

แขนกลจับวัตถุ

ROBOT ARM



จัดทำโดย

นางสาว ณัฐสุดา วงศ์สิทธิพิศาล

นางสาว ทิพวรรณ แก้วสังข์

เลขที่.....
เลขทะเบียน.....42481
วัน, เดือน, ปี.....23 พ.ค. 2545

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

ปฏิญานีพนธ์ฉบับนี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต
สาขาวิชาวิศวกรรมควบคุม
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
ปีการศึกษา 2543

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ปริญญาานิพนธ์ปีการศึกษา 2543

สาขาวิชาวิศวกรรมระบบควบคุม

คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เรื่อง แขนกลจับวัตถุ

ROBOT ARM

ผู้จัดทำ 1. นางสาวณัฐสุดา วงศ์สิทธิพิศาล

2. นางสาวทิพวรรณ แก้วสังข์

นาง เมธนา สุทธิ

อาจารย์ที่ปรึกษา

(อาจารย์ถาวร เภยจนราสุทธิ์)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

แขนกลจับวัตถุ

นางสาว ณัฐสุดา วงศ์สิทธิพิศาล

นางสาว ทิพวรรณ แก้วสังข์

อ. ถาวร เภญจนราษฎร์ อาจารย์ที่ปรึกษา

ปีการศึกษา 2543

บทคัดย่อ

ในปัจจุบันนี้ความสนใจทางด้าน Robot Vision และการประยุกต์ใช้งานมีเพิ่มขึ้นอย่างมาก แขนกลจับวัตถุก็ถือเป็นการประยุกต์ใช้งาน Robot Vision ประเภทหนึ่ง ซึ่งเสมือนเป็นการทำให้ แขนกลสามารถทราบตำแหน่งวัตถุ แล้วเคลื่อนที่ไปยังตำแหน่งของวัตถุ ประโยชน์ที่พบนี้นำเสนอ การจับภาพวัตถุด้วยกล้องดิจิทัลวิดีโอ(digital video camera) ซึ่งต่อเข้ากับพอร์ต USB ของคอมพิวเตอร์ที่ทำการประมวลผลหาตำแหน่งจุดศูนย์กลางของวัตถุ(centroid)โดยใช้โปรแกรม microsoft visual basic 6 แล้วส่งค่าไปยังไมโครคอนโทรลเลอร์ MCS-51เพื่อควบคุมสเต็ปเปอร์มอเตอร์ให้บังคับการเคลื่อนที่ของแขนกลที่ประดิษฐ์ขึ้นให้เคลื่อนไปยังตำแหน่งวัตถุที่ต้องการจับ

Abstract

Recently, the interest in the Robot Vision and its applications are increased. The Robot Arm to pick the object is one part of Robot Vision applications, liken to the Robot Arm can know the position of the object and then move to that object. This thesis is presents about the Robot Arm. Digital Video Camera which is connected to the USB port of computer captures the picture of the object and sends it to microsoft visual basic program. This program process the picture of the object to be the centroid value and then send this value to microcontroller MCS-51 in order to rotate the Robot Arm to the position of the object by controlling the stepper motors.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ

หน้า

บทคัดย่อ	
สารบัญ	
สารบัญรูปภาพ	
สารบัญตาราง	
บทที่ 1 บทนำ	1
บทที่ 2 ทฤษฎีและหลักการ	2
2.1 ทฤษฎีและหลักการเบื้องต้น	2
2.1.1 ความหมายและนิยามของภาพในระบบดิจิทัล	2
2.1.2 หลักการเบื้องต้นของการประมวลผลภาพ	2
2.1.3 ลักษณะข้อมูลภาพ	4
2.2 สเต็ปเปอร์มอเตอร์	4
2.2.1 ชนิดของสเต็ปเปอร์มอเตอร์	4
2.2.2 การทำงานของสเต็ปเปอร์มอเตอร์	5
2.2.3 การควบคุมการหมุนของสเต็ปเปอร์มอเตอร์	6
2.2.4 วงจรขับสเต็ปเปอร์มอเตอร์	7
2.3 ไมโครคอนโทรลเลอร์ตระกูล MCS-51	8
2.3.1 คุณสมบัติของไมโครคอนโทรลเลอร์ตระกูล MCS-51	8
2.3.2 สถาปัตยกรรมของ 8051	9
2.3.3 พอร์ตของ MCS-51	10
2.3.4 การรับส่งข้อมูลแบบอนุกรม(UART) ใน MCS-51	14
2.3.5 ชุดคำสั่งของ MCS-51	15
2.4 วงจรรับส่งข้อมูลแบบอนุกรมมาตรฐาน RS-232	15
บทที่ 3 หลักการออกแบบ	17
3.1 ส่วนของซอฟต์แวร์	17
3.1.1 โปรแกรมประมวลผลภาพดิจิทัล	17
3.1.2 โปรแกรมของไมโครคอนโทรลเลอร์ MCS-51	28
3.2 ส่วนของฮาร์ดแวร์	29

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.2.1 สเต็ปเปอร์มอเตอร์	29
3.2.2 ลิ้มิตสวิตช์	30
3.2.3 วงจรไมโครคอนโทรลเลอร์ MCS-51	31
3.2.4 วงจรขับสเต็ปเปอร์มอเตอร์	32
3.2.5 วงจรรับส่งข้อมูลแบบอนุกรม	33
3.2.6 วงจรรักษาระดับแรงดัน	33
3.2.7 กล้องดิจิตอลวิดีโอ	34
3.2.8 การออกแบบแขนกล ฐาน และขาตั้งกล้อง	35
บทที่ 4 การทดลองและผลการทดลอง	36
บทที่ 5 บทวิจารณ์และสรุป	44
ภาคผนวก	
กิตติกรรมประกาศ	
หนังสืออ้างอิง	



สารบัญรูปภาพ

	หน้า
รูปที่ 2.1 แสดงการแปลงข้อมูลภาพแบบดิจิทัลเฉดสี	2
2.2 แสดงระบบประมวลผลภาพดิจิทัล	3
2.3 ลักษณะการพันขดลวดบนสเตเตอร์	5
2.4 ทิศทางการหมุนโรเตอร์ของสเต็ปเปอร์มอเตอร์ 4 เฟส	5
2.5 บล็อกไดอะแกรมควบคุมด้วยสเต็ปเปอร์มอเตอร์	7
2.6 การขับสเต็ปเปอร์มอเตอร์แบบยูนิโพลาร์	7
2.7 บล็อกไดอะแกรมแสดงสถาปัตยกรรมของ MCS-51	10
2.8 การจัดวางขาของ MCS-51	10
2.9 แสดงโครงสร้างพอร์ต 0	11
2.10 แสดงโครงสร้างพอร์ต 1	11
2.11 แสดงโครงสร้างพอร์ต 2	12
2.12 แสดงโครงสร้างพอร์ต 3	12
2.13 แสดงการต่อขารีเซตให้กับ MCS-51	13
2.14 แสดงวงจรบล็อกของ MCS-51 แบบใช้บล็อกภายใน	14
2.15 แสดงคอนเนคเตอร์แบบ DB9	16
3.1 แสดงบล็อกไดอะแกรมของการทำงาน	17
3.2 แสดงแบบจำลองของแกนกลในระนาบ x-y	18
3.3 แสดงแบบจำลองของแกนกลในระนาบ x-y	18
3.4 แสดงแบบจำลองของแกนกลในระนาบ x-y	19
3.5 ไฟล์ชาร์ตของโปรแกรมการประมวลผลภาพของวัตถุสีดำ	22
3.6 ไฟล์ชาร์ตของโปรแกรมการประมวลผลภาพของวัตถุโทนสีส้ม	26
3.7 แสดงฟอร์มของส่วนควบคุม(Control Form)ของโปรแกรมประมวลผลภาพดิจิทัล	27
3.8 ไฟล์ชาร์ตของโปรแกรมไมโครคอนโทรลเลอร์ MCS-51 ควบคุมแกนกล	28
3.9 แสดงสเต็ปเปอร์ตัวล่าง	29
3.10 แสดงสเต็ปเปอร์ตัวบน	30
3.11 แสดงการติดตั้งลิมิตสวิทช์	30
3.12 แสดงวงจรไมโครโปรเซสเซอร์ MCS-51	31
3.13 แสดงวงจรขับสเต็ปเปอร์มอเตอร์	32

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.14 แสดงวงจรรับส่งข้อมูลแบบอนุกรม	33
3.15 แสดงวงจรปรับขนาดแรงดัน	34
3.16 แสดงกล้องดิจิทัลวิดีโอ	34
3.17 แสดงแขนกล ฐาน และขาตั้งกล้อง	35
4.1 Control Form แสดงผลการทดลองขั้นที่ 1	36
4.2 Control Form แสดงผลการทดลองขั้นที่ 2	37
4.3 Control Form แสดงผลการทดลองขั้นที่ 3	38
4.4(ก) Control Form แสดงผลการทดลองขั้นที่ 4	40
4.4(ข) Control Form แสดงผลการทดลองขั้นที่ 4	41
4.4(ค) Control Form แสดงผลการทดลองขั้นที่ 4	42
4.4(ง) Control Form แสดงผลการทดลองขั้นที่ 4	43



สารบัญตาราง

	หน้า
ตารางที่ 2.1 แสดงการกระตุ้นขดลวดแต่ละเฟสแบบสองเฟส	6
2.2 แสดงรายละเอียดการต่อคอนเนคเตอร์แบบ DB9 ตามมาตรฐาน RS-232	16



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 1

บทนำ

จากที่คณะผู้จัดทำได้ศึกษาด้าน โรโบติก(Robotic) ซึ่งแบ่งย่อยออกเป็นหลายสาขา แต่มีสาขาหนึ่งที่น่าสนใจอย่างยิ่งคือ โรบอทวิชัน(Robot Vision) ซึ่งเป็นการทำให้หุ่นยนต์สามารถมองเห็นภาพและนำภาพที่มองเห็นไปคิดและประมวลผล แล้วแสดงการทำงานต่างๆได้ เช่นการหยิบวัตถุจากพื้น การติดตามการเคลื่อนไหว ของขีปนาวุธ

คณะผู้จัดทำจึงคิดที่จะทำโครงการที่เกี่ยวกับการทำให้แขนกลสามารถหยิบวัตถุบนพื้นโดยใช้กล้องดิจิทัลวิดีโอ ในการจับภาพวัตถุ และใช้คอมพิวเตอร์ในการประมวลผลหาตำแหน่งของวัตถุ แล้วนำผลที่ได้ไปทำการควบคุมแขนกลให้สามารถหยิบวัตถุในตำแหน่งที่ต้องการ

โครงการนี้จะต้องอาศัยความรู้เกี่ยวกับการประมวลผลภาพ(image processing) สำหรับใช้วิเคราะห์ภาพที่ได้จากกล้องวิดีโอ เพื่อให้คอมพิวเตอร์หรือ แขนกลรู้ว่าวัตถุอยู่ที่ตำแหน่งใด และความรู้เกี่ยวกับการควบคุมมอเตอร์ สำหรับใช้ในการควบคุมการเคลื่อนที่ของแขนกล

ส่วนประกอบของโครงการแบ่งเป็นส่วนต่าง ๆ ดังนี้

- 1.1 กล้องวิดีโอสำหรับจับภาพวัตถุ และบันทึกเป็นภาพดิจิทัลลงบนคอมพิวเตอร์ โดยกล้องวิดีโอเป็นชนิดที่สามารถต่อกับพอร์ต USB ของคอมพิวเตอร์ เพื่อดึงภาพมาใช้ในการประมวลผลในคอมพิวเตอร์ได้โดยตรง
- 1.2 โปรแกรมหาตำแหน่งของวัตถุ โดยใช้โปรแกรม Microsoft Visual Basic 6 ทำการวิเคราะห์ภาพเพื่อให้ได้ตำแหน่งจุดศูนย์กลาง (Centroid) ของวัตถุ เพื่อคำนวณและส่งค่าไปควบคุมแขนกล
- 1.3 สเต็ปเปอร์มอเตอร์ และวงจรขับสเต็ปเปอร์มอเตอร์จำนวน 2 ชุด เพื่อให้แขนกลเคลื่อนที่ไปตามตำแหน่งที่ต้องการในระนาบ 2 มิติ โดยอาศัยไมโครคอนโทรลเลอร์ MCS-51 ในการควบคุมการเคลื่อนที่
- 1.4 โปรแกรมควบคุมแขนกลสำหรับไมโครคอนโทรลเลอร์ MCS-51 ด้วยภาษาแอสเซมบลี
- 1.5 แขนกลแบบ 2 มิติ เคลื่อนที่ตามแนวแกน X-Y ไปยังตำแหน่งของวัตถุ

บทที่ 2

ทฤษฎีและหลักการ

2.1 ทฤษฎีและหลักการเบื้องต้นเกี่ยวกับการประมวลผลภาพดิจิทัล

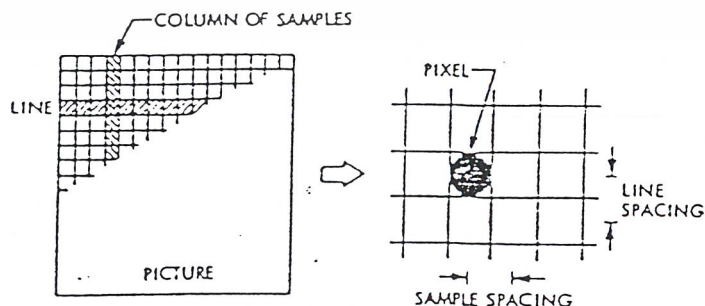
2.1.1 ความหมายและนิยามของภาพในระบบดิจิทัล

ภาพ (image) ในเชิงคณิตศาสตร์จะหมายถึง ฟังก์ชัน 2 มิติ $f(x,y)$ โดย x และ y เป็นแกนพิกัดในระนาบ 2 มิติ ค่าฟังก์ชัน $f(x,y)$ จะเป็นสัดส่วนกับความสว่างหรือความเข้มของภาพที่ตำแหน่ง (x,y) ซึ่งเราเรียกว่า ระดับสีเทา (gray level)

2.1.2 หลักการเบื้องต้นของการประมวลผลภาพ

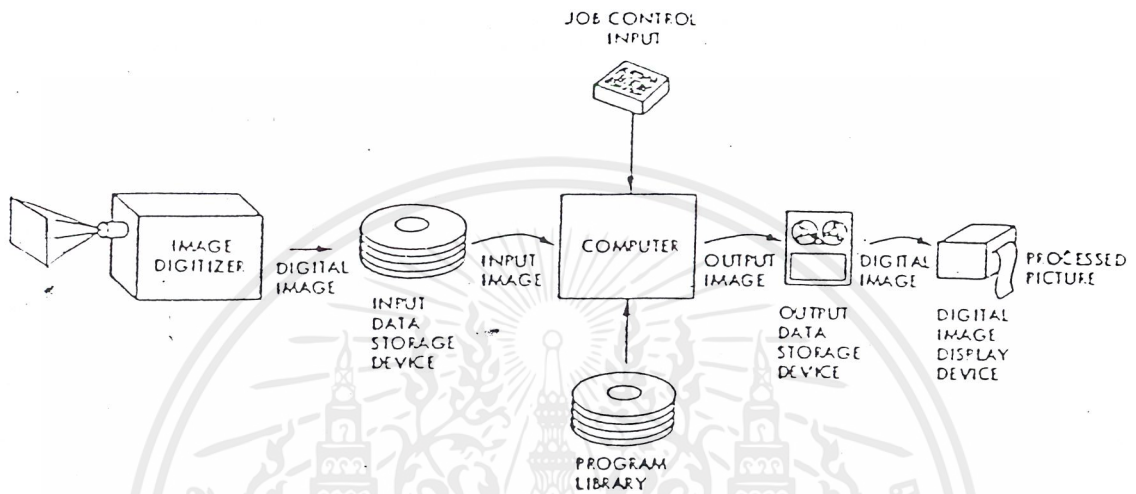
ในการประมวลผลภาพดิจิทัล (digital image processing) ต้องใช้อุปกรณ์พื้นฐาน 3 ชนิด คือ (1) หน่วยประมวลผล ได้แก่ คอมพิวเตอร์ (2) อุปกรณ์อินพุท (input device) ได้แก่ อิมเมจดิจิไทเซอร์ (image digitizer) และ (3) อุปกรณ์เอาต์พุท (output device) ได้แก่ อุปกรณ์แสดงผลภาพ

โดยทั่วไปนั้นข้อมูลภาพยังไม่สามารถนำไปวิเคราะห์ด้วยคอมพิวเตอร์ได้ทันทีเนื่องจากคอมพิวเตอร์ทำงานเกี่ยวกับระบบตัวเลข (digital) จึงต้องแปลงข้อมูลภาพให้อยู่ในรูปแบบของตัวเลขก่อน จึงเรียกรูปการแปลงนี้ว่า ดิจิไทเซชัน (digitizer) แสดงดังรูปที่ 2.1 โดยภาพจะถูกแบ่งเป็นส่วนเล็กๆ เรียกว่า พิกเซล (pixel) หรือ พิกเจอร์อีเลเมนต์ (picture element) ซึ่งมีลักษณะเป็นตารางสี่เหลี่ยมจัตุรัส โดยแต่ละพิกเซลจะมีค่าระดับความสว่างหรือมืดแตกต่างกันไป



รูปที่ 2.1 แสดงการแปลงข้อมูลภาพแบบดิจิไทเซชัน

ระบบการประมวลผลภาพประกอบด้วย 3 ส่วนใหญ่ๆ คือ ส่วนเปลี่ยนสัญญาณ อนาคตอกให้เป็นสัญญาณทางด้านอิิจิตอลซึ่งเรียกว่าดิจิไตเซอร์(digitizer) ส่วนประมวลผล(Processing) และส่วนแสดงผล (display) แสดงในรูปที่ 2.2



รูปที่ 2.2 ระบบประมวลผลภาพดิจิตอล

จากรูปที่ 2.2 ส่วนแรก คือส่วนรับภาพและเปลี่ยนสัญญาณอนาคตอกให้เป็นสัญญาณดิจิตอล อุปกรณ์ส่วนนี้ได้แก่ กล้องโทรทัศน์ดิจิไตเซอร์ โดยกล้องเปรียบเสมือนดวงตาของมนุษย์ทำหน้าที่เปลี่ยนภาพวัตถุ มาเป็นสัญญาณทางไฟฟ้าและส่งให้ดิจิไตเซอร์ ซึ่งทำหน้าที่เปลี่ยนสัญญาณไฟฟ้าให้เป็นสัญญาณดิจิตอล

ส่วนที่สอง คือส่วนประมวลผล ได้แก่ คอมพิวเตอร์ ซึ่งเปรียบเสมือนสมอง ที่ทำหน้าที่ประมวลผลและวิเคราะห์ข้อมูลภาพ

ส่วนสุดท้าย คือส่วนแสดงผล ทำหน้าที่เปลี่ยนข้อมูลตัวเลข (ซึ่งเป็นระดับสีเทา) ที่เก็บเป็นอาร์เรย์ (array) ในคอมพิวเตอร์ ให้อยู่ในรูปที่เหมาะสมและสื่อความหมายกับมนุษย์ได้ คือเป็นภาพปกติต่างๆ ไป อุปกรณ์ในส่วนนี้ได้แก่ มอนิเตอร์ โทรี เครื่องพิมพ์ที่สามารถแสดงผลในรูปแบบกราฟฟิกได้

ภาพหนึ่งภาพที่ถูกเปลี่ยนเป็นสัญญาณดิจิตอลสำหรับคอมพิวเตอร์ จะมีขนาดขึ้นกับความละเอียดของภาพที่ต้องการ โดยหากมีความละเอียดมากจะมีผลทำให้ใช้เนื้อที่ในหน่วยความจำมาก

ในการเก็บข้อมูล เช่น การเก็บภาพหนึ่งภาพ ขนาด 256 X 256 จุด ที่มีความแตกต่างของระดับความเข้มของแต่ละจุด เท่ากับ 256 ระดับ จะต้องใช้เนื้อที่ในหน่วยความจำถึง 64 กิโลไบต์ ในการเก็บภาพนี้ ดังนั้น ในปัจจุบันนี้ได้มีการค้นคว้าวิจัย หาวิธีการที่จะเก็บภาพด้วยคอมพิวเตอร์ โดยให้ใช้เนื้อที่ในหน่วยความจำให้น้อยที่สุด และยังรักษาความละเอียดของภาพตามการใช้งานได้อีกด้วย

2.1.3 ลักษณะข้อมูลภาพ

ข้อมูลภาพจะถูกแทนด้วยอาร์เรย์จัตุรัส (rectangular array) ของเลขจำนวนเต็ม โดยตำแหน่งที่ตั้งของแต่ละพิกเซลบอกด้วยแถวกับหลักของอาร์เรย์ และค่าในอาร์เรย์บอกระดับความเข้มของแต่ละพิกเซล (Gray – Level) ซึ่งมีถึง 256 ระดับ ที่ค่าระดับความเข้ม 256 แทนสีขาว คือบริเวณสว่าง ที่ค่าระดับความเข้ม 0 แทนสีดำ คือบริเวณมืด

2.2 สเต็ปเปอร์มอเตอร์

สเต็ปเปอร์มอเตอร์เป็นมอเตอร์ประเภทหนึ่งที่ถูกนำไปใช้งานมาก ไม่ว่าจะเป็นส่วนประกอบในหุ่นยนต์ เครื่องจักรกลที่ควบคุมด้วยคอมพิวเตอร์ ต่างก็ใช้สเต็ปเปอร์มอเตอร์เป็นตัวขับเคลื่อน หัวข้อต่างๆที่จะกล่าวก็จะประกอบไปด้วยการทำงานของสเต็ปเปอร์มอเตอร์ ตลอดจนการควบคุมการหมุนและวงจรอิเล็กทรอนิกส์ที่นำมาขับสเต็ปเปอร์มอเตอร์

2.2.1 ชนิดของสเต็ปเปอร์มอเตอร์

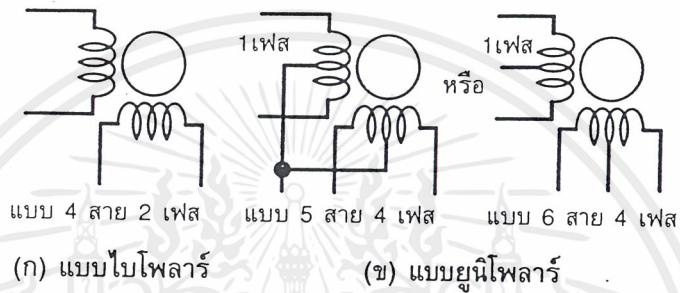
การแบ่งชนิดของสเต็ปเปอร์มอเตอร์แบ่งได้เป็น 6 ชนิดตามโครงสร้างพื้นฐานหรือความแตกต่างของรูปแบบของโรเตอร์ดังนี้

1. ชนิดวาริเอเบิลรีลักแตนซ์ (variable reluctance: VR)
2. ชนิดไฮบริด (hybrid)
3. ชนิดแม่เหล็กถาวร (permanent magnet)
4. ชนิดแรเอิร์ธเพอร์มาเนนต์แม็กเน็ต (rare earth permanent magnet)
5. ชนิดไบโพลาร์ (bipolar)
6. ชนิดยูนิโพลาร์ (unipolar)

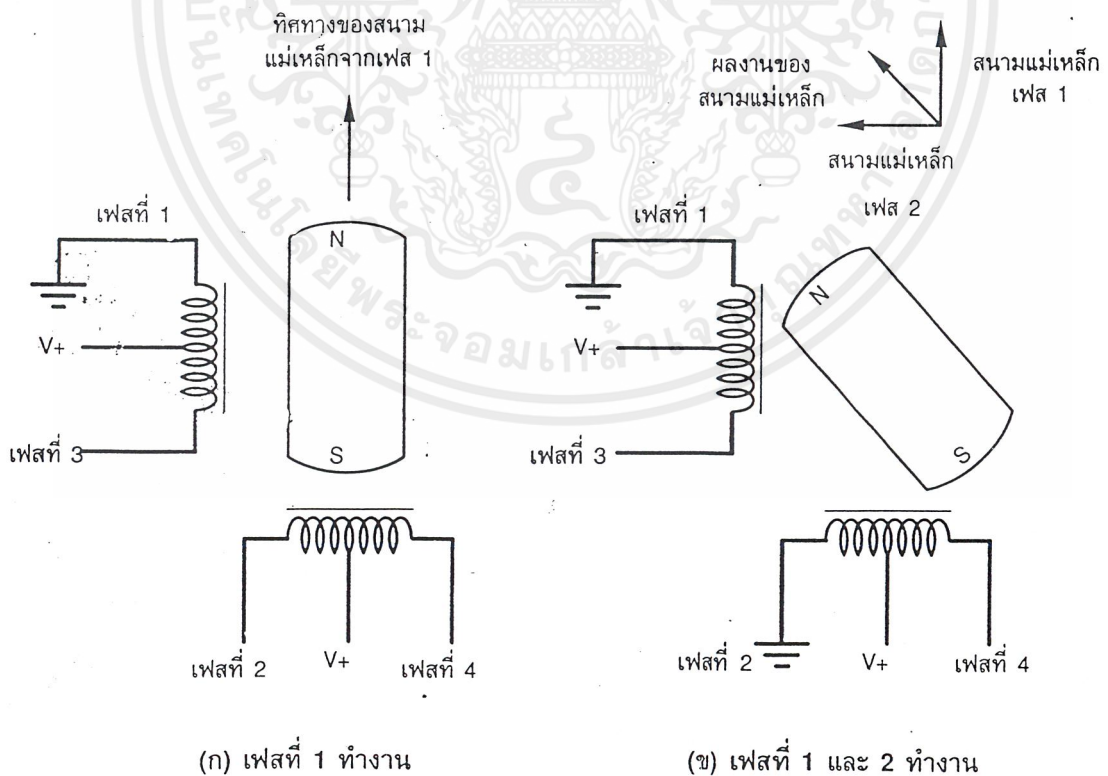
สำหรับโครงงานนี้ใช้สเต็ปเปอร์มอเตอร์แบบยูนิโพลาร์

2.2.2 การทำงานของสเต็ปเปอร์มอเตอร์

การทำงานของสเต็ปเปอร์มอเตอร์จะมีความแตกต่างจากมอเตอร์ทั่วไป คือเมื่อป้อนกำลังไฟให้สเต็ปเปอร์มอเตอร์ก็จะหมุนเพียงเล็กน้อยตามเส้นรอบวงแล้วหยุด สเต็ปเปอร์มอเตอร์สามารถกำหนดตำแหน่งการหมุนได้อย่างละเอียดโดยใช้คอมพิวเตอร์เป็นตัวกำหนดและจัดเก็บตัวเลขของตำแหน่งเหล่านั้นไว้



รูปที่ 2.3 ลักษณะการพันขดลวดบนสเตเตอร์



รูปที่ 2.4 ทิศทางการหมุนโรเตอร์ของสเต็ปเปอร์มอเตอร์ 4 เฟส

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หลักการการทำงานอย่างง่ายของสตีปเปอร์มอเตอร์ แบบยูนิโพลาร์ 4 เฟส ตัวมอเตอร์เป็นแม่เหล็ก การให้พลังงานแก่ขดลวดใดขดลวดหนึ่งโรเตอร์ก็จะหมุนไป 90 องศา ดังรูปที่ 2.4(ก) แต่ถ้าให้ที่เดียว 2 ขดลวดพร้อมกัน โรเตอร์ก็จะหมุนเพียง 45 องศา ดังรูปที่ 2.4(ข) ซึ่งแบบหลังจะสร้างแรงบิดได้มากกว่าแบบแรก สตีปเปอร์มอเตอร์จะมีมุมในการเคลื่อนที่แต่ละสเต็ปเป็น 1.8 องศา ดังนั้นที่โรเตอร์จะต้องมีขั้วแม่เหล็ก 50 ขั้ว ($90/50 = 1.8$)

2.2.3 การควบคุมการหมุนของสตีปเปอร์มอเตอร์

การกระตุ้นเพื่อควบคุมการหมุนของมอเตอร์ให้เคลื่อนที่ไปแต่ละสเต็ปในการใช้งานจริง ทำได้โดยจ่ายกำลังไฟฟ้าไปยังขดลวดแต่ละขดบนสเตเตอร์ ซึ่งต้องป้อนแบบซีควนเชียลในรูปแบบที่ถูกต้องซึ่งแบ่งเป็น 3 รูปแบบคือ (1)แบบเวฟ (2)แบบ 2 เฟส (two phase) และ (3)แบบครึ่งสเต็ป(half step) สำหรับโครงการนี้ใช้แบบ 2 เฟส ซึ่งมีหลักการดังนี้คือ

แบบ 2 เฟส (Two Phase)

เป็นการกระตุ้นรูปแบบหนึ่งซึ่งคล้ายกับแบบเวฟ แต่จ่ายกำลังไฟฟ้าไปที่ขดลวด 2 ขดที่อยู่ใกล้กันในเวลาเดียวกัน และถัดเรียงไปเช่นเดียวกับแบบเวฟคือขดลวดที่ 12,23,34,41,12 หรือ 14,43,32,21,14 ขึ้นอยู่กับทิศทางที่ต้องการให้หมุนการเพิ่มจำนวนของขดลวดที่ที่ถูกกระตุ้นนี้ทำให้เพิ่มแรงบิดได้มากกว่าแบบเวฟ แต่ต้องใช้แหล่งจ่ายไฟกำลังไฟฟ้ามากขึ้น การทำงานแสดงที่ตารางที่ 2.1

