

สำนักหอสมุดกลาง พระจอมเกล้าลาดกระบัง

ชุดควบคุมระบบแสงสว่างผ่านระบบ LAN
(IP Lighting)



โดย
นาย เกียรติกร แสงภูเขียว
นาย คุณาวุฒิ บุญมาก
นาย เฉลิมเกียรติ สายยศ
นาย นิวัฒน์ กุลครอง

๑/๗
๗ 852๕
9๕49

เลขหมู่.....
เลขทะเบียน..... 72737
วัน,เดือน,ปี..... 22 ส.ย. 2550

b. 11๓720๑๘
i.....

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต
สาขาวิชาวิศวกรรมไฟฟ้า
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2549

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ปีการศึกษา 2549

ชุดควบคุมระบบแสงสว่างผ่านระบบ LAN
(IP Lighting)



โดย
นาย เกียรติกร แสงภูเขียว
นาย คุณาวุฒิ บุญมาก
นาย เฉลิมเกียรติ สายยศ
นาย นิวัฒน์ กุลครอง

อาจารย์ที่ปรึกษา

ผศ.ดร. สุวัฒน์ กิตติรัตน์สัจจา

ดร. สมภาพ ผลไม้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ปริญญาโทปีการศึกษา 2549

ภาควิชาวิศวกรรมไฟฟ้า

คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เรื่อง ชุดควบคุมระบบแสงสว่างผ่านระบบ LAN

ผู้จัดทำ



1. นาย เกียรติกร แสงภูเขียว
2. นาย คุณาวุฒิ บุญมาก
3. นาย เฉลิมเกียรติ สายยศ
4. นาย นิวัฒน์ กุลครอง

.....*สุวัฒน์ กิตติรัตน์*.....อาจารย์ที่ปรึกษา
(ผศ.ดร. สุวัฒน์ กิตติรัตน์ สัจจา)

.....*สมภพ ผลไม้*.....อาจารย์ที่ปรึกษา
(ดร. สมภพ ผลไม้)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ชุดควบคุมระบบแสงสว่างผ่านระบบ LAN (IP Lighting)

นาย เกียรติกร	แสวงภูเขียว	
นาย คุณาวุฒิ	บุญมาก	
นาย เฉลิมเกียรติ	สายยศ	
นาย นิวัฒน์	กุลครอง	
ผศ.ดร. สุวัฒน์	กิตติรัตน์สัจจา	อาจารย์ที่ปรึกษา
ดร. สมภพ	ผลไม้	อาจารย์ที่ปรึกษา
ปีการศึกษา 2549		

บทคัดย่อ

ปัญญานิพนธ์ฉบับนี้นำเสนอเกี่ยวกับการควบคุมระบบแสงสว่างผ่านระบบ LAN ซึ่งประกอบด้วยโปรแกรมที่ใช้ในการควบคุมการทำงานระยะไกลซึ่งเขียนด้วยโปรแกรมวิซวลเบสิก การแปลงข้อมูลในระบบ LAN เป็น RS232 และชุดควบคุมโลคอลซึ่งควบคุมด้วย MCS-51 ชุดควบคุมโลคอลรับสัญญาณผ่าน RS232 จากโปรแกรมควบคุมแล้วส่งงานออกเอาต์พุต ชุดเอาต์พุต มี 3 ชุด คือชุดรีเลย์ และ ชุดขับสำหรับการหรี่ไฟสำหรับหลอด LED กำลังสูงจำนวน 2 ชุด ชุดรีเลย์ควบคุมการทำงานของหลอดไฟ 4 หลอดและสามารถนำไปประยุกต์ใช้ในงานอื่นได้อีก ชุดขับ LED ชุดที่หนึ่งควบคุมการทำงานของโคมหลอด LED สีขาว โดยสามารถปรับความสว่างได้ และชุดขับ LED อีกตัวหนึ่งทำการควบคุมการทำงานของโคมหลอด LED สามสี คือ สีแดง สีเขียว และ สีน้ำเงิน ซึ่งสามารถทำการผสมสีและตั้งสภาวะการทำงานได้หลากหลาย

IP Lighting

Kiattikorn	Sangphukhiaw	
Khunawut	Bunmak	
Chalermkiat	Saiyot	
Niwat	Kunklong	
Asst. Prof. Dr. Supat	Kittiratsatcha	Advisor
Dr. Somphob	Polmai	Advisor
2006		

ABSTRACT

This project presents LAN lighting control system. The system is consisting of a computer program written in Visual Basic, a LAN-to-RS232 conversion card, and a local control system based on MCS-51 microcontroller. Local control system receives command from remote program via LAN-to-RS232 conversion card, and then controls the level of light output of the lamp set according to the command received. There are three set of lamp designed for this experiment. The first lamp set has 4 lamps; each can be controlled "on" or "off" by its dedicated relay. The second set has one high-power white LED lamp which is dimmable from zero to 100 percent intensity. The third set is a three-color high-power LED lamp. Each color light intensity can be independently controlled to have color mixing effect.

กิตติกรรมประกาศ

ปริญญานิพนธ์ฉบับนี้สำเร็จได้ด้วยคำแนะนำและข้อคิดดีๆในการทำงานจากบุคคลหลายท่านดังต่อไปนี้

อันดับแรกคืออาจารย์ที่ปรึกษาทั้ง 2 ท่าน คือ ผศ.ดร.สุพัฒน์ กิตติรัตน์สัจจา และ ดร.สมภพ ผลไม้ ที่กรุณาช่วยเหลือและให้คำชี้แนะ เอาใจใส่ด้วยดีเสมอมา

ที่ลืมไม่ได้คือ พี่ๆนักศึกษาศรีอยุธยาโท ที่คอยช่วยเหลือ ให้คำปรึกษาและความเป็นกันเอง ทำให้บรรยากาศในการทำงานเป็นไปอย่างราบรื่น ต่อมาคือกลุ่มเพื่อนๆที่ร่วมทำโครงการในห้องเดียวกันที่มีส่วนทำให้บรรยากาศเป็นไปอย่างสนุกสนาน รวมไปถึงบุคลากรของภาควิศวกรรมไฟฟ้าทุกท่านที่ช่วยอำนวยความสะดวกเรื่องอุปกรณ์ เครื่องมือต่างๆด้วยดีเสมอมา

สุดท้ายขอขอบคุณ บิดา มารดา อันเป็นที่เคารพรักของข้าพเจ้าทุกคน ที่คอยให้กำลังใจในการทำงานข้าพเจ้าทุกคนขอระลึกถึงในพระคุณอันหาสิ่งดอบแทนมิได้จึงขอกราบขอบพระคุณเป็นอย่างสูงมา ณ ที่นี้

ผู้จัดทำ

นาย เกียรติกร	แสวงภูเขียว
นาย คุณาวุฒิ	บุญมาก
นาย เฉลิมเกียรติ	สายยศ
นาย นิวัฒน์	กุลครอง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่นุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ

	หน้า
บทคัดย่อ.....	I
ABSTRACT.....	II
สารบัญรูป.....	III
สารบัญตาราง.....	VI
บทที่ 1 บทนำ.....	1
1.1 ความสำคัญของปัญหา.....	1
1.2 วัตถุประสงค์ของการวิจัย.....	1
1.3 ขอบเขตและข้อกำหนดในการวิจัย.....	2
1.4 ขั้นตอนและวิธีการดำเนินงาน.....	2
1.5 ประโยชน์ที่ได้รับจากการวิจัย.....	2
บทที่ 2 ทฤษฎีที่เกี่ยวข้อง.....	3
2.1 โปรแกรมวิซวล เบสิก (Visual Basic).....	3
2.2 ทฤษฎีเบื้องต้นเกี่ยวกับระบบ LAN โปรโตคอล.....	4
2.2.1 แนวคิดด้านสื่อสารข้อมูล.....	5
2.2.2 การทำงานของระดับโปรโตคอลใน LAN.....	5
2.2.3 โปรโตคอลชั้นเน็ตเวิร์ค.....	7
2.3 การใช้งาน ET-XPORT V1 (RS232 / LAN Converter).....	8
2.3.1 คุณสมบัติของ ET-XPORT V1.....	9
2.3.2 โครงสร้างของ ET-XPORT V1.....	10
2.3.3 การกำหนดค่าการใช้งาน ET-XPORT V1 กับระบบเครือข่าย Ethernet LAN.....	10
2.4 การเชื่อมต่อสัญญาณด้าน Ethernet LAN (Ethernet Interface).....	11
2.4.1 สายสัญญาณ Ethernet LAN โดยตรงกับเครื่องคอมพิวเตอร์ PC (Peer to Peer).....	12
2.4.2 สายสัญญาณ Ethernet LAN ผ่าน HUB.....	13
2.5 ทฤษฎีการเขียนโปรแกรมส่งข้อมูลผ่าน TCP/IP ด้วย Visual Basic.....	14
2.6 ทฤษฎีการเชื่อมต่อ Serial Port (RS232).....	17
2.6.1 รูปแบบของข้อมูลอนุกรม.....	17
2.6.2 การเชื่อมต่อสัญญาณด้าน RS232.....	18
2.7 บอร์ด Controller.....	19

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ (ต่อ)

	หน้า
2.7.1 คุณสมบัติของบอร์ด.....	20
2.7.2 โครงสร้างบอร์ด ET-BASE51 V2.0 (AT89C51ED2).....	20
2.7.3 ขั้วต่อสัญญาณต่างๆ.....	21
2.7.4 วงจรของบอร์ด ET BASE V2.0 (AT89C51ED).....	23
บทที่ 3 หลักการทำงานและการออกแบบวงจร.....	24
3.1 การออกแบบโดยรวมของโครงการ.....	24
3.2 การออกแบบในส่วนของโปรแกรม Visual Basic.....	25
3.2.1 Form.....	25
3.2.2 Frame.....	26
3.2.3 OptionButton.....	27
3.2.4 TextBox.....	28
3.2.5 ComboBox.....	29
3.2.6 Label.....	29
3.2.7 CommandButton.....	30
3.2.8 HScrollBar.....	31
3.2.9 Timer.....	32
3.2.10 Winsock.....	32
3.2.11 หน้าต่างโปรแกรมที่ทำการออกแบบ.....	33
3.3 การกำหนดค่า IP Address ให้กับ ET-XPORT V1.....	34
3.4 การทดสอบการเชื่อมต่อกับเครือข่าย Network ด้วยการ Ping.....	37
3.5 การกำหนดค่า Configuration ให้กับ ET-XPORT V1 ด้วยโปรแกรม DeviceInstaller.....	39
3.6 การออกแบบโปรแกรม Microcontroller.....	41
บทที่ 4 ผลการทดลอง.....	42
4.1 การทดสอบการทำงานของชุด เปิด ปิด หลอดไฟ.....	42
4.2 การทดสอบการทำงานของชุดปรับความสว่างของ LED.....	42
4.3 การทำงานของชุดโคม LED ที่สามารถปรับสีได้.....	44
4.4 การทดลองการทำงานจริง.....	45

สารบัญ (ต่อ)

	หน้า
บทที่ 5 สรุปผลและวิจารณ์โครงการ.....	54
5.1 สรุปโครงการ.....	54
5.2 ปัญหา อุปสรรค และการแก้ไข.....	54
5.3 แนวทางการพัฒนา.....	55

ภาคผนวก

กิตติกรรมประกาศ

เอกสารอ้างอิง



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูป

รูปที่	หน้า
2.1	หน้าต่างพื้นฐานของโปรแกรมวิซวล เบสิก ให้เลือกโหมดการทำงาน..... 3
2.2	หน้าต่างพร้อมอุปกรณ์และเครื่องมือพื้นฐานของโปรแกรมวิซวล เบสิก..... 4
2.3	การแบ่งระดับโปรโตคอล..... 5
2.4	ข้อกำหนดทางฟิสิกส์..... 6
2.5	ชุดอุปกรณ์ที่ใช้สำหรับการแปลงข้อมูล..... 8
2.6	แสดงแผนผังโครงสร้างของ ET-XPORT V1..... 10
2.7	แสดงลักษณะของหัว RJ45 ที่ใช้เชื่อมต่อสัญญาณกับเครือข่าย Ethernet LAN..... 11
2.8	แสดงการเข้าสายสำหรับเชื่อมต่อ ET-XPORT V1 โดยตรงกับ Ethernet Port ของเครื่อง PC..... 13
2.9	แสดงการเข้าสายสำหรับเชื่อมต่อ ET-XPORT V1 กับเครือข่าย ผ่าน HUB..... 14
2.10	การเรียกใช้งาน Winsock Control..... 15
2.11	แบบสัญญาณไฟฟ้าของข้อมูล..... 17
2.12	แผนภาพสัญญาณเวลาของข้อมูลแบบอนุกรมจำนวน 8 บิต พร้อมด้วยบิตเริ่มต้นบิตพาริตี และ บิตสุดท้าย..... 18
2.13	แสดงการจัดเรียงสัญญาณ RS232 (DB9 ตัวเมีย) ของ ET-XPORT V1..... 18
2.14	บอร์ด ET-BASE51 V2.0 (AT89C51ED2)..... 20
2.15	การเชื่อมต่อสัญญาณ RS232..... 21
2.16	การจัดเรียงสัญญาณของพอร์ต I/O ต่าง ๆ ของบอร์ด ET-BASE51 V2.0..... 22
2.17	วงจรของบอร์ด ET-BASE51 V2.0..... 23
3.1	ไดอะแกรมของโครงการทั้งหมด..... 24
3.2	แถบเครื่องมือ (Tool Box) และอุปกรณ์พื้นฐานบนหน้าต่างโปรแกรม..... 25
3.3	ฟอร์มที่ปรากฏขึ้นมาเมื่อเริ่มเปิดโปรแกรม..... 26
3.4	สัญลักษณ์คอนโทรล Frame..... 26
3.5	สัญลักษณ์คอนโทรล OptionButton..... 27
3.6	การเลือกสีที่จะทำการควบคุม..... 28
3.7	สัญลักษณ์คอนโทรล TextBox..... 28
3.8	TextBox สำหรับใส่ค่า IP Address และ Port..... 28
3.9	สัญลักษณ์คอนโทรล ComboBox..... 29
3.10	สัญลักษณ์คอนโทรล Label..... 30
3.11	สัญลักษณ์คอนโทรล CommandButton..... 30
3.12	ปุ่ม Connect โดยใช้คอนโทรล CommandButton..... 31

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูป (ต่อ)

รูปที่	หน้า
3.13 สัญลักษณ์คอนโทรล HScrollBar.....	31
3.14 การปรับความสว่างโดยใช้คอนโทรล HScrollBar.....	32
3.15 สัญลักษณ์คอนโทรล Timer.....	32
3.16 สัญลักษณ์คอนโทรล Winsock.....	33
3.17 หน้าต่างโปรแกรมที่ทำการออกแบบ.....	33
3.18 แสดงขั้นตอนการ Assign IP.....	34
3.19 แสดงขั้นตอนการกรอกค่า MAC Address.....	35
3.20 แสดงขั้นตอนการ Assign a specific IP address.....	35
3.21 แสดงขั้นตอนการกรอกค่า IP Address, Subnet Mark และ Gateway.....	36
3.22 แสดงขั้นตอนการ Assign.....	36
3.23 สิ้นสุดขั้นตอนการกำหนดค่า IP Address ให้กับ ET-XPORT V1.....	37
3.24 แสดงการตรวจสอบสถานะการเชื่อมต่อ.....	37
3.25 แสดงขั้นตอนการทดสอบการเชื่อมต่อกับเครือข่าย Network ด้วยการ Ping.....	38
3.26 แสดงผลการทดสอบการเชื่อมต่อกับเครือข่าย.....	38
3.27 แสดงหน้าต่างการกำหนดค่า Configuration ให้กับ ET-XPORT V1 ด้วยโปรแกรม DeviceInstaller.....	39
3.28 กำหนดชื่อให้กับ ET-XPORT V1.....	40
3.29 แสดงการกำหนดค่าการทำงานทางด้าน RS232.....	40
3.30 Flow Chart การทำงานของ Microcontroller.....	41
4.1 ที่การทำงาน 20 % ของ Program Control.....	42
4.2 ที่การทำงาน 50 % ของ Program Control.....	43
4.3 ที่การทำงาน 80 % ของ Program Control.....	43
4.4 การทำงานที่สีแดง 20 % สีเขียว 50 % สีน้ำเงิน 80 %.....	44
4.5 การทำงานที่สีแดง 50 % สีเขียว 80 % สีน้ำเงิน 20 %.....	44
4.6 การทำงานที่สีแดง 50 % สีเขียว 80 % สีน้ำเงิน 20 %.....	45
4.7 อุปกรณ์การทดลองทั้งหมด.....	45
4.8 การสั่งการทำงานของหลอด 1 หลอด.....	46
4.9 การสั่งการทำงานของหลอด 2 หลอด.....	46
4.10 การสั่งการทำงานของหลอด 3 หลอด.....	47
4.11 การสั่งการทำงานของหลอด 4 หลอด.....	47
4.12 การสั่งการทำงานของโคมหลอดขาว 0%.....	48

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูป (ต่อ)

รูปที่	หน้า
4.13 การสั่งการทำงานของโคมพลอดขาว 10%.....	48
4.14 การสั่งการทำงานของโคมพลอดขาว 100%.....	49
4.15 การสั่งการทำงานของโคมพลอด 3 สี (สีแดง).....	49
4.16 การสั่งการทำงานของโคมพลอด 3 สี (สีเขียว).....	50
4.17 การสั่งการทำงานของโคมพลอด 3 สี (สีน้ำเงิน).....	50
4.18 การสั่งการทำงานของโคมพลอด 3 สี (ผสมสี).....	51
4.19 การสั่งการทำงานของโคมพลอด 3 สี (ผสมสี).....	51
4.20 การสั่งการทำงานของโคมพลอด 3 สี (ผสมสี).....	52
4.21 การสั่งการทำงานของโคมพลอด 3 สี (ผสมสี).....	52
4.22 การสั่งการทำงานของโคมพลอด 3 สี (ผสมสี).....	53
4.23 การสั่งการทำงานของโคมพลอด 3 สี (ผสมสี).....	53



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญตาราง

ตารางที่	หน้า
2.1 แสดงการจัดเรียงสัญญาณของ RJ45.....	12
2.2 แสดงผังการต่อสายสัญญาณ RS232 ระหว่าง คอมพิวเตอร์ PC และ ET-XPORT.....	19



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 1

บทนำ

ในปัจจุบันได้มีการนำเทคโนโลยีที่ทันสมัยเข้ามาใช้มากขึ้น ทั้งด้านการสื่อสาร อุปกรณ์อิเล็กทรอนิกส์ต่าง ๆ ซึ่งได้มีการพัฒนามาจากในอดีต โดยเฉพาะทางด้านการสื่อสาร ในปัจจุบัน ระบบ LAN ได้เป็นสื่อที่รู้จักกันอย่างแพร่หลายและเป็นสื่อที่ได้มีการพัฒนาอย่างต่อเนื่องควบคู่ไปกับการดำเนินชีวิตของมนุษย์ในปัจจุบัน

ทางกลุ่มได้รับหัวข้อโครงการเกี่ยวกับการควบคุมระบบไฟฟ้าผ่านระบบเครือข่าย LAN ซึ่งเป็นอีกแนวทางหนึ่งในการนำเทคโนโลยีสมัยใหม่มาประยุกต์ใช้กับการควบคุมการทำงานของอุปกรณ์ไฟฟ้าภายในที่พักอาศัย และอาคารสำนักงาน โดยจะเพิ่มความสะดวกสบายให้แก่ผู้ใช้งาน ตามความต้องการ ซึ่งในอดีตการควบคุมอุปกรณ์ไฟฟ้าแต่ละชนิด จะต้องควบคุมด้วยมือโดยตรงในระยะทางที่ไกลกับอุปกรณ์ แต่ในปัจจุบันมีการควบคุมหลายรูปแบบ แต่ในโครงการนี้จะใช้การควบคุมระบบไฟฟ้าผ่านเครือข่าย LAN ซึ่งจะสามารถพัฒนาการใช้งานให้มีความหลากหลายมากขึ้นอีกในอนาคต

1.1 ความสำคัญของปัญหา

การควบคุมอุปกรณ์ไฟฟ้าในอาคารที่พักอาศัย แต่ก่อนจะต้องทำการควบคุมการเปิด-ปิด ด้วยมือของผู้ใช้งานในอุปกรณ์ไฟฟ้านั้น ๆ โดยตรง และเมื่อผู้ใช้งานออกจากที่ทำงานหรือที่พักอาศัย อาจเกิดปัญหาการหลงลืมเปิด-ปิดอุปกรณ์ไฟฟ้าทิ้งไว้ ทำให้เกิดการสูญเสียและสิ้นเปลืองพลังงานไฟฟ้าโดยใช่เหตุ และถ้าอยู่ในที่ห่างไกลจากสถานที่ทำงานหรือที่พักอาศัย ทำให้ผู้ใช้งานต้องเสียเวลาในการเดินทางกลับมาเพื่อปิดด้วยตัวเอง

อุปกรณ์ชุดนี้จะเป็นชุดทดลอง เพื่อการพัฒนาให้สามารถควบคุมอุปกรณ์ไฟฟ้าได้ในระยะไกล โดยควบคุมอุปกรณ์ไฟฟ้าผ่านทางเครือข่าย LAN ซึ่งเราจะสามารถควบคุมการทำงานของอุปกรณ์ไฟฟ้าได้โดยไม่ต้องเสียเวลาในการเดินทาง ทำให้มีความสะดวกสบายในการใช้งานระบบไฟฟ้ามากขึ้น อีกทั้งยังสามารถทำให้เกิดการหลากหลายในการใช้งานมากยิ่งขึ้น

1.2 วัตถุประสงค์ของการวิจัย

1. ศึกษาการทำงานของเครือข่ายระบบ LAN
2. ศึกษาการเขียนโปรแกรม Visual Basic 6.0 (Software ใช้ในการควบคุม)
3. ศึกษาการเขียนโปรแกรมสำหรับ Microcontroller ที่ใช้ในการ Control Hardware
4. นำความรู้ที่ได้ศึกษามาประยุกต์ใช้สร้างชุดควบคุมอุปกรณ์ไฟฟ้า
5. เพื่อความสะดวกในการใช้งานและควบคุมการทำงานของเครื่องใช้ไฟฟ้าต่าง ในระยะทางไกล โดยผ่านเครือข่ายของระบบ LAN

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1.3 ขอบเขตและข้อกำหนดของโครงการ

ขอบเขตของโครงการคือการเขียนโปรแกรมการควบคุมการทำงานของ LED ผ่านออกทางระบบ LAN ของคอมพิวเตอร์ แล้วใช้อุปกรณ์อิเล็กทรอนิกส์สำเร็จรูป ET-XPORT (อุปกรณ์แปลงสัญญาณ LAN ออกมาเป็นสัญญาณ RS232) เพื่อส่งข้อมูลให้กับชุด Microcontroller เพื่อสั่งการทำงานของชุด LED โดยใช้สัญญาณ PWM และยังสามารถสั่งเปิด ปิด อุปกรณ์ไฟฟ้าทั่วไปด้วย

1.4 ขั้นตอนและวิธีการดำเนินงาน

1. เลือกหัวข้อในการดำเนินงาน
2. ศึกษาวัตถุประสงค์และขอบเขตการทำงานของโครงการ
3. ศึกษาและค้นคว้าข้อมูลที่เกี่ยวข้องกับโครงการ
4. เขียนโปรแกรมการควบคุมความสว่างโดยใช้โปรแกรม Visual Basic 6.0
5. ออกแบบและสร้างวงจรทดลองเพื่อศึกษาการทำงาน
6. ศึกษาผลการทดลองที่ได้และนำผลการทดลองมาวิเคราะห์ผลการทดลอง
7. สรุปผลการทดลอง รวมถึงปัญหาและอุปสรรคในการทำงาน

1.5 ประโยชน์ที่คาดว่าจะได้รับ

1. มีความรู้ ความเข้าใจ เกี่ยวกับระบบ LAN มากขึ้น
2. มีความรู้ ความเข้าใจ เกี่ยวกับการเขียนโปรแกรม Microcontroller มากขึ้น
3. มีความรวดเร็วในการควบคุมระบบไฟฟ้าภายในตึกและอาคารสูง ๆ
4. ความสะดวกสบายต่อผู้อยู่อาศัย
5. สามารถนำไปใช้ประยุกต์กับการควบคุมอุปกรณ์ไฟฟ้าอื่น ๆ ได้
6. เสริมสร้างทักษะในการปฏิบัติงานจริงเพื่อให้เกิดชิ้นงานที่ใช้งานได้
7. ฝึกการทำงานร่วมกันเป็นกลุ่ม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 2 ทฤษฎีที่เกี่ยวข้อง

2.1 โปรแกรมวิซวล เบสิก (Visual Basic)

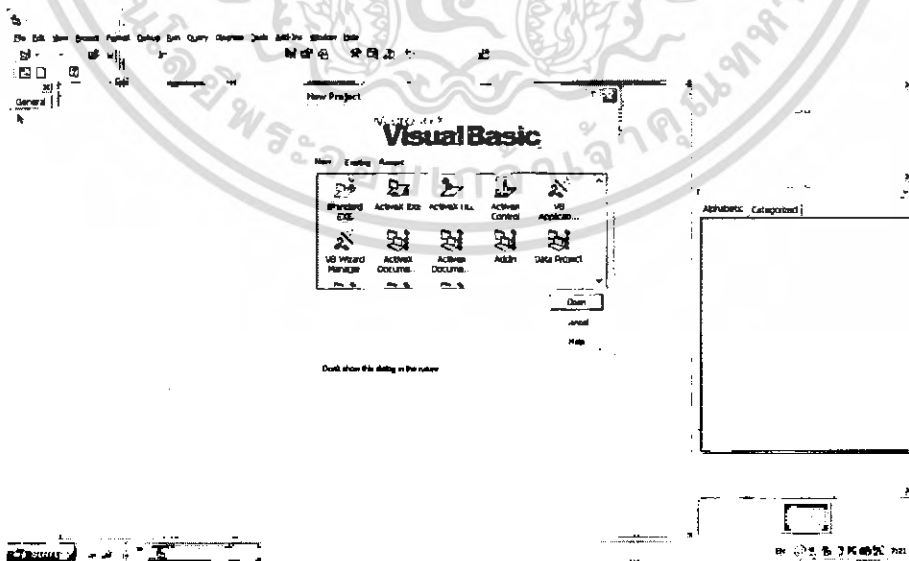
โปรแกรมวิซวล เบสิก เป็นโปรแกรมที่ใช้กันอย่างกว้างขวางในปัจจุบัน สามารถรองรับกับโปรแกรมต่าง ๆ ของบริษัทไมโครซอฟท์ได้เป็นอย่างดี แม้แต่ไมโครซอฟท์ออฟฟิศเองก็ยังคงใช้วิซวล เบสิก สำหรับเขียนโปรแกรมต่อเติมเพื่อการทำงานในระดับสูง หรือแก้ไขส่วนที่บกพร่องเล็ก ๆ น้อย ๆ ของโปรแกรม

ในส่วนของภาษาที่ใช้ในการเขียนโปรแกรมนั้น พัฒนามาจากภาษาเบสิก (Basic) ซึ่งเป็นภาษาที่มีมาแต่ดั้งเดิมในสมัยที่ยังใช้ ดอส (Dos) ในการพัฒนาโปรแกรม ในตอนนี้มีภาษาเบสิกหลายตัว อาทิ QBasic, MBasic เป็นต้น

