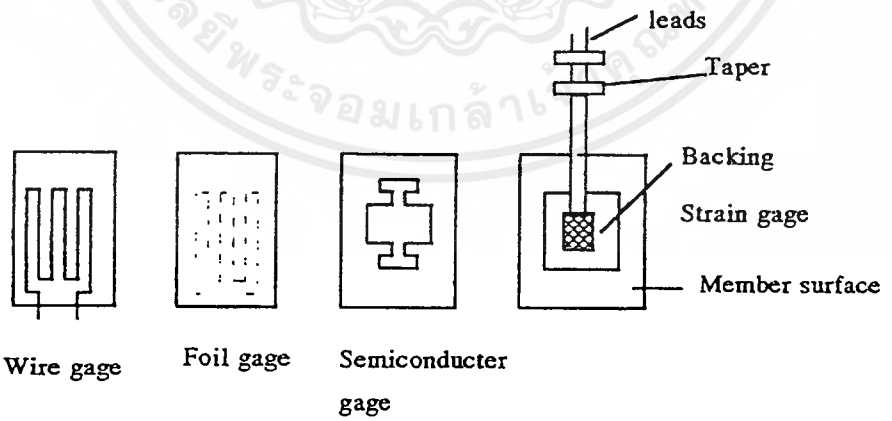
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 1 เกจความเครียดชนิดไม่ยึดติด

หน้าที่เช่นเดียวกับสปริงระหว่างเฟรมทั้งสอง เส้นลวดจะอยู่ภายใต้การป้อนโหลดอันใหม่ ซึ่งมากกว่าโหลดที่มีการอัดตัวที่ตั้งเอาไว้ในตอนแรก การเคลื่อนที่ของอินพิทที่แสดงในรูปที่ 1 จะขึงลวดเส้นที่ 1 และ 3 ให้ตึง ขณะเดียวกันก็จะลดความตึงในลวดเส้นที่ 2 และ 4 การเคลื่อนที่ในทิศทางที่ตรงข้ามกันก็จะให้ผลที่ตรงข้ามกัน เส้นลวดจะถูกนำมาต่อเข้ากับวงจรวีทซโทนบริดจ์ แสดงดังในรูปที่ 1 (ข) โดยทรานส์ดิวเซอร์ชนิดนี้สามารถวัดการเคลื่นที่ที่มีขนาดเล็กมากๆ ถึง 50 ไมโครเมตร และแรงที่มีขนาดเล็กมากๆอีกด้วย ทรานส์ดิวเซอร์ชนิดนี้อาจนำไปใช้เพื่อการวัด แรง ความดัน และความเร่งได้

เกจความเครียดชนิดยึดติด ทรานส์ดิวเซอร์เกจความเครียดชนิดยึดติด ถูกใช้อย่างกว้างขวางเพื่อวัดสิ่งที่แปรเปลี่ยนไปทางกายภาพหลายๆอย่างเช่น ความเครียด แรง แรงบิด ความดัน และการสั่นสะเทือนเป็นต้น เกจเหล่านี้อาจจะทำด้วยวัสดุจำพวกโลหะหรือสารกึ่งตัวนำและอยู่ในรูปของเกจเส้นลวด(เส้น)



รูปที่ 2 ชนิดของเกจความเครียด รูปที่ 3 การติดเกจความเครียด

ผ่าศูนย์กลางประมาณ 25 ไมโครเมตร) หรือโลหะแผ่นบางๆหรือแท่งเล็กๆ (ในกรณีของสารกึ่งตัวนำ) ดังแสดงในรูปที่ 2 เกจเหล่านี้มีกระดาษหรือวัสดุชนิดอื่นห่อหุ้ม เพื่อใช้ยึดติดกับพื้นผิวที่ต้องการ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

วัดความเครียด ดังแสดงในรูปที่ 3 ครั้นยึดติดแล้ว เกจจะได้รับความเครียดเหมือนกับความเครียดที่มีอยู่ในพื้นผิวนั้น เกจเหล่านี้มีความไวสูงมากและเมื่อนำไปใช้ร่วมกับอุปกรณ์อิเล็คทรอนิคส์ ความเครียดที่ต่ำประมาณ 10^{-7} อาจวัดได้

เกจที่ทำด้วยโลหะผสมทองแดง-นิคเกิ้ลมีเกจแฟกเตอร์ 2 ถึง 3 ขณะที่ เซมิคอนดัคเตอร์เกิจมีเกจแฟกเตอร์ 100 ถึง 200 ตารางที่ 1 จะให้ข้อมูลเกี่ยวกับวัสดุห่อหุ้มเกจ ชนิดเส้นลวดและสารยึดติด

ตารางที่ 1

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

กระแสที่ไหลผ่านเกจโดยปกติถูกจำกัดที่ 10 ถึง 30 mA ขึ้นอยู่กับระยะเวลาในการทดสอบเพื่อป้องกันความเสียหายที่จะเกิดขึ้นกับเส้นลวด เกจที่ห่อหุ้มด้วยเบคาไลท์สามารถต้านทานกระแสที่มีค่าค่อนข้างสูงได้

เราจะต้องให้ความระมัดระวังเอาใจใส่ในขณะทำการติดเกจ พื้นผิวที่ต้องการติดเกจจะต้องทำให้สะอาดโดยตลอด ต่อมาจึงใส่สารยึดติดตามคำแนะนำของโรงงานผู้ผลิต แล้วทำการเชื่อมต่อขาของเกจและยึดติดอย่างแน่นหนาเข้ากับส่วนที่ต้องการทดสอบดังแสดงในรูปที่ 1.30 จากนั้นตรวจสอบการเชื่อมต่อเกจและฉนวนเป็นลำดับสุดท้าย

บริดจ์ของเกจความเครียด

เกจความเครียดโดยปกติแล้ว จะถูกต่อเป็นส่วนหนึ่งของวงจรวีทซโตนบริดจ์เพื่อว่าการเปลี่ยนแปลงในค่าความต้านทานของมัน อันเนื่องจากความเครียดจะได้วัดได้หรือให้เอาท์พุทที่สามารถนำไปแสดงผลหรือบันทึกผลได้ มีการจัดวางวงจรบริดจ์อยู่สองรูปแบบ นั่นคือ

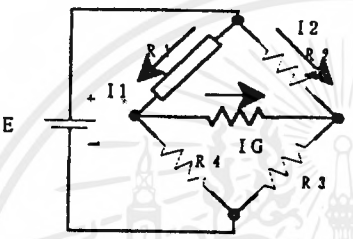
1. สมดุลย์บสริดจ์ (balanced bridge) และ
2. ไม่สมดุลย์บริดจ์ (unbalanced bridge)

ในการจัดวางแบบสมดุลย์บริดจ์ (รูปที่ 4) ค่าความต้านทานของเกจความเครียด R_1 เป็นแขนข้างหนึ่งของวีทซโตนบริดจ์ ขณะที่แขนที่เหลือมีค่าความต้านทาน R_2 R_3 และ R_4 บริดจ์จะถูกกระตุ้นด้วยแหล่งจ่ายดีซีด้วยแรงเคลื่อนไฟฟ้า E และ R_G เป็นค่าความต้านทานของกัลวาโนมิ

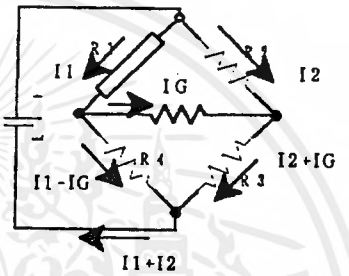
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เตอร์ บริดจ์กล่าวได้ว่าสมดุลย์ เมื่อไม่มีกระแสไหลผ่านกัลวาโนมิเตอร์ เงื่อนไขของการสมดุลย์ซึ่งเป็นที่รู้จักกันดีก็คือ

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



รูปที่ 4 สมดุลย์บริดจ์ของเกจความเครียด



รูปที่ 5 ไม่สมดุลย์บริดจ์ของเกจความเครียด

ถ้า R_1 มีการเปลี่ยนแปลงเนื่องจากความเครียด บริดจ์ซึ่งสมดุลย์ในตอนต้นก็จะไม่สมดุลย์ อันนี้อาจจะ ทำให้สมดุลย์ได้อีกครั้งโดยการเปลี่ยนแปลงค่า R_4 หรือ R_2 การเปลี่ยนแปลงอันนี้สามารถวัดได้และถูกใช้เพื่อแสดงการเปลี่ยนแปลงของ R_1 ซึ่งมีความสัมพันธ์กับความเครียด เทคนิคอันนี้สามารถนำมาใช้เพื่อวัดความเครียดทางสถิติเพียงอย่างเดียว

การจัดวางแบบไม่สมดุลย์บริดจ์ที่แสดงในรูปที่ 5 กระแสที่ไหลผ่านกัลวาโนมิเตอร์หรือแรงเคลื่นไฟฟ้าที่ตกคร่อมกัลวาโนมิเตอร์ถูกใช้ เพื่อแสดงความเครียดในเกจความเครียด วิธีนี้สามารถใช้เพื่อวัดความเครียดทางไดนามิกได้ดีเช่นเดียวกับการวัดความเครียดทางสถิติ เพื่อแสดงถึงการได้มาของความสัมพันธ์ระหว่างกระแสเอาท์พุท ที่ไม่สมดุลย์กับการเปลี่ยนแปลงในค่าความต้านทานของเกจ R_1 สมการที่ (1.23) ถึง(1.25) สามารถเขียนได้ดังนี้โดยใช้กฎของ kirchhoff

$$I_1R_1 + R_4(I_1 - I_G) = E \qquad (1.2)$$

$$I_1R_1 + I_GR_G - I_2R_2 = 0 \qquad (1.3)$$

$$I_GR_G + (I_2 + I_G)R_3 - (R_1 - I_G)R_4 = 0 \qquad (1.4)$$

แก้สมการที่ (1.2) (1.3) และ (1.4) เพื่อหาค่า I_G เราจะได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

$$I_G = \frac{E(R_2R_4-R_1R_3)}{[R_2(R_1+R_4)(R_G+R_3+R_4)+R_1R_3R_4-R_2R_4+R_GR_3(R_1+R_4)]} \quad (1.5)$$

จากสมการที่ (1.5) จะพบว่า $I_G = 0$ เมื่อ $R_2R_4=R_1R_3$ อันนี้สอดคล้องกับสมการที่ (1.2)

จากสมการที่ (1.5) ค่ากระแสที่ไม่สมดุลย์ I_G สามารถคำนวณได้สำหรับการเปลี่ยนแปลงใด ๆ ในค่าความต้านทานของเกจความเครียด R_1 สิ่งที่ได้เมื่อ $R_1= R_2 = R_3= R_4$ และถ้า R_1 เปลี่ยนแปลงเป็น $R_1 + \Delta R_1$ เราจะพบว่าสมการที่ (1.5) คือ

$$I_G = \frac{-E\Delta R_1}{4R_1 (R_1+R_G)} \quad (1.6)$$

$$= \frac{EF_{S1}}{(R_1+ I_G)} \quad (1.7)$$

เมื่อ เป็นความเครียดซึ่งเกิดจาก R_1 และ

$$S_1 = \frac{\Delta R_1}{R_1F}$$

F เป็นเกจแฟคเตอร์ของเกจความเครียด

แรงคลื่นไฟฟ้าเอาท์พุทตกคร่อม R_G (E_0) $= I_GR_G$

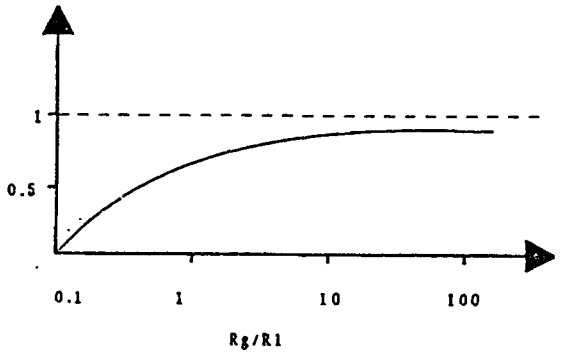
$$= \frac{-E_{s1}R_GF}{4(R_1+R_G)} \quad (1.8)$$

แรงเคลื่อนไฟฟ้าเอาท์พุทที่เปิดวงจร $(E'_0) = \frac{-EF1}{4}$ (1.9)

อันนี้เกิดขึ้นได้โดยการกำหนดให้ $R_G \rightarrow \infty$ ในสมการที่ (1.8)

อัตราส่วนของ E'_0 / E_0 ถูกพล็อทเทียบกับ R_G / R_1ในรูปที่ 1.33 เราจะพบว่า R_G จะต้องมีค่าอย่างน้อยร้อยเท่าของค่าความต้านทานของเกจความเครียด R_1 เพื่อจะให้ได้เอาท์พุท E_0 มากที่สุดเท่าที่จะเป็นได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 6 พล็อทของ E'_0/E_0 เทียบกับ R_G/R_1

เมื่อแขนของวีทซโตนบริดจ์มากกว่าหนึ่งแขน บรรจุด้วยเกจความเครียดและค่าความต้านทานของมันเปลี่ยนแปลงเนื่องจากความเครียด เอาท์พุทที่ได้รับก็คือผลรวมของการเปลี่ยนแปลงเหล่านี้นั่นคือในรูปที่ 4 ถ้า R_2 เปลี่ยนแปลงเป็น $R_2+\Delta R_2$ และในตอนต้นกำหนดให้ $R_1 = R_2 = R_3=R_4$

สมการที่ (1.6) 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 เป็น$R_2+\Delta R_2$ ผลรวมของการเปลี่ยนแปลงคือ

$$I_G = \frac{-E}{4(R_1+R_G)}\left[\frac{\Delta R_1}{R_1} - \frac{\Delta R_2}{R_2}\right] \quad (1.10)$$

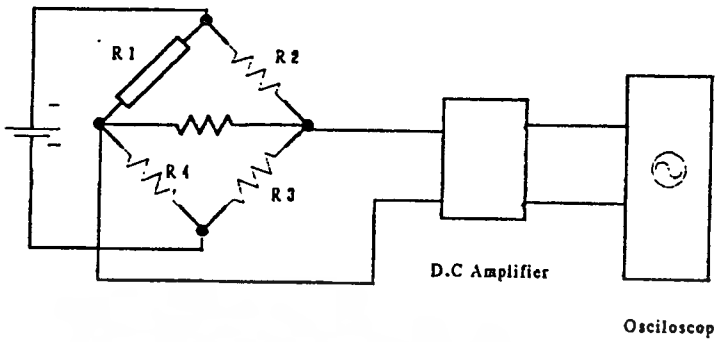
ในทำนองเดียวกัน ถ้าแขนทั้งสี่ของบริดจ์มีเกจความเครียดต่ออยู่ ซึ่งค่าความต้านทานของมันจะเปลี่ยนแปลงเนื่องจากความเครียด เราจะพบว่า

$$I_G = \frac{E}{4(R_2+R_G)}\left[-\frac{\Delta R_1}{R_1} + \frac{\Delta R_2}{R_2} - \frac{\Delta R_3}{R_3} + \frac{\Delta R_4}{R_4}\right] \quad (1.11)$$

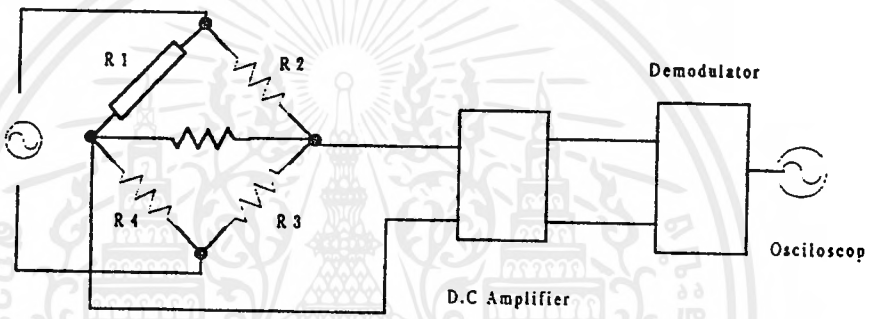
ด้วยเหตุนี้ เราจะพบว่ากระแสเอาท์พุทจะเพิ่มขึ้นถ้าการเปลี่ยนแปลงของค่าความต้านทาน ΔR_2 และ ΔR_4 และมีธรรมชาติที่เหมือนกับ ΔR_3 ดังนั้นแขนที่อยู่ใกล้ชิดกันของบริดจ์จะต้องมีความเครียดในธรรมชาติที่ตรงกันข้ามกัน เพื่อให้เอาท์พุทมีขนาดใหญ่หรือเพิ่มความไวของบริดจ์ อันนี้จะต้องให้ความสนใจขณะจัดวางเกจความเครียดบนพื้นที่ที่ต้องการวัดในลักษณะที่ว่าถ้า R_1 เป็นความตึงเครียด R_2 ควรจะเป็นความอัดเครียดR_3 เป็นความตึงเครียด และ R_4 เป็นความอัดเครียด ดังนั้น มันเป็นไปได้ที่จะให้คำนิยามของคำว่า "แฟคเตอร์สนับสนุนสัญญาณ (signal enhancement factor) ของบริด" เหมือนกับอัตราส่วนของเอาท์พุทสูงสุด เนื่องจากการเปลี่ยนแปลงในเกจความเครียดหลายๆ ตัวต่อเอาท์พุทสูงสุดที่ได้รับจากการใช้เกจความเครียดเพียงตัวเดียวบนพื้นที่ที่ต้องการวัด หรือกล่าวอีกนัยหนึ่ง เกจที่มีมากกว่าหนึ่งตัว (ภายใต้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ความเครียด) ถูกจัดวางไว้อย่างเหมาะสมสามารถทำให้ความไวเพิ่มมากขึ้นหรือทำให้สัญญาณเพิ่มสูงขึ้น



รูปที่ 7 ดีซีบริดจ์



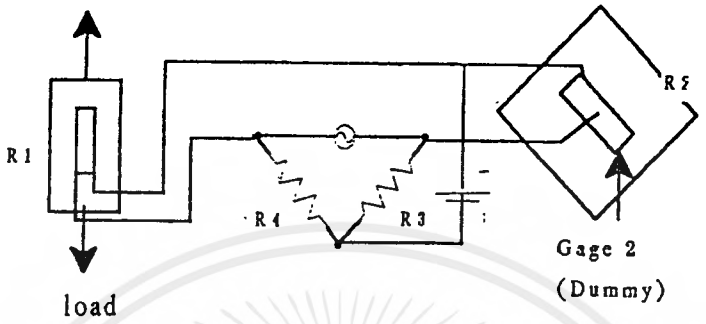
รูปที่ 8 เอซีบริดจ์

การกระตุ้นบริดจ์อาจจะทำด้วยแหล่งจ่ายไฟดีซีหรือเอซี ดังวงจรที่ให้ไว้ในรูปที่ 7 และ 8 ในทั้งสองกรณี ตัวขยายเป็นสิ่งที่ต้องการเพื่อการคัพปลิง ในกรณีของเอซีบริดจ์สัญญาณเป็นแอมปิลจูดมอดูเลทเหมือนแสดงในรูปที่ 8 (ข) ดังนั้นดีมอดูเลเตอร์จึงเป็นสิ่งต้องการเพื่อการรองความถี่พาหะเอาท์พุทที่ได้จะเหมือนกับความเครียดที่กำลังถูกวัด ในจำพวก ของบริดจ์ทั้งสองชนิดเอซีบริดจ์จะไม่มีการเปลี่ยนแปลงที่เอาท์พุท เนื่องจากอุณหภูมิและไม่มีสัญญาณรบกวน ซึ่งตามปกติแล้วมักจะพบในดีซีบริดจ์ อย่างไรก็ตาม เอซีบริดจ์มีราคาแพงและการตอบสนองต่อความถึงสูงของมันถูกลิมิตที่ประมาณหนึ่งในห้าของความถี่พาหะที่ใช้

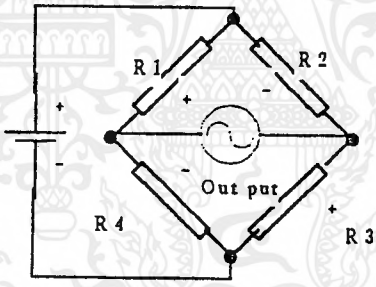
การชดเชยอุณหภูมิ นอกจากความเครียดแล้ว การเปลี่ยนแปลงของอุณหภูมิจะมีผลทำให้เกิดการเปลี่ยนแปลงของค่าความต้านทานในเกจความเครียดด้วย แต่มันไม่สะดวกที่จะใช้การคำนวณเพื่อการแก้ไขอุณหภูมิ การแก้ไขและการชดเชยอุณหภูมิที่ทำขึ้นโดยตัวของมันเองจะสะดวกกว่า อันนี้ทำได้โดย (1) ใช้ดัมมิเกจ (dummy gage) หรือ (2) ใช้เกจมากกว่าหนึ่งตัว ที่มีการจัดวางเกจไว้อย่างเหมาะสมในกรณีที่ (1) แสดงดังรูปที่ 9 เกจค่าความต้านทาน R_2 เท่ากับ R_1 ถูกยึดติดบนแท่งของวัสดุชนิดเดียวกับแท่งวัสดุทดสอบ (test specimen) ซึ่งกำลังได้รับโหลด อย่างไรก็ตาม แท่งของวัสดุที่มี R_2 ไม่ได้รับความเครียด แต่อยู่ภายใต้การ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เปลี่นแปลงของอุณหภูมิอันเดียวกันเหมือนกับแท่งวัสดุทดสอบดังนั้น R_1 ก็คือเกจใช้งานภายใต้ความเครียดและอุณหภูมิมีการเปลี่ยนแปลงไปพร้อม ๆ กัน ขณะที่ R_2 คือดัมมิเกจภายใต้การเปลี่ยนแปลงของอุณหภูมิอันเดียวกันด้วยเหตุที่ R_1 และ R_2 เป็นแขนที่ไกลชิดกันของบริดจ์เอาท์พุทเนื่องจากอุณหภูมิปลี่ยนแปลงจึงเป็นศูนย์ เพราะว่า R_1 และ R_2 มีการเปลี่ยนแปลงเนื่องจากอุณหภูมิเป็นจำนวนเท่ากัน



รูปที่ 9 การชดเชยอุณหภูมิด้วยดัมมิเกจ



รูปที่ 10 การจัดวางเกจความเครียดโดยใช้เกจสี่ตัว

ในกรณีที่ (2) เกจ R_2 ไม่ได้เป็นดัมมิเกจแต่ถูกจัดวางไว้อย่างเหมาะสมบนแท่งวัสดุทดสอบเพื่อว่ามันจะได้มีความเครียดในธรรมชาติที่ตรงกันข้ามกับความเครียดใน R_1 ในกรณีเช่นนี้จะทำได้ระบบมีความไวและมีการชดเชยอุณหภูมิไปพร้อม ๆกัน แท้จริงแล้ว ถ้าแขนทั้งสี่ของบริดจ์เป็นเกจใช้งานเมือนแสดงในรูปที่ 10 โดยแขนที่ไกลชิดกันของบริดจ์มีความเครียดในธรรมชาติที่ตรงกันข้ามกัน สัญญาณเอาท์พุจะมีค่าสูงสุดและมีการชดเชยอุณภูมิด้วย และเป็นเรื่องปกติที่ค่าความต้านทานเริ่มต้นของทุก ๆแขนของบริดจ์จะต้องมีค่าเท่ากัน

การชดเชยอุณหภูมิปีนสิ่งจำเป็นเมื่อต้องการวัดความเครียดทางสถิติ เพราะว่าการเปลี่ยนแปลงของอุณหภูมิจะเป็นผลให้เกิดการเปลี่ยนแปลงที่สัญญาณเอาท์พุท ซึ่งการเปลี่ยนแปลงนี้จะน้อยมากสำหรับการวัดความเครียดทางไดนามิค ดังนั้น การวัดความเครียดทางไดนามิคจึงไม่จำเป็นต้องมีการชดเชยอุณหภูมิ

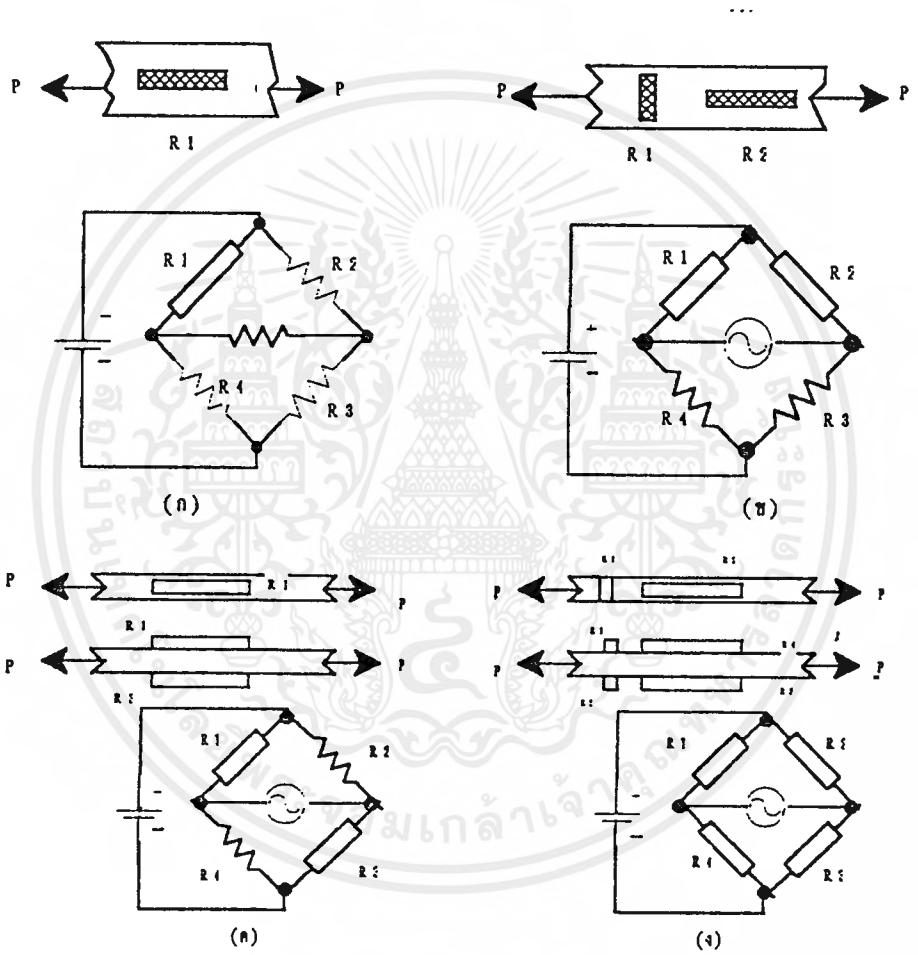
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การจัดวางเกจความเครียด (Strain gane arrangement) มีปัจจัยสองอย่างดังต่อไปนี้จะต้องคำนึงถึง

