โครงงาน สมาร์ต กริช แอลอีดี
Smart Grid LED

ปุณิกา เชื้อมั่นคง
พศิน ธรรมทวีวุฒิ
พัชราภรณ์ พิพัฒน์ภานุกูล

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต
สาขาวิศวกรรมการวัดคุม
คณะวิศวกรรมศาสตร์
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
ปีการศึกษา 2556

โครงงาน สมาร์ต กริช แอลอีดี
Smart Grid LED

ปุณิกา เชื้อมั่นคง
พศิน ธรรมทวีวุฒิ
พัชราภรณ์ พิพัฒน์ภานุกูล

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต
สาขาวิศวกรรมการวัดคุม
คณะวิศวกรรมศาสตร์
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
ปีการศึกษา 2556

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Smart Grid LED

PUNIKA CHUEAMANKONG
PASIN THAMTAVEEVOOT
PATCHARAPORN PIPATPANUKUL

A THESIS SUBMITTED IN PARTIAL FULFILLMENT
OF THE REQUIREMENT FOR THE DEGREE OF
BACHELOR OF ENGINEERING IN INSTRUMENTATION ENGINEERING
FACULTY OF ENGINEERING
KING MONGKUT'S INSTITUE OF TECHNOLOGY LARDKRABANG
ACADEMIC YEAR 2013

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ปริญญานิพนธ์ปีการศึกษา 2556
คณะวิศวกรรมศาสตร์
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
ใบรับรองปริญญานิพนธ์

.....

หัวข้อปริญญานิพนธ์	โครงงาน สมาร์ต กริช แอลอีดี Smart Grid LED		
นักศึกษาผู้จัดทำ	นางสาวปุณิกา	เชื้อมั่นคง	**รหัสนักศึกษา** 53011008
	นายพศิน	ธรรมทวีวุฒิ	**รหัสนักศึกษา** 53011077
	นางสาวพัชราภรณ์	พิพัฒน์ภานุกูล	**รหัสนักศึกษา** 53011093
ปริญญา	วิศวกรรมศาสตรบัณฑิต		
สาขาวิชา	วิศวกรรมการวัดคุม		
ปีการศึกษา	2556		

อาจารย์ผู้ควบคุมปริญญานิพนธ์	**ลายมือชื่อ**
ผศ.ดร.พงษ์ชัย นิลาศ	

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หัวข้อปริญญานิพนธ์	โครงงาน สมาร์ต กริช แอลอีดี Smart Grid LED		
นักศึกษาผู้จัดทำ	นางสาวปุณิกา	เชื้อมั่นคง	รหัสนักศึกษา 53011008
	นายพศิน	ธรรมทวีวุฒิ	รหัสนักศึกษา 53011077
	นางสาวพัชราภรณ์	พิพัฒน์ภานุกูล	รหัสนักศึกษา 53011093
อาจารย์ที่ปรึกษา	ผศ.ดร.พงษ์ชัย	นิลาศ	
ปีการศึกษา	2556		

บทคัดย่อ

โครงงานสมาร์ต กริช แอลอีดีนี้ เป็นโครงงานที่ได้ทำการศึกษา ออกแบบ รวมถึงการประกอบฮาร์ดแวร์ของการแสดงผลโดยหลอดแอลอีดี เมื่อมีการหมุน โครงงานนี้ได้ใช้ไมโครคอนโทรลเลอร์ Arduino Due, หลอดแอลอีดี 48 หลอด, บอร์ด Xbee สื่อสารแบบไร้สาย และ การนำเอารูปภาพหรือสัญลักษณ์มาใช้ในการสื่อความหมายระหว่างผู้ใช้กับตัวโปรแกรม ซึ่งเป็นการเขียนในโปรแกรม Visual Basic 2010 ให้สามารถรับคำสั่ง, ตัวอักษร, รูปภาพ จากคอมพิวเตอร์ และ ส่งข้อมูลไปยังบอร์ด Arduino Due เพื่อควบคุมการแสดงผลจากแอลอีดี โครงงานสมาร์ต กริช แอลอีดีจะทำการรับข้อมูลผ่านการสื่อสารแบบไร้สายจากคอมพิวเตอร์และแสดงผลบนหน้าจอแอลอีดี ขณะที่หมุน ในส่วนผลการทดลองนั้น สามารถแสดง ตัวอักษร , สัญลักษณ์ และ รูปภาพ บนแอลอีดีอย่างที่ต้องการได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Thesis Title	Smart Grid LED	
Authors	Miss. Punika	Chueamankong
	Mr. Pasin	Thamtaveevoot
	Miss. Patcharaporn	Pipatpanukul
Thesis Advisor	Assit.Prof.Dr.Phongchai	Nilas
Year	2013	

ABSTRACT

This Smart Grid LED project focuses on study, design as well as hardware assembly the rotational LED display. The project employs Arduino Due microcontroller, forty eight Leds, Xbee wireless board and a Graphical User Interface (GUI) program. The GUI is written in visual basic 2010 to receive commands, texts, images from computer and transmitted to Arduino Due to control the LED display. The Smart Grid LED receives data via wireless network from computer and display them on a rotating Leds. The experiment results could show visible texts, signs and images on the LED output.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กิตติกรรมประกาศ

ปริญญานิพนธ์ฉบับนี้ได้สำเร็จลุล่วงไปได้ด้วยดี เนื่องจากได้รับความเมตตาจากผู้ช่วยศาสตราจารย์ ดร.พงษ์ชัย นิลาศ ที่ได้ให้คำแนะนำและคำสอนแก่ผู้วิจัยปริญญานิพนธ์นี้ตลอดมา อีกทั้งยังเอื้อเฟื้อสถานที่ และเครื่องมือต่างๆที่ใช้ในการทำปริญญานิพนธ์นี้ ผู้วิจัยขอกราบขอบพระคุณเป็นอย่างสูง

ขอขอบพระคุณรองศาสตราจารย์ ประภาษ อุคคกิมาพันธุ์ ภาควิศวกรรมอัตโนมัติ เป็นอย่างสูงที่ได้เอื้อเฟื้อสถานที่ เครื่องมือต่างๆ ให้คำปรึกษาและช่วยเหลือตลอดมา ในการทำปริญญานิพนธ์ ผู้วิจัยได้รู้สึกซาบซึ้งเป็นอย่างมาก ผู้วิจัยขอกราบขอบพระคุณเป็นอย่างสูง

ขอขอบพระคุณผู้ช่วยศาสตราจารย์ เชื้อ นกอยู่ , รองศาสตราจารย์สุพรรณ กุลพาณิชย์ และอาจารย์ภาควิศวกรรมการวัดคุมทุกท่าน ที่ได้เอื้อเฟื้อเครื่องมือต่างๆ และให้คำปรึกษาในการทำปริญญานิพนธ์ ผู้วิจัยได้รู้สึกซาบซึ้งเป็นอย่างมาก

ขอขอบพระคุณน้องสมสิน ทองไกรรัตน์ รุ่น 50 ที่ให้คำปรึกษาอันเป็นประโยชน์ และช่วยเหลือด้วยดีตลอดมา

ขอกราบขอบพระคุณ คุณพ่อ,คุณแม่ และเพื่อนอันเป็นที่รัก ที่คอยสนับสนุน และเป็นแรงบันดาลใจในการทำปริญญานิพนธ์ฉบับนี้ ให้เสร็จลุล่วงไปได้

ผู้วิจัยขอให้ปริญญานิพนธ์ฉบับนี้ ได้ให้ ความรู้ ประโยชน์ และคุณค่า กับผู้อ่านไม่มากก็น้อย

คณะผู้จัดทำ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ

หน้า

บทคัดย่อภาษาไทย I

บทคัดย่อภาษาอังกฤษ II

กิตติกรรมประกาศ III

สารบัญ IV

สารบัญภาพ VII

บทที่ 1 บทนำ 1

1.1 ความสำคัญของปริญญานิพนธ์ 1

1.2 วัตถุประสงค์ของปริญญานิพนธ์ 1

1.3 ขอบเขตของปริญญานิพนธ์ 1

1.4 ขั้นตอนการศึกษา 1

1.5 ประโยชน์ที่คาดว่าจะได้รับ 2

บทที่ 2 ทฤษฎีและหลักการ 3

2.1 ไมโครคอนโทรลเลอร์ 3

2.1.1 หน่วยประมวลผลกลางหรือซีพี 3

2.1.2 หน่วยความจำ 3

2.1.3 ส่วนติดต่อกับอุปกรณ์ภายนอก หรือพอร์ต 3

2.1.4 ช่องทางเดินของสัญญาณ หรือบัส (BUS) 3

2.1.5 วงจรกำเนิดสัญญาณนาฬิกา 3

2.2 ไมโครโปรเซสเซอร์ 4

2.2.1 Reduced Instruction Set Computer 4

2.2.2 Complex Instruction Set Computer 4

2.3 Arduino Due 4

2.3.1 Overview 4

2.3.2 คุณสมบัติทางเทคนิคที่สำคัญ 5

2.4 ไดโอดเปล่งแสง (light-emitting diode) 5

2.4.1 Color (wavelength) 8

2.4.2 ข้อดีของแอลอีดี 8

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ(ต่อ)

หน้า

2.5 มอเตอร์ไฟฟ้ากระแสตรง (DC MOTOR) 8
2.5.1 หลักการของมอเตอร์กระแสไฟฟ้าตรง (Motor Action) 9
2.6 SolidWorks 10
2.6.1 ประสิทธิภาพการทำงาน 10
2.6.2 ลักษณะการทำงาน 10
2.7 Altium 11
2.7.1 การใช้ Altium Designer 11
2.8 Zigbee : XBee-Pro โมดูลสื่อสารข้อมูลไร้สาย 2.4 GHz 12
2.8.1 โมดูล XBee-PRO 12
2.8.1.1 คุณสมบัติทางเทคนิค 12
2.8.1.1.1 คุณสมบัติโดยทั่วไป 12
2.8.1.1.2 คุณสมบัติด้านการสื่อสารข้อมูล 12

บทที่ 3 วิธีการดำเนินงาน 14
3.1 ฮาร์ดแวร์ 14
3.1.1 ส่วนของแมคคานิกส์ 14
3.1.2 ส่วนของวงจร LED DISPLAY 18
3.2 ซอฟท์แวร์ 19
3.2.1 ส่วนโปรแกรม Visual Basic 2010 20
3.2.2 ส่วนโปรแกรม Arduino 1.5.5 28
3.2.3 ส่วนโปรแกรม X CTU 30

บทที่ 4 ผลการทดลอง 33
4.1 การทดสอบโปรแกรมกับบอร์ดไมโครคอนโทรลเลอร์ Arduino Due โดยการสั่งให้กดปุ่มไหนให้หลอดไฟติดเท่ากับเลขปุ่ม 33
4.2 การทดสอบโปรแกรมกับบอร์ดไมโครคอนโทรลเลอร์ Arduino Due โดยสั่งให้หลอดไฟติดหมดทุกพอร์ต 34
4.3 การทดสอบโปรแกรมกับบอร์ดไมโครคอนโทรลเลอร์ Arduino Due โดยสั่งให้หลอดไฟติดทีละพอร์ต 35

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ(ต่อ)

หน้า

4.4 การทดสอบโปรแกรมกับบอร์ดไมโครคอนโทรลเลอร์ Arduino Due โดยสั่งให้หลอดไฟติดสลับดวง 36
4.5 การทดสอบโปรแกรม โดยกดปุ่มในโปรแกรม Visual Basic 2010 จะส่งข้อมูลไปยังไมโครคอนโทรลเลอร์ เพื่อสั่งงานวงจร LED Display ให้แสดงรูป 36
4.5.1 การทดลองกดปุ่ม Button 1 รูปใบพัด 38
4.5.2 การทดลองกดปุ่ม Button 2 รูปใบไม้ 39
4.5.3 การทดลองกดปุ่ม Button 4 รูปหมากรุก 40
4.5.4 การทดลองกดปุ่ม Button 5 รูปตัวเลข 12 41
4.5.5 การทดลองกดปุ่ม Button 6 รูปตัวหนังสือ P 42
4.5.6 การทดลองกดปุ่ม Button 7 รูปตัวสามเหลี่ยมคว่ำ 43
4.5.7 การทดลองกดปุ่ม Button 8 รูปหมวก 44
4.5.8 การทดลองกดปุ่ม Button 9 รูปตัวหนังสือ Zine 45
4.5.9 การทดลองกดปุ่ม Button 10 รูปหน้ายิ้ม 46
4.6 การทดสอบโปรแกรม โดยการ openfile.txt ในโปรแกรม Visual Basic 2010 จะส่งข้อมูลไปยัง ไมโครคอนโทรลเลอร์ เพื่อสั่งงานวงจร LED Display ให้แสดงรูป 47

บทที่ 5 สรุปผลการทดลอง และข้อเสนอแนะ 50
5.1 สรุปผลการทดลอง 50
5.2 ข้อเสนอแนะ 50

บรรณานุกรม..... 51

ภาคผนวก 53
ภาคผนวก ก. การออกแบบโดยใช้โปรแกรม Solidworks 54
ภาคผนวก ข. การออกแบบวงจรโดยใช้โปรแกรม Altium 61
ภาคผนวก ค. โปรแกรม 64

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญภาพ

ภาพที่ หน้า

2.1 แสดงภาพสถาปัตยกรรมของบอร์ด 4
2.2 แสดงภาพไดโอดเปล่งแสง 5
2.3 แสดงภาพส่วนประกอบของไดโอด 6
2.4 แสดงภาพวงจรที่ทำให้ไดโอดเปล่งแสง 7
2.5 แสดงภาพการหาค่าความต้านทานของตัวต้านทาน 7
2.6 แสดงภาพโครงสร้างเบื้องต้นของมอเตอร์กระแสตรง 9
2.7 แสดงภาพทิศการหมุนของมอเตอร์ 9
3.1 แสดงภาพอุปกรณ์และแบบที่ใช้สร้างฐานของโครงงาน 14
3.2 แสดงภาพขั้นตอนการเชื่อมแท่งเหล็กเข้าด้วยกัน 15
3.3 แสดงภาพฐานของโครงงาน 15
3.4 แสดงภาพฐานของโครงงานเมื่อลงสีเรียบร้อยแล้ว 15
3.5 แสดงภาพดีซีมอเตอร์ 16
3.6 แสดงภาพพาวเวอร์ซัพพลาย 16
3.7 แสดงภาพฐานรองบอร์ดและวงจร LED Display 17
3.8 แสดงภาพ วงจร LED Display 17
3.9 แสดงภาพ Zigbee 18
3.10 แสดงภาพชิ้นงานเมื่อประกอบเสร็จ 18
3.11 แสดงภาพ Schematic ของวงจร LED 19
3.12 แสดงภาพ PCB 19
3.13 แสดงภาพการทำงานของโปรแกรม 20
3.14 แสดงภาพหน้าจอเริ่มต้นโปรแกรม Visual Basic 2010 ก่อนเริ่มทำงาน 20
3.15 แสดงภาพหน้าต่างโปรแกรม Visual Basic 2010 เลือก Serial Comport. 21
3.16 แสดงภาพหน้าต่างโปรแกรม Visual Basic 2010 กดปุ่ม Connect 21
3.17 แสดงภาพหน้าต่างโปรแกรม Visual Basic 2010 กดปุ่ม Button 1 21
3.18 แสดงภาพหน้าต่างโปรแกรม Visual Basic 2010 กดปุ่ม Button 2 22
3.19 แสดงภาพหน้าต่างโปรแกรม Visual Basic 2010 กดปุ่ม Button 3 22
3.20 แสดงภาพหน้าต่างโปรแกรม Visual Basic 2010 กดปุ่ม Button 4 22
3.21 แสดงภาพหน้าต่างโปรแกรม Visual Basic 2010 กดปุ่ม Button 5 23

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญภาพ(ต่อ)

ภาพที่ **หน้า**

3.22 แสดงภาพหน้าต่างโปรแกรม Visual Basic 2010 กดปุ่ม Button 6 23
3.23 แสดงภาพหน้าต่างโปรแกรม Visual Basic 2010 กดปุ่ม Button 7 23
3.24 แสดงภาพหน้าต่างโปรแกรม Visual Basic 2010 กดปุ่ม Button 8 24
3.25 แสดงภาพหน้าต่างโปรแกรม Visual Basic 2010 กดปุ่ม Button 9 24
3.26 แสดงภาพหน้าต่างโปรแกรม Visual Basic 2010 กดปุ่ม Button 10 24
3.27 แสดงภาพหน้าต่างโปรแกรม Visual Basic 2010 กดปุ่ม Disconnect 25
3.28 แสดงภาพหน้าจอเริ่มต้นโปรแกรม Visual Basic 2010 ก่อนเริ่มทำงาน 25
3.29 แสดงภาพหน้าต่างโปรแกรม Visual Basic 2010 เลือก Serial Comport 25
3.30 แสดงภาพหน้าต่างโปรแกรม Visual Basic 2010 กดปุ่ม Open 26
3.31 แสดงหน้าต่างโปรแกรม Visual Basic 2010 กดปุ่ม openfile 26
3.32 แสดงหน้าต่างโปรแกรม Visual Basic 2010 เมื่อกดปุ่ม openfile แล้วเลือกไฟล์ .txt ในการแสดงผล 26
3.33 แสดงหน้าต่างโปรแกรม Visual Basic 2010 เมื่อกดปุ่ม openfile เลือกไฟล์ที่ต้องการแล้ว 27
3.34 แสดงหน้าต่างโปรแกรม Visual Basic 2010 เมื่อกดปุ่ม Send ข้อมูลที่ต้องการไปยังไมโครคอนโทรลเลอร์เพื่อแสดงผลที่แผงวงจรไฟ LED 27
3.35 แสดงภาพหน้าต่างโปรแกรม Visual Basic 2010 กดปุ่ม Close 28
3.36 แสดงภาพการออกแบบโปรแกรมที่ใช้ในการควบคุม 29
3.37 แสดงภาพลำดับการส่งข้อมูลของโปรแกรม 30
3.38 แสดงภาพการทำ PC Setting 30
3.39 แสดงภาพเมื่อกด Test / Query 31
3.40 แสดงภาพการใส่ค่า DH,DL 31
3.41 แสดงภาพการส่งรับค่าสลับกัน 32
4.1 แสดงภาพการทดสอบโปรแกรม การสั่งให้กดปุ่ม 'Button1' 33
4.2 แสดงภาพการทดสอบโปรแกรม การสั่งให้กดปุ่ม 'Button2' 33
4.3 แสดงภาพการทดสอบโปรแกรม การสั่งให้กดปุ่ม 'Button3' 34
4.4 แสดงภาพการทดสอบโปรแกรม การสั่งให้กดปุ่ม 'Button4' 34
4.5 แสดงภาพการทดสอบโปรแกรมสั่งให้ติดทุกพอร์ต 35
4.6 แสดงภาพการทดสอบโปรแกรมสั่งให้ติดทีละพอร์ต 35
4.7 แสดงภาพการทดสอบโปรแกรมสั่งให้ติดติดสลับดวง 36
4.8 แสดงภาพการเลือกคอมพอร์ต แล้วทำการเชื่อมต่อไมโครคอนโทรลเลอร์ 36

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญภาพ(ต่อ)

ภาพที่ **หน้า**

4.9 แสดงภาพรูป Encoder โดยการใช้โปรแกรม Edit Bitmap 37
4.10 แสดงภาพการแสดงผลของวงจร LED Display รูป Encoder เมื่อจ่ายไฟ 37
4.11 แสดงภาพรูป ใบพัด โดยการใช้โปรแกรม Edit Bitmap 38
4.12 แสดงภาพการแสดงผลของวงจร LED Display ของใบพัด 38
4.13 แสดงภาพรูปใบไม้ โดยการใช้โปรแกรม Edit Bitmap 39
4.14 แสดงภาพการแสดงผลของวงจร LED Display ของใบไม้ 39
4.15 แสดงภาพรูป หมากรุก โดยการใช้โปรแกรม Edit Bitmap 40
4.16 แสดงภาพการแสดงผลของวงจร LED Display ของหมากรุก 40
4.17 แสดงภาพ รูปเลข 12 โดยการใช้โปรแกรม Edit Bitmap 41
4.18 แสดงภาพการแสดงผลของวงจร LED Display ของรูปเลข 12 41
4.19 แสดงภาพ รูปตัวหนังสือ P โดยการใช้โปรแกรม Edit Bitmap 42
4.20 แสดงภาพการแสดงผลของวงจร LED Display ของรูปตัวหนังสือ P 42
4.21 แสดงภาพ รูปสามเหลี่ยมคว่ำ โดยการใช้โปรแกรม Edit Bitmap 43
4.22 แสดงภาพการแสดงผลของวงจร LED Display ของรูปรูปสามเหลี่ยมคว่ำ 43
4.23 แสดงภาพรูปหมวก โดยการใช้โปรแกรม Edit Bitmap 44
4.24 แสดงภาพการแสดงผลของวงจร LED Display ของรูปหมวก 44
4.25 แสดงภาพรูปตัวหนังสือ Zine โดยการใช้โปรแกรม Edit Bitmap 45
4.26 แสดงภาพการแสดงผลของวงจร LED Display ของรูปตัวหนังสือ Zine 45
4.27 แสดงภาพรูปหน้ายิ้มโดยการใช้โปรแกรม Edit Bitmap 46
4.28 แสดงภาพการแสดงผลของวงจร LED Display ของรูปหน้ายิ้ม 46
4.29 แสดงภาพการเลือกคอมพอร์ต แล้วทำการเชื่อมต่อไมโครคอนโทรลเลอร์ 47
4.30 แสดงภาพหลอดไฟชุดแรกและชุดที่สี่ หลอดไฟ LED ติดทุกดวง 47
4.31 แสดงภาพหลอดไฟชุดที่สองและชุดที่สาม หลอดไฟ LED ติดแค่ดวงที่1 และดวงที่4 48
4.32 แสดงภาพ Square 8x8 48
4.33 แสดงภาพ triangle 8x8 49

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 1
บทนำ

1.1 ความสำคัญของปริญญานิพนธ์

เนื่องจากในสังคมปัจจุบันความเจริญก้าวหน้าในทางอิเล็คทรอนิคส์มีการพัฒนาไปอย่างไม่มีที่สิ้นสุด ความเจริญทางด้านเทคโนโลยีดังกล่าวไม่ว่าจะเป็นเทคโนโลยีทางด้านฮาร์ดแวร์หรือเทคโนโลยีทางด้านซอร์ฟแวร์ ได้ถูกนำมาใช้ในการพัฒนาคุณภาพชีวิตของมนุษย์ในปัจจุบันเป็นอย่างมาก จากการพัฒนาคุณภาพนี้ จึงทำให้มีการพัฒนาไมโครคอนโทรลเลอร์ขึ้นมาอย่างแพร่หลาย ทำให้การพัฒนาของซีพียูดีมากขึ้นด้วยพื้นฐานในการสร้างขึ้นมาจากสถาปัตยกรรมเดียวกันชื่อว่า ARM ทั้งด้านออกแบบและการผลิต แต่ในที่นี้เราจะใช้ Arduino Due เป็นบอร์ดไมโครคอนโทรลเลอร์รุ่นแรกสุดของ Arduino ที่ใช้ไมโครคอนโทรลเลอร์ 32 บิต ARM Cortex M3 เป็นไมโครคอนโทรลเลอร์ AT91SAM3X8E ซึ่ง Arduino Due มีขนาดพอร์ตอินพุตเอาต์พุตให้ใช้งานมากถึง 54 ขา ซึ่งเป็นคุณสมบัติเด่นที่เราเลือกนำมาใช้คือมีจำนวน I/O มาก ดังนั้นเราจึงใช้ Arduino Due มาประยุกต์ใช้งานกับระบบแสดงผลภาพด้วยหลอด LED โดยเป็นการควบคุมหลอด LED

1.2 วัตถุประสงค์ของปริญญานิพนธ์

1. ศึกษาค้นคว้าการทำงานของ Smart Grid LED
2. ทำการรวบรวมข้อมูล โดยการศึกษาวิเคราะห์การออกแบบและรูปแบบการทำงานของ Smart Grid LED
3. สามารถเลือกอุปกรณ์ต่างๆ รวมทั้งไมโครโปรเซสเซอร์ มาใช้งานในโครงงานได้
4. สามารถเขียนโปรแกรม ไมโครโปรเซสเซอร์ได้
5. สามารถทำการประกอบ ทำการทดสอบ และทำการทดลอง แล้วนำผลการทดลองที่ได้มาวิเคราะห์ได้

1.3 ขอบเขตของปริญญานิพนธ์

1. ทำการศึกษาและแสดงผลโดยการใช้หลอดLED
2. ศึกษาหลักการทำงานของหลอด LED

1.4 ขั้นตอนการศึกษา

1. ศึกษาหลักการทำงานของหลอด LED
2. ศึกษาการทำงานของไมโครคอนโทรลเลอร์ Arduino Due
3. ศึกษาการเขียนโปรแกรม SolidWorks 2013 เพื่อสำหรับออกแบบฮาร์ดแวร์
4. ศึกษาการเขียนโปรแกรม Altium Designer เพื่อสำหรับออกแบบวงจร LED DISPLAY
5. ศึกษาการทำบิตแมป เป็นการแปลงจากภาพสีเป็นภาพขาวดำ
6. ศึกษาการแมปภาพจากสี่เหลี่ยมเป็นภาพวงกลม
7. ศึกษาการเขียนโปรแกรม Visual Basic 2010 เพื่อติดต่อกับบอร์ดไมโครคอนโทรลเลอร์
8. ศึกษาการเขียนโปรแกรม Arduino 1.5.5 เพื่อควบคุมวงจร LED DISPLAY

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1.5 ประโยชน์ที่คาดว่าจะได้รับ

1. ทำให้เข้าใจในหลักการทำงานและการเขียนโปรแกรมเพื่อควบคุมไมโครคอนโทรลเลอร์ Arduino Due
2. ทำให้ได้เรียนรู้การออกแบบโดยใช้โปรแกรม SolidWorks 2013 เพื่อออกแบบฮาร์ดแวร์
3. ทำให้ได้เรียนรู้การออกแบบวงจรโดยใช้โปรแกรม Altium Designer เพื่อออกแบบวงจร LED DISPLAY
4. ทำให้ได้เรียนรู้และเข้าใจการเขียนโปรแกรม Visual Basic 2010
5. ทำให้ได้เรียนรู้ถึงเรื่องการบิตแมปภาพ

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 2
ทฤษฎีและหลักการ

2.1 ไมโครคอนโทรลเลอร์

ไมโครคอนโทรลเลอร์ เป็นอุปกรณ์ควบคุมขนาดเล็ก ซึ่งบรรจุความสามารถที่คล้ายคลึงกับระบบคอมพิวเตอร์ โดยในไมโครคอนโทรลเลอร์ได้รวมเอาซีพียู, หน่วยความจำ และพอร์ต ซึ่งเป็นส่วนประกอบหลักสำคัญของระบบคอมพิวเตอร์เข้าไว้ด้วยกัน โดยทำการบรรจุเข้าไว้ในตัวถังเดียวกัน โครงสร้างทั่วไป

โครงสร้างโดยทั่วไป ของไมโครคอนโทรลเลอร์นั้น สามารถแบ่งออกมาได้เป็น 5 ส่วนใหญ่ๆ ดังต่อไปนี้

2.1.1 หน่วยประมวลผลกลางหรือซีพียู (CPU: Central Processing Unit)

