เลเซอร์สแกนวัตถุ
LASER SCANNER

กันต์ดนัย ศักดิ์ศรีวิทยา
Kundonai Sakstriwittaya
ศุภชัย อาลัย
Suphachai Arlai
ศุภากร หลวงจอก
Supakhom Luangjok

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต
สาขาวิชาวิศวกรรมอิเล็กทรอนิกส์
คณะวิศวกรรมศาสตร์
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
พ.ศ. 2555

เลเซอร์สแกนวัตถุ

Laser scanner

กันต์ดนัย ศักดิ์ศรีวิทยา
Kundanai Saksriwittaya

ศุภชัย อาลัย
Suphachai Arlai

ศุภากร หลวงจอก
Supakhon Laungjok

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต
สาขาวิศวกรรมอิเล็กทรอนิกส์
คณะวิศวกรรมศาสตร์
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
พ.ศ.2555

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เลเซอร์สแกนวัตถุ

Laser scanner

โดย

กันต์ดนัย ศักดิ์ศรีวิทยา

ศุภชัย อาลัย

ศุภากร หลวงจอก

อาจารย์ที่ปรึกษา

รศ.ดร.สุริภณ สมควรพาณิชย์

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต

สาขาวิศวกรรมอิเล็กทรอนิกส์

คณะวิศวกรรมศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

พ.ศ.2555

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ปริญญานิพนธ์ปีการศึกษา 2555

สาขาวิชา วิศวกรรมอิเล็กทรอนิกส์

คณะ วิศวกรรมศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เรื่อง เลเซอร์สแกนวัตถุ

Laser scanner

ผู้จัดทำ	นายกันต์ดนัย	ศักดิ์ศรีวิทยา	รหัสนักศึกษา	52010048
	นายศุภชัย	อาลัย	รหัสนักศึกษา	52011216
	นางสาวศุภากร	หลวงจอก	รหัสนักศึกษา	52011225

ปริญญานิพนธ์นี้ผ่านการตรวจสอบโดยอาจารย์ที่ปรึกษาแล้ว

(รศ.ดร.สุริภณ สมควรพาณิชย์)

อาจารย์ที่ปรึกษา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หัวข้อปริญญานิพนธ์	เลเซอร์สแกนวัตถุ			
นักศึกษา	นายกันต์ดนัย	ศักดิ์ศรีวิทยา	รหัสนักศึกษา	52010048
	นายศุภชัย	อาลัย	รหัสนักศึกษา	52011216
	นางสาวศุภากร	หลวงจอก	รหัสนักศึกษา	52011225
ปริญญา	วิศวกรรมศาสตรบัณฑิต			
สาขา	วิศวกรรมอิเล็กทรอนิกส์			
ปีการศึกษา	2555			
อาจารย์ที่ปรึกษาปริญญานิพนธ์	รศ.ดร. สุริภณ สมควรพาณิชย์			

บทคัดย่อ

โครงงานเลเซอร์สแกนวัตถุ เป็นเครื่องมือที่ใช้การสร้างจำลองรูปทรงของวัตถุด้วยเลเซอร์ มีหลักการในการทำงานคือ ฉายเลเซอร์ลงบนพื้นผิวของวัตถุ และทำการถ่ายภาพวัตถุที่มีแสงเลเซอร์อยู่บนพื้นผิว นำภาพที่ได้ประมวลผลเพื่อหารูปทรงของวัตถุ การประมวลภาพถ่ายได้จะใช้หลักการแบ่งระดับความเข้มของแสงเลเซอร์ที่มีความเข้มสูง และสามารถหาพิกัดของแสงเลเซอร์ที่อยู่บนพื้นผิวของวัตถุ นำตำแหน่งของพิกัดจุดแสงเลเซอร์แต่ละจุดมารวมกันเป็นรูปทรงของวัตถุที่ทำการสแกน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Thesis Title	Laser scanner	
Student	Mr.Kundanai Saksriwittaya	ID 52010048
	Mr.Suphachai Arlai	ID 52011216
	Ms.Supakhon Laungjok	ID 52011225
Degree	Bachelor of Engineering	
Program	Electronics Engineering	
Year	2012	
Thesis Advisor	Assoc. Prof. Dr. Suripon Somkuarnpanit	

Abstract

This Laser Scanner Project presents a device used to form a shape of an object with laser. This device will project the laser on an object layer and take pictures of the object covered with the laser. The picture will be evaluated to identify the shape of the object. This evaluation processes by intensity level slicing of the laser. It will find the coordinate of the laser points on the object layer and combine each point to form the 3D imitation figure of the object scanned.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กิตติกรรมประกาศ

โครงงานนี้สำเร็จรุล่วงด้วยความกรุณา รศ.ดร.สุริภณ สมควรพาณิชย์ อาจารย์ที่ปรึกษาโครงงาน ที่ได้ให้คำแนะนำแนวทาง แนวคิดในการทำโครงงาน ตลอดจนการแก้ไขปัญหาความผิดพลาดในการทำโครงงานครั้งนี้มาตลอด จนโครงงานในครั้งนี้สามารถสำเร็จสมบูรณ์ จึงขอกราบขอบพระคุณ

ขอกราบขอบพระคุณ คุณพ่อ คุณแม่ และผู้ปกครองที่ได้ให้คำปรึกษาในเรื่องต่างๆ รวมทั้งเป็นกำลังที่ดีเสมอมา

ขอขอบคุณภาควิชาอิเล็กทรอนิกส์ คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง ที่ได้ให้เงินสนับสนุนในการทำโครงงานชิ้นนี้

ขอขอบพระคุณอาจารย์ประจำภาควิชาอิเล็กทรอนิกส์ คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง ที่ได้ให้คำปรึกษาเรื่องต่างๆที่เกี่ยวข้องกับพื้นฐานการเขียนโปรแกรม MatLab ข้อมูลเนื้อหาในการทำโครงงานครั้งนี้

ขอขอบคุณเพื่อนๆ ภาควิชาอิเล็กทรอนิกส์ คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง ที่ช่วยให้คำปรึกษา คำแนะนำ ในด้านการเขียนโปรแกรม Matlab และเป็นกำลังใจในการทำปริญญานิพนธ์นี้สำเร็จลุล่วงไปด้วยดี หากปริญญานิพนธ์ฉบับนี้มีข้อบกพร่องประการใด ข้าพเจ้ายินดีรับข้อเสมอและขออภัยมา ณ ที่นี้ด้วย

กันต์ดนัย ศักดิ์ศรีวิทยา
ศุภชัย อาลัย
ศุภากร หลวงจอก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ

หน้า

บทคัดย่อภาษาไทย.....I
บทคัดย่อภาษาอังกฤษ.....II
กิตติกรรมประกาศ.....III
สารบัญ.....IV
สารบัญตาราง.....VIII
สารบัญรูปภาพ.....IX

บทที่ 1 บทนำ

1.1 ความเป็นมาและความสำคัญของปัญหา.....1
1.2 วัตถุประสงค์ของโครงงาน.....1
1.3 ขอบเขตของโครงงาน.....1
1.4 ประโยชน์ที่ได้รับ.....2

บทที่ 2 ทฤษฎีและหลักการทำงานที่เกี่ยวข้อง

2.1 ชนิดของภาพ.....3
2.1.1 ภาพดิจิตอล(Digital Image).....3
2.1.2 ภาพความเข้มแสง(Intensity image).....3
2.1.3 ภาพไบนารี(Binary Image.....3
2.1.4 ภาพสี (RGB image).....3
2.2 การปรับปรุงภาพ (Image Enhancement).....5
2.2.1 การแบ่งระดับความเข้ม (Intensity Level Slicing).....5
2.2.2 การปรับปรุงขนาดของเส้น.....6
2.3 การติดต่อกล้อง USB ผ่าน Image Acquisition.....6
2.4 การโปรเจคชั่น(Projections).....7
2.4.1 โปรเจคชั่นแบบขนาน (Parallel Projections).....7
2.4.2 โปรเจคชั่นแบบเพอสเปกทีพ(Perspective Projection).....7
2.5 การเกิดภาพสเตอริโอ(stereo image).....8
2.6 เรขาคณิตสำหรับการเห็นใน 3 มิติ (Geometry for 3D vision).....9
2.6.1 พื้นฐานของการโปรเจคชันแบบเพอสเปกทีฟของกล้อง (Basis of Perspective Camera).....9
2.6.2 การแปลงจากหน่วยความยาวเป็นพิกเซล.....11
2.6.3 กรณีที่พิกัดของโลกไม่ซ้อนทับกับพิกัดของกล้อง.....11

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.7 การปรับเทียบกล้อง (calibration) 14
2.8 การพิกัด 3 มิติของจุดในฉากโดยใช้กล้อง 2 ตัว 15
(3D Coordinate Extraction)
2.9 ทฤษฎีและหลักการทำงานด้านฮาร์ดแวร์ (Hardware) 17
2.9.1 วงจรจ่ายไฟกระแสตรง (DC Power Supply) 17
2.9.2 โครงสร้างพื้นฐานของแหล่งจ่ายไฟกระแสตรงแบบเชิงเส้น 18
2.9.2.1 หม้อแปลงไฟฟ้า (Transformer) 18
2.9.2.2 วงจรเรียงกระแส 19
2.9.2.3 วงจรกรองลดการกระเพื่อม (Ripple Filter) 22
2.9.2.4 วงจรรักษาแรงดันใช้ไอซีรักษาแรงดัน (IC Regulator) 24
2.9.3 วงจรแยกกราวน์ (ground isolation Circuit) 26
2.9.4 Arduino Board 27
2.9.5 Microcontroller ที่ใช้บนบอร์ด AVR 28
2.9.6 Software Arduino 29
2.9.7 จอแสดงผล LCD (Liquid Crystal Display) 30
2.9.7.1 โครงสร้างทั่วไปของ LCD 31
2.9.7.2 การควบคุมการแสดงผลของ LCD 31
2.9.7.3 การเชื่อมต่อระหว่างไมโครคอนโทรลเลอร์ กับ LCD Controller 32
2.9.7.4 คำสั่งสำหรับ LCD Controller เบอร์ HD44780 หรือเทียบเท่า 33
2.9.7.5 คำสั่ง LCD ของ LCD_4_BIT_Library 36
2.9.8 Stepping Motor 37
2.9.8.1 ประเภทของ Stepping Motor 37
2.9.8.2 หลักการทำงานสเต็ปมอเตอร์ 39
2.9.8.3 การควบคุมการหมุนของสเต็ปมอเตอร์ 39
2.9.9 DC Motor 41
2.9.9.1 ประเภทของ DC Motor 41
2.9.9.2 วิธีการควบคุมความเร็วของ DC Motor 42
2.9.10 Laser Diode 43
2.9.10.1 กลไกการเปล่งแสงของเลเซอร์ไดโอด 43
2.9.10.2 ลักษณะสมบัติกำลังเอาต์พุตและกระแสไฟฟ้า 44

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 3 การออกแบบการทดลอง

3.1 ขั้นตอนการออกแบบ Software 46
3.1.1 ขั้นตอนการทำงาน 46
3.1.2 การเชื่อมต่อกล้องเข้ากับโปรแกรม MATLAB 47
3.1.3 การเริ่มต้นใช้งานกล้องผ่าน MATLAB 50
3.1.4 calibration 50
3.1.5 การรับคำสั่งถ่ายภาพต่อเนื่อง 51
3.1.6 การปรับปรุงภาพ : การดึงองค์ประกอบสีแดง 52
3.1.7 ปรับปรุงภาพ : เลือกระดับความเข้ม 52
3.1.8 ปรับปรุงภาพ : การลดขนาดของเส้น 53
3.1.9 รวมภาพหลังจากการปรับปรุงภาพ 53
3.1.10 การหาตำแหน่ง xy จากจุดตัดภาพในภาพ 54
3.1.11 พล็อตสามมิติ 55
3.1.12 หน้าต่างโปรแกรม 56
3.2 การออกแบบด้านฮาร์ดแวร์ (Hardware) 56
3.2.1 หลักการออกแบบโดยรวม 56
3.2.2 การออกแบบวงจรแหล่งจ่ายไฟคงที่ 5 โวลต์ 57
3.2.3 วงจรรักษาแรงดันใช้ไอซีรักษาแรงดัน (IC Regulator) 58
3.2.4 การออกแบบวงจรแยกกราวน์ (Ground Isolation) 58
3.2.5 วงจรขับกระแส 58
3.2.6 การออกแบบวงจรแสดงผลจอ LCD แบบ 4 บิต 60
3.2.7 การออกแบบวงจร detect switch 61
3.2.8 การควบคุมการหมุนของมอเตอร์ 61
3.2.9 การออกแบบวงจร Analog to digital convertor (A-to-D) 61
3.2.10 การออกแบบวงจรขับเลเซอร์ 62

บทที่ 4 ผลการทดลอง

4.1 ผลการทดลองในส่วนของ Software 63
4.1.1 ขั้นตอนการทำงาน 64
4.1.2 ขั้นตอนหลังจากาการปรับปรุงภาพ 67
4.1.3 วัตถุที่ใช้ในการทดลอง 67
4.2 ผลการทดลองในส่วนของ Hardware 75

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 5 สรุปผลการทดลอง

5.1 สรุปผลการทดลอง 77
5.2 ปัจจัยต่างๆในการทดลอง 77
5.3 ปัญหาในการทดลอง 77
5.3 แนวทางแก้ปัญหา 78

บรรณานุกรม 79
ภาคผนวก ก ..... ก
ภาคผนวก ข ..... ฐ
ภาคผนวก ค ..... ต

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญตาราง

หน้า

ตารางที่ 1 ตารางแสดงบิตที่เกี่ยวข้องกับตำแหน่ง Cursor 35
ตารางที่ 2 การทำงานของสเต็ปมอเตอร์แบบเต็มสเต็ปหนึ่งเฟส..... 40
ตารางที่ 3 การทำงานของสเต็ปมอเตอร์แบบเต็มสเต็ปสองเฟส..... 40
ตารางที่ 4 การทำงานของสเต็ปมอเตอร์แบบครึ่งสเต็ป..... 40
ตารางที่ 5 รูปแบบการขับมอเตอร์แบบ Wave Step Drive..... 75

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูป

หน้า

รูปที่ 2.1 stack RGB 4
รูปที่ 2.2 ตัวอย่างการแยกองค์ประกอบของภาพสี 4
รูปที่ 2.3 ตัวอย่างการรวมองค์ประกอบของภาพสี โดยใช้คำสั่ง cat 5
รูปที่ 2.4 การแบ่งระดับความเข้ม 5
รูปที่ 2.5 ตัวอย่างการเลือกความเข็มของภาพที่ระดับความเข้มต่างกัน 6
รูปที่ 2.6 การโปรเจคชั่นแบบขนาน 7
รูปที่ 2.7 การโปรเจคชั่นแบบเพอสเปกทีพ 8
รูปที่ 2.8 แบบจำลองขบวนการสร้างภาพสเตอริโอ 9
รูปที่ 2.9 แบบจำลองของระบบสร้างภาพ 9
รูปที่ 2.10 เรขาคณิตของการเกิดภาพกรณีที่พิกัดของโลกไม่ซ้อนทับกับพิกัดของกล้อง 12
รูปที่ 2.11 ภาพกล่องที่ใช้สำหรับการ Calibration จากกล้องสองตัว 15
รูปที่ 2.12 วงจรจ่ายไฟตรงจากไฟบ้าน (Line operated DC Power Supply) 17
รูปที่ 2.13 ขั้วหรือเฟสของหม้อแปลง 19
รูปที่ 2.14 วงจรบริดจ์เรคติไฟเออร์ 20
รูปที่ 2.15 แสดงการทำงานของวงจรบริดจ์เรคติไฟเออร์ 21
รูปที่ 2.16 การทำงานของตัวเก็บประจุกรองไฟในสภาวะที่มีโหลด 22
รูปที่ 2.17 กราฟการทำงานของวงจรกรองแรงดันด้วยตัวเก็บประจุในสภาวะที่ไม่มี โหลด 23
รูปที่ 2.18 กราฟแสดงการทำงานของวงจรกรองแรงดันด้วยตัวเก็บประจุ ในสภาวะมีโหลด 23
รูปที่ 2.19 พื้นฐานในการต่อ IC Regulator 25
รูปที่ 2.20 วงจรภายใน IC 25
รูปที่ 2.21 วงจรรักษาระดับแรงดันแบบปรับค่าแรงดันเอาท์พุทได้ 26
รูปที่ 2.22 วงจรที่มีกราวน์ร่วมกัน 26
รูปที่ 2.23 สัญลักษณ์ออปโต้แบบต่างๆ 27
รูปที่ 2.24 ตัวอย่างการใช้งานออปโต้ทรานซิสเตอร์ 27
รูปที่ 2.25 บอร์ด ET-BASE AVR EASY88 28
รูปที่ 2.26 ซีพียู ATMEGA88P ขนาด 28 ขา 28
รูปที่ 2.27 หน้าตาโปรแกรมพัฒนา Arduino Board 30
รูปที่ 2.28 ตัวอย่างจอแสดงผล LCD 30
รูปที่ 2.29 การ Initial LCD แบบ 8 bit Mode 32
รูปที่ 2.30 การ Initial LCD แบบ 4 bit Mode 33
รูปที่ 2.31 โครงสร้างของ PM Step Motor 38
รูปที่ 2.32 โครงสร้างของ VR Step Motor 38
รูปที่ 2.33 โครงสร้างของ Hybrid Step Motor 39
รูปที่ 2.34 โครงสร้างของ Shunt Motor 41

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งไม่อนุญาตให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 2.35 โครงสร้างของ Series Motor 42
รูปที่ 2.36 โครงสร้างของ Permanent Magnet Motor 42
รูปที่ 2.37 ตัวอย่างเลเซอร์ไดโอด 43
รูปที่ 2.38 รอยต่อพีเอ็นที่กำเนิดแสงเลเซอร์ในเลเซอร์ไดโอด 44
รูปที่ 2.39 ลักษณะความสัมพันธ์ของความเข้มแสงและกระแสไฟฟ้าในเลเซอร์ไดโอด 44
รูปที่ 2.40 ลักษณะสมบัติกระแสไฟฟ้าและแรงดันของเลเซอร์ไดโอด 45
รูปที่ 2.41 ลักษณะสมบัติกำลังของเลเซอร์เอาต์พุตและกระแสที่ฉีดเข้าเลเซอร์ไดโอด 45
รูปที่ 3.1 แสดงการทำงานส่วนซอฟแวร์ 47
รูปที่ 3.2 ภาพที่ได้จากกล้องที่เราเรียกใช้ 49
รูปที่ 3.3 กล่องลูกบาศก์ที่ใช้ในการปรับเทียบกล้อง 50
รูปที่ 3.4 การ Reproject หลังจากทำการปรับเทียบกล้อง 51
รูปที่ 3.5 ตัวอย่างการถ่ายภาพต่อเนื่องทั้งหมด 9 ภาพ 51
รูปที่ 3.6 การเลือกระดับความเข้ม ภาพ (A) ระดับความเข้มที่ 250, ภาพ (B) ระดับความเข้มที่ 100 52
รูปที่ 3.7 หลังการปรับปรุงภาพ : ภาพต้นฉบับที่ได้จากการถ่ายภาพ 53
รูปที่ 3.8 การ imadd กล่องกับภาพเส้นทั้งหมด 54
รูปที่ 3.9 เครื่องหมาย + แสดงการคลิ๊กเมาส์ลงบนจุดตัด 54
รูปที่ 3.10 การพล็อตสามมิติ จากการคลิ๊กเมาส์จำนวน 25+8 ครั้ง 55
รูปที่ 3.11 หน้าต่างโปรแกรมที่ใช้สำหรับการถ่ายรูปต่อเนื่องและปรับปรุงภาพ 56
รูปที่ 3.12 บล็อกไดอะแกรมการทำงานโดยรวม 56
รูปที่ 3.13 พื้นฐานในการต่อ IC Regulator 57
รูปที่ 3.14 การใช้งานออปโต้ทรานซิสเตอร์ 58
รูปที่ 3.15 วงจรขับกระแสโดยใช้ทรานซิสเตอร์ 58
รูปที่ 3.16 การใช้ IC 7815 ช่วยจ่ายกระแสให้กับมอเตอร์ 59
รูปที่ 3.17 วงจรแสดงผลจอ LCD Module 60
รูปที่ 3.18 วงจรควบคุม switch 60
รูปที่ 3.19 วงจร A-to-D 61
รูปที่ 3.20 วงจรขับเลเซอร์ 62
รูปที่ 4.1 หน้าต่างโปรแกรมที่ใช้สำหรับการถ่ายรูปต่อเนื่องและปรับปรุงภาพ 63
รูปที่ 4.2 การถ่ายภาพวัตถุ 64
รูปที่ 4.3 ถ่ายภาพจากการกดปุ่ม Capture one 64
รูปที่ 4.4 ถ่ายภาพจากการกดปุ่ม Capture two 65
รูปที่ 4.5 การปรับปรุงภาพ 65
รูปที่ 4.6 การรับภาพต่อเนื่องจำนวน 20 ภาพ 66
รูปที่ 4.7 ภาพต่อเนื่องหลังจากการปรับปรุงภาพดึงองค์ประกอบสีแดงและเลือกระดับความเข้มของแสง 66
รูปที่ 4.8 วัตถุชิ้นที่ 1 กล่องสี่เหลี่ยม 67
รูปที่ 4.9 การ imadd ทุกรูปเข้าด้วยกัน 67

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งไม่อนุญาตให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 4.10 การคลิ๊กมุมของกล่องทั้งแปดและการคลิ๊กลงบนจุดตัดที่เกิดขึ้น 68
รูปที่ 4.11 การพล็อตสามมิติ วัตถุชิ้นที่ 1 68
รูปที่ 4.12 วัตถุชิ้นที่ 2 กระดาษแข็งที่มีช่องว่าง 69
รูปที่ 4.13 การ imadd ทุกรูปเข้าด้วยกัน 69
รูปที่ 4.14 การคลิ๊กมุมของกล่องทั้งแปดและการคลิ๊กลงบนจุดตัดที่เกิดขึ้น 70
รูปที่ 4.15 การพล็อตสามมิติ วัตถุชิ้นที่ 2 70
รูปที่ 4.16 วัตถุชิ้นที่ 3 ท่อน้ำทรงกระบอก 71
รูปที่ 4.17 การ imadd ทุกรูปเข้าด้วยกัน 71
รูปที่ 4.18 การคลิ๊กมุมของกล่องทั้งแปดและการคลิ๊กลงบนจุดตัดที่เกิดขึ้น 72
รูปที่ 4.19 การพล็อตสามมิติ วัตถุชิ้นที่ 3 72
รูปที่ 4.20 วัตถุชิ้นที่ 4 ท่อน้ำขนาดไม่เป็นทรงกระบอก 73
รูปที่ 4.21 การ imadd ทุกรูปเข้าด้วยกัน 73
รูปที่ 4.22 การคลิ๊กมุมของกล่องทั้งแปดแและการคลิ๊กลงบนจุดตัดที่เกิดขึ้น 74
รูปที่ 4.23 การพล็อตสามมิติ วัตถุชิ้นที่ 4 74

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 1
บทนำ

1.1 ความเป็นมาและความสำคัญของปัญหา

คำว่า “สะดวก รวดเร็ว ทันความต้องการของผู้ใช้” มาพร้อมกับคำว่าพัฒนาอุตสาหกรรมให้มีความก้าวหน้า การจำลองรูปทรง 3 มิติเป็นเรื่องหนึ่งที่มีการพัฒนาและ มีความต้องการการใช้งานด้านการจำลองรูปทรง 3 มิติ เพิ่มมากขึ้น ในกระบวนการจำลองรูปทรงชิ้นงานขึ้นมาจากต้นแบบมีหลายวิธีการที่สามารถทำการจำลองรูปทรงขึ้นมา จากต้นแบบที่มาจากวัสดุหลายชนิดทำให้การจำลองรูปทรงวัตถุแต่ละชนิดพื้นผิวของวัตถุมีข้อจำกัดในการใช้เครื่องมือหนึ่ง เครื่องมือหนึ่งเครื่องกับความสามารถในการจำลองรูปทรงของวัตถุที่มีพื้นผิวต่างกันได้หมดทุกสภาพพื้นผิว

การที่เรานำเลเซอร์มาใช้การจำลองรูปทรงของวัตถุ เป็นวิธีที่ไม่ต้องคำนึงถึงลักษณะของพื้นผิววัตถุที่นำมาทำการจำลอง ด้วยคุณสมบัติของแสงเลเซอร์ที่มีความเข้มแสงมาก ซึ่งทำให้สามารถตรวจจับความเข้มแสงได้ง่าย และรวมไปถึงการใช้แสงเลเซอร์ในการสแกนรูปทรงของวัตถุไม่กระทำให้พื้นผิววัตถุเสียหายจากการสแกนแสงเลเซอร์ลงบนพื้นผิวของวัตถุ การจำลองรูปทรงของวัตถุเป็นเทคโนโลยีสแกนเนอร์แบบไม่สัมผัสพื้นผิวของวัตถุ โดยข้อมูลที่ได้จาการสแกนวัตถุสามารถบันทึกและเก็บข้อมูลลงในระบบคอมพิวเตอร์และนำข้อมูลที่ได้ไปประมวลผลและพัฒนารูปทรงต่อไป

โครงงานชิ้นนี้ทำขึ้นเพื่อศึกษาหลักการใช้เลเซอร์ในการสแกนรูปทรงวัตถุ เพื่อจำลองรูปทรงของวัตถุขึ้นมา โดยใช้เลเซอร์สแกนลงบนพื้นผิวของวัตถุ และใช้กล้องจับภาพในขณะที่แสงเลเซอร์กวาดไปบนชิ้นงาน ภาพที่ได้จะเป็นองค์ประกอบสีแดง ที่มีแสงเลเซอร์อยู่บนพื้นผิวของวัตถุ โดยแสงเลเซอร์ที่ปรากฏทั้งหมดจะเป็นพื้นผิวของวัตถุ

1.2 วัตถุประสงค์ของโครงงาน

1.2.1 เพื่อศึกษาหลักการทำงานของเทคโนโลยีสแกนเนอร์ 3 มิติ

1.2.2 เพื่อศึกษาขั้นตอนการทำงานควบคู่กันระหว่างการทำงานเก็บข้อมูลภาพของวัตถุกับกาควบคุมการทำงานของมอเตอร์

1.2.3 เพื่อศึกษาการนำข้อมูลของภาพวัตถุมาประมวลผลภาพให้เป็นรูปทรง 3 มิติ

1.3 ขอบเขตของโครงงาน

1.3.1 การทำงานในการเก็บข้อมูลรูปภาพวัตถุและการปรับปรุงภาพ ใช้โปรแกรม MATLAB

1.3.2 ใช้โปรแกรม ardunio ในการควบคุมการทำงานของมอเตอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1.4 ประโยชน์ที่ได้รับ

1.4.1 สามารถปรับปรุงภาพโดยใช้โปรแกรมMATLAB ได้

1.4.2 สามารถติดต่อกล้องกับคอมพิวเตอร์โดยการใช้โปรแกรม MATLAB

1.4.3 มีความรู้ด้าน GUI (Graphic User Interface) ของโปรแกรม MATLAB

1.4.3 สามารถควบคุมมอเตอร์โดยการใช้ arduino

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 2
ทฤษฎีและหลักการทำงานที่เกี่ยวข้อง

เรื่อง เลเซอร์สแกนวัตถุ มีหลายทฤษฎีและหลักการทำงานที่เกี่ยวข้อง ได้แก่ หลักการการปรับปรุงภาพ (Image Enhancement) การโปรเจคชั่น (Projection) การปรับเทียบกล้อง (Camera Calibration) การทำงานของสเต็ปมอเตอร์ การปรับความเร็วของ DCมอเตอร์ และหลักการอื่นๆที่ใช้สำหรับการทำงานนี้

2.1 ชนิดของภาพ

2.1.1 ภาพดิจิตอล(Digital image)

ภาพดิจิตอลคือภาพ f(x,y) ซึ่งถูกทำให้เป็นค่าดีสครีต ทั้งค่าพิกัดและค่าความสว่าง การแทนข้อมูลความสว่างด้วยเลขขนาด 8 บิต ค่าระดับเทาที่ “0” แทนความสว่างน้อยหรือ สีดำ ค่าระดับเทาที่ “255” แทนความสว่างมากหรือสีขาว ภาพดิจิตอลขนาด M ×N ซึ่งค่า อินเด็กซ์ของเมทริกซ์ (แถวและคอลัมน์) ระบุจุดในภาพและค่าของเมทริกซ์ที่ตำแหน่ง ดังกล่าวแทนค่าระดับเทาที่จุดนั้น เราเรียกหน่วยที่เล็กที่สุดของภาพดิจิตอลว่า “พิกเซล (Pixel)” สำหรับกรณี ภาพดิจิตอลสีขนาด M×N เราใช้เมทริกซ์ขนาด M×Nจำนวน 3 เมทริกซ์สำหรับ R G และ B ในการแทนค่าสีที่จุดใดๆ โดยที่ค่าสีที่จุดใดๆซึ่งเกินจากการผสม สีจะได้มาจากค่าในเมทริกซ์ทั้งสาม ตัวอย่างเช่นค่าในเมทริกส์ R G และB สำหรับสีแดงคือ (255,0,0) เป็นต้น

2.1.2 ภาพความเข้มแสง(Intensity image)

ภาพความเข้มแสงคือเมทริกซ์ของข้อมูลซึ่งค่าถูกสเกลลงให้เป็นค่าความเข้มแสง เมื่อ สมาชิกของค่าในภาพความเข้มแสงเป็นชนิด uint8 หรือ uint16 ค่าของภาพจะอยู่ใ น ช่ ว ง [0,255] และ [0,65535] ตามลำดับ ถ้าค่าในภาพความเข้มแสงชนิด double ค่า ของภาพจะอยู่ในช่วง [0,1]

2.1.3 ภาพไบนารี(Binary image)

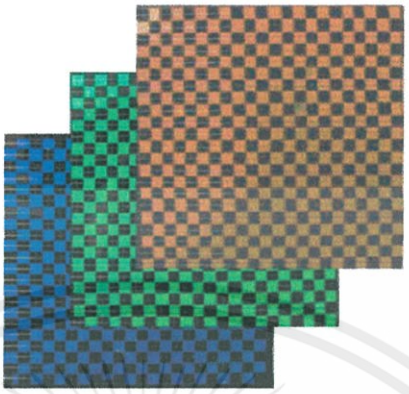
ภาพไบนารีมีความหมายที่เฉพาะสำหรับ Matlab ภาพไบนารีคืออาร์เรย์ของข้อมูลแบบตรรกะ (Logical Array) ที่มีค่า 0 หรือ 1 ภาพของข้อมูลชนิด uint8 ที่มีค่าตั้งแต่ 0 หรือ 1 ไม่ถือว่าเป็นภาพไบนารี

2.1.4 ภาพสี (RGB image)

ภาพสี RGB คืออาร์เรย์ขนาด M × N × 3 ของพิกเซลสี โดยที่แต่ละพิกเซลสีคือค่า Triplet สอดคล้องกับองค์ประกอบสีแดง เขียว และน้ำเงินของภาพ RGB ที่ตำแหน่งระบุ ภาพสีอาจจะมองว่าเป็นชั้น (Stack) ของภาพระดับเทาสามภาพ ที่ถูกป้อนให้กับอินพุทของจอมอนิเตอร์สี เพื่อทำให้เกิดสีบนจอภาพ โดยมาตรฐานแล้วภาพทั้งสามที่เป็นส่วนของภาพสี หมายถึงภาพองค์ประกอบสีแดงเขียวและน้ำเงิน คลาสของข้อมูลของภาพองค์ประกอบเป็นตัวกำหนดช่วงของค่า ถ้าภาพสีเป็นคลาส double ค่าจะอยู่ในช่วง [0,1] และถ้าภาพสีเป็นคลาส Uint8 ค่าจะอยู่ในช่วง [0,255] จำนวนบิตที่ใช้แทนค่าของพิกเซลของภาพองค์ประกอบเป็นตัวกำหนด Bit Depth ของภาพสี ตัวอย่าง ถ้าแต่ละภาพองค์ประกอบคือภาพขนาด 8 บิต ภาพสี ถูกเรียกว่ามีค่า Bit Depth

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งไม่อนุญาตให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ที่ 24 โดยทั่วไป จำนวนบิตในทุกองค์ประกอบของภาพมีค่าเดียวกัน ในกรณีนี้จำนวนของสีในภาพสี คือ $(2^b)^3$ โดยที่ b คือจำนวนบิตในแต่ละภาพองค์ประกอบ



รูปที่ 2.1 stack RGB

ในโปรแกรมMATLAB ให้ fR, fG และ fB แทนภาพองค์ประกอบของภาพสี RGB ภาพสีสามารถถูกสร้างโดยใช้คำสั่ง car(concatenate) เพื่อสร้างเป็นชั้น (Stack)

rgb_image = cat (3, fR, fG , fB);

ถ้าภาพองค์ประกอบทั้งสามเหมือนกันโดยมีความเข้มภาพมีค่ามาก ภาพที่ได้จะเข้าใกล้สีดำ ถ้าภาพองค์ประกอบทั้งสามเหมือนกันโดยความเข้มภาพมีค่าน้อย ภาพที่ได้จะเข้าใกล้สีขาว ถ้าภาพองค์ประกอบทั้งสามเหมือนกันโดยความเข้มภาพมีค่า125ภาพที่ได้เป็นภาพระดับเทา ภาพที่ได้เป็นภาพระดับเทา (Gray-Scale image) ให้ rgb_image แทนภาพสี RGB ในการแยกองค์ประกอบของภาพเราใช้คำสั่ง

```
fR=rgb_image (: , : , 1);
fG=rgb_image (: , : , 2);
fB=rgb_image (: , : , 3);
```

b1=b (: , : , 1); b2=b (: , : , 2); b3=b (: , : , 3);

รูปที่ 2.2 ตัวอย่างการแยกองค์ประกอบของภาพสี

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

b = cat (3, b1, b2, b3);

รูปที่ 2.3 ตัวอย่างการรวมองค์ประกอบของภาพสี โดยใช้คำสั่ง cat

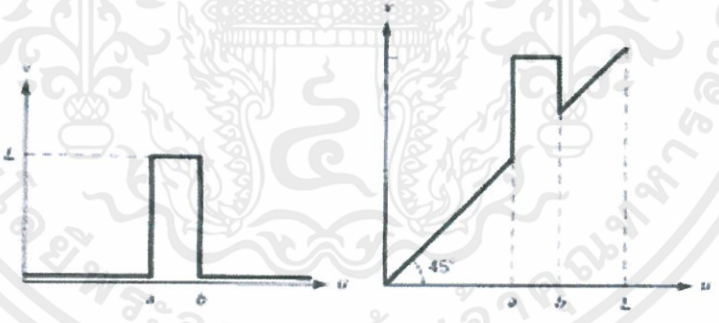
2.2 การปรับปรุงภาพ (Image Enhancement)

ในส่วนของการประมวลผลภาพดิจิตอล เป็นการปรับปรุงภาพดิจิตอลในส่วนที่เกี่ยวข้องกับการแบ่งระดับความเข้ม(Intensity Level Slicing) การปรับปรุงขนาดของเส้นบนภาพที่เราได้จากการถ่ายภาพ

2.2.1 การแบ่งระดับความเข้ม (Intensity Level Slicing)

การแบ่งระดับความเข้มใช้ในการเน้นส่วนหนึ่งส่วนใดของภาพ ซึ่งแบ่งเป็น 2 กรณีคือ

1) กรณีที่ไม่มีพื้นหลัง (without background)
2) กรณีที่มีพื้นหลัง (with background)



กรณีมีพื้นหลัง กรณีไม่มีพื้นหลัง

รูปที่ 2.4 การแบ่งระดับความเข้ม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การแปลงนี้จะเป็นการแบ่งออกเป็นส่วน(Segmentation) สำหรับบริเวณที่มีระดับเทาค่าหนึ่งออกจากบริเวณอื่นของภาพ เทคนิคนี้มีประโยชน์เมื่อลักษณะเฉพาะ (Feature) ต่างๆของภาพกระจายอยู่ในระดับเทาต่างๆ และเราต้องการแยกลักษณะเฉพาะนั้นๆ

รูปที่ 2.5 ตัวอย่างการเลือกความเข้มของภาพที่ระดับความเข้มต่างกัน

2.2.2 การปรับปรุงขนาดของเส้น

การปรับปรุงขนาดของเส้นมีหลายวิธี แต่จะนำเสนอวิธีที่ใช้คำสั่งที่มีอยู่ใน Matlab คือ การใช้ฟังก์ชั่น (bwmorph) ซึ่งเป็นฟังก์เกี่ยวกับการกระทำกับภาพไบนารี โดยการใช้คำสั่งนี้ทำให้ขนาดของเส้นมีขนาดลดลงหรือเปลี่ยนแปลงไป ซึ่งการเปลี่ยนแปลงขนาดของเส้นจะแตกต่างกันโดยจะขึ้นอยู่กับการระบุชนิดของการลดขนาดในคำสั่ง เช่น thin (ทำให้ขนาดของเส้นบางลง) fill (การเติมเต็มบางส่วนของเส้น)

2.3 การติดต่อกล้อง USB ผ่าน Image Acquistion

การติดต่อกล้อง USB กับ โปรแกรม MATLAB เป็นการเรียกใช้กล้องที่มีอยู่ในระบบ ซึ่งการติดต่อกล้องในระบบแต่ละครั้ง อาจจะเป็นกล้องที่มีอยู่ในตัวเครื่องคอมพิวเตอร์ หรือว่ากล้อง USB ที่เราได้ทำการเพิ่มเข้าไปในระบบ โดยการใช้กล้องแต่ละครั้งเราต้องเลือกชนิดของกล้องที่เราต้องทำการติดต่อ

โปรแกรม MATLAB มีคำสั่งเรียกใช้กล้องที่มีอยู่ในระบบ ดังนี้

1) ต้องทำการตรวจสอบก่อนว่ามีกล้องอยู่ในระบบ โดยใช้คำสั่ง

```
>>imaqhwinfo
```

2) ตรวจสอง ID ของอุปกรณ์ที่ต้องการใช้ ในกรณีนี้ใช้กล้อง USB ให้พิมพ์

```
>>Info = imaqhwinfo('winvideo')
```

3) ตรวจดูรูปแบบของสัญญาณวิดีโอที่ใช้ โดยใช้ DeviceID

```
>>Device_info = imaqhwinfo('winvideo',1) ; ตัวเลข 1 แทนกล้องตัวที่ 1
```

4) สร้างอ๊อบเจ็คสำหรับสัญญาณวิดีโออินพุทโดยใช้คำสั่ง

```
>>vid = videoinput('winvideo', '1' , 'RGB24_320x240' )
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของผลงานทุกครั้งที่มีการนำไปใช้

5) ตรวจดูคุณสมบัติของอ็อบเจ็คโดยใช้คำสั่ง

