สำนักหอสมุดกลาง พระจอมเกล้าลาดกระบัง

แขนกลอัตโนมัติ

AUTOMATIC ROBOT ARM

นายวิทวัส สีลพิพัฒน์

นายสมชาย คีรีกุลเสนา

นายเศรษฐวัฒน์ จงวัฒนาพาณิชย์

เลขหมู่.....
ทะเบียน 119187
วัน,เดือน,ปี - 6 ส.ค. 2554

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

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต

สาขาวิชาวิศวกรรมแมคคาทรอนิกส์

คณะวิศวกรรมศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2553

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

AUTOMATIC ROBOT ARM

Mr. Wittawas Silaphiphat

Mr. Somchai Keereegulsena

Mr. Sesstawat Chongwattanapanit

THIS THESIS IS SUBMITTED IN PARTIAL FULFILLMENT

OF THE REQUIREMENTS FOR THE DEGREE OF

BACHELOR OF ENGINEERING IN MECHATRONICS ENGINEERING

FACULTY OF ENGINEERING

KING MONGKUT'S INSTITUTE OF TECHNOLOGY LADKRABANG

ACADAMIC YEAR 2010

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ปริญญานิพนธ์ปีการศึกษา 2553

สาขาวิชาวิศวกรรมการวัดและควบคุม คณะวิศวกรรมศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เรื่อง แขนกลอัตโนมัติ

Automatic Robot Arm

ผู้จัดทำ	นาย วิทวัส	สีลพิพัฒน์	50011468
	นาย เศรษฐวัฒน์	จงวัฒนาพาณิชย์	50011605
	นาย สมชาย	คีรีกุลเสนา	50011620

.....อาจารย์ที่ปรึกษา

(รศ.ดร. เกียรติศักดิ์ คมวัชระ)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

แขนกลอัตโนมัติ

โดย

นาย วิทวัส สีลพิพัฒน์ 50011468

นาย เศรษฐวัฒน์ จงวัฒนาพานิชย์ 50011605

นาย สมชาย คีรีกุลเสนา 50011620

อาจารย์ที่ปรึกษา

รศ.ดร. เกียรติศักดิ์ คมวัชระ

ปีการศึกษา 2553

บทคัดย่อ

ปริญญานิพนธ์ฉบับนี้ นำเสนอทฤษฎีและการออกแบบระบบอัตโนมัติ สำหรับการควบคุมการทำงานของแขนกลในการจับวัตถุ โดยโครงสร้างของระบบประกอบด้วย แขนกล กล้องเว็บแคม คอมพิวเตอร์ และวงจรอิเล็กโทรนิกส์ที่เกี่ยวข้อง จุดมุ่งหมายของโครงงานนี้คือการให้แขนกลสามารถจับวัตถุ โดยการคำนวณหาตำแหน่งของวัตถุผ่านการประมวลผลภาพ

ขั้นตอนการดำเนินการ เริ่มจากการออกแบบและสร้างแขนกลตามที่ได้ศึกษาและออกแบบวงจรอิเล็กทรอนิกส์ที่จำเป็นสำหรับควบคุมแขนกล ซึ่งประกอบด้วยวงจรประกอบการใช้งานของไมโครคอน โทรลเลอร์และวงจรขับมอเตอร์หลังจากนั้นจึงเขียนโปรแกรมการหาตำแหน่งและคัดแยกสีผ่านกระบวน การประมวลผลภาพ และนำข้อมูลมาประมวลผลผ่านทางคอมพิวเตอร์ที่เขียนโปรแกรมด้วยภาษา ซีชาร์ป ซึ่งเปลี่ยนค่าตัวแปรต่างๆ ให้เหมาะสม และส่งข้อมูลไปยังไมโครคอนโทรลเลอร์ผ่านทางพอร์ตอนุกรม จากนั้นทดลองพบว่าระบบการทำงานที่ได้ออกแบบสามารถทำให้แขนกลจับวัตถุได้ตามต้องการ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ROBOT ARM AUTOMATIC

By

Mr. Wittawas Silaphiphat 50011468

Mr. Sesstawat Chongwattnapanit 50011605

Mr. Somchai Keereegulsena 50011620

Advisor

Assoc. Dr. Kiattisak Kumwachara

Academic Year 2553

ABSTRACT

This thesis is presented theory and design in automatic's system. Controlling robot's arm to crash objects is consisted of robot's arm , webcam , computer and electronical circuitry.Aim of this project is making robot's arm to crash a thing by calculating the position of objects though image processing.

First , we designed and made robot's arm from pattern that we had designed before.And then , we studied and designed electronical circuitry that is necessary for controlling robot's arm. This electronical circuitry consisted of microcontroller circuitry and motor circuitry. After that , we made a program that can find a position and divide color though image processing. And then , we processed data though computer which is written by C# language. We also changed variable to be suitabled. And then , we sent data to microcontroller though sequence ports. Micro-controller processed and sent signal to control robot's arm. From experiment's result is showed that working system is designed for robot's arm to crash anything

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กิตติกรรมประกาศ

การจัดทำปริญญานิพนธ์ฉบับนี้ สามารถสำเร็จลุล่วงไปได้ด้วยดี เพราะได้รับความช่วยเหลือเป็นอย่างดีจากอาจารย์เกียรติศักดิ์ ที่ได้กรุณาให้คำปรึกษา แนะนำที่ดีมาโดยตลอดนับตั้งแต่ต้นรวมทั้งเอื้อเฟื้ออุปกรณ์ที่จำเป็น และความช่วยเหลืออื่นๆ ที่เป็นประโยชน์ต่อโครงงาน ผู้จัดทำรู้สึกซาบซึ้งและขอกราบขอบพระคุณอย่างสูง

ขอขอบคุณ เพื่อนๆ พี่ๆ น้องๆ ทุกคนที่ให้กำลังในและช่วยสนับสนุนอุปกรณ์ที่ขาดเหลือ กระตุ้นตือนและคอยถามไถ่ความคืบหน้าของโครงงานอยู่เสมอ

สุดท้ายนี้ผู้จัดทำขอกราบขอบพระคุณบิดา มารดา และครอบครัว ที่คอยเป็นกำลังใจที่ดีตลอดมารวมถึงการสนับสนุนในเรื่องของงบประมาณที่ขาดเหลือ ตลอดจนเป็นแรงบันดาลใจที่ดีที่สุดที่ทำให้โครงงานนี้สำเร็จสมบูรณ์ลงได้

ผู้จัดทำ

นายวิทวัส สีลพิพัฒน์

นายสมชาย คีรีกุลเสนา

นายเศรษฐวัฒน์ จงวัฒนาพาณิชย์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ

	หน้า
บทคัดย่อ	**I**
บทคัดย่อภาษาอังกฤษ	**II**
กิตติกรรมประกาศ	**III**
สารบัญ	**IV**
สารบัญภาพ	**VII**
สารบัญตาราง	**IX**
บทที่ 1 บทนำ	1
1.1 กล่าวนำ	1
1.2 วัตถุประสงค์ในการทำปริญญานิพนธ์	1
1.3 ขั้นตอนการศึกษาและจัดทำโครงงาน	2
1.4 รายละเอียดของปริญญานิพนธ์	2
บทที่ 2 ความรู้และทฤษฎีที่เกี่ยวข้อง	4
2.1 ทฤษฎีพื้นฐานของการวิเคราะห์หุ่นยนต์	4
2.1.1 การแบ่งแยกประเภทหุ่นยนต์	4
2.1.2 ชนิดการทำงานของหุ่นยนต์ แบ่งตามลักษณะการทำงานของหุ่นยนต์	6
2.2 หลักการทำงานของเซอร์โวมอเตอร์	11
2.3 ทฤษฎีการประมวลผลภาพ	12

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ(ต่อ)

	หน้า
2.3.1 การกำหนดสี	12
2.3.2 องค์ประกอบของสี	13
2.3.3 ภาพดิจิตอล	14
2.3.4 การประมวลผลภาพเชิงตัวเลข	16
2.3.5 ข้อมูลภาพชนิดบิตแมป	16
2.3.6 การสร้างภาพไบนารี	17
2.4 การฟิลเตอร์แบบค่ากลาง	18
2.5 การวิเคราะห์รูปทรง	19
2.5.1 หลักการอธิบายพื้นที่	19
2.5.2 โมเมนต์	20
บทที่ 3 หลักการออกแบบ	21
3.1 หลักการทำงานของแขนกล	21
3.2 หลักการออกแบบชิ้นงาน	22
3.3 การออกแบบโปรแกรม	22
3.4 การออกแบบวงจรควบคุมการทำงานของแขนกล	23
3.5 สมการแขนกล	26
บทที่ 4 ผลการทดลอง	30
4.1 การทดสอบมอเตอร์	30

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ(ต่อ)

หน้า

4.2 การติดตั้งและเชื่อมต่ออุปกรณ์ 30

4.3 ปรับแต่งอุปกรณ์ ก่อนทำการทดลอง 31

4.3.1 การปรับตำแหน่งกล้องเว็บแคม 31

4.3.2 การปรับเทียบขนาดของพิกเซล 31

4.4 ขั้นตอนการทดลองและผลการทดลอง 32

4.4.1 การทดสอบความแม่นยำของแขนกล 32

4.4.2 การทดลองหาตำแหน่งของวัตถุ 33

4.4.3 การทดลองการจับวัตถุ 35

บทที่ 5 สรุปผลการทดลอง

37 5.1 สรุปผลการดำเนินงาน 37

5.2 ปัญหาและอุปสรรค 37

5.3 แนวทางการแก้ไข 38

5.4 แนวทางพัฒนา 38

ภาคผนวก ก วงจรภายในของบอร์ด ET-BASE Dspic30F4011 39

ภาคผนวก ข โปรแกรมประมวลผลภาพ 40

เอกสารอ้างอิง 56

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ(ภาพ)

รูปที่ **หน้า**

รูปที่ 2.1a Cartesian Robot 6

รูปที่ 2.1b Work envelope of Cartesian Robot 6

รูปที่ 2.2a Cylindrical Robot 7

รูปที่ 2.2b Work envelope of Cylindrical Robot 7

รูปที่ 2.3a Spherical Robot 8

รูปที่ 2.3b Work envelope of Spherical Robot 8

รูปที่ 2.4a Scara Robot 9

รูปที่ 2.4b Work envelope of Scara Robot 9

รูปที่ 2.5a Articulated Robot 10

รูปที่ 2.5b Work envelope of Articulated Robot 10

รูปที่ 2.6 กล่องสีอาร์จีบี 13

รูปที่ 3.1 แสดงแผนผังการทำงาน 21

รูปที่ 3.2 แสดงหน้าต่างของโปรแกรม Visualbasic ที่ใช้ในการประมวลสัญญาณภาพ 23

รูปที่ 3.3 ET-BASE DSPIC30F4011 24

รูปที่ 3.4 ระนาบต่างๆ ของแขนกล 26

รูปที่ 4.1 แสดงการติดตั้งเชื่อมต่ออุปกรณ์ 31

รูปที่ 4.2 รูปแสดงการปรับเทียบของพิกเซล 32

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ(ภาพ)

รูปที่ **หน้า**

รูปที่ 4.3 แสดงการจับวัตถุสีขาวที่โซน A 34

รูปที่ 4.4 แสดงการจับวัตถุสีขาวที่ตำแหน่งกึ่งกลาง 34

รูปที่ 4.5 แสดงผลการจับวัตถุสีขาวที่โซน B 35

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ(ตาราง)

ตารางที่	หน้า
ตารางที่ 2.1 ชนิดของข้อต่อหุ่นยนต์	5
ตารางที่ 2.2 ขอบเขตการทำงานของหุ่นยนต์ที่ขึ้นอยู่กับแกนหลัก	5
ตารางที่ 2.3 คุณลักษณะของหุ่นยนต์	11
ตารางที่ 3.1 ค่าพารามิเตอร์ต่างๆ ของแขนกล	27
ตารางที่ 4.1 สเปกของมอเตอร์	30
ตารางที่ 4.2 ผลการทดลองความแม่นยำแขนกลในแนวแกน X เมื่อกำหนดให้แกน Y = 0	33
ตารางที่ 4.3 ผลการทดลองความแม่นยำแขนกลในแนวแกน Y เมื่อกำหนดให้แกน X = 0	33
ตารางที่ 4.4 แสดงผลการจับวัตถุสีขาวของกล้องเว็บแคม	35

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 1

บทนำ

1.1 กล่าวนำ

ในปัจจุบันเทคโนโลยีทางด้านวิศวกรรมได้มีการพัฒนาอย่างรวดเร็วและพร้อมตอบสนองความต้อง การของมนุษย์มากขึ้น อุปกรณ์เครื่องใช้ต่างๆ มีการพัฒนาตามยุคสมัยทำให้กระบวนการกรผลิตสิ่งต่างๆ มีการพัฒนาด้วยเช่นกัน ด้วยเหตุนี้จึงทำให้ภาคอุตสาหกรรมต้องการศักยภาพในการผลิตมากยิ่งขึ้น ในสมัยก่อนการผลิตชิ้นงานต่างๆ ส่วนใหญ่จะใช้แรงงานจากมนุษย์ แต่ในปัจจุบันเมื่อต้องการความแม่นยำและรวดเร็วในการผลิต จึงมีการนำเทคโนโลยีหุ่นยนต์เข้ามาแทนที่มนุษย์ เนื่องจากหุ่นยนต์มีความแม่นยำ รวดเร็ว สามารถทำงานในที่ที่อันตรายได้ รวมทั้งเวลาในการหยุดงานน้อยกว่ามนุษย์ จึงถือได้ว่าการนำหุ่นยนต์มาใช้ในกระบวนการการผลิตเป็นอีกทางเลือกหนึ่งที่น่าสนใจ

แขนกลเป็นหนึ่งในเทคโนโลยีหุ่นยนต์ที่เข้ามาแทนที่มนุษย์ เพื่อช่วยเพิ่มประสิทธิภาพในการผลิตและได้รับความนิยมอย่างมากเพราะสามารถนำมาประยุกต์ใช้งานได้อย่างหลากหลาย อาทิ อุตสาหกรรมประกอบรถยนต์ อุสาหกรรมประกอบเครื่องจักร อุตสาหกรรมเคมีภัณฑ์ อุตสาหกรรมบรรจุภัณฑ์ อีกทั้งยังเป็นเทคโนโลยีที่สอดคล้องกับสาขาวิชาที่คณะผู้จัดทำกำลังศึกษาอยู่ คณะผู้จัดทำได้เล็งเห็นความสำคัญและประโยชน์ของแขนกล จึงได้จัดทำแขนกลจำลองเพื่อศึกษาการออกแบบ การควบคุมแขนกลให้ทำงานได้ตามความต้องการและเพื่อสะสมประสบการณ์ด้านวิศวกรรม

แขนกลที่ได้จัดทำขึ้น เป็นแขนกลเพื่อใช้ในงานจับวัตถุที่มีสีขาว โดยอัตโนมัติ ซึ่งแขนกลจะเชื่อมต่อกับคอมพิวเตอร์ เมื่อต้องการจับวัตถุให้ผู้ใช้สามารถสั่งงานผ่านคอมพิวเตอร์ โดยโปรแกรมที่เขียนไว้จะทำการประมวลผล และสั่งงานผ่านไมโครคอนโทรลเลอร์ (Microcontroller) ทำให้แขนกลรู้ตำแน่งของวัตถุสีที่ต้องการและสามารถจับวัตถุนั้นได้

1.2 วัตถุประสงค์ในการทำปริญญานิพนธ์

1. ศึกษาการออกแบบชิ้นงาน และการเลือกใช้วัสดุในการจัดทำแขนกล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2. ศึกษาและประยุกต์ใช้ความรู้ในการเขียนโปรแกรมและการใช้งานไมโครคอน โทรลเลอร์

3. ศึกษาระบบควบคุมของแขนกล

4. ศึกษาการประมวลผลภาพ (Image Processing)

1.3 ขั้นตอนการศึกษาและจัดทำโครงงาน

ขั้นตอนการศึกษาและจัดทำโครงงาน เริ่มจากการศึกษาองค์ความรู้และรูปแบบการทำงานของแขนกล จากนั้นออกแบบแขนกลแบบข้อต่อหมุน 4 ข้อต่อโดยเลือกใช้แผ่นอคิลิกในการผลิตโครงสร้าง และใช้เซอร์โวมอเตอร์เป็นตัวขับเคลื่อนในแต่ละข้อต่อ จากนั้นศึกษาการใช้งานไมโครคอนโทรลเลอร์ซึ่งในโครงงานนี้ได้เลือกใช้ ดีเอสปิค (dsPic) ขนาด 16 บิต เพราะถูกออกแบบมาให้สามารถใช้ในงานประมวลผลสัญญาณดิจิตอลสำหรับระบบควบคุมอัตโนมัติที่มีความสามารถสูงและรวดเร็ว ซึ่งหน้าที่หลักของไมโคร คอนโทรลเลอร์คือใช้ในการควบคุมเซอร์โวมอเตอร์ให้หมุนไปตามตำแหน่งที่ต้องการโดยใช้คอมพิวเตอร์ในการประมวลสัญญาณภาพเพื่อให้ทราบตำแหน่งที่แน่นอนของวัตถุ

ในโครงงานนี้เลือกพัฒนาโปรแกรมคอมพิวเตอร์ด้วยภาษาซีชาร์ป (C#) เพื่อใช้ในการประมวลผลภาพ โดยกระบวนการทำงานของระบบควบคุมแขนกลเริ่มจากการถ่ายภาพและส่งข้อมูลของภาพมายังคอมพิวเตอร์ซึ่งคอมพิวเตอร์จะทำการตรวจจับตำแหน่งของวัตถุและส่งค่าตำแหน่งที่ได้ให้กับไมโครคอน โทรลเลอร์เพื่อใช้ในการหาตำแหน่งของมุมแต่ละมุมที่จะเคลื่อนที่ไปของแต่ละแกนและให้จ่ายสัญญาณพัลซ์ไปควบคุมเซอร์โวให้หมุนไปตามตำแหน่งที่ต้องการ

1.4 รายละเอียดของปริญญานิพนธ์

เนื้อหาที่จะกล่าวในปริญญานิพนธ์ฉบับนี้ประกอบด้วย

บทที่ 1 บทนำ กล่าวถึงวัตถุประสงค์ หลักการใหม่ ขั้นตอนการศึกษา และการจัดทำโครงงานพร้อมทั้งรายละเอียดของปริญญานิพนธ์ในแต่ละบท

