



ระบบตรวจสอบและเตือนภัยผู้บุกรุกโดยผ่านเครือข่ายอินเทอร์เน็ต  
INTRUDER CHECKING AND WARNING SYSTEM BY INTERNET



ปริญญาานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต  
สาขาวิชาวิศวกรรมโทรคมนาคม  
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง  
ปีการศึกษา 2546

เลขหมู่.....  
เลขทะเบียน 54972.....  
วัน,เดือน,ปี - 4 เม.ย. 2548

b.....  
1.....

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
หากมีการละเมิดลิขสิทธิ์จะมีความผิดตามกฎหมายมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ระบบตรวจสอบและเตือนภัยผู้บุกรุกโดยผ่านเครือข่ายอินเทอร์เน็ต  
INTRUDER CHECKING AND WARNING SYSTEM BY INTERNET



ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต  
สาขาวิชาวิศวกรรมโทรคมนาคม  
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง  
ปีการศึกษา 2546

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ปริญญาโทปีการศึกษา 2546

ภาควิชาวิศวกรรมโทรคมนาคม

คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เรื่อง ระบบตรวจสอบและเตือนภัยผู้บุกรุกโดยผ่านเครือข่ายอินเทอร์เน็ต

**INTRUDER CHECKING AND WARNING SYSTEM BY INTERNET**

ผู้จัดทำ

- |                |         |          |
|----------------|---------|----------|
| 1. นายธีระวุฒิ | โพธิชัย | 44015060 |
| 2. นายสาธิต    | เสนคง   | 44015087 |



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ระบบตรวจสอบและเตือนภัยผู้บุกรุกโดยผ่านเครือข่ายอินเทอร์เน็ต  
INTRUDER CHECKING AND WARNING SYSTEM BY INTERNET

โดย นายธีระวุฒิ โพธิ์ชัย 44015060

นายสายันต์ แสนคง 44015087

อาจารย์ที่ปรึกษา อ.สุรพล บุญจันทร์

**บทคัดย่อ**

โครงการนี้จะขอนำเสนอถึงอุปกรณ์ที่ใช้ตรวจสอบและเตือนภัยในกรณีที่มีผู้บุกรุกเข้ามาโดยผ่านเครือข่ายอินเทอร์เน็ต โดยมีขีดความสามารถดังนี้ สามารถดูภาพและควบคุมการเคลื่อนที่ของกล้องได้โดยผ่านเครือข่ายอินเทอร์เน็ตและยังสามารถโทรศัพท์แจ้งเตือนทางโทรศัพท์เมื่อมีผู้บุกรุกเข้ามาโดยทันที

**ABSTRACT**

This project will present the equipment which can check and warn the user by internet whenever a stealer coming. By this equipment, we can see a picture and control camera by internet too. Moreover, it can call automatic the user by telephone or mobile phone when a robber coming.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## กิตติกรรมประกาศ

ปริญญาานิพนธ์ฉบับนี้ประสบความสำเร็จลุล่วงไปได้ด้วยดี เนื่องจากความร่วมมือของสมาชิกภายในกลุ่มทุกท่าน คณะผู้จัดทำขอกราบขอบพระคุณอาจารย์สุรพล บุญจันทร์ อาจารย์ที่ปรึกษาปริญญาานิพนธ์ ภาควิชาวิศวกรรมศาสตร์ รวมไปถึงคณาจารย์ประจำภาควิชาทุกท่านที่กรุณาให้คำแนะนำ แนวความคิด ความรู้และแนวทางแก้ไขปัญหาในการจัดทำปริญญาานิพนธ์ นอกจากนี้ คณะผู้จัดทำขอขอบคุณเพื่อนๆ ทุกคนที่ให้ความช่วยเหลือในด้าน ให้คำแนะนำปรึกษา เครื่องมือ และวัสดุอุปกรณ์เป็นอย่างดี สุดท้ายที่ควรระลึกถึงเป็นอย่างยิ่งคือ บิดา มารดา และญาติพี่น้องผู้ให้กำลังใจและสนับสนุนทุนการศึกษาด้วยดีตลอดมาตั้งแต่อดีตจนถึงปัจจุบัน