```
>>get(vid)
```

6) ตรวจสอบความถูกต้องของการติดตั้งด้วยคำสั่ง

```
>>Preview(vid)
```

โดยที่เราสามารถกำหนดชื่อของกล้องแต่ละตัวได้ตามต้องการ คำที่เราสามารถกำหนดชื่อตามที่เราต้องการแทนได้ คือ *info Device_info vidg(เป็นชื่อประจำตัวกล้องแต่ละตัวที่เราเรียกใช้แต่ละครั้ง)* ซึ่งโดยปกติเราจะกำหนดชื่อให้มีความสอดคล้องกัน เพื่อความสะดวกในการเรียกใช้ตัวกล้องหรือตรวจสอบคุณสมบัติของกล้องตัวนั้นๆ

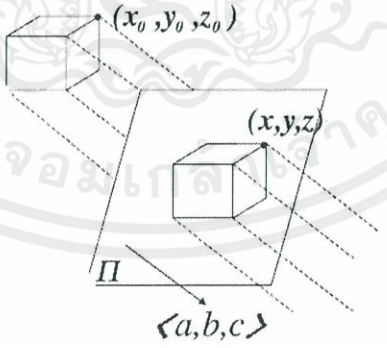
2.4 การโปรเจคชั่น(Projections)

การโปรเจคชั่นเป็นการแปลงจาก n มิติ ไปเป็น n-1 มิติ เมื่อสมมติให้มีวัตถุ 3 มิติที่ถูกโปรเจค (Projected) ลงบนระนาบโปรเจคชั่น (Projection Plane) โดยมีผู้สังเกตลักษณะการโปรเจคของวัตถุ และเพื่อให้สามารถมองลักษณะการโปรเจคได้จึงต้องกำหนดให้วัตถุและผู้สังเกตอยู่คนละข้างของระนามโปรเจคชั่น

โปรเจคชั่นสามารถแบ่งได้เป็นแบบเชิงเส้น (Linear Projections) และแบบไม่เป็นเชิงเส้น (Non-linear Projections) โดยที่โปรเจคชั่นแบบเชิงเส้นประกอบด้วย 2 ชนิด คือ โปรเจคชั่นแบบขนาน (Parallel Projections) และ โปรเจคชั่นแบบเพอสเปกทีพ (Perspective Projection)

2.4.1 โปรเจคชั่นแบบขนาน (Parallel Projections)

การโปรเจคชั่นแบบขนาน คือ การฉายแสงไปตามทิศทางที่กำหนดไว้ให้ปรากฏเป็นจุดใหม่บนระนาบโปรเจคชั่นซึ่งเป็น 2 มิติ โดยที่แสงและเงาของวัตถุที่ฉายจะพุ่งขนานกันไป กระทบกับระนาบของการโปรเจคและขนาดของรูปที่ได้จากการโปรเจคชั่นแบบนี้จะมีขนาดเท่ากับขนาดของวัตถุ ส่วนใหญ่ โปรเจคชั่นแบบขนานนี้ถูกใช้ในการวาดภาพทางวิศวกรรม



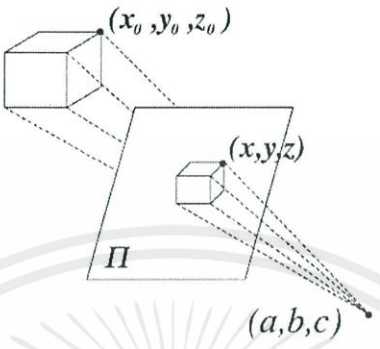
รูปที่ 2.6 การโปรเจคชั่นแบบขนาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของผลงานทุกครั้งที่มีการนำไปใช้

2.4.1 โปรเจคชั่นแบบเพอสเปกทีพ(Perspective Projection)

การโปรเจคชั่นแบบเพอสเปกทีพ เป็นการมองเห็นวัตถุในลักษณะเดียวกับโลกของความเป็นจริง เหมือนที่ตามนุษย์มองเห็น โปรเจคชั่นแบบนี้มีจุดศูนย์กลางของการโปรเจค

(Center of Projection) อยู่ที่จุดๆหนึ่งแนวลำแสงและเงาของวัตถุที่ฉายจะพุ่งเข้าไปยังจุดศูนย์กลางของการโปรเจคมีลักษณะเป็นสามเหลี่ยม คือ ผู้สังเกตจะเห็นวัตถุที่อยู่ไกลมีขนาด เล็กกว่าความเป็นจริง เนื่องจากขนาดของวัตถุเปลี่ยนแปลงโดยมีความสัมพันธ์แบบผกผันกับ ระยะห่างระหว่างวัตถุกับจุดศูนย์กลางของการโปรเจค



รูปที่ 2.7 การโปรเจคชั่นแบบเพอสเปกทีพ

2.5 การเกิดภาพสเตอริโอ(stereo image)

การเกิดภาพสเตอริโอเป็นการสร้างภาพของวัตถุเดียวกันด้วยกล้อง 2 ตัววางอยู่ข้างกัน ภาพที่ได้จากกล้องทั้งสองสามารถนำมาใช้การหาพิกัด 3 มิติของวัตถุได้ พิจารณาแบบจำลองการเกิดภาพสเตอริโอ ระยะระหว่างจุดกึ่งกลางของเลนส์ของกล้องทั้งสองเราเรียกว่า Base line B เราต้องหาพิกัด (X,Y,Z) ของจุด w ซึ่งมีจุดภาพ (x1,y1) และ (x2,y2) เราสมมติว่ากล้องทั้งสองมีลักษณะที่เหมือนกัน และระบบพิกัดของกล้องทั้งสองนั้นวางอยู่ในแนวที่ตรงกัน (Perfectly Aligned) ถ้าระบบพิกัดของกล้องและของโลกซ้อนทับกันพอดีระนาบ xy ของภาพวางอยู่ในแนวเดียวกับระนาบ xy ของโลก ดังนั้นค่าพิกัด Z ของ w สำหรับระบบพิกัดกล้องทั้งสองมีค่าเดียวกัน

ถ้าระบบพิกัดของกล้องที่หนึ่งและของโลกซ้อนทับกันพอดี เราได้

$$X_1=\frac{x_1}{f}(f-Z_1) \tag{2.1}$$

โดยที่ตัวห้อยบน X และ Z เป็นการบอกว่าพิกัดกล้องที่หนึ่งซ้อนทับกับพิกัดของโลก ทำนองเดียวกันถ้าพิกัดของกล้องที่สองและของโลกซ้อนทับกันพอดี เราได้ว่า

$$X_2=\frac{x_2}{f}(f-Z_2) \tag{2.2}$$

เนื่องจากการแยกกันของกล้องทั้งสอง และเนื่องจาก w เป็นค่าเดียวสำหรับระบบพิกัดของทั้งสองกล้อง เราได้ว่า

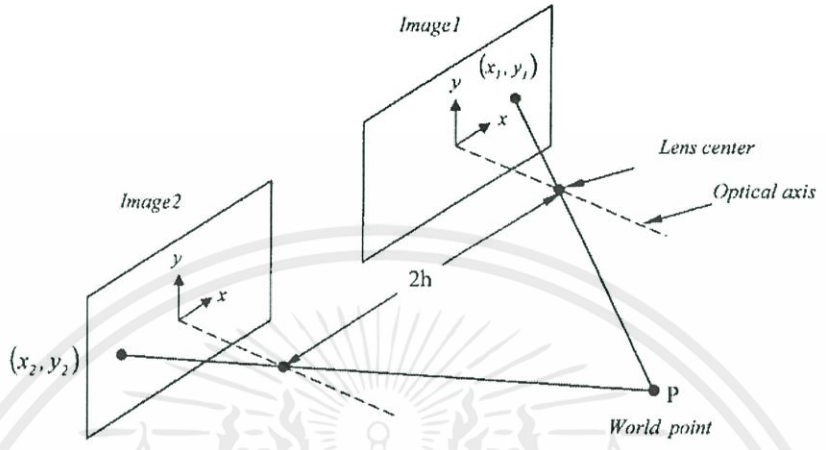
$$X_2=X_1+B \tag{2.3}$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

และ

$$Z_2=Z_1=Z \tag{2.4}$$

โดยที่ B เป็นระยะ Baseline

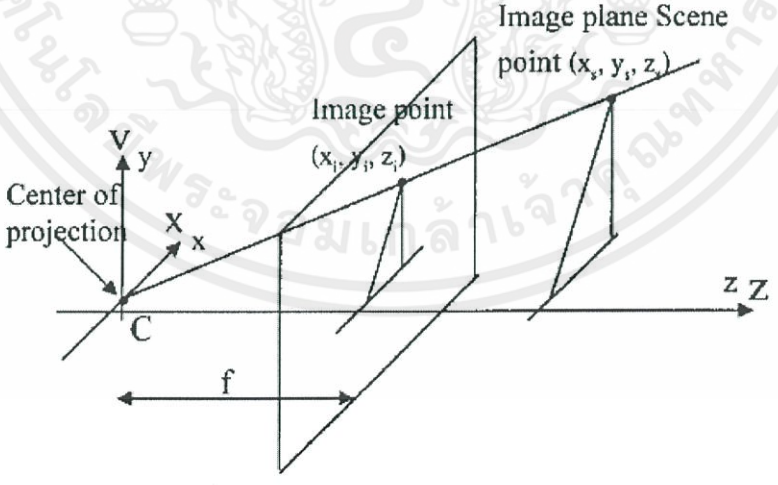


รูปที่ 2.8 แบบจำลองขบวนการสร้างภาพสเตอริโอ

2.6 เรขาคณิตสำหรับการเห็นใน 3มิติ (Geometry for 3D vision)

2.6.1 พื้นฐานของการโปรเจคชันแบบเพอสเปกทีฟของกล้อง (Basis of Perspective Camera)

ภาพ 2 มิติ ส่วนใหญ่ได้มาจากการมองวัตถุ 3 มิติ ผ่านเลนส์ของตาหรือเลนส์ของกล้อง ขบวนการเกิดภาพสามารถถูกจำลองได้โดยการใช้การแปลงแบบเพอสเปกทีฟ (Perspective Transformation)



รูปที่ 2.9 แบบจำลองของระบบสร้างภาพ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เราแทนแกนระบบพิกัดโลก (World Coordination System) ด้วยตัวอักษรใหญ่ (X, Y, Z) และแทนแกนของระบบพิกัดกล้องด้วยตัวอักษรเล็ก (x, y, z) เราสมมติว่าพิกัดของโลกซ้อนทับกับระบบพิกัดของกล้อง แสงจากวัตถุใน 3 มิติ ส่องผ่านศูนย์ของการโปรเจ็คชันและโปรเจ็คชันลงบนภาพให้จุด (x_s , y_s , z_s) แทนจุดพิกัดของวัตถุ 3 มิติ ให้ (x_i, y_i) แทนจุดพิกัดของวัตถุในระนาบของภาพ เราสมมติว่าระนาบของภาพอยู่ f ห่างไปทางขวาของจุดกำเนิดของพิกัดกล้องโดยที่ f คือความยาวโฟกัสของเลนส์ โดยการใช้สามเหลี่ยมคล้ายเราได้ว่า

$$x_i = \frac{fx_s}{Z_s} \tag{2.5}$$

$$y_i = \frac{fy_s}{Z_s} \tag{2.6}$$

ดังนั้นจุดที่โปรเจ็ค (x_i, y_i) มีความสัมพันธ์ไม่เป็นเชิงเส้นกับจุดของวัตถุ (x_s , y_s , z_s) ความสัมพันธ์นี้ สามารถมองเห็นได้ง่ายถ้าเราใช้ระบบพิกัดโฮโมจีเนียส
ให้

$$\tilde{v} = \begin{bmatrix} x_s \\ y_s \\ z_s \\ 1 \end{bmatrix} \tag{2.7}$$

เป็นเวคเตอร์ที่ประกอบด้วยจุดพิกัดของวัตถุ เวคเตอร์ในระบบโฮโมจีเนียส เวคเตอร์ในระบบพิกัด Catesian v สามารถได้จากเวคเตอร์ในระบบพิกัดโฮโมจีเนียสได้โดย การหารสามสมาชิกแรกด้วยสมาชิกที่สี่

พิจารณาเมทริกซ์ของการแปลงเพอสเปกทีพ (Perspective Transformation) ในอีกรูปหนึ่ง

$$P = \begin{bmatrix} f & 0 & 0 & 0 \\ 0 & f & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \tag{2.8}$$

ทำการคูณเวคเตอร์

$$\tilde{w} = P\tilde{v} \tag{2.9}$$

ได้เป็น

$$\tilde{w} = \begin{bmatrix} fx_s \\ fy_s \\ z_s \end{bmatrix} \tag{2.10}$$

ทำการนอร์มอไลซ์สมาชิกที่สามให้เป็น 1 เพื่อแปลงเป็นเวคเตอร์ในระพิกัด Cartesian
ดังนั้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

$$w=\begin{bmatrix}\frac{fx_s}{z_s}\\ \frac{fy_s}{z_s}\end{bmatrix} \tag{2.11}$$

สังเกตุว่าสองสมาชิกแรกของ w สอดคล้องกับพิกัดของภาพที่ได้ในสมการ (2.5) และ (2.6)

2.6.2 การแปลงจากหน่วยความยาวเป็นพิกเซล

ต่อไปคือการหาพิกัดของจุดที่ถูกโปรเจคชัน (x_s , y_s , z_s) ในหน่วยของพิกเซล กำหนดให้พิกัดของภาพมีจุดกำเนิดอยู่ที่มุมล่างซ้ายของภาพ การแปลงจากความยาวเป็นพิกเซลเราจำเป็นต้องรู้จักค่าการปรับสเกล (Aspect Ratio) ของแต่ละแกนของระบบพิกัดภาพ ค่าการปรับสเกลเป็นค่าที่ระบุจำนวนพิกเซลต่อหนึ่งหน่วยความยาว ในระบบภาพบางระบบ พิกเซลในแถวถัดไปอาจมีการเยื้องกัน จำเป็นต้องรู้ค่าที่เรียกว่าพารามิเตอร์ของการเฉือน (Skew parameter) นอกจากนี้ยังจำเป็นต้องรู้จุดหลัก (Principal Point) $U_0 = (x_0 , y_0)$ หรือบางครั้งถูกเรียกว่าจุดกึ่งกลางของภาพ

ดังนั้นพิกัดของจุดในพิกเซล (x_i ,y_i ,f) สามารถหาได้จากความสัมพันธ์

$$\begin{aligned} x_{pix} &= k_x x_i + s y_i + x_0 \\ &= k_x f\frac{x_s}{z_s} + sf\frac{y_s}{z_s} + 0 \end{aligned} \tag{2.12}$$

โดยที่ k_x คือ ค่าการปรับสเกลในทิศทาง x , k_y คือ ค่าการปรับสเกลในทิศทาง y และ s แทนพารามิเตอร์ของการเฉือน

เราสามารถทำให้อยู่ในรูปของเมทริกซ์ได้เป็น

$$\begin{bmatrix}u'\\ v'\\ w'\end{bmatrix} = \begin{bmatrix}\alpha_x & s & u_0 & 0\\ 0 & \alpha_y & v_0 & 0\\ 0 & 0 & 1 & 0\end{bmatrix}\begin{bmatrix}x_s\\ y_s\\ z_z\\ 1\end{bmatrix} \tag{2.13}$$

โดยที่ $\alpha_x = k_x f$ ซึ่งเป็นค่าความยาวโฟกัสในทิศ x ในหน่วยของพิกเซลและ $\alpha_y = k_y f$ f ซึ่งเป็นค่าความยาวโฟกัสในทิศ y ในหน่วยของพิกเซล เราสามารถเขียนได้เป็น

$$\begin{bmatrix}\alpha_x & s & u_0 & 0\\ 0 & \alpha_y & v_0 & 0\\ 0 & 0 & 1 & 0\end{bmatrix} = \begin{bmatrix}\alpha_x & s & u_0\\ 0 & \alpha_y & v_0\\ 0 & 0 & 1\end{bmatrix}\begin{bmatrix}1 & 0 & 0 & 0\\ 0 & 1 & 0 & 0\\ 0 & 0 & 1 & 0\end{bmatrix} = \kappa\,[I_3|O_3] \tag{2.14}$$

เมทริกซ์ K มีลักษณะเป็นสามเหลี่ยมข้างบน (Upper Triangle) ขนาด 3x3 มีชื่อว่า เมทริกซ์การเปรียบเทียบ (Calibration Matrix) เป็นเมทริกซ์ที่ประกอบพารามิเตอร์ภายใน (Intrinsic Parameter) ที่สำคัญของกล้อง

2.6.3 กรณีที่พิกัดของโลกไม่ซ้อนทับกับพิกัดของกล้อง

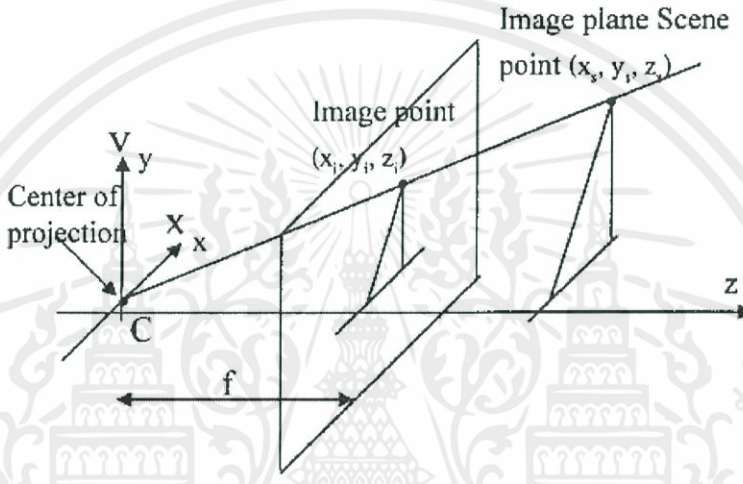
การหาพิกัดของภาพที่เกิดกรณีที่จุดกึ่งกลางของการโปรเจคชันของกล้องถูกวางไว้ที่จุดกึ่งกลางของระบบพิกัดอ้างอิงของโลก ไม่มีการส่ายหรือเอียงทำมุมกับแกนอ้างอิง จุดกึ่งกลางของ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งไม่อนุญาตให้นำไปใช้

หัวยึดกล้องและของระนาบภาพอยู่ที่เดียวกัน แบบจำลองการเกิดภาพ (Imaging Model) ในระบบพิกัดโฮโมจีเนียส คือ

$$\widetilde{W}=M\widetilde{V} \tag{2.15}$$

โดยที่ $\widetilde{V}$ คือ เวคเตอร์ของจุดบนวัตถุในระบบพิกัดโฮโมจีเนียส $\widetilde{W}$ คือ เวคเตอร์ของจุดบนระนาบภาพในระบบโฮโมจีเนียส และ M คือ เมทริกซ์ของ Perspective Transformation ในกรณีนี้แบบจำลองการเกิดภาพจากกล้องสามารถหาได้ง่ายโดยการเปลี่ยนแปลง (Modify) ของสมการ (3.5)



รูปที่ 2.10 เรขาคณิตของการเกิดภาพกรณีที่พิกัดของโลกไม่ซ้อนทับกับพิกัดของกล้อง

ระบบพิกัดโลกสามารถหมุนส่าย (Pan) ทำมุม β กับแนวนอน (Horizontal) และเอียง (Tilt) ทำมุม α กับแนวดิ่ง จุดกึ่งกลางของพิกัดโลกและจุดกึ่งกลางของการโปรเจคชันของระนาบภาพอยู่ห่างกันด้วยระยะ (X_0, Y_0, Z_0)

การที่จุดกึ่งกลางของพิกัดโลกและจุดกึ่งกลางของการโปรเจคชันของระนาบภาพอยู่ห่างกันด้วยระยะ (X_0, Y_0, Z_0) เราได้แบบจำลองใหม่ คือ

$$\widetilde{W}=MT_c\widetilde{V} \tag{2.16}$$

โดยที่

$$T_G=\begin{bmatrix}1 & 0 & 0 & -X_0\\ 0 & 1 & 0 & -Y_0\\ 0 & 0 & 1 & -Z_0\\ 0 & 0 & 0 & 1\end{bmatrix} \tag{2.17}$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น ทั้งนี้ ห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารนี้ทุกครั้งที่มีการนำไปใช้

การส่ายหรือเอียง (Pan and Tilt) สามารถถูกจำลองโดยใช้การคูณด้วยเมทริกซ์การหมุน

$$\widetilde{W} = MTT_c\widetilde{V} \tag{2.18}$$

โดยที่ $R = R_\beta R_\alpha$

$$R_\beta = \begin{bmatrix} \cos\beta & 0 & -\sin\beta & 0 \\ 0 & 1 & 0 & 0 \\ \sin\beta & 0 & \cos\beta & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{2.19}$$

และ

$$R_\beta = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos\alpha & \sin\alpha & 0 \\ 0 & -\sin\alpha & \cos\alpha & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{2.20}$$

ดังนั้นแบบจำลองของการเกิดภาพจากกล้องคือ

$$\widetilde{W}=MRT_c\widetilde{V} \tag{2.21}$$

โดยที่ สมการ (2.21) สามารถเขียนใหม่ได้เป็น

$$\begin{bmatrix} u' \\ v' \\ w' \end{bmatrix} = K[I_3|O_3] \begin{bmatrix} R & -T \\ O_3^T & 1 \end{bmatrix} \begin{bmatrix} x_s \\ y_s \\ z_z \\ 1 \end{bmatrix} \tag{2.22}$$

โดยที่ R แทนเมทริกซ์ที่เป็นผลรวมของการหมุนระบบพิกัด (ขนาด 3x3 ของระบบพิกัดธรรมดา) และ T แทนเวคเตอร์ที่เป็นผลรวมของการย้ายระบบพิกัด (ขนาด 3x1) เรากล่าวว่าเมทริกซ์

$$\begin{bmatrix} R & -T \\ O_3^T & 1 \end{bmatrix} \tag{2.23}$$

เป็นเมทริกซ์ที่รวมเอาพารามิเตอร์ภายนอก (Extrinsic Parameter) ของการปรับเทียบกล้อง เราสามารถเขียนสมการ (2.23) ในรูปแบบที่ง่ายได้เป็น

$$x = MX \tag{2.24}$$

โดยที่ $M=[KR \mid -KRT]$
และ M มีขนาด 3x4

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.7 การปรับเทียบกล้อง(calibration)

การปรับเทียบกล้องเป็นการหาประมาณแมทริกซ์ M จากจุดในฉาก 3 มิติ ที่รู้ตำแหน่งและภาพที่เกิดขึ้นของจุด จากนั้นทำการหาพารามิเตอร์ภายใน และ พารามิเตอร์ภายนอก พิจารณาการปรับเทียบกล้อง ในการหาเมทริกซ์ M เราทำการหาจุดในฉากที่เราทราบพิกัด $x = [x, y, z]^T$ และจุดในภาพ 2 มิติที่สอดคล้องกัน $[u,v]^T$ ซึ่งจะได้สมการที่อยู่ในรูปสมการดังนี้

$$\begin{bmatrix} \alpha u \\ \alpha v \\ \alpha \end{bmatrix} = \begin{bmatrix} m_{11} & m_{12} & m_{13} & m_{14} \\ m_{21} & m_{22} & m_{23} & m_{24} \\ m_{22} & m_{32} & m_{33} & m_{34} \end{bmatrix} \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix} \tag{2.25}$$

$$\begin{bmatrix} \alpha u \\ \alpha v \\ \alpha \end{bmatrix} = \begin{bmatrix} m_{11} & m_{12} & m_{13} & m_{14} \\ m_{21} & m_{22} & m_{23} & m_{24} \\ m_{22} & m_{32} & m_{33} & m_{34} \end{bmatrix} \tag{2.26}$$

ทำการนอร์มอลไลซ์ให้สมาชิกที่ 3 เป็น 1 จะได้สมการ

$$u(m_{31}x + m_{32}y + m_{33}z + m_{34}) = m_{11}x + m_{12}y + m_{13}z + m_{14} \tag{2.27}$$

$$u(m_{31}x + m_{32}y + m_{33}z + m_{34}) = m_{21}x + m_{22}y + m_{23}z + m_{24} \tag{2.28}$$

ดังนั้นเราจะได้สมการเชิงเส้น 2 สมการสำหรับจุดในฉาก 3 มิติหนึ่งจุดและจุดในภาพ 2 มิติที่สอดคล้องกันทำการเขียนสมการใหม่ในรูปของเมทริกซ์ได้เป็น

$$\begin{bmatrix} x & y & z & 1 & 0 & 0 & 0 & 0 & -ux & -uy & -uz & -u \\ 0 & 0 & 0 & 0 & x & y & z & 1 & -vu & -vy & -vz & -v \\ & & & & & & \vdots & & & & & \end{bmatrix} \begin{bmatrix} m_{11} \\ m_{12} \\ \vdots \\ m_{34} \end{bmatrix} = 0$$

$$AM=0 \tag{2.29}$$

จะเห็นได้ว่าเรามีตัวแปรที่ไม่ทราบค่า 11 ตัวแปรแทนที่จะเป็น 12 ตัวแปร เนื่องจากเฟคเตอร์ การย่อ/ขยายที่ไม่สามารถทราบค่าได้ ในการแก้สมการโฮโมจีเนียสเราต้องใช้อย่างน้อย 6 จุด ถ้ามีมากกว่า 6 จุดจะได้สมการ Over-determine ซึ่งสามารถแก้ได้โดยใช้วิธี Least Square หรือใช้การหา Singular Value Decomposition (SVD) ของ A เมทริกซ์ M ได้จากแถวสุดท้ายของ V

เมื่อได้เมทริกซ์ M ขั้นตอนต่อไปทำการแยกพารามิเตอร์ภายใน (Intrinsic Patameter) เนื่องจาก

$$M = [KR|-KRT] = [A|b] \tag{2.30}$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ดังนั้นเมทริกซ์ 3x3 แทนด้วย A เวคเตอร์คอลัมป์ขวามือสุดแทนด้วย b เวคเคอร์ที่มีพารามิเตอร์ระยะทางเคลื่อนที่สามารถแยกออกได้ง่ายโดย t=A-1B เวคเตอร์ t ให้ข้อมูลที่บอกถึงตำแหน่งกึ่งกลางของระนาบภาพ (Image Plane)

จากนั้นพิจารณา A=KR โดยที่ K เป็นเมทริกซ์สามาเหลี่ยมบน (Upper Triangle) และเมทริกซ์ R เป็นเมทริกซ์ออโธกอนอล การแยกเมทริกซ์ K และ R สามารถทำได้โดยใช้เทคนิกค QP Decomposition สำหรับเมทริกซ์ A

รูปที่ 2.11 ภาพกล่องที่ใช้สำหรับการ Calibration จากกล้องสองตัว

2.8 การพิกัด 3 มิติของจุดในฉากโดยใช้กล้อง 2 ตัว (3D Coordinate Extraction)

หัวข้อนี้เราพิจารณะการหาพิกัด 3 มิติ ของจุดในฉากโดยใช้กล้อง 2 ตัว โดยที่กล้องทั้งสองตัวผ่านขบวนการการปรับเทียบกล้อง ในขบวนการปรับเทียบกล้อง ให้กล้องทั้งสองถ่ายภาพวัตถุ ที่เราทราบพิกัด ซึ่งอาจจะใช้เป็นตารางหมากรุกหรือว่ากล้องที่เราทราบขนาดที่แน่นอน จากนั้นทำการหาพิกัดของจุดตัวในกล้องทั้งสอง ทำการคำนวณเมทริกซ์ของ Perspective Transformation M สำหรับแต่ละกล้อง ต่อไปทำการวางวัตถุที่เราต้องการพิกัด 3 มิติในฉาก ทำการหาจุดสอดคล้อง ของจุดในฉาก 3 มิติ ของกล้องทั้งสองตัว กำหนดให้เมทริกซ์ของPerspecion Transformation M ของกล้องตัวที่ 1 และ 2 เป็น M และ M' ตามลำดับ เราแทนแต่ละแถวของเมทริกซ์ M ด้วย m_1^T , m_2^T , m_3^T ทำนองเดียวกันสำหรับกล้องที่สองเราแทนแต่ละแถวของ เมทริกซ์ M' ด้วย $m_1'^T$, $m_2'^T$, $m_3'^T$

สำหรับกล้องที่ 1 เราได้

$$u=\begin{bmatrix}u\\v\\w\end{bmatrix}=\begin{bmatrix}m_{11} & m_{12} & m_{13} & m_{14}\\ m_{21} & m_{22} & m_{23} & m_{24}\\ m_{22} & m_{32} & m_{33} & m_{34}\end{bmatrix}\begin{bmatrix}x\\y\\z\\1\end{bmatrix}=MX=\begin{bmatrix}m_1^T\\m_2^T\\m_3^T\end{bmatrix}X \qquad (2.31)$$

สำหรับกล้องที่ 2 เราได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น ไม่อนุญาตให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

$$u' = \begin{bmatrix} u' \\ v' \\ w' \end{bmatrix} = M'X = \begin{bmatrix} m_1'^T \\ m_2'^T \\ m_3'^T \end{bmatrix} \tag{2.32}$$

โดยที่ u และ u' แทนพิกัดในภาพของวัตถุในกล้อง 1 และ 2 ตามลำดับ ในการกำจัดค่าสเกลที่ไม่ทราบค่า เราทำการหาสัดส่วนระหว่างสามแถวในเมทริกซ์ Perspective Transformation M

$$u : v : w = m_1^T X : m_2^T X : m_3^T X \tag{2.33}$$

$$u' : v' : w' = m_1'^T X : m_2'^T X : m_3'^T X \tag{2.34}$$

ดังนั้นเราได้สมการสำหรับกล้อง 1 และกล้อง 2

$$um_2^T = vm_1^T X \quad um_2'^T X = vm_1'^T X \tag{2.35}$$

$$um_3^T = vm_1^T X \quad um_3'^T X = vm_1'^T X \tag{2.36}$$

$$um_3^T = vm_2^T X \quad um_3'^T X = vm_2'^T X \tag{2.37}$$

สามารถเขียนได้เป็น

$$\begin{bmatrix} um_2^T - vm_1^T \\ um_3^T - wm_1^T \\ vm_3^T - wm_2^T \end{bmatrix} X = 0 \tag{2.38}$$

และ

$$\begin{bmatrix} um_2'^T - vm_1'^T \\ um_3'^T - wm_1'^T \\ vm_3'^T - wm_2'^T \end{bmatrix} X = 0 \tag{2.39}$$

ทำการคูณแถวแรกของสมการแรกด้วย w และคูณแถวที่สองด้วย –v แล้วทำการบวกเราได้

$$\left(uwm_2^T - vwm_1^T - uvm_3^T + vwm_1^T\right)X = \left(uwm_2^T - uvm_3^T\right) \tag{2.40}$$

จากสมการเห็นว่าสมการนี้มีคุณสมบัติ Linearly Dependent กับแถวที่สามของสมการ ดังนั้น แต่ละกล้องเราใช้ได้เพียง 2 สมการ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

$$\begin{bmatrix} um_3^T - wm_1^T \\ vm_3^T - wm_2^T \end{bmatrix} X = 0 \tag{2.41}$$

และ

$$\begin{bmatrix} um_3^{'T} - wm_1^{'T} \\ vm_3^{'T} - wm_2^{'T} \end{bmatrix} X=0 \tag{2.42}$$

เขียนสมการใหม่ได้ดังนี้

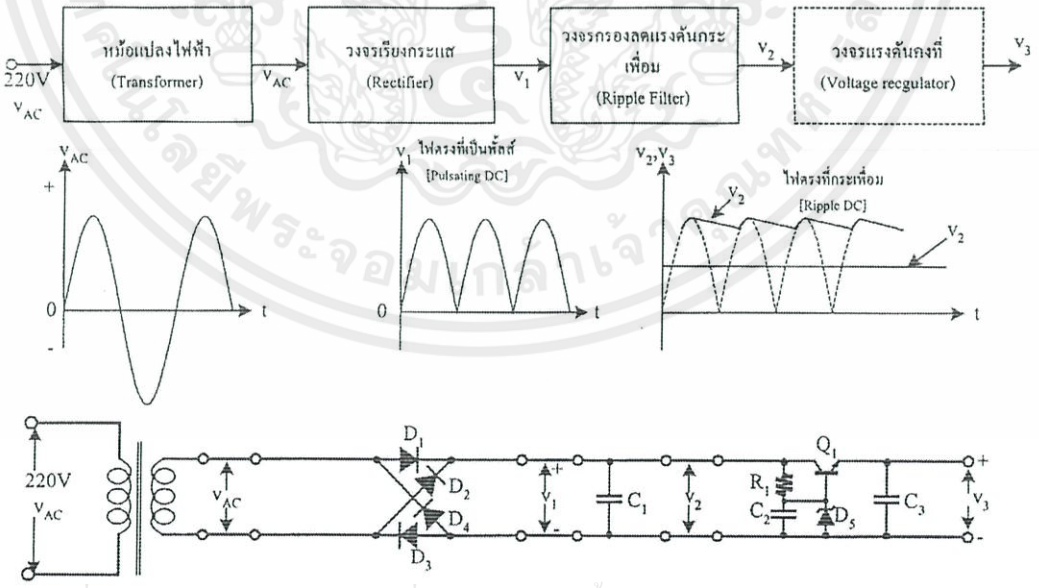
$$\begin{bmatrix} um_3^{T} - wm_1^{T} \\ vm_3^{T} - wm_2^{T} \\ um_3^{'T} - wm_1^{'T} \\ vm_3^{'T} - wm_2^{'T} \end{bmatrix} X = AX = 0 \tag{2.43}$$

ทำการแก้สมการโฮโมจีเนียสสำหรับพิกัด 3 มิติ หา X โดยเราสนใจเฉพาะคำตอบประเภท Non-Trival หรือกรณี det(A) = 0

2.9 ทฤษฎีและหลักการทำงานด้านฮาร์ดแวร์ (Hardware)

2.9.1 วงจรจ่ายไฟกระแสตรง (DC Power Supply)

วงจรอิเล็กทรอนิกส์เกือบทุกวงจรต้องการไฟกระแสตรงเพื่อเป็นพลังงานให้วงจรทำงานได้ ในอุปกรณ์หรือเครื่องมือขนาดเล็กๆ เคลื่อนที่ได้ เช่น เครื่องรับวิทยุ และเทปขนาดเล็ก ที่อาจใช้แบตเตอรี่ เป็นตัวจ่ายพลังงานให้ อย่างไรก็ตามอุปกรณ์อิเล็กทรอนิกส์ทั่วไป จะทำงานได้โดยใช้ไฟกระแสตรงจาก วงจรจ่ายไฟจากไฟบ้าน (Line operated DC Power Supply) บล็อกไดอะแกรมของวงจรจ่ายไฟ แสดงดังรูปที่ 2.12



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 2.12 วงจรจ่ายไฟตรงจากไฟบ้าน (Line operated DC Power Supply)

ค่าแรงดันไฟฟ้าของไฟบ้าน (Line Voltage) โดยทั่วไปจะมีค่า 220V หม้อแปลงไฟฟ้า จะทำหน้าที่เปลี่ยนค่าของแรงดันไฟฟ้ากระแสสลับ 220V ให้มีค่าตามที่ต้องการ ส่วนใหญ่จะเป็นการลดค่าแรงดันไฟฟ้าให้ต่ำลง และบางระบบอาจมีทั้งเพิ่มและลดแรงดันไฟฟ้า

เมื่อหม้อแปลงได้เปลี่ยนค่าแรงดันไฟฟ้าให้เป็นไปตามที่ต้องการแล้ว วงจรเรียงกระแส (Rectifier) ก็จะเปลี่ยนไฟกระแสสลับให้เป็นกระแสตรง ซึ่งวงจรเรียงกระแสอาจใช้ไดโอด 1 ตัว 2 ตัว 4 ตัวต่อวงจร

ไฟตรงที่ได้จากวงจรเรียงกระแสจะเป็นลักษณะเป็นพัลส์ (Pulsating DC) มีการกระเพื่อมสูงมาก (Ripple) วงจรอิเล็กทรอนิกส์ไม่สามารถทำงานได้ จากไฟตรงที่มีการกระเพื่อมอย่างมากเช่นนี้ วงจรกรองลดแรงดันกระเพื่อม (Ripple Filter) จะทำหน้าที่กรองแรงดันที่กระเพื่อมออกมาจากไฟตรงที่เป็นพัลส์ V_2 จะเป็นไฟตรงที่มีการกระเพื่อมน้อยลง วงจรอิเล็กทรอนิกส์โดยทั่วไปจะสามารถทำงานได้จากไฟตรงที่มีการกระเพื่อมเล็กน้อยเช่นนี้ (% ripple < 10%)

วงจรกรองลดแรงดันกระเพื่อม จะสร้างขึ้นมาได้จากตัวอุปกรณ์ที่เป็นรีแอคทีฟ (Reactive) ได้แก่ตัวเก็บประจุ (C) และตัวเหนี่ยวนำ (L) ซึ่งจะต่อวงจรในลักษณะกรองความถี่ต่ำผ่าน (Low pass Filer)

2.9.2 โครงสร้างพื้นฐานของแหล่งจ่ายไฟกระแสตรงแบบเชิงเส้น

2.9.2.1 หม้อแปลงไฟฟ้า (Transformer)

หม้อแปลงไฟฟ้าเป็นอุปกรณ์ที่ทำหนาที่แปลงแรงดันไฟฟ้ากระแสสลับใหมีขนาดของแรงดันสูงขึ้น หรือต่ำลงตามต้องการ โดยโครงสร้างของหม้อแปลงไฟฟ้า ประกอบด้วยขดลวดอย่างน้อยสองขดพันอยู่บนแกนเหล็ก การทำงานของหม้อแปลงอาศัยหลักการของการเหนี่ยวนำสนามแม่เหล็ก หม้อแปลงไฟฟ้าจะสามารถถ่ายทอดพลังงานได้เฉพาะไฟสลับหรือไฟตรงที่มีการเปลี่ยนแปลงเท่านั้น ($\frac{di}{dt} \neq 0$) หม้อแปลงบางตัวอาจมีขดทุติยภูมิได้หลายขด และขดทุติยภูมิอาจจะมีจุดแทปกึ่งกลาง (center tapped) ขดลวดด้านไฟเขาเรียกว่า ขดปฐมภูมิ (Primary และขดลวดทางด้านไฟออกเรียกว่า ขดทุติยภูมิ (Secondary) ในกรณีที่เป็นหม้อแปลงไฟฟ้าในอุดมคติ (Ideal Transformer) จะได้สมการความสัมพันธ์ระหว่างกระแสและแรงดันของขดลวดทั้งสองดังนี้

$$V_p I_p = V_s I_s \qquad (2.44)$$

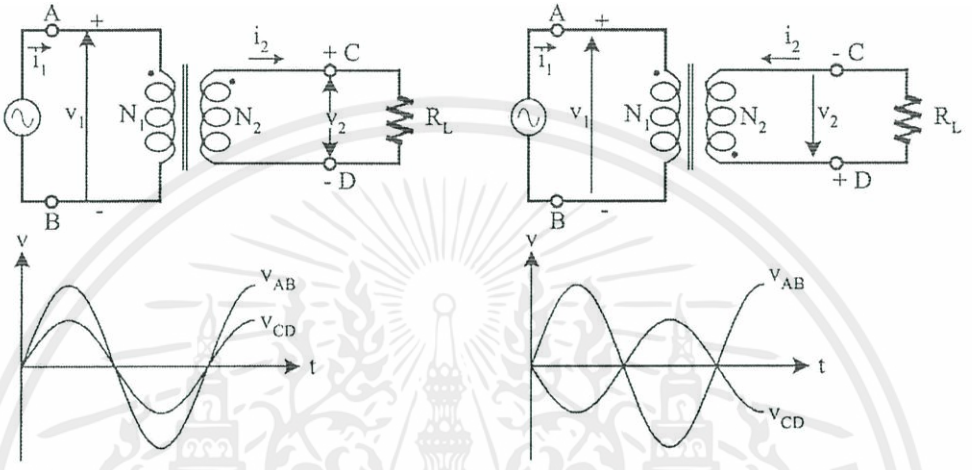
Vp คือแรงดันที่ปอนใหกับขดปฐมภูมิ
Vs คือแรงดันที่ไดจากขดทุติยภูมิ
