สำนักหอสมุดกลาง พระจอมเกล้าลาดกระบัง

แขนกล

ROBOT ARM

โดย

นางสาวชยาภรณ์ ศรีราชญา

นายชัยพัฒน์ อินดีคำ

นายวีรวัฒน์ ขวัญแก้ว

วพ
ช196ป
2549

เลขหมู่.....
เลขทะเบียน 72877
วัน,เดือน,ปี 25 มิ.ย. 2550

b 11123877
i.....

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตร์บัณฑิต

สาขาวิศวกรรมระบบควบคุม

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2549

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ปริญญานิพนธ์ปีการศึกษา 2548

ภาควิชาวิศวกรรมระบบควบคุม คณะวิศวกรรมศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เรื่อง แขนกล

ROBOT ARM

ผู้จัดทำ นางสาวชยาภรณ์ ศรีราชญา 46010148

นายชัยพัฒน์ อินดีคำ 46010158

นายวีรวัฒน์ ขวัญแก้ว 46010741

.....อาจารย์ที่ปรึกษา

(ผู้ช่วยศาสตราจารย์สุมิตร พนาอุดมทรัพย์)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

แขนกล

Robot Arm

โดย

นางสาวชยาภรณ์ ศรีราชญา 46010148

นายชัยพัฒน์ อินดีคำ 46010158

นายวีรวัฒน์ ขวัญแก้ว 46010741

อาจารย์ที่ปรึกษา

ผศ.สุมิตร พนาอุดมทรัพย์

ปีการศึกษา 2549

บทคัดย่อ

ปัจจุบันนี้แขนกลได้ใช้กันอย่างแพร่หลายในวงการอุตสาหกรรม ซึ่งปริญญานิพนธ์ฉบับนี้ได้นำเสนอเกี่ยวกับ แขนกล ในการทำงานของแขนกลนี้จะแบ่งออกเป็น 3 ส่วน คือ ส่วนของคอมพิวเตอร์ ส่วนของตัวควบคุมและส่วนของแขนกล นอกจากนี้ได้มีการออกแบบและเลือกวัสดุที่เหมาะสมเพื่อนำมาใช้ในโครงงาน โดยจุดประสงค์ของโครงงานนี้เพื่อนำหุ่นยนต์มาประยุกต์ในการใช้งานจริงและใช้งานได้อย่างหลากหลาย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Robot Arm

By

Miss Chayaporn Srisadya 46010148

Mr.Chaipat Indeekham 46010158

Mr.Weerawat Khwankeiw 46010741

Advisor

Asst.Prof.Sumit Panaudomsub

Academic Year 2006

ABSTRACT

Nowaday, the robot – arm are generally used in many industrial. This thesis presents "Robot arm". The robot arm consists of three parts ; computer, controller and robot-arm structure parts. Not only that we designs and chooses the suitable materials to use in this project. The robot arm is flexible for modifying in many task.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กิตติกรรมประกาศ

การจัดทำปริญญานิพนธ์ฉบับนี้จะประสบความสำเร็จไม่ได้ถ้าขาดการช่วยเหลือและส่งเสริมทั้งทางตรงและทางอ้อม ทางคณะผู้จัดทำขอกราบขอบพระคุณทางคณะวิศวกรรมศาสตร์ที่ได้เปิดโอกาสให้มีการฝึกงานซึ่งได้จุดประกายแนวคิดในการทำโปรเจค

ขอขอบคุณ ผศ.สุมิตร พนาอุดมทรัพย์ เป็นอาจารย์ที่ปรึกษาโปรเจค ที่ให้คำปรึกษา แนะแนวทางในแก้ปัญหาต่างๆและดูแลโปรเจคตลอดมา

ขอขอบคุณ อาจารย์ธวัชชัย คำศรี ที่ได้ให้ความรู้ทางด้านไมโครคอนโทรลเลอร์และการเขียนโปรแกรมภาษาซี

ขอขอบคุณนายพัลลภ กลิ่นขจร ที่ช่วยเหลือในการให้คำปรึกษาทางด้านการเขียนโปรแกรมและแนะแนวทางการแก้ไขปัญหาต่างๆ

คณะผู้จัดทำ

นางสาวชยาภรณ์ ศรีราชญา

นายชัยพัฒน์ อินดีคำ

นายวีรวัฒน์ ขวัญแก้ว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ

	หน้า
บทคัดย่อ	I
กิตติกรรมประกาศ	III
สารบัญ	IV
สารบัญภาพ	VI
สารบัญตาราง	VIII
บทที่ 1 บทนำ	
1.1 กล่าวนำและความเป็นมา	1
1.2 วัตถุประสงค์ของงาน	2
1.3 ขั้นตอนในการศึกษาและจัดทำโครงงาน	2
1.4 รายละเอียดของปริญญานิพนธ์	2
บทที่ 2 ทฤษฎีและหลักการ	
2.1 แขนกล	4
2.1.1 คุณลักษณะของแขนมนุษย์	4
2.1.2 ประเภทของแขนกล	4
2.2 การหมุน(Orientation)	8
2.2.1 โครงพิกัด	8
2.2.2 การถ่ายโอนพิกัด	10
2.2.3 การหมุนพื้นฐาน	11
2.2.4 การเลื่อนตำแหน่งและการหมุน	13
2.2.5 ความแม่นยำของการเคลื่อนที่	14
2.2.6 สมการแขนกล	15
2.3 Servo motor	17
2.3.1 ส่วนประกอบของ Servo motor	18
2.3.2 หลักการทำงานของ Servo motor	19

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ(ต่อ)

2.4 ส่วนควบคุมโดยไมโครคอนโทรลเลอร์ 21

2.4.1 ข้อมูลเบื้องต้นของ dsPIC 21

2.4.2 คุณสมบัติเด่นโดยรวมของ dsPIC 21

2.5 โปรแกรมที่ใช้ในการควบคุมหุ่นยนต์ 23

2.5.1 ส่วนประกอบต่างๆของโปรแกรมวิชวลเบสิก 23

บทที่ 3 การออกแบบ

3.1 โครงสร้างการออกแบบแขนกล 31

3.1.1 ส่วนฐาน 31

3.1.2 ส่วนแขน 34

3.1.3 ส่วนที่ใช้จับปากกา 36

3.2 การออกแบบโปรแกรมที่ใช้ในการควบคุม 38

3.3 โปรแกรมที่ใช้ในการประมวลผลในไมโครคอนโทรลเลอร์ 40

บทที่ 4 การทดลอง

4.1 การทดลองที่ 1 42

4.2 การทดลองที่ 2 45

4.2.1 ผลการทดลอง 45

บทที่ 5 สรุปและวิจารณ์

5.1 สรุปผลการทดลอง 52

5.2 ปัญหาที่พบและแนวทางแก้ไข 52

5.3 ขอบเขตการดำเนินงานและแนวทางการพัฒนา 53

ภาคผนวก ก โปรแกรมการประมวลผลภาพและโปรแกรมควบคุม 55

ภาคผนวก ข ผลการทดลองหาข้อมูลพื้นฐานของเซอร์โวมอเตอร์แต่ละตัว 71

ภาคผนวก ค ข้อมูลความรู้ทางด้านอิเล็กทรอนิกส์ 75

เอกสารอ้างอิง 78

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญภาพ

รูปที่	หน้า
รูปที่ 2.1 แขนกลที่เคลื่อนที่ในระบบพิกัดขั้วหรือพิกัดเชิงกลม	5
รูปที่ 2.2 แขนกลที่มีการเคลื่อนที่ในระบบพิกัดเชิงขั้ว	5
รูปที่ 2.3 แขนกลที่มีลักษณะแขนเป็นข้อต่อ	6
รูปที่ 2.4 แขนกลที่เคลื่อนที่ในระบบพิกัดคาร์ทิเชียน	6
รูปที่ 2.5 แขนกลที่มีการเคลื่อนที่ได้ในระบบพิกัดเชิงกลม	7
รูปที่ 2.6 พิกัดของ p เทียบกับโครงพิกัด x	9
รูปที่ 2.7 การหมุนพื้นฐานในปริภูมิ R^3	10
รูปที่ 2.8 การหมุนของโครงพิกัดรอบแกนของโครงพิกัดอยู่นิ่งที่ 1	11
รูปที่ 2.9 โครงพิกัดฉากใน R^3	15
รูปที่ 2.10 ส่วนประกอบของเซอร์โวมอเตอร์	18
รูปที่ 2.11 หลักการทำงานของเซอร์โวมอเตอร์	19
รูปที่ 2.12 การควบคุมมอเตอร์หยุดหมุน	20
รูปที่ 2.13 หน้าต่างการเลือกชนิดโปรเจค	23
รูปที่ 2.14 หน้าต่างหลักของโปรแกรมวิชวลเบสิก	24
รูปที่ 2.15 เมนูบาร์	25
รูปที่ 2.16 ทูลบาร์	25
รูปที่ 2.17 ทูลบ็อกซ์	26
รูปที่ 2.18 หน้าต่างโปรเจค	26
รูปที่ 2.19 หน้าต่างคุณสมบัติ	27
รูปที่ 2.20 หน้าต่างฟอร์ม	28
รูปที่ 2.21 หน้าต่างโค๊ดโปรแกรม	29
รูปที่ 3.1 ส่วนประกอบต่างๆของฐานเครื่อง	32
รูปที่ 3.2 ฐานเครื่องที่ออกแบบ	32
รูปที่ 3.3 ฐานหมุนที่ออกแบบ	33
รูปที่ 3.4 ลักษณะโครงสร้างและการติดตั้งของฐานหมุน	33
รูปที่ 3.5 แขนท่อนล่างและแขนท่อนบนที่ออกแบบ	34
รูปที่ 3.6 โครงสร้างของแขนท่อนล่าง	35

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญภาพ(ต่อ)

รูปที่ 3.7 โครงสร้างของแขนท่อนบน 35
รูปที่ 3.8 ส่วนปากกาที่ออกแบบ 36
รูปที่ 3.9 โครงสร้างของแขนส่วนปลายปากกา 36
รูปที่ 3.10 ภาพโครงสร้างของแขนกล 37
รูปที่ 3.11 โครงสร้างของแขนกล 37
รูปที่ 3.12 โฟลว์ชาร์ตแสดงขั้นตอนการทำงานของโปรแกรมวิชวลเบสิก 38
รูปที่ 3.13 หน้าต่างการใช้งานของโปรแกรมวิชวลเบสิก 39
รูปที่ 3.14 โฟลว์ชาร์ตแสดงการทำงานของโปรแกรม
ประมวลผลของไมโครคอนโทรลเลอร์ 40
รูปที่ 4.1 มุมในการคำนวณการเคลื่อนที่ของแขนกล 42
รูปที่ 4.2 ตำแหน่งมุมเริ่มต้นของแขนกล 44
รูปที่ 4.3 รูปที่วาดลงในฟอร์มครั้งที่ 1 45
รูปที่ 4.4 การทดลองวาดรูปครั้งที่ 1 46
รูปที่ 4.5 รูปที่วาดลงในฟอร์มครั้งที่ 2 46
รูปที่ 4.6 การทดลองวาดรูปครั้งที่ 2 47
รูปที่ 4.7 รูปที่วาดลงในฟอร์มครั้งที่ 3 47
รูปที่ 4.8 การทดลองวาดรูปครั้งที่ 3 48
รูปที่ 4.9 รูปที่วาดลงในฟอร์มครั้งที่ 4 48
รูปที่ 4.10 การทดลองวาดรูปครั้งที่ 4 49
รูปที่ 4.11 รูปที่วาดลงในฟอร์มครั้งที่ 5 49
รูปที่ 4.12 การทดลองวาดรูปครั้งที่ 5 ตอนมีข้อผิดพลาด 50
รูปที่ 4.13 การทดลองวาดรูปครั้งที่ 5 ตอนไม่มีข้อผิดพลาด 50
รูปที่ ก.1 หน้าต่างเชื่อมต่อกับผู้ใช้ในโปรแกรมการวาดรูป 55

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญตาราง

ตารางที่	หน้า
ตารางที่ ข.1 ผลการทดลองของเซอร์โวมอเตอร์ที่ควบคุมฐาน	71
ตารางที่ ข.2 ผลการทดลองของเซอร์โวมอเตอร์ที่ควบคุมแขนท่อนล่าง	72
ตารางที่ ข.3 ผลการทดลองของเซอร์โวมอเตอร์ที่ควบคุมแขนท่อนบน	73
ตารางที่ ข.4 ผลการทดลองของเซอร์โวมอเตอร์ที่ควบคุมส่วนปลายปากกา	74

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 1
บทนำ

1.1 กล่าวนำและความเป็นมา

ในปัจจุบันนี้ หุ่นยนต์นับได้ว่ามีบทบาทสำคัญอย่างมากในอุตสาหกรรมหลายๆประเภท โดยทั่วไปแล้วถ้ากล่าวถึงหุ่นยนต์คนทั่วไปมักจะคิดถึงเรื่องหุ่นยนต์ที่มีรูปร่างลักษณะเหมือนมนุษย์ สามารถคิดและทำงานได้ด้วยตัวเอง แต่ที่จริงแล้ว คำว่า Robotics ตามนิยามของ Robotics Industrial Associationและ robotics Institute of America นั้นหมายถึง เครื่องยนต์ที่ประกอบกันขึ้นมาเพื่อทำงานบางอย่างแทนมนุษย์ สามารถควบคุมได้โดยใช้โปรแกรมคอมพิวเตอร์ และสามารถ ที่จะเปลี่ยนการทำงานให้ทำงานอย่างอื่นได้

ปริญญานิพนธ์นี้เป็นโครงงานการสร้างแขนกล (Robot Arm)มีวัตถุประสงค์ของโครงงานคือ

1. เพื่อศึกษาทฤษฎีวิชา Robotics และสร้างต้นแบบแขนกล
2. เพื่อศึกษาและควบคุมการทำงานของแขนกลด้วยคอมพิวเตอร์
3. นำความรู้ที่ได้จากการศึกษา ทดลองไปประยุกต์การใช้งานและวางแนวทางพัฒนาต่อไป

ในปัจจุบันมีการนำแขนกลมาทำงานแทนมนุษย์กันมากขึ้น โดยเฉพาะในประเทศไทยมีหลายบริษัทได้นำแขนกลเข้ามาช่วยในงานอุตสาหกรรม ซึ่งแขนกลดังกล่าว ส่วนใหญ่ยังต้องสั่งซื้อจากต่างประเทศทำให้มีราคาสูงทำให้ต้นทุนในการผลิตสินค้าสูงไปด้วย การเลือกใช้แขนกลที่ผลิตในประเทศจะช่วยให้ต้นทุนการผลิตสินค้าถูกลง ดังนั้นจึงควรที่จะมีการขยายการศึกษาและค้นคว้าเกี่ยวกับแขนกลให้มากขึ้น เพื่อที่จะนำแขนกลไปใช้ในประโยชน์ต่างๆในงานอุตสาหกรรมได้อย่างเหมาะสม และสิ้นเปลืองงบประมาณการลงทุนน้อยที่สุด โดยใช้มันสมองของคนไทย งานหลายประเภท มีลักษณะการทำงานซ้ำๆกัน โดยเฉพาะงานการประกอบชิ้นงาน ซึ่งไม่ได้มีการซับซ้อนมากนัก น่าจะสามารถใช้เครื่องจักรมาทำงานแทนมนุษย์ได้ และเครื่องจักรที่ว่านี้จะต้องผลิตงานได้สม่ำเสมอ และมีคุณภาพเป็นมาตรฐานเดียวกัน จึงมีความคิดที่จะนำความรู้ที่ได้มาเรียนมาในด้านวิศวกรรมมาเพื่อแก้ปัญหานี้ ในที่สุดจึงเลือกที่จะใช้แขนกลมาเป็นหัวข้อในการทำโครงงานครั้งนี้ การตัดสินใจเลือกทำโครงงานนี้เพื่อที่จะศึกษาพื้นฐานหลักฐานหลักการทำงานของแขนกล โดยให้แขนกลประกอบชิ้นงานสมมติได้อย่างถูกต้อง มีการทำงานแบบอัตโนมัติ และหวังเป็นอย่างยิ่งว่า โครงงานแขนกลชิ้นนี้จะทำให้เกิดแนวคิดที่จะพัฒนาแขนกลให้สามารถทำงานอื่นๆตามที่ต้องการได้ รวมถึงการเพิ่มความสามารถทำงานอื่นๆตามที่ต้องการได้ รวมถึงการเพิ่มความสามารถในการทำงานที่ซับซ้อนขึ้นต่อไปในอนาคต

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1.2 วัตถุประสงค์ของงาน

1. เพื่อแสดงให้เห็นถึงการนำทฤษฎีของแขนกลมาประยุกต์ใช้ในงานอุตสาหกรรม

2. เพื่อศึกษาการออกแบบวัสดุและอุปกรณ์ในการสร้างแขนกล

3. เมื่อประกอบแขนกลเรียบร้อยแล้วสามารถออกแบบโปรแกรมในการควบคุมการทำงานของแขนกลได้

4. แขนกลสามารถทำงานตามจุดประสงค์ที่ต้องการได้

1.3 ขั้นตอนในการศึกษาและจัดทำโครงงาน

การทำโครงงานนี้เริ่มจากการคิดรูปแบบของลักษณะแขนกลว่าควรจะเป็นอย่างไรและทำการออกแบบ ในการออกแบบได้ใช้โปรแกรม Solid Work เป็นโปรแกรมสร้างโมเดลตามที่ต้องการเพื่อให้เห็นภาพจริงของแขนกล จากนั้นทำการสร้างชิ้นส่วนแต่ละส่วนของแขนตามที่ได้ออกแบบไว้ เมื่อได้อุปกรณ์ครบแล้วนำมาประกอบเป็นแขนกลตามที่ได้ออกแบบไว้

เมื่อประกอบแขนกลเรียบร้อยแล้วจะต้องตรวจสอบความสมดุลของแขน จากนั้นมีการเขียนโปรแกรมที่ใช้ในการควบคุมการทำงานของแขนกล โดยการเขียนโปรแกรมแบ่งเป็น 2 ส่วน คือส่วนของไมโครคอนโทรลเลอร์และส่วนของโปรแกรมควบคุมแขนกลผ่านทางคอมพิวเตอร์ โดยในส่วนของไมโครคอนโทรลเลอร์ทางผู้จัดทำได้เลือกใช้ตัวควบคุม dsPIC30F4011 และในส่วนของการติดต่อกับคอมพิวเตอร์ ทางผู้จัดทำได้เลือกใช้โปรแกรมวิชวลเบสิก เมื่อทำการออกแบบโปรแกรมและเขียนโปรแกรมเสร็จแล้วได้ทดลองลงในวงจรเพื่อตรวจสอบว่าแขนกลทำงานได้ตามการควบคุมหรือไม่ ถ้าไม่ได้ก็ทำการตรวจสอบและแก้ไขต่อไป

1.4 รายละเอียดของปริญญานิพนธ์

เนื้อหาที่จะกล่าวในปริญญานิพนธ์ฉบับนี้ประกอบด้วย

บทที่ 1 บทนำ กล่าวนำถึงวัตถุประสงค์ หลักการใหม่ ขั้นตอนการศึกษา และการจัดทำโครงงาน พร้อมทั้งรายละเอียดของปริญญานิพนธ์ของแต่ละบท

บทที่ 2 ทฤษฎีและความรู้ที่เกี่ยวข้อง กล่าวถึงหลักการและทฤษฎีที่เกี่ยวข้องในการประมวลผลภาพ การใช้งานพอร์ตขนาน วงจรอิเล็กทรอนิกส์ที่เกี่ยวข้องและนำเอาความรู้ไปประยุกต์ใช้ในการจัดทำโครงงาน

บทที่ 3 หลักการออกแบบ นำเสนอการประกอบโครงสร้างของระบบ รวมถึงแนวคิดในการออกแบบระบบควบคุม

บทที่ 4 การทดลอง เป็นส่วนการทดสอบองค์ประกอบต่างๆ ในระบบ ตลอดจนการทดลองระบบควบคุมตำแหน่งลูกบอลบนคาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 5 บทวิจารณ์และสรุป จะสรุปผลการดำเนินงาน ปัญหาที่เกิดขึ้น และแนวทางการปรับปรุงพัฒนาโครงงานนี้ต่อไป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 2
ทฤษฎีและความรู้ที่เกี่ยวข้อง

2.1 แขนกล

เมื่อกล่าวถึงหุ่นยนต์ทุกคนมักจะนึกถึงโครงสร้างเหล็กขนาดใหญ่คล้ายมนุษย์ เคลื่อนที่อย่างช้าๆ แต่ปัจจุบันได้มีการพัฒนามากขึ้น มีการนำหุ่นยนต์มาใช้ในงานผลิตอุตสาหกรรมโดยจะมีลักษณะแตกต่างกันออกไปตามลักษณะงานที่นำไปใช้งาน

และเมื่อกล่าวถึงแขนกลมักจะนึกถึงแขนของมนุษย์ โดยมีการเคลื่อนที่ การใช้งานเหมือนกับแขนของมนุษย์ทั้งสิ้น ในเมื่อเราต้องการที่จะสร้างแขนกลที่มีลักษณะเหมือนกับแขนของมนุษย์ เราจึงต้องศึกษาโครงสร้างและลักษณะการเคลื่อนไหวของแขนมนุษย์เสียก่อน

2.1.1 คุณลักษณะของแขนมนุษย์

แขนมนุษย์ประกอบด้วยส่วนที่แยกกันอย่างชัดเจนอยู่สองส่วนคือ ส่วนที่เป็นข้อมือและส่วนของแขนก่อนอื่นมาดูส่วนแรกคือ ข้อมือของมนุษย์ มีลักษณะดังต่อไปนี้

-การหมุนข้อมือ

-การพับข้อมือ

-การบิดข้อมือ

ส่วนที่สองคือ แขน ประกอบด้วยข้อต่อหลัก 2 ข้อต่อ ลักษณะหนึ่งที่สำคัญของโครงสร้างของแขนคือ อัตราส่วนระหว่างความยาวของแขนท่อนบนและแขนท่อนล่าง จะมีค่าประมาณ1.2 : 1 นั่นคือ แขนท่อนล่างจะต้องสั้นกว่าแขนท่อนบน ถ้าเราออกแบบให้แขนท่อนล่างยาวกว่าแขนท่อนบนจะทำให้ประสิทธิภาพของการทำงานของแขนลดลง

2.1.2 ประเภทของแขนกล

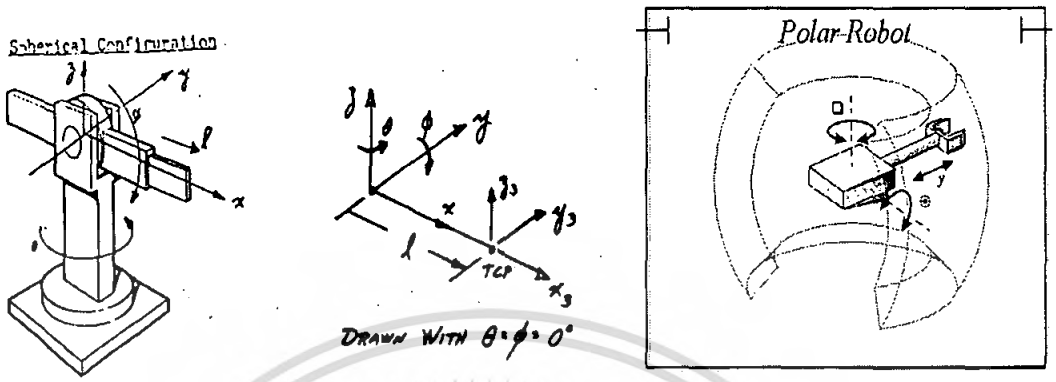
สามารถแบ่งประเภทของแขนกลโดยพิจารณาจากพื้นฐานใหญ่ๆ ดังนี้

1. โครงสร้างภายนอกและการเคลื่อนที่ของแขนกล
2. ทางเดินของแขนกลซึ่งขึ้นอยู่กับการควบคุมการเคลื่อนที่

1.1 การแบ่งประเภทของแขนกลโดยอาศัยการเคลื่อนที่และโครงสร้างภายนอก โดยทั่วไปสามารถแบ่งได้ 5 กลุ่ม ดังนี้

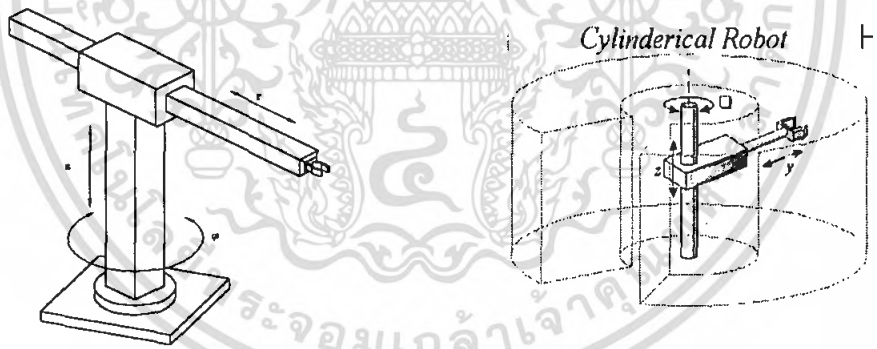
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1.1.1 แขนกลที่เคลื่อนที่ได้ในระบบพิกัดขั้ว (Polar Coordinate Configuration) การเคลื่อนที่ของแขนกลที่มีลักษณะเช่นนี้ อาจจะเรียกอีกอย่างว่า ระบบพิกัดเชิงกลม (Spherical Coordinate Configuration) ทั้งนี้เพราะลักษณะบริเวณที่วาดไปได้จะมีลักษณะเป็นรูปวงกลม