ส่วนสำคัญที่ทำให้ วิซวล เบสิก กลายมาเป็นโปรแกรมที่นิยมใช้กันมากในปัจจุบันก็เพราะตัวภาษาเบสิกถูกนำมาใช้เขียนโปรแกรมกับวินโดวส์ ทำให้ผู้ใช้งานสามารถใช้โปรแกรมได้อย่างสะดวก และมีความหลากหลายมากขึ้น

ในโครงการนี้เลือกใช้โปรแกรมวิซวล เบสิก เวอร์ชัน 6.0 ซึ่งเป็นเวอร์ชันที่นิยมใช้กันมากในปัจจุบัน ทั้งในงานด้านกราฟฟิคต่าง ๆ รวมไปถึงการนำมาใช้ในการควบคุมฮาร์ดแวร์หรืออุปกรณ์ภายนอก ในงานอุตสาหกรรมประเภทต่าง ๆ อีกด้วย

นอกจากนั้นยังรวมถึงแหล่งความรู้ทางด้านบุคคลหรือเอกสาร หนังสือ ซึ่งมีอยู่มากมาย สำหรับเวอร์ชัน 6.0 นี้ ทำให้เมื่อเกิดปัญหาจะสามารถแก้ไขได้โดยไม่ลำบากมากนัก รวมถึงการเพิ่มทักษะในการเขียนโค้ดต่าง ๆ ให้การควบคุมของระบบงานออกมาดีมากขึ้น



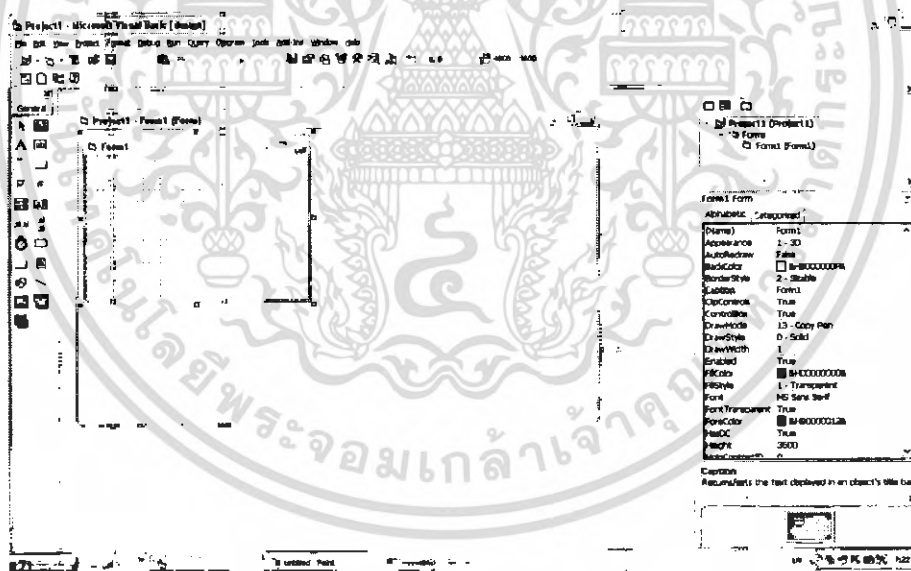
รูปที่ 2.1 หน้าต่างพื้นฐานของโปรแกรมวิซวล เบสิก ให้เลือกโหมดการทำงาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

คุณสมบัติเด่น ของโปรแกรมวิชวล เบสิก

โปรแกรมแล้ว วิซวล เบสิก มีข้อดีเด่น ๆ ที่พอสรุปได้ดังนี้

1. ใช้เขียนโปรแกรมได้ทั้งวินโดวส์และเว็บ โปรแกรมวิชวล เบสิก ในเวอร์ชันหลัง ๆ ได้รับการพัฒนาให้สามารถเขียนโปรแกรมให้ทำงานได้ทั้งบนวินโดวส์และบนอินเทอร์เน็ต โดยที่ผู้สนใจเขียนโปรแกรมบนวินโดวส์สามารถประยุกต์ใช้ความรู้ที่ตนมีอยู่ในการพัฒนาเว็บแอปพลิเคชันได้ง่าย โดยสามารถเรียนรู้ VB Script ซึ่งเป็นภาษาในตระกูลวิชวล เบสิก ได้อย่างรวดเร็ว
2. การแก้ไขโปรแกรมที่เขียนขึ้นสามารถทำได้ง่าย ในบางครั้งการเขียนโปรแกรมอาจจะมีข้อผิดพลาด การแก้ไขหรือปรับปรุงในภายหลังของวิชวล เบสิก นั้นทำได้ง่ายโดยมีเครื่องมือต่าง ๆ คอยช่วยเหลืออยู่
3. มีเอกสารให้ศึกษาอ้างอิงมาก จากที่กล่าวมาแล้วว่าโปรแกรมวิชวล เบสิก เป็นที่นิยมใช้กันมากในปัจจุบัน เมื่อมีปัญหาหรือข้อสงสัยเกิดขึ้นจะหาคำตอบได้ง่ายจากผู้มีความรู้ รวมทั้งเอกสารในรูปแบบต่าง ๆ รวมถึงจากเว็บไซต์ของไมโครซอฟท์และเว็บไซต์อื่น ๆ ด้วย
4. เป็นโปรแกรมที่มีอนาคต วิซวล เบสิก นั้นเป็นภาษาเขียนโปรแกรมที่ไม่โครซอฟท์ถือว่าเป็นภาษาที่สำคัญมาก จึงถูกพัฒนาอย่างต่อเนื่อง คาดว่าเวอร์ชันใหม่ ๆ จะถูกเพิ่มความสามารถอย่างมาก ทั้งตัวภาษาและตัวผลงานที่ได้จากการเขียนโปรแกรม



รูปที่ 2.2 หน้าต่างพร้อมอุปกรณ์และเครื่องมือพื้นฐานของโปรแกรมวิชวล เบสิก

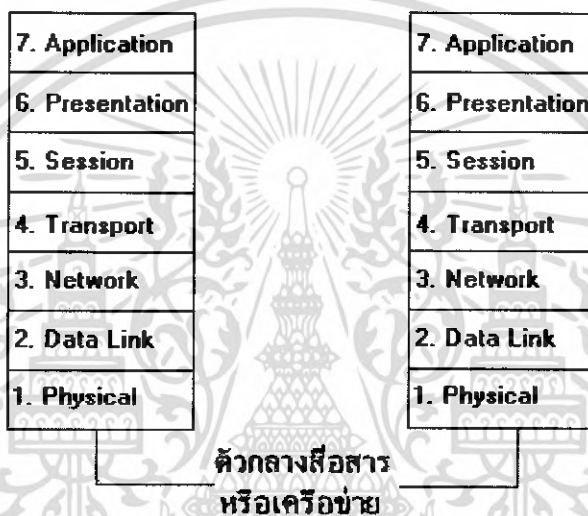
2.2 ทฤษฎีเบื้องต้นเกี่ยวกับระบบ LAN โปรโตคอล

โปรโตคอล (Protocol) คือระเบียบพิธีการในการติดต่อสื่อสาร เมื่อมาใช้กับเทคโนโลยีสื่อสารโทรคมนาคม จึงหมายถึงขั้นตอนการติดต่อสื่อสาร ซึ่งรวมถึง กฎ ระเบียบ และข้อกำหนดต่าง ๆ รวมถึงมาตรฐานที่ใช้ เพื่อให้ตัวรับและตัวส่งสามารถดำเนินกิจกรรมทางด้านสื่อสารได้สำเร็จ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.2.1 แนวคิดด้านสื่อสารข้อมูล

หัวใจในการสื่อสารข้อมูลอยู่ที่ว่าจะทำอย่างไรให้อุปกรณ์สื่อสารต่าง ๆ สื่อสารกันได้ อย่างอัตโนมัติ โดยเน้นการสื่อสารที่แตกต่างกันทางด้านเครื่องมือ อุปกรณ์และวิธีการต่าง ๆ เช่น คอมพิวเตอร์เมนเฟรมยี่ห้อหนึ่ง ติดต่อผ่านข่ายสื่อสารไปยังเครื่องคอมพิวเตอร์อีกยี่ห้อหนึ่ง โดยมีผลิตภัณฑ์ที่เชื่อมโยงในระบบสื่อสารที่มาจากหลายบริษัทผู้ผลิต ด้วยแนวคิดนี้ องค์กรว่า ด้วยเครื่องมาตรฐานระหว่างประเทศ หรือที่รู้จักกันในนาม ISO จึงได้วางมาตรฐานโปรโตคอลไว้ เป็นระดับ เพื่อให้การสื่อสารต่าง ๆ ยึดหลักการนี้และเรียกมาตรฐานโปรโตคอลนี้ว่า OSI PROTOCOL โดยวางเป็นระดับ 7 ชั้น



โมเดลมาตรฐานระดับโปรโตคอลตาม OSI

รูปที่ 2.3 การแบ่งระดับโปรโตคอล

การวางมาตรฐานโปรโตคอลต่าง ๆ ของเครือข่าย LAN จะอยู่ในระดับล่าง 2 ระดับ เท่านั้น โดยเน้นที่รูปร่างลักษณะของอุปกรณ์ รวมถึงรูปแบบสัญญาณไฟฟ้าที่ส่งรับกันโดย มาตรฐานโปรโตคอล ส่วนนี้จะกำหนดในระดับ 1 (Physical) และวิธีการจะทำให้ข้อมูลข่าวสาร จากอุปกรณ์หนึ่งส่งไปยังอีกอุปกรณ์หนึ่งภายในเครือข่ายเดียวกัน อยู่ในโปรโตคอลระดับ 2 เรียกว่า "ระดับดาต้าลิงค์ (Data Link)

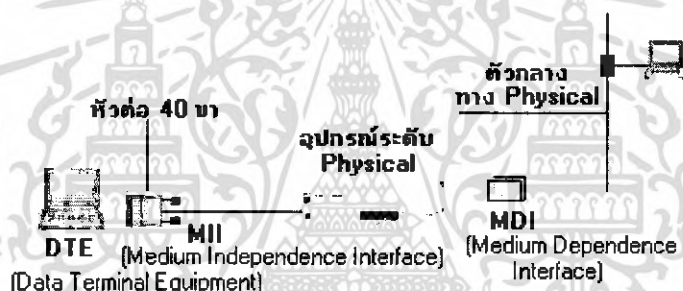
2.2.2 การทำงานของระดับโปรโตคอลใน LAN

ระบบ LAN ที่นิยมและแพร่หลายในปัจจุบัน ได้แก่ Ethernet, Token Ring และ FDDI โปรโตคอลที่ใช้ประกอบเป็น LAN ตามมาตรฐานข้อกำหนด จึงจัดอยู่ในระดับโปรโตคอลระดับ 1 และ 2 เท่านั้น

อีเทอร์เน็ต (Ethernet) เป็น LAN ที่มีผู้นิยมใช้กันมาก อีเทอร์เน็ตมีโปรโตคอลใน เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ระดับชั้นฟิสิคัล (Physical) ได้หลายรูปแบบ ตามสภาพความเร็วของการรับส่งข้อมูล รูปแบบสัญญาณและตัวกลางที่ใช้รับส่ง การกำหนดชื่อของ LAN แบบนี้ใช้วิธีการกำหนดเป็น XXBASEY เมื่อ XX คือความเร็ว BASE คือวิธีการส่งสัญญาณเป็นแบบ Digital Baseband ส่วน Y คือตัวกลางที่ใช้ส่งสัญญาณ เช่น 10BASE2 หมายถึงส่งความเร็ว 10 เมกะบิต แบบ Thin Ethernet ตัวกลางเป็นสายโคแอกเซียล 10BASE-T หมายถึงส่งความเร็ว 10 เมกะบิต แบบสาย UTP และถ้า 10BASE-FL ก็จะเป็นการใช้สายเส้นใยแก้วนำแสง

สัญญาณทางไฟฟ้าของอีเทอร์เน็ตเป็นแบบดิจิทัล จึงทำให้มีข้อจำกัดในเรื่องระยะทางที่ใช้ระเบียบข้อกำหนดเหล่านี้จึงอยู่ในกลุ่มโปรโตคอลระดับฟิสิคัล ส่วนในระดับโปรโตคอลดาต้าลิงค์เป็นวิธีการกำหนดแอดเดรสระหว่างกันในเครือข่าย ซึ่งแต่ละสถานีจะมีแอดเดรสเป็นตัวเลขขนาด 48 บิต การรับส่งเป็นการสร้างข้อมูลเป็นแพ็กเก็ตเรียกว่า "เฟรม" การส่งข้อมูลมีวิธีการใส่ข้อมูลแอดเดรสต้นทางและปลายทางและส่งกระจายออกไป ผู้รับจะตรวจสอบแอดเดรสของเฟรมถ้าตรงกับแอดเดรสตนก็จะรับข้อมูลเข้ามา



ข้อกำหนดทาง Physical ของ 100BASE-T

รูปที่ 2.4 ข้อกำหนดทางฟิสิคัล

FDDI เป็น LAN อีกชนิดหนึ่งที่ใช้เส้นใยแก้วนำแสงเป็นตัวกลางมีความเร็วในการรับส่ง 100 เมกะบิตต่อวินาที รูปแบบของเครือข่ายเป็นแบบวงแหวน การรับส่งภายในวงแหวนใช้โปรโตคอลแบบโทเก็นพาสซิง (Token Passing)

โทเก็นริง (Token Ring) ระบบ LAN ที่ใช้โครงสร้างเชื่อมโยงแบบวงแหวน แต่ใช้ตัวกลางเป็นสาย UTP การรับส่งสัญญาณเป็นแบบ Digital Baseband ความเร็วในการรับส่งมีทั้งแบบ 4 เมกะบิตต่อวินาที และ 16 เมกะบิตต่อวินาที

การกำหนดโปรโตคอลใน FDDI และ Token Ring ในระดับดาต้าลิงค์ ใช้รูปแบบข้อมูลเป็นเฟรม อุปกรณ์แต่ละตัวมีแอดเดรสประจำ การรับส่งข้อมูล ส่งต่อตามลำดับตามเส้นทางของสายต่อที่เป็นวงแหวน ตัวรับจะตรวจสอบแอดเดรส ซึ่งตัวตรงกับของตนก็จะคัดลอกข้อมูลขึ้นมาแล้วตอบรับว่าได้รับข้อมูลนั้นแล้ว จะเห็นได้ชัดว่า โปรโตคอลของ LAN ใน 2 ระดับล่าง เป็นการสื่อสารกันในกลุ่มของตนเอง ภายใต้กลุ่ม LAN นั้น ๆ เท่านั้น เช่น ถ้าเป็น Ethernet ก็จะสื่อสาร

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่นับญาติเห็นไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กันในอุปกรณ์ที่ต่ออยู่ในกลุ่มนั้นเท่านั้น เมื่อนำ LAN ต่างกลุ่มมาต่อเชื่อมรวมกัน การเชื่อมรวมกันนี้อาจเป็น LAN ที่ใช้โปรโตคอลเหมือนกัน หรือต่างกันก็ได้ เช่น นำ Ethernet มาเชื่อมต่อกับ Ethernet หรือ Ethernet กับ Token Ring การเชื่อมต่อระหว่าง LAN ด้วยกันนี้จำเป็นต้องมีโปรโตคอล ช่วยในการติดต่อระหว่างกัน โปรโตคอลในระดับนี้จึงอยู่ในชั้นระดับสามคือ โปรโตคอลชั้นเน็ตเวิร์ค

2.2.3 โปรโตคอลชั้นเน็ตเวิร์ค

ในระดับสามนี้ทำหน้าที่เชื่อมโยงระหว่างเครือข่ายย่อย เราอาจเรียกโปรโตคอลนี้ว่า เราตั้งโปรโตคอล (Routing Protocol) การกำหนดเส้นทางนี้จะต้องวางมาตรฐานกลางสำหรับการเชื่อมโยงอุปกรณ์ ซึ่งมาจากระดับล่างหลาย ๆ มาตรฐาน วิธีการหนึ่งที่ยอมรับคือ การกำหนดแอดเดรสของอุปกรณ์ระดับล่างใหม่ และให้แอดเดรสเป็นมาตรฐานกลาง เช่น การใช้โปรโตคอลอินเทอร์เน็ต (IP) ทุกอุปกรณ์มีแอดเดรสของตนเองมีการสร้างรูปแบบฟอร์แมตข้อมูลใหม่ที่เรียกว่า แพ็กเก็ต (Packet) ดังนั้น โปรโตคอลในระดับนี้จึงรับส่งข้อมูลกันเป็นแพ็กเก็ต ทุกแพ็กเก็ตมีการกำหนดแอดเดรสต้นทางและปลายทางโดยไม่ต้องคำนึงว่าระดับล่างที่ใช้คืออะไร

อุปกรณ์ที่ทำหน้าที่รับส่ง และรับรู้โปรโตคอลในระดับเน็ตเวิร์คนี้จะทำหน้าที่เป็นแปลงแพ็กเก็ตให้เข้าสู่เฟรมข้อมูลในระดับสอง และรับเฟรมข้อมูลระดับสองเปลี่ยนมาเป็นแพ็กเก็ตในระดับสามเช่นกัน ข้อเด่นในที่นี้ คือ ทำให้สามารถเชื่อม LAN ทุกมาตรฐานเข้าด้วยกันได้ ในระดับนี้ยังมีมาตรฐานโปรโตคอลอื่น ๆ เช่น IPX ของบริษัทเน็ตแวร์ เป็นต้น

ลองนึกเลยต่อไปว่า ขณะที่เราใช้โปรแกรมวินโดวส์ 95 เป็นเครื่องไคลเอนต์ (Client) ต่อเข้าสู่อินเทอร์เน็ตเชื่อมไปยังเครื่องให้บริการ (เซิร์ฟเวอร์) เครื่องใดเครื่องหนึ่ง นั้นหมายความว่า เราเชื่อมกันในระดับ 3 คือใช้ IP โปรโตคอล ทำให้ไม่ต้องคำนึงว่าทางฝ่ายไคลเอนต์หรือเซิร์ฟเวอร์ใช้ LAN แบบใด

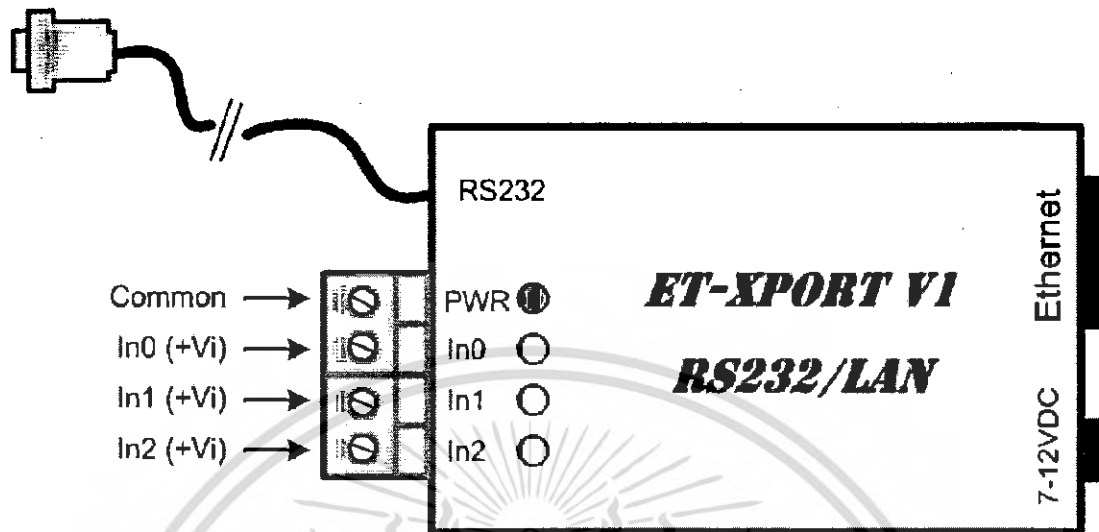
เครื่องไคลเอนต์ที่ใช้วินโดวส์ 95 ทำให้สามารถเปิดงานได้หลาย ๆ วินโดวส์พร้อมกันได้ ดังนั้นในเครื่องหนึ่งมีแอดเดรสในระดับสามตัวเดียว เชื่อมไปยังเซิร์ฟเวอร์ที่มีแอดเดรสในระดับสามตัวเดียวเช่นกัน แต่เปิดงานหลายงานได้ ดังนั้นจึงจำเป็นต้องสร้างโปรโตคอลในระดับสี่ แยกงานต่าง ๆ เหล่านี้ออกจากกันเราเรียกว่า โปรโตคอลระดับ 4 ว่า "ทรานสปอร์ต" (Transport)

ในระดับ 4 ก็มีแอดเดรสแยกอีก แต่คราวนี้เราเรียกว่า "หมายเลขพอร์ต" ซึ่งจะช่วยให้ตัวรับและตัวส่ง ทั้งฝ่ายไคลเอนต์และเซิร์ฟเวอร์ติดต่อแอดเดรส IP เดียวกัน แต่แยกกันด้วยโปรโตคอลระดับ 4 ในกรณีของอินเทอร์เน็ตจึงมีโปรโตคอล TCP (Transmission Control Protocol) เป็นตัวแยกที่ทำให้คอมพิวเตอร์เครื่องหนึ่งสามารถติดต่อกับเครื่องอื่นได้หลาย ๆ งานพร้อมกัน

การแบ่งแยกกลุ่มโปรโตคอลนี้เป็นหนทางอันชาญฉลาดของผู้ออกแบบที่ทำให้ระบบสื่อสารข้อมูลดำเนินไปอย่างมีระบบ จนสามารถประยุกต์ใช้งานได้อย่างกว้างขวาง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.3 การใช้งาน ET-XPORT V1 (RS232 / LAN Converter)



รูปที่ 2.5 ชุดอุปกรณ์ที่ใช้สำหรับการแปลงข้อมูล

ET-XPORT V1 เป็นชุดอุปกรณ์ที่ใช้สำหรับการแปลงข้อมูลระหว่าง Protocol ของ TCP/IP และ RS232 โดยด้านหนึ่งจะใช้การติดต่อสื่อสารด้วยพอร์ตสื่อสารอนุกรมแบบ RS232 ส่วนอีกด้านหนึ่งจะสื่อสารผ่านระบบ LAN แบบ Ethernet ด้วย TCP/IP Protocol โดยที่ ET-XPORT V1 จะทำหน้าที่เป็น Data Protocol Converter สำหรับแปลงข้อมูลของการสื่อสารระหว่างอุปกรณ์ที่เป็น RS232 กับอุปกรณ์ที่เป็น Ethernet LAN ซึ่งจะช่วยลดความยุ่งยากสำหรับผู้ใช้งานที่ต้องการติดต่อสื่อสารข้อมูลกับระบบเครือข่ายของ Ethernet LAN โดยผู้ใช้ไม่จำเป็นต้องเสียเวลาไปศึกษารายละเอียดของการรับส่งข้อมูลผ่าน Protocol แบบ TCP/IP มากมายนัก ก็สามารถทำการเชื่อมต่อสื่อสารกับเครือข่าย Ethernet LAN ด้วย TCP/IP Protocol ได้แล้ว

โดย ET-XPORT V1 มีความเหมาะสมในการนำไปประยุกต์ใช้งานที่เกี่ยวกับการสื่อสารผ่าน TCP/IP Protocol โดยเฉพาะระหว่างไมโครคอนโทรลเลอร์กับอุปกรณ์อื่น ๆ ในเครือข่าย โดยการเชื่อมต่อ ET-XPORT V1 ด้านที่เป็นพอร์ตสื่อสารอนุกรม RS232 เข้ากับระบบไมโครคอนโทรลเลอร์ ส่วนด้าน Ethernet Port (RJ45) ก็เชื่อมต่อเข้ากับเครือข่ายจากนั้นก็ทำให้ระบบไมโครคอนโทรลเลอร์สามารถติดต่อสื่อสารข้อมูลกับอุปกรณ์อื่น ๆ ที่ต่ออยู่ภายในเครือข่ายได้แล้ว ซึ่งรูปแบบและข้อกำหนดในการสื่อสารก็สามารถเลือกกำหนดได้มากมายหลายรูปแบบ

ซึ่งนอกจากจะนำ ET-XPORT V1 ไปประยุกต์ใช้งานในด้านของการสื่อสารกับ TCP/IP ร่วมกับระบบไมโครคอนโทรลเลอร์หรือกับอุปกรณ์อื่น ๆ ที่เป็น RS232 แล้วยังสามารถนำไป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ประยุกต์ใช้งานในด้านอื่น ๆ โดยไม่จำเป็นต้องเชื่อมตัวกับไมโครคอนโทรลเลอร์ก็ได้ เช่น ใช้เป็นระบบเตือนภัยหรือตรวจจับสัญญาณ sensor ต่าง ๆ

2.3.1 คุณสมบัติของ ET-XPORT V1

1. มีจุดต่อพอร์ตสื่อสารอนุกรม (Serial Port) แบบ RS232 โดยใช้หัวต่อแบบ DB9 ตัวเมีย สำหรับเชื่อมต่อกับอุปกรณ์ที่ใช้การรับส่งข้อมูลด้วย RS232 เช่น คอมพิวเตอร์ หรือไมโครคอนโทรลเลอร์ต่าง ๆ

2. มีจุดต่อพอร์ต Ethernet LAN แบบ Modula RJ45 จำนวน 1 ช่อง สำหรับเชื่อมต่อกับเครือข่ายระบบ LAN แบบ Ethernet พร้อม LED แสดงสถานะของการสื่อสารข้อมูลของระบบ Ethernet LAN

3. มีจุดต่อสัญญาณ Input แบบ OPTO-ISOLATE สำหรับต่อ Sensor จำนวน 3 Input

4. มี LED แสดงสถานะการทำงานดังนี้

- LED สีแดง ใช้แสดงสถานะของแหล่งจ่ายไฟเลี้ยงวงจร (Power)
- LED สีเขียว ใช้แสดงสถานะของ Input ซึ่งมี 3 ชุด(In0, In1 และ In2)

5. สามารถตั้งค่า Configuration การทำงานผ่านเว็บ Browser หรือ Telnet หรือ RS232 ได้

6. มีระบบป้องกันความปลอดภัยด้วย Password Protection

7. มีระบบป้องกันความปลอดภัยของการสื่อสารข้อมูลด้วยการเข้ารหัส Encryption แบบ Rijndael algorithm