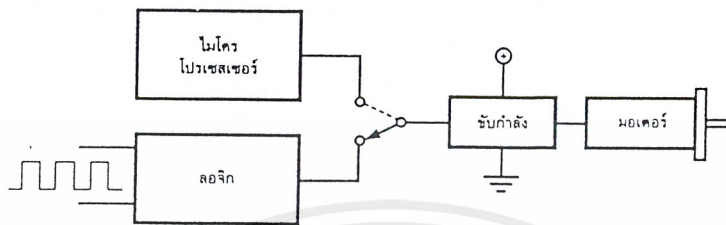
สเต็ปที่	เฟส 1	เฟส 2	เฟส 3	เฟส 4
1	ทำงาน	ทำงาน	-	-
2	-	ทำงาน	ทำงาน	-
3	-	-	ทำงาน	ทำงาน
4	ทำงาน	-	-	ทำงาน

ตารางที่ 2.1 แสดงการกระตุ้นขดลวดแต่ละเฟสแบบ 2 เฟส

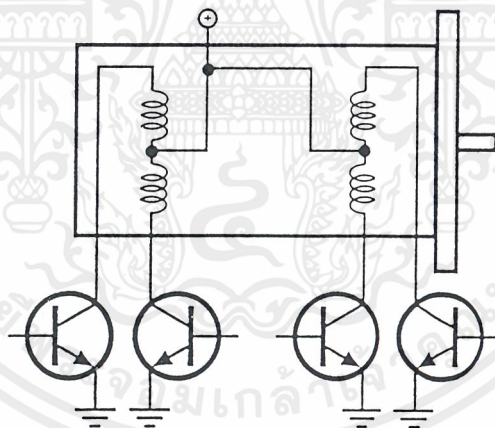
2.2.4 วงจรขับสเต็ปเปอร์มอเตอร์

การนำเอาวงจรอิเล็กทรอนิกส์มาใช้กับสเต็ปเปอร์มอเตอร์แสดงดังบล็อกไดอะแกรมรูปที่

2.5



รูปที่ 2.5 บล็อกไดอะแกรมควบคุมด้วยสเต็ปเปอร์มอเตอร์



รูปที่ 2.6 การขับสเต็ปเปอร์มอเตอร์แบบยูนิโพลาร์

จากรูปที่ 2.6 จากวงจะเห็นว่าวงจรขับสเต็ปเปอร์มอเตอร์แบบยูนิโพลาร์จะใช้ทรานซิสเตอร์ ขับขดลวดมอเตอร์ในลักษณะ 1 ขดต่อทรานซิสเตอร์ 1 ตัว

ถ้าต้องการควบคุมสเต็ปเปอร์มอเตอร์ด้วยคอมพิวเตอร์เพื่อให้มอเตอร์ทำงานตามต้องการ สามารถต่อเอาท์พุทพอร์ทโดยตรง และสามารถสั่งงานโดยใช้ซอฟต์แวร์ เพื่อสั่งงานให้มอเตอร์ หมุนเดินหน้าหรือถอยหลังได้ตามต้องการและนอกจากนี้ยังสามารถเลือกได้ว่าจะหมุนเป็นแบบเวฟ แบบ 2 เฟส หรือแบบครึ่งสเต็ปก็ได้ จากการปรับช่วงห่างระหว่างเวลาแต่ละสเต็ป วิธีการควบคุมก็ โดยอาศัยวงจรลอจิกนำเอาท์พุทมาควบคุมทรานซิสเตอร์ผ่านวงจร RS ไบสเตเบิล เพื่อป้องกันข้อผิดพลาดที่อาจเกิดขึ้น เช่น กรณีของการนำกระแสพร้อมๆกันของทรานซิสเตอร์ทั้ง 4 ตัวแบบบริดจ์ ในบางกรณีลอจิกเกตจะถูกใช้เป็นตัวรีเซตวงจรไบสเตเบิลเพื่อกำหนดทิศทางการหมุนก็ได้

มีวงจรรวมหรือ ไอซีที่ผลิตขึ้นมาเพื่อใช้ในการออกแบบเป็นวงจรควบคุมสเต็ปเปอร์ มอเตอร์ ที่ควบคุมความเร็ว ด้วยการเปลี่ยนแปลงความเร็วของพัลส์ ตัวอย่างเช่น ไอซี เบอร์ SAA1027, L297, L298, TL376 และ ULN2002-UL2005 เป็นต้น

2.3 ไมโครคอนโทรลเลอร์ ตระกูล MCS-51

ไมโครคอนโทรลเลอร์เป็นไมโครโปรเซสเซอร์ประเภทหนึ่งที่ได้รับการออกแบบมาเพื่อใช้งานกับระบบควบคุมที่มีขนาดเล็ก โดยภายในไอซีไมโครคอนโทรลเลอร์หนึ่งตัวจะประกอบด้วยการทำงานหลักของระบบคอมพิวเตอร์ครบถ้วน เช่น หน่วยประมวลผลกลาง (CPU) หน่วยความจำพอร์ตในการติดต่อหรือควบคุมอุปกรณ์ต่างๆ เป็นต้น ซึ่งหากว่าเป็นการใช้งานไมโครโปรเซสเซอร์ทั่วไปก็ต้องใช้ไอซีภายนอกมาประกอบเพื่อทำหน้าที่เหล่านี้ ดังนั้นจึงอาจกล่าวได้ว่า ไมโครคอนโทรลเลอร์เป็นระบบคอมพิวเตอร์เพื่องานควบคุมที่สมบูรณ์โดยบรรจุอยู่ใน ไอซีเพียงหนึ่งตัวเท่านั้น จึงอาจเรียกว่าระบบไมโครคอมพิวเตอร์ชิปเดี่ยว (single chip microcomputer)

2.3.1 คุณสมบัติของไมโครคอนโทรลเลอร์ตระกูล MCS-51

- ต้องการแหล่งจ่ายไฟ +5 โวลต์ เพียงชุดเดียว
- หน่วยประมวลผลกลางขนาด 8 บิต และหน่วยประมวลผลข้อมูลเกี่ยวกับการคำนวณ (arithmetic logic unit) ขนาด 8 บิต
- มีหน่วยประมวลผลสำหรับข้อมูลแบบบิต (boolean processor) และมีคำสั่งในการประมวลผลเช่น AND OR และ COMPLEMENT
- มีหน่วยความจำโปรแกรม (program memory) ขนาด 4 KB แบบ ROM สำหรับเบอร์ 8051 และ 8031 และมีหน่วยความจำถึง 8 KB เป็นแบบ EPROM สำหรับเบอร์ 8052
- มีหน่วยความจำสำหรับเก็บข้อมูล (data memory) แบบ RAM ขนาด 128 ไบต์ สำหรับเบอร์ 8052 เป็นต้นไปมีถึง 256 ไบต์

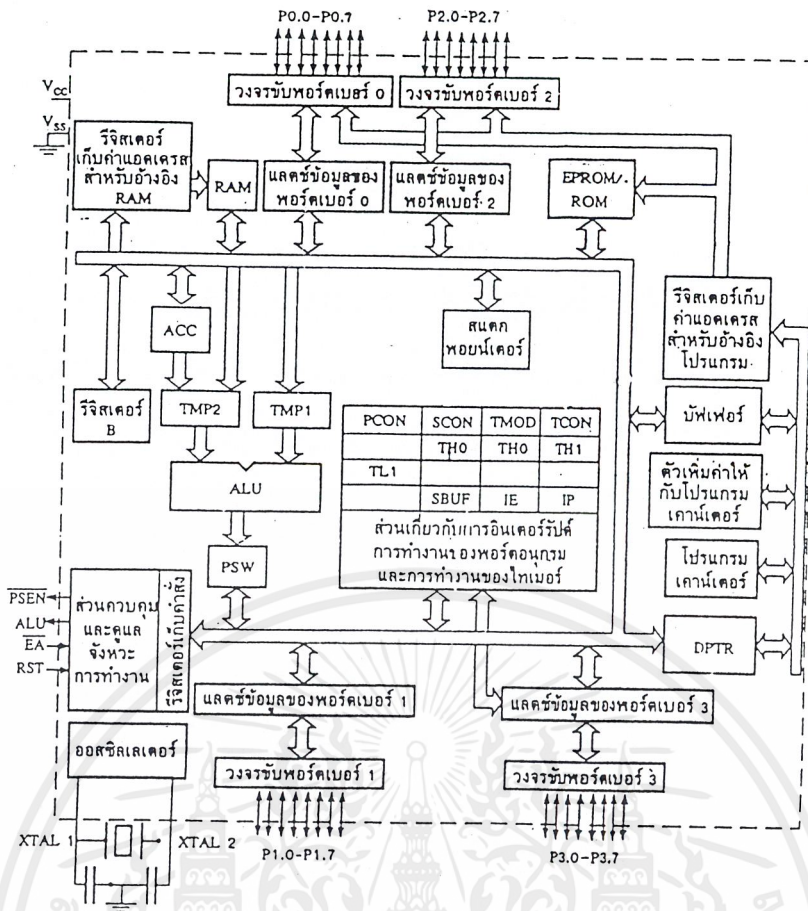
- ความสามารถในการอ้างตำแหน่งของหน่วยความจำสำหรับเก็บโปรแกรมและข้อมูล แยกจากกันอย่างละ 64 KB
- มีวงจรรนับ/จับเวลา (Counter/Timer) ขนาด 16 บิต 2 ชุด (สำหรับ 8052 มี 3 ชุด) ทำงานได้ 4 โหมด
- เส้นส่งสัญญาณอินพุท/เอาต์พุท(input/output) แบบขนาน จำนวน 32 เส้น ซึ่งสามารถแยกทำงานได้อย่างอิสระ
- พอร์ตรับส่งข้อมูลอนุกรม (UART) 2 พอร์ต แบบฟูลดูเพล็กซ์ (Full Duplex)
- ขารับสัญญาณอินเตอร์รัพ (Interrupt) ภายนอก 2 ขา มีวงจรถบคุมการอินเตอร์รัพ รับอินเตอร์รัพได้จากแหล่งกำเนิดสัญญาณ 6 แหล่ง 5 เวกเตอร์ (สำหรับเบอร์ 8052 ขึ้นไป มี 8 แหล่ง 6 เวกเตอร์) พร้อมการกำหนดระดับความสำคัญได้ 2 ระดับ
- คำสั่งส่วนใหญ่ใช้เวลาทำงานเพียง 1 ไมโครวินาที เมื่อใช้คริสตอลความถี่ 12 เมกะเฮิร์ต
- มีชุดคำสั่ง 111 คำสั่ง

2.3.2 สถาปัตยกรรมของ 8051

จากรูปที่ 2.7 เป็นบล็อกไดอะแกรมแสดงสถาปัตยกรรมของ 8051 ซึ่งจะเห็นว่า การเชื่อมต่อของชิพกับภายนอกนั้นไม่มีความซับซ้อนมากนัก ตัวชิพดังในรูปจะมีขาสัญญาณจำนวน 32 ขา สำหรับใช้เป็นพอร์ตอินพุท/เอาต์พุท ขนาด 8 บิต จำนวน 4 พอร์ต ที่สามารถส่งข้อมูลได้ 2 ทิศทาง นอกจากนี้ ตัวชิพยังมีขาสำหรับป้อนไปเลี้ยงขาสำหรับใช้เชื่อมต่อคริสตอลที่สร้างสัญญาณนาฬิกา และขาส่งสัญญาณกำกับจังหวะการทำงานและขาส่งสัญญาณควบคุม ซึ่งพบว่า 8051 มีโครงสร้างภายนอกที่ไม่สลับซับซ้อน ซึ่งก็หมายความว่าเราจะต้องเพิ่มอุปกรณ์ภายนอกไม่กี่ชิ้นในการที่จะทำให้ระบบที่ใช้ 8051 ทำงาน

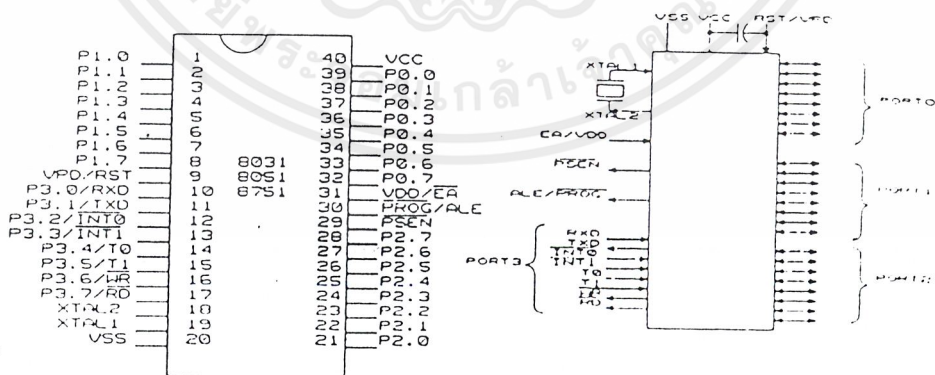
ในทางกลับกัน โครงสร้างภายใน ของ 8051 จะมีความสลับซับซ้อนพอสมควร เราจะพบส่วนที่ทำหน้าที่สำคัญในไมโครโปรเซสเซอร์ได้ตรงกลางบล็อกไดอะแกรมรวมถึง ALU แอ็คคิวมูเลเตอร์ สเตตทอพอยต์เตอร์ รีจิสเตอร์ที่มีหน้าที่ทั่วไปต่างๆ (เช่น รีจิสเตอร์ B) โดยส่วนต่าง ๆ เหล่านี้จะเชื่อมต่อกับ บัสภายในของ 80510

พอร์ตอินพุท/เอาต์พุท แต่ละตัวจะถูกเชื่อมต่อกับบัสข้อมูลภายในขนาด 8 บิต โดยผ่าน รีจิสเตอร์ที่ต่ออนุกรมกัน รีจิสเตอร์เหล่านี้จะมีหน้าที่ในการเก็บข้อมูลในช่วงที่มีการโอนย้ายข้อมูลของพอร์ตอินพุท/เอาต์พุท จากบล็อกไดอะแกรมนี้ยังแสดงถึง ROM, RAM ของ 8051 ด้วย



รูปที่ 2.7 บล็อก ไดอะแกรม แสดงสถาปัตยกรรมของ MCS-51

2.3.3 พอร์ตของ MCS-51

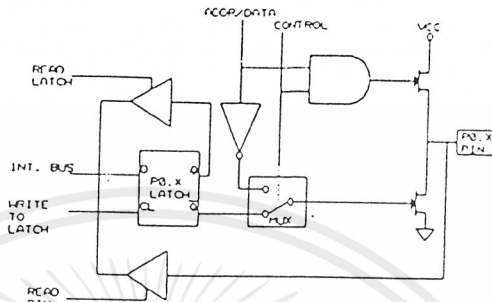


รูปที่ 2.8 การจัดวางขาของ MCS-51

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

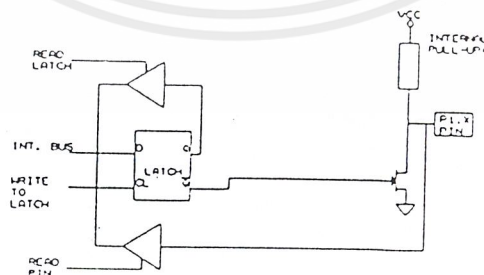
MCS-51 เป็นไมโครคอนโทรลเลอร์ขนาด 40 ขา ซึ่งแต่ละขาจะมีหน้าที่ระบุชัดเจนตามสัญลักษณ์ชื่อย่อที่กำกับในแต่ละขา ดังนี้

- ขา VCC (ขา 40) ต่อกับไฟ +5 โวลต์
- ขา VSS (ขา 20) ต่อเป็นขากราวด์ (GND)



รูปที่ 2.9 แสดงโครงสร้างของพอร์ต 0

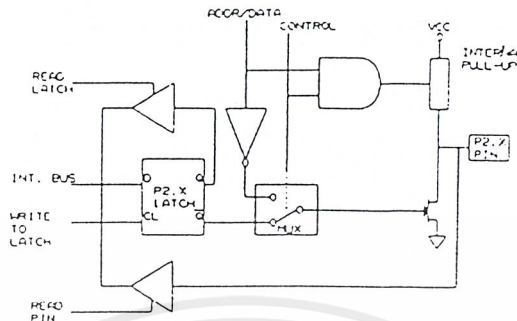
- ขาพอร์ต 0 (ขา 32-39) มี 8 ขา (P0.0 – P0.7) ใช้งานได้ 2 หน้าที่คือ แอคเตอเรสกับ และ คาต้าบัส เมื่อต้องการติดต่อกับหน่วยความจำภายนอกหรือเป็น อินพุท/เอาต์พุทพอร์ต โดยถ้าต้องการให้ทำงานเป็นอินพุทพอร์ต ต้องส่ง ลอจิก “1” ไปยังแต่ละบิตของพอร์ต จะมีผลให้ Q ของ D-FF เป็น “0” ทำให้ FET ตัวล่างมีสถานะ OFF สัญญาณที่ให้อ่าน อินพุทพอร์ตแลทโดยส่งสัญญาณ Read Latch ไปกระตุ้นที่ไตรสเททบัฟเฟอร์ตัวบน และการอ่านพอร์ต(Pin)จะใช้สัญญาณ Read (Pin) นอกจากนี้ใช้งานเป็นอินพุท/เอาต์พุทพอร์ตแล้ว พอร์ต 0 ยังใช้ในการติดต่อกับหน่วยความจำสำหรับเก็บ โปรแกรมและข้อมูล ภายนอกชิพด้วย



รูปที่ 2.10 แสดงโครงสร้างของพอร์ต 1

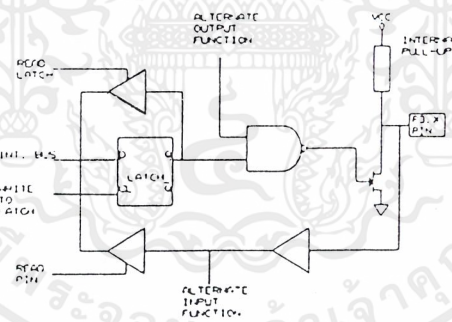
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและไม่ต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- ขาพอร์ต 1 (ขา 1-8) มี 8 ขา (P1.0 – P1.7) อันมีโครงสร้างคล้ายพอร์ต 0 แต่จะใช้ความต้านทานภายในพูลอัพแทน โดยอินเทอร์นอลพูลอัพฟรีจิสเตอร์



รูปที่ 2.11 แสดง โครงสร้างของพอร์ต 2

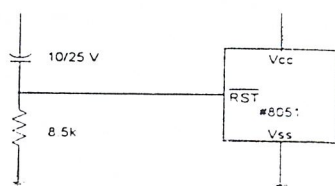
- ขาพอร์ต 2 (ขา 21-28) มีทั้งหมด 8 ขา (P2.0 - P2.7) มีโครงสร้างคล้ายพอร์ต 0 โดยมี FET ตัวล่างตัวเดียว ส่วนด้านบนใช้ความต้านทานพูลอัพแทน(internal pull up) พอร์ตนี้ทำงาน 2 หน้าที่ คือ (1) สามารถใช้เป็นแอดเดรสสับขนาด 8 บิต (A8-A15) และ (2) สามารถใช้งานเป็นอินพุท/เอาต์พุทพอร์ตใช้งานได้



รูปที่ 2.12 แสดง โครงสร้างของพอร์ต 3

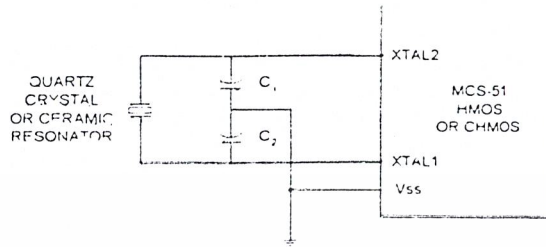
- ขาพอร์ต 3 (ขา 10 –17) มีทั้งหมด 8 ขา (P3.0-P3.7) มีโครงสร้างคล้ายพอร์ต 1 ทำงานได้ 2 หน้าที่คือ (1) เป็นอินพุท/เอาต์พุทพอร์ต และ (2) ใช้ส่งสัญญาณควบคุมออกมา และรับสัญญาณเข้าไป สัญญาณต่างๆ มีดังนี้
 - P3.0 /RXD (ขา 10) อินพุทพอร์ตอนุกรม(serial input port) เป็นขาที่ใช้รับข้อมูลแบบอนุกรม (UART)

- P3.1/TXD (ขา 11) เอาท์พุทพอร์ตอนุกรม(serial output port) เป็นขาที่ใช้ส่งข้อมูลแบบอนุกรม(UART)
 - P3.2/INT0 (ขา 12) อินเทอร์รัพภายนอก 0(External Interrupt 0) ใช้รับสัญญาณการขัดจังหวะจากภายนอกเบอร์ 0
 - P3.3/INT1 (ขา 13) อินเทอร์รัพภายนอก 1(External Interrupt 1) ใช้รับสัญญาณการขัดจังหวะจากภายนอกเบอร์ 1
 - P3.4/T0 (ขา 14) เคาท์เตอร์ 0 อินพุทภายนอก(Counter 0 External Input) ขารับสัญญาณพัลส์อินพุทเข้าไปยังวงจรเคาท์เตอร์ 0 (เป็นอินพุทโหมดเคาน์เตอร์)
 - P3.5/T1 (ขา 15) เคาท์เตอร์ 1 อินพุทภายนอก(Counter 1 External Input)ขารับสัญญาณพัลส์อินพุทเข้าไปยังวงจรเคาท์เตอร์ 1 (เป็นอินพุทโหมดเคาน์เตอร์)
 - P3.6/WR (ขา 16) (External Data Memory Write Strobe) ขาสัญญาณควบคุมการเขียนข้อมูลลงหน่วยความจำภายนอก
 - P3.7/RD (ขา 17) (External Data Memory Read Strobe) ขาสัญญาณควบคุมการอ่านข้อมูลลงหน่วยความจำภายนอก
- ALE (ขา 30) เป็นขาส่งสโตรบสำหรับใช้ในการแลตซ์แอดเดรสไบต์ต่ำ (A0-A7) ที่ส่งออกมาจากพอร์ต 0 สัญญาณนี้จะแอกทีฟทุกๆ 2 ครั้งใน 1 เมทซิงไซเคิล
 - PSEN (ขา 29) เป็นขาสโตรบที่ใช้สำหรับอ่านข้อมูลจาก โปรแกรมเมมโมรี่ภายนอก สัญญาณนี้จะส่งออกมา 2 ครั้งในแต่ละเมทซิงไซเคิล แต่ถ้าเป็นการอ่านโปรแกรมเมมโมรี่ภายใน จะไม่มีสัญญาณออกที่ขานี้
 - EA (ขา 31) ใช้เลือกหน่วยความจำโปรแกรมภายนอก โดยป้อน “0” จะอ่านโปรแกรมจากภายนอกชิพ ป้อน “1” จะอ่านโปรแกรมจากภายในชิพ



รูปที่ 2.13 แสดงการต่อขารีเซตให้กับ MCS-51

- RST (ขา 9) ขารีเซต จะรีเซตได้ก็ต่อเมื่อป้อนลอจิก “1” เข้าที่ขานี้นานอย่างน้อย 2 แมทซึน ไซเคลิส



รูปที่ 2.14 แสดงวงจรคล็อกของ MCS-51 แบบใช้คล็อกภายใน (using the on-chip oscillator)

- XTAL1 (ขา19) ใช้ต่อคริสตอลภายนอกโดยเป็นอินพุทเข้าของวงจรรอสซิลเลเตอร์ภายใน
- XTAL2 (ขา18) ใช้ต่อคริสตอลภายนอก โดยเป็นเอาต์พุทของวงจรรอสซิลเลเตอร์ภายใน

2.3.4 การรับส่งข้อมูลแบบอนุกรม (UART) ใน MCS-51

พอร์ตสื่อสารอนุกรมของ MCS-51 มีโครงสร้างการทำงานในแบบที่เรียกว่า ฟูลดูเพลกซ์ ที่สามารถรับและส่งข้อมูลอนุกรมได้ในเวลาเดียวกัน โดยทางด้านส่งใช้ขา TXD(P3.1) ทางด้านรับใช้ขา RXD(P3.0) บัฟเฟอร์ SBUF ใช้เป็นบัฟเฟอร์สำหรับรับและส่งข้อมูลอนุกรมจึงทำให้ MCS-51 สามารถกำหนดการรับข้อมูลไบต์ที่สองที่ถูกส่งตามเข้ามา ก่อนที่ข้อมูลไบต์แรกที่ได้รับเข้ามาจะถูกอ่านจากรีจิสเตอร์ใช้งานเฉพาะที่ใช้สำหรับรับข้อมูล (Resist Register) เพื่อนำไปเก็บไว้ในหน่วยความจำต่อไป

พอร์ตสื่อสารข้อมูลแบบอนุกรมใน MCS-51 ประกอบด้วยรีจิสเตอร์ขนาด 8 บิต จำนวน 2 ตัว คือ (1) รีจิสเตอร์สำหรับรับข้อมูล (Resist Register) ใช้รับข้อมูลที่ส่งเข้ามาจากภายนอก และ (2) รีจิสเตอร์สำหรับส่งข้อมูล (Transmit Register) ใช้ส่งข้อมูลจาก MCS-51 ออกไปภายนอก ซึ่งรีจิสเตอร์ทั้งสองมีตำแหน่งเดียวกันในรีจิสเตอร์ใช้งานเฉพาะ SBUF (ตำแหน่ง 99H) พอร์ตสื่อสารอนุกรมของ MCS-51 สามารถโปรแกรมการทำงานได้หลายโหมดด้วยกัน โดยเลือกที่บิต SM1 และ SM0 ซึ่งอยู่ในรีจิสเตอร์ควบคุม SCON

2.3.5 ชุดคำสั่งของ MCS-51

ไมโครคอนโทรลเลอร์ MCS-51 ประกอบด้วยชุดคำสั่ง 111 คำสั่ง ซึ่งสามารถจะจัดกลุ่มคำสั่งตามลักษณะหน้าที่การทำงานที่คล้ายคลึงกันได้ดังนี้

2.3.5.1 กลุ่มคำสั่งในการโอนย้ายข้อมูล ทำหน้าที่ในการโอนย้ายข้อมูลระหว่างรีจิสเตอร์หรือหน่วยความจำโดยรูปแบบของคำสั่ง จะอยู่ในลักษณะ

MOV ปลายทาง, ต้นทาง

เป็นคำสั่งโอนย้ายข้อมูล นำข้อมูลจากต้นทาง (source) ไปเก็บไว้ยังปลายทาง (destination)

2.3.5.2 กลุ่มคำสั่งดำเนินการแบบลอจิก ทำหน้าที่เกี่ยวกับการประเมินผลแบบลอจิกต่างๆ คำสั่งทางลอจิก(logical operations) โดยพื้นฐานได้แก่ AND, OR, XOR, NOT ระหว่างข้อมูลในรีจิสเตอร์ A กับรีจิสเตอร์ทั่วไป และการทำคำสั่งเกี่ยวกับการเลื่อนหรือหมุนข้อมูลภายในรีจิสเตอร์ A เป็นต้น

2.3.5.3 กลุ่มคำสั่งในการกระโดด(jump)ไปยังตำแหน่งต่างๆ ภายในโปรแกรมซึ่งจะเปลี่ยนลำดับการประมวลผลภายในโปรแกรมไปยังส่วนต่างๆ แทนที่จะดำเนินการไปเป็นลำดับต่อเนื่องตามปกติ ซึ่งสามารถจะทำการกระโดดได้ทั้งแบบมีและไม่มีเงื่อนไข นอกจากนั้นกลุ่มคำสั่งนี้ยังรวมถึง การเรียกใช้โปรแกรมย่อย(call program) และการกลับมาทำโปรแกรมหลักด้วย

2.3.5.4 กลุ่มคำสั่งการคำนวณทางคณิตศาสตร์ เช่นการบวก ลบ คูณ และหารข้อมูลภายในรีจิสเตอร์ต่าง ๆ ซึ่งมีความเหมาะสมกับงานประยุกต์และงานควบคุมต่าง ๆ ที่ไม่ค่อยมีการคำนวณมากนัก แต่ถ้าหากว่าต้องการทำคำสั่งคณิตศาสตร์ที่ซับซ้อนและไม่มีชุดคำสั่งของไมโครคอนโทรลเลอร์สนับสนุนโดยตรงก็ควรใช้วิธีการทำซอฟต์แวร์เพื่อสร้างเป็นโปรแกรมมาดำเนินการแทน

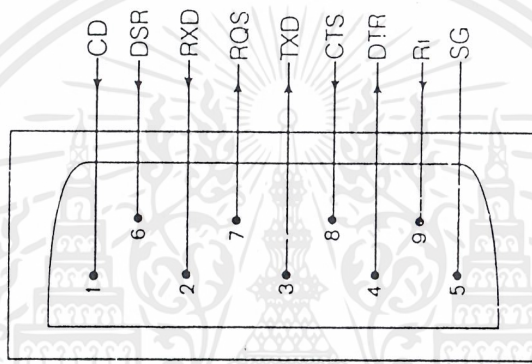
2.3.5.5 กลุ่มคำสั่งการประมวลผลแบบบิต ซึ่งเป็นความสามารถของ MCS-51ที่จะดำเนินการประมวลผลแบบบิต นอกเหนือไปจากการประมวลผลแบบไบต์เหมือนกับไมโครโปรเซสเซอร์โดยทั่วไป โดยมีชุดคำสั่งที่จัดการโดยตรง