รูปที่ 2.1 แขนกลที่เคลื่อนที่ในระบบพิกัดขั้วหรือพิกัดเชิงกลม

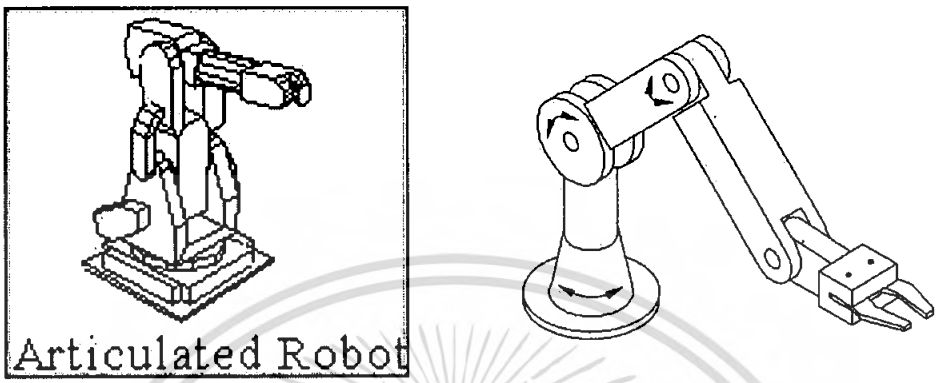
1.1.2 แขนกลที่มีการเคลื่อนที่ได้ในระบบพิกัดเชิงขั้ว (Cylinderical Coordinate Configuration) มีลักษณะการกวาดแขนไปได้เป็นรูปทรงกระบอก



รูปที่ 2.2 แขนกลที่มีการเคลื่อนที่ในระบบพิกัดเชิงขั้ว

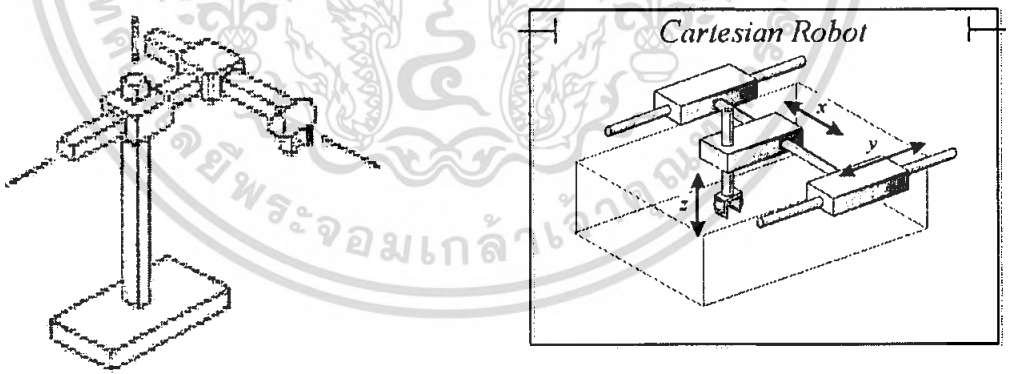
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1.1.3 แขนกลที่มีลักษณะแขนเป็นข้อต่อ (Joint Arm Configuration หรือ Articulated Robot) แขนกลในลักษณะนี้จะเป็นการเลียนแบบแขนของมนุษย์โดยส่วนต่างๆ สามารถยืดหดได้ เช่นเดียวกับไหล่ ตัวแขน ข้อศอกและข้อมือ ทำให้การเคลื่อนที่คล่องตัว โดยจะมีการกวาดได้เป็นลักษณะรูปครึ่งวงกลม



รูปที่ 2.3 แขนกลที่มีลักษณะแขนเป็นข้อต่อ

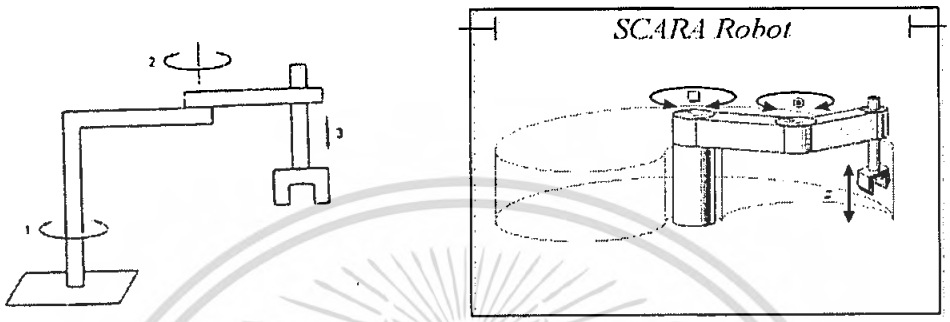
1.1.4 แขนกลที่มีการเคลื่อนที่ได้ในระบบพิกัดคาร์ทิเชียน (Cartesian Coordinate Robot) แขนกลประเภทนี้จะเคลื่อนที่ใน 3 แกนหลักคือ แกน X แกน Y และแกน Z ของพิกัดคาร์ทิเชียน ดังนั้นขอบเขตของการกวาดของมือจะเป็นกรอบสี่เหลี่ยม



รูปที่ 2.4 แขนกลที่เคลื่อนที่ในระบบพิกัดคาร์ทิเชียน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1.1.5 แขนกลที่มีการเคลื่อนที่ได้ในระบบพิกัดเชิงกลมหรือหุ่นยนต์ SCARA(Selective Compliance Assembly Robot Arm) มีลักษณะคล้ายหุ่นยนต์พิกัดทรงกลม มีสัญลักษณ์ RRP แต่แกนตั้งทั้ง3 จะอยู่ในแนวตั้ง โดยข้อที่ 2 ทำให้แขนหมุนรอบแกนตั้งในแนวนอน เหมือนการหมุนของข้อแรก ภาพตัดขวางของแนวนอนในพื้นที่การทำงานค่อนข้างซับซ้อนขึ้นอยู่กับข้อจำกัดในการเคลื่อนที่ของสองแกนแรก



รูปที่ 2.5 แขนกลที่มีการเคลื่อนที่ได้ในระบบพิกัดเชิงกลม

ข้อดี-ข้อเสีย ของหุ่นยนต์แต่ละชนิดนี้แตกต่างกันออกไป เพราะลักษณะทางกายภาพนั้นแตกต่างกัน แต่ถ้ามองในแง่ของการทำงานที่เป็นแบบซ้ำๆที่เดิมตลอด หุ่นยนต์พิกัดคาร์ทีเซียนจะสามารถทำงานได้ดีกว่า คือสามารถเคลื่อนที่ไปหาเป้าหมายโดยความผิดพลาดน้อยที่สุด แต่ถ้ามองในแง่การเข้าถึงวัตถุ ชนิดพิกัดทรงกลมและข้อต่อหมุนจะสามารถเข้าถึงวัตถุได้ดีกว่าชนิดอื่น ชนิดพิกัดทรงกระบอกมีข้อดีที่สามารถยกวัตถุได้มากกว่า ในงานทั่วไปแล้ว ใช้แบบชนิดพิกัดทรงกลมและพิกัดทรงกระบอก เพราะทั้ง 2 ชนิดสามารถที่จะทำงานเป็นแบบ Load และ Unload โดยมีการเคลื่อนที่ของแขนออกไปในด้านข้างได้ดีกว่าชนิดอื่นๆ

1.2 การแบ่งประเภทของแขนกลโดยอาศัยทางเดินแขนกล การควบคุมทางเดินแขนกลแบ่งเป็น 2 แบบใหญ่ๆคือ

1.2.1 การควบคุมแบบจุด แขนกลได้รับการตั้งโปรแกรมให้หยุดที่จุดใดจุดหนึ่งและใช้จุดนั้นในการกำหนดจุดที่ต้องการให้หยุดจุดต่อไป แม้ว่าไม่สามารถควบคุมการเคลื่อนที่ระหว่างจุดให้อยู่ในเส้นทางที่ต้องการได้ แต่การเคลื่อนไปยังจุดที่กำหนดก็ยังถูกต้องตามที่กำหนด ในทางตรงข้ามการควบคุมแบบต่อเนื่องจะสามารถเคลื่อนที่ตามเส้นทางต่างๆได้อย่างถูกต้อง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1.2.2 การควบคุมแบบต่อเนื่อง การควบคุมแบบจุดจะมีความแม่นยำสูงทั้งในแง่ความเที่ยงตรงและการทำงานซ้ำๆ แม้ว่าการควบคุมแบบต่อเนื่องจะเคลื่อนที่ได้นุ่มนวลกว่าในเส้นทางที่ได้ระบุไว้ แต่มีข้อเสียในด้านความเร็ว ซึ่งเป็นฟังก์ชันของจังหวะการเคลื่อนที่ที่คำนวณโดยคอมพิวเตอร์ในลักษณะเวลาจริง ข้อเสียนี้จะทำให้ความเร็วของการเคลื่อนที่แบบต่อเนื่องลดลงถึง 15-25 % และประสิทธิภาพการทำงานจะลดลงเมื่อเปรียบเทียบกับแขนกลตัวเดียวกันที่ควบคุมแบบจุด

2.2 การหมุน(Orientation)

ให้ x และ y เป็นเวคเตอร์ใน R^3 และ θ เป็นมุมจาก x ไป y ดังนั้น

$$X \cdot Y = \|x\| \|y\| \cos\theta$$

ดังนั้น Dot Product จึงเป็นการวัดการหมุนระหว่างสองเวคเตอร์นั้น นอกจากนี้ยังมีการกระทำของเวคเตอร์อีกชนิดที่เป็นประโยชน์ในการวิเคราะห์แขนกลคือ Cross Product โดยนิยามได้ดังนี้

นิยาม 5 Cross Product ของเวคเตอร์ u และ v ใน R^3 คือเวคเตอร์ w=u x v ซึ่งตั้งฉากกับกฎมือขวาและ

$$w \equiv \det\left\{\begin{bmatrix} i^1 & i^2 & i^3 \\ u_1 & u_2 & u_3 \\ v_1 & v_2 & v_3 \end{bmatrix}\right\} = \begin{bmatrix} (u_2 v_3 - u_3 v_2) i^1 \\ (u_3 v_1 - u_1 v_3) i^2 \\ (u_1 v_2 - u_2 v_1) i^3 \end{bmatrix}$$

โดยเวคเตอร์หนึ่งหน่วย $\{i^1, i^2, i^3\}$ เป็นเวคเตอร์ตามแนวแกนของโครงพิกัดฉากตามกฎมือขวา ดังในรูปที่ 2.12 โดย $i^1 = i^2 \times i^3$, $i^2 = i^3 \times i^1$ และ $i^3 = i^1 \times i^2$ ถ้า θ เป็นมุมจาก u ไป v ดังนั้นCross Product

$$\|x \times y\| = \|u\| \|v\| \sin\theta$$

2.2.1 โครงพิกัด(Coordinate Frames)

นิยาม 1 พิกัด (Coordinates) ให้ p เป็นเวคเตอร์ใน R^n และ $x = \{x^1, x^2, x^3, \ldots, x^n\}$ เป็น Complete Orthogonal Set ของ R^n

ดังนั้นพิกัดของ p เทียบกับ x เขียนเป็น $[p]^x$ และนิยามดังนี้

$$P = \sum_{k=1}^{n} [p]_k^x \, x^k$$

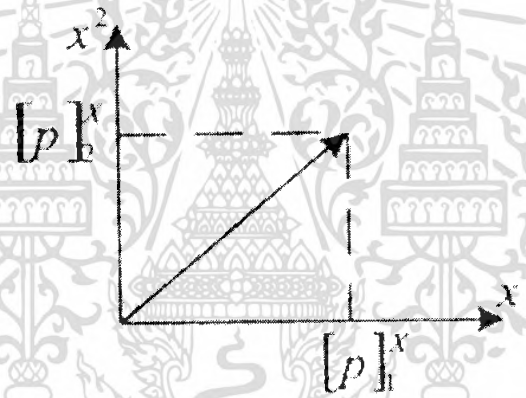
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Complete Orthogonal Set X บางครั้งเรียกว่า โครงพิกัดหนึ่งหน่วยตั้งฉาก(Orthogonal coordinate frame) ซึ่งการใช้เวคเตอร์โครงพิกัดเป็นยูนิตเวคเตอร์ที่ตั้งฉากกันนี้ ทำให้หาพิกัดของเวคเตอร์อื่นเทียบกับโครงพิกัดนี้ได้ง่าย โดยใช้ทฤษฎีต่อไปนี้

โครงพิกัดฉาก (Orthogonal coordinates) ให้ p เป็นเวคเตอร์ใน R^n และ$[p]^x$ เป็นพิกัดของของ p เทียบกับโครงพิกัดฉาก $x = \{x^1,x^2,x^3,\ldots,x^n\}$ ดังนั้นพิกัดที่ k ของ p เทียบกับ x คือ

$$[p]_k^x = p \cdot x^k : 1 \le k \le n$$

จะเห็นได้ว่า พิกัดที่ k ของ p เทียบกับโครงพิกัดฉาก x ก็คือ Dot Product ของ p กับสมาชิกตัวที่ k ของเซต x โดยสามารถแสดงความสัมพันธ์ระหว่างเวคเตอร์กับพิกัดได้ดังรูปที่ 2.13 ซึ่งเป็นปริภูม 2 มิติ โดย p คือผลบวกของภาพฉายตั้งฉากลงบนเวคเตอร์ x^1,x^2 ตามลำดับ



รูปที่ 2.6 พิกัดของ p เทียบกับโครงพิกัด x

คำตอบของปัญหา Direct Kinematics ในแขนกล เราต้องแสดงตำแหน่งและการหมุนของเครื่องมือที่เคลื่อนที่เทียบกับโครงพิกัดที่ติดกับฐานที่อยู่นิ่ง ดังนั้นจึงต้องมีการถ่ายโอนพิกัดจากเรื่องมือไปข้อมือ ข้อมือไปข้อศอก ข้อศอกไปไหล่ และต่อไปเรื่อยๆ การถ่ายโอนพิกัดแต่ละอันสามารถแสดงได้ด้วยเมตริกซ์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.2.2 การถ่ายโอนพิกัด(Coordinate Transformations)

ให้ F=$\{f^1, f^2, f^3, \ldots, f^n\}$และ M=$\{m^1, m^2, m^3, \ldots, m^n\}$ เป็นโครงพิกัด R^n และ F เป็นโครงพิกัด Orthogonal ให้ A เป็นเมตริกซ์ขนาด n×n โดย $A_{kj} = f^k \cdot m^j$ เมื่อ $1 \leq k,j \leq n$ ดังนั้น สำหรับแต่ละจุด p ใน R^n

$$[p]^F = A[p]^M$$

ปัญหาของการถ่ายโอนพิกัดคือการให้พิกัดของ p เทียบกับโครงพิกัด M มาแล้วให้หาพิกัดของ p เทียบกับโครงพิกัด F เมตริกซ์ A เรียกว่า เมตริกซ์ถ่ายโอนพิกัด (Coordinate Transformation Matrix)เป็นคอลัมน์ที่ j ของ A จาก $a^j = [m^j]^F$ สำหรับ$1 \leq j \leq n$

หรือกล่าวได้ว่า คอลัมน์ที่ j ของเมตริกซ์ A คือพิกัดของเวคเตอร์ลำดับที่ j ของโครงพิกัดต้นกำเนิด M เทียบกับโครงพิกัดปลายทาง F

อินเวร์สของการถ่ายโอนโครงพิกัด(Inverse Coordinate Transformation) ให้ F และ M เป็นโครงพิกัดฉาก Orthogonal ใน R^n มีจุดกำเนิดเดียวกัน และเป็นเมตริกซ์ถ่ายโอนพิกัดจาก M ไป F ดังนั้นเมตริกซ์ถ่ายโอนพิกัดจาก F ไป M คือ A^{-1} โดย

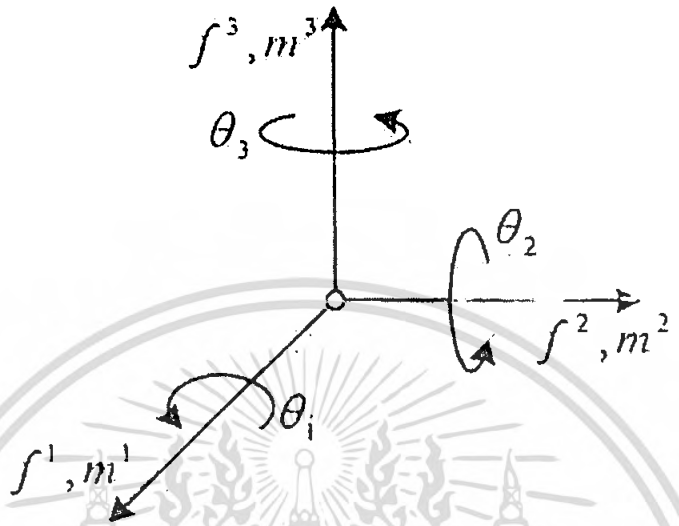
$$A^{-1} = A^T$$

ในการระบุตำแหน่งและการหมุนของเครื่องมือในเทอมของพิกัดที่ติดอยู่กับฐานที่อยู่นิ่ง การถ่ายโอนพิกัดจึงเกี่ยวข้องกับทั้งการหมุนและการเคลื่อนที่เปลี่ยนตำแหน่ง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

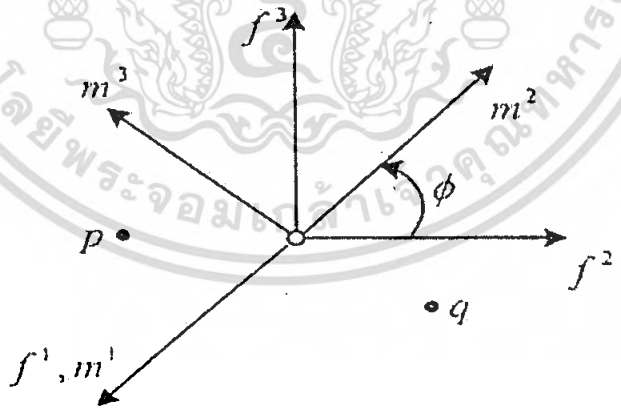
2.2.3 การหมุนพื้นฐาน(Fundamental Rotations)

ถ้าโครงพิกัดเคลื่อนที่ M รอบยูนิตเวคเตอร์อันหนึ่งของโครงพิกัดอยู่นิ่ง F ดังนั้นเมตริกซ์การถ่ายโอนพิกัดที่ได้เรียกว่า เมตริกซ์การหมุนพื้นฐาน ในปริภูมิ R^3 มีทางเป็นไปได้ 3 ทางดังแสดงในรูป 2.14



รูปที่ 2.7 การหมุนพื้นฐานในปริภูมิ R^3

ถ้าเราหมุนโครงพิกัดเคลื่อนที่ M รอบแกน f^1 ของโครงพิกัดอยู่นิ่ง F ให้ ϕ เป็นองศาการหมุนวัดตามกฎมือขวา ดังรูป 2.8



รูปที่ 2.8 การหมุนของโครงพิกัดรอบแกนของโครงพิกัดอยู่นิ่งที่ 1

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ให้ $R_1(\phi)$ เป็นเมตริกซ์การถ่ายโอนพิกัดเคลื่อนที่ M ไป F (จากการถ่ายโอนพิกัด) จะได้

$$R_1(\phi) = \begin{bmatrix} f^1 \cdot m^1 & f^1 \cdot m^2 & f^1 \cdot m^3 \\ f^2 \cdot m^1 & f^2 \cdot m^2 & f^2 \cdot m^3 \\ f^3 \cdot m^1 & f^3 \cdot m^2 & f^3 \cdot m^3 \end{bmatrix}$$

เนื่องจากเราหมุนรอบแกน f^1 จากรูปที่ 2.15 จะได้ $f^1 = m^1$ แทนใน $R_1(\phi)$ และจากรูปที่ 2.15 เวคเตอร์ $\{f^1, f^2, f^3\}$ และ $\{m^1, m^2, m^3\}$ เป็นเซตของยูนิตเวคเตอร์ที่ตั้งฉากกัน ดังนั้นจะได้

$$R_1(\phi) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & f^2 \cdot m^2 & f^2 \cdot m^3 \\ 0 & f^3 \cdot m^2 & f^3 \cdot m^3 \end{bmatrix}$$

จากรูปที่ 2.15 เวคเตอร์ $\{f^1, f^2, f^3\}$ อยู่ในระนาบตั้งฉากกับ $f^1 = m^1$ จากการหมุน (Orientation) จะได้ว่า Dot Product ของยูนิตเวคเตอร์เท่ากับค่า cosine ของมุมระหว่าง 2เวคเตอร์นั้น จากรูปที่ 2.15 มุมจาก f^2 ไป m^2 และ f^3 ไป m^3 เท่ากับ ϕ มุมจาก f^2 ไป m^3 เท่ากับ $\pi/2 + \phi$ และมุมจาก f^3 ไป m^2 เท่ากับ $-\pi/2 + \phi$ ดังนั้นจะได้

$$R_1(\phi) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\phi & -\sin\phi \\ 0 & \sin\phi & \cos\phi \end{bmatrix}$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ในทำนองเดียวกันถ้า $R_2(\phi)$ และ $R_3(\phi)$ แสดงการหมุนของโครงพิกัด Mรอบ f^2และ f^3 ของ F ตามลำดับจะได้

$$R_2(\phi) = \begin{bmatrix} \cos\phi & 0 & \sin\phi \\ 0 & 1 & 0 \\ -\sin\phi & 0 & \cos\phi \end{bmatrix}$$

$$R_3(\phi) = \begin{bmatrix} \cos\phi & -\sin\phi & 0 \\ \sin\phi & \cos\phi & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

แถวที่ k และคอลัมน์ที่ k ของ $R_k(\phi)$ จะเหมือนกับแถวที่ k คอลัมน์ที่ k ของเมตริกซ์เอกลักษณ์ *I*ส่วน Submatrix 2×2 จะได้ว่าเทอมในแนวทแยงมุมจะเป็น cos(ส่วนเทอมนอกแนวทแยงมุมเป็น$\pm\sin\phi$ เมื่อ ϕเป็นมุมการหมุน โดยเครื่องหมายของเทอมที่อยู่เหนือแนวทแยงมุมเป็น $(-1)^k$ สำหรับ $R_k(\phi)$ เช่น $R_1(\phi)$จะได้ –sin ϕเป็นต้น

2.2.4 การเลื่อนตำแหน่งและการหมุน(Translations and Rotations)

การกระทำพื้นฐานของการเลื่อนตำแหน่งและการหมุน จัดว่าเป็นกรณีพิเศษในเมตริกซ์ถ่ายโอนโฮโมจีเนียส 4×4 เช่น กรณีการหมุนสมมติให้ F และ M เป็นโครงพิกัดฉากที่เริ่มต้นทับกันสนิท ถ้าหมุน M ไปϕ รอบเวคเตอร์ลำดับที่ k ของ F ดังนั้นจะแสดงในเทอมของพิกัดโฮโมจีเนียสเป็นเมตริกซ์ 4×4 ,Rot(ϕ,k) ได้ว่า

$$Rot(\phi,k) \cong \left[\begin{array}{ccc:c} & & & 0 \\ & R_k(\phi) & & 0 \\ \hdashline 0 & 0 & & 1 \end{array}\right] \qquad 1 \le k \le 3$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.2.5 ความแม่นยำของการเคลื่อนที่

ความสำคัญประการหนึ่งของการสร้างหุ่นยนต์คือความแม่นยำในการทำงาน ซึ่งความแม่นยำนี้ขึ้นอยู่กับตัวแปร 3 อย่าง คือ

1.Spatial Resolution คือช่วงการเคลื่อนที่ที่มีระยะสั้นที่สุดที่หุ่นยนต์แต่ละตัวสามารถที่จะทำได้ ซึ่ง Spatial Resolution นี้ขึ้นอยู่กับองค์ประกอบสำคัญ 2 ประการ

1.) ระบบการควบคุม (Control System) ระบบการควบคุมนี้จะรวมถึงการวัดสัญญาณป้อนกลับของหุ่นยนต์ด้วย ช่วงการเคลื่อนที่ที่ระบบการควบคุมสามารถที่จะทำได้ขึ้นอยู่กับหน่วยความจำหลักของคอมพิวเตอร์ ที่มีหน่วยความจำ 8 บิตจะสามารถแบ่งการเคลื่อนที่ได้เป็น 256 ช่วงคือ ช่วงการเคลื่อนที่ที่คอมพิวเตอร์แบ่งได้มีค่าเท่ากับ 2n เมื่อ n คือหน่วยความจำหลักคอมพิวเตอร์

2.) ความคลาดเคลื่อนเชิงกล (Mechanical inaccuracy) ของหุ่นยนต์แต่ละตัวขึ้นอยู่กับลักษณะของข้อหมุน(joint) และข้อต่อ (link) และระบบต้นกำลังของหุ่นยนต์ตัวนั้นด้วย