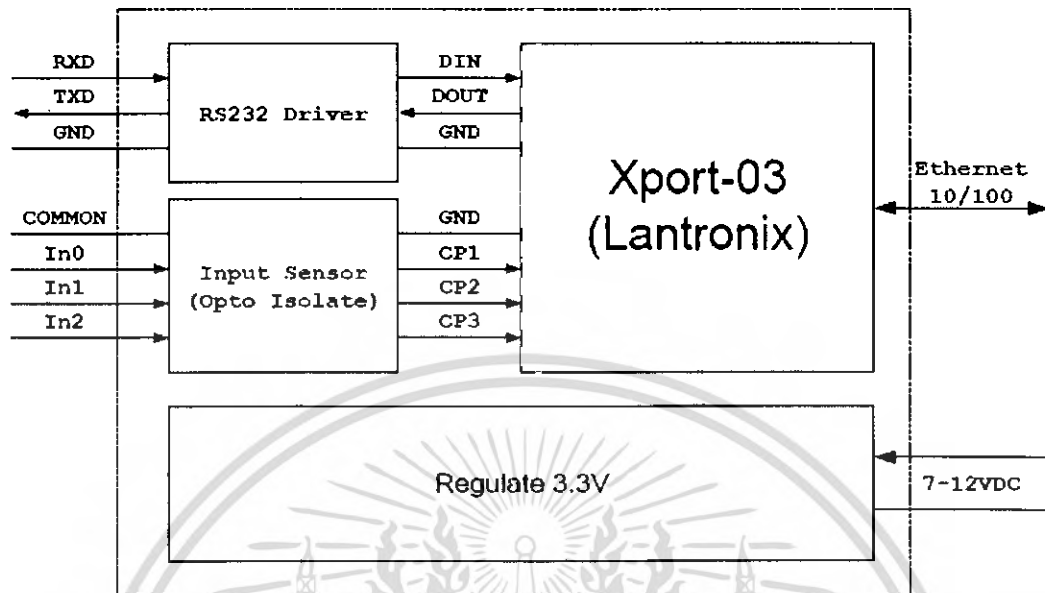
8.รองรับการสื่อสาร TCP/IP Protocol หลายรูปแบบ ตัวอย่างเช่น

- ARP,UDP,TCP,ICMP,Telnet,TFTP,Auto IP,DHCP,HTTP,SNMP สำหรับ Network
- TCP,UDP,Telnet สำหรับการรับส่งผ่าน Serial Port (RS232)
- User Datagram Protocol(UDP)
- SMTP สำหรับการรับส่ง E-mail

9. มี MAC Address หรือ Ethernet Address ที่เป็นมาตรฐานสากลประจำอยู่ภายในชุด

10. ใช้ Power Supply 7 VDC – 12 VDC

2.3.2 โครงสร้างของ ET-XPORT V1



รูปที่ 2.6 แสดงแผนผังโครงสร้างของ ET-XPORT V1

2.3.3 การกำหนดค่าการใช้งาน ET-XPORT V1 กับระบบเครือข่าย Ethernet LAN

สำหรับการนำ ET-XPORT V1 ไปประยุกต์ใช้งานนั้น สามารถทำได้หลายรูปแบบ ขึ้นอยู่กับผู้ที่จะนำไปประยุกต์ดัดแปลง แต่ในที่นี้จะขอเน้นเฉพาะในส่วนของการใช้งานในส่วนของการเชื่อมต่อสัญญาณเพื่อใช้ในการรับส่งข้อมูลระหว่างอุปกรณ์ที่เป็น RS232 และอุปกรณ์ที่ต่อกันอยู่ในเครือข่ายของ Ethernet LAN เป็นหลัก

โดยในโหมดของการเชื่อมต่อสัญญาณเพื่อรับส่งข้อมูลระหว่าง RS232 กับเครือข่ายของ Ethernet LAN นั้น ET-XPORT V1 จะสามารถรองรับการเชื่อมต่อกับอุปกรณ์ในเครือข่าย โดยใช้ TCP/IP Protocol ได้ 3 รูปแบบด้วยกัน คือ TCP, UDP และ Telnet ซึ่งผู้ใช้สามารถเลือกกำหนดรูปแบบของ Protocol ที่จะใช้ในการเชื่อมต่อกับระบบเครือข่ายได้ตามต้องการ โดยก่อนที่จะนำ ET-XPORT V1 ไปเชื่อมต่อเข้ากับระบบเครือข่ายได้นั้น ผู้ใช้จะต้องทำการกำหนดค่าที่จำเป็นต่อใช้งานในระบบเครือข่ายให้กับ ET-XPORT V1 ให้เรียบร้อยเสียก่อน โดยในระบบเครือข่ายของ Ethernet LAN นั้น มีความจำเป็นต้องกำหนดค่าต่างๆให้กับอุปกรณ์ที่จะนำมาเชื่อมต่อเพื่อใช้งานร่วมกันในระบบเครือข่ายดังนี้

ค่า Hardware Address หรือ MAC Address ซึ่งมีค่าเป็นเลข HEX ขนาด 2 หลัก ระหว่าง 00-FF จำนวน 6 ชุด โดยในส่วนนี้ อุปกรณ์แต่ละตัวที่จะนำมาเชื่อมต่อกันในเครือข่าย ต้องมีค่ารหัสที่ไม่ซ้ำกัน แต่สำหรับ ET-XPORT V1 นั้นค่ารหัสนี้จะถูกกำหนดไว้ที่ตัว ET-XPORT V1 แต่ละตัวอยู่แล้ว ดังนั้นในการอ้างอิงค่ารหัส MAC Address ของ ET-XPORT V1

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

นั้นผู้ใช้สามารถตรวจสอบได้จากรหัสตัวเลขที่กล่องของ ET-XPORT V1 ได้โดยตรง โดย ET-XPORT V1 ทุกๆตัวจะมี MAC Address ของ 3 ชุดแรกเหมือนกัน คือ 00-20-4A ส่วน 3 ชุดสุดท้ายจะมีความแตกต่างกัน เช่น 00-20-4A-86-AB-AB

ค่า IP Address ซึ่งเป็นหมายเลข IP Address ที่จะใช้ในการสื่อสารข้อมูลกับอุปกรณ์อื่นๆในเครือข่ายโดยใช้ TCP/IP Protocol มีค่าเป็นตัวเลขฐานสิบขนาด 3 หลัก ระหว่าง 0-255 จำนวน 4 ชุด ซึ่งค่าของ IP Address นี้ผู้ใช้ต้องกำหนดให้กับ ET-XPORT V1 เอง และต้องมีค่า IP Address ที่ไม่ซ้ำกับอุปกรณ์อื่นๆที่อยู่ในเครือข่ายด้วย เช่น 192.168.1.50 เป็นต้น

ค่า Port Numbers ซึ่งเป็นค่า Port ที่ใช้อ้างอิงในการรับส่งข้อมูลของ TCP และ UDP โดยสามารถกำหนดได้ตามต้องการ แต่ไม่ต้องตรงกับหมายเลข Port ที่สงวนไว้ใช้ในระบบ โดย ET-XPORT V1 นั้นจะมีการสงวนค่าหมายเลข Port ไว้ใช้งานเป็นการเฉพาะ(ห้ามใช้งานโดย User) ดังต่อไปนี้

-1-1024 จะถูกสงวนไว้ใช้เกี่ยวกับการติดต่อสื่อสารทั่วไป

-9999 จะถูกสงวนไว้เพื่อใช้สำหรับการสั่ง Setup ค่า Configuration ให้กับ ET-XPORT V1 ผ่านระบบเครือข่ายด้วย Telnet Protocol

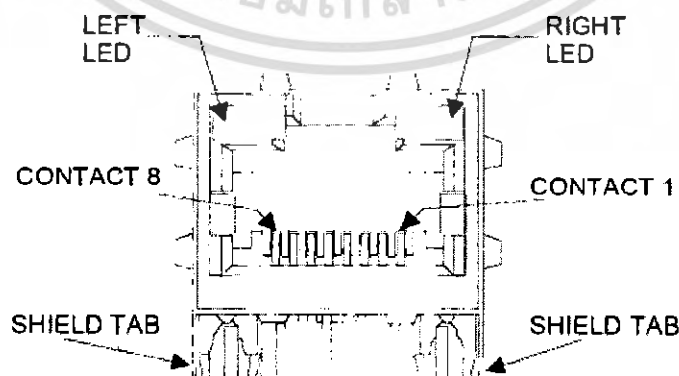
-14000-14009 จะถูกสงวนไว้สำหรับใช้งานในการสื่อสารร่วมกับโปรแกรม Redirector

-30704 จะถูกสงวนไว้ใช้ในระบบด้านการรักษาความปลอดภัย (77F0H)

-30718 จะถูกสงวนไว้ใช้ในระบบด้านการรักษาความปลอดภัย (77FEH)

2.4 การเชื่อมต่อสัญญาณด้าน Ethernet LAN (Ethernet Interface)

สำหรับการเชื่อมต่อสัญญาณกับระบบเครือข่ายแบบ Ethernet LAN นั้น ชุด ET-XPORT V1 ของ ETT จะใช้การเชื่อมต่อผ่านหัวต่อสัญญาณแบบ RJ45 มาตรฐาน โดยหัวต่อสัญญาณดังกล่าวจะมีการจัดเรียงสัญญาณการเชื่อมต่อที่เป็นมาตรฐานตามข้อกำหนดของระบบ Ethernet LAN ทุกประการ โดยการจัดเรียงสัญญาณของ Connector แบบ RJ45 ของ ET-XPORT V1 จะมีการจัดเรียงสัญญาณดังตารางต่อไปนี้



รูปที่ 2.7 แสดงลักษณะของหัว RJ45 ที่ใช้เชื่อมต่อสัญญาณกับเครือข่าย Ethernet LAN

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่นิยมนำไปใช้ประโยชน์ทางการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Signal Name	Dir	Pin No	Primary Function
TX+	Out	1	Transmit Data +
TX-	Out	2	Transmit Data -
RX+	In	3	Receive Data +
RX-	In	6	Receive Data -
Not Used		4	Terminated
Not Used		5	Terminated
Not Used		7	Terminated
Not Used		8	Terminated
SHIELD			Chassis Ground

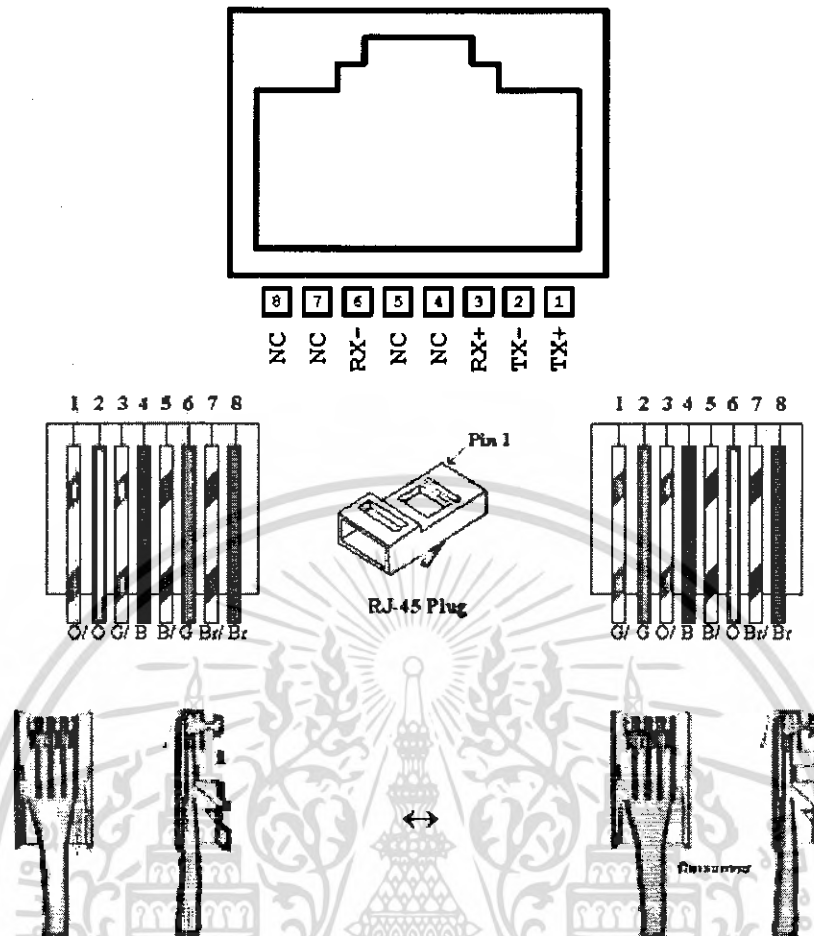
ตารางที่ 2.1 แสดงการจัดเรียงสัญญาณของ RJ45

2.4.1 สายสัญญาณ Ethernet LAN โดยตรงกับเครื่องคอมพิวเตอร์ PC (Peer to Peer)

ในกรณีที่มีการต่อบอร์ด ET-XPORT V1 กับ Ethernet Port ของคอมพิวเตอร์โดยตรง โดยไม่ผ่าน HUB นั้น ลักษณะของสายสัญญาณจะใช้หัวต่อแบบ RJ45 ตัวผู้ทั้ง 2 ด้าน ซึ่งรูปแบบของการเข้าสายจะใช้ลักษณะการเข้าแบบต่อสลับกัน โดยต่อ TX(+) เข้ากับ RX(+) และต่อ TX(-) เข้ากับ RX(-) โดยมีแผนผังการต่อสายดังนี้

การต่อสาย RJ45 ตัวผู้ 2 ด้าน ระหว่างต้นทางและปลายทาง

- สีขาวส้ม ใช้ ต่อระหว่าง Pin-1 (TX+) กับ Pin-3 (RX+)
- สีส้ม ใช้ ต่อระหว่าง Pin-2 (TX-) กับ Pin-6 (RX-)
- สีขาวเขียว ใช้ ต่อระหว่าง Pin-3 (RX+) กับ Pin-1 (TX+)
- สีน้ำเงิน ใช้ ต่อระหว่าง Pin-4 (INC) กับ Pin-4 (NC)
- สีขาวน้ำเงิน ใช้ ต่อระหว่าง Pin-5 (NC) กับ Pin-5 (NC)
- สีเขียว ใช้ ต่อระหว่าง Pin-6 (RX-) กับ Pin-2 (TX-)
- สีขาวน้ำตาล ใช้ ต่อระหว่าง Pin-7 (NC) กับ Pin-7 (NC)
- สีน้ำตาล ใช้ ต่อระหว่าง Pin-8 (NC) กับ Pin-8 (NC)



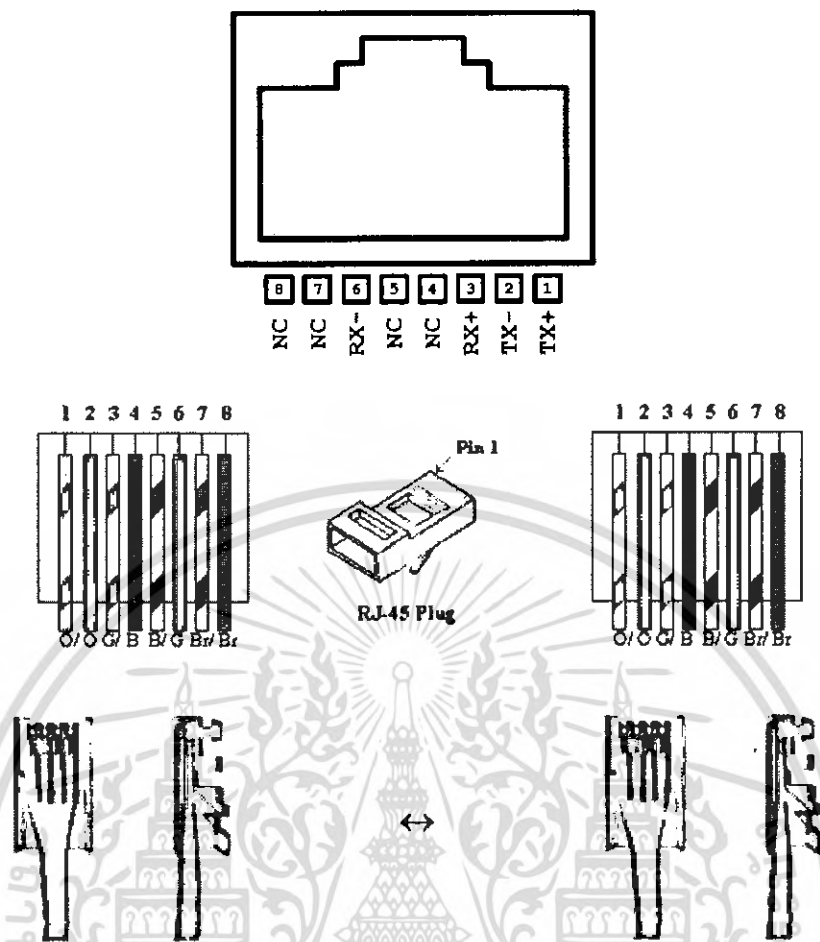
รูปที่ 2.8 แสดงการเข้าสายสำหรับเชื่อมต่อ ET-XPORT V1 โดยตรงกับ Port ของเครื่อง PC

2.4.2 สายสัญญาณ Ethernet LAN ผ่าน HUB

ในกรณีที่มีการต่อบอร์ด ET-XPORT V1 กับเครือข่าย โดยการต่อผ่าน HUB นั้น ลักษณะของสายสัญญาณจะใช้หัวต่อแบบ RJ45 ตัวผู้ทั้ง 2 ด้าน ซึ่งรูปแบบของการเข้าสายจะใช้ลักษณะการเข้าสายแบบต่อขนานกัน โดยไม่มีการสลับสายสัญญาณ โดยมีแผนผังการต่อสายดังนี้

- การต่อสาย RJ45 ตัวผู้ ทั้ง 2 ด้าน
- RJ45 Pin-1 ต่อกับสายสี ขาวส้ม (TX+)
 - RJ45 Pin-2 ต่อกับสายสี ส้ม (TX-)
 - RJ45 Pin-3 ต่อกับสายสี ขาวเขียว (RX+)
 - RJ45 Pin-4 ต่อกับสายสี น้ำเงิน (NC)
 - RJ45 Pin-5 ต่อกับสายสี ขาวน้ำเงิน (NC)
 - RJ45 Pin-6 ต่อกับสายสี เขียว (RX-)
 - RJ45 Pin-7 ต่อกับสายสี ขาวน้ำตาล (NC)
 - RJ45 Pin-8 ต่อกับสายสี น้ำตาล (NC)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.9 แสดงการเข้าสายสำหรับเชื่อมต่อ ET-XPORT V1 กับเครือข่าย ผ่าน HUB

2.5 ทฤษฎีการเขียนโปรแกรมส่งข้อมูลผ่าน TCP/IP ด้วย Visual Basic

ในการเขียนโปรแกรมได้ใช้โปรแกรม Visual Basic ในการเขียนโปรแกรม เพื่อควบคุมการทำงานของอุปกรณ์ และต้องการที่จะควบคุมผ่านระบบ LAN (Local Area Network) ดังนั้นจึงต้องส่งข้อมูลผ่านระบบ LAN ซึ่งในการติดต่อผ่านระบบ LAN นั้น มีหลายมาตรฐานในการเชื่อมต่อ โดยมาตรฐานในการเชื่อมต่อหลัก ๆ ที่ใช้ในการเขียนก็จะมีด้วยกัน 2 มาตรฐาน คือ TCP/IP และ UDP/IP ซึ่งเป็นโพรโตคอลที่ใช้กันอย่างแพร่หลาย และเป็นมาตรฐานเดียวกับอินเทอร์เน็ตในปัจจุบัน

Winsock ระบบปฏิบัติการยูนิกซ์จะมีความสามารถอย่างหนึ่งที่เรียกว่า Socket ซึ่งก็คือการที่โปรแกรมต่าง ๆ จะสามารถสื่อสารข้อมูลระหว่างกันได้โดยไม่จำเป็นต้องเป็นว่าโปรแกรมนั้นจะต้องรันอยู่บนคอมพิวเตอร์เครื่องเดียวกัน

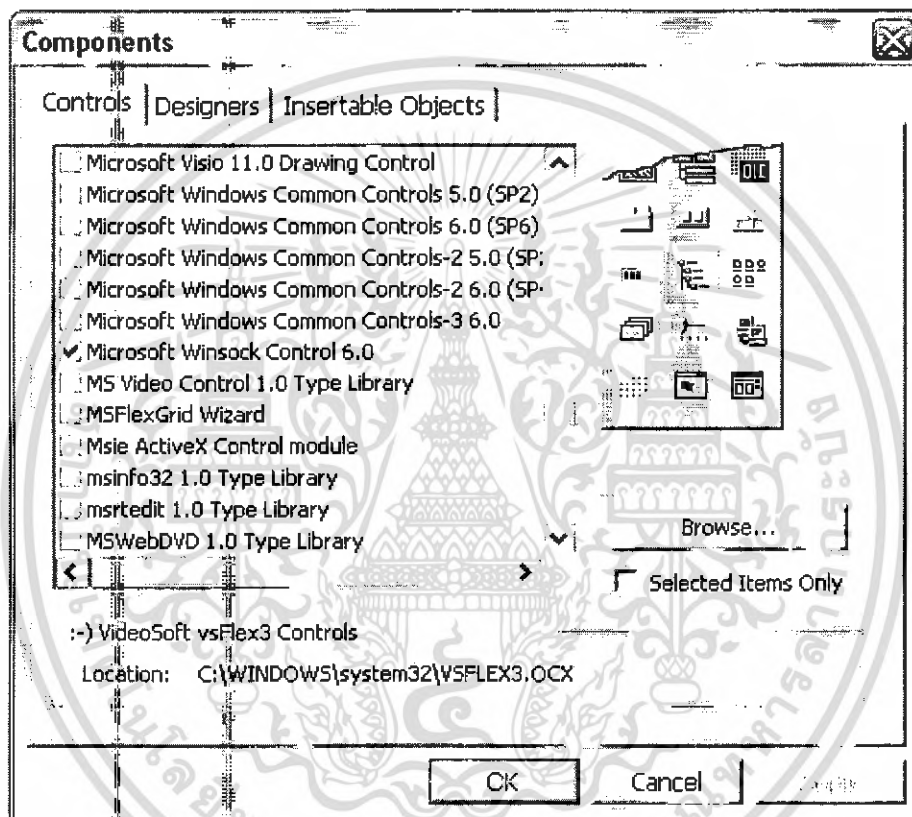
การติดต่อผ่าน Socket ก็คือ โปรแกรมส่วนที่ทำหน้าที่เป็น Server จะสร้าง Socket (ซึ่งถ้าแปลตรงตามตัวก็คือ ช่องเสียบ) โดยสามารถมีได้หลายช่อง จากนั้นเมื่อ Client ต้องการติดต่อสื่อสาร ก็จะติดต่อมาที่ Socket ที่ฝั่ง Server เตรียมไว้ให้ เมื่อเชื่อมต่อกันเสร็จก็พร้อมที่จะ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่นิยมนำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

แลกเปลี่ยนข้อมูลระหว่างกันได้

การใช้งาน Winsock ในระบบปฏิบัติการของ Microsoft ในระบบปฏิบัติการของ Microsoft ได้เตรียมการทำงานในลักษณะ Socket เอาไว้ให้ได้ใช้งาน โดยตั้งชื่อว่า Winsock โดยมีหลักการคล้ายคลึงกับ Socket ของระบบปฏิบัติการยูนิกซ์

ในมุมมองของการพัฒนาแอปพลิเคชัน Winsock ถือเป็น API (Application Program Interface) หรือ ชุดฟังก์ชันมาตรฐาน ซึ่งออกแบบมาเพื่อให้การเขียนโปรแกรมใช้งาน อินเทอร์เน็ตเป็นเรื่องง่าย



รูปที่ 2.10 การเรียกใช้งาน Winsock Control

การใช้งาน Winsock Control ใน Visual Basic ได้เตรียม Active X Control ที่มีชื่อว่า Winsock Control ไว้ให้เราใช้งาน ซึ่งพร้อมให้เราใช้งานเพื่อสร้างการเชื่อมต่อระหว่างเครื่องคอมพิวเตอร์ 2 เครื่องในเครือข่าย ทำให้สามารถแลกเปลี่ยนข้อมูลระหว่างกันได้ สำหรับ Winsock Control จะถูกเก็บไว้ในไฟล์ MSWINSOCK.OCX

โหมดการทำงานของ Winsock Control ใน Winsock Control สามารถรองรับการทำงานได้ 2 โหมด คือ TCP และ UDP ซึ่งสามารถเลือกลักษณะการทำงานได้โดย

TCP (Transmission Control Protocol) เป็นการทำงานในลักษณะ Connection-

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Base เปรียบเสมือนการทำงานของโทรศัพท์ คือจะต้องมีการเชื่อมต่อสื่อสารก่อน ถึงจะสามารถแลกเปลี่ยนข้อมูลได้ จนกระทั่งยุติการสื่อสาร ซึ่งในระหว่างที่เชื่อมต่อกันอยู่ จะมีการตรวจสอบการทำงาน ทำให้การสื่อสารแบบนี้มีประสิทธิภาพมากกว่า โดย TCP จะเหมาะกับการส่งข้อมูลที่ต้องการตรวจสอบความถูกต้องของข้อมูล เช่น การส่งภาพ เสียง ในระบบเครือข่าย แต่เนื่องจากการตรวจสอบความถูกต้องของข้อมูล จึงทำให้ใช้ทรัพยากรของระบบค่อนข้างสูง

UDP (User Datagram Protocol) เป็นการทำงานแบบ Connection Less คือไม่มีการสร้างการเชื่อมต่อ จึงทำให้ไม่เปลืองช่องสัญญาณในการติดต่อ เนื่องจากไม่มีการสร้างการเชื่อมต่อไว้ แต่ไม่สามารถตรวจสอบได้ว่าปลายทางได้รับข้อมูลหรือยัง โดย UDP เหมาะสำหรับการส่งข้อมูลที่ไม่สำคัญนัก เช่น สถานการณ์ทำงาน โดยเป็นข้อมูลขนาดเล็ก จึงทำให้ UDP ใช้ทรัพยากรของระบบน้อยกว่า TCP

Property ที่สำคัญของ Winsock Control

Protocol	เป็นการเลือกโปรโตคอลที่จะใช้งาน TCP/UDP
Local IP	หมายเลข IP ของเครื่องที่รันโปรแกรม
Local Port	กำหนด Port ที่จะรอรับการเชื่อมต่อ
Remote HostIP	หมายเลข IP Address ของเครื่องปลายทาง
Remote Port	หมายเลข Port ของเครื่องปลายทาง

Method ที่สำคัญของ Winsock

Listen	เป็นการสร้าง Socket เพื่อให้เครื่องคอมพิวเตอร์เครื่องอื่น ๆ ติดต่อเข้ามาได้
Connect	เป็นการสร้างการติดต่อสื่อสารไปยังเครื่องที่เปิด Socket รอรับการติดต่อ โดยจะต้องระบุหมายเลข IP Address และ Port ของเครื่องที่รอรับการติดต่อ
Accept	เมื่อมีการติดต่อเพื่อต้องการ Connect จากเครื่อง Client ที่เครื่อง Server จะได้รับการ Request Connect ดังนั้น ถ้าอนุญาตให้เครื่อง Client เข้ามาเชื่อมต่อได้ ต้อง Accept เพื่อสร้างการเชื่อมต่อ
Close	เป็นการยุติการเชื่อมต่อ
SendData	เป็นการส่งข้อมูลในตัวแปรที่กำหนดไปยัง Buffer ของการส่งข้อมูล
GetData	เป็นการรับข้อมูลที่อยู่ใน Buffer เข้ามาเก็บในตัวแปรที่กำหนด