2.1.2 หน่วยความจำ (Memory) สามารถแบ่งออกเป็น 2 ส่วน คือ หน่วยความจำที่มีไว้สำหรับเก็บโปรแกรมหลัก (Program Memory) เปรียบเสมือนฮาร์ดดิสก์ของเครื่องคอมพิวเตอร์ตั้งโต๊ะ คือข้อมูลใดๆ ที่ถูกเก็บไว้ในนี้จะไม่สูญหายไปแม้ไม่มีไฟเลี้ยง อีกส่วนหนึ่งคือหน่วยความจำข้อมูล (Data Memory) ใช้เป็นเหมือนกระดาษทดในการคำนวณของซีพียู และเป็นที่พักข้อมูลชั่วคราวขณะทำงาน แต่หากไม่มีไฟเลี้ยง ข้อมูลก็จะหายไปคล้ายกับหน่วยความจำแรม (RAM) ในเครื่องคอมพิวเตอร์ทั่วๆ ไป แต่สำหรับไมโครคอนโทรลเลอร์สมัยใหม่ หน่วยความจำข้อมูลจะมีทั้งที่เป็นหน่วยความจำแรม ซึ่งข้อมูลจะหายไปเมื่อไม่มีไฟเลี้ยง และเป็นอีอีพรอม (EEPROM : Erasable Electrically Read-Only Memory) ซึ่งสามารถเก็บข้อมูลได้แม้ไม่มีไฟเลี้ยง

2.1.3 ส่วนติดต่อกับอุปกรณ์ภายนอก หรือพอร์ต (Port) มี 2 ลักษณะคือ พอร์ตอินพุต (Input Port) และพอร์ตส่งสัญญาณหรือพอร์ตเอาต์พุต (Output Port) ส่วนนี้จะใช้ในการเชื่อมต่อกับอุปกรณ์ภายนอก ถือว่าเป็นส่วนที่สำคัญมาก ใช้ร่วมกันระหว่างพอร์ตอินพุต เพื่อรับสัญญาณอาจจะด้วยการกดสวิตช์ เพื่อนำไปประมวลผลและส่งไปพอร์ตเอาต์พุต เพื่อแสดงผลเช่น การติดสว่างของหลอดไฟ เป็นต้น

2.1.4 ช่องทางเดินของสัญญาณ หรือบัส (BUS) คือเส้นทางการแลกเปลี่ยนสัญญาณข้อมูลระหว่าง ซีพียู หน่วยความจำและพอร์ต เป็นลักษณะของสายสัญญาณ จำนวนมากอยู่ภายในตัวไมโครคอนโทรลเลอร์ โดยแบ่งเป็นบัสข้อมูล (Data Bus), บัสแอดเดรส (Address Bus) และบัสควบคุม (Control Bus)

2.1.5 วงจรกำเนิดสัญญาณนาฬิกา นับเป็นส่วนประกอบที่สำคัญมากอีกส่วนหนึ่ง เนื่องจากการทำงานที่เกิดขึ้นในตัวไมโครคอนโทรลเลอร์ จะขึ้นอยู่กับการกำหนดจังหวะ หากสัญญาณนาฬิกามี

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ความถี่สูง จังหวะการทำงานก็จะสามารถทำได้ถี่ขึ้นส่งผลให้ไมโครคอนโทรลเลอร์ตัวนั้น มีความเร็วในการประมวลผลสูงตามไปด้วย

2.2 ไมโครโปรเซสเซอร์

ไมโครโปรเซสเซอร์เป็นวงจรที่มีโครงสร้างตรรกะ คล้ายกับหน่วยประมวลผลกลางของดิจิตอลคอมพิวเตอร์ ซึ่งโครงสร้างของมันจะประกอบด้วยไอซีแบบ LSI (Large-Scale Integrated Circuit) ทำหน้าที่จัดการข้อมูล ทำการคำนวณภายใต้การควบคุมของโปรแกรม เราอาจเรียกไมโครโปรเซสเซอร์อีกอย่างหนึ่งว่าเป็นการประมวลผลข้อมูล การประมวลผลข้อมูลเป็นหน้าที่หลักของไมโครโปรเซสเซอร์ ซึ่งประกอบด้วยข้อมูล สำหรับการคำนวณจะใช้วงจรเฉพาะที่มีชื่อว่า ALU (Arithmetic & Logical Unit) โดยวงจรนี้สามารถทำการบวก ลบ AND OR เปรียบเทียบเพิ่มค่า ลดค่าข้อมูลในการประมวลผล ไมโครโปรเซสเซอร์จะต้องมีหน่วยควบคุม เพื่อที่บอกให้แปลความหมายคำสั่ง และปฏิบัติตามคำสั่งของโปรแกรม โดยขั้นตอนแรกหน่วยควบคุมจะอ่านคำสั่งจากโปรแกรมหน่วยความจำ เราเรียกขั้นตอนนี้ว่า เฟตช์ – เอ็กซีคิวต์ไซเคิล

ไมโครโปรเซสเซอร์หมายถึงชิปที่ใช้เป็นหน่วยประมวลผลกลางของเครื่องไมโครคอมพิวเตอร์ เราสามารถแบ่งไมโครโพรเซสเซอร์ตามสถาปัตยกรรมได้เป็น 2 ชนิด คือ

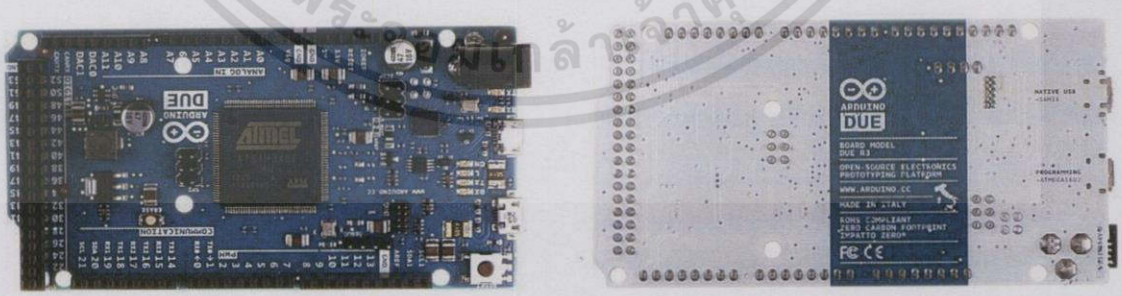
2.2.1 Reduced Instruction Set Computer

RISC (Reduced Instruction Set Computer) คือ ไมโครโพรเซสเซอร์ที่มีคำสั่งน้อย แต่คำสั่งทำงานได้เร็ว

2.2.2 Complex Instruction Set Computer

CISC (Complex Instruction Set Computer) เป็นสถาปัตยกรรมของไมโครโพรเซสเซอร์ที่มีคำสั่งมากกว่าและซับซ้อนกว่า

2.3 Arduino Due



รูปที่ 2.1 แสดงภาพสถาปัตยกรรมของบอร์ด

2.3.1 Overview

Arduino Due เป็นบอร์ดไมโครคอนโทรลเลอร์รุ่นแรกสุดของครอบครัว Arduino ใช้ไมโครคอนโทรลเลอร์ 32บิต ARM Cortex M3 เป็นบอร์ด Arduino ที่เปลี่ยนชิป MCU (Multipoint

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Control Unit) ใหม่ ซึ่งจากเดิมเป็นตระกูล AVR เปลี่ยนเป็นเบอร์ AT91SAM3X8E (ตระกูล ARM Cortex-M3) แทน ทำให้การประมวลผลเร็วขึ้น แต่ยังคงรูปแบบโค้ดโปรแกรมของ Arduino ที่ง่ายอยู่ ข้อควรระวัง: เนื่องจาก MCU เป็นคนละเบอร์กับ Arduino Uno R3 อาจะทำให้บอร์ด Shield บางตัวหรือ Library ใช้ร่วมกันกับบอร์ด Arduino Leonardo ไม่ได้ ผู้ใช้งานจำเป็นต้องตรวจสอบก่อนใช้งาน

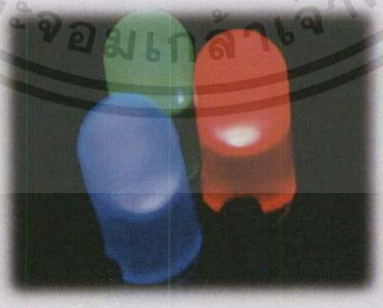
Arduino Due มีพอร์ตอินพุตเอาต์พุตให้ใช้งานมากถึง 54 ขา โดยรวมเอาความสามารถของโมดูล PWM ไว้ 12 ช่อง, อินพุตอนาล็อก 12 ช่อง, 4 UART, DAC 2 ชุด, I^2C 2 ชุด, โมดูล CAN และรองรับการทำงานกับอุปกรณ์ USB ในแบบ USB-OTG จึงนับเป็นบอร์ด Arduino ที่มีความสามารถสูง ภายใต้ราคาย่อมเยาว์ Arduino Due ต้องใช้กับซอฟต์แวร์ Arduino IDE version 1.5 ขึ้นไป

2.3.2 คุณสมบัติทางเทคนิคที่สำคัญ

ไมโครคอนโทรลเลอร์ : AT91SAM3X8E จาก Atmel หน่วยความจำแฟลช 512 กิโลไบต์ , แรม 96 กิโลไบต์ มีความสามารถ USB-OTG ทำหน้าที่เป็นอุปกรณ์ USB โฮสต์ เพื่อต่อกับเมาส์ คีย์บอร์ด และสมาร์ตโฟนได้

- ความถี่สัญญาณนาฬิกา: 84 MHz
- ไฟเลี้ยงทำงาน: 3.3 V
- ย่านไฟเลี้ยงอินพุต: 7 ถึง 12 V
- พอร์ตอินพุตเอาต์พุต: 54 ช่อง (รวมเอาต์พุต PWM 12 ช่อง)
- อินพุตอนาล็อก: 12 ช่อง ความละเอียด 12 บิต รับแรงดัน 0 ถึง +3.3 V
- เอาต์พุตอนาล็อก (DAC): 2 ช่อง ความละเอียด 12 บิต ให้แรงดัน 0 ถึง +3.3V
- บัสสื่อสารข้อมูล: I^2C 2 ชุด, SPI, JTAG, CAN, UART 4 ชุด
- ติดต่อคอมพิวเตอร์: พอร์ต USB โดยใช้สาย micro-USB

2.4 ไดโอดเปล่งแสง (light-emitting diode)



รูปที่ 2.2 แสดงภาพไดโอดเปล่งแสง

เรียกย่อ ๆ ว่า LED คือ ซึ่งสามารถเปล่งแสงออกมาได้ แสงที่เปล่งออกมาประกอบด้วยคลื่นความถี่เดียวและเฟสต่อเนื่องกัน ซึ่งต่างกับแสงธรรมดาที่ตาคนมองเห็น โดยหลอด LED สามารถ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

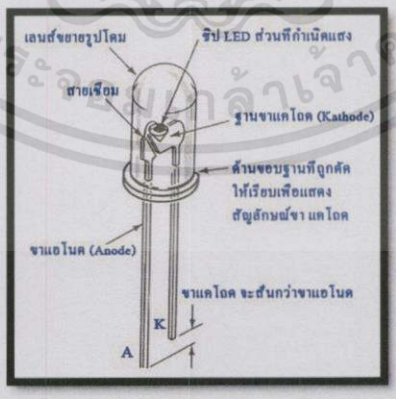
เปล่งแสงได้เมื่อจ่ายกระแสไฟฟ้าเข้าเพียงเล็กน้อยเท่านั้น และประสิทธิภาพในการให้แสงสว่างก็ยังดีกว่าหลอดไฟขนาดเล็กทั่วๆ ไป

ไดโอดชนิดนี้ (LED เหมือนไดโอดทั่ว ๆ ไปที่ประกอบด้วยสารกึ่งตัวนำชนิด P และ N ประกบกันมีผิวข้างหนึ่งเรียบเป็นมันคล้ายกระจก เมื่อไดโอดตกไบแอสตรงจะทำให้อิเล็กตรอนที่สารกึ่งตัวนำชนิด N มีพลังงานสูงขึ้นจนสามารถวิ่งข้ามรอยต่อไปรวมกับโฮลใน P ต่อให้เกิดพลังงานในรูปของประจุโฟตอน ซึ่งจะส่งแสงออกมา การประยุกต์ LED ไปใช้งานอย่างกว้างขวางส่วนมากใช้ในภาคแสดงผล (LED display) LED โดยทั่วไปมี 2 ชนิดใหญ่ ๆ คือ LED ชนิดที่ตาคนเห็นได้ กับชนิดที่ตาคนมองไม่เห็นต้องใช้ทรานซิสเตอร์มาเป็นตัวรับแสงแทนตาคน

ปัจจุบันจากความก้าวหน้าอย่างรวดเร็วของเทคโนโลยีเซมิคอนดักเตอร์ ทำให้เทคโนโลยีของ LED ก้าวหน้าอย่างรวดเร็วตามไปด้วย ได้มีการนำ LED มา ใช้ประโยชน์แพร่หลายมากขึ้นเรื่อยๆ เช่น ในเครื่องคิดเลข สัญญาณจราจร ไฟท้ายรถยนต์ ป้ายสัญญาณต่างๆ ไฟฉาย ไฟให้สัญญาณของประภาคาร จอภาพยนตร์ขนาดใหญ่ ยิ่งไปกว่านั้น หน้าจอ LCD ของโทรศัพท์มือถือที่เราใช้กันทั่วไปเกือบทั้งหมดจะให้แสงสว่างด้วย LED

LED นับเป็นอุปกรณ์เซมิคอนดักเตอร์แบบหนึ่งที่ยอมให้กระแสไฟฟ้าไหลผ่านและจะปล่อยแสงสว่างออกมา ความจริงแล้ว LED ไม่ใช่เรื่องใหม่แต่อย่างใด โดยนักวิทยาศาสตร์ได้สังเกตมาตั้งแต่ปี 2450 ว่าเซมิคอนดักเตอร์จะเปล่งแสงออกมาเมื่อกระแสไฟฟ้าไหลผ่าน อย่างไรก็ตามแสงที่เปล่งออกมามีปริมาณน้อยมาก จึงทำให้เทคโนโลยีนี้ไม่ได้รับความสนใจ

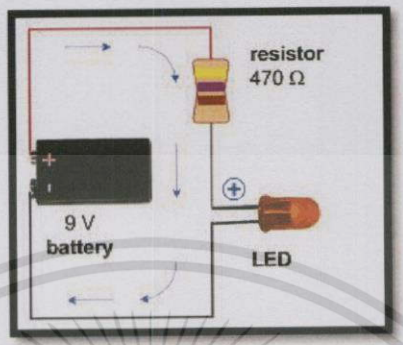
ไดโอดเปล่งแสง เป็นอุปกรณ์อิเล็กทรอนิกส์ ที่ทำงานโดยการเปล่งแสงออกมา ส่วนใหญ่แล้วจะถูกนำไปใช้งาน สำหรับแสดงผลสถานะ การทำงานของระบบต่าง ๆ ในวงจรอิเล็กทรอนิกส์ เช่น แสดงสถานะปิดและเปิดเครื่องรับโทรทัศน์ โทรศัพท์มือถือ เครื่องคอมพิวเตอร์ เป็นต้น โครงสร้างของไดโอดเปล่งแสงแสดงดังรูป 2.3



รูปที่ 2.3 แสดงภาพส่วนประกอบของไดโอด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ไดโอดเปล่งแสง จะถูกใช้งานสำหรับการแสดงผลสถานะ การทำงานของส่วนต่าง ๆ ของวงจรอิเล็กทรอนิกส์ในรูปของหลอดไฟเป็นสีต่าง ๆ ไดโอดจะทำงานได้ก็ต้องได้รับแรงดันไฟฟ้าแบบไบแอสตรง ดังนั้นสิ่งที่ต้องพิจารณาการนำไดโอดเปล่งแสงมาใช้งานมีดังนี้



รูปที่ 2.4 แสดงภาพวงจรที่ทำให้ไดโอดเปล่งแสง

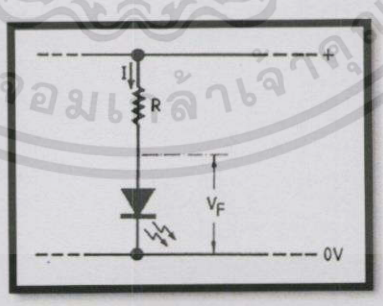
1. การต่อวงจรไดโอดเปล่งแสงให้สามารถทำงานได้นั้น จะต้องต่อแบบไบแอสตรง

2. แรงดันและกระแสไฟฟ้าใช้งานของไดโอด ในการนำไดโอดมาต่อใช้งานนั้น เราจะต้องทราบค่า แรงดันและกระแสไฟฟ้าที่ไดโอดสามารถทำงานได้โดยไม่เกิดความเสียหาย โดยปกติแล้วไดโอดเปล่งแสงจะมีค่าแรงดันและกระแสไฟฟ้าดังนี้

2.1 แรงดันไฟฟ้าใช้งาน 2 – 5 โวลต์ เป็นแรงดันที่ไดโอดเปล่งแสงทำงานได้

2.2 กระแสไฟฟ้าใช้งาน 5 – 20 มิลลิแอมป์

ดังนั้นเราจึงต้องต่อตัวต้านทานอนุกรมกับไดโอดเปล่งแสงและคำนวณหาค่าความต้านทานเพื่อจำกัดกระแสไฟฟ้าให้อยู่ในช่วง 5 – 20 มิลลิแอมป์ แรงดันตกคร่อมไดโอดเปล่งแสงประมาณ 1.2 – 5 โวลต์ (V_F)



รูปที่ 2.5 แสดงภาพการหาค่าความต้านทานของตัวต้านทาน

สำหรับการหาค่าความต้านทานของตัวต้านทาน จะใช้กฎของโอห์มในการคำนวณ

$$R = \frac{E}{I}$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โดย E = V_F = แรงดันตกคร่อมไดโอดเปล่งแสง (V), I = กระแสไฟฟ้าใช้งานของไดโอด (mA)

2.4.1 Color (wavelength)

เป็นตัวบอกสี ซึ่งหมายถึงขนาดของความยาวคลื่นที่ LED เปล่งแสงออกมา เช่น

สีฟ้า จะมีความยาวคลื่น ประมาณ 468 nm

สีขาว จะมีความยาวคลื่น ประมาณ 462 nm

สีเหลือง จะมีความยาวคลื่น ประมาณ 468 nm

สีเขียว จะมีความยาวคลื่น ประมาณ 565 nm

สีแดง จะมีความยาวคลื่น ประมาณ 630 nm

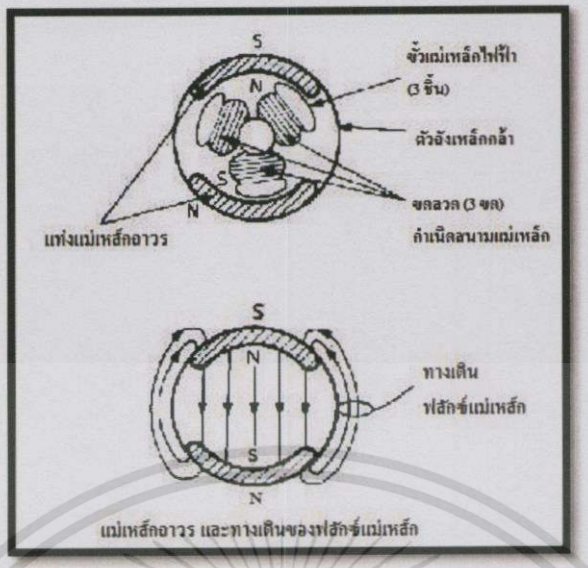
2.4.2 ข้อดีของแอลอีดี

- ประสิทธิภาพในการให้แสงสว่างดีกว่าหลอดไฟธรรมดาทั่วๆไป
- ตัวหลอด LED เองเมื่อทำให้เกิดแสงขึ้นจะกินกระแสน้อยมากประมาณ 1-20 mA
- มีอายุการใช้งานที่ยาวนาน ประมาณ 50,000 – 100,000 ชั่วโมง ขึ้นอยู่กับคุณภาพของแอลอีดี วงจรขับกระแส สภาพภูมิอากาศ ความชื้น และอุณหภูมิ ซึ่งก็มีอายุการใช้งานที่ยาวนานกว่าหลอดที่ให้แสงสว่างชนิดอื่นๆมาก
- ไม่มีรังสีอินฟราเรด รังสีอัลตราไวโอเลต ซึ่งเป็นอันตรายต่อผิวหนัง
- ทนทานต่อสภาวะอากาศ
- ทนทานต่อการสั่นสะเทือน
- มีหลากหลายสีให้เลือกใช้

2.5 มอเตอร์ไฟฟ้ากระแสตรง (DC MOTOR)

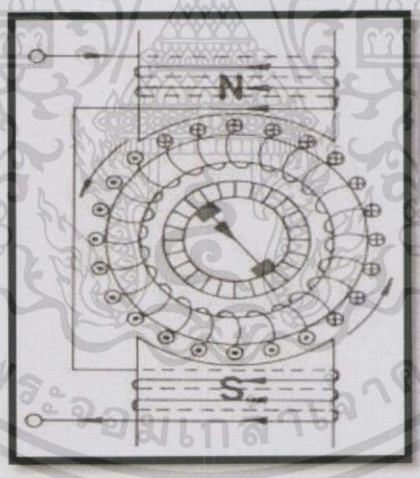
มอเตอร์ไฟฟ้า (MOTOR) หมายถึง เป็นเครื่องกลไฟฟ้าชนิดหนึ่งที่เปลี่ยนแปลงพลังงานไฟฟ้ามาเป็นพลังงานกล มอเตอร์ไฟฟ้าที่ใช้พลังงานไฟฟ้าเปลี่ยนเป็นพลังงานกลมีทั้งพลังงานไฟฟ้ากระแสสลับและพลังงานไฟฟ้ากระแสตรง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.6 แสดงภาพโครงสร้างเบื้องต้นของมอเตอร์กระแสตรง

มอเตอร์กระแสตรงจะมีหลักการทำงานโดยวิธีการผ่านกระแสให้กับขดลวดในสนามแม่เหล็ก ซึ่งจะทำให้เกิดแรงแม่เหล็ก โดยส่วนของแรงนี้จะขึ้นอยู่กับกระแสและกำลังของสนามแม่เหล็ก



รูปที่ 2.7 แสดงภาพทิศการหมุนของมอเตอร์

2.5.1 หลักการของมอเตอร์กระแสไฟฟ้าตรง (Motor Action)

หลักการของมอเตอร์ไฟฟ้ากระแสตรง (Motor Action) เมื่อเป็นแรงดันกระแสไฟฟ้าตรงเข้าไปในมอเตอร์ ส่วนหนึ่งจากแปรงถ่านผ่านคอมมิวเตเตอร์เข้าไปในขดลวดอาร์เมเจอร์สร้างสนามแม่เหล็กขึ้น และกระแสไฟฟ้าอีกส่วนหนึ่งจะไหลเข้าไปในขดลวดสนามแม่เหล็ก (Field coil) สร้างขั้วเหนือ-ใต้ขึ้น จะเกิดสนามแม่เหล็ก 2 สนาม ในขณะเดียวกัน ตามคุณสมบัติของเส้นแรงแม่เหล็ก จะไม่ตัดกันทิศทางตรงข้ามจะหักล้างกัน และทิศทางเดียวจะเสริมแรงกัน ทำให้เกิดแรงบิดในตัวอาร์เมเจอร์ ซึ่งวางแกนเพลาและแกนเพลานี้ สวมอยู่กับตลับลูกปืนของมอเตอร์ ทำให้อาร์เมเจอร์นี้หมุนได้ ขณะที่ตัวอาร์เมเจอร์ทำหน้าที่หมุนได้นี้เรียกว่า โรเตอร์ (Rotor) ซึ่งหมายความว่า

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตัวหมุน การที่อำนาจเส้นแรงแม่เหล็กทั้งสองมีปฏิกิริยาต่อกัน ทำให้ขดลวดอาร์เมเจอร์ หรือโรเตอร์หมุนไปนั้นเป็นไปตามกฎซ้ายของเฟลมมิ่ง (Fleming' left hand rule)

2.6 SolidWorks

SolidWorks พัฒนาขึ้นในปี 1995 โดยบริษัท Dassault System ในฝรั่งเศสเป็นซอฟต์แวร์เพื่อให้นักออกแบบใช้ เป็นเครื่องมือในการออกแบบทางวิศวกรรม เพื่อสร้างตัวอย่างผลิตภัณฑ์จำลองใน Computer ก่อนที่จะสร้างผลิตภัณฑ์ต้นแบบจริง โดยตัวซอฟต์แวร์จะจัดอยู่ในตระกูล CAD (Computer Aided Design) ซึ่งสามารถสร้างชิ้นงานจำลองในรูปแบบ 3D Solid Models เป็นแบบงานแยกชิ้น (Part) และแบบงานประกอบ (Assembly) เพื่อนำไปสร้างเป็น 2D Standard Engineering (CADD = Computer Aided Design and Drafting)

โปรแกรม Solidwork เป็นโปรแกรมที่มีความยืดหยุ่นในการทำงานสูงมาก คือ สามารถที่จะทำงานมากมายหลายรูปแบบ ไม่ว่าจะเป็นชิ้นงานที่ต้องขึ้นเป็น solid หรือ surface ก็มีเครื่องที่รองรับเป็นอย่างดี เมื่อสร้างชิ้นงานเสร็จเรียบร้อยสามารถที่จะประกอบชิ้นงานได้ใน Mode ของชุดคำสั่ง Assembly รวมทั้งผู้ต้องการ Drawing ของชิ้นงาน ก็เพียงลากชิ้นงานมาวางในใบงานแล้วขนาด จะมองเห็นได้ว่าผู้ใช้งาน สามารถที่จะประหยัดเวลาในการทำงาน

2.6.1 ประสิทธิภาพการทำงาน

ประสิทธิภาพ ของ SolidWorks เป็นการเจาะลึกให้นักออกแบบสามารถสร้างชิ้นงานจำลองทางด้าน Mechanical Engineering Design ได้อย่างสมบูรณ์แบบ นอกจากนี้ยังสามารถนำไปใช้ในการคำนวณทางวิศวกรรม และการตรวจสอบความผิดพลาดของ 3D Solid Models เพื่อลดต้นทุนในการผลิต และลดระยะเวลาการทำงานในการออกแบบ เพื่อเพิ่มประสิทธิภาพการทำงานในบริษัทและองค์กร

2.6.2 ลักษณะการทำงาน

SolidWorks แบ่งหมวดการทำงานหลักออกเป็น 3 หมวดคือ Part, Assembly และ Drawing โดยรูปแบบการทำงานทั้ง 3 หมวดมีลักษณะการใช้งานดังนี้

Part Mode เป็นหมวดการทำงานเริ่มต้นก่อนที่จะก้าวสู่การทำงานในหมวด Assembly และ Drawing ในขั้นนี้จะมีการแบ่งการทำงานออกเป็น 2 ส่วน คือ การใช้ 2D Sketch เพื่อนำไปสู่การสร้างเป็น 3D Feature และมีเงื่อนไขเป็น Feature-Based Modeling และ Parametric โดยมีการอ้างอิงจาก Solid Mode

1. Feature-Based Modeling คือ การออกแบบซอฟต์แวร์ให้สามารถทราบถึงคุณสมบัติต่างๆของ Solid Model ที่สร้างขึ้นมา เพื่อให้ผู้ใช้งานสามารถเปลี่ยนแปลงและแก้ไข Model ในลำดับการทำงานแต่ละขั้นได้ง่ายและรวดเร็ว