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 2 ทฤษฎีและความรู้ที่เกี่ยวข้องกล่าวถึงหลักการและทฤษฎีที่เกี่ยวข้องกับการออกแบบแขนกลการประมลผลภาพ ทฤษฎีระบบควบคุมหลักการทำงานของเซอร์โวมอเตอร์ และวงจรอิเล็กทรอนิกสที่เกี่ยวข้องรวมทั้งการนำเอาความรู้ไปประยุกต์ใช้ในการจัดทำโครงงาน

บทที่ 3 หลักการออกแบบ นำเสนอแผนผังการทำงานของระบบแขนกล หลักการออกแบบแขนกล การออกแบบโปรแกรมควบคุมแขนกล การออกวงจรควบคุมมอเตอร์ หลีกการของสมการแขนกล

บทที่ 4 การทดลอง แสดงผลการทดสอบมอเตอร์ ทดสอบความแม่นยำของแขนกล ทดสอบความแม่นยำของการประมวลผลภาพ ทดสอบว่าแขนกลสามารถจับวัตถุได้หรือไม

บทที่ 5 บทวิจารณ์และสรุป สรุปผลการดำเนินงาน ปัญหาที่เกิดขึ้น และแนวทางการปรับปรุงและพัฒนาโครงงาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 2

ทฤษฎีและความรู้ที่เกี่ยวข้อง

การจัดทำโครงงานต่างๆนั้น ขั้นแรกในการเริ่มการทำงาน คือการศึกษาหาความรู้และทฤษฎีต่างๆ ที่เกี่ยวข้องกับโครงงาน เพื่อที่สามารถจัดทำโครงงานได้อย่างมีประสิทธิภาพ ด้วยเหตุนี้เองการทำโครงการแขนกลอัจฉริยะ จึงต้องมีการศึกษาหาความรู้และทฤษฎีที่เกี่ยวข้องดังต่อไปนี้

2.1 ทฤษฎีพื้นฐานของการวิเคราะห์หุ่นยนต์

หุ่นยนต์ที่มีในปัจจุบันนั้นมีหลากหลายและมีการเพิ่มขึ้นอย่างมาก เนื่องจากมีความต้องการในท้องตลาดมากขึ้นอย่างต่อเนื่อง เพราะฉะนั้นความรู้เบื้องต้นเกี่ยวกับหุ่นยนต์จึงมีความสำคัญเพิ่มมากขึ้นเช่นกัน พื้นฐานความรู้ที่จำเป็นต้องทราบเพื่อสามารถใช้งานหุ่นยนต์ได้อย่างถูกต้องและมีประสิทธิภาพดังนี้

2.1.1 การแบ่งแยกประเภทหุ่นยนต์

การแบ่งยากประเภทของหุ่นยนต์อาจจำแนกกว้างๆ ได้ 2 ประเภท โดยอาศัยเทคโนโลยีการขับเคลื่อน หรือ รูปทรงของขอบเขตการทำงาน ดังนี้

1. เทคโนโลยีการขับเคลื่อน

พิจารณาจากต้นกำลังที่ใช้ขับข้อของหุ่นยนต์ที่นิยมมี 2 แบบคือ การขับเคลื่อนด้วยไฟฟ้าและขับเคลื่อนด้วยไฮดรอลิค (H1ydrolic) แขนกลส่วยใหญ่ในปัจจุบันเป็นแบบขับเคลื่อนด้วยไฟฟ้าโดยใช้ดีซีเซอร์โวมอเตอร์ (DC Servo Motor) หรือ สเต็บปิ้งมอเตอร์ (Stepping Motor) แต่ในการเคลื่อนย้ายวัตถุที่ต้องการความเร็วสูง มักนิยมใช้แบบขับเคลื่อนด้วยไฮดรอลิค

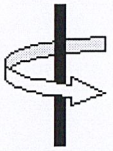
2. รูปทรงของขอบเขตการทำงาน

ขอบเขตการทำงานสุทธิหมายถึง ขอบเขตในปริภูมิสามมิติที่ข้อมือของแขนกลสามารถเคลื่อนที่ไปยังตำแหน่งนั้นได้ เราจะเรียกแกนของ 3 ข้อต่อแรกของแขนกลว่า แกนหลัก ซึ่ง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สามารถพิจารณาตำแหน่งขอบเขตการทำงานได้จาก ลำดับชนิดของข้อต่อที่ใช้ใน 3 แกนแรก ข้อต่อมีหลายแบบ แต่ที่เป็นพื้นฐานและนิยมใช้มี 2 แบบ ดังตารางที่ 2.1

ตารางที่ 2.1 ชนิดของข้อต่อหุ่นยนต์

ชนิด	สัญลักษณ์	หมายเหตุ
Revolute (R)		เป็นการหมุนรอบแกน (Rotary)
Prismatic (P)		การเคลื่อนที่เชิงเส้น (Linear motion)

การรวมกันของข้อต่อหมุนและข้อต่อเลื่อนใน 3 แกนหลัก ก่อให้เกิดรูปทรงการเคลื่อนที่หลายรูปแบบ ดังตัวอย่างในตารางที่ 2.2 ซึ่งแสดงเฉพาะรูปแบบที่นิยมใช้ในการวิเคราะห์การเคลื่อนที่ของแขนกล ข้อต่อหมุน จะวิเคราะห์ยากกว่าแบบข้อต่อเลื่อน ดังนั้นยิ่งมีข้อต่อหมุนมาก แขนกลก็จะยิ่งซับซ้อนขึ้น

ตารางที่ 2.2 ขอบเขตการทำงานของหุ่นยนต์ที่ขึ้นอยู่กับแกนหลัก

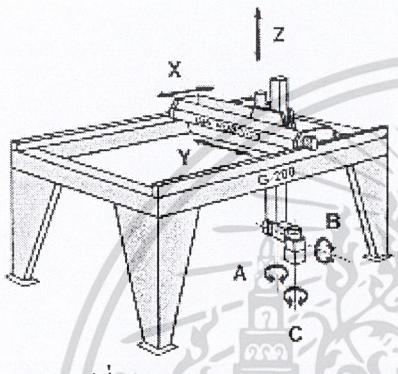
ชนิดของหุ่นยนต์	แกนที่ 1 (เอว)	แกนที่ 1 (เอว)	แกนที่ 3 (ข้อศอก)
Cartesian (gantry)	P	P	P
Cylindrical	R	P	P
Spherical (Polar)	R	R	P
SCARA	R	P	R
Articulated	R	R	R
R = Revolute, P = Prismatic			

โดย P = (การเลื่อน), R = Revolute (การหมุน)

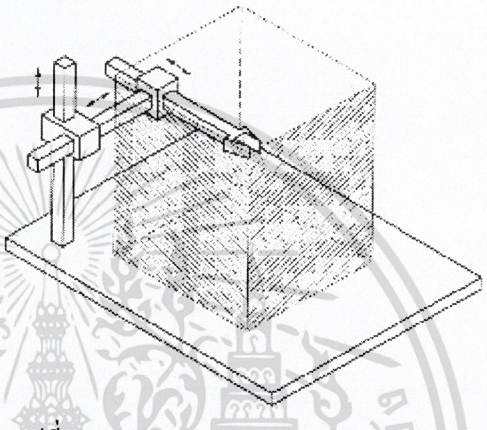
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.1.2 ชนิดการทำงานของหุ่นยนต์ แบ่งตามลักษณะการทำงานของหุ่นยนต์

1. หุ่นยนต์พิกัดคาร์ทีเซียน หรือหุ่นยนต์พิกัดฉาก (Cartesian – coordinate Robot, Rectangular – coordinate Robot) สัญลักษณ์ PPP ดังรูปที่ 2.1a และ 2.1b โดยข้อมือจะเลื่อน ขึ้น – ลง, เข้า – ออก และ เดินหน้า – ถอยหลัง ทำให้พื้นที่การทำงานมีลักษณะเป็นกล่องสี่เหลี่ยมมุมฉาก



รูปที่ 2.1a Cartesian Robot



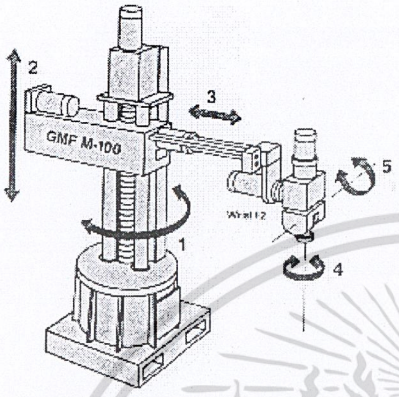
รูปที่ 2.1b Work envelope of Cartesian Robot

ข้อดี	ข้อเสีย
• เคลื่อนที่เป็นแนวเส้นตรงทั้ง 3 มิติ • การเคลื่อนที่สามารถทำความเข้าใจง่าย • มีส่วนประกอบง่าย ๆ • โครงสร้างแข็งแรงตลอดการเคลื่อนที่	• ต้องการพื้นที่ติดตั้งมาก • บริเวณที่หุ่นยนต์เข้าไปทำงานได้ จะเล็กกว่าขนาดของตัวหุ่นยนต์ • ไม่สามารถเข้าถึงวัตถุจากทิศทางข้างใต้ได้ • แกนแบบเชิงเส้นจะ Seal เพื่อป้องกันฝุ่นและของเหลวได้ยาก

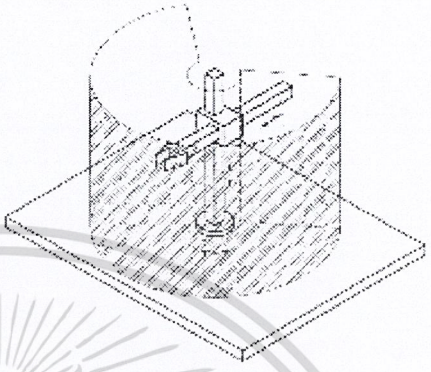
- การประยุกต์ใช้งาน
  เนื่องจากโครงสร้างมีความแข็งแรงตลอดแนวการเคลื่อนที่ ดังนั้นจึงเหมาะ กับงานเคลื่อนย้ายของหนัก ๆ หรือเรียกว่างาน Pick-and-Place เช่นใช้โหลด ชิ้นงานเข้าเครื่องจักร (Machine loading) ,ใช้จัดเก็บชิ้นงาน (Stacking)นอกจากนี้ยังสามารถใช้ในงานประกอบ (Assembly) ที่ไม่ต้องการเข้าถึงในลักษณะที่มีมุมหมุน เช่น ประกอบอุปกรณ์อิเล็กทรอนิกส์ และงาน Test ต่าง ๆ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2. หุ่นยนต์พิกัดทรงกระบอก (Cylindrical – coordinate Robot) สัญลักษณ์ RPP ดังรูปที่ 2.2a และ2.2b สามารถเคลื่อนที่ขึ้นลงข้างบนได้ตามแกนตั้งที่เป็นหลักสามารถเคลื่อนที่เข้าออกตามแนวรัศมี และแขนหมุนรอบแกนตั้งฉากกับฐานได้ พื้นที่การทำงานจึงเป็นแบบทรงกระบอก



รูปที่ 2.2a Cylindrical Robot



รูปที่ 2.2b Work envelope of Cylindrical Robot

ข้อดี

- มีส่วนประกอบไม่ซับซ้อน
- การเคลื่อนที่สามารถเข้าใจได้ง่าย
- สามารถเข้าถึงเครื่องจักรที่มีการเปิด-ปิด หรือเข้าไปในบริเวณที่เป็นช่องหรือโพรงได้ง่าย (Loading) เช่น การโหลดชิ้นงานเข้าเครื่อง CNC

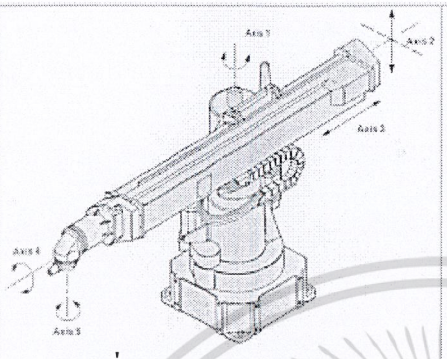
ข้อเสีย

- มีพื้นที่ทำงานจำกัด
- แกนที่เป็นเชิงเส้นมีความยุ่งยากในการ seal เพื่อป้องกันฝุ่นและของเหลว

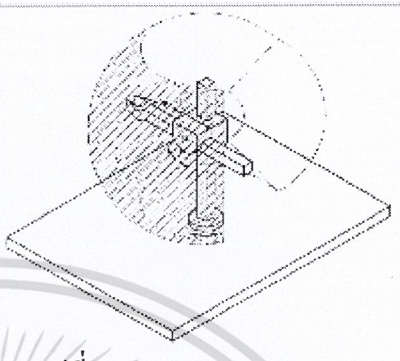
- **การประยุกต์ใช้งาน** โดยทั่วไปจะใช้ในการหยิบยกชิ้นงาน (Pick-and-Place) หรือป้อนชิ้นงานเข้าเครื่องจักร เพราะสามารถเคลื่อนที่เข้าออกบริเวณที่เป็นช่องโพรงเล็ก ๆ ได้สะดวก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3. หุ่นยนต์พิกัดทรงกลม (Spherical – Coordinate Robot) สัญลักษณ์ RRP แสดงดังรูปที่ 2.3a และ 2.3b ลักษณะการเคลื่อนที่ของแขนจะสามารถยกขึ้นลงได้ในแนวดิ่งโดยยกทำมุมกับฐาน แขนสามารถหมุนได้รอบแกนแนวตั้งของฐาน พื้นที่การทำงานเป็นแบบทรงกลม



รูปที่ 2.3a Spherical Robot



รูปที่ 2.3b Work envelope of Spherical Robot

ข้อดี

- มีปริมาตรการทำงานมากขึ้น เนื่องจากการหมุนของแกนที่ 2 (ไหล่)
- สามารถที่จะก้มลงมาจับชิ้นงานบนพื้นได้สะดวก

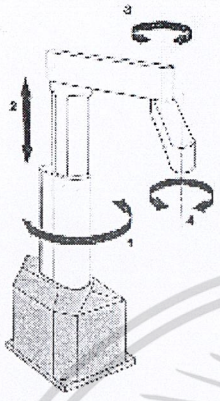
ข้อเสีย

- มีระบบพิกัด (Coordinate) และส่วนประกอบ ที่ซับซ้อน
- การเคลื่อนที่และระบบควบคุมมีความซับซ้อนขึ้น

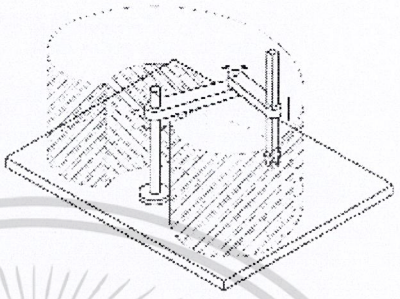
- **การประยุกต์ใช้งาน**
  ใช้ในงานที่มีการเคลื่อนที่ในแนวตั้ง (Vertical) เพียงเล็กน้อย เช่น การโหลดชิ้นงานเข้าออกจากเครื่องปั๊ม (Press) หรืออาจจะใช้งานเชื่อมจุด (Spot Welding)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4. หุ่นยนต์ SCARA (Selective Compliance Assembly Robot Arm) ดังรูปที่ 2.4a และ 2.4b มีลักษณะคล้ายหุ่นยนต์พิกัดทรงกลม มีสัญลักษณ์ RRP แต่แกนทั้ง 3 จะอยู่ในแนวตั้งโดยข้อต่อที่ 2 ทำให้แขนหมุนรอบแกนตั้งในแนวนอน เหมือนการหมุนของข้อแรก ภาพตัดขวางในแนวนอนของพื้นที่การทำงานค่อนข้างซับซ้อนขึ้นอยู่กับข้อจำกัดในการเคลื่อนที่ของสองแกนแรก



รูปที่ 2.4a Scara Robot



รูปที่ 2.4b Work envelope of Scara Robot

ข้อดี	**ข้อเสีย**
• สามารถเคลื่อนที่ในแนวระนาบ และขึ้นลงได้รวดเร็ว • มีความแม่นยำสูง	• มีพื้นที่ทำงานจำกัด • ไม่สามารถหมุน (rotation) ในลักษณะมุมต่างๆได้ • สามารถยกน้ำหนัก (Payload) ได้ไม่มากนัก

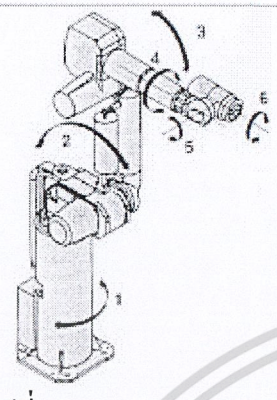
การประยุกต์ใช้งาน

เนื่องจากการเคลื่อนที่ในแนวระนาบและขึ้นลงได้รวดเร็วจึงเหมาะกับ งานประกอบ ชิ้นส่วนทางอิเล็กทรอนิกส์ซึ่ง ต้องการความรวดเร็วและการเคลื่อนที่ก็ไม่ต้องการ การหมุนมากนัก แต่จะไม่เหมาะกับงานประกอบชิ้นส่วนทางกล (Mechanical part)ซึ่งส่วนใหญ่การประกอบจะอาศัยการหมุน(rotation)ในลักษณะมุมต่างๆ นอกจากนี้ Scara robot ยังเหมาะ กับงานตรวจสอบ (Inspection) งานบรรจุภัณฑ์ (Packaging)

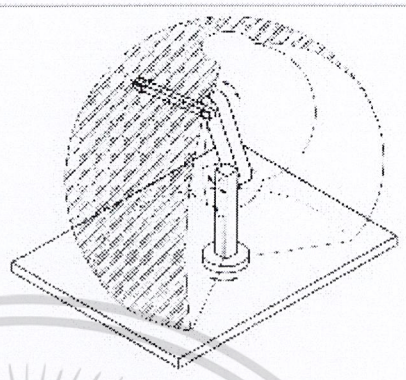
5. หุ่นยนต์ข้อต่อหมุน (Articulate – coordinate Robot หรือ Revolute Robot) ดังรูปที่ 2.5a และ 2.5b สัญลักษณ์ RRR ทั้ง 3 ข้อเป็นข้อต่อหมุน หุ่นยนต์แบบนี้มีลักษณะใกล้เคียงกับแขนมนุษย์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