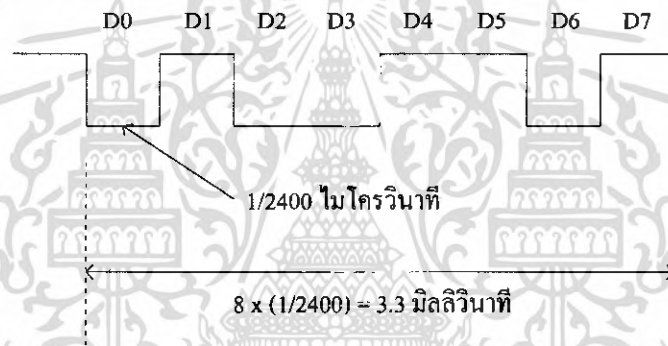
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สำนักหอสมุดกลาง พระจอมเกล้าลาดกระบัง

2.6 ทฤษฎีการเชื่อมต่อ Serial Port (RS232)

การสื่อสารข้อมูลอนุกรมเป็นการรับหรือส่งข้อมูลในลักษณะของบิตหรือกลุ่มของบิต คราวละหนึ่งบิตเป็นลำดับเรื่อยไปจนสิ้นสุด

โดยทั่วไปมักจะระบุกันในหน่วยของจำนวนบิตข้อมูลภายในเวลาหนึ่งวินาที เรียกว่า อัตราบอเรียต ตามค่ามาตรฐานเหล่านี้ ได้แก่ 110, 150, 300, 1200, 2400, 4800, และ 9600 จากรูปที่ 2.11 แสดงให้เห็นลักษณะของรูปแบบสัญญาณข้อมูลอนุกรมที่ปรากฏในสายส่ง สัญญาณ ข้อมูลทั้ง 8 บิตนี้หากว่าถูกส่งออกมาด้วยอัตรา 2400 บอดจะใช้เวลาในการส่งข้อมูลหนึ่งบิตมีค่าเท่ากับ $1/2400$ หรือ 416 ไมโครวินาที และเวลาในการส่งข้อมูลทั้ง 8 บิตมีค่าเท่ากับ 8×416 หรือ 3,328 ไมโครวินาทีแต่ในโครงการงานของเราจะใช้อัตราบอเรียตเท่ากับ 9600 และส่งข้อมูลที่ละ 8 บิต



รูปที่ 2.11 แบบสัญญาณไฟฟ้าของข้อมูล ซึ่งเป็นการส่งข้อมูลแบบอนุกรมด้วยอัตราเร็ว 2400 บิต/วินาที (สังเกตว่าจะเริ่มส่งจากบิต D0 ซึ่งเป็นบิตนัยสำคัญต่ำออกมาก่อนเป็นลำดับแรก)

2.6.1 รูปแบบของข้อมูลอนุกรม

วิธีการที่จะทำให้ข้อมูลสื่อสารอนุกรมมีความถูกต้องมากยิ่งขึ้น จะใช้การเพิ่มเติมบิตข้อมูลบางอย่างรวมไปกับการส่งข้อมูลจริง ได้แก่

1. บิตเริ่มต้น (Start Bit)

มีหน้าที่สำหรับการบ่งบอกให้วงจรฮาร์ดแวร์ทางด้านรับทราบถึง ตำแหน่งจุดเริ่มต้นของบิตข้อมูลกลุ่มใหม่ เพื่อที่จะทำการปรับจังหวะของสัญญาณการรับข้อมูลให้ตรงกัน ดังนั้นบิตเริ่มต้นนี้จึงจะถูกเพิ่มเข้าไปก่อนระดับลอจิกของสภาวะของสายสื่อสาร ขณะเมื่อไม่มีการส่งข้อมูล

2. บิตแสดงภาวะความเป็นเลขคู่หรือเลขคี่ (Parity Bit)

มีหน้าที่เพื่อตรวจสอบความถูกต้องของข้อมูล โดยทั่วไปมักเรียกว่า บิตพาริตีและจะนำไปแทรกต่อท้ายบิตข้อมูล ค่าของบิตนี้ขึ้นอยู่กับจำนวนค่าของบิตข้อมูลที่เป็น 1 ซึ่งจะเป็นได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

72737

สองลักษณะ คือ พาริตีคู่ (Even Parity) หรือ พาริตีคี่ (Odd Parity)

3. บิตสุดท้าย (Stop Bit)

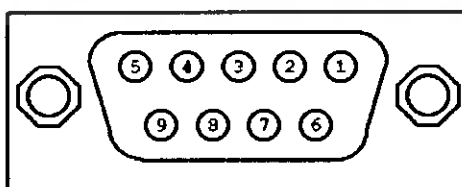
เป็นบิตที่เพิ่มเติมขึ้นเพื่อระบุถึงขอบเขตการสิ้นสุดของกลุ่มบิตข้อมูล บิตสุดท้ายนี้อาจจะมีจำนวนมากกว่าหนึ่งบิตได้ คือ 1.5 บิต และ 2 บิต ดังนั้นกรณีของการส่งข้อมูล 8 บิต พร้อมบิตที่เพิ่มเติมเข้าไปโดยสมบูรณ์ คือ บิตเริ่มต้น บิตพาริตี และ บิตสุดท้าย รวมทั้งสิ้น 12 บิต ตามแผนภาพสัญญาณเวลาในรูปที่ 2.12 หากข้อมูลถูกส่งออกไปด้วยอัตราเร็ว 2400 บอดเรต เวลาโดยรวมในการส่งข้อมูลหนึ่งไบต์จะมีค่าเป็น 12×416 ไมโครวินาที หรือ 4.99 มิลลิวินาที



รูปที่ 2.12 แผนภาพสัญญาณเวลาของข้อมูลแบบอนุกรมจำนวน 8 บิต พร้อมด้วยบิตเริ่มต้นบิตพาริตี และ บิตสุดท้าย

2.6.2 การเชื่อมต่อสัญญาณด้าน RS232

ตามปกติแล้วเครื่อง ET-XPOR V1 จะมีพอร์ตสื่อสารอนุกรม RS232 จัดเตรียมไว้ให้ใช้งานอยู่แล้ว ซึ่งขั้วต่อสัญญาณของพอร์ตสื่อสารอนุกรมแบบ RS232 ของคอมพิวเตอร์ ET-XPOR V1 นั้นจะเป็นแบบ DB9 ตัวผู้ โดยการต่อสายกับบอร์ด ET-XPOR V1 นั้นจะใช้สายเพียง 3 เส้นเท่านั้น คือ RXD, TXD และ GND ซึ่งการเชื่อมต่อสามารถกระทำดังนี้



2 = TXD
3 = RXD
5 = GND

รูปที่ 2.13 แสดงการจัดเรียงสัญญาณ RS232 (DB9 ตัวเมีย) ของ ET-XPOR V1

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่นิยมนำไปใช้ประโยชน์ทางการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ขั้วต่อ Com Port ของเครื่องคอมพิวเตอร์ (RS232)			ขั้วต่อ RS232 ของ ET-XPORT V1
DB25 ตัวผู้	DB9 ตัวผู้		DB9 ตัวเมีย
3 = RXD	2 = RXD	←	2 = TXD
2 = TXD	3 = TXD	→	3 = RXD
20 = DTR	4 = DTR		4 = NC
7 = GND	5 = GND	↔	5 = GND
6 = DSR	6 = DSR		6 = NC
4 = RTS	7 = RTS		7 = NC
5 = CTS	8 = CTS		8 = NC

ตารางที่ 2.2 แสดงผังการต่อสายสัญญาณ RS232 ระหว่าง คอมพิวเตอร์ PC และ ET-XPORT

สำหรับกรณีที่มีการเชื่อมต่อ ET-XPORT V1 เข้ากับ RS232 ของอุปกรณ์อื่นๆซึ่งมีการจัดเรียงขั้วสัญญาณที่แตกต่างไปจากนี้ ผู้ใช้สามารถดัดแปลงรูปแบบการต่อสายสัญญาณ โดยอ้างอิงหลักการเดียวกันนี้ได้กล่าวคือ ต่อสัญญาณ TXD ของ ET-XPORT V1 เข้ากับ RXD ของอุปกรณ์ที่จะนำมาเชื่อมต่อ และต่อ RXD ของ ET-XPORT V1 เข้ากับ TXD ของอุปกรณ์ที่จะนำมาเชื่อมต่อ ส่วน GND จะต้องต่อถึงกันด้วย

2.7 บอร์ด Controller

ET-BASE51 V2.0 เป็นบอร์ดไมโครคอนโทรลเลอร์ในตระกูล MCS51 ขนาด 40 Pin ซึ่งเลือกใช้ไมโครคอนโทรลเลอร์ เบอร์ AT89C51ED2 ของ ATMEL เป็น MCU ประจำบอร์ด โดย MCU รุ่นนี้จะบรรจุอยู่ในตัวถังแบบ DIP40 โดย MCU ตัวนี้จะมีจุดเด่น คือ เรื่องของความเร็วในการประมวลผล ซึ่งสามารถทำงานได้ด้วยความเร็วสูงสุด 60MHz ที่ 12 Clock / 1 Machine Cycle นอกจากนี้แล้วยังมีความเพียบพร้อมด้วยอุปกรณ์พื้นฐานต่างๆที่จำเป็นต่อการใช้งาน ไม่ว่าจะเป็นหน่วยความจำสำหรับเก็บข้อมูลแบบ EEPROM ขนาด 2 KByte หรือหน่วยความจำใช้งานแบบ RAM ซึ่งมีมากถึง 1792 Byte ส่วนในด้านของอุปกรณ์ Peripheral นั้นก็นับว่าครบถ้วนเหมาะแก่การนำไปประยุกต์ใช้งานเกี่ยวกับการควบคุมและประมวลผลต่างๆ ได้เป็นอย่างดี โดยจะมีทั้ง SPI, UART, Watchdog, Timer/Counter, PWM ฯลฯ โดยการออกแบบโครงสร้างของบอร์ดนั้นจะเน้นเรื่องขนาดของบอร์ดให้มีขนาดเล็กเพื่อให้ง่ายต่อการนำไปประยุกต์ใช้งาน และสะดวกต่อการพัฒนาโปรแกรม

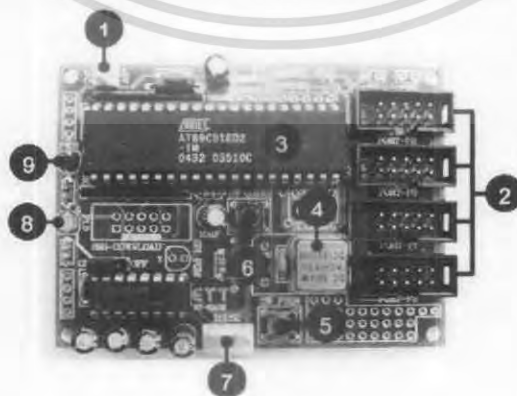
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.7.1 คุณสมบัติของบอร์ด

เลือกใช้ MCU ตระกูล MCS51 เบอร์ AT89C51ED2 ของ ATMEL เป็น MCU ประจำบอร์ด โดยเลือกใช้แหล่งกำเนิดสัญญาณนาฬิกาแบบ Oscillator Module ค่า 29.4912 MHz ซึ่งสามารถกำหนดการทำงานของ MCU ให้ทำงานในโหมดความเร็ว 2 เท่า (X2 Mode) ได้ ทำให้ MC สามารถประมวลผลด้วยความเร็วสูงสุดที่ 58.9824 MHz โดยคุณสมบัติเด่นๆของ MCU ได้แก่

1. มีหน่วยความจำ Flash สำหรับเขียนโปรแกรมขนาด 64KByte
 2. มี EEPROM ขนาด 2KByte สำหรับเก็บข้อมูล และสามารถเขียนซ้ำได้กว่า 1 ล้านครั้ง
 3. มีพอร์ต I/O ขนาด 8 บิต จำนวน 4 พอร์ต (P0,P1,P2 และ P3)
 4. มี RAM ใช้งาน 1792 Byte เพียงพอต่อการใช้งาน
 5. มีวงจรรีโมตสื่อสารอนุกรม UART จำนวน 1 พอร์ต และมีวงจรรีโมตสื่อสาร SPI จำนวน 1 พอร์ต
 6. มีวงจรร Timer/Counter ขนาด 16 บิต จำนวน 3 ชุด
 7. มีวงจรร Watchdog, Power-ON Reset, Capture/Compare ,PWM
- มีพอร์ตสื่อสารอนุกรมแบบ RS232 จำนวน 1 ช่อง สำหรับใช้ในการ Download โปรแกรมให้กับบอร์ด และประยุกต์ใช้งานทั่วไป
- มีขั้วต่อสัญญาณ I/O แบบ TTL แบบ Header ขนาด 2x5 จำนวน 4 ชุด (P0,P1,P2 และ P3)
 - มี LED แสดงสถานะแหล่งจ่าย Power และ Self-Test สำหรับใช้ทดสอบการทำงานของบอร์ด
 - ใช้แหล่งจ่ายไฟขนาด +5VDC
 - ขนาด PCB Size เล็กเพียง 8 x 6 cm.

2.7.2 โครงสร้างบอร์ด ET-BASE51 V2.0 (AT89C51ED2)



รูปที่ 2.14 บอร์ด ET-BASE51 V2.0 (AT89C51ED2)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หมายเลข 1 คือ ขั้วต่อแหล่งจ่ายไฟเลี้ยงวงจรของบอร์ด ใช้กับแหล่งจ่ายไฟตรง +5VDC

หมายเลข 2 คือ PORT-P2, PORT-P0, PORT-P1 และ PORT-P3 ตามลำดับ

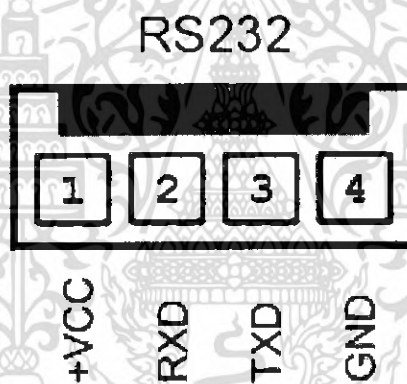
หมายเลข 3 คือ MCU เบอร์ AT89C51ED2 ซึ่งเป็น MCU ตระกูล MCS51 จาก ATMEL

หมายเลข 4 คือ Oscillator Module ค่า 29.4912 MHz

หมายเลข 5 คือ SW PSEN ใช้สำหรับกำหนดสถานะลอจิก "0" ให้ขา PSEN สำหรับใช้ในขั้นตอนของการ Download HEX File ให้กับหน่วยความจำ Flash ของ MCU ในบอร์ด

หมายเลข 6 คือ SW RESET ใช้สำหรับรีเซ็ตการทำงานของ MCU ภายในบอร์ด

หมายเลข 7 คือ พอร์ต RS232 หรือ Serial Port สำหรับติดต่อกับอุปกรณ์มาตรฐาน RS232 และใช้เป็น ISP Download Connector สำหรับ Download HEX File ให้กับ MCU ของบอร์ด



รูปที่ 2.15 การเชื่อมต่อสัญญาณ RS232

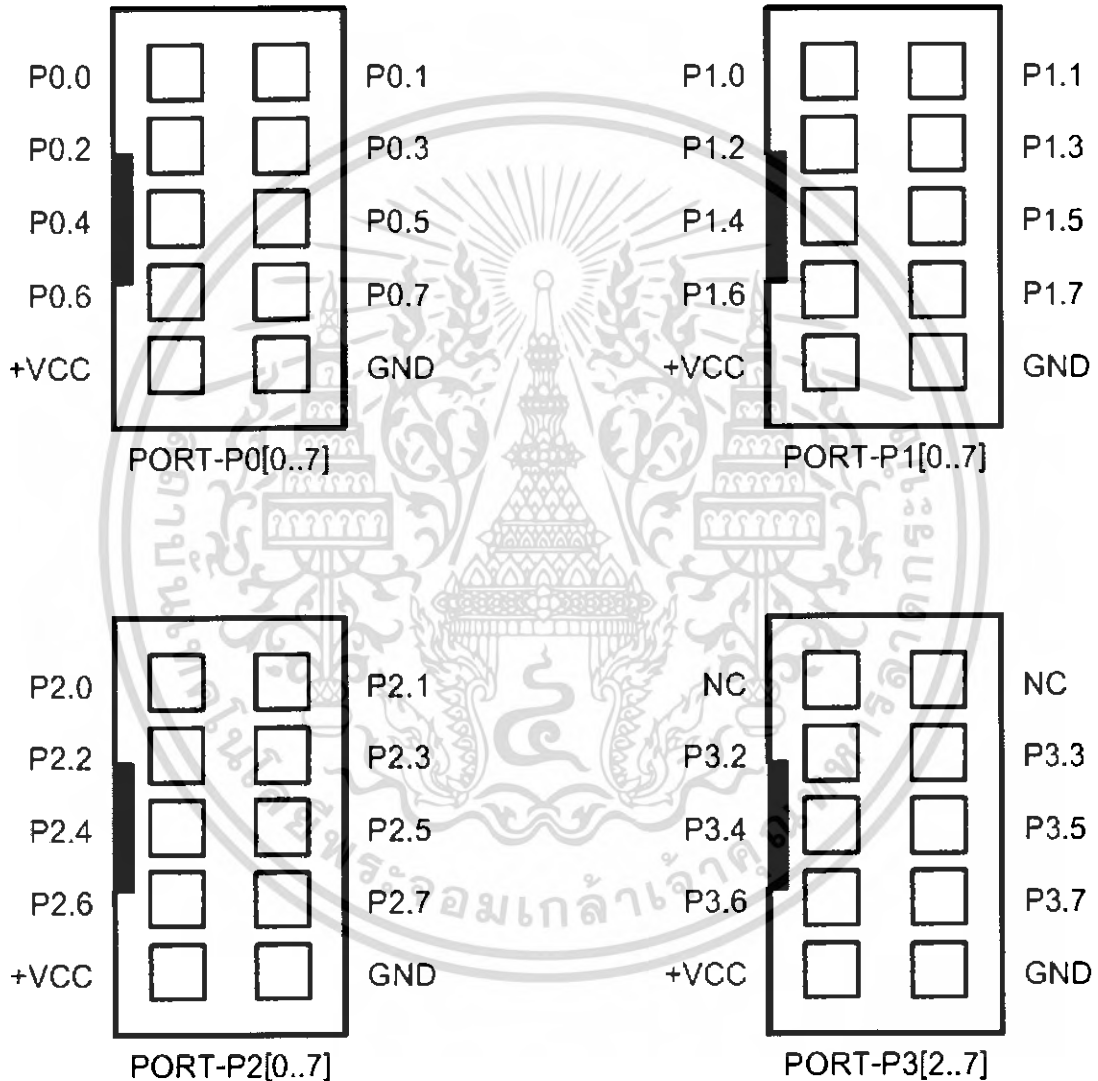
หมายเลข 8 คือ LED แสดงสถานะของ P1.0 ในกรณีที่เลือก Jumper ไว้ทางด้าน ON เพื่อ Test

หมายเลข 9 เป็น LED Power สีแดง ใช้แสดงสถานะของแหล่งจ่ายไฟเลี้ยงวงจรของบอร์ด

2.7.3 ขั้วต่อสัญญาณต่าง ๆ

สำหรับขั้วต่อสัญญาณของพอร์ต I/O จาก MCU นั้น จะถูกออกแบบและจัดเตรียมไว้ผ่านทางขั้วต่อแบบ IDE Header ขนาด 2x5 จำนวน 4 ชุด คือ PORT-P0, PORT-P1, PORT-P2 และ PORT-P3ตามลำดับ โดยที่ขั้วต่อสัญญาณแต่ละชุด จะประกอบไปด้วยสัญญาณของ I/O ที่เชื่อมต่อมาจากขาสัญญาณของ MCU โดยตรงทั้งหมด โดยแต่ละพอร์ตจะมีสัญญาณเอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปเผยแพร่ขึ้นด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

พอร์ตละ 8 บิต ยกเว้น PORT-P3 ซึ่งจะมีเพียง 6 บิต คือ P3.2-P3.7 เท่านั้น ส่วน P3.0 และ P3.1 จะถูกสงวนไว้ใช้งานเป็นขาสัญญาณ RXD และ TXD สำหรับรับส่งข้อมูลของ RS232 ซึ่งสัญญาณทั้ง 2 เส้น (P3.0 และ P3.1) จะถูกเชื่อมต่อผ่านวงจร Line Driver (MAX232) สำหรับแปลงระดับสัญญาณจากระดับลอจิก TTL ของ MCU ให้เป็นสัญญาณแรงดันตามมาตรฐานของ RS232 โดยสัญญาณที่ได้รับการแปลงเป็นแบบ RS232 จะถูกเชื่อมต่อไปรอไว้ที่ขั้วต่อแบบ CPA ขนาด 4 PIN (RS232) โดยการจัดเรียงสัญญาณของแต่ละชุด จะเป็นดังรูป



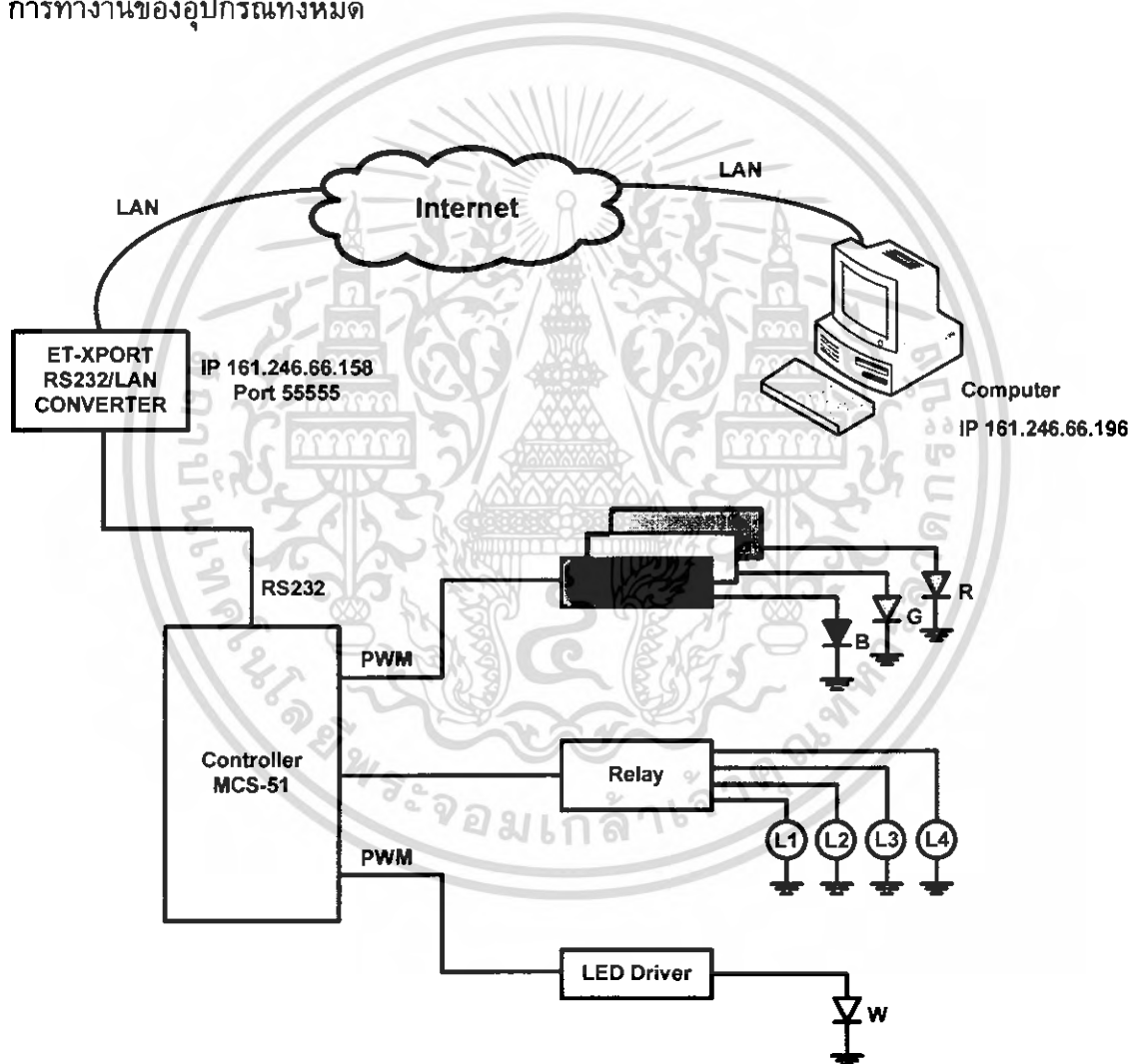
รูปที่ 2.16 การจัดเรียงสัญญาณของพอร์ต I/O ต่างๆของบอร์ด ET-BASE51 V2.0

บทที่ 3

หลักการทํางานและการออกแบบวงจร

3.1 การออกแบบโดยรวมของโครงการ

เริ่มจากการเขียนโปรแกรมสั่งงานด้วยโปรแกรม Visual basic 6.0 บนวินโดวส์ แล้วส่งงานออกทาง Port LAN แล้วส่งคำสั่งผ่านระบบ LAN เพื่อให้ตัวรับ (ET X-PORT) โดยกำหนด IP Address ที่เราทำการตั้งค่าไว้ที่ตัวของ ET X-POR เมื่อมันรับค่ามาจะทำหน้าที่แปลงสัญญาณ LAN ให้เป็น RS232 เพื่อส่งไปยัง Microcontroller เพื่อแปลงออกมาเป็นคำสั่งการทำงานของอุปกรณ์ทั้งหมด

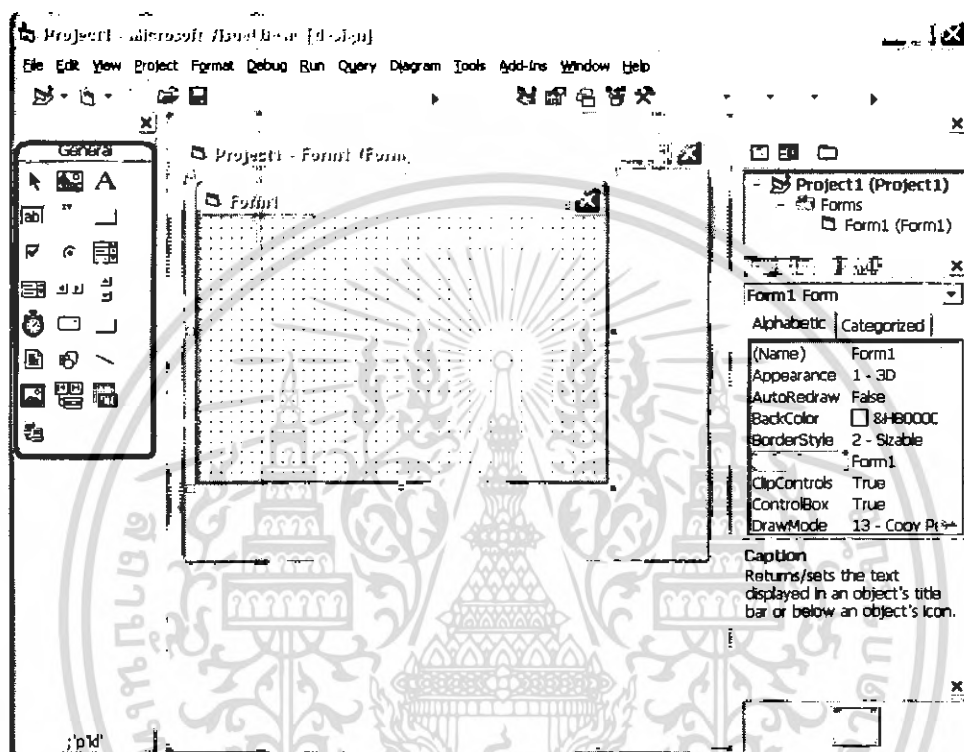


รูปที่ 3.1 ไดอะแกรมของโครงการทั้งหมด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.2 การออกแบบในส่วนของโปรแกรม Visual Basic