ขณะตัดสินใจวางเกจความเครียดบนชิ้นส่วนยืดหยุ่นเพื่อวัดการเปลี่ยนแปลงทางกายภาพคือ

1. ความไวสูงและ
2. การชดเชยอุณหภูมิ

การจัดวางที่แสดงในรูปที่ 10 เป็นสิ่งที่สอดคล้องกับการพิจารณาทั้งสองประการข้างต้น



รูปที่ 11 รูปแบบการจัดวางที่เป็นไปได้สำหรับการวัดแรง P

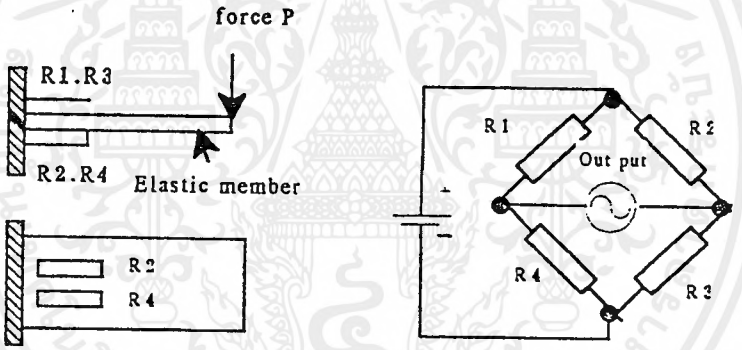
รูปแบบของการจัดวางเกจที่เป็นไปได้ของการวัดแรง P ที่ป้อนไปที่ชิ้นส่วนยืดหยุ่น แสดงดังในรูปที่ 11 (ก) (ข) (ค) และ (ง) โดยเกจความเครียดแสดงด้วยรูปสี่เหลี่ยมผืนผ้า ในรูปที่ 11 (ก) เกจ R_1 ถูกยึดติดอยู่บนชิ้นส่วนยืดหยุ่น เพื่อวัดความเครียดตามแนวเกนในชิ้นส่วนยืดหยุ่น แต่ระบบไม่มีการชดเชยอุณหภูมิและเอาท์พุทจะขึ้นอยู่กับการเปลี่ยนแปลงของ R_1 เพียงอย่างเดียว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ในการจัดวางของรูปที่ 11 (ข) เกจสองตัว R_1 และ R_2 ถูกวางไว้เป็นมุมฉาก หรือเรียกว่า "การจัดวางแบบพอยสัน (Poisson's arraghement)" ดังนั้น R_1 จะเปลี่ยนแปลงเนื่องจากความตึงเครียดตามแนวแกน ขณะที่ R_2 เปลี่ยนเนื่องจากความอัดเครียดตามแนวขวางในชิ้นส่วนยืดหยุ่น โดยความเครียดอันหลังจะเป็น v เท่าของความเครียดอันแรก เมื่อ v หมายถึงอัตราส่วนของพอยสัน (Poisson' ration) แฟคเตอร์สนับสนุนสัญญาณจะเป็น (1+v) เพราะว่าค่าความต้านทานของ R_1 และ R_2 มีความเครียดในธรรมชาติที่ตรงกันข้ามกัน และระบบมีการชดเชยอุณหภูมิด้วยเนื่องจาก

การเปลี่ยนแปลงของอุณหภูมิจะมีผลกระทบต่อ R_1 และ R_2 เท่า ๆกันจึงไม่มีผลต่อเอาท์พุท การจัดวางของรูปที่11 (ก) ซึ่งทั้ง R_1 และ R_3 อยู่ภายใต้ความตึงเครียดตามแนวแกนด้วยจำนวนที่เท่ากันและ R_1 และ R_3 ถูกติดไว้บนแขนของบริดจ์ด้านตรงข้ามกัน จึงให้แฟคเตอร์สนับสนุนสัญญาณเป็น 2 แต่จะไม่มีการชดเชยอุณหภูมิ ดังนั้น การจัดวางแบบนี้จึงต้องการสองดัมมิเกจเพื่อทำให้เกิดความมั่นใจในเรื่องการชดเชยอุณหภูมิ

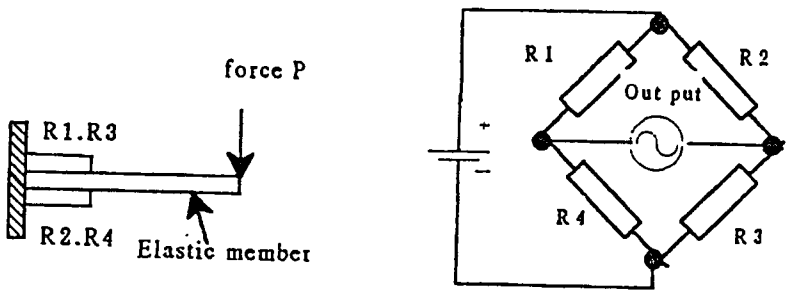
การจัดวางของรูปที่ 11 (ง) ซึ่งใช้เกจสี่ตัวโดย R_2 และ R_4 ถูกจัดวางไว้เป็นมุมฉากกับ R_1 และ R_3 จะให้แฟคเตอร์สนับสนุนสัญญาณเป็น 21+v และมีการชดเชยการเปลี่ยนแปลงของอุณหภูมิ



รูปที่ 12 การจัดวางเกจความเครียดเพื่อการวัดแรง P

อีกตัวอย่างหนึ่งแรง P สามารถวัดได้โดยใช้คานยืดหยุ่น (elastic cantilever) ดังแสดงในรูปที่ 1.39 เกจความเครียดถูกยึดติดอยู่ที่โคนของคาน ตรงที่ความเครียดของการโค้งมากที่สุด R_1 และ R_3 มีความตึงเครียด ขณะที่ R_2 และ R_4 มีความอัดเครียด การจัดวางบริดจ์ของรูปที่ 12 (ข) จะให้แฟคเตอร์สนับสนุนสัญญาณเป็น 4 และมีการชดเชยอุณหภูมิด้วย การจัดวางในลักษณะอื่นแสดงดังในรูปที่ 13 ซึ่ง R_1 และ R_2 ถูกยึดติดตามการจัดวางของพอยสัน และอันนี้ก็ใช้กับ R_1 และ R_2 ด้วยการจัดวางบริดจ์ของรูปที่ 13 (ข) แขนที่ติดกันจะมีความเครียดในธรรมชาติที่ตรงกันข้ามกัน แม้ว่าจะมี ขนาดไม่เท่ากันก็ตาม การจัดวางของรูปที่ 13 จะให้แฟคเตอร์สนับสนุนสัญญาณเป็น 2(1+ v) เมื่อ v เป็นอัตราส่วนของพอยสันและอุณหภูมิก็จะถูกชดเชยด้วย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่13 การจัดวางในอีกลักษณะหนึ่ง

<u>การสมดุลย์ของบริดจ์(Balancing of Bridges)</u>

บริดจ์ต้องทำให้สมบูรณในตอนเริ่มต้น อย่างไรก็ตามในทางปฏิบัติ ค่า ความต้านทานที่แขนทั้งสี่อาจจะไม่เท่ากัน วิธีใดวิธีหนึ่งดังต่อไปนี้จะถูกนำมาใช้เพื่อการสมดูลย์บริดย์

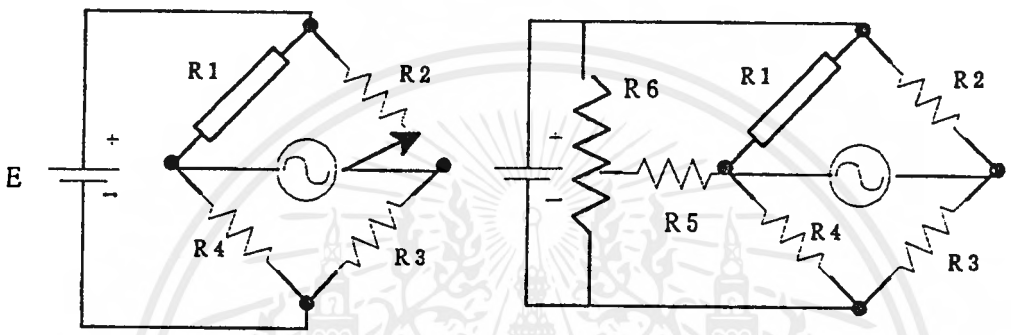
1.วิธีการสมดุลแบบอนุกรม (Series balancing methold) โดยใช้ Apex resistor ดังแสดงในรูป* เนื่องจากการเคลื่อนที่ที่จุดสัมผัสของ Apex resistance ถ้า R2 เพิ่มขึ้นเป็น r และ R3 ลดลงเป็น r ด้วย ในทางตรงข้าม ถ้า R3 เพิ่มขึ้นเป็น r R2 ก็จะลดลงเป็น r ด้วย

2.วิธีการสมดุลแบบขนาน(Parallet Balancing Method) บริดย์สามารถทำให้สมดุลได้โดยการเคลื่อนที่จุดสัมผัส P ดังแสดงในรูปที่

<u>การสอบเทียบ</u>

การสอบเทียบกลายเป็นสิ่งจำเป็นสำหรับการจัดวางวงจรบริดย์แบบไม่สมดุลย์ ซึ่งเอาท์พุทของบริดย์ได้จากความเครียดของเกจความเครียด การสอบเทียบอาจทำได้ทั้งทางไฟฟ้าและทางกล วิธีการทางไฟฟ้าโดยปกติจะสร้างขึ้นในวงจรของบริดย์ความเครียดสามารถพบเห็นได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 14 การใช้ Apex resister เพื่อการสมดุลย์และการสอบเทียบ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

<u>วิธีการทางไฟฟ้า</u>

วิธีแรกการเปลี่ยนแปลงของค่าความต้านทานที่แขนอางเลียนแบบโดยการใช้ Apex resistance ของรูปที่ 14 มันเป็นไปได้ที่จะสอบเทียบ Apex resistance ในเทอมของความเครียด(ς) ในเกจตัวเดียว (สมมุติว่าเป็น R1) ความสามารถอันนี้แสดงดังนี้

จากรูปที่ 14 ถ้าบริดย์ถูกทำให้สมดุลย์ในตอนแรกนั้นคือ

$$\frac{R1}{R4} = \frac{R2}{R3} = 1 \qquad (1.12)$$

จากนั้น Apex resistance เปลี่ยนแปลงไป นั้นคือ R2 เพิ่มขึ้นเป็น R2+r และ R3 ลดลงเป็น R3+r บริดย์เปลี่ยนป็นไม่สมดุลย์ แต่ค่าของ $\Delta R1/R1$ สอดคล้องกับการเปลี่ยนแปลงข้างต้น

$$\frac{R1 + \Delta R1}{R4} = \frac{R2 + r}{R3 - r} \qquad (1.13)$$

กำหนดให้ x=r/R3 เราจะพบว่าสมการที่ (1.33) และ (1.34)จะได้

$$\frac{\Delta R1}{R1} = \frac{2x}{1 - x} \qquad (1.14)$$

ดังนั้น Apex resistance สามารถสอบเทียบได้ในเทอมของ$\Delta R1/R1$หรือความเครียด ξ ถ้ารู้ค่าเกจเฟคเตอร์

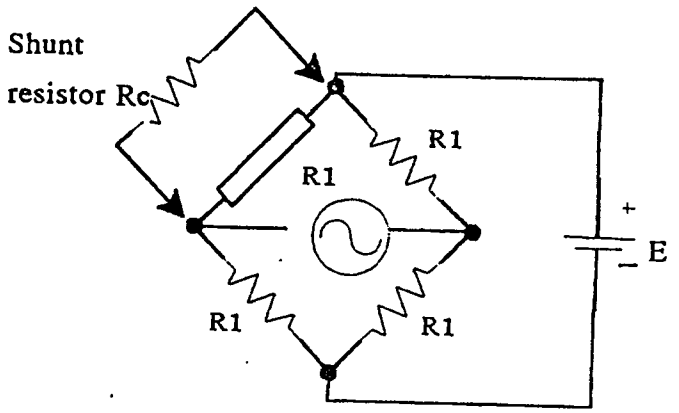
การสอบเทียบทางวงจรทางไฟฟ้าวิธีที่สองโดยการใช้ค่าความต้านทาน (Shunt resister) ดังแสดงในรูปที่ 15 ถ้าตัวต้านทานขนาน Rc ถูกต่อขนานกับเกจความเครียด R1 การเปลี่ยนแปลง $\Delta R1$ คือ

$$\Delta R1 = R1 - R3 / / Rc \qquad (1.15)$$

หรือเราอาจเขียนสมการใหม่เป็น

$$\frac{\Delta R1}{R1} = \frac{\Delta R1}{R1 + Rc} \approx \frac{R1}{Rc} \qquad (1.16)$$

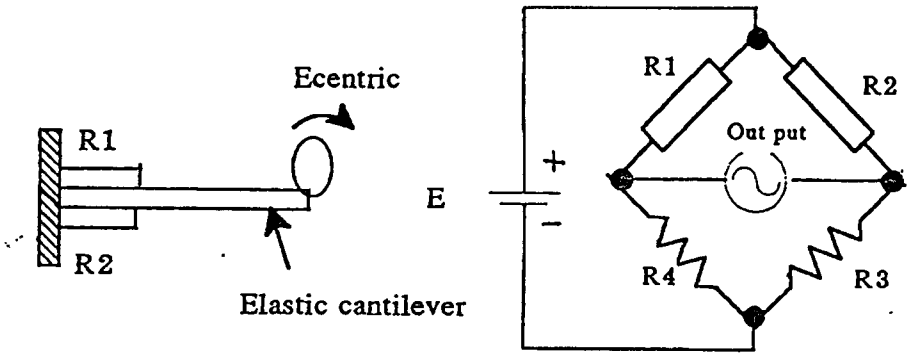
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



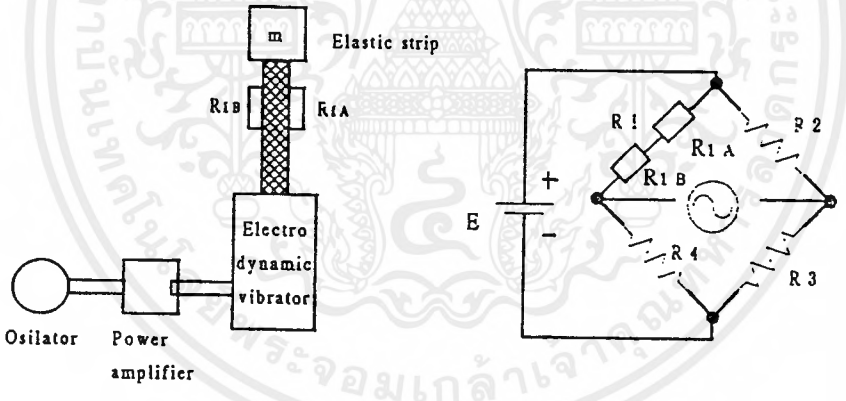
รูปที่ 15การสอบเทียบโดยการขนานตัวต้านทาน

ซึ่งกรณีโดยทั่วๆไปสำหรับ $\Delta R1/R1$,มีค่าน้อยมากๆ ดังนั้น การเปลี่ยนแปลงที่เอาท์พุทซึ่งเป็นผลมาจากการต่อ Rc ขนานกับค่าความต้านทาน R1 สามารถวัดได้หรือมีความสัมพันธ์กับ $\Delta R1/R1$ หรือความเครียด ในการสอบเทียบทางกล สำหรับการสอบทียบทาง Statiic dead weightsสามารถนำมาใช้ได้โดยตรงและความเครียดสามารคำนวณได้และมีความสัมพันธ์กับเอาท์พุท สำหรับการสอบเทียบทางไดนามิกมีสองวิธี ดังแสดงในรูปที่ 17 และวิธีแรกจะใช้การหมุนของลูกเบี้ยว (rotating eccentric) และวิธีหลังจะใช้ตัวสั่นสะเทือน (vibration)ในรูปที่ 17 คานยึดหยุ่นถูกใช เหมือนกับอุปกรณ์ยึดหยุ่น ความเครียดที่แท้จริงสามารถคำนวณได้และเฟคเตอร์ของการสอบเทียบ ซึ่งมีความสัมพันธ์กับเอาท์พุทและความเครียดสามารถคำนวณได้ในรูปที่ 17 ความถี่แอปมปริจูดของตัวสั่นสะเทือนสามารถคำนวณได้ เพราะว่าแรงอย่าง sinusoidal โดยมวล m ถูกป้อนไปที่แท่งยึดหยุน (elastic strip)เป็น $m\omega^2 x0$ เมื่อ ω เป็น circular frequency ของการสั่น และ x0 เป็นแอปมปริจูดของมวลที่เคลื่อนที่ซึ่งจำเป็นที่จะต้องถูกวัดโดยทรานสดิวเซอร์เคลื่อนที่ใดๆ ที่อธิบายไปแล้ว ตรงข้ามกกับอันนี้ เอาท์พุทของบริดจ์สามารถวัดได้ ในรูปที่ 17 ทั้ง R1A และ R1B ถูกจัดวางเป็นแขนข้างหนึ่งของวีทซโตนบริดจ์ การกระทำเช่นนี้เพื่อการขจัดผลของการโค้งงอใดๆของแท่งยึดหยุน การโค้งงอในแท่งยึดหยุ่น การโค้งงอจะมีการเปลี่ยนแปลงที่เหมือนกันแต่ในธรรมชาติที่ตรงข้ามกันของ R1A และ R1B ทำให้ค่าความต้านทานรวมของแขนไม่เปลี่ยนแปลง ดังนั้นเกจจะตอบสนองความเครียด อันเนื่องจากแรงทางไดนามิกที่ได้จากการเคลื่อนที่ของมวล m เพียงอย่างเดียว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 16 การสอบเทียบเกจความเครียดทางไดนามิกโดยใช้ลูกเบี้ยว



รูปที่ 17 การสอบเทียบเกจความเครียดทางไดนามิกโดยใช้ตัวสั่นสะเทือน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2. การติดต่อกับไมโครคอนโทลเลอร์

2.1 การสื่อสารอนุกรมทางพอร์ต RS - 232C

การสื่อสารข้อมูลอนุกรมนิยมนำมาใช้กับการโอนถ่ายข้อมูล ระหว่างตัวควบคุมด้วยไมโครคอนโทรลเลอร์หรืออุปกรณ์คอมพิวเตอร์ต่าง ๆ เพราะใช้สัญญาณเพียง 2 หรือ 3 เส้น สำหรับข้อมูลสื่อสารที่ส่งหรือรับเท่านั้น แต่เนื่องจากการประมวลข้อมูลในคอมพิวเตอร์เป็นไปลักษณะขนานทำให้ต้องมีการแปลงผัน (Conversion) ข้อมูลให้เหมาะสมก่อนการส่งข้อมูล รวมทั้งภายหลังการรับข้อมูลแบบอนุกรม

การส่งข้อมูลของ MCS 51 นั้นทำได้ง่ายเพียงแต่การเขียนโปรแกรมการโอนถ่ายข้อมูล (แบบขนาน)ให้กับรีจิสเตอร์ที่จัดการเกี่ยวกับข้อมูลอนุกรมเท่านั้น หรือการรับข้อมูลก็อ่านจากรีจิสเตอร์นี้เช่นกัน โดยไม่ต้องเขียนโปรแกรมเพื่อแปลงผันข้อมูลระหว่างอนุกรมและขนาน รวมทั้งการเลื่อนบิต (Shift Bite) จากขาสัญญาณแต่อย่างใด จึงนับว่าสะดวกกับผู้ใช้งานมาก

2.2 พอตร์สื่อสารข้อมูลอนุกรม RS -232c

รูปแบบของข้อมูลอนุกรมที่ใชกันอย่างแพร่หลายเป็นลักษณะ การสื่อสารแบบอะซิงโครนัส (Asynchronous communication) และเป็นไปตามมาตราฐาน RS-232C รูปแบบของข้อมูลภายในเส้นสัญญาณจะมีบิตข้อมูลเพิ่มขึ้นจากบิตสื่อสารจริง เช่นบิตเริ่มต้น (Start bit) และบิตสิ้นสุด(Stop bit) เพื่อนำมาใช้ในการบอกจังหวะการเริ่มต้นและสิ้นสุดแต่ละไบต์

ตามมาตราฐาน RS-232C ระดับสัญญาณข้อมูลของวงจร (TTL voltage levels) ซึ่มีค่าอยู่ในช่วง 0 ถึง 5 โวลต์ จะต้องนำมาปรับให้เป็นระดับแบบ RS-232 (RS-232 voltage levels) ซึ่งมีค่าสูงขึ้นในช่วงจาก +15 ถึง -15 โวลต์ ดังรูป แสดงการเชื่อมต่อ RS-232C ของบอร์ด JAZZ -31 โดยมีไอซี MAX 232 เป็นตัวปรับระดับสัญญาณ(RS-232C line diver and receiver)

สำหรับความเร็วของการส่งข้อมูลอนุกรมมักจะเรียกกันว่า “อัตราบอต” (Baud Rate) ซึ่งสามารถอธิบายโดยง่ายสำหรับรูปแบบข้อมูลอนุกรมแบบอะซิงโครนัสว่า เป็นจำนวนของบิตข้อมูลภายในหนึ่งช่วงเวลา อัตราบอดที่ใช้งานกันโดยทั่วไป เช่น 1200,2400,4800, และ 9600 เป็นต้น

2.3 การใช้งานของพอร์ตอนุกรม 8031

รีจิสเตอร์ที่เกี่ยวข้องเกี่ยวกับการสื่อสารอนุกรมของ 8031 ประกอบด้วย รีจิสเตอร์ SBUF สำหรับเก็บข้อมูลสื่อสาร รีจีสเตอร์ SCON ควบคุมการสื่อสาร รีจีสเตอร์ PCON ควบคุมการสื่อสารและขาสัญญาณ R*D (P3.0) และ T*D (P3.1) เป็นขาสัญญาณที่เกี่ยวกับอุปกรณ์หรือวงจรภายนอก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้
038510

เนื่องจากการส่งข้อมูลหนึ่งไบต์ออกไปทางพอร์ตอนุกรมในคราวหนึ่งๆ นั้นจะใช้เวลามาก ดังนั้นเพื่อไม่ให้การเสียเวลาในการประมวลผลไป 8031 จึงนำแฟล็ก T1 และ R1 ซึ่งเป็นตำแหน่งบิตอยู่ภายในรีจีสเตอร์ SCON เพื่อแจ้งกับผู้ใช้งานทราบ โดยเมื่อใดก็ตามที่มีการส่งข้อมูลอนุกรมเสร็จสิ้น แฟร์ก T1 จะมีค่าเป็นหนึ่ง หรือเมื่อได้รับข้อมูลอนุกรมเข้ามา แฟร์ก R1 จะมีค่าเป็น 1 เช่นกัน

3. การควบคุม ADC

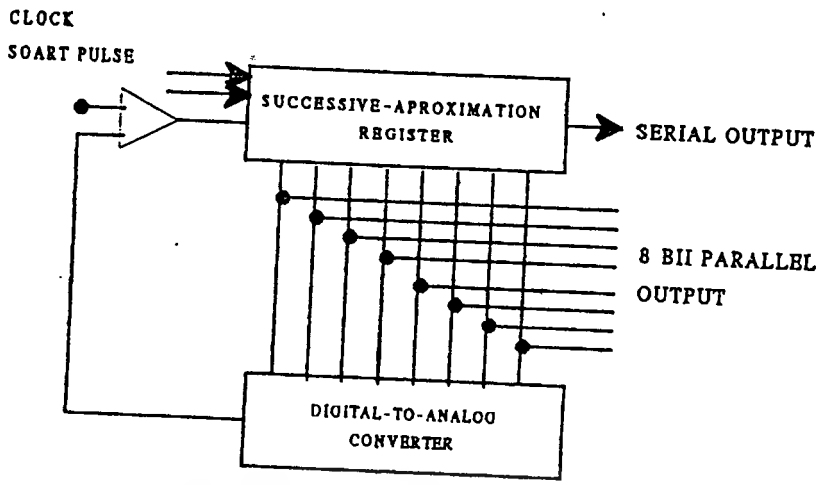
3.1การควบคุม Analog-to-digital Convertion