Ip คือกระแสที่ไหลในขดปฐมภูมิ
Is คือกระแสที่ไหลในขดทุติยภูมิ

อัตราสวนของแรงดันจะขึ้นอยูกับอัตราสวนของจำนวนรอบของหม้อแปลงไฟฟ้าโดยมีความสัมพันธดังสมการ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

$$\frac{V_p}{V_s} = \frac{N_p}{N_s} \qquad (2.45)$$

Np คือจำนวนรอบของของปฐมภูมิ
Ns คือจำนวนรอบของขดทุติยภูมิ



รูปที่ 2.13 ขั้วหรือเฟสของหม้อแปลง

2.9.2.2 วงจรเรียงกระแส

วงจรเรียงกระแสสำหรับไฟเฟสเดียว (Single phase) ถ้าแบ่งตามมุมของกระแสที่ไหลผ่านโหลด จะสามารถแบ่งได้เป็น 2 แบบ

1. วงจรเรียงกระแสแบบครึ่งคลื่น (Half Wave Rectifier)
2. วงจรเรียงกระแสแบบเต็มคลื่น (Full Wave Rectifier)

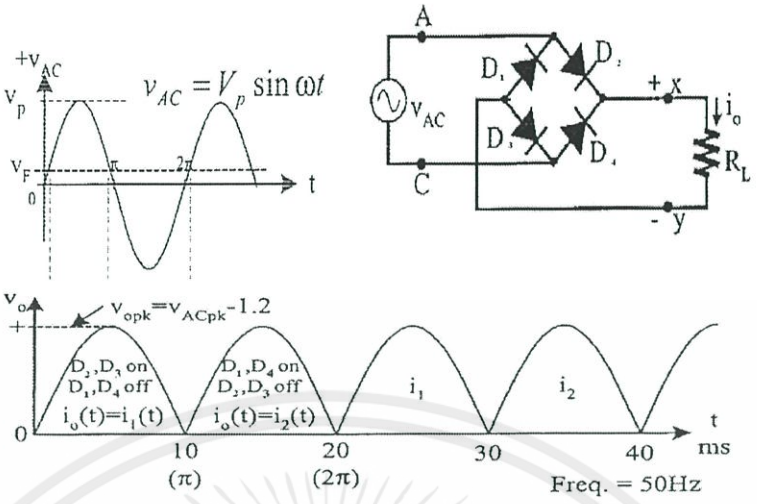
สามารถต่อวงจรได้ 2 แบบ คือ วงจรเรียงกระแสเต็มคลื่นแบบหม้อแปลงมีแทป (FWCT) วงจรเรียงกระแสแบบเต็มคลื่นแบบบริดจ์ (FWB)

*(** หมายเหตุ : โครงงานฉบับนี้ขอกล่าวเฉพาะ วงจรเรียงกระแสแบบเต็มคลื่นแบบบริดจ์ (FWB) เท่านั้น)*

2.9.2.2.1 วงจรเรียงกระแสเต็มคลื่นแบบบริดจ์ (FWB)

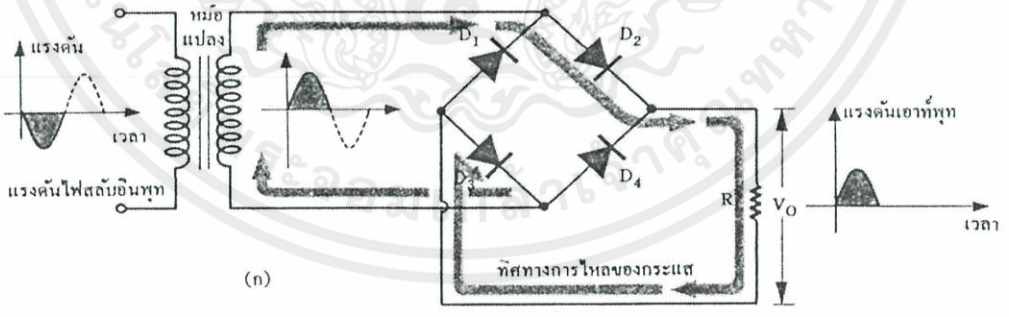
วงจรที่ให้สัญญาณออกเป็นรูปฟูลเวฟ (Full Wave) อีกแบบหนึ่งคือ วงจรเรียงกระแสเต็มคลื่นแบบบริดจ์ (FWB) แรงดันไฟสลับจะต่อเข้าที่สองมุมของวงจรบริดจ์ และเอาท์พุทจะถูกนำออกที่สองมุมที่เหลือดังวงจรในรูปที่ 2.14

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

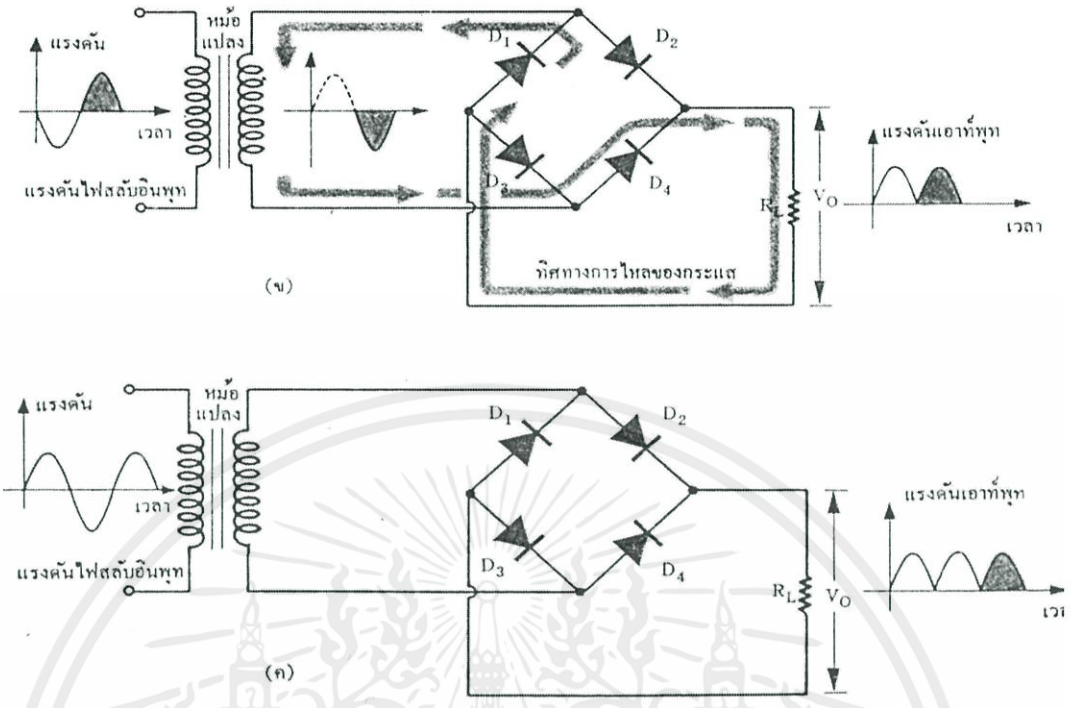


รูปที่ 2.14 วงจรบริดจ์เรคติไฟเออร์

ในแต่ละครึ่งไซเคิลของวงจรอินพุทสมมติว่าเมื่อขั้วบนของหม้อแปลงมีค่าเป็นบวก และขั้วล่างมีค่าเป็นลบหรือครึ่งไซเคิลลบถูกป้อนเข้าทางขดทุติยภูมิของหม้อแปลง ไดโอด D_2 และ D_3 จะอยู่ในลักษณะไบแอสตรง ดังนั้นกระแสจึงไหลครบวงจร จากขั้วบนของหม้อแปลงผ่านไดโอด D_2 ความต้านทานโหลด ไดโอด D_3 แล้วกลับเข้าขั้วล่างของหม้อแปลง (ดูรูปที่ 2.15) และเมื่อแรงดันไฟสลับเปลี่ยนข้างมาเป็นบวกที่ขั้วล่าง และเป็นลบที่ขั้วบนของขดทุติยภูมิ การนำกระแสของไดโอดจะเป็นไปโดยเริ่มจากจุดล่างของหม้อแปลงผ่านไดโอด D_4 ความต้านทานโหลด ไดโอด D_1 กลับเข้าขั้วบนของหม้อแปลง ทิศทางแรงดันตกคร่อมโหลดจะมีทิศทางเดียวกันกับตอนแรกคือ มีทิศบวกอยู่ข้างบน ดังนั้นการนำกระแสของไดโอดจะเกิดสลับกันทีละสองตัว คือ D_2 กับ D_3 และ D_1 กับ D_4 ดังรูป 4ข



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.15 แสดงการทำงานของวงจรบริดจ์เรคติไฟเออร์

เมื่อกำหนดหม้อแปลงให้วงจรบริดจ์เรคติไฟเออร์ วงจรสามารถทำให้แรงดันเอาท์พุทมีค่าเป็นสองเท่าของวงจรเรียงกระแสเต็มคลื่นแบบหม้อแปลงมีแทป (FWCT) ทั้งนี้เป็นเพราะแรงดันที่คร่อมหม้อแปลงทั้งหมดไม่มีการแบ่งครึ่ง แต่จะป้อนเข้ายังวงจรบริดจ์ได้โดยตรง จึงทำให้ Vp ของวงจรบริดจ์มีค่าเป็นสองเท่าของวงจรเรียงกระแสเต็มคลื่นแบบหม้อแปลงมีแทป (FWCT)

เมื่อพิจารณาขณะที่ไดโอดบางตัวถูกไบแอสกลับ สมมติว่าไดโอด D_2 และ D_3 นำกระแส ไดโอด D_1 และ D_4 จะไม่นำกระแส ขณะที่ไดโอด D_1 กับ D_4 ไม่นำกระแสนั้น แรงดันที่ตกคร่อมไดโอด D_2 และ D_3 จะมีค่ามากที่สุดได้เพียงค่า Vp เท่านั้น ซึ่งมีค่าน้อยกว่าวงจรเรียงกระแสเต็มคลื่นแบบหม้อแปลงมีแทป (FWCT) ซึ่งเป็นข้อดีอีกประการหนึ่งที่ไดโอดที่ใช้ไม่ต้องให้ทนแรงดันไบแอสกลับได้สูงมากเกินไปนัก ทนแรงดันได้ Vp/2 ก็พอเพียง

จึงเห็นได้ว่าวงจรเรียงกระแสเต็มคลื่นแบบบริดส์ (FWB) ให้ผลดีจึงมีผู้นิยมใช้กันมากในวงจรจ่ายไฟกระแสตรง

$$V_{DC\,(full\ wave)} = 0.636\ V_P \qquad (2.46)$$

ถ้าคิดแรงดันตกคร่อมไดโอด แรงดันเอาท์พุทเท่ากับ

$$V_{DC\,(full\ wave)} = 0.636(\ V_P\text{-}1.4) \qquad (2.47)$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.9.2.3 วงจรกรองลดการกระเพื่อม (Ripple Filter)

ไฟตรงที่ได้จากวงจรเรียงกระแสไม่ว่าแบบใด จะมีการกระเพื่อมสูงมาก โดยการวิเคราะห์ทางโดเมนความถี่ แรงดันไฟตรงที่ทางออกมิได้เป็นรูปคลื่นซายน์ (sinusoidal) จะประกอบด้วยความถี่ฮาร์โมนิกส์จำนวนมาก โดยการวิเคราะห์ด้วยอนุกรมฟูริเยร์ (Fourier Series) จะได้ผลดังนี้

$$V_{HW} = V_p\left(\frac{1}{\pi} + \frac{1}{2}\cos\omega t + \frac{2}{3\pi}\cos 2\omega t - \frac{2}{15\pi}\cos 4\omega t + \ldots - \ldots + \ldots\right) \quad (2.48)$$

$$V_{FW} = V_p\left(\frac{2}{\pi} + \frac{4}{3\pi}\cos 2\omega t - \frac{4}{15\pi}\cos 4\omega t + \ldots\right) \quad (2.49)$$

$$CV = \frac{I_O}{FR} \quad (2.50)$$

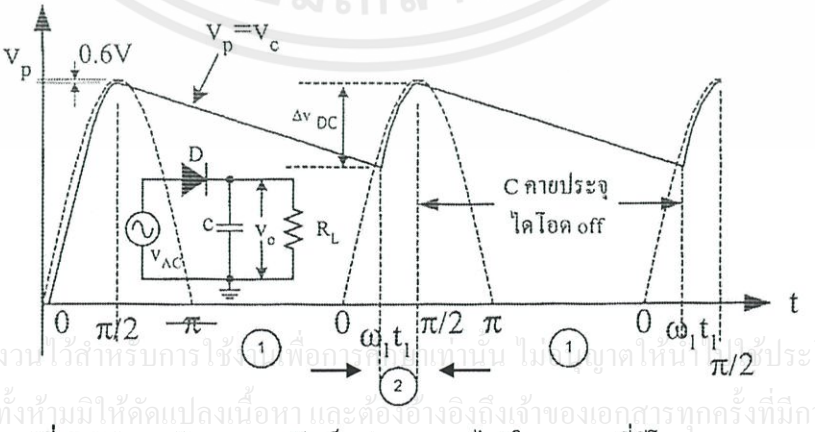
จากสมการพบว่าในการลดไฟตรงที่ได้จากวงจรเรียงกระแสยังมีความถี่สูงอยู่ดังนั้นในการลดการกระเพื่อม เราจะต้องลดส่วนของพจน์ที่เป็นกระแสสลับ (พจน์ที่มี ωt เนื่องจากความถี่ของไฟตรงคือ 0 H_Z โดยการใช้วงจรกรองความถี่ต่ำผ่าน (Low pass filter) องค์ประกอบของแรงดันกระเพื่อมซึ่งมีความถี่สูงกว่าไฟตรง ($\omega > 0$) จะถูกทำให้มีขนาดลดลง วงจรกรองความถี่ต่ำผ่านอาจทำได้โดยการใช้ตัวเหนี่ยวนำต่ออนุกรมกับ R_L หรือ ตัวเก็บประจุต่อขนานกับ R_L

วงจรกรองแรงดันมีอยู่หลายประเภท คือ

1. วงจรกรองแรงดันด้วยตัวเก็บประจุ (Capacitor Filter)
2. วงจรกรองแรงดันด้วยตัวเหนี่ยวนำ (Inductor Filter)

เนื่องจากวงจรกรองแรงดันด้วยตัวเหนี่ยวนำ ถาจะให้มีประสิทธิภาพเท่ากับวงจรกรองแรงดันด้วยตัวเก็บประจุ จะมีขนาดใหญ่และมีน้ำหนักมากจึงนิยมใช่วงจรกรองแรงดันด่วยตัวเก็บประจุมากกวา จึงขอกล่าวเฉพาะรายละเอียดของวงจรกรองแรงดันด้วยตัวเก็บประจุเท่านั้น

1 วงจรกรองแรงดันดวยตัวเก็บประจุ (Capaciter Filter) จะใชตัวเก็บประจุขนานเข้าไปหลังสวนของตัวเรียงกระแส โดยมีลักษณะวงจรดังรูปที่ 2.16

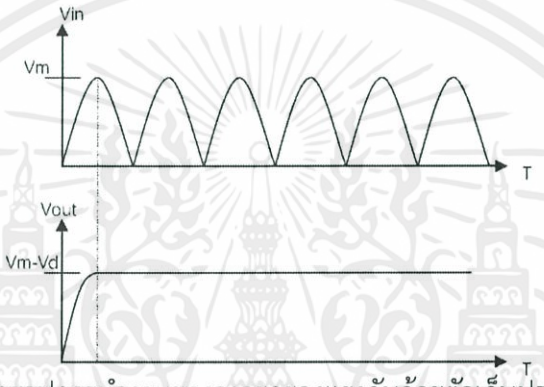


รูปที่ 2.16 การทำงานของตัวเก็บประจุกรองไฟ ในสภาวะที่มีโหลด

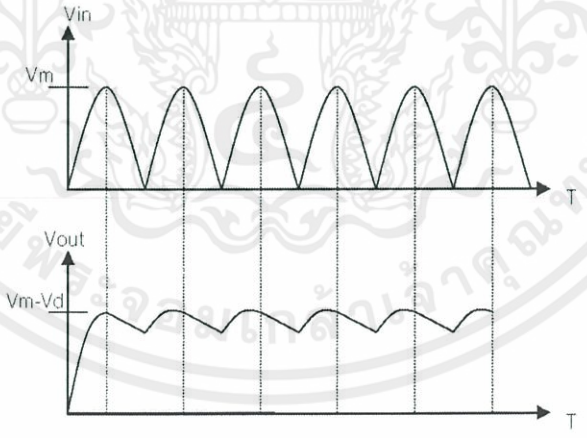
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากวงจรในรูปขณะที่ไดโอดนำกระแส ตัวเก็บประจุจะสะสมพลังงานเอาไว้ เมื่อไดโอดหยุดนำกระแส ตัวเก็บประจุจะคายพลังงานออกมาให้กับ R_R ทำให้โหลดได้รับพลังงานอย่างต่อเนื่อง

ในสภาวะที่ไม่มีโหลด(R_L =open) ไดโอดเริ่มจะนำกระแสเมื่อได้รับไฟสลับทางด้านบวกครั้งแรกเป็นเวลา ¼ คาบของไฟสลับ ซึ่งเมื่อเวลาตัวเก็บประจุจะได้รับการประจุให้มีแรงดันตกคร่อมเท่ากับ $V_m - V_d$ เนื่องจากไม่มีโหลดค่าแรงดันค่านี้จะอยู่เช่นนี้ตลอดไปจนกว่าจะได้รับแรงดันไฟสลับทางด้านบวกครั้งทำให้เราได้ไฟตรงที่มีความเรียบมากๆเหมือนกับไฟตรงในอุดมคติดังรูปที่ 2.17



รูปที่ 2.17 กราฟการทำงานของวงจรกรองแรงดันด้วยตัวเก็บประจุในสภาวะที่ไม่มีโหลด



รูปที่ 2.18 กราฟแสดงการทำงานของวงจรกรองแรงดันด้วยตัวเก็บประจุ ในสภาวะมีโหลด

จะเห็นว่าช่วงเวลาที่ตัวเก็บประจุ (C) คายประจุจะมีผลอย่างมากต่อความเรียบของไฟตรงที่ได้ดังนั้นจึงควรเลือกใช้ตัวเก็บประจุ ที่มีค่ามากพอที่จะทำให้ค่าคงตัวทางเวลา (Time Constant) มีค่ามากพอ ซึ่งหลักในการเลือกตัวเก็บประจุมีหลักการง่ายๆ ดังสมการที่ 2.51

เอกสารนี้เป็นกรรมสิทธิ์ของ ... ไม่ว่ากรณีใดๆ ... ทุกครั้งที่มีการนำไปใช้

$$CV = \frac{I_o}{FR} \quad (2.51)$$

C คือค่าความจุไฟฟาของตัวเก็บประจุ มีหนวยเปน ฟารัด (Farad)

V คือค่าแรงดันกระเพื่อมจากยอดถึงยอดมีหนวยเปนโวลต์พีคทูพีค (V_{PP})

I_o คือค่ากระแสเฉลี่ยที่จ่ายให้กับโหลด มีหน่วยเป็นแอมแปร์ (A)

FR คือความถี่ของการกระเพื่อมมีหน่วยเป็นเรเดียนต่อวินาที (rad/sec หรือ Hz)

โดยที่ $FR = F_{line}$ สำหรับ HW และ $FR = 2F_{line}$ สำหรับ FW

ค่าริปเปิ้ล (Ripple Voltage)

ในการแปลงไฟจากกระแสสลับเป็นกระแสตรงนั้น รูปคลื่นที่ได้ทางเอาท์พุทมีลักษณะเป็นพัลส์ (Pulse) ไม่เหมือนกระแสตรงที่ได้จากแบตเตอร์รี่ เนื่องจากมีรูปคลื่นที่เปลี่ยนแปลงเสมอ เรียก Ripple Factor ในทางปฏิบัติการแปลงไฟต้องพยายามลดค่าริปเปิ้ลที่เกิดขึ้นให้น้อยที่สุด เพื่อให้ได้ประสิทธิภาพของวงจร

ในการแสดงถึงคุณสมบัติของวงจรกรองลดการกระเพื่อมว่ามีคุณสมบัติดีเพียงใด เราสามารถดูได้จากค่าของเปอร์เซ็นต์ของการกระเพื่อม (Percentage of Ripple : R) ซึ่งนิยามโดย

$$\%R = \frac{V_{rms}}{V_{dc}} \times 100 \quad (2.52)$$

หากแหล่งจ่ายไฟมีค่าริปเปิ้ลสูงจะเป็นอันตรายต่อระบบการทำงานและอาจเกิดสัญญาณรบกวนขึ้น

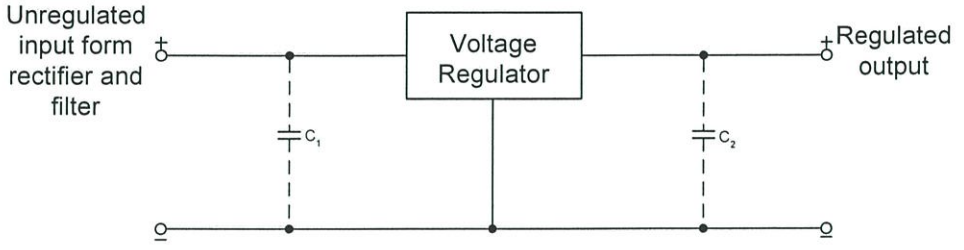
2.9.2.4 วงจรรักษาแรงดันใช้ไอซีรักษาแรงดัน(IC Regulator)

ข้อดีของวงจรแบบนี้ คือ ราคาถูก มีขนาดเล็ก และรูปแบบวงจรที่ง่าย สามารถจ่ายกระแสเอาท์พุทได้ตั้งแต่ 3 mA ถึง 100mA ตามเบอร์ที่เราเลือกใช้ ยังมีวงจรป้องกันกระแสเกินภายในและวงจรป้องกันอุณหภูมิเกินภายในด้วยโดย IC เบอร์ต่างๆ จะมีคุณสมบัติด้านกระแสเอาท์พุทสูงสุด, แรงดันอินพุท, Line Regulation, Load Regulation และช่วงอุณหภูมิที่เราทำงานให้เราเลือกตามต้องการ วงจรรักษาระดับแรงดันแบบ IC แบ่งได้เป็น 2 ชนิด คือ

1 วงจรรักษาระดับแรงดันแบบแรงดันเอาท์พุทคงที่

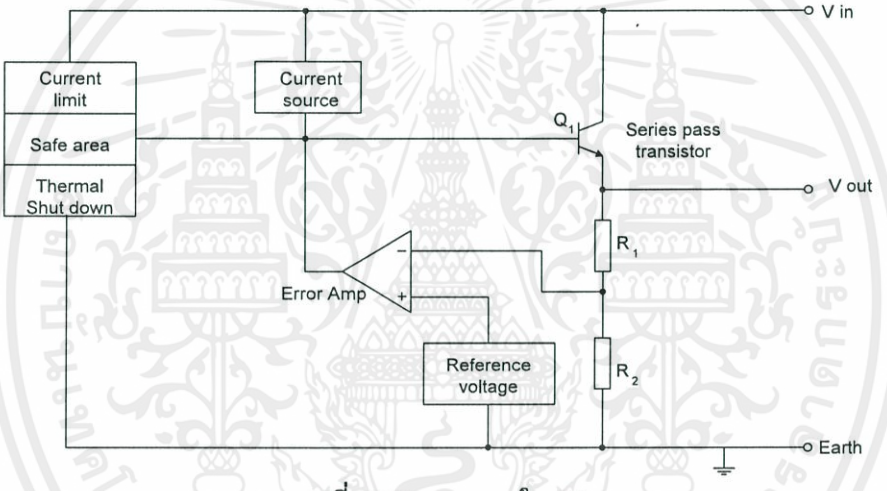
เบอร์ที่นิยมใช้จะเป็น IC ตระกูล 78xx และ 79xx และต่างกันที่ 78xx ให้ไฟบวก และ 79xx ให้ไฟลบ โดยมีเลข xx เป็นค่าแรงดันเอาท์พุท เช่น 7805 ให้ไฟบวกแรงดันคงที่ 5V, 7812 ให้ไฟบวกแรงดันคงที่ 12V เป็นต้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.19 พื้นฐานในการต่อ IC Regulator

ในการทำงานของ IC Regulator ถ้าเกิดผลต่างของแรงดันทางอินพุทและเอาท์พุทมาก จะทำให้เกิดแรงดันตกคร่อมที่ IC (Drop Out) ดังนั้นจึงเกิดการสูญเสียพลังงานในรูปของความร้อน ถ้าความร้อนมีมากเกินไป ทำให้วงจรส่วน Thermal Shutdown ภายใน IC ทำงาน แต่อย่างไรก็ตามควรติดแผ่นระบายความร้อน (Heat sink) เพื่อช่วยระบายความร้อน

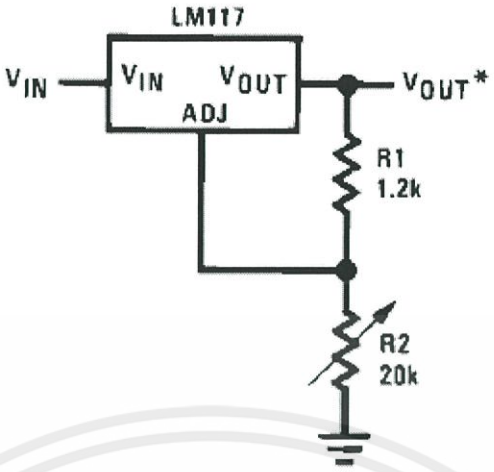


รูปที่ 2.20 วงจรภายใน IC

2.9.2.4.2 วงจรรักษาระดับแรงดันแบบปรับค่าแรงดันเอาท์พุทได้

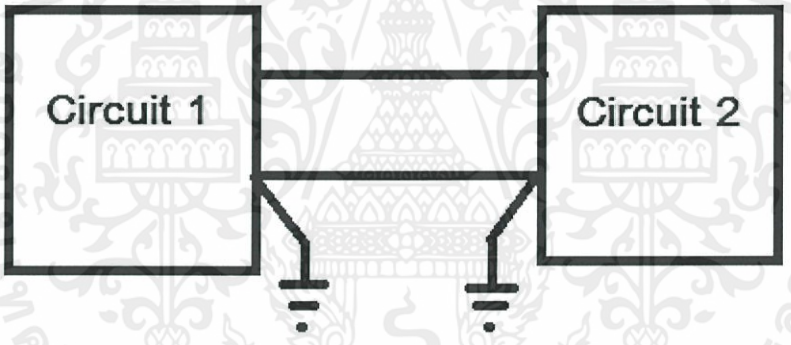
เป็นวงจรที่สามารถปรับค่าแรงดันเอาท์พุทได้ในช่วงที่กำหนด เช่น LM117,LM317 เป็นต้น ปรับค่าแรงดันเอาท์พุทได้ตั้งแต่ 1.25V-37V แต่ถ้าต้องการจ่ายเอาท์พุทตั้งแต่ 0V สามารถทำได้โดยเพิ่มส่วนไฟลบให้กับวงจร แต่ละเบอร์ IC นั้นจะมีคุณสมบัติทางแรงดันเอาท์พุท,กระแสสูงสุดช่วงแรงดันที่สามารถทำงานได้ ,ความแตกต่างระหว่างแรงดันอินพุทกับเอาท์พุท (Vo-Vin) ต่ำสุดที่สามารถทำงานได้ ,Line Regulation ,Load Regulation และการป้องกันต่างๆ เช่น วงจรจำกัดกระแส วงจรป้องกันให้อยู่ในช่วงที่ปลอดภัย (Safe Operation Area Protection) เป็นต้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.21 วงจรรักษาระดับแรงดันแบบปรับค่าแรงดันเอาท์พุทได้

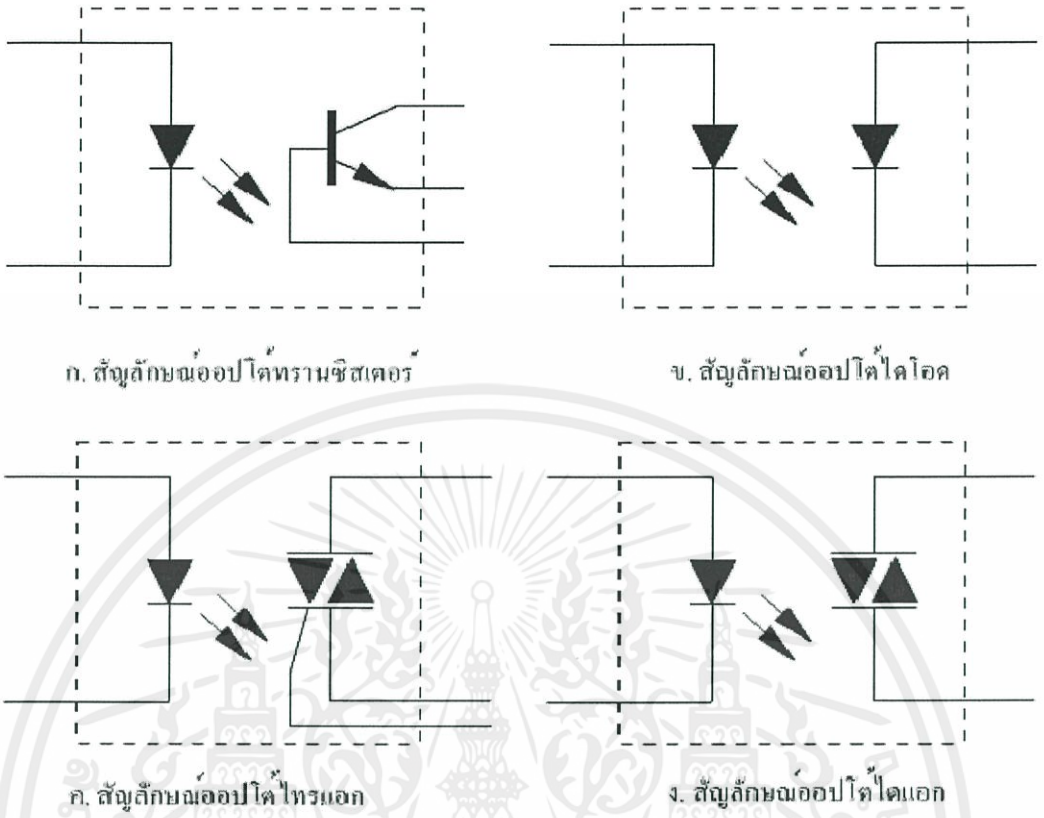
2.9.3 วงจรแยกกราวน์ (ground isolation Circuit)
โดยทั่วไปวงจร 2 วงจร ที่เชื่อมต่อกันมักจะใช้กราวน์ร่วมกัน ดังรูป



รูปที่ 2.22 วงจรที่มีกราวน์ร่วมกัน

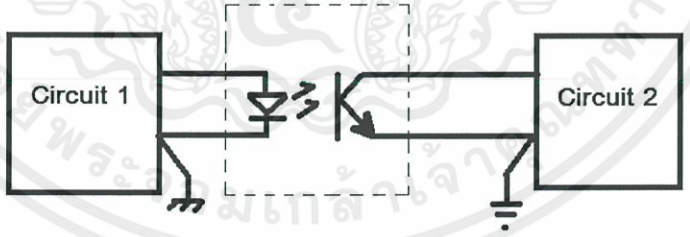
เราสามารถใช้อุปกรณ์เชื่อมต่อทางแสง (Opto-Isolator) หรือที่เรียกว่าออปโต้คัป-เปลอร์ (Opto-Coupler) เป็นอุปกรณ์อิเล็กทรอนิกส์ที่ใช้ในการเชื่อมต่อทางแสงโดยใช้หลักการเปลี่ยนสัญญาณไฟฟ้าเป็นสัญญาณแสง และเปลี่ยนกลับจากแสงเป็นไฟฟ้าตามเดิม ใช้สำหรับการเชื่อมต่อสัญญาณระหว่างสองวงจรที่ต้องการแยกทางไฟฟ้าอย่างเด็ดขาดเพื่อป้องกันการรบกวนกันทางไฟฟ้า แบ่งออกเป็นหลายชนิดแต่ละชนิดจะประกอบด้วย LED ส่งแสงซึ่งปติจะเป็นชนิดอินฟาเรด และตัวรับแสงที่เป็นโฟโต้ทรานซิสเตอร์หรือโฟโต้ไดโอด โดยจะถูกผลิตรวมอยู่ในตัวเดียวกัน ดังรูป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.23 สัญลักษณ์ออปโต้แบบต่างๆ

เราสามารถใช้ opto isolator แยกกราวน์ทั้งสองวงจรได้ดังรูป



รูปที่ 2.24 ตัวอย่างการใช้งานออปโต้ทรานซิสเตอร์

2.9.4 Arduino Board

Arduino เป็นบอร์ดไมโครคอนโทรลเลอร์โดยใช้ AVR ขนาดเล็กเป็นตัวประมวลผลและสั่งงาน เหมาะสาหรับนำไปใช้ในการศึกษาเรียนรู้ระบบไมโครคอนโทรลเลอร์ และนำไปประยุกต์ใช้งานเกี่ยวกับการควบคุมอุปกรณ์ Input / Output ต่างๆได้มาก ทั้งแบบที่เป็นการทำงานตัวเดียวอิสระ หรือเชื่อมต่อสั่งงานร่วมกับอุปกรณ์อื่นๆ เช่น คอมพิวเตอร์ PC เนื่องจากว่า Arduino สนับสนุนการเชื่อมต่อกับอุปกรณ์ Input / Output ต่างๆได้มากมาย ทั้งแบบ Digital และ Analog เช่น การ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น

รับค่าจากสวิตซ์ หรืออุปกรณ์ตรวจจับ (Sensor) แบบต่างๆ รวมไปถึง การควบคุมอุปกรณ์ Output ต่างๆ ตั้งแต่ LED, หลอดไฟ, มอเตอร์, รีเลย์ ฯลฯ

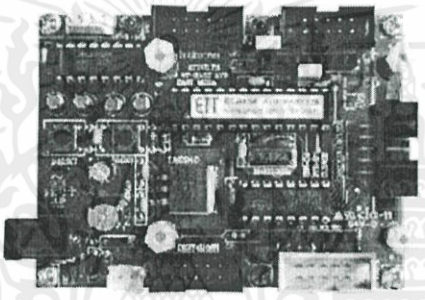
ข้อดีของระบบฮาร์ดแวร์ของ Arduino

1. สามารถสร้างและประกอบขึ้นใช้งานได้เอง กรณีที่ผู้ใช้พอมีความรู้ด้านอิเล็กทรอนิกส์ หรือสามารถซื้อแผงวงจรสาเร็จรูปที่มีการผลิตออกจาหน่ายกันในราคาที่ไม่แพง

2. เผยแพร่ Source Code และตัวอย่างต่างๆ ให้ผู้ใช้นาไปใช้งาน หรือพัฒนาดัดแปลงต่อยอดได้โดยไม่เสียค่าใช้จ่ายใดๆ

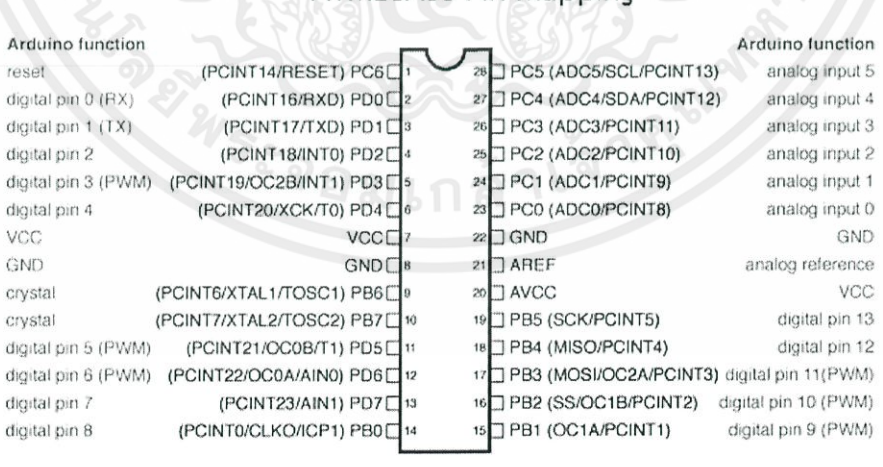
3. โปรแกรมพัฒนา Arduino ทางานได้ทั้งบนวินโดวส์ Macintosh OSX และ บนลีนุกซ์

การทำโครงงานครั้งนี้ได้เลือกเบอร์ Arduino ET-BASE AVR EASY88 บอร์ดรุ่นนี้บรรจุ MCU ภายในตัวถังแบบ 28 Pin DIP ซึ่งมีจุดเด่น คือ เป็นไมโครคอนโทรลเลอร์ขนาดเล็กแต่เพียบพร้อมไปด้วยทรัพยากรพื้นฐานต่างๆครบถ้วน



รูปที่ 2.25 บอร์ด ET-BASE AVR EASY88

2.9.5 Microcontroller ที่ใช้บนบอร์ด AVR



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 2.26 ซีพียู ATMEGA88P ขนาด 28 ขา

คุณสมบัติของ Microcontroller AVR ATMEGA88

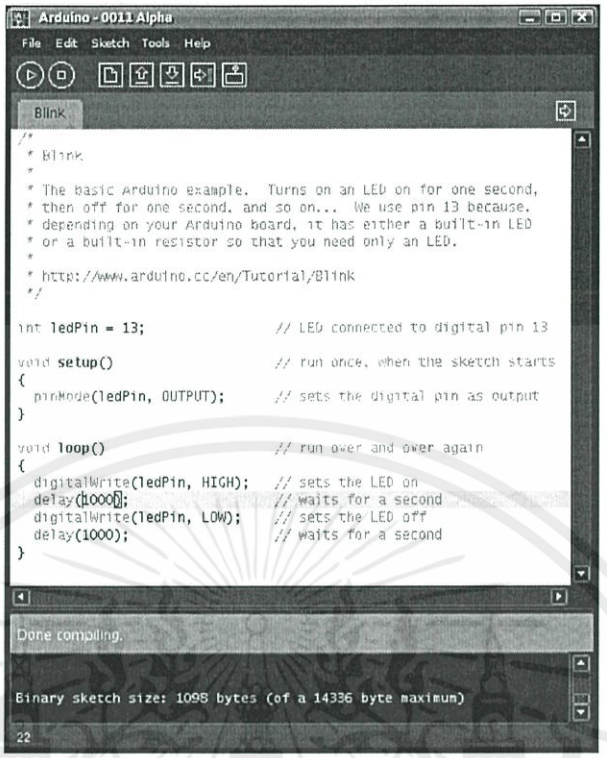
- มีหน่วยความจำ 8 Kbyte Flash / 512 Byte EEPROM / 1024 Byte SRAM
- สามารถเลือกใช้แหล่งกำเนิดสัญญาณนาฬิกาแบบ Crystal Oscillator ตั้งแต่ 8-20MHz สำหรับโครงงานฉบับนี้จะใช้ 19.6608 MHz เพื่อให้ทำงานสัมพันธ์กับโปรแกรม Boot loader
- มี Port Digital I/O 13 port , Analog Input 6 port
- ติดตั้ง Boot loader สำหรับใช้ Upload Code ผ่านทางพอร์ตสื่อสารอนุกรม RS232 โดยไม่ต้องใช้เครื่องโปรแกรมจากภายนอก

ตัวบอร์ด Arduino ที่ใช้ในโปรเจคนี้จะกล่าวถึงสถาปัตยกรรมของ AVR ขนาด 8 บิต โดยในสถาปัตยกรรม AVR เป็นซีพียูแบบ RISC (Reduced Instruction Set Computer) สถาปัตยกรรมการต่อหน่วยความจำแบบ Harvard ซึ่งแยกหน่วยความจำโปรแกรม และหน่วยความจำ ข้อมูลออกจากกันโดยเด็ดขาด โดยใช้หน่วยความจำแบบ Flash สำหรับเป็นหน่วยความจำโปรแกรม และใช้หน่วยความจำแบบ SRAM สำหรับหน่วยความจำข้อมูล และ นอกจากนี้ยังมีหน่วยความจำ แบบ EEPROM ซึ่งสามารถเก็บ ข้อมูลเอาไว้ได้โดยไม่จำเป็นต้องมีไฟเลี้ยง

2.9.6 Software Arduino

โปรแกรม Arduino Environment เป็นซอฟต์แวร์ที่พัฒนาขึ้นมาเพื่อการเขียนโปรแกรมลงบนบอร์ด Arduino โดยจะสามารถรองรับได้ทั้งระบบปฏิบัติการ Window, Linux และ Mac OS X และภายในเว็ปไซต์ก็จะมีตัวอย่าง Open Source

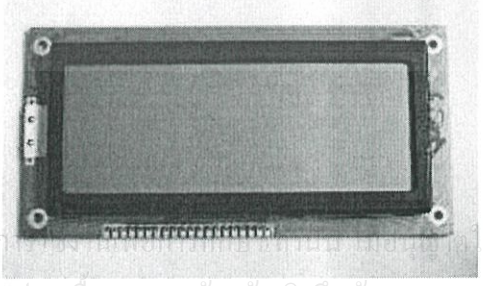
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.27 หน้าตาโปรแกรมพัฒนา Arduino Board

ความหมายของแต่ละคำสั่งต่างบนหน้าโปรแกรม Arduino
Verify ใช้ตรวจสอบ Code โปรแกรมว่าเขียนถูกหรือไม
Stop ใช้สั่งหยุดการทำงายของโปรแกรม Terminal
New ใช้สร้างไฟล์ที่เก็บไว้ขึ้นมาใช้งานบนโปรแกรม
Open ใช้สั่งเปิดไฟล์ที่เก็บไว้ขึ้นมาใช้งานบนโปรแกรม
Save ใช้ Save Code โปรแกรมเป็นไฟล์เก็บไว้ใน PC
Upload to I/O Board ใช้ Upload โปรแกรมให้บอร์ด
Serial Monitor ใช้ทดสอบ รับ ส่ง ข้อมูลกับบอร์ด

2.9.7. จอแสดงผล LCD (Liquid Crystal Display)



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 2.28 ตัวอย่างจอแสดงผล LCD

จอแสดงผลแบบ LCD (Liquid Crystal Display) จัดเป็นจอแสดงผลอีกรูปแบบหนึ่ง ซึ่งเป็นที่นิยมนำมาใช้งานกันอย่างแพร่หลายมากขึ้นในปัจจุบัน โดยที่พบเห็นกันในปัจจุบันอาจมีหลากหลายขนาดให้เลือกใช้ ที่พบเห็นกันทั่วไปได้แก่ 16 ตัวอักษร ไปจนถึง 40 ตัวอักษร และมีจำนวนบรรทัดตั้งแต่ 1 บรรทัด ไปจนถึง 4 บรรทัด หรืออาจมากกว่า แล้วแต่ผู้ผลิตแต่ส่วนมากแล้วจะมีโครงสร้างการทำงานและชุดคำสั่งที่เหมือนกันเกือบทุกประการอาจแตกต่างกันบาง ในเรื่องของความเร็วในการอ่าน/เขียน (Access Time) ซึ่งโดยส่วนมากแล้ว LCD ทุกตัว จะใช้ Controller ของบริษัท Hitachi เบอร์ HD44780

ตามปกติแล้ว LCD จะมี Chip Controller บรรจุไว้ภายในชุดโมดูลด้วยอยู่แล้ว การสั่งงานต่างๆจะกระทำผ่าน Controller ของแผงโมดูลเป็นหลัก

2.9.7.1 โครงสร้างทั่วไปของ LCD

ตามปกติโครงสร้างทั่วไปของ LCD จะประกอบด้วยแผ่นแก้ว 2 แผ่น ประกบกันอยู่โดยเว้นช่องว่างตรงกลางไว้ 6-10 μm. ผิวด้านในของแผ่นแก้วจะเคลือบด้วยตัวนำไฟฟ้าชนิดใสเพื่อแสดงตัวอักษร ระหว่างตัวนำไฟฟ้ากับผลึกเหลวจะมีชั้นสารที่ทำให้โมเลกุลของผลึกรวมตัวกัน ในทิศทางที่แสงส่องมากระทบซึ่งเรียกว่า Alignment Layer และผลึกเหลวที่ใช้โดยทั่วไปจะเป็นแบบ Magnetic โดย LCD สามารถแสดงผลได้ทั้งหมด 3 แบบ คือ

- แบบใช้การสะท้องแสง (Reflective Mode) ซึ่งจะใช้สารประเภทโลหะเคลือบอยู่ที่แผ่นหลังของ LCD ซึ่ง LCD ประเภทนี้จะเหมาะกับการนำมาใช้งานในที่มีแสงสว่างเพียงพอ
- แบบใช้การส่องผ่าน (Transmissive Mode) โดย LCD แบบนี้จะวางหลอดไฟไว้ด้านหลังจอ LCD แสดงผลเพื่อทำให้อ่านค่าได้ชัดเจน
- แบบส่งผ่าน/สะท้อน (Transflective Mode) แบบนี้จะเป็นนำเอาข้อดีของจอแสดงผล LCD ทั้งสองแบบแรกมารวมกัน

2.9.7.2 การควบคุมการแสดงผลของ LCD

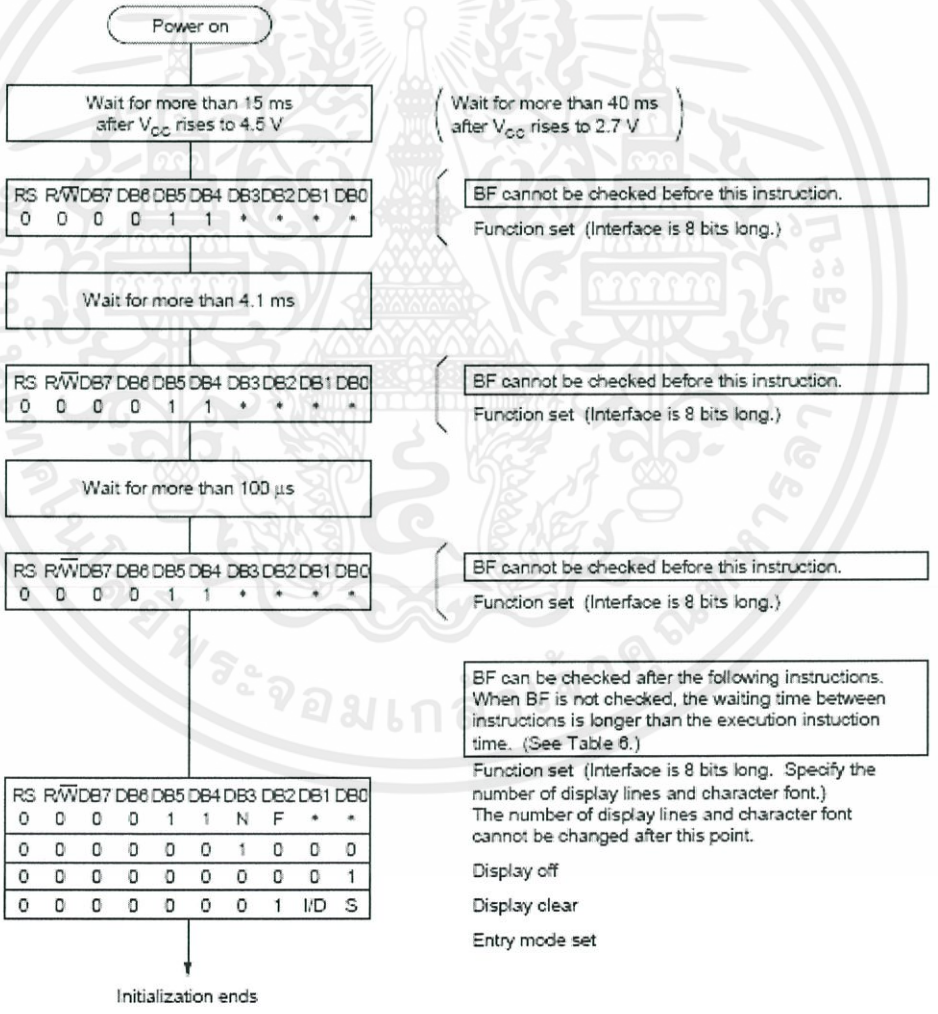
เราไม่สามารถจ่ายไฟกระแสตรงให้กับ LCD ค้างไว้ตลอดเวลาเพื่อให้ LCD แสดงผลตามที่เราต้องการได้เนื่องจากจะทำให้เกิดปฏิกิริยาเคมีไฟฟ้าขึ้นและจะทำให้อายุการใช้งาน LCD สั้นลง ฉะนั้น จำเป็นจะต้องป้อนสัญญาณสลับระหว่างปิดกับเปิด (SCAN) ด้วยความถี่ไม่น้อยกว่า 30 Hz เพื่อไม่ให้หน้าจอกระพริบ โดยปกติแล้ว LCD ที่นำมาใช้งานทั่วไปจะเป็นแบบที่มีส่วนควบคุม (Controller) รวมไว้ในตัว LCD ด้วยอยู่แล้ว ผู้ใช้งานเพียงแค่ส่งรหัสคำสั่งควบคุมการทำงานของ LCD ให้กับ Controller เท่านั้นว่าต้องการให้ LCD ทำงานอย่างไร แสดงข้อความอะไร ซึ่งส่วนใหญ่แล้วจะใช้ Controller ของ Hitachi เบอร์ HD44780 ซึ่งจะมีสัญญาณในการเชื่อมต่อระหว่าง LCD กับ ไมโครคอนโทรลเลอร์ ดังนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1. GND เป็นกราวน์ของไมโครคอนโทรลเลอร์กับ LCD
2. VCC เป็นไฟเลี้ยงวงจรที่ต้องป้อนให้กับ LCD มีขนาด +5V DC
3. VO เป็นขาสำหรับปรับความสว่างของหน้าจอ LCD
4. RS ใช้สำหรับบอกให้ LCD Controller ทราบว่าข้อมูลที่ส่งให้ทางขา Data เป็นคำสั่งหรือข้อมูล
5. R/W ใช้สำหรับกำหนดว่าจะอ่านหรือเขียนข้อมูล
6. E เป็นขา Enable หรือ Chip Select เพื่อกำหนดการทำงานให้กับจอ LCD
7. DB0-DB7 เป็นขาสัญญาณ Data ใช้สำหรับเขียนหรืออ่านข้อมูล/คำสั่ง

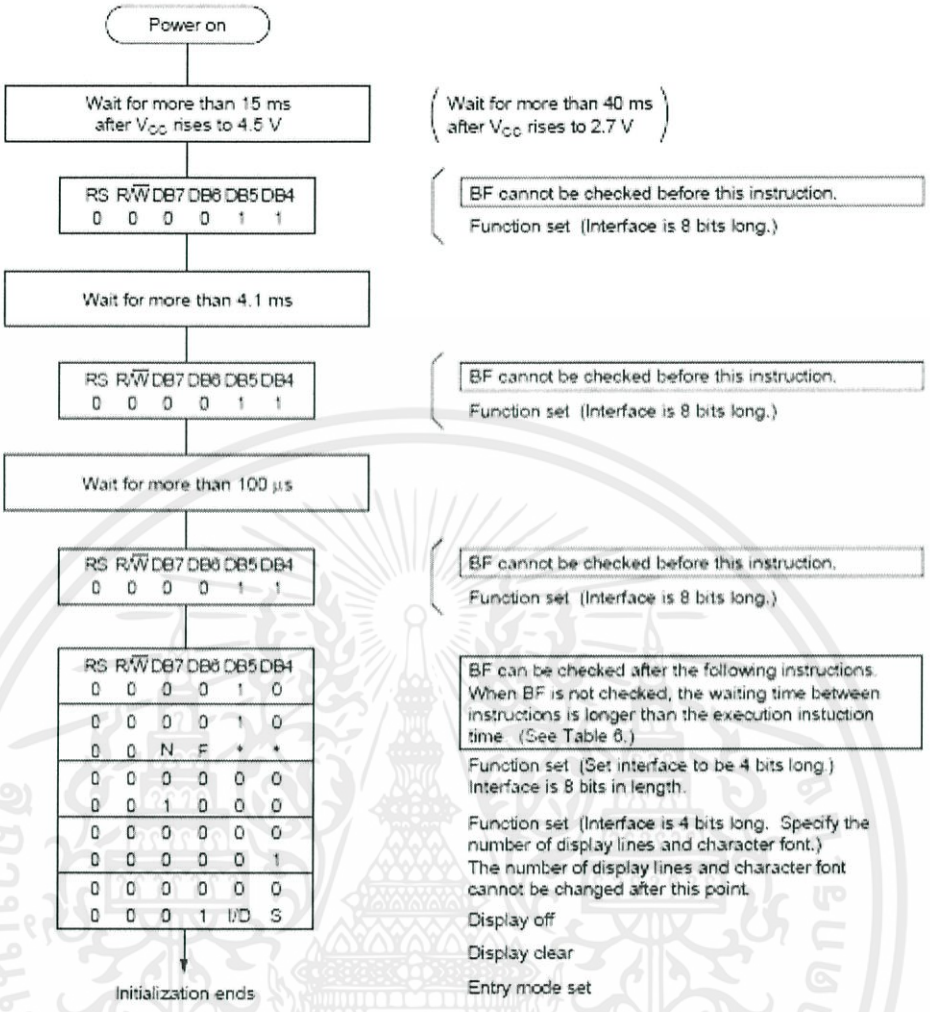
2.9.7.3 การเชื่อมต่อระหว่าง ไมโครคอนโทรลเลอร์ กับ LCD Controller

สำหรับการเชื่อมต่อสัญญาณระหว่างไมโครคอนโทรลเลอร์ กับ LCD Controller นั้นสามารถทำได้ 2 ลักษณะ คือ การเชื่อมต่อแบบ 8 บิต และการเชื่อมต่อแบบ 4 บิต โดยมีข้อกำหนดดังนี้



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 2.29 การ Initial LCD แบบ 8 bit Mode



รูปที่ 2.30 การ Initial LCD แบบ 4 bit Mode

2.9.7.4 คำสั่งสำหรับ LCD Controller เบอร์ HD44780 หรือเทียบเท่า

คำสั่งสำหรับการสั่งงานให้ LCD ทำงานตามต้องการนั้น จะมีอยู่ทั้งหมด 11 คำสั่ง โดยแต่ละคำสั่ง จะมีหน้าที่การใช้งานและวิธีการสั่งงานที่แตกต่างกันไป โดย LCD สามารถรับคำสั่งจากสัญญาณ RS , R/W และ DB7-DB0 ในขณะที่สัญญาณ E มีค่า logic = 1 ซึ่งสัญญาณเหล่านี้จะใช้ร่วมกันเพื่อกำหนดเป็นรหัสคำสั่งสำหรับสั่งงาน LCD ได้ดังนี้

สัญญาณ	RS	RW	DB7	DB6	DB5	DB4	DB3	DB2	DB1	DB0

- E เป็นสัญญาณ Enable บอกให้ LCD รู้ว่าอุปกรณ์ภายนอกต้องการติดต่ออ่าน/เขียนข้อมูล E=0 Disable ไม่สนใจสัญญาณ RS/RW และ DB7-DB0 , E=1 Enable LCD จะทำการถอดรหัสคำสั่ง หรือข้อมูลที่ปรากฏที่ขาสัญญาณ และทำงานตามคำสั่งนั้นทันที

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา

- **RS** เป็นสัญญาณสำหรับกำหนดให้ LCD ทราบว่า อุปกรณ์ภายนอกต้องการติดต่อกับ LCD นั้น เป็น รหัสคำสั่งหรือข้อมูล RS=0 คำสั่ง , RS=1 ข้อมูล

- **R/W** เป็นสัญญาณบอก LCD ให้ทราบว่าต้องการอ่านหรือเขียนกับ LCD ถ้า R/W=0 เขียน , R/W=1 อ่าน
- **DB7-DB0** เป็นสัญญาณแบบ 2 ทิศทาง สัมพันธ์กับสัญญาณ R/W ใช้ส่งคำสั่งหรือข้อมูลระหว่าง LCD กับ อุปกรณ์ภายนอก ถ้า R/W=0 สัญญาณ DB7-DB0 จะส่งจากอุปกรณ์ภายนอกไปหา LCD แต่ถ้า R/W=1 สัญญาณ DB7-DB0 จะส่งจาก LCD มายังอุปกรณ์ภายนอก

โดยมีจะมีคำสั่งที่ใช้อ่านและเขียนข้อมูลทั้งหมด 11 คำสั่ง แบ่งออก 4 กลุ่ม

- **Write Command** เป็นการเขียนรหัสคำสั่งต่างๆ จำนวน 8 คำสั่งไปยัง LCD
- **Read Command** เป็นการอ่านค่าสถานะความพร้อมของ LCD
- **Write Data** เป็นการเขียนรหัส ASCII แสดงผลบน LCD
- **Read Data** เป็นการอ่านรหัส ASCII กลับคืนมาจากจอ LCD

1 Clear Display (0x01)

คำสั่งนี้จะใช้สำหรับลบข้อความหน้าจอทั้งหมดพร้อมกับกำหนดตำแหน่งของ Cursor ให้กลับไปยังตำแหน่งศูนย์

2 Return Home (0x02-0x03)