ในส่วนของการออกแบบโปรแกรม Visual Basic เป็นส่วนที่สำคัญที่จะใช้ในการส่งข้อมูลการควบคุมอุปกรณ์ไฟฟ้าให้กับชุดไมโครคอนโทรลเลอร์ ตัวอุปกรณ์ที่ใช้ในการออกแบบจะเป็นแบบพื้นฐานซึ่งมีบรรจุอยู่แล้วในแถบเครื่องมือ (Tool Box) บนหน้าต่างของโปรแกรม Visual Basic



รูปที่ 3.2 แถบเครื่องมือ (Tool Box) และอุปกรณ์พื้นฐานบนหน้าต่างโปรแกรม

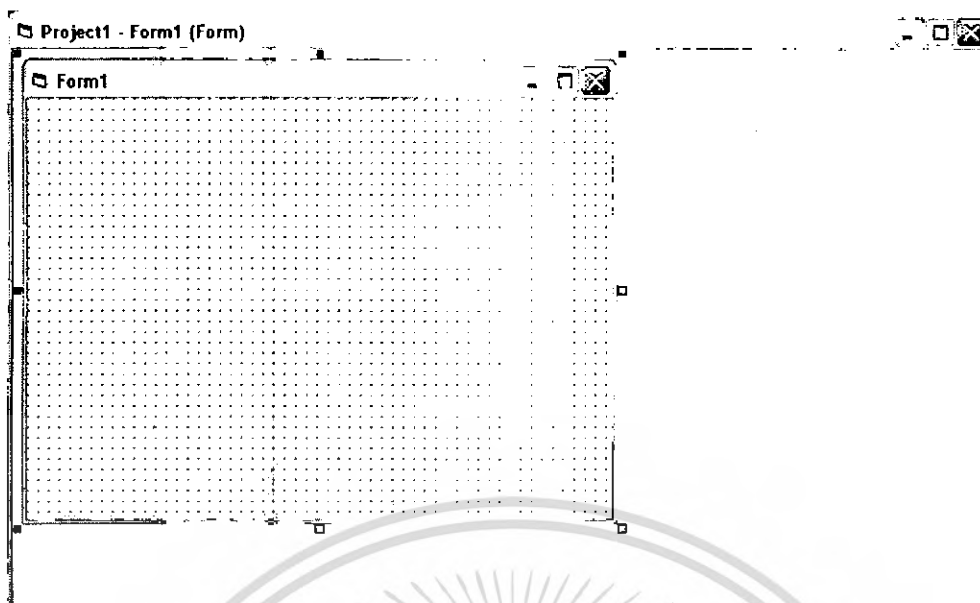
แถบเครื่องมือของโปรแกรม Visual Basic ที่เลือกมาใช้งานมีดังต่อไปนี้

3.2.1 Form

Form เป็นออบเจกต์แรกสุดที่โปรแกรม Visual Basic จะติดตั้งให้เองโดยอัตโนมัติ และเป็นออบเจกต์ที่ไม่ได้อยู่บน Tool Box แต่จะนำขึ้นมาหลังจากได้เรียกโปรแกรม Visual Basic ขึ้นมาเป็นที่เรียบร้อยแล้ว

Form จะประกอบไปด้วย คุณสมบัติ (Property), กฎเกณฑ์ (Method), และเหตุการณ์ (Event) ต่าง ๆ ที่ช่วยในการเขียนโปรแกรมเพื่อควบคุมการทำงานของฟอร์มตามที่เรากำลังต้องการ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.3 ฟอর্মที่ปรากฏขึ้นมาเมื่อเริ่มเปิดโปรแกรม

ฟอर्मเป็นเหมือนกับแผ่นกระดาษเปล่า ๆ แผ่นหนึ่งจะแต่งเติมอะไรลงไปก็ได้ ไม่ว่าจะ เป็นข้อความ ช่องกรอกข้อมูล ปุ่มสั่งการทำงาน หรือการแสดงรูปหรือผลการทำงานต่าง ๆ ซึ่ง สิ่งต่าง ๆ เหล่านี้จะเป็นส่วนตอบโต้หรือแสดงผลให้กับผู้ใช้โปรแกรม ดังนั้นการทำงานบน โปรแกรม Visual Basic อันดับแรกต้องเป็นส่วนของฟอर्मเสมอ

3.2.2 Frame

Frame เป็นคอนโทรลใช้สำหรับจัดกลุ่มของคอนโทรลต่าง ๆ ที่ได้ทำการออกแบบไว้บน ฟอर्मให้อยู่ในกลุ่มหรือหมวดหมู่เดียวกัน และมีชื่อกำกับโดยไม่จำเป็นว่าในเฟรมนั้นจะต้องมี คอนโทรลกี่ตัว คอนโทรลเฟรมมีประโยชน์ในการแบ่งกลุ่มคำสั่ง เพื่อความสะดวกในการเลือก ทำงานสำหรับผู้ใช้โปรแกรม อีกประการหนึ่งคือเพื่อที่เวลาเกิดความผิดพลาด การแก้ไขหรือการ เคลื่อนย้ายตำแหน่งของคอนโทรลต่าง ๆ ก็จะทำให้ง่ายและรวดเร็ว

xy

รูปที่ 3.4 สัญลักษณ์คอนโทรล Frame

การเลือก Frame มาใช้งานนั้นสามารถลากมาจากแถบเครื่องมือ (Tool Box) ออกมา วางบนฟอर्म และสามารถลากกำหนดขนาดได้ตามความต้องการ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Property ที่สำคัญสำหรับ Frame ประกอบด้วย Caption, Enabled, Font, ForeColor, BackColor และ BorderStyle

Caption : เป็นพรอเพอร์ตี้สำหรับเก็บข้อความเอาไว้แสดงเป็นหัวข้อของเฟรม ตามปกติแล้วถ้าไม่มีการเปลี่ยนแปลงจะเป็นข้อความที่ชื่อ Frame1 แต่ถ้าเราต้องการจะเปลี่ยนข้อความ ก็ให้พิมพ์ข้อความที่ต้องการให้เป็นหัวข้อลงไปในพรอเพอร์ตี้ Caption ที่ Property Box

Enabled : เป็นพรอเพอร์ตี้มีหน้าที่คืนค่า หรือกำหนดให้เฟรม สามารถตอบรับหรือตอบสนองเหตุการณ์ต่าง ๆ ที่อาจเกิดจากผู้ใช้ การกำหนดค่าการใช้งานนั้นจะสามารถกำหนดได้ด้วยกัน 2 ค่า คือ True กับ False ถ้าเป็น True จะยอมให้เฟรมสามารถตอบสนองเหตุการณ์ต่าง ๆ ได้ตามปกติ แต่ถ้าเป็น False จะไม่อนุญาตให้เฟรมตอบสนองเหตุการณ์ใด ๆ ที่เกิดขึ้นทั้งสิ้น

Font and ForeColor : เป็นพรอเพอร์ตี้สำหรับปรับแต่งตัวหนังสือที่แสดงบนพรอเพอร์ตี้ Caption อย่างเดียว

BackColor : เป็นพรอเพอร์ตี้สำหรับกำหนดสีจากหลังให้กับเฟรม

BorderStyle : เป็นพรอเพอร์ตี้สำหรับกำหนดเส้นขอบของเฟรม โดยจะมีค่าเป็นตัวเลข คือ 0 หรือ 1 ที่สามารถกำหนดจาก Property Box ถ้าเลือก 0 จะไม่แสดงเส้นขอบของเฟรม รวมทั้งข้อความที่แสดงที่ Caption แต่ถ้าเลือก 1 จะแสดงเส้นขอบตามปกติ

3.2.3 OptionButton

OptionButton เป็นคอนโทรลที่มีลักษณะเป็นปุ่ม ใช้สำหรับเลือกอย่างใดอย่างหนึ่ง OptionButton จะต้องมียังน้อย 2 ตัวหรือมากกว่าก็ได้ ถ้า OptionButton ตัวใดถูกเลือกแล้ว ตัวที่เหลือก็就会被เลือกไม่ได้อีกโดยอัตโนมัติ

รูปที่ 3.5 สัญลักษณ์คอนโทรล OptionButton

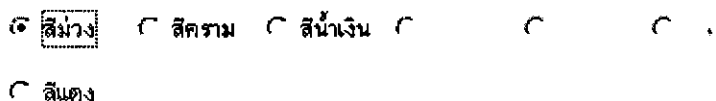
การเลือก OptionButton ออกมาใช้งานนั้นสามารถลากออกมาจากแถบเครื่องมือ (Tool Box) โดยทั่วไปจะใช้ร่วมกับFrame ที่ถูกวางบนฟอร์มอยู่ก่อนแล้ว โดยการกำหนดชื่อของสิ่งที่ต้องการให้เลือก

ค่าสถานะของ OptionButton จะมีค่าเป็น True หรือ False เท่านั้น โดยจะมีค่าเป็น True เมื่อเราคลิกที่ปุ่มกลม ๆ จนมีจุดดำแสดงอยู่บนกึ่งกลางของวงกลม และจะเป็น False โดยอัตโนมัติเมื่อเราไปคลิกที่ปุ่ม OptionButton อื่นอื่น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ในโครงงานนี้เราจะใช้ `OptionButton` ร่วมกับ `Frame` ในการเลือกสีที่แสดงออกของการผสมสี ซึ่งจะมีให้เลือก 7 สีด้วยกัน

๑ ปรับ - แดง - ผสมสี



รูปที่ 3.6 การเลือกสีที่จะทำการควบคุม

3.2.4 TextBox

`TextBox` เป็นคอนโทรลใช้สำหรับบันทึกข้อความในรูปแบบตัวเลขและตัวอักษร ขณะที่โปรแกรมทำงาน ผู้ใช้งานสามารถพิมพ์ข้อความหรือตัวเลขลงใน `TextBox` ได้

รูปที่ 3.7 สัญลักษณ์คอนโทรล TextBox

การเลือก `TextBox` ออกมาใช้งานนั้นสามารถลากออกมาจากแถบเครื่องมือ (`Tool Box`) โดยทั่วไปก็จะใช้งานร่วมกับ `Frame` ที่ถูกวางบนฟอร์มอยู่ก่อนแล้วเช่นเดียวกับ `OptionButton`

กำหนดค่า

IP Address :

Port :

รูปที่ 3.8 TextBox สำหรับใส่ค่า IP Address และ Port

ในโครงงานนี้จะใช้ `TextBox` ร่วมกับ `Frame` ในการใส่ค่าหรือกำหนดค่าหมายเลข IP Address และหมายเลข Port เพื่อที่จะทำการติดต่อ ส่งข้อมูล และควบคุมการทำงานอุปกรณ์ไฟฟ้าผ่านระบบ LAN

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

พรอเพอร์ตี้ที่สำคัญสำหรับ TextBox ประกอบด้วย BorderStyle, Enabled และ Name
 BorderStyle : เป็นพรอเพอร์ตี้ที่เกี่ยวกับการติกรอบของคอนโทรลที่ใช้ ซึ่งมีใช้ในคอนโทรลต่อไปนี้คือ From, Picture, TextBox และ Label โดยค่าของ BorderStyle จะมีอยู่ 2 ค่า คือ 0 และ 1 ถ้าเลือก 0 จะไม่แสดงเส้นกรอบ ถ้าเป็น 1 จะแสดงเส้นกรอบ

Enabled : เป็นพรอเพอร์ตี้สำหรับให้คอนโทรลตัวนั้นทำงานได้หรือไม่ โดยจะมีค่าเป็น True หรือ False ถ้าเป็น True จะทำงาน แต่ถ้าเป็น False จะไม่ทำงาน ในส่วนของ TextBox มีความสำคัญคือ ถ้าเป็น False โปรแกรมจะไม่สามารถนำค่าตัวเลขหรือข้อความที่ใส่ไปกระทำได้ จะเป็นเพียงแค่การแสดงให้เห็นว่าใส่อะไรลงไปเท่านั้น

Name : เป็นพรอเพอร์ตี้ที่ใช้ในเกือบทุกคอนโทรล โดยปกติใน TextBox จะแสดงคำว่า Text1 หากต้องการเปลี่ยน ไปเปลี่ยนได้ที่พรอเพอร์ตี้ Name ที่ Property Box ซึ่งการเปลี่ยนนั้นจะเปลี่ยนเป็นไม่มีข้อความเลยเพื่อที่จะรอข้อความที่ผู้ใช้งานจะใส่ลงไปก็ได้ แต่หากข้อความที่ใส่ลงไปนั้นเป็นตัวเลข ให้เปลี่ยนจากเดิมเป็นเลข 0

3.2.5 ComboBox

ComboBox เป็นคอนโทรลที่เปิดโอกาสให้ผู้ใช้โปรแกรมนั้นสามารถทำการเลือกค่าที่แอปพลิเคชันกำหนดไว้ แทนที่จะพิมพ์ค่านั้นเข้าไปโดยตรงในลักษณะเช่นเดียวกับ TextBox ซึ่งจะทำให้ผู้ใช้โปรแกรมมีความสะดวกในการทำงานมากกว่าเดิม



รูปที่ 3.9 สัญลักษณ์คอนโทรล ComboBox

การเลือก TextBox ออกมาใช้งานนั้นสามารถลากออกมาจากแถบเครื่องมือ (Tool Box) โดยทั่วไปก็จะใช้งานร่วมกับ Frame ที่ถูกวางบนฟอร์มอยู่ก่อนแล้วเช่นเดียวกันกับ OptionButton และ TextBox

พรอเพอร์ตี้ที่สำคัญสำหรับ ComboBox บางพรอเพอร์ตี้ก็จะมีลักษณะที่คล้ายกันกับ TextBox แต่ที่ไม่เหมือนกันก็คือ List

List : สามารถกำหนดค่าลงไป ใน ComboBox ได้โดยการเพิ่มข้อความหรือข้อมูลที่ต้องการลงไปได้ List ใน Property Box ได้เลย

3.2.6 Label

Label เป็นคอนโทรลที่ใช้ในการนำเสนอข้อความบนฟอร์มคล้ายกับ TextBox แต่ว่า Label จะไม่สามารถรับข้อมูลที่ต้องการใส่ลงไปได้ มันจะแสดงข้อความหรือตัวเลขที่ใส่ไว้ตอนเขียนโปรแกรมหรือข้อมูลที่ได้จากการกระทำของโปรแกรมมาแสดงให้เห็น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.10 สัญลักษณ์คอนโทรล Label

การเลือก Label ออกมาใช้งานนั้นสามารถลากออกมาจากแถบเครื่องมือ (Tool Box) มาวางบนฟอร์มและลากกำหนดขนาดได้ตามต้องการ

พรอเพอร์ตี้ที่สำคัญของ Label คือ Caption, Alignment และ BorderStyle

Caption : มีหน้าที่กำหนดข้อความที่ใช้บรรยายคอนโทรลนั้น ซึ่งสามารถจะบรรยายได้ใน Property Box ที่ Caption ของ Label ข้อความที่พิมพ์ลงไปก็จะปรากฏใน Label นั้น ๆ

Alignment : เป็นคำสั่งสำหรับจัดตัวอักษรให้ชิดซ้าย ชิดขวา หรือกึ่งกลาง เมื่อคลิกที่พรอเพอร์ตี้ Alignment ใน Property Box ก็จะพบตัวเลือก 3 ตัวเลือก คือ 0 1 และ 2 ถ้าเป็น 0 จะชิดซ้าย ถ้าเป็น 1 จะชิดขวา และถ้าเป็น 2 จะอยู่กึ่งกลาง

BorderStyle : เป็นพรอเพอร์ตี้ที่เกี่ยวกับการติดกรอบของคอนโทรลที่ใช้ โดยค่าของ BorderStyle จะมีอยู่ 2 ค่า คือ 0 และ 1 ถ้าเลือก 0 จะไม่แสดงเส้นกรอบ ถ้าเป็น 1 จะแสดงเส้นกรอบ

ในโครงการนี้การทำงานของ Label จะขึ้นอยู่กับเหตุการณ์ที่เกิดขึ้นตามที่เรากำหนดการคือค่าที่ Label แสดงจะเป็นค่าที่เกิดขึ้นจากการทำงานของโปรแกรมนั้นได้ทำไปแล้ว

3.2.7 CommandButton

CommandButton เป็นคอนโทรลที่มีลักษณะเป็นปุ่มสี่เหลี่ยม ออกแบบมาเพื่อให้ใช้เมาส์คลิกตัวหนังสือที่อยู่บนปุ่มที่คลิก



รูปที่ 3.11 สัญลักษณ์คอนโทรล CommandButton

พรอเพอร์ตี้ที่สำคัญได้แก่ พรอเพอร์ตี้ที่ใช้ในการแก้ไขเปลี่ยนแปลงข้อความบนปุ่ม พรอเพอร์ตี้ที่ใช้ในการเปลี่ยนแปลงรูปแบบตัวอักษรที่แสดงข้อความ ซึ่งมีดังต่อไปนี้

Caption : มีหน้าที่กำหนดข้อความต่าง ๆ บนปุ่ม โดยปกติจะมีค่าชื่อเป็น CommandButton1 การเปลี่ยนแปลงสามารถทำได้ที่ Property Box ที่ Caption ของ CommandButton ข้อความที่พิมพ์ลงไปก็จะปรากฏใน CommandButton นั้น ๆ

Font : เป็นพรอเพอร์ตี้สำหรับปรับแต่งตัวหนังสือที่แสดงบนพรอเพอร์ตี้ Caption สามารถปรับปรุงขนาดได้ตามต้องการ

Style and BackColor : เป็นพรอเพอร์ตี้ที่ใช้ในการเปลี่ยนสีของ CommandButton โดยเราจะใช้พรอเพอร์ตี้ 2 ตัวนี้ในการเปลี่ยนสี เริ่มจากการไปที่ พรอเพอร์ตี้ Style ซึ่งจะมีค่าให้เลือกระหว่าง 0 คือไม่ปรับเปลี่ยน กับ 1 คือการปรับเปลี่ยน เลือก 1 แล้วไปที่พรอเพอร์ตี้ชื่อ BackColor แล้วทำการเลือกสีของ CommandButton ตามต้องการ

ในโครงการนี้ใช้ CommandButton เป็นปุ่มกดเพื่อทำการติดต่อและส่งข้อมูลออกทาง Port LAN ของเครื่องคอมพิวเตอร์ ดังแสดงในรูปที่ 3.11

Connect

รูปที่ 3.12 ปุ่ม Connect โดยใช้คอนโทรล CommandButton

3.2.8 HScrollBar

HScrollBar เป็นคอนโทรลที่มีลักษณะเป็นตัวเลื่อนใช้สำหรับการเปลี่ยนแปลงค่า โดยการเคลื่อนย้ายตัวเลื่อนบนแถบบาร์ (Bar)



รูปที่ 3.13 สัญลักษณ์คอนโทรล HScrollBar

พรอเพอร์ตี้ที่สำคัญคือ Min, Max, SmallChange และ Value

Min : ใช้สำหรับการกำหนดค่าต่ำสุดของ HScrollBar

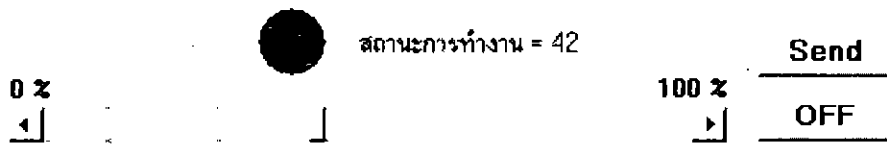
Max : ใช้สำหรับการกำหนดค่าสูงสุดของ HScrollBar

SmallChange : ใช้สำหรับกำหนดค่าที่จะมีการเปลี่ยนแปลงไป เมื่อผู้โปรแกรมคลิกที่ปุ่มลูกศรชี้ขวาและซ้ายของ HScrollBar

Value : ใช้สำหรับกำหนดค่าเริ่มต้นของ HScrollBar โดยปกติจะมีค่าเท่ากับค่า Min ของ HScrollBar ซึ่งมีค่าปกติอยู่ที่ 0

ในโครงการนี้จะใช้ HScrollBar เป็นตัวเลื่อนเพื่อปรับเปลี่ยนความสว่างของแสง โดยจะใช้ทั้งหมด 4 ชุด ด้วยกัน

ปรับความสว่าง



รูปที่ 3.14 การปรับความสว่างโดยใช้คอนโทรล HScrollBar

3.2.9 Timer

Timer เป็นคอนโทรลสำหรับตั้งเวลาทำงาน หน่วงเวลาการทำงาน โดยจะทำงานเป็นรอบ เมื่อจบรอบหนึ่งก็จะเริ่มรอบใหม่ในเวลาที่ตั้งไว้ หรืออาจจะให้จบการทำงานเลยก็ได้ ซึ่งสามารถจะกำหนดได้ในพรอเพอร์ตี้

รูปที่ 3.15 สัญลักษณ์คอนโทรล Timer

สำหรับ Timer เมื่อทำการติดตั้งลงบนฟอร์มแล้ว เวลาโปรแกรมทำงานจะไม่แสดงคอนโทรล Timer ให้เห็น ดังนั้นเราสามารถนำไปติดตั้งไว้ที่ใดบนฟอร์มก็ได้

พรอเพอร์ตี้ที่สำคัญของ Timer มีดังนี้

Interval : เป็นพรอเพอร์ตี้สำหรับตั้งเวลาแก่ Timer ซึ่งมีค่าเป็นตัวเลข โดยที่หนึ่งหน่วยจะมีค่าหนึ่งในพันของวินาที (1 หน่วย = 1/1000 วินาที) ดังนั้นหากต้องการตั้งเวลา 1 วินาที จะต้องกำหนดค่า Interval เป็น 1000 การกำหนดสามารถกำหนดได้ใน Property Box หรือเขียนเป็นโปรแกรมก็ได้

Enabled : เป็นพรอเพอร์ตี้สำหรับให้คอนโทรลนั้นทำงานได้หรือไม่ โดยมีค่าเป็น True หรือ False ถ้าเป็น True นั้น Timer ก็ทำงาน ถ้าเป็น False Timer ก็ไม่ทำงาน

3.2.10 Winsock

Winsock เป็นคอนโทรลที่ทำหน้าที่จัดการเกี่ยวกับการติดต่อสื่อสารโดยใช้ Protocol ที่อยู่บน IP (Internet Protocol) โดยสามารถสร้างการติดต่อสื่อสารที่เป็น TCP (Transfer Control Protocol) หรือ UDP (User Datagram Protocol)



รูปที่ 3.16 สัญลักษณ์คอนโทรล Winsock

พารามิเตอร์ที่สำคัญของ Winsock

RemoteHost : เป็นพารามิเตอร์ที่ต้องกำหนดค่า IP Address ของเครื่องปลายทางที่ต้องการติดต่อสื่อสารกัน

RemotePort : เป็นพารามิเตอร์ที่ต้องกำหนดค่า Local Port ของเครื่องปลายทางที่จะทำการติดต่อสื่อสารด้วย

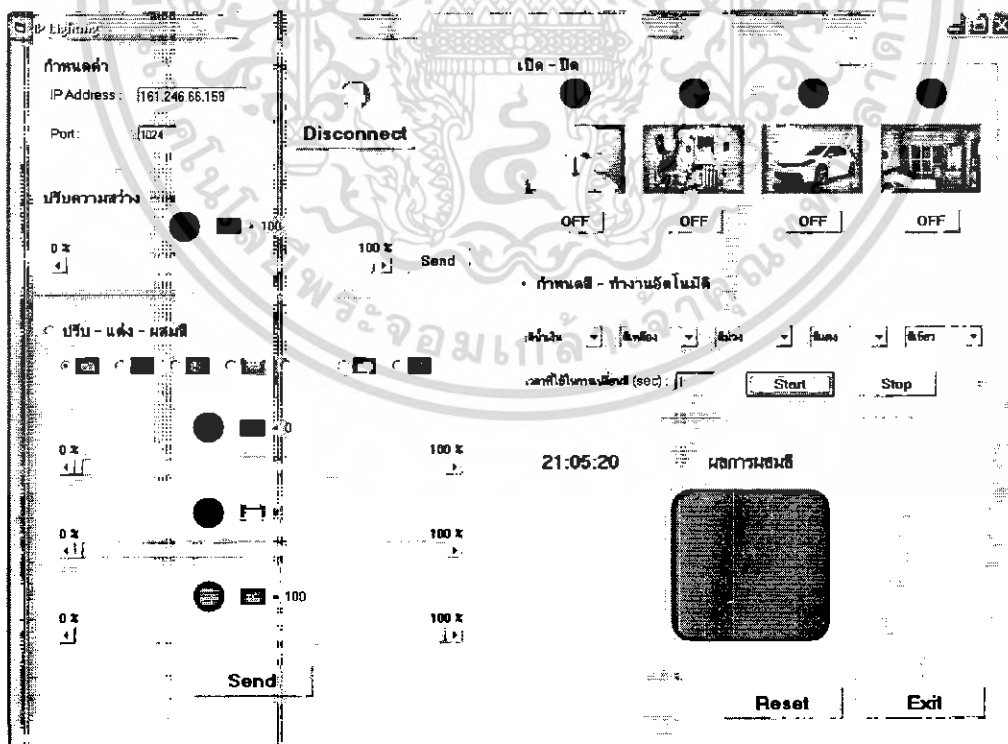
Methods ที่สำคัญของ Winsock

Close : ทำหน้าที่ปิดการติดต่อ หรือปิดการรอรับการติดต่อ

Connect : ทำหน้าที่สร้างการเชื่อมต่อของ Protocol ไปยังหมายเลข RemotePort ที่อยู่บน RemoteHost