มากที่สุดมีข้อหมุนต่างๆ เหมือนกัน ดังนั้นพื้นที่การทำงานจึงสามารถที่จะทำงานได้ในทุกตำแหน่งในระยะความยาวของแขน



รูปที่ 2.5a Articulated Robot



รูปที่ 2.5b Work envelope of Articulated Robot

ข้อดี

- เนื่องจากทุกแกนจะเคลื่อนที่ในลักษณะของการหมุนทำให้มีความ ยืดหยุ่นสูงในการเข้าไปยังจุด ต่าง ๆ
- บริเวณข้อต่อ (Joint) สามารถ Seal เพื่อป้องกันฝุ่น ความชื้น หรือน้ำ ได้ง่าย
- มีพื้นที่การทำงานมาก

ข้อเสีย

- มีระบบพิกัด (Coordinate) ที่ซับซ้อน
- การเคลื่อนที่และระบบควบคุมทำความเข้าใจได้ยากขึ้น
- ควบคุมให้เคลื่อนที่ในแนวเส้นตรง (Linear) ได้ยาก
- โครงสร้างไม่มั่นคงตลอดช่วงการเคลื่อนที่ เพราะบริเวณขอบ Work envelope ปลายแขนจะมีการสั่น ทำให้ความแม่ยำลดลง

การประยุกต์ใช้งาน

หุ่นยนต์ชนิดนี้สามารถใช้งานได้กว้างขวางเพราะสามารถเข้าถึงตำแหน่งต่าง ๆ ได้ดี เช่นงานเชื่อม Spot Welding, Path Welding ,งานยกของ , งานตัด ,งานทากาว ,งานที่มีการเคลื่อนที่ยาก ๆ เช่น งานพ่นสี งาน sealing ฯลฯ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ข้อดี – ข้อเสีย ของหุ่นยนต์แต่ละชนิดนี้แตกต่างกันออกไป เพราะลักษณะทางกายภาพแตกต่างกันแต่ถ้ามองในแง่ของการทำงานที่เป็นแบบซ้ำๆ ที่เดิมตลอด หุ่นยนต์พิกัดคาร์ทีเซียนจะสามารถทำงานได้ดีกว่า คือสามารถเคลื่อนที่ไปหาเป้าหมายโดยมีความผิดพลาดน้อยที่สุด แต่ถ้ามองในแง่การเข้าถึงวัตถุชนิดพิกัดทรงกลมและข้อต่อหมุนจะสามารถเข้าถึงวัตถุได้มากกว่า

2.1.3 ลักษณะเฉพาะของหุ่นยนต์

นอกจากเกณฑ์ต่างๆ ที่ใช้การแบ่งประเภทของหุ่นยนต์ดังที่กล่าวมาแล้ว ยังมีลักษณะบางอย่างที่ช่วยผู้ใช้ในการเลือกหุ่นยนต์ที่ต้องการ ลักษณะต่างๆ เหล่านี้แสดงดังตารางที่ 2.3

ตารางที่ 2.3 คุณลักษณะของหุ่นยนต์

คุณลักษณะ (Characteristics)	หน่วย (Units)
จำนวนแกน	-
ความสามารถในการยกน้ำหนัก	kg
ความเร็วสูงสุด, เวลาครบรอบ	mm / sec
ระยะเอื้อมถึงและสโตรก	mm
การหมุนของเครื่องมือ	deg
การซ้ำตำแหน่งเดิม	mm
ความละเอียดและความแม่นยำ	mm

2.2 หลักการทำงานของเซอร์โวมอเตอร์

เซอร์โวมอเตอร์ คือ มอร์เตอร์ไฟฟ้ากระแสตรง DC MOTOR ที่ถูกประกอบร่วมด้วยชุดเกียร์ และส่วนควบคุมต่างๆ ไว้ในโมดูลเดียวกัน โดยจะมีสัญญาณใช้งาน 1 เส้น และอีก2 เส้น เป็น VCC และ GND เท่านั้น ซึ่งสามารถควบคุมให้ตัว SERVO MOTOR หมุนซ้าย หรือ ขวาได้ +90 องศา - 90 องศา (180 องศา) โดยสามารถสั่งงานในการหมุนให้หมุนไปได้ตามองศาต่างๆ ที่ต้องการ ได้ด้วยตัวของ SERVO MOTOR เอง เช่น ต้องการหมุน 1 องศา หรือ 15 องศา ก็ได้ ไม่ต้องมีส่วนควบคุม หรือ SENSOR ใดๆ กลับมาตรวจสอบอีกทำให้ง่าย และสะดวกในการในการนำไปประยุกต์ใช้งานต่างๆได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

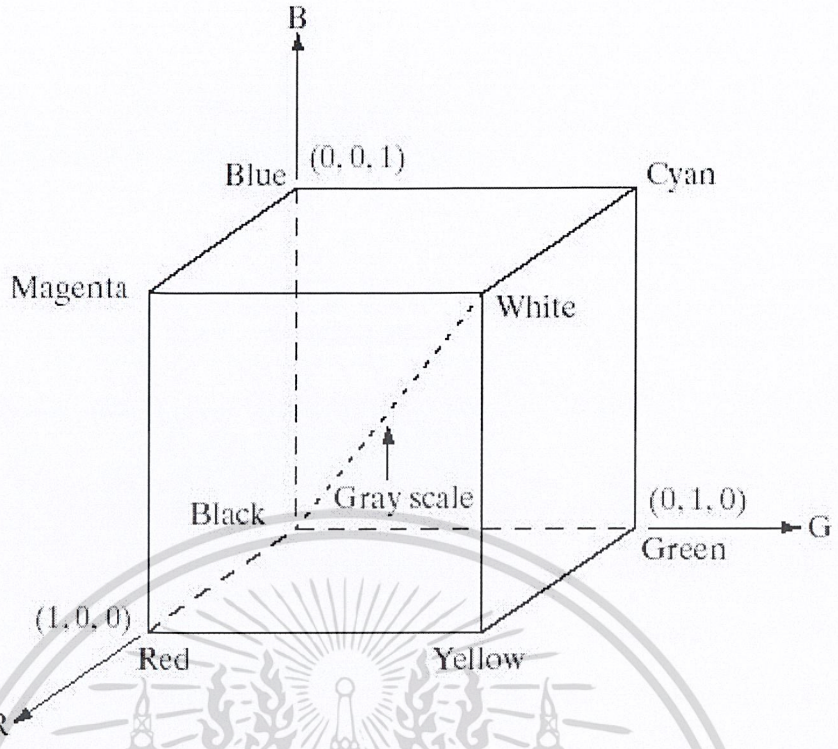
2.3 ทฤษฎีการประมวลผลภาพ

ความรู้และความเข้าใจเกี่ยวกับการประมวลผลภาพ นับเป็นส่วนสำคัญที่จำเป็นจะต้องศึกษาเพื่อให้สามารถประยุกต์ใช้ในการระบุตำแหน่งของวัตถุที่อยู่บนสนามได้ ดังนั้นต้องทำความเข้าใจเกี่ยวกับหลักการของการประมวลผลภาพเบื้องต้นซึ่งประกอบไปด้วยหัวข้อต่างๆ ดังนี้

2.3.1 การกำหนดสี

การกำหนดสีใช้หลักการดังนี้ ในคอมพิวเตอร์สีทุกสีจะประกอบด้วย 3 สีพื้นฐาน คือ สีแดง สีเขียวและน้ำเงิน (RGB) ซึ่งสามารถที่จะสังเคราะห์สีอื่นๆ ได้เกือบทุกสีโดยการผสมสีพื้นฐานทั้งสามนี้ในสัดส่วนต่างๆ กัน การผสมสีตั้งอยู่บนพื้นฐานของความเข้มข้นขององค์ประกอบของอาร์จีบีซึ่งเรียกว่าอาร์จีบีโมเดล (RGB Model) โดยแต่ละสีถูกนำเสนอโดยสัดส่วนของสีแต่ละสี ซึ่งปริมาณของสีพื้นฐานทั้งสามในแต่ละสีนิยมนำเสนอด้วยค่าขนาด 1 ไบต์หรือเท่ากับ 8 บิต โดยค่าที่เล็กที่สุดคือค่า 0 สอดคล้องกับการไม่มีสีพื้นฐานสีนั้นเลย และค่าที่ใหญ่ที่สุดคือค่า 255 บ่งบอกถึงความเข้มข้นสูงสุดของสีพื้นฐานสีนั้นอาทิเช่น สัดส่วน (0,0,0) สอดคล้องกับสีดำ เพราะสีพื้นฐานทั้งสามสีไม่มีความเข้มข้นเลยในขณะที่สัดส่วน (255,255,255) สอดคล้องกับสีขาว ส่วนสีอื่นๆ ซึ่งเกิดจากการรวมกันนั้นมีได้มากมาย เช่น (255,0,0) จะเป็นสีแดงบริสุทธิ์ (0,255,255) เป็นสีน้ำเงินเข้ม และ (0,128,128) เป็นสีน้ำเงินอ่อนขั้น ดังนั้นการรวมกันที่เป็นไปได้ของสีพื้นฐานในกรณีนี้จะมีขนาด 24 บิต หรือมีค่าได้ 256 x 256 x 256 ซึ่งเท่ากับ 16,777,216 ระดับสีจึงนิยมจัดเก็บค่าสีในรูปแบบตัวแปรลองอินติเจอร์ (Long Integer) กระบวนการการสร้างสีซึ่งมีสีพื้นฐาน 3 สีนี้ ตั้งอยู่บนพื้นฐานของกล่องสีอาร์จีบีดังแสดงในรูปที่ 2.6 โดยมุมของกล่องสีนี้จะสอดคล้องกับสีต่างๆ ดังรูป ปริมาณของสีพื้นฐานทั้ง 3 สีในสีใดๆตะมีค่าตามแกน x y และ z ตามลำดับ สีที่เป็นส่วนกลับ (Complementary Colors) สามารถคำนวณได้อย่างง่ายๆ โดยการลบค่าของสีจาก 255 ตัวอย่างเช่นสี (0,0,255) เป็นสีน้ำเงินบริสุทธิ์ เป็นสีส่วนกลับของสีเหลืองบริสุทธิ์ที่สอดคล้องกับค่า (255,255,0) ซึ่งทั้งสองสีดังกล่าวอยู่ที่มุมตรงข้ามกันของกล่องสี ส่วนมุมอื่นๆ ก็เช่นเดียวกัน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.6 กล่องสีอาร์จีบี

องค์ประกอบของสี ที่มุมของกล่องสีจะมีวามเข้มข้นสูงที่สุด หรือไม่มีความเข้มข้นเลยเท่านั้น เมื่อเคลื่อนที่จากมุมหนึ่งไปยังอีกมุมหนึ่งตามขอบที่เหมือนกัน จะมีเพียงแค่องค์ประกอบเดียวเท่านั้นที่เปลี่ยนแปลง ยกตัวอย่างเช่นเมื่อเคลื่อนที่จากมุมสีเขียวไปยังมุมสีเหลือง องค์ประกอบของ สีแดงจะเปลี่ยนแปลงจาก 0 ถึง 255 โดยเมื่อเคลื่อนที่ระหว่างมุมของสีคู่นี้ จะได้ทุกโทนสี จากสีเขียวไปสีเหลือง

2.3.2 องค์ประกอบของสี

ในการประมวลผลภาพบางประเภทอาจต้องการอ่านค่าขององค์ประกอบสีของแต่ละพิกเซลแล้วแยกองค์ประกอบของ อาร์จีบี เพื่อนำไปใช้งานแยกกัน ดังนั้นจึงจำเป็นต้องมีฟังก์ชั่นซึ่งทำหน้าที่แยกองค์ประกอบของสีทั้งสามซึ่งจะถูกเก็บไว้ในตัวแปรลองอินติเจอร์ ดังที่กล่าวไว้ในหัวข้อ 2.3.1 ในตัวแปรลองอินติเจอร์ดังกล่าวจะประกบอด้วย 4 ไบต์โดยไบต์แรก (The Most Significant Byte) เก็บ 0 ต่อมาเป็นองค์ประกอบของสีน้ำเงิน เขียว และแดงตามลำดับค่าลองอินติเจอร์ซึ่งสอดคล้องกับสัดส่วนอาร์จีบี (64,32,192) มีค่าเป็น 12,591,168 ซึ่งค่านี้ดูไม่เหมือนกับจำนวนเดิมเลยถ้านำเสนอองค์ประกอบของสีด้วยเลขฐานสิบหก (Hexadecimal Format) ซึ่งใน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตัวอย่างนี้มีค่าเป็น $C02040_h$ จะเป็นเลขฐานสิบหกของ 12,591,168 ในเลขฐานสิบ จะเห็นว่าสองหลักสุดท้ายในเลขฐานสิบหกสอดคล้องกับองค์ประกอบสีแดง (40_h = 64) สองหลักถัดมาสอดคล้องกับองค์ประกอบสีเขียว (20_h = 32) และสองหลักที่มีความสำคัญสูงสุดสอดคล้องกับค่าสีน้ำเงิน ($C0_h$ = 192) ด้วยเหตุนี้เลขฐานสิบหกจึงถูกใช้มากในการกำหนดค่าสี

2.3.3 ภาพดิจิตอล

ในการที่จะนำภาพภาพหนึ่งมาประมวลด้วยเครื่องคอมพิวเตอร์นั้น ภาพดังกล่าวจำเป็นจะต้องถูกนำเสนอหรือแทนด้วยตัวเลข แต่ภาพที่ได้มาโดยส่วนมากจะเป็นภาพที่ได้จากตัวรับสัญญาณซึ่งอยู่ในรูปของฟังก์ชั่น $f(x,y)$ ที่ต่อเนื่องในระนาบสองมิติ โดยจะเป็นสัดส่วนกับความสว่างหรือความเข้มของภาพที่ตำแหน่ง (x,y) ซึ่งเรียกว่าระดับสีเทา (Grey Level) สำหรับภาพขาวดำ และจะเป็น สัดส่วนกับระดับค่าอาร์จีบีสำหรับภาพสี

2.3.3.1 การแทนภาพด้วยข้อมูลดิจิตอล

ภาพแบบดิจิตอลเป็นภาพที่ถูกแปลงมาจากอนาลอกเพื่อให้อยู่ในรูปของตัวเลข โดยภาพจะถูกแบ่งพื้นที่เป็นสี่เหลี่ยมเล็กๆ ที่เรียกว่าจุดภาพหรือพิกเซล (Pixel) ซึ่งถูกระบุตำแหน่งของพิกเซลโดยพิกัด (x,y) และในแต่ละพิกเซลจะระบุข้อมูลระดับสีหรือค่า อาร์จีบี สำหรับภาพสี และระดับความเข้มสำหรับระดับสีเทา โดยการแหลงข้อมูลแบบดิจิตอลสามารถทำได้ดังขั้นตอนต่อไปนี้

เมื่อนำภาพแบบอนาล็อกที่ต้องการประมวลผลมาผ่านส่วนที่เรียกว่าดิจิไทเซอร์ (Digitizer) ทำหน้าที่เปลี่ยนสัญญาณอนาล็อกเป็นสัญญาณดิจิตอลอุปกรณ์ประเภทนี้ได้แก่กล้องดิจิไทเซอร์ ซึ่งจะทำให้ภาพที่นำเสนอด้วยฟังก์ชั่นที่ต่อเนื่อง $f(x,y)$ ให้กลายเป็นฟังก์ชั่นที่ไม่ต่อเนื่องทั้งระนาบของภาพซึ่งเรียกว่าการสุ่ม (Sampling) และผ่านการควอนไตซ์ (Quantization) จะได้ข้อมูลที่เป็นดิจิตอลสมมติว่าสัญญาณภาพต่อเนื่อง $f(x,y)$ ได้ถูกแปลงให้อยู่ฟังก์ชั่นที่ไม่ต่อเนื่องซึ่งจะสามารถนำเสนอในรูปเมตริกซ์และสมมติว่ามีขนาด N x N ได้ตามสมการ (2.3)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

$$f(x,y) = \begin{matrix} f(0,0) & f(0,1) & f(0,2) & \cdots & f(0,N-1) \\ f(1,0) & f(1,1) & f(1,2) & \cdots & f(1,N-1) \\ \vdots & \vdots & \vdots & & \vdots \\ f(N-1,0) & f(N-1,1) & f(N-1,2) & \cdots & f(N-1,N-1) \end{matrix} \quad (2.3)$$

โดยทางขวาของฟังก์ชั่น $f(x,y)$ จะเรียกว่า ภาพดิจิตอล และทุกสมาชิกของเมตริกซ์จะเรียกว่า พิกเซล ซึ่งความละเอียดของภาพตามสมการ (2.3) เป็น N x N พิกเซล และในกรณีภาพขาวดำ ขนาดของข้อมูลภาพที่เป็นดิจิตอลจะมีขนาดดังสมการ (2.4)

$$B = M \times N \times N \text{ บิต} \quad (2.4)$$

เมื่อ B คือขนาดของข้อมูลภาพที่เป็นดิจิตอล (บิต)

M คือจำนวนบิตที่ใช้ในการแทนข้อมูลภาพพิกเซล 1 พิกเซล สามารถได้จากสมการที่ (2.5)

$$G = 2^M \quad (2.5)$$

เมื่อ G คือจำนวนระดับสีเทา (Grey Level) ที่ต้องการใช้ในการเก็บภาพ

2.3.3.2 ลักษณะการจัดเก็บข้อมูลภาพดิจิตอล

โดยทั่วไปแล้วข้อมูลภาพสีหรือภาพขาวดำจะมีค่าตั้งแต่สองระดับขึ้นไป โดยเฉพาะในกรณภาพขาวดำ นิยมใช้ระดับความเข้มของจุดภาพระดับของสีเทาเท่ากับ 256 ระดับ ซึ่งทำให้จุดภาพมีค่าอยู่ในช่วง 0–255 โดยใช้พื้นที่เก็บข้อมูล 1 ไบต์ หรือ 8 บิต สำหรับข้อมูล 1 จุดภาพ (M = 8) ในกรณีที่ต้องการภาพที่มีความละเอียดของระดับความเข้มหรือระดับสีจำนวนสูงๆ อาจจะต้องการจำนวนบิตสำหรับเก็บข้อมูลมากกว่า 8 บิต อาจเป็น 16 หรือ 24 บิต โดยระดับความเข้มของจุดภาพเท่ากับ 65,536 และ 16,777,216 ระดับ โดยจะแยกให้ชัดเจนดังนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1. ภาพ 2 ระดับ คือมีเพียงแต่จุดขาวกับจุดดำเท่านั้น โดยแต่ละจุดภาพเป็นข้อมูลขนาด 1 บิต