2. Parametric Model คือ การออกแบบซอฟต์แวร์ซึ่งใช้เงื่อนไขทางคณิตศาสตร์ในการแก้ไขขนาดรูปร่าง ทางเรขาคณิตของ Model ที่สร้างขึ้นมา

3. Solid Model คือ แบบจำลองบนคอมพิวเตอร์ที่สามารถแสดงค่าต่างๆ เช่น Density, Material, Mass, Weight เป็นต้น และยังสามารถมองเห็น 3D Model ได้ทุกมุมมอง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Assembly Mode เป็นหมวดการทำงานเพื่อนำ Part Model เข้าไปประกอบเป็นเครื่องจักรกลหรือกลไกต่างๆ และมีเงื่อนไขเป็น Feature Base และ Parametric เช่นเดียวกับ Part Model โดย Part Model และ Assembly จะมีความสัมพันธ์ซึ่งกันและกัน เมื่อทำการแก้ไขในหมวดใด อีก หรือมีการประกอบที่ซ้อนหรือทับกันหมวดจะมีการเปลี่ยนแปลงตามการแก้ไขไปด้วย การทำงานใน Assembly สามารถช่วยให้นักออกแบบหรือวิศวกรสามารถตรวจสอบความผิดพลาดในการสร้าง Part ได้โดยการใช้คำสั่งต่างๆ เช่น คำสั่ง Interference Detection เพื่อตรวจสอบการขัดกันเมื่อมีการเคลื่อนที่ โดยใช้คำสั่ง Move Component เพื่อตรวจสอบการเคลื่อนที่ของกลไก คำสั่ง Simulation เพื่อจำลองต้นกำลังในการทำงานจริงของเครื่องจักร หรือหากชิ้นงานจำลองที่ออกแบบมีข้อผิดพลาด ก็สามารถแก้ไข Part ใน Assembly ได้เลย ทำให้การออกแบบเป็นเรื่องง่าย

การทำงานใน Assembly Mode มีลักษณะการทำงาน 2 กรณีได้แก่

1. Bottom-Up Assembly คือ การนำ 3D Models ต่างๆที่สร้างเสร็จแล้วใน Part Mode ไปวางในหน้าต่าง Assembly เพื่อทำการประกอบ โดยการใช้คำสั่ง Mate หรือ Smart Mate ซึ่งวิธีนี้จะเหมาะสำหรับผู้ใช้ในระดับเริ่มต้นหรือขั้น Basic

2. Top-Down Assembly คือการสร้าง 2D Sketch เป็นโครงร่างระหว่างชิ้นส่วนต่าง ๆ ระหว่าง Part หรือการสร้าง Part ใน Assembly โดยให้มีขนาดและรูปร่างที่มีการอ้างอิงกับ Part อื่น ๆ ทั้งในส่วน Sketch และ Feature วิธีนี้เหมาะกับผู้ใช้ในระดับ Advance

Drawing Mode เป็นหมวดการทำงานเพื่อสร้าง 2D Standard Engineering โดยในหมวดนี้เป็นการสร้างมุมมองและกำหนดรายละเอียดตามระบบมาตรฐานต่าง ๆ โดยจะแบ่งการทำงานออกเป็น 2 ส่วนคือ

1. Generative Drafting ซึ่งเป็นการสร้าง 2D Sketch และ Interaction Drafting ซึ่งเป็นการนำ 3D Model จาก Part และ Assembly มาวางใน Drawing เพื่อสร้างเป็น 2D Drafting จะมีลักษณะเป็น Parametric และ Relation เช่นกัน แต่จะไม่สามารถใช้คำสั่งใน Drawing Commands ได้ เพราะคำสั่งต่าง ๆ จะต้องอ้างอิงกับ 3D Model

2. Interaction Drafting คือการนำ 3D Model จาก Part และ Assembly มาวาง Drawing เพื่อสร้างเป็น 2D Drafting การทำงานในหมวดนี้สามารถใช้คำสั่งจาก Annotation Command และ Drawing Command เพื่อสร้างมุมมองและกำหนดรายละเอียดได้โดยอัตโนมัติ

2.7 Altium

2.7.1 การใช้ Altium Designer

ความสามารถหลักๆครอบคลุมฟังก์ชั่นการออกแบบทางด้านอิเล็กทรอนิกส์ 6 ด้านดังนี้

- Board level design – การออกแบบ Printed circuit board (PCB)
- Programmable device – การจัดการอุปกรณ์โปรแกรมได้ ทั้ง embedded และ Field Programmable Gate Array (FPGA)
- FPGA/PCB integration – การประสานระหว่างการใช้งาน Field Programmable Gate
  Array (FPGA) และการออกแบบ Printed circuit board (PCB)
- Design to manufacture – การออกแบบให้สามารถนำไปเข้าสู่กระบวนการผลิตได้
- Library management – การจัดการไลบรารี

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

• Document management – การจัดการเอกสารทั้งหมดของชิ้นงาน

2.8 Zigbee : XBee-PRO โมดูลสื่อสารข้อมูลไร้สาย 2.4 GHz

2.8.1 โมดูล XBee-PRO

2.8.1.1 คุณสมบัติทางเทคนิค

2.8.1.1.1 คุณสมบัติโดยทั่วไป

- ความถี่ในการทำงาน: 2.4GHz
- สายอากาศ: มีสายอากาศแบบ Whip
- ระยะทำการในร่ม: สูงสุด 300 ฟุตหรือประมาณ 100 เมตร
- ระยะทำการกลางแจ้ง (แบบ line-of-sight): สูงสุดถึง 1 ไมล์หรือประมาณ 1,500 เมตร
- กำลังส่ง: 60mW (18dBm)
- ความไวในการรับสัญญาณ: -100 dBm (1% packet error rate) การทำงานของขาพอร์ต: สามารถกำหนดผ่านทางซอฟต์แวร์ X-CTU เพื่อให้ทำงานเป็น
  - อินพุตอนาล็อกสำหรับวงจรแปลงสัญญาณอนาล็อกเป็นดิจิตอลความละเอียด 10 บิต
  - อินพุตเอาต์พุตดิจิตอล
- ขนาด: 0.96 x 1.297 นิ้ว หรือ 2.438 x 3.294 เซนติเมตร ไฟเลี้ยง: 2.8 ถึง 3.4 V
- กระแสไฟฟ้า: เมื่อส่งข้อมูล 215mA, รับข้อมูล 55mA, น้อยกว่า 10A ในโหมดลดพลังงาน ที่ไฟเลี้ยง +3.3 V
- อุณหภูมิใช้งาน: -40 ถึง 85 ºc

2.8.1.1.2 คุณสมบัติด้านการสื่อสารข้อมูล

- สามารถทำงานเป็นอุปกรณ์มาสเตอร์ และ สเลฟได้
- อัตราถ่ายทอดข้อมูลผ่านคลื่นวิทยุ: 250,000 บิตต่อวินาที
- อัตราการถ่ายทอดข้อมูลอนุกรม (บอดเรต): 1,200 ถึง 115,200 บิตต่อวินาที
- รูปแบบโครงข่ายข้อมูลที่รองรับ: จุดต่อจุด (Point-to-point), จุดต่อหลายจุด (Point-to-multipoint) และเข้ากันได้กับอุปกรณ์ ตามมาตรฐานรหัส 802.15.4
- ทางเลือกแอดเดรส: PAN ID, ชอง (Channel) และแอดเดรส (Addresses) สำหรับแอดเดรสสามารถกำหนดรหัสแอดเดรสได้มากถึง 65,000 รหัส
- เทคโนโลยี ในการกระจายคลื่น: DSSS (Direct Sequence Spread Spectrum)
- รองรับการทำงานทั้งแบบ API และ AT command สามารถ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กำหนดได้ ผ่านทางซอฟต์แวร์ X-CTU

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 3
วิธีการดำเนินงาน

สำหรับปริญญานิพนธ์ฉบับนี้สามารถแบ่งส่วนการทำงานได้ 2 ส่วนคือ ทางด้านฮาร์ดแวร์และทางด้านซอฟต์แวร์

- ทางด้านฮาร์ดแวร์ ประกอบไปด้วย
  1. ฐาน
  2. มอเตอร์
  3. พาวเวอร์ซัพพลาย
  4. ฐานอะคริลิกรองวงจร LED Display
  5. วงจร LED Display
  6. Zigbee
- ทางด้านซอฟต์แวร์ ประกอบไปด้วย
  1. โปรแกรม Visual Basic 2010
  2. โปรแกรม Arduino 1.5.5
  3. โปรแกรม X - CTU

3.1 ฮาร์ดแวร์

สำหรับในส่วนนี้จะแบ่งเป็น 2 ส่วนใหญ่ คือ ส่วนของแมคคานิกส์ และส่วนของวงจร

3.1.1 ส่วนของแมคคานิกส์

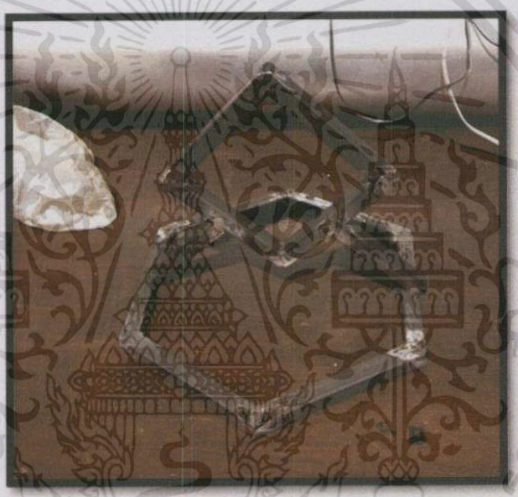
ประกอบไปด้วย 6 ส่วนหลัก คือ

1. ฐาน

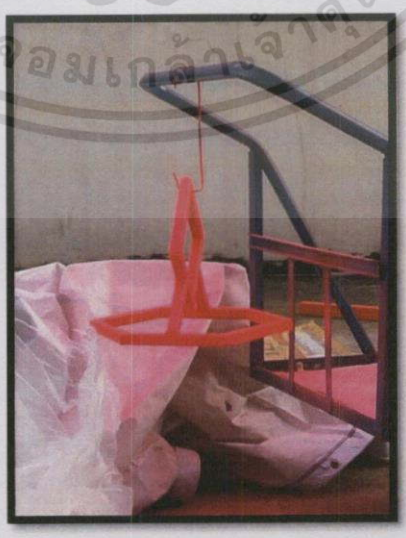
รูปที่ 3.1 แสดงภาพอุปกรณ์และแบบที่ใช้สร้างฐานของโครงงาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 3.2 แสดงภาพขั้นตอนการเชื่อมแท่งเหล็กเข้าด้วยกัน



รูปที่ 3.3 แสดงภาพฐานของโครงงาน

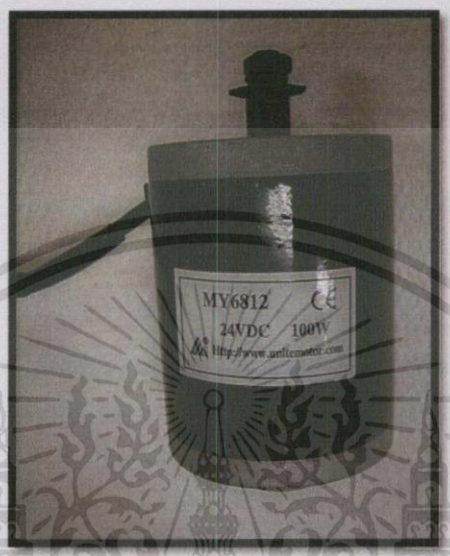


รูปที่ 3.4 แสดงภาพฐานของโครงงานเมื่อลงสี

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตัวฐานถูกออกแบบให้มีความมั่นคงแข็งแรง มีความสมดุล และมีลักษณะโดดเด่น สีสันสดใส โดยเป็นที่ยึดมอเตอร์ รองรับน้ำหนักทั้งหมดและแรงเหวี่ยงจากการหมุนของมอเตอร์ ดังรูปที่ 3.4

2. มอเตอร์

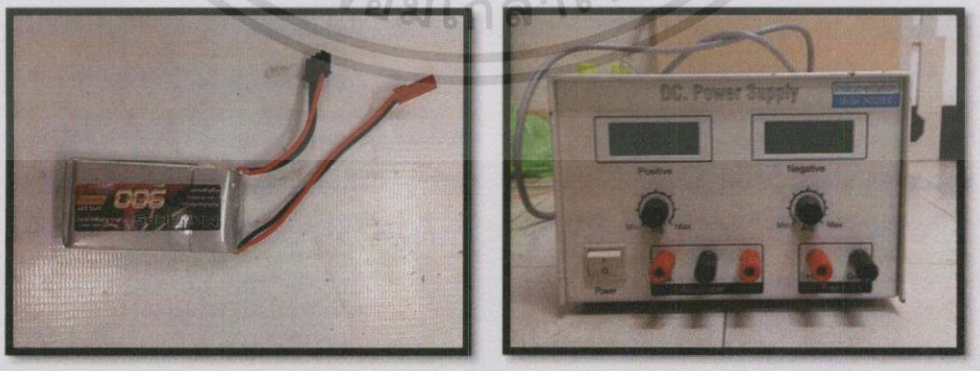


รูปที่ 3.5 แสดงภาพดีซีมอเตอร์

มอเตอร์ที่ใช้ในโครงงานนี้ เป็นดีซีมอเตอร์ 24VDC 100W 3,500rpm 4.16A หน้ากว้าง 6.5 เซนติเมตร ขนาดแกน 8 mm. ความยาวไม่รวมแกน 10 ซม. มีเฟืองโซ่ เบอร์ 25/ 11 ฟัน ติดพร้อมมอเตอร์ ดังรูปที่ 3.5

3. พาวเวอร์ซัพพลาย

พาวเวอร์ซัพพลายที่ใช้ในโครงงานนี้จะแยกเป็น 2 ตัวคือ พาวเวอร์ซัพพลายขนาด 24 VDC ใช้สำหรับขับมอเตอร์ และ 5 VDC ใช้สำหรับป้อนไฟให้กับบอร์ดไมโครคอนโทรลเลอร์ ดังรูปที่ 3.6

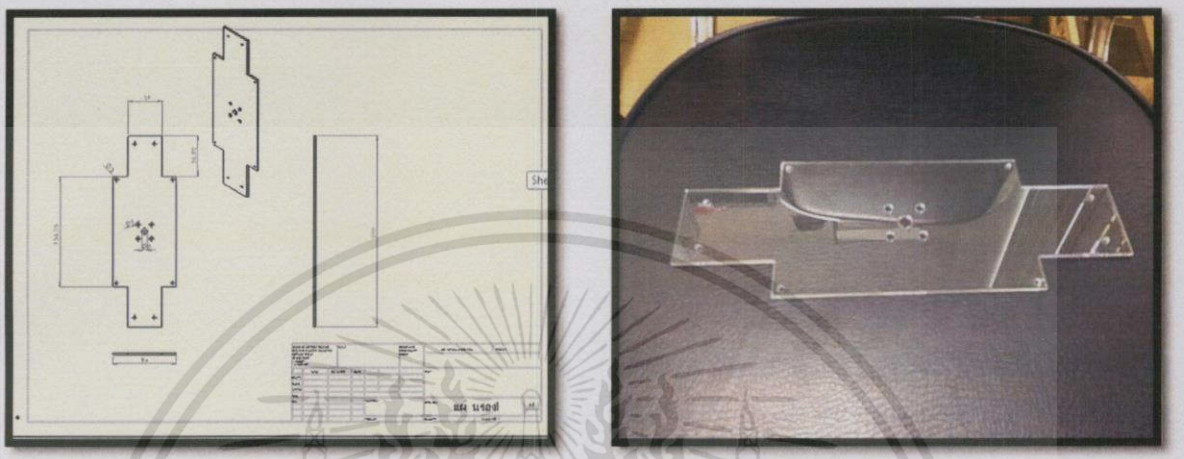


รูปที่ 3.6 แสดงภาพพาวเวอร์ซัพพลาย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4. ฐานอะคริลิกรองวงจร LED Display

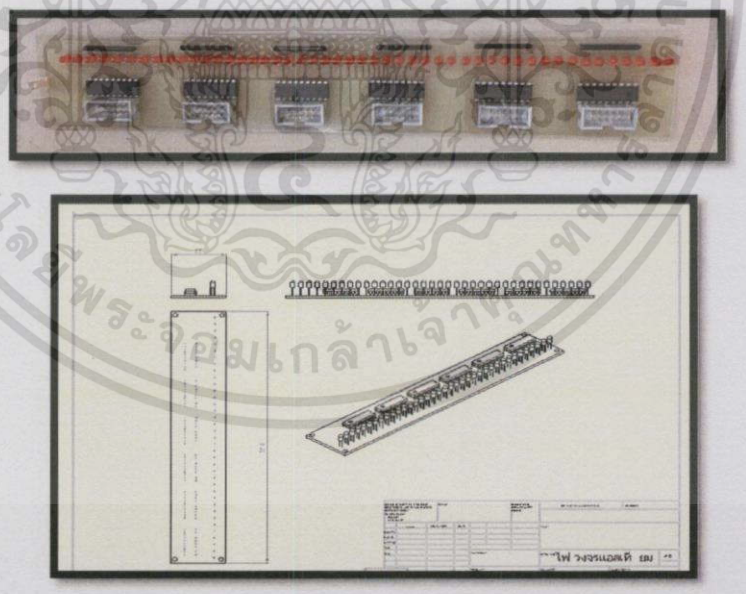
ฐานอะคริลิกรองบอร์ดและวงจร LED Display ซึ่งเลือกใช้วัสดุเป็นอะคริลิกใส เนื่องจากอะคริลิกนั้นมีน้ำหนักที่เบาเมื่อรวมกับน้ำหนักบอร์ดและน้ำหนักวงจร LED Display แล้วจะทำให้ไม่เกิดแรงเหวี่ยงและหนักจนเกินไป ดังรูปที่ 3.7



รูปที่ 3.7 แสดงภาพฐานรองบอร์ดและวงจร LED Display

5. วงจร LED Display

วงจร LED Display เป็นวงจรที่ใช้ในการแสดงผลของโครงงานนี้



รูปที่ 3.8 แสดงภาพ วงจร LED Display

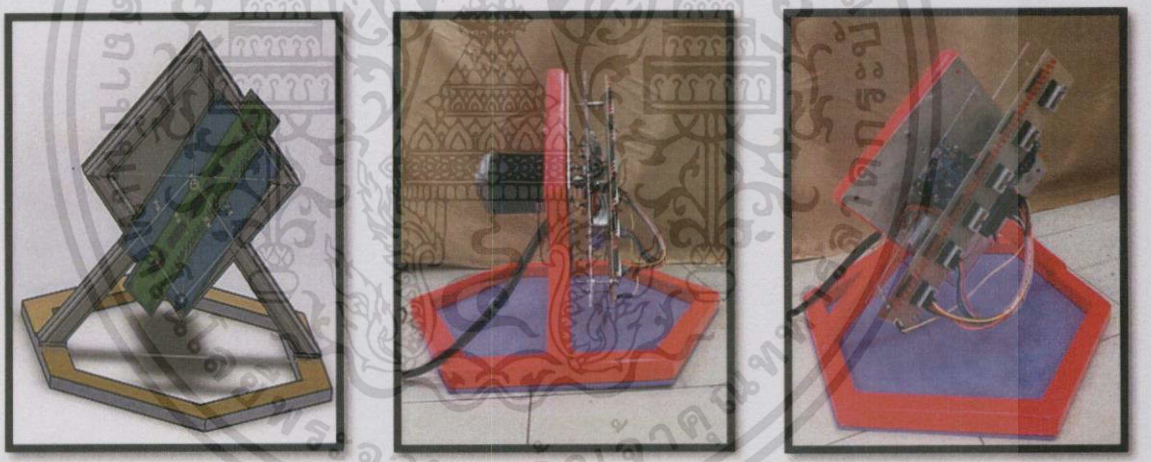
6. Zigbee

Zigbee ที่นำมาใช้เป็น XBee Pro Series 1 ซึ่งเป็น Bluetooth นำมาใช้แทนสาย USB เพื่อติดต่อกับบอร์ด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 3.9 แสดงภาพ Zigbee

เมื่อนำส่วนประกอบฮาร์ดแวร์ทุกส่วนมาประกอบกัน และทำการเดินสายไฟ จะได้ดังรูปที่ 3.10



รูปที่ 3.10 แสดงภาพชิ้นงานเมื่อประกอบเสร็จ

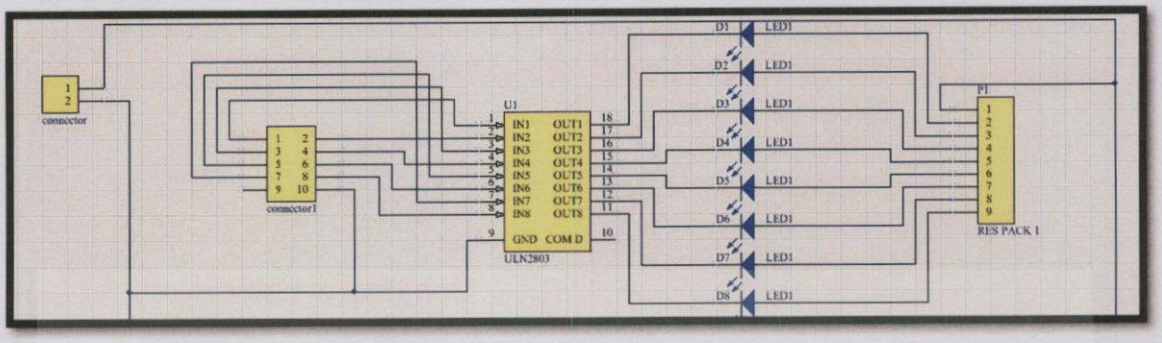
3.1.2 ส่วนของวงจร LED DISPLAY

ส่วนประกอบหลัก

1. ไอซี ULN2803A	6	ตัว
2. หลอด LED 3 mm.	48	หลอด
3. Connector Box 10 pin	6	ตัว
4. R pack 1 K	6	ตัว
5. สายไฟ		

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

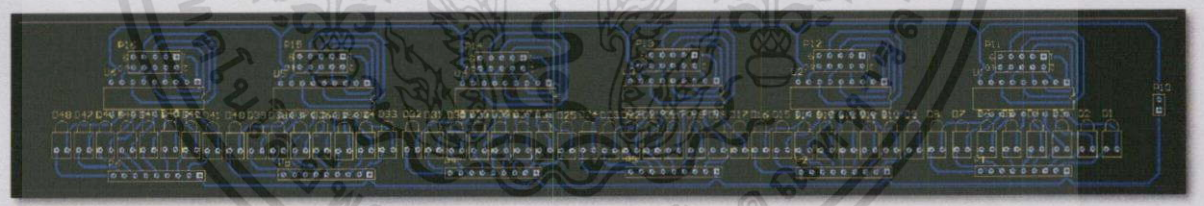
โดยวงจร LED DISPLAY มีหน้าที่แสดงภาพโดยรับคำสั่งจาก บอร์ด Arduino Due



รูปที่ 3.11 แสดงภาพ Schematic ของวงจร LED

หลักการทำงานของวงจร LED DISPLAY

เมื่อนำ Connector ต่อกับ บอร์ด Arduino โดย ขาที่ 1 ถึงขาที่ 8 ต่อกับ Port ของ บอร์ด Arduino Due เมื่อบอร์ด Arduino Due ซึ่งเป็นไมโครคอนโทรลเลอร์ ส่งข้อมูลเข้ามาที่ Connector จะเชื่อมกับอินพุตของ IC ULN2803 และเมื่อ IC ULN2803 ได้รับอินพุตเป็น Logic 1 ก็จะทำหน้าที่เชื่อมต่อขาเอาต์พุตกับกราวด์ ทำให้ไฟ VCC ไหลผ่าน R Pack ไปยังหลอดไฟ LED แล้วลงกราวด์ และเมื่อครบวงจรดังนี้แล้ว หลอดไฟ LED ก็จะติด ในทางกลับกัน ถ้า IC ULN2803 ได้รับอินพุตเป็น Logic 0 ก็จะทำให้ไฟไหลไม่ครบวงจร และหลอดไฟ LED จะไม่สามารถติดได้ หลอดไฟจึงดับ เราจึงใช้ บอร์ด Arduino Due เป็นไมโครคอนโทรลเลอร์ในการควบคุมการติดดับของหลอดไฟได้



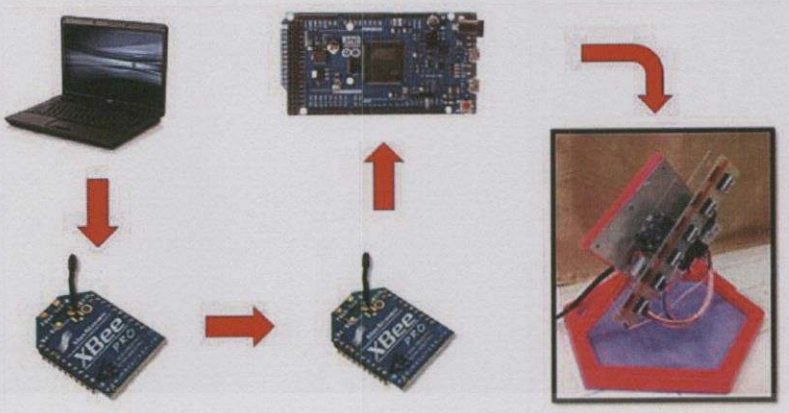
รูปที่ 3.12 แสดงภาพ PCB

โดยวงจร LED DISPLAY ทำการออกแบบวงจรโดยใช้โปรแกรม Altium Designer 6

3.2 ซอฟต์แวร์

ซอฟต์แวร์ในโครงงานฉบับนี้ประกอบด้วย 3 ส่วน คือ โปรแกรม Visual Basic 2010, โปรแกรม Arduino 1.5.5 และ โปรแกรม X CTU

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.13 แสดงภาพการทำงานของโปรแกรม

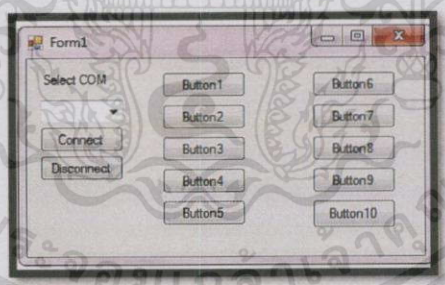
ขั้นตอนการทำงานของโปรแกรม

1. นำรูปภาพที่ต้องการ และแปลงรูปภาพให้เป็นค่า 0, 1
2. นำค่า 0, 1 มาทำการประมวลผลด้วย โปรแกรม Arduino 1.5.5 จากนั้นทำการส่งค่า จาก Zigbee (Sender) ไปยัง Zigbee(Receiver) แทนการใช้สายUSB ไปยังไมโครคอนโทรลเลอร์ บอร์ด Arduino Due
3. ไมโครคอนโทรลเลอร์ บอร์ด Arduino Due จะทำการส่งค่าที่ได้มายังแผงวงจร LED Display