2.4 วงจรรับส่งข้อมูลแบบอนุกรมมาตรฐาน RS-232

ในการรับส่งข้อมูลระหว่างไมโครคอนโทรลเลอร์และคอมพิวเตอร์ได้เลือกใช้การรับส่งข้อมูลแบบอนุกรม เนื่องจากสัญญาณจากคอมพิวเตอร์เป็นมาตรฐาน RS-232 (-12V ถึง 12V) แต่สัญญาณของไมโครคอนโทรลเลอร์เป็นแบบTTL(0 ถึง 5V) ทำให้ต้องมีตัวแปลงเพื่อแปลงจากมาตรฐาน RS-232 ไปเป็น TTL(อินพุท) และจาก TTL ไปเป็นมาตรฐาน RS-232(เอาต์พุท) ซึ่งเลือกใช้วงจรรวมเบอร์ MAX232 มาเป็นตัวแปลง

การเชื่อมต่อตามมาตรฐาน RS-232 แบ่งเป็น 2 ลักษณะ คือ การต่อกับเทอร์มินอล (DTE : Data Terminal Equipment) และการต่อกับอุปกรณ์สื่อสารข้อมูล (DCE : Data Communication Equipment) ซึ่งโดยปกติ DCE จะต้องต่อเข้ากับ DTE เสมอ เช่นการต่อโมเด็ม(อุปกรณ์ DCE) กับเครื่องคอมพิวเตอร์ส่วนบุคคล(อุปกรณ์ DTE)

การเชื่อมต่อตามมาตรฐาน RS-232 โดยใช้ คอนเนคเตอร์แบบ DB9(แบบ 9 ขา) กำหนดให้ ปลายสายสัญญาณด้านหนึ่งเป็นคอนเนคเตอร์ตัวผู้ต่อกับอุปกรณ์ DCE และปลายของสายสัญญาณอีกด้านหนึ่งเป็นคอนเนคเตอร์ตัวเมียต่อกับอุปกรณ์ DTE



รูปที่ 2.15 แสดงคอนเนคเตอร์แบบ DB9

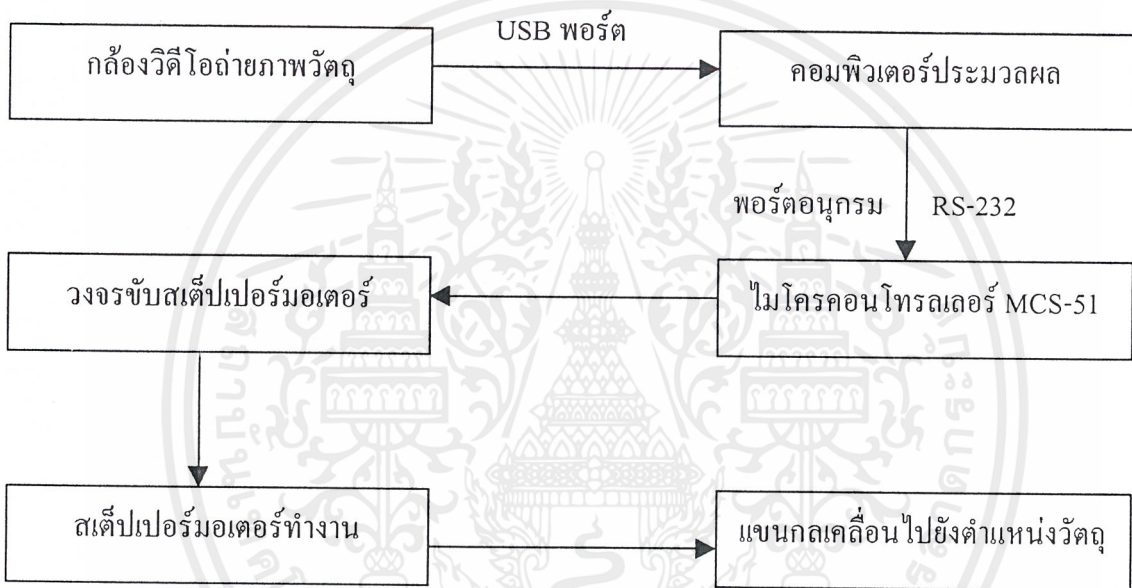
หมายเลขขาสัญญาณ	ชื่อของสายสัญญาณ
1	Carrier Detect
2	Received Data
3	Transmitted Data
4	Data Terminal Ready
5	Signal Common
6	Data Ser Ready
7	Request To Send
8	Clear To Send
9	Ring Indicator

ตารางที่ 2.2 แสดงรายละเอียดการต่อกอนเนคเตอร์แบบ DB9 ตามมาตรฐาน RS-232

บทที่ 3

หลักการออกแบบ

โครงการนี้ประกอบด้วยส่วนของซอฟต์แวร์และฮาร์ดแวร์ ซึ่งมีบล็อกไดอะแกรมของการทำงานโครงการ ดังรูปที่ 3.1



รูปที่ 3.1 แสดง บล็อกไดอะแกรมของการทำงาน

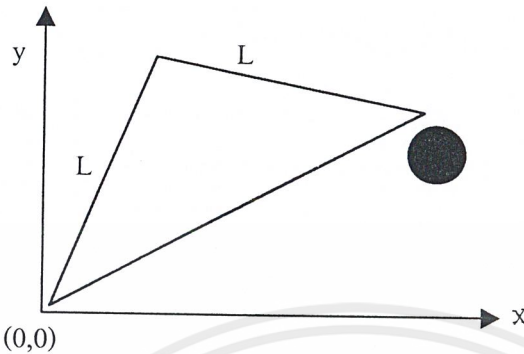
3.1 ส่วนของซอฟต์แวร์

3.1.1 โปรแกรมประมวลผลภาพดิจิทัล

ใช้โปรแกรม Microsoft Visual Basic 6 ในการประมวลผลภาพเพื่อหาตำแหน่งจุดศูนย์กลางของวัตถุและคำนวณหาค่าจำนวนสตีปการหมุนของสตีปเปอร์มอเตอร์แต่ละตัว ซึ่งสามารถเขียนโฟลว์ชาร์ต(flow chart) แสดงการทำงานได้ดังรูปที่ 3.5 และ รูปที่ 3.6 ส่วนฟอร์มของส่วนควบคุม (Fontrol Form) แสดงดังรูปที่ 3.7

หลักการคำนวณหาค่าจำนวนสตีปการหมุนของสตีปเปอร์มอเตอร์แต่ละตัว

1. กรณีที่มุมระหว่างแกนท่อนล่างกับแกน y (angleA) มีค่ามากกว่า 0



รูปที่ 3.2 แสดงแบบจำลองของแขนกล ในระนาบ x-y

กำหนดให้

- L แทนความยาวของแขนกลแต่ละท่อน
- (X,Y) แทนตำแหน่งจุดศูนย์กลางของวัตถุ
- B แทนค่ามุมระหว่างแกนท่อนล่างกับแกนท่อนบน
- A แทนมุมระหว่างแกนท่อนล่างกับแกน y

ระยะทางระหว่างจุดศูนย์กลางของวัตถุกับจุดกำเนิด = $H = \text{รากที่สองของ}(X^2 + Y^2)$

มุมของวัตถุ = $\text{angleP} = \arctan(Y / X)$

$L = 200 \text{ pixel}$

$Q = H / (2 * L)$

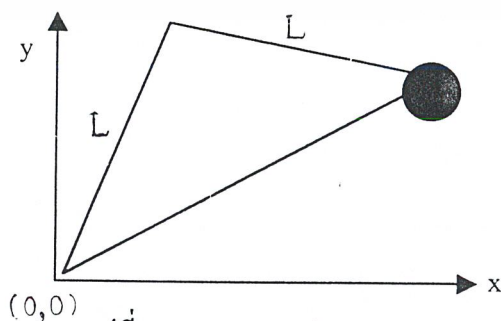
มุมระหว่างแขนทั้งสองท่อน = $2 \arcsin Q$

มุมระหว่างแกนท่อนแรกกับแกน y = $\text{angleA} = \text{angleP} - (90^\circ - \text{angleB} / 2)$

จำนวนสตีปของสตีปเปอร์มอเตอร์ตัวล่าง = $S = \text{angleA} / 1.8^\circ$

จำนวนสตีปของสตีปเปอร์มอเตอร์ตัวบน = $T = \text{angleB} / 7.5^\circ$

2. กรณีที่มุมระหว่างแกนท่อนล่างกับแกน y (angleA) มีค่าเท่ากับ 0

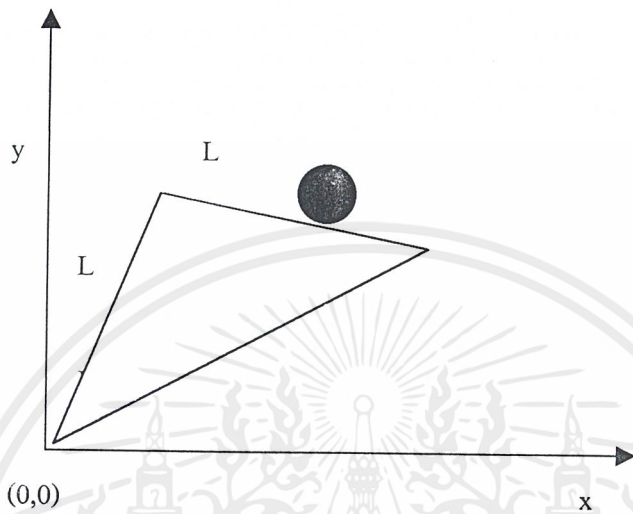


รูปที่ 3.3 แสดงแบบจำลองของแขนกลในระนาบ x-y

จากกรณีที่ 1 จะได้สตีปการหมุนของสตีปเปอร์มอเตอร์ตัวล่าง = 1

และจะได้สตีปการหมุนของสตีปเปอร์มอเตอร์ตัวบน = $\text{angerB}/7.5^\circ$

3. กรณีที่มุมระหว่างแกนท่อนล่างกับแกน y(angerA) มีค่าน้อยกว่า 0

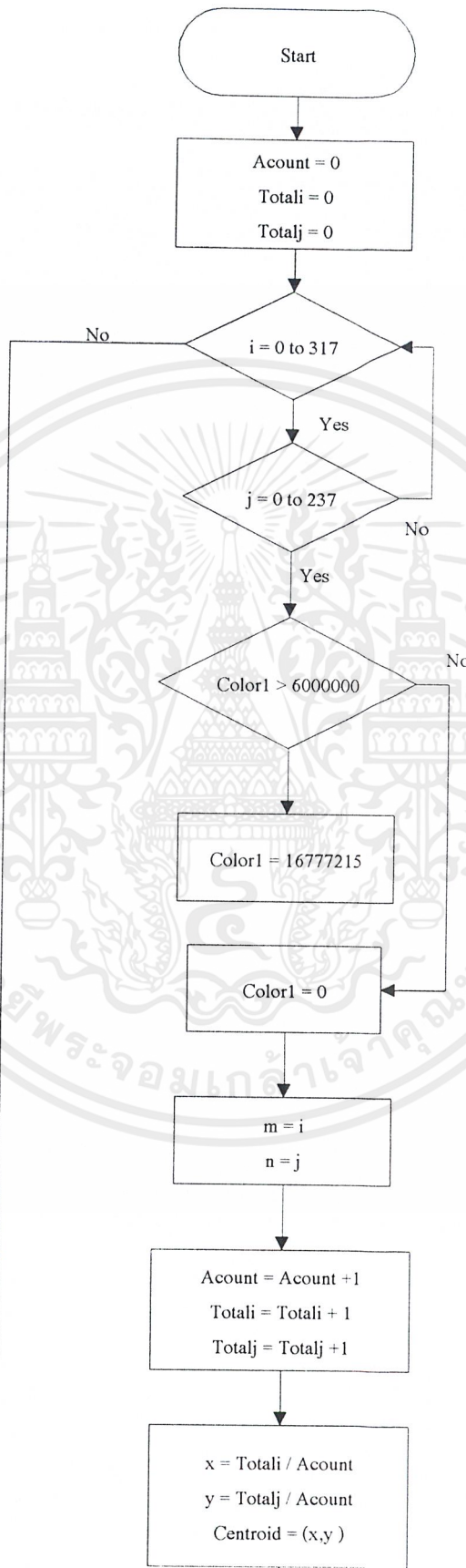


รูปที่ 3.4 แสดงแบบจำลองของแกนกลในระนาบ x-y

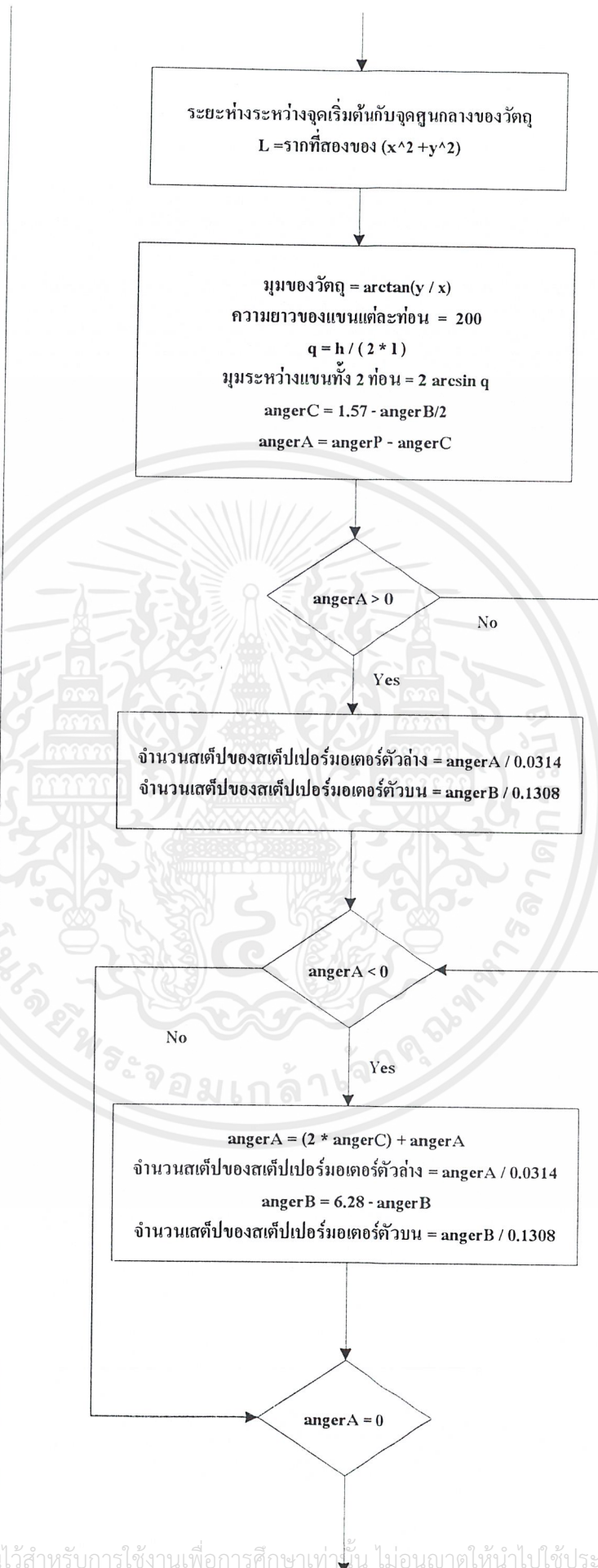
จากกรณีที่ 1 จะได้ $\text{angerA} = (2 * (1.57 - \text{angerB}/2)) + \text{angerA}$

สตีปการหมุนของสตีปเปอร์มอเตอร์ตัวล่าง = $\text{angerA}/1.8^\circ$

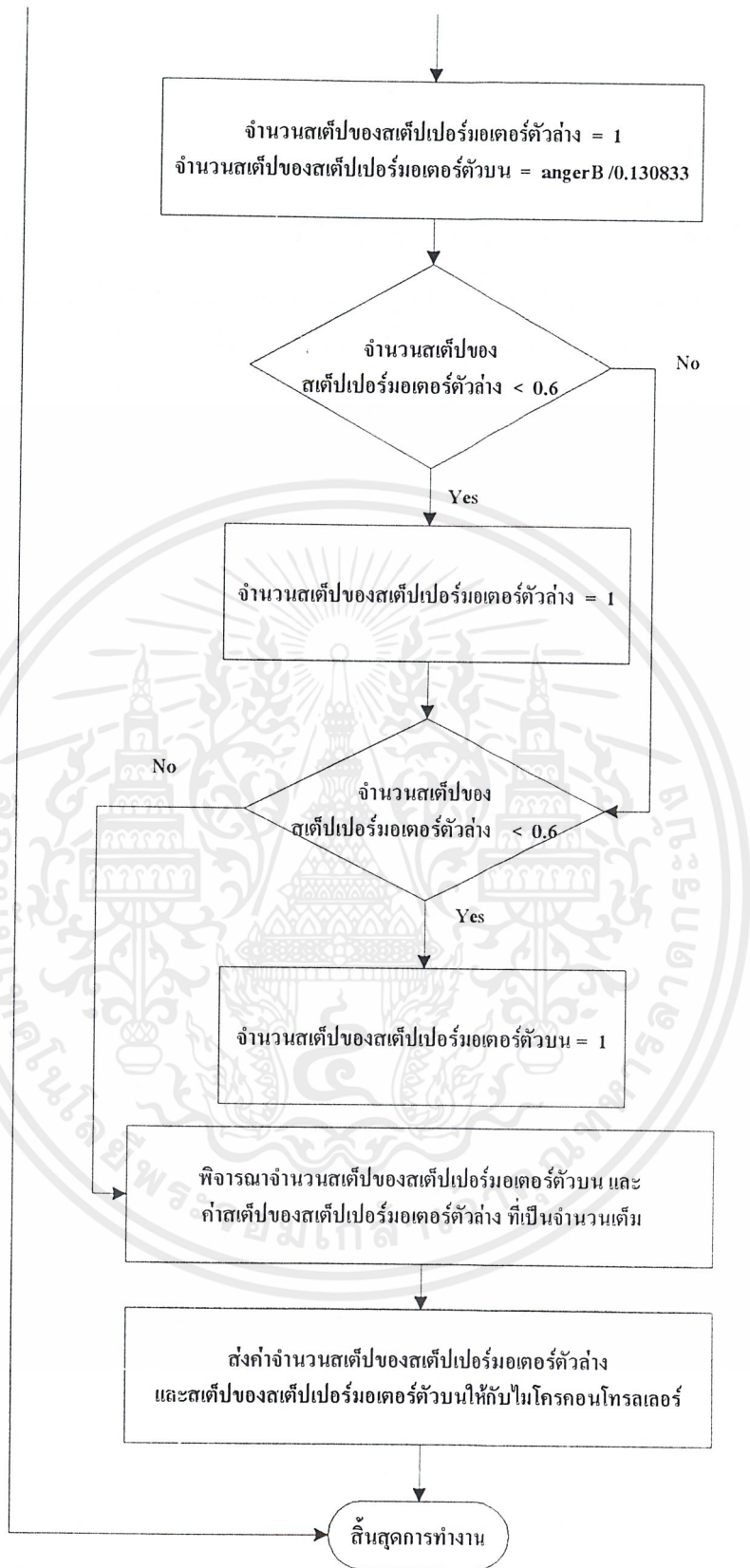
สตีปการหมุนของสตีปเปอร์มอเตอร์ตัวบน = $\text{angerB}/7.5^\circ$



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและ 20 ้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

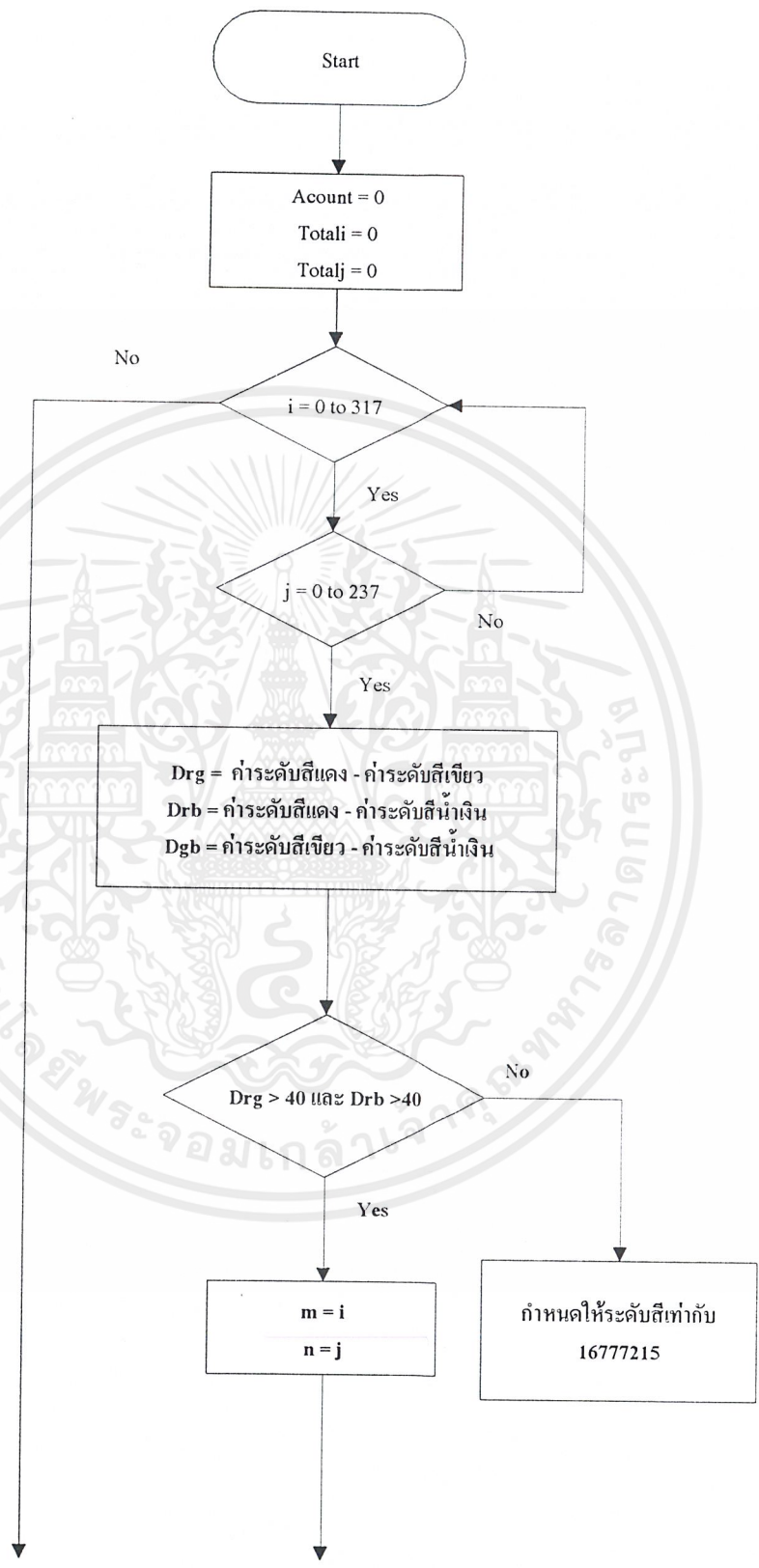


เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาแล211ห้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

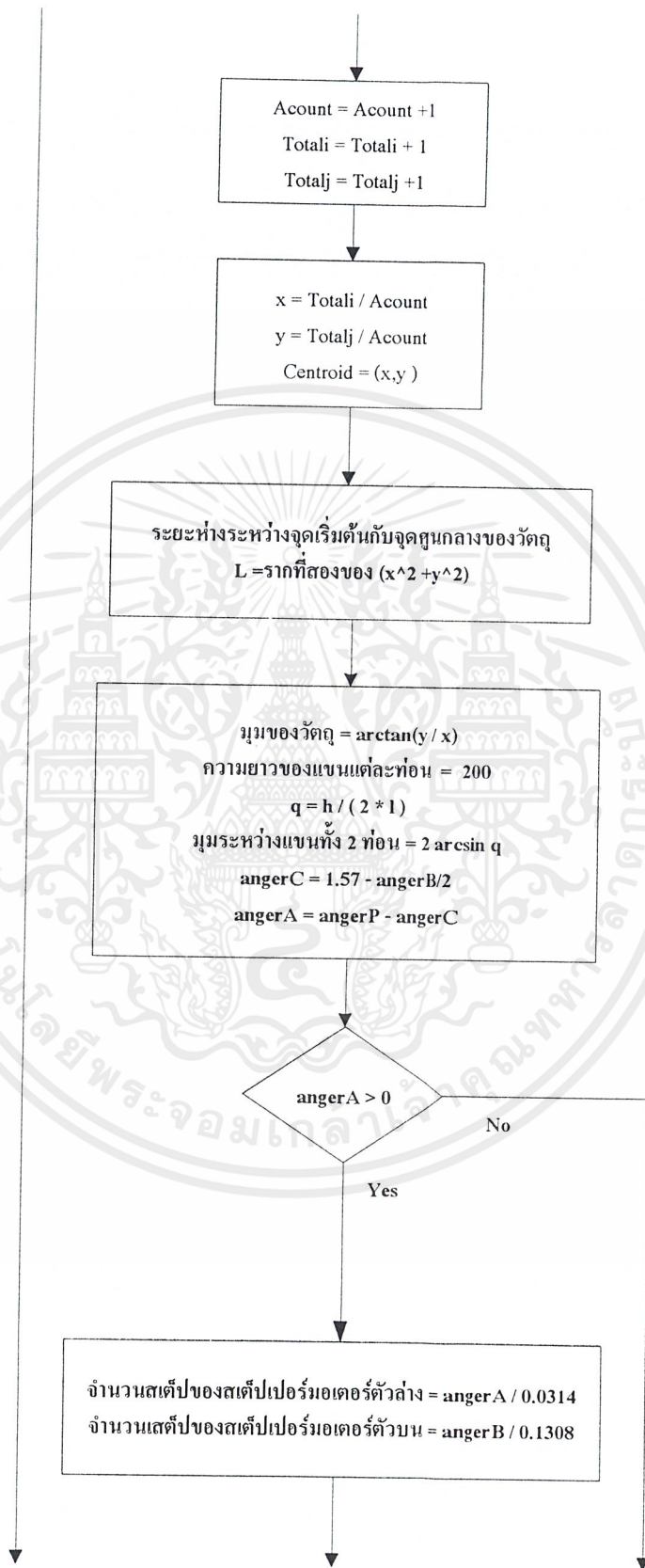


รูปที่ 3.5 โฟลว์ชาร์ตของโปรแกรมการประมวลผลภาพของวัตถุสีดำ

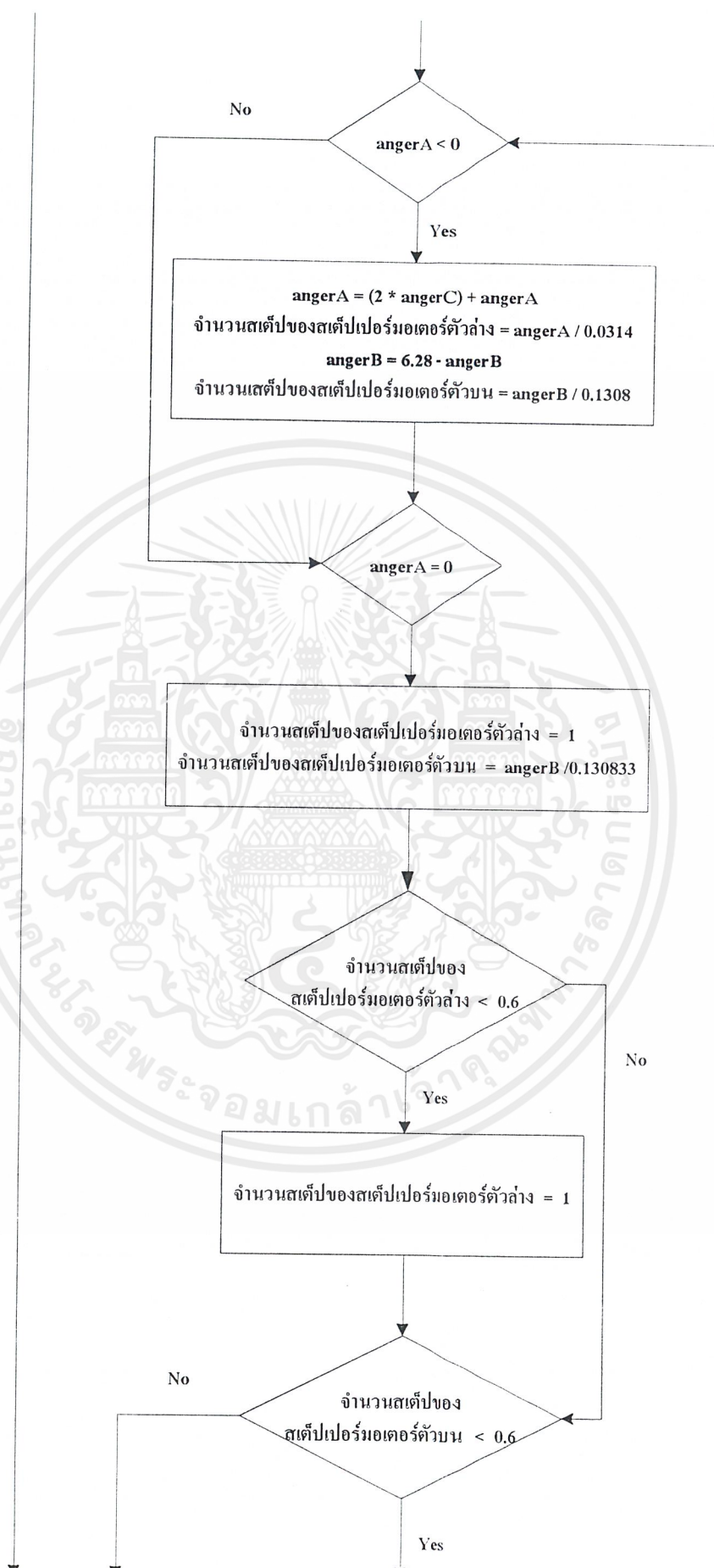
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและ222 อ่างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