งานควบคุมของไมโครคอนโทรเลอร์หลายลักษณะ จะไม่ใช่เพียงการควบคุมสถานะของการเปิด-ปิด วงจรหรืออุปกรณ์เท่านั้น ตัวอย่างเช่น การวัดหรืออ่านยค่าอนาลอกเป็นสัญญาณดิจิตอล เป็นต้น งานลักษณะเช่นนี้ จำเป็นจะต้องนำวงจรในการแปลงสัญญาณอนาลอกเป็นสัญญาณดิจิตอล (Analog-to-digital Converter หรือ ADC)มาใช้งาน ซึ่งมีวิธีการแปลงสัญญาณของวงจร ADC หลายลักษณะด้วยกัน แต่การเสนอนี้จะนำมากล่าวเฉพาะวงจร ADC ที่ใช้หลักการ Successive-Aproximation ซึ่งนิยมนำมาใช้ในการแปลงข้อมูลที่ต้องการความเร็วระดับปานกลาง-ถึงความเร็วสูงเท่านั้น

3.2 วงจร ADC แบบ Successive-Aproximation

พื้นฐานทางเทคนิคการทำงานของวงจรชนิดนี้ เป็นการพิจราณาระดับแรงดันไฟฟ้าของสัญญาณอินพุทที่เข้ามา และประมาณค่า (Aproximation) ด้วยรหัสเลขฐานสองค่าหนึ่งแล้วทำการติดตามเพื่อทำการทวนค่าที่ทำการประมวลค่าในแต่ละบิต ๆ ของระหัสเลขบานสองนั้นจนกระทั้งได้ค่าประมาณที่ดีที่สุด โดยในแต่ละขั้นตอนของการประมวลผล จะเก็ค่าของเลขระหัสเลขฐานสองขณะนั้นไว้ตลอดเวลา ภายในหน่วยความจำเฉพาะ หรือที่เรียกว่า Successive-Aproximatio register (SAR)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



ในรูปที่ 3.1แสดงรูปแบบการทำงานพื้นฐานของวงจร ADC ขนาด 3 บิต

กระบวนการแปลงจะเริ่มจากการกำหนดค่าเริ่มต้น 1 ให้กับค่าบิตนัยสำคัญสูงสุด (MSB)ของรีจิสเตอร์ ซึ่งเท่ากับเป็นการเริ่มต้นในการประมาณค่า ด้วยค่าเพียงรึงหนึ่งของระดับสัญญาณอินพุทแบบอนาลอกสูงสุดที่สามารถรับเข้ามาได้ ค่านี้จะนำไปเข้าวงจร DAC ภายในเพื่อเปรียบเทียบกับระดับสัญญาณอินพุทอีกครั้งหนึ่ง ผลลัพธ์จากการเปรียบเทียบจะนำไปใช้ในการกำหนดเลือกค่าให้กับบิต MSB ของรีจิสเตอร์ SAR โดยหากค่าที่ประมาณมีค่าสูงเกินไปจะกำหนดเป็นค่า 0 แต่ในกรณีตรงข้ามเป็นค่า 1 ระยะเวลาถัดมาเมื่อมีสัญญาณนาฬิกาเข้ามาหนึ่งครั้ง ตัวควบคุมภายในจะเริ่มกำหนดค่า 1 ให้กับบิต MSB หลักถัดไป ซึ่งก็จะเปรียบเทียบกับสัญญาณอินพุทอีกครั้ง เพื่อให้ทราบว่าจะกำหนดค่าของบิตนั้นให้เป็น 0 หรือ 1 กระบวนการทำงานทั้งหมดนี้จะวนรอบซ้ำไปเรื่อย ๆจนกระทั่งสิ้นสุดบิตนัยสำคัญต่ำสุด (LSB) ซึ่งจะได้ค่าของการประมาณที่ดีที่สุดเก็บอยู่ภายในรีจิสเตอร์ SAR นี้ และพร้อมส่งออกไปให้กับระบบภายนอก

3.3 การควบคุมไอซี MAX 192 เบื้องต้น

การเสนอในที่นี้ได้ใช้ไอซี MAX 192 ทำหน้าที่เป็น ADC แบบ Suiccessive aproximation จำนวน 8 ช่วงสัญญาณ โดยให้ค่าของเอาท์พุท ที่เป็นข้อมูลดิจิตอลจำนวน 10 บิต ในรูปแบบของข้อมูลแบบอนุกรม ดังนั้นจึงต้องมีการใช้สัญญาณนาฬิกาเป็นตัวกำหนดจังหวะในการรับส่งข้อมูล

เมื่อเริ่มต้นการใช้งาน MAX 192 จะต้องกำหนดค่าให้กับบิตเริ่มต้นให้กับบิตต่างๆภายในรีจิสเตอร์ควบคุม (ดังตาราง) เพื่อเป็นการกำหนดรูปแบบของวงจร ADC เป็นลำดับแรก ซึ่งในงานนี้จะใช้งานเฉพาะช่องสัญญาณ 0 โดยใช้สัญญาณนาฬิกาจากภายนอกไอซี

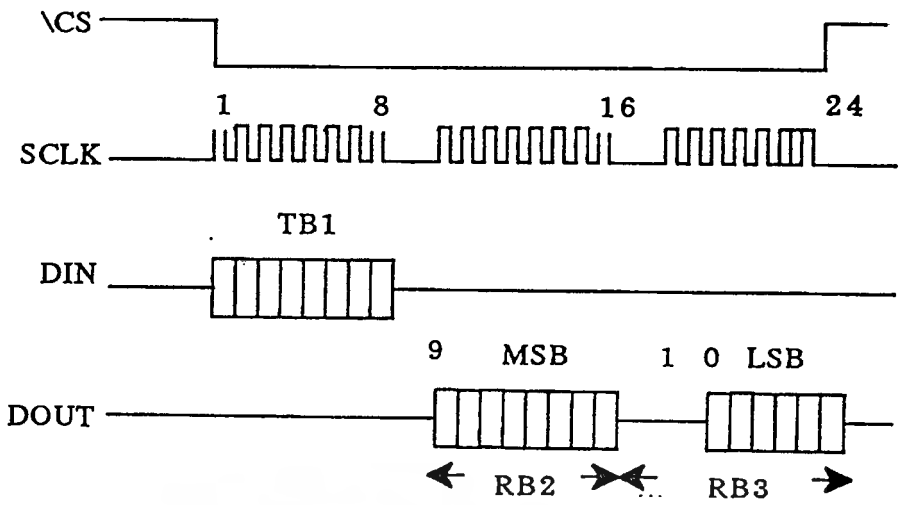
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จำนวน 24 พัลส์ จะได้ค่าเอาท์พุตของการแปลงสัญญาณอนาลอกได้ครบถ้วน (24 bit extemal clock mold conversion) ตามรูปแบบของแผนภาพการติดต่อในรูป ที่ 3.2

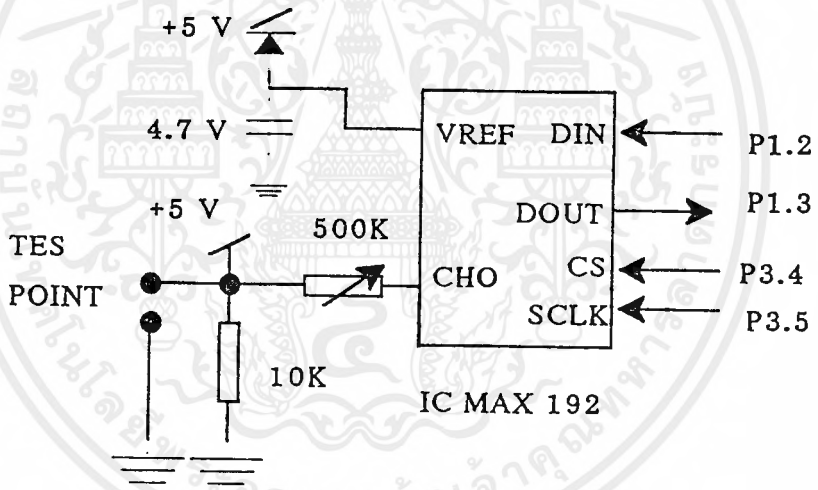
ตำแหน่งบิต	ชื่อ	การทำงาน
7	START	มีค่าเป็นหนึ่งแสดงว่าเป็นไบต์ควบคุม
6 5 4	SEL 2 SEL 1 SEL 0	หมายเลขช่องสัญญาณที่จะใช้ในการแปลงข้อมูล ซึ่งมีทั้งหมด 8 ช่อง (เริ่มจากช่องสัญญาณที่ 0 ถึง 7)
3	UNI/BIP\	เลือกระบุขอบเขตของสัญญาณอนาลอกอินพุตว่า อยู่ในช่วงจาก 0-VREF โวลต์ (Unipolar)หรือเป็นสัญญาณบวกลบ อยู่ในช่วงจาก -Vref/2 ถึง +Vref/2
2	SGL/DIF\	เลือกรูปแบบของสัญญาณอนาลอกว่าเป็นสัญญาณเส้นเดียว (Single ended)เทียบกับระดับกราวด์ในอีกเส้นหรือสัญญาณแบบดิฟเฟอเรนเชียล(Differential)ซึ่งเป็นสัญญาณความแตกต่างในเส้นสัญญาณทั้งสอง
1 0	PDI PD0	เลือกการทำงานโดยใช้ที่มาของสัญญาณนาฬิกาจากภายนอกหรือในตัวไอซี และการทำงานในโหมด POWER DOWN

ตารางที่ 3.1 รูปแบบของไบต์ควบคุมที่จะต้องส่งให้กับรีจิสเตอร์ควบคุมภายในไอซี MAX 192

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.2 แผนภาพของสัญญาณติดต่อกับไอซี MAX 192 ในโหมด 24 บิต External clock Mode Conversion จากรูปนี้ TB1 เป็นไบต์ข้อมูลควบคุม และไบต์ RB2 และ RB3 เป็นไบต์ที่ได้จากการแปลงสัญญาณอนาลอก



รูปที่ 3.3 แผนภาพอย่างง่ายของวงจร ADC โดยไอซี MAX 192

ค่าระดับอ้างอิง (VREF) ของไอซี Max 192 จากแผนภาพวงจรข้างต้น จะมีค่าประมาณ 4.096 โวลต์ ทำให้ค่าของระดับสัญญาณที่เข้ามายังขาของช่องสัญญาณใด ๆ (เช่น ช่องสัญญาณ CH0) อยู่ในช่วงระหว่างค่าต่ำสุดที่ระดับ 0 โวลต์ และค่าสูงสุดที่ระดับ 4.096 โวลต์ ซึ่งจะทำให้ได้ค่าดิจิตอลผลลัพย์ (เลขฐานสอง)จากการแปลงของไอซี MAX 192 (จำนวน 10 บิต)เป็นค่า 00000 00000 และ 11111 11111 ตามลำดับ

จากการทำงานขั้นต้นนั้นจึงอาจจะกล่างในอีกในลักษณะหนึ่งได้ว่าระดับสัญญาณในช่วง 0-4.096 โวลต์ จะสามารถแปลงผันออกมาเป็นเลขฐานสองได้ถึง 2^{10} (หรือเท่ากับ 1024

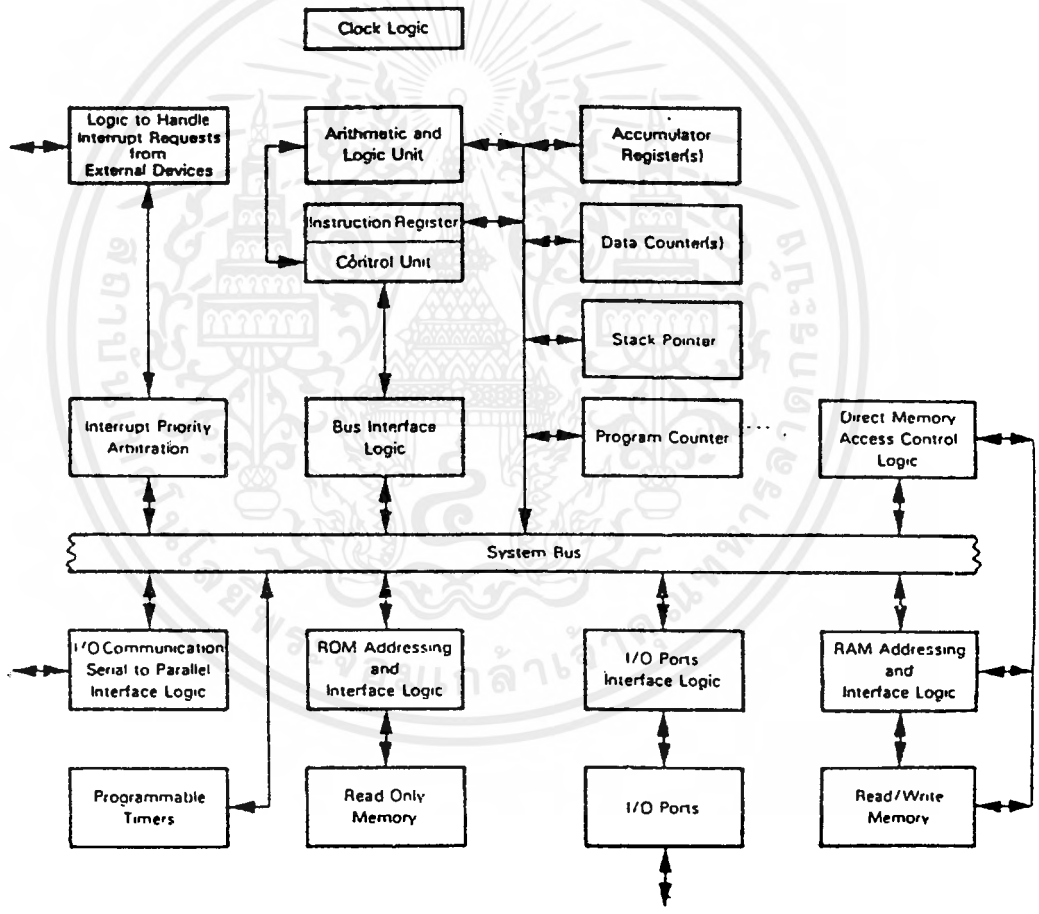
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ค่า) และโวลต์เตจในแต่ละระดับขั้นที่จะมีผลทำให้ระดับการเปลี่ยนแปลงบิตเกิดขึ้นได้(1LSB)จะมีค่าเป็น 4.096/1024 หรือจะเท่ากับ 4.00 มิลลิโวลต์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4. Z80 ไมโครโปรเซสเซอร์

ในปัจจุบันนี้ เราสามารถนำวงจรอิเลคทรอนิกส์ที่ยุ่งยากซับซ้อนมาบรรจุลงบนแผ่นวงจรเดียวที่มีขนาดเล็กมากซึ่งเรียกว่า LSI (LAST SCALE INTEGRATED CTRCUIT) และบรรจุอยู่ในตัวถัง ซึ่งต่อขาออกมาเพื่อใช้งานติดต่อกับวงจรภายนอก สิ่งนี้เรียกว่า "ชิพ (CHIP)" องค์ประกอบย่อยๆ ในไมโครคอมพิวเตอร์ทั้งหมด จะประกอบด้วยชิพเหล่านี้ เช่น หน่วยความจำประเภท ROM (READ ONLY MEMORY) อุปกรณ์สนับสนุนต่างๆ และสิ่งที่เป็นหัวใจของระบบไมโครคอมพิวเตอร์ คือ หน่วยประมวลผลกลาง หรือ CPU (CENTRAL PROCESSING UNIT) ซึ่งภายในประกอบด้วยส่วนต่างๆ ดังบล็อกไดอะแกรมดังรูป



รูปที่ 4.1 แสดงบล็อกไดอะแกรมของ Z80

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ซึ่งแต่ละบล็อกมีลักษณะการทำงานดังต่อไปนี้

1.ARITHMETICS LOGIC UNIT (ALU) เป็นหน่วยที่ทำหน้าที่ในการคำนวณฟังก์ชันพื้นฐานทางคณิตศาสตร์ และการกระทำฟังก์ชันทางลอจิก เช่น AND , OR , ALU จะสามารถทำหน้าที่ได้อย่างมีประสิทธิภาพมากเพียงใดนั้นขึ้นอยู่กับการออกแบบวงจรภายในของ ALU

2.CONTROL BUS เป็นหน่วยที่ทำหน้าที่ในการส่งสัญญาณไปควบคุมอุปกรณ์ต่างๆ ที่ต่อเชื่อมกันได้อย่างถูกต้อง

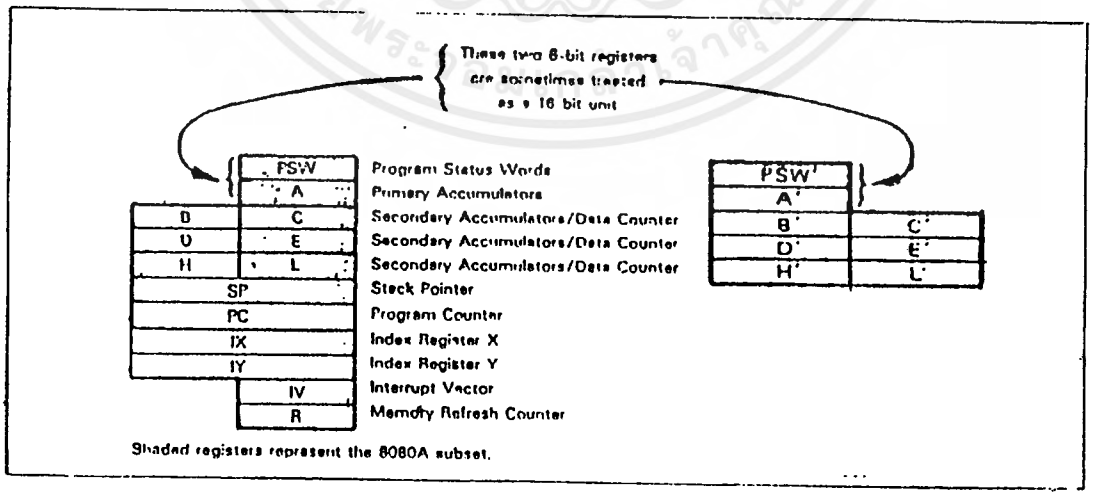
3.DATA BUS เป็นบัสสองทิศทาง (BI -DIRECTIONAL) ที่ใช้ในการส่งผ่านข้อมูลระหว่าง CPU กับอุปกรณ์อื่นๆ ภายในระบบ จำนวนเส้นของบัสข้อมูล (DATA BUS) จะขึ้นอยู่กับชนิดของ CPU เช่นในกรณีของ Z80 CPU จะส่งผ่านข้อมูลทีละ 8 บิท ดังนั้นจะมีเส้นของบัสข้อมูล 8 เส้น

4.CONTROL BUS หรือ บัสควบคุม เป็นบัสทางเดียว (UNI DIRECTIONAL BUS) ที่ใช้ในการส่งผ่านสัญญาณควบคุมให้กับอุปกรณ์ต่างๆ ในระบบ

5.ADDRESS BUS เป็นบัสทางเดียว ใช้ส่งผ่านค่าแอดเดรสจาก CPU ออกไปยังหน่วยความจำเพื่อระบุตำแหน่งที่ต้องการรับส่งข้อมูล หรือใช้ระบุตำแหน่งของพอร์ท I/O (INPUT/OUTPUT PORT) ที่ CPU ต้องการติดต่อด้วย

รีจิสเตอร์ต่างๆ ใน Z80 CPU

Z80 CPU จะประกอบไปด้วยรีจิสเตอร์ถึง 22 ตัวดังรูป รีจิสเตอร์เหล่านี้จะแบ่งได้เป็น 2 กลุ่มคือ รีจิสเตอร์ที่ทำหน้าที่ทั่วๆ ไป และ รีจิสเตอร์ที่ทำหน้าที่เฉพาะงาน



รูปที่ 4.2 แสดงรีจิสเตอร์ต่างๆ ภายใน Z80

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1.รีจิสเตอร์ที่ทำหน้าที่ทั่วไป แบ่งเป็นรีจิสเตอร์หลัก ได้แก่ A,B,C,D,E,H และ L มีความจุขนาด 8 บิท รีจิสเตอร์เหล่านี้ใช้เก็บข้อมูลชั่วคราว นอกจากนี้ยังสามารถรับข้อมูล จากหน่วยความจำหรืออาจจะทำการย้ายข้อมูลไปเก็บไว้ในหน่วยความจำก็ได้ และ รีจิสเตอร์สำรองได้แก่ A',B',C',D',E',H', และ L' ซึ่งเป็นรีจิสเตอร์ที่ทำหน้าที่เก็บข้อมูลที่มาจากรีจิสเตอร์หลักในกรณีที่ต้องใช้รีจิสเตอร์หลักในการทำงานอย่างอื่นก่อน ดังนั้นรีจิสเตอร์กลุ่มนี้จึงไม่สามารถกระทำขบวนการทางคณิตศาสตร์และลอจิกได้

รีจิสเตอร์ A เรียกว่า แอคคิวมูเลเตอร์ (ACCUMULATOR) ทำหน้าที่เก็บข้อมูลชั่วคราวที่ได้จากการทำขบวนการทางคณิตศาสตร์ เช่น บวกหรือลบข้อมูล 2 จำนวน ผลลัพท์ที่ได้จะเก็บไว้ในรีจิสเตอร์ A นี้ นอกจากนี้ ในการปฏิบัติตามคำสั่งที่ใช้กับข้อมูลขนาด 16 บิท Z80 จะนำเอารีจิสเตอร์แฟลก "F" (FLAG REGISTER) มาใช้ร่วมกับรีจิสเตอร์ A เรียกว่า คู่รีจิสเตอร์ AF ซึ่งมีขนาด 16 บิท นอกจากนี้ยังมีคู่รีจิสเตอร์ 16 บิท อื่นๆ อีก คือ BC, DE และ HL

2.รีจิสเตอร์ที่ใช้งานเฉพาะอย่าง ได้แก่ รีจิสเตอร์ I, R, IX, IY, SP และ PC ซึ่งทำหน้าที่ต่างๆดังนี้

รีจิสเตอร์ I (INTERRUPT PAGE ADDRESS REGISTER) เมื่อมีการอินเตอร์รัพท์เกิดขึ้นจำเป็นต้องบอกตำแหน่งของหน่วยความจำที่เก็บโปรแกรมตอบสนองการอินเตอร์รัพท์ รีจิสเตอร์ I จะทำหน้าที่เก็บค่า 8 บิทบนของตำแหน่งข้อมูลในหน่วยความจำ ส่วนค่า 8 บิทล่างจะป้อนจากภายนอกให้แก่ CPU ค่าทั้งสองจะประกอบกันเป็นค่าแอดเดรสที่ระบุตำแหน่งของโปรแกรมการตอบสนองการอินเตอร์รัพท์

รีจิสเตอร์ R (MEMORY REFRESH REGISTER) เป็นรีจิสเตอร์ขนาด 7 บิท ที่ถูกใช้ในการรีเฟรช (REFRESH) DYNAMIC RAM และค่ารีจิสเตอร์ R จะเพิ่มขึ้นเองโดยอัตโนมัติในทุกๆ ครั้งที่มีการเฟทช์คำสั่งจากหน่วยความจำ

รีจิสเตอร์ IX และ IY; (INDEX REGISTER) เป็นรีจิสเตอร์ที่มีขนาด 16 บิท มีประโยชน์ใช้บ่งชี้ตำแหน่งในหน่วยความจำแบบ INDEX ADDESSING MODE โดยจะกำหนดให้ค่าใน INDEX REGISTER เป็นค่าอ้างอิง แล้วใช้คำสั่งบ่งบอกว่าตำแหน่งของข้อมูลที่ต้องการอยู่ห่างจากค่าอ้างอิงนี้เท่าใด โดยจะบอกระยะห่างในรูปของ TWO COMPLEMENT

รีจิสเตอร์ SP (STACK POINTER) มีขนาด 16 บิท ในหน่วยความจำชนิด RAM จะมีส่วนหนึ่งที่ถูกกำหนดให้เป็นที่เก็บข้อมูลชั่วคราว ส่วนนี้เรียกว่าสแตก (STACK) ซึ่งมีลักษณะการเก็บข้อมูลแบบ LIFO (LAST IN FIRST OUT) เราสามารถที่จะเก็บข้อมูลลงบนสแตกโดยใช้คำสั่ง PUSH และเมื่อต้องการดึงข้อมูลออกจากสแตกต้องใช้คำสั่ง POP

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รีจิสเตอร์ PC (PROGRAM COUNTER) เป็นรีจิสเตอร์ขนาด 16 บิทที่ใช้ในการเก็บตำแหน่งของหน่วยความจำที่ CPU จะเฟทช์ (FETCH) คำสั่งเรียบร้อยแล้ว ค่าในรีจิสเตอร์ PC จะเพิ่มขึ้น และจะชี้ไปยังตำแหน่งของคำสั่งถัดไป เราจะสามารถเปลี่ยนแปลงค่าใน PC ได้โดยใช้คำสั่ง CALL หรือ JUMP