3.2.1 ส่วนโปรแกรม Visual Basic 2010

- แบบแรก เป็นการส่งข้อมูลแบบทีละบิตสั่งให้ติดโดยการกดทีละปุ่ม ซึ่งเป็นแบบPattern

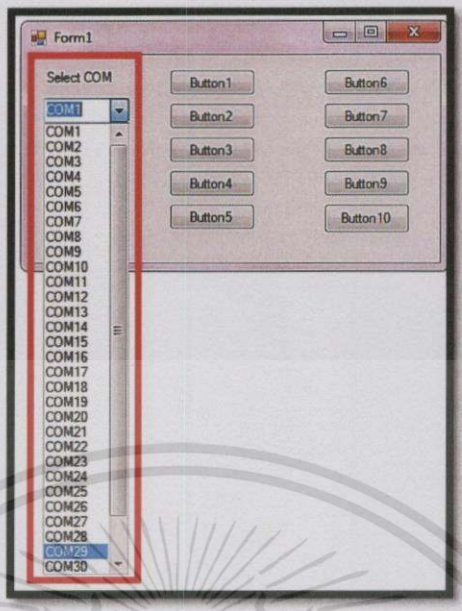
1. โดยเมื่อเปิดโปรแกรมขึ้น หน้าจอเริ่มต้น ก่อนเริ่มทำงาน ดังรูปที่ 3.14



รูปที่ 3.14 แสดงภาพหน้าจอเริ่มต้นโปรแกรม Visual Basic 2010 ก่อนเริ่มทำงาน

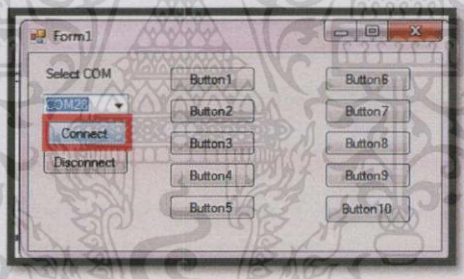
2. ทำการเลือก Comport เนื่องจากในคอมแต่ละเครื่อง Serial Comport จะไม่เหมือนกัน โครงงานเราจึงทำปุ่มที่สามารถเลือก Serial Comport ได้ จึงทำการเลือก Serial Comport ให้ตรงกับเครื่องคอมพิวเตอร์ โดยกดปุ่ม 'Select Com' ดังรูปที่ 3.15

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



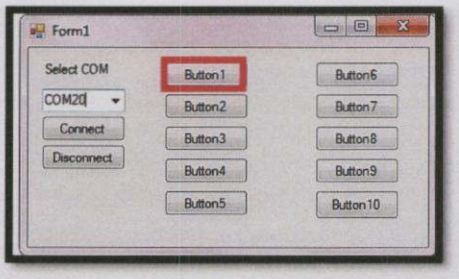
รูปที่ 3.15 แสดงภาพหน้าต่างโปรแกรม Visual Basic 2010 เลือก Serial Comport

3. เมื่อเลือก Comport เสร็จแล้ว จึงทำการเชื่อมต่อกับไมโครคอนโทรลเลอร์ โดยกดปุ่ม 'Connect' ดังรูปที่ 3.16



รูปที่ 3.16 แสดงภาพหน้าต่างโปรแกรม Visual Basic 2010 กดปุ่ม Connect

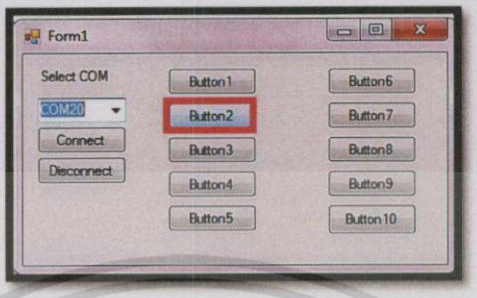
4. เมื่อทำการเชื่อมต่อกับไมโครคอนโทรลเลอร์แล้ว จึงทำการเลือกภาพที่จะให้แสดงผลออก โดยกดปุ่ม Button ซึ่งรูปตัวอย่างมีทั้งหมด 10 รูป ที่นำมาแสดง ถ้าต้องการรูปที่ 1 ซึ่งเป็นรูปใบพัด ให้ทำการกดปุ่ม 'Button1' ดังรูปที่ 3.17



รูปที่ 3.17 แสดงภาพหน้าต่างโปรแกรม Visual Basic 2010 กดปุ่ม Button 1

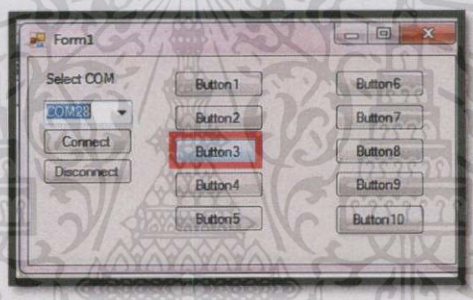
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5. ถ้าต้องการรูปที่ 2 ซึ่งเป็นรูปใบไม้ ให้ทำการกดปุ่ม ‘Button2’ ดังรูปที่ 3.18



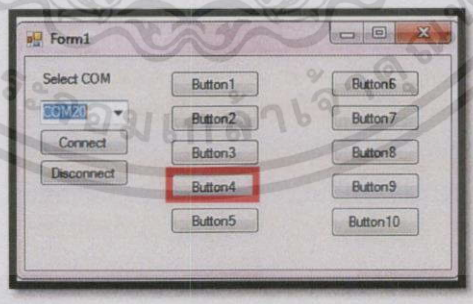
รูปที่ 3.18 แสดงภาพหน้าต่างโปรแกรม Visual Basic 2010 กดปุ่ม Button 2

6. ถ้าต้องการรูปที่ 3 ซึ่งเป็นรูป Encoder ให้ทำการกดปุ่ม ‘Button3’ ดังรูปที่ 3.19



รูปที่ 3.19 แสดงภาพหน้าต่างโปรแกรม Visual Basic 2010 กดปุ่ม Button 3

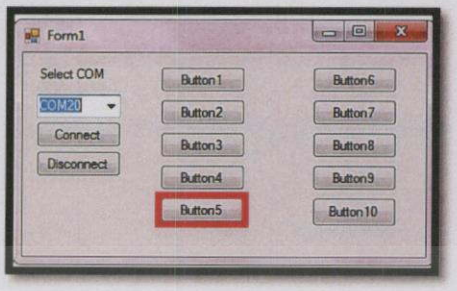
7. ถ้าต้องการรูปที่ 4 ซึ่งเป็นรูป หมากรุก ให้ทำการกดปุ่ม ‘Button4’ ดังรูปที่ 3.20



รูปที่ 3.20 แสดงภาพหน้าต่างโปรแกรม Visual Basic 2010 กดปุ่ม Button 4

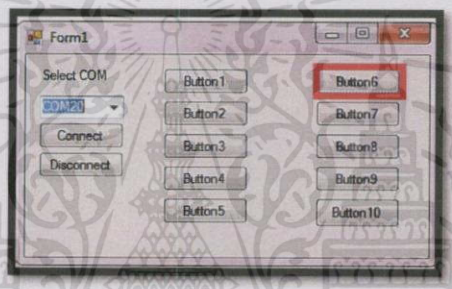
8. ถ้าต้องการรูปที่ 5 ซึ่งเป็นรูปตัวเลข 12 ให้ทำการกดปุ่ม ‘Button5’ ดังรูปที่ 3.21

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



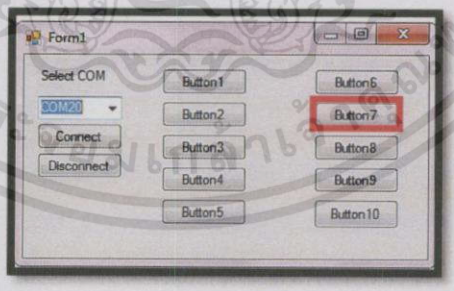
รูปที่ 3.21 แสดงภาพหน้าต่างโปรแกรม Visual Basic 2010 กดปุ่ม Button 5

9. ถ้าต้องการรูปที่ 6 ซึ่งเป็นรูปตัวอักษร P ให้ทำการกดปุ่ม 'Button6' ดังรูปที่ 3.22



รูปที่ 3.22 แสดงภาพหน้าต่างโปรแกรม Visual Basic 2010 กดปุ่ม Button 6

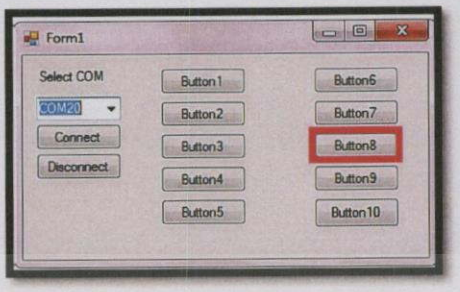
10. ถ้าต้องการรูปที่ 7 ซึ่งเป็นรูปสามเหลี่ยมคว่ำ ให้ทำการกดปุ่ม 'Button7' ดังรูปที่ 3.23



รูปที่ 3.23 แสดงภาพหน้าต่างโปรแกรม Visual Basic 2010 กดปุ่ม Button 7

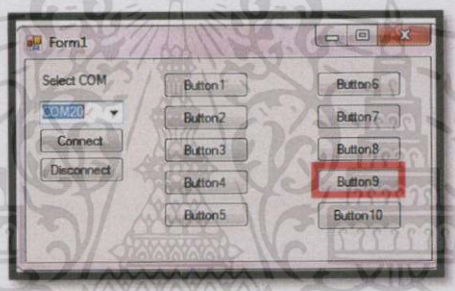
11. ถ้าต้องการรูปที่ 8 ซึ่งเป็นรูปหมวก ให้ทำการกดปุ่ม 'Button8' ดังรูปที่ 3.24

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



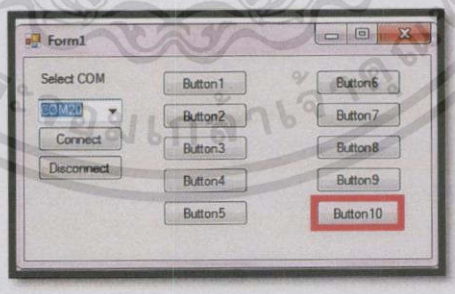
รูปที่ 3.24 แสดงภาพหน้าต่างโปรแกรม Visual Basic 2010 กดปุ่ม Button 8

12. ถ้าต้องการรูปที่ 9 ซึ่งเป็นรูปตัวหนังสือ Zine ให้ทำการกดปุ่ม 'Button9' ดังรูปที่ 3.25



รูปที่ 3.25 แสดงภาพหน้าต่างโปรแกรม Visual Basic 2010 กดปุ่ม Button 9

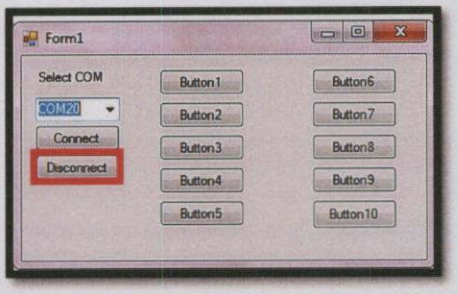
13. ถ้าต้องการรูปที่ 10 ซึ่งเป็นรูปหน้ายิ้ม ให้ทำการกดปุ่ม 'Button10' ดังรูปที่ 3.26



รูปที่ 3.26 แสดงภาพหน้าต่างโปรแกรม Visual Basic 2010 กดปุ่ม Button 10

14. เมื่อต้องการยกเลิกการเชื่อมต่อ ให้กดปุ่ม 'Disconnect' ดังรูปที่ 3.27

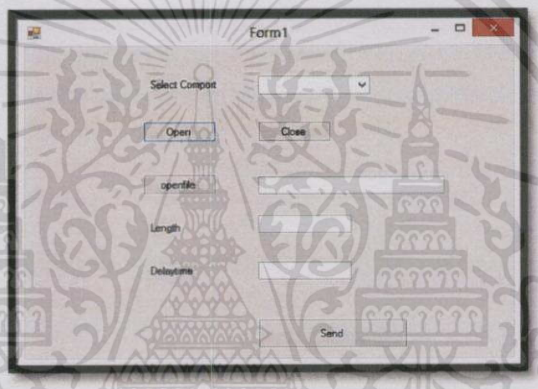
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.27 แสดงภาพหน้าต่างโปรแกรม Visual Basic 2010 กดปุ่ม Disconnect

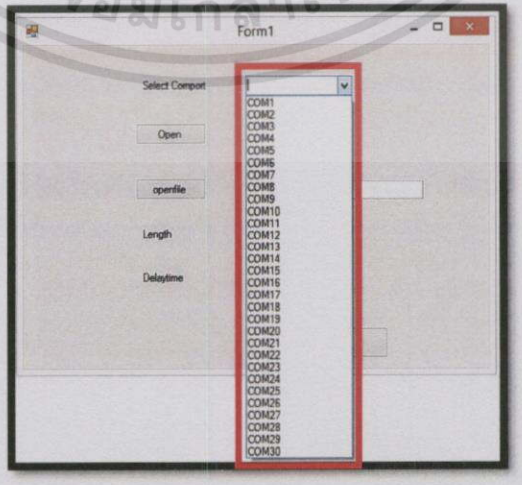
- แบบที่สอง เป็นการส่งข้อมูลแบบทีละบิต โดยใช้คำสั่ง Open file แล้วกดส่งข้อมูล

1. โดยเมื่อเปิดโปรแกรมขึ้น หน้าจอเริ่มต้น ก่อนเริ่มทำงาน ดังรูปที่ 3.28



รูปที่ 3.28 แสดงภาพหน้าจอเริ่มต้นโปรแกรม Visual Basic 2010 ก่อนเริ่มทำงาน

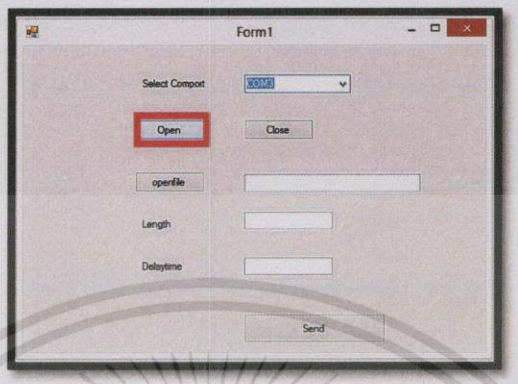
2. ทำการเลือก Comport เนื่องจากในคอมแต่ละเครื่อง Serial Comport จะไม่เหมือนกัน โครงงานเราจึงทำปุ่มที่สามารถเลือก Serial Comport ได้ จึงทำการเลือก Serial Comport ให้ตรงกับเครื่องคอมพิวเตอร์ โดยกดปุ่ม 'Select Com' ดังรูปที่ 3.29



รูปที่ 3.29 แสดงภาพหน้าต่างโปรแกรม Visual Basic 2010 เลือก Serial Comport

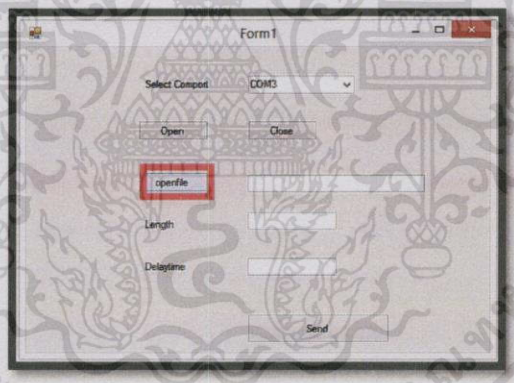
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3. เมื่อเลือก Serial Comport เสร็จแล้ว จึงทำการเชื่อมต่อกับไมโครคอนโทรลเลอร์ โดยกดปุ่ม ‘Open’ ดังรูปที่ 3.30

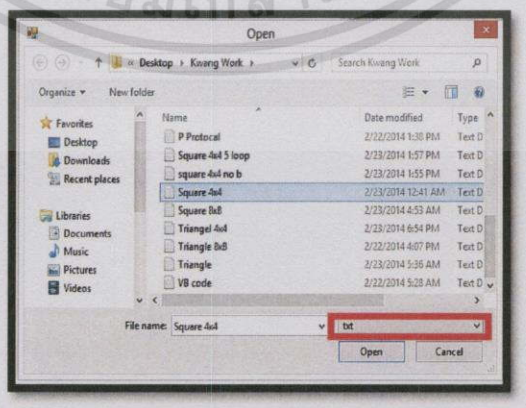


รูปที่ 3.30 แสดงภาพหน้าต่างโปรแกรม Visual Basic 2010 กดปุ่ม Open

4. เมื่อทำการเชื่อมต่อกับไมโครคอนโทรลเลอร์แล้ว จึงทำการเลือกไฟล์ ที่จะให้แสดงผลออกมา โดยกดปุ่ม Open file ดังรูปที่ 3.31 ซึ่งไฟล์ที่ใช้ในการแสดงผลเป็นไฟล์ .txt ดังรูปที่ 3.32



รูปที่ 3.31 แสดงหน้าต่างโปรแกรม Visual Basic 2010 กดปุ่ม Open file



รูปที่ 3.32 แสดงหน้าต่างโปรแกรม Visual Basic 2010 เมื่อกดปุ่ม Open file แล้วเลือกไฟล์ .txt ในการแสดงผล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5. เมื่อทำการ Open file เลือกไฟล์ที่ต้องการแล้ว หน้าต่างโปรแกรม Visual Basic 2010 ซึ่งจะมีช่อง length บอกขนาดความยาวของข้อมูลในไฟล์ที่เลือกมา จะเป็นดังรูปที่ 3.33

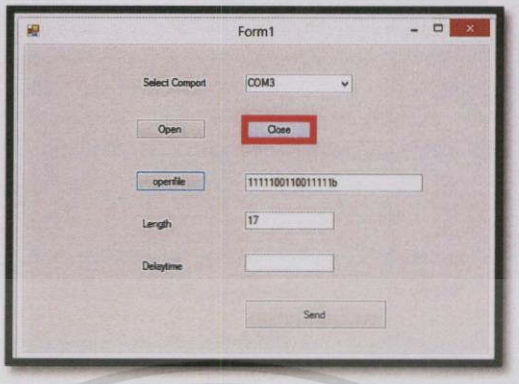
รูปที่ 3.33 แสดงหน้าต่างโปรแกรม Visual Basic 2010 เมื่อกดปุ่ม Open file เลือกไฟล์ที่ต้องการแล้ว

6. เมื่อเลือกข้อมูลที่ต้องการส่งเสร็จเรียบร้อยแล้วให้กดปุ่ม Send ไปยังไมโครคอนโทรลเลอร์เพื่อแสดงผลที่แผงวงจรไฟ LED เป็นดังรูปที่ 3.34

รูปที่ 3.34 แสดงหน้าต่างโปรแกรม Visual Basic 2010 เมื่อกดปุ่ม Send ข้อมูลที่ต้องการไปยังไมโครคอนโทรลเลอร์เพื่อแสดงผลที่แผงวงจรไฟ LED

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

7. เมื่อต้องการยกเลิกการเชื่อมต่อ ให้กดปุ่ม Close ดังรูปที่ 3.35

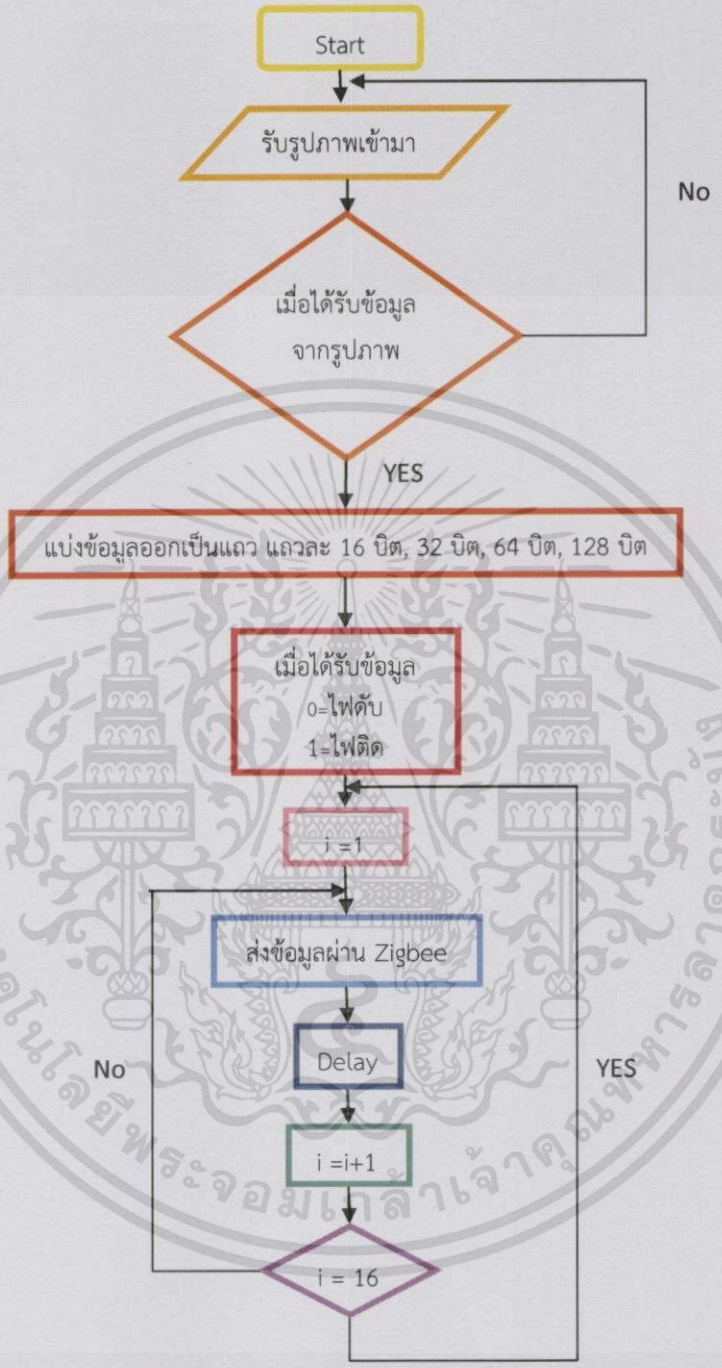


รูปที่ 3.35 แสดงภาพหน้าต่างโปรแกรม Visual Basic 2010 กดปุ่ม Close

3.2.2 ส่วนโปรแกรม Arduino 1.5.5

ก่อนจะออกแบบโปรแกรมและเขียนโปรแกรมที่ใช้ในการควบคุมแผงวงจร LED ติดดับของหลอด LED ได้นั้น ซึ่งต้องเข้าใจในส่วนของโปรแกรมทั้งหมดก่อน เมื่อทำความเข้าใจในแต่ละส่วนของโปรแกรมแล้ว จึงเริ่มคิดออกแบบโปรแกรมที่ใช้ในการควบคุม มีรูปแบบขั้นตอนการทำงาน ดังรูปที่ 3.36

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.36 แสดงภาพการออกแบบโปรแกรมที่ใช้ในการควบคุม

ซึ่งเมื่อได้ทำการออกแบบโปรแกรมได้แล้ว จึงนำมาออกแบบและเขียนลำดับการส่งข้อมูลของโปรแกรม ดังรูปที่ 3.37

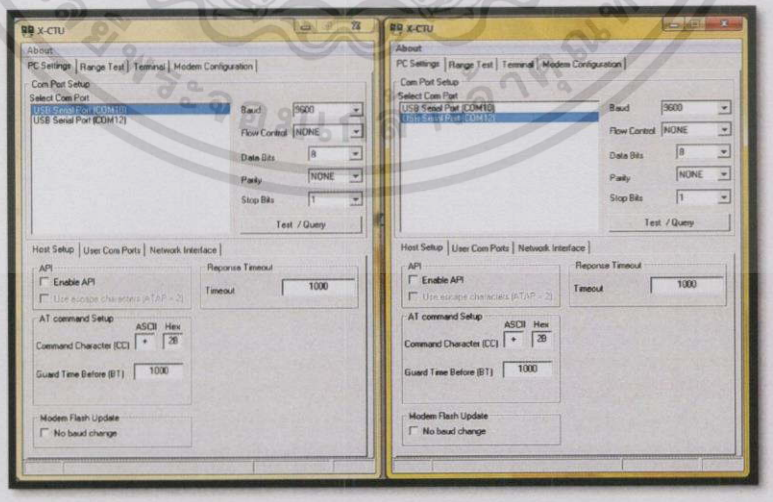
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 3.37 แสดงภาพลำดับการส่งข้อมูลของโปรแกรม

3.2.3 ส่วนโปรแกรม X CTU

ในการส่งข้อมูลนั้น คอมพิวเตอร์ต้องทำการเชื่อมต่อกับบอร์ดโดยผ่านสาย USB แต่เนื่องจากบอร์ดที่หมุนไปพร้อมกับวงจร LED ที่แสดงผล ทำให้ไม่สามารถที่จะส่งข้อมูลผ่านสาย USB ได้ จึงใช้ Zigbee เป็นตัวส่งข้อมูลแทนสาย USB ซึ่งในการส่งข้อมูลได้นั้น ต้องทำการ Modem Configuration ของตัว Zigbee ก่อน จากนั้นจะทำให้สามารถเห็นถึงการส่งและรับข้อมูลได้ โดยใช้โปรแกรม X CTU

การทำ Modem Configuration เริ่มจาก กดที่ PC Setting แล้วทำการเลือก Comport เมื่อเลือกเสร็จเรียบร้อย ทำการกดปุ่ม Test / Query ดังรูปที่ 3.38

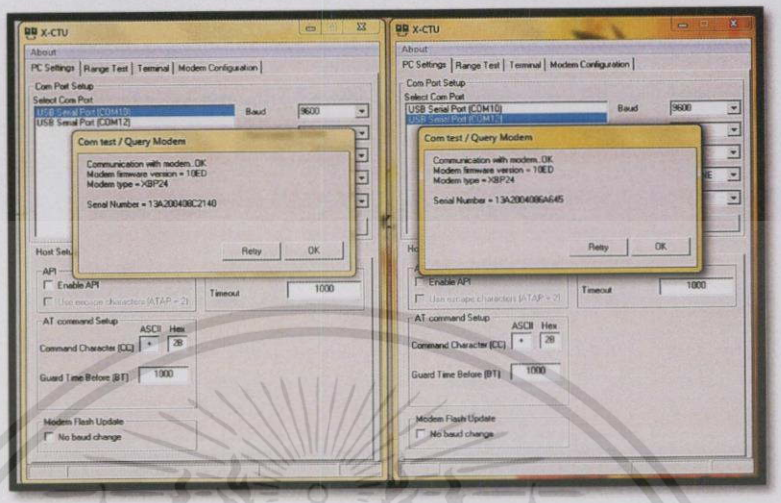


รูปที่ 3.38 แสดงภาพการทำ PC Setting

เนื่องจาก Zigbee ที่ใช้งาน มี 2 ตัวซึ่งตัวแรก ต่อกับคอมพิวเตอร์ และอีกตัวนั้นติดกับ

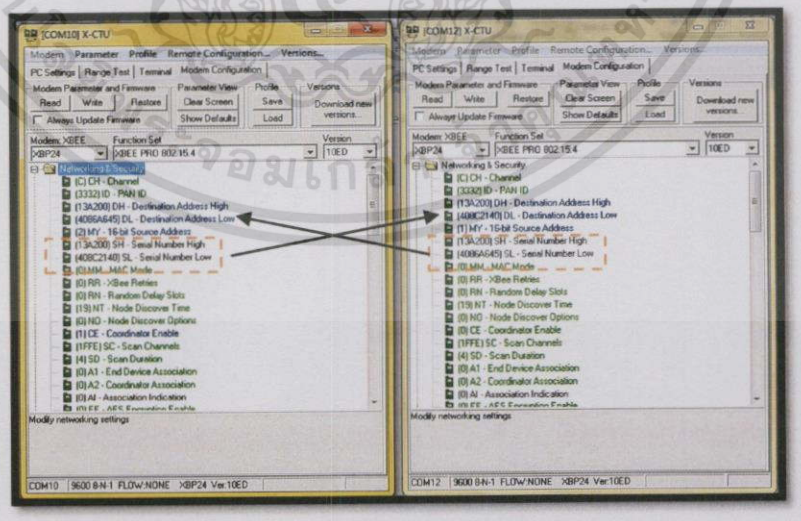
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ชิ้นงาน จึงทำดังรูปที่ 3.38 เมื่อกดปุ่ม Test / Query จะปรากฏหน้าจอ ให้กดปุ่ม OK ดังรูปที่ 3.39 ซึ่ง Zigbee ในแต่ละตัวจะมีค่าที่ไม่เหมือนกัน