2.Accurary คือตัวที่แสดงถึงความสามารถของหุ่นยนต์ในการเคลื่อนที่เข้าใกล้ตามจุดเป้าหมายตามที่เราสั่ง Accuracy สามารถที่จะกำหนดให้อยู่ในรูปเทอมของ Spatial Resolution ได้ทั้งนี้เพราะว่าการเคลื่อนที่ให้เข้าใกล้จุดเป้าหมาย ก็ต้องขึ้นอยู่กับช่วงของการเคลื่อนที่ว่ามีความละเอียดมากน้อยเพียงใด ในการทำงานเราต้องวางจุดที่เราต้องการให้หุ่นยนต์ทำงานอยู่ระหว่างกลางของตำแหน่งของการเคลื่อนที่ของหุ่นยนต์ ทั้งนี้เพราะว่าความคลาดเคลื่อนเชิงกล มีผลต่อความแม่นยำของหุ่นยนต์ ความแม่นยำของหุ่นยนต์กำหนดให้เท่ากับครึ่งหนึ่งของระยะทางการเคลื่อนที่ที่สั้นที่สุดของหุ่นยนต์ที่สามารถทำได้ โดยขึ้นอยู่กับองค์ประกอบต่อไปนี้

1.) พื้นที่การทำงานของหุ่นยนต์ ถ้าแขนทำงานในพื้นที่การทำงานจะมีความแม่นยำมากกว่าเมื่อแขนออกนอกพื้นที่การทำงาน

2.)วงรอบการทำงาน ถ้าวงรอบการทำงานเป็นวงนอบที่แน่นอนความแม่นยำจะมีมากขึ้น

3.)น้ำหนักที่ได้รับ ถ้าหุ่นยนต์ทำงานโดยรับน้ำหนักมากๆ ความแม่นยำจะลดลง

3. Repeatability คือความสามารถของหุ่นยนต์ในการกลับมาทำงานซ้ำที่เดิม หลายๆครั้งได้ เนื่องจากสาเหตุต่างๆ เช่น Backlash ในเฟืองและความยืดหยุ่น (Flexibility) ของส่วนต่างๆซึ่งมักทำให้เกิดความผิดพลาดใน Repeatability

2.2.6 สมการแขนกล (Direct Kinematics : The Arm Equation)

แขนกลมีลักษณะเป็นวัตถุเกร็ง (Rigid body) เป็น link หลายๆอันนำมาเชื่อมต่อเข้าด้วยกัน โดยใช้ข้อต่อดังรูปที่ 2.11 ปลายข้างหนึ่งของ link ริมสุดจะติดแน่นกับฐานส่วนปลายของ link อีกอันที่อยู่คนละด้านจะเคลื่อนที่ได้อย่างอิสระ ปลายข้างที่เคลื่อนที่นี้จะต่อเข้ากับเครื่องมือ (Toolor End-effector) ตามปกติมีข้อต่อ 2 ชนิดที่ใช้เชื่อมต่อ links คือข้อต่อหมุน (Revolute joint) และข้อต่อเลื่อน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จุดประสงค์คือ การควบคุมทั้งตำแหน่งและการหมุนของเครื่องมือในระนาบ 3 มิติ เพื่อโปรแกรมให้เครื่องมือเคลื่อนที่ไปตามเส้นทางที่กำหนดไว้และใช้ในการเคลื่อนย้ายวัตถุในพื้นที่การทำงาน โดยการโปรแกรมการเคลื่อนที่ของเครื่องมือนั้น จำเป็นต้องคำนวณความสัมพันธ์ของตัวแปรข้อต่อกับตำแหน่งและการหมุนของอุปกรณ์ ซึ่งเรียกว่าปัญหา Direct Kinematics โดยนิยามดังนี้

ปัญหา Direct Kinematics คือการกำหนดเวกเตอร์ของตัวแปรข้อต่อ (Joint Variables) ของแขนกลโดยแยกตามลักษณะของข้อต่อเป็น

-ข้อต่อหมุน ตัวแปรได้แก่ ค่าองศาการหมุนรอบตัวแปรข้อต่อ

-ข้อต่อเลื่อน ตัวแปรได้แก่ ระยะการเคลื่อนที่ตามแนวแกนข้อต่อ

แล้วทำการคำนวณหาตำแหน่ง (Position) และการหมุน (Orientation)ของเครื่องมือเทียบกับโครงพิกัดที่ติดอยู่กับฐานของแขนกล

Dot and Cross Products

เวกเตอร์ในปริภูมิ n มิติ R^n สามารถแทนด้วยลูกศรพุ่งออกจากจุดกำเนิดดังรูปที่2..12 ซึ่งเป็นกรณี n=3

รูปที่ 2.9 โครงพิกัดฉากใน R^3

นิยาม Dot Product ของ 2 เวคเตอร์ X และ Y ใน R^n เขียนเป็น $X \cdot Y$

$$X \cdot Y \equiv \sum_{k=1}^{n} X_k Y_k$$

ถ้าใช้การ Transpose ของเมตริกซ์ จะแสดงได้ในรูป $X \cdot Y = X^T Y$ โดย Dot product ใน R^n มีคุณสมบัติพื้นฐานดังนี้

ให้ {x , y , z} เป็นเวคเตอร์ใน R^n และ {α , ß} เป็นสเกลาร์ ดังนั้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1. $x \cdot x \geq 0$
2. $x \cdot x = 0 \Leftrightarrow x = 0$
3. $x \cdot y = y \cdot x$
4. $(\alpha x + \beta y) \cdot z = \alpha(x \cdot z) + \beta(y \cdot z)$

นิยาม 2 การตั้งฉาก (Orthogonality) เวคเตอร์ x และ y ใน R^n จะตั้งฉากกันก็ต่อเมื่อ $x \cdot y = 0$

นิยาม 3 Completenness เซตของเวคเตอร์ที่ตั้งฉากกัน $\{ x^1, x^2, x^3, \ldots, x^n \}$ ใน R^n เป็น complete ก็ต่อเมื่อ

$$y \cdot x^k = 0 \text{ เมื่อ } 1 \leq k \leq n \Rightarrow y = 0$$

จำนวนเวคเตอร์ใน complete Orthogonal Set เรียกว่า มิติ ดังนั้น R^3 ก็คือปริภูมิสามมิติที่มีรูปทั่วไปคือ R^n

นิยาม4 ขนาด (Norm) ของเวคเตอร์ x ใน R^n เขียนเป็น $\| x \|$ นิยามได้ว่า

$$\|x\| \cong (x \cdot x)^{1/2} = (\sum_{k=1}^{n} x_k^2)^{1/2}$$

โดยยูนิคเวคเตอร์ที่ตั้งฉากกันเรียกว่า Orthogonal

2.3 Servo motor

Servo motor คือ มอเตอร์ไฟฟ้ากระแสตรง (DC motor) ที่ถูกประกอบรวมกับชุดเกียร์และส่วนควบคุมต่างๆไว้ในโมดูลเดียวกันหรือภายในกล่องพลาสติกเดียวกัน โดยมอเตอร์ชนิดนี้จะมีสาย ต่อใช้งานเพียง 3 เส้นเท่านั้น คือ VCC,GND และสายสัญญาณควบคุม(Control Line) ซึ่ง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สามารถควบคุมให้มอเตอร์หมุนซ้าย หรือ ขวาได้จากสายสัญญาณเพียงเส้นเดียวโดยสัญญาณที่ใช้ควบคุมนี้จะเป็นสัญญาณพัลส์วิดมอด (PWM) แบบ TTL Level ระดับแรงดันที่จ่ายให้มอเตอร์นี้จะอยู่ในช่วงประมาณ 4 ถึง 6 โวลท์ ขึ้นอยู่กับคุณสมบัติของมอเตอร์แต่ละตัวข้อดีของมอเตอร์ชนิดนี้ก็คือ จะมีขนาดเล็กน้ำหนักเบา,ให้แรงบิตสูง ,กินพลังงานน้อยและสามารถควบคุมด้วยแรงดันลอจิกที่เป็น TTL ได้โดยตรงไม่จำเป็นต้องต่อวงจรขับ(Driver) อื่นๆเพราะมอเตอร์ชนิดนี้จะมีวงจรควบคุมบรรจุไว้ภายในอยู่แล้ว ซึ่งมอเตอร์ชนิดนี้สามารถควบคุมให้หมุนไปในตำแหน่งหรือทิศทางองศาที่

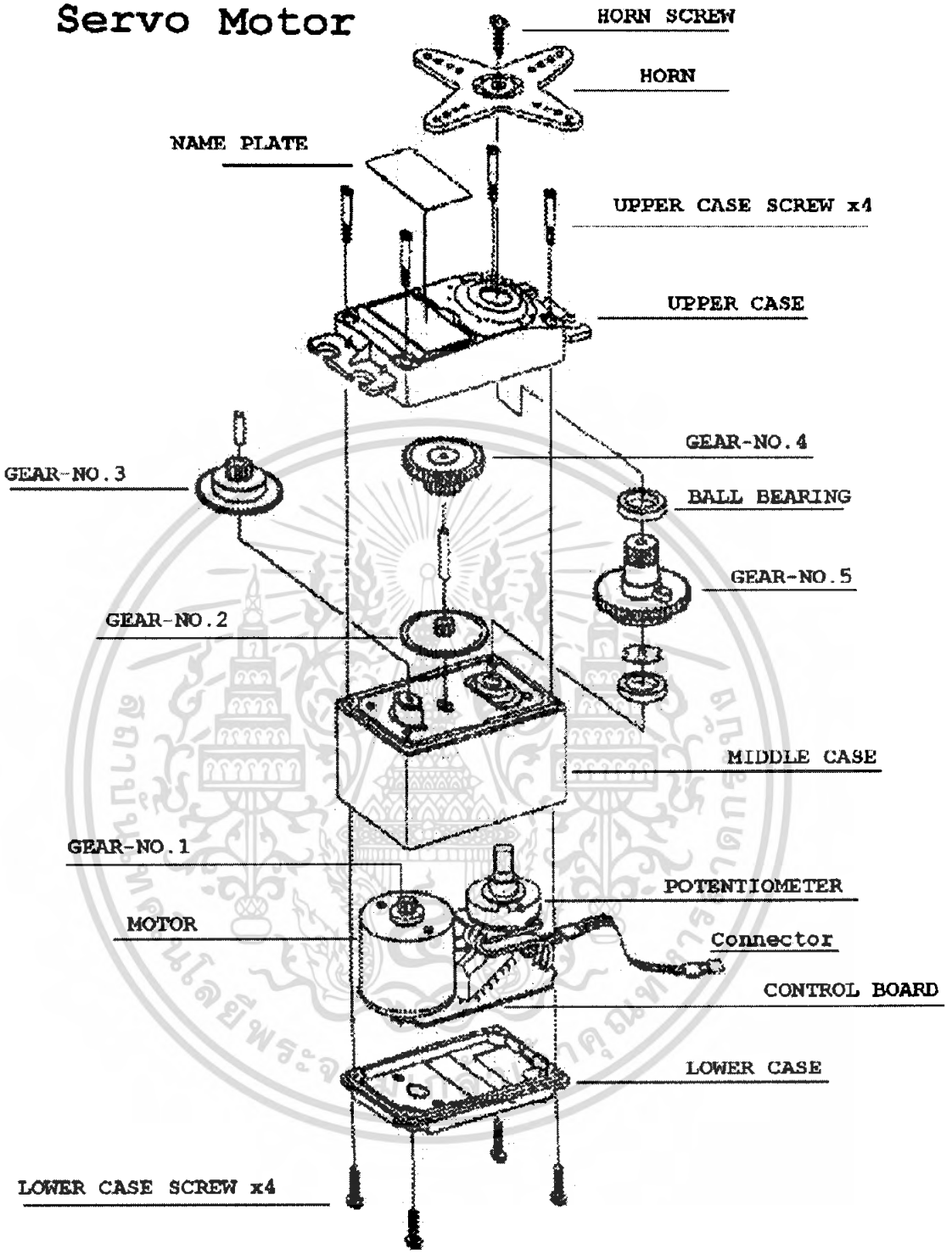
ต้องการได้ โดยอาศัยสัญญาณความกว้างพัลส์ที่ป้อนให้มอเตอร์ แต่เซอร์โวมอเตอร์นี้จะหมุนได้แค่เพียงในช่วงประมาณ 180° หรือครึ่งรอบเท่านั้นหรือบางรุ่นอาจหมุนได้ถึง 210°แต่จะไม่สามารถหมุนเป็นวงรอบได้

เนื่องจากโครงสร้างภายในจะประกอบด้วย ตัวต้านทานชนิดปรับค่าได้ (VR) ที่ทำหน้าที่ตรวจสอบตำแหน่งการหมุนของมอเตอร์และตัวต้านทานนี้จะถูกยึดติดกับแกนหมุนของมอเตอร์ ซึ่งจากการที่ตัวต้านทานปรับค่านี้ไม่สามารถหมุนเป็นวงรอบได้ ดังนั้นเซอร์โวมอเตอร์จึงถูกออกแบบให้หมุนได้เพียงแค่ประมาณ 180 องศาหรือครึ่งรอบเท่านั้นเพื่อป้องกันความเสียหายที่จะเกิดกับตัวต้านทานปรับค่าได้

2.3.1 ส่วนประกอบของ Servo motor

72877

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

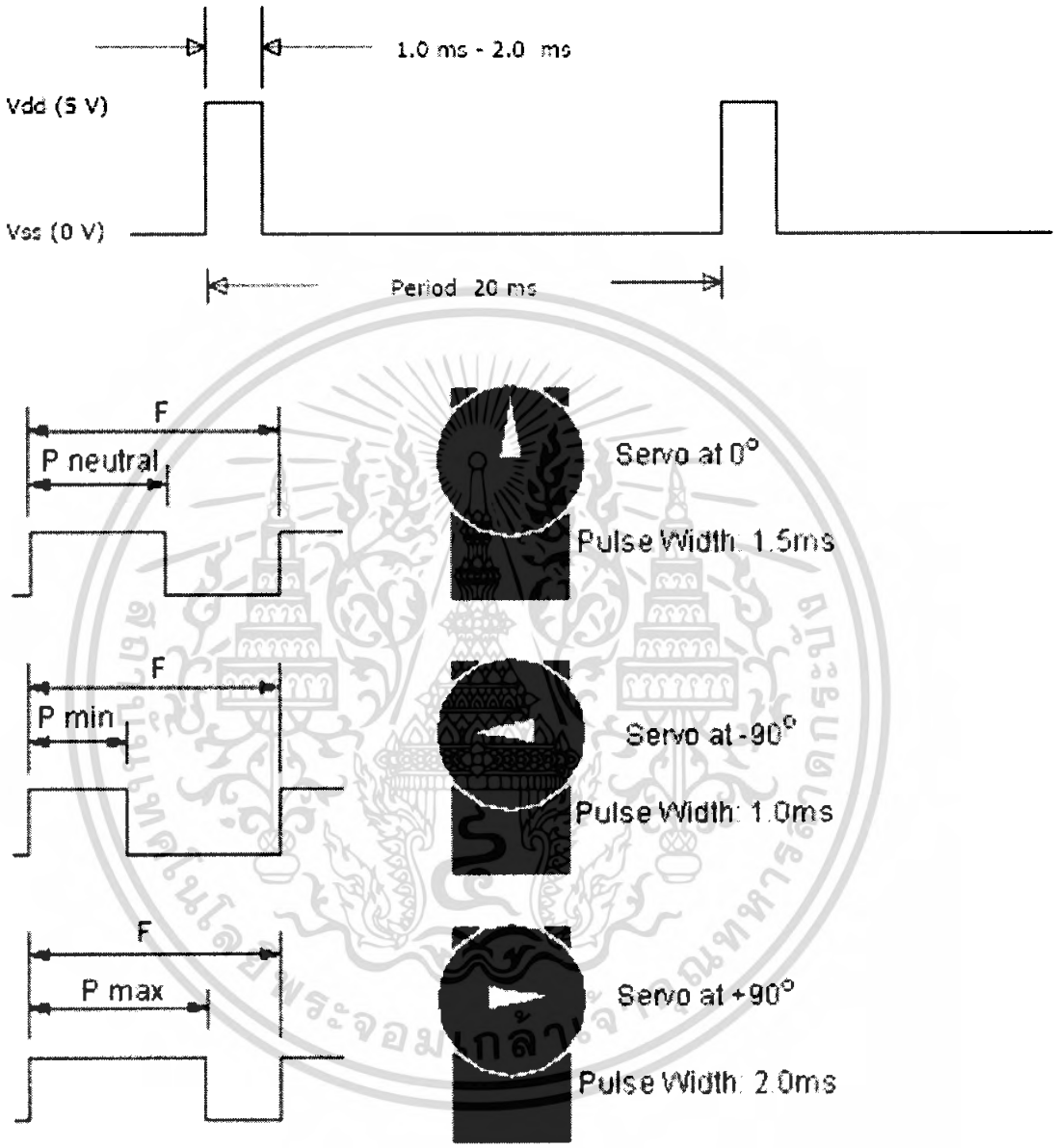


รูปที่ 2.10 ส่วนประกอบของเซอร์โวมอเตอร์

2.3.2 หลักการทำงานของ Servo motor

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การควบคุมการทำงานของเซอร์โวมอเตอร์ทำได้โดย การป้อนสัญญาณความกว้างพัลส์ให้กับมอเตอร์ซึ่งตำแหน่งและทิศทางการหมุนของมอเตอร์นี้จะขึ้นอยู่กับขนาดของความกว้างของพัลส์นั้นๆโดยทั่วไปแล้วความกว้างของสัญญาณพัลส์จะมีจุดให้อ้างอิง 3 จุด ดังรูป คือ



รูปที่ 2.11 หลักการทำงานของเซอร์โวมอเตอร์

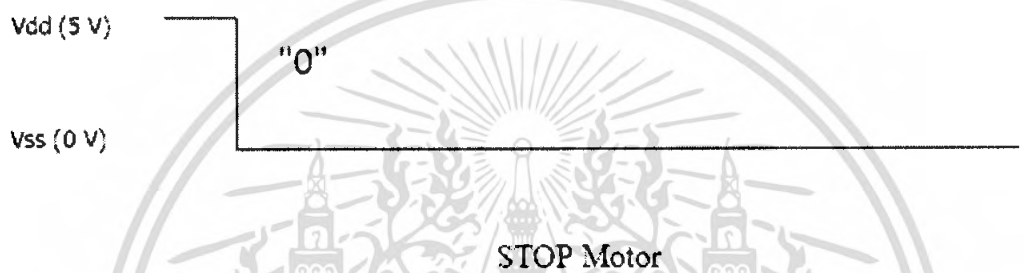
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- สัญญาณความกว้างพัลส์ขนาด 1.5 ms จะควบคุมให้เซอร์โวมอเตอร์หมุนไปอยู่ที่ตำแหน่งมุม 0 องศาหรือจุด กึ่งกลางของมอเตอร์

- สัญญาณความกว้างพัลส์ขนาด 1 ms จะควบคุมให้เซอร์โวมอเตอร์หมุนไปอยู่ที่ตำแหน่งมุม – 90 องศาหรือในทิศทางทวนเข็มนาฬิกา

- สัญญาณความกว้างพัลส์ขนาด 2 ms จะควบคุมให้เซอร์โวมอเตอร์หมุนไปอยู่ที่ตำแหน่งมุม + 90 องศา หรือในทิศทางตามเข็มนาฬิกา

-การควบคุมให้มอเตอร์หยุดหมุนทำได้โดยการส่งลอจิก “0” ให้กับมอเตอร์หรือก็คือการไม่จ่ายสัญญาณพัลส์ให้กับมอเตอร์นั่นเอง



รูปที่ 2.12 การควบคุมมอเตอร์หยุดหมุน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.4 ส่วนควบคุมโดยไมโครคอนโทรลเลอร์

ในหุ่นยนต์สมัยใหม่จะใช้อุปกรณ์อิเล็กทรอนิกส์ที่เรียกว่า " ไมโครคอนโทรลเลอร์ (Microcontroller) " เป็นอุปกรณ์หลักในการควบคุมและประมวลผล โดยในส่วนควบคุมนี้จะบรรจุโปรแกรมควบคุมที่ผู้ใช้งานเขียนขึ้นมา จะถูกบรรจุลงในหน่วยความจำ และสามารถเปลี่ยนแปลงแก้ไขได้

ในโครงงานนี้ใช้ตัวไมโครคอนโทรลเลอร์ประเภท dsPIC เป็นตัวควบคุม

2.4.1 ข้อมูลเบื้องต้นของ dsPIC

dsPIC คือชื่อของไมโครคอนโทรลเลอร์ 16 บิตจาก Microchip Technology Inc. ผู้ผลิตไมโครคอนโทรลเลอร์ PIC ซึ่งรู้จักกันเป็นอย่างดีในแวดวงนักพัฒนาระบบไมโครคอนโทรลเลอร์ โดยบริษัทไมโครชิพ เทคโนโลยี จำกัด (Microchip Technology) ได้กำหนดชื่ออย่างเป็นทางการสำหรับไมโครคอนโทรลเลอร์นุกรมใหม่นี้ว่าดิจิตอลซิกแนลคอนโทรลเลอร์ (Digital Signal Controller หรือ DSC) นั่นหมายความว่า dsPIC เป็นไมโครคอนโทรลเลอร์ที่ได้รับการออกแบบมาเป็นพิเศษเพื่องานประมวลผลสัญญาณดิจิตอลสำหรับสร้างระบบควบคุมอัตโนมัติที่มีความสามารถสูง

2.4.2 คุณสมบัติเด่นโดยรวมของ dsPIC

1. คุณสมบัติของซีพียู

- ความเร็วในการทำงานสูงถึง 30 ล้านคำสั่งต่อวินาที
- มี 84 คำสั่งภาษาแอสเซมบลีมาตรฐาน รองรับรูปแบบการอ้างแอดเดรสได้อย่างอิสระ
- ชุดคำสั่งมีขนาด 24 บิต สามารถประมวลผลข้อมูลได้ 16 บิต
- มีหน่วยความจำโปรแกรมเป็นแบบแฟลช สามารถลบและเขียนใหม่ได้ไม่น้อยกว่า 100,000 ครั้ง สามารถป้องกันการอ่านได้และสามารถโปรแกรมตัวเอง โดยใช้กระบวนการทางซอฟต์แวร์
- มีหน่วยความจำข้อมูลอีอีพรอมที่สามารถลบและเขียนใหม่ได้ไม่น้อยกว่า 100,000 ครั้ง
- มีอินเตอร์รัปต์เวกเตอร์จำนวนมาก จึงรองรับการตอบสนองสัญญาณอินเตอร์รัปต์ได้ดี
- มีวงจรตรวจจับแรงดันไฟเลี้ยงต่ำกว่ากำหนดแบบโปรแกรมได้
- มีเพาเวอร์-ออนรีเซต, เพาเวอร์-อัปไทเมอร์ และออสซิลเลเตอร์สตาร์ต-อัปไทเมอร์
- มีวอตช์ด็อกไทเมอร์แบบโปรแกรมได้
- มีวงจรตรวจสอบการทำงานของวงจรกำเนิดสัญญาณนาฬิกา
- รองรับการโปรแกรมในวงจรแบบอนุกรม(ICSP : In-Circuit Serial Programming)
- สามารถเลือกโหมดการใช้พลังงานได้

2. คุณสมบัติด้านการประมวลสัญญาณดิจิตอล

- มีแอกคิวมูเลเตอร์ขนาด 40 บิต 2 ตัว รองรับการประมวลผลทางคณิตศาสตร์ได้เป็นอย่างดี

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- มีหน่วยประมวลผลด้านการคูณและหารเลข 17 บิตในรูปของฮาร์ดแวร์ จึงทำให้สามารถคูณและหารเลขได้อย่างรวดเร็ว
- ทำการคูณเลข 16 บิต ได้ภายในสัญญาณนาฬิกาเพียง 1 ไซเกิล
- มีตัวเลื่อนข้อมูลบาร์เรล 40 สเตจ ช่วยให้การประมวลผลข้อมูลที่จำนวนบิตมากๆสามารถทำได้อย่างรวดเร็ว
- มีวงจรเฟตช์ข้อมูลคู่ จึงทำให้สามารถประมวลผลข้อมูลได้อย่างรวดเร็ว

3. คุณสมบัติของโมดูลฟังก์ชันพิเศษ

- สามารถจ่ายกระแสออกทางขาพอร์ตได้ 25 mA ทั้งแบบกระแสซิงก์และซอร์ส
- ไทเมอร์/เคาน์เตอร์มีขนาด 16 บิต ไม่น้อยกว่า 3 ตัวต่อใช้งานร่วมกันเป็นไทเมอร์ 32 บิตได้
- มีโมดูลตรวจจับและเปรียบเทียบสัญญาณดิจิตอล
- มีส่วนเชื่อมต่ออุปกรณ์อนุกรมทั้งแบบ SPI และผ่านระบบบัส I2C
- มีโมดูลสื่อสารข้อมูลอนุกรม UART พร้อมบัฟเฟอร์แบบ FIFO
- มีวงจรแปลงสัญญาณอนาลอกเป็นดิจิตอล ความละเอียด 10 หรือ 12 บิต (เป็นคุณสมบัติที่มีในบางเบอร์)
- มีโมดูลสร้างสัญญาณ PWM สำหรับควบคุมมอเตอร์
- มีโมดูลเชื่อมต่อตัวเข้ารหัสแบบควอดราเจอร์

สำหรับในโครงงานนี้คณะผู้จัดทำได้เลือกใช้ไมโครคอนโทรลเลอร์เบอร์ dsPIC30F4011 ซึ่งมีขาการใช้งาน 40 ขา และมีไทเมอร์ 5 ตัวด้วยกัน