SendData : ทำหน้าที่ส่งข้อมูลไปยังหมายเลข RemoteHost

3.2.11 หน้าต่างโปรแกรม Control



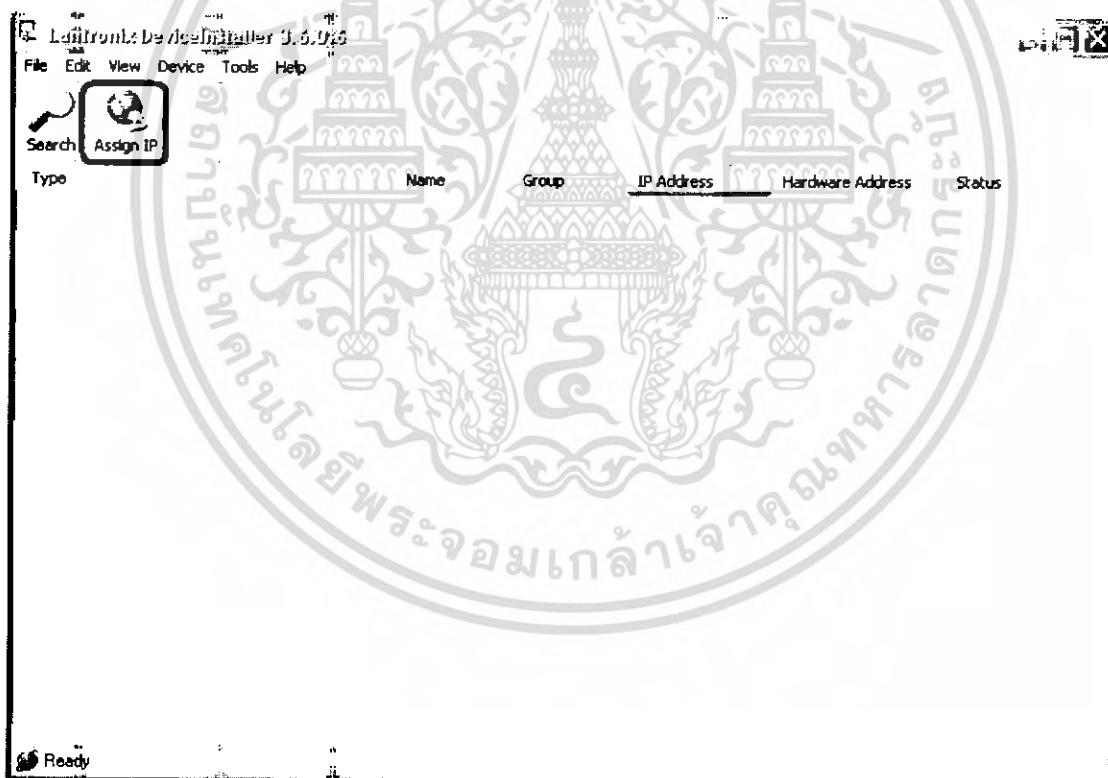
รูปที่ 3.17 หน้าต่างโปรแกรม Control

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.3 การกำหนดค่า IP Address ให้กับ ET-XPORT V1

ก่อนที่จะนำ ET-XPORT V1 ไปต่อใช้งานร่วมกับอุปกรณ์อื่นๆในเครือข่ายได้นั้น ในอันดับแรกจะต้องทำการกำหนดค่า Configuration ให้กับตัวเครื่องให้เรียบร้อยเสียก่อน โดยเฉพาะอย่างยิ่ง การกำหนดค่าหมายเลข IP Address ประจำตัวให้กับ ET-XPORT V1 เพื่อใช้ในระบบเครือข่าย โดยในการกำหนดค่าให้กับสำหรับวิธีการกำหนดค่า Configuration ให้กับ ET-XPORT V1 นั้น จะใช้โปรแกรม “DeviceInstaller” เป็นตัวจัดการ โดยลำดับขั้นตอนของการกำหนดค่า Configuration ต่าง ๆ มีดังนี้

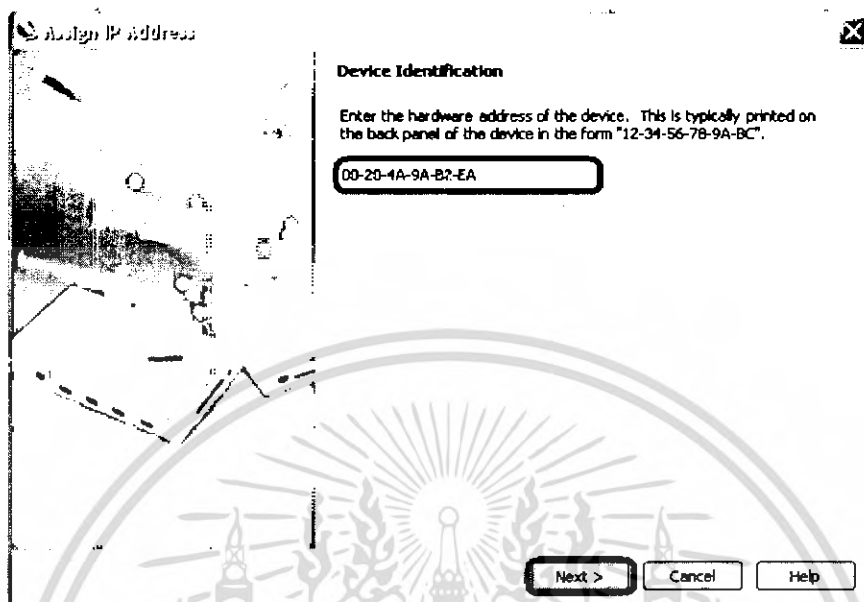
1. ต่อสายสัญญาณ RJ45 ของชุด ET-XPORT V1 เข้ากับเครือข่าย โดยถ้าต่อผ่าน HUB จะต้องใช้สายสัญญาณแบบต่อตรง (Direct) แต่ถ้าต่อโดยตรงกับเครื่องคอมพิวเตอร์จะต้องใช้สายสัญญาณแบบสลับ (Peer-to-Peer) พร้อมทั้งจ่ายไฟเลี้ยงวงจรให้กับตัวเครื่องให้เรียบร้อย
2. เปิดโปรแกรม “DeviceInstaller” โดยไปที่ **“Start Menu” > “Programs” > “Lantronix” > “DeviceInstaller” > “DeviceInstaller”**
3. คลิกที่ Icon **Assign IP** ดังรูปที่ 3.17



รูปที่ 3.18 แสดงขั้นตอนการ Assign IP

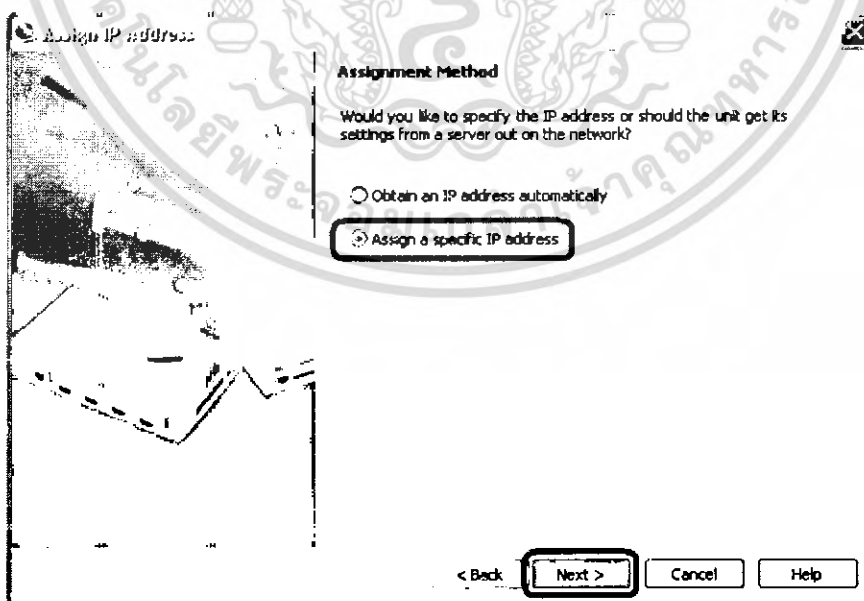
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4. กรอกค่า **MAC Address** ซึ่งจะระบุอยู่ที่ตัวกล่อง ET-XPORT V1 ดังตัวอย่างในรูป แล้วคลิก **Next >**



รูปที่ 3.19 แสดงขั้นตอนการกรอกค่า MAC Address

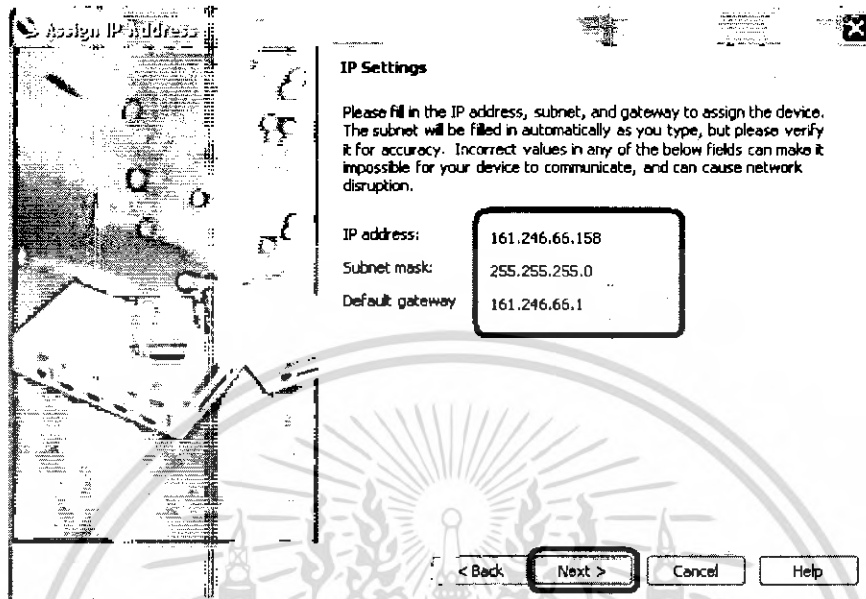
5. เลือก **Assign a specific IP address** เพื่อกำหนดค่า IP แบบคงที่ให้กับ ET-XPORT แล้วคลิก **Next >**



รูปที่ 3.20 แสดงขั้นตอนการ Assign a specific IP address

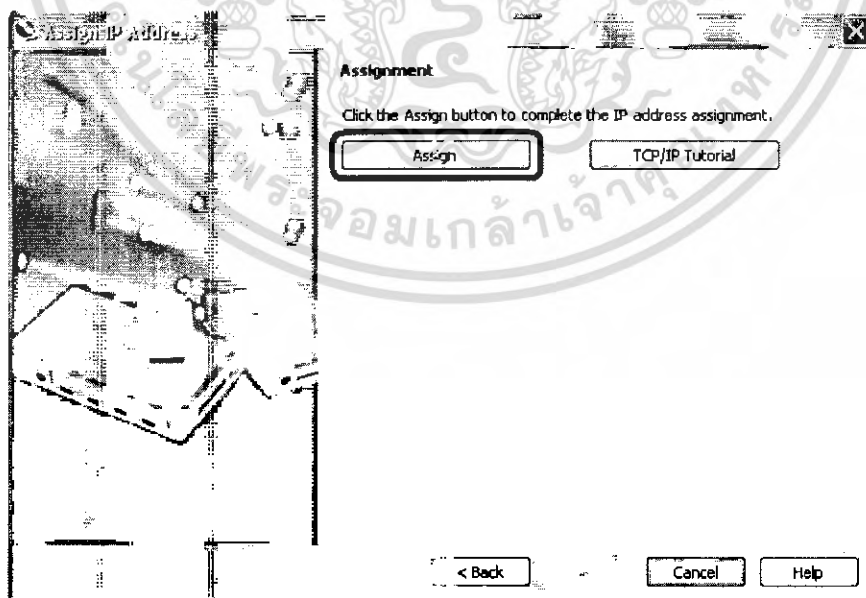
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

6. กรอกค่า **IP Address, Subnet Mark** ส่วน **Gateway** จะกรอกหรือไม่ก็ได้ ซึ่งมีค่าเริ่มต้นเป็น 0



รูปที่ 3.21 แสดงขั้นตอนการกรอกค่า IP Address, Subnet Mark และ Gateway

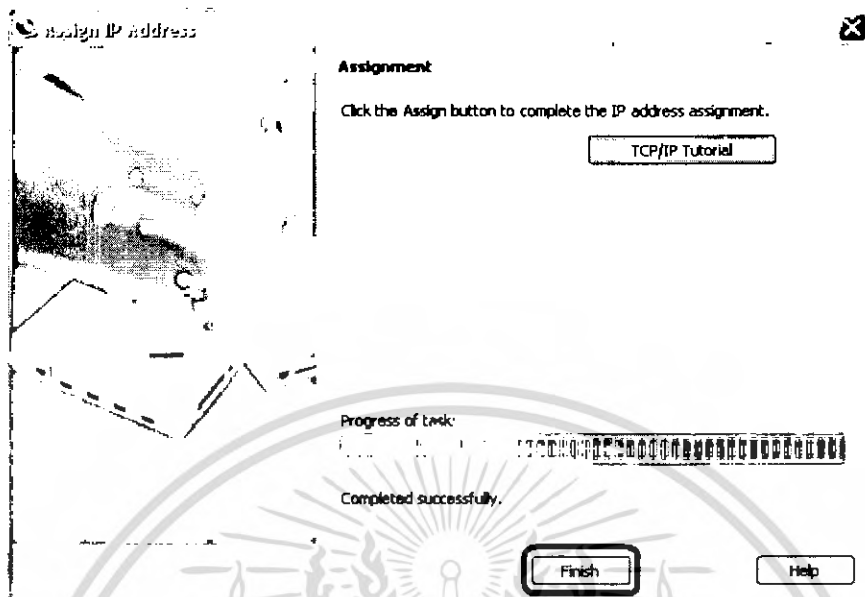
7. จากนั้นโปรแกรมจะให้เรา Assign IP คือตั้งค่าให้กับ ET-XPORT โดยคลิกที่ **Assign** รอสักครู่



รูปที่ 3.22 แสดงขั้นตอนการ Assign

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

8. เมื่อสักครู่นี้เสร็จสิ้นกระบวนการ ดังรูป ให้คลิก **Finish**



รูปที่ 3.23 สิ้นสุดขั้นตอนการกำหนดค่า IP Address ให้กับ ET-XPORT V1



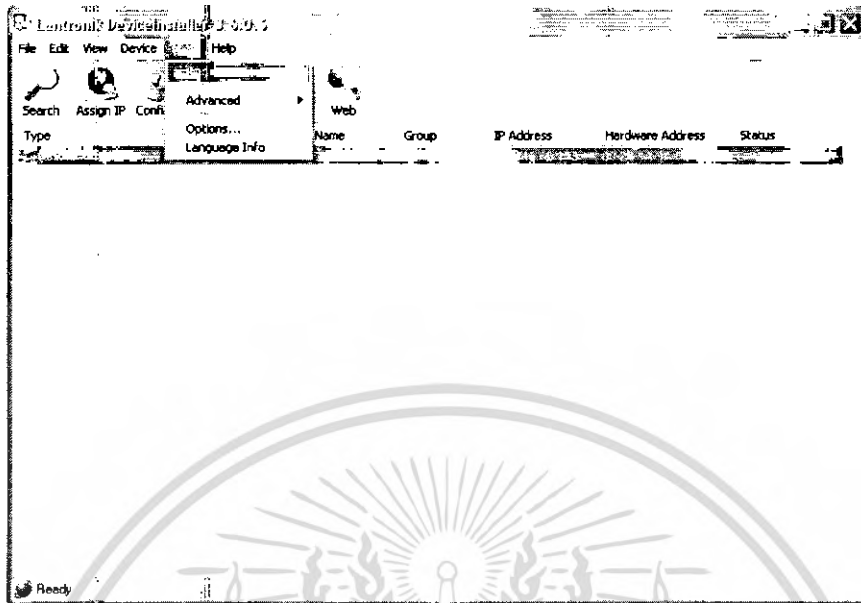
รูปที่ 3.24 แสดงการตรวจสอบสถานะการเชื่อมต่อ

3.4 การทดสอบการเชื่อมต่อกับเครือข่าย Network ด้วยการ Ping

หลังจากที่ทำการกำหนดค่า IP Address ให้กับชุด ET-XPORT V1 เป็นที่เรียบร้อยแล้ว ในเบื้องต้น เราสามารถทำการทดสอบการเชื่อมต่อสื่อสารข้อมูลของ ET-XPORT V1 กับเครือข่ายได้ดังตัวอย่างต่อไปนี้

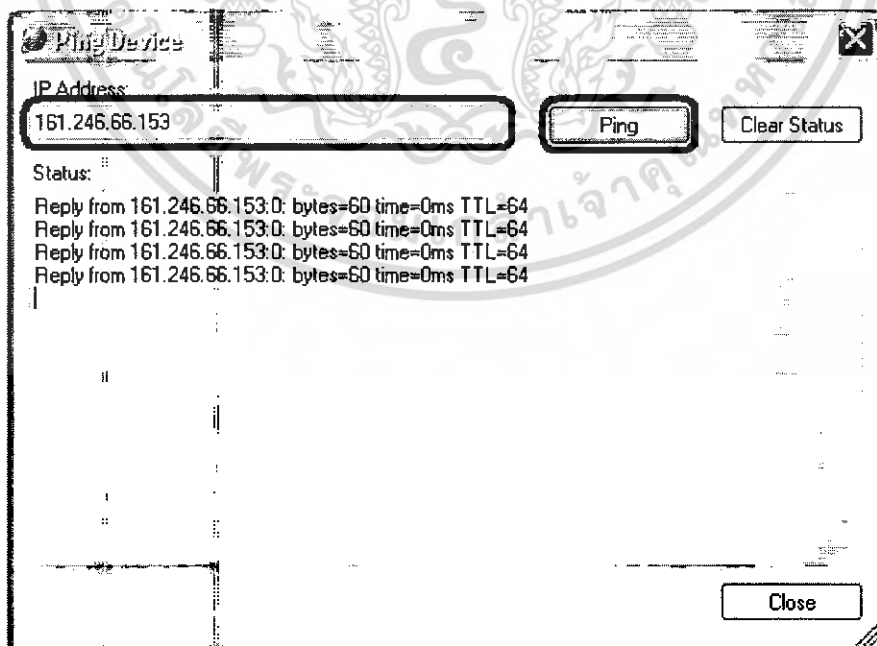
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1. คลิกเมาส์ที่คำสั่ง **Tools > Ping...** ดังรูป



รูปที่ 3.25 แสดงขั้นตอนการทดสอบการเชื่อมต่อกับเครือข่าย Network ด้วยการ Ping

2. จากนั้นให้เราระบุ **IP Address** ของ ET-XPORT V1 ตามที่เราได้ Assign IP ไป แล้วจึงคลิกที่ **Ping** ถ้าการติดต่อกับ ET-XPORT V1 เป็นผลสำเร็จ ที่ช่อง Status จะแสดงผล ดังรูป จากนั้นปิดหน้าต่างนี้



รูปที่ 3.26 แสดงผลการทดสอบการเชื่อมต่อกับเครือข่าย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.5 การกำหนดค่า Configuration ให้กับ ET-XPORT V1 ด้วยโปรแกรม DeviceInstaller

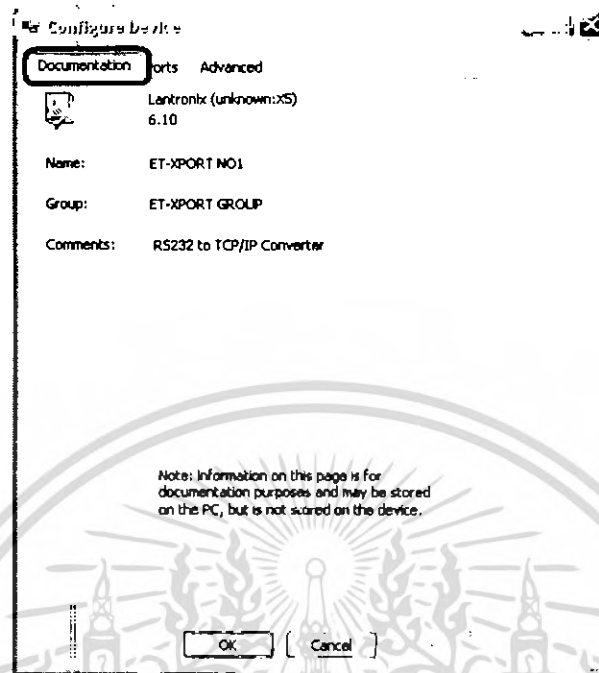
โปรแกรม DeviceInstaller นอกจากจะใช้สำหรับทำการกำหนดค่า IP Address ให้กับ ET-XPORT V1 แล้วยังสามารถใช้ทำการกำหนดค่า Configuration ต่างๆให้กับ ET-XPORT V1 ได้อีกด้วย โดยในกรณีนี้ต้องทำการกำหนดค่า IP Address ให้กับ ET-XPORT V1 ให้เรียบร้อยเสียก่อนจากนั้นจึงจะสามารถสั่งกำหนดค่าการทำงานต่างๆให้กับ ET-XPORT V1 ตามหมายเลข IP Address ที่ต้องการได้ ซึ่งลำดับขั้นตอนการ Setup ค่ามีดังนี้

1. สั่ง Run โปรแกรม DeviceInstaller โดยเลือกที่เมนูคำสั่ง Start > Program > Lantronix > DeviceInstaller > DeviceInstaller
2. เชื่อมต่อ ET-XPORT V1 เข้ากับเครือข่ายเดียวกับเครื่องคอมพิวเตอร์ PC ที่ใช้ Run โปรแกรม พร้อมทั้งจ่ายไฟให้ ET-XPORT V1 พร้อมทำงาน
3. คลิกเมาส์ที่คำสั่ง Search ของโปรแกรม DeviceInstaller เพื่อค้นหา ET-XPORT V1 ที่ต่ออยู่กับเครือข่ายอยู่ในขณะนั้น
4. คลิกเมาส์ที่รายการของ ET-XPORT V1 ที่ต้องการ Setup ค่าจะปรากฏแถบสีฟ้าที่รายการนั้น พร้อมกับคลิกเมาส์ที่ Configure ดังรูป



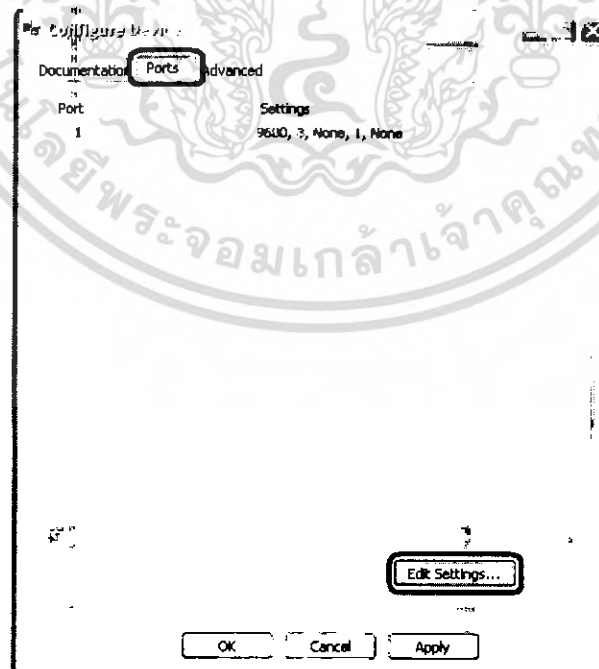
รูปที่ 3.27 แสดงหน้าต่างการกำหนดค่า Configuration ให้กับ ET-XPORT V1 ด้วยโปรแกรม DeviceInstaller

5. ที่ Tab ของ Documentation ทดลองกำหนดรายละเอียดต่างๆ ดังรูป



รูปที่ 3.28 กำหนดชื่อให้กับ ET-XPORT V1

6. คลิกเมาส์ไปที่ Tab ของ Ports พร้อมกับเลือกที่ "Edit Setting..." พร้อมกับเลือกกำหนดค่าการทำงานของ RS232 (Channel1) ตามต้องการ ดังรูปในตัวอย่าง



รูปที่ 3.29 แสดงการกำหนดค่าการทำงานทางด้าน RS232

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.6 การออกแบบโปรแกรม Microcontroller

ทำโดยเริ่มจากรับค่าข้อมูลขนาด 8 Bit จากรูปแบบสัญญาณอนุกรมที่ทำการส่งงานจาก Program Control (Visual Basic 6.0) เมื่อรับค่ามาแล้วตรวจสอบเงื่อนไขเพื่อทำให้โปรแกรมสามารถส่งงานการทำงานของ Output ได้ตามต้องการ

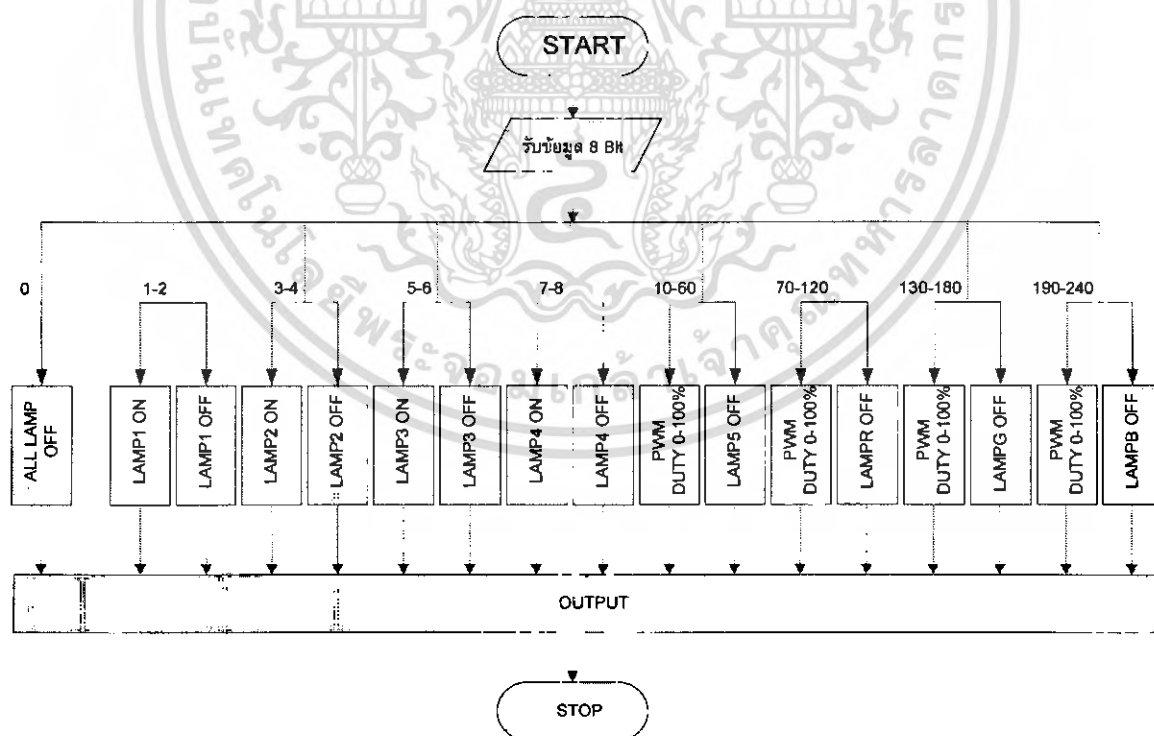
ข้อมูลจาก Program Control ในช่วง 1 ถึง 8 จะเป็นการ เปิด ปิด การทำงานของ อุปกรณ์ 4 ตัว ซึ่งเราสามารถนำไปใช้กับอุปกรณ์ที่ใช้ที่มีการแค่ Fuction เปิด ปิด เท่านั้น

ข้อมูลจาก Program Control ในช่วง 10 ถึง 60 จะเป็นการควบคุมหลอด LED ที่สามารถปรับความสว่างได้โดยการควบคุมสัญญาณ PWM ที่ออกจาก Microcontroller โดยสัญญาณที่รับมาจะนำมาควบคุม Duty Cycle จาก 0 ถึง 100 %

ข้อมูลจาก Program Control ในช่วง 70 ถึง 190 จะใช้ในการควบคุมการทำงานของ หลอด LED 3 สีในคอมเดียวกัน จึงทำให้สามารถควบคุมสีได้ตามใจชอบเพราะทั้ง 3 ชุดสามารถทำการปรับความสว่างได้จึงเกิดสีได้หลากหลายมาก