รีจิสเตอร์ F (FALG REGISTER) ประกอบด้วย

SIGN FLAG (S) : แฟลกเครื่องหมาย

ZERO FLAG (Z) : แฟลกศูนย์

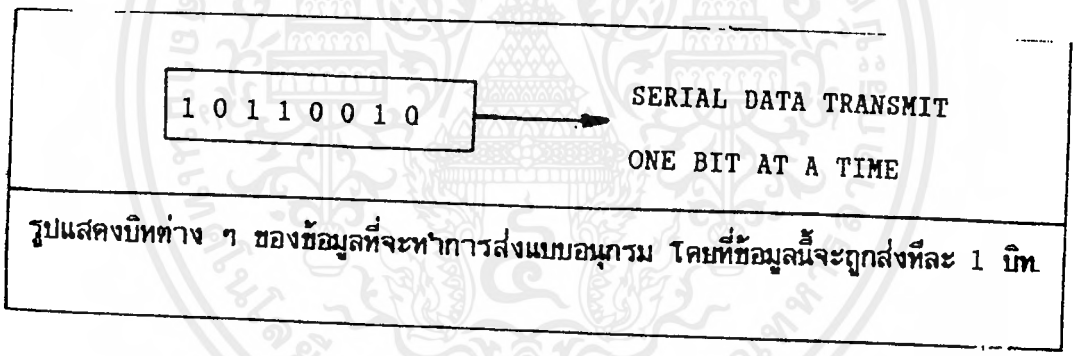
HALF CARRY FLAG (H) : แฟลกทดครึ่ง

PARITY/OVERFLOW FLAG (P/V) : แฟลกพาริตีหรือโอเวอร์โฟล์ว

SUBTRACT FLAG (N) : แฟลกตัวทด

การรับส่งข้อมูลแบบอนุกรม

การรับส่งข้อมูลแบบอนุกรมเป็นการรับส่งข้อมูลทีละบิทแทนที่จะทำการส่งข้อมูลพร้อมกันทุกบิทในเวลาเดียวกัน การรับส่งข้อมูลแบบนี้มีชื่อว่า "การรับส่งข้อมูลแบบอนุกรม (SERIAL COMMUNICATION)"



รูปแสดงบิทต่าง ๆ ของข้อมูลที่จะทำการส่งแบบอนุกรม โดยที่ข้อมูลนี้จะถูกส่งทีละ 1 บิท

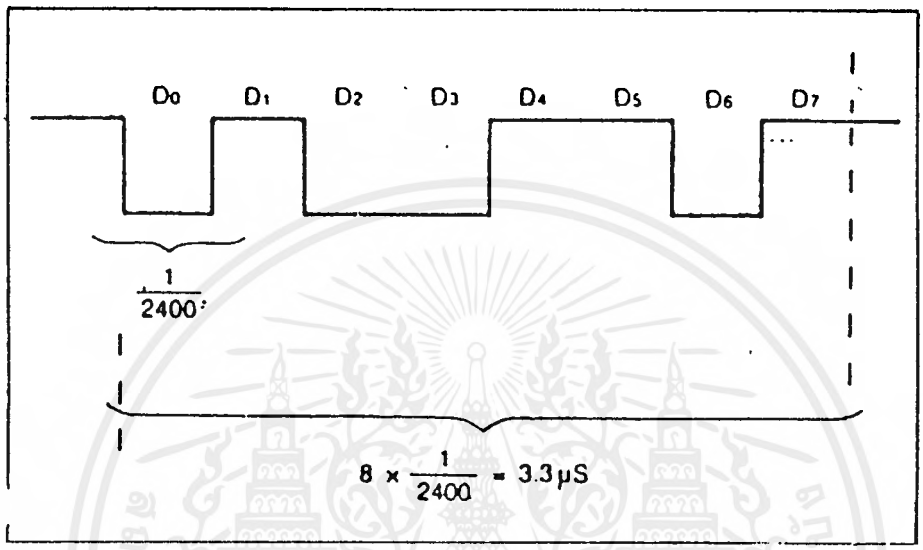
สำหรับการรับส่งข้อมูลแบบขนานนั้น ถึงแม้ว่าจะมีความเร็วสูงกว่าแบบอนุกรมอยู่มากก็ตามแต่ก็ต้องใช้จำนวนสายในการส่งผ่านข้อมูลเป็นจำนวนมากกว่าแบบอนุกรม ทำให้สิ้นเปลืองค่าใช้จ่ายในการวางสายโดยไม่จำเป็น และยังมีการลดทอนของสัญญาณมากกว่าแบบอนุกรมอีกด้วย ทำให้เกิดความผิดพลาดในการส่งผ่านข้อมูลขึ้นได้ง่าย ดังนั้นในการส่งผ่านข้อมูลในระยะทางไกลๆ เรามักจะเลือกใช้การรับส่งข้อมูลแบบอนุกรม เพื่อลดจำนวนของสายส่งซึ่งจะช่วยในการลดค่าใช้จ่ายในการวางสายลงได้อย่างมาก ถึงแม้ว่าการรับส่งข้อมูลแบบนี้จะมีความยุ่งยากและช้ากว่าการรับส่งข้อมูลแบบขนานอยู่บ้างก็ตาม

BAUD RATE สิ่งที่สำคัญมากสิ่งหนึ่งในการรับส่งข้อมูลแบบอนุกรมนี้ก็คือ ความถี่ที่ใช้ในการส่งข้อมูลซึ่งจะต้องสัมพันธ์กันระหว่างอุปกรณ์ที่ทำการรับส่งข้อมูล และความถี่ที่ใช้นี้เรียกว่า "BAUD RATE" ซึ่งหมายถึง "อัตราการรับส่งข้อมูลเป็นจำนวนบิทใน 1 วินาที" ถ้าหากว่า

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เครื่องส่งใช้ BAUD RATE ที่ไม่สัมพันธ์กับเครื่องรับแล้ว ก็จะทำให้การรับส่งข้อมูลเกิดผิดพลาดขึ้นได้

โดยทั่วไปค่าของ BAUD RATE นั้นจะใช้ค่าต่างๆ ดังต่อไปนี้คือ 110,150,3000 ,1200,2400,4800 และ 9600 สำหรับในในที่นี้จะสมมติว่าเราต้องการที่จะส่งข้อมูลแบบอนุกรมด้วยอัตรา 2400 BAUD (2400 บิท/วินาที) และข้อมูลที่ต้องการจะส่งก็คือ 0B2H หรือ 10110010B ซึ่งเราสามารถที่จะแสดงได้ในรูปของสัญญาณดังรูปที่ 4.3



รูปที่ 4.3 แสดงรูปสัญญาณของข้อมูลที่ถูกส่งไปตามสายส่งแบบอนุกรม

จากรูป จะเห็นว่าความกว้างของสัญญาณแต่ละบิทจะเท่ากับ 1/BAUD RATE วินาที ซึ่งจาก BAUD RATE ที่ต้องการที่จะใช้คือ 2400 BAUD นั้นจะทำให้ความกว้างของแต่ละบิทมีค่าเท่ากับ 1/2400 วินาที หรือ เท่ากับ 416 microsccconds ซึ่งจากความกว้างของแต่ละบิทที่จะส่งไปตามสายส่งนี้ ทำให้เราสามารถที่จะคำนวนเวลาที่จะต้องใช้ในการรับส่งข้อมูลแต่ละไบท์ (8 บิท) ได้ดังนี้คือ เท่ากับ 8 416 microseconds หรือ 3328 microseconds อย่างไรก็ตามเพื่อป้องกันความผิดพลาดที่อาจเกิดขึ้นได้ จึงมีการเพิ่มบิทต่างๆ ลงไปในแต่ละไบท์ของข้อมูล เพื่อช่วยในการตรวจสอบความถูกต้องของข้อมูลที่เครื่องรับได้รับเข้ามา สำหรับบิทต่างๆที่เพิ่มเข้ามานี้ ก็คือ START ,STOP และ PARITY BIT ซึ่งจะทำให้ข้อมูลในแต่ละไบท์นี้มีมากกว่า 8 บิท และเวลาที่ใช้ในการรับส่งข้อมูลก็จะมากขึ้นตามไปด้วย

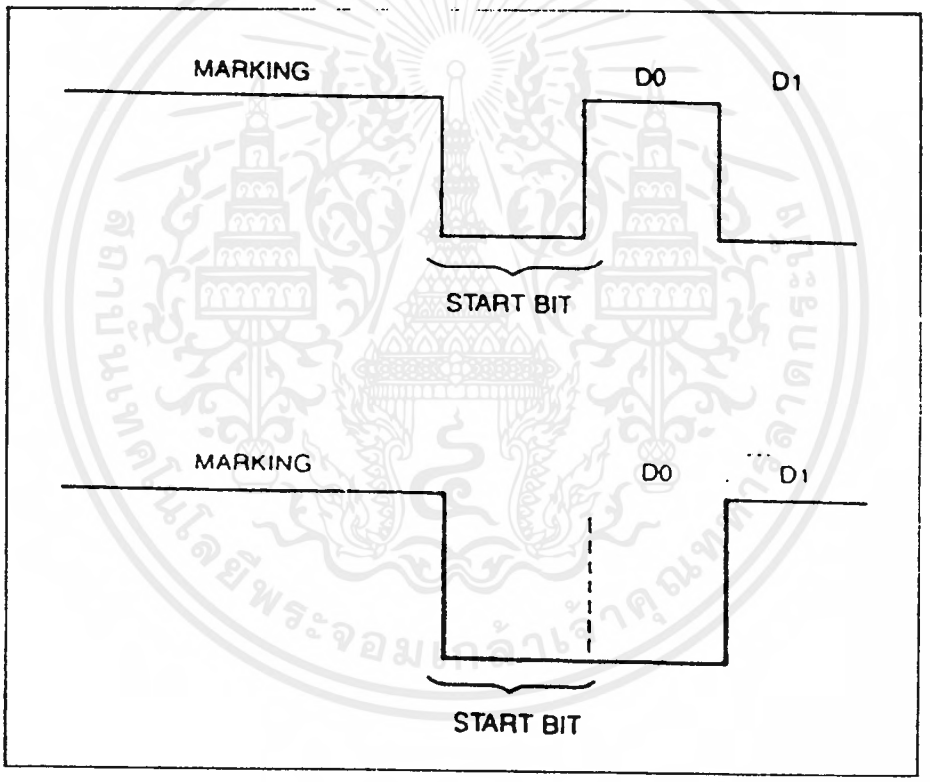
START BIT ในการส่งผ่านข้อมูลแบบอนุกรมนี้ เราจำเป็นที่จะต้องทำให้อุปกรณ์ที่จะรับข้อมูลทราบว่า ข้อมูลที่ส่งมานั้นเริ่มต้นที่จุดใด ดังนั้นเราจึงจำเป็นที่จะต้องเพิ่มข้อมูล 1 บิท ลงไปก่อนหน้าข้อมูลจริง (ACTUAL DATA) ที่จะทำการส่ง (การส่งอนุกรมจะส่งบิท D0 เป็นบิทแรก และ D7 เป็นบิทสุดท้าย) คือทำการเพิ่มบิทนี้ลงไปหน้าบิท D0 นั่นเอง และเรียกบิทนี้ว่า "START BIT"

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หน้าที่ของ START BIT นี้นอกจากจะใช้ในการบอกว่าข้อมูลนั้นเริ่มต้นที่ใดแล้ว ยังทำงานร่วมกับ STOP BIT เพื่อช่วยในการแยกข้อมูลแต่ละชุดออกจากกัน และความกว้างของบิทนี้จะเท่ากับความกว้างของบิทอื่น ๆ ในข้อมูลที่จะส่ง

เมื่ออุปกรณ์ที่จะส่งข้อมูลยังไม่ได้ทำการส่งข้อมูลใด ๆ ออกมานั้น สายส่งจะอยู่ในสภาวะที่เรียกว่า "MARKING" ซึ่งเป็นสภาวะที่ไม่มีการรับส่งข้อมูลใด ๆ เกิดขึ้นในที่นี้เราสมมติให้ MARKING ของสายส่งเป็นลอจิก "1" START BIT ที่จะเพิ่มเข้าไปนี้จะมีลอจิกที่ตรงข้ามกับลอจิกของ MARKING ดังนั้นในกรณีนี้ START BIT จะมีลอจิก "0"

สำหรับ START BIT นี้จะมีความกว้างเท่ากับ 1 บิทของข้อมูล เช่น ใน 1บิทของ ข้อมูลมีความยาวเท่ากับ 416 microseconds START BIT ก็จะมีความกว้างของสัญญาณเท่ากับ 416 microseconds ด้วย ในรูปที่ 404 จะแสดงให้เห็นถึง START BIT ที่เพิ่มเข้าไปก่อนหน้าข้อมูล



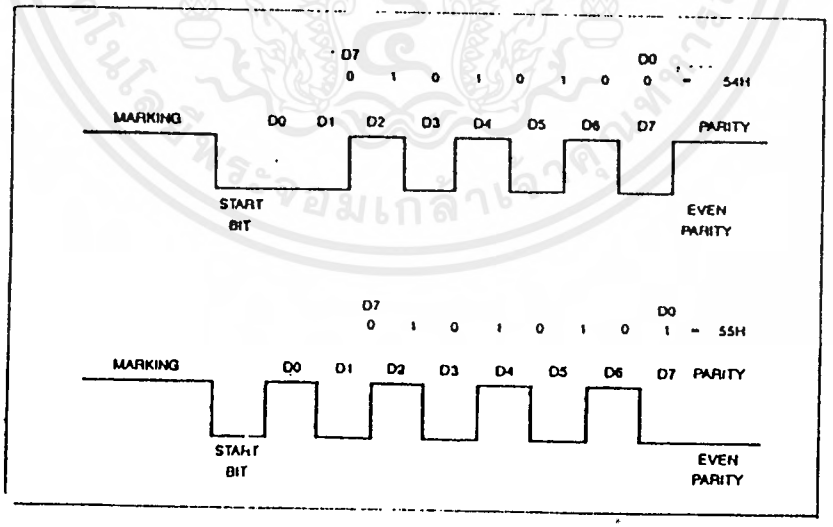
รูปที่ 4.4 การเพิ่ม START BIT เข้าไปก่อนหน้าบิท D0
ในกรณีที่บิท D0 เป็น "1" และ "0" ตามลำดับ

PARITY BIT สำหรับบิทนี้จะทำหน้าที่ในการบอกให้ส่วนรับข้อมูลทราบว่า ข้อมูลที่ได้รับเข้ามานั้นมีความถูกต้องเหมือนกับข้อมูลที่ถูกออกมาหรือไม่ โดยที่บิทนี้จะทำหน้าที่ในการบอกให้ส่วนรับข้อมูลที่ส่งออกมาแต่ละไบท์นั้นมีจำนวนบิทที่เป็น "1" อยู่เป็นจำนวนคี่ หรือเป็นจำนวนคู่ เช่นข้อมูล 54H หรือ 01010111B จะมีจำนวนบิทที่เป็น "1" อยู่เป็นจำนวนคี่เป็นต้น สำหรับบิทที่ใช้ในการตรวจสอบนี้เรียกว่า "PARITY BIT"

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

PARITY BIT นี้จะถูกส่งออกมาโดยอุปกรณ์ส่งข้อมูล ซึ่งบิทนี้จะเป็น "1" หรือ "0" นั้นขึ้นอยู่กับข้อมูลที่ส่งออกมา (D0-D7) ว่ามีจำนวนบิทที่เป็น "1" เป็นจำนวนคู่หรือคี่ และยังขึ้นกับอุปกรณ์รับส่งข้อมูลด้วยว่าถูกออกแบบ ไว้ให้รับส่ง PARITY BIT ในลักษณะของ PARITY คู่หรือ PARITY คี่ อีกด้วย

ในกรณีที่อุปกรณ์รับส่งข้อมูลถูกออกแบบไว้ให้เป็น PARITY คู่ อุปกรณ์ส่งข้อมูลจะทำการส่ง PARITY BIT เป็นลอจิก "1" ออกไปเมื่อจำนวนบิทที่เป็น "1" ของข้อมูล (D0-D7) เป็นจำนวนคี่และจะทำการส่ง PARITY BIT เป็นลอจิก "0" เมื่อจำนวนบิทที่เป็น "1" ของข้อมูลเป็นจำนวนคู่ สำหรับ PARITY คี่ก็เช่นกัน คือ PARITY BIT จะเป็น "1" ในกรณีที่จำนวนบิทที่เป็น "1" ของข้อมูลเป็นจำนวนคู่และจะเป็น "0" ในกรณีที่เป็นจำนวนคี่ ในที่นี้จะสมมติว่าอุปกรณ์ถูกออกแบบไว้สำหรับ PARUTY คู่ และเราต้องการที่จะส่งข้อมูลออกไปให้กับส่วนรับข้อมูลเป็นจำนวน 2 ไบท์คือ 54H และ 55H เมื่อเราส่งข้อมูล 54H ออกไปซึ่งมีจำนวนบิทที่เป็น "1" เป็นจำนวนคี่ ดังนั้นในกรณีนี้อุปกรณีส่งข้อมูลก็จะทำการส่ง PARITY BIT เป็นลอจิก "1" ตามออกมาด้วย เพื่อจะให้จำนวนบิทที่เป็น "1" ของข้อมูล (54H) รวมกับ PARITY BIT แล้วได้เป็นจำนวนคู่ ส่วนข้อมูล 55H นั้นจำนวนบิทที่เป็น "1" นั้นเป็นจำนวนคู่แล้ว ดังนั้นอุปกรณ์ส่งข้อมูลก็จะส่งข้อมูล PARITY BIT เป็น "0" ให้กับส่วนรับข้อมูล ดังในรูปที่ 4.5 สำหรับส่วนรับข้อมูลนั้นเมื่อทำการรับข้อมูลเข้ามาแล้วก็จะตรวจสอบสัญญาณว่าจำนวนบิทที่เป็น "1" ของข้อมูลรวมกับ PARITY BIT นั้นเป็นจำนวนคู่หรือไม่ ถ้าหากเป็นจำนวนคี่ก็แสดงว่าข้อมูลที่ได้รับเข้ามานี้ความผิดพลาดเกิดขึ้น



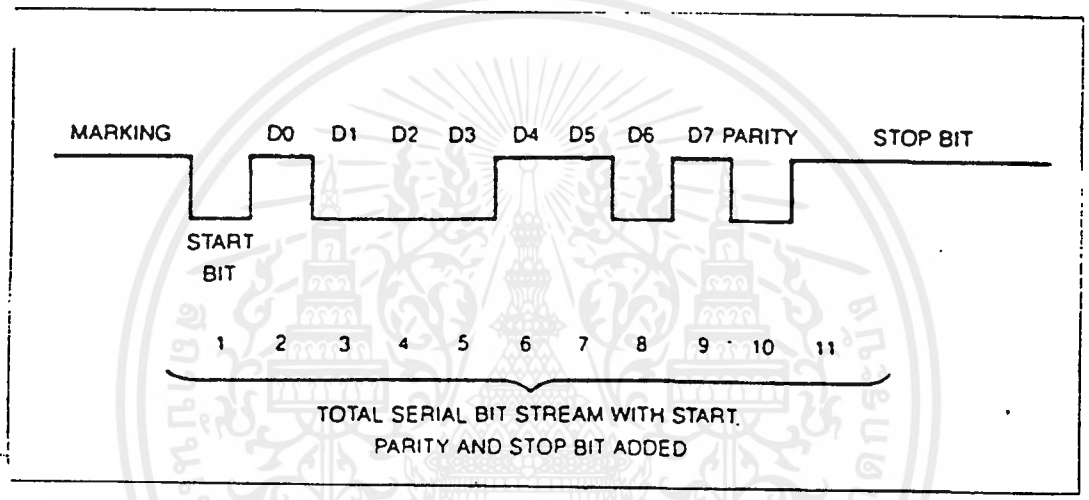
รูปที่ 4.5 การเพิ่ม PARITY BIT ลงไปในข้อมูลแต่ละไบท์

สิ่งสำคัญอีกสิ่งหนึ่งก็คือ ถ้าอุปกรณ์ส่งข้อมูลทำการส่งในลักษณะ PARITY คู่หรือคี่ก็ตามส่วนรับข้อมูลก็จะต้องทำการรับในลักษณะ PARITY เดียวกับอุปกรณ์ส่งข้อมูลด้วย เช่นในกรณี

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ที่อุปกรณ์ส่งข้อมูลทำการส่งข้อมูลในลักษณะของ PARITY คู่อุปกรณ์รับข้อมูลก็จะต้องทำการรับข้อมูลในลักษณะของ PARITY คู่ด้วย เป็นต้น

STOP BIT สำหรับบิทสุดท้ายที่เพิ่มเข้าไปนี้ จะใช้การตรวจสอบจุดสิ้นสุดของข้อมูลบิทนี้ จะถูกเพิ่มเข้าไปหลัง PARITY BIT ถ้าอุปกรณ์รับข้อมูลตรวจไม่พบบิทนี้ก็แสดงว่าข้อมูลที่ได้รับเข้ามานั้นมีความผิดพลาดเกิดขึ้น สำหรับ STOP BIT นี้อาจมีจำนวนบิทเป็น 1, 1.5 หรือ 2 ก็ได้ รูปที่ 4.6 จะแสดงข้อมูลทั้ง 8 บิทที่ส่งออกมารวมทั้ง START, STOP และ PARITY BIT ด้วยซึ่งจะเห็นว่าสิ่งที่ส่งออกมาในแต่ละไบท์นั้นไม่ได้มีเพียงข้อมูล 8 บิทเท่านั้น แต่อาจจะมีได้ถึง 12 บิท ดังนั้นถ้าเราทำการส่งด้วยอัตรา 2400 BAUD เราจะต้องใช้เวลาทั้งหมดเป็น 12 416 microseconds หรือ 4.99 milliseconds ไม่ใช่ 3328 microseconds ดังที่ได้คำนวณไว้ในตอนต้น



รูปที่ 4.6 รูปแบบของข้อมูลแต่ละไบท์ในการรับส่งข้อมูลแบบอนุกรม

หลักการเบื้องต้นของการรับส่งข้อมูลแบบอนุกรม

สำหรับหลักการเบื้องต้นของการรับส่งข้อมูลแบบอนุกรมที่ได้กล่าวถึงนั้นสามารถสรุปได้เป็นข้อ ๆ ดังนี้

1.ข้อมูลแบบขนานจากระบบจะถูกเปลี่ยนให้เป็นข้อมูลแบบอนุกรมเพื่อเตรียมที่จะส่งออกไปให้กับส่วนรับข้อมูล

2.ข้อมูลจะถูกส่งออกไปด้วยอัตราคงที่ค่าหนึ่ง ซึ่งเรียกว่า "BAUD RATE" คือ ถ้าทำการส่งข้อมูลด้วยอัตรา 1200 BAUD ก็แสดงว่าเป็นการส่งข้อมูลด้วยอัตรา 1200 บิทต่อ 1 วินาที ซึ่งก็คือ การส่งข้อมูลโดยใช้ความถี่ 1200 Hz นั่นเอง

3.ข้อมูลอนุกรมจะถูกส่งออกไปทีละบิท โดยทำการส่งบิท D0 เป็นบิทแรก และบิท D7 เป็นบิทสุดท้าย

4.ในขณะที่ยังไม่มีการส่งข้อมูลเข้าไปในสายส่ง สายส่งจะถูกทำให้อยู่ในสภาวะลอจิกใดลอจิกหนึ่ง และเราเรียกสภาวะนี้ว่า "MARKING"

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5.อุปกรณ์ส่งข้อมูลจะเพิ่มข้อมูลอีก 1 บิทเข้าไปหน้า D0 ของข้อมูลที่จะส่งให้กับเครื่องรับ บิทที่เพิ่มเข้าไปนี้เรียกว่า "START BIT" สำหรับบิทนี้จะมีลอจิกตรงข้ามกับลอจิกของ MARKING เช่น ถ้าลอจิกของ MARKING เป็น "1" ลอจิกของบิทนี้ก็จะเป็น "0"

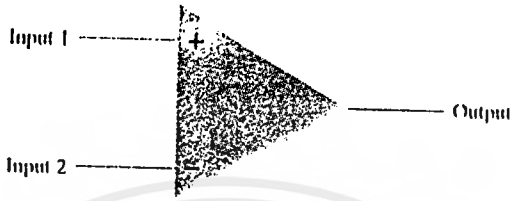
6.อุปกรณ์ส่งข้อมูลจะทำการเพิ่ม PARITY BIT เข้าไปหลังบิท D7 ของข้อมูล เพื่อใช้ในการตรวจสอบความผิดพลาดของข้อมูลที่เครื่องรับ

7.สำหรับบิทสุดท้ายที่ถูกเพิ่มเข้าไปหลัง PARITY BIT เรียกว่า "STOP BIT" ซึ่งอาจจะมีจำนวน 1, 1.5, หรือ 2 บิทก็ได้ และลอจิกของบิทนี้จะเป็นลอจิกเดียวกับลอจิกของ MARKING

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5. Operational Amplifiers เบื้องต้น