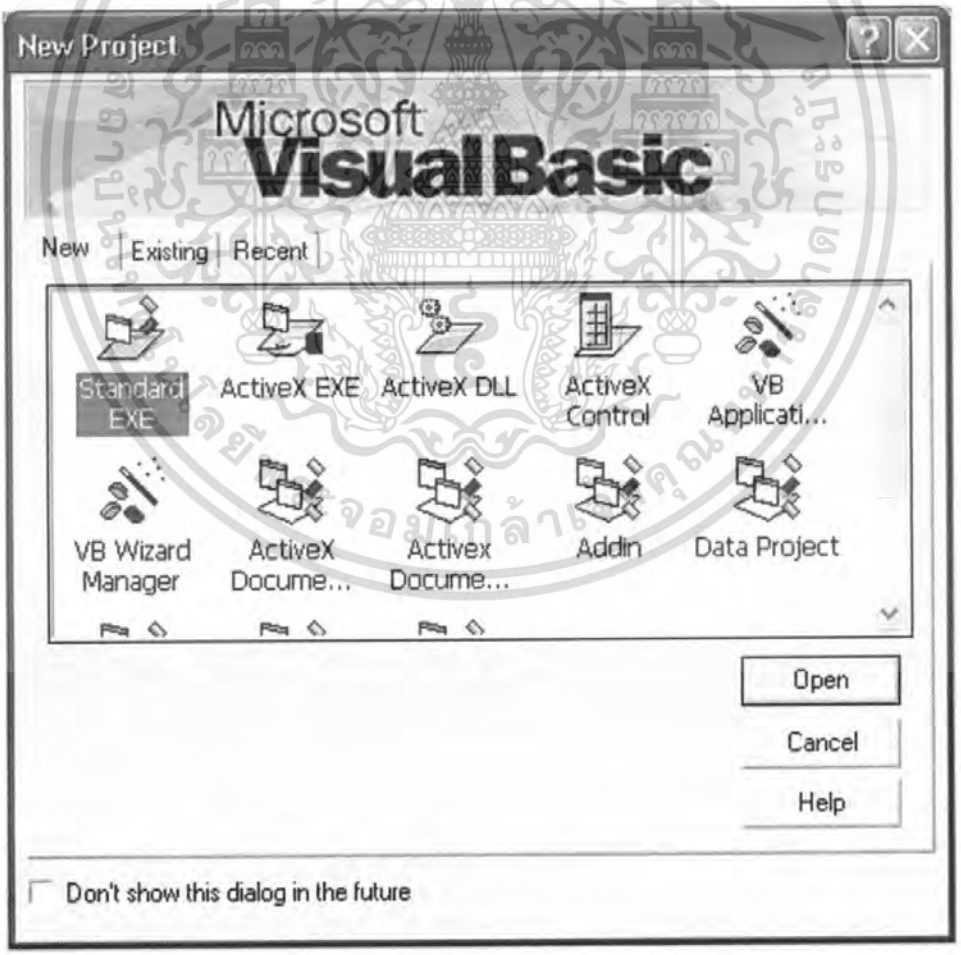
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.5 โปรแกรมที่ใช้ในการควบคุมหุ่นยนต์

ในที่นี้ทางผู้จัดทำได้เลือกใช้โปรแกรมวิชวลเบสิก ซึ่งเป็นโปรแกรมที่ใช้ติดต่อกับผู้ใช้ผ่านทางคอมพิวเตอร์ เนื่องจากเป็นเครื่องมือแอปพลิเคชั่นบนวินโดว์ ที่มีประสิทธิภาพสูงสุดตัวหนึ่ง ซึ่งสามารถรองรับความสามารถต่างๆที่อยู่ในวินโดว์ได้อย่างครบถ้วน และยังง่ายในการใช้งาน ทำให้สามารถพัฒนาแอปพลิเคชั่นต่างๆได้ในเวลาอันรวดเร็ว ไม่ว่าจะเป็นส่วนของการติดต่อกับผู้ใช้ การเชื่อมโยงฐานข้อมูล ตลอดจนการติดต่อสื่อสารผ่านทางพอร์ตอนุกรม โดยหลักการทำงานของโปรแกรมวิชวลเบสิก มีดังนี้

2.5.1 ส่วนประกอบต่างๆของโปรแกรมวิชวลเบสิก

สำหรับก่อนอื่นให้เราเปิดโปรแกรมวิชวลเบสิกขึ้นมาก่อน เมื่อเปิดแล้วจะปรากฏหน้าจอดังรูป ซึ่งโดยทั่วไปแล้ว เราจะเลือก Standard.EXE ซึ่งเป็นการใช้ วิชวลเบสิกในการโปรแกรมที่รันบนวินโดว์ทั่วไป



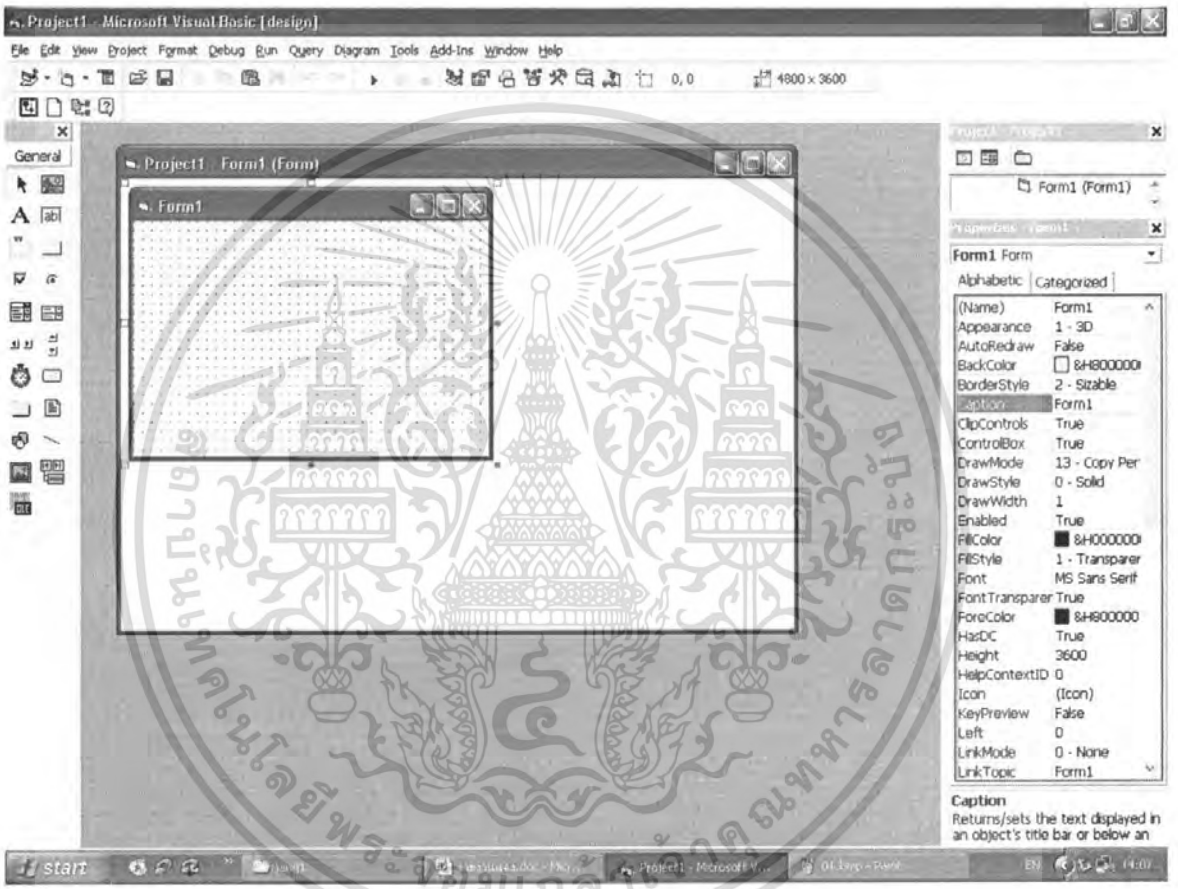
รูปที่ 2.13 หน้าต่างการเลือกชนิดโปรเจค

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เนื่องจากโปรเจคชนิด Standard.EXE นั้น เป็นการเขียนโปรแกรมทั่วไปที่รันบนวินโดว์ โดยโปรเจคคือกลุ่มของไฟล์ที่เรานำมารวมกันเพื่อสร้างโปรแกรมในวิชวลเบสิก

เมื่อเราเลือกเสร็จแล้ว ต่อไปจะปรากฏหน้าจอของวิชวลเบสิกโดยมีส่วนประกอบต่างๆที่เราจำเป็นต้องทราบ ดังนี้

1. หน้าต่างหลัก

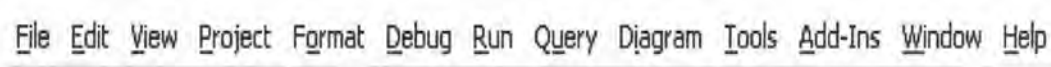


รูปที่ 2.14 หน้าต่างหลักของโปรแกรมวิชวลเบสิก

หน้าต่างหลักเป็นหน้าต่างแรกที่ปรากฏเมื่อเรียกใช้โปรแกรมวิชวลเบสิกซึ่งประกอบด้วยส่วนต่างๆ คือ เมนูบาร์(Menubar), ทูลบาร์(Toolbar), ทูลบอกซ์(Toolbox), โปรเจคเอกซ์โพลเรอร์(Project Explorer), พร็อพเพอร์ตี้วินโดว์(Properties Window), ฟอร์ม(Form) โดยมีรายละเอียด ดังนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1.1 เมนูบาร์



รูปที่ 2.15 เมนูบาร์

เมนูบาร์เป็นคำสั่งที่เราสามารถใช้งานได้ทั้งหมดในวิชวลเบสิกประกอบด้วยเมนูทำงานกับไฟล์(File), วิว(View) และ วินโดว์(Windows)

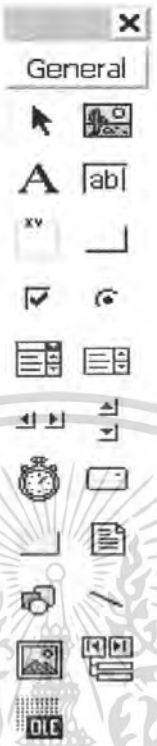
1.2 ทูลบาร์

รูปที่ 2.16 ทูลบาร์

ทูลบาร์เป็นที่แสดงเครื่องมือต่างๆที่ช่วยให้เราใช้คำสั่งของวิชวลเบสิกได้รวดเร็วยิ่งขึ้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

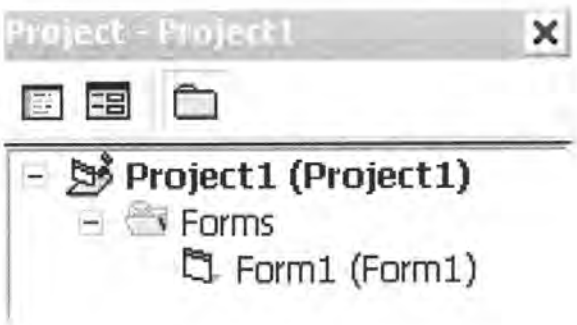
1.3 ทูลบ็อกซ์



รูปที่ 2.17 ทูลบ็อกซ์

ทูลบ็อกซ์เป็นที่แสดงเครื่องมือต่างๆที่เราเรียกว่า คอนโทรล ซึ่งเป็นเครื่องมือที่เราสามารถนำไปวางลงบนฟอร์มได้ เพื่อออกแบบหน้าจอของโปรแกรม (เรียกว่าส่วนติดต่อกับผู้ใช้ หรือ User Interface)

1.4 หน้าต่างโปรเจค



รูปที่ 2.18 หน้าต่างโปรเจค

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หน้าต่างโปรเจคเป็นหน้าต่างที่แสดงโมดูลต่างๆที่มีอยู่ในโปรเจคทั้งหมด เราสามารถกดปุ่ม <CTRL> + <R> เพื่อให้แสดงหน้าต่างนี้ได้

1.5 หน้าต่างคุณสมบัติ

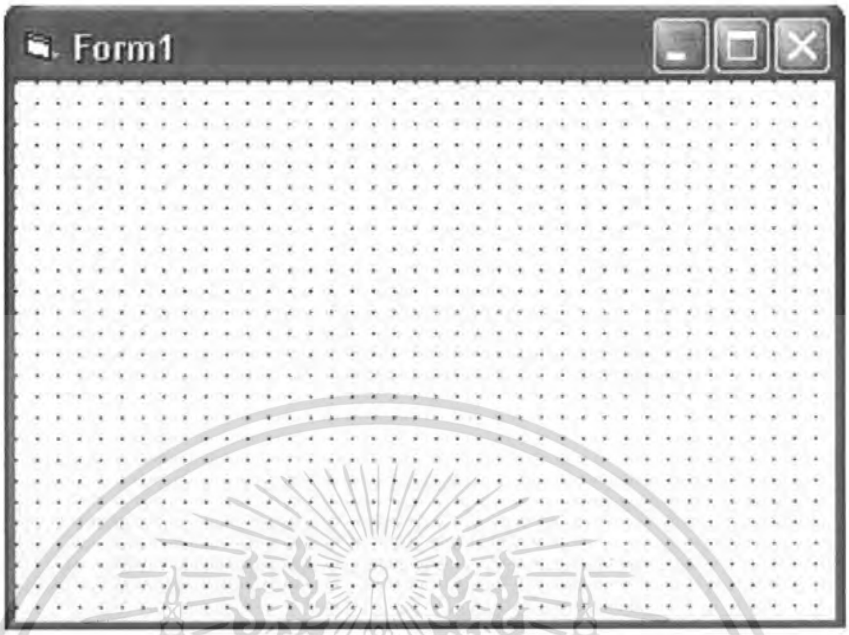
รูปที่ 2.19 หน้าต่างคุณสมบัติ

หน้าต่างคุณสมบัติเป็นหน้าต่างที่แสดงคุณสมบัติของคอนโทรลที่เลือกอยู่ในขนาดนั้น เราสามารถกดปุ่ม <F4> เพื่อให้แสดงหน้าต่างนี้ได้

คุณสมบัติ (Properties) คือลักษณะต่างๆของคอนโทรลที่ถูกนำมาวางบนฟอร์มที่เราสามารถกำหนดได้ เช่น ข้อความที่ปรากฏบนคอนโทรล รูปแบบฟอร์มของคอนโทรล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

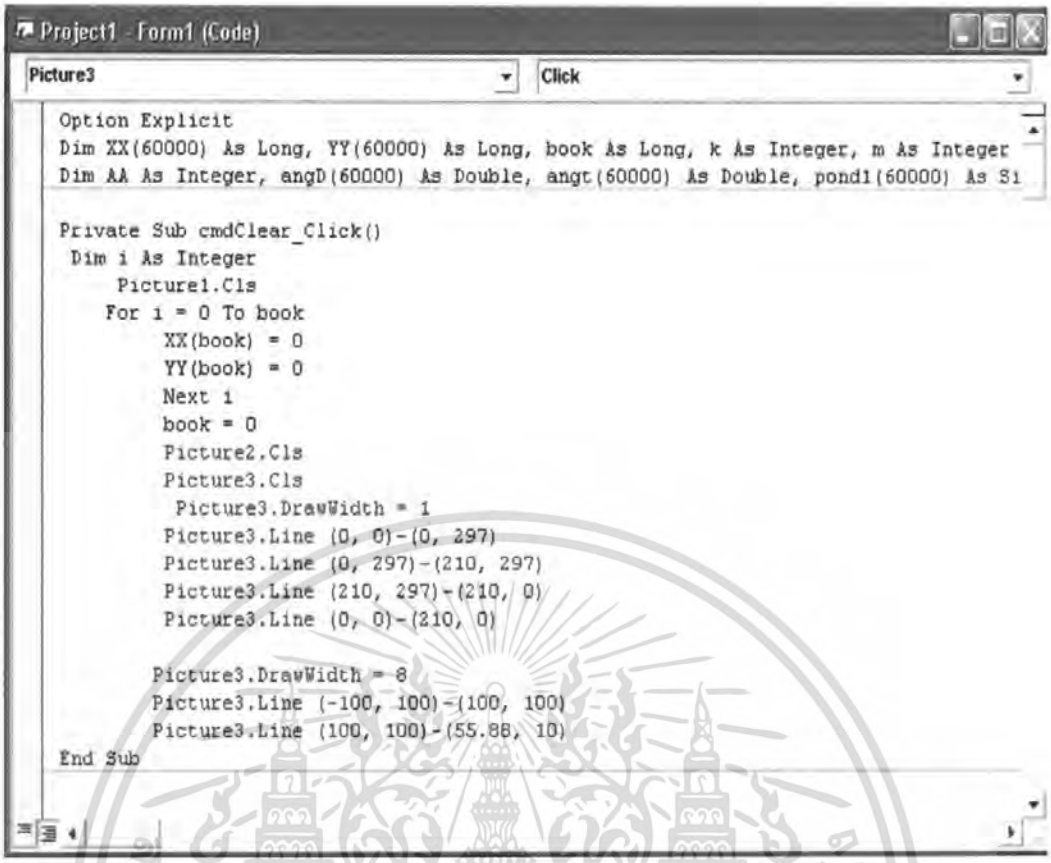
1.6 ฟอร์ม



รูปที่ 2.20 หน้าต่างฟอร์ม

ฟอร์มเป็นฟอร์มที่ใช้ในการออกแบบ เป็นหน้าต่างที่ใช้ในการออกแบบหน้าจอโปรแกรม สำหรับส่วนประกอบที่มีความสำคัญอีกส่วนคือ หน้าต่างโค๊ดโปรแกรม(Code Editor) ซึ่งเป็นหน้าต่างที่ให้เราใส่คำสั่ง เพื่อควบคุมการทำงานของโปรแกรม เราสามารถเรียกหน้าต่างโค๊ดโปรแกรม (Code Editor) ขึ้นมาโดยคลิกไอคอนเพื่อดูโค๊ดโปรแกรม (View Code) ที่หน้าต่างโปรเจคหรือดับเบิ้ลคลิกที่ฟอร์มเลยก็ได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.21 หน้าต่างโค๊ดโปรแกรม

1.7 คุณสมบัติ เมดถอดและอีเว็นต์

ในการทำงานกับฟอร์มและคอนโทรล เราจะต้องทำความคุ้นเคยกับคำ 3 คำนี้ คือ

- คุณสมบัติ (Properties) ที่ให้เรากำหนดลักษณะต่างๆของฟอร์มและคอนโทรล
- เมดถอด (Method) เป็นการสั่งฟอร์มและคอนโทรลทำงานตามที่เราร้องขอไป
- อีเว็นต์ (Event) เป็นเหตุการณ์ที่เกิดขึ้นกับฟอร์มหรือคอนโทรลที่เราสามารถใส่คำสั่งเพื่อตอบสนองได้

ในที่นี้จะกล่าวถึงคุณสมบัติ เมดถอดและอีเว็นต์ ที่โครงงานนี้ใช้งานมากเท่านั้น ดังนี้

1. ฟอร์ม (Forms)

ฟอร์มจะเป็นเครื่องมือที่เราจะต้องทำงานด้วยบ่อยมาก ในการสร้างโปรแกรมด้วยวิชวลเบสิกโดยที่ฟอร์มจะเป็นหน้าต่างที่ผู้ใช้ติดต่อทำงานด้วย ผ่านทางคอนโทรลต่างๆที่เราวางบนฟอร์ม

- คุณสมบัติที่สำคัญของฟอร์ม

Name	กำหนดชื่อของฟอร์มที่เราใช้อ้างอิงถึงในโปรแกรม
BorderStyle	กำหนดลักษณะในการเปลี่ยนขนาดของฟอร์มเป็นอย่างไร

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

MinButton, MaxButton	กำหนดให้ฟอร์มสามารถขยายให้มีขนาดใหญ่และลดขนาดเป็นขนาดเล็กได้หรือไม่
Caption	กำหนดข้อความที่แสดงบนไตเติ้ลบาร์ของฟอร์ม
Icon	กำหนดรูปไอคอนของฟอร์มเมื่อฟอร์มมีขนาดเล็ก
Height, Width	กำหนดความสูงและความกว้างของฟอร์ม
Left, Top	กำหนดตำแหน่งของฟอร์มโดยคิดจากตำแหน่งบนซ้ายของหน้าจอ
Movable	กำหนดว่าฟอร์มสามารถเคลื่อนย้ายได้หรือไม่
WindowState	กำหนดว่าเมื่อเริ่มต้นเรียกฟอร์มจะให้อยู่ในแบบขนาดใหญ่, ขนาดเล็ก หรือธรรมดา
Enabled	กำหนดให้ฟอร์มสามารถตอบสนองอีเว็นต์ได้หรือไม่

- เมดถอดและอีเว็นต์ที่สำคัญของฟอร์ม

Load	เป็นอีเว็นต์ที่เกิดเมื่อเรียกฟอร์มขึ้นมาครั้งแรก
Click	เป็นอีเว็นต์ที่เกิดเมื่อมีการกดเมาส์ (Click mouse) บนฟอร์ม
Hide	เป็นเมดถอดที่ใช้ซ่อนฟอร์มไว้

2. ปุ่มคำสั่ง (Command Button)

ปุ่มคำสั่งนั้นมีหน้าที่หลักคือ การตอบสนองผู้ใช้ที่สั่งงานมายังโปรแกรมว่าต้องการให้โปรแกรมทำอะไรตอบกลับไป

- คุณสมบัติของปุ่มคำสั่ง

Style	กำหนดให้เป็นปุ่มคำสั่งธรรมดาหรือมีรูปภาพบนปุ่มด้วย
Picture	กำหนดรูปภาพที่แสดงบนปุ่มคำสั่ง (ใช้ได้เมื่อคุณสมบัติ Style เท่ากับ 1)
DownPicture	กำหนดรูปภาพที่แสดงบนปุ่มคำสั่งเมื่อปุ่มถูกกด
DisabledPicture	กำหนดรูปภาพที่แสดงบนปุ่มคำสั่ง เมื่อคุณสมบัติการใช้งาน (Enabled) เท่ากับ False
ToolTipText	เป็นข้อความที่แสดง เมื่อหยุดตัวชี้เมาส์ที่ปุ่มคำสั่งช่วงเวลาหนึ่ง
Caption	ข้อความที่ปรากฏบนปุ่มคำสั่ง
Enabled	กำหนดให้ปุ่มคำสั่งใช้งานได้หรือไม่ได้
Visible	กำหนดให้คอนโทรลสามารถมองเห็นได้ในตอนรันหรือไม่

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 3
การออกแบบโครงสร้างของแขนกล

3.1 โครงสร้างการออกแบบแขนกล

แขนกลในโครงงานนี้เป็นแขนกลที่มีลักษณะเป็นข้อต่อหรือแบบอาร์ติคูเลต ซึ่งฐานของแขนกลสามารถเคลื่อนที่ในลักษณะหมุนและส่วนของปลายเป็นปากกาเพื่อใช้เขียน การทำงานของแขนกลในส่วนของจุดหมุนทั้งหมดเราใช้เซอร์โวมอเตอร์ (Servo Motor)ในการขับเคลื่อน สำหรับอุปกรณ์ที่ใช้ในการประกอบแขนกลนี้คือ อะคริลิก แผ่นอีพร็อกซี โฟม ไฟเบอร์กลาส ไม้บาร์ซ่าร์ และอุปกรณ์ต่างๆในแต่ละตำแหน่งตามความเหมาะสมและวัสดุที่หาได้ ในการออกแบบโครงสร้างนั้นแบ่งออกเป็น 3 ส่วน คือ

1. ส่วนฐาน
2. ส่วนแขน
3. ส่วนปลายจับปากกา

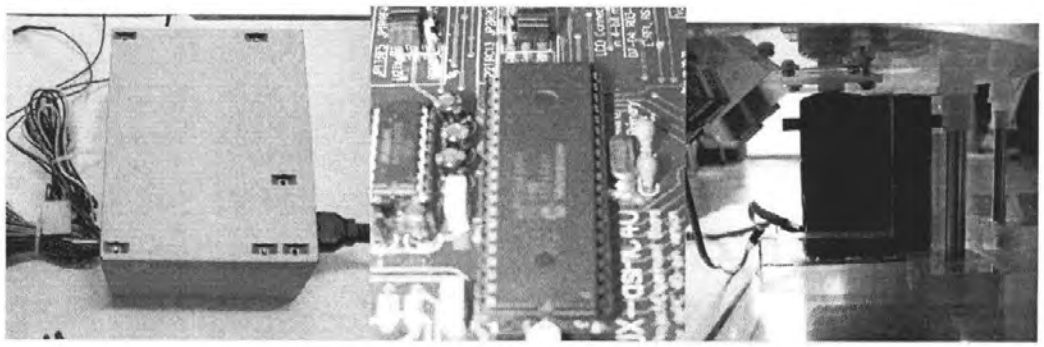
3.1.1 ส่วนฐาน

ส่วนฐานแบ่งออกเป็น 2 ส่วนคือ ฐานเครื่องและฐานหมุน

1. ฐานเครื่อง เป็นส่วนที่ติดตั้งมอเตอร์และชุดขับเคลื่อนแขนกล โดยฐานเครื่องทำมาจากอะคริลิกหนา 1 เซนติเมตร