ข้อมูล 0 จะเป็นการหยุดการทำงานทั้งหมดของอุปกรณ์ทุกตัว

Flowchart การทำงานของ Microcontroller



รูปที่ 3.30 Flow Chart การทำงานของ Microcontroller

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 4

ผลการทดลอง

จากการออกแบบการทำงานของระบบและสร้างชิ้นงานขึ้นมาจึงได้ทดสอบการทำงานของอุปกรณ์โดยการทำ Test Run จริงว่ามีประสิทธิภาพในการทำงานเป็นอย่างไร และได้ทำการเก็บค่าบางส่วนเพื่อการวิเคราะห์การทำงานของอุปกรณ์ว่าเป็นไปตามความต้องการตามการออกแบบหรือไม่

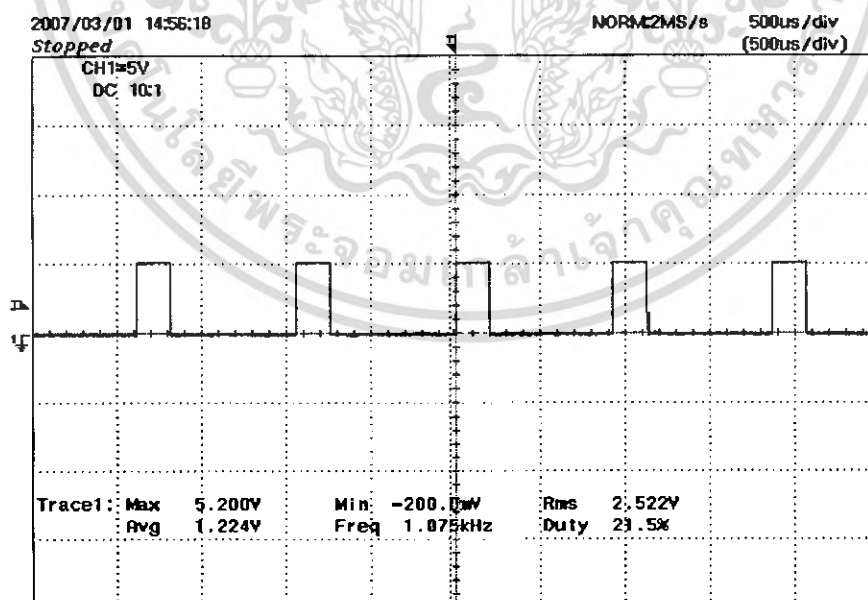
4.1 การทดสอบการทำงานของชุด เปิด ปิด หลอดไฟ

จากการทดสอบการทำงานจะเห็นว่าชุด Microcontroller สามารถทำตามคำสั่งของ Program Control ได้ทุกคำสั่งทั้ง 4 ชุด

4.2 การทดสอบการทำงานของชุดปรับความสว่างของ LED

จากการทดสอบโดยทำการต่อกับวงจรขับ LED แล้วสั่งการทำงานโดยปรับเปลี่ยนค่า Duty Cycle ของสัญญาณ PWM ที่จ่ายออกมาจาก Microcontroller ตามการสั่งการจาก Program Control และได้ทำการเก็บรูปสัญญาณ PWM บางค่ามาเพื่อวิเคราะห์การทำงานของ Program ว่ามีประสิทธิภาพมากน้อยเพียงใด

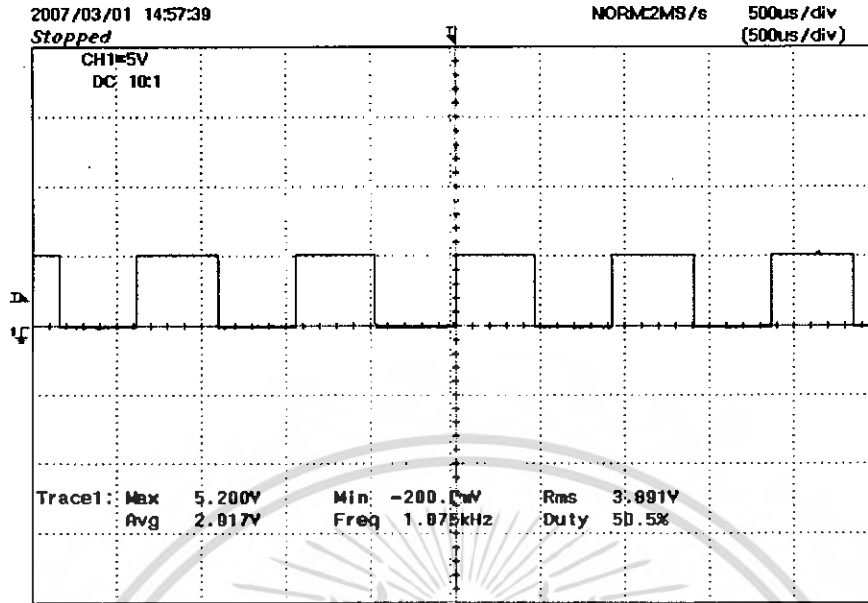
การทำงานที่ 20%



รูปที่ 4.1 ที่การทำงาน 20 % ของ Program Control

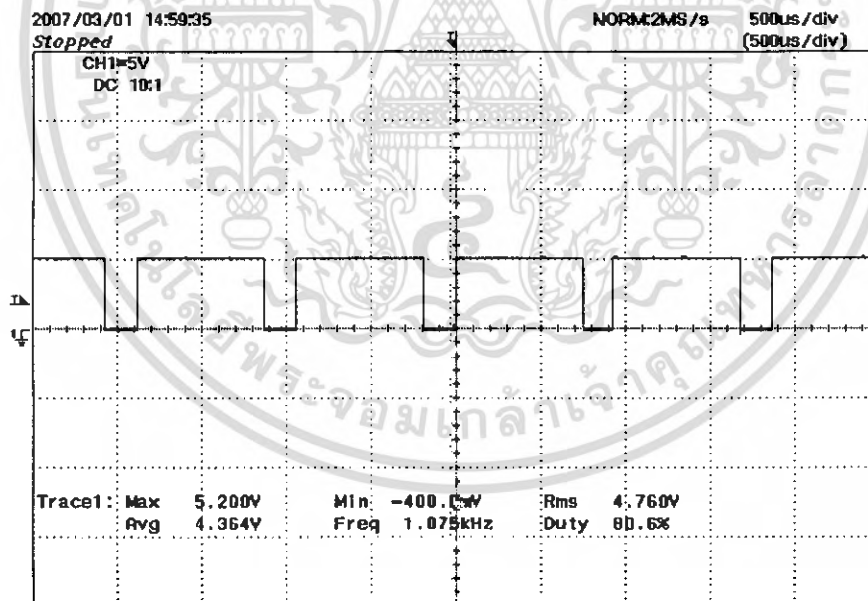
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การทำงานที่ 50 %



รูปที่ 4.2 ที่การทำงาน 50 % ของ Program Control

การทำงานที่ 80 %



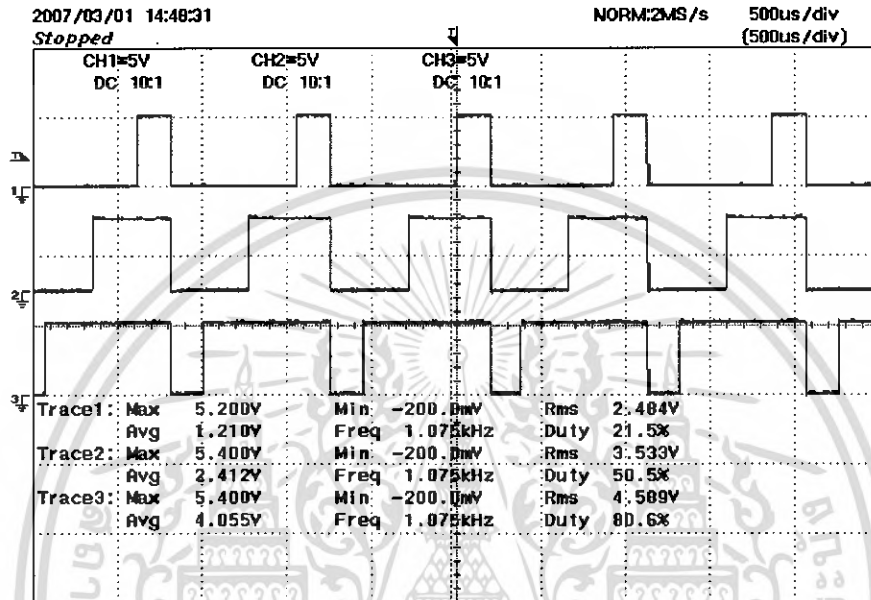
รูปที่ 4.3 ที่การทำงาน 80 % ของ Program Control

จากลักษณะของสัญญาณ PWM ที่ได้ทำการเก็บรูปมาจะเห็นได้ว่าเป็นค่า Error เพียงเล็กน้อยเท่านั้น ซึ่งสามารถสรุปได้ว่าการทำงานของ Program Control ที่ทำงานร่วมกับ Microcontroller สามารถทำงานได้ค่อนข้างสมบูรณ์

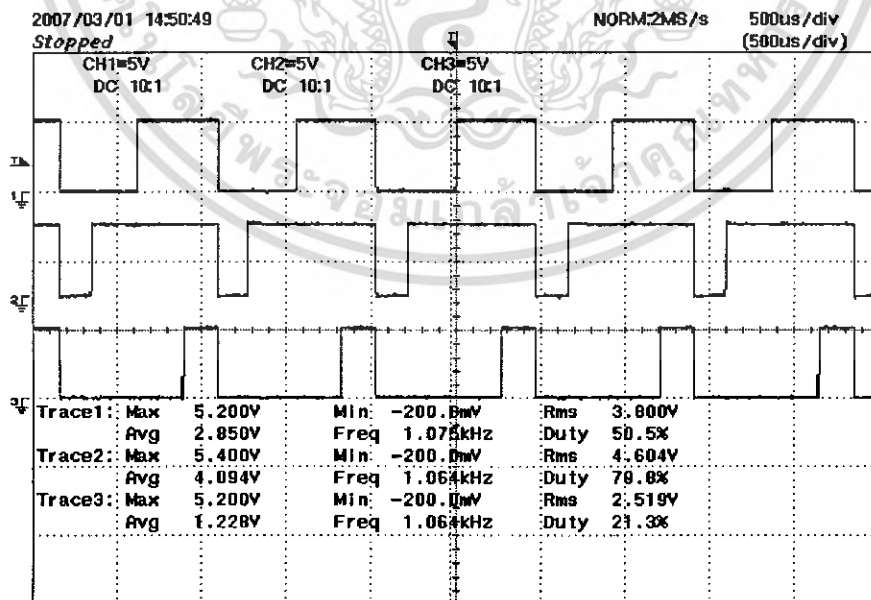
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.3 การทำงานของชุดโคม LED ที่สามารถปรับสีได้

การทำงานของชุดนี้จะประกอบไปด้วยชุดควบคุมการทำงานของหลอด LED 3 ชุดที่สามารถปรับความสว่างของหลอดทั้ง 3 ได้โดยการปรับเปลี่ยนค่าสัญญาณ PWM และหลอดทั้ง 3 สีได้ทำการติดตั้งภายในโคมเดียวกันทำให้สามารถปรับความสว่างของสีต่างๆ (สีแดง สีเขียวและสีน้ำเงิน) ภายในโคมเพื่อผสมสีให้เกิดสีต่างๆได้ตามความต้องการของผู้ใช้งาน

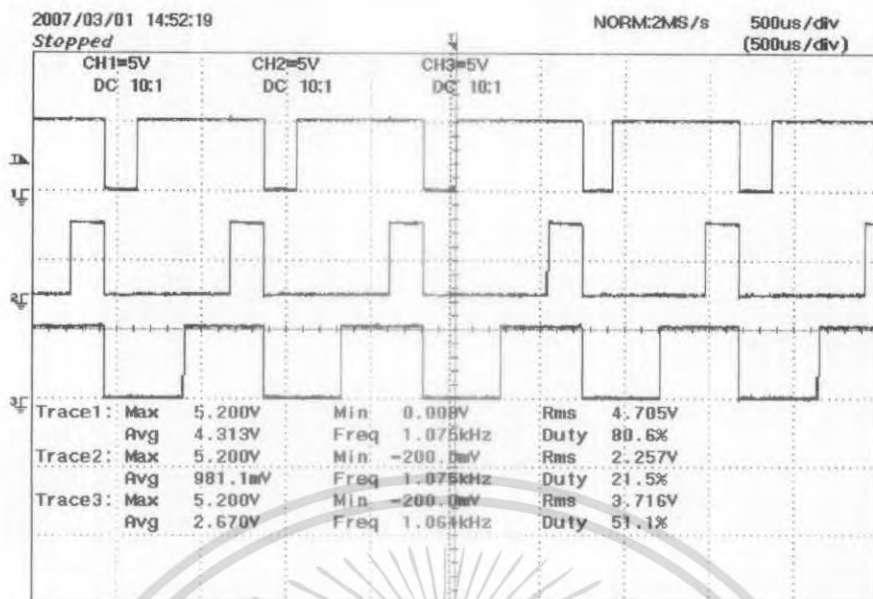


รูปที่ 4.4 การทำงานที่สีแดง 20 % สีเขียว 50 % สีน้ำเงิน 80 %



รูปที่ 4.5 การทำงานที่สีแดง 50 % สีเขียว 80 % สีน้ำเงิน 20 %

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.6 การทำงานที่สีแดง 50 % สีเขียว 80 % สีน้ำเงิน 20 %

จากการทดลองและเก็บตามากีแควะที่ผลการทำงานของ Program Control ที่ทำงานร่วมกับ Microcontroller สามารถเห็นได้จากการทำงานของทั้ง 3 สี จะเกิดค่า Error เพียงเล็กน้อยเท่านั้นจึงทำให้ได้สีออกมาที่ Output ที่โคมไฟใกล้เคียงกับความต้องการของผู้ควบคุมการทำงานที่ Program Control

4.4 การทดลองการทำงานจริง

อุปกรณ์การทดลองทั้งหมด



รูปที่ 4.7 อุปกรณ์การทดลองทั้งหมด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

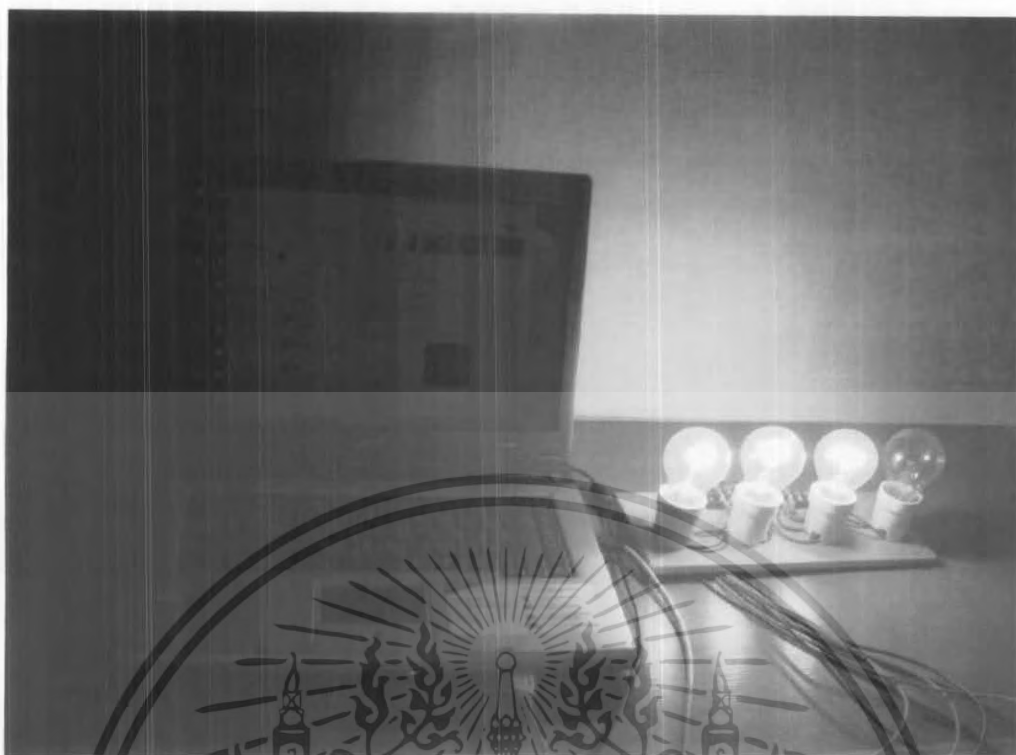


รูปที่ 4.8 การสั่งการทำงานของหลอด 1 หลอด



รูปที่ 4.9 การสั่งการทำงานของหลอด 2 หลอด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

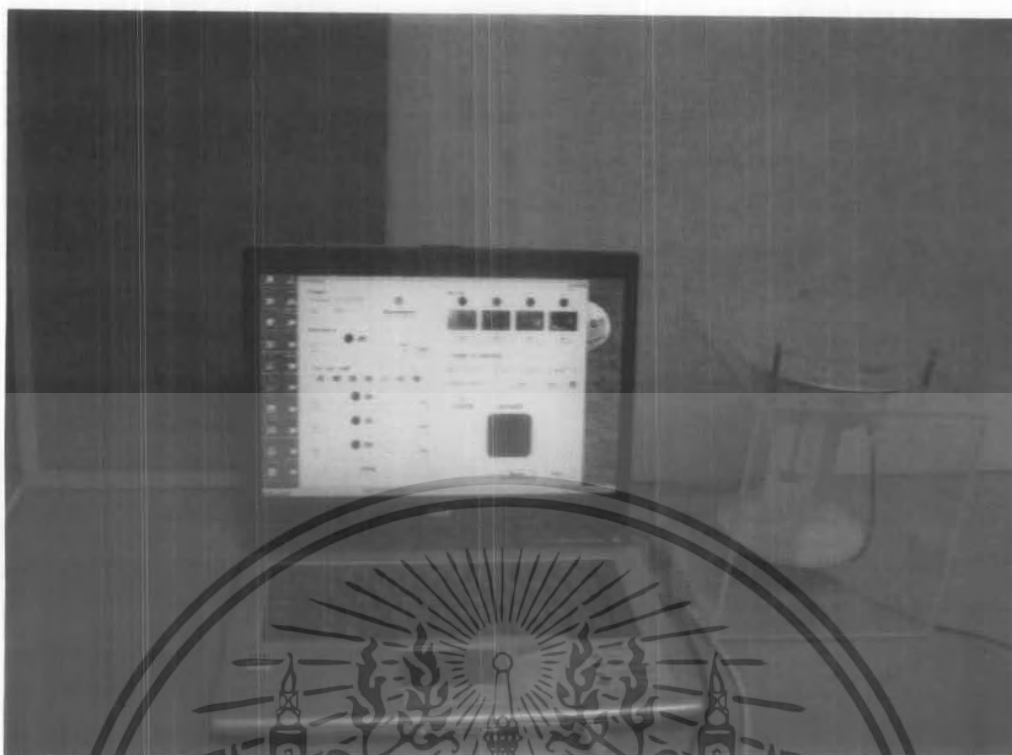


รูปที่ 4.10 การสั่งการทำงานของหลอด 3 หลอด



รูปที่ 4.11 การสั่งการทำงานของหลอด 4 หลอด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

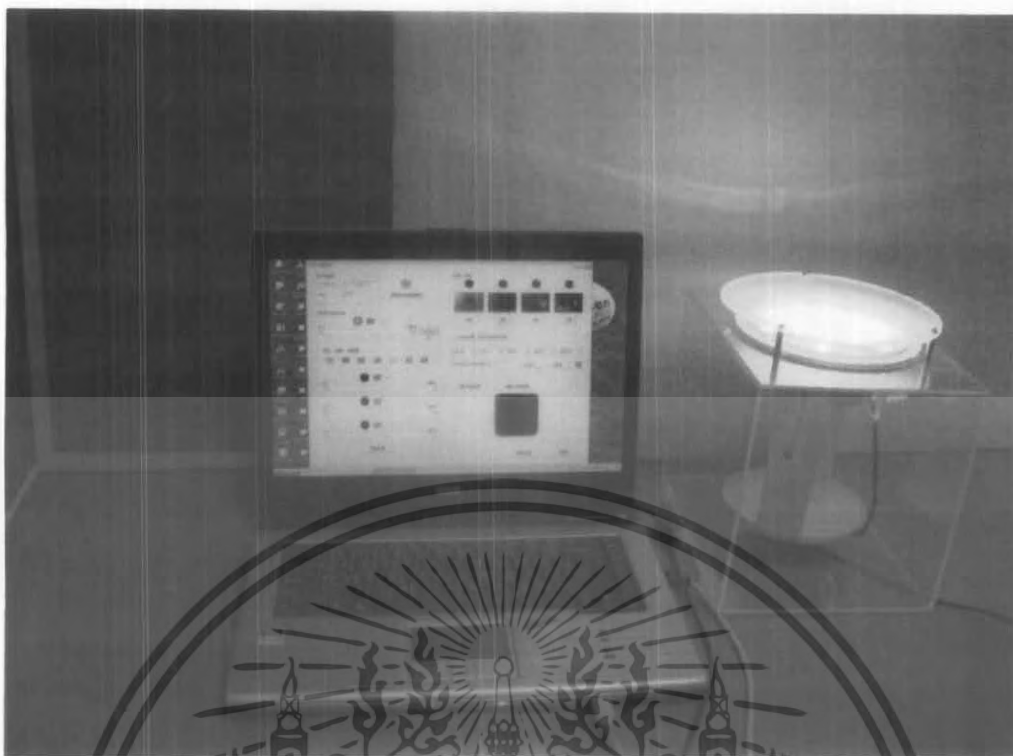


รูปที่ 4.12 การสั่งการทำงานของโคมหลอดขาว 0%



รูปที่ 4.13 การสั่งการทำงานของโคมหลอดขาว 10%

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

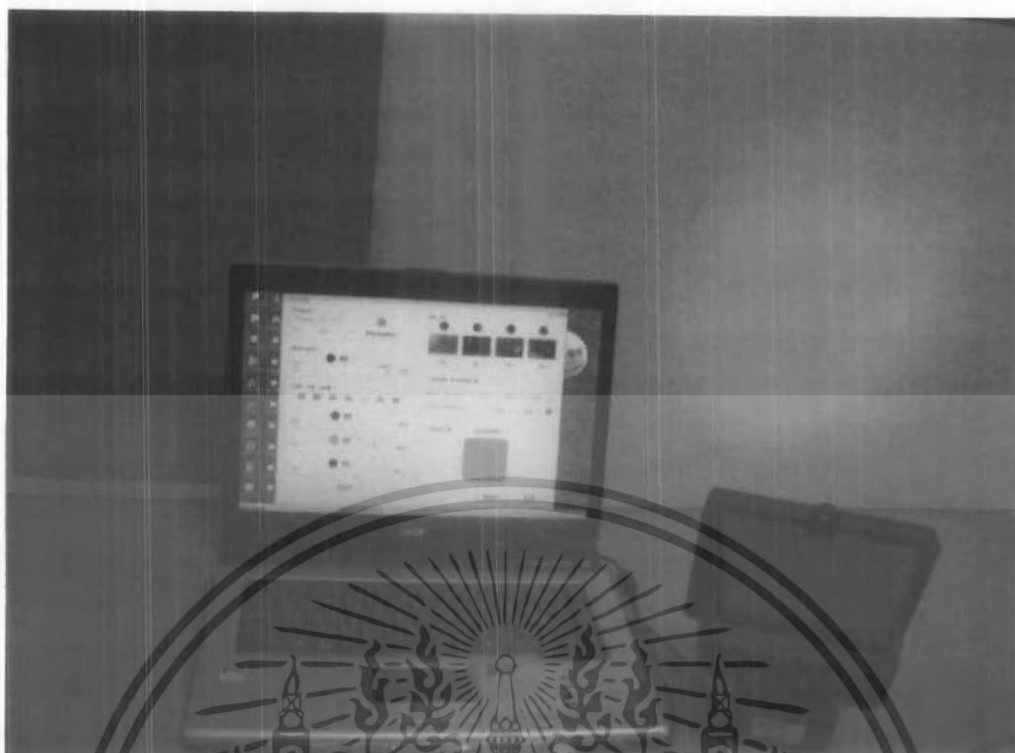


รูปที่ 4.14 การสังการทำานของโคมหลอดขาว 100%



รูปที่ 4.15 การสังการทำานของโคมหลอด 3 สี (สีแดง)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.16 การสั่งการทำงานของคอมหลอด 3 สี (สีเขียว)



รูปที่ 4.17 การสั่งการทำงานของคอมหลอด 3 สี (สีน้ำเงิน)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.18 การสั่งการทำงานของคอมหลอด 3 สี (ผสมสี)



รูปที่ 4.19 การสั่งการทำงานของคอมหลอด 3 สี (ผสมสี)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

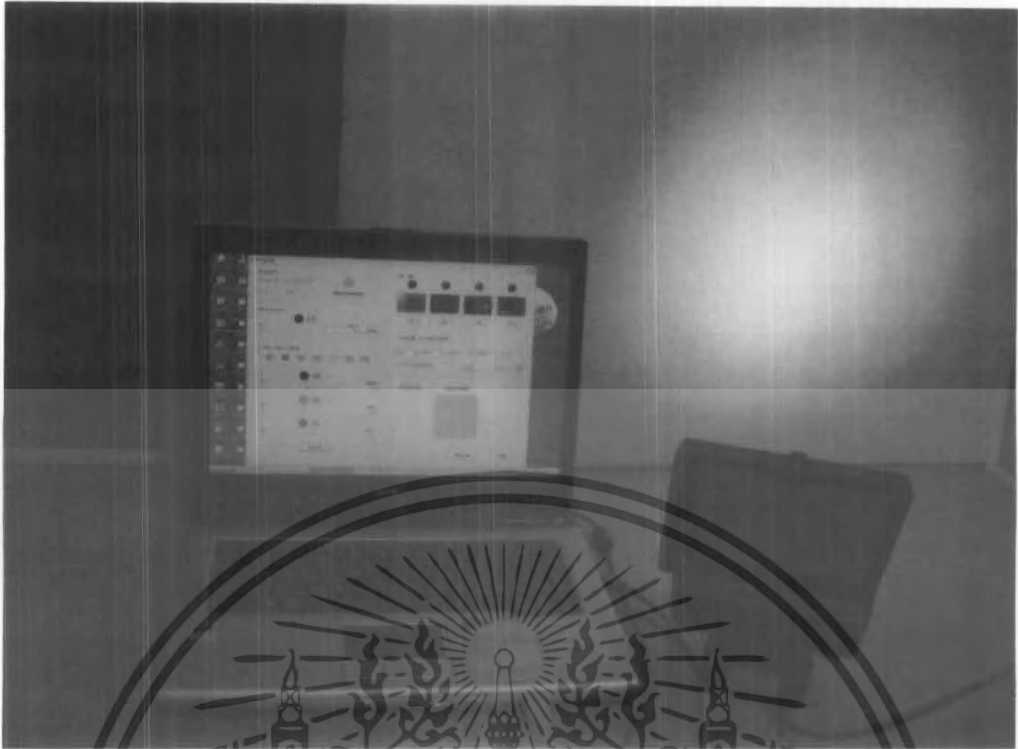


รูปที่ 4.20 การสั่งการทำงานของคอมพิวเตอร์ 3 ดี (ผสมสี)