รูปที่ 3.39 แสดงภาพเมื่อกด Test / Query

จากนั้นคลิกไปที่ Modem Configuration แล้ว ไปในช่อง Modem Parameter and Firmware แล้วทำการกดปุ่ม Read และนำค่า SH,SL ไปใส่ในค่า DH,DL ของอีกตัวหนึ่ง ให้ตรงกัน ทำการ set parameter โดยให้ฝั่งหนึ่งเป็น End Device ด้วยการ Set Parameter ในหมวด Networking & Security ให้ CE=0 และ ค่า MY = 1 และ ฝั่งอีกหนึ่ง เป็น Coordinator ด้วยการ Set Parameter ในหมวด Networking & Security ให้ CE=1 และ ค่า MY = 2 ทั้งนี้ได้ set baud rate ที่ตัว Xbee ทั้ง 2 ตัวที่ 9600 bps (BD=3) ดังรูปที่ 3.40



รูปที่ 3.40 แสดงภาพการใส่ค่า DH,DL

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากนั้นเข้าไปที่ Terminal ทำการลองส่งและรับค่า สลับกัน เป็นการเสร็จเรียบร้อย ดังรูปที่ 3.41

รูปที่ 3.41 แสดงภาพการส่งรับค่าสลับกัน

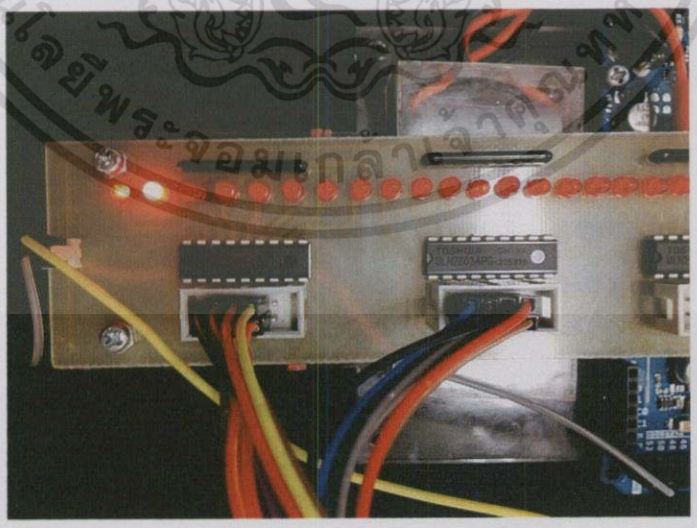
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 4
ผลการทดลอง

4.1 การทดสอบโปรแกรมกับบอร์ดไมโครคอนโทรลเลอร์ Arduino Due โดยการสั่งให้กดปุ่มไหนให้หลอดไฟติดเท่ากับเลขปุ่ม

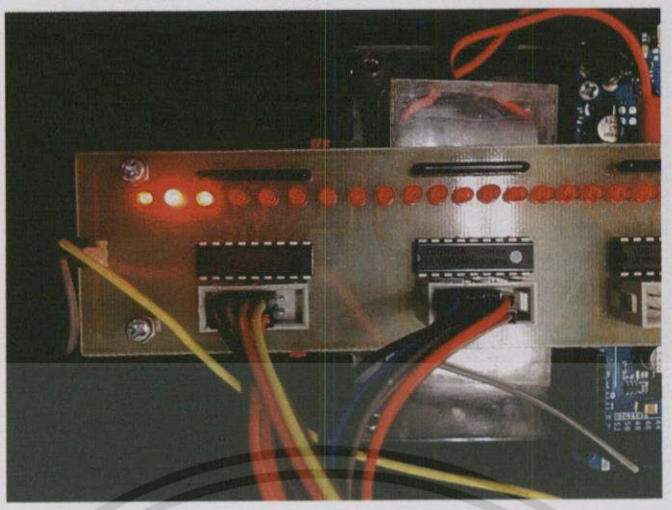
เป็นการสั่งกดปุ่มไหนให้หลอดไฟติดเท่ากับเลขปุ่ม เช่น เมื่อกดปุ่ม 'Button1' หลอดไฟจะติด 1 ดวง, กดปุ่ม 'Button2' หลอดไฟจะติด 2 ดวง เป็นต้น ดังรูป 4.1, รูป4.2, รูป 4.3 และรูป 4.4

รูปที่ 4.1 แสดงภาพการทดสอบโปรแกรม การสั่งให้กดปุ่ม 'Button1'

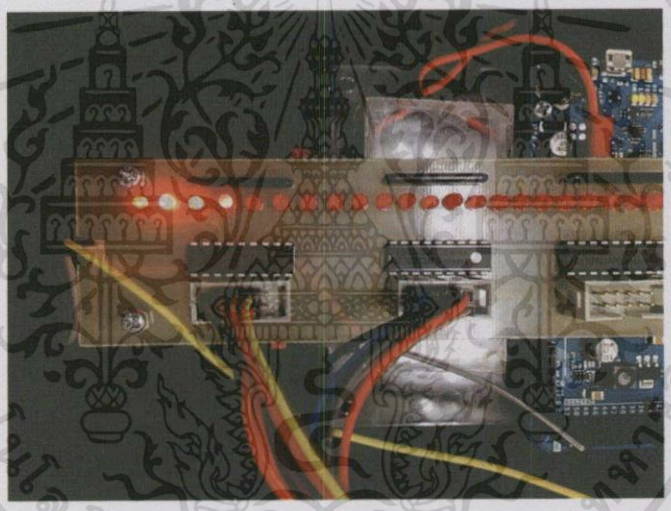


รูปที่ 4.2 แสดงภาพการทดสอบโปรแกรม การสั่งให้กดปุ่ม 'Button2'

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.3 แสดงภาพการทดสอบโปรแกรม การสั่งให้กดปุ่ม 'Button3'



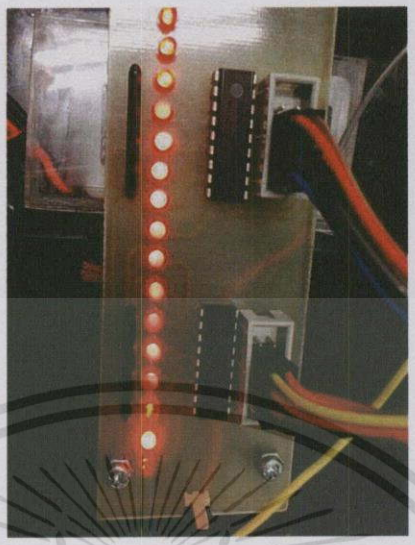
รูปที่ 4.4 แสดงภาพการทดสอบโปรแกรม การสั่งให้กดปุ่ม 'Button4'

โดยในการทดลองนี้จะทำการเขียนโปรแกรมให้หลอดไฟติดเรียงหลอด โดยจะทดสอบกับบอร์ดไมโครคอนโทรลเลอร์

4.2 การทดสอบโปรแกรมกับบอร์ดไมโครคอนโทรลเลอร์ Arduino Due โดยสั่งให้หลอดไฟติดหมดทุกพอร์ต

ในการทดลองนี้จะทำการเขียนโปรแกรม โดยสั่งให้หลอดไฟติดหมดทุกพอร์ตทั้งหมด 16 ดวง ดังรูป 4.5

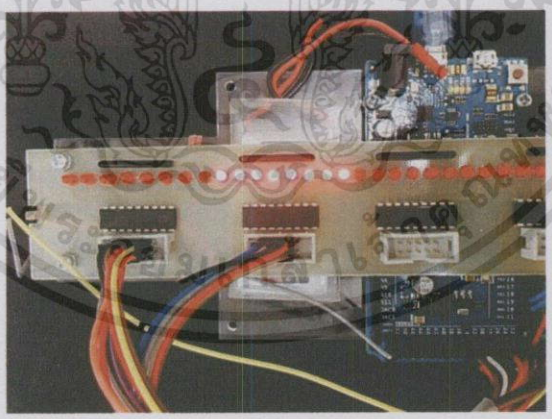
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.5 แสดงภาพการทดสอบโปรแกรมสั่งให้ติดทุกพอร์ต

4.3 การทดสอบโปรแกรมกับบอร์ดไมโครคอนโทรลเลอร์ Arduino Due โดยสั่งให้หลอดไฟติดทีละพอร์ต

ในการทดลองนี้จะทำการเขียนโปรแกรม โดยสั่งให้หลอดไฟติดทีละพอร์ต ซึ่งมี 2 พอร์ต พอร์ตละ 8 หลอด ดังรูป 4.6

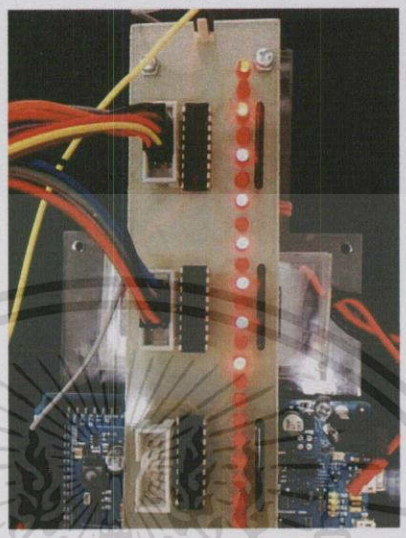


รูปที่ 4.6 แสดงภาพการทดสอบโปรแกรมสั่งให้ติดทีละพอร์ต

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.4 การทดสอบโปรแกรมกับบอร์ดไมโครคอนโทรลเลอร์ Arduino Due โดยสั่งให้หลอดไฟติดสลับดวง

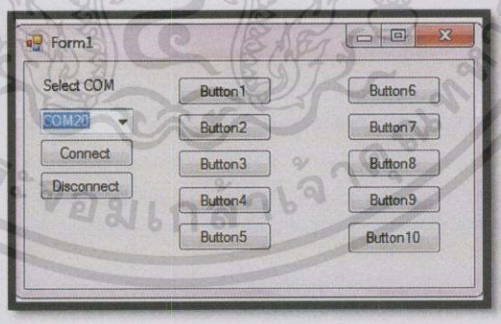
ในการทดลองนี้จะทำการเขียนโปรแกรม โดยสั่งให้หลอดไฟติดสลับดวง ดังรูป 4.7



รูปที่ 4.7 แสดงภาพการทดสอบโปรแกรมสั่งให้ติดติดสลับดวง

4.5 การทดสอบโปรแกรม โดยกดปุ่มในโปรแกรม Visual Basic 2010 จะส่งข้อมูลไปยังไมโครคอนโทรลเลอร์ เพื่อสั่งงานวงจร LED Display ให้แสดงรูป

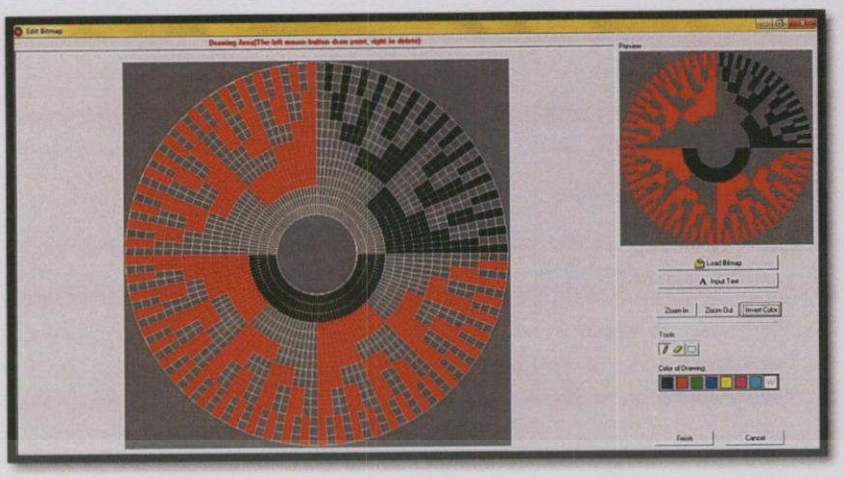
ขั้นแรก เปิดหน้าเริ่มโปรแกรม Visual Basic 2010 ขึ้นมา จากนั้น เลือก Comport แล้วทำการเชื่อมต่อไมโครคอนโทรลเลอร์ ดังรูป 4.8



รูปที่ 4.8 แสดงภาพการเลือก Comport แล้วทำการเชื่อมต่อไมโครคอนโทรลเลอร์

ขั้นตอนต่อไป เมื่อกดปุ่ม ‘Button1’ ถึง ‘Button10’ ก็จะทำการส่งข้อมูลไปยังไมโครคอนโทรลเลอร์ ที่เขียนโปรแกรมเชื่อมกับปุ่มไว้อยู่แล้ว เมื่อไมโครคอนโทรลเลอร์ได้ข้อมูลมาแล้วจะสั่ง วงจร LED Display ให้แสดงผลเป็นรูปได้ตามที่กดปุ่มไว้ และจะจ่ายไฟให้มอเตอร์หมุน เช่น กดปุ่ม ‘Button3’ ซึ่งเป็นรูป Encoder ดังรูป 4.9 และ ดังรูป 4.10

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



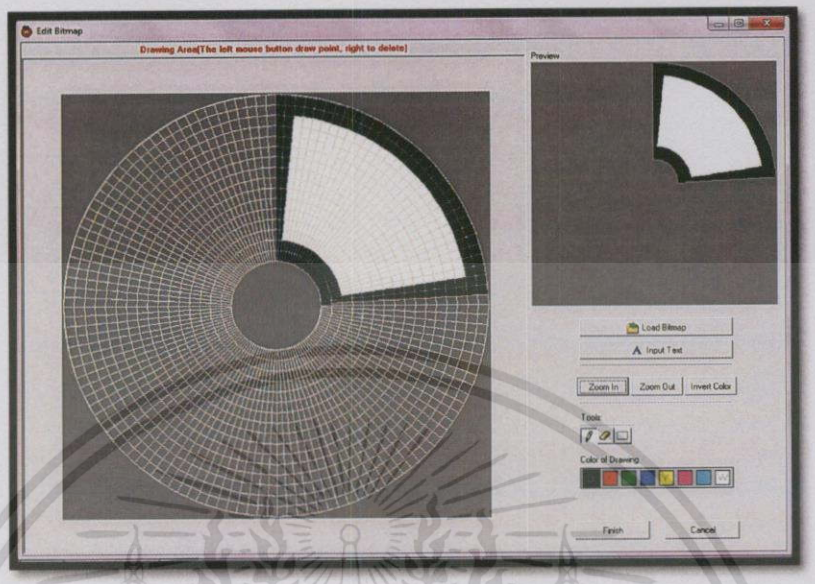
รูปที่ 4.9 แสดงภาพรูป Encoder โดยการใช้โปรแกรม Edit Bitmap

รูปที่ 4.10 แสดงภาพการแสดงผลของวงจร LED Display รูป Encoder เมื่อจ่ายไฟมอเตอร์หมุน

เมื่อเราจ่ายไฟให้มอเตอร์แล้วและจ่ายไฟให้กับวงจร LED Display เมื่อให้ไฟติดเป็นรูป Encoder ก็จะหมุนไปด้วยความเร็วประมาณ 3500 รอบ ต่อ นาที จะทำให้เห็นเป็นรูป Encoder

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.5.1 การทดลองกดปุ่ม Button 1 รูปใบพัด



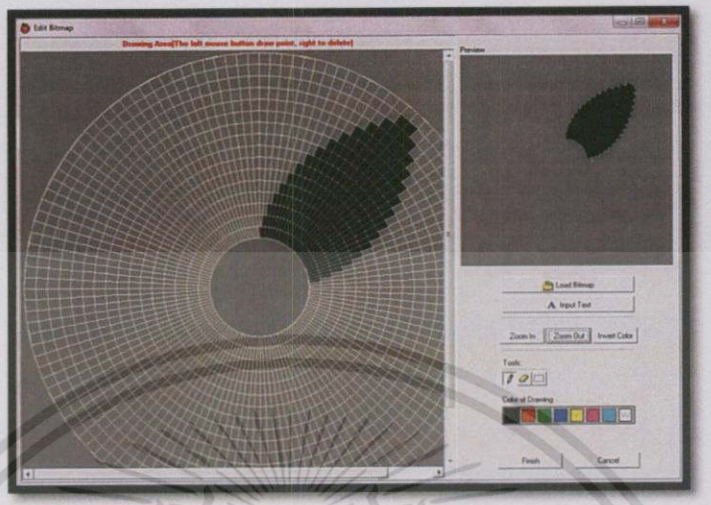
รูปที่ 4.11 แสดงภาพรูป ใบพัด โดยการใช้โปรแกรม Edit Bitmap

รูปที่ 4.12 แสดงภาพ การแสดงผลของวงจร LED Display ของรูปใบพัด

จากรูปที่ 4.12 การแสดงผลของ LED Display จะแสดงไปเป็นรูปใบพัดได้ตามข้อมูลภาพที่ป้อนไป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.5.2 การทดลองกดปุ่ม Button 2 รูปใบไม้



รูปที่ 4.13 แสดงภาพ รูปใบไม้ โดยการใช้โปรแกรม Edit Bitmap

รูปที่ 4.14 แสดงภาพ การแสดงผลของวงจร LED Display ของรูปใบไม้

จากรูปที่ 4.14 การแสดงผลของ LED Display จะแสดงไปเป็นรูปใบไม้ได้ตามข้อมูลภาพที่ป้อนไป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.5.3 การทดลองกดปุ่ม Button 4 รูปหมากรุก

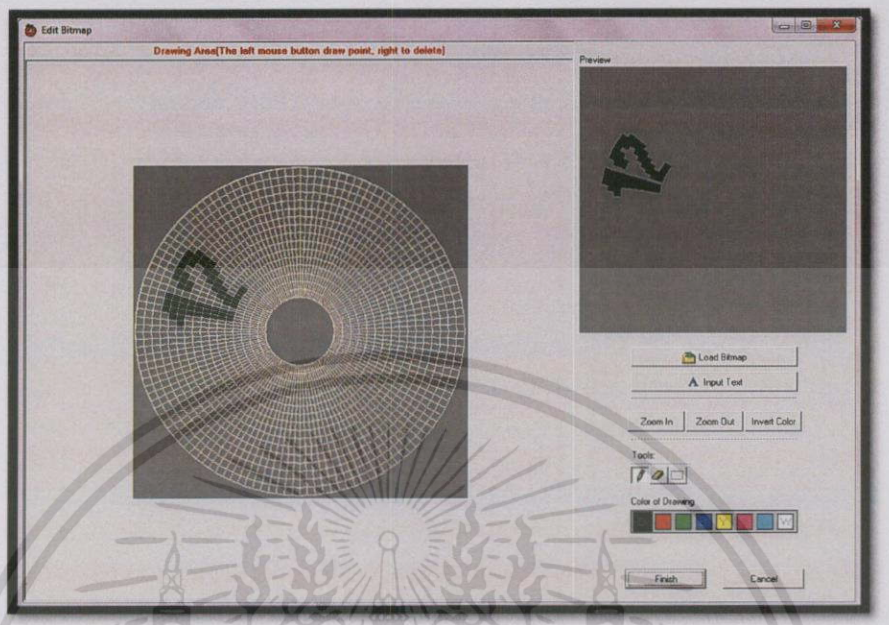
รูปที่ 4.15 แสดงภาพ รูปหมากรุก โดยการใช้โปรแกรม Edit Bitmap

รูปที่ 4.16 แสดงภาพ การแสดงผลของวงจร LED Display ของรูปหมากรุก

จากรูปที่ 4.16 การแสดงผลของ LED Display จะแสดงไปเป็นรูปหมากรุกได้ตามข้อมูลภาพที่ป้อนไป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.5.4 การทดลองกดปุ่ม Button 5 รูปตัวเลข 12



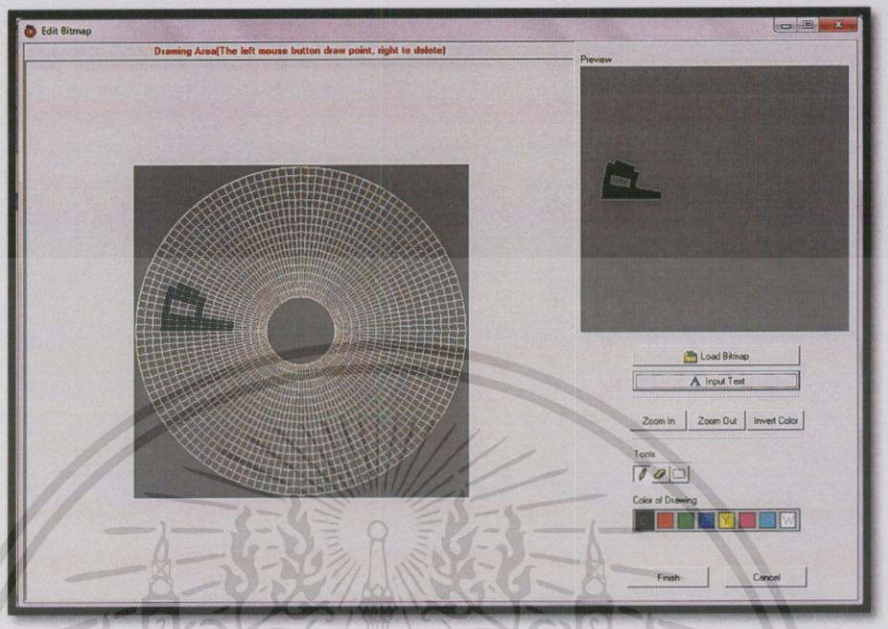
รูปที่ 4.17 แสดงภาพ รูปตัวเลข 12 โดยการใช้โปรแกรม Edit Bitmap

รูปที่ 4.18 แสดงภาพ การแสดงผลของวงจร LED Display ของรูปตัวเลข 12

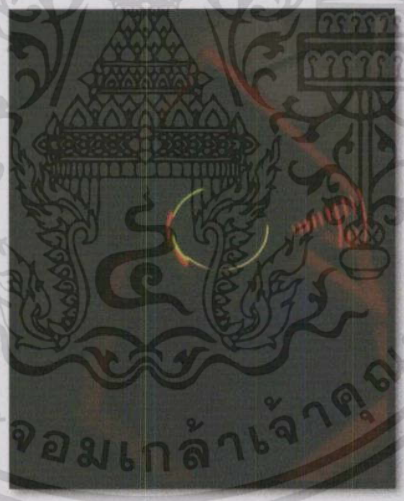
จากรูปที่ 4.18 การแสดงผลของ LED Display จะแสดงไปเป็นรูปตัวเลข 12 ได้ตามข้อมูลภาพที่ป้อนไป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.5.5 การทดลองกดปุ่ม Button 6 รูปตัวอักษร P



รูปที่ 4.19 แสดงภาพ รูปตัวอักษร P โดยการใช้โปรแกรม Edit Bitmap

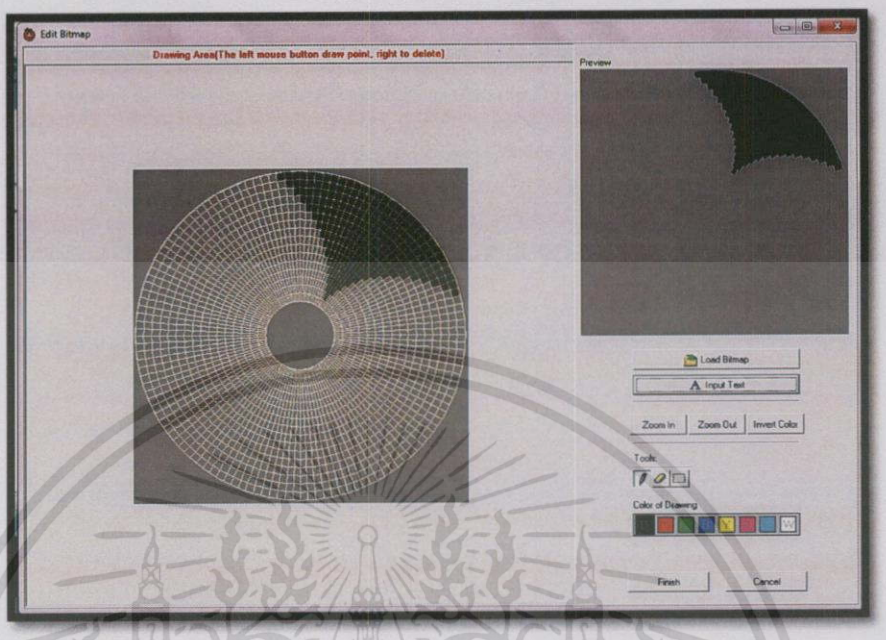


รูปที่ 4.20 แสดงภาพ การแสดงผลของวงจร LED Display ของรูปตัวอักษร P

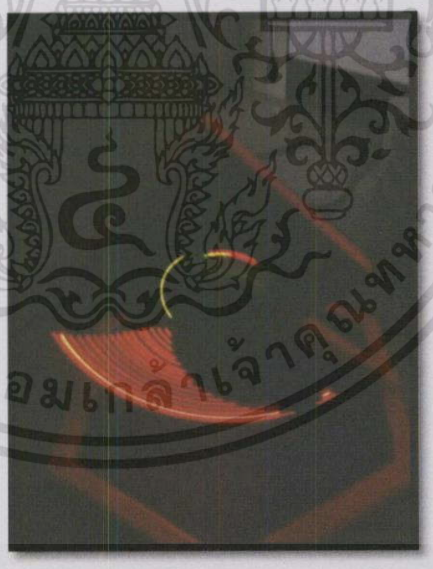
จากรูปที่ 4.20 การแสดงผลของ LED Display จะแสดงไปเป็นรูปตัวอักษร P ได้ตามข้อมูลภาพที่ป้อนไป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.5.6 การทดลองกดปุ่ม Button 7 รูปตัวสามเหลี่ยมคว่ำ



รูปที่ 4.21 แสดงภาพ รูปสามเหลี่ยมคว่ำ โดยการใช้โปรแกรม Edit Bitmap



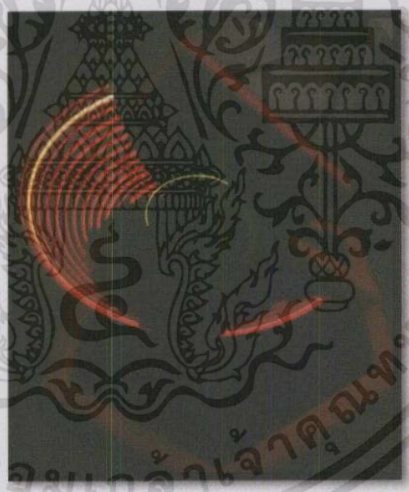
รูปที่ 4.22 แสดงภาพ การแสดงผลของวงจร LED Display ของรูปสามเหลี่ยมคว่ำ