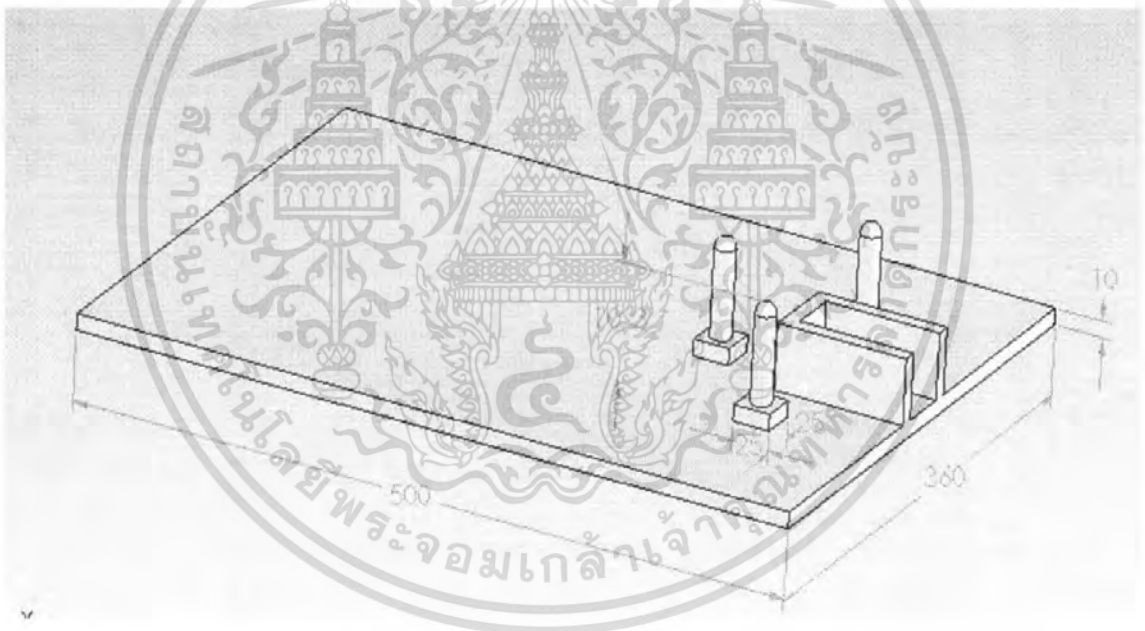
- พาวเวอร์ซัพพลาย (Power Supply) เป็นพาวเวอร์ซัพพลายของบริษัทดีแทค (DTECH) รุ่น Power P4 450W 24M เพื่อแปลงไฟฟ้าจาก 220 VAC ให้เป็น 5 VDC
- วงจรไมโครคอนโทรลเลอร์ที่มีไมโครคอนโทรลเลอร์ dsPIC เบอร์ dsPIC30F4011
- เซอร์โวมอเตอร์เป็นมอเตอร์ที่ใช้หมุนฐานหมุน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.1 ส่วนประกอบต่างๆของฐานเครื่อง

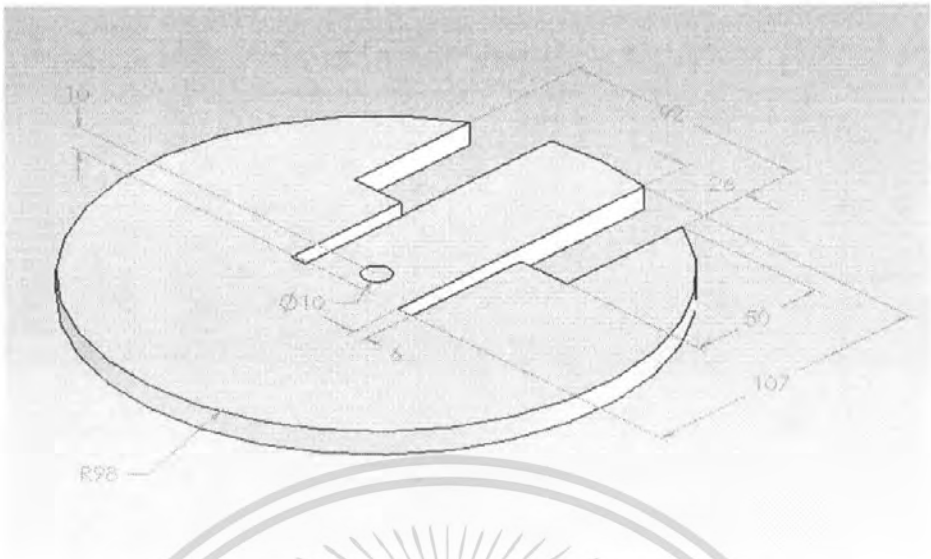
อุปกรณ์ต่างๆที่กล่าวมาแล้วนั้นได้ทำการติดตั้งไว้ที่ฐานซึ่งฐานมีขนาดที่ได้ออกแบบไว้ดังนี้



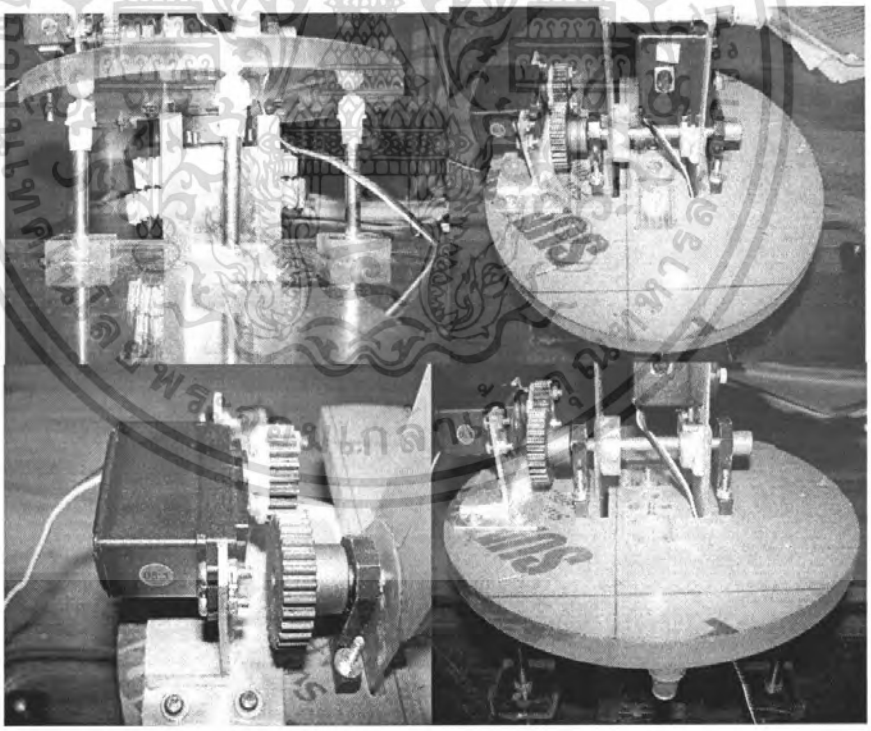
รูปที่ 3.2 ฐานเครื่องที่ออกแบบ

2. ฐานหมุน เป็นส่วนติดตั้งชุดแขนทั้งหมด เมื่อฐานส่วนนี้หมุนจะเป็นการหมุนชุดแขนทั้งชุดโดยฐานหมุนติดตั้งอยู่บนเซอร์โวมอเตอร์ของฐานเครื่องและใช้เป็นที่ติดตั้งของเซอร์โวมอเตอร์ตัวอื่นๆ ซึ่งฐานหมุนมีขนาดดังนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.3 ฐานหมุนที่ออกแบบ

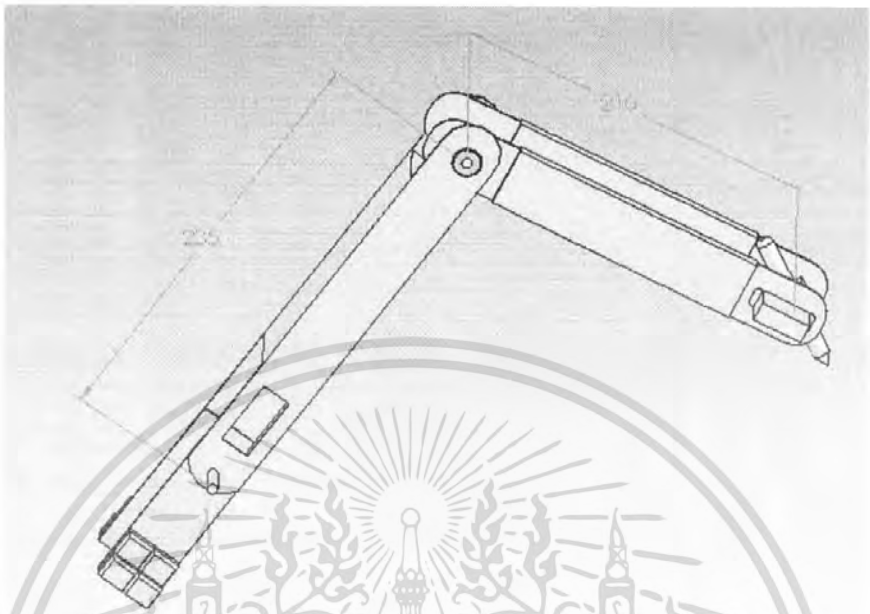


รูปที่ 3.4 ลักษณะโครงสร้างและการติดตั้งของฐานหมุน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.1.2 ส่วนแขน

แบ่งออกเป็น 2 ส่วนคือ แขนท่อนล่างและแขนท่อนบน

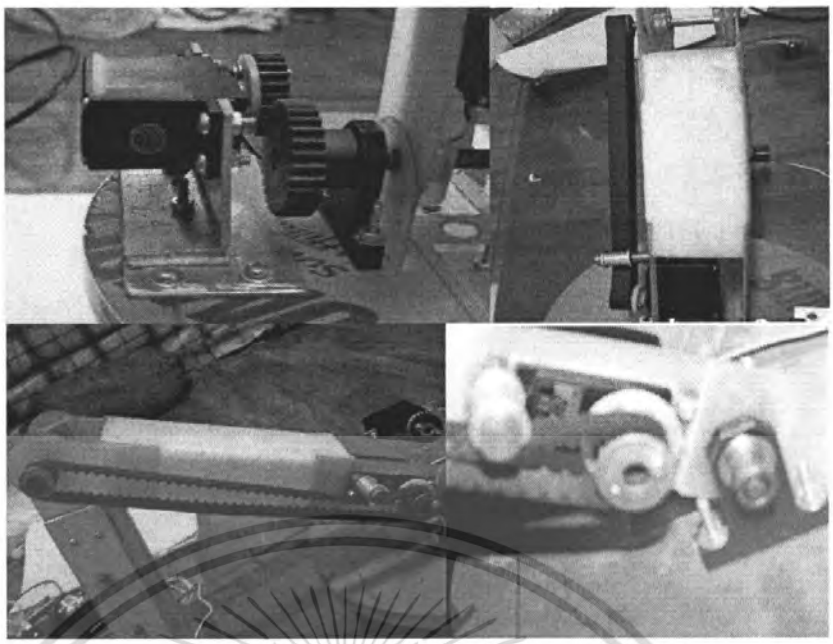


รูปที่ 3.5 แขนท่อนล่างและแขนท่อนบนที่ออกแบบ

1. แขนท่อนล่าง ขับกำลังจากเซอร์โวมอเตอร์ที่ติดกับฐานหมุน เมื่อหมุนไปตามตำแหน่งที่ต้องการ โดยปลายอีกด้านหนึ่งจะเป็นจุดเชื่อมโยงกับแขนท่อนบน แขนท่อนล่างนี้จะใช้แผ่นอีพร็อกซี โฟมและใยไฟเบอร์ ใช้ในการหล่อแขนเป็นโครงสร้างในการประกอบแขน ดังรูป

ในการติดตั้งแขนท่อนล่างมี 2 ด้าน ด้านหนึ่งติดกับเซอร์โวมอเตอร์โดยตรง และส่วนในของแขนได้มีการเจาะรูติดเซอร์โวมอเตอร์อีกตัวเพื่อใช้ในการขับเคลื่อนแขนท่อนบน อีกด้านของแขนใช้ แบริ่งเป็นจุดหมุน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.6 โครงสร้างของแขนท่อนล่าง

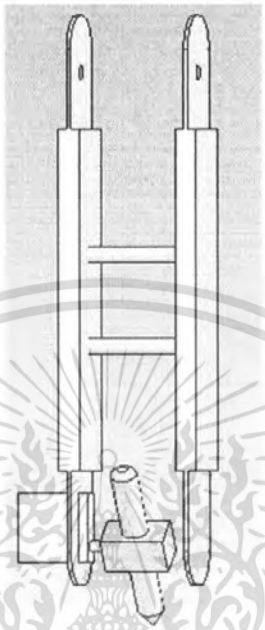
2. แขนท่อนบน ขับเคลื่อนจากเซอร์โวมอเตอร์ เพื่อหมุนแขนไปยังตำแหน่งที่ต้องการได้ ตำแหน่งที่ติดเซอร์โวมอเตอร์ที่ส่งกำลังคือบนแขนท่อนล่าง ดังนั้นจึงส่งกำลังผ่านสายพานมาหมุนได้ แขนท่อนนี้ที่ปลายได้มีการติดเซอร์โวมอเตอร์อีกตัวไว้ซึ่งมีขนาดเล็กเพื่อใช้ในการหมุนปากกา

รูปที่ 3.7 โครงสร้างของแขนท่อนบน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.1.3 ส่วนที่ใช้จับปากกา

ส่วนนี้ใช้ในการจับปากกาเพื่อนำไปเขียนตามจุดต่างๆ โดยเราออกแบบง่ายๆใช้อะคริลิกเจาะรูเป็นที่ใส่ปากกาและติดกับเซอร์โวมอเตอร์

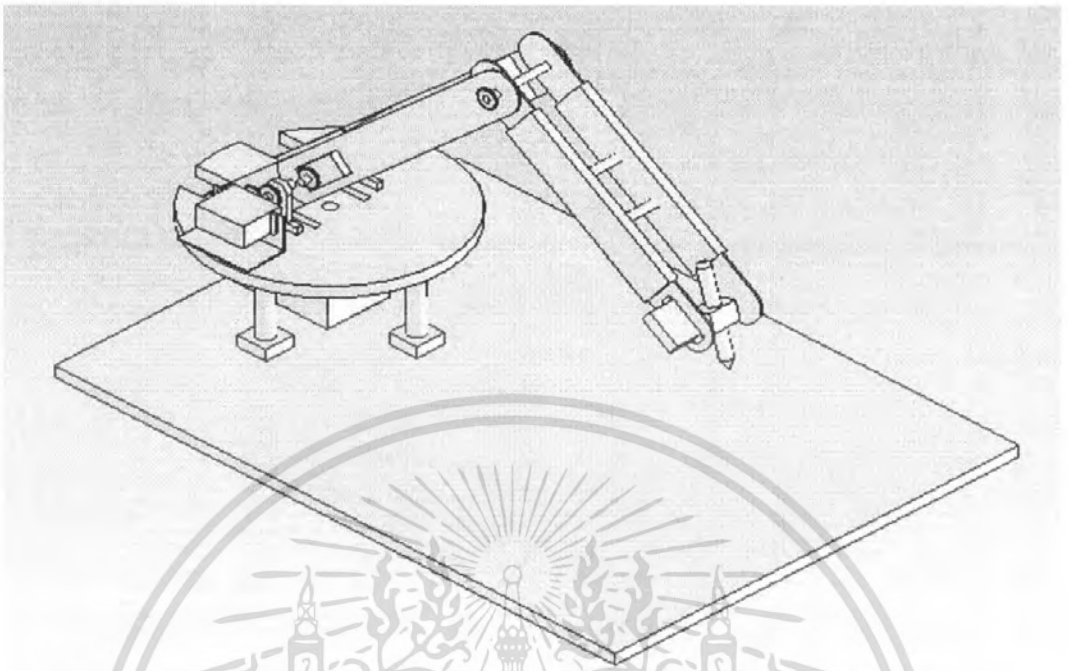


รูปที่ 3.8 ส่วนปากกาที่ออกแบบ

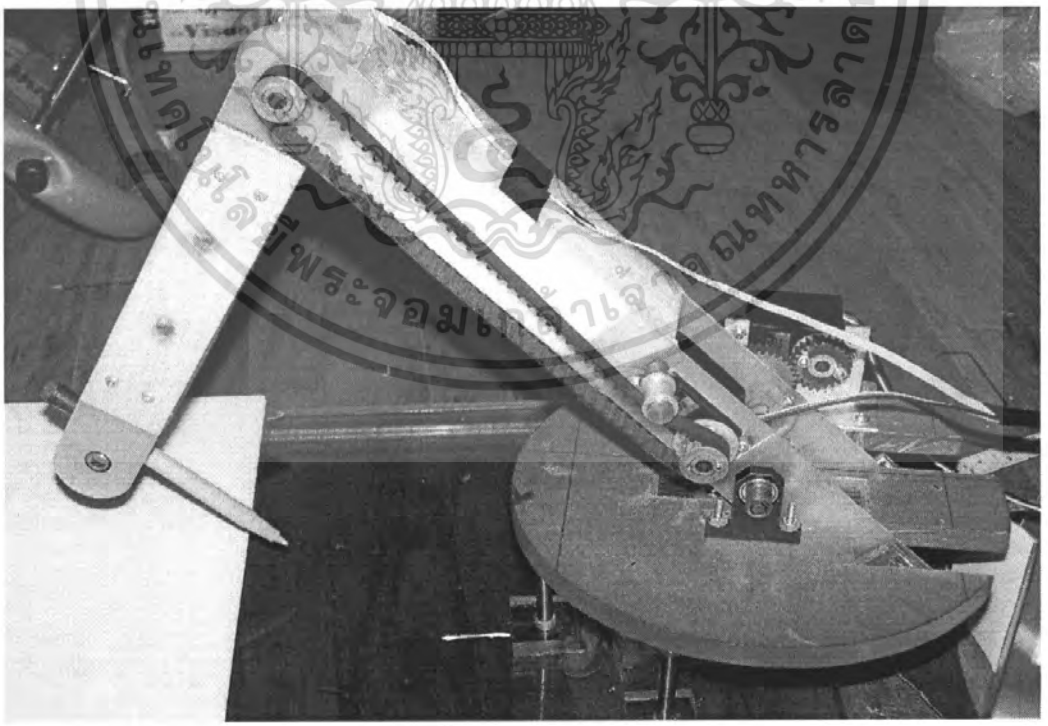
รูปที่ 3.9 โครงสร้างของแขนส่วนปลายปากกา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เมื่อนำชิ้นส่วนทั้งหมดมาประกอบเข้าด้วยกันจะได้โครงสร้างของแขนกลดังต่อไปนี้



รูปที่ 3.10 ภาพโครงสร้างของแขนกล

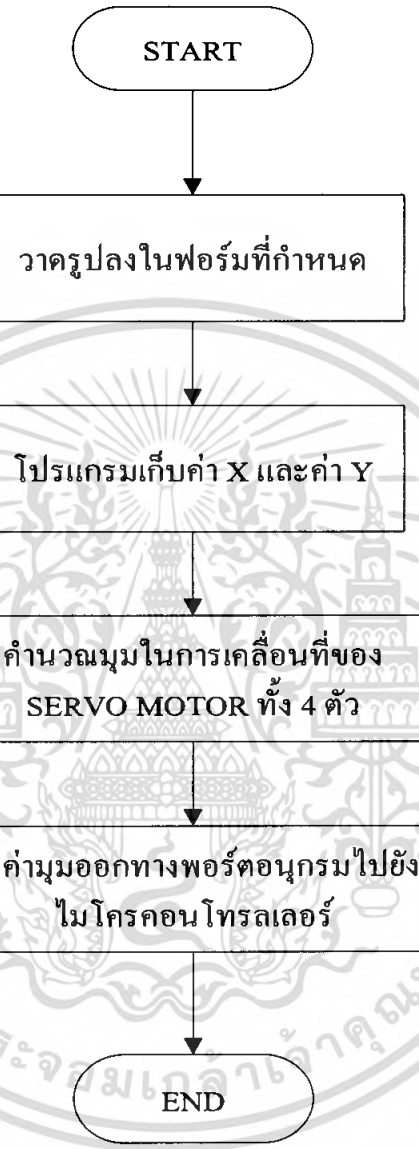


รูปที่ 3.11 โครงสร้างของแขนกล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.2 การออกแบบโปรแกรมที่ใช้ในการควบคุม

โปรแกรมที่ใช้งานในส่วนของการวาดรูปในคอมพิวเตอร์คือ โปรแกรมวิชวลเบสิกโดย ขั้นตอนในการทำงานของโปรแกรมสามารถอธิบายได้ดังนี้

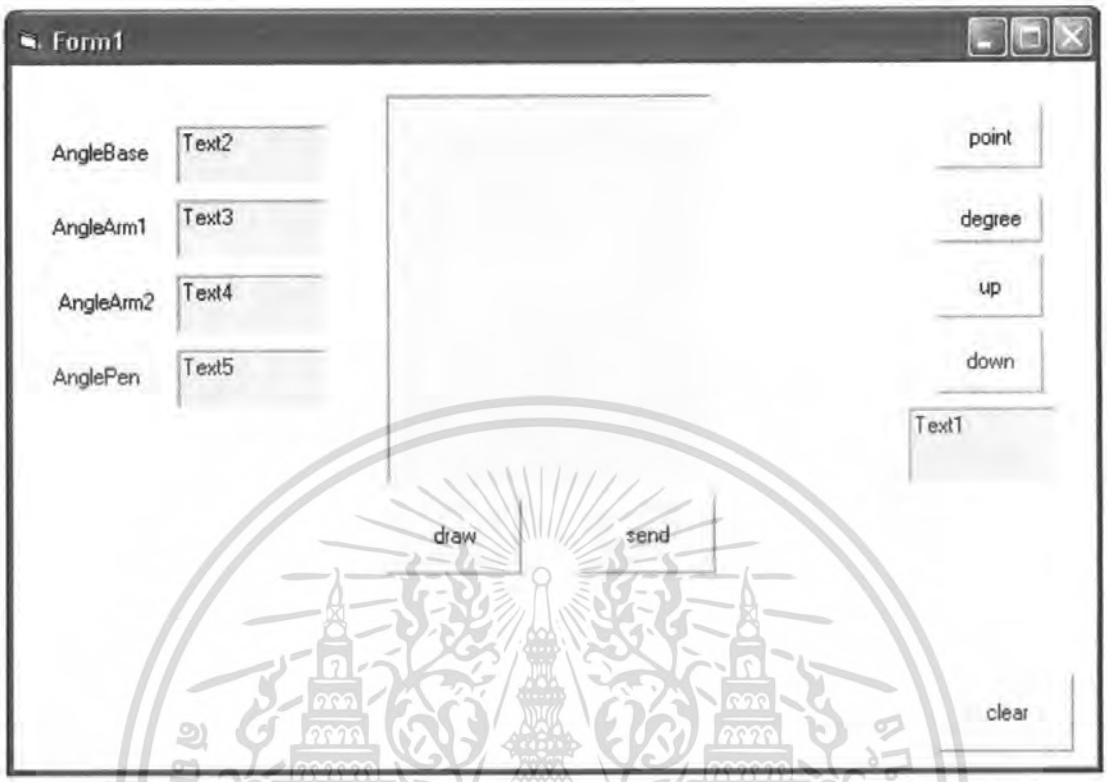


รูปที่ 3.12 โฟลว์ชาร์ตแสดงขั้นตอนการทำงานของโปรแกรมวิชวลเบสิก

การทำงานของโปรแกรมวิชวลเบสิกเริ่มต้นจากผู้ใช้งานกดปุ่มวาดรูป(Draw) จากนั้นผู้ใช้จะวาดรูปลงไปในฟอร์มที่กำหนดไว้ และระหว่างนี้ทางโปรแกรมได้ทำการเก็บค่าจุด (X,Y) ทุกจุดไว้ เพื่อนำไปคำนวณหาค่ามุมในการเคลื่อนที่ของเซอร์โวมอเตอร์ (Servo Motor) ทั้ง 4 ตัว หลังจากนั้นเมื่อกดปุ่มส่ง(Send) ทางโปรแกรมจะจัดส่งข้อมูลมุมทั้งหมดผ่านทางพอร์ตอนุกรมไปยังไมโครคอนโทรลเลอร์เพื่อให้ประมวลผลต่อไป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ส่วนที่ได้ออกแบบไว้เพื่อใช้ติดต่อกับผู้ใช้งานเป็นดังต่อไปนี้



รูปที่ 3.13 หน้าต่างการใช้งานของโปรแกรมวิชวลเบสิก

การใช้งานสามารถอธิบายได้ดังนี้

- ปุ่ม draw กดเพื่อเป็นการบอกว่าสามารถวาดรูปลงในฟอร์มที่กำหนดได้
- ปุ่ม send กดเพื่อส่งค่ามุมการเคลื่อนที่ของเซอร์โวมอเตอร์แต่ละตัวออกไปยังไมโครคอนโทรลเลอร์โดยผ่านทางพอร์ตอนุกรม
- ปุ่ม point กดเพื่อแสดงค่าจุดทุกจุดที่เก็บค่าไว้จากการวาดรูป
- ปุ่ม degree กดเพื่อแสดงค่ามุมของเซอร์โวมอเตอร์ทั้ง 4 ตัว โดยมุมที่แสดงมีหน่วยเป็นองศา
- ปุ่ม up/down กดเพื่อทดสอบหาข้อมูลพื้นฐานของเซอร์โวมอเตอร์
- ปุ่ม clear กดเพื่อต้องการยกเลิกการทำงานทั้งหมด เคลียร์ค่าทุกค่าให้หมดไป