มีผลทำให้ตำแหน่ง Cursor กลับไปอยู่ที่ตำแหน่งแรก โดย LCD จะกำหนดค่าให้กับตำแหน่งของ DD RAM Address ให้มีค่าศูนย์ ส่วนข้อความเดิมที่แสดงในจอจะไม่เกิดการเปลี่ยนแปลงใดๆ โดยคำสั่งนี้จะยึดค่าข้อมูลในบิต DB1=1 เป็นหลัก ส่วน DB0=0 หรือ 1 ก็ได้ ไม่มีผลต่อการทำงาน แต่ปกติควรให้ DB0=0 เสมอ

3 Entry Mode set (0x04-0x07)

ใช้กำหนดรูปแบบการเปลี่ยนแปลงข้อมูลและตำแหน่งของ Cursor เมื่อมีการเขียนข้อมูลให้ LCD แต่ละครั้ง

- **I/D** เป็นบิตที่กำหนดว่ามาทำการอ่านหรือเขียนข้อมูลให้กับ LCD แต่ละครั้ง จะให้ตำแหน่งของ Cursor (DD RAM Address) เพิ่มค่าหรือลดค่าลง 1 ค่า ถ้า I/D=1 Cursor เพิ่มค่า , I/D=0 Cursor ลดค่าลง 1 ค่า
- **S** ใช้กำหนดลักษณะการดันข้อมูลหน้าจอ ถ้า S=1 แต่ละครั้งที่เขียนข้อมูล Cursor จะอยู่กับที่ ส่วนข้อมูลจะถูกดันไปซ้าย 1 ตำแหน่ง และ ถ้า S=0 ข้อมูลจะอยู่กับที่แต่ตำแหน่ง Cursor จะเลื่อนไปขวา 1 ตำแหน่งแทน

4 Display ON/OFF Control (0x08-0x0F)

ใช้กำหนดรูปแบบการแสดงผลของ Cursor โดยมีบิตที่เกี่ยวข้องทั้งหมด ดังนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารนี้ทุกครั้งที่มีการนำไปใช้

- D เป็นบิตกำหนดให้เปิดหรือปิดหน้าจอแสดงผล โดยถ้าบิต D=1 หน้าจอ ON , D=0 หน้าจอ OFF
- C ใช้กำหนดการแสดงผลของ Cursor ถ้า C=1 Cursor แสดงผล , C=0 ไม่มีการแสดงผลของ Cursor
- B ใช้กำหนดการกระพริบของ Cursor ถ้า B=1 Cursor กระพริบ , B=0 ไม่มีการกระพริบของ Cursor

5 Cursor or Display Shift (0x10-0x1F)

ใช้กำหนดรูปแบบการเลื่อนของ Cursor และข้อมูลของ LCD เมื่อมีการอ่านหรือเขียนข้อมูล มีบิตที่เกี่ยวข้องทั้งหมด ดังนี้

ตารางที่ 1 ตารางแสดงบิตที่เกี่ยวข้องกับตำแหน่ง Cursor

S/C	R/L	ลักษณะการเลื่อนของ Cursor และข้อมูลหน้าจอ
0	0	ย้าย Cursor จากเดิมไปซ้าย 1 ตำแหน่ง
0	1	ย้าย Cursor จากเดิมไปขวา 1 ตำแหน่ง
1	0	แทรกตัวอักษรไปทางซ้าย 1 ตำแหน่ง
1	1	แทรกตัวอักษรไปทางขวา 1 ตำแหน่ง

6 Function Set (0x20-0x3F)

ใช้สำหรับกำหนดรูปแบบการเชื่อมต่อระหว่าง LCD กับ Microcontroller ว่าจะให้มีการติดต่อแบบ 4 บิต หรือ 8 บิต และขนาดหน้าจอกว้างเท่าใด

- DL บิตที่กำหนดว่า จะให้มีการติดต่อกันแบบ 4 หรือ 8 บิต ถ้า DL=0 จะติดต่อแบบ 4 บิต , DL=1 ติดต่อแบบ 8 บิต
- N ใช้กำหนดจำนวนบรรทัดในการแสดงผลของ LCD ถ้า N=0 แสดงผล 1 บรรทัด , N=1 จะติดต่อกันแบบ 8 บิต
- F ใช้กำหนดขนาดของตัวอักษร ถ้า F=1 ตัวอักษรมีขนาด 5x10 Dot , F=0 ตัวอักษรจะมีขนาด 5x7 Dot
- DB1 : DB0 กำหนดให้มีค่าเป็น 0 หรือ 1 ก็ได้ ไม่มีผลต่อการทำงานของคำสั่งนี้ แต่ควรกำหนดให้มีค่าเป็น 0 เสมอ

7 Set CG RAM Address (0x04-0x7F)

ใช้กำหนดตำแหน่ง Address ของ Character Generator หรือ CG RAM โดยต้องทำการกำหนดค่า Address ให้กับ CG RAM ทุกครั้ง เมื่อต้องการอ่านหรือเขียนให้กับค่า CG RAM โดยค่าตำแหน่ง Address กำหนดโดยบิต DB0-DB5 ของรหัสคำสั่ง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

8 Set DD RAM Address (0x80-0xFF)

ใช้กำหนดตำแหน่ง Address ของ Display Data RAM หรือ DD RAM หรือตำแหน่งของ Cursor ตามปกติเมื่ออ่านหรือเขียนตัวอักษร ตำแหน่ง DD RAM หรือ Cursor จะเพิ่มขึ้นทีละ 1 ตำแหน่งโดยอัตโนมัติ แต่เราสามารถกำหนดตำแหน่ง Address ให้กับ DD RAM ณ. ตำแหน่งใดๆ ของหน้าจอได้ โดยสามารถกำหนดค่า Address นี้ก่อนทำการอ่านหรือเขียนค่าให้กับ CG RAM เสมอ โดยค่าตำแหน่ง Address จะกำหนดโดยบิต DB0-DB6 ของรหัสคำสั่ง ซึ่งตำแหน่ง Address จะขึ้นอยู่กับการกำหนดค่า N ในคำสั่ง Function Set ด้วย คือ

- ถ้ากำหนดค่า N เป็น 0 (1 บรรทัด) ค่า Address จะอยู่ระหว่าง 00H-4FH
- ถ้ากำหนดค่า N เป็น 1 (2 บรรทัด) ค่า Address จะอยู่ระหว่าง 00H-27H ส่วนบรรทัดที่ 2 จะอยู่ที่ตำแหน่ง 40H-67H ตามลำดับ

9 Read Busy Flag & Address

ใช้สำหรับอ่านค่า Busy Flag ซึ่งเป็นบิตสถานะของ LCD Controller ว่ากำลังทำงานอยู่หรือพร้อมที่จะรับข้อมูลจากไมโครคอนโทรลเลอร์ ซึ่งทุกครั้งที่ต้องการติดต่อกับ LCD นั้นจะต้องมาอ่านค่าสถานะ Busy Flag ก่อนเสมอ โดยถ้าไม่ทำการตรวจสอบความพร้อมของ LCD ก่อนอาจจะมีการทำงานผิดพลาดได้

- BF=1 แสดงว่า LCD กำลังทำงานอยู่ ไม่สามารถส่งคำสั่งหรือข้อมูลในเวลานี้ได้
- BF=0 แสดงว่า LCD พร้อมรับคำสั่งหรือข้อมูลจากไมโครคอนโทรลเลอร์ได้

10 Write Data to CG RAM or DD RAM

ใช้สำหรับเขียนข้อมูลหรือรหัสตัวอักษรให้กับ LCD โดยเมื่อเขียนข้อมูลให้กับ LCD แต่ละครั้ง จะทำให้ตำแหน่ง Address เพิ่มหรือลดครั้งละ 1 ตำแหน่ง ขึ้นอยู่กับการกำหนดคำสั่ง Entry Mode จะขึ้นอยู่กับว่า ก่อนใช้คำสั่งนี้ ได้ทำการกำหนดค่าตำแหน่ง Address เป็นของ CG RAM หรือ DD RAM

11 Read Data From CG RAM or DD RAM

ใช้สำหรับอ่านข้อมูลจาก CG RAM หรือ DD RAM ทุกครั้งก่อนที่จะทำการอ่านค่าจาก CG RAM หรือ DD RAM ต้องทำการกำหนดค่า Address ที่จะอ่านก่อนทุกครั้งว่าต้องการอ่านค่าของ CG RAM หรือ DD RAM โดยใช้คำสั่ง Set DD RAM Address หรือ Set CG RAM Address

2.9.7.5 คำสั่ง LCD ของ LCD_4_BIT_Library

LCD_4BIT เป็น Library ใช้สำหรับติดต่อสั่งงานกับจอแสดงผล LCD แบบ Character ทุกรุ่น ที่ใช้ Driver ที่มีคำสั่งเข้ากันได้กับ HD44780 ของ Hitachi โดยเชื่อมต่อวงจรของ LCD แบบ 4 บิต โดยใช้ขาสัญญาณ 6 หรือ 7 เส้น สามารถเลือกกำหนดใช้ขาใดๆก็ได้ อิสระจากคำสั่งในโปรแกรม ประกอบด้วยคำสั่งต่างๆ สำหรับติดต่อสั่งงาน และควบคุมการแสดงผลของ LCD ทั้งหมด 14 คำสั่ง มีรายละเอียดดังนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- void Initial (void);
- void Command (int value);
- void Print (char);
- void Print (const char []);
- void Print (uint8_t);
- void Print (int);
- void Print (unsigned int);
- void Print (long);
- void Print (unsigned long);
- void Print (long , int);
- int ReadDisplay (void);
- int ReadAddress (void);
- void SetCursor (int Cursor);
- void ClearScreen (void);
- void ClearLCD (int Cursor , int length);

2.9.8 Stepping Motor

เป็นมอเตอร์แบบหนึ่งที่มีการหมุนเป็น Step กล่าวคือ เมื่อจ่ายสัญญาณให้แก่มอเตอร์อย่างถูกต้อง มอเตอร์จะหมุนไปเป็นจังหวะ ไม่เหมือนกับมอเตอร์ในแบบทั่วไป ที่เมื่อจ่ายไฟแล้วก็จะหมุนทันที ความละเอียดในการหมุนของ Stepping Motor นี้จะขึ้นอยู่กับองค์ประกอบสำคัญ 2 ประการ คือ โครงสร้างภายใน และ วงจรขับ

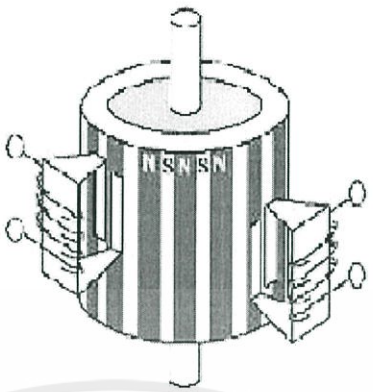
2.9.8.1 ประเภทของ Stepping Motor

ตามปกติแล้ว Stepping Motor ที่มีอยู่โดยส่วนใหญ่ สามารถจำแนกออก ตามลักษณะโครงสร้างได้ 3 แบบใหญ่ๆ คือ

1 แบบแม่เหล็กถาวร (Permanent Magnet Step Motor หรือ PM)

Stepping Motor แบบ PM นี้ จะใช้ส่วนของ Stator สำหรับพันขดลวดไว้หลายๆ โพล(Pole) และ จะมีส่วน Rotor เป็นรูปทรงกระบอกฟันเลื่อย ซึ่งส่วนของ Rotor จะทำจากแม่เหล็กถาวร โดยเมื่อป้อนไฟฟ้ากระแสตรงให้กับขดลวดที่พันอยู่บนส่วนของแกน Stator จะทำให้เกิดสนามแม่เหล็กไฟฟ้าขึ้น ในขดลวด Stator จะทำให้เกิดแรงดัน-ผลัก กับขั้วแม่เหล็กของส่วน Rotor จึงส่งผลให้แกน Rotor ของมอเตอร์สามารถเคลื่อนที่ไปได้ ลักษณะเด่นของมอเตอร์แบบนี้คือ จะให้แรงบิดสูง แต่มีความเร็วและระยะในการเคลื่อนที่ไม่ละเอียดมากนัก

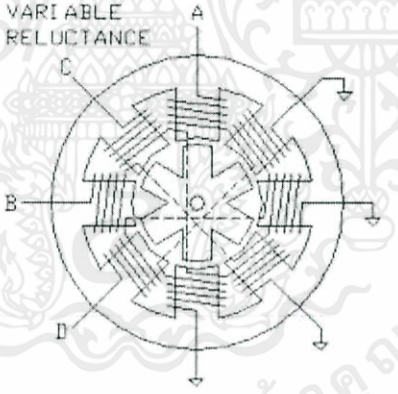
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.31 โครงสร้างของ PM Step Motor

2 แบบแปรผันค่ารีลัคแตนซ์ (Variable Reluctance Step Motor หรือ VR)

Stepping Motor แบบนี้ ส่วนของ Rotor สามารถเคลื่อนที่ได้อย่างงอิสระ โดย Rotor จพทำขึ้นจากสารแม่เหล็กกำลังอ่อน (Force Low Magnetic) มีลักษณะเป็นทรงกระบอกฟันเลื่อย ซึ่งจะมีความสัมพันธ์โดยตรง กับ จำนวนโพล (Pole) ของขดลวดใน Stator เพื่อใช้สำหรับกำหนดระยะของมุมที่จะหมุนไปในแต่ละครั้งของการเคลื่อนที่ เมื่อเราป้อนกระแสไฟฟ้าให้กับขดลวดใน Stator จะทำให้เกิดแรงบิด มอเตอร์แบบนี้มีความเร็วสูงและมีตำแหน่งของการเคลื่อนที่มีความแม่นยำสูง แต่มีแรงบิดน้อยกว่าแบบ PM

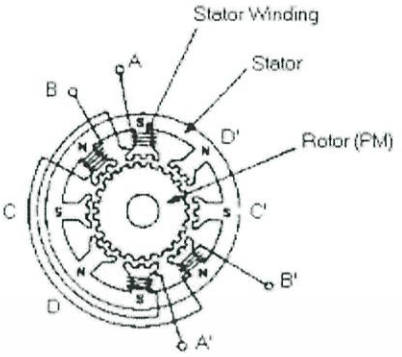


รูปที่ 2.32 โครงสร้างของ VR Step Motor

3 แบบผสม (Hybrid Stepping Motor หรือ H)

Stepping Motor แบบนี้ เป็นการนำเอาข้อดีของมอเตอร์แบบ PM และ VR มารวมเข้าด้วยกัน โดยจะออกแบบ ให้มีส่วนของ Stator มีลักษณะคล้ายกับของมอเตอร์แบบ VR สำหรับในส่วนของ Rotor จะมีหมวกหุ้มปลายทำด้วยสารแม่เหล็กกำลังสูง โดยในการออกแบบจะควบคุมขนาดรูปร่างของหมวกแม่เหล็กเป็นอย่างดีเพื่อให้ได้มุมการหมุนของมอเตอร์ในแต่ละครั้งที่ละเอียดและแม่นยำมากที่สุด คุณสมบัติเด่นของมอเตอร์แบบนี้ คือ ให้แรงบิดสูง มีขนาดกะทัดรัด มีระยะการหมุนที่ละเอียดและแม่นยำมาก แต่มีราคาสูงกว่าแบบอื่น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.33 โครงสร้างของ Hybrid Step Motor

2.9.8.2 หลักการทำงานสเต็ปมอเตอร์

Step Motor

สเต็ปมอเตอร์(Step Motor) หรือว่า สเต็บบิ้งมอเตอร์ (Stepping Motor) คือ อุปกรณ์ที่เคลื่อนที่เป็นจังหวะโดยการกระตุ้นด้วยวิธีทางแม่เหล็กไฟฟ้า ซึ่งเปลี่ยนเอาท์พุทจากสัญญาณดิจิตอลที่เป็นพัลส์ (Pulse)ไปเป็นการเคลื่อนที่เชิงกล สเต็ปมอเตอร์จะเคลื่อนที่เพิ่มขึ้น 1 สเต็ป เมื่อถูกกระตุ้นด้วยโวลท์เตจ (Voltage) ซึ่งส่วนมากเป็นไฟฟ้ากระแสตรง และเนื่องจากสเต็ปมอเตอร์สามารถควบคุมตำแหน่งได้ จึงไม่จำเป็นต้องใช้ตัววัดตำแหน่ง ในการป้อนกลับ สำหรับเอาท์พุทของสเต็ปมอเตอร์จะมีจำนวนเท่ากับจำนวนคำสั่งอินพุทซึ่งมีลักษณะเป็นพัลส์ (Pulse) ลักษณะการขับเคลื่อนจะหมุนรอบแกนได้ 360 องศา มีลักษณะเป็นสเต็ป โดยแต่ละสเต็ปจะขับเคลื่อนได้ 1, 1.5, 1.8 หรือ 2 องศา แล้วแต่ลักษณะที่นำไปใช้

2.9.8.3 การควบคุมการหมุนของสเต็ปมอเตอร์

การที่จะทำให้สเต็ปมอเตอร์หมุนต้องมีการป้อนสัญญาณพัลส์ที่ขดลวดแต่ละเฟสของสเต็ปมอเตอร์ให้ไปในทิศทางเดียวกัน หรือเรียงลำดับ (Sequence) ถ้าต้องการให้สเต็ปมอเตอร์หมุนกลับทิศทางจะต้องป้อนสัญญาณพัลส์ในทิศทางตรงข้ามกัน การควบคุมแบ่งออกเป็น 3 ลักษณะ ดังนี้

- **แบบเต็มสเต็ปหนึ่งเฟส (Full – Step 1 Phase)**

การควบคุมนี้เป็นการกระตุ้นขดลวดทีละขดเรียงตามลำดับ 1, 2, 3, 4 การกระตุ้นจะมีขดลวดขดเดียวที่ถูกกระตุ้นในเวลาหนึ่งเท่านั้น เช่น ขดลวด 1, 2, 3, 4 หรือถ้าต้องการให้หมุนสวนทิศทางกันจะกระตุ้นขดลวดที่ 4, 3, 2, 1 เป็นต้น การกระตุ้นแบบนี้จะทำงานง่ายที่สุดและกินกระแสไฟฟ้าน้อยที่สุด แสดงดังตารางที่ 2

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 2 การทำงานของสเต็ปมอเตอร์แบบเต็มสเต็ปหนึ่งเฟส

สเต็ปที่	เฟสที่ 1	เฟสที่ 2	เฟสที่ 3	เฟสที่ 4
1	ทำงาน	-	-	-
2	-	ทำงาน	-	-
3	-	-	ทำงาน	-
4	-	-	-	ทำงาน

- **แบบเต็มสเต็ปสองเฟส (Full – Step 2 Phase)**

การควบคุมลักษณะนี้เป็นการกระตุ้นขดลวดที่ละ 2 ขดที่อยู่ใกล้กันให้ทำงานพร้อมกัน และเรียงลำดับกัน เช่น ขดที่ 12, 23, 34, 41 หรือถ้าต้องการให้หมุดสวนทิศทางต้องกระตุ้นขดลวดที่ 14, 43, 32, 21ข้อดีของการกระตุ้นแบบนี้ คือ สเต็ปมอเตอร์จะมีแรงบิดมากกว่าแบบ 1 เฟส

ข้อเสีย การกระตุ้นแบบนี้จะต้องใช้กำลังไฟฟ้าเพิ่มขึ้นเป็น 2 เท่าของแบบ 1 เฟส เพราะต้องกระตุ้นขดลวด 2 ขดพร้อมกัน แสดงดังตารางที่ 3

ตารางที่ 3 การทำงานของสเต็ปมอเตอร์แบบเต็มสเต็ปสองเฟส

สเต็ปที่	เฟสที่ 1	เฟสที่ 2	เฟสที่ 3	เฟสที่ 4
1	ทำงาน	ทำงาน	-	-
2	-	ทำงาน	ทำงาน	-
3	-	-	ทำงาน	ทำงาน
4	ทำงาน	-	-	ทำงาน

- **แบบครึ่งสเต็ป (Half Step)**

การควบคุลักษณะนี้เป็นการกระตุ้นแบบ 1 เฟส กับ 2 เฟส มารวมกัน โดยการกระตุ้นจะเรียงลำดับกันไป เช่น ขดที่ 1, 12, 2, 23, 3, 34, 4, 41 หรือถ้าต้องการให้หมุนสวนทิศทางกันต้องกระตุ้นขดลวดที่ 1, 41, 4, 43, 3, 32, 2, 21

ข้อดี การกระตุ้นแบบนี้ สเต็ปมอเตอร์จะมีแรงบิดเพิ่มมากขึ้น ละเอียดขึ้นและการควบคุมตำแหน่งถูกต้องมากยิ่งขึ้น

ข้อเสีย การกระตุ้นแบบนี้ต้องใช้กำลังไฟฟ้าเพิ่มขึ้นเป็น 2 เท่า เหมือนกับการกระตุ้นแบบ 2 เฟส แสดงดังตารางที่ 4

ตารางที่ 4 การทำงานของสเต็ปมอเตอร์แบบครึ่งสเต็ป

สเต็ปที่	เฟสที่ 1	เฟสที่ 2	เฟสที่ 3	เฟสที่ 4
1	ทำงาน	-	-	-
2	ทำงาน	ทำงาน	-	-

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3	-	ทำงาน	-	-
4	-	ทำงาน	ทำงาน	-
5	-	-	ทำงาน	-
6	-	-	ทำงาน	ทำงาน
7	-	-	-	ทำงาน
8	ทำงาน	-	-	ทำงาน

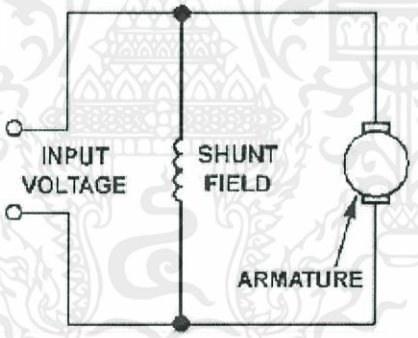
2.9.9 DC Motor

โดยปกติแล้ว DC Motor สามารถจำแนกออกได้หลายประเภท ขึ้นอยู่กับการออกแบบและวิธีการสร้างตัวมอเตอร์นั้นๆ แต่ในปัจจุบันนี้ DC Motor ที่มักพบเห็นกันโดยทั่วไปนี้มีอยู่ด้วยกัน 4 แบบ หลักๆ แบบขนาน (Shunt) แบบอนุกรม (Series) แบบผสมระหว่างขนานและอนุกรม (Compound) และ แบบแม่เหล็กถาวร (Permanent Magnet) โดยแต่ละแบบจะมีคุณสมบัติดังนี้

2.9.9.1 ประเภทของ DC Motor

1 SHUNT MOTOR

มอเตอร์แบบนี้สามารถปรับขนาดเส้นแรงแม่เหล็กได้อย่างอิสระต่อกระแสของ อาร์เมเจอร์ เป็นผลให้สามารถควบคุมค่าพารามิเตอร์ต่างๆ ของมอเตอร์ให้คงที่ได้ตลอดในช่วงพิสัยที่กว้าง ทำให้มอเตอร์ประเภทนี้นิยมนำไปใช้งานระบบเคลื่อนที่ๆต้องการแรงบิดสูง

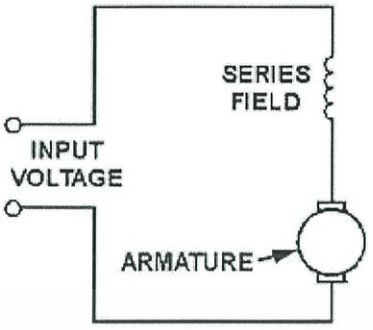


รูปที่ 2.34 โครงสร้างของ Shunt Motor

2 SERIES MOTOR

มอเตอร์แบบนี้จะมีเส้นแรงแม่เหล็กเป็นสัดส่วนกับกระแส ดังนั้นแส้นแรงแม่เหล็กจึงสามารถปรับค่าได้ ทำให้ความสัมพันธ์ระหว่างความเร็วและแรงบิดเป็นแบบ Nonlinear จึงเหมาะนำไปใช้งานประเภทที่ต้องการแรงบิดสูงแต่ความเร็วต่ำ และงานที่แรงบิดสูงและความเร็วสูง เช่น ระบบขับเคลื่อนของรถลาก เป็นต้น

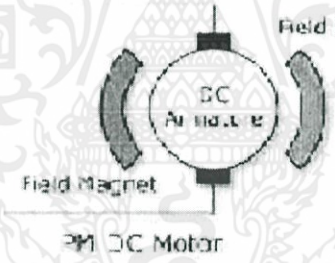
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.35 โครงสร้างของ Series Motor

3 PERMANENT MAGNET MOTOR

มอเตอร์แบบนี้จะใช้การกระตุ้น ฟิลด์ ของตัวมอเตอร์ซึ่งเป็น แม่เหล็กถาวรซึ่งต่างจากทั้งสองแบบที่กล่าวมาแล้ว ซึ่งมอเตอร์แบบนี้จะใช้ฟิลด์ของเส้นแรงแม่เหล็กมีค่า คงที่ ดังนั้น อัตราส่วนระหว่างกระแสอาร์มาเจอร์และแรงบิดจึงมีค่าคงที่ตามไปด้วย ซึ่งมอเตอร์แบบนี้มีข้อดี คือ ไม่มีการสูญเสียกำลังในฟิลด์ ทำให้มีประสิทธิภาพสูงและมีขนาดเล็กกว่าเมื่อเทียบกับมอเตอร์แบบ ที่ใช้ขดลวดในการกระตุ้นในขณะที่มีขนาดกำลังเท่ากัน จึงเหมาะกับงานที่ต้องการแรงบิดของโหลดสูง



รูปที่ 2.36 โครงสร้างของ Permanent Magnet Motor

2.9.9.2 วิธีการควบคุมความเร็วของ DC Motor

1 Armature Voltage Control

เป็นการคบคุมแรงดันไฟตรงที่จะป้อนให้กับขดลวดอาร์มาเจอร์ เนื่องจากความเร็วของ DC Motor นั้นจะแปรผันตรงกับแรงบิดที่ป้อนให้กับขดลวดอาร์มาเจอร์ ดังนั้นสามารถควบคุมความเร็วช่วงที่มอเตอร์มีความเร็วต่ำกว่าความเร็วที่กำหนด (Base Speed) หรือ n Base การควบคุมความเร็วของมอเตอร์แบบนี้จะทำให้แรงบิดสูงสุดส่วนกำลังของมอเตอร์จะเพิ่มขึ้นตามค่าของความเร็วซึ่งแปรผันเป็นเส้นตรงโดยจะมีกำลังสูงสุดตามที่กำหนด

2 Field Control

เป็นการควบคุมความเข้มของสนามแม่เหล็ก ซึ่งเป็นเทคนิคการควบคุมความเร็วของมอเตอร์ในย่านที่สูงกว่าความเร็วที่กำหนดไว้ ซึ่งจะทำให้การควบคุมกระแสของขดลวดสำหรับสร้างสนามแม่เหล็กให้กับมอเตอร์ วิธีนี้เมื่อต้องการเพิ่มความเร็วจะต้องลดขนาดกระแสในขดลวดลง ซึ่งผล

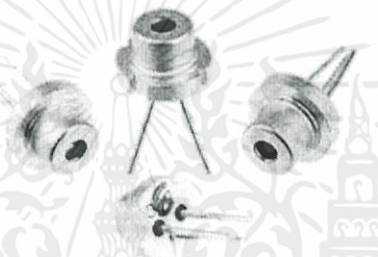
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้นอีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากการลดความเข้มของสารแม่เหล็กในขดลวดของมอเตอร์จะมีผลทำให้แรงบิดสูงสุดของมอเตอร์ลดลงในขณะที่กำลังออกของมอเตอร์ไม่เปลี่ยนแปลง วิธีนี้นิยมใช้กับ Load ที่ต้องการความเร็วสูง โดยที่แรงบิดของ Load จะต้อง ลดลงเมื่อความเร็วเพิ่มขึ้น

2.9.10 Laser Diode

เลเซอร์ชนิดสารกึ่งตัวนำ (เลเซอร์ไดโอด)

เลเซอร์ไดโอดมีโครงสร้างรอยต่อแบบ p-n ของสารกึ่งตัวนำทำจากสารประกอบกึ่งตัวนำ เช่น GaAs, InP มีขนาดเล็กมาก สิ่งประดิษฐ์มีปริมาตรเพียงไม่ถึง 1 ลูกบาศก์มิลลิเมตร บรรจุอยู่ในกล่องโลหะ (metal package) ที่มีขาติดอยู่เพื่อใช้ป้อนกระแสไฟ รูปที่ 1 แสดงตัวอย่างเลเซอร์ไดโอดที่มีขายในท้องตลาด



รูปที่ 2.37 ตัวอย่างเลเซอร์ไดโอด

คุณสมบัติเด่นของเลเซอร์ไดโอดซึ่งเลเซอร์ชนิดอื่นไม่มี ได้แก่

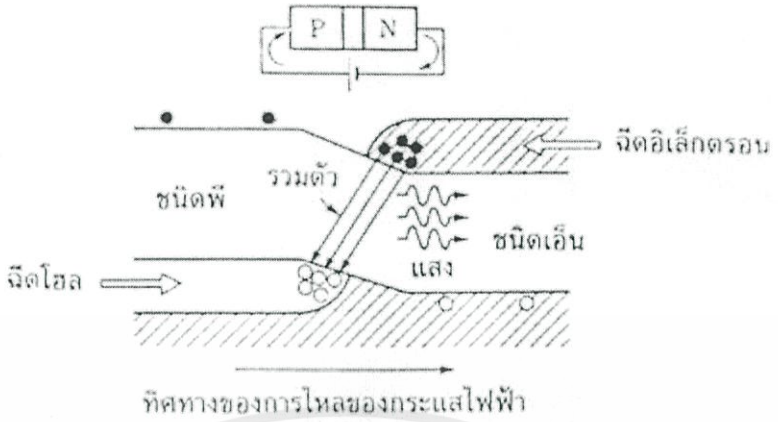
1. มีขนาดเล็กและเบามาก
2. มีประสิทธิภาพสูง
3. สามารถมอดูเลตความเข้มของการเปล่งแสงด้วยกระแสไฟฟ้าได้โดยตรง
4. มีอายุการใช้งานยาวนาน
5. ผลิตจำนวนมาได้ง่ายและราคาถูก

ปัจจุบันเลเซอร์ไดโอดถูกนำไปใช้งานอย่างกว้างขวาง เช่น การสื่อสารด้วยแสง การวัด สารสนเทศ คอมแพกต์ดิสก์ เลเซอร์ดิสก์ ซีดีรอม การแพทย์ การบันเทิง ฯลฯ

2.9.10.1 กลไกการเปล่งแสงของเลเซอร์ไดโอด

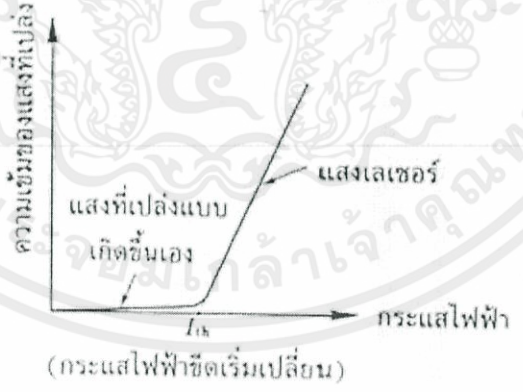
ในการกระตุ้นพาหะเพื่อให้เกิดเลเซอร์นั้น จะใช้วิธีฉีดกระแสไฟฟ้าจากภายนอกให้ไหลเข้าสู่รอยต่อ p-n โดยป้อนแรงดันไฟฟ้าแบบไบแอสตาม นั่นคือ ป้อนแรงดันบวกเข้าด้าน p และป้อนแรงดันลบเข้าด้าน n จะทำให้แถบพลังงานเปลี่ยนรูป อิเล็กตรอนจะไหลจากด้าน n เข้าสู่รอยต่อ และโหลจะไหลจากด้าน p เข้าสู่รอยต่อ ที่บริเวณรอยต่อจะมีคู่ของอิเล็กตรอนและโฮลเกิดขึ้นจำนวนมาก เมื่ออิเล็กตรอน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.38 รอยต่อพีเอ็นที่กำเนิดแสงเลเซอร์ในเลเซอร์ไดโอด

และโฮลรวมตัวกัน (recombine) จะเกิดการเปล่งแสงออกมาด้วยพลังงานโฟตอนเท่ากับช่องว่างพลังงานของรอยต่อ ($E_g = hv$) หลักการเปล่งแสงเช่นนี้คล้ายกับกรณีของ LED แต่การทำให้เกิดเลเซอร์นั้นจะต้องฉีดกระแสไฟฟ้าให้มากเป็นพิเศษ ถ้ากระแสไฟฟ้ามีค่าน้อย การเปล่งแสงจะเป็นแบบเกิดขึ้นเอง (spontaneous) และเมื่อเพิ่มกระแสไฟฟ้าให้สูงกว่ากระแสไฟขีดเริ่มเปลี่ยน (threshold current) (I_{th}) จะเกิดปรากฏการณ์ประชากรผกผัน (population inversion) กล่าวคือ ถ้าจำนวนคู่ของอิเล็กตรอนและโฮลมีมากกว่าค่าๆหนึ่ง ก็จะทำให้เกิดการรวมตัวของอิเล็กตรอนและโฮลและเกิดการเปล่งแสงแบบเร่งเร้า (stimulated emission) ที่ด้านข้างภายนอกของรอยต่อ p-n จะถูกออกแบบให้เป็นผิวนอกแบบกระจก ให้ทำหน้าที่เป็นออปติคัลเรโซเนเตอร์ (optical resonator) แสงที่สะท้อนกลับไปมาภายในเรโซเรเตอร์นี้ จะกลายเป็นแสงเลเซอร์วิ่งออกสู่ภายนอกในที่สุด



รูปที่ 2.39 ลักษณะความสัมพันธ์ของความเข้มแสงและกระแสไฟฟ้าในเลเซอร์ไดโอด

2.9.10.2 ลักษณะสมบัติกำลังเอาต์พุตและกระแสไฟฟ้า

เราเรียกความสัมพันธ์ระหว่างกระแสไฟฟ้า (Injection current) ที่ฉีดเข้าสู่เลเซอร์ไอโอดและพลังงานเอาท์พุต (energy output) ว่า ลักษณะสมบัติความสัมพันธ์ของอินพุตและเอาต์พุต (input-output characteristics) ในรูปที่ 29 เมื่อเพิ่มแรงดันไฟฟ้าให้ใกล้เคียงค่าช่องว่างพลังงาน จะทำให้กระแสไฟฟ้า เริ่มไหลและจะเริ่มมีแสง เปล่งแบบเกิดขึ้นเองออกมาดังรูปที่ 30 และ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งไม่อนุญาตให้นำไปดัดแปลง เผยแพร่ หรือนำไปเผยแพร่ต่อในรูปแบบอื่นๆ

เมื่อเพิ่มกระแสอีกมากๆ จนสูงกว่าค่าขีดเริ่มเปลี่ยนค่าหนึ่งจะเกิดการ (lasing) และกำลังของแสงเอาท์พุตจะสูงขึ้นอย่างรวดเร็ว ค่ากระแสไฟฟ้าขีดเริ่มเปลี่ยนนี้จะขึ้นกับอุณหภูมิขณะทำงาน ถ้าอุณหภูมิสูงขึ้น จะทำให้อัตราขยาย (gain) ลดลงและจะทำให้กระแสไฟฟ้าขีดเริ่มเปลี่ยนมีค่าสูงขึ้น

ความสัมพันธ์ระหว่างกระแสไฟฟ้าขีดเริ่มเปลี่ยน (I_{th}) และอุณหภูมิของชั้นแอกทิฟ (T_j) คือ

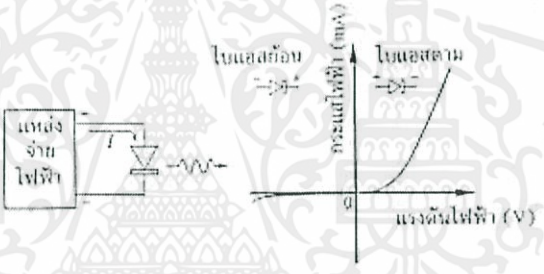
$$I_{th} = I_{tho} \exp(T_j / T_o) \qquad (2.53)$$

โดยที่ I_{th} : กระแสไฟฟ้าขีดเริ่มเปลี่ยนที่อุณหภูมิห้อง

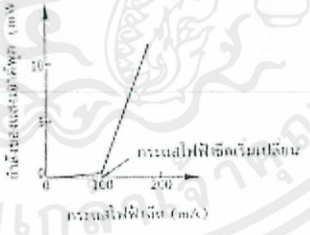
T_o : ค่าคงตัวที่ขึ้นกับวัสดุ มีชื่อเรียกว่า characteristic temperature

ตัวอย่างในกรณีของ GaAs : T_o = 100~160 K

In GaAs : T_o = 50~70 K



รูปที่ 2.40 ลักษณะสมบัติกระแสไฟฟ้าและแรงดันของเลเซอร์ไดโอด



รูปที่ 2.41 ลักษณะสมบัติกำลังของเลเซอร์เอาต์พุตและกระแสที่ฉีดเข้าเลเซอร์ไดโอด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 3
การออกแบบการทดลอง

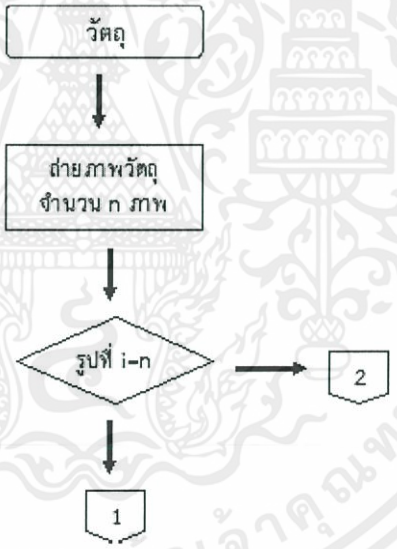
การทำงานแบบออกเป็น 2 ส่วน

คือ ส่วนของซอฟแวร์ เป็นการประมวลผลภาพที่ได้จากการถ่ายรูปเลเซอร์ที่กวาดไปวัตถุอย่างต่อเนื่องโดยใช้โปรแกรม MATLAB รวมถึงการติดต่อกล้องที่ใช้ในการเก็บภาพ และสองส่วนของฮาร์ดแวร์ใช้สำหรับการควบคุมมอเตอร์โดยใช้โปรแกรม arduino

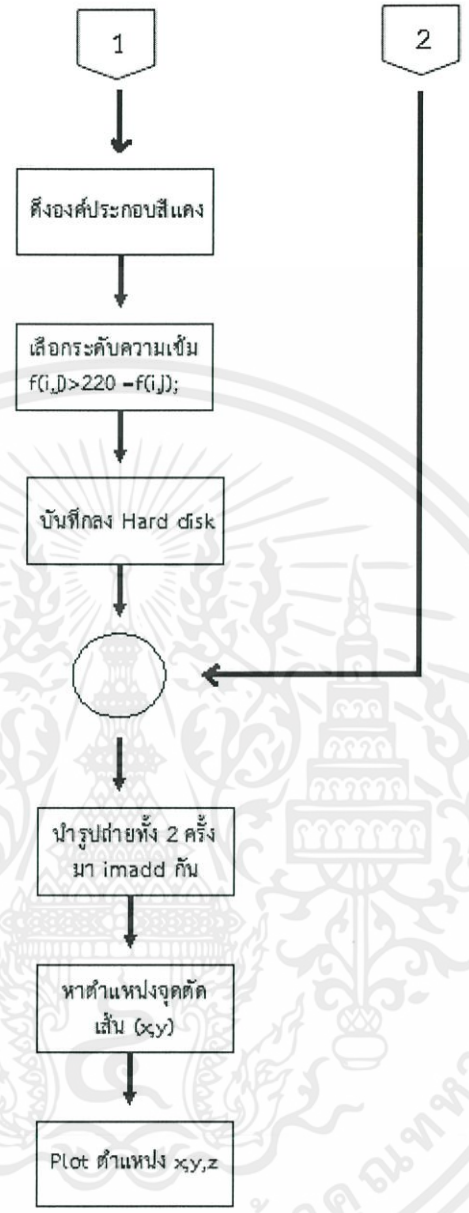
3.1 ขั้นตอนการออกแบบ Software

การทำงานในส่วน Software เป็นการเชื่อมต่อกับกล้องเข้ามาในระบบโดยใช้โปรแกรม MATLABการปรับปรุงภาพที่ได้จากการถ่ายวัตถุรวมถึงการพล็อตสามมิติ

3.1.1 ขั้นตอนการทำงาน



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.1 แสดงการทำงานส่วนซอฟแวร์

3.1.2. การเชื่อมต่อกล้องเข้ากับโปรแกรม MATLAB

เป็นการเชื่อมต่อกล้องเว็บแคม (Webcam) สองตัวเข้ากับระบบเพื่อทำการถ่ายวัตถุที่เราต้องการนำมาสแกน โดยการที่จะเชื่อมต่อกล้องทั้งสองตัวเข้ามาในระบบนั้น สามารถใช้วิธีการเดียวกับการเชื่อมต่อกล้องตัวเดียวเข้ามาในระบบ แตกต่างกันเพียงการเรียกใช้กล้องแต่ละตัวนั้น จะกำหนดชื่อขึ้นมาให้แตกต่างกัน เพื่อความสะดวกในการเรียกใช้กล้องทั้งสองตัว การเชื่อมต่อกล้องเข้ากับระบบ ทำได้มากกว่าสองตัวถ้า คอมพิวเตอร์ของเรามีพอร์ตในการเชื่อมต่อได้หลายพอร์ต

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ขั้นตอนการเชื่อมต่อกล้องกับโปรแกรม MATLAB

โปรแกรม MATLAB มีคำสั่งเรียกใช้กล้องที่มีอยู่ในระบบ ดังนี้

1. ทำการตรวจสอบก่อนว่ามีกล้องอยู่ในระบบ โดยใช้คำสั่ง