2. ภาพ 16 ระดับ คือในแต่ละจุดภาพจะมีขนาดของข้อมูล 4 บิต สามารถทำให้แสดงได้ 16 ระดับความเข้มหรือระดับสีขึ้นอยู่กับภาพนั้นเป็นภาพสีหรือภาพขาวดำ

3. ภาพ 256 ระดับ คือในแต่ละจุดภาพจะมีขนาดของข้อมูลขนาด 8 บิต ซึ่งสามารถทำให้แสดงได้ 256 ระดับความเข้มหรือระดับสี ขึ้นอยู่กับภาพนั้นเป็นภาพสีหรือขาวดำ

4. ภาพทรูคัลเลอร์ (True Colors) ในแต่ละจุดภาพจะมีขนาดของข้อมูล 24 บิต สามารถทำให้แสดงภาพสีได้เหมือนจริงที่สุดเพราะสามารถแสดงสีได้ถึง 16,777,216 ระดับสี

2.3.4 การประมวลผลภาพเชิงตัวเลข

โดยทั่วไปวิธีการประมวลผลภาพเชิงตัวเลขโดยคอมพิวเตอร์นั้น อาจแบ่งโดยคร่าวๆ เป็นสองระดับด้วยกัน คือ การประมวลผลภาพในระดับต่ำ (Low Level Processing) และการประมวลผลภาพในระดับสูง (High Level Processing) การประมวลผลภาพในระดับต่ำจะเป็นการประมวลผลเชิงตัวเลขเกือบทั้งหมดเพื่อหาตัวแปรต่างๆ มาอธิบายข้อมูลภาพ มีวัตถุประสงค์เพื่อนำตัวแปรเหล่านั้นไปใช้ในการประมวลผลในระดับสูงต่อไป

ส่วนการประมวลผลภาพในระดับสูง เป็นการนำผลลัพธ์ที่ได้จากการประมวลผลภาพในระดับต่ำมาตีความหรือเพื่อส่งข้อมูลให้คอมพิวเตอร์สามารถรู้และเข้าใจได้ ดังนั้นความแตกต่างที่สำคัญของการประมวลผลภาพระดับต่ำและระดับสูง คือ ข้อมูลที่นำเข้ามาในการประมวลผลโดยการประมวลผลภาพในระดับต่ำจะใช้ความเข้มของจุดโดยตรง ส่วนการประมวลผลภาพในระดับสูงนั้น ข้อมูลที่นำมาประมวลผลจะถูกแสดงในรูปของสัญลักษณ์ ซึ่งจะแสดงถึงสิ่งต่างๆ ที่อยู่ในภาพ เช่น ขนาด หรือรูปร่างของวัตถุในภาพ

2.3.5 ข้อมูลภาพชนิดบิตแมป

รูปแบบของไฟล์ข้อมูลภาพชนิดบิตแมป (Bitmap) เป็นรูปแบบไฟล์มาตรฐานสำหรับภาพกราฟฟิกบนวินโดว์ซึ่งจะใช้ในการตัดต่อ หรือทำสำเนาภาพต่างๆ ลงบนคลิปบอร์ด (Clipboard) เวลาจัดเก็บไฟล์จะมีนามสกุล .BMP

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สำนักหอสมุดกลาง พระจอมเกล้าลาดกระบัง

โครงสร้างของไฟล์ข้อมูลภาพชนิดบิตแมปจะประกอบด้วย 3 ส่วนคือ

1. ข้อมูลที่อยู่บริเวณส่วนหัวของไฟล์ (Header) ซึ่งประกอบด้วยข้อมูลที่บอกรายละเอียดต่างๆ ของภาพ เช่น ความกว้าง ความยาวของภาพ จำนวนสี จำนวนบิต ความละเอียดของภาพ

2. ข้อมูลจานสี (Palette) เป็นค่าแม่สีอาร์จีบี ของภาพ

3. ข้อมูลภาพ (Data) เป็นส่วนเก็บข้อมูลสีของแต่ละพิกเซลของภาพ โดยข้อมูลแรกจะเป็นค่าสีของพิกเซลที่อยู่แถวบนสุดที่ตำแหน่งซ้ายสุดข้อมูลลำดับ ต่อไปจะเรียงจากทางขวาจากแถวบนจนถึงแถวล่างสุด

2.3.6 การสร้างภาพไบนารี

การสร้างภาพไบนารี หมายถึง การแปลงข้อมูลภาพที่มีความเข้มหลายระดับ ให้เป็นภาพที่มีความเข้มเพียง 2 ระดับ คือหนึ่งจุดภาพมีค่าได้สองค่าเท่านั้น โดยเป็นค่า 0 กับค่า 1 ซึ่งค่าที่เป็น 1 จะหมายถึงจุดภาพที่มีสีดำ และค่าที่เป็น 0 จะหมายถึงจุดภาพที่มีสีขาว

การสร้างภาพไบนารีสามารถทำโดยใช้เทคนิคการทำเทรชโฮล (Thresholding Techniques) โดยพิจารณาว่าจุดใดควรเป็นจุดขาวหรือจุดดำ ซึ่งทำได้โดยการเปรียบเทียบระหว่างจุดภาพเริ่มต้นกับค่าคงที่ค่าหนึ่ง เรียกว่าค่าเทรชโฮล (Threshold) เทคนิคนี้ใช้กันมากในกรณีที่ข้อมูลภาพมีลักษณะแตกต่างกันระหว่างวัตถุและพื้นหลัง โดยค่าของจุดภาพใดๆ ที่มีค่าน้อยกว่าค่าเทรลโฮลจะถูกกำหนดค่าเป็น 1 (จุดสีดำ) และถ้าค่าของจุดภาพที่มีค่ามากกว่าค่าเทรชโอลจจะถูกเปลี่ยนให้เป็นค่า 0 (จุดสีขาว) ซึ่งการทำงานสามารถแสดงได้ดังสมการที่ (2.6) และ ตามลำดับ

$$\mathrm{b(x,y)} = \begin{cases} 1, & g(x,y) \le Thr \\ 0, & g(x,y) > Thr \end{cases} \qquad (2.6)$$

เมื่อ $b(x,y)$ คือ ข้อมูลภาพผลลัพธ์ภาพเป็นไบนารี

$g(x,y)$ คือ ข้อมูลภาพอินพุตที่มีระดับความเข้ม 0 ถึง L ระดับ

1 คือ จุดภาพที่เป็นสีดำ

0 คือ จุดภาพที่เป็นสีขาว

L คือ ระดับความเข้มของจุดภาพสูงสุด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้
119187

ในการสร้างภาพไบนารีโดยใช้เทคนิคเทรชโฮล เพื่อให้ผลลัพธ์ที่ได้มีความคมชัดและเหมะสม สิ่งที่สำคัญที่สุด คือ ค่าเทรชโฮล เนื่องจากถ้าเลือกค่าเทรชโฮลที่ไม่เหมาะสม ภาพที่ได้จะไม่คมชัดและรายละเอียดบางส่วนอาจขาดหายไป ดังนั้นปัญหาการหร้างภาพไบนารีด้วยเทคนิคการทำเทรชโฮล คือทำอย่างไรจึงสามารถคำนวณหาค่าเทรชโฮลที่เหมะสมได้ ซึ่งในที่นี้จะขอกล่าวถึงวิธีที่นิยมใช้กันคือการหาค่าเทรชโฮลโดยวิธีการกำหนดล่วงหน้า

การหาค่าเทรชโฮลโดยวิธีการกำหนดล่วงหน้า (Preassigned Threshold Value) นี้เป็นวิธีที่ง่ายที่สุดเป็นการกำหนดค่าเทรชโฮลเองจากผู้ใช้ ซึ่งจะขึ้นอยู่กับประสบการณ์ของผู้ใช้ โดยการเลือกค่าคงที่ค่าหนึ่งซึ่งจะเป็นค่าที่อยู่ระหว่างค่าต่ำสุดและค่าสูงสุดของระดับความเข้มของข้อมูลอินพุต

วิธีหนึ่งสำหรับการหาค่าเทรชโฮลล่วงหน้าคือ การหาค่าเทรชโฮลจากค่ากลาง (Midrange Thredhold Value) การหาวิธีนี้จะพิจารณาจากค่ากลาง โดยอาศัยการคำนวณพื้นฐานทางสถิติในการหาค่ากลางมาประยุกต์ค่าที่ได้ เป็นค่ากึ่งกลางระหว่างค่าระดับวามเข้มสูงสุดกับค่าต่ำสุดของข้อมูลภาพอินพุตสำหรับการหาค่ากึ่งกลางจะแสดงได้ดังสมการที่ (2.7)

$$Thr = \frac{\text{Max}(g(x,y)) + \text{Min}(g(x,y))}{2} \qquad (2.7)$$

เมื่อ Thr คือ ค่าเทรชโฮล

$g(x,y)$ คือ ข้อมูลภาพอินพุตที่มีระดับความเข้ม 0 ถึง L ระดับ

$\text{Max}(g(x,y))$ คือ ค่าสูงสุดเกรย์เสกลของข้อมูลอินพุต

$\text{Min}(g(x,y))$ คือ ค่าต่ำสุดเกรย์สเกลของข้อมูลอินพุต

L คือ ระดับความเข้มของจุดภาพสูงสุด

การหาค่าเทรชโฮลจากค่าเฉลี่ยเลขคณิตหาได้จากสมการที่ (2.8)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

$$Thr = \frac{\sum g(x,y)}{N \times N} \quad (2.8)$$

2.4 การฟิลเตอร์แบบค่ากลาง

ในกรณีนี้พิกเซลอินพุตจะถูกแทนที่ด้วยค่ากลางของพิกเซลที่อยู่ในหน้าต่างรอบๆ พิกเซลนั่นคือ

$$V(m,n) = \text{median}\{y(m-k,n-1),(k,1) \in W\} \quad (2.9)$$

เมื่อ W คือ หน้าต่าง (Window)

อัลกอริทึมสำหรับการฟิลเตอร์ (Filtering) แบบกลางจะทำการเรียงค่าพิกเซลในลักษณะที่เพิ่มขึ้นหรือลดลงแล้วทำการเลือกค่าที่อยู่ตรงกลาง โดยทั่วไปแล้วขนาดของหน้าต่างถูกเลือกให้ N_w มีค่าเป็นเลขคี่ ถ้า N_w เป็นเลขคู่ ค่ามีเดียนจะเป็นค่าเฉลี่ยระหว่างสองค่าตรงกลาง โดยทั่วไปแล้วหน้าต่างจะเป็นแบบขนาด 3x3 5x5 7x7 มีประโยชน์ในการจำกัดเส้นหรือพิกเซลที่แยกออกมา (Isolate) โดยฟิลเตอร์โดยใช้ค่ากลางนั้นจะใช้ได้ดีสำหรับสัญญาณรบกวนแบบไบนารี แต่ใช้ไม่ได้ดีสำหรับสัญญาณรบกวนแบบ เกาซ์เซียน (Gaussian) และจะมีประสิทธิภาพต่ำเมื่อจำนวนของพิกเซลที่ถูกปนเปื้อนด้วยสัญญาณรบกวนในหน้าต่าง มีค่ามากกว่าหรือมีค่าครึ่งหนึ่งของจำนวนพิกเซลในหน้าต่าง

2.5 การวิเคราะห์รูปทรง

การคำนวณหาค่าจากภาพที่ได้ถ่ายมานั้น เพื่อใช้ในการคำนวณนั้นสามารถทำได้หลายวิธีเพื่อบอกคุณลักษณะของภาพในรูปของเลขฐานสอง

2.5.1 หลักการอธิบายพื้นที่

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เป็นวิธีหนึ่งที่สามารถบอกพื้นที่ของกลุ่มภาพ โดยแปลงกลุ่มภาพนั้นเป็นในรูปแบบเลขฐานสองซึ่งวิธีนี้มีความแม่นยำค่อนข้างสูงแต่มีความยากในการคำนวณโดยเฉพาะถ้ารูปมีขนาดใหญ่

ในการคำนวณหาค่าการอธิบายพื้นที่โดยที่ R แทนค่าพื้นที่ของรูปภาพ I(k,j) โดยที่ R เป็นกลุ่มค่าของแต่ละพิกเซล แม้ว่ารูปที่คิดนั้นอาจมีรูหรือช่องว่างในรูปภาพโดยสมมติให้ภาพ I(k,j) นั้นเป็นรูปภาพแบบฐานสองโดยที่ ค่าของภาพที่จับได้ (Foreground) มีค่าเป็น 1 และส่วนที่เหลือ (Background) มีค่าเป็น 0

2.5.2 โมเมนต์

เป็นกระบวนการหาค่าของโมเมนต์ (Moments) โดยสมมติให้ x แทนค่าแถวและ y แทนค่าของหลักของภาพฐานสอง โดยเราสามารถคำนวณหาค่าของโมเมนต์ได้ดังสมการที่ (2.10)

$$M_{kj} = \sum_{(x,y)\in R} x^k y^j; k \geq 0, j \geq 0 \quad (2.10)$$

โมเมนต์ระดับต่ำ (Low – order Moment) เป็นกระบวนการที่นำทฤษฎีของโมเมนต์ มาหาค่าของจุดศูนย์กลางของวัตถุ (x,y) โดยสามารถหาค่าจุดศูนย์กลางของวัตถุได้ดังนี้

$$X_c = \frac{m_{10}}{m_{00}} \quad (2.11)$$

$$Y_c = \frac{m_{01}}{m_{00}} \quad (2.12)$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 3

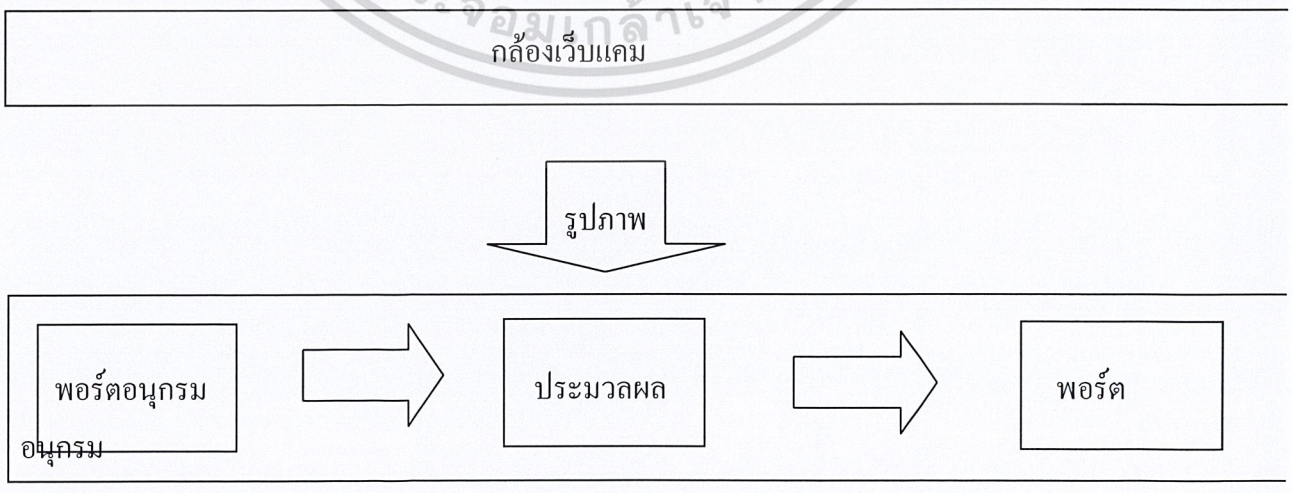
หลักการออกแบบ

การจัดทำโครงงานนั้นจำเป็นต้องมีความเข้าใจในภาพรวมของระบบการทำงานทั้งหมด และหลักการออกแบบ ซึ่งจะช่วยให้การทำงานนั้นง่ายขึ้น และเมื่อเกิดปัญหาขึ้นจะสามารถแก้ไขปัญหาต่างๆ ได้อย่างมีประสิทธิภาพ และรวดเร็ว

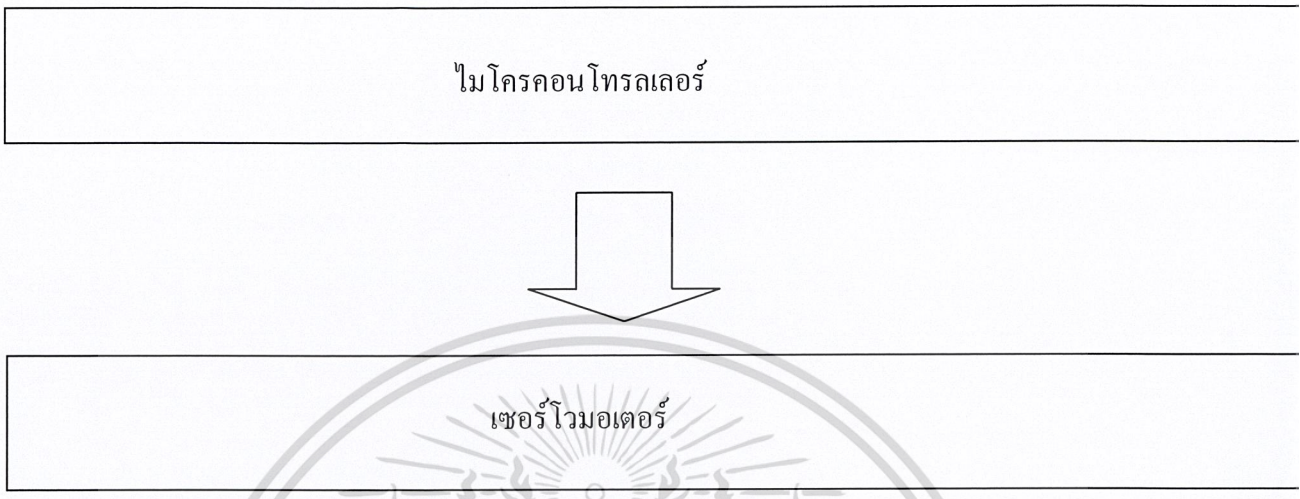
ภาพรวมของระบบการทำงานของโครงงานและหลักการออกแบบสามารถแยกได้ 5 ประเภทดังนี้