จากรูปที่ 4.22 การแสดงผลของ LED Display จะแสดงไปเป็นรูปสามเหลี่ยมคว่ำ ได้ตามข้อมูลภาพที่ป้อนไป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.5.7 การทดลองกดปุ่ม Button 8 รูปหมวก

รูปที่ 4.23 แสดงภาพรูปหมวก โดยการใช้โปรแกรม Edit Bitmap

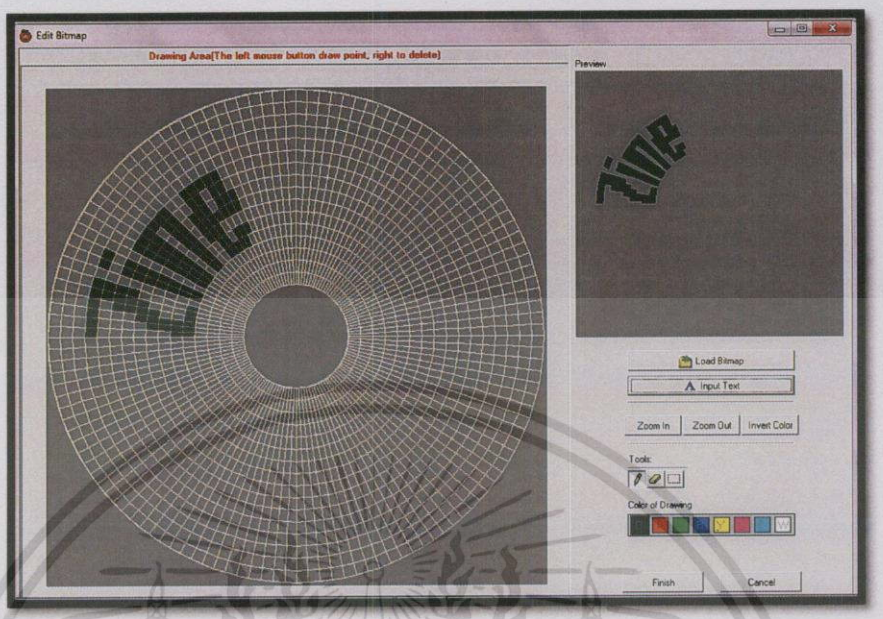


รูปที่ 4.24 แสดงภาพ การแสดงผลของวงจร LED Display ของรูปหมวก

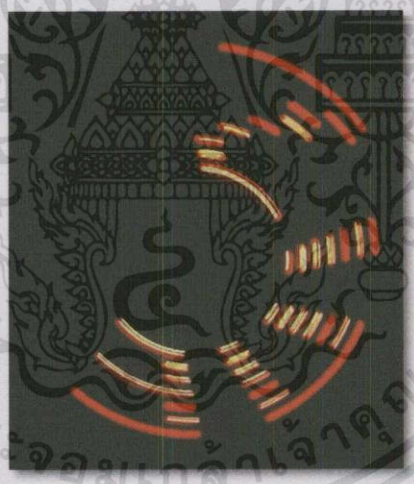
จากรูปที่ 4.24 การแสดงผลของ LED Display จะแสดงไปเป็นรูปหมวกได้ตามข้อมูลภาพที่ป้อนไป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.5.8 การทดลองกดปุ่ม Button 9 รูปตัวหนังสือ Zine



รูปที่ 4.25 แสดงภาพ รูปตัวหนังสือ Zine โดยการใช้โปรแกรม Edit Bitmap

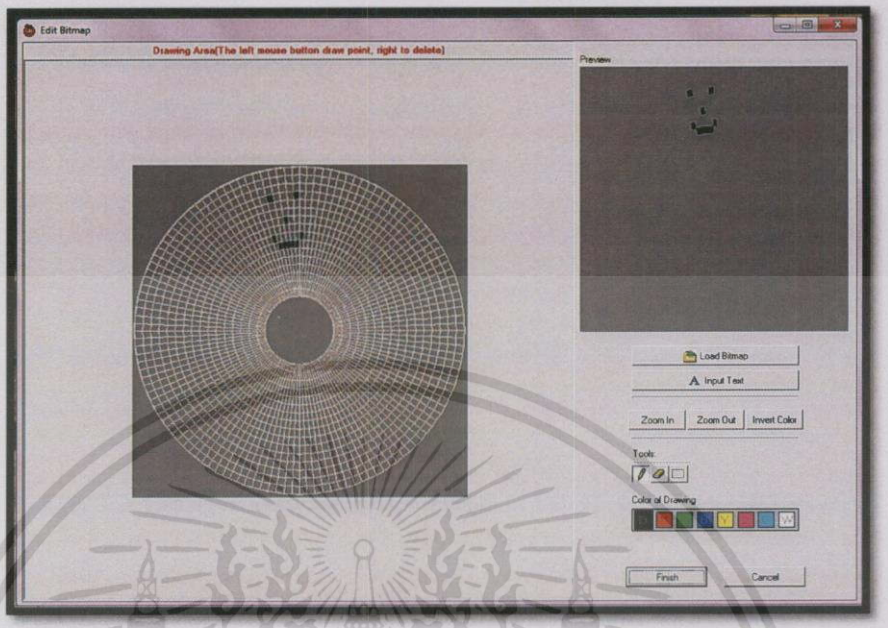


รูปที่ 4.26 แสดงภาพ การแสดงผลของวงจร LED Display ของรูปตัวหนังสือ Zine

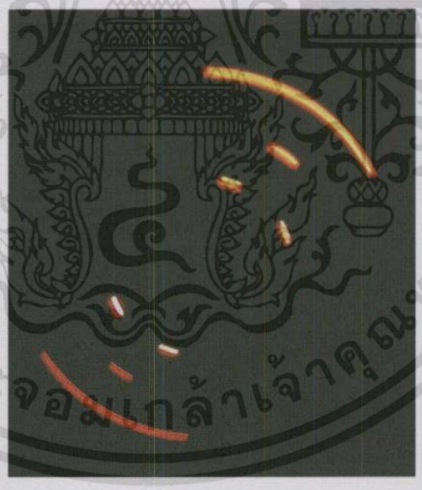
จากรูปที่ 4.26 การแสดงผลของ LED Display จะแสดงไปเป็นรูปตัวหนังสือ Zine ได้ตามข้อมูลภาพที่ป้อนไป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.5.9 การทดลองกดปุ่ม Button 10 รูปหน้ายิ้ม



รูปที่ 4.27 แสดงภาพรูปหน้ายิ้มโดยการใช้โปรแกรม Edit Bitmap



รูปที่ 4.28 แสดงภาพ การแสดงผลของวงจร LED Display ของรูปหน้ายิ้ม

จากรูปที่ 4.28 การแสดงผลของ LED Display จะแสดงไปเป็นรูปหน้ายิ้มได้ตามข้อมูลภาพที่ป้อนไป

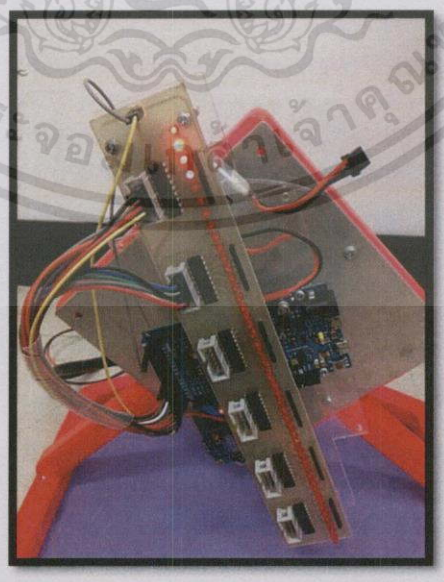
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.6 การทดสอบโปรแกรม โดยการ Openfile.txt ในโปรแกรม Visual Basic 2010 จะส่งข้อมูลไปยัง ไมโครคอนโทรลเลอร์ เพื่อสั่งงานวงจร LED Display ให้แสดงรูป

ขั้นแรก เปิดหน้าโปรแกรม Visual Basic 2010 ขั้นแรก เปิดหน้าเริ่มโปรแกรม Visual Basic 2010 ขึ้นมา จากนั้น เลือก Comport แล้วทำการเชื่อมต่อไมโครคอนโทรลเลอร์ ดังรูป 4.29

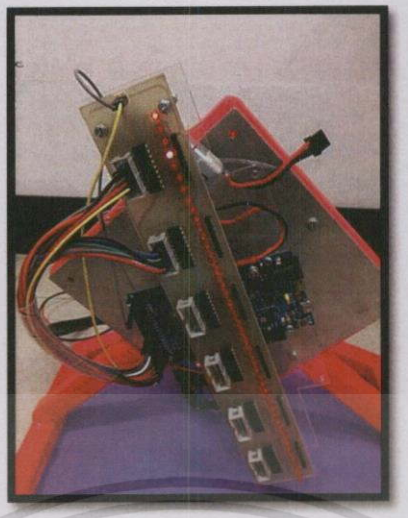
รูปที่ 4.29 แสดงภาพการเลือก Comport แล้วทำการเชื่อมต่อไมโครคอนโทรลเลอร์

ขั้นตอนต่อไป เมื่อกดปุ่ม Openfile เพื่อเลือกไฟล์ .txt ที่ต้องการส่งข้อมูลไปยังไมโครคอนโทรลเลอร์ที่เขียนโปรแกรมรับค่าจากโปรแกรม Visual Basic 2010 เมื่อไมโครคอนโทรลเลอร์ได้ข้อมูลมาแล้วก็จะสั่ง วงจร LED Display ให้แสดงผลหลอด LED ติดดับตามที่เลือกไฟล์มา เช่น Openfile square 4x4 จะเป็นการติดดับของหลอดไฟ LED ทีละ 4 ชุดๆละ 4 ดวง ซึ่งไฟล์จะเป็น 1111100110011111b โดยชุดแรกและชุดที่สี่หลอดไฟ LED จะติดทุกดวง ชุดที่สองและชุดที่สามหลอดไฟ LED จะติดแค่ดวงที่ 1 และดวงที่ 4 ดังรูป 4.30 และดังรูปที่ 4.31



รูปที่ 4.30 แสดงภาพ หลอดไฟชุดแรกและชุดที่สี่ หลอดไฟ LED ติดทุกดวง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

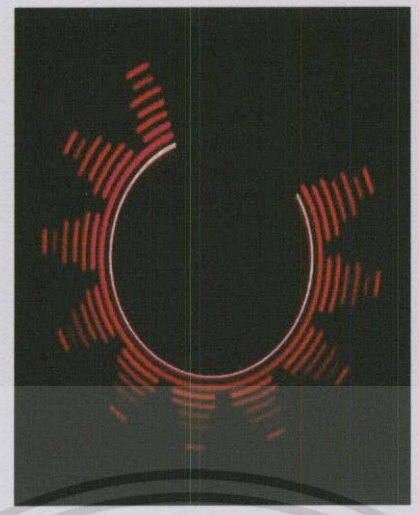


รูปที่ 4.31 แสดงภาพ หลอดไฟชุดที่สองและชุดที่สาม หลอดไฟ LED ติดแค่ดวงที่ 1 และดวงที่ 4

จากนั้นทดลองส่งไฟล์ Openfile square 8x8, triangle 8x8 แบบที่ให้หลอดไฟติด 8 bits คือติด 8 หลอด และทำการหมุนเพื่อให้แสดงผล ซึ่งจะได้ดังรูปที่ 4.32 และรูปที่ 4.33

รูปที่ 4.32 แสดงภาพ Square 8x8

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.33 แสดงภาพ Triangle 8x8

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 5
สรุปผลการทดลอง และข้อเสนอแนะ

5.1 สรุปผลการทดลอง

ปริญญานิพนธ์ฉบับนี้เป็นการศึกษาและออกแบบโปรแกรมที่ใช้ในการควบคุมแผงวงจร LED Display โดยสั่งงานผ่านไมโครคอนโทรลเลอร์ บอร์ด Arduino Due จากการศึกษาเบื้องต้นนั้นทำให้รู้ถึงการออกแบบวงจร LED Display โดยใช้โปรแกรม Altium, การออกแบบ Hardware โดยใช้โปรแกรม Solidwork, การเขียนโปรแกรมโดยใช้ Arduino 1.5.5 เพื่อใช้ในการสั่งงานไมโครคอนโทรลเลอร์ บอร์ด Arduino Due และได้เขียนโปรแกรม Visual Basic 2010 เป็นตัวเชื่อมต่อไปยังโปรแกรม Arduino 1.5.5

จากการทดลองเมื่อกดปุ่มส่งค่าในโปรแกรม Visual Basic 2010 จะทำการเชื่อมต่อโปรแกรม Arduino 1.5.5 และโปรแกรม Arduino 1.5.5 จากนั้นจะส่งข้อมูลที่ได้ทำการกดปุ่มนั้น ไปยังไมโครคอนโทรลเลอร์ บอร์ด Arduino Due โดยผ่าน zigbee เพื่อสั่งงานให้กับ LED Display แสดงผลเป็นรูปอย่างที่ต้องการ ได้ทั้ง แบบบิต, แบบไบท์, แบบคำ, แบบ Pattern และแบบ Send file

5.2 ข้อเสนอแนะ

1. วงจรไฟ LED Display ที่ใช้ในการทดลองนั้น ใช้ LED เพียงสีเดียวคือสีแดง สีเดียวทำให้แสดงผลแค่เป็นรูปขาวดำหรือติดดับเท่านั้น แต่สามารถนำไปเปลี่ยนเป็นหลอด 3 สี หรือหลอดRGB เพื่อแสดงผลเป็นภาพหลายสีได้
2. การส่งรูปภาพนั้นให้กับไมโครคอนโทรลเลอร์ สามารถส่งผ่าน Wireless ได้ในโครงงานนี้ใช้ Zigbee แทนสาย USB
3. มอเตอร์ไฟฟ้ากระแสตรงที่ใช้มีความเร็วในการหมุนนั้นไม่คงที่ ทำให้ใช้ ตัว Drive มอเตอร์เข้ามาทำการสอบเทียบมอเตอร์ให้มีความเร็วที่คงที่ หรือควรเปลี่ยนไปใช้มอเตอร์ที่คงที่ เช่น Stepping motor จะทำให้รูปภาพสามารถอยู่ในตำแหน่งที่ต้องการ
4. ฮาร์ดแวร์ควรจะออกแบบให้มีความสมดุล และมีน้ำหนักเบา เพื่อลดแรงเหวี่ยงของใบพัด จะได้ปลอดภัยมากยิ่งขึ้น
5. เนื่องจากหลอด LED มีความสว่างไม่เท่ากัน จึงทำให้รูปภาพที่เห็นนั้นไม่ชัดเจน ควรที่จะซื้อหลอดแอลอีดีให้เหมือนกันทั้งหมด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บรรณานุกรม

[1] เอกชัย มะการ. 2552. **เรียนรู้ เข้าใจ ใช้งาน ไมโครคอนโทรลเลอร์ตระกูล AVR ด้วย.** กรุงเทพฯ : บริษัท อีทีที จำกัด.

[2] ณัฐวุฒิ ประเสริฐอำไพสกุล. 2552.“ระบบสมองกลฝังตัวอย่างง่ายสำหรับแสดงผลบนจอ LED ” ปริญญานิพนธ์วิศวกรรมศาสตรบัณฑิต สาขาวิชาวิศวกรรมการวัดคุม, สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง.

[3] Arduino Due. [Online].
Available:
http://www.inex.co.th/inexstore/index.php?page=shop.product_details&flypage=flypage.tpl&product_id=513&category_id=14&option=com_virtuemart&Itemid=11&vmcchk=1&Itemid=11#.Utz4bRD-LIU

[4] Arduino
Available: http://arduino.cc/

[5] XBee Pro 60mW Wire Antenna - Series 1 (802.15.4). [Online].
Available: https://www.sparkfun.com/products/8742

[6] XBee®/XBee-PRO® RF Modules. [Online].
Available: https://www.sparkfun.com/datasheets/Wireless/Zigbee/XBee-Datasheet.pdf

[7] ไมโครคอนโทรลเลอร์. [Online].
Available: http://th.wikipedia.org/wiki/ไมโครคอนโทรลเลอร์

[8] ไมโครโพรเซสเซอร์. [Online].
Available: http://th.wikipedia.org/wiki/ไมโครโพรเซสเซอร์

[9] LED คืออะไร? [Online].
Available:
http://www.ledmaxwell.com/index.php?option=com_content&view%20%09%09%09=article&id=90&Itemid=158&lang=th

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

[10] ความรู้เกี่ยวกับมอเตอร์ไฟฟ้า. [Online].
Available: http://202.129.59.73/tn/motor10-52/motor1.htm

[11] Solidworks. [Online].
Available: http://solidworkweb.blogspot.com/2009/08/solidwork.html

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก

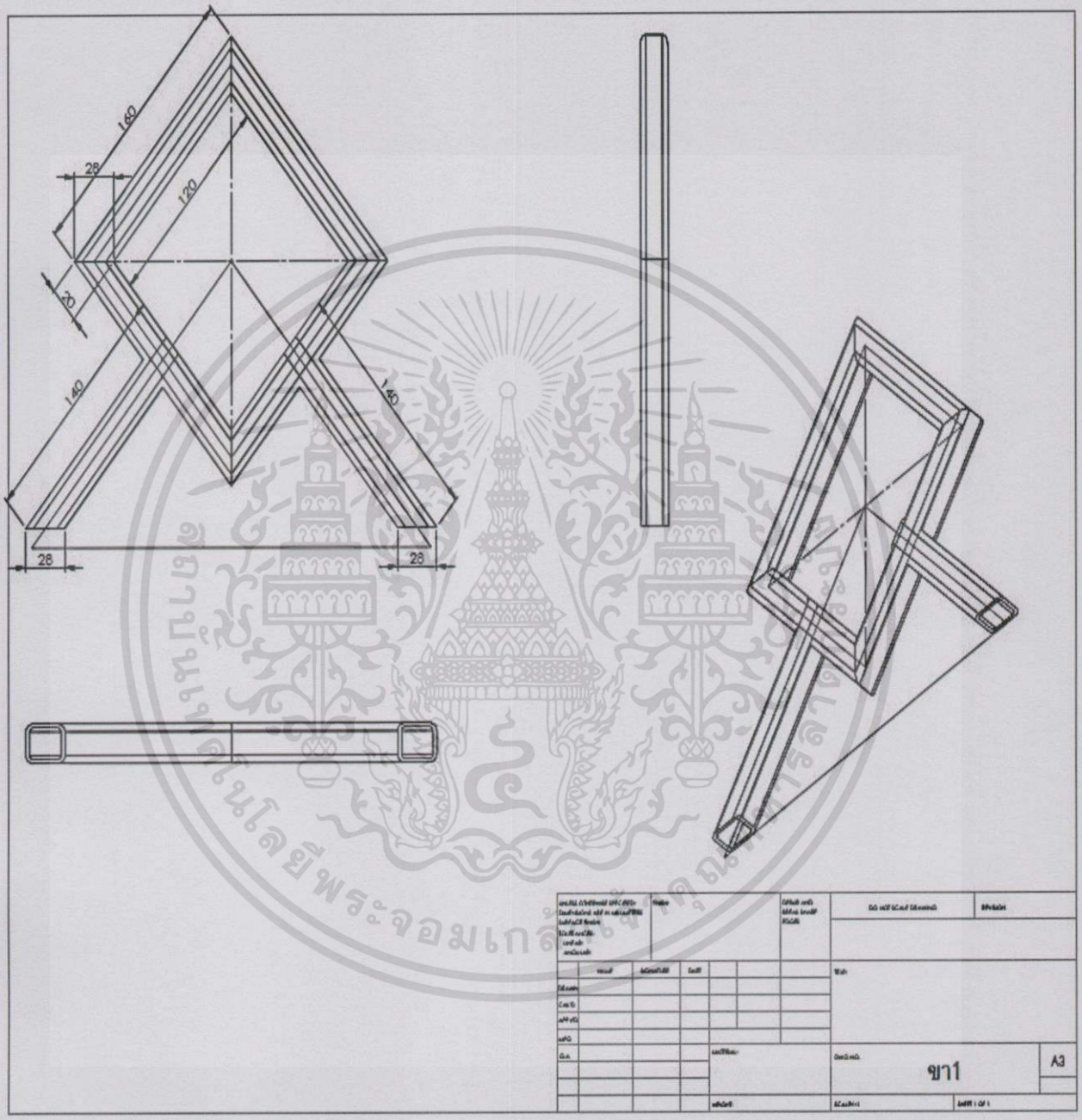
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก ก.

การออกแบบโดยใช้โปรแกรม Solidworks

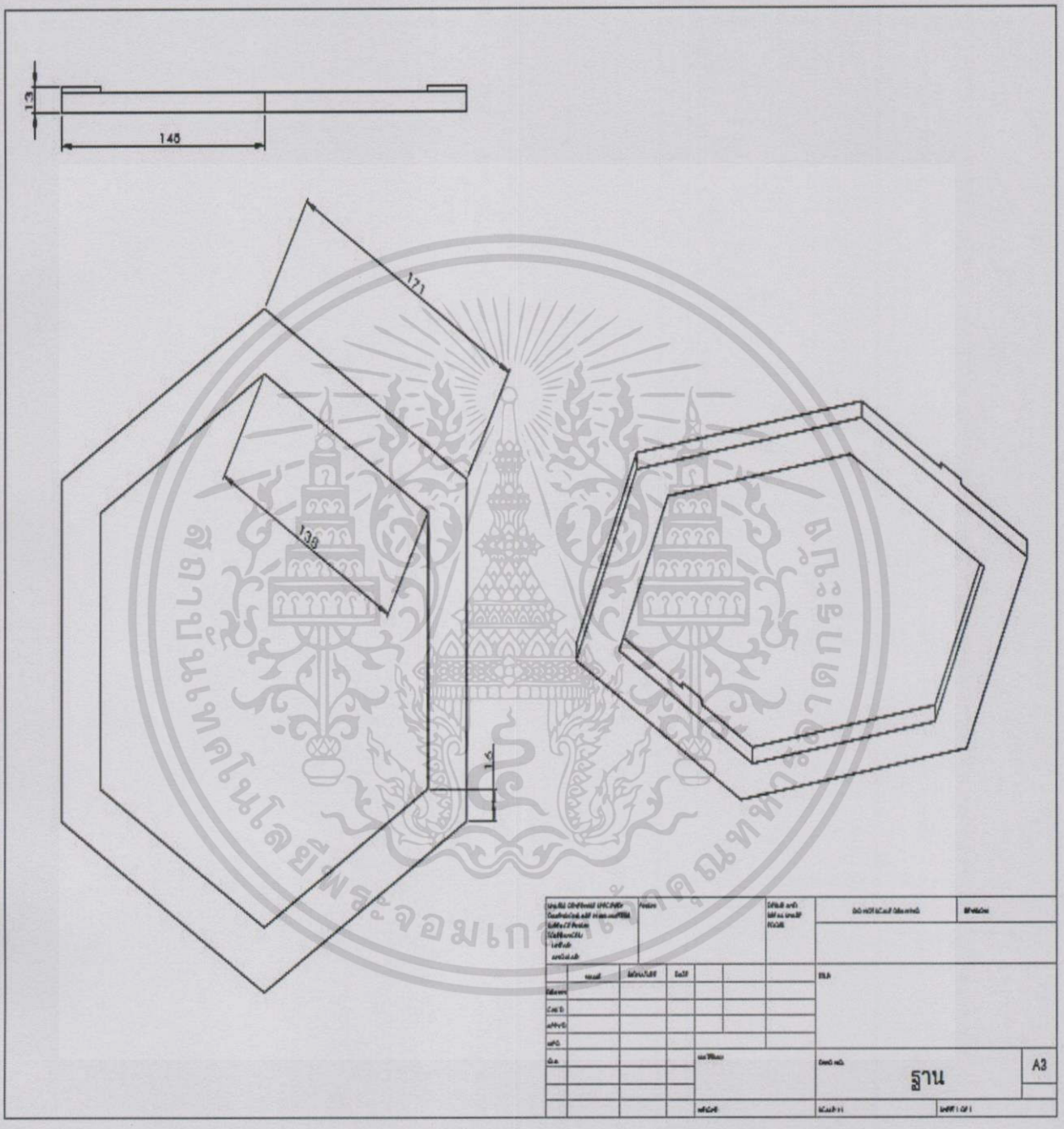
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

แบบร่างชิ้นส่วนขาตั้งวางชิ้นงาน



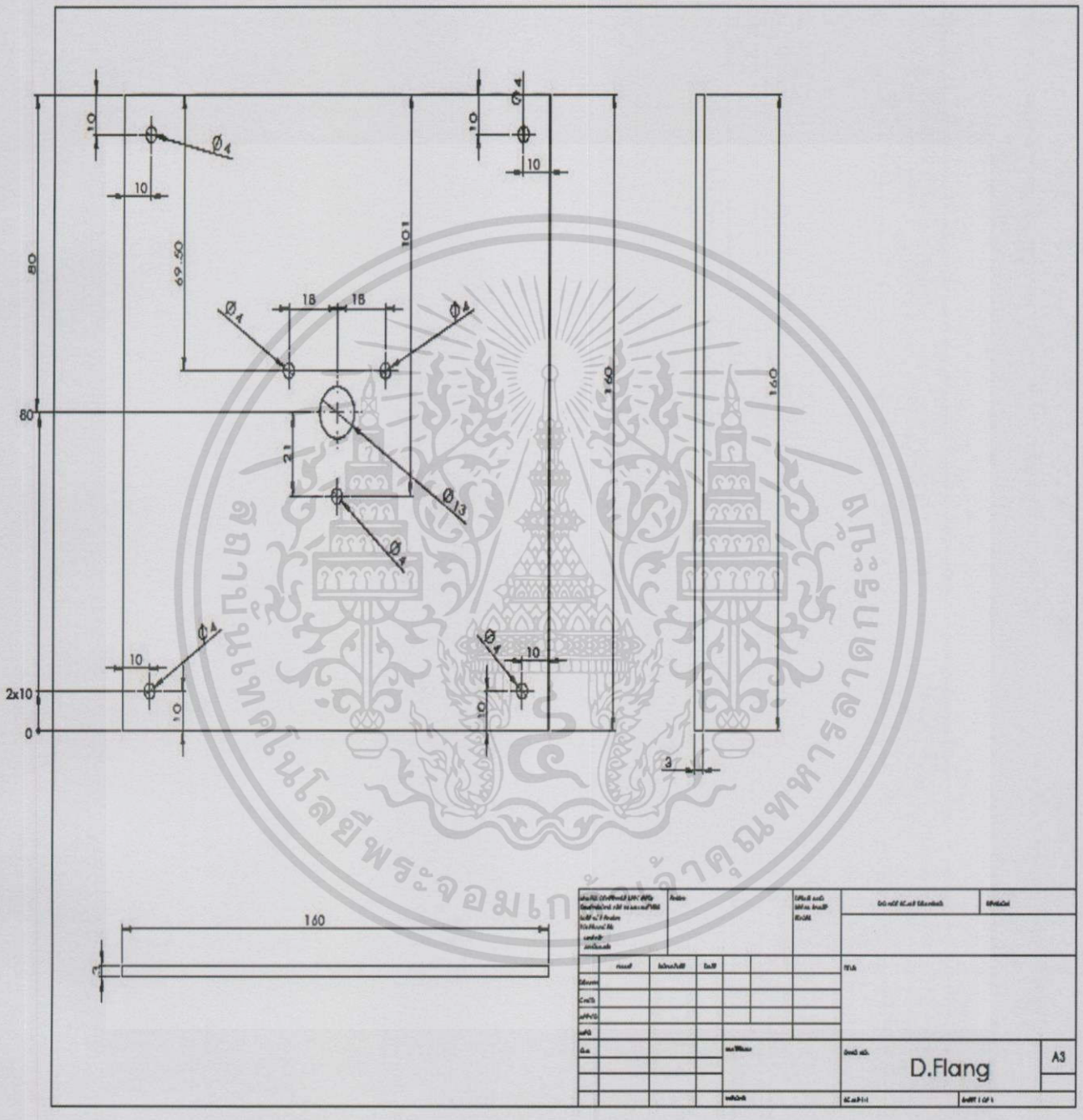
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