```
>>imaqhwinfo
ans =
        InstalledAdaptors: {'coreco'  'winvideo'}
        MATLABVersion: '7.10 (R2010a)'
        ToolboxName: 'Image Acquisition Toolbox'
        ToolboxVersion: '3.5 (R2010a)'
```

2. ตรวจสอง ID ของอุปกรณ์ที่ต้องการใช้ ในกรณีนี้ใช้กล้อง USB ให้พิมพ์

```
>> info=imaqhwinfo('winvideo')
info =
        AdaptorDllName: [1x81 char]
        AdaptorDllVersion: '3.5 (R2010a)'
        AdaptorName: 'winvideo'
        DeviceIDs: {[1]}          %จำนวนกล้องที่สามารถเรียกใช้ได้
        DeviceInfo: [1x1 struct]
```

3. ตรวจดูรูปแบบของสัญญาณวิดีโอที่ใช้ โดยใช้ DeviceID

```
>> info=imaqhwinfo('winvideo',1) ; % ตัวเลข 1 แทนกล้องตัวที่ 1
info =
        DefaultFormat: 'YUY2_160x120'
        DeviceFileSupported: 0
        DeviceName: 'Lenovo EasyCamera'
        DeviceID: 1
        ObjectConstructor: 'videoinput('winvideo', 1)'
        SupportedFormats: {'YUY2_160x120'  'YUY2_320x240'  'YUY2_640x400'
'YUY2_640x480'}          % สามารถเลือกความความละเอียดของกล้องได้
```

4. สร้างอ๊อบเจ็คสำหรับสัญญาณวิดีโออินพุทโดยใช้คำสั่ง

```
>> vid=videoinput('winvideo','1','YUY2_640x400')      %ตั้งชื่อที่ใช้สำหรับเรียกใช้
กล้องๆ
Summary of Video Input Object Using 'Lenovo EasyCamera'.
  Acquisition Source(s):  input1 is available.
  Acquisition Parameters:  'input1' is the current selected source.
                           10 frames per trigger using the selected source.
                           'YUY2_640x400' video data to be logged upon START.
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารนี้ทุกครั้งที่มีการนำไปใช้

Grabbing first of every 1 frame(s).
Log data to 'memory' on trigger.

Trigger Parameters: 1 'immediate' trigger(s) on START.

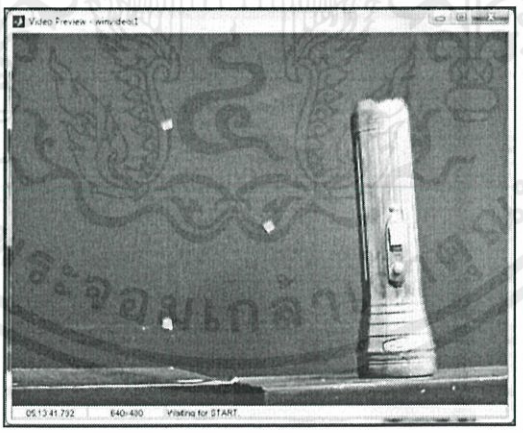
Status: Waiting for START.
0 frames acquired since starting.
0 frames available for GETDATA.

5. ตรวจดูคุณสมบัติของอ็อบเจ็คโดยใช้คำสั่ง

```
>> get(vid)
  General Settings:
    DeviceID = 1
....
....
  Acquisition Sources:
    SelectedSourceName = input1
    Source = [1x1 videosource]
```

6. ตรวจสอบความถูกต้องของการติดตั้งด้วยคำสั่ง

```
>>Preview(vid)
```



รูปที่ 3.2 ภาพที่ได้จากกล้องที่เราเรียกใช้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.1.3. การเริ่มต้นใช้งานกล้องผ่าน MATLAB

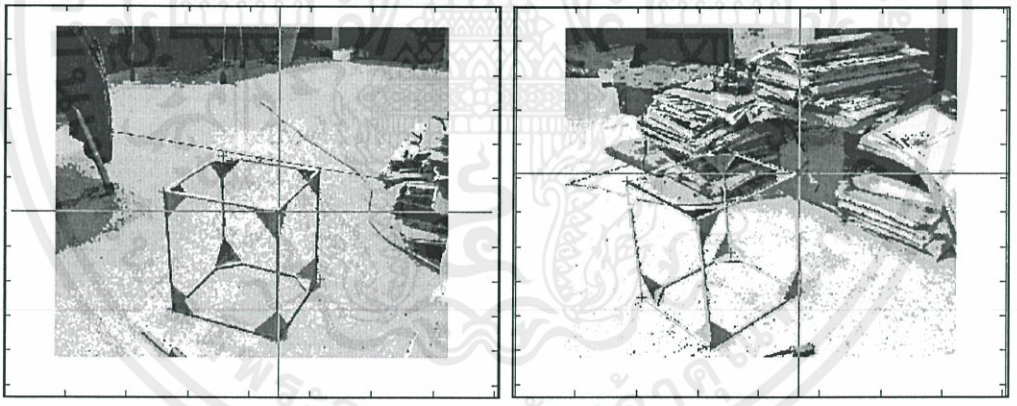
หลังจากเชื่อมต่อกล้องเข้ากับโปรแกรม MATLAB จะมีชื่อที่ถูกตั้งขึ้นเฉพาะกล้องแต่ละตัวสำหรับการเรียกใช้กล้องตัวที่เราต้องการ

```
Imaqhwinfo                                          % ตรวจสอบกล้องในระบบ
imaqhwinfo('winvideo')                              % ตรวจสอบ ID ของอุปกรณ์
vid=imaqhwinfo('winvideo',1)                        % กล้องที่เราเลือกใช้รูปแบบใด
vid=videoinput('winvideo','1','RGB24_640x480')      % กำหนดชื่อสำหรับเรียกใช้กล้อง
start(vid)                                          % เริ่มการทำงาน
preview(vid)                                        % โชว์การทำงานของกล้อง
stop(vid)
```

3.1.4 calibration

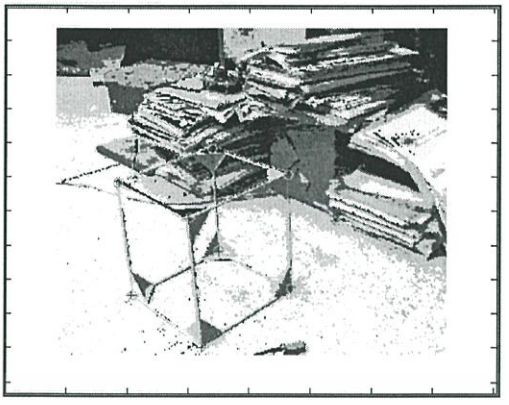
การหาจุดตัดในฉาก 3 มิติ ที่รู้ตำแหน่งและภาพที่เกิดขึ้นของจุด แล้วทำการหาพารามิเตอร์ภายในและพารามิเตอร์ภายนอก

ในการทดลองครั้งนี้ ใช้การปรับเทียบกล้องโดยการใช้กล่องลูกบาศก์ที่มีขนาดใกล้เคียงกับวัตถุที่จะนำมาทำการสแกน ทำการคลิ๊กลงบนมุมของกล่องทั้ง 8 ซึ่งลำดับการคลิ๊กที่มุมในแต่ละครั้งภาพที่หนึ่งและภาพที่สองต้องมีความสอดคล้องกัน



รูปที่ 3.3 กล่องลูกบาศก์ที่ใช้ในการปรับเทียบกล้อง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

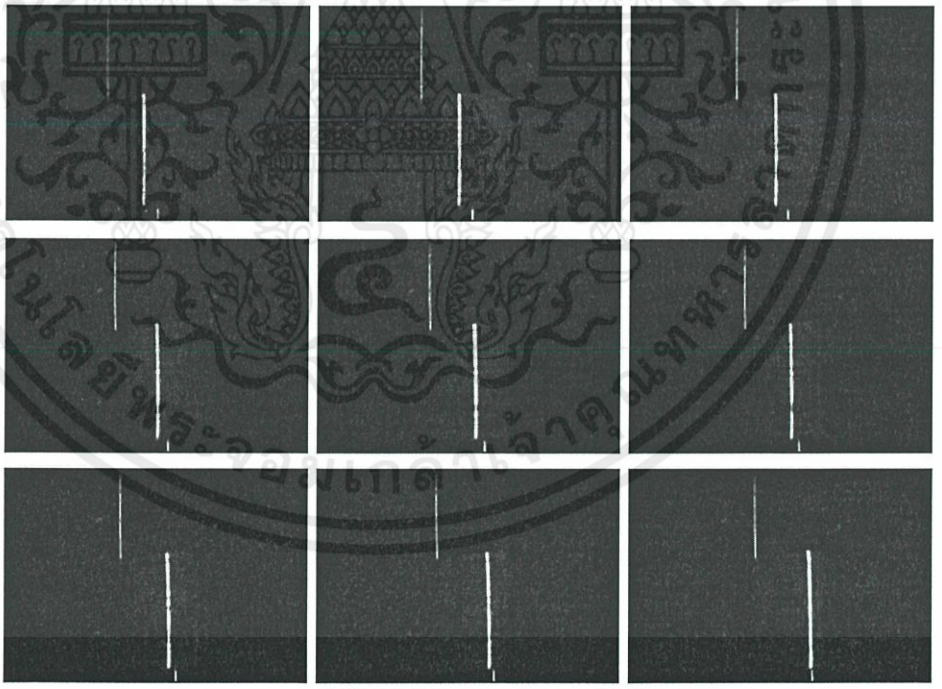


รูปที่ 3.4 การ Reproject หลังจากทำการปรับเทียบกล้อง

หลังจากทำการคลิ๊ก จะปรากฏจุดสีแดงขึ้น ซึ่งเป็นจุดที่เกิดจากการคลิ๊กเมาส์ และวงกลมคือพิกัดในการ Reproject การ Reproject เป็นการตรวจสอบว่าค่าพารามิเตอร์ของกล้องที่เราได้มามีความถูกต้องหรือไม่

3.1.5 การรับคำสั่งถ่ายภาพต่อเนื่อง

ทำการถ่ายภาพต่อเนื่องตามจำนวนภาพที่กำหนดขึ้น โดยภาพที่ได้จากการถ่ายต่อเนื่อง เป็นภาพที่เกิดจากกวาดเลเซอร์ลงไปบนพื้นผิวของวัตถุ แต่ละตำแหน่งที่แตกต่างกันไป



รูปที่ 3.5 ตัวอย่างการถ่ายภาต่อเนื่อง ทั้งหมด 9 ภาพ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.1.6. การปรับปรุงภาพ : การดึงองค์ประกอบสีแดง

จำนวนภาพที่ได้หลังจากการถ่ายภาพต่อเนื่อง นำภาพทั้งหมดที่ได้มาดึงองค์ประกอบสีออกมาแต่ละสี โดยองค์ประกอบของสี แบ่งออกเป็น 3 องค์ประกอบ องค์ประกอบสีแดง ® องค์ประกอบสีเขียว(G) องค์ประกอบสีฟ้า (B) เราต้องการเฉพาะองค์ประกอบสีแดงของ laser การดึงองค์ประกอบสีแดงของเลเซอร์มาทำให้เราได้ภาพที่มีตำแหน่งของเลเซอร์ปรากฏอยู่บนพื้นผิวของวัตถุ

โค้ดที่ใช้สำหรับการดึงองค์ประกอบสีแดง

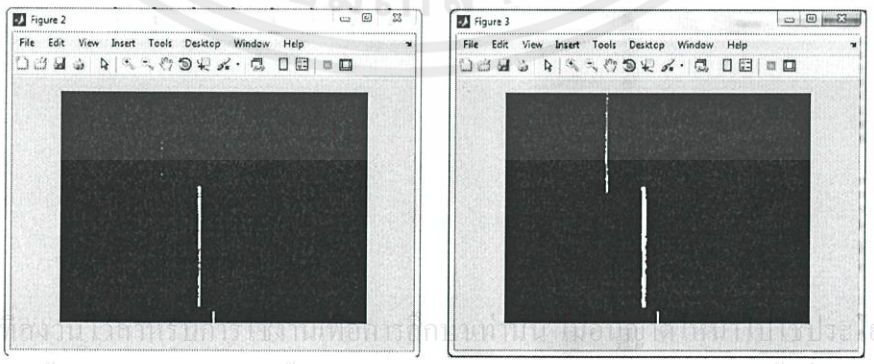
```
f=imread('A.bmp')          %อ่านภาพที่เราต้องการ
f1=f (:,:,1);              % การดึงองค์ประกอบสีแดงออกมาจากรูปภาพ
```

3.1.7. ปรับปรุงภาพ : เลือกระดับความเข้ม

การเลือกระดับความเข้มแสงแตกต่างกันไปทำให้ภาพที่ได้แตกต่างกันไป โดยภาพที่ได้จากการดึงองค์ประกอบ จะเหลือเพียงแค่องค์ประกอบสีแดงในภาพ ภาพที่เราต้องการนำไปใช้ในขั้นตอนต่อไป คือ การทำความหนาของเส้นให้บางลงนั้นต้องการภาพองค์ประกอบสีแดงที่มีความเข้มของแสงเลเซอร์ จึงต้องทำการกำหนดระดับความเข้มของแสงขึ้นมา เพราะแสงเลเซอร์มีความเข้มสูง

โค้ดที่ใช้สำหรับการเลือกระดับความเข้ม

```
for i=1:size(oneredleft,1)
    for j=1:size(oneredleft,2)
        if oneredleft(I,j)>=250        % สามารถกำหนดระดับความเข้มที่เราต้องการ
          oneredleft(I,j)= 255;
        else
          oneredleft(I,j)=0;
        end
    end
end
```



(A) (B)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

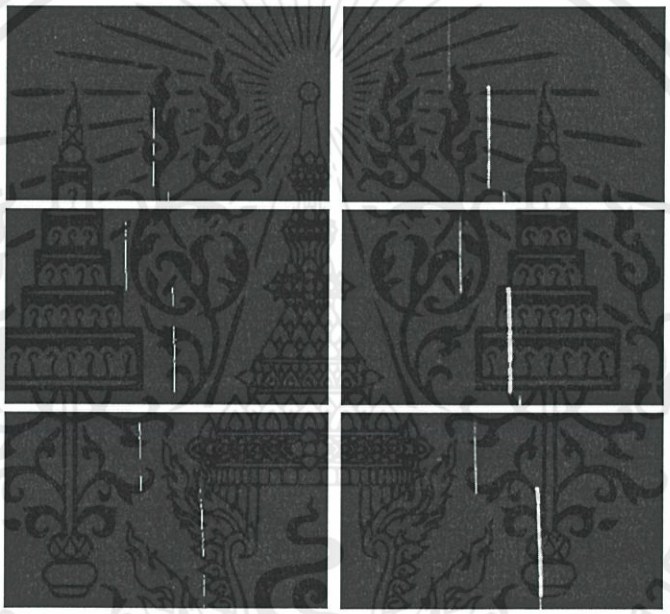
รูปที่ 3.6 การเลือกระดับความเข้ม ภาพ (A) ระดับความเข้มที่ 250, ภาพ (B) ระดับความเข้มที่ 100

3.1.8. ปรับปรุงภาพ : การลดขนาดของเส้น

วิธีการลดขนาดของเส้นมีหลายวิธี แต่การทดลองครั้งในใช้คำสั่ง bwmorph ซึ่งเป็นฟังก์ที่มีอยู่ในโปรแกรม MATLAB ในฟังก์ชัน bwmorph มีการปรับปรุงภาพหลายอย่างแต่การทำการลดขนาดของเส้นเราใช้คำสั่ง “thin”

โค้ดที่ใช้สำหรับการลดขนาดของเส้น

```
newthin2 = bwmorph(newthin,'thin');
        % newthin      ภาพที่เราต้องการลดขนาดเส้น
        % newthin2     ภาพที่ทำการลดขนาดเส้นเรียบร้อยแล้ว
        % 'thin'       ชนิดการปรับปรุงภาพ ในฟังก์ชัน bwmorph
```