3.1 หลักการทำงานของแขนกล

เริ่มต้นจากคอมพิวเตอร์ประมวลสัญญาณภาพโดยการจับวัตถุที่มีสีขาวและหาตำแหน่งของวัตถุที่มีสีขาวได้ จากนั้นส่งค่าที่ได้มายังไมโครคอนโทรลเลอร์โดยการส่งผ่านพอร์ตอนุกรม ซึ่งในที่โครงงานนี้เราใช้ dspic เป็นไมโครคอนโทรลเลอร์ เมื่อไมโครคอนโทรลเลอร์รับค่าตำแหน่งได้จากคอมพิวเตอร์ไมโครคอนโทรลเลอร์จะทำการอินเวอร์ไคนิเมติกส์เพื่อให้มุมต่างๆที่แขนกลต้องการเมื่อได้มุมก็จะสั่งจ่ายสัญญาณพัลซ์ขนาดต่างๆไปให้กับเซอร์โวเพื่อให้เซอร์หมุนไปตามองศาที่ต้องการ ซึ่งหลักการทำงานของแขนกลดังแสดงในรูปที่ 3.1



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.1 แสดงแผนผังการทำงาน

3.2 หลักการออกแบบชิ้นงาน

ในการจัดทำแขนกลซึ่งถือได้ว่าเป็นองค์ประกอบที่มีความสำคัญและต้องมีความแม่นยำ เพราะฉะนั้นจึงจำเป็นต้องมีการออกแบบในคอมพิวเตอร์ เพื่อดูความเหมาะสมและความยากง่ายในการจัดทำแขนกล โดยในโครงงานนี้ได้ออกแบบโดยใช้โปรแกรม Solid Work 2007

3.3 การออกแบบโปรแกรม

โปรแกรมที่ได้จัดทำขึ้นมีส่วนสำคัญที่มีการประมวลผลคือ

1. การประมวลผลภาพ

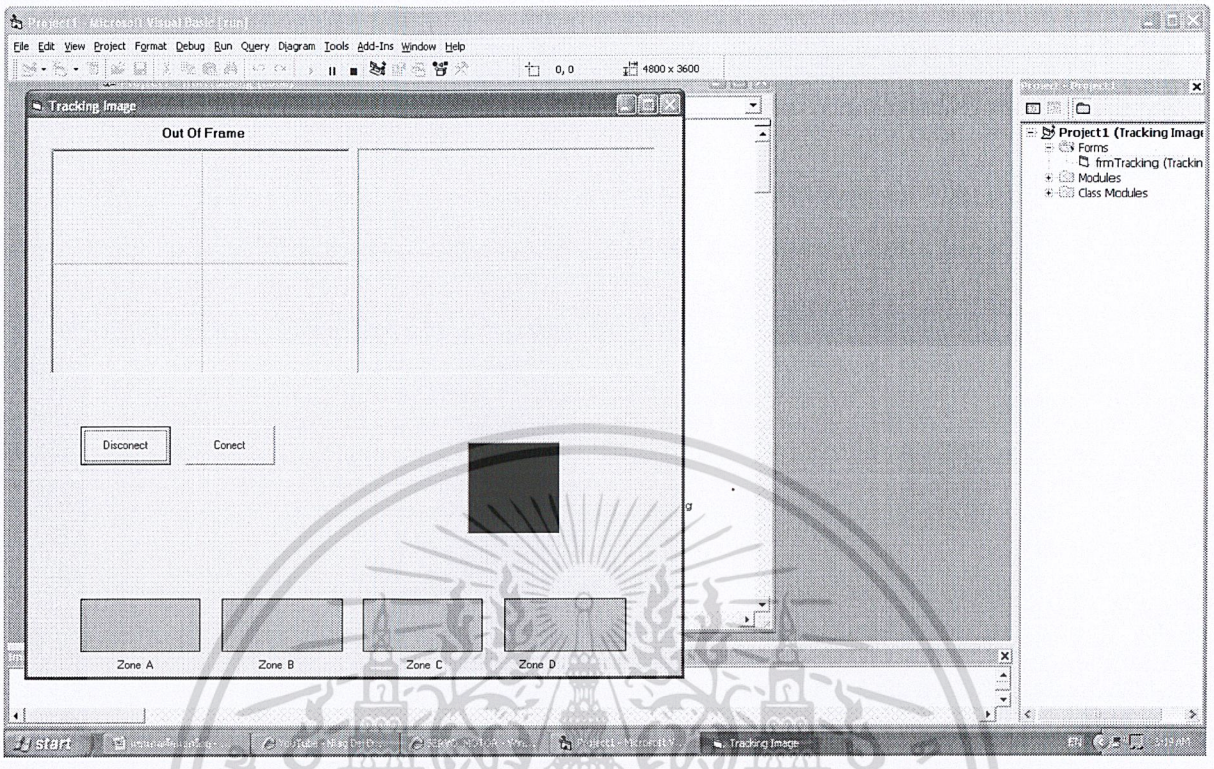
2. คำนวณสมการด้วยสมการอินเวิร์สไคนิเมติกส์

3. ประมวลผลเพื่อคำนวณหาค่าของมุมที่ต้องการ

ส่วนประกอบของโปรแกรมประกอบด้วยหน้าจอแสดงภาพจากกล้องเว็บแคมสองหน้าจอ หน้าจอด้านซ้ายจะเป็นหน้าจอที่แบบเป็นสี่โซน ส่วนหน้าจอด้านขวาจะเป็นหน้าจอแสดงผลปกติ ถัดลงมาจะเป็นด้านขวาจะเป็นปุ่มกดสองปุ่ม เป็นปุ่ม Connect กับปุ่ม DisConnect เป็นใช้ในการติดต่อกับกล้อง และด้านซ้ายจะเป็นหน้าจอที่ใช้โฟกัสวัตถุโดยตรง ด้านล่างสุดเป็นแถบสีที่ใช้ในการแสดงโซนตำแหน่งของวัตถุแบ่งเป็นสี่โซน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ดังแสดงในรูปที่ 3.2

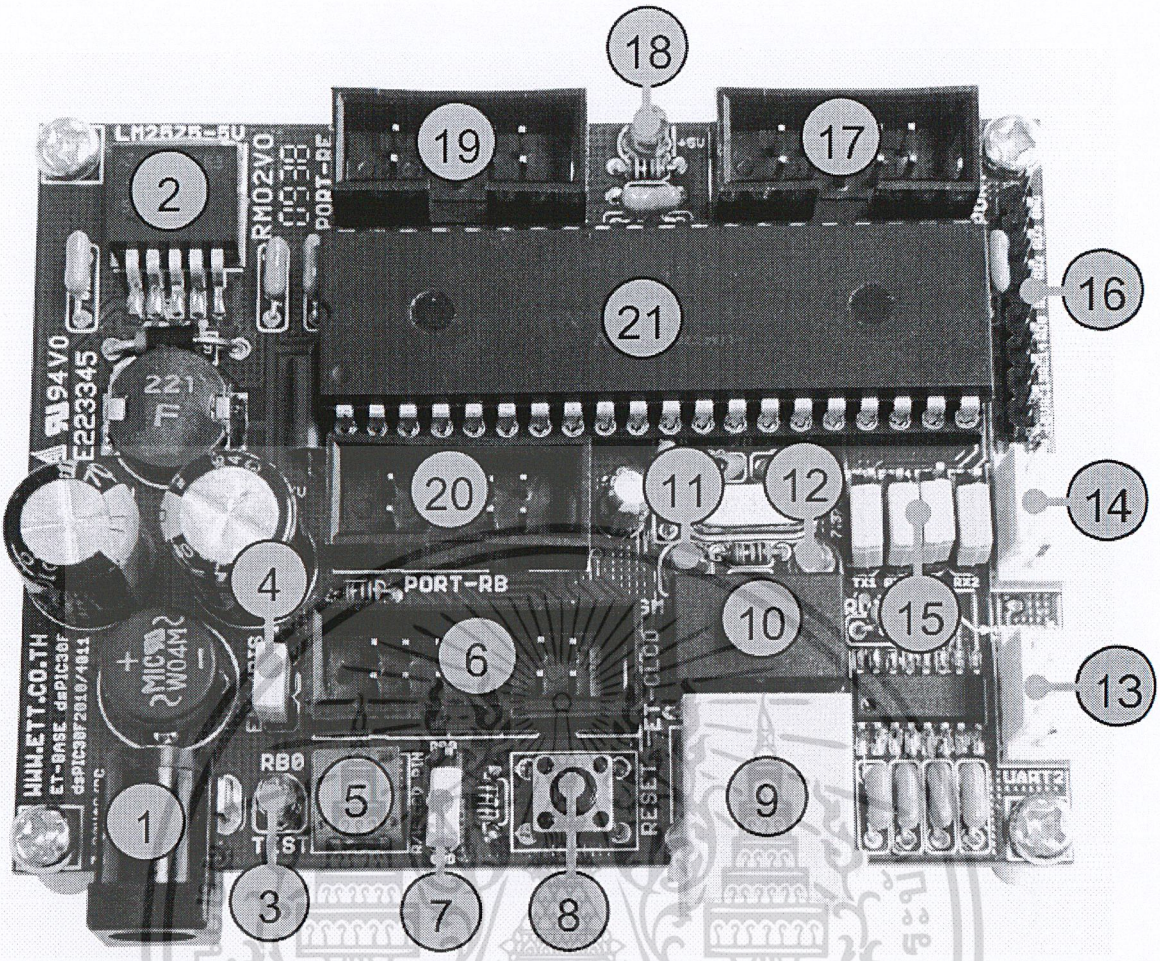


รูปที่ 3.2 แสดงหน้าต่างของโปรแกรม Visualbasic ที่ใช้ในการประมวลสัญญาณภาพ

3.4 การออกแบบวงจรควบคุมการทำงานของแขนกล

ในการจัดทำโครงงานนี้นอกจากจะอาศัยหลักการข้างต้นดังที่กล่าวมาแล้ว ยังต้องอาศัยวงจรควบคุมการทำงานของแขนกลที่มีประสิทธิภาพ จึงจะส่งผลให้แขนกลทำงานได้ตามเป้าหมาย ซึ่งทางคณะผู้จัดทำได้ใช้บอร์ดของอีทีทีรุ่น ET-BASE DSPIC30F4011 ดังแสดงในรูปที่ รูปที่ 3.3

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.3 ET-BASE dsPIC30F4011

3.4.1 คู่มือการใช้งานบอร์ด ET-BASE dsPIC30F4011

หมายเลข 1 คือ ขั้วต่อแหล่งจ่ายไฟเลี้ยงวงจรของบอร์ด ใช้กับแหล่งจ่ายไฟ 7-20VAC/DC

หมายเลข 2 คือ IC Regulate แบบ Switching ขนาด 5V/1A

หมายเลข 3 คือ LED TEST สำหรับทดสอบการทางานของบอร์ด โดยควบคุมจาก RB0

หมายเลข 4 คือ Jumper สำหรับ ตัด ต่อ สัญญาณ RB0 กบ LED TEST

หมายเลข 5 คือ VR ปรับค่า สำหรับใช้ปรับความสว่างของหนาจอแสดงผล LCD

หมายเลข 6 คือ ขั้วต่อ 14PIN IDE สำหรับเชื่อมต่อกับ LCD แบบ Character

หมายเลข 7 คือ Jumper สำหรับเลือกรูปแบบการควบคุมขา R/W ของ LCD โดยถ้าใช้ MCU รุ่น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

28Pin ต้องเลือกไว้ด้าน GND เสมอและไม่สามารถส่งอ่านข้อมูลจาก LCD ได้

หมายเลข 8 คือ สวิตช์ Reset สำหรับ Reset การทำงานของ MCU เมื่ออยู่ในโหมด Run

หมายเลข 9 คือ ขั้วต่อ ICD2 สาหรบใช้เชื่อมต่อกับเครื่องโปรแกรมและดีบักตามมาตรฐาน ICD2

หมายเลข 10 คือ สวิตช์ สำหรับเลือกโหมดการทำงานระหว่าง Run(RUN) และ Program(PGM)

หมายเลข 11 คือ LED สีแดง แสดงสถานะ PGM เมื่อบอร์ดทำงานใน Program Mode

หมายเลข 12 คือ LED สีเขียว แสดงสถานะ RUN เมื่อบอร์ดทำงานใน Run Mode

หมายเลข 13 คือ ขั้วต่อ UART2 ซึ่งมีเฉพาะใน MCU รุ่น 40 Pin (dsPIC30F4011) เท่านั้น โดย เป็นสัญญาณแบบ RS232 โดยใช้ Pin ของ RF4(RX2) และ RF5(TX2) เป็นสัญญาณเชื่อมต่อ

หมายเลข 14 คือขั้วต่อ UART1 โดยเป็นสัญญาณแบบ RS232 มีอยู่ใน MCU ทั้งรุ่น 28 Pin และรุ่น 40 Pin ซึ่งใช้ Pin ของ RC13(TX1),RC14(RX1) เป็นสัญญาณเชื่อมต่อ

หมายเลข 15 คือ Jumper สำหรับเลือกการเชื่อมต่อสัญญาณ RC13,RC14,RF4,RF5 ว่าจะใช้ ขาสัญญาณดังกล่าวทำหน้าที่เป็นขาสัญญาณรับส่งของ RS232 หรอ GPIO สำหรับใช้งานทั่วไป

หมายเลข 16 คือ ขั้วต่อสัญญาณ RC13,RC14,RD0,RD1,RD2 และ RD3 สำหรับใช้งาน โดยถ้า เป็น MCU รุ่น 28Pin จะไม่มีสัญญาณ RD2 และ RD3 ขาสัญญาณดังกล่าวจะปล่อยวางไว

หมายเลข 17 คือ ขั้วต่อสัญญาณ Port-RF ซึ่งถ้าเป็น MCU รุ่น 40 Pin จะมี 7 บิต คือ RF[0..6] แต่ถ้าเป็น MCU รุ่น 28 Pin จะมีเพียง 2 บิต คือ RF[2] และ RF[3] เท่านั้น

หมายเลข 18 คือ LED สำหรับแสดงสถานะ ของแหล่งจ่ายไฟ +5V ของบอร์ด

หมายเลข 19 คือ ขั้วต่อสัญญาณ Port-RE ซึ่งจะมี 7 บิต คือ RE[0..6 และ 8]

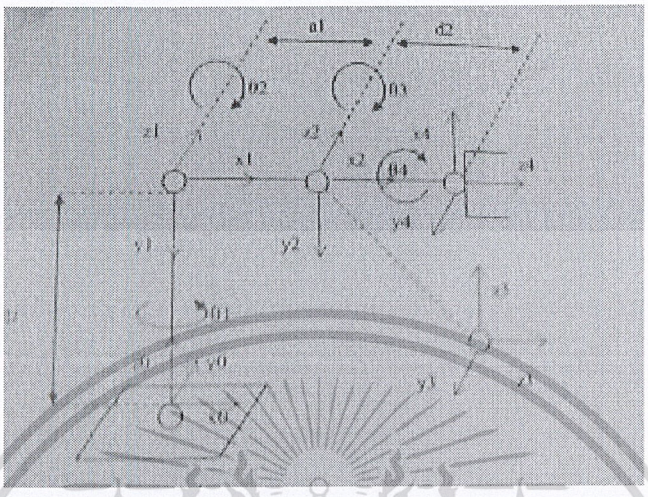
หมายเลข 20 คือ ขั้วต่อสัญญาณ Port-RB ซึ่งถ้าเป็น MCU40Pin จะมี 8 บิต คือ RB[0..7] แต่ถ้า เป็น MCU 28Pin จะมีเพียง 6 บิต คือ RB[0..5] เท่านั้น

หมายเลข 21 คือ MCU ประจำบอร์ด โดยถ้าเป็น รุ่น 28Pin จะใช้เบอร์ dsPIC30F2010 แต่ถ้าเป็น รุ่น 40Pin จะใช้เบอร์ dsPIC30F4011

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.5 สมการแขนกล

ฟอเวิร์สไคนิแตติกส์ (Forward Kinematics) เป็นการแปลงค่ามุมไปเป็นพิกัดโดยจะอ้างอิงจากระนาบต่างๆ ของแขนกลดังรูปที่ 3.4



รูปที่ 3.4 ระนาบต่างๆ ของแขนกล

จากระนาบต่างๆ ได้มีการหมุนแกนจากระนาบที่ฐานไปยังข้อต่างๆ โดยมีสมการการหมุนในแกน X, Y, Z โดยเป็นเมตริกซ์ 3 x 3 ดังนี้

แกน X

$$R_x(\theta) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\theta & -\sin\theta \\ 0 & \sin\theta & \cos\theta \end{bmatrix} \quad (3.1)$$

แกน Y

$$R_x(\theta) = \begin{bmatrix} \cos\theta & 0 & \sin\theta \\ 0 & 1 & 0 \\ -\sin\theta & 0 & \cos\theta \end{bmatrix} \quad (3.2)$$

แกน Z

$$R_x(\theta) = \begin{bmatrix} \cos\theta & -\sin\theta & 0 \\ \sin\theta & \cos\theta & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad (3.3)$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

แต่ในระบบของแขนกลนั้นจะใช้แกน Z แทนแกนการหมุนของมอเตอร์ และจะใช้แค่สมการการหมุนแกนของแกน Z และแกน X เท่านั้น และจากรูปที่ 3.19 จะได้ค่าพารามิเตอร์ต่างๆ ที่ไว้ช่วยในการหาสมการไคนิเมติกส์ (Kinematics Equation) ดังนี้

ตารางที่ 3.1 ค่าพารามิเตอร์ต่างๆ ของแขนกล

Axis	θ	D	A	α	Home (θ)
1	θ_1	D_1	0	$\frac{-\pi}{2}$	0
2	θ_2	0	A_1	0	0
3	θ_3	0	0	$\frac{-\pi}{2}$	$\frac{-\pi}{2}$
4	θ_4	$-D_2$	0	0	0