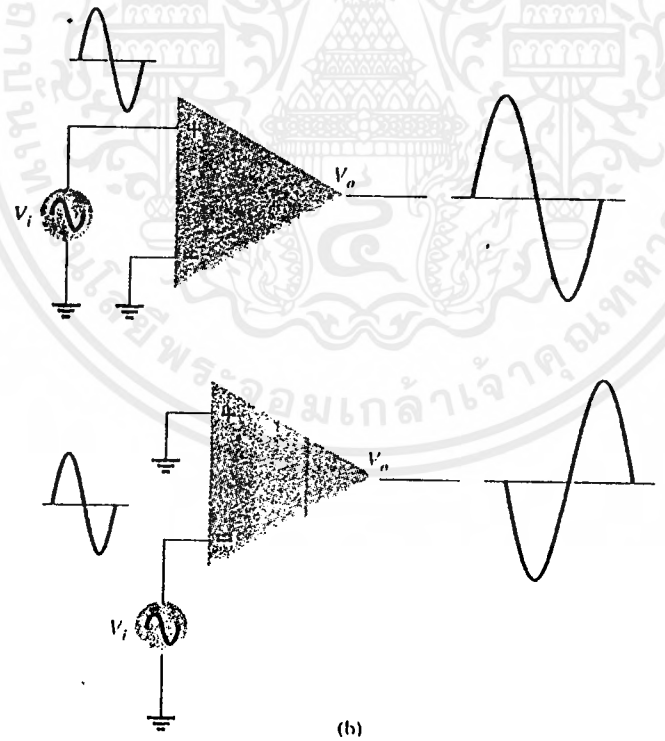
- Op-Amp เป็นอุปกรณ์ที่มีอัตราขยายสูงและมี input impedance สูง output impedance ต่ำ การใช้งาน Op-Amp โดยสามารถทำการเปลี่ยนระดับแรงดัน การ Oscillator และการวงจร Fillter รูปที่ 5.1 แสดงรูปแบบพื้นฐานของ Op-Amp ซึ่งจะมี 2 input และ 1 output การทำงานจะนำผลต่างของ 2 input นี้มาทำการขยายสัญญาณ



รูปที่ 5.1 Basic Op-Amp

Single-Ended Input

ผลที่ได้ของ Single-Ended Input คือเมื่อทำการต่อสัญญาณเข้าทาง input เดียวแล้วอีก input ต่อลง ground จะได้ลักษณะสัญญาณดังรูปที่ 5.2



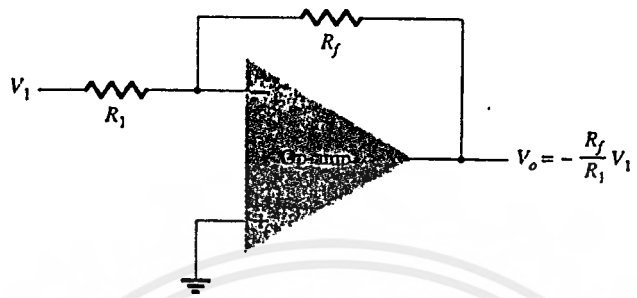
รูปที่ 5.2 Single-ended operation

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Inverting Amplifier

เป็นวงจรที่ให้อัตราการขยายที่คงที่ โดยจะให้ output ที่ลักษณะสัญญาณที่กลับกันกับสัญญาณของ input

$$Vo = -[Rf/Ri]VI$$



รูปที่5.3 Inverting constant-gain multiplier

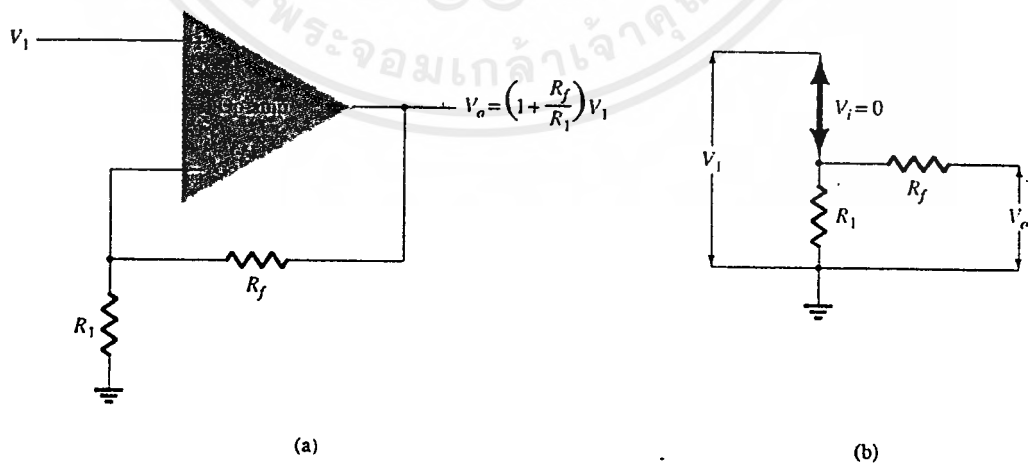
Noninverting Amplifier

เป็นการต่อวงจรตามรูปที่ 5.4 การใช้งานจะใช้ในรูปแบบของ noninverting จะให้ความคงที่ของความถี่ได้ดีกว่าการคำนวณค่าอัตราการขยายของวงจรจะสามารถหาได้จากสูตร

$$VI = [R1/(R1+Rf)]Vo$$

ซึ่งก็คือ

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



รูปที่ 5.4 Noninverting constant-gain multiplier

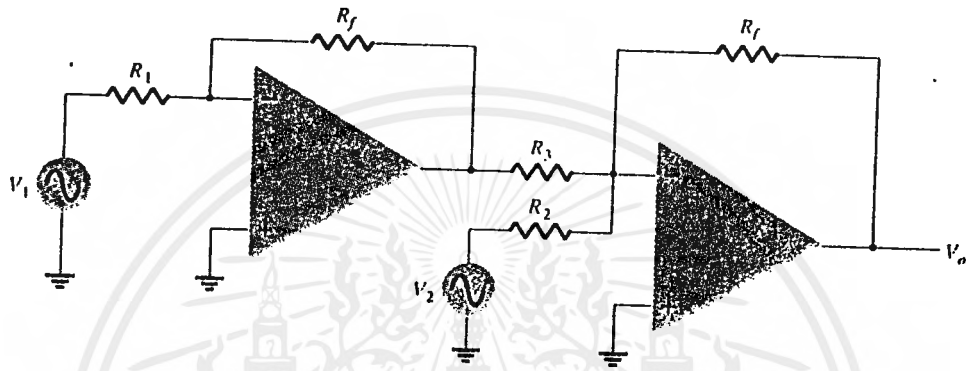
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Voltage Subtraction

สัญญาณ 2 สัญญาณสามารถมาหักล้างกันได้ โดยสัญญาณหนึ่งมาหักล้างกับอีกสัญญาณหนึ่ง ตามรูปที่ 5.5 โดยใช้ Op-Amp 2 ตัวผลที่ได้ของ output จะเป็น

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

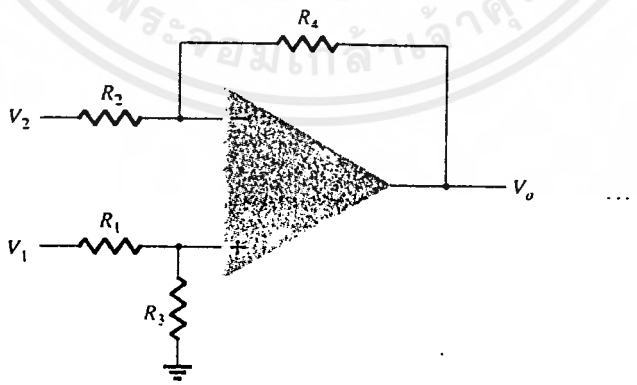
$$\boxed{V_o = -\left(\frac{R_f}{R_2}V_2 - \frac{R_f}{R_3}\frac{R_f}{R_1}V_1\right)}$$



รูปที่ 5.5 Op-Amp Applications Voltage Subtraction

การต่อวงจรในรูปแบบอื่นๆ ที่ทำการหักล้างสัญญาณ 2 สัญญาณตามรูปที่ 5.6 การต่อแบบนี้จะใช้ Op-Amp เพียงตัวเดียวและ output ที่ได้จะเป็น

$$\boxed{V_o = \frac{R_3}{R_1 + R_3}\frac{R_2 + R_4}{R_2}V_1 - \frac{R_4}{R_2}V_2}$$



รูปที่ 5.6 Subtraction circuit only one 0p-amp

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

วงจรอินสตรูเมนเตชั่นแอมปลิไฟเออร์

จากที่ผ่านมาพบว่า หากเราต้องการจะปรับเปลี่ยนอัตราขยายโดยรวมให้กับวงจรขยายผลต่างนั้น ค่อนข้างจะเป็นเรื่องยุ่งยากทีเดียว เพราะจะต้องรักษาอัตราส่วนของ R2/R1 และ R4/R2 ให้เท่ากันตลอดเวลา เช่นถ้า R2 มีค่าเพิ่มขึ้น 25% R4 ก็จะต้องมีค่าเพิ่มขึ้น 25% ด้วยเช่นกัน ในทางปฏิบัติเราไม่สามารถจะควบคุมอัตราส่วนต่าง ๆ เหล่านี้ได้ในเวลาเดียวกัน จึงไม่สะดวกที่จะนำวงจรขยายผลต่างไปใช้ในงานที่ต้องการปรับเปลี่ยนอัตราขยาย

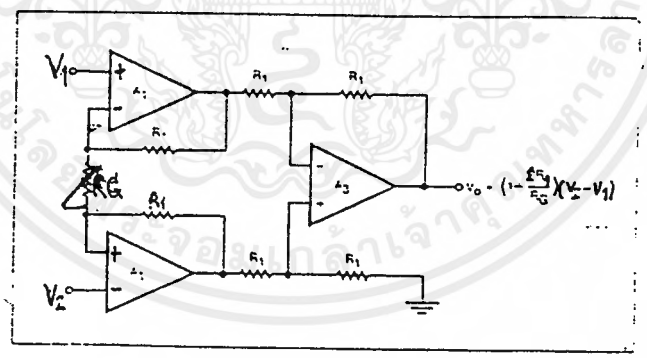
จึงมีการปรับปรุงขุดด้อยของวงจรขยายผลต่างมาเป็นวงจรอินสตรูเมนเตชั่นแอมปลิไฟเยอร์ (instrumentation amplifier) ดังรูปที่ 5.7 เพียงเพิ่มออปแอมป์เข้าไป 2 ตัว ตัวต้านทานอีก 3 ตัว ต่อทางด้านอินพุทของวงจรขยายผลต่างเดิม ออปแอมป์ทั้งสองจะถูกต่อแบบโวลเตจฟอลโลเวอร์ ซึ่งจะมีอิมพีแดนซ์ด้านอินพุทที่สูงและบัฟเฟอร์ให้วงจรด้วยโดยมีอินพุท V1 และ V2 อยู่ที่ขา noninverting ดังรูป และหากกำหนดให้ตัวต้านทานปรับอัตราขยาย (Rg) ก็จะได้สมการอัตราขยายความแตกต่าง Ad ดังนี้

$$Ad = 1+2R1/Rg$$

และสมการแรงดัน output ของวงจรอินสตรูเมนเตชั่นแอมปลิไฟเออร์ก็คือ

$$Vo = (1+2R1/Rg)(V2-V1)$$

จากสมการจะเห็นว่าเราสามารถควบคุมอัตราขยายวงจรขยายผลต่างได้ง่าย ๆ ด้วยการเปลี่ยนค่า Rg เท่านั้น



รูปที่ 5.7 วงจร instrumenttation amplifier

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

6. ความรู้พื้นฐานของ DOT MATRIX LCD MODULE

อุปกรณ์ในปัจจุบันในส่วนแสดงผลนั้นจะใช้ LCD เสียเป็นส่วนใหญ่ไม่ว่าจะเป็นเครื่องคิดเลข เครื่องถ่ายเอกสาร เครื่องมือวัดต่าง ๆ ไปจนถึง คอมพิวเตอร์ เราสามารถแบ่งการแสดงผลแบบ Dot Matrix LCD ได้ 3 แบบด้วยกันคือ

1.Character LCD MODULE

2.Graphic LCD MODULE

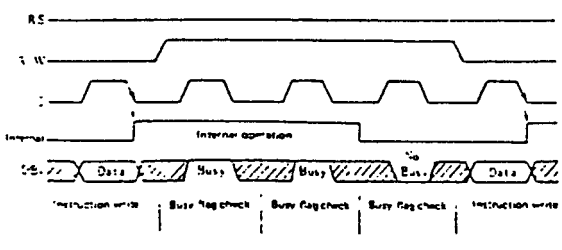
3.Segment Display ชนิด LCD MODULE

โดยในแต่ละแบบนี้ก็จะมีส่วนประกอบใหญ่ ๆ แบ่งได้เป็น

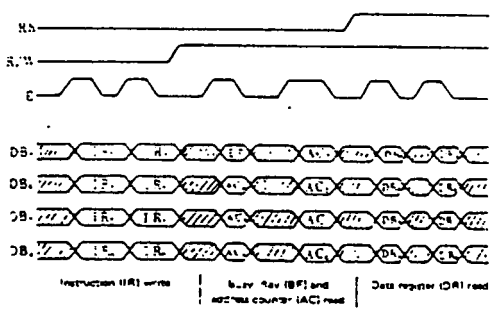
1. DOT MATRIX LCD เป็นตัวแสดงผลให้เราได้มองเห็นในลักษณะการปิด และ เปิดตัวเอง กับแสงภายนอก ส่วนของที่เป็นตัวกระจกบรรจุผลึก
2. DRIVER เป็นตัวรับสัญญาณจากตัวควบคุมมาขับผลึก LCD อีกทีหนึ่งโดยมีเบอร์ที่นิยมใช้ใน LCD Module เช่น HD44100H , MSM5259
3. CONTROLLER เป็นตัวรับข้อมูลจากตัวอุปกรณ์ภายนอกมาและจัดการควบคุม LCD Module ให้ทำงานแสดงผลต่าง ๆ เช่น การลบจอภาพ , การเกิดตัวอักษร เป็นต้น

การควบคุม Controller ของ LCD นั้นต้องเข้าใจหลักการทำงานของ Controller ก่อน LCD Module ของแต่ละบริษัทจะใช้ตัว Controller ที่มีหลักการทำงานเหมือน ๆ กันเป็นส่วนใหญ่และใน LCD MODULE แต่ละขนาดจำนวนตัวอักษรหรือจำนวนบรรทัดก็มีหลักการทำงานแบบเดียวกันทั้งหมด IC ที่นิยมใช้มากที่สุดตัวหนึ่งที่เป็น Controller LCD ก็คือเบอร์ HD44780 โดยรูปแบบการทำงานของมันได้เป็นมาตรฐานให้กับ Controller LCD ตัวอื่น ๆ ด้วย

HD44780 เป็นไอซี LSI ตัวหนึ่งใช้ควบคุม LCD โดยแสดงผลในรูปตัวอักษรหรือสัญลักษณ์ต่าง ๆ ตัวมันเองสามารถต่อใช้งานได้ทั้งแบบ 4 Bit และ 8 Bit โดยถ้าเราต่อแบบ 4 Bit จะต่อใช้งานที่ DB7-DB4 เท่านั้นโดยข้อมูลครั้งแรกที่ส่งนั้น HD44780 จะถือเป็นข้อมูล 4 Bit บน และข้อมูลที่ส่งต่อมานั้นเป็นข้อมูล 4 Bit ล่าง จากรูปจะแสดงโครงสร้าง และการทำงานภายในตัว Controller และ Timing Diagram ในการรับข้อมูลของตัว Controller

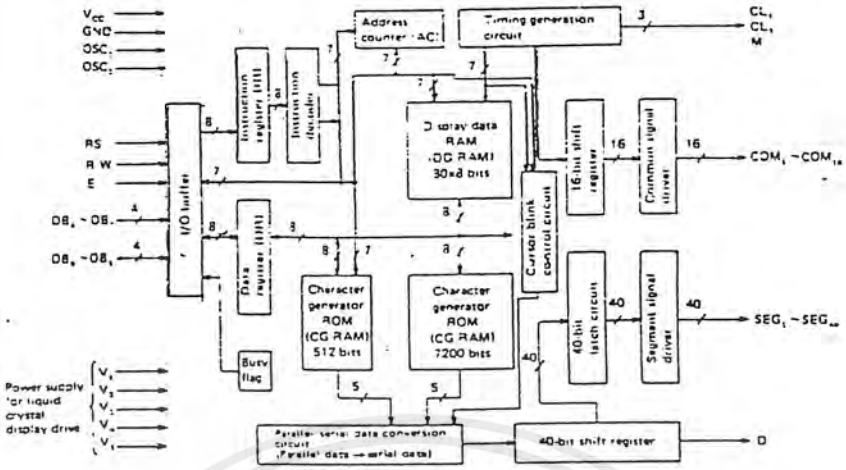


สัญญาณการติดต่อแบบ 8 Bit



สัญญาณการติดต่อแบบ 4 Bit

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



โครงสร้างของ Controller

รูปที่ 6.1แสดงโครงสร้างของ Controller และ Time Diagram

จากรูปที่ 6.1 จะเห็นว่าโครงสร้างภายในของ Controller จะประกอบหลายส่วนด้วยกัน ได้แก่

I/O Buffer เป็นวงจรในการติดต่อข้อมูลจากภายนอกเพื่อที่จะส่งข้อมูลเข้าไปยังภายในตัว Controller

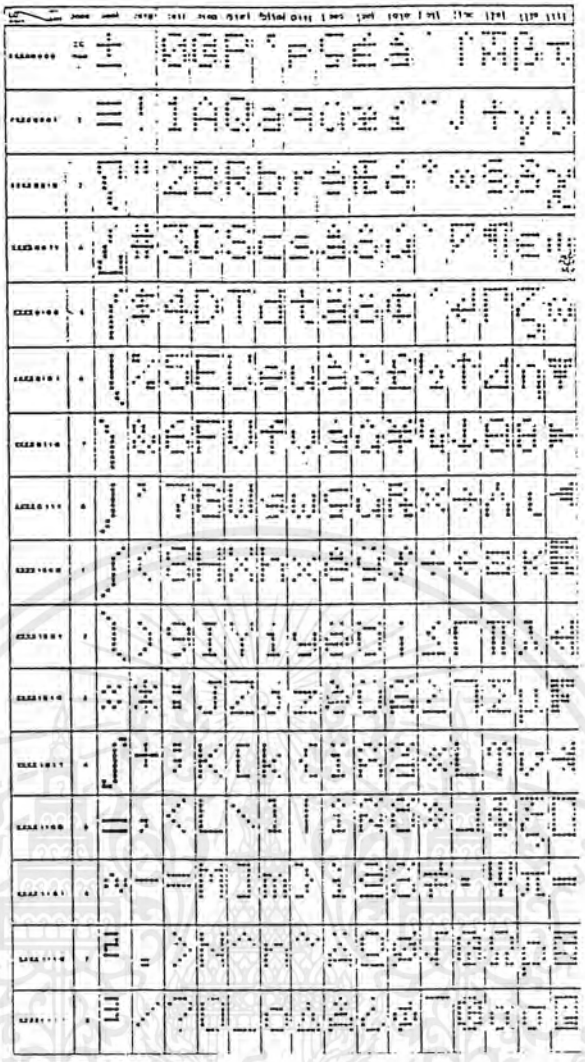
Instruction Register (IR) เป็นรีจิสเตอร์ที่จะรับส่งจากภายนอกเพื่อจะควบคุมการแสดงผล เช่น การจัดการแสดงผล,การ Clear จอ ,กำหนดตำแหน่งของการแสดงผล เป็นต้น

DATA Register (DR) เป็นรีจิสเตอร์ที่จะใช้รับส่งข้อมูลจากภายนอกส่งไปให้ DD RAM หรือ สร้างตัวอักษรเพิ่มเติมให้กับ CG RAM

Display Data RAM (DD RAM) เป็นหน่วยความจำที่สามารถเขียนข้อมูลเข้าไปได้โดยผ่าน DATA Register (DR) ซึ่งตัว Controller จะนำข้อมูลที่อยู่ใน DD RAM ไปชี้ค่าที่ตาราง หรือ ตำแหน่งของ CG ROM หรือ CG RAM เพื่อจะนำไปแสดงที่หน้าจอ LCD

Character Generater ROM (CG) เป็นหน่วยความจำที่ถาวรใช้เก็บ Character ของตัวอักษรหรือ สัญลักษณ์ ที่จะนำไปแสดงที่หน้าจอ LCD และยังสามารถสร้าง Font ขึ้นมาใช้เองได้ โดยการเขียนข้อมูลลงไปใน CG RAM และชี้ FONT ได้โดยการกำหนดค่าที่ DD RAM ตารางของข้อมูล CGROM จะแสดงในตารางที่ 1

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



ตารางที่ 1 แสดงข้อมูลภายใน CG ROM

Busy Flag เป็นตัวบอกสถานะการทำงานของ Controller ให้อุปกรณ์ภายนอกทราบว่า Controller พร้อมที่จะรับข้อมูล หรือ คำสั่ง ที่จะทำการส่งไปให้ตัว Controller หรือยัง โดยอุปกรณ์ภายนอกทำการอ่านสัญญาณตัวนี้เสร็จหรือยัง

เมื่อเราป้อนไฟให้ HD4478 นั้นก็จะทำการ Reset ตัวเองโดยใช้เวลาประมาณ 10 ms หลังจากไฟ VDD ถึง 4.5 V แล้วโดยจะ SET ตัวเองดังนี้

1. Display Clear จะทำการลบข้อมูล
2. Function Set โดยจะ SET ค่าภายใน

   DL = 1 เป็นการ Set ให้การติดต่อเป็นแบบ 8 Bit

   N = 0 Set เป็น 1 บรรทัดการแสดงผล

   F = 0 5*7 Dot ต่อหนึ่งตัวอักษร

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3. Display ON/OFF

D = 0 Display OFF

C = 0 Cursor OFF

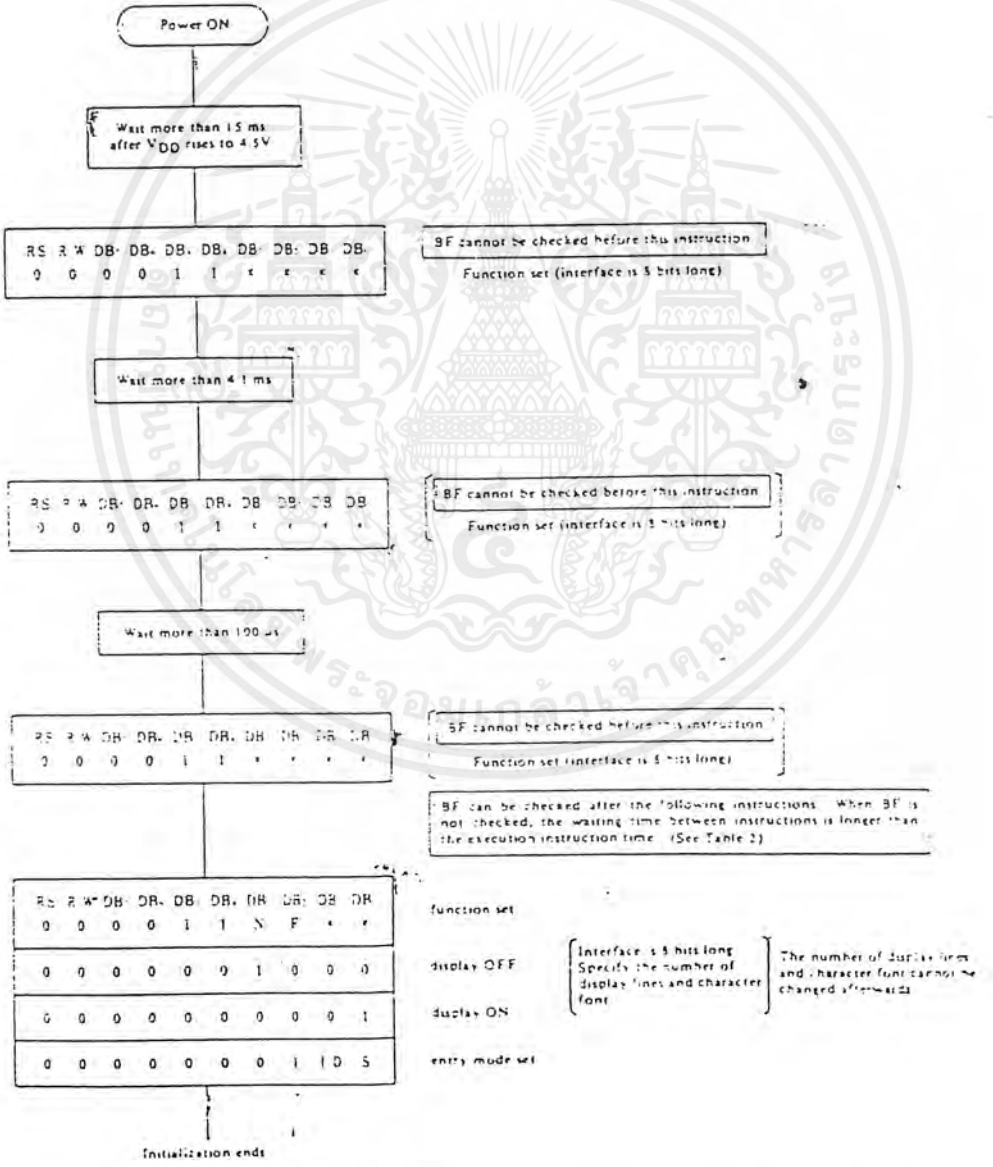
B = 0 Blink OFF

4. Entry Mode Set

I/O = 1 (เพิ่มค่า Counter ขึ้น 1)

S = 0 NO Shift

เมื่อเราเริ่มเปิดเครื่องทำงานแล้วก็จะต้องส่งคำสั่งควบคุมให้มันเริ่มทำงานตาม Flow ที่ 1 ดังนี้