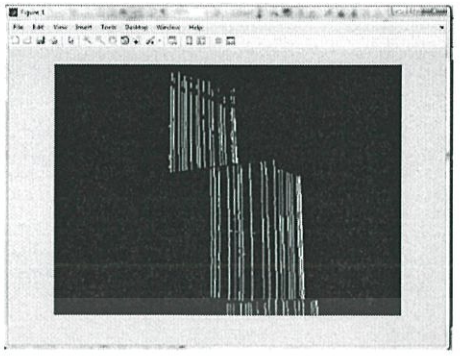


รูปที่ 3.7 หลังการปรับปรุงภาพ : ภาพต้นฉบับที่ได้จากการถ่าย

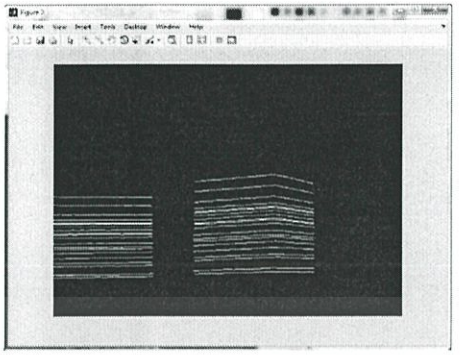
3.1.9 รวมภาพหลังจากการปรับปรุงภาพ

ภาพหลังจากทำการปรับปรุงภาพจะเป็นภาพที่มีการผ่านกระบวนการดึงองค์ประกอบสีแดง การเลือกระดับความเข้ม และการลดขนาดของเส้น หลังจากนั้นเรานำภาพทั้งหมดที่ได้มาทำการรวมกันโดยใช้คำสั่ง imadd หลังจากใช้คำสั่ง imadd ภาพที่ได้จะเป็นภาพของเส้นที่มีแนวตั้ง แนวนอนตัดกันปรากฏอยู่

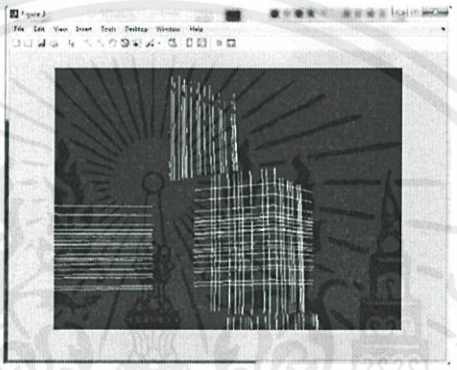
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



(A)



(B)



(C)

รูปที่ 3.8 (A) การ imadd ของเส้นแนวตั้ง , (B) การ imadd ของเส้นแนวนอน, (C) การ imadd ของเส้นทั้งหมด

รูปที่ 3.9 การ imadd กล่องกับภาพเส้นทั้งหมด

จากรูปจะเห็นได้ว่านอกจากภาพของเส้นแนวตั้งและแนวนอนตัดกันอยู่ จะปรากฏภาพของกล่องสี่เหลี่ยมขึ้นมาพร้อมกัน ซึ่งเป็นภาพที่เกิดจากการ .ใช้คำสั่ง imadd ภาพกล่องเข้าไป ซึ่งกล่องที่ปรากฏอยู่จะใช้สำหรับการ Calibration กล้องที่เราใช้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารนี้

3.1.10 การหาตำแหน่ง xy จากจุดตัดภาพในภาพ

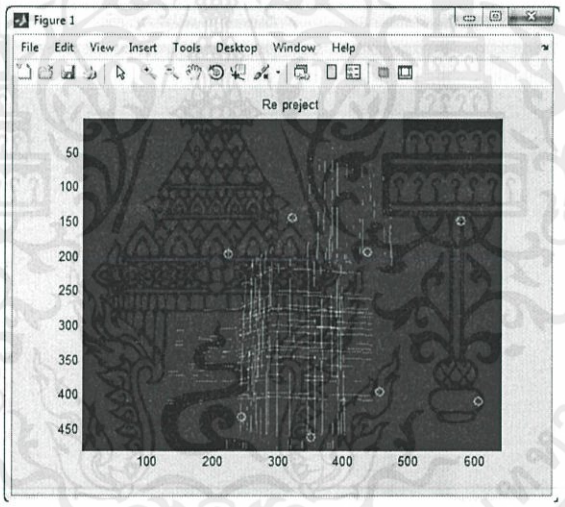
หลังจากได้ภาพที่เป็นจุดตัด นำภาพที่ได้มาหาจุดตัด xy โดยค่าจุดตัด xy ที่ได้นั้นเกิดจากการใช้ คำสั่ง ginput ซึ่ง คำสั่งนี้จะเป็นการรับค่าจากการคลิ๊กเมาส์ จำนวนครั้งที่เราทำการ

คลิ๊กจะถูกกำหนดก่อนเริ่มทำการคลิ๊ก โดยจะนวนทีทำการคลิ๊กจะต้องบวกเพิ่มขึ้นอีก จำนวนที่บวกเพิ่มเกิดจากจุดที่เราต้องการคลิ๊กลงบนภาพกล่องที่ปรากฏ เพื่อเป็นตำแหน่งอ้างอิง และเป็นการ calibration กล้องที่ใช้สำหรับถ่ายรูป

โค้ดสำหรับการรับค่าตำแหน่ง x y จากการคลิ๊กเมาส์

```
for     i=1:8
        [u (i,1),u(i,2)]=ginput(1);
        Plot (u (i,1),u(i,2),'r+');
end
% i คือจำนวนจุดที่เราทำการคลิ๊กเมาส์ที่จุดตัด+จำนวนจุดที่คลิ๊กเมาส์ลงบนภาพกล่อง
% ginput เป็นการรับค่าจากการ คลิ๊กเมาส์
```

จากโค้ด ค่าที่เราคลิ๊กเมาส์ที่จุดตัดต่างๆจะถูกเก็บไว้ที่ค่า u ซึ่งค่า u ที่ได้คือตำแหน่ง x y ที่เราจะนำไปใช้สำหรับการพล็อต สามมิติ



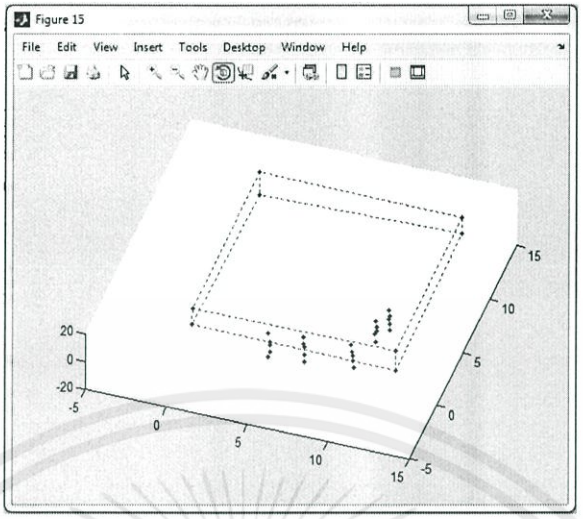
รูปที่ 3.10 เครื่องหมาย + แสดงการคลิ๊กเมาส์ลงบนจุดตัด

3.1.11 พล็อตสามมิติ

ค่าที่นำมาพล็อตสามมิติ คือค่าตำแหน่ง x y ที่ได้จากการคลิ๊กเมาส์ลงบนจุดตัด ซึ่งภาพที่ได้จากการพล็อตสามมิติจะมีความละเอียดมากเท่าใด ขึ้นอยู่กับจำนวนจุดที่เราทำการคลิ๊กลงบนภาพ

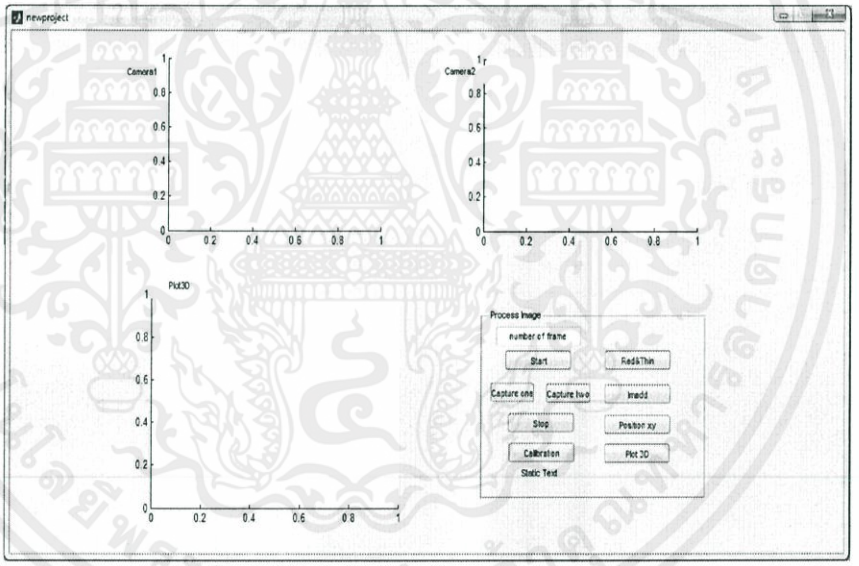
จากรูป 3.10 จำนวนที่เราทำการคลิ๊กลงไปมีจำนวนน้อยเมื่อเทียบกับจุดตัดที่เกิดขึ้นทั้งหมดในภาพ การพล็อตสามมิติจึงมีความละเอียดลดลงไปด้วย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.11 การพล็อตสามมิติ จากการคลิ๊กเมาส์จำนวน 25+8 ครั้ง

3.1.12 หน้าต่างโปรแกรม



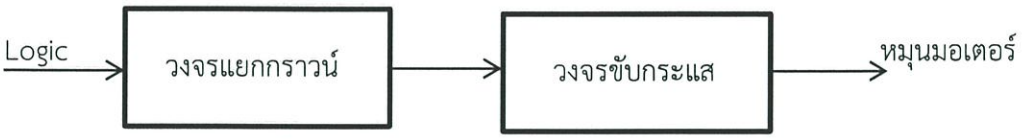
รูปที่ 3.12 หน้าต่างโปรแกรมที่ใช้สำหรับการถ่ายรูปต่อเนื่องและปรับปรุงภาพ

3.2 การออกแบบด้านฮาร์ดแวร์ (Hardware)

3.2.1. หลักการออกแบบโดยรวม

เริ่มจากไมโครคอนโทรลเลอร์ส่งค่าลอจิกหรือพัลส์ค่าๆหนึ่ง ออกทาง digital port แล้ว สัญญาณลอจิกจะผ่านวงจรแยกกราวน์ โดยใช้ Opto Isolator ซึ่งออปโต้ได้รับไฟเลี้ยงจากวงจรแหล่งจ่ายไฟ 5 โวลต์คงที่ หลังจากนั้นลอจิกจะทำให้ทรานซิสเตอร์ ON ซึ่งทรานซิสเตอร์ทำงานอยู่ใน Saturation Mode และลอจิกค่าดังกล่าวจะไปควบคุมการเคลื่อนที่ของมอเตอร์ ดังรูป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.13 บล็อกไดอะแกรมการทำงานโดยรวม

3.2.2. การออกแบบวงจรแหล่งจ่ายไฟคงที่ 5 โวลต์

พิกัดหม้อแปลงที่มีแรงดัน 24 Vac กระแส 1 A เนื่องจากขดลวดมอเตอร์ โดยแต่ละขดใช้กระแสเฉลี่ยขดละ 650 mA ฉะนั้นสเปคหม้อแปลงดังกล่าวสามารถใช้ได้

ในการออกแบบกำหนดให้ ripple 1.7% และ $I_{o(DC)}$ = 700 mA , $V_{o(dc)}$ 31 V. จะได้

จาก $$\Delta V_{(pp)} = \frac{2\sqrt{3} \times \%ripple \times Vo(dc)}{100} \quad (2.54)$$

$$= \frac{2\sqrt{3} \times 1.7 \times 31}{100}$$

$$\Delta V_{(pp)} \leq 1.82 \ V_{(pp)}$$

สำหรับ Full Wave

$$C \geq \frac{I_o}{F_r \Delta V_{(pp)}} = \frac{700mA}{2x50x1.82} \approx 4 \times 10^{-3} \text{ F.} \quad (\text{เลือก } 4700 \ \mu\text{F.})$$

คำนวณหาอัตราทนแรงดันของตัวเก็บประจุ (C_w)

$$C_w \geq 1.25 \ (|V_o| + 0.5\Delta V_{(pp)})$$

$$C_w \geq 1.25 \ (31 + 0.5x1.82) \approx 39.88 \text{ V.} \quad (\text{เลือกทนได้ 50 V.})$$

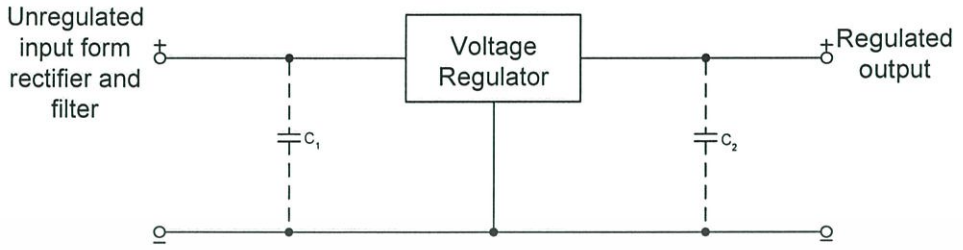
$$V_{AC\ (rms)} = \frac{|V_o| + 0.5\Delta V_{(pp)} + V_{F(diode)}}{\sqrt{2}} = \frac{31 + (0.5 \times 1.82) + 1.4}{\sqrt{2}} = 23.55 \ V_{rms}$$

FWB ; พิกัดกระแสของขดลวดทุติยภูมิ (I_{AC}) = $1.8I_{o\ (DC)}$ =1.8 x 700 mA = 1.26 $A_{(rms)}$

ฉะนั้นสามารถใช้ Bridge Diode ทนค่ากระแสไบอัสสูงสุด 2 A. และแรงดันพังทลาย (Breakdown Voltage) ที่ 50 V. และ พิกัดหม้อแปลงที่มีแรงดัน 24 $Vac_{(rms)}$ กระแส 1 A_{rms} ได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.2.3. วงจรรักษาแรงดันใช้ไอซีรักษาแรงดัน(IC Regulator)

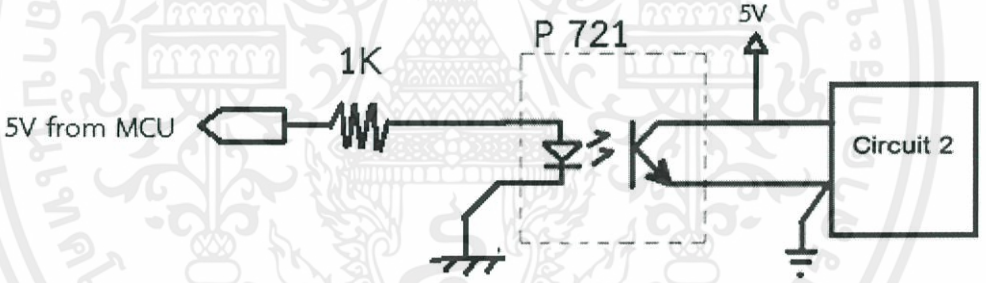


รูปที่ 3.14 พื้นฐานในการต่อ IC Regulator

ใช้ IC 7805 ในการออกแบบวงจรรักษาแรงดัน โดยใช้แหล่งจ่ายไฟจากการออกแบบที่กล่าวไปโดยแรงดันที่ผ่านวงจร filter คือ $\sqrt{2}\ Vac_{(rms)} - \frac{\Delta V_{(pp)}}{2} - 1.4 = 31.63$ V. เมื่อผ่าน IC 7805 จะได้แรงดัน 5 โวลต์คงที่

3.2.4. การออกแบบวงจรแยกกราวน์ (Ground Isolation)

ใช้ Optocoupler ชนิด Phototransistor เบอร์ P721 ในการ transfer สัญญาณลอจิกจากไมโครคอนโทรลเลอร์



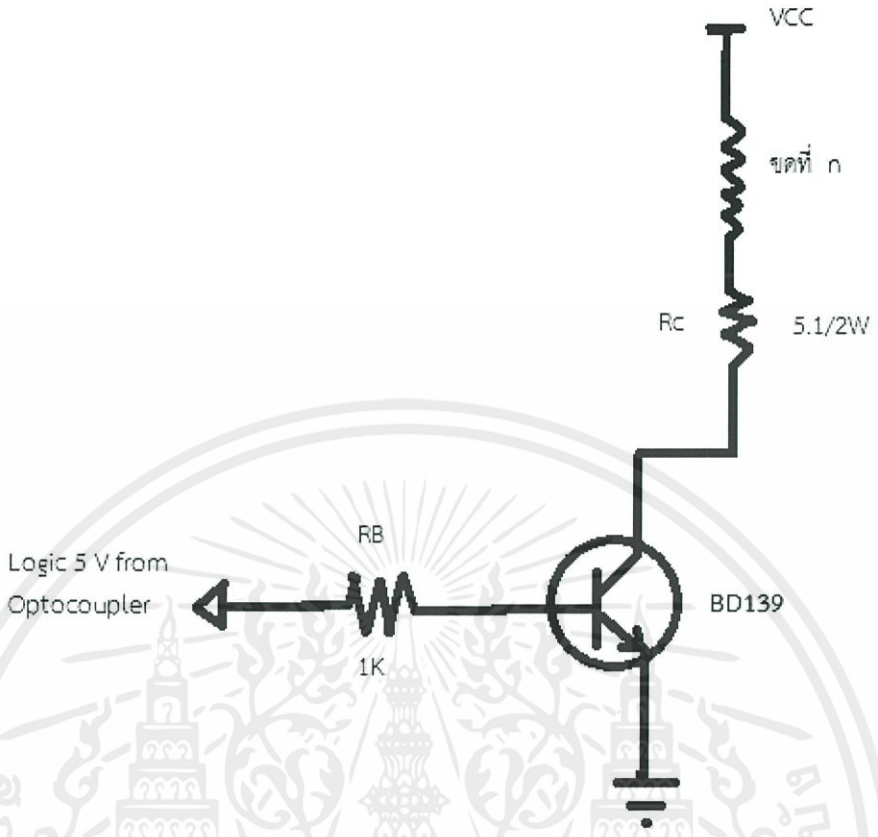
รูปที่ 3.15 การใช้งานออปโต้ทรานซิสเตอร์

จาก datasheet Optocoupler เบอร์ P721 Forward Voltage (V_F) 1.15-1.4 V. และ Maximum DC Forward Current (I_F) = 50 mA. ฉะนั้นลอจิกขนาด 5 V. จากไมโครคอนโทรลเลอร์ และกำหนดให้ I_F = 3.8 mA. ฉะนั้นจะได้ R = $\frac{5-1.15}{3.8x10^{-3}} \approx 1K\Omega$. ดังรูปที่ 3.15

3.2.5 วงจรขับกระแส

ใช้ทรานซิสเตอร์ เป็นตัวขับกระแส โดยให้ทรานซิสเตอร์ทำงานในโหมด Saturation Mode เสมือนทรานซิสเตอร์ทำงานเป็นสวิตช์ ON-OFF ตามลอจิกที่ส่งผ่านโดยไมโครคอนโทรลเลอร์ โดยการออกแบบขดลวดแต่ละขดของมอเตอร์ต้องการกระแสเฉลี่ยขดละประมาน 600 mA. (load current (I_L = 600 mA.)) ดังนั้นทรานซิสเตอร์จะต้องจ่ายกระแสได้อย่างน้อยที่สุด 600 mA. ในการออกแบบจะใช้ทรานซิสเตอร์เบอร์ BD139 จ่ายกระแสสูงสุดที่ 1.5 mA. ดังรูปที่ 34

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.16 วงจรขับกระแสโดยใช้ทรานซิสเตอร์

การคำนวณหาค่าพารามิเตอร์อุปกรณ์

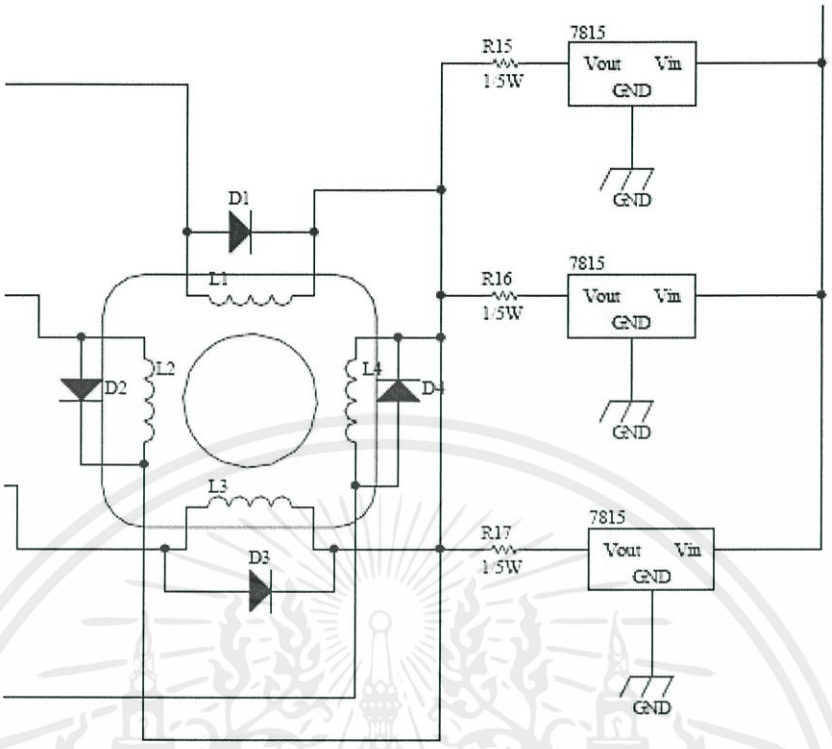
กำหนดให้ ต้องการ V_{CC} = 15 V. , I_C = 600 mA.

1. หาค่า R_C จาก Load Current (I_L) โดยที่ $R_C = \frac{V_{cc}}{I_c} = \frac{15}{600x10^{-3}}$ = 25 Ω. (เลือก R=5.1/2W เนื่องจากต้องการให้ motor ดึงกระแสสูงสุด)

2. หาค่า I_B จาก ($I_C = \beta I_B$) จะได้ $I_B = \frac{I_C}{\beta} = \frac{0.6\ A}{70}$ = 0.00857 A. (หรือ 8.6 mA. ดังนั้น เลือกที่ 5 mA) จะได้ $R_B = \frac{5-0.7}{5\ mA}$ = 860 Ω. (เลือก 1KΩ.)

นอกจากนี้เราสามารถใช้ IC 7815 ช่วยในการจ่ายกระแส ดังรูปที่ 3.16

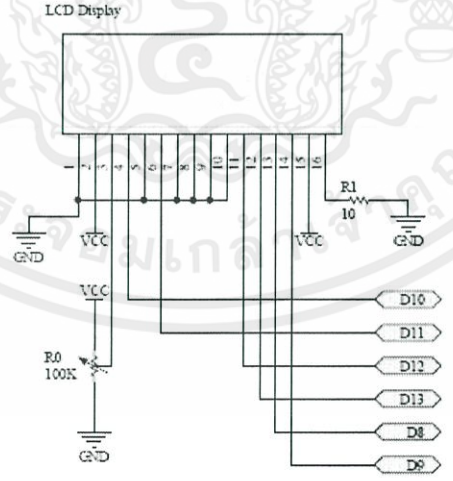
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.17 การใช้ *IC 7815* ช่วยจ่ายกระแสให้กับมอเตอร์

3.2.6 การออกแบบวงจรแสดงผลจอ LCD แบบ 4 บิต

ในการออกแบบวงจรแสดงผล LCD จะใช้สัญญาณทั้งหมด 6 เส้น คือ port digital 8,9,10,11,12,13 และสามารถปรับความเข้มหน้าจอแสดงผล โดยใช้ตัวต้านทานปรับค่าได้ ดังรูป

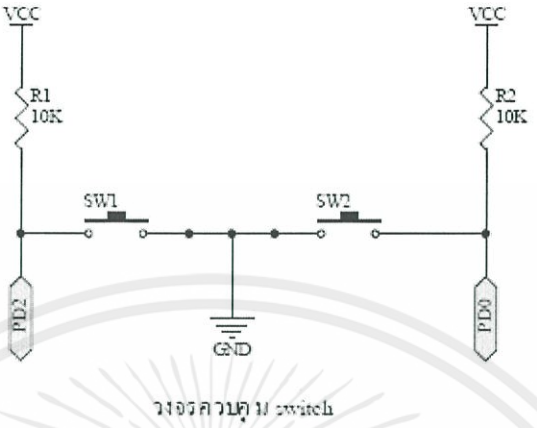


รูปที่ 3.18 วงจรแสดงผลจอ *LCD Module*

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.2.7 การออกแบบวงจร detect switch

จะใช้ port digital 2 port คือ 0,2 ต่อกันแบบ active low ดังรูปที่ 37



วงจรควบคุม switch

รูปที่ 3.19 วงจรควบคุม switch

3.2.8 การควบคุมการหมุนของมอเตอร์

การออกแบบในส่วนการควบคุมมอเตอร์จะให้มอเตอร์หมุนแบบเต็มสเต็ปหนึ่งเฟส (Full – Step 1 Phase) โดยเป็นการกระตุ้นขดลวดทีละขดเรียงตามลำดับ 1, 2, 3, 4 การกระตุ้นจะมีขดลวดขดเดียวที่ถูกกระตุ้นในเวลาหนึ่งเท่านั้น เช่น ขดลวด 1, 2, 3, 4 หรือถ้าต้องการให้หมุนสวนทิศทางกันจะกระตุ้นขดลวดที่ 4, 3, 2, 1 เป็นต้น การกระตุ้นแบบนี้จะทำงาน ง่ายที่สุดและกินกระแสไฟฟ้าน้อยที่สุด โดยใช้ digital port 4,5,6 และ 7

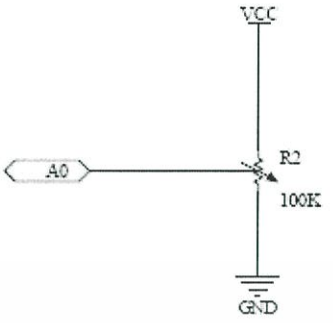
ตารางที่ 4 การทำงานของสเต็ปมอเตอร์แบบเต็มสเต็ปหนึ่งเฟส

สเต็ปที่	เฟสที่ 1	เฟสที่ 2	เฟสที่ 3	เฟสที่ 4
1	ทำงาน	-	-	-
2	-	ทำงาน	-	-
3	-	-	ทำงาน	-
4	-	-	-	ทำงาน

3.2.9 การออกแบบวงจร Analog to digital convertor (A-to-D)

จะใช้ port analog ที่ 0 โดยต่อกับตัวต้านทานปรับค่าได้ และสามารถปรับค่าความละเอียดขนาด 10 บิต หรือ 0-1023 ดังรูป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.20 วงจร A-to-D

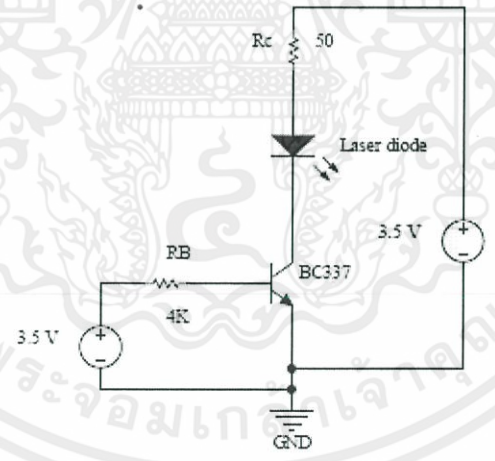
3.2.10 การออกแบบวงจรขับเลเซอร์

กำหนดให้ ต้องการ V_{CC} = 3.5 V. , I_C = 70 mA.

1. หาค่า R_C จาก Load Current (I_L) โดยที่ $R_C = \frac{V_{cc}}{I_c} = \frac{3.5-0.07-0.3}{70x10^{-3}}$ = 50 Ω. (เลือก R=50)

2. หาค่า I_B จาก ($I_C = \beta I_B$) จะได้ $I_B = \frac{I_C}{\beta} = \frac{70\ mA}{100}$ = 0.7 mA.

จะได้ $R_B = \frac{3.5-0.7}{0.7\ mA}$ = 4 KΩ. (เลือก 3.9K) ดังรูป



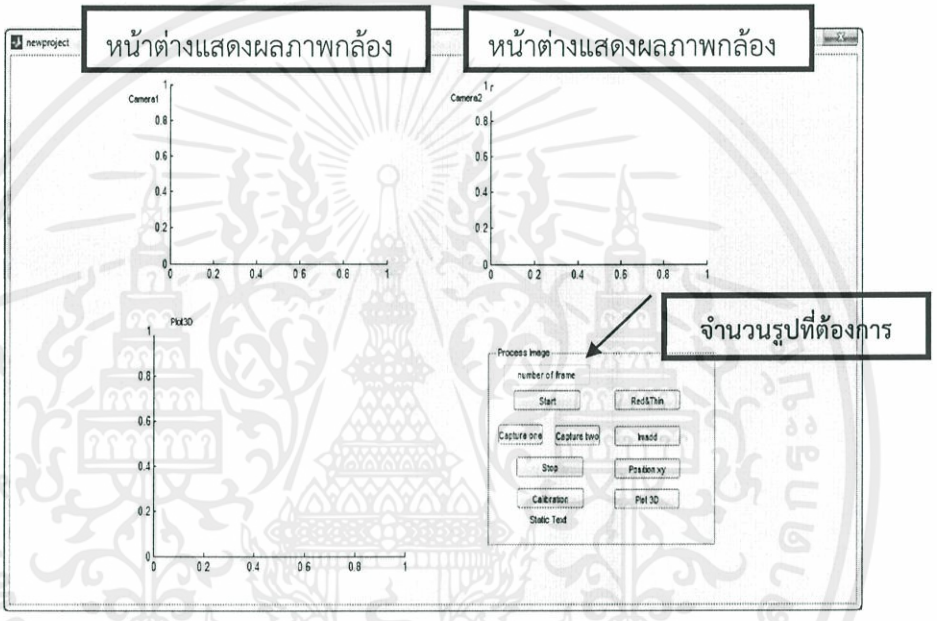
รูปที่ 3.21 วงจรขับเลเซอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 4
ผลการทดลอง

4.1 ผลการทดลองในส่วนของ Software

การทำงานในส่วน Software เป็นการเชื่อมต่อกับกล้องเข้ามาในระบบโดยใช้โปรแกรม MATLABการปรับปรุงภาพที่ได้จากการถ่ายวัตถุ รวมถึงการพล็อตสามมิติ โดยแบ่งผลการทดลองออกเป็นขั้นตอน



รูปที่ 4.1 หน้าต่างโปรแกรมที่ใช้สำหรับการถ่ายรูปต่อเนื่องและปรับปรุงภาพ

หน้าต่างของโปรแกรมประกอบด้วย

- ช่องใส่จำนวนรูป
- Start % เริ่มการทำงานของกล้อง
- Capture one % การถ่ายภาพในแนวแรก
- Capture two % การถ่ายภาพในแนวสอง
- Stop %หยุดการทำงานของกล้อง
- Calibration สำหรับการปรับเทียบกล้องโดย เราจะกดปุ่มนี้เพื่อทำการถ่ายรูปกล่อง และหาค่าพารามิเตอร์ของกล้องทั้งสองตัว โดยจะทำการปรับเทียบในครั้งแรก ในการเปิดโปรแกรม

Red&thin ดึงองค์ประกอบสีแดง และลดขนาดเส้น ภาพที่ทำการปรับปรุงเสร็จจะถูกบันทึกเก็บไว้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากรูปภาพหน้าต่างโปรแกรมยังมีปุ่มการทำงานปุ่มอื่นอีก แต่ในการทดลองครั้งนี้ได้ ส่วนของการทำงานหลังจากปรับปรุงภาพเสร็จแล้ว ได้ทำการรันโปรแกรมที่หน้าคำสั่ง command window แทน

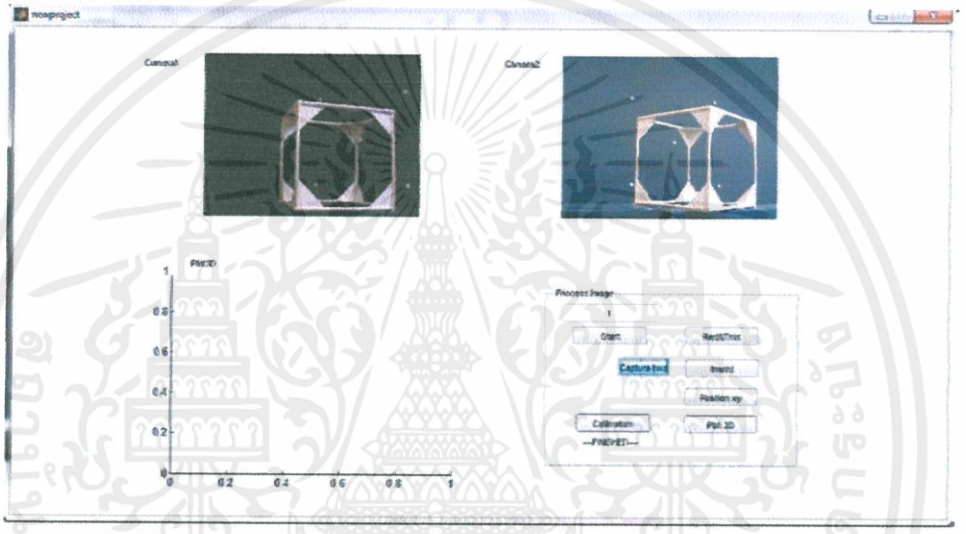
4.1.1 ขั้นตอนการทำงาน

4.1.1.1 การรับภาพเข้ามา

โดยการกำหนดจำนวนภาพที่เราต้องการทำการถ่ายลงในช่องว่าง

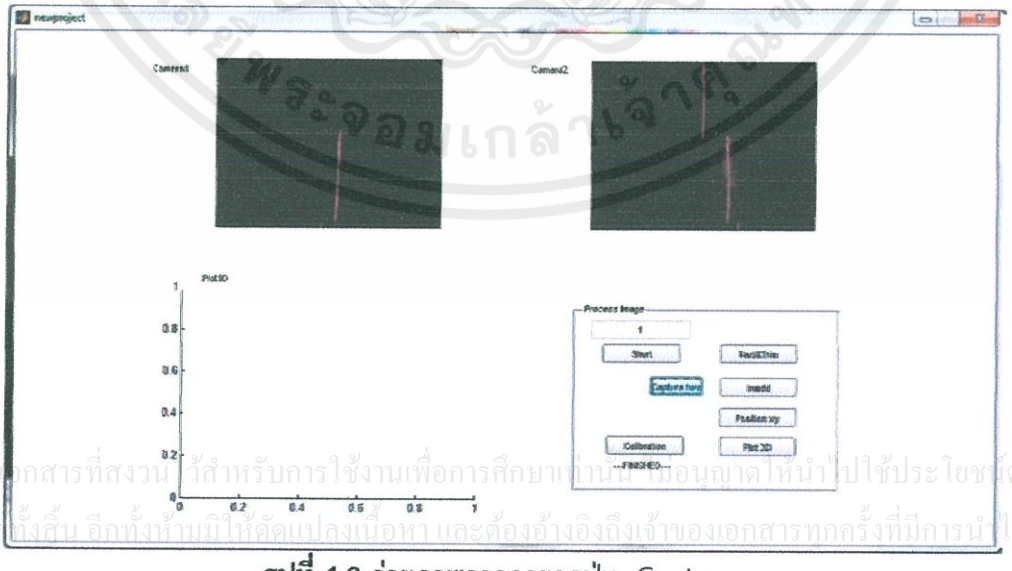
กดปุ่ม - start % เริ่มการทำงานของกล้อง

- capture one ถ่ายภาพครั้งแรก



รูปที่ 4.2 การถ่ายภาพวัตถุ

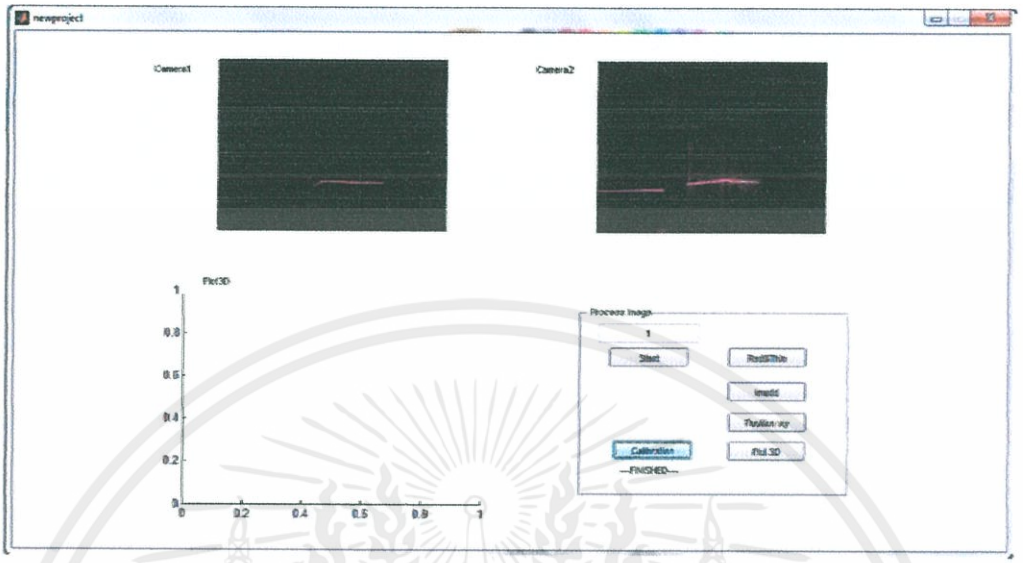
4.1.1.2. การถ่ายภาพวัตถุที่มีแสงเลเซอร์ในแนวแรก