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## สารบัญ

เรื่อง	หน้า
บทคัดย่อภาษาไทย-อังกฤษ	II
กิตติกรรมประกาศ	III
สารบัญ	IV
สารบัญตาราง	IX
สารบัญรูป	X
บทที่ 1 บทนำ	1
1.1 ความเป็นมาและความสำคัญของปริยญาวิพนธ์	1
1.2 ชี้ดความสามารถของโครงการ	1
1.3 เนื้อหาโดยสังเขป	1
บทที่ 2 ทฤษฎีและหลักการ	3
2.1 พื้นฐานการสื่อสารผ่านทางพอร์ตสื่อสารแบบอนุกรมโดยวิซวลเบสิก	3
2.1.1 การรับและส่งข้อมูลผ่านพอร์ตอนุกรมโดยคอนโทรล MSComm	5
2.1.2 การใช้งานคอนโทรล MSComm	6
2.2 ความรู้เบื้องต้นเกี่ยวกับพอร์ตอนุกรม	6
2.2.1 การสื่อสารข้อมูลแบบซิงโครนัส	7
2.2.2 การสื่อสารข้อมูลแบบอะซิงโครนัส	7
2.2.3 มาตรฐานพอร์ตอนุกรมแบบ RS-232	9
2.2.4 คอนเน็คเตอร์สำหรับพอร์ต RS-232 และการเชื่อมต่อ	9
2.2.5 มาตรฐานการรับส่งข้อมูลแบบอะซิงโครนัส (UART)	13
2.3 การเชื่อมต่อพอร์ตอนุกรมกับระบบบัส I <sup>2</sup> C	14
2.3.1 คุณสมบัติโดยทั่วไปของบัส I <sup>2</sup> C	14
2.3.2 หลักการของบัส I <sup>2</sup> C	16
2.3.3 วงจรแปลงสัญญาณพอร์ตอนุกรม เพื่อเชื่อมต่อกับระบบบัส I <sup>2</sup> C	16
2.3.4 การเขียนโปรแกรมติดต่อบัส I <sup>2</sup> C ด้วยวิซวลเบสิก	17
2.4 หลักในการพัฒนาแอปพลิเคชันด้วย ASP	18
2.4.1 หลักการทำงานของ ASP	18

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## สารบัญ (ต่อ)

เรื่อง	หน้า
2.4.2 หลักในการเขียน ASP	18
2.4.3 การใช้โปรแกรมพัฒนาและเขียน ASP	19
2.4.4 ความสามารถของ ASP	19
2.5 มัลติมีเดียบนอินเทอร์เน็ต	22
2.5.1 การบีบอัดสัญญาณภาพ	23
2.5.2 พื้นฐานการบีบอัดสัญญาณภาพ	24
2.5.3 มาตรฐานวิดีโอ	25
2.5.4 Windows Media Player MPEG-4	27
2.6 ActiveX	28
2.6.1 ActiveX Control	29
2.6.2 ActiveX Scripting	29
2.6.3 ActiveX Document	30
2.6.4 ActiveX Security	30
2.6.5 ActiveX และ Java	30
2.7 สวิตซ์ลำแสง	30
2.7.1 ชนิดของตัวรับแสงและตัวกำเนิดแสงในสวิตซ์ลำแสง	31
2.7.2 เทคนิคในการรับส่งลำแสง	31
2.7.3 หลักการเลือกใช้สวิตซ์ลำแสงให้เหมาะสมกับงาน	32
2.8 สวิตซ์แม่เหล็กไฟฟ้าหรือรีเลย์	32
2.9 ความรู้เบื้องต้นเกี่ยวกับสแต็ปปีงมอเตอร์	34
บทที่ 3 การออกแบบ การสร้าง และการทำงาน	38
3.1 กล่าวนำ	38
3.2 การออกแบบด้านฮาร์ดแวร์	38
3.2.1 แผงวงจรควบคุมการทำงานหลัก	39
3.2.2 วงจรขยายพอร์ตอินพุตเอาต์พุต	42
3.2.3 วงจรขับสแต็ปปีงมอเตอร์	43
3.2.4 วงจรแหล่งจ่ายไฟ	44