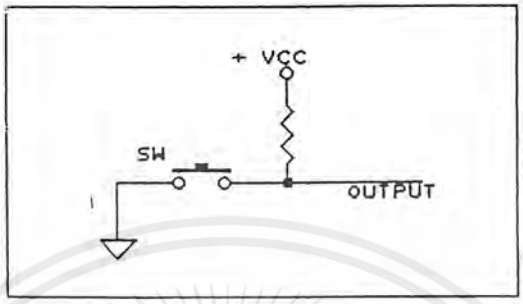


Flow Chart ที่ 1 การติดตั้งระบบการแสดงของ LCD เมื่อเปิดเครื่อง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

7. การต่อสวิทช์จำนวนมาก (Key Board)

สวิทช์เป็นอุปกรณ์พื้นฐานอย่างหนึ่งสำหรับระบบไมโคร เราใช้สวิทช์เพื่อเป็นอุปกรณ์ในการตรวจจับสัญญาณบางอย่างที่ต้องการ เช่น การเข้ารหัส ลิมิตสวิทช์ เป็นต้น โดยจะทำการต่อสวิทช์ เข้ากับบิทใดของพอร์ตอินพุท ลักษณะการต่อสวิทช์เข้ากับพอร์ตอินพุตจะต่อวงจรตามรูปที่ 7.1

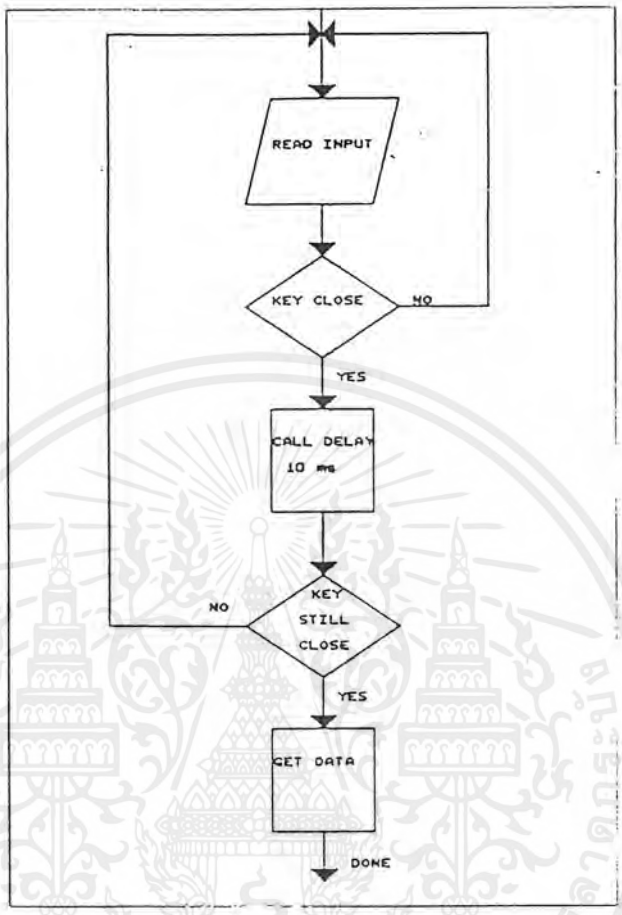


รูปที่ 7.1 วงจรในการต่อสวิทช์ เข้ากับพอร์ตอินพุต

จากรูปที่ 7.1 D0-D7 ของพอร์ตอินพุตต่อกับสวิทช์ และมีรีจิสเตอร์ Pull up ในการยกระดับของอินพุต เมื่อไม่มีการกดสวิทช์ ข้อมูลที่รับเข้าไปจะเป็น "1" ถ้ามีการกดสวิทช์โดยทั่วไปจะปรากฏ พัลส์ที่ไม่ต้องการขึ้นมา ซึ่งเกิดจากการกระเด้งของหน้าสัมผัสปรากฏการณ์นี้เรียกว่า การ บราวซ์ (Bouncc) สามารถทำได้โดยวิธีทาง Hard warc หรือ Sofc warc ก็ได้ วิธีการทาง Hard warc โดยทั่วไปจะใช้วงจร Mono stable เพื่อที่จะหน่วงเวลาของสัญญาณ ในลักษณะการเกิดบาวนซ์ มีผลต่อการป้อนสัญญาณในวงจร Logic เป็นอย่างมาก คือเมื่อเราต้องการกดสวิทช์ 1 ครั้งเพื่อที่จะได้ 1 พัลส์ แต่ถ้ามีการเกิดบราวส์ สัญญาณจะได้มีมากกว่า 1 พัลส์ ค่าที่จะทำการตรวจสอบที่ไม่แน่นอน อาจจะเป็น 0 หรือ 1 ก็ได้ ดังนั้นการกำจัดสัญญาณบาวนซ์ อาจจะทำได้โดยวิธีการทางด้าน ฮาร์แวร์ หรือ ซอฟแวร์แต่ที่นิยมใช้ในระบบไมโครคอมพิวเตอร์ก็คือด้วยซอฟแวร์เพราะไม่ต้องเพิ่มอุปกรณ์เข้าไป

การแก้สัญญาณ บาวนซ์โดยใช้ซอฟแวร์ โปรแกรมจะทำงาน โดยการตรวจสอบ การกดสวิทช์ครั้งแรก แล้วรอสักระยะเวลาช่วงหนึ่ง เพื่อที่จะให้ผ่านช่วงเวลาที่เกิดการ บาวนซ์ผ่านไป แล้วก็ทำการอ่านข้อมูลจากสวิทช์อีกครั้งหนึ่งเพื่อจะนำไปเป็นข้อมูล การแก้สัญญาณบาวนซ์นั้นสามารถนำมาเขียนเป็น Flow Chart ที่ 1 ดังต่อไปนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



Flow Chart ที่ 1 ในการแก้สัญญาณบาวนซ์

จากรูปที่ 1 จะเห็นว่าการรับข้อมูลจากสวิทช์จะใช้ 1 Bit ต่อสวิทช์ 1 ตัวแต่ในการใช้งานใน Micro Controller ต้องการใช้สวิทช์หรือ Key Board จำนวนมาก ดังนั้นถ้าใช้ 1 สวิทช์ต่อ 1 Bit จะทำให้สิ้นเปลืองพอร์ต เป็นจำนวนมาก เช่น ถ้าต้องการรับข้อมูลจาก สวิทช์ 32 ตัว ต้องใช้พอร์ตแบบ 8 Bit จำนวนถึง 4 พอร์ตด้วยกัน

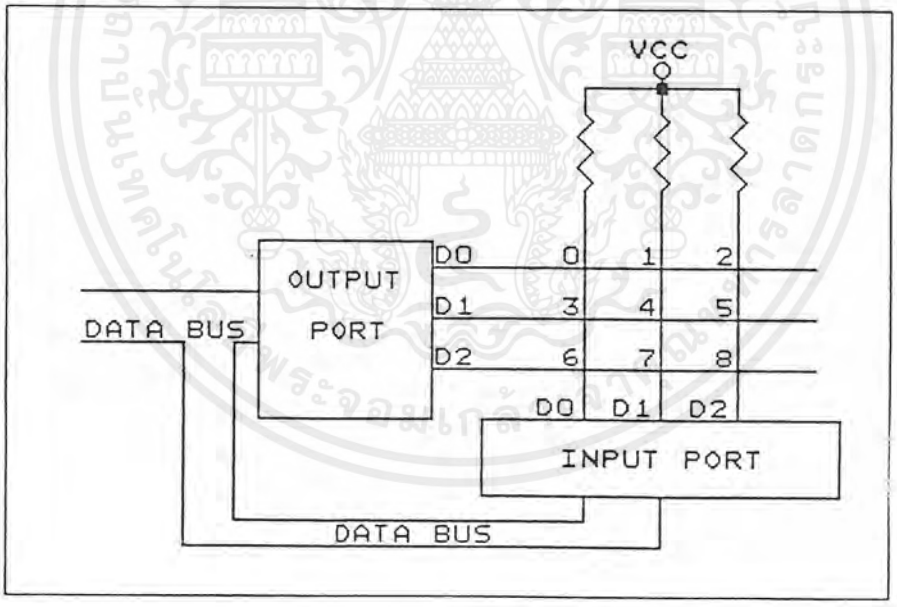
การต่อสวิทช์แบบที่ประหยัดพอร์ต และนิยมใช้ในปัจจุบัน คือ การต่อแบบแมทริกซ์ การต่อด้วยวิธีนี้จะลดจำนวนการใช้พอร์ตลง และจำนวนของสายในการต่อลงด้วยจากตารางที่ 1 เป็นการแสดงการใช้จำนวนของสายในการรับข้อมูลจากสวิทช์แบบธรรมดาและแบบแมทริกซ์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จำนวนของสวิทช์	จำนวนของสายที่ใช้ในการต่อสวิทช์	
	แบบธรรมดา	แบบแมทริกซ์
1*1	1	1
2*2	4	4
4*4	16	8
5*5	25	10
8*8	64	16

ตารางที่ 1

จากตารางที่ 1 จะว่าการต่อสวิทช์แบบแมทริกซ์จะประหยัดสายได้มากกว่าการต่อสวิทช์แบบธรรมดาสะดวกในการออกแบบทาง ฮาร์ดแวร์ และการตรวจสอบสวิทช์ที่กด จะใช้ ซอฟท์แวร์เข้ามาช่วยในการรับข้อมูลจากสวิทช์ โดยซอฟแวร์

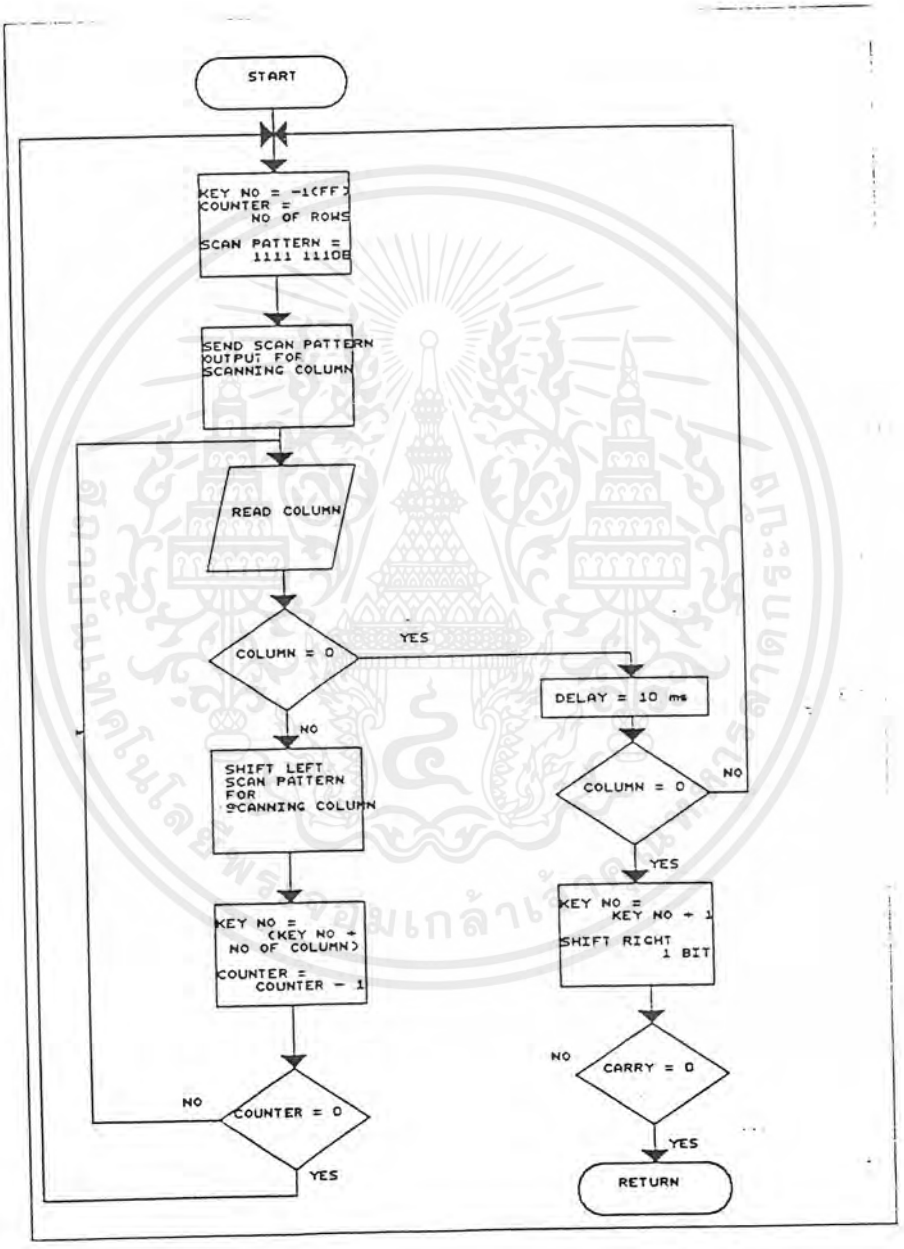


รูปที่ 7.3 การต่อสวิทช์แบบ แมทริกซ์ 3*3

หลักการทำงานของโปรแกรมจะใช้หลักการในการกวาด (Scanning) เพื่อหาว่าสวิทช์ใดถูกกดโดยอาจเริ่มต้นที่ แถวที่ 0 โดยกำหนดให้แถวที่ 0 มีระดับเป็น 0 ส่วนแถวอื่นเป็น 1 แล้วทำการตรวจสอบทีละ Column โดยเริ่มจากคอลัมน์ 0 ตรวจสอบว่าทาง Row ว่าเป็น 0 หรือไม่ ถ้าไม่ก็ให้ทำการเพิ่มค่าของการ Scan ทาง Column แล้วทำการเช็คทาง Row ทำไปเรื่อย ๆ จนกว่าจะครบการ Scan ทาง Column แล้วทำการเช็คทาง Row ทำไปเรื่อย ๆ จนกว่าครบการ Scan

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ทาง Column ทุก Column ถ้ายังตรวจสอบไม่เจอว่าทาง Row ไม่เป็น 0 ก็แสดงว่าไม่มีการกด Key แต่ถ้ามีถ้าเช็คว่ามี 0 เกิดขึ้นที่ Row ก็ให้ทำการ หน่วงเวลา 10 ms แล้วทำการเช็คข้อมูลทาง Row อีกครั้ง นำเอามา ทำให้เป็น ตำแหน่งของ Key ทาง Row และ Column ดังแสดงใน Flow Chart ที่ 2



Flow Chart ที่ 2 แสดงการรับข้อมูลจาก Key 3*3

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รายละเอียดส่วนต่างๆของเครื่องชั่งแบบละเอียด

ส่วนของ SOFT WARE

เครื่องชั่งแบบละเอียดนี้จะอ่านค่าจากการเปลี่ยนแปลงความต้านทาน ของ stain gage แล้วใช้ไฟฟ้ากระแสตรงป้อนโดยนำหลักการ ของวงจร Bridg ทำให้ได้การเปลี่ยนแปลงของระดับแรงดันจากวงจรขยายเบื้องต้น นำมาป้อนให้กับไมโครโปรเซสเซอร์ โดยผ่านอุปกรณ์เปลี่ยนสัญญาณ Analog เป็น Digital ตัวไมโครโปรเซสเซอร์จะทำการประมวลผลและทำงานต่างๆตาม SOFT WARE ที่ได้เขียนขึ้นนี้ จะทำงาน

1.แสดงค่าน้ำหนักที่ได้เป็นตัวเลขออกทางจอแสดงผล

2.คำนวนราคาต่อหน่วยน้ำหนักโดยใส่ราคาต่อหน่วยน้ำหนักเข้าให้เครื่อง เครื่องจะทำการประมวลผลออกมาเป็นราคาทั้งหมด

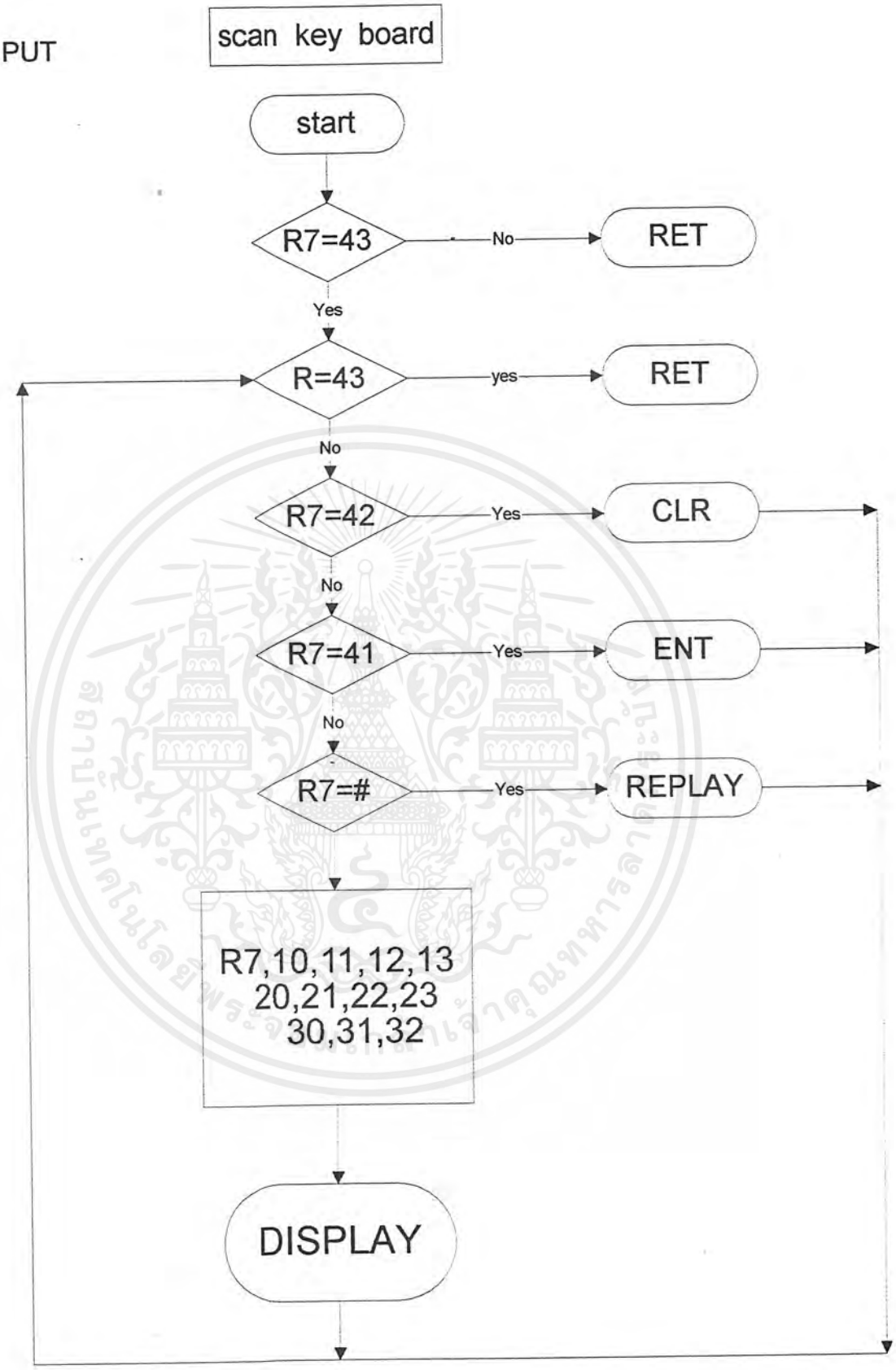
3.เก็บค่าน้ำหนักที่ชั่งครั้งสุดท้าย

SOFT WARE ที่เขียนขึ้นสามารถอธิบายได้โดย FLOW CHART ได้ดังนี้

และในส่วนของ SOFT WARE ในส่วนถัดไป

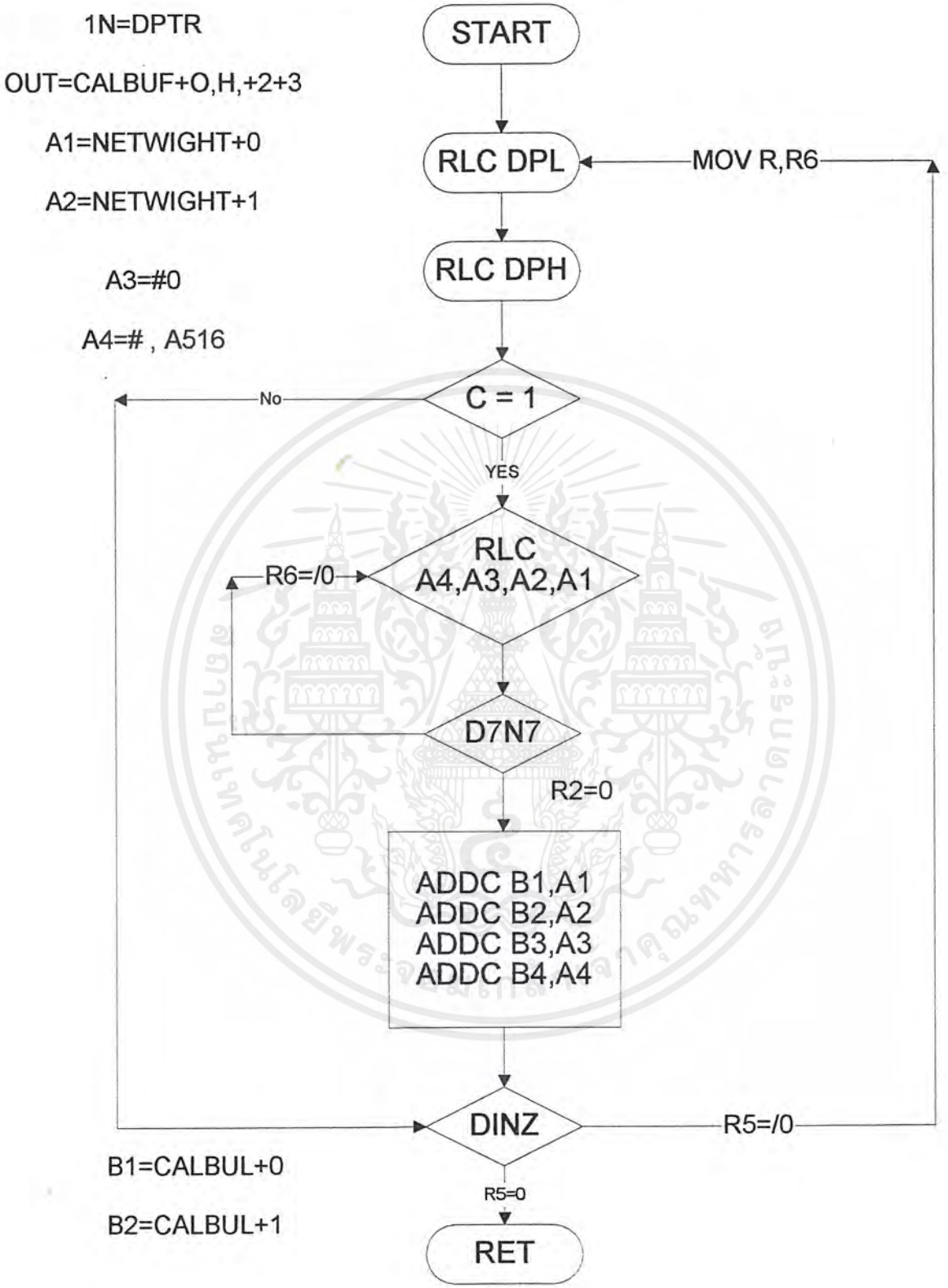
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

R7 = OUT PUT



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โปรแกรมคำนวน ขนาด 16 บิต (ผลลัพท์ 32 บิต)

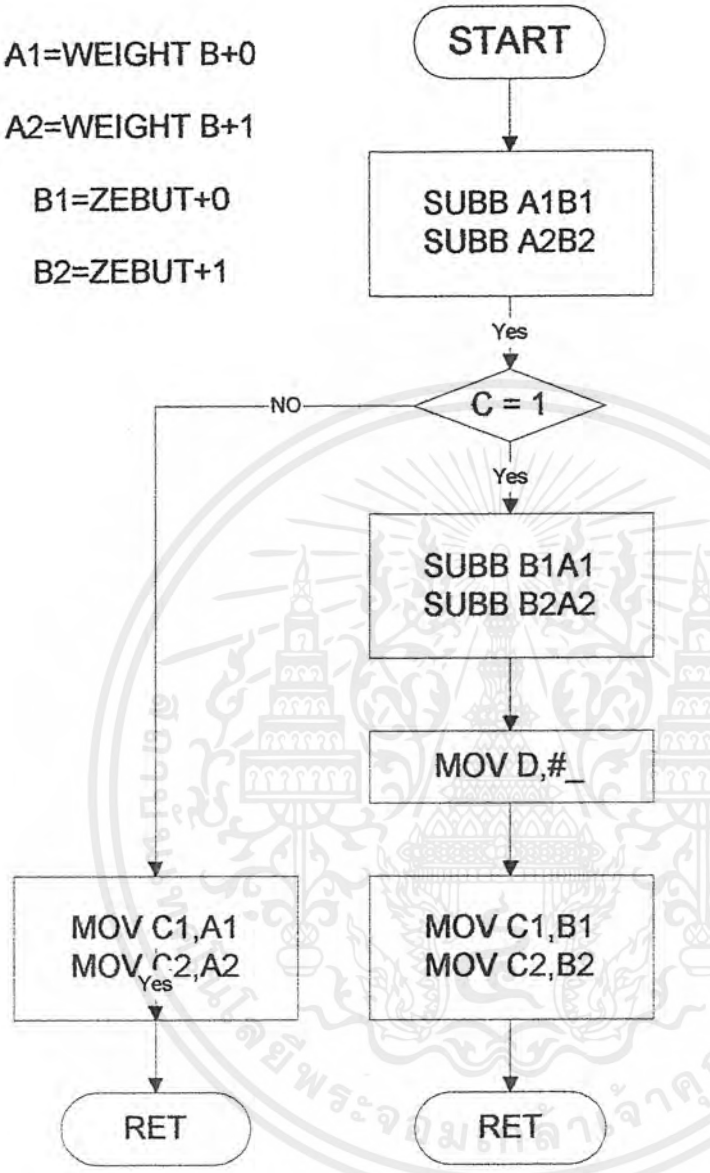


เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

B3=CALBUL+2

B4=CALBUL+3

โปรแกรม ZERO SET



C1=NETWEIGHT+0

C2=NETWEIGHT+1

D=LCDBUFA+0

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
; FILENAME      FINESCAL.ASM
; ASSEMBLER    SXA51
; WRITER       CHAO KASAMKUL
; FACULTY      INSTRUMENTATION OF ENGINEERING
; I/O PORT     MAX192          JAZZ-31
;              ============    ==========
;              PIN-18 CS\  --- PIN-1 P1.0
;              PIN-19 SCLK --- PIN-2 P1.1
;              PIN-17 DIN  --- PIN-3 P1.2
;              PIN-15 DOUT --- PIN-4 P1.3