รูปที่ 4.3 ถ่ายภาพจากการกดปุ่ม Capture one

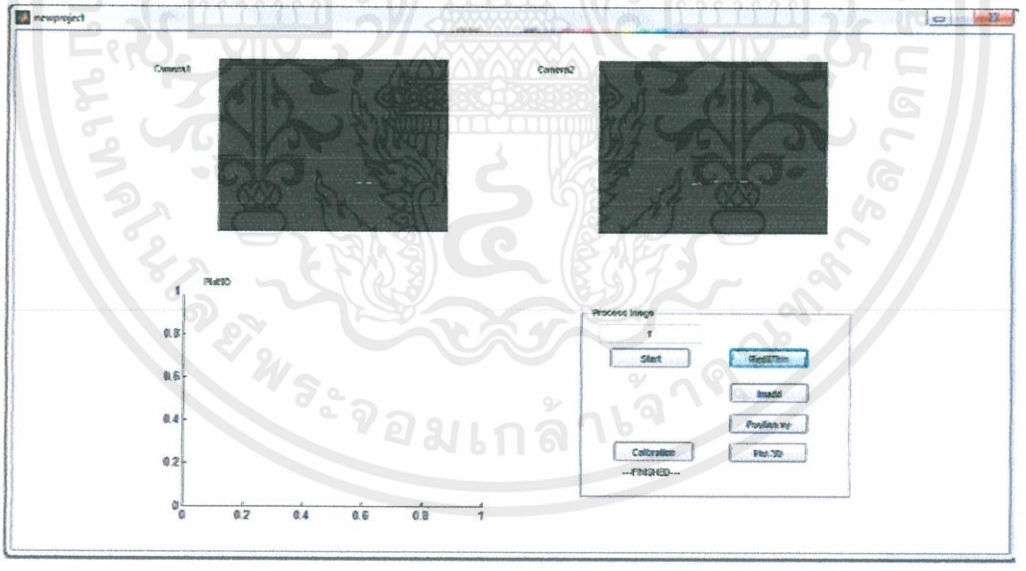
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.1.1.3. การถ่ายภาพวัตถุที่มีแสงเลเซอร์ในแนวที่สอง



รูปที่ 4.4 ถ่ายภาพจากการกดปุ่ม Capture two

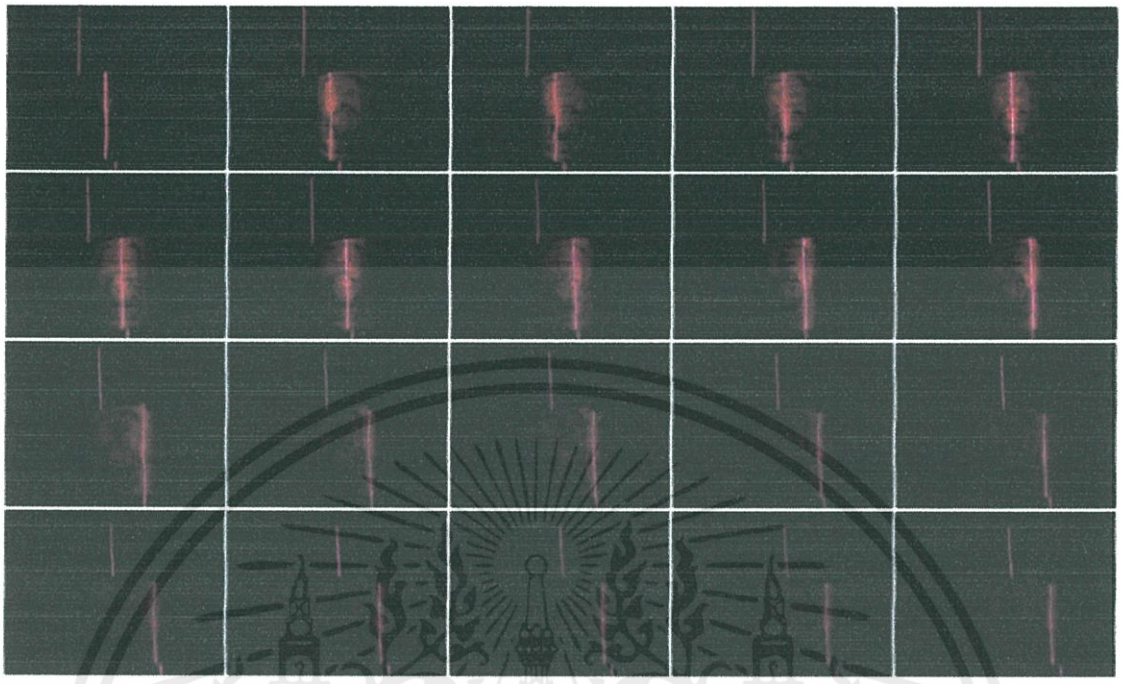
4.1.1.4 การดึงองค์ประกอบสีแดงและลดขนาดของเส้น



รูปที่ 4.5 การปรับปรุงภาพ

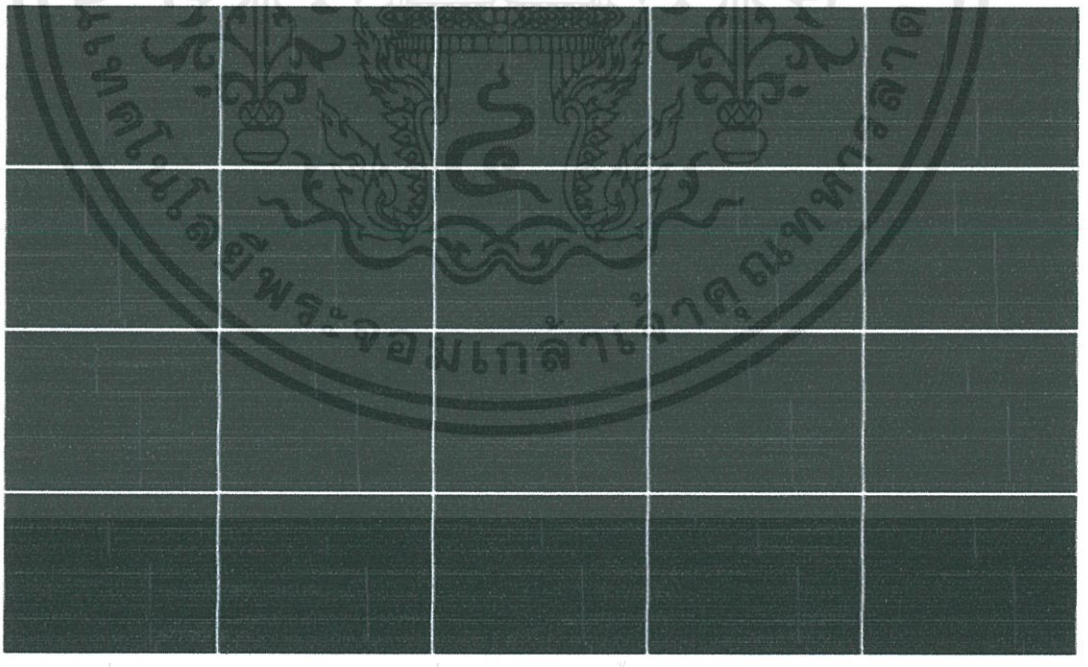
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตัวอย่างการรับภาพต่อเนื่อง กำหนดจำนวนการถ่าย 20 ภาพ



รูปที่ 4.6 การรับภาพต่อเนื่องจำนวน 20 ภาพ

ตัวอย่างภาพต่อเนื่องหลังจากการปรับปรุงภาพดึงองค์ประกอบสีแดงและเลือกระดับความเข้มของแสง



รูปที่ 4.7 ภาพต่อเนื่องหลังจากดึงองค์ประกอบสีแดงและเลือกระดับความเข้มของแสง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.1.2 ขั้นตอนหลังจากาการปรับปรุงภาพได้ใช้การรันโปรแกรมในหน้า command window

4.1.2.1 การอ่านภาพส่วนที่ทำกการปรับปรุงภาพ

4.1.2.2 ทำการ รวม ภาพที่เป็นลักษณะเส้นเลเซอร์ในแนวแรก และแนวที่สองเข้าด้วยกันด้วยการใช้คำสั่ง imadd

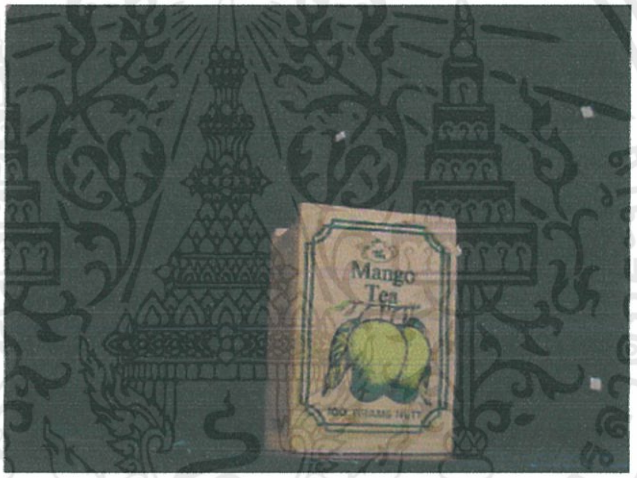
4.1.2.3 ทำการรวมภาพจากข้อที่ 2 กับภาพกล่องสี่เหลี่ยม ซึ่งกล่องสี่เหลี่ยมที่รวมเพิ่มเข้ามา เพื่อใช้สำหรับเป็นปรับเทียบกล้องทั้งสองตัว

4.1.2.4 คลิ๊กเมาส์ลงบนจุดทั้งแปดของกล่องที่ปรากฏในภาพทำการคลิ๊กเมาส์ลงบนจุดตัดที่ปรากฏขึ้น (จำนวนจุดที่คลิ๊กมีผลต่อความละเอียด ของภาพที่ใช้ทำการพล็อต)

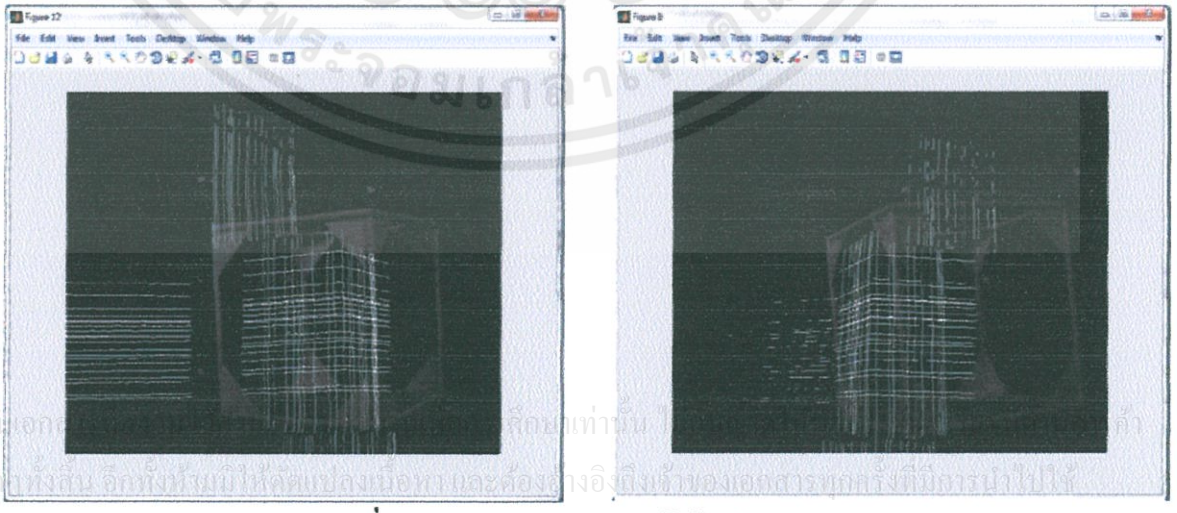
4.1.2.5 นำค่าพิกัด xy ที่ได้จากการคลิ๊ก พล็อต 3 มิติ

4.1.3. วัตถุที่ใช้ในการทดลอง

4.1.3.1 วัตถุชิ้นที่ 1 กล่องสี่เหลี่ยม

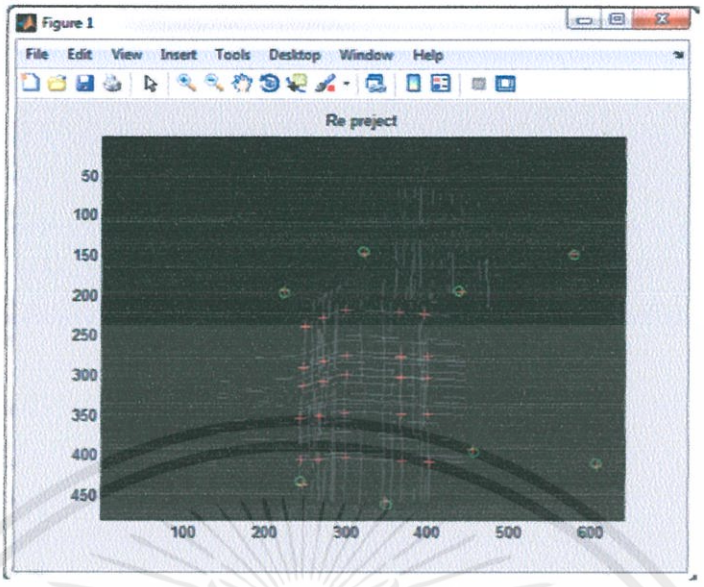


รูปที่ 4.8 วัตถุชิ้นที่ 1 กล่องสี่เหลี่ยม



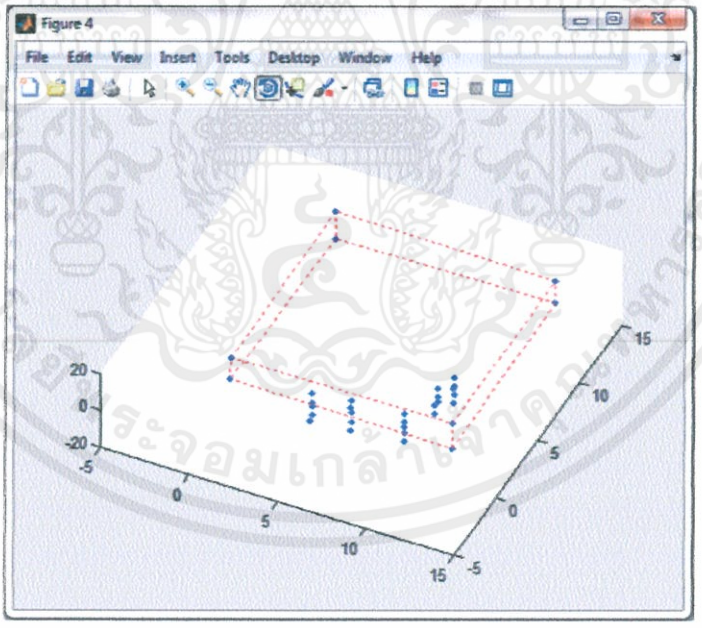
รูปที่ 4.9 การ imadd ทุกรูปเข้าด้วยกัน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.10 การคลิ๊กมุมของกล่องทั้งแปดและการคลิ๊กลงบนจุดตัดที่เกิดขึ้น

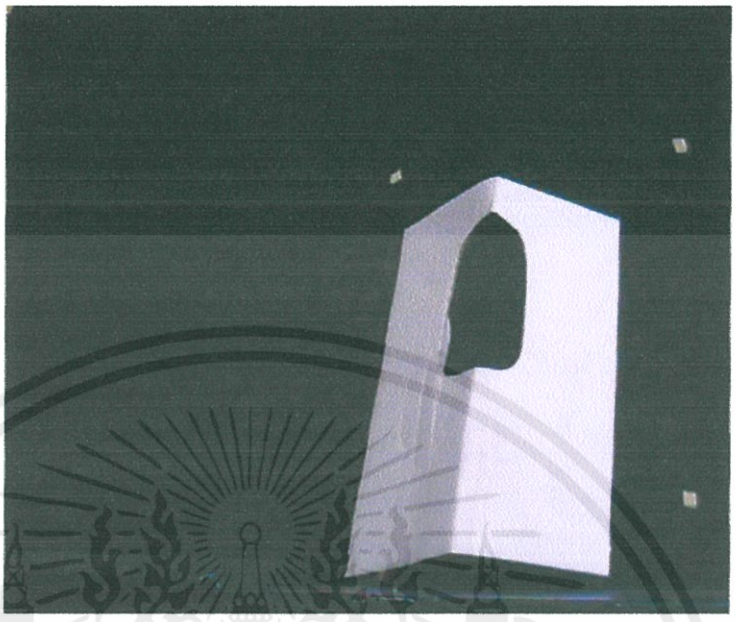
จากรูปการคลิ๊กลงบนจุดตัดที่เกินขึ้นจำนวนจุดที่ทำการคลิ๊กลงไปมีจำนวนน้อย ทำให้ค่าที่ได้การพล็อตมีค่าไม่ละเอียด เห็นเพียงแต่มุมที่เกิดขึ้นของกล่องเท่านั้น (จำนวนจุดที่คลิ๊ก25)



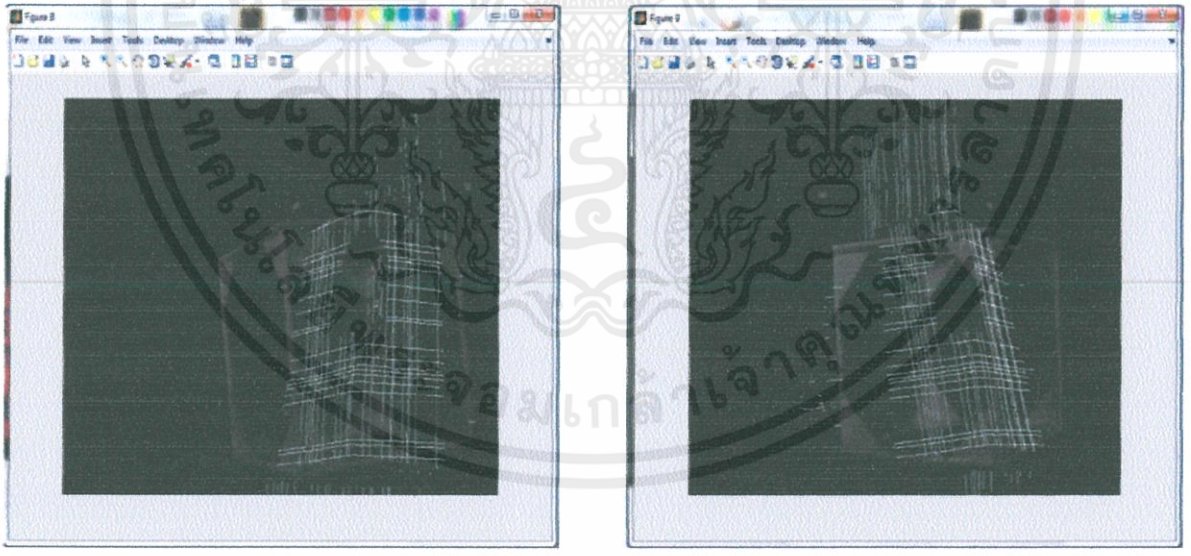
รูปที่ 4.11 การพล็อตสามมิติ วัตถุชิ้นที่ 1

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.1.3.2 วัตถุชิ้นที่ 2 กระดาษแข็งที่มีช่องว่าง

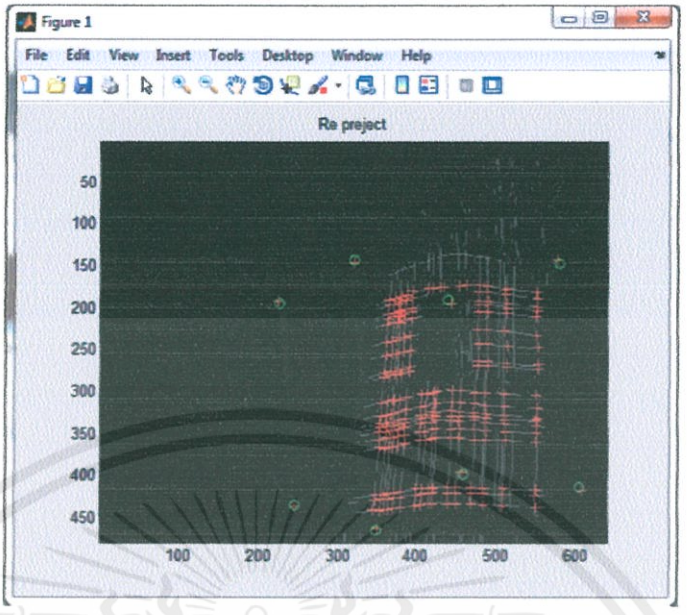


รูปที่ 4.12 วัตถุชิ้นที่ 2 กระดาษแข็งที่มีช่องว่าง

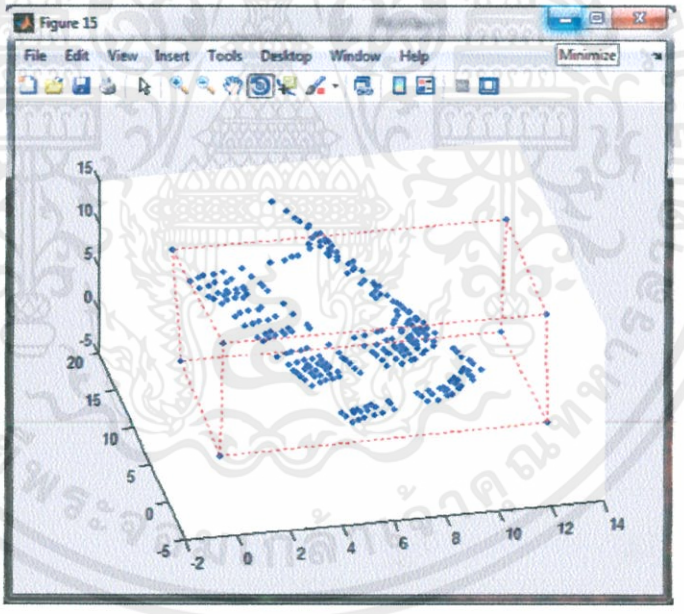


รูปที่ 4.13 การ imadd ทุกรูปเข้าด้วยกัน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



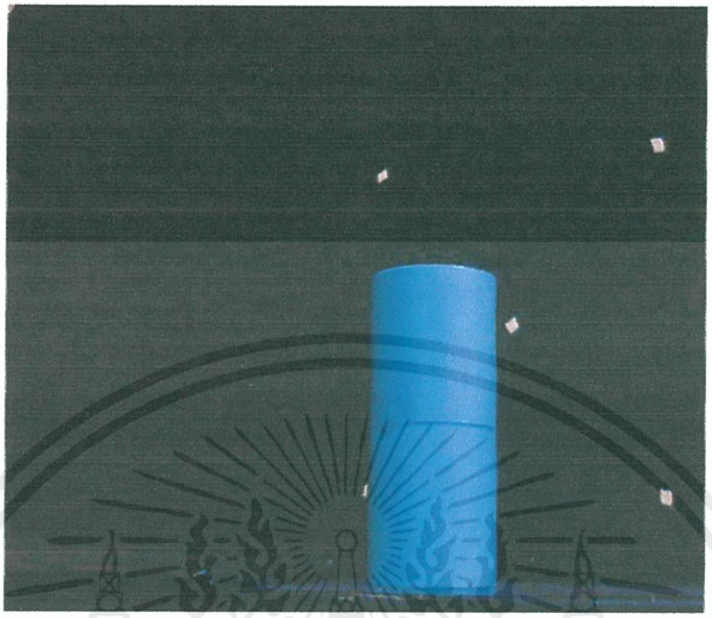
รูปที่ 4.14 การคลิ๊กมุมของกล่องทั้งแปดและการคลิ๊กลงบนจุดตัดที่เกิดขึ้น



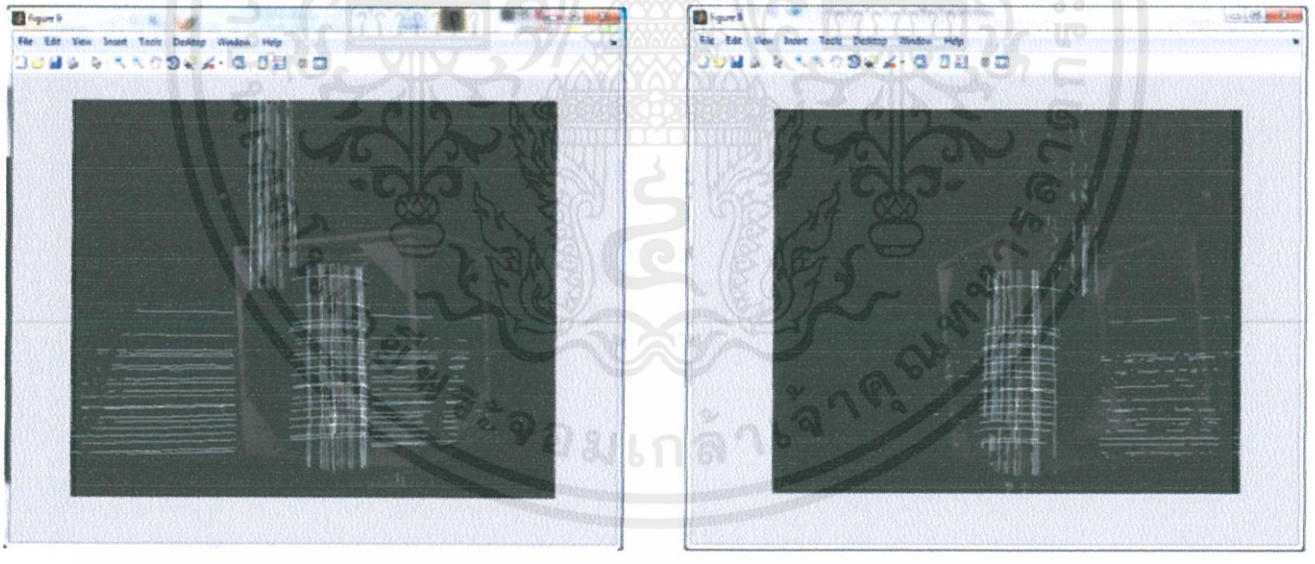
รูปที่ 4.15 การพล็อตสามมิติ วัตถุชิ้นที่ 2

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.1.3.3 วัตถุชิ้นที่ 3 ท่อ PVC ทรงกระบอก

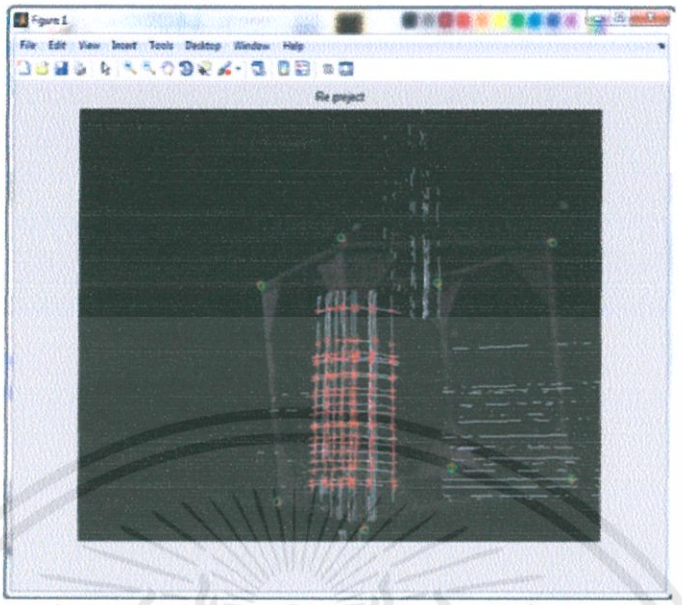


รูปที่ 4.16 วัตถุชิ้นที่ 3 ท่อน้ำทรงกระบอก

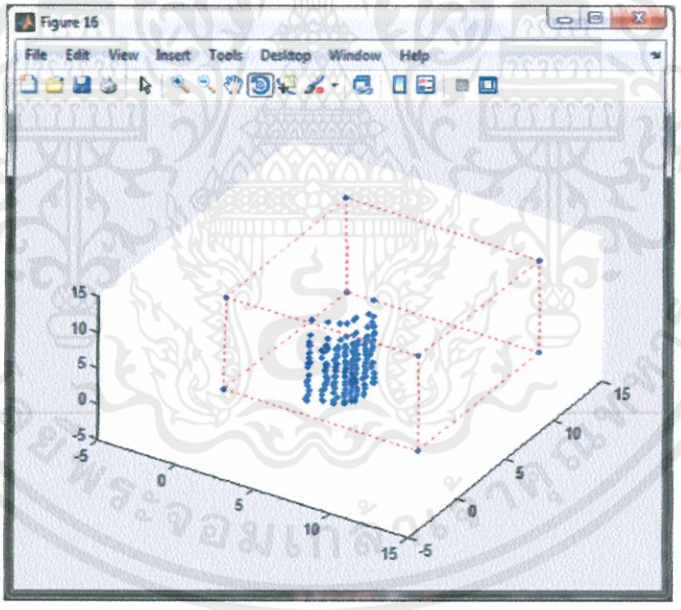


รูปที่ 4.17 การ imadd ทุกรูปเข้าด้วยกัน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



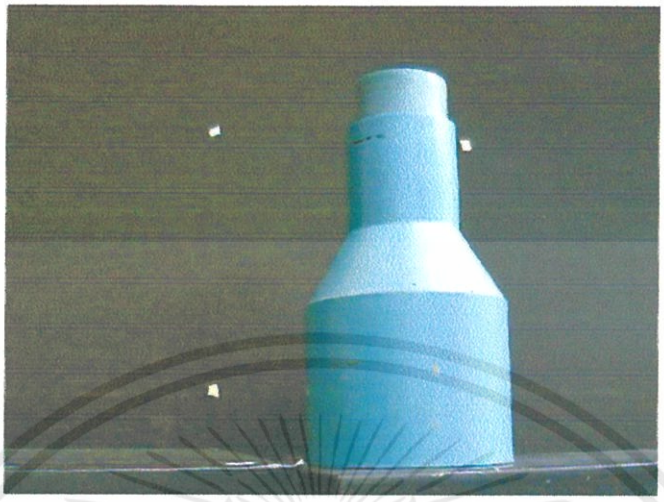
รูปที่ 4.18 การคลิ๊กมุมของกล่องทั้งแปดและการคลิ๊กลงบนจุดตัดที่เกิดขึ้น



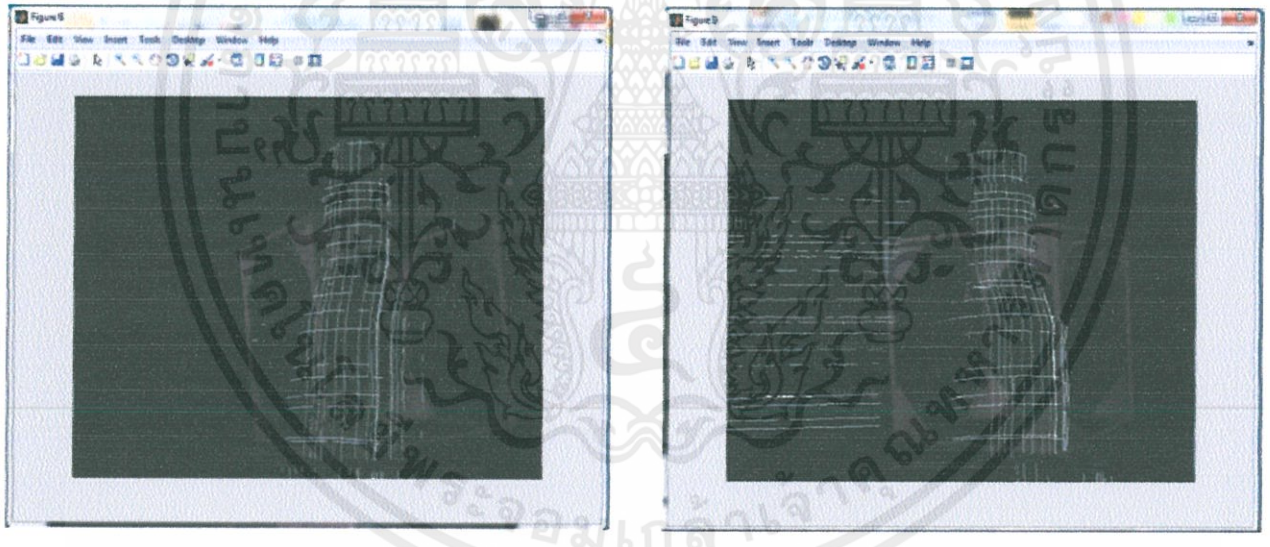
รูปที่ 4.19 การพล็อตสามมิติ วัตถุชิ้นที่ 3

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.1.3.4 วัตถุชิ้นที่ 4 ท่อ PVC ขนาดไม่เป็นทรงกระบอก

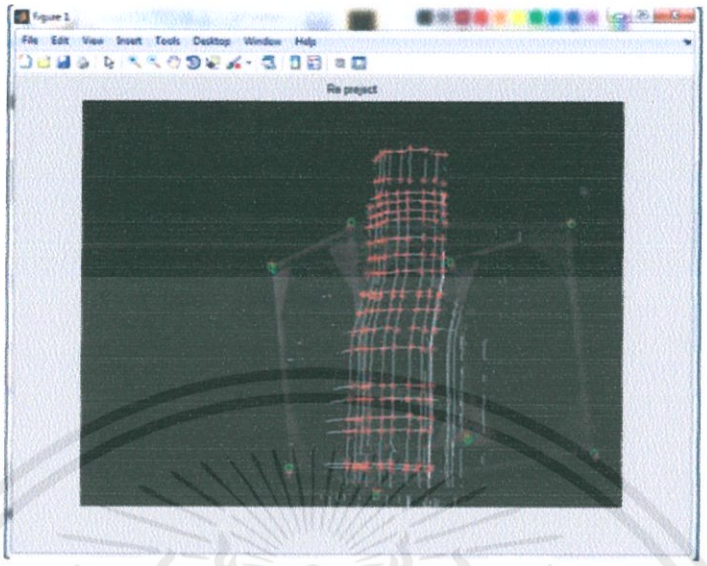


รูปที่ 4.20 วัตถุชิ้นที่ 4 ท่อน้ำขนาดไม่เป็นทรงกระบอก

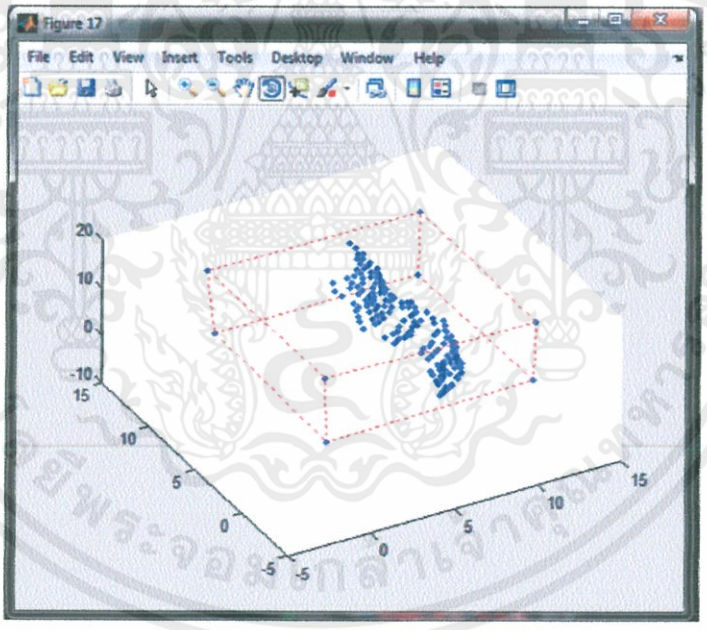


รูปที่ 4.21 การ imadd ทุกรูปเข้าด้วยกัน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.22 การคลิ๊กมุมของกล่องทั้งแปดและการคลิ๊กลงบนจุดตัดที่เกิดขึ้น



รูปที่ 4.23 การพล็อตสามมิติ วัตถุชิ้นที่ 4

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.2 ผลการทดลองในส่วนของ Hardware

เป็นส่วนที่ใช้ควบคุมการทำงานของ Motor , จอแสดงผล , และเลเซอร์ โดยการทำงานในส่วนมอเตอร์ควบคุมการทำงานโดย microcontroller โดยส่งลอจิกมาควบคุมการทำงานขดลวด 4 ขวด แบบ Wave Step Drive ข้างละเฟสเรียงกัน เมื่อต้องการกลับทิศก็แค่สลับเฟส

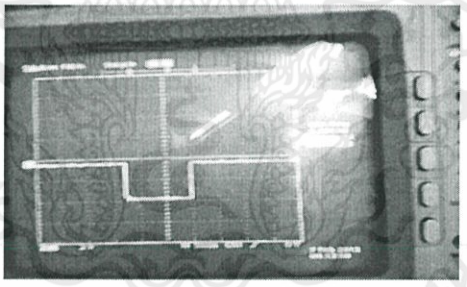
ตารางที่ 5 รูปแบบการขับมอเตอร์แบบ Wave Step Drive

สเต็ปที่	เฟสที่ 1	เฟสที่ 2	เฟสที่ 3	เฟสที่ 4
1	ทำงาน	-	-	-
2	-	ทำงาน	-	-
3	-	-	ทำงาน	-
4	-	-	-	ทำงาน

4.2.1 วงจรขับมอเตอร์

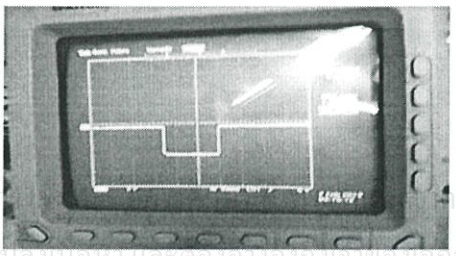
ในการป้อนกระแสให้กับมอเตอร์ จะใช้วงจรขับกระแสโดยใช้ Transistor เบอร์ BD139 จำนวน 4 ตัวในการขับกระแส วัดสัญญาณคร่อมตัวต้านทานขนาด 5.1/2W (R11,R12,R13,R14) ได้ดังนี้

<u>ขดที่ 1</u>



Voltage..... 3.36 ..... V. ที่ 2V/div. , 50ms

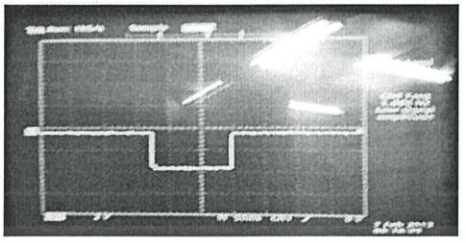
<u>ขดที่ 2</u>



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

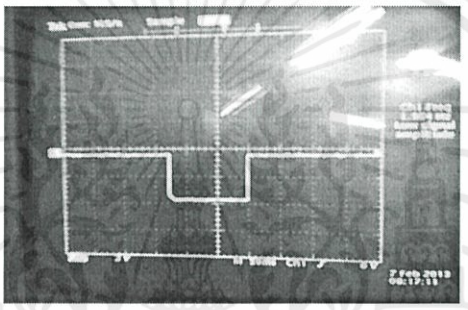
Voltage..... 3.44 ..... V. ที่ 2V/div. , 50ms

ขดที่ 3



Voltage =..... 3.20 ..... V. ที่ 2V/div. , 50ms

ขดที่ 4



Voltage =..... 3.28 ..... V. ที่ 2V/div. , 50ms

ฉะนั้นสามารถหาค่ากระแสที่จ่ายให้กับขดลวดแต่ละขดได้ดังนี้

ขดลวดขดที่	กระแส (A)
1	0.659
2	0.675
3	0.627
4	0.643

กระแสเฉลี่ยในแต่ละขด (0.659 + 0.675 + 0.627 + 0.643) / 4 = ... **0.651**..... A.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 5
สรุปผลการทดลอง

5.1 สรุปผลการทดลอง

การทดลองนำวัตถุที่มีรูปร่างแตกต่างกันมาเพื่อสแกนรูปร่างของวัตถุ

5.1.1 การทำงานในส่วนของ software

การทำงานในส่วนของ Software เป็นการทำงานในส่วน ติดต่อกล้องเข้าไปในระบบ โดยใช้โปรแกรม MATLAB และปรับปรุงภาพที่ได้จากการถ่ายวัตถุ สามารถถ่ายภาพต่อเนื่องตามจำนวนเฟรมที่กำหนด โดยภาพที่ได้จากการถ่ายสามารถดึงองค์ประกอบสีแดงและเลือกระดับความเข้มของแสง รวมถึงลดขนาดของเส้น และหาจุดตัดของแต่ละเส้นได้ ซึ่งเป็นตำแหน่งพิกัด สองมิติ ก่อนจะถูกแปลงเป็นสามมิติ

5.1.2 การทำงานในส่วนของ Hardware

มอเตอร์สามารถหมุนกลับไปกลับมาได้ สามารถควบคุมความเร็ว และแสดงผลออกทางจอ LCD Module ได้

5.2 ปัจจัยต่างๆในการทดลอง

5.2.1 แสงสว่างรบกวน

เนื่องจากแสงที่กล้องรับเข้ามาเป็นระบบสีแบบ RGB ซึ่งแสงสว่างมีผลกระทบต่อการดึงองค์ประกอบสีแดง เมื่อช่วงเวลาในการถ่ายรูปต่อเนื่องทำให้แสงที่เข้ามามีความเข้มแสงต่างกัน ผลของการดึงองค์ประกอบสีแดง และระดับความเข้มแสงนั้นจะมีการผิดเพี้ยนของแสงเกิดขึ้น

แนวทางการแก้ปัญหา ทำการถ่ายภาพในสถานที่มีแสงรบกวนน้อย อาจจะสร้างกล่องเพื่อคลุมส่วนทีทำการฉายแสงเลเซอร์ หรือทำการปิดไฟในการทดลองแต่ละครั้ง

5.2.2 ชนิดของวัตถุ

วัตถุที่นำมาทำการสแกนจะต้องเป็นวัตถุที่ทึบแสง มีการสะท้อนแสงที่น้อย ต้องไม่เป็นวัตถุใส เพราะแสงเลเซอร์ที่กระทบลงบนพื้นผิวของวัตถุจะมีค่าตำแหน่งที่ผิดเพี้ยน โดยกล้องจะรับภาพของเลเซอร์ที่ถูกสะท้อนหรือทะลุไปข้างหลังของวัตถุ แทนที่จะรับตำแหน่งเลเซอร์บนพื้นผิวของวัตถุ

5.2.3 ความเร็วในการเคลื่อนที่ของแท่นวางวงจรขับเลเซอร์

ความเร็วของแท่นวางไม่คงที่ โดยเฉพาะการเคลื่อนที่ขึ้นลงในแนวดิ่ง

5.3 ปัญหาในการทดลอง

5.3.1 การหาพิกัดจุดตัดสองมิติ

เราสามารถคาดเดาตำแหน่งจุดที่ไม่มีการตัดกันจริงในภาพได้ เพื่อเพิ่มความละเอียดของภาพ แต่การทำแบบนี้เป็นการเสียงเวลาเป็นอย่างมาก ถ้าจุดที่คลิ๊กมีความไม่สอดคล้องกัน จะทำให้รูปทรงของวัตถุผิดเพี้ยนไปหมด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5.3.2 ความเข้มของแสงเลเซอร์

ความเข้มของแสงเลเซอร์ที่มีความเข้มลดลงเมื่อใช้งานติดต่อกันเป็นเวลานาน ค่าระดับความเข้มที่ถูกกำหนดไว้ตอนเริ่มต้น ส่งผลให้เส้นเลเซอร์ที่ได้จากการปรับปรุงภาพ มีความชัดเจนลดลง

5.4 แนวทางแก้ปัญหา

5.4.1 เลือกชนิดของวัตถุ

เลือกชนิดของวัตถุให้มีลักษณะทึบแสง หรือปรับแต่งให้วัตถุมีลักษณะทึบแสง

5.4.2 เพิ่มวงจรขับเลเซอร์

เพื่อให้แสงเลเซอร์มีความเข้มคงที

5.4.3 ส่วน Hardware

การเคลื่อนที่บน linear scale สามารถเพิ่มแรงบิดการหมุนของมอเตอร์ได้ โดยเปลี่ยนรูปแบบการหมุนเป็น Half step drive (One-Two Excitation Drive)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บรรณานุกรม

[1] ชูชาติ ปิณฑวิรุจน์. 2555. **การประมวลผลภาพดิจิตอลด้วย MATLAB.** พิมพ์ครั้งที่ 2. กรุงเทพมหานคร : มีน เซอร์วิส ซัพพลาย.