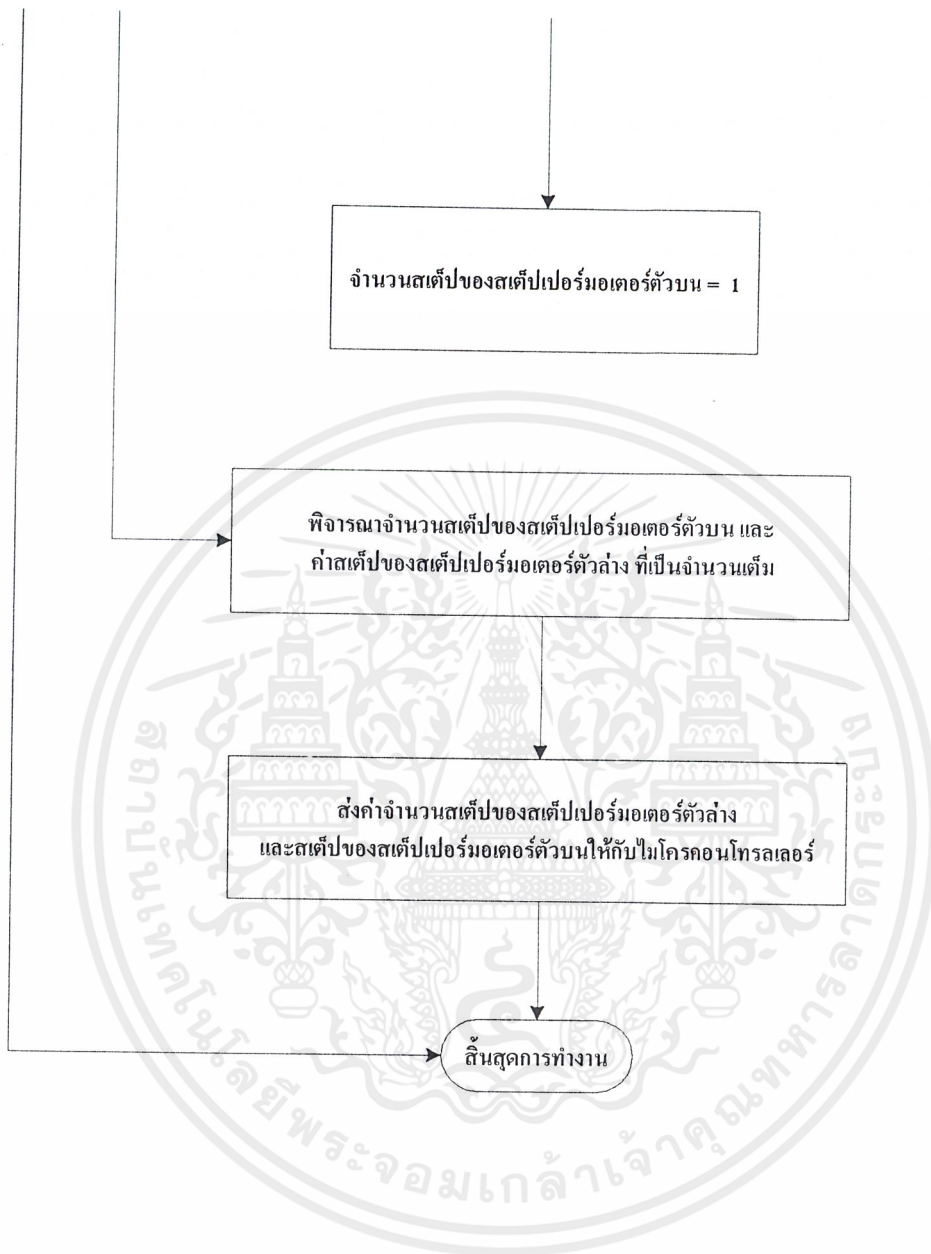
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาแล23้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและ24 อ่างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

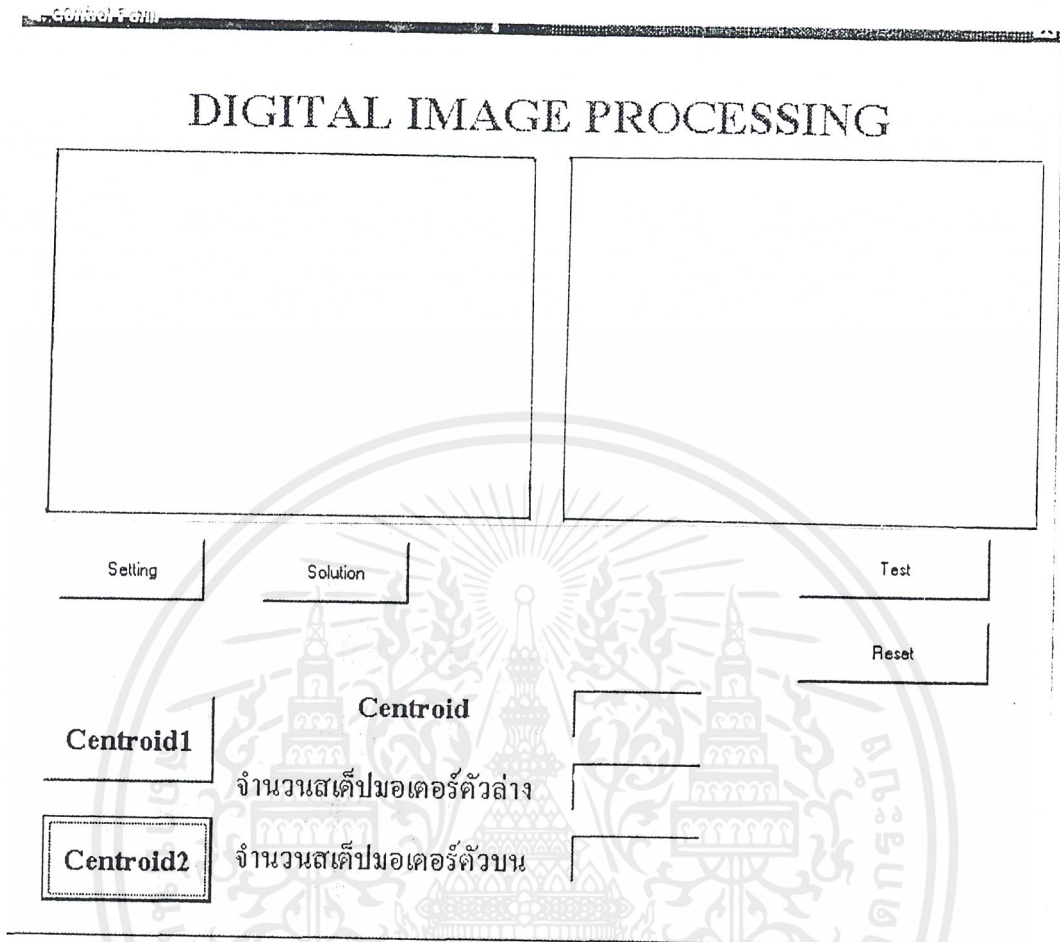


เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาแล25องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.6 โฟลว์ชาร์ตของโปรแกรมการประมวลผลภาพของวัตถุโทนสีส้ม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาแล266องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

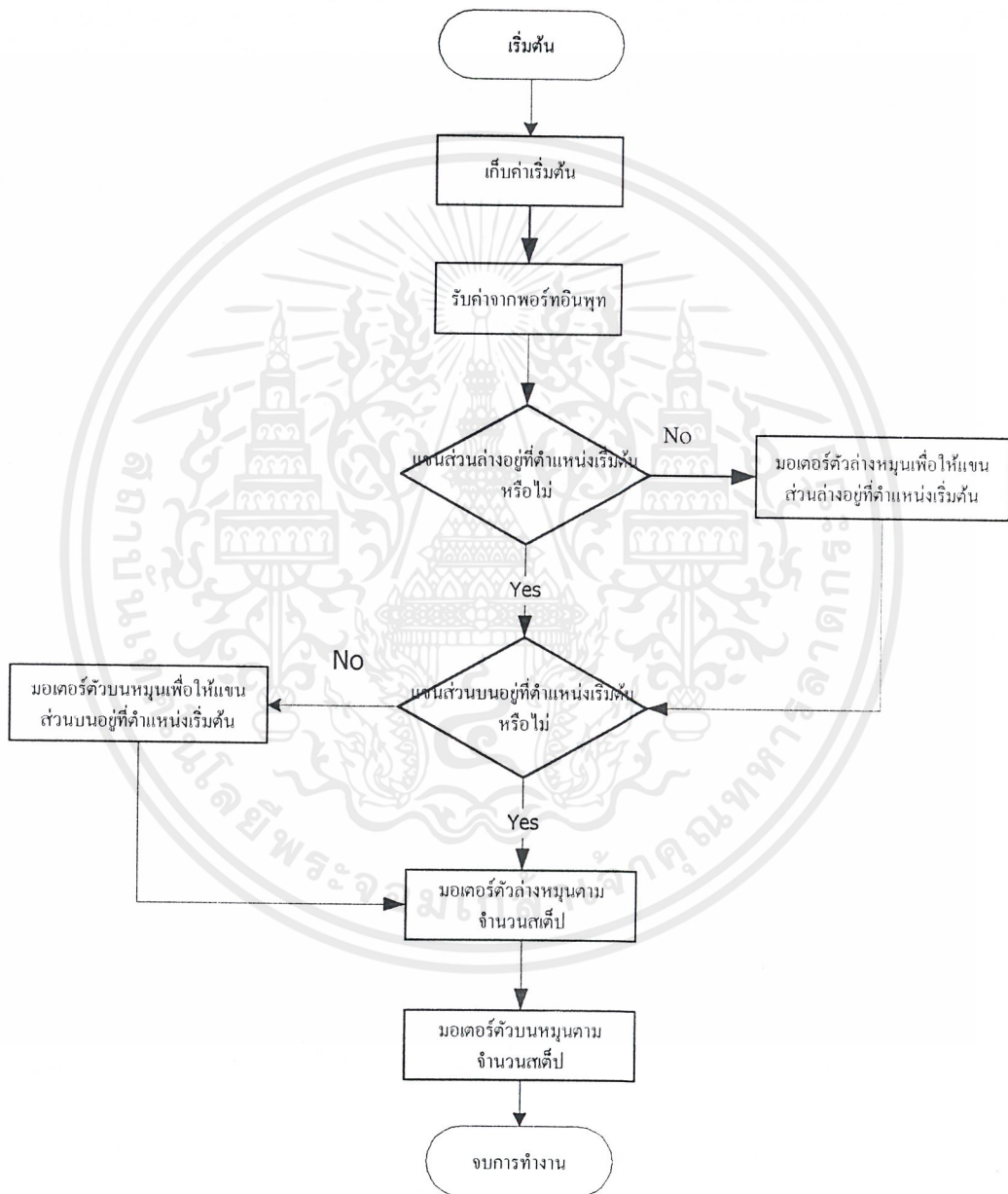


รูปที่ 3. 7 แสดงฟอร์มของส่วนควบคุม (Control Form) ของโปรแกรมประมวลผลภาพดิจิทัล

- Setting** : ใช้กำหนดค่าคุณสมบัติต่างๆของกล้องดิจิตอลวิดีโอ เช่น ค่าความสว่าง
- Solution** : ใช้กำหนดค่าขนาดของรูปที่ถ่ายด้วยกล้องดิจิตอลวิดีโอ เช่น ขนาด 240*320
- Test** : ใช้ทดสอบการทำงานของสแต็ปเปอร์มอเตอร์แต่ละตัวว่าสามารถหมุนได้ตามจำนวนสแต็ปที่ต้องการหรือไม่
- Reset** : ใช้สั่งให้สแต็ปเปอร์มอเตอร์แต่ละตัวหมุนให้แขนกลพับที่ตำแหน่งเริ่มต้น
- Centroid1** : เมื่อกดปุ่มนี้โปรแกรม microsoft visual basic จะทำการประมวลผลภาพของวัตถุสีดำ
- Centroid2** : เมื่อกดปุ่มนี้โปรแกรม microsoft visual basic จะทำการประมวลผลภาพของวัตถุสีส้ม

3.1.2 โปรแกรมของไมโครคอนโทรลเลอร์ MCS-51

โปรแกรมของไมโครคอนโทรลเลอร์ MCS-51 เป็นโปรแกรมภาษาแอสเซมบลี ซึ่งควบคุมการทำงานของแขนกล สามารถเขียนไฟล์ชาร์ต แสดงการทำงานได้ดังรูปที่ 3.8 และสำหรับตัวโปรแกรมแสดงในภาคผนวก



รูปที่ 3.8 ไฟล์ชาร์ตของโปรแกรมไมโครคอนโทรลเลอร์ MCS-51 ควบคุมแขนกล

3.2 ส่วนของฮาร์ดแวร์

3.2.1 สเต็ปเปอร์มอเตอร์

สเต็ปเปอร์มอเตอร์ใช้ในการเคลื่อนที่เชิงกลไปยังตำแหน่งของวัตถุ ซึ่งสเต็ปเปอร์มอเตอร์แต่ละตัวจะถูกควบคุมการหมุนโดยวงจรขับเคลื่อนสเต็ปเปอร์มอเตอร์และไมโครคอนโทรลเลอร์ซึ่งรับค่าจำนวนสเต็ปการหมุนมาจากโปรแกรมประมวลผลภาพ

โดยในโครงการนี้ต้องใช้สเต็ปเปอร์มอเตอร์แบบยูนิโพลาร์ 4 เฟส จำนวน 2 ตัว ซึ่งควบคุมการเคลื่อนที่แบบ 2 เฟส ได้แก่

สเต็ปเปอร์มอเตอร์ตัวล่าง (ตัวที่ฐาน)



รูปที่ 3.9 แสดงสเต็ปเปอร์มอเตอร์ตัวล่าง

จากรูปที่ 3.9 เป็นสเต็ปเปอร์มอเตอร์แบบยูนิโพลาร์ 4 เฟส 6 สาย มีค่าสเต็ปของการหมุนประมาณ 1.8 องศาต่อสเต็ป

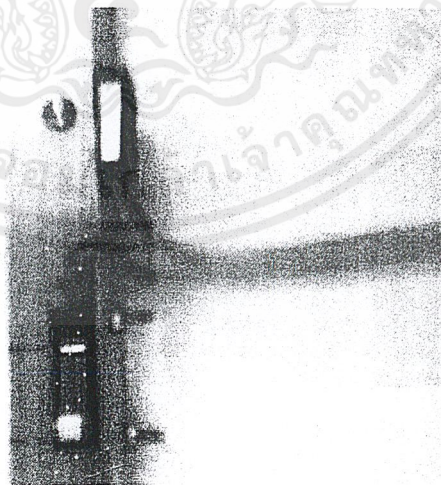
สเต็ปเปอร์มอเตอร์ตัวบน (ตัวที่อยู่ระหว่างแขนท่อนบนและล่าง)



รูปที่ 3.10 แสดงสเต็ปเปอร์มอเตอร์ตัวบน

จากรูปที่ 3.10 เป็นสเต็ปเปอร์มอเตอร์แบบยูนิโพลาร์ 4 เฟส 5 สาย มีค่าสเต็ปของการหมุนประมาณ 7.5 องศาต่อสเต็ป

3.2.2 ลิ้มิตสวิตช์ (Limit Switch)



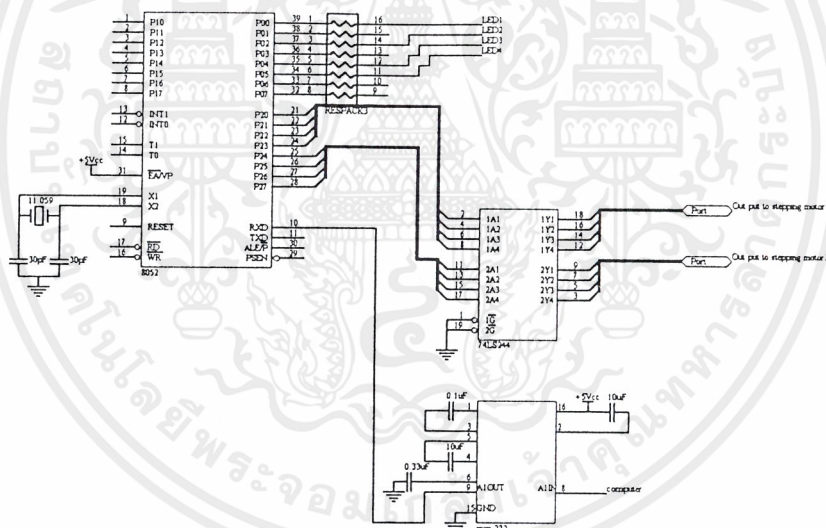
รูปที่ 3.11 แสดงการติดตั้ง ลิ้มิตสวิตช์

จากรูปที่ 3.11 ลิ้มิตสวิตช์ที่ใช้ 2 ตัว เป็นชนิดปกติเปิด(normally open) นั่นคือ ปกติจะมีค่าสถานะเป็น “1” แต่เมื่อกดจะมีค่าสถานะเป็น “0” โดยจะต่อขา GND ลงกราวด์ และต่อขา NO ของ ลิ้มิตสวิตช์ตัวบนและตัวล่าง เข้ากับขา P1.3 และ P1.5 ของไอซี 89C52 ตามลำดับ

สำหรับโครงการนี้เราต่อลิ้มิตสวิตช์เพื่อตรวจสอบว่าแขนกลแต่ละท่อนเคลื่อนที่มาอยู่ที่ ตำแหน่งเริ่มต้นหรือไม่ ก่อนที่จะเริ่มเคลื่อนที่ไปยังตำแหน่งวัตถุที่ต้องการ เมื่อได้รับค่าข้อมูลที่ส่งมาจากโปรแกรมประมวลผลภาพ

3.2.3 วงจรไมโครคอนโทรลเลอร์ MCS-51

วงจรมิโครคอนโทรลเลอร์ MCS-51 จะมีชิพไอซี 89C52 เป็นส่วนประกอบหลักใช้ควบคุมการทำงานของสตีปเปอร์มอเตอร์ซึ่งต้องโปรแกรมก่อนโดยโปรแกรมที่เขียนลงชิพไอซี 89C52 เป็นโปรแกรมภาษาแอสเซมบลี

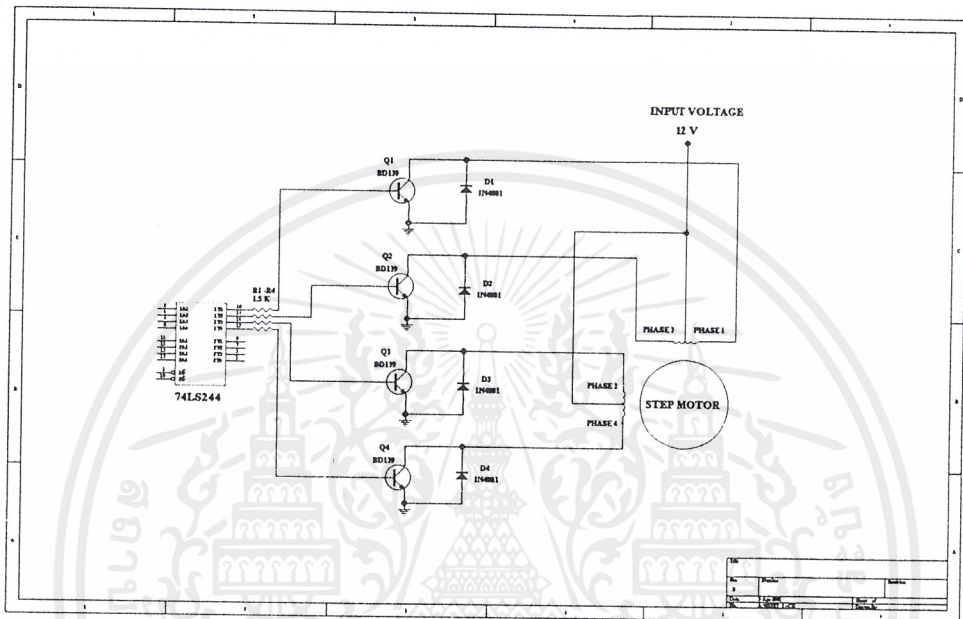


รูปที่ 3.12 แสดงวงจรมิโครโปรเซสเซอร์ MCS-51

จากรูปที่ 3.12 ตัวไอซี 89C52 ที่ขา 21-28 (P2.0-P2.7) ต่อกับ 74LS244 ซึ่งเป็นตัวบัฟเฟอร์ต่อกับวงจรขับเคลื่อนมอเตอร์ทั้ง 2 ตัว และที่ขา 10(RXD) ต่อกับ MAX-232 เพื่อการรับส่งข้อมูลแบบอนุกรมระหว่างไมโครคอนโทรลเลอร์และคอมพิวเตอร์ โดยคอมพิวเตอร์จะส่งข้อมูลเข้าที่ขา 8(AIN) ของMAX-232และไมโครคอนโทรลเลอร์จะรับข้อมูลเข้าที่ขา 10(RXD) ของ MAX-232

3.2.4 วงจรขับสเต็ปเปอร์มอเตอร์

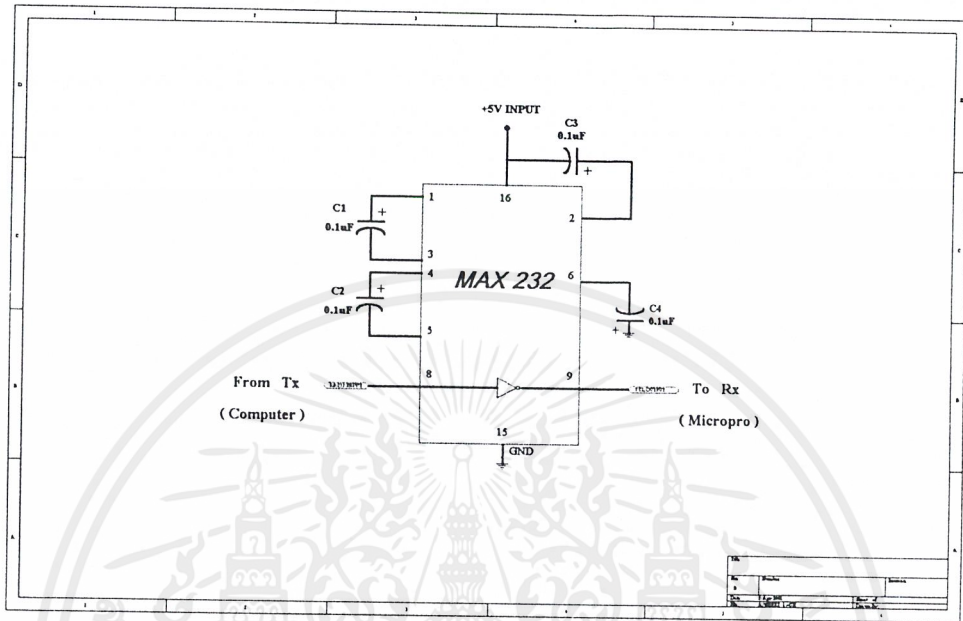
เนื่องจากโครงงานนี้ใช้สเต็ปเปอร์มอเตอร์ 2 ตัวจึงต้องมีวงจรขับ 2 วงจรซึ่งมีลักษณะการต่อเหมือนกัน วงจรขับสเต็ปเปอร์มอเตอร์แสดงดังรูปที่ 3.13



รูปที่ 3.13 วงจรขับสเต็ปเปอร์มอเตอร์

จากรูปที่ 3.13 จะได้ว่าพัลส์ควบคุมมอเตอร์ที่ได้นั้นจะถูกกำหนดโดยลำดับการทำงานของโปรแกรมภายในไมโครคอนโทรลเลอร์ MCS-51 และจะไปกระตุ้นให้ทรานซิสเตอร์ Q1-Q4 ทำงาน ทำให้ขดลวดมีกระแสไหลผ่าน เกิดการเหนี่ยวนำได้ โรเตอร์เคลื่อนไปยังอีกที่หนึ่งได้ทำให้สเต็ปเปอร์มอเตอร์หมุนโดยมีไดโอด D1-D4 ทำหน้าที่ป้องกันกระแสย้อนกลับอันเกิดจากการยุบตัวของสนามแม่เหล็กของสเต็ปเปอร์มอเตอร์ซึ่งอาจทำอันตรายให้แก่ทรานซิสเตอร์ได้

3.2.5 วงจรรับส่งข้อมูลแบบอนุกรม

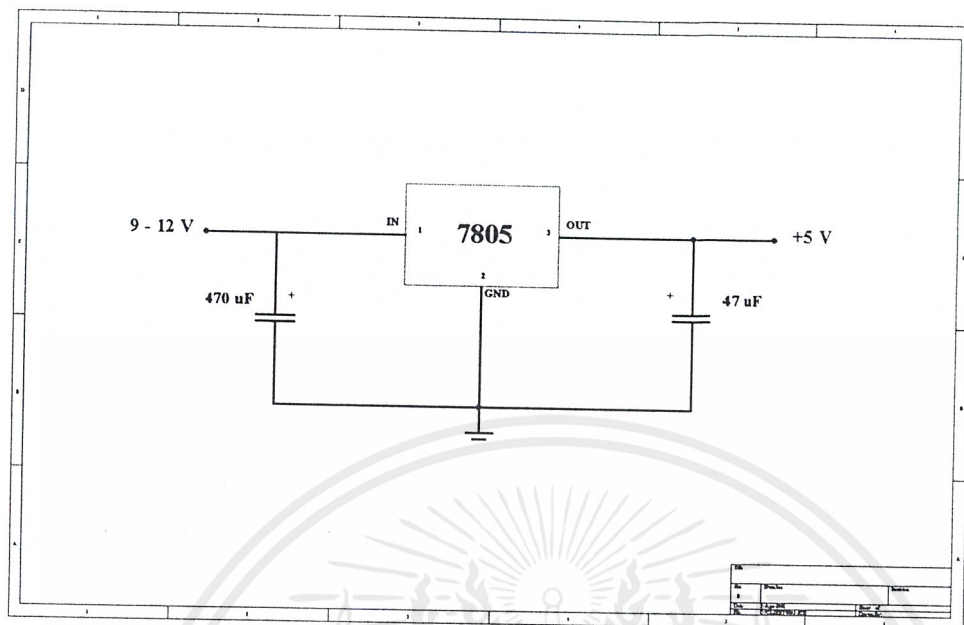


รูปที่ 3.14 แสดงวงจรรับส่งข้อมูลแบบอนุกรม

จากรูปที่ 3.14 ในการรับส่งข้อมูลระหว่างคอมพิวเตอร์และไมโครคอนโทรลเลอร์ ได้เลือกใช้การรับส่งข้อมูลแบบอนุกรม เพราะสัญญาณจากคอมพิวเตอร์เป็นมาตรฐาน RS-232 (-12V ถึง 12V) แต่สัญญาณของไมโครคอนโทรลเลอร์เป็นแบบ TTL (0 ถึง 5V) ทำให้ต้องมีตัวแปลงเพื่อแปลงจากมาตรฐาน RS232 ไปเป็น TTL (อินพุท) และจาก TTL ไปเป็นมาตรฐาน RS232 (เอาต์พุท) ซึ่งเลือกใช้วงจรรวมเบอร์ MAX232 มาเป็นตัวแปลง

3.2.6 วงจรรักษาระดับแรงดัน (regulator voltage circuit)

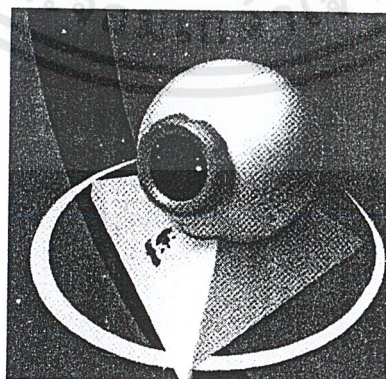
จากรูปที่ 3.15 เป็นวงจรที่ปรับแรงดันขนาด 9-24 โวลต์ให้เป็นแรงดันขนาด 5 โวลต์ เพื่อนำไปใช้เป็นไฟเลี้ยงสำหรับวงจรไมโครคอนโทรลเลอร์ และไอซี 74LS244 ที่ต่อเป็นบัฟเฟอร์สำหรับวงจรขับสเต็ปมอเตอร์