; FUNCTION     DISPLAY ON 7-SEGMENT
;              (DISPLAY VALUE X 4 = ANALOG INPUT VOLTAGE)

; ********** VARIABLE **********

        CSBIT       EQU   P1.0      ;CHIP SELECT
        SCLKBT      EQU   P1.1      ;SERIAL CLOCK
        DINBIT      EQU   P1.2      ;DATA IN
        DOUTBT      EQU   P1.3      ;DATA OUT
        XLCDWRC     EQU   0C000H    ;LCD WRITE CONTROL (ANT-31PJ)
        XLCDRDC     EQU   0C001H    ;LCD READ CONTROL
        XLCDWRD     EQU   0C002H    ;LCD WRITE DATA
        XLCDRDD     EQU   0C003H    ;LCD READ DATA
        PORTA       EQU   0E000H    ;PORTA
        PORTB       EQU   0E001H    ;PORTB
        PORTC       EQU   0E002H    ;PORTC
        CONT        EQU   0E003H    ;8255 CONTROL
        REPLAYB         EQU   8840H
        ADCBUF      EQU   20H       ;ADC BUFFER (2 BYTES)
        TB1BUF      EQU   22H       ;TABEL1 BUFFER (1 BYTE)
        ZERBUF      EQU   23H       ;ZEROBUF 2 BYTE
        HEXBUF      EQU   25H       ;HEX BUFFER (1 BYTES)
        WEIGHTB         EQU   26H       ;WEIGHT BUFFER 2 BYTES
        CALBUF      EQU   28H       ;BUFFER OF WEIGHT (4 BYTES)
        NETWB       EQU   2DH       ;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
DIGIT           EQU     2FH
LCDBUFA                 EQU     30H     ;BUFFER FOR LCD MODULE
; ********** MAIN **********
; 24-BIT EXTERNAL CLOCK MODE
; ********** INTERNAL RAM **********

                ORG     8100H
                                        ;**** USER AREA ****
                DS      8               ;USER AREA (REGISTER BANK-0)
SYSSTK:         DS      24              ;SYSTEM STACK
ENDINT:

                ORG     8100H

RES:            MOV     R2,#80H         ;POWER UP DELAY
RES1:           MOV     R3,#0
                DJNZ    R3,$
                DJNZ    R2,RES1
                MOV     SP,#SYSSTK

                MOV     A,#81H          ;SET CONTROL (CONSOLE PORT)
                MOV     DPTR,#CONT
                MOVX    @DPTR,A

                MOV     R0,#08H         ;CLEAR INT-RAM (ALL)
                MOV     R2,#ENDINT-08H
RES4:   MOV     @R0,#0
                INC     R0
                DJNZ    R2,RES4
MAIN:           SETB    CSBIT           ;SET CS\
                CLR     SCLKBT          ;CLEAR SCLK

                MOV     ZERBUF+0,#0H
                MOV     ZERBUF+1,#0H
                MOV     WEIGHTB+0,#0H
                MOV     WEIGHTB+1,#0H
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                LCALL   LCDSET
CHAO:           LCALL   CLEAR
                LCALL   MAIN0
                LCALL   RANG
                LCALL   RANGSET
                LCALL   MAIN1
                LCALL   RANGPLAC
                LCALL   RANGLOOP
                LCALL   RANGVAL
                LCALL   DTOH
                LCALL   ZEROSET
                LCALL   CALCU
                LCALL   LCDCHECK
                LCALL   LCDOUT
                LCALL   LAY
                LCALL   KEY
                CJNE    R7,#40H,SCAN1
                LCALL   ZERO
                CALL    KEEP
                CALL    DELAY
                LCALL   KEY
SCAN1:          CJNE    R7,#43H,SCAN2
                CALL    TEST
SCAN2:          LCALL   KEEP
                LCALL   DELAY
                LJMP    CHAO
;************************TEST************************************

TEST:           CALL    KEEP
                CALL    DELAY
TESTPER:        CALL    KEY
                CJNE    A,#0FFH,TEST2
                SJMP    TESTPER

TEST2:          CJNE    R7,#42H,TEST3   ;CLEAR DISPLAY
                CALL    CLEARK
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                CALL    LCDOUTA
                CALL    KEEP
                CALL    DELAY
                SJMP    TESTPER

TEST3:          CJNE    R7,#41H,TEST4   ;ENTER OF PRICE
                CALL    MOVING
                CALL    ENTLOOP
                CALL    WEIGHTVAL
                CALL    DTOH
                CALL    MULTIPLY
                CALL    HTOD32
                CALL    CLEARMUL
                CALL    DIGITSET
                CALL    CONVEST
                CALL    CHECKZE
                CALL    LCDOUTMUL
                CALL    KEEP
                CALL    DELAY
                SJMP    TESTPER

TEST4:          CJNE    R7,#33H,TEST5       ;LOAD OLD PRICE
                CALL    LCDOUTFULL
                CALL    KEEP
                CALL    DELAY
                SJMP    TESTPER

TEST5:          CJNE    R7,#40H,TEST6       ;LOAD OLD PRICE
                CALL    KEEP
                CALL    DELAY
                SJMP    TESTPER

TEST6:          CJNE    R7,#43H,TEST7   ;RETURN   FUNCTION SET
                CALL    KEEP
                CALL    DELAY           ;DELAY TIME
                CALL    LCDSET
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                RET

;*************************************************************

TEST7:          MOV   DPTR,#TABLE
                MOV   DPL,A
                CALL  LCDHEX
                CALL  PLACE
                CALL  LCDOUTA
                CALL  KEEP
                CALL  DELAY       ;DELAY TIME
                LJMP  TESTPER
;*************************************************************
;REG=R0,R2

CLEARK:         MOV   R0,#LCDBUFA+11
                MOV   R2,#5
CLEARKA:        MOV   @R0,#20H
                INC   R0
                DJNZ  R2,CLEARKA
                RET
;*************************************************************
;REG=R2,R0,

MOVING:         MOV   CALBUF+0,#0
                MOV   CALBUF+1,#0
                MOV   CALBUF+2,#0
                MOV   CALBUF+3,#0
                MOV   CALBUF+4,#0
                MOV   DIGIT,#0
                MOV   R0,#LCDBUFA+15
                MOV   R2,#5
GOUP:           MOV   A,@R0
                INC   DIGIT
                CJNE  A,#2EH,GOON
                DEC   DIGIT
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                SJMP    ENT0
GOON:           DEC     R0
                DJNZ    R2,GOUP
                MOV     DIGIT,#0

ENT0:
                MOV     R0,#CALBUF+0
                MOV     A,LCDBUFA+15
                CJNE    A,#20H,ENT1
                RET
ENT1:           CJNE    A,#2EH,ENT11
                SJMP    ENTA
ENT11:          MOV     @R0,A
                INC     R0
ENTA:           MOV     A,LCDBUFA+14
                CJNE    A,#20H,ENT2
                RET
ENT2:           CJNE    A,#2EH,ENT12
                SJMP    ENTB
ENT12:          MOV     @R0,A
                INC     R0
ENTB:           MOV     A,LCDBUFA+13
                CJNE    A,#20H,ENT3
                RET
ENT3:           CJNE    A,#2EH,ENT13
                SJMP    ENTC
ENT13:          MOV     @R0,A
                INC     R0
ENTC:           MOV     A,LCDBUFA+12
                CJNE    A,#20H,ENT4
                RET
ENT4:           CJNE    A,#2EH,ENT14
                SJMP    ENTD
ENT14:          MOV     @R0,A
                INC     R0
ENTD:           MOV     A,LCDBUFA+11
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                    CJNE    A,#20H,ENT5
                    RET
        ENT5:       CJNE    A,#2EH,ENT15
                    SJMP    ENTF
        ENT15:      MOV     @R0,A
        ENTF:       NOP
                    RET

;**********************ENTLOOP*******************************
;R;EG=R2,R3
;OU;T=CALBUF+0,1,2

        ENTLOOP:
                    MOV     R3,CALBUF+0
                    MOV     R2,CALBUF+1
                    CALL    ATOH
                    MOV     CALBUF+0,A
                    MOV     R3,CALBUF+2
                    MOV     R2,CALBUF+3
                    CALL    ATOH
                    MOV     CALBUF+1,A
                    MOV     R3,CALBUF+4
                    MOV     R2,#0
                    CALL    ATOH
                    MOV     CALBUF+2,A
                    RET

; ********** ATOH SUB *********************************
; ;ASCII TO HEX CONVERT
; I;N  = R2,R3   30H,41H
; OU;T = A      0AH
; REG; = A,R2
;*****************************************************
        ATOH:       MOV   A,R2
                    CALL  ATOHS
                    SWAP  A
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                MOV   R2,A
                MOV   A,R3
                CALL  ATOHS
                ORL   A,R2
                RET

ATOHS:          CJNE  A,#'A',$+3
                JC    ATOHS1
                ADD   A,#9
ATOHS1:         ANL   A,#0FH
                RET


;**********************WEIGHTVAL**************************

WEIGHTVAL:      MOV     R3,CALBUF+0
                MOV     R2,CALBUF+1
                MOV     R1,CALBUF+2
                RET


; **********************DTOH SUB ****************************
; DECIMAL TO HEX
; IN  = R1,R2,R3
; OUT = DPTR
; REG = A,R0,R1,R2,R3,R4,R5,DPTR
;************************************************************

DTOH:           MOV   DPTR,#0
                MOV   R4,#16
DTOH1:          MOV   R5,#3         ;SHIFT & SUB
                MOV   R0,#1         ;INDEX TO R1
                CLR   C
DTOH2:          MOV   A,@R0
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                RRC     A
                PUSH    PSW             ;-[
                JNB     ACC.7,DTOH3
                CLR     C
                SUBB    A,#30H
DTOH3:          JNB     ACC.3,DTOH4
                CLR     C
                SUBB    A,#03H
DTOH4:          MOV     @R0,A
                INC     R0
                POP     PSW             ;-]
                DJNZ    R5,DTOH2
                MOV     A,DPH
                RRC     A
                MOV     DPH,A
                MOV     A,DPL
                RRC     A
                MOV     DPL,A
                DJNZ    R4,DTOH1
                RET
;*********************************************************
;IN=DPTR,WEIGHTB+0,1,
;OUT=R,1,2,3,4
;REG R0,R1,R2,R3,R4,R5,R6
;*********************************************************
MULTIPLY:       MOV     R1,#0
                MOV     R2,#0
                MOV     R3,#0
                MOV     R4,#0
                MOV     CALBUF+0,#0
                MOV     CALBUF+1,#0
                MOV     CALBUF+2,#0
                MOV     CALBUF+3,#0
                MOV     R5,#15
                MOV     R0,#2
                MOV     R6,#0
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
CLOSE:      CLR   C
            CLR   A
            MOV   A,DPL
            RLC   A
            MOV   DPL,A
            MOV   A,DPH
            RLC   A
            MOV   DPH,A
            JNC   CLOSELOOP
            MOV   CALBUF+3,NETWB+0    ;FD
            MOV   CALBUF+2,NETWB+1    ;20
            MOV   CALBUF+1,#0
            MOV   CALBUF+0,#0
            MOV   A,R5
            JZ    CLOSEUP
            MOV   R6,A
            CLR   C
BITLOOP:    MOV   A,CALBUF+3
            RLC   A
            MOV   CALBUF+3,A
            MOV   A,CALBUF+2
            RLC   A
            MOV   CALBUF+2,A
            MOV   A,CALBUF+1
            RLC   A
            MOV   CALBUF+1,A
            MOV   A,CALBUF+0
            RLC   A
            MOV   CALBUF+0,A
            DJNZ  R6,BITLOOP
CLOSEUP:    CALL  ADDVALUE
CLOSELOOP:  DJNZ  R5,CLOSE
            DJNZ  R0,CLOSE
            RET
;*********************************************
ADDVALUE:   CLR   C
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                MOV     A,R4
                ADDC    A,CALBUF+3
                MOV     R4,A
                MOV     A,R3
                ADDC    A,CALBUF+2
                MOV     R3,A
                MOV     A,R2
                ADDC    A,CALBUF+1
                MOV     R2,A
                MOV     A,R1
                ADDC    A,CALBUF+0
                MOV     R1,A
                RET
; ********** HTOD32 SUB **********************************
; HEX TO DECIMAL
; IN  = R,1,2,3,4
; OUT=CALBUF+0,1,2,3,4
; REG = A,R0,R1,R2,R3,R4,R5,DPTR
;******************************************************
        HTOD32:     CLR   A             ;CLEAR OUTPUT
                    MOV     CALBUF+0,A
                    MOV     CALBUF+1,A
                    MOV     CALBUF+2,A
                    MOV     CALBUF+3,A
                    MOV     CALBUF+4,A

                    MOV   R6,#32        ;SHIFT 16 BIT
        HTOD132:    MOV   A,R4
                    RLC   A
                    MOV   R4,A
                    MOV   A,R3
                    RLC   A
                    MOV   R3,A
                    MOV   A,R2
                    RLC   A
                    MOV   R2,A
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                MOV   A,R1
                RLC   A
                MOV   R1,A
                MOV   R5,#5              ;ADD DECIMAL
                MOV   R0,#CALBUF+4       ;INDEX TO R3
HTOD232:        MOV   A,@R0
                ADDC  A,ACC
                DA    A
                MOV   @R0,A
                DEC   R0
                DJNZ  R5,HTOD232
                DJNZ  R6,HTOD132
                RET
;*****************************************************
;IN=A
;OUT=R2,R3
;REG=R2,R3,R4,DPTR,R0
;*****************************************************

CONVEST:        MOV   DPTR,#REPLAYB+5
                MOV   R0,#CALBUF+0
                MOV   R5,#5
CON1:           MOV   A,@R0
                CALL  HTOA
                MOVX  A,@DPTR
                CJNE  A,#2EH,CON2
                INC   DPTR
CON2:           MOV   A,R2
                MOVX  @DPTR,A
                INC   DPTR
                MOVX  A,@DPTR
                CJNE  A,#2EH,CON3
                INC   DPTR
CON3:           MOV   A,R3
                MOVX  @DPTR,A
                INC   DPTR
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                INC     R0
                DJNZ    R5,CON1
                RET
;***************************************************************
        CHECKZE:    MOV     R2,#11
                    MOV     DPTR,#REPLAYB+5
        CHECKLOOP:  MOVX    A,@DPTR
                    CJNE    A,#30H,CHE
                    MOV     A,#20H
                    MOVX    @DPTR,A
                    INC     DPTR
                    DJNZ    R2,CHECKLOOP
        CHE:        CJNE    A,#2EH,CHETO
                    CALL    CLEARMUL
        CHETO:      RET
;***************************************************************
        DIGITSET:   ;INC    DIGIT
                    MOV     A,#15
                    CLR     C
                    SUBB    A,DIGIT
                    MOV     DIGIT,A
                    MOV     DPTR,#REPLAYB
        DISS:       INC     DPTR
                    DJNZ    DIGIT,DISS
                    MOV     A,#'.'
                    MOVX    @DPTR,A
                    RET

;***************************************************************
;USE REG ACC,R2,DPTR,B
;A=OFFH NO KEY
;***************************************************************
        KEY:
                    MOV     R2,#0           ;R2=LOW
                    MOV     R7,#0
        KEY1:       MOV     DPTR,#PORTC     ;PORTB=OUTPUT
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        MOV   A,R2          ;R2=B5-B4
        SWAP  A             ;PB=00**0000

        MOVX  @DPTR,A       ;SCAN0
        MOV   DPTR,#PORTC   ;PORTC=INPUT
        MOVX  A,@DPTR       ;INPUT KEY
        ANL   A,#0FH        ;PC=00000***
        CJNE  A,#0FH,KEY3   ;IF ACC.7=0 CLOSE KEY

KEY2:   INC   R2            ;SCAN=SCAN+1
        CJNE  R2,#4,KEY1    ;IF SCAN = 4
        MOV   A,#0FFH       ;NO KEY
        RET

KEY3:   ACALL DELAY         ;DELAY TIME 10 ms
        MOV   DPTR,#PORTC
        MOVX  A,@DPTR       ;INPUT KEY
        ANL   A,#0FH        ;A=00000***
        MOV   B,#0

KEY4:   INC   B             ;TABLE COLUM
        RRC   A
        JC    KEY4          ;JUMP IF CARRY FLAG=0
        MOV   A,B           ;A=1,2,3
        SWAP  A             ;A=00010000B
        ORL   A,R2
        MOV   R7,A
        RET

;*************************************************
;IN=PORTC
;OUT=A
;REG=A,DPTR,
;*************************************************
        KEEP:
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                MOV   DPTR,#PORTC
KEEP2:          MOVX  A,@DPTR          ;INPUT KEY
                ANL   A,#0FH
                CJNE  A,#0FH,KEEP2     ;IF KEY CLOSE
                RET
;*************************************************

DELAY:          MOV   R0,#10
DELAY1:         MOV   R1,#250
DE:             NOP
                NOP
                DJNZ  R1,DE
                DJNZ  R0,DELAY1
                RET
;*************************************************

LAY:            MOV   R0,#0FFH
LAY1:           MOV   R1,#250
LDE:            NOP
                NOP
                DJNZ  R1,LDE
                DJNZ  R0,LAY1
                RET

;*************************************************
;ZERSET
ZERO:           MOV   ZERBUF+0,WEIGHTB+0
                MOV   ZERBUF+1,WEIGHTB+1
                RET
;*************************************************
LCDHEX:         MOV   A,LCDBUFA+12
                MOV   LCDBUFA+11,A
                MOV   A,LCDBUFA+13
                MOV   LCDBUFA+12,A
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                MOV     A,LCDBUFA+14
                MOV     LCDBUFA+13,A
                MOV     A,LCDBUFA+15
                MOV     LCDBUFA+14,A
                RET

;*************************************************
        PLACE:
                MOVX    A,@DPTR
                MOV     LCDBUFA+15,A
                RET

; ********** LCDSET SUB ****************************
; LCD SETUP COMMAND
;*************************************************
        LCDSET:         MOV   A,#00111000B    ;FUNCTION SET
                        CALL  LCDWI
                        MOV   A,#00001110B    ;DISPLAY ON/OFF
                        CALL  LCDWI
                        MOV   A,#01H          ;CLEAR
                        CALL  LCDWI
                        RET


; ********** LCDOUT SUB ********************************
; LOAD LCDBUF TO LCD-MODULE (DMC202)
; IN  = LCDBUF
; REG = A,R0,R2
;*************************************************

        LCDOUT:         MOV   R0,#LCDBUFA+0   ;LCD BUFFER
                        MOV   A,#80H          ;SET ADDRESS LINE 1
                        CALL  LCDWI           ;LOAD ONE LINE
                        MOV   R2,#8           ;20 CHAR.
        LCDOUTS1:       MOV   A,@R0
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                CALL    LCDWD           ;WRITE DATA
                INC     R0
                DJNZ    R2,LCDOUTS1
                MOV     A,#0C0H
                CALL    LCDWI
                MOV     R2,#3
LCDOUTS2:       MOV     A,@R0
                CALL    LCDWD           ;WRITE DATA
                INC     R0
                DJNZ    R2,LCDOUTS2
                MOV     R0,#0
                RET

; ********** LCDOUT SUB ************************************
; LOAD LCDBUF TO LCD-MODULE (DMC202)
; IN  = LCDBUF
; REG = A,R0,R2
;************************************************************

LCDOUTA:        MOV     R0,#LCDBUFA+11  ;LCD BUFFER
                MOV     A,#0C3H         ;SET ADDRESS LINE 1
                CALL    LCDWI           ;LOAD ONE LINE
                MOV     R2,#5           ;20 CHAR.
LCDOUTA1:       MOV     A,@R0
                CALL    LCDWD           ;WRITE DATA
                INC     R0
                DJNZ    R2,LCDOUTA1
                RET
; ********** LCDOUTFULL SUB ********************************
; LOAD LCDBUF TO LCD-MODULE (DMC202)
; IN  = LCDBUF
; REG = A,R0,R2
;************************************************************

LCDOUTFULL:     MOV     R0,#LCDBUFA+0   ;LCD BUFFER
                MOV     A,#80H          ;SET ADDRESS LINE 1
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                    CALL   LCDOUTSF
                    MOV    A,#0C0H        ;SET ADDRESS LINE 2
                    CALL   LCDOUTSF
                    RET

LCDOUTSF:           CALL   LCDWI          ;LOAD ONE LINE
                    MOV    R2,#8          ;20 CHAR.
LCDOUTSF1:          MOV    A,@R0
                    CALL   LCDWD          ;WRITE DATA
                    INC    R0
                    DJNZ   R2,LCDOUTSF1
                    RET
; ********** LCDOUTCLEAR SUB ****************************
; LOAD LCDBUF TO LCD-MODULE (DMC202)
; IN  = LCDBUF
; REG = A,R0,R2
;*************************************************************

CLEARMUL:           MOV    DPTR,#REPLAYB+0     ;LCD BUFFER
                    MOV    R2,#15
                    MOV    A,#20H
CLEARLOOP:          MOVX   @DPTR,A
                    INC    DPTR
                    DJNZ   R2,CLEARLOOP
                    MOV    A,#'0'
                    MOVX   @DPTR,A
                    RET
;*****************REPLAY BUFFER ***************************

LCDOUTMUL:
                    MOV    DPTR,#REPLAYB       ;LCD BUFFER
                    MOV    A,#80H              ;SET ADDRESS LINE 1
                    CALL   LCDOUTSR
                    MOV    A,#0C0H
                    CALL   LCDOUTSR
                    RET
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
LCDOUTSR:    CALL  LCDWI          ;LOAD ONE LINE
             MOV   R2,#8          ;20 CHAR.
LCDOUTSR1:   MOVX  A,@DPTR
             CALL  LCDWD          ;WRITE DATA
             INC   DPTR
             DJNZ  R2,LCDOUTSR1
             RET


;*****************************************************
; ********** LCDWI SUB **********
; LCD WRITE INSTRUCTION (RS=0)
; IN  = A
; REG = A
;*****************************************************

        LCDWI:       PUSH  DPH
                     PUSH  DPL
                     MOV   DPTR,#XLCDWRC
                     MOVX  @DPTR,A
                     MOV   DPTR,#XLCDRDC
        LCDWI1:      MOVX  A,@DPTR        ;WAIT FOR BF=0
                     JB    ACC.7,LCDWI1
                     POP   DPL
                     POP   DPH
                     RET

; ********** LCDWD SUB **********************************
; LCD WRITE DATA (RS=1)
; IN  = A
; REG = A
;*****************************************************

        LCDWD:       PUSH  DPH
                     PUSH  DPL
                     MOV   DPTR,#XLCDWRD
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                MOVX  @DPTR,A
                MOV   DPTR,#XLCDRDC
LCDWD1:         MOVX  A,@DPTR        ;WAIT FOR BF=0
                JB    ACC.7,LCDWD1
                POP   DPL
                POP   DPH
                RET

; ********** CLEAR SUB **********
; CLEAR DISBUF

CLEAR:          MOV   R2,#11
                MOV   R0,#LCDBUFA
CLEAR1:         MOV   @R0,#0
                INC   R0
                DJNZ  R2,CLEAR1
                RET

;*************************************************************

MAIN0:
                MOV    WEIGHTB+0,#0
                MOV    WEIGHTB+1,#0
                MOV    NETWB+0,#0
                MOV    NETWB+1,#0
                MOV    LCDBUFA+0,#20H
                MOV    TB1BUF,#11001111B      ;SEL CH1
                CALL   TB1                    ;TRANSMIT TB1
                CALL   D10B                   ;10 BITS
                MOV    DPH,ADCBUF+0
                MOV    DPL,ADCBUF+1
                CALL   HADD
                RET
;*****************************************************

MAIN1:          MOV   TB1BUF,#10011111B ;SEL CH2
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                CALL  TB1               ;TRANSMIT TB1
                CALL  D10B              ;10 BITS
                MOV   DPH,ADCBUF+0
                MOV   DPL,ADCBUF+1
                RET