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## สารบัญ (ต่อ)

เรื่อง	หน้า
3.2.5 การออกแบบฐานควบคุมการหมุนของกลิ้ง	45
3.3 การออกแบบด้านซอฟต์แวร์	47
3.3.1 การออกแบบโปรแกรมควบคุมสตีปีจิ่งมอเตอร์	47
3.3.2 การออกแบบโปรแกรมตรวจสอบสัญญาณรบกวน	53
3.3.3 การออกแบบโปรแกรมโทรศัพท์ที่แจ้งเตือนภัย	53
3.3.4 การออกแบบการควบคุมกลิ้งผ่านอินเทอร์เน็ต	54
3.3.5 การออกแบบการส่งสัญญาณภาพผ่านเครือข่ายอินเทอร์เน็ต	55
บทที่ 4 การทดลอง และผลการทดลอง	56
4.1 กล่าวนำ	56
4.2 การทดลองแผงวงจรควบคุมการทำงานหลัก	56
4.2.1 ลำดับขั้นการทดลอง	56
4.2.2 ผลการทดลอง	57
4.3 การทดลองวงจรขยายพอร์ตอินพุตเอาต์พุต	57
4.3.1 ลำดับขั้นการทดลอง	58
4.3.2 ผลการทดลอง	59
4.4 การทดลองวงจรขับเคลื่อนสตีปีจิ่งมอเตอร์	60
4.4.1 ลำดับขั้นการทดลอง	60
4.4.2 ผลการทดลอง	61
4.5 การทดลองการส่งสัญญาณภาพ	61
4.6 การทดลองการใช้งานเว็บเพจติดต่อกับฐานข้อมูลและควบคุม สตีปีจิ่งมอเตอร์ผ่านเครือข่าย	66
4.6.1 ลำดับขั้นการทดลอง	66
4.6.2 ผลการทดลอง	67
4.7 การทดลองการเขียนฐานข้อมูลโดยใช้ โปรแกรม Microsoft Access	70
4.7.1 ลำดับขั้นการทดลอง	70
4.7.2 ผลการทดลอง	71

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## สารบัญ (ต่อ)

เรื่อง	หน้า
4.8 การทดลองการใช้งาน โปรแกรมระบบเตือนภัยผู้บุกรุกสำนักงาน ผ่านเครือข่ายอินเทอร์เน็ต	72
4.9 การทดลอง โปรแกรมควบคุมการหมุนของสเต็ปปีงมอเตอร์	73
4.9.1 ลำดับขั้นการทดลอง	73
4.9.2 ผลการทดลอง	74
4.10 การทดลอง โปรแกรมควบคุมการหมุนของสเต็ปปีงมอเตอร์ โดยใช้ ค่าองศาจากฐานข้อมูล	74
4.10.1 ลำดับขั้นการทดลอง	75
4.10.2 ผลการทดลอง	76
4.11 การทดลอง การตรวจสอบค่าองศาการหมุนของสเต็ปปีงมอเตอร์	77
4.11.1 ลำดับขั้นการทดลอง	77
4.11.2 ผลการทดลอง	77
4.12 การทดลอง โปรแกรมตรวจสอบการทำงานของอุปกรณ์เซนเซอร์	78
4.12.1 ลำดับขั้นการทดลอง	78
4.12.2 ผลการทดลอง	79
4.13 การทดลอง โปรแกรมสั่งงาน ไมโคร	79
4.13.1 ลำดับขั้นการทดลอง	79
4.13.2 ผลการทดลอง	80
บทที่ 5 บทสรุป ปัญหา และแนวทางแก้ไขและพัฒนา	81
5.1 บทสรุป	81
5.2 ปัญหาและแนวทางการแก้ไข	81
5.3 แนวทางการพัฒนา โครงการงาน	83
บรรณานุกรม	
ภาคผนวก	