แบบร่างชิ้นส่วนฐานวางชิ้นงาน



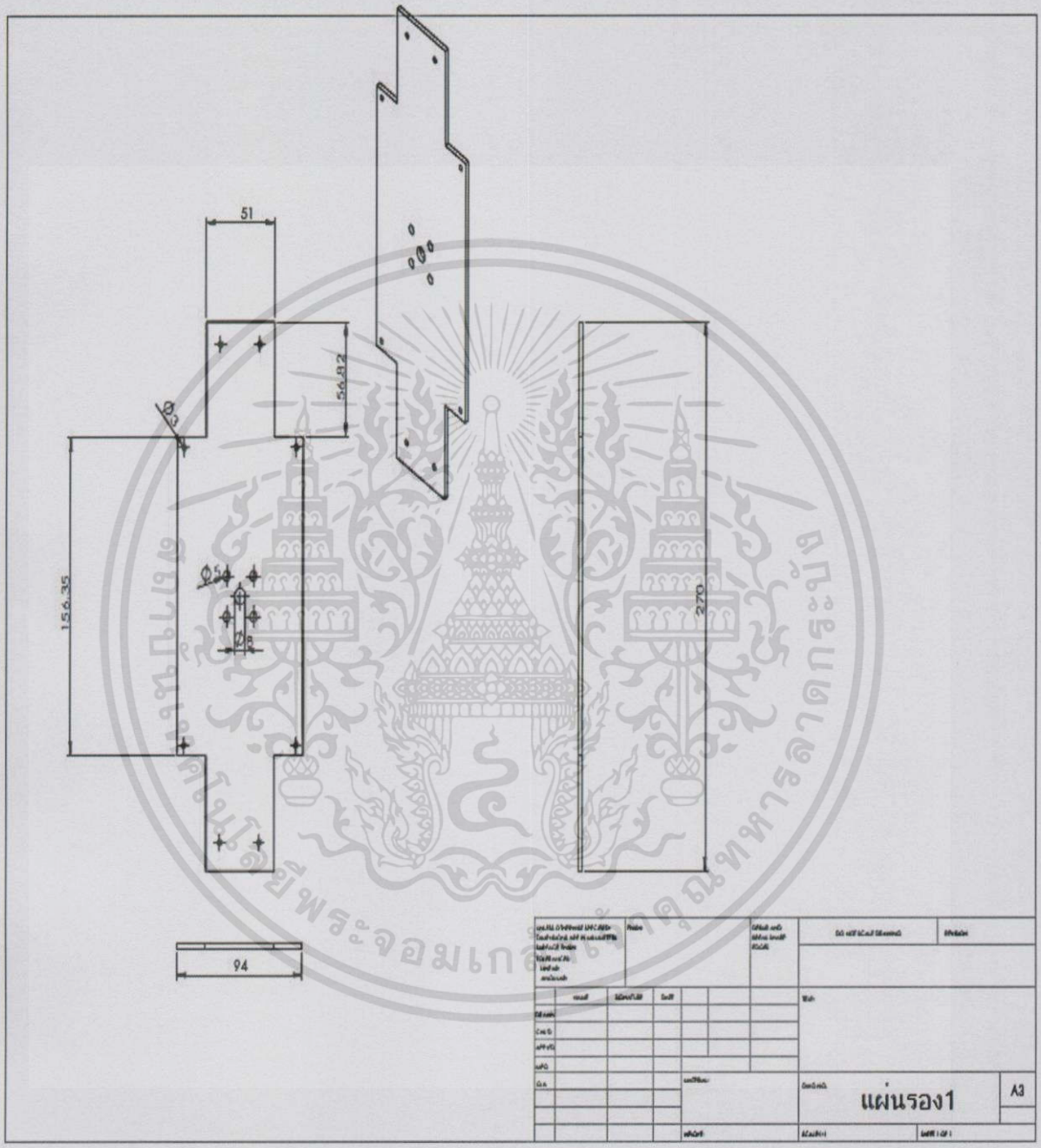
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

แบบร่างชิ้นส่วน Flange



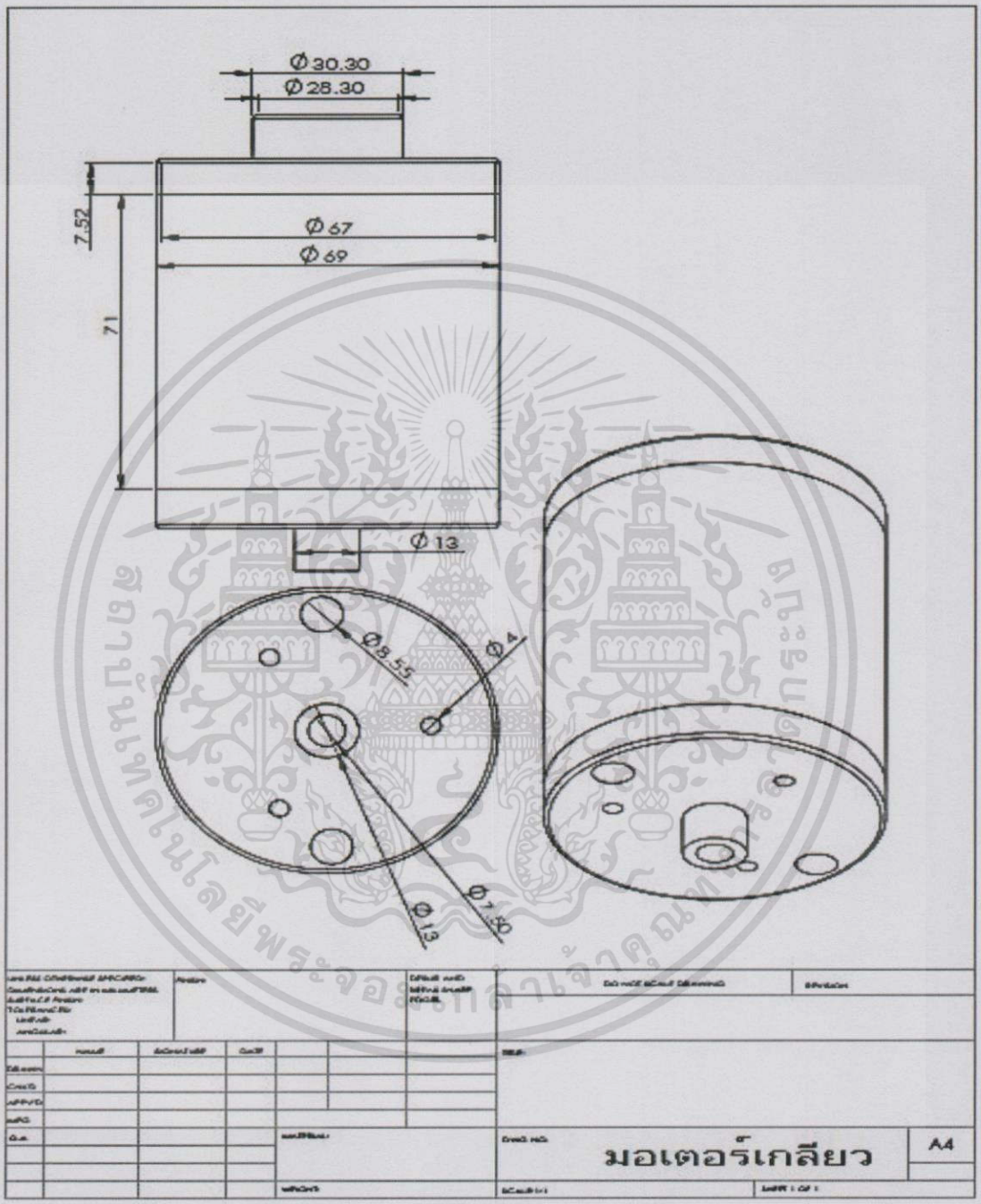
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

แบบร่างชิ้นส่วนของแผ่นรองแผงวงจรไฟ



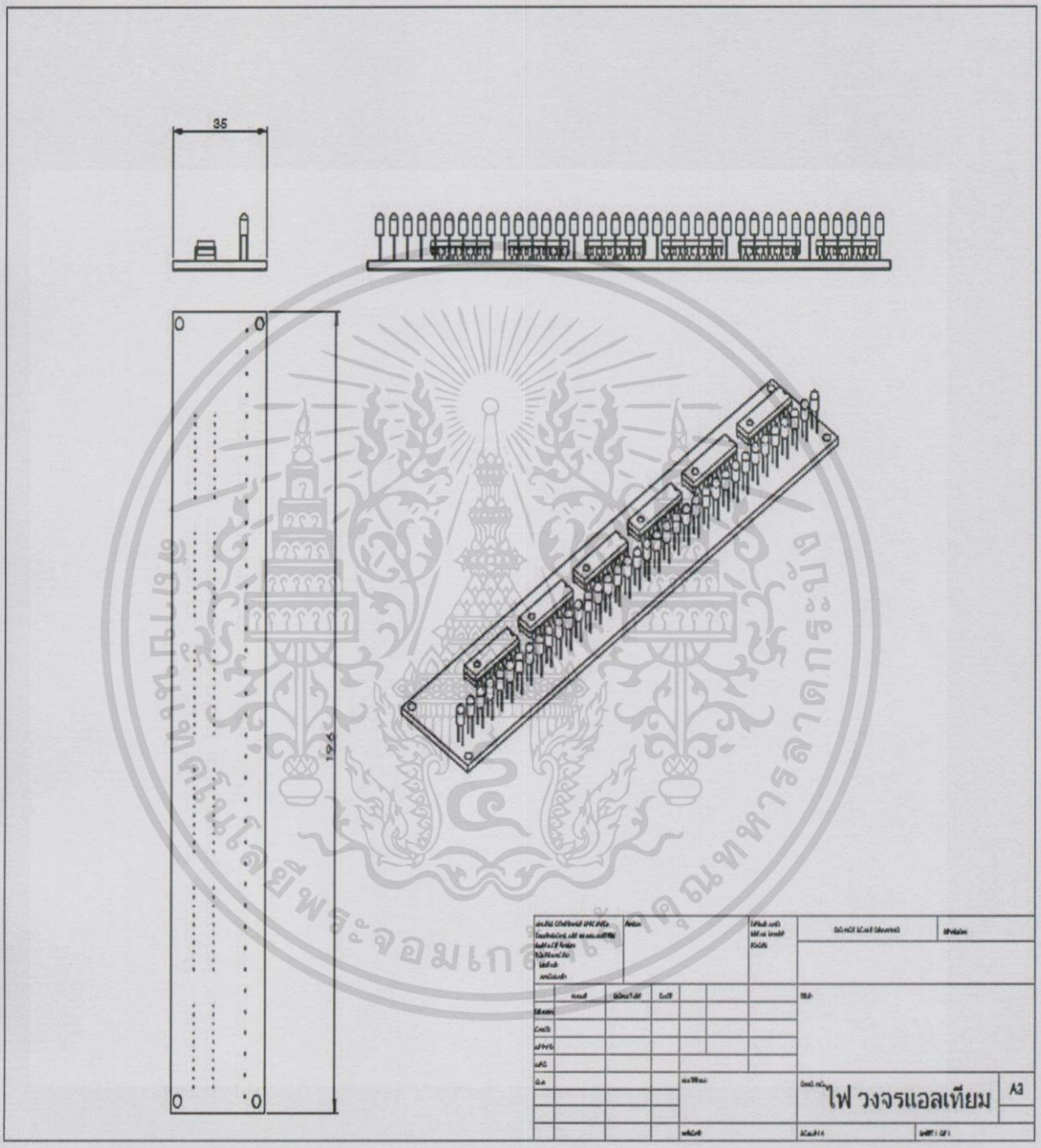
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

แบบร่างชิ้นส่วนของมอเตอร์



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

แบบร่างชิ้นส่วนแผงวงจรไฟ



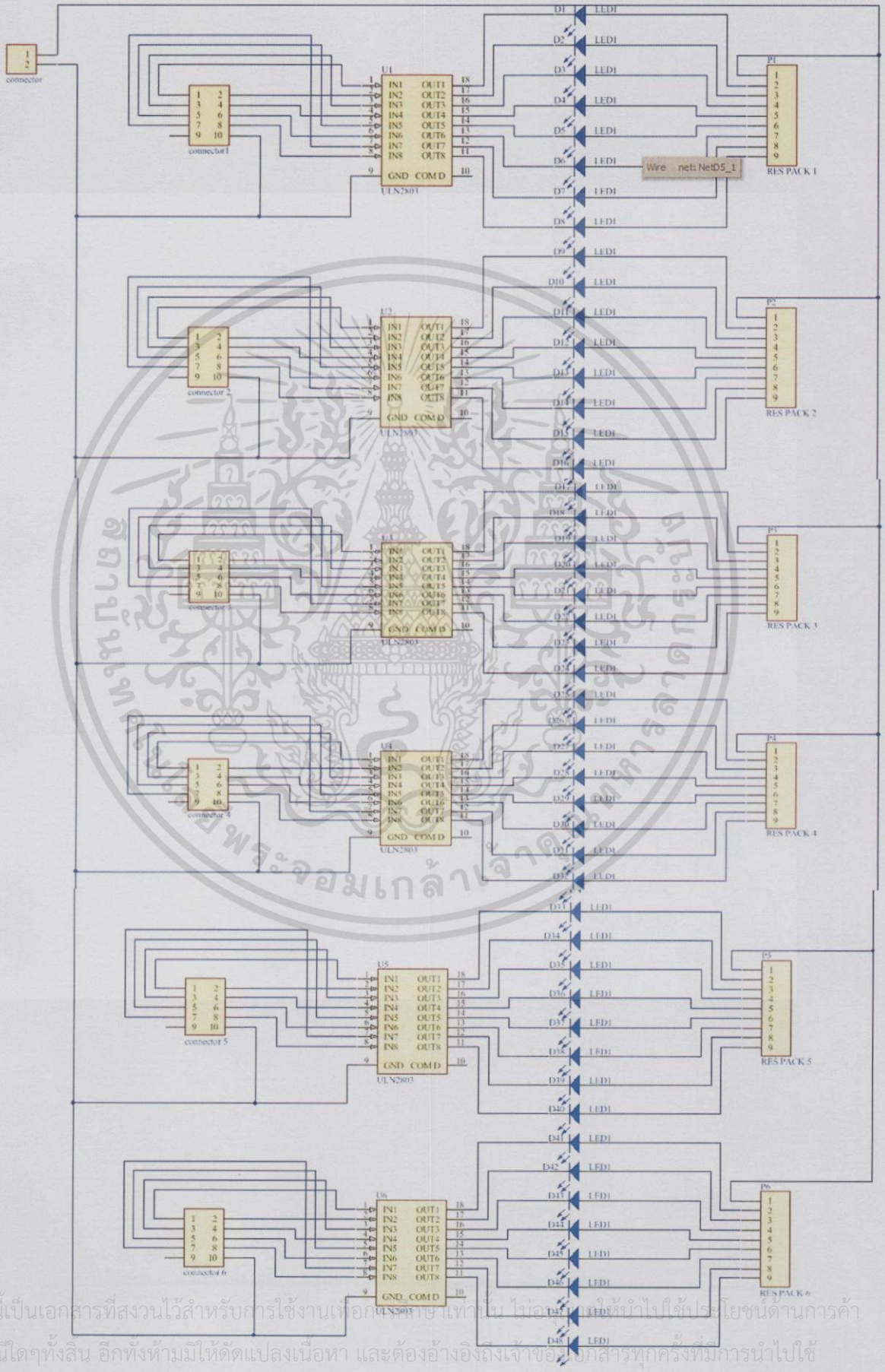
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก ข.

การออกแบบวงจรโดยใช้โปรแกรม Altium

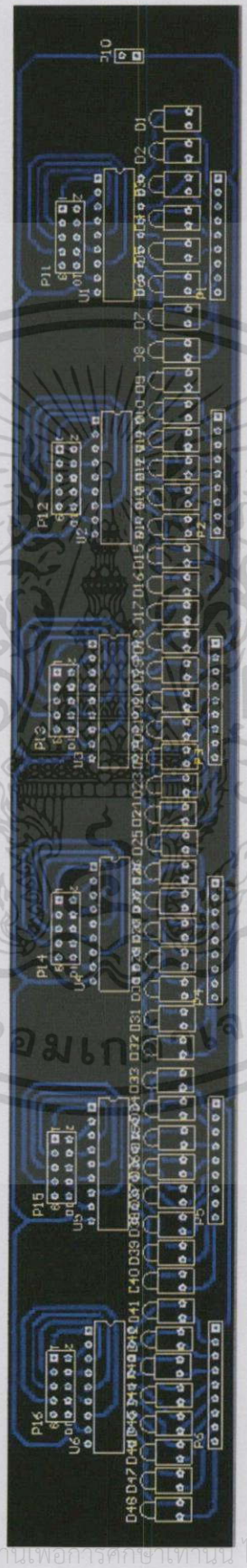
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

แผงวงจร แบบ Schematic



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

แผงวงจร แบบ PCB



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก ค.

โปรแกรม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โปรแกรม Arduino 1.5.5