รูปที่ 3.15 วงจรปรับขนาดแรงดัน

3.2.7 กล้องดิจิทัลวิดีโอ

กล้องดิจิทัลวิดีโอที่ใช้ในโครงการนี้เป็นกล้องแบบ กล้องอินเทอร์เน็ตวิดีโอ (Internet Video Camera) ซึ่งสามารถเชื่อมต่อกับคอมพิวเตอร์โดยผ่านทางพอร์ต USB และภาพที่ได้เป็นภาพ RGB 24 บิต โดยเรากำหนดขนาดภาพเท่ากับ 240×320 พิกเซล



รูปที่ 3.16 แสดงกล้องดิจิทัลวิดีโอ

3.2.8 การออกแบบแขนกล ฐานและขาตั้งกล้อง

แขนกลที่สร้างขึ้นทำด้วยอลูมิเนียมแท่งเหลี่ยม โดยสร้างแขนกลให้มี 2 ท่อนและ 2 ข้อต่อกำหนดให้แขน 2 ท่อนมีความยาวเท่ากันเพื่อความสะดวกต่อการหามุมการเคลื่อนที่ของแขน โดยแขนแต่ละท่อนมีความยาวเท่ากับ 43 เซนติเมตร ที่ข้อต่อยึดระหว่างแขน 2 ท่อน(ท่อนบนและท่อนล่าง) ยึดด้วยน็อตที่ยึดติดกับสเต็ปเปอร์มอเตอร์ตัวบน ซึ่งมีค่าสเต็ปของการหมุนประมาณ 7.5 องศาต่อสเต็ป และแขนท่อนล่างยึดติดกับฐานและสเต็ปเปอร์มอเตอร์ตัวล่าง ซึ่งมีค่าสเต็ปของการหมุน 1.8 องศาต่อสเต็ป โดยแขนท่อนบนสามารถหมุนได้ 360 องศา ส่วนแขนท่อนล่างสามารถหมุนได้ 180 องศา

ฐาน เป็นแผ่นไม้สีขาวยาวขนาดกว้าง-ยาวด้านละ 65 เซนติเมตร ใช้เป็นฐานยึดติดกับแขนกลและขาตั้งกล้อง

ขาตั้งกล้อง ทำด้วยอลูมิเนียมแท่งฉาก ความสูง 120 เซนติเมตร เพื่อติดกล้องวิดีโอให้สามารถถ่ายภาพวัตถุที่อยู่บนฐานได้ครอบคลุมทุกตำแหน่ง



รูปที่ 3.17 แสดงแขนกล ฐานและขาตั้งกล้อง

บทที่ 4

การทดลองและผลการทดลอง

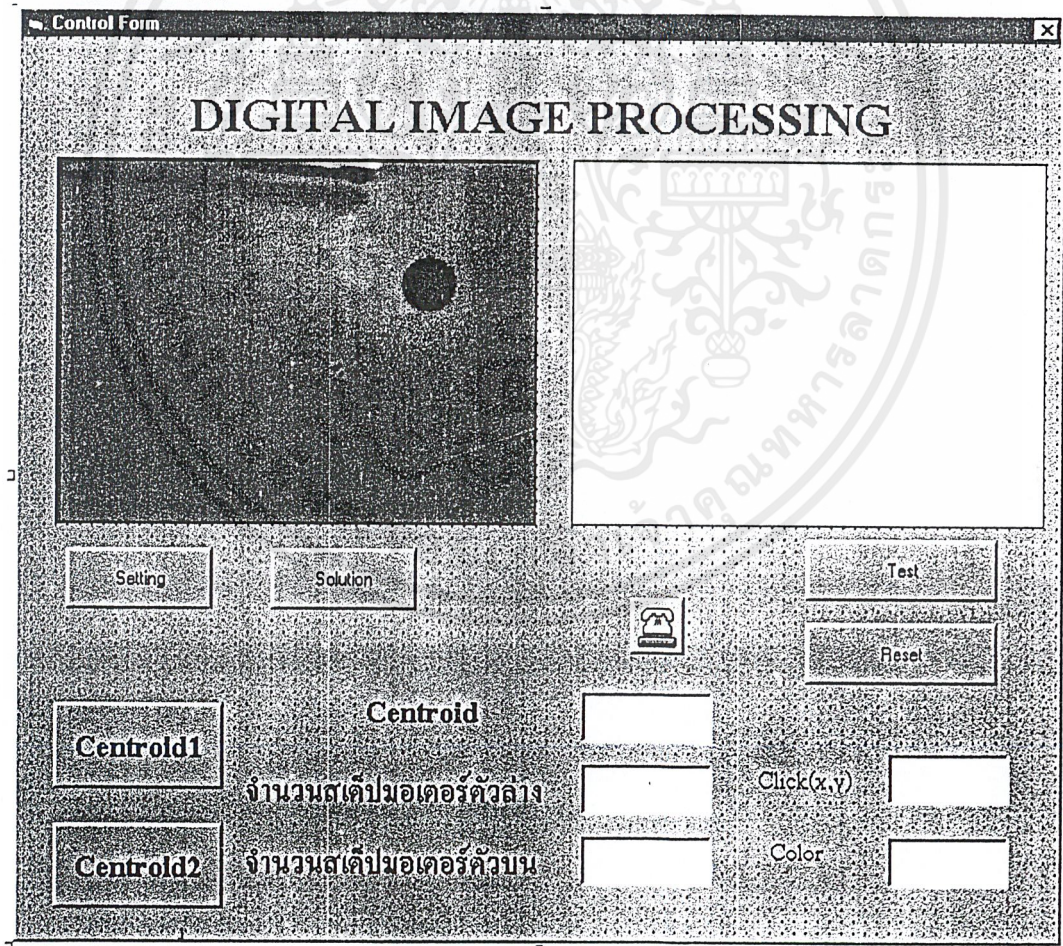
หลังจากที่ได้ทำการเชื่อมต่อส่วนของฮาร์ดแวร์และซอฟต์แวร์ต่างๆที่ได้ทำการออกแบบไว้เข้าด้วยกันแล้ว ก็ได้ทำการทดลองและแสดงผลการทดลองที่ Control Form ของโปรแกรม microsoft visual basic บนหน้าจคอมพิวเตอร์ ดังนี้

ขั้นที่ 1

การทดลอง

ต่อกล้องดิจิทัลวิดีโอเข้ากับพอร์ต USB ของคอมพิวเตอร์แล้วทำการถ่ายภาพวัตถุบนพื้นสีขาว และดึงภาพจากกล้องเข้ามาทำการประมวลผลใน โปรแกรม microsoft visual basic โดยตรง

ผลการทดลอง



รูปที่ 4.1 Control Form แสดงผลการทดลองขั้นที่ 1

จากรูปที่ 4.1 ภาพทางซ้ายมือแสดงภาพที่ดึงจากกล้องเข้ามาประมวลผลโดยตรง

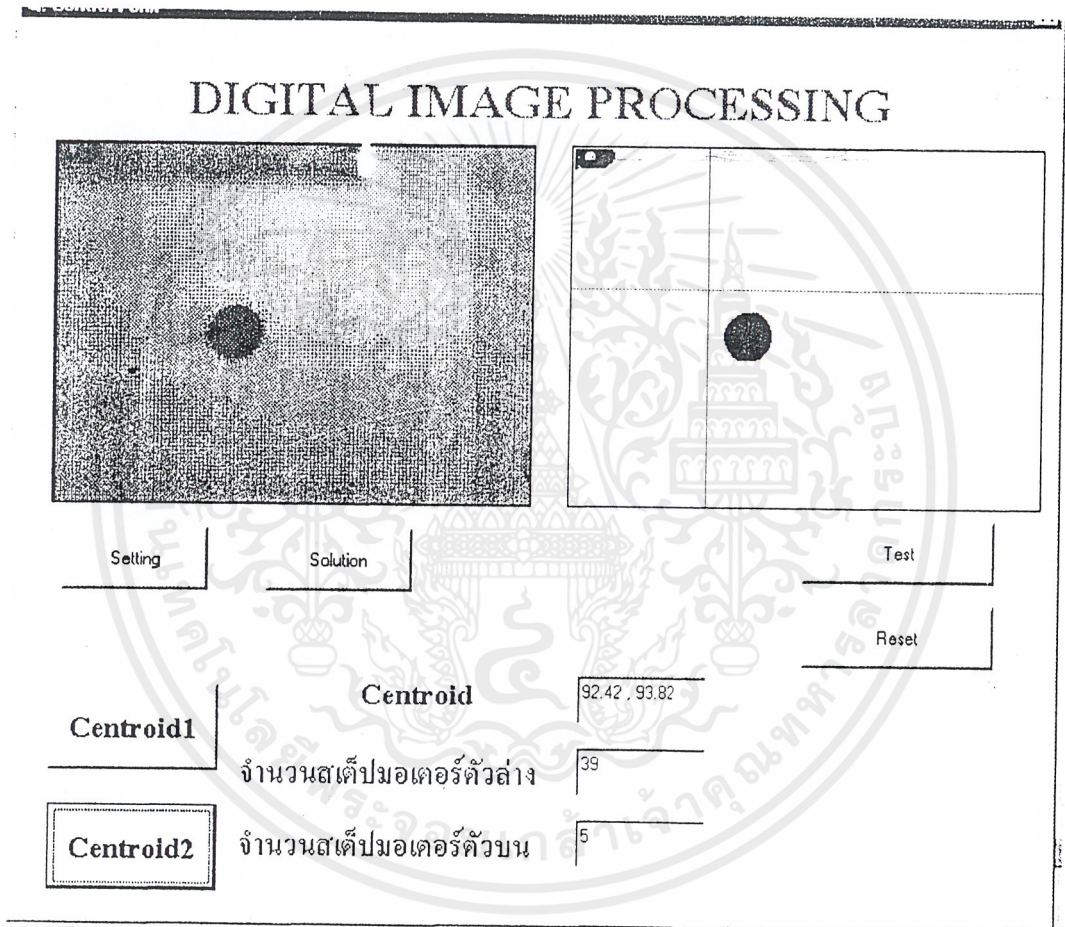
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาแล36องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ขั้นที่ 2

การทดลอง

ทำการประมวลผลภาพที่ถ่ายด้วยกล้องดิจิทัลวิดีโอ โดยการแปลงภาพสีให้กลายเป็นภาพขาวดำ แล้วหาค่าจุดศูนย์กลาง(Centroid)ของภาพสีดำ และจำนวนสเต็ปการหมุนของสเต็ปเปอร์มอเตอร์ทั้งตัวบนและล่าง

ผลการทดลอง



รูปที่ 4.2 Control Form แสดงผลการทดลองขั้นที่ 2

จากรูปที่ 4.2 ภาพทางซ้ายมือ แสดงภาพที่ถ่ายด้วยกล้องดิจิทัลวิดีโอ

ภาพทางขวามือแสดงภาพที่แปลงให้เป็นภาพขาวดำ และจุดที่เส้นตรงแนวแกน X-Y ตัดกัน คือตำแหน่งจุดศูนย์กลางของวัตถุ ซึ่งจะเห็นว่าค่าที่ได้ผิดพลาดไม่ตรงกับตำแหน่งจุดศูนย์กลางของวัตถุทรงกลมที่ต้องการ เนื่องจากผลของวัตถุที่ไม่ต้องการและเงาของวัตถุที่เกิดขึ้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา 37 ต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ส่งผลให้ค่าจำนวนสตีปการหมุนของมอเตอร์แต่ละตัวผิดพลาดด้วย กรณีนี้เซนกลก็ไม่สามารถเคลื่อนที่ไปยังตำแหน่งวัตถุที่ต้องการได้ เราจึงต้องทำการแก้ไขปัญหานี้ต่อไป

ขั้นที่ 3

การทดลอง

เพื่อลดปัญหาเรื่องวัตถุที่ไม่ต้องการและเงาที่เกิดขึ้น จึงทำการแก้ไขโปรแกรมการประมวลผลภาพดิจิทัลใหม่ โดยถ้าภาพที่ถ่ายด้วยกล้องดิจิทัล(ภาพทางขวามือ)เป็นภาพโทนีสัมเท่านั้น ภาพทางขวามือจึงจะเป็นสีดำ แล้วนำไปคำนวณหาค่าจุดศูนย์กลางและค่าจำนวนสตีปการหมุนของมอเตอร์แต่ละตัวต่อไป

ผลการทดลอง

Centroid	Value	
Centroid1	จำนวนสตีป มอเตอร์ตัวล่าง	38
Centroid2	จำนวนสตีปมอเตอร์ตัวบน	37

รูปที่ 4.3 Control Form แสดงผลการทดลองขั้นที่ 3

จากรูปที่ 4.3 ภาพทางขวามือแสดงตำแหน่งจุดศูนย์กลางวัตถุทรงกลมได้ถูกต้อง นั่นคือโปรแกรมประมวลผลภาพดิจิทัลทำงานได้ถูกต้องแล้ว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาแล38 ้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ขั้นที่ 4

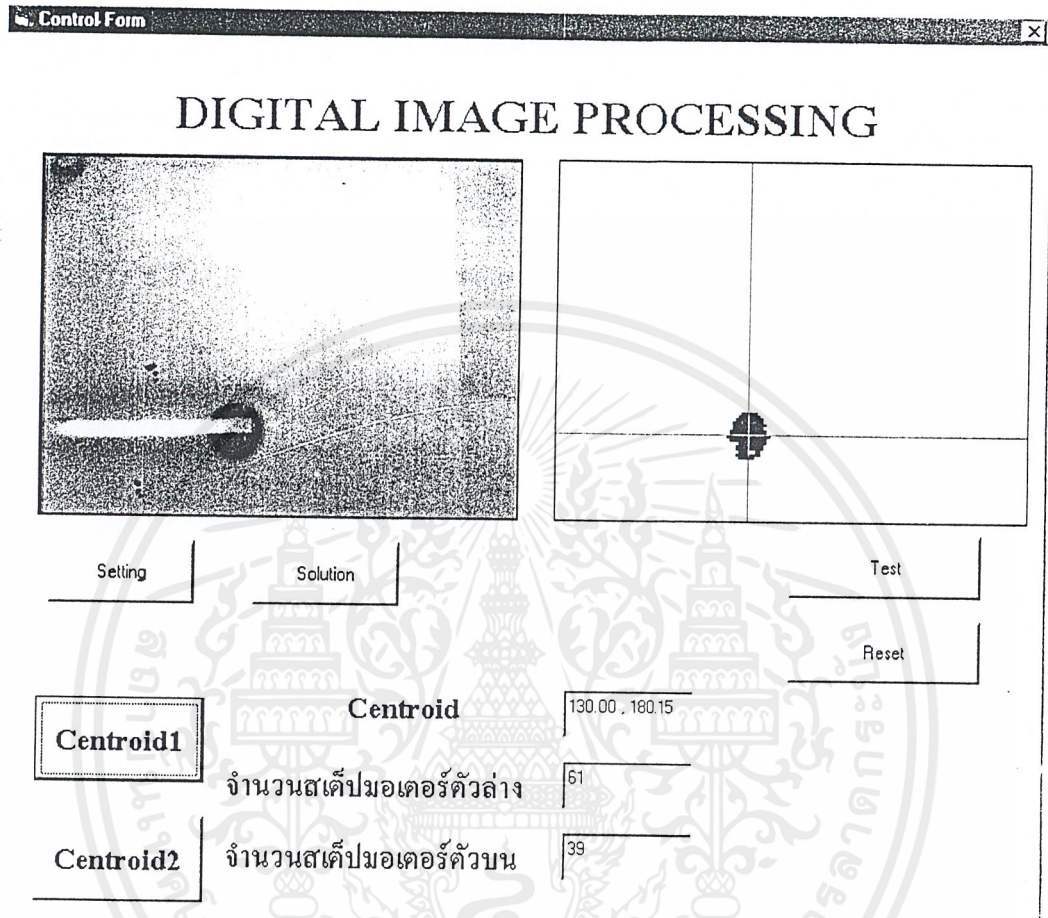
การทดลอง

หลังจากทดลองขั้นที่ 1-ขั้นที่ 3 พบว่าโปรแกรมประมวลผลภาพดิจิทัลทำงานได้ถูกต้องแล้วและทำการต่อวงจรอิเล็กทรอนิกส์ต่างๆตามที่ได้ออกแบบไว้เสร็จแล้วจึงส่งค่าจำนวนสตีปการหมุนของมอเตอร์แต่ละตัวที่ได้จากโปรแกรมการประมวลผลภาพดิจิทัลผ่านทางพอร์ตอนุกรม RS-232 มายังไมโครคอนโทรลเลอร์ซึ่งได้ใส่โปรแกรมแอสเซมบลีควบคุมการเคลื่อนที่ของสตีปเปอร์มอเตอร์แต่ละตัวไว้แล้ว และเมื่อได้รับค่าจำนวนสตีปการหมุน วงจรขับสตีปเปอร์มอเตอร์ก็จะขับให้สตีปเปอร์มอเตอร์แต่ละตัวเคลื่อนที่เพื่อให้แขนกลแต่ละท่อนเคลื่อนที่ไปยังตำแหน่งวัตถุที่ถูกต้องได้

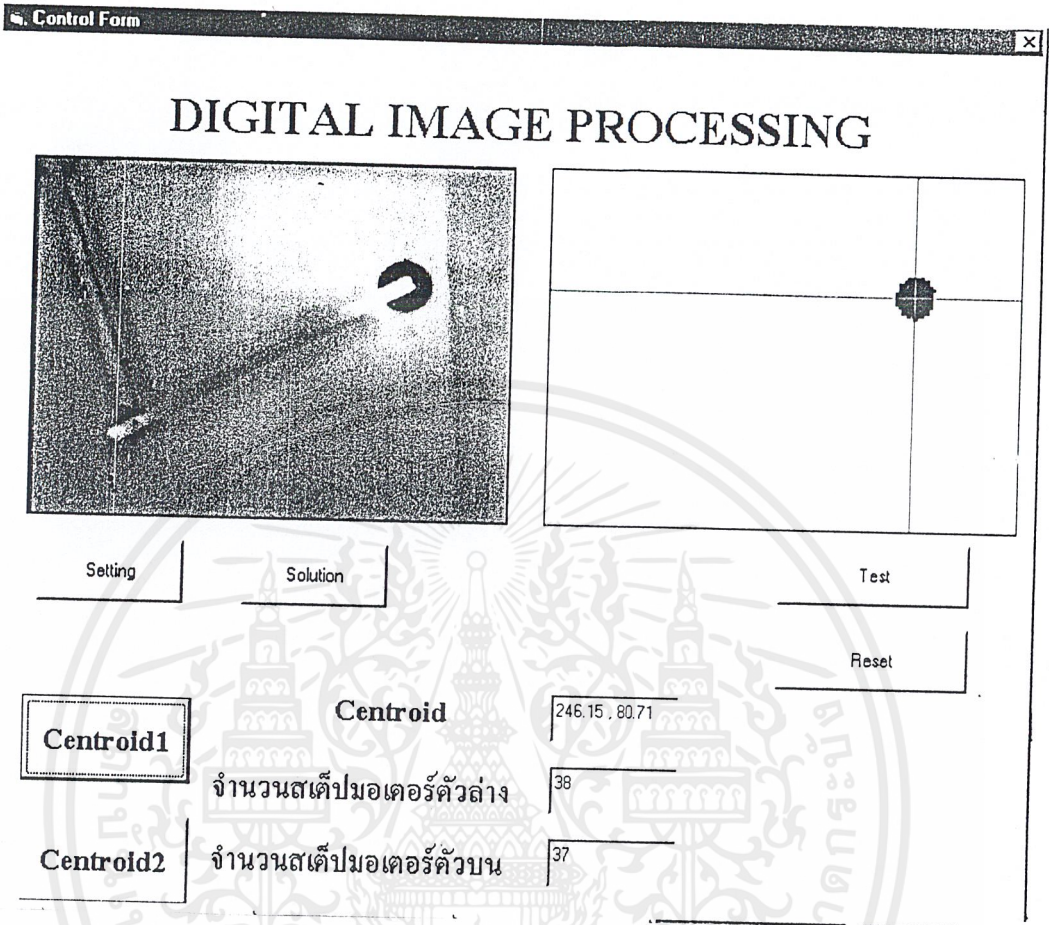
ขั้นตอนการทดลองการเคลื่อนที่ของแขนกล

1. กดปุ่ม run โปรแกรมประมวลผลภาพ
2. กดปุ่ม Reset เพื่อพับแขนกลให้มาอยู่ที่ตำแหน่งเริ่มต้น
3. กดปุ่ม Centroid 1 โปรแกรมประมวลผลภาพจะทำการคำนวณและแสดงค่าตำแหน่ง Centroid และจำนวนสตีปการหมุนของมอเตอร์แต่ละตัวที่ Control Form แล้วส่งค่าไปยังไมโครคอนโทรลเลอร์ โดยอัตโนมัติ
4. ไมโครคอนโทรลเลอร์ควบคุมการเคลื่อนที่ของแขนกล โดยสตีปเปอร์มอเตอร์ตัวล่างจะเริ่มหมุนเท่ากับจำนวนสตีปเปอร์มอเตอร์ซึ่งแสดงที่ Control Form ให้แขนกลทั้งท่อนบนและล่างซึ่งอยู่ในแนวเดียวกันเคลื่อนที่ออกจากจุดเริ่มต้นในทิศทางทวนเข็มนาฬิกา แล้วสตีปเปอร์มอเตอร์ตัวล่างก็จะเริ่มหมุนเท่ากับจำนวนสตีปเปอร์มอเตอร์ซึ่งแสดงที่ Control Form ให้แขนท่อนบนซึ่งเดิมอยู่ในแนวเดียวกับแขนท่อนล่างเคลื่อนที่ออกในทิศทางตามเข็มนาฬิกา เพื่อให้ปลายแขนชี้ที่ตำแหน่งจุดศูนย์กลางของวัตถุทรงกลมที่ต้องการได้
5. เมื่อเปลี่ยนตำแหน่งวัตถุบนฐาน ก็ทำการทดลองตามขั้นตอน 2-4 ตามลำดับ

ผลการทดลอง

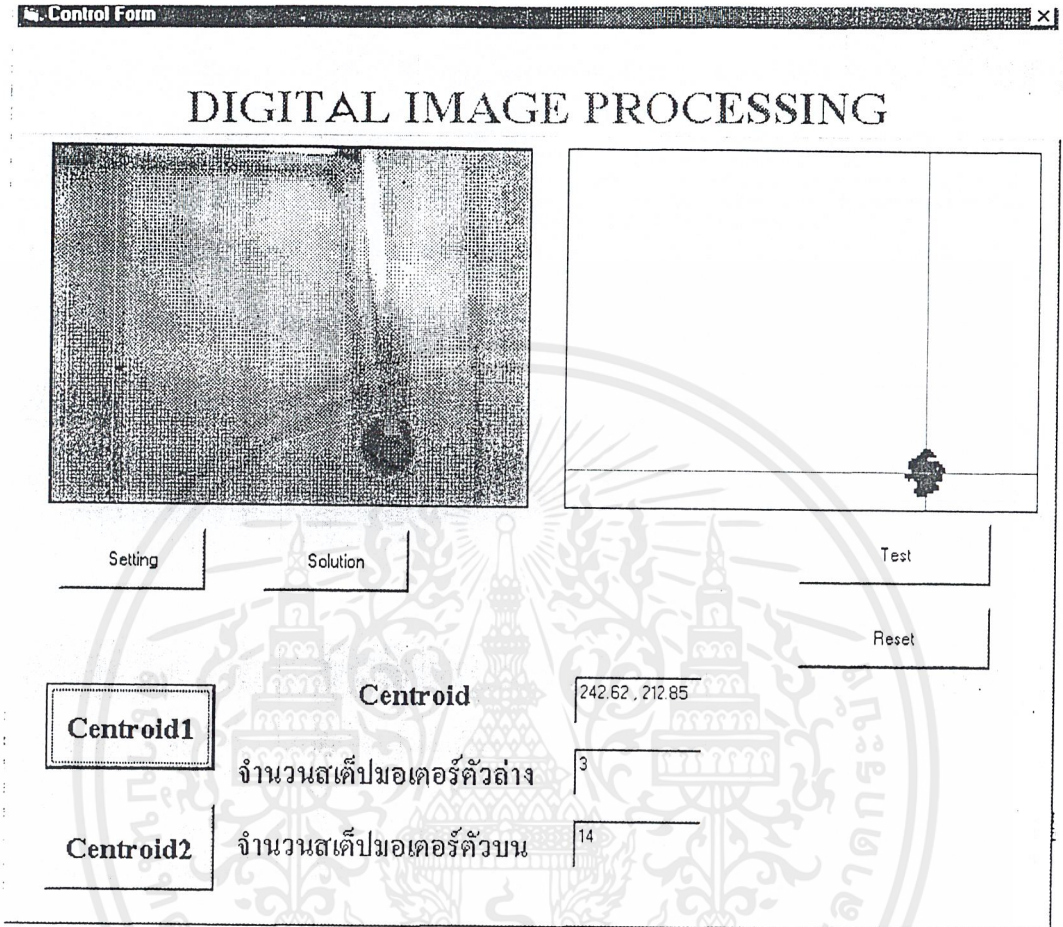


รูปที่ 4.4(ก) Control Form แสดงผลการทดลองขั้นที่ 4

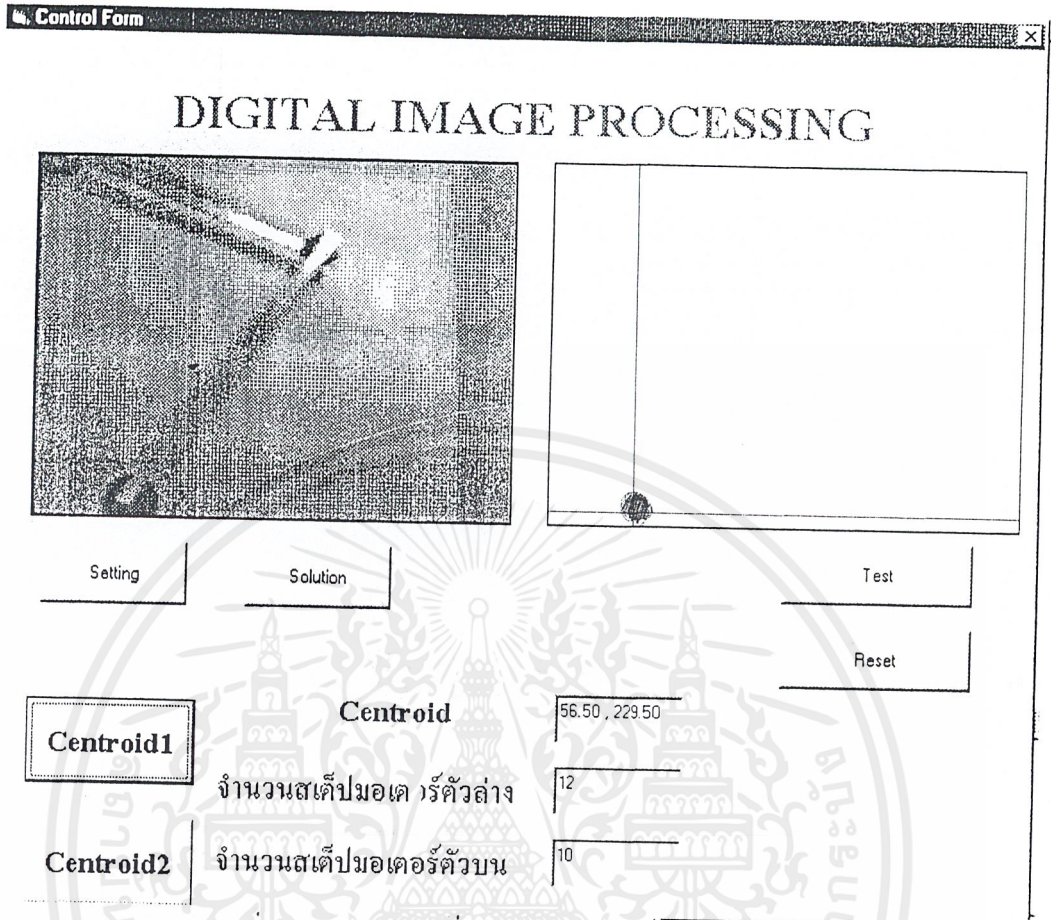


รูปที่ 4.4(ข) Control Form แสดงผลการทดลองขั้นที่ 4

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาแล41องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.4(ก) Control Form แสดงผลการทดลองขั้นที่ 4



รูปที่ 4.4(ง) Control Form แสดงผลการทดลองขั้นที่ 4

รูปที่ 4.3(ก) – รูปที่ 4.3(ง) รูปทางซ้ายมือแสดงลักษณะการเคลื่อนที่ของแขนมายังตำแหน่งวัตถุที่ต้องการ ส่วนรูปทางขวามือแสดงภาพที่ได้จากการประมวลผลภาพดิจิทัลของภาพที่ดึงมาจากกล้อง โดยตรง และแสดงตำแหน่งจุดศูนย์กลางวัตถุทรงกลม

สรุปผลการทดลอง

เมื่อทดลองวางวัตถุทรงกลมโทสนีสัมไว้ที่ตำแหน่งต่างๆ บนฐานที่กำหนด แล้วทำการทดลองให้แขนกลเคลื่อนที่ไปยังที่ตำแหน่งวัตถุนั้น พบว่าแขนกลสามารถเคลื่อนที่ไปยังตำแหน่งจุดศูนย์กลางของวัตถุที่ต้องการได้อย่างถูกต้อง