- ค่า θ ต่างๆ เป็นค่าที่ต้องการหรือที่ป้อนเข้าสู่ระบบ
- ค่า D คือค่าระยะทางในแนวแกน Z
- ค่า A คือค่าระยะทางในแนวแกน X
- ค่า α เป็นค่ามุมการหมุนระนาบโดยมีแกน X เป็นจุดหมุน
- ค่า Home (θ) เป็นค่ามุมในระนาบโดยมีแกน Z เป็นจุดหมุน

หลังจากได้ตารางค่าพารามิเตอร์มาแล้ว เราสามารถพิจารณาหาสมการแขนกลได้จากสมการการหมุนแกน Z และแกน X ดังนี้

$$T_{k-1}^{k} = \begin{bmatrix} \cos\theta_k & -\cos\alpha_k \sin\theta_k & \sin\alpha_k \sin\theta_k & A_k \cos\theta_k \\ \sin\theta_k & \cos\alpha_k \cos\theta_k & -\sin\alpha_k \cos\theta_k & A_k \sin_k \\ 0 & \sin\alpha_k & \cos\theta_k & D_k \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (3.4)$$

เพราะฉะนั้นสมการแขนกลทั้งหมดได้แก่

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

$$T_0^1 T_1^2 T_2^3 T_3^4 = \begin{bmatrix} c_1 & 0 & -s_1 & 0 \\ s_1 & 0 & c_1 & 0 \\ 0 & -1 & 0 & D \\ 0 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} c_2 & s_2 & 0 & A_1c_2 \\ s_2 & c_2 & 0 & A_2s_2 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} c_2 & s_2 & 0 & A_1c_2 \\ s_2 & c_2 & 0 & A_2s_2 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} c_2 & s_2 & 0 & A_1c_2 \\ s_2 & c_2 & 0 & A_2s_2 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$= \begin{bmatrix} c_1c_2 & -c_1s_2 & -s_1 & A_1c_1c_2 \\ s_1c_2 & -s_1s_2 & c_1 & A_1s_1c_2 \\ -s_1 & -c_1 & 0 & D_1 - A_1s_2 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} c_3c_4 & -c_3s_4 & 0 & A_1c_2 \\ s_3c_4 & -s_3s_4 & 0 & A_2s_2 \\ -s_4 & -c_4 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$= \begin{bmatrix} c_1c_4c_{2+3} + s_1s_2 & -c_1s_4c_{2+3} & -c_1s_{2+3} & c_1(-D_2s_{2+3} + A_1c_2) \\ s_1c_4c_{2+3} - c_1s_4 & -s_1s_4c_{2+3} - c_1s_4 & -s_1s_{2+3} & s_1(-D_2s_{2+3} + A_1c_2) \\ -s_{2+3}c_4 & -s_{2+3}s_4 & -c_{2+3} & -D_2c_{2+3} - A_1s_2 + D_1 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (3.5)$$

เมื่อได้สมการแขนกลซึ่งเป็นฟอร?เวร์สไคนิเมติกส?และทราบมุมของข้อต่อแต่ละข้อต่อ จะสามารถหาพิกัดปลายแขนได้

นอกจากฟอร์เวิร์สไคนิเมติกส์แล้ว ยังต้องการอินเวิร์สไคนิเมติกส์เพื่อให้ทราบมุมของแต่ละข้อต่อที่จะทำให้แขนกลขยับไปยังตำแหน่งที่ต้องการ โดยอ้างอิงจากเมตริกซ์ของฟอร์เวิร์สไคนิเมติกส์ ดังนี้

สิ่งที่ทราบในการทำอินเวิร์สไคนิเมติกส์คือ ตำแหน่ง X, Y, Z ระนาบแกน Z ของปลายแขนกลและมุมของมือจับ

หลักที่ 4 ของเมตริกซ์ฟอร์เวิร์สไคนิเมติกส์ของสมการที่ (3.5) นั้นจะเป็นตัวบอกพิกัด X, Y, Z

หลักที่ 3 ของเมตริกซ์ฟอร์เวิร์สไคนิเมติกส์ของสมการที่ (3.5) นั้นจะเป็นตัวระนาบแกน Z ของมือจับเทียบฐาน

และจากรูปที่ 3.19 สมมติให้ $W_4W_5W_6$ เป็นค่าระนาบแกน Z ของแขนกบทำมุมกับระนาบของฐานในแกน X, Y, Z ตามลำดับจะได้สมการที่ (3.6)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

$$\begin{bmatrix} c_1(-D_2 s_{2+3} + A_1 c_2) \\ s_1(-D_2 s_{2+3} + A_1 c_2) \\ -D_2 c_{2+3} - A_1 s_2 + D_1 \\ -(Exp\left(\frac{\theta_4}{\pi}\right))c_1 s_{2+3} \\ -(Exp\left(\frac{\theta_4}{\pi}\right))s_1 s_{2+3} \\ -(Exp\left(\frac{\theta_4}{\pi}\right))c_{2+3} \end{bmatrix} = \begin{bmatrix} X \\ Y \\ Z \\ W_4 \\ W_5 \\ W_6 \end{bmatrix} \quad (3.6)$$

หากแก้สมการหาค่า $\theta_1, \theta_2, \theta_3$ ได้ดังนี้

$$\theta_1 = \arctan\left(\frac{Y}{X}\right) \quad (3.7)$$

$$\theta_{2+3} = \arctan\left(-\frac{(\cos\theta_1 W_4 + \sin\theta_1 W_5)}{(-W_6)}\right) \quad (3.8)$$

$$\theta_3 = \arcsin\left(\frac{(\cos\theta_1 X_4 + \sin\theta_1 Y_5)^2 + (D_1 - Z_1)^2 - D_2^2 - A_1^2}{2A_1 D_2}\right) \quad (3.9)$$

$$\theta_1 = \arctan\left(\frac{\tan\theta_{2+3}}{\tan\theta_3 + (\tan\theta_{2+3} \times \tan\theta_3)}\right) \quad (3.10)$$

เมื่อ $c_i = \cos\theta_i$

$s_j = \sin\theta_j$

$c_{i+j} = \cos(\theta_i + \theta_j)$

$s_{i+j} = \sin(\theta_i + \theta_j)$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 4

ผลการทดลอง

4.1 การทดสอบมอเตอร์

เซอร์โวมอเตอร์เป็นมอเตอร์ที่อาศัยสัญญาณพัลซ์ควบคุมตำแหน่งโดยการจ่ายสัญญาณพัลซ์ที่ High 1.5 ms และ low ที่ 20 ms จะทำให้มอเตอร์หมุนไปที่ตำแหน่งตรงกลาง ถ้าเราจ่ายสัญญาณพัลซ์ที่ High 2.0 ms และ lowที่ 20 ms จะทำให้มอเตอร์หมุนไปที่ตำแหน่งซ้ายและถ้าเราจ่ายสัญญาณพัลซ์ที่ High 1.0 ms และ low ที่ 20 ms จะทำให้มอเตอร์หมุนไปที่ตำแหน่งขวาสุด

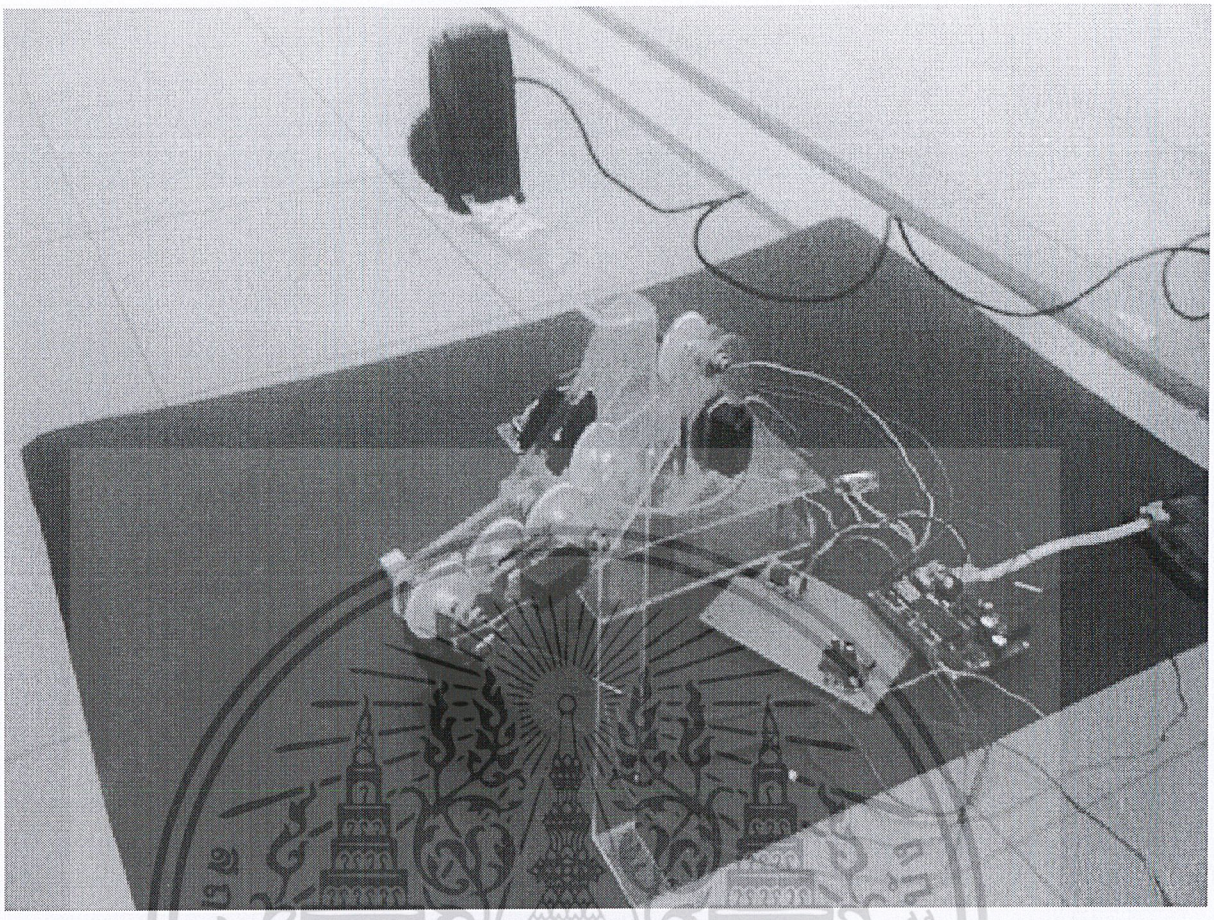
ตารางที่ 4.1 สเปกของมอเตอร์

รุ่นของเซอร์โวมอเตอร์	น้ำหนักที่รับได้
Tower Pro 945	15 kg
Tower Pro 945	15 kg
S3003	3kg
S3003	3kg
S3003	3kg

4.2 การติดตั้งและเชื่อมต่ออุปกรณ์

ในการทำการทดลองนั้นมีความจำเป็นที่ต้องมีการเตรียมอุปกรณ์ต่างๆ ให้เหมาะเพื่อความสะดวกและความแม่นยำในการทำการทดลองดังรูปที่ 4.1 จะเห็นว่ามีอุปกรณ์ดังนี้ คือ กล้องเว็บแคม ขาตั้งกล้อง คอมพิวเตอร์ กล่องใส่วงจรควบคุมแขนกล แขนกล ซึ่งสามารถเชื่อมต่ออุปกรณ์โดยนำสายเชื่อมต่อข้อมูลแบบ ยูเอสบี จากกล้องเว็บแคมต่อกับคอมพิวเตอร์ จากนั้นนำสายเชื่อมต่อข้อมูลแบบอาร์เอส-232 เชื่อมต่อระหว่างคอมพิวเตอร์กับพอร์ตอนุกรมอาร์เอส-232 (ซึ่งอยู่ที่กล่องใส่วงจรควบคุมแขนกล) และสุดท้ายทำการเชื่อมต่อระหว่างแขนกลกับวงจรควบคุมแขนกล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.1 แสดงการติดตั้งเชื่อมต่ออุปกรณ์

4.3 ปรับแต่งอุปกรณ์ ก่อนทำการทดลอง

4.3.1 การปรับตำแหน่งกล้องเว็บแคม

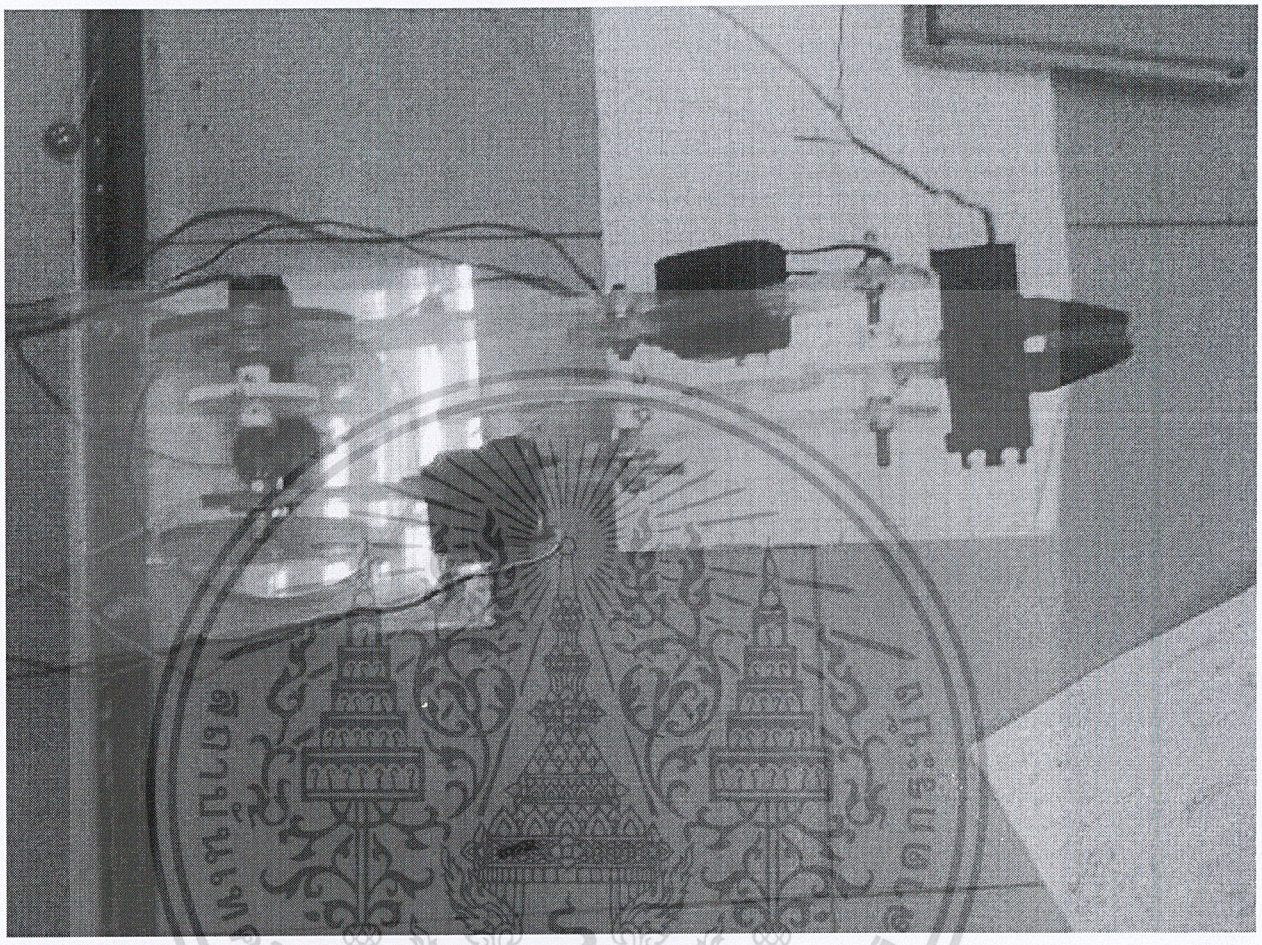
ในการจัดเตรียมอุปกรณ์นั้นสิ่งที่สำคัญอย่างหนึ่ง คือการตำแหน่งของกล้องให้สามารถเห็นสนามตามที่เราต้องการ

4.3.2 การปรับเทียบขนาดของพิกเซล

ในการประมวลผลด้วยภาพนั้น มีความจำเป็นที่ต้องทราบขนาดของพิกเซล ซึ่งคณะผู้จัดทำนำกระดาษสีขาว ที่ตีสเกลเรียบร้อยมาวางไว้บนสนาม เพื่อคำนวณหาขนาดของพิกเซลว่า 1 พิกเซล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

มีความยาวเท่าไรเมื่อเทียบกับขนาดจริง โดยอาศัยการเทียบบัญญัติไตรยางในการคำนวณค่า หลังจากนั้นนำค่าที่ได้ไปใช้ในการคำนวณหาตำแหน่งของมุมต่อไปต่อไป



รูปที่ 4.2 รูปแสดงการปรับเทียบของพิกเซล

4.4 ขั้นตอนการทดลองและผลการทดลอง

4.4.1 การทดสอบความแม่นยำของแขนกล

การทดลองความแม่นยำของแขนกล ทำได้โดยการป้อนตำแหน่งที่ต้องการให้แขนกลเคลื่อนที่ไปในคอมพิวเตอร์ เมื่อแขนกลเคลื่อนที่ไปอยู่ที่ตำแหน่งนั้นแล้ว จากนั้นจึงทำการวัดตำแหน่งของวัตถุด้วยตลับเมตร เมื่อวัดเสร็จก็สั่งให้แขนกลเคลื่อที่ไปอยู่ในตำแหน่งเริ่มต้นก่อน แล้วจึงป้อนค่าที่ต้องการให้แขนกลเคลื่อนที่ไป ลงใหม่ทำเช่นนี้เรื่อยๆ ซึ่งได้ผลดังนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 4.2 ผลการทดลองความแม่นยำแขนกลในแนวแกน X เมื่อกำหนดให้แกน Y = 0

ตำแหน่งที่ต้องการให้แขนกลเคลื่อนที่ไป (มิลลิเมตร)	ตำแหน่งที่แขนกลเคลื่อนที่ไปได้จริง(มิลลิเมตร)			
	ครั้งที่ 1	ครั้งที่ 2	ครั้งที่ 3	ค่าเฉลี่ย
20	23	24	25	24
30	31	33	29	31
50	52	52	52	52
70	71	72	70	71
100	103	104	104	103.6