ทางด้านซ้ายมือของฟอร์มสำหรับวาดรูป มีการแสดงค่ามุมที่ใช้ในการเคลื่อนที่ของเซอร์โวมอเตอร์แต่ละตัวตามจุดนั้นๆที่มีการเคลื่อนที่ไป โดย

- AngleBase คือมุมที่ใช้ในการเคลื่อนที่ของเซอร์โวมอเตอร์ควบคุมส่วนของฐาน
- AngleArm1 คือมุมที่ใช้ในการเคลื่อนที่ของเซอร์โวมอเตอร์ที่ควบคุมส่วนของแขนท่อนล่าง

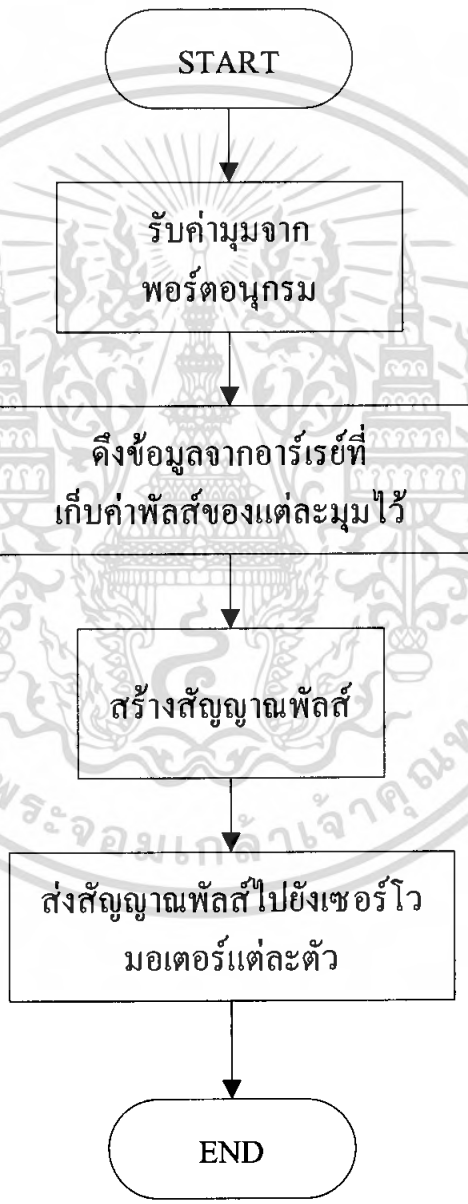
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- AngleArm2 คือมุมที่ใช้ในการเคลื่อนที่ของเซอร์โวมอเตอร์ที่ควบคุมส่วนของแขนท่อนบน

- AnglePen คือมุมที่ใช้ในการเคลื่อนที่ของเซอร์โวมอเตอร์ที่ควบคุมส่วนของปากกา

3.3 โปรแกรมที่ใช้ในการประมวลผลในไมโครคอนโทรลเลอร์

โปรแกรมที่ใช้ในการประมวลผลในไมโครคอนโทรลเลอร์นี้ใช้โปรแกรมMPLAB IDE และโปรแกรม MPLAB C30 ซึ่งสามารถอธิบายการทำงานได้ดังต่อไปนี้



รูปที่ 3.14 โฟลว์ชาร์ตแสดงการทำงานของโปรแกรมประมวลผลของไมโครคอนโทรลเลอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การทำงานของโปรแกรมการประมวลผลในไมโครคอนโทรลเลอร์เริ่มต้นจากรับค่ามุมที่ส่งมาจากโปรแกรมวิชวลเบสิกผ่านทางพอร์ตอนุกรม จากนั้นมีการดึงข้อมูลค่าสัญญาณพัลส์ของมุมนั้นๆที่ถูกส่งมาซึ่งได้มีการจัดเก็บไว้ในตัวแปรชนิดอาร์เรย์ แล้วนำไปสร้างสัญญาณพัลส์ตามข้อมูลที่เก็บค่าไว้ หลังจากนั้นส่งสัญญาณพัลส์ออกทางขาสัญญาณออก(Output) เพื่อส่งต่อไปยังเซอร์โวมอเตอร์และขับเคลื่อนให้เซอร์โวมอเตอร์ทำงานต่อไป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 4
การทดลอง

4.1 การทดลองที่ 1

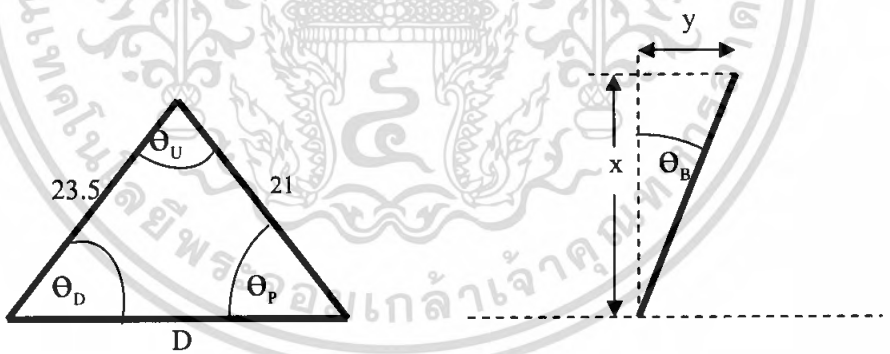
การทดลองที่ 1ในโครงงานนี้เป็นการทดลองเพื่อหาข้อมูลพื้นฐานการเคลื่อนที่ของเซอร์โวมอเตอร์ทั้ง 4 ตัวคือ

-- เซอร์โวมอเตอร์ตัวที่ 1 ควบคุมการเคลื่อนที่ของฐาน

- เซอร์โวมอเตอร์ตัวที่ 2 ควบคุมการเคลื่อนที่ของแขนท่อนล่าง(ท่อนที่1)
- เซอร์โวมอเตอร์ตัวที่ 3 ควบคุมการเคลื่อนที่ของแขนท่อนบน(ท่อนที่2)
- เซอร์โวมอเตอร์ตัวที่ 4 ควบคุมการเคลื่อนที่ของส่วนปากกา(ส่วนปลาย)

โดยการทดลองนี้จะตั้งตำแหน่งเริ่มต้นของเซอร์โวมอเตอร์ทุกตัวไว้ที่ 90 องศา และทำการจ่ายสัญญาณพัลส์ทั้งเพิ่มขึ้นและลดลง ระหว่างนั้นสังเกตการเคลื่อนที่ของเซอร์โวมอเตอร์แล้วบันทึกผลค่ามุมและสัญญาณพัลส์ที่จ่ายให้กับมุมนั้นๆ หลังจากการทดลองเราจะนำมาเลือกว่าค่ามุมช่วงไหนที่เหมาะสมต่อการใช้งานเพื่อนำไปขับเคลื่อนมอเตอร์ให้แขนกลสามารถทำงานได้จริง

และการหามุมในการเคลื่อนที่ของเซอร์โวมอเตอร์ ณ พิกัดนั้นๆ สามารถคำนวณได้ดังต่อไปนี้



รูปที่ 4.1 มุมในการคำนวณการเคลื่อนที่ของแขนกล

แขนท่อนล่างยาว 23.5 เซนติเมตร และแขนท่อนบนยาว 21 เซนติเมตร

พิกัด (x , y) จะถูกเก็บค่าไว้ในตัวแปร p1 และ q1 ตามลำดับ

p1 = X

q1 = Y

p2 = p1 + 240

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

$$D = ((p2 * p2) + (q1 * q1))^{1/2}$$

โดย p1 คือ ตัวแปรเก็บพิกัดค่าของ x

q1 คือ ตัวแปรเก็บพิกัดค่าของ y

p2 คือ ตัวแปรที่เก็บค่าระยะทางจากพิกัด x จนถึงส่วนจุดหมุนของฐาน ในแนวแกนX

D คือ ระยะทางจากจุด (x , y) ใดๆจนถึงจุดหมุนของฐาน

1. การเคลื่อนที่ของเซอร์โวมอเตอร์ที่ควบคุมฐาน

$$Base = q1 / p2$$

$$\Theta_B = \arctan(Base)$$

$$\Theta_{BASE} = (\Theta_B - 90^0) + 10^0)$$

โดย Base คือ อัตราส่วนระหว่างระยะในแนวแกน y ต่อระยะในแนวแกน x

Θ_B คือ มุมที่ได้จากการคำนวณมุมที่ฐาน

Θ_{BASE} คือ มุมในการเคลื่อนที่ของเซอร์โวมอเตอร์ที่ควบคุมฐาน

มุมที่ควบคุมฐานสามารถเคลื่อนที่ไปทั้งทางซ้ายมือและขวามือได้ หรือ เซอร์โวมอเตอร์ที่ควบคุมฐานจะเคลื่อนที่ทำมุม 0 - 180 องศา

2. การเคลื่อนที่ของเซอร์โวมอเตอร์ที่ควบคุมแขนท่อนล่าง

$$(21)^2 = (23.5)^2 + D^2 - 2(23)D \cos(\Theta_D)$$

$$\Theta_D = \arccos ((D^2 + 112.25) / 47D)$$

$$\Theta_{DOWN} = 90^0 - (55^0 - \Theta_D)$$

โดย Θ_D คือ มุมที่ได้จากการคำนวณมุมของแขนท่อนล่าง

Θ_{DOWN} คือ มุมในการเคลื่อนที่ของเซอร์โวมอเตอร์ที่ควบคุมแขนท่อนล่าง

มุมในการเคลื่อนที่ของเซอร์โวมอเตอร์ที่ควบคุมแขนท่อนล่างจะเคลื่อนที่ทำมุม 0 - 90 องศา โดยจะเคลื่อนที่ลงด้านล่างเท่านั้น ไม่เคลื่อนที่ขึ้นข้างบน จากตำแหน่งเริ่มต้น

3. การเคลื่อนที่ของเซอร์โวมอเตอร์ที่ควบคุมแขนท่อนบน

$$D^2 = (21)^2 + (23.5)^2 - 2(21)(23.5) \cos(\Theta_U)$$

$$\Theta_U = \arccos((993.25 - D^2) / 987)$$

$$\Theta_{UP} = 90^0 - (\Theta_U - 77^0)$$

โดย Θ_U คือ มุมที่ได้จากการคำนวณมุมของแขนท่อนบน

Θ_{UP} คือ มุมในการเคลื่อนที่ของเซอร์โวมอเตอร์ที่ควบคุมแขนท่อนบน

มุมในการเคลื่อนที่ของเซอร์โวมอเตอร์ที่ควบคุมแขนท่อนบนจะเคลื่อนที่ทำมุม 0 - 90 องศา โดยจะเคลื่อนที่ขึ้นข้างบนเท่านั้น ไม่เคลื่อนที่ลงด้านล่าง จากตำแหน่งเริ่มต้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4. การเคลื่อนที่ของเซอร์โวมอเตอร์ที่ควบคุมส่วนปลายปากกา

$$(23.5)^2 = (21)^2 + D^2 - 2(21)D\cos(\Theta_P)$$

$$\Theta_P = \text{acrcos}((D^2 - 111.25) / 42D)$$

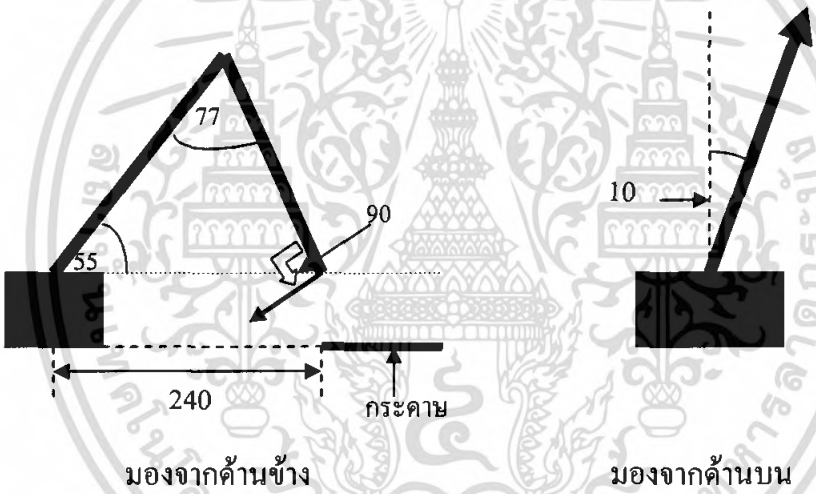
$$\Theta_{PEN} = 90 - \Theta_P$$

โดย Θ_P คือ มุมที่ได้จากการคำนวณมุมของส่วนปลายปากกา

Θ_{PEN} คือ มุมในการเคลื่อนที่ของเซอร์โวมอเตอร์ที่ควบคุมส่วนปลายปากกา

มุมในการเคลื่อนที่ของเซอร์โวมอเตอร์ที่ควบคุมปลายปากกาจะเคลื่อนที่ทำมุม 0 - 90 องศา โดยจะเคลื่อนที่ออกนอกฐานเท่านั้นและจะไม่เคลื่อนที่เข้าหาฐานจากตำแหน่งเริ่มต้น

จากการทดลองเพื่อหาข้อมูลพื้นฐานของเซอร์โวมอเตอร์แต่ละตัวนั้น ทำให้สามารถตั้งค่าตำแหน่งเริ่มต้นของแขนกลได้ ดังนี้



รูปที่ 4.2 ตำแหน่งมุมเริ่มต้นของแขนกล

เมื่อได้มุมเริ่มต้นของแขนกลและสูตรในการคำนวณมุมในการเคลื่อนที่ของเซอร์โวมอเตอร์ทั้งสี่ตัวที่พิกัดต่างๆมาแล้ว ลำดับต่อไปเราได้ทำการทดสอบว่า แขนกลสามารถเคลื่อนที่ได้จริงหรือไม่ โดยทำการทดลองที่ 2

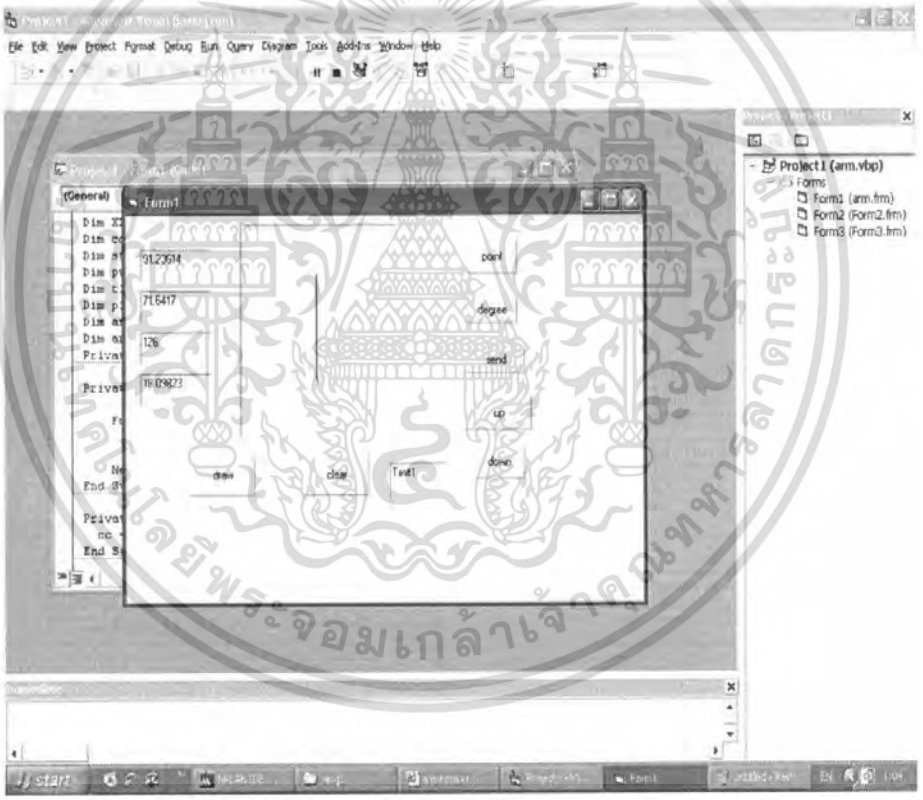
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.2 การทดลองที่ 2

การทดลองที่2 เป็นการทดลองดูว่า แขนกลสามารถใช้งานได้หรือวาดรูปได้หรือไม่ โดยการทดลองเริ่มจากการวาดรูปที่เราต้องการทดสอบลงในฟอร์มที่กำหนดในโปรแกรมวิชวลเบสิกที่จัดสร้างขึ้นมา จากนั้นก็กดปุ่มส่งค่าไปยังไมโครคอนโทรลเลอร์ที่มีโปรแกรมรองรับการทำงานแล้ว และไมโครคอนโทรลเลอร์ก็จะส่งค่าสัญญาณพัลส์ออกไปยังเซอร์โวมอเตอร์แต่ละตัว

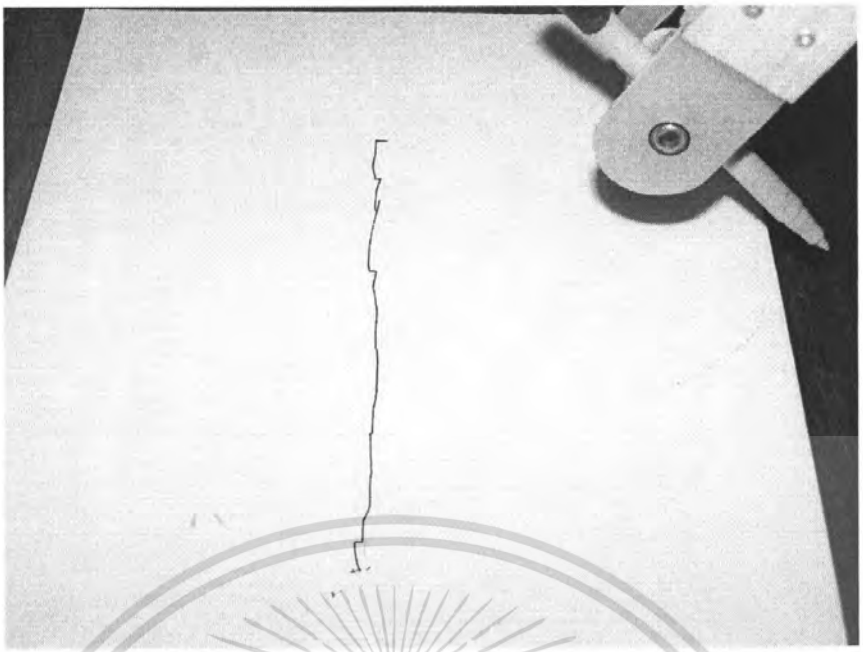
4.2.1 ผลการทดลอง

ผลการทดลองนี้จะเป็นการเปรียบเทียบกันระหว่างรูปที่เราวาดลงในฟอร์มกับรูปที่แขนกลทำการวาด ดังนี้