; ********** TB1 SUB **********
; TRANSMIT TB1 (1XXXXX11B)
; IN  = TB1BUF
; OUT = NONE
; REG = A,R2,R3

TB1:            CLR   CSBIT             ;CLEAR CS\
                MOV   R2,#8             ;8 SCLK
                MOV   A,TB1BUF
TB11:           RLC   A
                MOV   DINBIT,C          ;OUT TO DIN
                SETB  SCLKBT            ;SET SCLK (CLK = 0.1536 MHZ)
                MOV   R3,#1             ;COMPENSATE FOR DUTY CYCLE 45%-55%
                DJNZ  R3,$
                NOP
                CLR   SCLKBT            ;CLEAR SCLK
                DJNZ  R2,TB11
                RET


; ********** D10B SUB **********
; DATA 10 BITS (XXXXXXXXXXB)
; IN  = NONE
; OUT = ADCBUF+0 (000000XXB) HIGH BYTE
;       ADCBUF+1 (XXXXXXXXB) LOW BYTE
; REG = A,R4

D10B:           CALL  RB2X
                RL    A
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                    RL   A
                    MOV  R4,A
                    ANL  A,#00000011B
                    MOV  ADCBUF+0,A
                    CALL RB3X
                    RL   A
                    RL   A
                    ORL  A,#11111100B
                    ANL  A,R4
                    MOV  ADCBUF+1,A
                    RET

; ********** RB2X SUB **********
; RECEIVE BYTE RB2
; IN  = NONE
; OUT = A
; REG = A,R2,R3

RB2X:               MOV  R2,#8          ;8 SCLK
                    CLR  A
RB2X1:  SETB  SCLKBT      ;SET SCLK (CLK = 0.1536 MHZ)
                    MOV  R3,#1          ;COMPENSATE FOR DUTY CYCLE 45%-55%
                    DJNZ R3,$
                    NOP
                    CLR  SCLKBT         ;CLEAR SCLK
 SETB  DOUTBT       ;SET DOUTBT=INPUT
                    MOV  C,DOUTBT       ;DATA IN
                    RLC  A
                    DJNZ R2,RB2X1
                    RET

; ********** RB3X SUB **********
; RECEIVE BYTE RB3
; IN  = NONE
; OUT = A
; REG = A,R2,R3
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
RB3X:                MOV   R2,#8          ;8 SCLK
                     CLR   A
RB3X1:  SETB  SCLKBT        ;SET SCLK (CLK = 0.1536 MHZ)
                     MOV   R3,#1          ;COMPENSATE FOR DUTY CYCLE 45%-55%
                     DJNZ  R3,$
                     NOP
                     CLR   SCLKBT         ;CLEAR SCLK
SETB  DOUTBT         ;SET DOUTBT=INPUT
                     MOV   C,DOUTBT       ;DATA IN
                     RLC   A
                     DJNZ  R2,RB3X1
                     SETB  CSBIT          ;SET CS\
                     RET
;************ HADD***************
;BUFFER FOR WEIGHT
;IN  WEIGHT BUFF
;REG A
        HADD:  CLR    C
               MOV    A,DPL
               ADD    A,WEIGHTB+0
               MOV    WEIGHTB+0,A
               MOV    A,DPH
               ADDC   A,WEIGHTB+1
               MOV    WEIGHTB+1,A
               RET
;*************************************************
        RANG:          MOV    R2,#2
        RANG1:         CLR    C
                       MOV    A,DPH
                       RRC    A
                       MOV    DPH,A
                       MOV    A,DPL
                       RRC    A
                       MOV    DPL,A
                       DJNZ   R2,RANG1
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                MOV     A,DPL
                MOV     DPTR,#TABLE0
                MOV     DPL,A
                MOVX    A,@DPTR
                MOV     DPTR,#TABLE1
                MOV     DPL,A
                MOVX    A,@DPTR
                ANL     A,#0F0H
                SWAP    A
                MOV     R3,A
                MOVX    A,@DPTR
                ANL     A,#0FH
                MOV     R4,A
                MOV     DPTR,#TABLE3
                MOV     DPL,R3
                MOVX    A,@DPTR
                MOV     R3,A
                MOV     DPTR,#TABLE2
                MOV     DPL,R4
                MOVX    A,@DPTR
                MOV     R4,A
                MOV     DPTR,#PORTB
                MOV     A,R3
                MOVX    @DPTR,A
                MOV     DPTR,#PORTA
                MOV     A,R4
                MOVX    @DPTR,A

                RET
;********************RANGSET*********************
;WEIGHT TO DISPLAY
;REG DPTR A
        RANGSET:    MOV     DPL,WEIGHTB+0
                    MOV     DPH,WEIGHTB+1
                    CALL    HTOD            ;OUT R1,R2,R3 ,IN DPTR
                    MOV     HEXBUF,R3
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                MOV     A,R2
                CALL    HTOA            ;OUT R2,R3
                MOV     LCDBUFA+1,R2
                MOV     LCDBUFA+2,R3
                MOV     A,HEXBUF
                CALL    HTOA            ;OUT R2,R3
                MOV     LCDBUFA+3,R2
                MOV     LCDBUFA+4,R3
                RET
;********************RANGPLAC*********************
;WEIGHT TO DISPLAY
;REG DPTR A
        RANGPLAC:
                CALL    HTOD            ;OUT R1,R2,R3 ,IN DPTR
                MOV     A,R3
                CALL    HTOA            ;OUT R2,R3
                MOV     LCDBUFA+5,R3
                RET
;*****************************************************
;R;EG=R2,R3
;OU;T=CALBUF+0,1,2

        RANGLOOP:
                MOV     R3,LCDBUFA+5
                MOV     R2,LCDBUFA+4
                CALL    ATOH
                MOV     LCDBUFA+5,A
                MOV     R3,LCDBUFA+3
                MOV     R2,LCDBUFA+2
                CALL    ATOH
                MOV     LCDBUFA+4,A
                MOV     R3,LCDBUFA+1
                MOV     R2,#0
                CALL    ATOH
                MOV     LCDBUFA+3,A
                RET
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
;*****************************************************

        RANGVAL:    MOV     R3,LCDBUFA+5
                    MOV     R2,LCDBUFA+4
                    MOV     R1,LCDBUFA+3
                    RET

;********************CALCU*********************
;WEIGHT TO DISPLAY
;REG DPTR A
            CALCU:
                    CALL    HTOD            ;OUT R1,R2,R3 ,IN DPTR
                    MOV     A,R2
                    MOV     R4,A
                    MOV     A,R3
                    MOV     R5,A
                    MOV     A,R1
                    CALL    HTOA            ;OUT R2,R3
                    MOV     LCDBUFA+1,R2
                    MOV     LCDBUFA+2,R3
                    MOV     A,R4
                    CALL    HTOA            ;OUT R2,R3
                    MOV     LCDBUFA+3,R2
                    MOV     LCDBUFA+4,R3
                    MOV     A,R5
                    CALL    HTOA
                    MOV     LCDBUFA+5,R2
                    MOV     LCDBUFA+6,#'.'
                    MOV     LCDBUFA+7,R3
                    MOV     LCDBUFA+8,#' '
                    MOV     LCDBUFA+9,#'g'
                    MOV     LCDBUFA+10,#' '
                    RET
;*****************************************
        ZEROSET:    MOV     WEIGHTB+0,DPL
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        MOV     WEIGHTB+1,DPH
        CLR     C
        MOV     A,WEIGHTB+0
        SUBB    A,ZERBUF+0
        MOV     A,WEIGHTB+1
        SUBB    A,ZERBUF+1
        JNC     ZER
        CLR     C
        MOV     A,ZERBUF+0
        SUBB    A,WEIGHTB+0
        MOV     DPL,A
        MOV     NETWB+0,A
        MOV     A,ZERBUF+1
        SUBB    A,WEIGHTB+1
        MOV     DPH,A
        MOV     NETWB+1,A
        MOV     LCDBUFA+0,#'-'
        SJMP    ZERA
ZER:    CLR     C
        MOV     A,WEIGHTB+0
        SUBB    A,ZERBUF+0
        MOV     DPL,A
        MOV     NETWB+0,A
        MOV     A,WEIGHTB+1
        SUBB    A,ZERBUF+1
        MOV     DPH,A
        MOV     NETWB+1,A
ZERA:   NOP
        RET


; ********** HTOD SUB **********
; HEX TO DECIMAL
; IN  = DPTR
; OUT = R1,R2,R3
; REG = A,R0,R1,R2,R3,R4,R5,DPTR
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
HTOD:       CLR   A          ;CLEAR OUTPUT
            MOV   R1,A
            MOV   R2,A
            MOV   R3,A

            MOV   R4,#16     ;SHIFT 16 BIT
HTOD1:      MOV   A,DPL
            RLC   A
            MOV   DPL,A
            MOV   A,DPH
            RLC   A
            MOV   DPH,A

            MOV   R5,#3      ;ADD DECIMAL
            MOV   R0,#3      ;INDEX TO R3
HTOD2:      MOV   A,@R0
            ADDC  A,ACC
            DA    A
            MOV   @R0,A
            DEC   R0
            DJNZ  R5,HTOD2
            DJNZ  R4,HTOD1
            RET

; ********** HTOA SUB **********
; CONVERT DECIMAL TO ASCII
; IN  = A
; OUT = R2,R3
; REG = A,R2,R3

HTOA:       PUSH  ACC
            SWAP  A
            CALL  HTOAS
            MOV   R2,A
            POP   ACC
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                    CALL  HTOAS
                    MOV   R3,A
                    RET

HTOAS:              ANL   A,#0FH
                    CJNE  A,#0AH,$+3
                    JNC   HTOAS1
                    ORL   A,#30H
                    RET
HTOAS1:             SUBB  A,#9
                    ORL   A,#40H
                    RET

;************************* LCDCHECK ********************************
        LCDCHECK:   MOV    R0,#LCDBUFA+1
                    MOV    R2,#5
        LCDCH1:             MOV    A,@R0
                    CJNE   A,#30H,LCDCH2
                    MOV    @R0,#20H
                    INC    R0
                    DJNZ   R2,LCDCH1
                    MOV    LCDBUFA+5,#30H
        LCDCH2:             RET


;%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
                    ORG    8710H
        TABLE:      DB "3","6","9","."
                    ORG    8720H
                    DB "2","5","8","0"
                    ORG    8730H
                    DB "1","4","7"

;************************************************

                    ORG    8900H
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

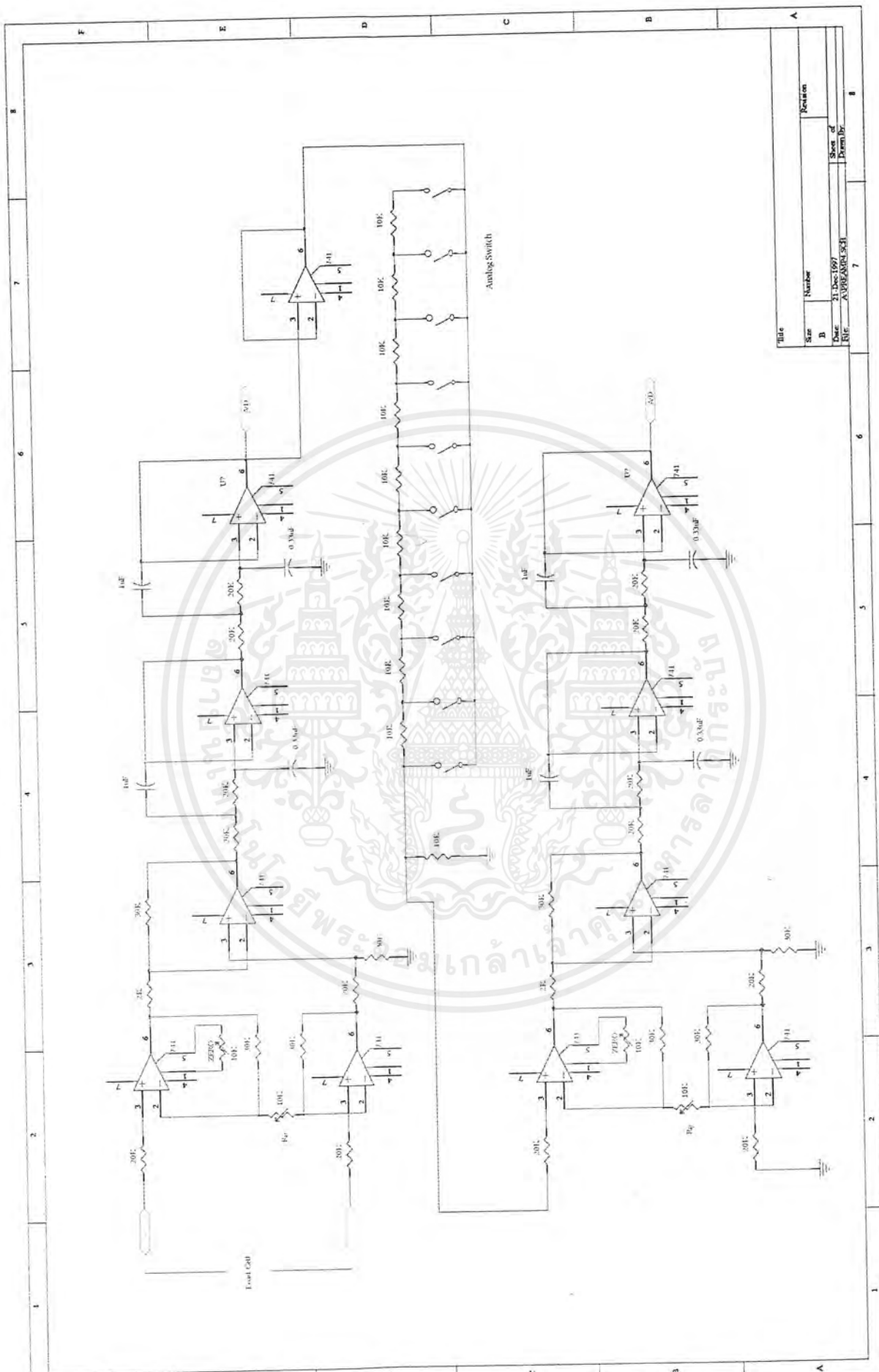
```
TABLE0:     DB    0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0
            DB    1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1
            DB    2,2,2,2,2,2,2,2,2,2,2,2,2,2,2,2,2,2,2,2,2,2,2,2,2
            DB    3,3,3,3,3,3,3,3,3,3,3,3,3,3,3,3,3,3,3,3,3,3,3,3,3
            DB    4,4,4,4,4,4,4,4,4,4,4,4,4,4,4,4,4,4,4,4,4,4,4,4,4
            DB    5,5,5,5,5,5,5,5,5,5,5,5,5,5,5,5,5,5,5,5,5,5,5,5,5
            DB    6,6,6,6,6,6,6,6,6,6,6,6,6,6,6,6,6,6,6,6,6,6,6,6,6
            DB    7,7,7,7,7,7,7,7,7,7,7,7,7,7,7,7,7,7,7,7,7,7,7,7,7
            DB    8,8,8,8,8,8,8,8,8,8,8,8,8,8,8,8,8,8,8,8,8,8,8,8,8
            DB    9,9,9,9,9,9,9,9,9,9,9,9,9,9,9,9,9,9,9,9,9,9,9,9,9
            DB    9,9,9,9,9,9,9

;*************************************************
            ORG   8A00H
TABLE1:     DB    01H,02H,03H,04H,05H,06H,07H,08H,10H,20H

;*************************************************
            ORG   8B00H
TABLE2:     DB    00H,01H,02H,04H,08H,10H,20H,40H,80H
;*************************************************
            ORG   8C00H
TABLE3:     DB    00H,01H,02H
;*************************************************
            END
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การทำงานของวงจร Pre-Amplifier ประกอบด้วย 2 ส่วน

1.ส่วนขยายผลต่างของแรงดันจาก Load cell ซึ่งมีย่านขยายแรงดันเป็น 0-20 mv มีอัตราขยายประมาณ 200 เท่า อัตราขยายสามารถปรับได้จาก Rg ซึ่งมีค่าประมาณ 10 กิโลโอห์ม เมื่อลดค่า Rg อัตราขยายจะเพิ่มขึ้น ในทางกลับกันเมื่อเพิ่มค่า Rg อัตราขยายจะลดลง ตามรายละเอียดดังนี้ เมื่อ Load cell ได้รับโหลด ทำให้เกิดการเปลี่ยนแปลงของระดับแรงดันไฟชุดของ Op-Amp จะทำการขยายในลักษณะการขยายผลต่าง มาผ่านชุดกรองสัญญาณ ส่งผ่านมายังชุด Analog switch แรงดันที่ผ่าน Analog switch แล้วจะมาทำการขยายอีกครั้ง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.การทำงานของ Auto Rang

เมื่อทำการชั่ง ค่าของน้ำหนักเป็นตัวเลือก Rang การวัดตามความเหมาะสมของน้ำหนักและจะสัมพันธ์กับแรงดันด้วย เช่น ที่น้ำหนัก 100 g ก็จะได้แรงดันประมาณ 0.4 Volts เมื่อทำการแบ่ง Rang ก็จะได้ 10 Rang พอดี ในการเลือก Rang จะกระทำด้วยคอนโทลเลอร์ โดยที่น้ำหนักตามค่าตามตาราง ซึ่งจะมีค่าคงที่ แต่ก่อนที่จะนำค่าของน้ำหนักมาเปิดตารางเพื่อความเหมาะสมและประหยัดหน่วยความจำ เราจึงทำการหารค่าน้ำหนักด้วย 4 ก็จะเหลือค่าน้ำหนักจริงเพียง 0-256 ค่า เพื่อใช้ในการเปิดตาราง ที่แบ่งออกเป็น 10 ช่วง คือ 0-9 แล้วนำค่านี้มาเปิดตาราง เพื่อทำการเลือกค่าออกเป็น PORT A และ PORT B เลือกจาก ตาราง 3 ตามความสัมพันธ์ในตาราง

ระดับแรงดัน	Switch Number									
0 - 0.4	B1	B2	A7	A6	A5	A4	A3	A2	A1	A0
0.4- 0.8	L	L	L	L	L	L	L	L	L	H
0.8- 1.2	L	L	L	L	L	L	L	L	H	L
1.2- 1.6	L	L	L	L	L	L	L	H	L	L
1.6- 2.0	L	L	L	L	L	L	H	L	L	L
2.0- 2.4	L	L	L	L	L	H	L	L	L	L
2.4- 2.8	L	L	L	L	H	L	L	L	L	L
2.8- 3.2	L	L	L	H	L	L	L	L	L	L
3.2- 3.6	L	L	H	L	L	L	L	L	L	L
3.6- 4.0	L	H	L	L	L	L	L	L	L	L
4.0- 4.4	H	L	L	L	L	L	L	L	L	L

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

วิธีการใช้งานเครื่องชั่งน้ำหนักแบบละเอียด

SET กด SET จะเข้าโปรแกรมการคำนวณราคา และการ IN PUT ค่าเข้าไปโดยจะทำการ SCAN KEY BOARD ตลอดเวลา ถ้ากด SET อีกครั้งจะออกจากโปรแกรมจากคำนวณ

CLR ทำการเคลียร์ DISPLAY เฉพาะส่วน IN PUT ของค่ที่กดจาก KET BOARD และเคลียร์ค่าน้ำหนักเดิมที่ใส่ไว้ครั้งก่อน

ENT ทำการคูณค่าที่ปรากฏตามความจริงที่อยู่บน DISPLAY โดยโปรแกรมจะทำการตรวจสอบค่าใน LCD BUFFER ที่เก็บไว้ใน INTERNAL RAM โดยที่ความสามารถในการคูณตัวเลขขนาด 16 บิต ทั้งตัวตั้งและตัวคูณโดยจะไม่สนใจเครื่องหมาย โดยผลลัพย์สูงสุดไม่เกิน 32 บิต

ZERO จะทำหน้าที่ ในการปรับ ZERO ให้กับน้ำหนักโดยจะทำการ MOVค่าน้ำหนักจริงเข้ามาเก็บไว้ แล้วจะทำการลบกับค่าน้ำหนักนั้น ทุกๆ ครั้งที่ทำการกด ZERO โดยที่สามารถกด KEY นี้ได้ตลอดเวลาขณะที่ทำการชั่งอยู่

REPLAY จะทำการดึงค่าเก่าออกมาทำการแสดง โดยดึงค่าน้ำหนักที่ทำการ Memmory เข้าไป

<u>ความสามารถในการคำนวณ</u>

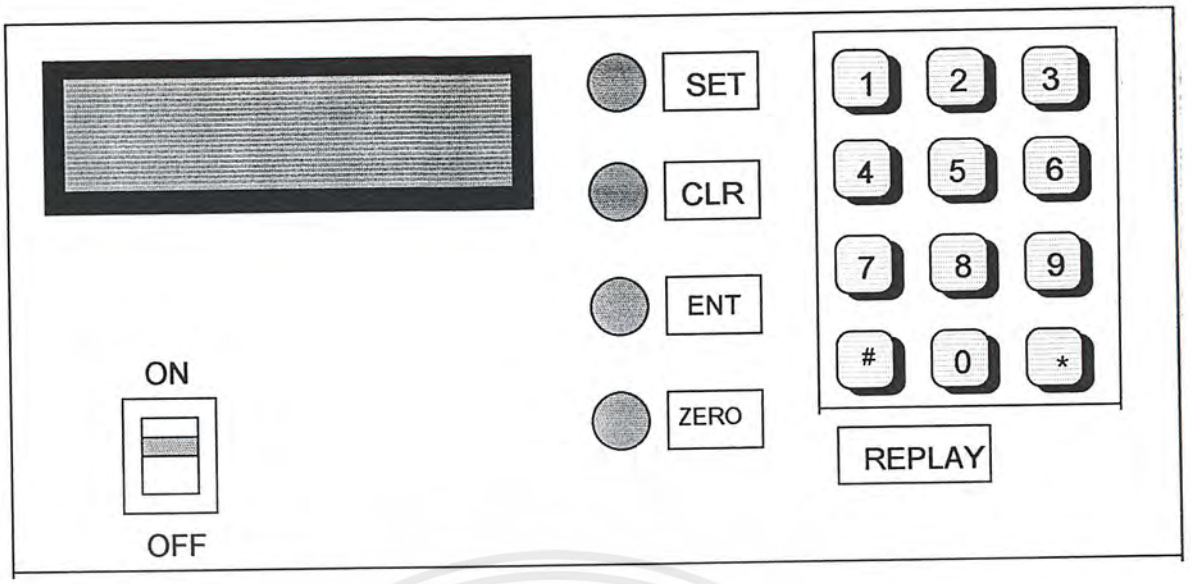
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สามารถปรับค่าอินพุตเข้ามา 65535 หรือ FFFF H ของเลขฐาน 16 และปรับค่าที่เป็นเลขทศนิยมที่ไม่จำกัด ซึ่งคำนวณให้เองโดยอัตโนมัติซึ่งได้กำหนดพื้นที่การรับค่าจาก key board ไว้ 5 ตำแหน่ง ซึ่งสามารถ อินพุต ค่าได้ไม่เกินค่านี้ คือ 65535 เมื่อค่าที่เกินจากนี้ จะทำให้เกิดการผิดพลาด ของค่าที่ได้คำนวณ

การแสดงค่าน้ำหนัก

จะทำการ RUN ค่า ACD แต่ละช่องมาทำการบวกกัน ซึ่งแต่ละช่องสามารถให้ค่าสูงสุด 10 บิตหรือ 1024 ค่า แต่ละช่องของ ACD มีแรงดันอ้างอิงที่ 0-4.096 V หรือ 4 mA ต่อ 1 ค่า เมื่อนำแต่ละช่องมาร่วมกันก็จะได้ค่าสูงสุด 1028 * 8 =8192 ค่า เมื่อนำมาทำเป็นทศนิยมก็จะได้ความละเอียดเท่ากับ 0.1 หรือ 819.2 มีหน่วยเป็นกรัมของค่าของน้ำหนัก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กิตติกรรมประกาศ

โครงงานปริญญานิพนธ์นี้สำเร็จได้โดยดีจากความร่วมมือและคำปรึกษาจากท่านอาจารย์ วิริยะ กองรัตน์ และอุปกรณ์เครื่องมือต่างๆ จากภาควิชาเครื่องมือวัดและควบคุมอุตสาหกรรม รวมทั้งอุปกรณ์ Load cell จาก บริษัท อินโทรเอ็น เตอร์ไพรส์ จำกัด และที่ขาดเสียมิได้คือท่านอาจารย์ทุกท่านในภาคและเพื่อน ๆ ที่ได้แนะนำและเป็นที่ปรึกษา

ทุกท่านที่ได้กล่าวมานั้นล้วนแต่ได้ส่งเสริมให้โครงงานนี้สำเร็จลงได้ ทางผู้จัดทำโครงงานต้องขอขอบพระคุณเป็นอย่างสูงและหวังว่าโครงงานนี้จะได้เป็นประโยชน์ต่อไป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บรรณานุกรม

1. ไมโครโปรเซสเซอร์และไมโครคอมพิวเตอร์/รศ.ยืน ภู่วรวรรณ ภาควิชาวิศวกรรมไฟฟ้า คณะวิทยาศาสตร์ ม.เกษตร/สำนักพิมพ์ซีเอ็ด
2. ELECTRONIC DEVICE & CIRCUIT / Robert Boylestad / Louis Naselsky / PRENTICE-HALL INTERNATIONAL EDITION
3. การทดลองไมโครโปรเซสเซอร์/ผศ.พิพัฒน์เลาหสงคราม
4. การใช้งาน Z80 / หจก.สำนักพิมพ์ฟิสิกส์เซ็นเตอร์/อ.นรินทร์ เนาวประทีป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้