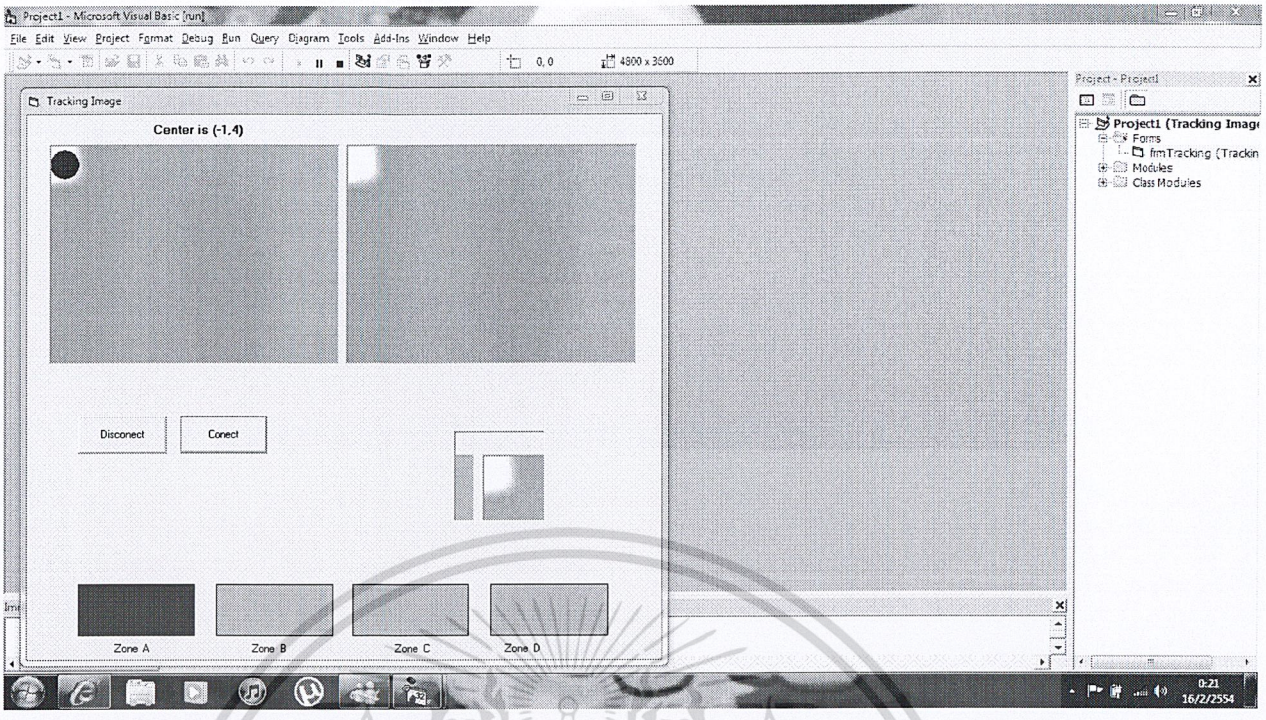
ตารางที่ 4.3 ผลการทดลองความแม่นยำแขนกลในแนวแกน Y เมื่อกำหนดให้แกน X = 0

ตำแหน่งที่ต้องการให้แขนกลเคลื่อนที่ไป (มิลลิเมตร)	ตำแหน่งที่แขนกลเคลื่อนที่ไปได้จริง(มิลลิเมตร)			
	ครั้งที่ 1	ครั้งที่ 2	ครั้งที่ 3	ค่าเฉลี่ย
20	23	23	23	23
30	30	31	32	31
50	52	51	50	51
70	71	72	70	71
100	104	104	104	104

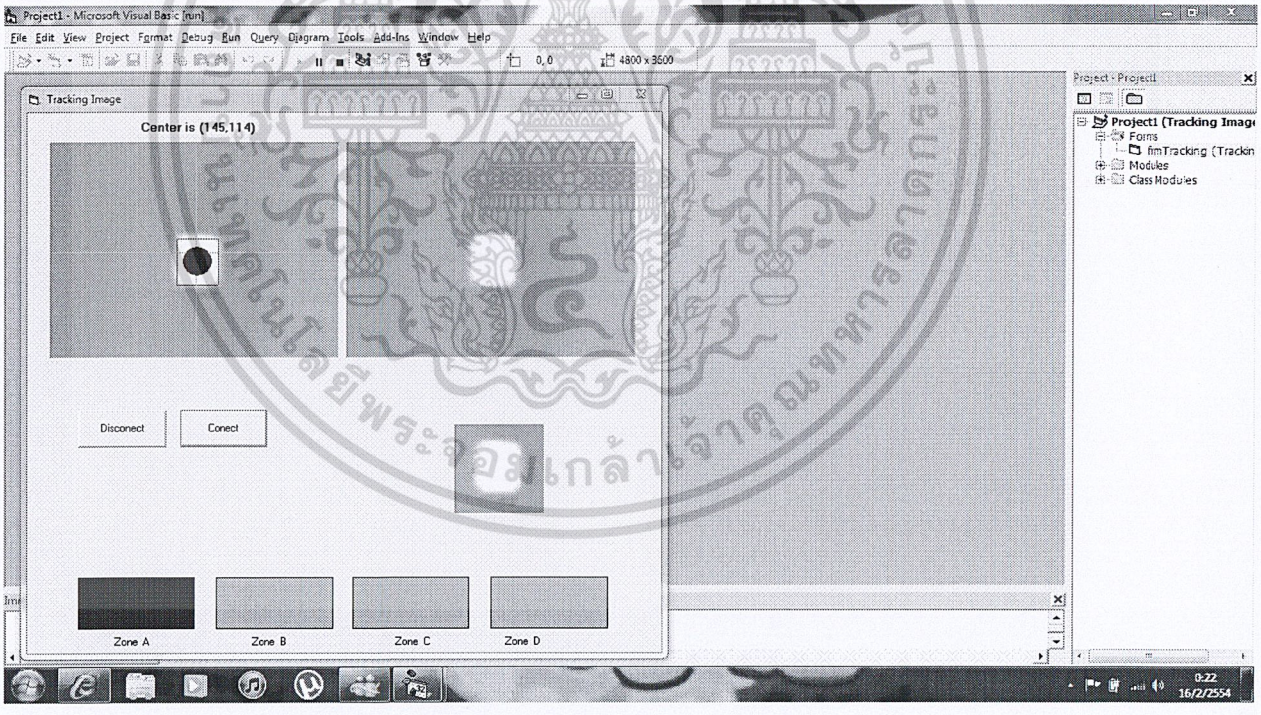
4.4.2 การทดลองหาตำแหน่งของวัตถุ

ทดลองหาตำแหน่งของวัตถุสีขาว ดังแสดงในรูปที่ 4.4, 4.5, 4.6

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

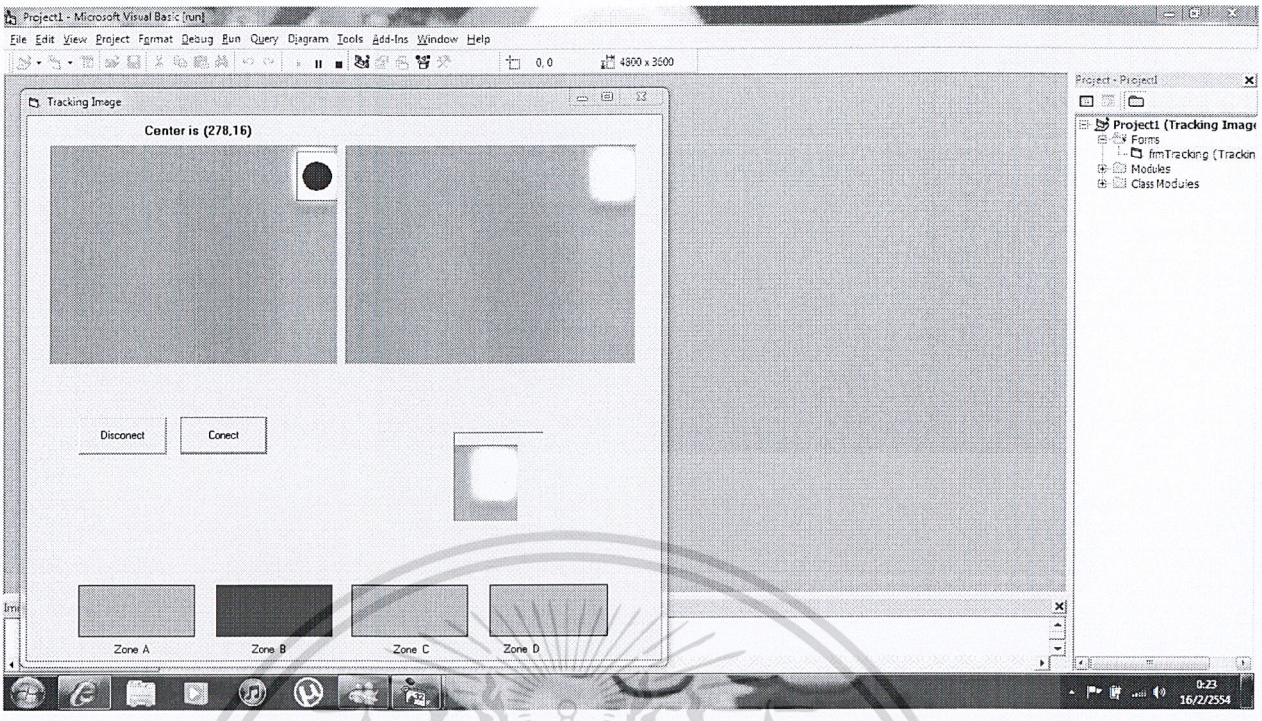


รูปที่ 4.3 แสดงการจับวัตถุสีขาวที่โซน A



รูปที่ 4.4 แสดงการจับวัตถุสีขาวที่ตำแหน่งกึ่งกลาง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.5 แสดงผลการจับวัตถุสีขาวที่โซน B

ตารางที่ 4.4 แสดงผลการจับวัตถุสีขาวของกล้องเว็บแคม

ตำแหน่ง X,Yค่าจริง (มิลลิเมตร)	ผลการทดลองโดยใช้กล้องเว็บแคมจับ			
	ครั้งที่ 1	ครั้งที่ 2	ครั้งที่ 3	เฉลี่ย
(0,0)	(0,0)	(-1,0)	(-1,-1)	(-0.67,-0.33)
(120,0)	(276,4)	(274,0)	(280,0)	(276.67,1.33)
(0,90)	(0,200)	(-1,208)	(-2,206)	(-1,204.67)
(120,90)	(268,208)	(270,206)	(268,210)	(268.67,208)

4.4.3 การทดลองการจับวัตถุ

หลังจากทำการจัดอุปกรณ์ และตั้งค่าเริ่มต้นพร้อมทั้งประมวลผลภาพ จะส่งข้อมูลให้แขนกลเคลื่อนที่ไปยังตำแหน่งของวัตถุดังรูปที่ 4.6 โดยแบ่งการทำงานของแขนกลได้ 2 ช่วงการทำงาน ช่วงการทำงานแรกคือการเคลื่อนที่จากตำแหน่งเริ่มต้น ไปยังตำแหน่ง (120,90,40) ซึ่งเป็นตำแหน่ง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ของแขนกลอยู่เหนือวัตถุ จากนั้นในช่วงการทำงานต่อมา แขนกลจะเคลื่อนที่จากตำแหน่งดังกล่าวไปยังตำแหน่ง (120,90,20) ซึ่งเป็นตำแหน่งที่แขนกลสามารถจับวัตถุได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 5

สรุปผลการทดลอง

5.1 สรุปผลการดำเนินงาน

1 .สามารถสื่อสารนำข้อมูลภาพถ่ายจากกล้องดิจิตอลเข้าสู่คอมพิวเตอร์ และประมวลเปลี่ยนเป็นข้อมูลภาพให้อยู่ในรูปของไบนารี และประยุกต์ใช้ในการประมวลผลจับวัตถุสีขาวได้และหาตำแหน่งของวัตถุได้

2. สามารถนำข้อมูลที่ได้จากการประมวลผลภาพเปลี่ยนเป็นองศา และส่งสัญญาณพัลซ์ไปควบคุมให้เซอร์โวมอเตอร์หมุนไปยังตำแหน่งที่ต้องการได้

3. สามารถติดต่อแลกเปลี่ยนข้อมูลระหว่างไมโครคอนโทรลเลอร์ กับคอมพิวเตอร์ผ่านพอร์ตอนุกรมได้

4. สามารถประยุกต์ใช้งานไมโครคอนโทรลเลอร์ ในการควบคุมตำแหน่งของแขนกลให้มีความแม่นยำได้อย่างมีประสิทธิภาพ

5. แขนกลสามารถทำงานรับข้อมูลจากไมโครคอนโทรลเลอร์ได้

5.2 ปัญหาและอุปสรรค

1. เนื่องจากคณะผู้จัดทำได้ใช้เฟืองในการเป็นตัวขับเคลื่อนแขนกล เมื่อทดลองจริงเฟืองเกิดการรุดทำให้ผลการทดลองคลาดเคลื่อน

2. แขนกลมีน้ำหนักค่อนข้างมาก ทำให้รับโหลดได้น้อย

3. การประมวลผลภาพ จากผลการทดลองจะเห็นได้ว่าจะเกิดความคลาดเคลื่อนค่อนข้างมาก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4. เนื่องจากแสงสว่างที่ส่องลงบนพื้นสนามต้องมีความสว่างสม่ำเสมอถึงจะสามารถประมวลผลได้อย่างแม่นยำ

5.3 แนวทางการแก้ไข

1. ควรใช้เฟืองตัวหนอนเป็นตัวขับเคลื่อนแขนกลเพื่อกันการรุด

2. ทำโครงสร้างให้มีขนาดเบา แต่แข็งแรงเพื่อสามารถจะทำให้แขนกลรับโหลดได้มาก

3. ควรใช้กล้องที่มีความแม่นยำมากว่านี้

4. จัดบริเวณที่ทำการทดลองให้มีแสงสว่างให้เหมาะสม

5.4 ประโยชน์ที่ได้รับจากโครงงาน

1. สามารถนำความรู้และประสบการณ์มาประยุกต์ใช้ในการทำงานจริง

2. ฝึกและเรียนรู้หาความรู้ใหม่ๆมาประยุกต์ใช้ในการทำงาน

3. สามารถทำงานเป็นทีมได้

4. ฝึกกระบวนการคิดในการทำงานได้

5. รู้จักแก้ปัญหาที่แก้ขึ้นได้อย่างมีประสิทธิภาพ

5.5 แนวทางพัฒนา

พัฒนาระบบให้มีประสิทธิภาพมากขึ้นเช่น สามารถจับวัตถุที่มีรูปร่าง ขนาด ที่แตกต่างกันและสีสันที่หลากหลาย สามารถจับวัตถุได้อย่างแม่นยำ สามารถใช้งานได้ในระนาบ 3 มิติ

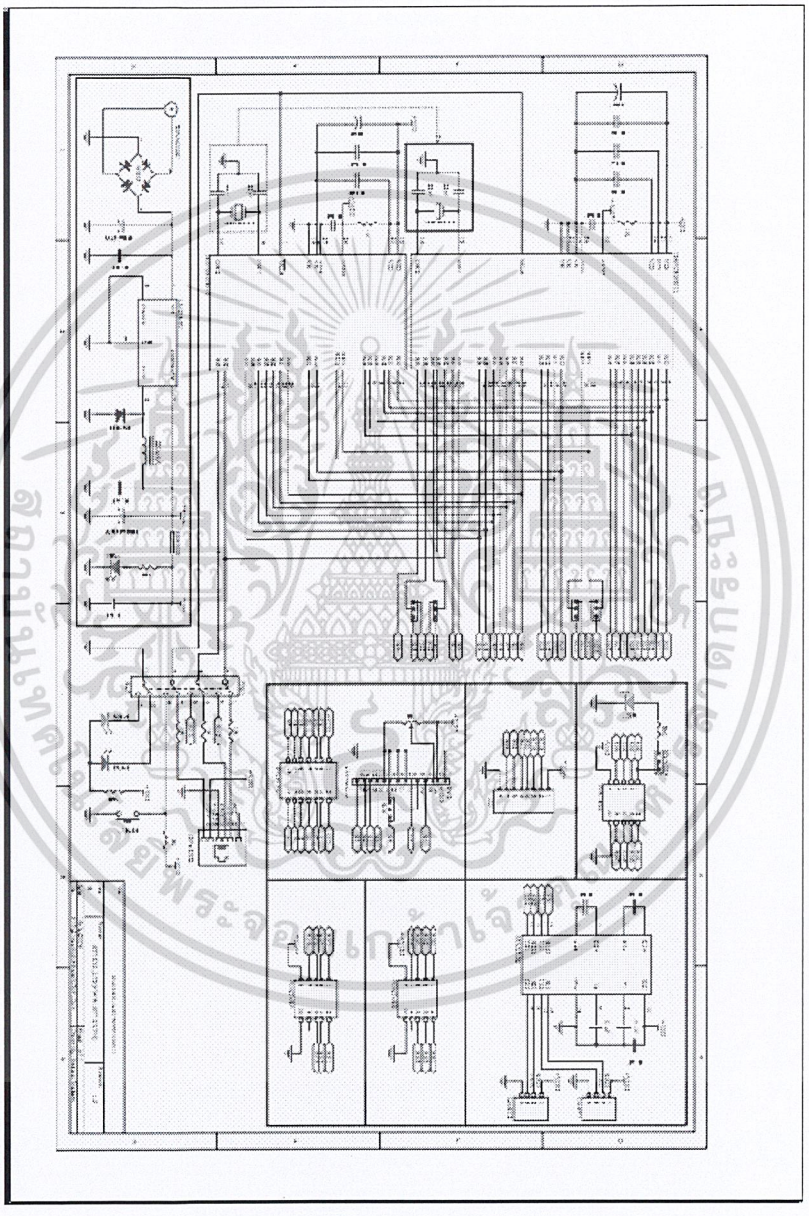
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก ก