สำหรับกรณีที่เกิดความผิดพลาด แขนกลจะชี้ที่ตำแหน่งคลาดเคลื่อนจากจุดศูนย์กลางวัตถุไม่เกิน 2 เซนติเมตร ตามแนวแขนท่อนบน

บทที่ 5

บทวิจารณ์และสรุป

การทำโครงการนี้ต้องทำการศึกษาความรู้พื้นฐานที่จำเป็น ได้แก่ การประมวลผลภาพ ดิจิตอล วิธีการรับส่งข้อมูลแบบอนุกรม การควบคุมการทำงานของไมโครคอนโทรลเลอร์ MCS-51 และสตีปเปอร์มอเตอร์

ประโยชน์ที่ได้รับจากการทำโครงการนี้

1. ได้นำความรู้ ทฤษฎี และหลักการต่างๆ ที่ได้เรียนมา ประยุกต์ใช้ในการทำโครงการนี้ เช่น ไมโครคอนโทรลเลอร์ตระกูล MCS-51 และหลักการทำงานของสตีปเปอร์มอเตอร์
2. ได้รับประโยชน์จากการศึกษาค้นคว้าหาข้อมูลเพิ่มเติมในส่วนที่เกี่ยวข้องกับโครงการ เช่น การประมวลผลภาพดิจิตอล
3. เกิดทักษะในการเขียนโปรแกรมภาษาแอสเซมบลี และโปรแกรม microsoft visual basic 6 ซึ่งสามารถนำไปประยุกต์ใช้ในการทำงานได้ต่อไป
4. ได้รู้ความรู้เกี่ยวกับอุปกรณ์อิเล็กทรอนิกส์ และวงจรต่างๆ รวมทั้งหลักการทำงานของอุปกรณ์แต่ละตัว
5. ได้ฝึกฝนการทำงานเป็นทีม ทำให้เกิดความร่วมมือในการคิด การแก้ปัญหาาร่วมกัน
6. การวางแผนการทำงานอย่างเป็นระบบ มีการศึกษา ลงมือปฏิบัติ และการแก้ปัญหา

ปัญหาที่เกิดขึ้นระหว่างทำโครงการ

1. ขาดความรู้และประสบการณ์ในการทำงาน ทำให้ต้องใช้เวลาในการศึกษาหาความรู้เพิ่มเติม
2. สำหรับการทำโครงการในช่วงแรกนั้น ไม่สามารถหาวิธีการดึงภาพจากกล้องเข้ามาประมวลผลในโปรแกรม microsoft visual basic 6 ได้โดยตรงซึ่งทำให้ใช้เวลาในการประมวลผลภาพค่อนข้างมาก
3. เกิดปัญหาจากการออกแบบในส่วนของฮาร์ดแวร์ และการเลือกใช้อุปกรณ์ในการทำโครงการ
4. เกิดปัญหาจากการออกแบบในส่วนของซอฟต์แวร์ สำหรับการเขียนโปรแกรมในการควบคุมแขนกล

เนื่องจากปัญหาที่กล่าวมาข้างต้นและข้อจำกัดของเวลา ทำให้ผู้จัดทำโครงการไม่สามารถจัดทำโครงการให้ลุล่วงได้ตามเป้าหมายเดิมที่วางไว้ คือให้แขนกลสามารถจับวัตถุที่ตำแหน่งที่

ต้องการได้ ซึ่งจากโครงการนี้ผู้จัดทำสามารถทำให้แขนกลรู้ตำแหน่งของวัตถุและเคลื่อนที่ไปยังตำแหน่งนั้นได้เท่านั้น

แนวทางการพัฒนา

หลังจากสามารถทำให้แขนกลรู้ตำแหน่งของวัตถุและเคลื่อนที่ไปยังตำแหน่งนั้นได้แล้วนั้น การทำให้แขนกลสามารถหยิบวัตถุในตำแหน่งนั้นได้ ก็ต้องอาศัยการพัฒนาเพิ่มเติมทั้งในส่วนของ ฮาร์ดแวร์และซอฟต์แวร์ ดังนี้

1. ส่วนของฮาร์ดแวร์ จะต้องสร้างส่วนของแขนกลเพิ่มเติมในส่วนที่ใช้สำหรับหยิบวัตถุ โดยสร้างเป็นมือจับหรือหนีบต่อจากปลายแขนท่อนบน และจะต้องเพิ่มสเต็ปเปอร์มอเตอร์ รวมทั้งวงจรขับสเต็ปเปอร์มอเตอร์ และวงจรอิเล็กทรอนิกส์อื่นๆที่ใช้เพื่อเคลื่อนที่มือจับให้สามารถหยิบวัตถุที่ต้องการได้
2. ส่วนของซอฟต์แวร์ จะต้องทำการสร้างโปรแกรม Microsoft Visual Basic เพื่อหาเส้นขอบของวัตถุ และส่วนของโปรแกรมภาษาแอสเซมบลีก็ต้องเพิ่มเติมในส่วนที่ใช้สำหรับควบคุมการทำงานของมือจับให้สามารถหยิบวัตถุที่ต้องการได้

ภาคผนวก



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

DIGITAL IMAGE PROCESSING PROGRAM

Option Explicit

Private Declare Function BitBlt Lib "gdi32" (ByVal hDestDC As Long, ByVal x As Long, ByVal y As Long, ByVal nWidth As Long, ByVal nHeight As Long, ByVal hSrcDC As Long, ByVal xSrc As Long, ByVal ySrc As Long, ByVal dwRop As Long) As Long

Private Declare Function GetPixel Lib "gdi32" (ByVal hdc As Long, ByVal x As Long, ByVal y As Long) As Long

Private Declare Function SetPixel Lib "gdi32" (ByVal hdc As Long, ByVal x As Long, ByVal y As Long, ByVal crColor As Long) As Long

Private Sub CmdSetting_Click()

ezVidCap1.ShowDlgVideoSource

End Sub

Private Sub CmdSol_Click()

ezVidCap1.ShowDlgVideoFormat

End Sub

Private Sub Command1_Click()

MSComm1.Output = Chr(Text6.Text)

MSComm1.Output = Chr(Text7.Text)

End Sub

Private Sub Command2_Click()

MSComm1.Output = Chr(0)

End Sub

Private Sub Form_Load()

MSComm1.PortOpen = True

End Sub

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

Private Sub Picture2_MouseUp(Button As Integer, Shift As Integer, x As Single, y As Single)
    Text1.Text = x
    Text2.Text = y
    Text3.Text = GetPixel(Picture2.hdc, x, y)
End Sub

```

```

Private Sub Picture3_MouseUp(Button As Integer, Shift As Integer, x As Single, y As Single)
    Text1.Text = Format(x, "00.00") & " , " & Format(y, "00.00")
    Text3.Text = GetPixel(Picture3.hdc, x, y)
End Sub

```

```

Private Sub Cmdcalcentroid1_click()
    Dim m, n, totalj, totali, ACount, i, j As Long
    Dim color1 As Long
    Dim x, y, z, P, d, a As Single
    Dim Drg%, Drb%, Dgb%, R%, g%, b As Integer
    Dim angerA As Single, angerB As Single, angerC As Single, angerP As Single, l As Single, h As
    Single, s As Single, t As Single, q As Single

    BitBlt Picture2.hdc, 0, 0, 320, 240, Picture1.hdc, 0, 0, vbSrcCopy
    ACount = 0
    totali = 0
    totalj = 0
    For i = 1 To 317
        For j = 1 To 237
            color1 = GetPixel(Picture2.hdc, i, j)
            Call ColorRGB(color1, R, g, b)

            Drg = R - g
            Drb = R - b
            Dgb = g - b

```

```

If Drg > 40 And Drb > 40 Then
    color1 = 0
    ACount = ACount + 1
    m = i
    n = j
    totali = totali + m
    totalj = totalj + n
Else
    color1 = 16777215
End If
SetPixel Picture3.hdc, i, j, color1

Next j
Next i
'On Error Resume Next
If ACount = 0 Then ACount = 1
x = totali / ACount
y = totalj / ACount
Text4.Text = Format(x, "00.00") & " , " & Format(y, "00.00")
Me.Picture3.Line (x, 0)-(x, Picture3.Height), vbBlue
Me.Picture3.Line (0, y)-(Picture3.Width, y), vbRed
h = Sqr(x * x + y * y)
angerP = Atn(y / x)
l = 200
q = h / (2 * l)
angerB = 2 * Atn(q / Sqr(-q * q + 1))
angerC = 1.57 - angerB / 2
angerA = angerP - angerC
If angerA > 0 Then
    s = angerA / 0.0314

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

t = angerB / 0.1308333
ElseIf angerA < 0 Then
    angerA = (2 * angerC) + angerA
    s = angerA / 0.0314
    angerB = 6.28 - angerB
    t = angerB / 0.1308333
Else
    s = 1
    t = angerB / 0.1308333
End If
If s < 0.6 Then s = 1
If t < 0.6 Then t = 1
s = Round(s, 0)
t = Round(t, 0)
Text6.Text = s
MSComm1.Output = Chr(s)
Text7.Text = t
MSComm1.Output = Chr(t)
End Sub

Private Function ColorRGB(color1 As Long, R As Integer, g As Integer, b As Integer)
    R = color1 Mod 256
    b = color1 \ 65536
    g = (color1 \ 256) Mod 256
End Function

Private Sub Cmdcalcentroid2_click()
    Dim m, n, totalj, totali, ACount, color1, i, j As Long
    Dim x, y, z, P, d, a As Single
    Dim Drg%, Drb%, Dgb%, R%, g%, b As Integer

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Dim angerA As Single, angerB As Single, angerC As Single, angerP As Single, l As Single, h As Single, s As Single, t As Single, q As Single

BitBlt Picture2.hdc, 0, 0, 320, 240, Picture1.hdc, 0, 0, vbSrcCopy

ACount = 0

totali = 0

totalj = 0

For i = 1 To 317

For j = 1 To 237

color1 = GetPixel(Picture2.hdc, i, j)

If color1 > 6000000 Then

color1 = 16777215

Else

ACount = ACount + 1

m = i

n = j

totali = totali + m

totalj = totalj + n

color1 = 0

End If

SetPixel Picture3.hdc, i, j, color1

Next j

Next i

'On Error Resume Next

If ACount = 0 Then ACount = 1

x = totali / ACount

y = totalj / ACount

Text4.Text = Format(x, "00.00") & " , " & Format(y, "00.00")

Me.Picture3.Line (x, 0)-(x, Picture3.Height), vbBlue

Me.Picture3.Line (0, y)-(Picture3.Width, y), vbRed

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

h = Sqr(x * x + y * y)
angerP = Atn(y / x)
l = 200
q = h / (2 * l)
angerB = 2 * Atn(q / Sqr(-q * q + 1))
angerC = 1.57 - angerB / 2
angerA = angerP - angerC
If angerA > 0 Then
    s = angerA / 0.0314
    t = angerB / 0.1308333
ElseIf angerA < 0 Then
    angerA = (2 * angerC) + angerA
    s = angerA / 0.0314
    angerB = 6.28 - angerB
    t = angerB / 0.1308333
Else
    s = 1
    t = angerB / 0.1308333
End If
If s < 0.6 Then s = 1
If t < 0.6 Then t = 1
s = Round(s, 0)
t = Round(t, 0)
Text6.Text = s
MSComm1.Output = Chr(s)
Text7.Text = t
MSComm1.Output = Chr(t)
End Sub

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ASSEMBLY PROGRAM

```
UP_MOTOR EQU P1.5
LO_MOTOR EQU P1.3
DATA_D EQU 2FH
DATA_U EQU 2EH
CARRY EQU 31H
UP_STATE EQU 20H
LO_STATE EQU 21H
STATE EQU 22H
```

```
ORG 0000H

START: MOV TMOD,#20H
MOV SCON,#52H
MOV TH1,#0FDH
SETB TR1
MOV UP_STATE,#11001100B
MOV LO_STATE,#11001100B

LOWER: MOV A,LO_STATE
RR A
MOV LO_STATE,A
ORL A,#11110000B
MOV STATE,UP_STATE
ORL STATE,#00001111B
ANL A,STATE
MOV P2,A
CALL DELAY
JB LO_MOTOR,LOWER
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

UPPER:MOV  A,UP_STATE
        RL   A
        MOV  UP_STATE,A
        ORL  A,#00001111B
        MOV  STATE,LO_STATE
        ORL  STATE,#11110000B
        ANL  A,STATE
        MOV  P2,A
        CALL DELAY1
        JB   UP_MOTOR,UPPER
        MOV  P2,#11000000B
        MOV  UP_STATE,#11001100B
        CLR  RI
jim:    JNB  RI,jim
        MOV  A,SBUF
        MOV  DATA_D,A
        CLR  RI
        JNB  RI,$
        MOV  A,SBUF
        MOV  DATA_U,A

        MOV  R0,DATA_D
LO:    MOV  A,LO_STATE
        RL   A
        MOV  LO_STATE,A
        ORL  A,#11110000B
        MOV  STATE,UP_STATE
        ORL  STATE,#00001111B
        ANL  A,STATE
        MOV  P2,A

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

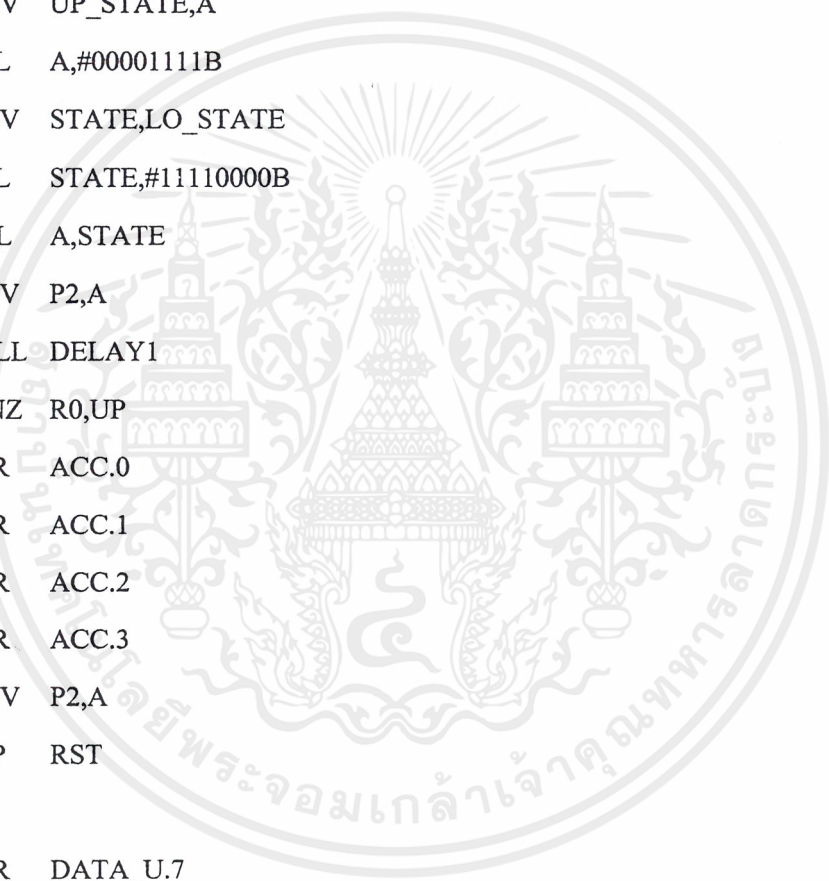
CALL DELAY
DJNZ R0,LO

JB DATA_U.7,UP_R
MOV R0,DATA_U
UP: MOV A,UP_STATE
RR A
MOV UP_STATE,A
ORL A,#00001111B
MOV STATE,LO_STATE
ORL STATE,#11110000B
ANL A,STATE
MOV P2,A
CALL DELAY1
DJNZ R0,UP
CLR ACC.0
CLR ACC.1
CLR ACC.2
CLR ACC.3
MOV P2,A
JMP RST

UP_R: CLR DATA_U.7
MOV R0,DATA_U
AAAA: MOV A,UP_STATE
RL A
MOV UP_STATE,A
ORL A,#00001111B
MOV STATE,LO_STATE
ORL STATE,#11110000B