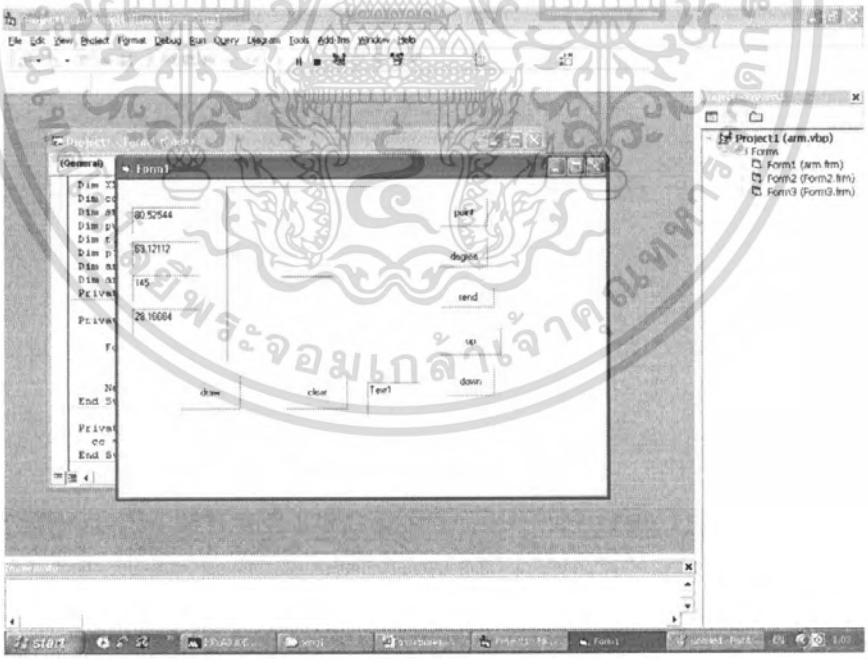


รูปที่ 4.3 รูปที่วาดลงในฟอร์มครั้งที่ 1

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

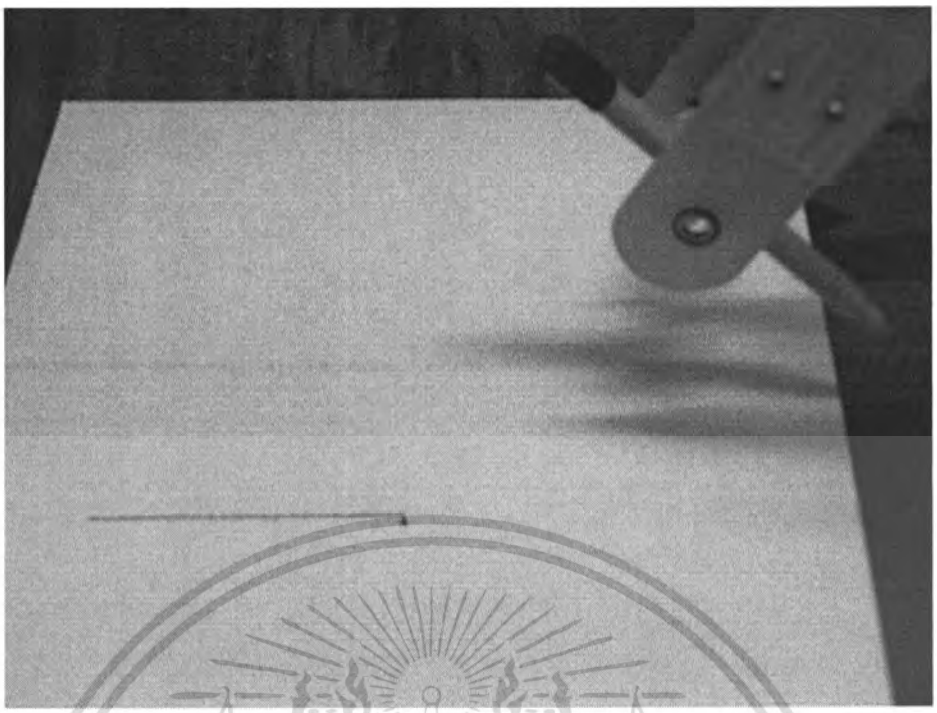


รูปที่ 4.4 การทดลองวาดรูปครั้งที่ 1

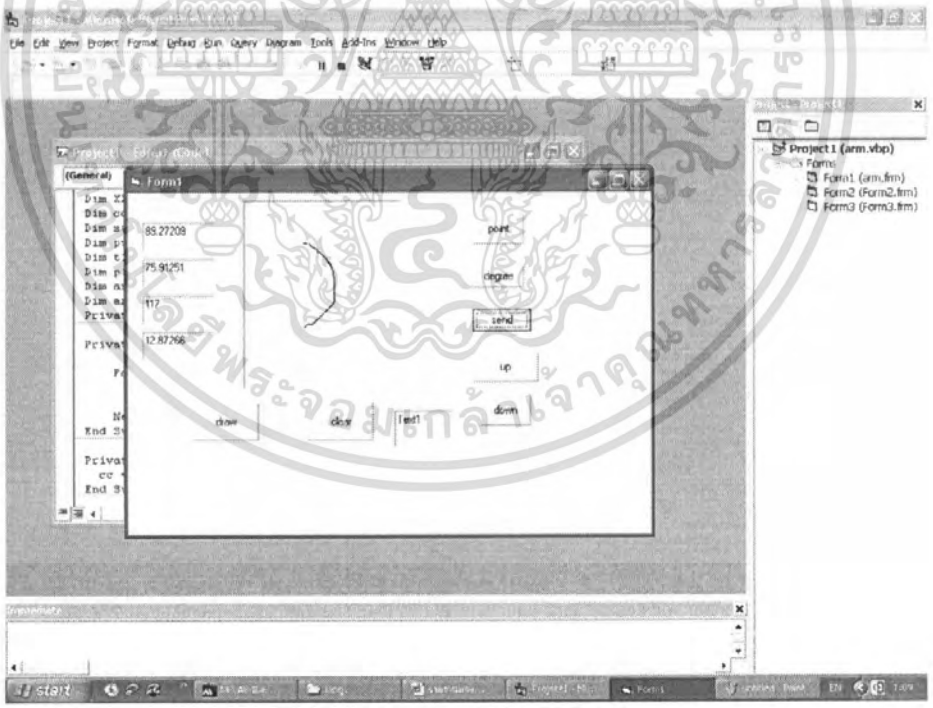


รูปที่ 4.5 รูปที่วาดลงในฟอร์มครั้งที่ 2

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

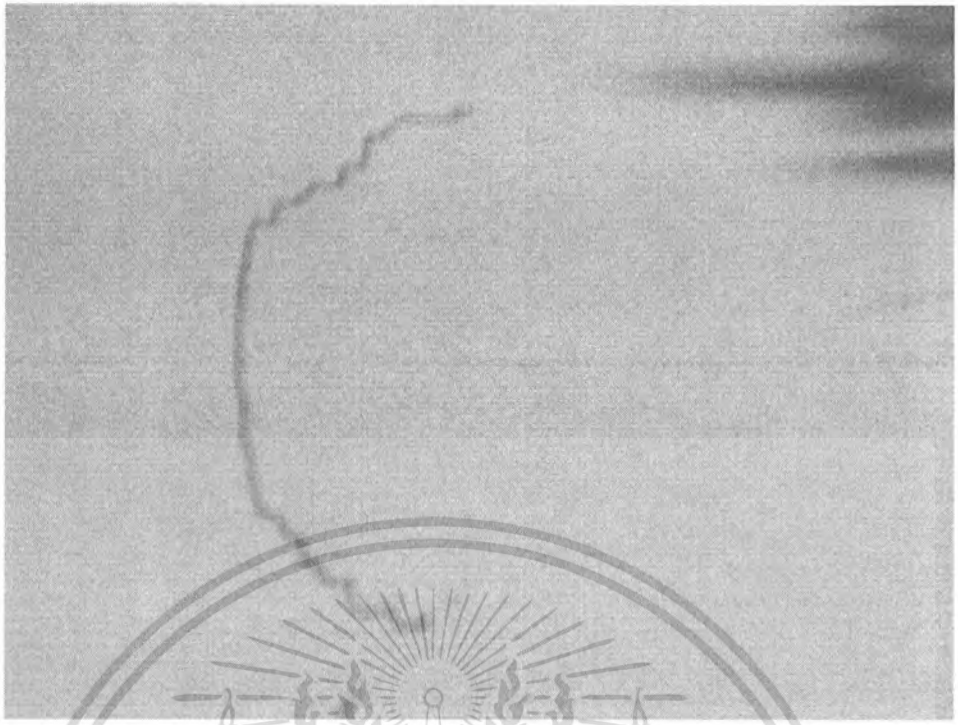


รูปที่ 4.6 การทดลองวาดรูปครั้งที่ 2

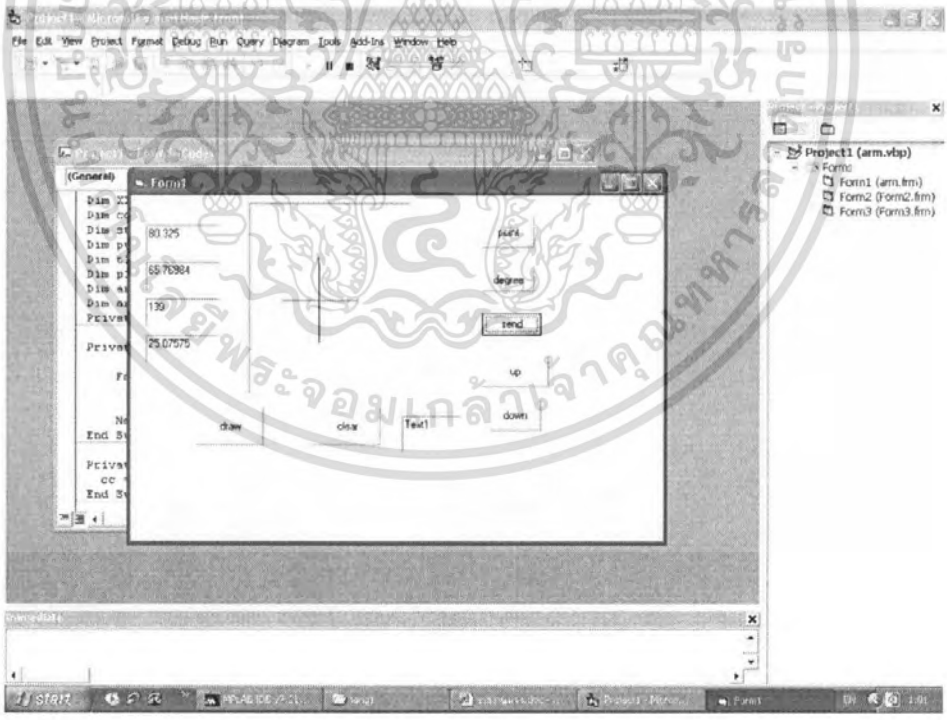


รูปที่ 4.7 รูปที่วาดลงในฟอร์มครั้งที่ 3

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

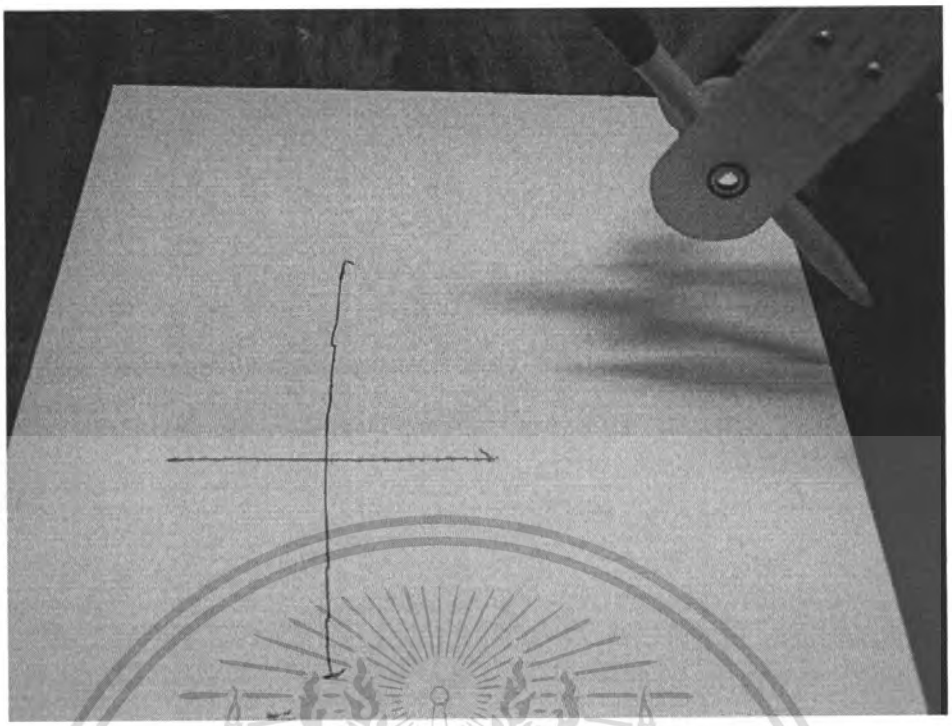


รูปที่ 4.8 การทดลองวาดรูปครั้งที่ 3

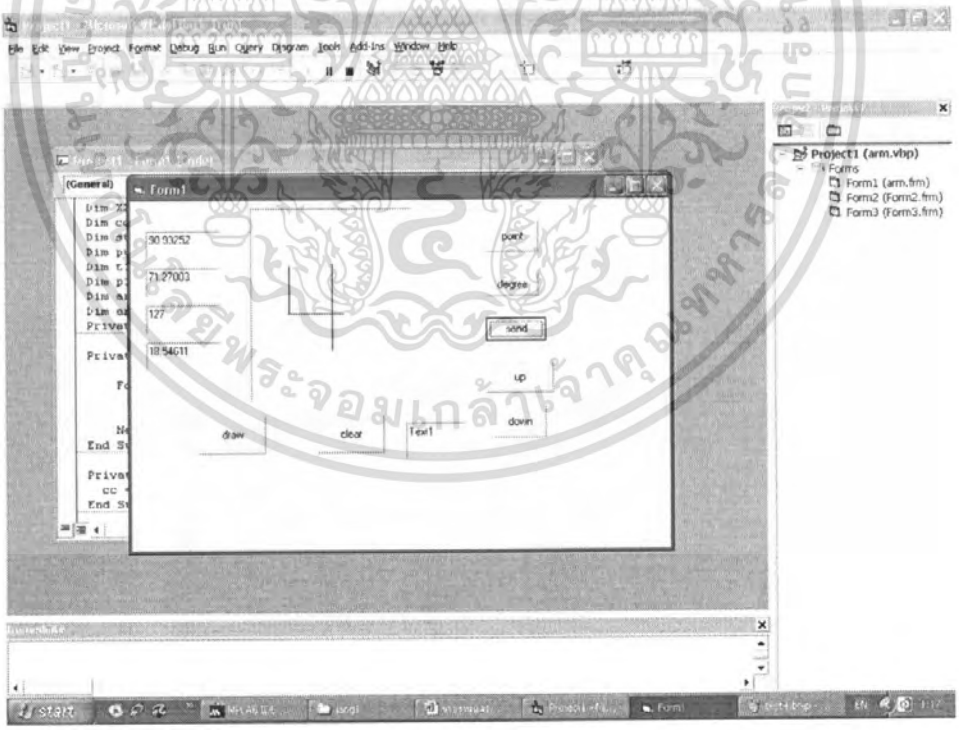


รูปที่ 4.9 รูปที่วาดลงในฟอร์มครั้งที่ 4

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

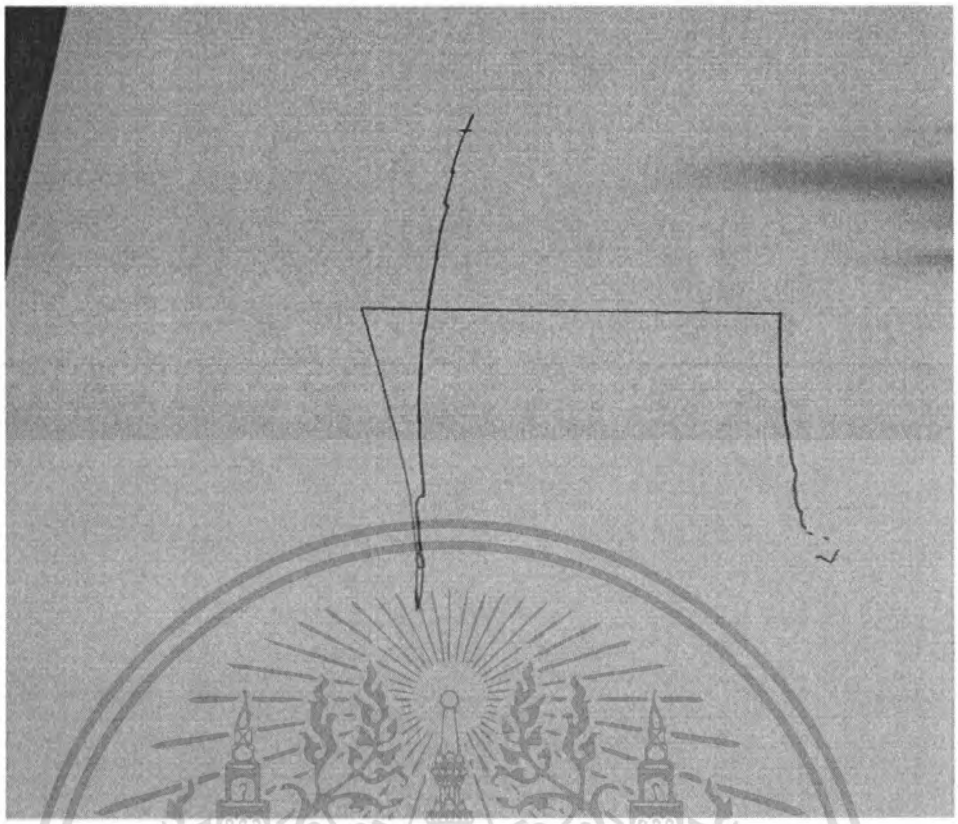


รูปที่ 4.10 การทดลองวาดรูปครั้งที่ 4



รูปที่ 4.11 รูปที่วาดลงในฟอร์มครั้งที่ 5

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.12 การทดลองวาดรูปครั้งที่ 5 ตอนมีข้อผิดพลาด

รูปที่ 4.13 การทดลองวาดรูปครั้งที่ 5 ตอนไม่มีข้อผิดพลาด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การวาดรูปครั้งที่ 1และ 2 เป็นการวาดเส้นตรงทั้งแนวนอนและแนวตั้งเทียบจากตำแหน่งแขน จะเห็นได้ว่าแขนสามารถเคลื่อนที่เป็นลักษณะของเส้นที่เกือบจะตรงทั้งสองแนว และการวาดรูปในครั้งที่ 3 กำหนดวาดรูปโค้ง จะเห็นได้ว่าแขนวาดออกมาเป็นแนวโค้งต่ำไม่เรียบ จะมีการเคลื่อนตัวเป็นขั้น นั่นเป็นเพราะความละเอียดในการเคลื่อนที่ของเซอร์โวมอเตอร์ยังคงไม่ละเอียดพอ ทำให้เกิดการกระชากขึ้นในทุกตำแหน่งของการเคลื่อนที่ รูปจึงออกมาไม่เรียบ

จะเห็นได้ว่าทั้งรูปที่ 1 , 2 และ 3 เป็นการวาดโดยไม่ยกเมาส์หรือวาดครั้งเดียวไม่ยกมือ แต่รูปที่ 4 และ 5 เป็นการวาดแบบยกมือ ซึ่งผลที่ออกมาในครั้งแรกมีแขนไม่ยกขึ้นจากจุดสุดท้ายของการลากครั้งแรกแต่มันจะเคลื่อนจากจุดสุดท้ายไปยังจุดแรกของการลากครั้งต่อไป ทำให้แขนกลลากเส้นเพิ่มขึ้นมานั่นคือ เส้นที่เรายกมือจากจุดสุดท้ายของลากครั้งแรกไปยังจุดแรกของการลากครั้งที่ 2 แต่เมื่อไปแก้ไขที่โปรแกรมวิชวลเบสิกให้มีการยกแขนขึ้นเมื่อวาดเสร็จ และลงต้องการวาดต่อก็สามารถแก้ไขได้ผลออกมาดังรูปข้างต้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 5
สรุปและวิจารณ์

5.1 สรุปผลการทดลอง

จะเห็นได้ว่าแขนกลที่ได้ออกแบบไว้สามารถนำมาสร้างเป็นแขนกลและใช้งานได้จริงโดยมีวงจรควบคุมและโปรแกรมควบคุมผ่านทางคอมพิวเตอร์ แขนกลในโครงงานนี้เป็นแขนแบบ 4 แกน โดยมีการใช้เซอร์โวมอเตอร์ 4 ตัว และแต่ละตัวได้มีการหาข้อมูลการเคลื่อนที่พื้นฐานเพื่อนำมาใช้ในการทำงานของแขน

จากการทดลอง เมื่อนำแขนมาทดสอบการใช้งานจริง จะเห็นได้ว่า แขนกลสามารถวาดรูปได้ตามแนววิถีนั้นๆ แต่ยังคงไม่แม่นยำนักเป็นเพราะความละเอียดของ การเคลื่อนที่ของเซอร์โวมอเตอร์ยังคงไม่ละเอียดพอ ยังคงมีความผิดพลาดเกิดขึ้น แต่ก็ยอมรับได้ว่า แขนกลสามารถทำงานได้ตามที่ต้องการ

5.2 ปัญหาที่พบและแนวทางแก้ไข

5.2.1 ขีดจำกัดการทำงานของเซอร์โวมอเตอร์ ในโครงงานนี้ใช้เซอร์โวมอเตอร์จำนวน 4 ตัวด้วยกัน โดยใช้เซอร์โวที่มีขนาดเล็ก 1ตัว กลาง 2 ตัวและใหญ่ 1 ตัว ซึ่งเมื่อนำมาใช้งาน ตัวขนาดกลางตัวหนึ่ง ที่นำมาขับเคลื่อนแขนท่อนล่าง ถือว่ามีแรงบิดของเซอร์โวมอเตอร์น้อย เมื่อคำนวณจากน้ำหนักของแขนที่ต้องรองรับ แต่เซอร์โวมอเตอร์ยังคงทำงานได้ถึงแม้จะทำงานค่อนข้างหนักก็ตาม และเซอร์โวมอเตอร์ตัวนี้ทำให้เกิดความผิดพลาดในการเคลื่อนที่ของแขนกลได้ และเซอร์โวมอเตอร์ตัวอื่นก็ทำให้เกิดความคลาดเคลื่อนได้ทั้งนี้เนื่องจาก เมื่อเราเก็บค่าข้อมูลพื้นฐานของเซอร์โวมอเตอร์ตัวต่างๆที่เราใช้จะพบว่าเราไม่สามารถเก็บค่ามุมที่ละเอียดกว่า 1 องศาได้

- แนวทางการแก้ไข คือ ควรเปลี่ยนไปใช้เซอร์โวมอเตอร์ที่มีแรงบิดสูงขึ้นและจะต้องมีเครื่องมือในการวัดมุม(องศา)ที่แน่นอนและละเอียดพอที่จะเก็บค่าข้อมูลพื้นฐานได้

5.2.2 น้ำหนักของแขน ตอนแรกได้ทดลองโดยใช้วัสดุที่ทำมาจากอะคริลิกมาเป็นส่วนประกอบของแขน แต่แผ่นอะคริลิกมีข้อจำกัดต่างๆ เช่น เกิดการแตกเมื่อมีแรงมากระทำมากๆ ดังนั้นเราจึงเลือกใช้โฟมหุ้มด้วยใยไฟเบอร์ เพื่อลดน้ำหนัก แต่น้ำหนักที่มากนี้ส่งผลให้แขนกลกดลงเมื่อมีการทำงาน แม้ว่าเราจะมีการถ่วงตุ้มน้ำหนักไว้แล้วก็ตาม ล่งผลให้เวลาเราเคลื่อนที่แขนไปยังจุดที่เราวาดจะเกิดการตกทำให้เกิดรอยวาดในจุดที่เราไม่ต้องการ

-แนวทางการแก้ไข คือ ควรเลือกใช้วัสดุที่เบา แข็งแรงและทนทาน และจะต้องคำนวณแรงที่ต้องใช้ตามหลักโมเมนต์ และถ่วงตุ้มน้ำหนักให้มากขึ้นเพื่อให้แขนกลทำงานได้เรียบขึ้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5.2.3 ความยืดหยุ่นของปลายปากกา

เมื่อเราทำงานสั่งให้แขนกลวาดรูป เวลาเลื่อนจุดไปน้ำหนักของแขนเป็นเหตุให้ปลายปากกาเคลื่อนที่ลงมากกว่าปกติ ทำให้เกิดความผิดพลาดได้

-แนวทางการแก้ไข คือ สร้างความยืดหยุ่นให้กับปลายปากกาและปากกาก็ยังคงหมุนไปได้

5.2.4 ระบบกลไกในการขับเคลื่อน เฟืองที่เราใช้มีความห่างของฟันค่อนข้างมากทำให้ส่งผลในการเคลื่อนที่และระบบกำลงอีกส่วนเราเลือกใช้สายพาน ส่วนต่างๆในกลไกเหล่านี้ส่งผลให้เกิดข้อผิดพลาดได้

-เลือกระบบกลไกในการขับเคลื่อนที่มีความละเอียดและแน่นอนกว่ากลไกที่ใช้อยู่

5.3 ขอบเขตการดำเนินงานและแนวทางการพัฒนา

ในภาคเรียนนี้ได้สร้างแขนกลเสร็จแล้วโดยมีวงจรควบคุมและโปรแกรมที่ใช้ควบคุมจากคอมพิวเตอร์ สามารถพัฒนาได้โดยการทำให้แขนกลเคลื่อนที่ให้แม่นยำมากยิ่งขึ้น สามารถนำไปพัฒนาเพื่อหาความละเอียดของการเคลื่อนที่ของเซอร์โวมอเตอร์ให้มากยิ่งขึ้นแล้วจะทำให้ได้ข้อมูลพื้นฐานของเซอร์โวมอเตอร์ละเอียดขึ้นด้วย ซึ่งจะทำให้แขนกลเคลื่อนที่ได้ดีและราบเรียบยิ่งขึ้น ซึ่งแขนกลนี้สามารถนำไปพัฒนาเพื่อนำไปใช้งานในอุตสาหกรรมขนาดเล็กได้ เช่น อาจใช้ในการวาดแบบ เจาะงาน หรือตัดชิ้นงานเล็กๆได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เอกสารอ้างอิง

[1] สัจจะ จรัสรุ่งรวีวร. **คู่มือเขียนโปรแกรม Visual Basic 6.0.** (พิมพ์ครั้งที่ 3). นนทบุรี : สำนักพิมพ์ DEV BOOK, 2548

[2] นคร ภักดีชาติ , ชัยวัฒน์ ลิ้มพรจิตรวิไล. **คู่มือการทดลอง dsPIC Microcontroller เบื้องต้นด้วยโปรแกรมภาษา C กับ MPLAB C30.** กรุงเทพฯ : บ.อินโนเวทีฟ เอ็กเพอริเมนท์ จำกัด, 2547.

[3] John J.Craig. **Introduction To Robotics Mechanic & Control.** (Third Edition). USA.: PEARSON Education, 2005

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

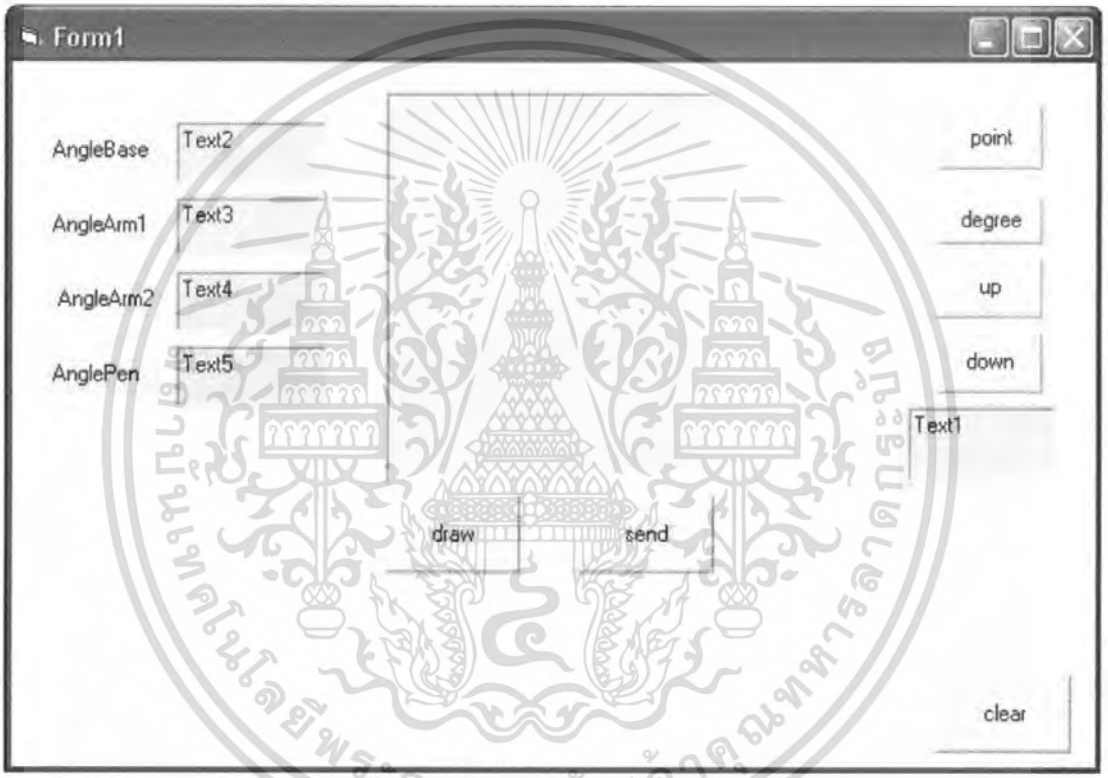
ภาคผนวก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก ก
โปรแกรมการประมวลผลและโปรแกรมควบคุม

ก.1 โปรแกรมควบคุมทางคอมพิวเตอร์

ในที่นี้ใช้โปรแกรมวิชวลเบสิก ซึ่งมีหน้าต่างที่ติดต่อกับผู้ใช้งานและโค๊ดโปรแกรมดังต่อไปนี้