วงจรภายในของบอร์ด ET-BASE Dspic30F4011



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก ข

โปรแกรมประมวลผลภาพ

```
Option Explicit

Dim RoddersMotion As New ClassMotion


Public M As Integer

Public N As Integer

Public O As Integer

Public p As Integer

Public Q As Integer

Public r As Integer

Public S As Integer

Public T As Integer

Public U As Integer

Public V As Integer

Public minx As Integer

Public miny As Integer

Public maxx As Integer

Public maxy As Integer


Private Declare Function SendMessage Lib "user32" Alias "SendMessageA" _

(ByVal hWnd As Long, ByVal wMsg As Long, ByVal wParam As Long, _

lParam As Any) As Long
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
Private Declare Function capCreateCaptureWindow Lib "avicap32.dll" Alias _
"capCreateCaptureWindowA" (ByVal lpszWindowName As String, ByVal _
dwStyle As Long, ByVal x As Long, ByVal y As Long, ByVal nWidth As Long, _
ByVal nHeight As Long, ByVal hwndParent As Long, ByVal nID As Long) As Long
Private mCapHwnd As Long

Private Const CONNECT As Long = 1034
Private Const DISCONNECT As Long = 1035
Private Const GET_FRAME As Long = 1084
Private Const COPY As Long = 1054
Private Const VIDEOFORMAT As Long = 1065
Private Const VIDEOSOURCE As Long = 1066
Private Const VIDEODISPLAY As Long = 1067
Private Const VIDEOCOMPRESSION As Long = 1070
Private Const PREVIEW As Long = 1074

Private Declare Function GetObject Lib "gdi32" Alias "GetObjectA" _
(ByVal hObject As Long, ByVal nCount As Long, lpObject As Any) As Long

Private Declare Function GetDIBits Lib "gdi32" (ByVal aHDC As Long, _
ByVal hBitmap As Long, ByVal nStartScan As Long, ByVal nNumScans As _
Long, lpBits As Any, lpBI As BITMAPINFO, ByVal wUsage As Long) As Long

Private Declare Function SetDIBits Lib "gdi32" (ByVal hdc As Long, _
ByVal hBitmap As Long, ByVal nStartScan As Long, ByVal nNumScans _
As Long, lpBits As Any, lpBI As BITMAPINFO, ByVal wUsage As Long) As Long
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
Private Type BITMAP '14 bytes
    bmType As Long
    bmWidth As Long
    bmHeight As Long
    bmWidthBytes As Long
    bmPlanes As Integer
    bmBitsPixel As Integer
    bmBits As Long
End Type
Private Type BITMAPINFOHEADER '40 bytes
    biSize As Long
    biWidth As Long
    biHeight As Long
    biPlanes As Integer
    biBitCount As Integer
    biCompression As Long
    biSizeImage As Long
    biXPelsPerMeter As Long
    biYPelsPerMeter As Long
    biClrUsed As Long
    biClrImportant As Long
End Type
Private Type RGBQUAD
    rgbBlue As Byte
    rgbGreen As Byte
    rgbRed As Byte
    rgbReserved As Byte
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
End Type
Private Type BITMAPINFO
   bmiHeader As BITMAPINFOHEADER
   bmiColors As RGBQUAD
End Type
Private Const DIB_RGB_COLORS = 0&
Private Const BI_RGB = 0&

Private Const pixR As Integer = 3
Private Const pixG As Integer = 2
Private Const pixB As Integer = 1

Private Sub Find_Object(ByVal picColor As PictureBox)
Dim bitmap_info As BITMAPINFO
Dim pixels() As Byte
Dim bytes_per_scanLine As Integer
Dim pad_per_scanLine As Integer
Dim x As Integer
Dim y As Integer
Dim ave_color As Byte


   minx = picColor.Width
   miny = picColor.Height
   maxx = 0
   maxy = 0
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
' Prepare the bitmap description.
With bitmap_info.bmiHeader
    .biSize = 40
    .biWidth = picColor.ScaleWidth
    ' Use negative height to scan top-down.
    .biHeight = -picColor.ScaleHeight
    .biPlanes = 1
    .biBitCount = 32
    .biCompression = BI_RGB
    bytes_per_scanLine = ((((.biWidth * .biBitCount) + 31) \ 32) * 4)
    pad_per_scanLine = bytes_per_scanLine - (((.biWidth * .biBitCount) + 7) \ 8)
    .biSizeImage = bytes_per_scanLine * Abs(.biHeight)
End With

' Load the bitmap's data.
ReDim pixels(1 To 4, 1 To picColor.ScaleWidth, 1 To picColor.ScaleHeight)
GetDIBits picColor.hdc, picColor.Image, _
    0, picColor.ScaleHeight, pixels(1, 1, 1), _
    bitmap_info, DIB_RGB_COLORS

' Modify the pixels.
For y = 1 To picColor.ScaleHeight
    For x = 1 To picColor.ScaleWidth

        If (pixels(pixB, x, y) > 246) And (pixels(pixG, x, y) > 246) And (pixels(pixR, x, y) > 246)
Then

        pixels(pixR, x, y) = 0
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
            pixels(pixG, x, y) = 0
            pixels(pixB, x, y) = 0

            If minx > x Then minx = x
            If maxx < x Then maxx = x
            If miny > y Then miny = y
            If maxy < y Then maxy = y

            'Else
            'pixels(pixR, X, Y) = 255
            'pixels(pixG, X, Y) = 255
            'pixels(pixB, X, Y) = 255
            End If

        Next x
    Next y

    N = (minx + maxx) / 2
    M = (minx + maxx) / 2

    'picColor.Line (N, miny)-(M, maxy), vbGreen

    O = (miny + maxy) / 2
    p = (miny + maxy) / 2
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
'picColor.Line (minx, O)-(maxx, P), vbGreen



picColor.Line (minx, miny)-(maxx, miny), vbRed
picColor.Line (maxx, miny)-(maxx, maxy), vbRed
picColor.Line (minx, maxy)-(maxx, maxy), vbRed
picColor.Line (minx, miny)-(minx, maxy), vbRed

picColor.Line (160, 0)-(160, 240), vbGreen
picColor.Line (0, 120)-(320, 120), vbGreen

shpLED0(0).Visible = True
shpLED0(0).BackColor = vbBlue


Q = (N + M) / 2
r = (O + p) / 2

shpLED0(0).Width = 32
shpLED0(0).Height = 32

shpLED0(0).Top = O - 16
shpLED0(0).Left = N - 16

Label1.Caption = "Center is (" & (N - 16) & "," & (O - 16) & ")"
U = N - 16
V = O - 16
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    check
End Sub



Private Sub cmdcon_Click()


  mCapHwnd = capCreateCaptureWindow("WebcamCapture", 0, 0, 0, 320, 240, Me.hWnd,
0)
  DoEvents
  SendMessage mCapHwnd, CONNECT, 0, 0
  Picture3.Width = 320 * Screen.TwipsPerPixelX
  Picture3.Height = 240 * Screen.TwipsPerPixelY


  picColor.AutoSize = True
  picColor.AutoRedraw = True


  shpLED0(0).Visible = False
  shpLED0(1).BackColor = vbGreen
  shpLED0(2).BackColor = vbGreen
  shpLED0(3).BackColor = vbGreen
  shpLED0(4).BackColor = vbGreen
  Timer1.Enabled = True


End Sub
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
Private Sub cmdDiscon_Click()
 DoEvents
   SendMessage mCapHwnd, DISCONNECT, 0, 0

   Timer1.Enabled = False
End Sub

Private Sub Form_Load()
picFovea.Width = 1500
picFovea.Height = 1500
End Sub

Private Sub Timer1_Timer()

   SendMessage mCapHwnd, GET_FRAME, 0, 0
   SendMessage mCapHwnd, COPY, 0, 0

   Picture3.Picture = Clipboard.GetData
   Clipboard.Clear

   picColor.Picture = Picture3.Image
   picColor.Visible = True
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    ' Convert to gray scale.
    Screen.MousePointer = vbHourglass
    DoEvents
    Find_Object picColor
    Screen.MousePointer = vbDefault

     Call RoddersMotion.showFovea(Picture3, picFovea)
End Sub

Private Sub check()

If U < 160 And V < 120 Then
shpLED0(1).BackColor = vbBlue
shpLED0(2).BackColor = vbGreen
shpLED0(3).BackColor = vbGreen
shpLED0(4).BackColor = vbGreen
End If

If U > 160 And V < 120 Then
shpLED0(1).BackColor = vbGreen
shpLED0(2).BackColor = vbBlue
shpLED0(3).BackColor = vbGreen
shpLED0(4).BackColor = vbGreen
End If

If U < 160 And V > 120 Then
shpLED0(1).BackColor = vbGreen
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
shpLED0(2).BackColor = vbGreen
shpLED0(3).BackColor = vbBlue
shpLED0(4).BackColor = vbGreen
End If

If U > 160 And V > 120 Then
shpLED0(1).BackColor = vbGreen
shpLED0(2).BackColor = vbGreen
shpLED0(3).BackColor = vbGreen
shpLED0(4).BackColor = vbBlue
End If

If U > 320 And V > 240 Then
shpLED0(1).BackColor = vbGreen
shpLED0(2).BackColor = vbGreen
shpLED0(3).BackColor = vbGreen
shpLED0(4).BackColor = vbGreen
Label1.Caption = "Out Of Frame"
End If


End Sub
//Modules


Option Explicit
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
Public Type RGBthingy
 Value As Long
End Type

Public Type RGBpoint
 Red As Byte
 Green As Byte
 Blue As Byte
End Type

Public Const ws_child As Long = &H40000000
Public Const ws_visible As Long = &H10000000
Public Const SWP_NOSIZE As Long = &H1&
Public Const SWP_NOMOVE As Long = &H2&
Public Const SWP_NOZORDER As Long = &H4&
Public Const SWP_NOSENDCHANGING As Long = &H400&   ' /* Don't send
WM_WINDOWPOSCHANGING */
Public Const HWND_BOTTOM As Long = 1&

Public Const SM_CYCAPTION As Long = 4
Public Const SM_CXBORDER As Long = 5
Public Const SM_CYBORDER   As Long = 6

Public Const SM_CYMENU   As Long = 15

Public Const SM_CXEDGE     As Long = 45
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
Public Const SM_CYEDGE    As Long = 46


Declare Function ShellAbout Lib "shell32" Alias "ShellAboutA" _
                (ByVal hWnd As Long, _
                ByVal szApp As String, _
                ByVal szOtherStuff As String, _
                ByVal hIcon As Long) As Long
Declare Function SetWindowTextAsLong Lib "user32" Alias "SetWindowTextA" (ByVal hWnd
As Long, ByVal LPCSTR As Long) As Long ' C BOOL
Declare Function GetSystemMetrics Lib "user32" (ByVal nIndex As Long) As Long
Declare Function SetParent Lib "user32" (ByVal hWndChild As Long, ByVal hWndNewParent
As Long) As Long
Declare Function SetWindowPos Lib "user32" (ByVal hWnd As Long, ByVal hWndInsertAfter
As Long, ByVal x As Long, ByVal y As Long, ByVal cx As Long, ByVal cy As Long, ByVal wFlags
As Long) As Long
Declare Function DestroyWindow Lib "user32" (ByVal hWnd As Long) As Long 'C BOOL
Declare Function GetDiskFreeSpace Lib "kernel32" Alias "GetDiskFreeSpaceA" _
                (ByVal lpRootPathName As String, _
                lpSectorsPerCluster As Long, _
                lpBytesPerSector As Long, _
                lpNumberOfFreeClusters As Long, _
                lpTtoalNumberOfClusters As Long) As Long 'C BOOL


'== Global Memory Functions ==========================================
Declare Function GlobalAlloc Lib "kernel32" (ByVal wFlags As Long, ByVal dwBytes As Long)
As Long
Declare Function GlobalFree Lib "kernel32" (ByVal hMem As Long) As Long
Declare Function GlobalLock Lib "kernel32" (ByVal hMem As Long) As Long
Declare Function GlobalUnlock Lib "kernel32" (ByVal hMem As Long) As Long
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

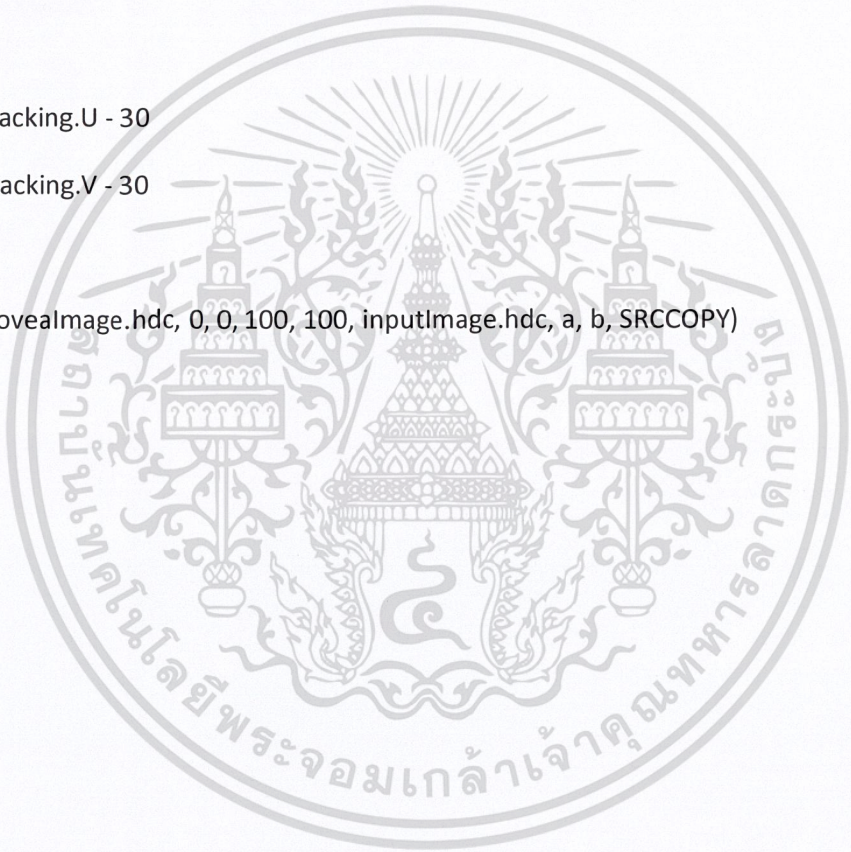
```
Public Declare Sub CopyMemory Lib "kernel32" Alias "RtlMoveMemory" (lpvDest As Any,
lpvSource As Any, ByVal cbCopy As Long)


Declare Function lstrcpy Lib "kernel32" Alias "lstrcpyA" (ByVal lpStringDest As Long, ByVal
lpStringSrc As Long) As Long

Declare Sub CopyPTRtoANY Lib "kernel32.dll" Alias "RtlMoveMemory" (ByRef Dest As Any,
ByVal PtrSrc As Long, ByVal length As Long)

Declare Sub CopyPTRtoLONG Lib "kernel32.dll" Alias "RtlMoveMemory" (ByRef LONGDest As
Long, ByVal PtrSrc As Long, ByVal length As Long)


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

Declare Function CreateBitmap Lib "gdi32" (ByVal nWidth As Long, ByVal nHeight As Long,
ByVal nPlanes As Long, ByVal nBitCount As Long, lpBits As Any) As Long

Public Declare Function SetBkColor Lib "gdi32" (ByVal hdc As Long, ByVal crColor As Long) As
Long

Public Declare Function SelectObject Lib "gdi32" (ByVal hdc As Long, ByVal hObject As Long)
As Long

Public Declare Function CreateCompatibleBitmap Lib "gdi32" (ByVal hdc As Long, ByVal
nWidth As Long, ByVal nHeight As Long) As Long

Public Declare Function CreateCompatibleDC Lib "gdi32" (ByVal hdc As Long) As Long

Public Declare Function DeleteDC Lib "gdi32" (ByVal hdc As Long) As Long

Public Declare Function DeleteObject Lib "gdi32" (ByVal hObject As Long) As Long


Public Const GMEM_MOVEABLE = &H2&

Public Const GMEM_SHARE = &H2000&

Public Const GMEM_ZEROINIT = &H40&


'VFW stuff
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
Global Const WM_USER = 1024

Global Const WM_CAP_EDIT_COPY = WM_USER + 30

Global Const wm_cap_driver_connect = WM_USER + 10

Global Const wm_cap_set_preview = WM_USER + 50

Global Const wm_cap_set_overlay = WM_USER + 51

Global Const WM_CAP_SET_PREVIEWRATE = WM_USER + 52

Global Const WM_CAP_SEQUENCE = WM_USER + 62

Global Const WM_CAP_SINGLE_FRAME_OPEN = WM_USER + 70

Global Const WM_CAP_SINGLE_FRAME_CLOSE = WM_USER + 71

Global Const WM_CAP_SINGLE_FRAME = WM_USER + 72


Public Const WM_CAP_DLG_VIDEOFORMAT As Long = WM_USER + 41


Global Const DRV_USER = &H4000

Global Const DVM_DIALOG = DRV_USER + 100

Global Const WM_CAP_DRIVER_DISCONNECT As Long = WM_USER + 11

Declare Function SendMessage Lib "user32" Alias "SendMessageA" (ByVal hWnd As Long,
ByVal wMsg As Long, ByVal wParam As Long, ByVal lParam As Long) As Long

Declare Function capCreateCaptureWindow Lib "avicap32.dll" Alias
"capCreateCaptureWindowA" (ByVal a As String, ByVal b As Long, ByVal c As Integer, ByVal d
As Integer, ByVal e As Integer, ByVal f As Integer, ByVal g As Long, ByVal h As Integer) As
Long

Declare Function SendMessageAsLong Lib "user32" Alias "SendMessageA" _

                    (ByVal hWnd As Long, _

                    ByVal wMsg As Long, _

                    ByVal wParam As Long, _

                    ByVal lParam As Long) As Long

Global Const WM_CAP_GRAB_FRAME As Long = WM_USER + 60

//ClassMotion
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
Option Explicit

Dim hwndc As Long

Const SRCCOPY = &HCC0020

Public Sub showFovea(inputImage As PictureBox, foveaImage As PictureBox)

  Dim rc As Long

 foveaImage.Cls

 Dim a As Integer

 Dim b As Integer


   a = frmTracking.U - 30

   b = frmTracking.V - 30


rc = BitBlt(foveaImage.hdc, 0, 0, 100, 100, inputImage.hdc, a, b, SRCCOPY)


End Sub
```



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เอกสารอ้างอิง

สิทธิโชค ยอดระยับ (2550), การเขียนโปรแกรม Digital Image Processing ด้วย Visual Basic, สำนักพิมพ์สสท.

Appsofttech, Dspic30Fprogramming กับ MPLAB C คอมไพเลอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้