```

~



```

ANL  A,STATE
MOV  P2,A
CALL DELAY1
DJNZ R0,AAAA
CLR  ACC.0
CLR  ACC.1
CLR  ACC.2
CLR  ACC.3
MOV  P2,A
JMP  RST

RST: CLR  RI
      JNB  RI,$
      MOV  A,SBUF
      CJNE A,#00H,RST
      JMP  LOWER

DELAY: MOV  R3,#4AH

DEL:  MOV  R5,#0FFH
      DJNZ R5,$
      DJNZ R3,DEL
      RET

DELAY1: MOV  R3,#7AH

DEL1:  MOV  R5,#0FFH
      DJNZ R5,$
      DJNZ R3,DEL1
      RET
      END

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางสัญลักษณ์ที่ใช้

สัญลักษณ์	ความหมาย
Rn	รีจิสเตอร์ภายใน R0 - R7
direct	รีจิสเตอร์ SFR และหน่วยความจำข้อมูลภายใน
@Ri	ค่าแอดเดรสหน่วยความจำภายใน อ้างแอดเดรส โดยอ้อมผ่าน รีจิสเตอร์ R0 หรือ R1
#data	ค่าคงที่ขนาด 8 บิต (ค่าจาก 0 - FF)
#data16	ค่าคงที่ขนาด 16 บิต (ค่าจาก 0 - FFFF)
addr11	ค่าตำแหน่งแอดเดรสจำนวน 11 บิต สำหรับคำสั่ง ACALL หรือ AJMP
addr16	ค่าตำแหน่งแอดเดรสจำนวน 16 บิต สำหรับคำสั่ง LCALL หรือ LJMP
rel	ค่าออฟเซต (offset) หรือค่าบอกความสัมพันธ์ (ค่าจาก -17 ถึง 18 ไบต์)
bit	ตำแหน่งบิตของหน่วยความจำภายในที่อ้างถึง ได้แบบบิต หรือ SFR
@DPTR	แอดเดรสของหน่วยความจำภายนอก หรือพอยน์เตอร์ โดยอ้อมผ่าน รีจิสเตอร์ DPTR

คำสั่งทางตรรก(Logical Operations)

รูปแบบคำสั่ง	ความหมาย	จำนวนไบต์	จำนวนแอมรีนโรเคิล
ANL A,Rn	AND ค่าใน Rn กับ A	1	1
ANL A,direct	AND ค่าในหน่วยความจำ direct กับ A	2	1
ANL A,@Ri	AND ค่าในหน่วยความจำที่เก็บใน Rn กับ A	1	1
ANL A,#data	AND ค่าคงที่ 8 บิตกับ A	2	1
ANL direct,A	AND ค่า A กับหน่วยความจำ direct	2	1
ANL direct,#data	AND ค่าคงที่ 8 บิตกับหน่วยความจำ direct	3	2
ORL A,Rn	OR ค่าใน Rn กับ A	1	1
ORL A,direct	OR ค่าในหน่วยความจำ direct กับ A	2	1
ORL A,@Ri	OR ค่าในหน่วยความจำที่เก็บอยู่ใน Rn กับ A	1	1
ORL A,#data	OR ค่าคงที่ 8 บิตกับ A	2	1
ORL direct,A	OR ค่า A กับหน่วยความจำ direct	2	1
ORL direct,#data	OR ค่าคงที่ 8 บิตกับหน่วยความจำ direct	3	2
XRL A,Rn	EX-OR ค่าใน Rn กับ A	1	1
XRL A,direct	EX-OR ค่าในหน่วยความจำ direct กับ A	2	1
XRL A,@Ri	EX-OR ค่าในหน่วยความจำที่เก็บอยู่ใน Ri กับ A	1	1
XRL A,#data	EX-OR ค่าคงที่ 8 บิตกับ A	2	1
XRL direct,A	EX-OR ค่า A กับหน่วยความจำ direct	2	1
XRL direct,#data	EX-OR ค่าคงที่ 8 บิตกับหน่วยความจำ direct	3	2
CLR A	ทำค่าใน A ให้เป็นศูนย์	1	1
CPL A	กลับค่าบิตใน A เป็นตรงข้ามทุกบิต	1	1
RL A	หมุนบิตใน A ไปทางซ้าย 1 บิตและบิต 0 มีค่า 0	1	1
RLC A	หมุนบิตใน A ไปทางซ้าย 1 บิต และบิต 0 เป็นค่าบิตที่อยู่ในแฟล็ก Carry	1	1
RR A	หมุนบิตใน A ไปทางขวา 1 บิต และบิต 0 เป็นค่าจากบิต 7	1	1
RRC A	หมุนบิตใน A ไปทางขวา 1 บิต และค่าจาก บิต 0 นำไปเก็บในแฟล็ก Carry และบิตที่อยู่ในแฟล็ก Carry เดิมจะย้ายมาเก็บในบิต 7	1	1
SWAP A	สลับค่าสวิตช์บนกับสวิตช์ล่างภายใน A	1	1

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

คำสั่งประมวลผลทางคณิตศาสตร์ (Arithmetic Operations)

รูปแบบคำสั่ง	ความหมาย	จำนวนไบต์	จำนวนแอมรีนไคเคิล
ADD A,Rn	บวกค่า Rn กับ A	1	1
ADD A,direct	บวกค่าในหน่วยความจำ direct กับ A	2	1
ADD A,@Ri	บวกค่าในหน่วยความจำที่เก็บอยู่ใน Ri กับ A	1	1
ADDC A,#data	บวกค่าคงที่ 8 บิตกับ A	2	1
ADDC A,Rn	บวกค่า Rn กับ A พร้อมแฟล็ก Carry	1	1
ADDC A,direct	บวกค่าในหน่วยความจำ direct กับ A พร้อมแฟล็ก Carry	2	1
ADDC A,@Ri	บวกค่าในหน่วยความจำที่เก็บอยู่ใน Ri กับ A พร้อมแฟล็ก Carry	1	1
ADDC A,#data	บวกค่าคงที่ 8 บิตกับ A พร้อมแฟล็ก Carry	2	1
SUBB A,Rn	ลบค่า Rn กับ A พร้อมแฟล็ก Borrow	1	1
SUBB A,direct	ลบค่าในหน่วยความจำ direct กับ A พร้อมแฟล็ก Borrow	2	1
SUBB A,@Ri	ลบค่าในหน่วยความจำที่เก็บอยู่ใน Ri กับ A พร้อมแฟล็ก Borrow	1	1
SUBB A,#data	ลบค่าคงที่ 8 บิตกับ A พร้อมแฟล็ก Borrow	2	1
INC A	เพิ่มค่าใน A	1	1
INC Rn	เพิ่มค่าใน Rn	1	1
INC direct	เพิ่มค่าในหน่วยความจำ direct	2	1
INC @Ri	เพิ่มค่าในหน่วยความจำที่เก็บอยู่ใน Ri	1	1
DEC A	ลดค่าใน A	1	1
DEC Rn	ลดค่าใน Rn	1	1
DEC direct	ลดค่าในหน่วยความจำ direct	2	1
DEC @Ri	ลดค่าในหน่วยความจำที่เก็บอยู่ใน Ri	1	1
INC DPTR	เพิ่มค่าใน DPTR	1	2
MUL: AB	คูณ A กับ B แล้วเก็บค่าใน A	1	4
DIV AB	หาร A ด้วย B แล้วเก็บค่าใน A	1	4
DA A	ทำ decimal adjust ค่าใน A	1	1

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

คำสั่งเคลื่อนย้ายข้อมูล (Data Transfer)

รูปแบบคำสั่ง	ความหมาย	จำนวนไบต์	จำนวนแอมรีนไรเคิล
MOV A,Rn	ย้ายข้อมูลจาก Rn ไป A	1	1
MOV A,direct	ย้ายข้อมูลจากหน่วยความจำ direct ไป A	2	1
MOV A,@Ri	ย้ายข้อมูลจากหน่วยความจำที่เก็บอยู่ใน Ri ไป A	1	1
MOV A,#data	ย้ายค่าคงที่ 8 บิตไปเก็บที่ A	2	1
MOV Rn,A	ย้ายข้อมูลจาก A ไป Rn	1	1
MOV Rn,direct	ย้ายข้อมูลจากหน่วยความจำ direct ไป Rn	2	2
MOV direct,A	ย้ายข้อมูลจาก A ไปยังหน่วยความจำ direct	2	1
MOV direct,Rn	ย้ายข้อมูลจาก Rn ไปยังหน่วยความจำ direct	2	2
MOV direct,direct	ย้ายข้อมูลระหว่างหน่วยความจำภายใน	3	2
MOV direct,@Ri	ย้ายข้อมูลจากหน่วยความจำที่เก็บอยู่ใน Ri ไปยังหน่วยความจำ direct	2	2
MOV direct,#data	ย้ายค่าคงที่ 8 บิตไปยังหน่วยความจำ direct	3	2
MOV @Ri,A	ย้ายข้อมูลใน A ไปยังหน่วยความจำที่เก็บอยู่ใน Ri	1	1
MOV @Ri,direct	ย้ายข้อมูลจากหน่วยความจำ direct ไปยังหน่วยความจำที่เก็บอยู่ใน Ri	2	2
MOV @Ri,#data	ย้ายค่าคงที่ 8 บิตไปยังหน่วยความจำที่เก็บอยู่ใน Ri	2	1
MOV DPTR,#data16	ย้ายค่าคงที่ 16 บิตไปยัง DPTR	3	2
MOVC A,@A+DPTR	ย้ายข้อมูลจากหน่วยความจำข้อมูลที่สัมพันธ์กับ DPTR ไปยัง A	1	2
MOVC A,@A+PC	ย้ายข้อมูลจากหน่วยความจำข้อมูลที่สัมพันธ์กับ PC ไปยัง A	1	2
MOVX A,@Ri	ย้ายข้อมูลจากหน่วยความจำที่เก็บอยู่ใน Ri ไปยัง A	1	2
MOVX A,@DPTR	ย้ายข้อมูลจากหน่วยความจำที่เก็บอยู่ใน DPTR ไปยัง A	1	2
MOVX @Ri,A	ย้ายข้อมูลที่เก็บอยู่ใน A ไปยังหน่วยความจำที่เก็บอยู่ใน Ri	1	2
MOVX @DPTR,A	ย้ายข้อมูลที่เก็บอยู่ใน A ไปยังหน่วยความจำที่เก็บอยู่ใน DPTR	1	2

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

PUSH	direct	ย้ายข้อมูลหน่วยความจำ direct ไปเก็บยัง stack	2	2
POP	direct	ย้ายข้อมูลจาก stack ไปยังหน่วยความจำ direct	2	2
XCH	A,Rn	แลกเปลี่ยนข้อมูลระหว่าง A กับ Rn	1	1
XCH	A,direct	แลกเปลี่ยนข้อมูลระหว่างหน่วยความจำ direct กับ A	2	1
XCH	A,@Ri	แลกเปลี่ยนข้อมูลระหว่างหน่วยความจำ ที่เก็บอยู่ใน Ri กับ A	1	1
XCHD	A,@Ri	แลกเปลี่ยนข้อมูลสลับบิตล่างจากหน่วยความจำ ที่เก็บอยู่ใน Ri กับ A	1	1

คำสั่งจัดการข้อมูลแบบบิต (Boolean Variable Manipulation)

รูปแบบคำสั่ง	ความหมาย	จำนวนไบต์	จำนวนแอมรินไรเคิล
CLR C	ทำค่าแฟล็ก Carry ให้เป็น 0	1	1
CLR bit	ทำค่า bit ให้เป็น 0	2	1
SETB C	ทำค่าแฟล็ก Carry ให้เป็น 1	1	1
SETB bit	ทำค่า bit ให้เป็น 1	2	1
CPL C	กลับค่าแฟล็ก Carry ให้เป็นตรงข้าม	1	1
CPL bit	กลับค่า bit ให้เป็นตรงข้าม	2	1
ANL C,bit	AND ค่า bit กับแฟล็ก Carry	2	2
ANL C,/bit	AND ค่าตรงข้ามของ bit กับแฟล็ก Carry	2	2
ORL C,bit	ORL ค่า bit กับแฟล็ก Carry	2	2
ORL C,/bit	ORL ค่าตรงข้ามของ bit กับแฟล็ก Carry	2	2
MOV C,bit	ย้ายค่า bit มายังแฟล็ก Carry	2	1
MOV bit,C	ย้ายค่าแฟล็ก Carry มายัง bit	2	2
JC rel	กระโดด ถ้าค่าแฟล็ก Carry เป็น 1	2	2
JNC rel	กระโดด ถ้าค่าแฟล็ก Carry เป็น 0	2	2
JB bit,rel	กระโดด ถ้าค่า bit เป็น 1	3	2
JNB bit,rel	กระโดด ถ้าค่า bit เป็น 0	3	2
JBC bit,rel	กระโดด ถ้าค่า bit เป็น 1 และเปลี่ยนค่า bit เป็น 0	3	2

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

คำสั่งควบคุมการทำงานโปรแกรม (Program and Machine Control)

รูปแบบคำสั่ง		ความหมาย	จำนวนไบต์	จำนวนแรมรีนไซเคิล
CLR	C	ทำค่าแฟล็ก Carry ให้เป็น 0	1	1
ACALL	addr 11	ไปทำโปรแกรมย่อยจากค่าแอดเดรส 11 บิต	2	2
LCALL	addr 16	ไปทำโปรแกรมย่อยจากค่าแอดเดรส 16 บิต	3	2
RET		คำสั่งสิ้นสุดการทำงานโปรแกรมย่อย	1	2
RETI		คำสั่งสิ้นสุดการทำงานโปรแกรมย่อย อินเตอร์รัปต์	1	2
AJMP	addr 11	กระโดดไปยังตำแหน่งจากค่าแอดเดรส 11 บิต	2	2
LJMP	addr 16	กระโดดไปยังตำแหน่งจากค่าแอดเดรส 16 บิต	3	2
SJMP	rel	กระโดดไปยังตำแหน่งที่สัมพันธ์กับตำแหน่งปัจจุบัน	2	2
JMP	@A+DPTR	กระโดดไปยังตำแหน่งที่สัมพันธ์กับ DPTR	1	2
JZ	rel	กระโดดไปยังตำแหน่งที่สัมพันธ์กับตำแหน่งปัจจุบัน ถ้าหากค่า A เป็นค่า 0	2	2
JNZ	rel	กระโดดไปยังตำแหน่งที่สัมพันธ์กับตำแหน่งปัจจุบัน ถ้าหากค่า A เป็นค่า 1	2	2
CJNE	A,direct,rel	เปรียบเทียบ A กับหน่วยความจำ direct และ กระโดดไปยังตำแหน่งที่สัมพันธ์กับตำแหน่งปัจจุบัน ถ้าค่าไม่เท่ากัน	3	2
CJNE	A,#data,rel	เปรียบเทียบ A กับค่าคงที่ และกระโดดไป ยังตำแหน่งที่สัมพันธ์กับตำแหน่งปัจจุบัน ถ้าค่าไม่เท่ากัน	3	2
CJNE	Rn,#data,rel	เปรียบเทียบ Rn กับค่าคงที่ และ กระโดดไป ยังตำแหน่งที่สัมพันธ์กับตำแหน่งปัจจุบัน ถ้าค่าไม่เท่ากัน	3	2
CJNE	@Ri,#data,rel	เปรียบเทียบค่าในหน่วยความจำที่เก็บใน Ri กับ ค่าคงที่และกระโดดไปยังตำแหน่งที่สัมพันธ์กับ ตำแหน่งปัจจุบัน ถ้าค่าไม่เท่ากัน	3	2
DJNZ	Rn,rel	ลดค่าใน Rn และกระโดดไปยังตำแหน่งที่ สัมพันธ์ กับตำแหน่งปัจจุบัน ถ้าค่าไม่เป็น 0	2	2
DJNZ	direct,rel	ลดค่าในหน่วยความจำ direct และกระโดด ไปยัง ตำแหน่งที่สัมพันธ์กับตำแหน่งปัจจุบัน ถ้าค่าไม่เป็น 0	3	2
NOP		ไม่มีการทำงานใดๆ เกิดขึ้น	1	1

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

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

MAX2220-MAX249

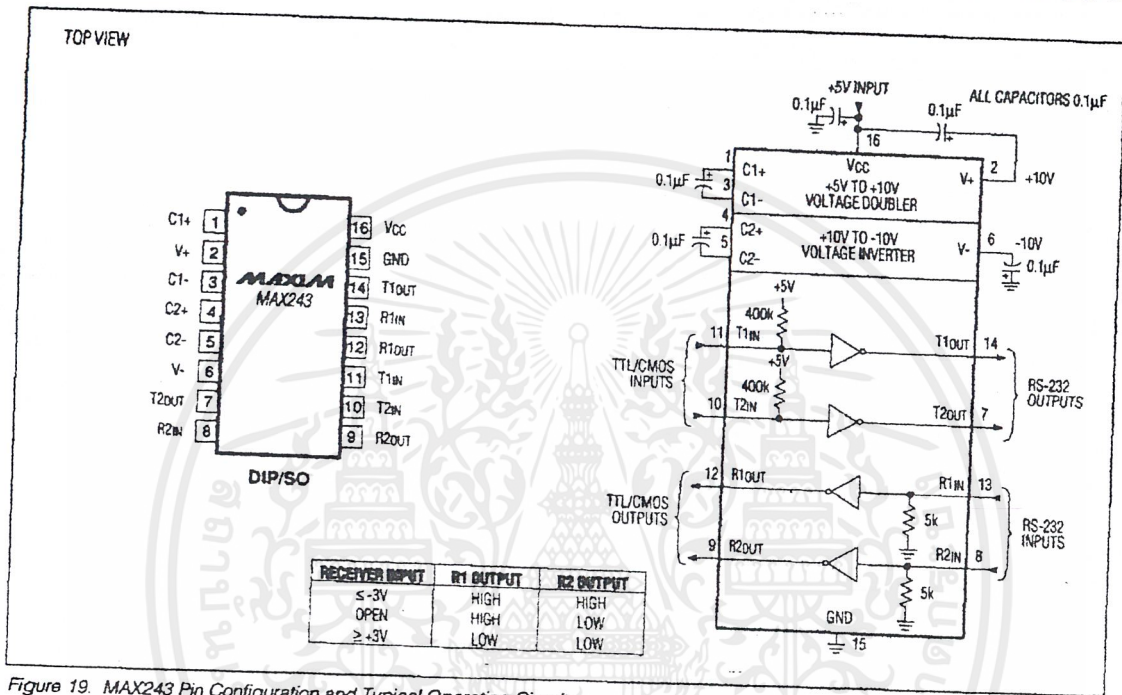


Figure 19. MAX243 Pin Configuration and Typical Operating Circuit

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

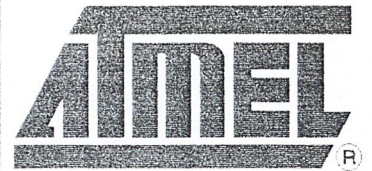
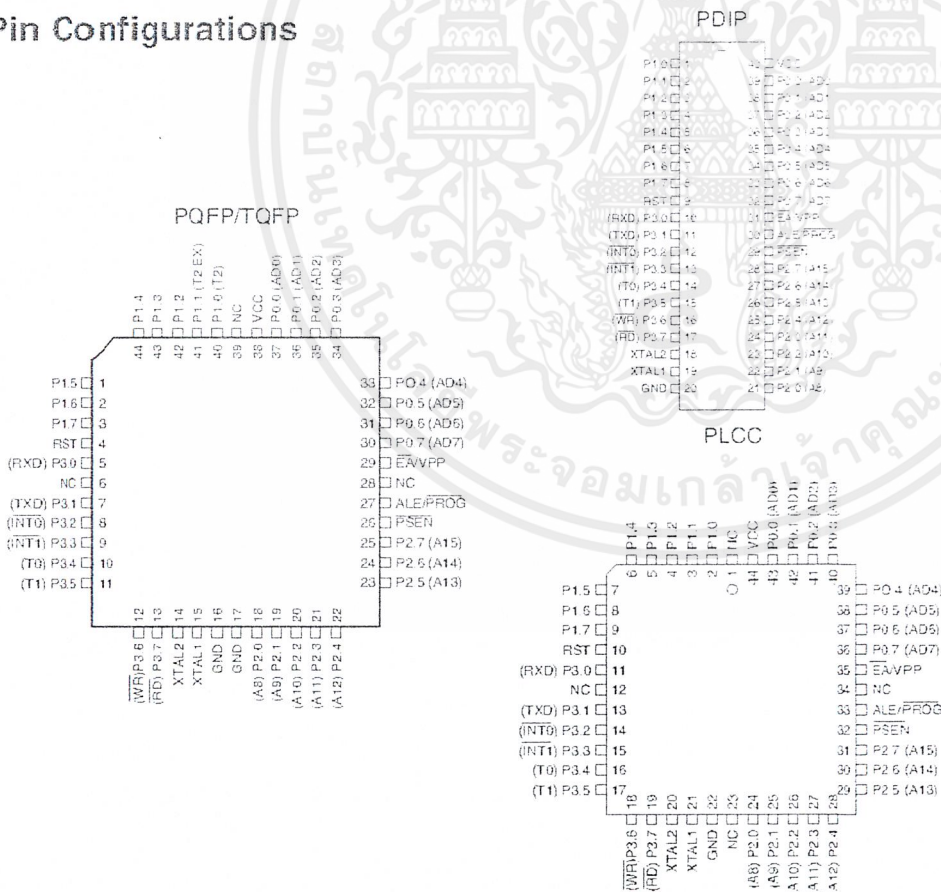
Features

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

Description

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

Pin Configurations



8-bit Microcontroller with 4K Bytes Flash

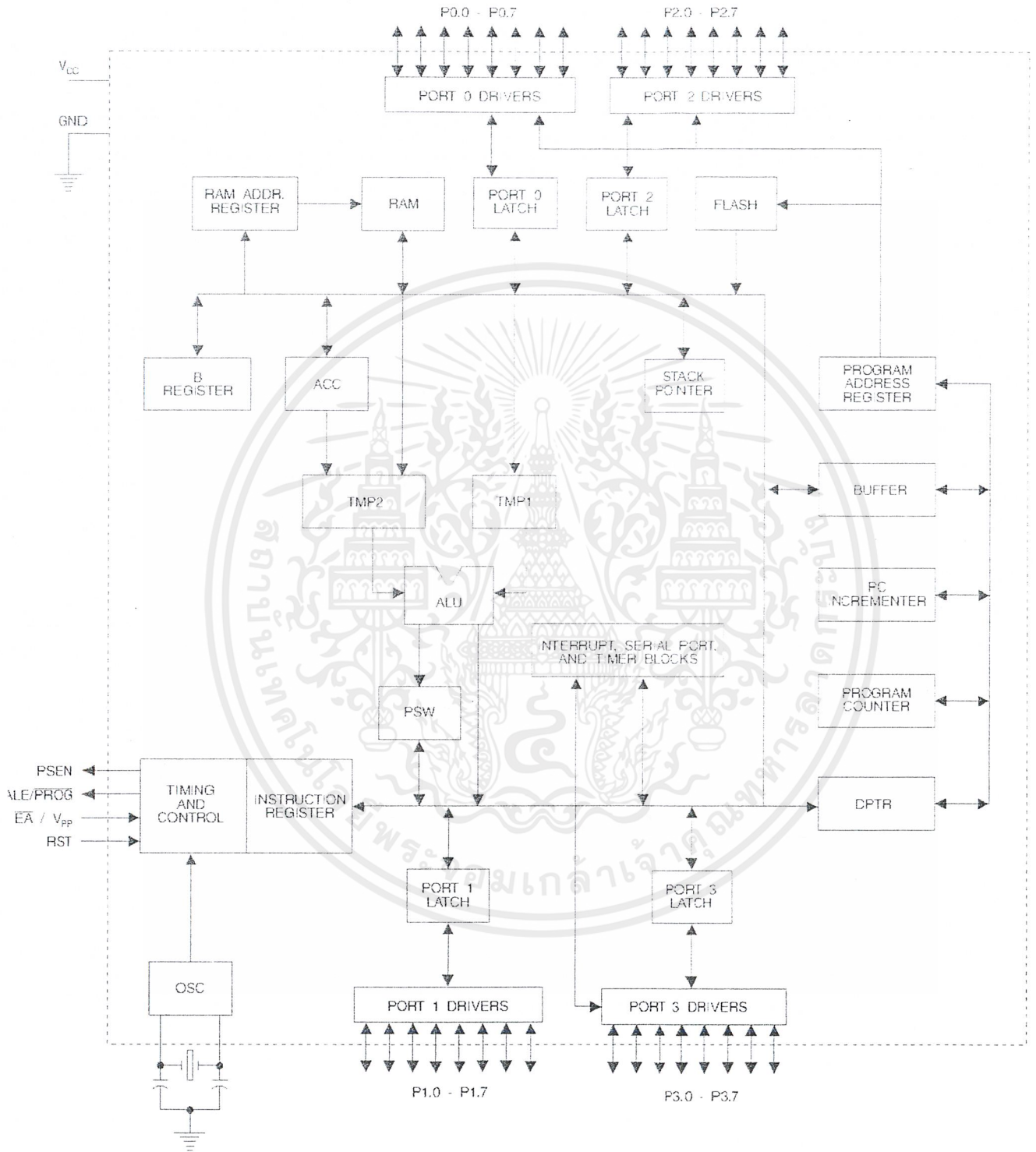
AT89C51

Rev. 0265G-02/00



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Block Diagram



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Flash Programming and Verification Characteristics

T_A = 0°C to 70°C, V_{CC} = 5.0 ± 10%

Symbol	Parameter	Min	Max	Units
V _{PP} ⁽¹⁾	Programming Enable Voltage	11.5	12.5	V
I _{PP} ⁽¹⁾	Programming Enable Current		1.0	mA
1/t _{CLCL}	Oscillator Frequency	3	24	MHz
t _{AVGL}	Address Setup to $\overline{\text{PROG}}$ Low	48t _{CLCL}		
t _{GHAX}	Address Hold After $\overline{\text{PROG}}$	48t _{CLCL}		
t _{DVGL}	Data Setup to $\overline{\text{PROG}}$ Low	48t _{CLCL}		
t _{GHDX}	Data Hold After $\overline{\text{PROG}}$	48t _{CLCL}		
t _{ESHSH}	P2.7 ($\overline{\text{ENABLE}}$) High to V _{PP}	48t _{CLCL}		
t _{SHGL}	V _{PP} Setup to $\overline{\text{PROG}}$ Low	10		μs
t _{GHSL} ⁽¹⁾	V _{PP} Hold After $\overline{\text{PROG}}$	10		μs
t _{GLGH}	$\overline{\text{PROG}}$ Width	1	110	μs
t _{AVQV}	Address to Data Valid		48t _{CLCL}	
t _{ELOV}	$\overline{\text{ENABLE}}$ Low to Data Valid		48t _{CLCL}	
t _{EHQZ}	Data Float After $\overline{\text{ENABLE}}$	0	48t _{CLCL}	
t _{GHBL}	$\overline{\text{PROG}}$ High to $\overline{\text{BUSY}}$ Low		1.0	μs
t _{WC}	Byte Write Cycle Time		2.0	ms

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





Absolute Maximum Ratings*

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

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

DC Characteristics

$T_A = -40^\circ\text{C}$ to 85°C , $V_{CC} = 5.0\text{V} \pm 20\%$ (unless otherwise noted)

Symbol	Parameter	Condition	Min	Max	Units
V_{IL}	Input Low-voltage	(Except EA)	-0.5	$0.2 V_{CC} - 0.1$	V
V_{IL1}	Input Low-voltage (EA)		-0.5	$0.2 V_{CC} - 0.3$	V
V_{IH}	Input High-voltage	(Except XTAL1, RST)	$0.2 V_{CC} + 0.9$	$V_{CC} + 0.5$	V
V_{IH1}	Input High-voltage	(XTAL1, RST)	$0.7 V_{CC}$	$V_{CC} + 0.5$	V
V_{OL}	Output Low-voltage ⁽¹⁾ (Ports 1,2,3)	$I_{OL} = 1.6 \text{ mA}$		0.45	V
V_{OL1}	Output Low-voltage ⁽¹⁾ (Port 0, ALE, PSEN)	$I_{OL} = 3.2 \text{ mA}$		0.45	V
V_{OH}	Output High-voltage (Ports 1,2,3, ALE, PSEN)	$I_{OH} = -60 \mu\text{A}$, $V_{CC} = 5\text{V} \pm 10\%$	2.4		V
		$I_{OH} = -25 \mu\text{A}$	$0.75 V_{CC}$		V
		$I_{OH} = -10 \mu\text{A}$	$0.9 V_{CC}$		V
V_{OH1}	Output High-voltage (Port 0 in External Bus Mode)	$I_{OH} = -800 \mu\text{A}$, $V_{CC} = 5\text{V} \pm 10\%$	2.4		V
		$I_{OH} = -300 \mu\text{A}$	$0.75 V_{CC}$		V
I_{IL}	Logical 0 Input Current (Ports 1,2,3)	$V_{IN} = 0.45\text{V}$		-50	μA
		$V_{IN} = 2\text{V}$, $V_{CC} = 5\text{V} \pm 10\%$		-650	μA
I_U	Input Leakage Current (Port 0, EA)	$0.45 < V_{IN} < V_{CC}$		± 10	μA
RRST	Reset Pull-down Resistor		50	300	$\text{K}\Omega$
C_{IO}	Pin Capacitance	Test Freq. = 1 MHz, $T_A = 25^\circ\text{C}$		10	pF
I_{CC}	Power Supply Current	Active Mode, 12 MHz		20	mA
		Idle Mode, 12 MHz		5	mA
	Power-down Mode ⁽²⁾	$V_{CC} = 6\text{V}$		100	μA
		$V_{CC} = 3\text{V}$		40	μA

Notes: 1. Under steady state (non-transient) conditions, I_{OL} must be externally limited as follows:

Maximum I_{OL} per port pin: 10 mA

Maximum I_{OL} per 8-bit port: Port 0: 26 mA

Ports 1, 2, 3: 15 mA

Maximum total I_{OL} for all output pins: 71 mA

If I_{OL} exceeds the test condition, V_{OL} may exceed the related specification. Pins are not guaranteed to sink current greater than the listed test conditions.

2. Minimum V_{CC} for Power-down is 2V.

AC Characteristics

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

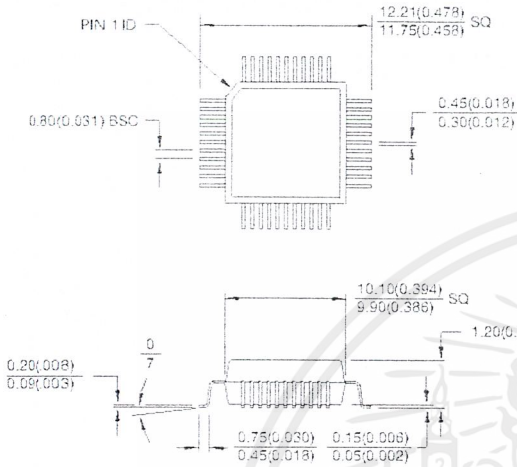
External Program and Data Memory Characteristics

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

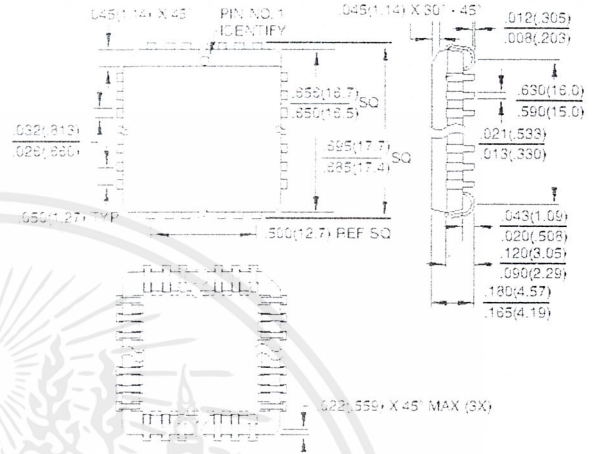


Packaging Information

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

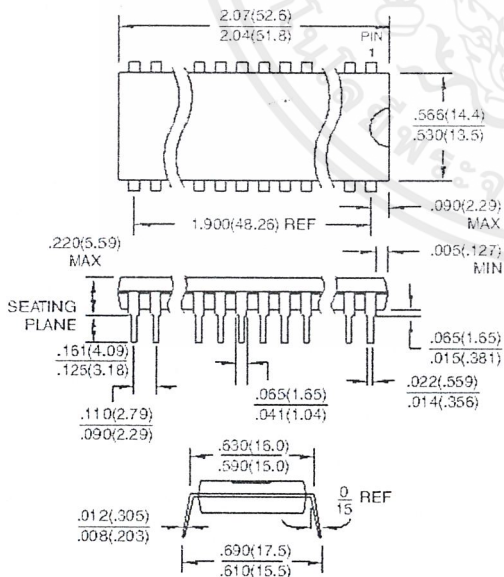


44J, 44-lead, Plastic J-leaded Chip Carrier (PLCC)
 Dimensions in Inches and (Millimeters)
 JEDEC STANDARD MS-018 AC

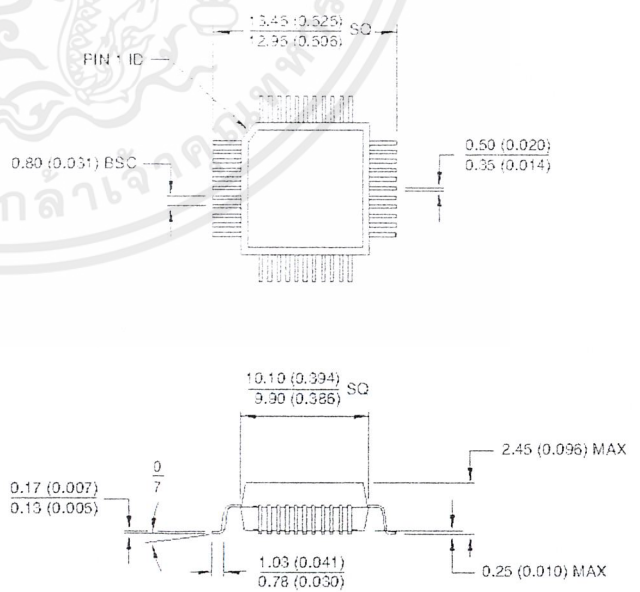


Controlling dimension: millimeters

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



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



Controlling dimension: millimeters

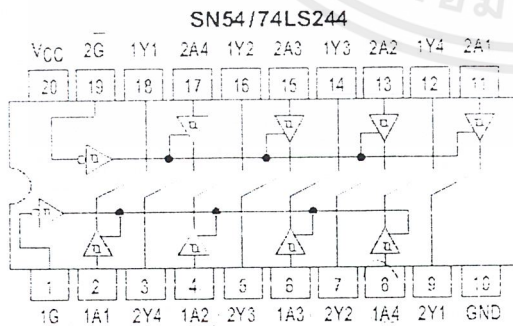
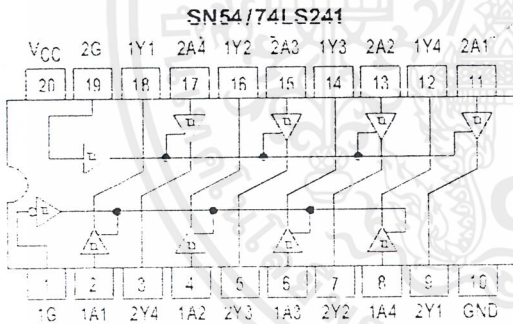
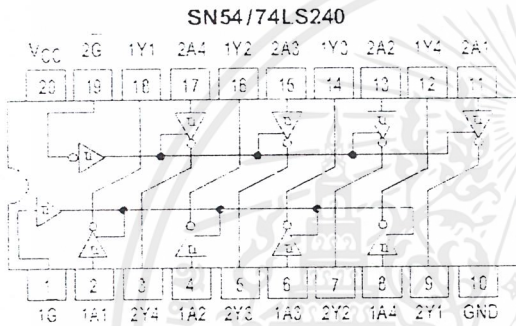


OCTAL BUFFER/LINE DRIVER WITH 3-STATE OUTPUTS

The SN54/74LS240, 241 and 244 are Octal Buffers and Line Drivers designed to be employed as memory address drivers, clock drivers and bus-oriented transmitters/receivers which provide improved PC board density.

- Hysteresis at Inputs to Improve Noise Margins
- 3-State Outputs Drive Bus Lines or Buffer Memory Address Registers
- Input Clamp Diodes Limit High-Speed Termination Effects

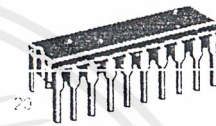
LOGIC AND CONNECTION DIAGRAMS DIP (TOP VIEW)



**SN54/74LS240
SN54/74LS241
SN54/74LS244**

OCTAL BUFFER/LINE DRIVER
WITH 3-STATE OUTPUTS

LOW POWER SCHOTTKY



J SUFFIX
CERAMIC
CASE 732-03



N SUFFIX
PLASTIC
CASE 738-03



DW SUFFIX
SOIC
CASE 751D-03

ORDERING INFORMATION

SN54LSXXXJ Ceramic
SN74LSXXXN Plastic
SN74LSXXXDW SOIC

FAST AND LS TTL DATA

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

SN54/74LS240 • SN54/74LS241 • SN54/74LS244

TRUTH TABLES

SN54/74LS240

INPUTS		OUTPUT
1G, 2G	D	
L	L	H
L	H	L
H	X	(Z)

SN54/74LS244

INPUTS		OUTPUT
1G, 2G	D	
L	L	L
L	H	H
H	X	(Z)

SN54/74LS241

INPUTS		OUTPUT	INPUTS		OUTPUT
1G	D		2G	D	
L	L	L	H	L	L
L	H	H	H	H	H
H	X	(Z)	L	X	(Z)

H = HiGH Voltage Level
 L = LOW Voltage Level
 X = Imaterial
 Z = HiGH impedance

GUARANTEED OPERATING RANGES

Symbol	Parameter		Min	Typ	Max	Unit
VCC	Supply Voltage	54	4.5	5.0	5.5	V
		74	4.75	5.0	5.25	
TA	Operating Ambient Temperature Range	54	-55	25	125	°C
		74	0	25	70	
IOH	Output Current — High	54, 74			-3.0	mA
		54 74			-12 -15	mA
IOL	Output Current — Low	54			12	mA
		74			24	

FAST AND LS TTL DATA

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
 ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

SN54/74LS240 • SN54/74LS241 • SN54/74LS244

DC CHARACTERISTICS OVER OPERATING TEMPERATURE RANGE (unless otherwise specified)

Symbol	Parameter	Limits			Unit	Test Conditions
		Min	Typ	Max		
V _{IH}	Input HIGH Voltage	2.0			V	Guaranteed Input HIGH Voltage for All Inputs
V _{IL}	Input LOW Voltage	54		0.7	V	Guaranteed Input LOW Voltage for All Inputs
		74		0.8		
V _{T+} –V _{T–}	Hysteresis	0.2	0.4		V	V _{CC} = MIN
V _{IK}	Input Clamp Diode Voltage		–0.65	–1.5	V	V _{CC} = MIN, I _{IN} = –18 mA
V _{OH}	Output HIGH Voltage	54, 74	2.4	3.4	V	V _{CC} = MIN, I _{OH} = –3.0 mA
		54, 74	2.0		V	V _{CC} = MIN, I _{OH} = MAX
V _{OL}	Output LOW Voltage	54, 74	0.25	0.4	V	I _{OL} = 12 mA
		74	0.35	0.5	V	I _{OL} = 24 mA
						V _{CC} = V _{CC} MIN, V _{IN} = V _{IL} or V _{IH} per Truth Table
I _{OZH}	Output Off Current HIGH			20	μA	V _{CC} = MAX, V _{OUT} = 2.7 V
I _{OZL}	Output Off Current LOW			–20	μA	V _{CC} = MAX, V _{OUT} = 0.4 V
I _{IH}	Input HIGH Current			20	μA	V _{CC} = MAX, V _{IN} = 2.7 V
				0.1	mA	V _{CC} = MAX, V _{IN} = 7.0 V
I _{IL}	Input LOW Current			–0.2	mA	V _{CC} = MAX, V _{IN} = 0.4 V
I _{OS}	Output Short Circuit Current (Note 1)	–40		–225	mA	V _{CC} = MAX
I _{CC}	Power Supply Current Total, Output HIGH			27	mA	V _{CC} = MAX
	Total, Output LOW	LS240		44		
		LS241/244		46		
	Total at HIGH Z	LS240		50		
LS241/244			54			

Note 1: Not more than one output should be shorted at a time, nor for more than 1 second.

AC CHARACTERISTICS (T_A = 25°C, V_{CC} = 5.0 V)

Symbol	Parameter	Limits			Unit	Test Conditions
		Min	Typ	Max		
t _{PLH}	Propagation Delay, Data to Output LS240		9.0	14	ns	C _L = 45 pF, R _L = 667 Ω
t _{PHL}			12	18		
t _{PLH}	Propagation Delay, Data to Output LS241/244		12	18	ns	
t _{PHL}			12	18		
t _{PZH}	Output Enable Time to HIGH Level		15	23	ns	
t _{PZL}	Output Enable Time to LOW Level		20	30	ns	
t _{PLZ}	Output Disable Time from LOW Level		15	25	ns	C _L = 5.0 pF, R _L = 667 Ω
t _{PHZ}	Output Disable Time from HIGH Level		10	18	ns	

FAST AND LS TTL DATA

SN54/74LS240 • SN54/74LS241 • SN54/74LS244

AC WAVEFORMS

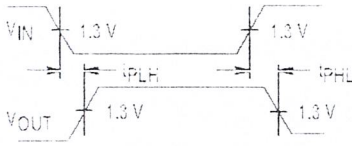


Figure 1

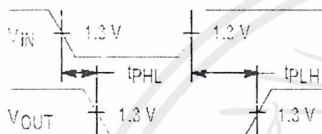


Figure 2

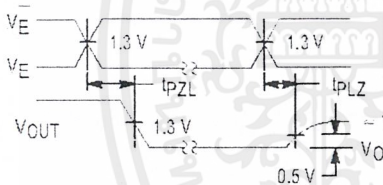


Figure 3

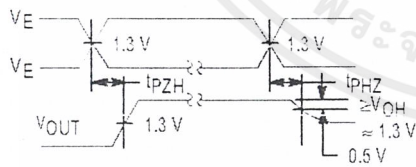
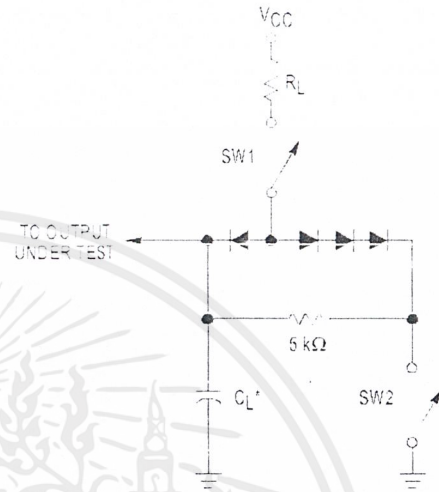


Figure 4



SWITCH POSITIONS

SYMBOL	SW1	SW2
tPZH	Open	Closed