รูปที่ ก.1 หน้าต่างเชื่อมต่อกับผู้ใช้ในโปรแกรมการวาดรูป

```
Dim XX(10000) As Single, YY(10000) As Single, book As Integer, i As Integer
Dim cc As Integer, AA As Integer, tom As Integer, dat As Integer, tom1 As Integer
Dim st1 As Integer, st2 As Integer, st3 As Integer
Dim pulse As Integer
Dim t1 As Integer, t2 As Integer, t3 As Integer, tsum As Integer
Dim p1 As Single, p2 As Single, q1 As Single, Base As Single, D As Single, s1 As Single, s2 As
Single, s3 As Single, b As Single, ss1 As Single, sss1 As Single, ss2 As Single, ss3 As Single
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
Dim angA As Single, angB As Single, angC As Single, angBase As Single
Dim ang1(10000) As Integer, ang2(10000) As Integer, ang3(10000) As Integer, ang4(10000) As
Integer
Private Declare Sub Sleep Lib "kernel32" (ByVal dwMilliseconds As Long)
Private Sub cmdDeg_Click()
   For i = 0 To book - 1
      Form3.Show vbModeless
      Form3.Print ang1(i); ang2(i); ang3(i); ang4(i); i
   Next i
End Sub
Private Sub cmdDraw_Click()
 cc = o
End Sub
Private Sub cmdPoint_Click()
 For i = 0 To book - 1
      Form2.Show vbModeless
      Form2.Print XX(i); YY(i)
 Next i
End Sub
Private Sub Command1_Click()
Text2.Text = " "
End Sub
Private Sub Command4_Click()
 Dim dd As Integer
 Dim da As String
 dd = 90
    mscsend.Output = Chr(dd)
     Sleep (1)
    mscsend.Output = Chr(dd)
    Sleep (1)
    mscsend.Output = Chr(dd)
    Sleep (1)
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    mscsend.Output = Chr(dd)
     Sleep (100)
     pulse = pulse + 10
     Text1.Text = pulse
     Data = mscsend.Input
     'Text2.Text = Data
      'Sleep (1000)
      'Text2.Text = " "
End Sub
Private Sub Command3_Click()
 For i = 0 To book - 1
   Sleep (1)
     mscsend.Output = Chr(ang1(i))
     Sleep (1)
    mscsend.Output = Chr(ang3(i))
    Sleep (1)
    mscsend.Output = Chr(ang4(i))
     Sleep (1)
    mscsend.Output = Chr(ang2(i))
  Sleep (20)
   Next i
   Sleep (500)
   mscsend.Output = Chr(90)
     Sleep (500)
    mscsend.Output = Chr(90)
    Sleep (500)
    mscsend.Output = Chr(90)
     Sleep (500)
    mscsend.Output = Chr(90)
  Sleep (500)
End Sub
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
Private Sub Command5_Click()
Dim dd As Integer, ddd As Integer
 dd = 90
 ddd = 80
    mscsend.Output = Chr(ddd)
     Sleep (1)
    mscsend.Output = Chr(ddd)
    Sleep (1)
    mscsend.Output = Chr(ddd)
     Sleep (1)
    mscsend.Output = Chr(ddd)
     Sleep (100)
     pulse = pulse - 10
     Text1.Text = pulse
End Sub
Private Sub Form_Load()
Dim i As Integer
mscsend.CommPort = 1
mscsend.Settings = "9600,n,8,1"
mscsend.PortOpen = True
dat = 90
pulse = 12000
 tom = 0
End Sub
Private Sub MSComm1_OnComm()
End Sub
Private Sub picture1_MouseDown(Button As Integer, Shift As Integer, x As Single, y As Single)
   Picture1.CurrentX = x
   Picture1.CurrentY = y
End Sub
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
Private Sub Picture1_MouseMove(Button As Integer, Shift As Integer, x As Single, y As Single)
  Dim ang11 As Single, ang22 As Single, ang33 As Single, ang44 As Single
  Picture1.ScaleMode = 0
  Picture1.Scale (0, 145)-(200, -145)
  AA = 1
  If Button = vbLeftButton Then
    If cc = 0 Then
      Picture1.Line (Picture1.CurrentX, Picture1.CurrentY)-(x, y)
      XX(book) = x
      YY(book) = y
      p1 = XX(book)
      q1 = YY(book)
        p2 = p1 + 240
        Base = q1 / p2
        angBase = Atn(Base)
        ang4(book) = Abs(((angBase * 57.3) - 90) + 10) 'base
        ang44 = Abs(((angBase * 57.3) - 90) + 10)
        D = Sqr((p2 * p2) + (q1 * q1))
        s1 = D * D
        sss1 = s1 + 11125
        ss1 = sss1 / (470 * D)
        angA = Atn(-ss1 / Sqr(-ss1 * ss1 + 1)) + 2 * Atn(1)
        ang1(book) = 90 - (55 - (angA * 57.3)) 'down
        ang11 = 90 - (55 - (angA * 57.3))
        s2 = 99325 - (D * D)
        ss2 = s2 / 98700
        angB = Atn(-ss2 / Sqr(-ss2 * ss2 + 1)) + 2 * Atn(1)
        ang2(book) = Abs(90 - ((angB * 57.3) - 77) - 102) + 90 'up
        ang22 = ang2(book)

        s3 = (D * D) - 11125
        ss3 = s3 / (420 * D) ' c1 = 2 * R * R
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
            angC = Atn(-ss3 / Sqr(-ss3 * ss3 + 1)) + 2 * Atn(1) 'c2 = a1 * a1
            ang3(book) = 90 - (angC * 57.3) 'c3 = c1 - c2 'pen
            ang33 = 90 - (angC * 57.3)
            'c4 = c3 / c1
            'radc = Atn(-c4 / Sqr(-c4 * c4 + 1)) + 2 * Atn(1)
            'degc(no) = 97 - (radc * 57.3) * 0.5 'small
            'degc(no) = 40 + (radc * 57.3)
            book = book + 1
            Text2.Text = ang44
            Text3.Text = ang11
            Text4.Text = ang22
            Text5.Text = ang33
        End If
    End If
End Sub
Private Sub cmdCls_Click()
    cc = 1
    Picture1.Cls
    Form2.Cls
    Form3.Cls
    book = 0
End Sub
```

และโค๊ดโปรแกรมประมวลผล ดังนี้

```
#include <p30f4011.h>
#include <uart.h>
#include <timer.h>
_FOSC(CSW_FSCM_ON & XT_PLL8);

_FWDT(WDT_OFF);                 // disable WDT
_FBORPOR(PBOR_OFF & MCLR_EN);   // disable PBOR, enable MCLR
_FGS(CODE_PROT_OFF);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
#define L1          PORTBbits.RB0   //big down
//#define L2        PORTCbits.RC13 //big on
//#define L3        PORTDbits.RD0   //small
#define L2          PORTEbits.RE5//big on
#define L3          PORTDbits.RD0   //small
#define L4          PORTEbits.RE0  //smallest


char aaa1=0;
char aaa2=0;
char aaa3=0;
char aaa4=0;
int ti=0;
int dat;
int deg=0;
long set1=160000;
long set2=160000;
long set3=160000;
long set4=160000;
int D1=12000;
int D2=12000;
int D3=12000;
int D4=12000;

int
motorbase[152]={0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,6644,6706,6772,6830,6904,6970,7038,
7114,7186,7248,7318,7380,7446,7514,7584,7672,7748

,7830,7920,8034,8130,8198,8264,8350,8434,8512,8574,8658,8730,8800,8884,8942,9038,9108,9
180,9246,9334,9398,9482,9576,9638,9720
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

,9786,9868,9952,10054,10114,10192,10290,10352,10430,10512,10586,10668,10754,10832,1092
8,10990,11096,11174,11250,11338,11422,11504

,11570,11650,11750,11834,11912,12000,12240,12328,12414,12494,12574,12658,12742
,12834,12928,13006,13094,13176,13272,13332,13440,13508

,13608,13686,13772,13862,13936,14006,14088,14178,14262,14344,14420,14508,14576
,14648,14724,14814,14902,14970,15064,15142,15240,15320

,15384,15468,15550,15626,15702,15788,15874,15946,16028,16094,16196,16254,16334
,16408,16490,16566,16628,16730,16796,16882,16950,17016,17132};

int
motordown[101]={0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0
,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0

,14688,14474,14376,14258,14168,14110,14022,13916,13846,13706,13576,13510,13330,13186,1
3124,12984,12894,12778,12662,12546,12470

,12412,12292,12116,12000,11904,11654,11548,11480,11416,11282,11220,11098,10964,10876};

int
motorup[171]={0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,
0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,11587,11692,11790,
11868,11910,11962,12000,12186,12336,12452,12538,12614,12680,12762,12840,12914,12964,1
3044,13146,13254,13332,13448,13524,13618,13678,13732,13796,13896,14034,14104,14158,14
224,14330,14386

,14488,14578,14620,14706,14792,14868,14932,15028,15110,15166,15238,15340,15410,15454,1
5524,15608,15726,15802,15876,15932,16006,16106,16182,16254,16316,16374,16468,16520,16

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
598,16704,16766,16842,16914416980,17060,17134,17222,17282,17330,17452,17532,17630,177
00,17780,17840,17920,17976,18052,18106,18170,18264,18364,18430};
int motorpen[106
]={0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0


        ,8577,8643,8733,8840,8917,9010,9097,9183,9270,9353,9457,9540,9627,9720,9827,992
3,9997,10103,10177,10277


        ,10370,10473,10540,10600,10677,10750,10830,10917,11020,11143,11223,11313,11397
,11463,11567,11660,11727,11793,11860,11920,12000


        ,12153,12263,12377,12530,12537,12677,12840,12910,13003,13087,13190,13270,13373
,13480,13590};




void _ISR _U2TXInterrupt(void)
{
  IFS1bits.U2TXIF=0;
}
void _ISR _U2RXInterrupt(void)
{
  IFS1bits.U2RXIF=0;
}
/*void _ISR _U1TXInterrupt(void)
{
  IFS0bits.U1TXIF=0;
}
void _ISR _U1RXInterrupt(void)
{
  IFS0bits.U1RXIF=0;// if(L5==0) {L5=1;} else {L5=0;}
  //L5=1;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
}*/
void __attribute__((__interrupt__)) _T1Interrupt( void )
{  T1CONbits.TON = 0;
        IFS0bits.T1IF = 0;

   if(aaa1==0) {  L1=0;
            //PR1=D1;
             if(set1>D1) {PR1=D1; set1=set1-D1;} else {PR1=set1;aaa1=1;}

            T1CONbits.TON = 1;

            }
          else {L1=1;
             PR1=D1;aaa1=0;
             set1=160000-D1;
             T1CONbits.TON = 1;

             }
  /* if(aaa1==0) {PR1=16928;aaa1=1;
            T1CONbits.TON = 1;
             L1=0;}
          else {PR1=12000;aaa1=0;
             T1CONbits.TON = 1;
            L1=1;}*/
}
void __attribute__((__interrupt__)) _T2Interrupt( void )
{  T2CONbits.TON = 0;
        IFS0bits.T2IF = 0;


   if(aaa2==0) {  L2=0;
            //PR1=D1;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
            if(set2>D2) {PR2=D2; set2=set2-D2;} else {PR2=set2;aaa2=1;}

           T2CONbits.TON = 1;

           }
         else {L2=1;
            PR2=D2;aaa2=0;
            set2=160000-D2;
            T2CONbits.TON = 1;}
}
void __attribute__((__interrupt__)) _T3Interrupt( void )
{
        T3CONbits.TON = 0;
        IFS0bits.T3IF = 0;

if(aaa3==0) {  L3=0;
            //PR1=D1;
            if(set3>D3) {PR3=D3; set3=set3-D3;} else {PR3=set3;aaa3=1;}

           T3CONbits.TON = 1;

           }
         else {L3=1;
            PR3=D3;aaa3=0;
            set3=160000-D3;
            T3CONbits.TON = 1;}
}
void __attribute__((__interrupt__)) _T4Interrupt( void )
{
        T4CONbits.TON = 0;
        IFS1bits.T4IF = 0;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
  if(aaa4==0) {  L4=0;
                //PR1=D1;
                 if(set4>D4) {PR4=D4; set4=set4-D4;} else {PR4=set4;aaa4=1;}

               T4CONbits.TON = 1;

               }
             else {L4=1;
                 PR4=D4;aaa4=0;
                 set4=160000-D4;
                 T4CONbits.TON = 1;}

}
void Init_Timer1(void)
{  T1CON = 0;
          IFS0bits.T1IF = 0;
          IPC0bits.T1IP = 1;
          IEC0bits.T1IE = 1;

          PR1 = D1;
          T1CONbits.TCS = 0;

          T1CONbits.TON = 1;
}
void Init_Timer2( void )
{

    T2CON = 0;
          IFS0bits.T2IF = 0;
          IPC1bits.T2IP = 2;
          IEC0bits.T2IE = 1;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        PR2 = D2;
        T2CONbits.TCS = 0;


        T2CONbits.TON = 1;

}

void Init_Timer3(void)
{  T3CON = 0;
        IFS0bits.T3IF = 0;
        IPC1bits.T3IP = 3;
        IEC0bits.T3IE = 1;

        PR3 = D3;
        T3CONbits.TCS = 0;

        T3CONbits.TON = 1;
}
void Init_Timer4( void )
{

   T4CON = 0;
        IFS1bits.T4IF = 0;
        IPC5bits.T4IP = 4;
        IEC1bits.T4IE = 1;

        PR4 = D4;
        T4CONbits.TCS = 0;


        T4CONbits.TON = 1 ;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
}
void delay(int t)
{
  int i,j;
   for(i=0;i<t;i++)
   for(j=0;j<250;j++);
}



void init()
{
 unsigned int baudvalue;
 unsigned int U2MODEvalue;
 unsigned int U2STAvalue;

 CloseUART2();
 ConfigIntUART2(UART_RX_INT_EN
&UART_RX_INT_PR6&UART_TX_INT_EN&UART_TX_INT_PR2 );
 baudvalue=51;//
  U2MODEvalue=UART_EN &
          UART_IDLE_CON &
          UART_RX_TX &
          UART_DIS_WAKE &
          UART_DIS_LOOPBACK &
          UART_DIS_ABAUD &
          UART_NO_PAR_8BIT &
          UART_1STOPBIT;
 U2STAvalue= UART_INT_TX_BUF_EMPTY &
          UART_TX_PIN_NORMAL &
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        UART_TX_ENABLE &
        UART_INT_RX_3_4_FUL &
        UART_ADR_DETECT_DIS &
        UART_RX_OVERRUN_CLEAR ;
  OpenUART2(U2MODEvalue,U2STAvalue,baudvalue);
}

main()
{
 //char Txdata[]="\r\n TOM  \r\n";
 //int i,j;
//int way;
  TRISB=0;
  TRISF=0;
  TRISD=0;
  TRISE=0;
  //uart1_init();
  //uart2_init();
   init();
  Init_Timer1();
  Init_Timer2();
  Init_Timer3();
  Init_Timer4();

   //putsUART1((unsigned int *)Txdata);
  // while(BusyUART1());
    putsUART2((unsigned int *)"ready");
   while(BusyUART2());
 while(1)
 {

  if(DataRdyUART2())
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    {
     /*dat = ReadUART2();//database
     if(deg==0)  {if(dat==90) {D1=D1+100;deg=1;} else {D1=D1-100;deg=1;}}
     else if(deg==1)   {if(dat==90) {D2=D2+100;deg=2;} else {D2=D2-100;deg=2;}}
     else if(deg==2){if(dat==90) {D3=D3+100;deg=3;} else {D3=D3-100;deg=3;}}
     else {if(dat==90) {D4=D4+100;deg=0;} else {D4=D4-100;deg=0;}}
     */

     dat = ReadUART2();//send
     if(deg==0)       {D1=motorbase[dat];deg=1;}
     else if(deg==1)   {D2=motordown[dat];deg=2;}
     else if(deg==2)   {D3=motorup[dat];deg=3;}
     else              {D4=motorpen[dat];deg=0;}


    }

}


}
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก ข

ผลการทดลองหาข้อมูลพื้นฐานของเซอร์โวมอเตอร์แต่ละตัว

ข.1 ข้อมูลพื้นฐานของเซอร์โวมอเตอร์ที่ควบคุมฐาน

ตารางที่ ข.1 ผลการทดลองของเซอร์โวมอเตอร์ที่ควบคุมฐาน

มุม	ความกว้าง พัลส์		มุม	ความกว้าง พัลส์		มุม	ความกว้าง พัลส์	
(องศา)	หมุนไป	หมุนกลับ	(องศา)	หมุนไป	หมุนกลับ	(องศา)	หมุนไป	หมุนกลับ
61	9610	9630	81	11240	11240	101	13120	13070
62	9710	9720	82	11320	11330	102	13230	13140
63	9760	9780	83	11420	11400	103	13290	13250
64	9860	9870	84	11480	11490	104	13370	13300
65	9950	9940	85	11560	11560	105	13500	13380
66	10050	10040	86	11620	11630	106	13540	13480
67	10100	10110	87	11720	11740	107	13630	13580
68	10170	10180	88	11800	11820	108	13690	13650
69	10280	10290	89	11900	11900	109	13790	13730
70	10360	10340	90	12000	12000	110	13880	13820
71	10430	10420	91	12240	12230	111	13970	13910
72	10500	10510	92	12340	12320	112	14060	13960
73	10580	10570	93	12420	12400	113	14180	14060
74	10650	10660	94	12500	12490	114	14270	14160
75	10730	10740	95	12590	12560	115	14350	14240
76	10810	10820	96	12670	12640	116	14440	14320
77	10920	10930	97	12750	12730	117	14520	14380
78	10980	10990	98	12870	12800	118	14610	14480
79	11090	11070	99	12940	12910	119	14640	14550
80	11150	11160	100	13040	12960	120	14670	14630

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ข.2 ข้อมูลพื้นฐานของเซอร์โวมอเตอร์ที่ควบคุมแขนท่อนล่าง

ตารางที่ ข.2 ผลการทดลองของเซอร์โวมอเตอร์ที่ควบคุมแขนท่อนล่าง

มุม	ความกว้าง พัลส์		มุม	ความกว้าง พัลส์	
(องศา)	หมุนไป	หมุนกลับ	(องศา)	หมุนไป	หมุนกลับ
66	14730	14670	90	12000	12000
67	14470	14480	91	11900	11900
68	14370	14350	92	11670	11640
69	14250	14270	93	11530	11560
70	14160	14180	94	11490	11510
71	14100	14130	95	11420	11420
72	14020	14030	96	11280	11280
73	13920	13920	97	11210	11220
74	13850	13850	98	11090	11100
75	13700	13720	99	10950	10980
76	13570	13580	100	10870	10880
77	13500	13510	101	10890	10800
78	13300	13310	102	10670	10670
79	13180	13180	103	10480	10490
80	13120	13120	104	10360	10370
81	13000	12990	105	10280	10290
82	12890	12900	106	10180	10190
83	12770	12770	107	10080	10090
84	12660	12650	108	10000	10000
85	12550	12540	109	9860	9850
86	12490	12460	110	9790	9790
87	12440	12390	111	9700	9690
88	12310	12270	112	9590	9580
89	12120	12110	113	9470	9470

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ข.3 ข้อมูลพื้นฐานของเซอร์โวมอเตอร์ที่ควบคุมแขนท่อนบน

ตารางที่ ข.3 ผลการทดลองของเซอร์โวมอเตอร์ที่ควบคุมแขนท่อนบน

มุม	ความกว้าง พัลส์		มุม	ความกว้าง พัลส์		มุม	ความกว้าง พัลส์	
(องศา)	หมุนไป	หมุนกลับ	(องศา)	หมุนไป	หมุนกลับ	(องศา)	หมุนไป	หมุนกลับ
84	11570	11590	108	13670	13670	132	15510	15530
85	11650	11730	109	13720	13730	133	15590	15600
86	11820	11780	110	13780	13800	134	15710	15720
87	11880	11880	111	13890	13890	135	15800	15790
88	11910	11910	112	14010	14020	136	15860	15870
89	11960	11950	113	14090	14100	137	15920	15930
90	12000	12000	114	14150	14160	138	15980	16010
91	12170	12190	115	14200	14220	139	16100	16100
92	12330	12330	116	14320	14320	140	16180	16180
93	12450	12450	117	14380	14380	141	16240	16260
94	12530	12540	118	14460	14490	142	16310	16320
95	12610	12610	119	14570	14580	143	16370	16380
96	12670	12680	120	14620	14630	144	16450	16470
97	12750	12760	121	14690	14700	145	16520	16510
98	12830	12830	122	14780	14790	146	16580	16590
99	12910	12910	123	14860	14870	147	16690	16700
100	12960	12960	124	14920	14930	148	16750	16750
101	13030	13040	125	15020	15020	149	16830	16820
102	13140	13130	126	15090	15100	150	16890	16920
103	13240	13230	127	15160	15160	151	16970	16980
104	13320	13320	128	15220	15250	152	17040	17060
105	13440	13450	129	15330	15340	153	17120	17130
106	13520	13530	130	15400	15400	154	17220	17200
107	13610	13620	131	15450	15460	155	17270	17280

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ข.4 ข้อมูลพื้นฐานของเซอร์โวมอเตอร์ที่ควบคุมส่วนปากกา

ตารางที่ ข.4 ผลการทดลองของเซอร์โวมอเตอร์ที่ควบคุมส่วนปากกา

มุม	ความกว้าง พัลส์		มุม	ความกว้าง พัลส์		มุม	ความกว้าง พัลส์	
(องศา)	หมุนไป	หมุนกลับ	(องศา)	หมุนไป	หมุนกลับ	(องศา)	หมุนไป	หมุนกลับ
50	8600	8570	74	10670	10680	98	12930	12890
51	8640	8640	75	10740	10770	99	13030	13000
52	8730	8730	76	10810	10860	100	13110	13080
53	8800	8840	77	10880	10940	101	13210	13190
54	8870	8920	78	10950	11080	102	13300	13230
55	8980	9010	79	11110	11170	103	13400	13350
56	9060	9090	80	11180	11270	104	13490	13470
57	9150	9180	81	11280	11350	105	13590	13590
58	9240	9270	82	11360	11430	106	13740	13700
59	9320	9330	83	11440	11480	107	13830	13830
60	9470	9390	84	11520	11580	108	13950	13920
61	9600	9470	85	11620	11680	109	14060	14020
62	9660	9570	86	11700	11760	110	14110	14060
63	9730	9670	87	11790	11790	111	14270	14200
64	9830	9810	88	11850	11870	112	14350	1432
65	9940	9910	89	11900	11930	113	14390	14390
66	9990	10000	90	12000	12000	114	14460	14470
67	10120	10130	91	12160	12140	115	14560	14550
68	10180	10200	92	12270	12270	116	14650	14650
69	10280	10290	93	12380	12390	117	14800	14800
70	10360	10390	94	12530	12540	118	14870	14870
71	10480	10490	95	12610	12600	119	14960	14970
72	10550	10540	96	12690	12670	120	15020	15010
73	10580	10590	97	12830	12840	121	15150	15170

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก ค

ข้อมูลความรู้ทางด้านอิเล็กทรอนิกส์

dsPIC30F4011

dsPIC30F4011 Enhanced Flash

16-bit Digital Signal Controller

Note: This data sheet summarizes features of this group of dsPIC30F devices and is not intended to be a complete reference source. For more information on the CPU, peripherals, register descriptions and general device functionality, refer to the dsPIC30F Family Reference Manual (DS70046). For more information on the device instruction set and programming, refer to the dsPIC30 Programmer's Reference Manual (DS70030).

High Performance Modified RISC CPU:

• Modified Harvard architecture

• C compiler optimized instruction set architecture with flexible addressing modes

• 84 base instructions

• 24-bit wide instructions, 16-bit wide data path

• 48 Kbytes on-chip Flash program space (16K Instruction words)

• 2 Kbytes of on-chip data RAM

• 1 Kbytes of non-volatile data EEPROM

• Up to 30 MIPs operation:

-DC to 40 MHz external clock input 4 MHz-10 MHz oscillator input with PLL active (4x, 8x, 16x)

• 30 interrupt sources 3 external interrupt sources 8 user selectable priority levels for each interrupt source 4 processor trap sources

• 16 x 16-bit working register array

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

DSP Engine Features:

- Dual data fetch
- Accumulator write back for DSP operations
- Modulo and Bit-Reversed Addressing modes
- Two, 40-bit wide accumulators with optional saturation logic
- 17-bit x 17-bit single cycle hardware fractional/ integer multiplier
- All DSP instructions single cycle
- 16-bit single cycle shift

Peripheral Features:

- High current sink/source I/O pins: 25 mA/25 mA
- Timer module with programmable prescaler:

-Five 16-bit timers/counters; optionally pair 16-bit timers into 32-bit timer modules

- 16-bit Capture input functions
- 16-bit Compare/PWM output functions
- 3-wire SPI™ modules (supports 4 Frame modes)
- I2C™ module supports Multi-Master/Slave mode and 7-bit/10 bit addressing
- 2 UART modules with FIFO Buffers • 1 CAN modules, 2.0B compliant

Motor Control PWM Module Features:

- 6 PWM output channel

-Complementary or Independent Output modes

-Edge and Center Aligned modes

- 3 duty cycle generators
- Dedicated time base
- Programmable output polarity
- Dead-time control for Complementary mode
- Manual output control
- Trigger for A/D conversions

Quadrature Encoder Interface Module Features:

- Phase A, Phase B and Index Pulse input
- 16-bit up/down position counter
- Count direction status
- Position Measurement (x2 and x4) mode
- Programmable digital noise filters on inputs

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

• Alternate 16-bit Timer/Counter mode
• Interrupt on position counter rollover/underflow

Analog Features:

• 10-bit Analog-to-Digital Converter (A/D) with 4 S/H Inputs: 500 Ksps conversion rate 9 input channels

-Conversion available during Sleep and Idle

• Programmable Brown-out Detection and Reset generation

Special Microcontroller Features:

• Enhanced Flash program memory: 10,000 erase/write cycle (min.) for industrial temperature range, 100K (typical)

• Data EEPROM memory: 100,000 erase/write cycle (min.) for industrial temperature range, 1M (typical)

• Self-reprogrammable under software control

• Power-on Reset (POR), Power-up Timer (PWRT) and Oscillator Start-up Timer (OST)

• Flexible Watchdog Timer (WDT) with on-chip low power RC oscillator for reliable operation

• Fail-Safe clock monitor operation detects clock failure and switches to on-chip low power RC oscillator

• Programmable code protection

• In-Circuit Serial Programming™ (ICSP™)

• Selectable Power Management modes

-Sleep, Idle and Alternate Clock modes

CMOS Technology:

• Low power, high speed Flash technology

• Wide operating voltage range (2.5V to 5.5V)

• Industrial and Extended temperature ranges

• Low power consumption

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้