[2] ชูชาติ ปิณฑวิรุจน์. 2551. **การประมวลผลภาพดิจิตอลขั้นสูงด้วย MATLAB.** กรุงเทพมหานคร : วีเจ.

[3] มนัส สังวรศิลป์ และ วรรัตน์ ภัทรอมรกุ. 2543. **คู่มือการใช้งาน MATLAB ฉบับสมบูรณ์** พิมพ์ครั้งที่ 2. กรุงเทพมหานคร : อินโฟเพรส

[4] เอกชัย มะการ. 2552. **เรียนรู้เข้าใจใช้งานไมโครคอนโทรลเลอร์ตระกูล AVR ด้วย Arduino** กรุงเทพมหานคร : บริษัท อีทีที จำกัด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก ก

Code MATLAB

การถ่ายภาพต่อเนื่อง, การปรับปรุงภาพโดยการดึงองค์ประกอบสีแดง, เลือกระดับความเข้ม และ ลดขนาดของเส้น

```
function varargout = newproject(varargin)
% NEWPROJECT M-file for newproject.fig
%      NEWPROJECT, by itself, creates a new NEWPROJECT or raises the existing
%      singleton*.
%
%      H = NEWPROJECT returns the handle to a new NEWPROJECT or the handle to
%      the existing singleton*.
%
%      NEWPROJECT('CALLBACK',hObject,eventData,handles,...) calls the local
%      function named CALLBACK in NEWPROJECT.M with the given input arguments.
%
%      NEWPROJECT('Property','Value',...) creates a new NEWPROJECT or raises the
%      existing singleton*.  Starting from the left, property value pairs are
%      applied to the GUI before newproject_OpeningFcn gets called.  An
%      unrecognized property name or invalid value makes property application
%      stop.  All inputs are passed to newproject_OpeningFcn via varargin.
%
%      *See GUI Options on GUIDE's Tools menu.  Choose "GUI allows only one
%      instance to run (singleton)".
%
% See also: GUIDE, GUIDATA, GUIHANDLES

% Edit the above text to modify the response to help newproject

% Last Modified by GUIDE v2.5 21-Jan-2013 21:18:21

% Begin initialization code - DO NOT EDIT
gui_Singleton = 1;
gui_State = struct('gui_Name',       mfilename, ...
                   'gui_Singleton',  gui_Singleton, ...
                   'gui_OpeningFcn', @newproject_OpeningFcn, ...
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                'gui_OutputFcn',  @newproject_OutputFcn, ...
                'gui_LayoutFcn',  [] , ...
                'gui_Callback',   []);
if nargin && ischar(varargin{1})
    gui_State.gui_Callback = str2func(varargin{1});
end

if nargout
    [varargout{1:nargout}] = gui_mainfcn(gui_State, varargin{:});
else
    gui_mainfcn(gui_State, varargin{:});
end
% End initialization code - DO NOT EDIT
%%

%% --- Executes just before newproject is made visible.
function newproject_OpeningFcn(hObject, eventdata, handles, varargin)
% This function has no output args, see OutputFcn.
% hObject    handle to figure
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)
% varargin   command line arguments to newproject (see VARARGIN)

% Choose default command line output for newproject
handles.output = hObject;

% Update handles structure
guidata(hObject, handles);
%%

%% --- Outputs from this function are returned to the command line.
function varargout = newproject_OutputFcn(hObject, eventdata, handles)
% varargout  cell array for returning output args (see VARARGOUT);
% hObject    handle to figure
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)

% Get default command line output from handles structure
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
varargout{1} = handles.output;
%%


%% --- Executes on button press in pushbutton1.
function pushbutton1_Callback(hObject, eventdata, handles)
  set(handles.text1,'visible','off');

                                    handles.nframe=
str2double(get(handles.edit1,'string'));
                                    guidata(hObject,handles);

handles.vidleft=videoinput('winvideo','1','RGB24_640x480');
                                    guidata(hObject,handles);

handles.vidright=videoinput('winvideo','2','RGB24_640x480');
                                    guidata(hObject,handles);
                                        set(handles.vidleft,'TriggerRepeat',inf);
                                                handles.vid.FrameGrabInterval=1;

vid_src=getselectedsource(handles.vidleft);
                                                set(vid_src,'Tag','motion detection
setup');
                                        set(handles.vidright,'TriggerRepeat',inf);

handles.vidright.FrameGrabInterval=1;

vid_src=getselectedsource(handles.vidright);
                                                set(vid_src,'Tag','motion detection
setup');
                                                start(handles.vidleft);
                                                start(handles.vidright);
                                    preview(handles.vidleft);
                                    preview(handles.vidright);
                                    stop(handles.vidleft);
                                    stop(handles.vidright);
                                    set(handles.pushbutton2,'visible','on');
                                    set(handles.pushbutton3,'visible','on');
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของผลงานทุกครั้งที่มีการนำไปใช้

```
                                    set(handles.pushbutton4,'visible','on');
                                    set(hObject  ,'visible','off');
%%

%% --- Executes on button press in pushbutton2.
function pushbutton2_Callback(hObject, eventdata, handles)
% handles.nframe= str2double(get(handles.edit1,'string'));
start(handles.vidleft);
start(handles.vidright);
set(handles.text1,'visible','on');
%               pause(3)
              set(handles.pushbutton4,'visible','on');
              n2= handles.nframe;
              n=[1:n2]; %-------------number of writing picure to hardisk
              handles.number = size(n,2);
              guidata(hObject,handles);
                    for k=1:size(n,2)

%                       pause(5) % delay time
                      oneleft = getsnapshot(handles.vidleft);
                      oneright = getsnapshot(handles.vidright);
                      guidata(hObject,handles);
                        %imagename = strcat(int2str(n),'.jpeg');
                           imwrite(oneleft,strcat('oneleft',int2str (k),'.bmp'),'bmp'); %-
--------file anme is n.bmp and it's format is "bmp"
                           imwrite(oneright,strcat('oneright',int2str (k),'.bmp'),'bmp');
%---------file anme is n.bmp and it's format is "bmp"
%                          pic1 = imread('oneleft1.bmp');
%                          pic2 = imread('oneright1.bmp');
%                          imwrite(pic1,strcat('calbox1','.bmp'),'bmp');
%                          imwrite(pic2,strcat('calbox2','.bmp'),'bmp');

                          axes(handles.axes1),imshow(oneleft);
                          axes(handles.axes2),imshow(oneright);
                             flushdata(handles.vidleft);
                             flushdata(handles.vidright);
                             handles.currentnum=k;
                                guidata(hObject,handles);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                    textbox=strcat('DONE =  ',num2str(k));
                    set(handles.text1,'string', textbox);
                end
            set(handles.pushbutton1,'visible','on');
            set(handles.pushbutton3,'visible','on');
            set(handles.pushbutton4,'visible','off');
            set(handles.text1,'string', '---FINISHED---');
            set(hObject,'visible','off');
               stop(handles.vidleft);
               stop(handles.vidright);
%                   delete(handles.vidleft);%----------------------back to beginning clear
ram
%                   delete(handles.vidright);

 %%

%% --- Executes on button press in pushbutton3.
function pushbutton3_Callback(hObject, eventdata, handles)
% handles.nframe= str2double(get(handles.edit1,'string'));
start(handles.vidleft);
start(handles.vidright);
set(handles.text1,'visible','on');
%          pause(3)
            set(handles.pushbutton4,'visible','on');
            n2= handles.nframe;
            n=[1:n2]; %-------------number of writing picure to hardisk
            handles.number = size(n,2);
            guidata(hObject,handles);
                    for k=1:size(n,2)
%                     pause(5) % delay time
                    twoleft = getsnapshot(handles.vidleft);
                    tworight = getsnapshot(handles.vidright);
                    guidata(hObject,handles);
                       %imagename = strcat(int2str(n),'.jpeg');
                          imwrite(twoleft,strcat('twoleft',int2str (k),'.bmp'),'bmp'); %-
--------file anme is n.bmp and it's format is "bmp"
                          imwrite(tworight,strcat('tworight',int2str (k),'.bmp'),'bmp');
%---------file anme is n.bmp and it's format is "bmp"
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของผลงานทุกครั้งที่มีการนำไปใช้

```
                axes(handles.axes1),imshow(twoleft);
                axes(handles.axes2),imshow(tworight);
                    flushdata(handles.vidleft);
                    flushdata(handles.vidright);
                    handles.currentnum=k;
                        guidata(hObject,handles);
                textbox=strcat('DONE =  ',num2str(k));
                set(handles.text1,'string', textbox);
                end
            set(handles.pushbutton1,'visible','on');
            set(handles.pushbutton3,'visible','on');
            set(handles.pushbutton4,'visible','off');
            set(handles.text1,'string', '---FINISHED---');
            set(hObject,'visible','off');
                stop(handles.vidleft);
                stop(handles.vidright);
                    delete(handles.vidleft);%----------------------back to beginning
                    delete(handles.vidright);
%                   close(handles.vidleft);
%                   close(handles.vidright);


%%

%% --- Executes on button press in pushbutton4.
function pushbutton4_Callback(hObject, eventdata, handles)
stop(handles.vidleft);
stop(handles.vidright);

            set(handles.pushbutton1,'visible','on');
            set(handles.pushbutton2,'visible','on');
            set(handles.pushbutton3,'visible','on');
            set(hObject,'visible','off');
            delete(handles.vidleft);
        delete(handles.vidright);
        clear all;
%%
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
%% --- Executes on button press in pushbutton5.
function pushbutton5_Callback(hObject, eventdata, handles)

% start(handles.vidleft);
% start(handles.vidright);
a=getsnapshot(handles.vidleft);
    imwrite(a,strcat('calbox1','.bmp'),'bmp');
b=getsnapshot(handles.vidright);
    imwrite(b,strcat('calbox2','.bmp'),'bmp');
    guidata(hObject,handles);
    stop(handles.vidleft);
    stop(handles.vidright);
     set(handles.pushbutton1,'visible','on');
        delete(handles.vidleft);%----------------------back to beginning
        delete(handles.vidright);
for j=1:2
    switch(j)
        case 1
        a=imread('calbox1.bmp','bmp');
        aa=a(:,:,1);
        figure(1);imagesc(aa);colormap('gray');

        case 2
        a=imread('calbox2.bmp','bmp');
        aa=a(:,:,1);
        figure(1);imagesc(aa);colormap('gray');
    end

hold on;
    for i=1:8
    [u(i,1),u(i,2)]=ginput(1);
    plot(u(i,1),u(i,2),'r+');
    end
%¾Õ¡Ñ´Õèá·è¨ÃÔ§¢Í§¡Åèí§
 xx=[0      0       0 1
    10.5    0       0 1
    10.5    10.5    0 1
    0       10.5    0 1
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    0       0       11 1
    10.5    0       11 1
    10.5    10.5    11 1
    0       10.5    11 1];
%¤Ó¹Ç³ M
for i=1:8
G(2*i,1)=0;
G(2*i,2)=0;
G(2*i,3)=0;
G(2*i,4)=0;
G(2*i,5)=xx(i,1);
G(2*i,6)=xx(i,2);
G(2*i,7)=xx(i,3);
G(2*i,8)=1;
G(2*i,9)=-u(i,2)*xx(i,1);
G(2*i,10)=-u(i,2)*xx(i,2);
G(2*i,11)=-u(i,2)*xx(i,3);
G(2*i,12)=-u(i,2);

G((2*i)-1,1)=xx(i,1);
G((2*i)-1,2)=xx(i,2);
G((2*i)-1,3)=xx(i,3);
G((2*i)-1,4)=1;
G((2*i)-1,5)=0;
G((2*i)-1,6)=0;
G((2*i)-1,7)=0;
G((2*i)-1,8)=0;
G((2*i)-1,9)=-u(i,1)*xx(i,1);
G((2*i)-1,10)=-u(i,1)*xx(i,2);
G((2*i)-1,11)=-u(i,1)*xx(i,3);
G((2*i)-1,12)=-u(i,1);
end
[U,D,V]=svd(G);
VT=V;
M=VT(:,end); %last columm
%M=tranpose(reshape(M_vector,4,3));
m(1,1)=M(1);
m(1,2)=M(2);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
m(1,3)=M(3);
m(1,4)=M(4);
m(2,1)=M(5);
m(2,2)=M(6);
m(2,3)=M(7);
m(2,4)=M(8);
m(3,1)=M(9);
m(3,2)=M(10);
m(3,3)=M(11);
m(3,4)=M(12);
%°Ñ¹·Öį M äÇé
switch (j)
    case 1
        u1=u;
        m1=m;
        para1 = m1;
        handles.cal01=para1;
        guidata(hObject,handles)
    case 2
        u2=u;
        m2=m;
        para2 = m2;
        handles.cal02=para2;
        guidata(hObject,handles)
end


end%for j=1:3

size(u);
close
%%

%% --- Executes on button press in pushbutton6.
function pushbutton6_Callback(hObject, eventdata, handles)
inpic=handles.currentnum;
for n=1:inpic
    oneleft = imread(strcat('oneleft',int2str(n),'.bmp'));
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    oneredleft  = oneleft(:,:,1);
    oneredleft  = 255*im2double(oneredleft);
          for i=1:size(oneredleft,1)
             for j=1:size(oneredleft,2)
                if oneredleft(i,j)>=250 %%& imred(i,j)<=245
                   oneredleft(i,j)= 255;
                else
                   oneredleft(i,j)=0;
                end
             end
          end

      oneright = imread(strcat('oneright',int2str(n),'.bmp'));
      oneredright =  oneright(:,:,1);
      oneredright = 255*im2double(oneredright);
       for i=1:size(oneredright,1)
          for j=1:size(oneredright,2)
             if oneredright(i,j)>=250 %%& imred(i,j)<=245
                oneredright(i,j)= 255;
             else
                oneredright(i,j)=0;
             end
          end
       end

    imwrite(oneredleft,strcat('oneredleft',int2str(n),'.bmp'),'bmp'); %file anme is n.jpeg
and it's format is "JPG"
    imwrite(oneredright,strcat('oneredright',int2str(n),'.bmp'),'bmp'); %file anme is n.jpeg
and it's format is "JPG"

end


% inpic= handles.currentnum;
for n=1:inpic
      oneimredleft   = imread(strcat('oneredleft',int2str(n),'.bmp'));
      oneimnewleft   = elevatline(oneimredleft);
      oneimredright  = imread(strcat('oneredright',int2str(n),'.bmp'));
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งไม่อนุญาตให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        oneimnewright  = elevatline(oneimredright);

    imwrite(oneimnewleft,strcat('oneimnewleft',int2str(n),'.bmp'),'bmp'); %file anme is
n.jpeg and it's format is "JPG"
    imwrite(oneimnewright,strcat('oneimnewright',int2str(n),'.bmp'),'bmp'); %file anme is
n.jpeg and it's format is "JPG"

end


    %inpic=handles.num
inpic=handles.currentnum;
for n=1:inpic
    twoleft = imread(strcat('twoleft',int2str(n),'.bmp'));
    tworedleft  = twoleft(:,:,1);
    tworedleft  = 255*im2double(tworedleft);
          for i=1:size(tworedleft,1)
             for j=1:size(tworedleft,2)
                if tworedleft(i,j)>=240 %%& imred(i,j)<=245
                   tworedleft(i,j)= 255;
                else
                   tworedleft(i,j)=0;
                end
             end
          end

     tworight = imread(strcat('tworight',int2str(n),'.bmp'));
     tworedright =  tworight(:,:,1);
     tworedright = 255*im2double(tworedright);
       for i=1:size(tworedright,1)
          for j=1:size(tworedright,2)
             if tworedright(i,j)>=240 %%& imred(i,j)<=245
                tworedright(i,j)= 255;
             else
                tworedright(i,j)=0;
             end
          end
       end
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
%     imred = elevatline(imred);
%grayred=mat2gray(im1);


    imwrite(tworedleft,strcat('tworedleft',int2str(n),'.bmp'),'bmp'); %file anme is n.jpeg
and it's format is "JPG"
    imwrite(tworedright,strcat('tworedright',int2str(n),'.bmp'),'bmp'); %file anme is n.jpeg
and it's format is "JPG"

end


% two = zeros(inpic*inpic,2);
for n=1:inpic
        twoimredleft   = imread(strcat('tworedleft',int2str(n),'.bmp'));
        twoimnewleft   = elevatline(twoimredleft);
        twoimredright  = imread(strcat('tworedright',int2str(n),'.bmp'));
        twoimnewright  = elevatline(twoimredright);


    imwrite(twoimnewleft,strcat('twoimnewleft',int2str(n),'.bmp'),'bmp'); %file anme is
n.jpeg and it's format is "JPG"
    imwrite(twoimnewright,strcat('twoimnewright',int2str(n),'.bmp'),'bmp'); %file anme is
n.jpeg and it's format is "JPG"

    axes(handles.axes1),imshow(strcat('twoimnewleft',int2str(n),'.bmp'));
    axes(handles.axes2),imshow(strcat('twoimnewright',int2str(n),'.bmp'));
end
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก ข

Code Arduino

(ควบคุมการทำงานของมอเตอร์ , จอแสดงผล , detect switch และ A-to-D)

```
#define StepRate 1000
#define d 60
#include <LCD_4BIT.h>
 #define LCD_RS 10              //3
 #define LCD_EN 11              //5
 #define LCD_D4 12              //6
 #define LCD_D5 13              //7
 #define LCD_D6 8
 #define LCD_D7 9
LCD_4BIT lcd=LCD_4BIT(LCD_RS,LCD_EN,LCD_D4,LCD_D5,LCD_D6,LCD_D7);
int VR =0;
int INA = 4;                            //Phase-A Signal Drive
int INB = 5;                            //Phase-B Signal Drive
int INC = 6;                            //Phase-C Signal Drive
int IND = 7;                            //Phase-D Signal Drive
int REFSTATE1 = HIGH;
int REFSTATE2 = HIGH;
int SW1 = 0;
int SW2 = 2;

int READSW1;
int READSW2;
int READSW11;
int READSW22;
int READSW33 ;
int READSW44 ;

static byte Step = 0;                   //Position Step Buffer
static byte *pStep;
void WaveStepDrive(byte *Step)
  {
        byte MyStep = *Step;            //Get Step Position
    switch (MyStep)
     {
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                case 0:
                  digitalWrite(INA, HIGH);
                  digitalWrite(INB, LOW);
                  digitalWrite(INC, LOW);
                  digitalWrite(IND, LOW);
                  break;

                case 1:
                  digitalWrite(INA, LOW);
                  digitalWrite(INB, HIGH);
                  digitalWrite(INC, LOW);
                  digitalWrite(IND, LOW);
                  break;

                case 2:
                  digitalWrite(INA, LOW);
                  digitalWrite(INB, LOW);
                  digitalWrite(INC, HIGH);
                  digitalWrite(IND, LOW);
                  break;

                case 3:
                  digitalWrite(INA, LOW);
                  digitalWrite(INB, LOW);
                  digitalWrite(INC, LOW);
                  digitalWrite(IND, HIGH);
                  break;
                }
        }
//-------------------------------------//-------------------------------------//-------------------
-----------------/


  void setup()
        {       lcd.Initial();
                pinMode(INA, OUTPUT);
                pinMode(INB, OUTPUT);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                pinMode(INC, OUTPUT);
                pinMode(IND, OUTPUT);

                        digitalWrite(INA, LOW);
                        digitalWrite(INB, LOW);
                        digitalWrite(INC, LOW);
                        digitalWrite(IND, LOW);

                pinMode(SW1, INPUT);
                pinMode(SW2, INPUT);

        }
//-
================================================================
=======================================

        void loop()
        { /*(AAA)*/

int val= analogRead(VR);
val= map(val,0,1023,0,StepRate);
lcd.ClearScreen();
lcd.Print(val);
delay(10);
                        digitalWrite(INA, LOW);
                        digitalWrite(INB, LOW);
                        digitalWrite(INC, LOW);
                        digitalWrite(IND, LOW);
                                READSW1 = digitalRead(SW1) ;
                                READSW2 = digitalRead(SW2) ;
                if(REFSTATE1==HIGH && READSW2==LOW)
            {

                                delay(d);
                        READSW22 = digitalRead(SW2);
                        while (READSW22 == LOW )
                {/*-----------------------------------------------(4)*/
                        pStep = &Step; //1-Stop Reverse
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                *pStep -= 1; //3,2,1,0,3,2,1,0,...,3,2,1,0
                *pStep &= 0x03; //Step=0..3
                WaveStepDrive(&Step);

                if (*pStep ==4)
            {  *pStep = 0;}
                READSW1 = digitalRead(SW1);
                READSW2 = digitalRead(SW2);
                if (READSW1 == LOW){ break;}
                if (READSW1 == LOW && READSW2 == LOW){break;}
                 delay(val);   }/*--------------(2)*/
        }/*-------------------------------------------------------------------------(1)*/
              else if(REFSTATE2==HIGH && READSW1==LOW)
        {/*-----------------(3)*/
              delay(d);
              READSW11 = digitalRead(SW1);
                  while (READSW11 == LOW)
            {/*--------------(2)*/
              READSW2 = digitalRead(SW2) ;
              pStep = &Step;
              *pStep &= 0x03;
              WaveStepDrive(&Step);
              *pStep += 1;
              if (*pStep ==4)
              {*pStep = 0;}
              READSW1 = digitalRead(SW1);
              READSW2 = digitalRead(SW2);
              if (READSW2 == LOW) { break;}
             if (READSW1 == LOW && READSW2 == LOW) {break;}
              delay(val);
                    }/*-----------------------------------------------------(4)*/

    }/*----------------(3)*/
              else {/*--------DO NOTHING--------*/}

                    /*ถ้ามีการกดปุ่มสองปุ่มพร้อมกัน == การหยุดการทำงานของ
มอเตอร์*/
                   if (READSW2 == LOW && READSW1 == LOW)
```

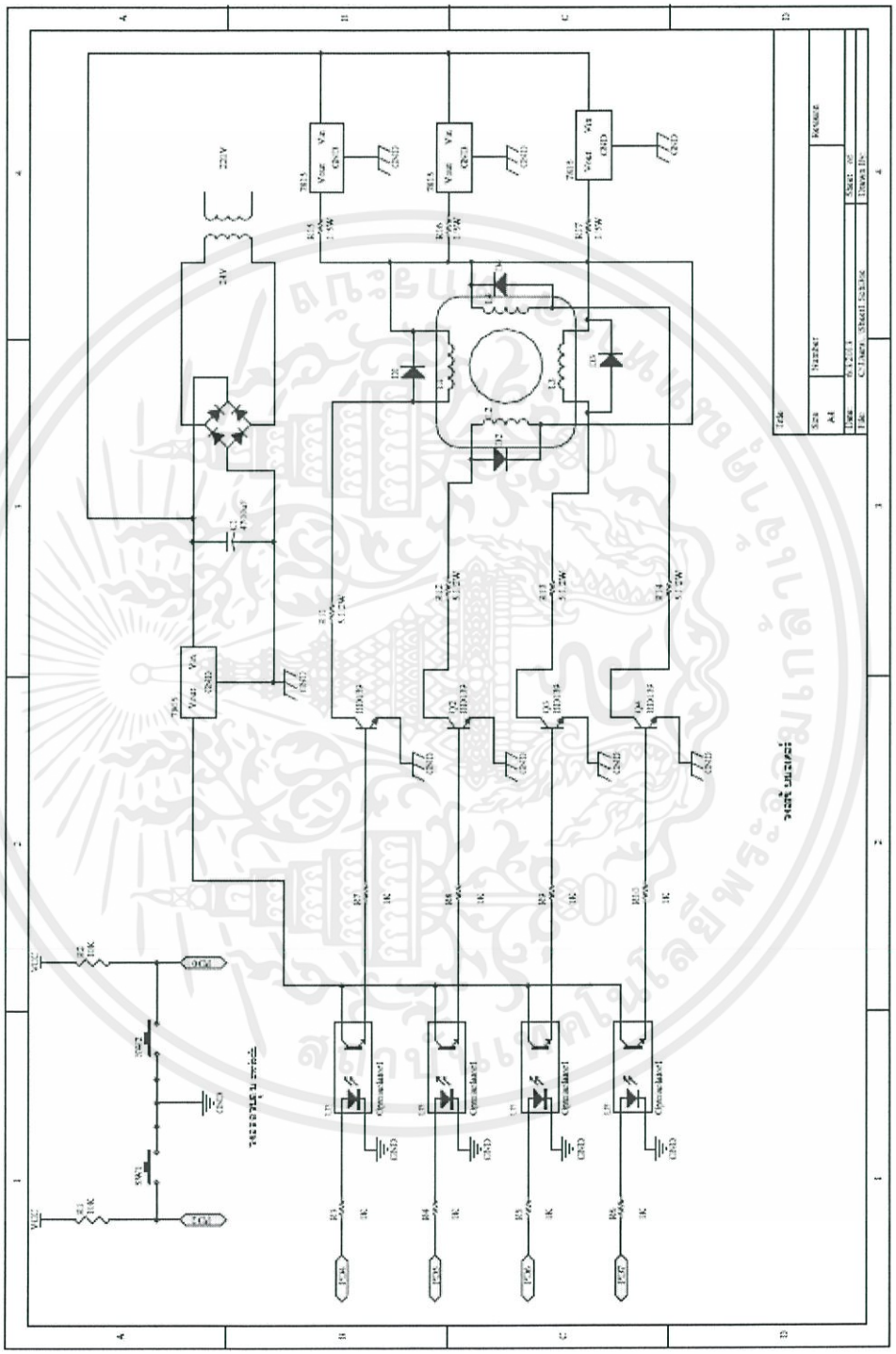
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

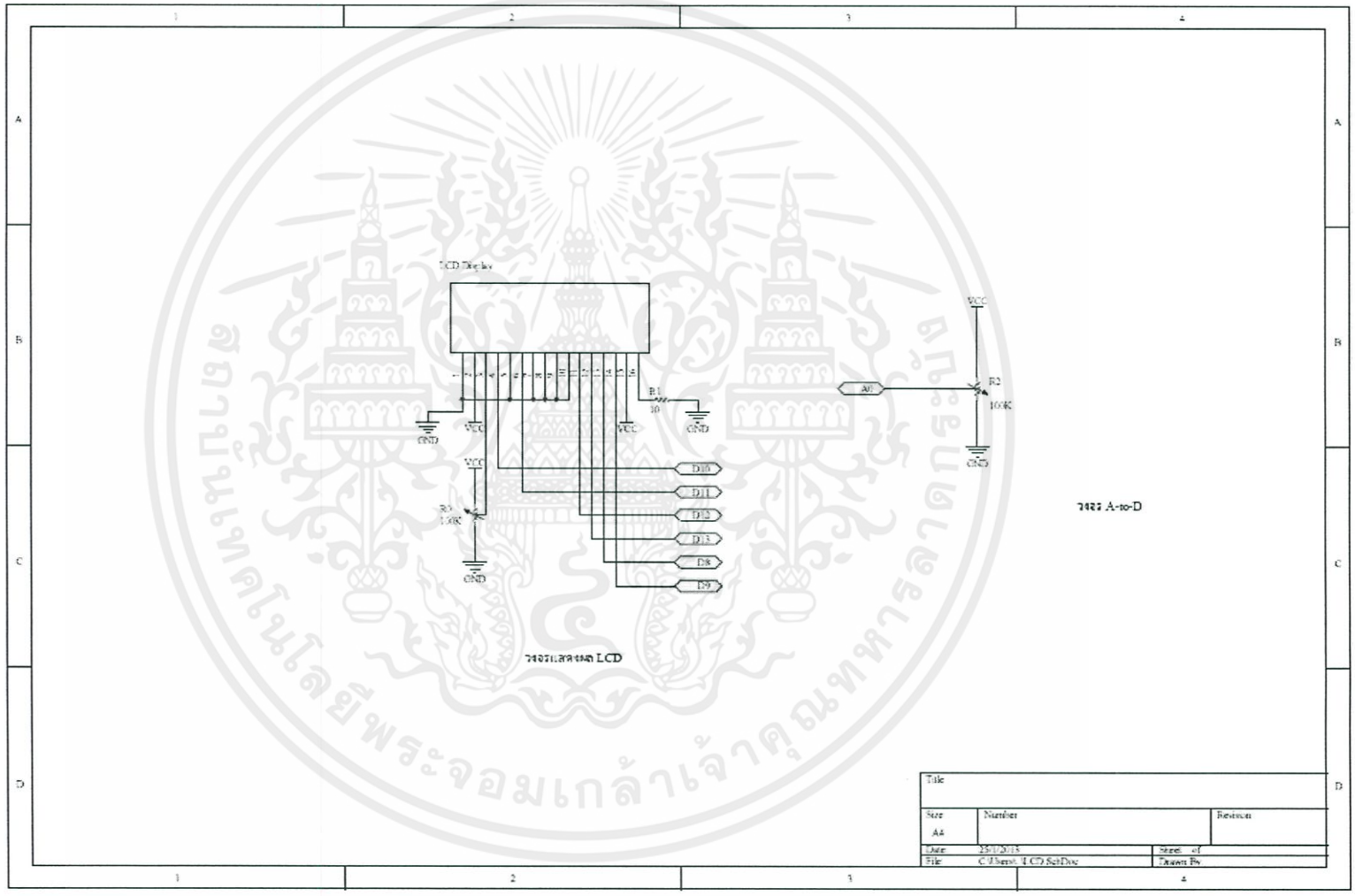
```
        {
         digitalWrite(INA, LOW);
        digitalWrite(INB, LOW);
        digitalWrite(INC, LOW);
        digitalWrite(IND, LOW);

        delay(d);
        READSW33 = digitalRead(SW1);
        READSW44 = digitalRead(SW2);
        if(READSW33 == LOW && READSW44 == LOW)
        {
        digitalWrite(INA, LOW);
        digitalWrite(INB, LOW);
        digitalWrite(INC, LOW);
        digitalWrite(IND, LOW);
        }
    }
        delay(50);
}
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

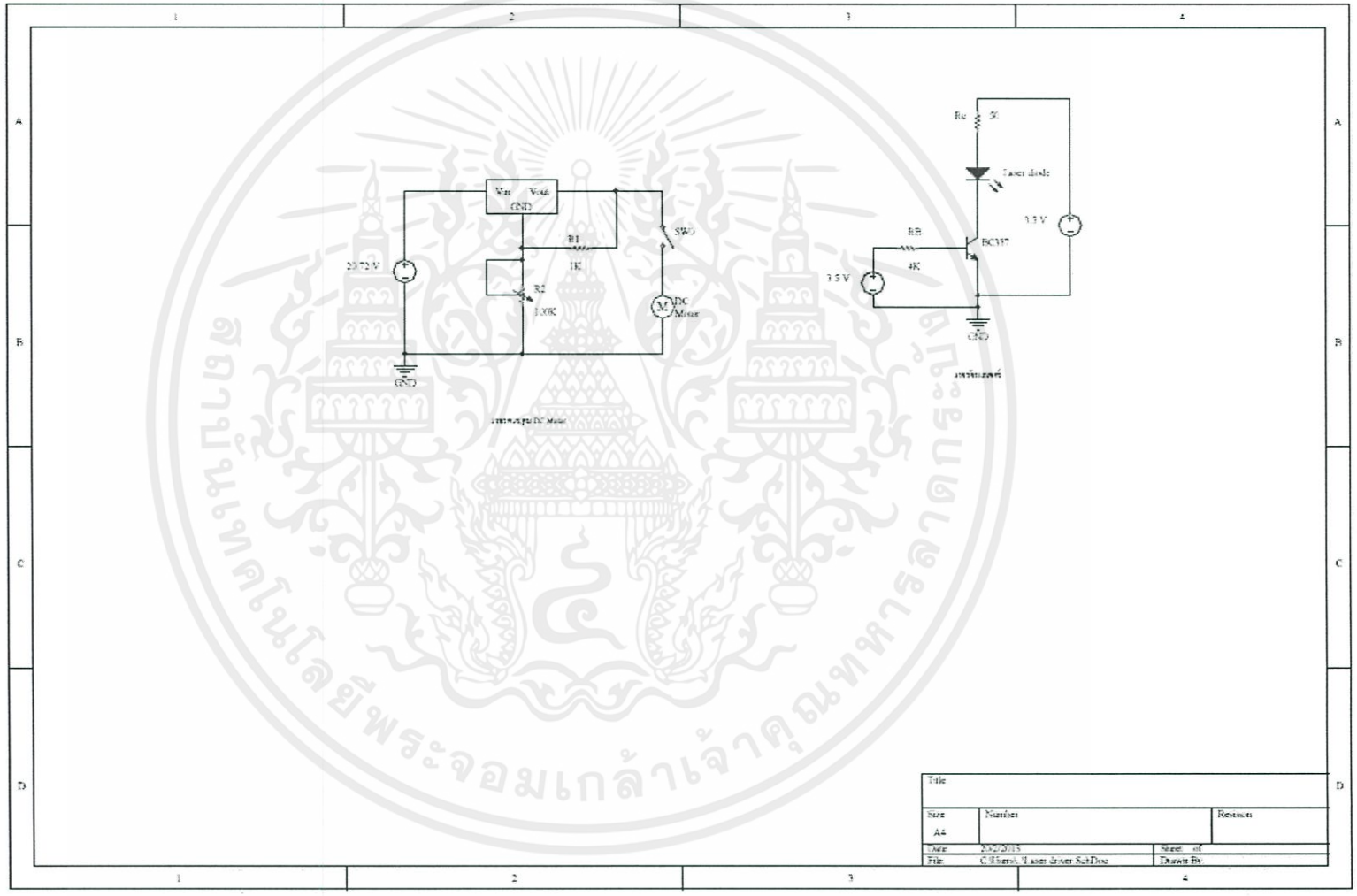
ภาคผนวก ค

รูปวงจรในส่วนฮาร์ดแวร์ (Hardware)





เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า