รูปที่ 4.21 การสั่งการทำงานของคอมพิวเตอร์ 3 ดี (ผสมสี)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.22 การสั่งการทำงานของคอมหลอด 3 สี (ผสมสี)



รูปที่ 4.23 การสั่งการทำงานของคอมหลอด 3 สี (ผสมสี)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 5

สรุปผลและวิจารณ์โครงการ

5.1 สรุปโครงการ

การควบคุมในโครงการที่ทำนี้ จะช่วยในเรื่องของความสะดวกสบายในการสั่งปิด-เปิดการทำงานของอุปกรณ์ไฟฟ้า เพื่อให้มีความรวดเร็วและสะดวกยิ่งขึ้น

จากการทดลองที่ผ่านมาทางกลุ่มสามารถที่จะควบคุมสั่งให้การปิด-เปิด หลอดไฟได้ด้วยการสั่งจากหน้าต่างโปรแกรมวิซวล เบสิก ที่ได้สร้างขึ้น และทำการรับค่าด้วย โปรแกรมภาษาซี ซึ่งสามารถที่จะตอบสนองความต้องการในการสั่งงานได้หลากหลาย โดยสั่งผ่านทางหน้าต่างโปรแกรม วิซวล เบสิก ได้เป็นอย่างดี

ในส่วนของการควบคุมอุปกรณ์ไฟฟ้านี้ โครงการนี้ได้ใช้การควบคุมแบบ UDP (User Datagram Protocol) ซึ่งการทดลองจะพบว่าระบบจะตอบสนองต่อการสั่งการได้รวดเร็วยิ่งขึ้น และเกิดความผิดพลาดน้อยลง ทำให้ระบบมีประสิทธิภาพในการทำงานมากขึ้น

เมื่อมองโครงการโดยรวมจะเห็นว่าการศึกษาและค้นคว้าเนื้อหาของโครงการและการประกอบชิ้นงานและทำการทดลองนั้น ผลที่ได้สามารถที่จะยืนยันการสั่งงาน โดยควบคุมผ่านทางโปรแกรม วิซวล เบสิก ได้เป็นอย่างดี เป็นไปตามวัตถุประสงค์และขอบเขตของโครงการที่วางไว้ทุกประการ

สุดท้าย จากการเริ่มต้นทำโครงการและทดลองในส่วนต่าง ๆ นั้น ทำให้ผู้ทำได้รับรู้และประสบการณ์ในการทำงานใหม่ๆ ได้มองเห็นถึงคุณสมบัติของอุปกรณ์ต่างๆ และความผิดพลาดรวมถึงการแก้ปัญหาในการทำงาน ซึ่งจะนำไปปรับแก้และประยุกต์ใช้ในการทำงานและการดำเนินชีวิตประจำวันต่อไปในอนาคต

5.2 ปัญหา อุปสรรค และการแก้ไข

จากการทำงานที่ผ่านมาสามารถที่จะสรุปปัญหา และอุปสรรค รวมถึงการแก้ไขดังนี้

ในส่วนของการเขียนโปรแกรมนั้น ผู้จัดทำไม่มีความรู้ ความชำนาญเกี่ยวกับการเขียนโปรแกรมมาก่อน จึงเสียเวลาในเรื่องของการเขียนโปรแกรมทั้งด้านวิซวล เบสิก และ ภาษาซี ค่อนข้างมาก ทำให้การดำเนินงานในขั้นตอนนี้ค่อนข้างล่าช้ากว่ากำหนดที่คาดเอาไว้มาก

โดยเฉพาะทางด้านโปรแกรมวิซวล เบสิก นั้น ผู้จัดทำจำเป็นต้องเรียนรู้ในเรื่องที่ค่อนข้างใหม่ ทำให้เสียเวลาในการศึกษาอยู่มากพอสมควร แต่ก็ได้ขอคำแนะนำจากพี่ ๆ ปรินญาโท และอาจารย์ที่ปรึกษา ทำให้เกิดความเข้าใจมากขึ้นในการเขียนโปรแกรม

และที่สำคัญคือ รูปแบบข้อมูลที่ใช้ในการสื่อสารระหว่างโปรแกรม คือ ทางด้านหนึ่งจะเขียนด้วยวิซวล เบสิก และอีกด้านหนึ่งเขียนด้วยภาษาซี ทำให้ต้องศึกษาถึงการแปลงค่าเล็กน้อย ในการส่งข้อมูลไปสั่งการทำงาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ในส่วนต่อมาก็คือ การใช้โปรแกรมช่วยในการออกแบบลาย PCB นั้น ทางผู้จัดทำไม่ค่อยมีความรู้ ความชำนาญเกี่ยวกับการใช้โปรแกรมมากเท่าไร จึงทำให้เกิดการล่าช้า ลองผิดลองถูก เสียเวลาในการทำความเข้าใจ จึงได้ขอคำแนะนำจากเพื่อน ๆ และพี่ปริญญาโทที่ทำอยู่ก็ได้รับคำแนะนำเป็นอย่างดี ทำให้เข้าใจเกี่ยวกับการออกแบบมากขึ้น

อุปสรรคที่สำคัญอีกอย่างหนึ่งคือ การใช้เครื่องมือวัดต่าง ๆ ที่ใช้ประกอบในการทดลอง ซึ่งเครื่องมือบางชนิด ทางกลุ่มยังไม่มี ความชำนาญพอในการใช้งาน จึงต้องศึกษาการใช้งานอย่างละเอียดอยู่นานพอสมควร เพื่อความถูกต้องในการทดลอง และที่สำคัญคือป้องกันความเสียหายที่อาจจะเกิดขึ้นได้กับเครื่องมือวัดนั้น ๆ ซึ่งในส่วนนี้ก็คือว่าเป็นส่วนหนึ่งที่ทำให้การทดลองล่าช้าไปพอสมควรแต่ก็มีประโยชน์ที่ได้รับความรู้จากการได้ใช้เครื่องมือหลากหลายประเภท

ทางด้านปัญหาของการจัดทำปริญญานิพนธ์นั้น ส่วนใหญ่จะมาจากการรวบรวมข้อมูล การคัดเลือกข้อมูลที่ต้องการเหมาะสมกับโครงงาน ซึ่งข้อมูลของอุปกรณ์ มีหลากหลายเนื้อหาที่ต่างกันจึงต้องนำมาประยุกต์ให้ได้ข้อมูลที่ดีและเหมาะสมซึ่งทั้งหมดนี้ก็เป็นปัญหาอีกทางหนึ่งที่ทำให้เกิดความล่าช้าในการทำรูปเล่มปริญญานิพนธ์

จากปัญหาและอุปสรรคที่เกิดขึ้นนั้น มีผลทำให้โครงงานสะดุดหรือล่าช้า ตามที่ได้กล่าวมาแล้วนั้น แนวทางการแก้ไขของกลุ่ม อันดับแรกคือการศึกษาหาข้อมูลเพิ่มเติมในขอบเขตที่พอจะสามารถศึกษาและค้นคว้าด้วยตัวเองได้ แต่ถ้าไม่สามารถแก้ไขได้ ก็จะขอคำปรึกษาจากพี่ ๆ ปริญญาโทก่อน สุดท้ายถ้ายังไม่ได้ก็จะต้องทำการขอคำปรึกษา คำอธิบาย หรือการแก้ไข ปัญหาจากอาจารย์ที่ปรึกษา ซึ่งจากการปรึกษาและขอคำแนะนำ ทำให้ข้อผิดพลาดของโครงงานลดน้อยลง และประสบความสำเร็จในที่สุด

5.3 แนวทางการพัฒนา

สามารถที่จะนำไปใช้ในการควบคุมการทำงานของอุปกรณ์ไฟฟ้าอย่างอื่นได้ในงานอุตสาหกรรม สามารถทำการควบคุม รับ ส่งข้อมูล และนำข้อมูลมาแสดง ได้หลายอย่างภายในโปรแกรมเดียวกัน ในส่วนของการสื่อสารแบบ LAN ที่ใช้สายในการเชื่อมต่อสื่อสาร ก็อาจจะพัฒนาเปลี่ยนมาใช้ในการสื่อสารแบบไร้สายโดยใช้ Wire Less หรือพัฒนาเป็นแบบ Web Application แทน

เอกสารอ้างอิง

- (1). อุดม รานอก, "ภาษา C สำหรับงานควบคุมไมโครคอนโทรลเลอร์", นนทบุรี : โอดีซี, 2548.
- (2). อภิชาติ ภูพลับ, "สนุก! กับการประยุกต์ใช้ Visual Basic", นนทบุรี : อินโฟเพรส, 2546
- (3). อภิชาติ ภูพลับ, "เริ่มต้นการเขียนโปรแกรมติดต่อและควบคุมฮาร์ดแวร์ด้วย Visual Basic", นนทบุรี : อินโฟเพรส, 2546
- (4). ธนะศักดิ์ โสสุวรรณ และคณะ, "การขับเคลื่อนดีซีมอเตอร์ แบบ 4 คอร์ตแดรนท์โดยใช้ วิชาล เบสิก ควบคุมผ่านทาง ไมโครคอนโทรลเลอร์", ภาควิชาวิศวกรรมไฟฟ้า, สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง.
- (5). www.thaiio.com
- (6). www.thaidev.com



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก

โปรแกรม Control (Visual Basic 6.0)

Option Explicit

Dim data As String

Dim data1, data2, data3, data4, x, y As Integer

Dim dataA, dataB, dataC, dataD As String

Dim dataX, dataY, dataZ As Long

Dim data_color1, data_color2, data_color3, data_color4, data_color5 As String

Dim data_time As Integer

Private Sub cmd1_Click() ' ##### ON - OFF 1 #####

Select Case cmd1.Caption

Case "ON"

cmd1.Caption = "OFF"

data = Chr(1)

Winsock1.SendData data

Shape1(0).BackColor = &HFF&

Picture1.Visible = True

Picture2.Visible = False

Case "OFF"

cmd1.Caption = "ON"

data = Chr(2)

Winsock1.SendData data

Shape1(0).BackColor = vbBlack

Picture1.Visible = False

Picture2.Visible = True

End Select

End Sub

Private Sub cmd10_Click() ' ##### ส่งค่า ปรับ - แต่ง - ผสมสี 6,7,8 #####

Winsock1.SendData dataB

Delay1Time 1

Winsock1.SendData dataC

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

Delay1Time 1
Winsock1.SendData dataD
End Sub

Private Sub cmd2_Click() ' ##### ON - OFF 2 #####
Select Case cmd2.Caption
    Case "ON"
        cmd2.Caption = "OFF"
        data = Chr(3)
        Winsock1.SendData data
        Shape1(1).BackColor = &HFF&
        Picture3.Visible = True
        Picture4.Visible = False
    Case "OFF"
        cmd2.Caption = "ON"
        data = Chr(4)
        Winsock1.SendData data
        Shape1(1).BackColor = vbBlack
        Picture3.Visible = False
        Picture4.Visible = True
End Select
End Sub

```

```

Private Sub cmd3_Click() ' ##### ON - OFF 3 #####
Select Case cmd3.Caption
    Case "ON"
        cmd3.Caption = "OFF"
        data = Chr(5)
        Winsock1.SendData data
        Shape1(2).BackColor = &HFF&
        Picture5.Visible = True
        Picture6.Visible = False
    Case "OFF"
        cmd3.Caption = "ON"
        data = Chr(6)

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
Winsock1.SendData data
Shape1(2).BackColor = vbBlack
Picture5.Visible = False
Picture6.Visible = True
End Select
End Sub
```

```
Private Sub cmd4_Click() ' ##### ON - OFF 4 #####
```

```
Select Case cmd4.Caption
```

```
Case "ON"
```

```
cmd4.Caption = "OFF"
data = Chr(7)
Winsock1.SendData data
Shape1(3).BackColor = &HFF&
Picture7.Visible = True
Picture8.Visible = False
```

```
Case "OFF"
```

```
cmd4.Caption = "ON"
data = Chr(8)
Winsock1.SendData data
Shape1(3).BackColor = vbBlack
Picture7.Visible = False
Picture8.Visible = True
```

```
End Select
```

```
End Sub
```

```
Private Sub cmd5_Click() ' ##### ส่งค่า ปรับความสว่าง 5 #####
```

```
Winsock1.SendData dataA
```

```
End Sub
```

```
Private Sub cmdConnect_Click() ' ##### Connect #####
```

```
Select Case cmdConnect.Caption
```

```
Case "Connect"
```

```
Winsock1.RemoteHost = TxtIPAddress.Text
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

Winsock1.RemotePort = TxtPort.Text
Winsock1.Connect
cmdConnect.Caption = "Disconnect"
Shape4.BackColor = vbGreen
Opt1.Enabled = True
Opt2.Enabled = True
Frame1.Enabled = True
Frame2.Enabled = True

Case "Disconnect"
Timer2.Enabled = False
Shape6.BackColor = &HFF&
x = 100000
Winsock1.Close
cmdConnect.Caption = "Connect"
Shape4.BackColor = &HFF&
Opt1.Enabled = False
Opt2.Enabled = False
Frame1.Enabled = False
Frame2.Enabled = False
End Select
End Sub

Private Sub cmdExit_Click() ' ##### Exit Program #####
End
End Sub

Private Sub cmdReset_Click() ' ##### Reset #####
Timer1.Enabled = False
Timer2.Enabled = False
Winsock1.SendData Chr(0)

HScroll1.Value = 10
HScroll2.Value = 70
HScroll3.Value = 130

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

HScroll4.Value = 190

cmd1.Caption = "ON"

cmd2.Caption = "ON"

cmd3.Caption = "ON"

cmd4.Caption = "ON"

Picture2.Visible = True

Picture4.Visible = True

Picture6.Visible = True

Picture8.Visible = True

Shape1(0).BackColor = vbBlack

Shape1(1).BackColor = vbBlack

Shape1(2).BackColor = vbBlack

Shape1(3).BackColor = vbBlack

Shape6.BackColor = &HFF&

Text3.Text = ""

Combo1.Text = "เลือกสี 1"

Combo2.Text = "เลือกสี 2"

Combo3.Text = "เลือกสี 3"

Combo4.Text = "เลือกสี 4"

Combo5.Text = "เลือกสี 5"

End Sub

Private Sub cmdStart_Click() ' ##### Start กำหนดสี - ทำงานอัตโนมัติ #####

On Error GoTo CHECK

Shape6.BackColor = &HFFFF&

data_time = Text3.Text

x = 0

Do Until x >= 100000

data_color1 = Combo1.Text ' ##### Color 1#####

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

If data_color1 = "สีม่วง" Then
    Color1
Elseif data_color1 = "สีคราม" Then
    Color2
Elseif data_color1 = "สีน้ำเงิน" Then
    Color3
Elseif data_color1 = "สีเขียว" Then
    Color4
Elseif data_color1 = "สีเหลือง" Then
    Color5
Elseif data_color1 = "สีส้ม" Then
    Color6
Elseif data_color1 = "สีแดง" Then
    Color7
End If
Send_Data
Delay2Time (data_time)
data_color2 = Combo2.Text '#### Color 2 ####'
If data_color2 = "สีม่วง" Then
    Color1
Elseif data_color2 = "สีคราม" Then
    Color2
Elseif data_color2 = "สีน้ำเงิน" Then
    Color3
Elseif data_color2 = "สีเขียว" Then
    Color4
Elseif data_color2 = "สีเหลือง" Then
    Color5
Elseif data_color2 = "สีส้ม" Then
    Color6
Elseif data_color2 = "สีแดง" Then
    Color7
End If
Send_Data

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

Delay2Time (data_time)
data_color3 = Combo3.Text ' ##### Color 3 #####
If data_color3 = "สีม่วง" Then
    Color1
Elseif data_color3 = "สีคราม" Then
    Color2
Elseif data_color3 = "สีน้ำเงิน" Then
    Color3
Elseif data_color3 = "สีเขียว" Then
    Color4
Elseif data_color3 = "สีเหลือง" Then
    Color5
Elseif data_color3 = "สีส้ม" Then
    Color6
Elseif data_color3 = "สีแดง" Then
    Color7
End If
Send_Data
Delay2Time (data_time)
data_color4 = Combo4.Text ' ##### Color 4 #####
If data_color4 = "สีม่วง" Then
    Color1
Elseif data_color4 = "สีคราม" Then
    Color2
Elseif data_color4 = "สีน้ำเงิน" Then
    Color3
Elseif data_color4 = "สีเขียว" Then
    Color4
Elseif data_color4 = "สีเหลือง" Then
    Color5
Elseif data_color4 = "สีส้ม" Then
    Color6
Elseif data_color4 = "สีแดง" Then
    Color7

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

End If
Send_Data
Delay2Time (data_time)
data_color5 = Combo5.Text ' ##### Color 5 #####
If data_color5 = "สีม่วง" Then
    Color1
Elseif data_color5 = "สีคราม" Then
    Color2
Elseif data_color5 = "สีน้ำเงิน" Then
    Color3
Elseif data_color5 = "สีเขียว" Then
    Color4
Elseif data_color5 = "สีเหลือง" Then
    Color5
Elseif data_color5 = "สีส้ม" Then
    Color6
Elseif data_color5 = "สีแดง" Then
    Color7
End If
Send_Data
Delay2Time (data_time)
x = x + 1
Loop
CHECK:
If Text3.Text = "" Then
    MsgBox "กรุณากำหนดเวลาที่ใช้ในการเปลี่ยนสีด้วย", vbExclamation + vbOKOnly,
"Error"
End If
End Sub

Private Sub cmdStop_Click() ' ##### Stop กำหนดสี - ทำงานอัตโนมัติ #####
Timer2.Enabled = False
Shape6.BackColor = &HFF&
x = 100000

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

End Sub

Private Sub Form_Load() '#### กำหนดค่าต่าง ๆ ของโปรแกรม ####

Timer3.Interval = 100

Opt1.Enabled = False

Opt2.Enabled = False

Frame1.Enabled = False

Frame2.Enabled = False

Frame3.Enabled = False

Frame4.Enabled = False

HScroll1.Min = 10

HScroll1.Max = 60

HScroll1.Value = 10

HScroll2.Min = 70

HScroll2.Max = 120

HScroll2.Value = 70

HScroll3.Min = 130

HScroll3.Max = 180

HScroll3.Value = 130

HScroll4.Min = 190

HScroll4.Max = 240

HScroll4.Value = 190

Shape1(0).BackColor = vbBlack

Shape1(1).BackColor = vbBlack

Shape1(2).BackColor = vbBlack

Shape1(3).BackColor = vbBlack

Shape4.BackColor = &HFF&

Shape6.BackColor = &HFF&

End Sub

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
Private Sub HScroll1_Change() ' ##### HScroll1 #####
data1 = HScroll1.Value
dataA = Chr(data1)
lb1(0).Caption = "= " & (data1 - 10) * 2
Shape2.BackColor = (data1 - 10) * 5.1
End Sub
```

```
Private Sub HScroll2_Change() ' ##### HScroll2 #####
data2 = HScroll2.Value
dataB = Chr(data2)
dataX = (data2 - 70) * 5.1
lb1(1).Caption = "= " & (data2 - 70) * 2
Shape3(0).BackColor = RGB(dataX, 0, 0)
Updatecolor
End Sub
```

```
Private Sub HScroll3_Change() ' ##### HScroll3 #####
data3 = HScroll3.Value
dataC = Chr(data3)
dataY = (data3 - 130) * 5.1
lb1(2).Caption = "= " & (data3 - 130) * 2
Shape3(1).BackColor = RGB(0, dataY, 0)
Updatecolor
End Sub
```

```
Private Sub HScroll4_Change() ' ##### HScroll4 #####
data4 = HScroll4.Value
dataD = Chr(data4)
dataZ = (data4 - 190) * 5.1
lb1(3).Caption = "= " & (data4 - 190) * 2
Shape3(2).BackColor = RGB(0, 0, dataZ)
Updatecolor
End Sub
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
Sub Updatecolor() ' ##### MIX Color #####  
Shape5.BackColor = RGB(dataX, dataY, dataZ)  
End Sub
```

```
Private Sub Delay1Time(Sec1 As Long) ' ##### Delay1Time #####  
Timer1.Enabled = True  
Timer1.Interval = 10 * Sec1  
While Timer1.Interval > 0  
    DoEvents  
Wend  
Timer1.Enabled = False  
End Sub
```

```
Private Sub Delay2Time(Sec2 As Long) ' ##### Delay2Time #####  
Timer2.Enabled = True  
Timer2.Interval = 1000 * Sec2  
While Timer2.Interval > 0  
    DoEvents  
Wend  
Timer2.Enabled = False  
End Sub
```

```
Private Sub Opt1_Click() ' ##### เลือกการทำงานเป็นแบบ ปรับ - แต่ง - ผสมสี #####  
Timer2.Enabled = False  
Shape6.BackColor = &HFF&  
x = 100000
```

```
Frame3.Enabled = True  
Frame4.Enabled = False  
End Sub
```

```
Private Sub Opt2_Click() ' ##### เลือกการทำงานเป็นแบบ กำหนดสี - ทำงานอัตโนมัติ #####  
Frame3.Enabled = False  
Frame4.Enabled = True
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

End Sub

Private Sub Opt3_Click()

Color1

Send_Data

End Sub

Private Sub Opt4_Click()

Color2

Send_Data

End Sub

Private Sub Opt5_Click()

Color3

Send_Data

End Sub

Private Sub Opt6_Click()

Color4

Send_Data

End Sub

Private Sub Opt7_Click()

Color5

Send_Data

End Sub

Private Sub Opt8_Click()

Color6

Send_Data

End Sub

Private Sub Opt9_Click()

Color7



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
Send_Data
End Sub
Private Sub Timer1_Timer() '#### Timer 1 ####
Timer1.Enabled = False
Timer1.Interval = 0
End Sub
```

```
Private Sub Timer2_Timer() '#### Timer 2 ####
Timer2.Enabled = False
Timer2.Interval = 0
End Sub
```

```
Private Sub Timer3_Timer() '#### แสดงเวลา ####
Label8.Caption = Format(Time, "Hh:Mm:Ss")
End Sub
```

```
Private Sub Color1() '#### สีม่วง ####
HScroll2.Value = 89.8
HScroll3.Value = 134.7
HScroll4.Value = 213.53
End Sub
```

```
Private Sub Color2() '#### สีคราม ####
HScroll2.Value = 78.04
HScroll3.Value = 139.02
HScroll4.Value = 211.96
End Sub
```

```
Private Sub Color3() '#### สีน้ำเงิน ####
HScroll2.Value = 70
HScroll3.Value = 130
HScroll4.Value = 240
End Sub
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
Private Sub Color4() '#### สีเขียว ####
HScroll2.Value = 70
HScroll3.Value = 180
HScroll4.Value = 190
End Sub
```

```
Private Sub Color5() '#### สีเหลือง ####
HScroll2.Value = 120
HScroll3.Value = 180
HScroll4.Value = 190
End Sub
```

```
Private Sub Color6() '#### สีส้ม ####
HScroll2.Value = 117.65
HScroll3.Value = 152.74
HScroll4.Value = 196.27
End Sub
```

```
Private Sub Color7() '#### สีแดง ####
HScroll2.Value = 120
HScroll3.Value = 130
HScroll4.Value = 190
End Sub
```

```
Private Sub Send_Data()
Winsock1.SendData dataB
Delay1Time 1
Winsock1.SendData dataC
Delay1Time 1
Winsock1.SendData dataD
End Sub
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โปรแกรมของชุด Microcontroller

```
#include<reg51.h>    // Header include register of P89C51RD2
#define ON            1
#define OFF          0

sbit LAMP1 = P0^0;
sbit LAMP2 = P0^1;
sbit LAMP3 = P0^2;
sbit LAMP4 = P0^3;
sbit LAMP5 = P1^3;
sbit LAMP6 = P1^4;
sbit LAMP7 = P1^5;
sbit LAMP8 = P1^6;

////////// Add register ///////////
sfr CMOD    = 0xD9;
sfr CCAPM0  = 0xDA;
sfr CCAPM1  = 0xDB;
sfr CCAPM2  = 0xDC;
sfr CCAPM3  = 0xDD;

sfr CCAP0H  = 0xFA;
sfr CCAP1H  = 0xFB;
sfr CCAP2H  = 0xFC;
sfr CCAP3H  = 0xFD;

sfr CCON    = 0xD8;

/***** Sevice Interrupt Serial Port *****/

void sevice_serial() interrupt 4    // Vector interrupt for serial port
{
    if (RI)                          // IF interrupt by reciver data work in block
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

    {
        RI = 0;           // Clear RI flag
        P2 = SBUF;
    }
}

/***** Main loop *****/
void main(void)
{
    TMOD = 0x22;        // Timer1 Mode2(8 bit auto reload) for serial port
    SCON = 0x50;        // Mode serial port TX/RX data
    TH1 = 0xF8;         // Set 9600 bps Timer1 default
    TL1 = 0xF8;
    RI = 0;             // Clear RI flag
    TI = 0;             // Clear TI flag
    EA = 1;             // Enable interrupt all
    ES = 1;             // Enable interrupt serial port
    TR1 = 1;           // Start Timer1
    TH0 = 0xF7;
    TL0 = 0xF7;
    TR0 = 1;

    CMOD = 0x04;
    CCAPM0 = 0x42;
    CCAPM1 = 0x42;
    CCAPM2 = 0x42;
    CCAPM3 = 0x42;

    CCON = 0x40;

    CCAP0H = 0; // p1.3
    CCAP1H = 0; // p1.4
    CCAP2H = 0; // p1.5
    CCAP3H = 0; // p1.6
}

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

P2 = 0x00;          // Clear data port0

While (1)          // Infinite loop
{
If (P2==0x01)
    {
        LAMP1=ON;
    }

If (P2==0x02)
    {
        LAMP1=OFF;
    }
If (P2==0x03)
    {
        LAMP2=ON;
    }
If (P2==0x04)
    {
        LAMP2=OFF;
    }
If (P2==0x05)
    {
        LAMP3=ON;
    }
If (P2==0x06)
    {
        LAMP3=OFF;
    }
If (P2==0x07)
    {
        LAMP4=ON;
    }
If (P2==0x08)

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

    {
        LAMP4=OFF;
    }
if (P2>=10&&P2<=60)
    {
        CCAP0H =~(SBUF-10)*5;
        if(P2==10)
            {
                LAMP5 = P1^3;
                LAMP5=0;
            }
    }
if (P2>70&&P2<=120)
    {
        CCAP1H =~(SBUF-70)*5;
    }
if (P2==70)
    {
        CCAP1H =~(SBUF*0);
    }
if (P2>130&&P2<=180)
    {
        CCAP2H =~(SBUF-130)*5;
    }
if (P2==130)
    {
        CCAP2H =~(SBUF*0);
    }
if (P2>190&&P2<=240)
    {
        CCAP3H =~(SBUF-190)*5;
    }
if (P2==190)

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

    {
    CCAP3H=~(SBUF*0);
    }
If (P2==0x00)
    {
    LAMP1=OFF;
    LAMP2=OFF;
    LAMP3=OFF;
    LAMP4=OFF;
    CCAP0H=~(SBUF*0);
    CCAP1H=~(SBUF*0);
    CCAP2H=~(SBUF*0);
    CCAP3H=~(SBUF*0);
    }
}
}

```



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้