tPZL	Closed	Open
tPLZ	Closed	Closed
tPHZ	Closed	Closed

Figure 5

FAST AND LS TTL DATA

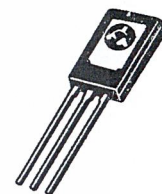
Plastic Medium Power Silicon NPN Transistor

BD135
BD137
BD139

... designed for use as audio amplifiers and drivers utilizing complementary or quasi complementary circuits.

- DC Current Gain — $h_{FE} = 40$ (Min) @ $I_C = 0.15$ Adc
- BD 135, 137, 139 are complementary with BD 136, 138, 140

1.5 AMPERE
POWER TRANSISTORS
NPN SILICON
45, 60, 80 VOLTS
10 WATTS



CASE 77-08
TO-225AA TYPE

MAXIMUM RATINGS

Rating	Symbol	Type	Value	Unit
Collector-Emitter Voltage	V_{CEO}	BD 135 BD 137 BD 139	45 60 80	Vdc
Collector-Base Voltage	V_{CBO}	BD 135 BD 137 BD 139	45 60 100	Vdc
Emitter-Base Voltage	V_{EBO}		5	Vdc
Collector Current	I_C		1.5	Adc
Base Current	I_B		0.5	Adc
Total Device Dissipation @ $T_A = 25^\circ\text{C}$ Derate above 25°C	P_D		1.25 10	Watts mW/ $^\circ\text{C}$
Total Device Dissipation @ $T_C = 25^\circ\text{C}$ Derate above 25°C	P_D		12.5 100	Watt mW/ $^\circ\text{C}$
Operating and Storage Junction Temperature Range	T_J, T_{stg}		-55 to +150	$^\circ\text{C}$

THERMAL CHARACTERISTICS

Characteristic	Symbol	Max	Unit
Thermal Resistance, Junction to Case	θ_{JC}	10	$^\circ\text{C}/\text{W}$
Thermal Resistance, Junction to Ambient	θ_{JA}	100	$^\circ\text{C}/\text{W}$

REV 7

© Motorola, Inc. 1995



MOTOROLA

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

BD135 BD137 BD139

ELECTRICAL CHARACTERISTICS ($T_C = 25^\circ\text{C}$ unless otherwise noted)

Characteristic	Symbol	Type	Min	Max	Unit
Collector–Emitter Sustaining Voltage* ($I_C = 0.03 \text{ Adc}$, $I_B = 0$)	BV_{CE0}^*	BD 135 BD 137 BD 139	45 60 80	— — —	Vdc
Collector Cutoff Current ($V_{CB} = 30 \text{ Vdc}$, $I_E = 0$) ($V_{CB} = 30 \text{ Vdc}$, $I_E = 0$, $T_C = 125^\circ\text{C}$)	I_{CBO}		— —	0.1 10	μAdc
Emitter Cutoff Current ($V_{BE} = 5.0 \text{ Vdc}$, $I_C = 0$)	I_{EBO}		—	10	μAdc
DC Current Gain ($I_C = 0.005 \text{ A}$, $V_{CE} = 2 \text{ V}$) ($I_C = 0.15 \text{ A}$, $V_{CE} = 2 \text{ V}$) ($I_C = 0.5 \text{ A}$, $V_{CE} = 2 \text{ V}$)	h_{FE}^*		25 40 25	— 250 —	—
Collector–Emitter Saturation Voltage* ($I_C = 0.5 \text{ Adc}$, $I_B = 0.05 \text{ Adc}$)	$V_{CE(sat)}^*$		—	0.5	Vdc
Base–Emitter On Voltage* ($I_C = 0.5 \text{ Adc}$, $V_{CE} = 2.0 \text{ Vdc}$)	$V_{BE(on)}^*$		—	1	Vdc

* Pulse Test: Pulse Width $\leq 300 \mu\text{s}$, Duty Cycle $\leq 2.0\%$.

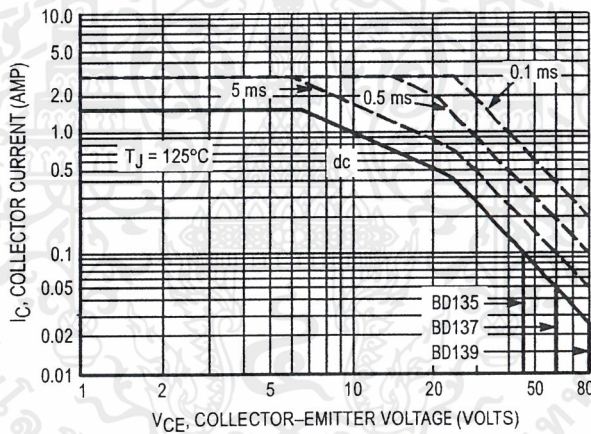
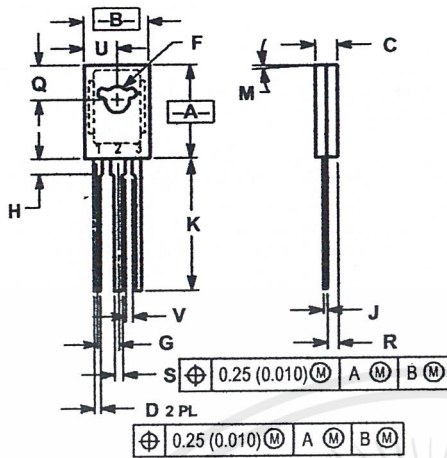


Figure 1. Active–Region Safe Operating Area

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

PACKAGE DIMENSIONS



NOTES:
 1. DIMENSIONING AND TOLERANCING PER ANSI Y14.5M, 1982.
 2. CONTROLLING DIMENSION: INCH.

DIM	INCHES		MILLIMETERS	
	MIN	MAX	MIN	MAX
A	0.425	0.435	10.80	11.04
B	0.295	0.305	7.50	7.74
C	0.095	0.105	2.42	2.66
D	0.020	0.026	0.51	0.66
F	0.115	0.130	2.93	3.30
G	0.094	BSC	2.39	BSC
H	0.050	0.095	1.27	2.41
J	0.015	0.025	0.39	0.63
K	0.575	0.655	14.61	16.63
M	5° TYP		5° TYP	
Q	0.148	0.158	3.76	4.01
R	0.045	0.055	1.15	1.39
S	0.025	0.035	0.64	0.88
U	0.145	0.155	3.69	3.93
V	0.040	—	1.02	—

STYLE 1:
 PIN 1. EMITTER
 2. COLLECTOR
 3. BASE

CASE 77-08
 TO-225AA TYPE
 ISSUE V



Motorola reserves the right to make changes without further notice to any products herein. Motorola makes no warranty, representation or guarantee regarding the suitability of its products for any particular purpose, nor does Motorola assume any liability arising out of the application or use of any product or circuit, and specifically disclaims any and all liability, including without limitation consequential or incidental damages. "Typical" parameters can and do vary in different applications. All operating parameters, including "Typicals" must be validated for each customer application by customer's technical experts. Motorola does not convey any license under its patent rights nor the rights of others. Motorola products are not designed, intended, or authorized for use as components in systems intended for surgical implant into the body, or other applications intended to support or sustain life, or for any other application in which the failure of the Motorola product could create a situation where personal injury or death may occur. Should Buyer purchase or use Motorola products for any such unintended or unauthorized application, Buyer shall indemnify and hold Motorola and its officers, employees, subsidiaries, affiliates, and distributors harmless against all claims, costs, damages, and expenses, and reasonable attorney fees arising out of, directly or indirectly, any claim of personal injury or death associated with such unintended or unauthorized use, even if such claim alleges that Motorola was negligent regarding the design or manufacture of the part. Motorola and (M) are registered trademarks of Motorola, Inc. Motorola, Inc. is an Equal Opportunity/Affirmative Action Employer.

How to reach us:

USA / EUROPE: Motorola Literature Distribution;
P.O. Box 20912; Phoenix, Arizona 85036. 1-800-441-2447

JAPAN: Nippon Motorola Ltd.; Tatsumi-SPD-JLDC, Toshikatsu Otsuki,
6F Seibu-Butsuryu-Center, 3-14-2 Tatsumi Koto-Ku, Tokyo 135, Japan. 03-3521-8315

MFAX: RMFAX0@email.sps.mot.com – TOUCHTONE (602) 244-6609
INTERNET: http://Design-NET.com

HONG KONG: Motorola Semiconductors H.K. Ltd.; 8B Tai Ping Industrial Park,
51 Ting Kok Road, Tai Po, N.T., Hong Kong. 852-26629298



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

LM78XX Series Voltage Regulators

General Description

The LM78XX series of three terminal regulators is available with several fixed output voltages making them useful in a wide range of applications. One of these is local on card regulation, eliminating the distribution problems associated with single point regulation. The voltages available allow these regulators to be used in logic systems, instrumentation, HiFi, and other solid state electronic equipment. Although designed primarily as fixed voltage regulators these devices can be used with external components to obtain adjustable voltages and currents.

The LM78XX series is available in an aluminum TO-3 package which will allow over 1.0A load current if adequate heat sinking is provided. Current limiting is included to limit the peak output current to a safe value. Safe area protection for the output transistor is provided to limit internal power dissipation. If internal power dissipation becomes too high for the heat sinking provided, the thermal shutdown circuit takes over preventing the IC from overheating.

Considerable effort was expended to make the LM78XX series of regulators easy to use and minimize the number

of external components. It is not necessary to bypass the output, although this does improve transient response. Input bypassing is needed only if the regulator is located far from the filter capacitor of the power supply.

For output voltage other than 5V, 12V and 15V the LM117 series provides an output voltage range from 1.2V to 57V.

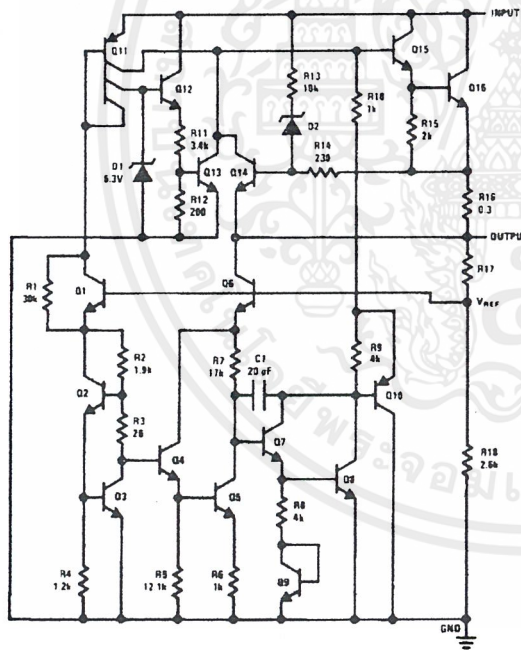
Features

- Output current in excess of 1A
- Internal thermal overload protection
- No external components required
- Output transistor safe area protection
- Internal short circuit current limit
- Available in the aluminum TO-3 package

Voltage Range

LM7805C	5V
LM7812C	12V
LM7815C	15V

Schematic and Connection Diagrams



TL/H/7746-1

Metal Can Package
TO-3 (K)
Aluminum

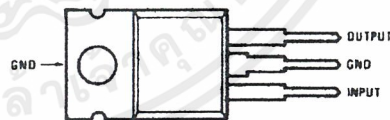


TL/H/7746-2

Bottom View

Order Number LM7805CK,
LM7812CK or LM7815CK
See NS Package Number KC02A

Plastic Package
TO-220 (T)



TL/H/7746-3

Top View

Order Number LM7805CT,
LM7812CT or LM7815CT
See NS Package Number T03B

Absolute Maximum Ratings

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

Input Voltage ($V_O = 5V, 12V$ and $15V$) 35V
 Internal Power Dissipation (Note 1) Internally Limited
 Operating Temperature Range (T_A) 0°C to $+70^\circ\text{C}$

Maximum Junction Temperature
 (K Package) 150°C
 (T Package) 150°C
 Storage Temperature Range -65°C to $+150^\circ\text{C}$
 Lead Temperature (Soldering, 10 sec.)
 TO-3 Package K 300°C
 TO-220 Package T 230°C

Electrical Characteristics LM78XXC (Note 2) $0^\circ\text{C} \leq T_j \leq 125^\circ\text{C}$ unless otherwise noted.

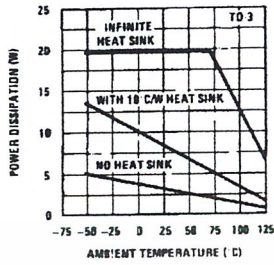
Output Voltage			5V			12V			15V			Units	
Input Voltage (unless otherwise noted)			10V			19V			23V				
Symbol	Parameter	Conditions	Min	Typ	Max	Min	Typ	Max	Min	Typ	Max		
V_O	Output Voltage	$T_j = 25^\circ\text{C}, 5\text{ mA} \leq I_O \leq 1\text{ A}$	4.8	5	5.2	11.5	12	12.5	14.4	15	15.6	V	
		$P_D \leq 15\text{ W}, 5\text{ mA} \leq I_O \leq 1\text{ A}$	4.75		5.25	11.4		12.6	14.25		15.75	V	
		$V_{\text{MIN}} \leq V_{\text{IN}} \leq V_{\text{MAX}}$	(7.5 $\leq V_{\text{IN}} \leq 20$)				(14.5 $\leq V_{\text{IN}} \leq 27$)			(17.5 $\leq V_{\text{IN}} \leq 30$)			V
ΔV_O	Line Regulation	$I_O = 500\text{ mA}$	$T_j = 25^\circ\text{C}$	3	50	4	120	4	150			mV	
			ΔV_{IN}	(7 $\leq V_{\text{IN}} \leq 25$)			(14.5 $\leq V_{\text{IN}} \leq 30$)			(17.5 $\leq V_{\text{IN}} \leq 30$)			V
		$0^\circ\text{C} \leq T_j \leq +125^\circ\text{C}$	ΔV_{IN}	50		120			150				mV
			ΔV_{IN}	(8 $\leq V_{\text{IN}} \leq 20$)			(15 $\leq V_{\text{IN}} \leq 27$)			(18.5 $\leq V_{\text{IN}} \leq 30$)			V
		$I_O \leq 1\text{ A}$	$T_j = 25^\circ\text{C}$	50		120			150				mV
			ΔV_{IN}	(7.5 $\leq V_{\text{IN}} \leq 20$)			(14.6 $\leq V_{\text{IN}} \leq 27$)			(17.7 $\leq V_{\text{IN}} \leq 30$)			V
$0^\circ\text{C} \leq T_j \leq +125^\circ\text{C}$	ΔV_{IN}	25		60			75				mV		
	ΔV_{IN}	(8 $\leq V_{\text{IN}} \leq 12$)			(16 $\leq V_{\text{IN}} \leq 22$)			(20 $\leq V_{\text{IN}} \leq 26$)			V		
ΔV_O	Load Regulation	$T_j = 25^\circ\text{C}$	$5\text{ mA} \leq I_O \leq 1.5\text{ A}$	10	50	12	120	12	150			mV	
			$250\text{ mA} \leq I_O \leq 750\text{ mA}$			25		60		75		mV	
		$5\text{ mA} \leq I_O \leq 1\text{ A}, 0^\circ\text{C} \leq T_j \leq +125^\circ\text{C}$			50		120		150			mV	
I_Q	Quiescent Current	$I_O \leq 1\text{ A}$	$T_j = 25^\circ\text{C}$	8		8		8		8		mA	
			$0^\circ\text{C} \leq T_j \leq +125^\circ\text{C}$			8.5		8.5		8.5		mA	
ΔI_Q	Quiescent Current Change	$5\text{ mA} \leq I_O \leq 1\text{ A}$	$T_j = 25^\circ\text{C}$	0.5		0.5		0.5		0.5		mA	
			$T_j = 25^\circ\text{C}, I_O \leq 1\text{ A}$			1.0		1.0		1.0		mA	
			$V_{\text{MIN}} \leq V_{\text{IN}} \leq V_{\text{MAX}}$	(7.5 $\leq V_{\text{IN}} \leq 20$)			(14.8 $\leq V_{\text{IN}} \leq 27$)			(17.9 $\leq V_{\text{IN}} \leq 30$)			V
			$I_O \leq 500\text{ mA}, 0^\circ\text{C} \leq T_j \leq +125^\circ\text{C}$			1.0		1.0		1.0		1.0	mA
$V_{\text{MIN}} \leq V_{\text{IN}} \leq V_{\text{MAX}}$	(7 $\leq V_{\text{IN}} \leq 25$)			(14.5 $\leq V_{\text{IN}} \leq 30$)			(17.5 $\leq V_{\text{IN}} \leq 30$)			V			
V_N	Output Noise Voltage	$T_A = 25^\circ\text{C}, 10\text{ Hz} \leq f \leq 100\text{ kHz}$	40		75		90				μV		
$\frac{\Delta V_{\text{IN}}}{\Delta V_{\text{OUT}}}$	Ripple Rejection	$f = 120\text{ Hz}$	$I_O \leq 1\text{ A}, T_j = 25^\circ\text{C}$ or $I_O \leq 500\text{ mA}$	62	80	55	72	54	70			dB	
			$0^\circ\text{C} \leq T_j \leq +125^\circ\text{C}$			62		55		54		dB	
			$V_{\text{MIN}} \leq V_{\text{IN}} \leq V_{\text{MAX}}$	(8 $\leq V_{\text{IN}} \leq 18$)			(15 $\leq V_{\text{IN}} \leq 25$)			(18.5 $\leq V_{\text{IN}} \leq 28.5$)			V
R_O	Dropout Voltage	$T_j = 25^\circ\text{C}, I_{\text{OUT}} = 1\text{ A}$	2.0		2.0		2.0		2.0		V		
	Output Resistance	$f = 1\text{ kHz}$	8		18		19		19		$\text{m}\Omega$		
	Short-Circuit Current	$T_j = 25^\circ\text{C}$	2.1		1.5		1.2		1.2		A		
	Peak Output Current	$T_j = 25^\circ\text{C}$	2.4		2.4		2.4		2.4		A		
	Average TC of V_{OUT}	$0^\circ\text{C} \leq T_j \leq +125^\circ\text{C}, I_O = 5\text{ mA}$	0.6		1.5		1.8		1.8		$\text{mV}/^\circ\text{C}$		
V_{IN}	Input Voltage Required to Maintain Line Regulation	$T_j = 25^\circ\text{C}, I_O \leq 1\text{ A}$	7.5		14.6		17.7				V		

Note 1: Thermal resistance of the TO-3 package (K, KC) is typically $4^\circ\text{C}/\text{W}$ junction to case and $35^\circ\text{C}/\text{W}$ case to ambient. Thermal resistance of the TO-220 package (T) is typically $4^\circ\text{C}/\text{W}$ junction to case and $50^\circ\text{C}/\text{W}$ case to ambient.

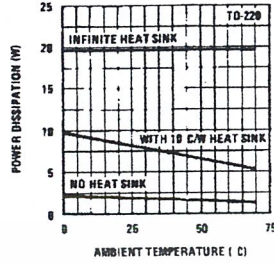
Note 2: All characteristics are measured with capacitor across the input of $0.22\ \mu\text{F}$, and a capacitor across the output of $0.1\ \mu\text{F}$. All characteristics except noise voltage and ripple rejection ratio are measured using pulse techniques ($t_w < 10\text{ ms}$, duty cycle $< 5\%$). Output voltage changes due to changes in internal temperature must be taken into account separately.

Typical Performance Characteristics

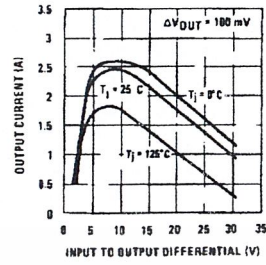
Maximum Average Power Dissipation



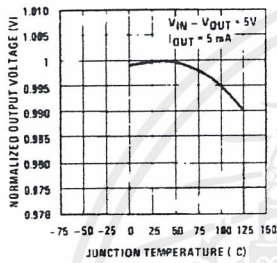
Maximum Average Power Dissipation



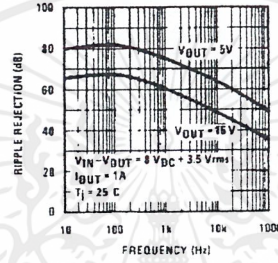
Peak Output Current



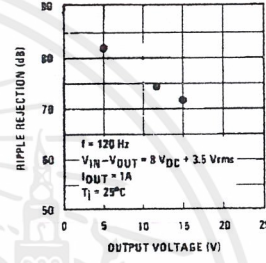
Output Voltage (Normalized to 1V at $T_J = 25^\circ\text{C}$)



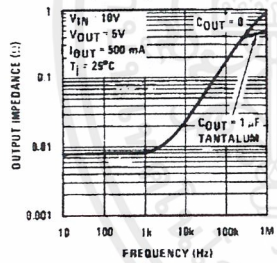
Ripple Rejection



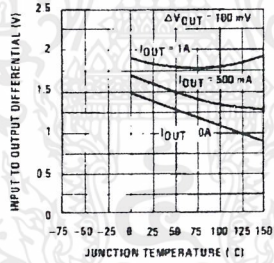
Ripple Rejection



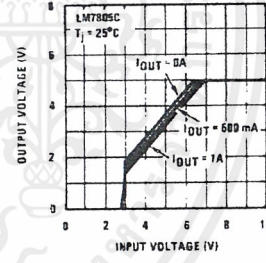
Output Impedance



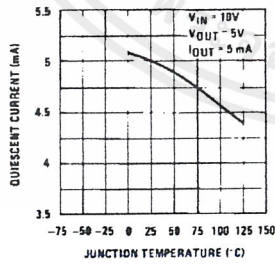
Dropout Voltage



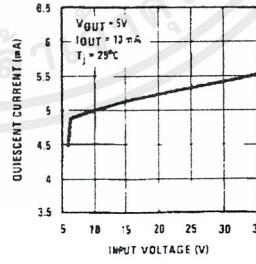
Dropout Characteristics



Quiescent Current

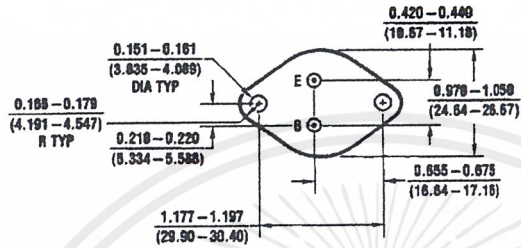
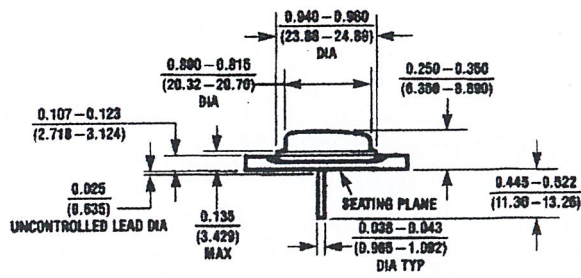


Quiescent Current



TL/H/7746-4

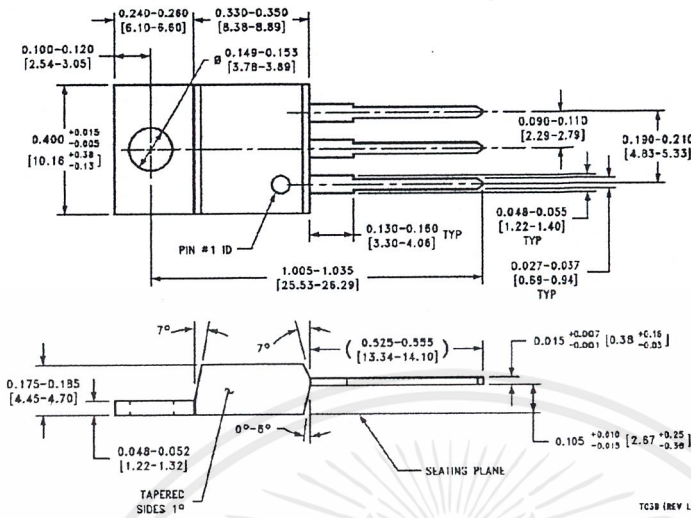
Physical Dimensions inches (millimeters)



Aluminum Metal Can Package (KC)
 Order Number LM7805CK, LM7812CK or LM7815CK
 NS Package Number KC02A

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
 ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Physical Dimensions inches (millimeters) (Continued)



TO-220 Package (T)
 Order Number LM7805CT, LM7812CT or LM7815CT
 NS Package Number T03B

LIFE SUPPORT POLICY

NATIONAL'S PRODUCTS ARE NOT AUTHORIZED FOR USE AS CRITICAL COMPONENTS IN LIFE SUPPORT DEVICES OR SYSTEMS WITHOUT THE EXPRESS WRITTEN APPROVAL OF THE PRESIDENT OF NATIONAL SEMICONDUCTOR CORPORATION. As used herein:

1. Life support devices or systems are devices or systems which, (a) are intended for surgical implant into the body, or (b) support or sustain life, and whose failure to perform, when properly used in accordance with instructions for use provided in the labeling, can be reasonably expected to result in a significant injury to the user.
2. A critical component is any component of a life support device or system whose failure to perform can be reasonably expected to cause the failure of the life support device or system, or to affect its safety or effectiveness.



National Semiconductor Corporation
 1111 West Bardin Road
 Arlington, TX 76017
 Tel: 1(800) 272-9959
 Fax: 1(800) 737-7018

National Semiconductor Europe
 Fax: (+49) 0-180-530 85 86
 Email: cnjwge@tevm2.nsc.com
 Deutsch Tel: (+49) 0-180-530 85 85
 English Tel: (+49) 0-180-532 78 32
 Français Tel: (+49) 0-180-532 93 58
 Italiano Tel: (+49) 0-180-534 16 80

National Semiconductor Hong Kong Ltd.
 13th Floor, Straight Block,
 Ocean Centre, 5 Canton Rd.
 Tsimshatsui, Kowloon
 Hong Kong
 Tel: (852) 2737-1600
 Fax: (852) 2736-9960

National Semiconductor Japan Ltd.
 Tel: 81-043-299-2309
 Fax: 81-043-299-2408

National does not assume any responsibility for use of any circuitry described, no circuit patent licenses are implied and National reserves the right at any time without notice to change said circuitry and specifications.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
 ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กิตติกรรมประกาศ

คณะผู้จัดทำขอขอบพระคุณอาจารย์ที่ปรึกษา อาจารย์ถาวร เบญจนาสุทท์ ที่ได้ให้คำแนะนำเกี่ยวกับการทำโครงการนี้จนประสบความสำเร็จและขอขอบคุณ เด็ย เจี๊ยบ จอย ฤทธิ์ รัน และเพื่อนๆทุกคนที่ให้ความช่วยเหลือคำแนะนำที่มีประโยชน์อย่างยิ่งต่อการทำโครงการนี้จนสำเร็จลุล่วงไปด้วยดี

สุดท้ายนี้ขอกราบขอบพระคุณ คุณพ่อ คุณแม่ที่ให้กำเนิดและคอยช่วยเหลือสนับสนุนในทุกๆด้านตลอดมา

คณะผู้จัดทำ

นางสาว ณัฐศดา วงศ์สิทธิพิศาล

นางสาว ทิพวรรณ แก้วสังข์



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เอกสารอ้างอิง

- ◆ ฉันทวุฒิ พิษผล , พิชิต สันติภูพานนท์ , คู่มือเรียน Microsoft Visual Basic 6 , กรุงเทพฯ , 2542
- ◆ จูริพงษ์ นามแดง , “บอร์ดควบคุมสเต็ปเปอร์มอเตอร์” , วารสารฮอปปี้อิเล็กทรอนิกส์ , ฉบับที่93, 2542 , หน้า 34-39
- ◆ ยอดพงษ์ จริยวิชาววัฒน์ , เอกรัศน์ จีระเพ็ญฟูศิลป์ , “กล้องคิดตามวัตถุ” , ปรินูญานิพนธ์ปีการศึกษา 2537
- ◆ สมยศ จุณณะปิยะ , “การประยุกต์ใช้งานไมโครคอนโทรลเลอร์ตระกูล MCS-51” , คณะวิศวกรรมศาสตร์สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง , 2541 , หน้า 1-42
- ◆ สุนทร วิฑูสุรพจน์ , “ การโปรแกรมภาษาแอสเซมบลีของไมโครคอนโทรลเลอร์ตระกูล MCS-51” , บริษัทซีเ็ดยูเคชั่น จำกัด(มหาชน) , 198 หน้า , 2537
- ◆ Kenneth R. Castleman , Prentice Hall , “Digital Image Processing” , 340 P. , 1993

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้