- **โปรแกรมที่รับค่าจากกดปุ่มในโปรแกรม Visual Basic 2010 ออกเป็นรูปภาพ**

```
int led[] = {22,23,24,25,26,27,28,29,30,31,32,33,34,35,36,37};  // LED 16 หลอด
int led_NUM = 16;                                                   // จำนวนหลอด
char koh;
int a,b,c,d,e,i=0,j=0,l;
float x = 3;                                                        // Delay
void setup() {
Serial1.begin(9600);                                                // Serial baud rate
for(l=0;l<led_NUM;l++)
      {
      pinMode(led[l],OUTPUT);                          //เป็นการกำหนดขาเอาต์พุต
      digitalWrite(led[l], LOW);                                 // Turn off LED
      }
}

void loop() {
if (Serial1.available() > 0) {
koh = Serial1.read();                                 // กำหนด port ที่รับ-ส่ง ค่า
}
if(koh == '1')          //เมื่อได้รับค่า '1' จากการกดปุ่ม '1' ในโปรแกรม Visual Basic
{
bool pic[77][77]=  {{1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1},
     {1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1},
     {1,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,1},
     {1,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,1},
     {1,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,1},
     {1,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,1},
     {1,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,1},
     {1,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,1},
     {1,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,1},
     {1,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,1},
     {1,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,1},
     {1,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,1},
     {1,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,1},
     {1,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,1},
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    {1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1},
    {1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1}};
{
        for(i=0;i<32;i++)   // แบ่งความละเอียด หรือ Segment เป็น 32 Segment
        {
        for(j=0;j<16;j++)                                          //จำนวนหลอดที่ใช้
        {
        digitalWrite(led[j],pic[j][i]); //เป็นการแสดงผลเอาต์พุต led[j] = pic[j][i]
                                        โดยการวนลูป
        }
        delay(x);                                                  //เป็นการหน่วงเวลา x mS
        }
}
}
else if(koh == '2') //เมื่อได้รับค่า '2' จากการกดปุ่ม '2' ในโปรแกรม Visual Basic
{
bool pic[77][77]=  {{0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0},
    {0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0},
    {0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,1,1,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0},
    {0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,1,1,1,1,1,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0},
    {0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,1,1,1,1,1,1,1,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0},
    {0,0,0,0,0,0,0,0,0,0,0,0,1,1,1,1,1,1,1,1,1,1,1,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0},
    {0,0,0,0,0,0,0,0,0,0,0,1,1,1,1,1,1,1,1,1,1,1,1,1,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0},
    {0,0,0,0,0,0,0,0,0,0,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,0,0,0,0,0,0,0,0,0,0,0,0,0},
    {0,0,0,0,0,0,0,0,0,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,0,0,0,0,0,0,0,0,0,0,0,0},
    {0,0,0,0,0,0,0,0,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,0,0,0,0,0,0,0,0,0,0,0},
    {0,0,0,0,0,0,0,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,0,0,0,0,0,0,0,0,0,0},
    {0,0,0,0,0,0,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,0,0,0,0,0,0,0,0,0},
    {0,0,0,0,0,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,0,0,0,0,0,0,0,0},
    {0,0,0,0,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,0,0,0,0,0,0,0},
    {0,0,0,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,0,0,0,0,0,0},
    {0,0,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,0,0,0,0,0},
    {0,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,0,0,0,0},
    {0,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,0,0,0},
    {0,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,0,0},
    {0,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,0}};
{
        for(i=0;i<32;i++) // แบ่งความละเอียด หรือ Segment เป็น 32 Segment
        {
        for(j=0;j<16;j++)                                          //จำนวนหลอดที่ใช้
        {
        digitalWrite(led[j],pic[j][i]); //เป็นการแสดงผลเอาต์พุต led[j] = pic[j][i]
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                                        โดยการวนลูป
        }
        delay(x);                                   //เป็นการหน่วงเวลา x mS
        }
}
}
else if(koh == '3') //เมื่อได้รับค่า '3' จากการกดปุ่ม '3' ในโปรแกรม Visual Basic
{
bool pic[77][77]=  {{0,1,0,1,0,1,0,1,0,1,0,1,0,1,0,1,0,1,0,1,0,1,0,1,0,1,0,1,0,1,0,1},
    {0,1,0,1,0,1,0,1,0,1,0,1,0,1,0,1,0,1,0,1,0,1,0,1,0,1,0,1,0,1,0,1},
    {0,1,0,1,0,1,0,1,0,1,0,1,0,1,0,1,0,1,0,1,0,1,0,1,0,1,0,1,0,1,0,1},
    {0,0,1,1,0,0,1,1,0,0,1,1,0,0,1,1,0,0,1,1,0,0,1,1,0,0,1,1,0,0,1,1},
    {0,0,1,1,0,0,1,1,0,0,1,1,0,0,1,1,0,0,1,1,0,0,1,1,0,0,1,1,0,0,1,1},
    {0,0,1,1,0,0,1,1,0,0,1,1,0,0,1,1,0,0,1,1,0,0,1,1,0,0,1,1,0,0,1,1},
    {0,0,0,0,1,1,1,1,0,0,0,0,1,1,1,1,0,0,0,0,1,1,1,1,0,0,0,0,1,1,1,1},
    {0,0,0,0,1,1,1,1,0,0,0,0,1,1,1,1,0,0,0,0,1,1,1,1,0,0,0,0,1,1,1,1},
    {0,0,0,0,0,0,0,1,1,1,1,1,1,1,0,0,0,0,0,0,0,1,1,1,1,1,1,1,1},
    {0,0,0,0,0,0,0,1,1,1,1,1,1,1,0,0,0,0,0,0,0,1,1,1,1,1,1,1,1},
    {0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1},
    {0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1},
    {0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1},
    {1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1},
    {1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1},
    {1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1}};
{
        for(i=0;i<32;i++) // แบ่งความละเอียด หรือ Segment เป็น 32 Segment
        {
        for(j=0;j<16;j++)                                  //จำนวนหลอดที่ใช้
        {
        digitalWrite(led[j],pic[j][i]); //เป็นการแสดงผลเอาต์พุต led[j] = pic[j][i]
                                        โดยการวนลูป
        }
        delay(x);                                   //เป็นการหน่วงเวลา x mS
        }
}
}
else if(koh == '4') //เมื่อได้รับค่า '4' จากการกดปุ่ม '4' ในโปรแกรม Visual Basic
{
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
bool pic[77][77]=  {{0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0},
    {0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0},
    {0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,1,1,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0},
    {0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,1,1,1,1,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0},
    {0,0,0,0,0,0,0,0,0,0,0,0,0,1,1,1,1,1,1,1,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0},
    {0,0,0,0,0,0,0,0,0,0,0,0,1,1,1,1,1,1,1,1,1,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0},
    {0,0,0,0,0,0,0,0,0,0,0,1,1,1,1,1,1,1,1,1,1,1,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0},
    {0,0,0,0,0,0,0,0,0,0,0,1,1,1,1,1,1,1,1,1,1,1,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0},
    {0,0,0,0,0,0,0,0,0,0,0,1,1,1,1,1,1,1,1,1,1,1,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0},
    {0,0,0,0,0,0,0,0,0,0,0,1,1,1,1,1,1,1,1,1,1,1,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0},
    {0,0,0,0,0,0,0,0,0,0,0,1,1,1,1,1,1,1,1,1,1,1,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0},
    {0,0,0,0,0,0,0,0,0,0,0,1,1,1,1,1,1,1,1,1,1,1,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0},
    {0,0,0,0,0,0,0,0,0,0,0,1,1,1,1,1,1,1,1,1,1,1,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0},
    {0,0,0,0,0,0,0,0,0,0,0,1,1,1,1,1,1,1,1,1,1,1,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0},
    {0,0,0,0,0,0,0,0,0,0,0,1,1,1,1,1,1,1,1,1,1,1,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0},
    {0,0,0,0,0,0,0,0,0,0,0,1,1,1,1,1,1,1,1,1,1,1,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0},
    {0,0,0,0,0,0,0,0,0,0,0,1,1,1,1,1,1,1,1,1,1,1,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0},
    {0,0,0,0,0,0,0,0,0,0,0,1,1,1,1,1,1,1,1,1,1,1,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0}};
{
    for(i=0;i<32;i++)  //แบ่งความละเอียด หรือ Segment เป็น 32 Segment
    {
    for(j=0;j<16;j++)                                   //จำนวนหลอดที่ใช้
    {
    digitalWrite(led[j],pic[j][i]); //เป็นการแสดงผลเอาต์พุต led[j] = pic[j][i]
                                    โดยการวนลูป
    }
    delay(x);                                           //เป็นการหน่วงเวลา x mS
    }
}
}
else if(koh == '5') //เมื่อได้รับค่า ‘5’ จากการกดปุ่ม ‘5’ ในโปรแกรม Visual Basic
{
        Boolpic[177][177]={{0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,
        0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0},
        {0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,
        0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0},
        {0,0,0,0,0,0,0,0,0,0,0,0,0,1,0,0,0,0,1,1,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,
        0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0},
        {0,0,0,0,0,0,0,0,0,0,0,1,1,0,0,0,1,0,0,0,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,
        0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0},
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        {0,0,0,0,0,0,0,0,0,0,1,0,0,0,1,0,0,0,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,
        0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0},
        {0,0,0,0,0,0,0,0,0,0,1,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,
        0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0},
        {0,0,0,0,0,0,0,0,0,0,1,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,
        0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0},
        {0,0,0,0,0,0,0,0,0,0,1,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,
        0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0},
        {0,0,0,0,0,0,0,0,0,0,1,0,0,0,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,
        0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0},
        {0,0,0,0,0,0,0,0,0,0,1,1,1,1,1,0,1,1,1,1,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,
        0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0},
        {0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,
        0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0},
        {0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,
        0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0},
        {0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,
        0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0},
        {0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,
        0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0},
        {0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,
        0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0},
        {0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,
        0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0}};
{
        for(i=0;i<64;i++)        // แบ่งความละเอียด หรือ Segment เป็น 64 Segment
        {
        for(j=0;j<16;j++)                                          //จำนวนหลอดที่ใช้
        {
        digitalWrite(led[j],pic[j][i]);        //เป็นการแสดงผลเอาต์พุต led[j] = pic[j][i]
                                               โดยการวนลูป
        }
        delay(x);                                       //เป็นการหน่วงเวลา x mS
        }
}
}
else if(koh == '6') //เมื่อได้รับค่า ‘6’ จากการกดปุ่ม ‘6’ ในโปรแกรม Visual Basic
{
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
bool pic[77][77]=  {{0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0},
    {0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0},
    {0,0,1,1,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0},
    {0,0,1,0,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0},
    {0,0,1,0,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0},
    {0,0,1,1,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0},
    {0,0,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0},
    {0,0,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0},
    {0,0,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0},
    {0,0,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0},
    {0,0,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0},
    {0,0,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0},
    {0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0},
    {0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0},
    {0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0},
    {0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0}};
{
    for(i=0;i<32;i++)  //แบ่งความละเอียด หรือ Segment เป็น 32 Segment
    {
    for(j=0;j<16;j++)                                    //จำนวนหลอดที่ใช้
    {
    digitalWrite(led[j],pic[j][i]); //เป็นการแสดงผลเอาต์พุต led[j] = pic[j][i]
                                    โดยการวนลูป
    }
    delay(x);                                    //เป็นการหน่วงเวลา x mS
    }
}
}
else if(koh == '7') //เมื่อได้รับค่า '7' จากการกดปุ่ม '7' ในโปรแกรม Visual Basic
{
bool pic[77][77]=  {{0,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,0},
    {0,0,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,0,0},
    {0,0,0,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,0,0,0},
    {0,0,0,0,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,0,0,0,0},
    {0,0,0,0,0,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,0,0,0,0,0},
    {0,0,0,0,0,0,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,0,0,0,0,0,0},
    {0,0,0,0,0,0,0,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,0,0,0,0,0,0,0},
    {0,0,0,0,0,0,0,0,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,0,0,0,0,0,0,0},
    {0,0,0,0,0,0,0,0,0,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,0,0,0,0,0,0,0},
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
{0,0,0,0,0,0,0,0,0,1,1,1,1,1,1,1,1,1,1,1,1,1,0,0,0,0,0,0,0,0,0,0},
{0,0,0,0,0,0,0,0,0,1,1,1,1,1,1,1,1,1,1,1,1,1,0,0,0,0,0,0,0,0,0,0},
{0,0,0,0,0,0,0,0,0,1,1,1,1,1,1,1,1,1,1,1,1,1,0,0,0,0,0,0,0,0,0,0},
{0,0,0,0,0,0,0,0,0,1,1,1,1,1,1,1,1,1,1,1,1,1,0,0,0,0,0,0,0,0,0,0},
{0,0,0,0,0,0,0,0,0,1,1,1,1,1,1,1,1,1,1,1,1,1,0,0,0,0,0,0,0,0,0,0},
{0,0,0,0,0,0,0,0,0,0,1,1,1,1,1,1,1,1,1,1,0,0,0,0,0,0,0,0,0,0,0,0},
{0,0,0,0,0,0,0,0,0,0,1,1,1,1,1,1,1,1,1,1,0,0,0,0,0,0,0,0,0,0,0,0},
{0,0,0,0,0,0,0,0,0,0,0,1,1,1,1,1,1,1,1,1,0,0,0,0,0,0,0,0,0,0,0,0},
{0,0,0,0,0,0,0,0,0,0,0,0,1,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0},
{0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0}};
{
    for(i=0;i<32;i++) //แบ่งความละเอียด หรือ Segment เป็น 32 Segment
    {
    for(j=0;j<16;j++)                                    //จำนวนหลอดที่ใช้
    {
    digitalWrite(led[j],pic[j][i]); //เป็นการแสดงผลเอาต์พุต led[j] = pic[j][i]
                                    โดยการวนลูป
    }
    delay(x);                                  //เป็นการหน่วงเวลา x mS
    }
}
}
else if(koh == '8') //เมื่อได้รับค่า '8' จากการกดปุ่ม '8' ในโปรแกรม Visual Basic
{
bool pic[77][77]=  {{1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1},
    {1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,0,0},
    {1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,0,0,0},
    {1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,0,0,0,0},
    {1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,0,0,0,0,0},
    {1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,0,0,0,0,0,0},
    {1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,0,0,0,0,0,0,0},
    {1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,0,0,0,0,0,0,0,0},
    {1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,0,0,0,0,0,0,0,0,0},
    {1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,0,0,0,0,0,0,0,0,0,0},
    {1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,0,0,0,0,0,0,0,0,0,0,0},
    {1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,0,0,0,0,0,0,0,0,0,0,0,0},
    {1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,0,0,0,0,0,0,0,0,0,0,0,0,0},
    {1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0},
    {1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0},
    {1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0},
    {1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0},
    {1,1,1,1,1,1,1,1,1,1,1,1,1,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0},
    {1,1,1,1,1,1,1,1,1,1,1,1,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0},
    {1,1,1,1,1,1,1,1,1,1,1,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0},
    {1,1,1,1,1,1,1,1,1,1,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0},
    {1,1,1,1,1,1,1,1,1,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0},
    {1,1,1,1,1,1,1,1,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0},
    {1,1,1,1,1,1,1,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0},
    {1,1,1,1,1,1,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0},
    {1,1,1,1,1,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0},
    {1,1,1,1,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0},
    {1,1,1,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0},
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    {1,1,1,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0},
    {1,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0}};
{
      for(i=0;i<32;i++) //แบ่งความละเอียด หรือ Segment เป็น 32 Segment
      {
      for(j=0;j<16;j++)                                    //จำนวนหลอดที่ใช้
      {
      digitalWrite(led[j],pic[j][i]); //เป็นการแสดงผลเอาต์พุต led[j] = pic[j][i]
                                      โดยการวนลูป
      }
      delay(x);                                            //เป็นการหน่วงเวลา x mS
      }
}
}
else if(koh == '9') //เมื่อได้รับค่า '9' จากการกดปุ่ม '9' ในโปรแกรม Visual Basic
{
bool pic[77][77]=  {{0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0},
    {0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0},
    {0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0},
    {0,0,0,0,1,1,1,1,1,1,0,1,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0},
    {0,0,0,0,0,0,0,0,1,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0},
    {0,0,0,0,0,0,0,1,1,1,0,1,1,0,1,1,1,1,1,1,1,0,0,1,1,1,1,1,0,0,0,0,0,0,0},
    {0,0,0,0,0,0,0,1,1,1,0,1,1,0,1,1,1,0,1,1,0,0,1,1,0,1,1,0,0,0,0,0,0,0,0},
    {0,0,0,0,0,0,0,1,1,1,0,0,1,1,0,1,1,1,0,1,1,0,1,1,1,0,1,1,1,0,0,0,0,0,0},
    {0,0,0,0,0,1,1,1,0,0,0,1,1,0,1,1,1,0,1,1,0,1,1,1,1,1,1,1,1,0,0,0,0,0,0},
    {0,0,0,0,1,1,1,0,0,0,0,1,1,0,1,1,1,0,1,1,0,1,1,1,0,0,0,0,0,0,0,0,0,0,0},
    {0,0,0,0,1,1,0,0,0,0,0,1,1,0,1,1,1,0,1,1,0,0,1,1,0,0,0,0,0,0,0,0,0,0,0},
    {0,0,0,0,1,1,1,1,1,1,0,1,1,0,1,1,1,0,1,1,0,0,0,1,1,1,1,1,1,0,0,0,0,0,0},
    {0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0},
    {0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0},
    {0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0},
    {0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0},
    {0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0}};
{
      for(i=0;i<32;i++) //แบ่งความละเอียด หรือ Segment เป็น 32 Segment
      {
      for(j=0;j<16;j++)                                    //จำนวนหลอดที่ใช้
      {
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        digitalWrite(led[j],pic[j][i]);  //เป็นการแสดงผลเอาต์พุต led[j] = pic[j][i]
                                         โดยการวนลูป
        }
        delay(x);                                    //เป็นการหน่วงเวลา x mS
        }
}
}
else if(koh == 'a') //เมื่อได้รับค่า 'a' จากการกดปุ่ม 'a' ในโปรแกรม Visual Basic
{
bool pic[77][77]=  {0,0,0,0,0,0,0,0,0,0,0,0,0,0,0},
                   {0,0,0,0,0,0,0,0,0,0,0,0,0,0,0},
                   {0,0,0,0,0,0,0,0,0,0,0,0,0,0,0},
                   {0,0,0,0,0,1,1,1,1,1,0,0,0,0,0},
                   {0,0,0,0,0,0,0,0,0,0,0,0,0,0,0},
                   {0,0,0,0,0,0,0,0,0,0,0,0,0,0,0},
                   {0,0,0,0,0,0,0,1,0,0,0,0,0,0,0},
                   {0,0,0,0,0,0,0,0,0,0,0,0,0,0,0},
                   {0,0,0,0,0,0,0,0,0,0,0,0,0,0,0},
                   {0,0,0,0,0,1,0,0,0,1,0,0,0,0,0},
                   {0,0,0,0,0,0,0,0,0,0,0,0,0,0,0},
                   {0,0,0,0,0,0,0,0,0,0,0,0,0,0,0},
                   {0,0,0,0,0,0,0,0,0,0,0,0,0,0,0},
                   {0,0,0,0,0,0,0,0,0,0,0,0,0,0,0},
                   {0,0,0,0,0,0,0,0,0,0,0,0,0,0,0},
                   {0,0,0,0,0,0,0,0,0,0,0,0,0,0,0}};{
        for(i=0;i<16;i++) //แบ่งความละเอียด หรือ Segment เป็น 16 Segment
        {
        for(j=0;j<16;j++)                            //จำนวนหลอดที่ใช้
        {
        digitalWrite(led[j],pic[j][i]);  //เป็นการแสดงผลเอาต์พุต led[j] = pic[j][i]
                                         โดยการวนลูป

        }
        delay(x);                                    //เป็นการหน่วงเวลา x mS
        }
}
}
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

}

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- **โปรแกรมที่รับค่าจากปุ่มกด Open file เลือกไฟล์ที่ต้องการจะแสดงผลจากโปรแกรม Visual Basic 2010 แบบ 4 หลอด**

```
char inChar;                                  // กำหนดตัวแปร
int led[4] = {22,23,24,25};                   // กำหนดตัวแปรของหลอดไฟ LED จำนวน 4
                                              หลอด
int data[4][4];                               // กำหนดตัวแปรอาเรย์เพื่อเก็บค่า
int count = 0;                                // กำหนดตัวแปร
int count1 = 0;                               // กำหนดตัวแปร
int i=0,j=0,x=0,y=0;                          // กำหนดตัวแปร
void setup() {                                // ฟังชั่นนี้จะถูกเรียกใช้งานเพียงครั้งเดียวเมื่อ
                                              โปรแกรมถูกเริ่มใช้งาน ใช้สำหรับการกำหนดค่า
                                              ของขาของตัว AVR หรือตั้งค่าการติดต่อแบบ
                                              serial โปรแกรมนี้เป็นส่วนจำเป็นที่จะต้องมี
                                              ถึงแม้มันจะไม่มีคำสั่งภายในตัวมันก็ตาม
  Serial1.begin(9600);                        // คำสั่งนี้จะเปิด serial port และตั้งค่า
                                              ความเร็วในการรับส่ง ปกติการติดต่อกับ
                                              เครื่องพีซีจะตั้งไว้ที่ ความเร็ว 9600 แม้ว่ามัน
                                              สามารถรับส่งได้ที่ความเร็วมากกว่านี้ก็ตาม
  Serial2.begin(9600);

  pinMode(led[0] ,OUTPUT);                    // กำหนดหน้าที่ขาของไมโครคอนโทรลเลอร์ให้
                                              เป็น ขา รับสัญญาณ INPUT หรือขาส่งสัญญาณ
                                              OUTPUT
  pinMode(led[1] ,OUTPUT);
  pinMode(led[2] ,OUTPUT);
  pinMode(led[3] ,OUTPUT);
  digitalWrite(led[0] ,LOW);                  // ส่งค่าลอจิก HIGH หรือ LOW (เปิด หรือ ปิด)
                                              ไปยังขา digital ที่กำหนด
  digitalWrite(led[1] ,LOW);
  digitalWrite(led[2] ,LOW);
  digitalWrite(led[3] ,LOW);
}
void loop() {                                 // ฟังก์ชั่นนี้จะทำงานต่อจาก void setup()
                                              และจะทำงานเป็นวนลูป เพื่อให้โปรแกรม
                                              ทำงานตามสถานะต่างๆ
    if (Serial1.available() > 0) {            // ใช้ตรวจสอบว่าบัฟเฟอร์รับข้อมูลไว้มากกว่า
                                              ศูนย์หรือไม่
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    inChar = (char)Serial1.read();          // รับค่าตัวแปรทีละ char ทางพอร์ตอนุกรมม 1 แล้วเก็บไว้ใน inChar
    Serial2.write(inChar);                  // ส่งค่าตัวแปร inChar ออกทางพอร์ตอนุกรม 2
     if (inChar == 'b'){                    // ใช้ตรวจสอบค่าตัวแปร inChar ว่าเท่ากับ b หรือไม่
     Serial1.flush();                       // เคลียร์บัฟเฟอร์ของพอร์ตอนุกรม(Serial) ให้ว่าง
     count = 0;
     count1 = 0;
      }
      else if (inChar == '0' || inChar == '1'){   // ใช้ตรวจสอบค่าตัวแปร inChar ว่าเท่ากับ 0 หรือ 1
       data[count][count1] = inChar-48;     // นำค่าตัวแปร inChar มาลบกับ 48 เพื่อให้ได้ค่าจริงเป็น 1 หรือ 0 แล้วเก็บค่าในอาเรย์ data
      count1++;
      if (count1 == 4){                     // ใช้ตรวจสอบค่าตัวแปร count1 ว่าเท่ากับ 4 หรือไม่
       count++;
       count1=0;
       if (count == 4)   i = inChar-48;     // ใช้ตรวจสอบค่าตัวแปร count ว่าเท่ากับ 4 หรือไม่ นำค่าตัวแปร inChar มาลบกับ 48 เพื่อให้ได้ค่าจริงเป็น 1 หรือ 0 แล้วเก็บค่าในตัวแปร i
      }
     }
    }
   else {
    for (x=0; x<4; x++){                    // ใช้ในการวนลูปเก็บค่าในอาเรย์ data แล้วแสดงผลไปที่ led
      for (y=0; y<4; y++){
       digitalWrite(led[y], data[x][y]);    // คำสั่งที่กำหนดให้ PIN ในแบบดิจิตอลมีค่าเป็นค่าที่ถูกเก็บในในอาเรย์ data
    }
      delay (i);                            // ค่าดีเลย์ที่สามารถกำหนดได้เองจากตัวแปร i
   }
   }
}
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- **โปรแกรมที่รับค่าจากปุ่มกด open file เลือกไฟล์ที่ต้องการจะแสดงผลจากโปรแกรม Visual Basic 2010 แบบ 8 หลอด**

```
char inChar;                                  // กำหนดตัวแปร
int led[8] = {22,23,24,25,26,27,28,29};       // กำหนดตัวแปรของหลอดไฟ LED
                                              จำนวน 8 หลอด
bool data[8][8];                              // กำหนดตัวแปรอาเรย์เพื่อเก็บค่า
int count = 0;                                // กำหนดตัวแปร
int count1 = 0;                               // กำหนดตัวแปร
int i=0,j=0,x=0,y=0;                          // กำหนดตัวแปร

void setup() {                                // ฟังชั่นนี้จะถูกเรียกใช้งานเพียงครั้งเดียวเมื่อ
                                              โปรแกรมถูกเริ่มใช้งาน ใช้สำหรับการกำหนดค่า
                                              ของขาของตัว AVR หรือตั้งค่าการติดต่อแบบ
                                              serial โปรแกรมนี้เป็นส่วนจำเป็นที่จะต้องมี
                                              ถึงแม้มันจะไม่มีคำสั่งภายในตัวมันก็ตาม
  Serial1.begin(9600);                        // คำสั่งนี้จะเปิด serial port และตั้งค่า
                                              ความเร็วในการรับส่ง ปกติการติดต่อกับ
                                              เครื่องพีซีจะตั้งไว้ที่ ความเร็ว 9600 แม้ว่ามัน
                                              สามารถรับส่งได้ที่ความเร็วมากกว่านี้ก็ตาม

  Serial2.begin(9600);
  for (i=0; i<8; i++){                        // ใช้ในการวนลูปจำนวนหลอด led ที่แสดงผล
    pinMode(led[i], OUTPUT);                  // กำหนดหน้าที่ขาของไมโครคอนโทรลเลอร์ให้
                                              เป็น ขา รับสัญญาณ INPUT หรือขาส่งสัญญาณ
                                              OUTPUT

    digitalWrite(led[i] ,LOW);                // ส่งค่าลอจิก HIGH หรือ LOW (เปิด หรือ ปิด)
                                              ไปยังขา digital ที่กำหนด

  }
}
void loop() {                                 // ฟังก์ชั่นนี้จะทำงานต่อจาก void setup()
                                              และจะทำงานเป็นวนลูป เพื่อให้โปรแกรมทำงาน
                                              ตามสถานะต่างๆ

  if (Serial1.available() > 0) {              // ใช้ตรวจสอบว่าบัฟเฟอร์รับข้อมูลไว้มากกว่า
                                              ศูนย์หรือไม่
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    inChar = (char)Serial1.read();          // รับค่าตัวแปรทีละ char ทางพอร์ตอนุกรม
                                            1 แล้วเก็บไว้ใน inChar
    Serial2.write(inChar);                  // ส่งค่าตัวแปร inChar ออกทางพอร์ตอนุกรม
                                            2
      if (inChar == 'b'){                   // ใช้ตรวจสอบค่าตัวแปร inChar ว่าเท่ากับ b
                                            หรือไม่
      Serial1.flush();                      // เคลียร์บัฟเฟอร์ของพอร์ตอนุกรม(Serial) ให้
                                            ว่าง
      count = 0;
      count1 = 0;
       }
       else if (inChar == '0' || inChar == '1'){   // ใช้ตรวจสอบค่าตัวแปร inChar ว่า
                                                   เท่ากับ 0 หรือ 1
        data[count][count1] = inChar-48;           // นำค่าตัวแปร inChar มาลบกับ 48 เพื่อให้ได้
                                                   ค่าจริงเป็น 1 หรือ 0 แล้วเก็บค่าในอาเรย์ data
      count1++;
      if (count1 == 8){                    // ใช้ตรวจสอบค่าตัวแปร count1 ว่า
                                           เท่ากับ 4 หรือไม่
       count++;
       count1=0;
       if (count == 8)   i = inChar-48;    // ใช้ตรวจสอบค่าตัวแปร count ว่า
                                           เท่ากับ 4 หรือไม่ นำค่าตัวแปร inChar มาลบ
                                           กับ 48 เพื่อให้ได้ค่าจริงเป็น 1 หรือ 0 แล้วเก็บ
                                           ค่าในตัวแปร i
      }
     }
    }
   else {
    for (x=0; x<8; x++){                   // ใช้ในการวนลูปเก็บค่าในอาเรย์ data แล้ว
                                           แสดงผลไปที่ led
      for (y=0; y<8; y++){
       digitalWrite(led[y], data[x][y]);   // คำสั่งที่กำหนดให้ PIN ในแบบดิจิตอลมีค่า
                                           เป็นค่าที่ถูกเก็บในในอาเรย์ data
    }
      delay (i);                           // ค่าดีเลย์ที่สามารถกำหนดได้เองจากตัวแปร i
   }
   }
}
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โปรแกรม Visual Basic 2010

- **โปรแกรมส่งค่าจากปุ่มกดไปยังไมโครคอลโทรลเลอร์ เพื่อแสดงผลที่แผงไฟวงจร LED**

```
Imports System.Threading
Imports System.IO.Ports
Imports System.Windows.Forms.ComboBox

Public Class Form1
Shared data(100) As String
Private Sub Form1_Load(sender As System.Object, e As System.EventArgs) Handles
MyBase.Load
SerialPort1.Close()
SerialPort1.BaudRate = 9600                    'Serial Baud Rate
SerialPort1.DataBits = 16                      'จำนวนหลอด
SerialPort1.Parity = Parity.None
SerialPort1.StopBits = StopBits.One
SerialPort1.Handshake = Handshake.None
SerialPort1.Encoding = System.Text.Encoding.ASCII      'very important!
End Sub

Private Sub Connect_Click(sender As System.Object, e As System.EventArgs) Handles
Connect.Click
Dim port As String
port = ComboBox1.Text
SerialPort1.PortName = port
SerialPort1.Open()
Disconnect.Enabled = True
Connect.Enabled = False
End Sub

Private Sub Disconnect_Click(sender As System.Object, e As System.EventArgs) Handles
Disconnect.Click
SerialPort1.Close()
Disconnect.Enabled = False
Connect.Enabled = True
End Sub
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
Private Sub Button1_Click(sender As System.Object, e As System.EventArgs) Handles
Button1.Click
SerialPort1.Write("1")                                    'ส่งหมายเลข'1'ไปยังบอร์ดArduino
End Sub

Private Sub Button2_Click(sender As System.Object, e As System.EventArgs) Handles
Button2.Click
SerialPort1.Write("2")                                    'ส่งหมายเลข'2'ไปยังบอร์ดArduino
End Sub

Private Sub Button3_Click(sender As System.Object, e As System.EventArgs) Handles
Button3.Click
SerialPort1.Write("3")                                    'ส่งหมายเลข'3'ไปยังบอร์ดArduino
End Sub

Private Sub Button4_Click(sender As System.Object, e As System.EventArgs) Handles
Button4.Click
SerialPort1.Write("4")                                    'ส่งหมายเลข'4'ไปยังบอร์ดArduino
End Sub

Private Sub Button5_Click(sender As System.Object, e As System.EventArgs) Handles
Button5.Click
SerialPort1.Write("5")                                    'ส่งหมายเลข'5'ไปยังบอร์ดArduino
End Sub

Private Sub Button6_Click(sender As System.Object, e As System.EventArgs) Handles
Button6.Click
SerialPort1.Write("6")                                    'ส่งหมายเลข'6'ไปยังบอร์ดArduino
End Sub

Private Sub Button7_Click(sender As System.Object, e As System.EventArgs) Handles
Button7.Click
SerialPort1.Write("7")                                    'ส่งหมายเลข'7'ไปยังบอร์ดArduino
End Sub

Private Sub Button8_Click(sender As System.Object, e As System.EventArgs) Handles
Button8.Click
SerialPort1.Write("8")                                    'ส่งหมายเลข'8'ไปยังบอร์ดArduino
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
End Sub

Private Sub Button9_Click(ByVal sender As System.Object, ByVal e As System.EventArgs)
Handles Button9.Click
SerialPort1.Write("9")                                    'ส่งหมายเลข'9'ไปยังบอร์ดArduino
End Sub

Private Sub Button10_Click(ByVal sender As System.Object, ByVal e As System.EventArgs)
Handles Button10.Click
SerialPort1.Write("a")                                    'ส่งหมายเลข'a'ไปยังบอร์ดArduino
End Sub

End Class
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- **โปรแกรมส่งข้อมูลโดยใช้วิธี open file ไปยังไมโครคอลโทรลเลอร์ เพื่อแสดงผลที่แผงไฟวงจร LED**

```
Imports System.IO
Imports System.IO.FileInfo
Imports System.Threading
Imports System.IO.Ports
Imports System.Windows.Forms.ComboBox


Public Class Form1
    Private Sub Form1_Load(ByVal sender As System.Object, ByVal e As System.EventArgs)
Handles MyBase.Load
        SerialPort1.Close()
    'คำสั่งนี้ใช้ปิด serial port
        SerialPort1.BaudRate = 9600
    'ความเร็วในการส่งข้อมูลของ Serial port
        SerialPort1.DataBits = 8
        SerialPort1.Parity = Parity.None
    'การตรวจสอบความถูกต้องของข้อมูลที่ส่งผ่าน Serial port จะใช้วิธีเช็ค Parity bit
        SerialPort1.StopBits = StopBits.One
    'เป็นการกำหนดจำนวนบิตที่ใช้ในการปิดท้ายข้อมูล ที่ทำการส่งในแต่ละไบต์ ก่อนที่จะทำการส่ง
    ข้อมูลในไบต์ต่อไป
        SerialPort1.Handshake = Handshake.None
        SerialPort1.Encoding = System.Text.Encoding.Default  'very important!

        SerialPort2.Close()
        SerialPort2.BaudRate = 9600
        SerialPort2.DataBits = 8
        SerialPort2.Parity = Parity.None
        SerialPort2.StopBits = StopBits.One
        SerialPort2.Handshake = Handshake.None
        SerialPort2.Encoding = System.Text.Encoding.Default  'very important!
    End Sub

    Private Sub Open_Click(ByVal sender As System.Object, ByVal e As System.EventArgs)
Handles Open.Click
        Dim comport1 As String
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        'ประกาศตัวแปร comport1 เป็น String
          comport1 = ComboBox1.Text
        'เป็นข้อความที่ถูกเลือกหรือส่งให้ ComboBox แล้วเก็บค่าที่ comport1
          SerialPort1.PortName() = comport1
        'เป็นการเลือก serial port name ที่ใช้ติดต่อระหว่างโปรแกรมกับ zigbee
          SerialPort1.Open()
        'คำสั่งนี้ใช้เปิด serial port

    End Sub

    Private Sub Close_Click(ByVal sender As System.Object, ByVal e As System.EventArgs)
Handles Close.Click
        SerialPort1.Close()
        'คำสั่งนี้ใช้ปิด serial port
    End Sub

    Private Sub Send_Click(ByVal sender As System.Object, ByVal e As System.EventArgs)
Handles Send.Click
        SerialPort1.Write(openfile.Text)
        'เขียนข้อมูลไปยังพอร์ตอนุกรมแสดงผลบัฟเฟอร์ จากการเปิดไฟล์
        SerialPort1.Write("b")
        'เขียนข้อมูลไปยังพอร์ตอนุกรมแสดงผลบัฟเฟอร์ เป็นค่า b
    End Sub

    Private Sub openfileBtn_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles openfileBtn.Click
        OpenFileDialog1.Filter = "jpegs|*.jpg|gifs|*.gif|Bitmaps|*bmp| txt|*.txt||*.*"
        'ได้รับหรือตั้งค่าสตริงที่ระบุรายละเอียดและ ประเภท ไฟล์ที่จะแสดงใน openfiledialog ได้
        Dim result As DialogResult = OpenFileDialog1.ShowDialog()
        'ประกาศตัวแปร result เป็น DialogResult โดยที่ให้ OpenFileDialog1.ShowDialog() เก็บค่าที่
        result

        If result = Windows.Forms.DialogResult.OK Then
            Dim path As String = OpenFileDialog1.FileName
            Dim fileSize As New FileInfo(path)
            Dim length As Long = fileSize.Length
        'ประกาศตัวแปร length เป็นชนิดข้อมูล long เพื่อเก็บค่าขนาดของไฟล์
            Length_TextBox.Text = length
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
'ค่า length ถูกแสดงผลไปที่ textbox เพื่อให้รู้ขนาดของไฟล์
        Try
            Dim text As String = File.ReadAllText(path)
            openfile.Text = text
        Catch ex As Exception
            Me.Text = "Error"
        End Try
        OpenFileDialog1.Reset()
    End If
End Sub
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้