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## สารบัญตาราง

ตาราง	หน้า
ตารางที่ 2.1 ค่าพื้นฐานในการคิดต่อพอร์ตคอนูกรมของคอน โทรล MSComm	4
ตารางที่ 2.2 ค่าพาริตีในการตั้งค่าพรอฟเพอร์ตี	4
ตารางที่ 2.3 การจัดขาสัญญาณของพอร์ตคอนูกรมในแบบต่างๆ และหน้าที่การทำงาน	10
ตารางที่ 2.4 แอปพลิเคชันมัลติมีเดียและเบนควิคว์ที่ต้องการ	22
ตารางที่ 2.5 แอปพลิเคชันทางด้านมัลติมีเดียและเบนควิคว์ที่ต้องการ	23
ตารางที่ 2.6 รูปแบบการขับสแต็ปิ่งมอเตอร์แบบฟูลสแต็ป 1 เฟส	36
ตารางที่ 2.7 รูปแบบการขับสแต็ปิ่งมอเตอร์แบบฟูลสแต็ป 2 เฟส	37
ตารางที่ 2.8 รูปแบบการขับสแต็ปิ่งมอเตอร์แบบฮาล์ฟสแต็ป	37
ตารางที่ 4.1 การทดลองวงจรขยายอินพุตเอาต์พุต	59



## สารบัญรูป

รูป	หน้า
รูปที่ 2.1 แผนผังการทำงานเวลาของการสื่อสารข้อมูลแบบซิงโครนัส	7
รูปที่ 2.2 รูปแบบของข้อมูลแบบอะซิงโครนัส	8
รูปที่ 2.3 คอนเน็กเตอร์อนุกรม	10
รูปที่ 2.4 การต่ออุปกรณ์ภายนอกเข้ากับคอมพิวเตอร์ผ่านทางพอร์ตอนุกรมในรูปแบบต่างๆ	12
รูปที่ 2.5 แผนผังการทำงานภายในของขาสัญญาณต่างๆ ของพอร์ตอนุกรม	13
รูปที่ 2.6 โครงสร้างวงจรเอาต์พุตของอุปกรณ์ที่ใช้การเชื่อมต่อบนระบบบัส I <sup>2</sup> C	14
รูปที่ 2.7 การเชื่อมต่ออุปกรณ์บนระบบบัส I <sup>2</sup> C ที่ใช้ไฟเลี้ยงไม่เท่ากัน	15
รูปที่ 2.8 การต่อตัวต้านทานเพื่อป้องกันแรงดันกระชากที่อาจปะปนเข้ามาในไฟเลี้ยงของ อุปกรณ์ในระบบบัส I <sup>2</sup> C	15
รูปที่ 2.9 วงจรสร้างสายสัญญาณเพื่อเชื่อมต่อกับระบบบัส I <sup>2</sup> C ของพอร์ตอนุกรม	17
รูปที่ 2.10 แผนผังการทำงานของเว็บเซิร์ฟเวอร์ กับไฟล์ชนิด HTML	20
รูปที่ 2.11 แผนผังการทำงานของเว็บเซิร์ฟเวอร์ กับไฟล์ชนิด ASP	21
รูปที่ 2.12 การเชื่อมต่อสัญญาณวิดีโอจากกล้องถ่ายภาพกับการ์ดวีดีโอ	28
รูปที่ 2.13 รีเลย์ (Relays)	33
รูปที่ 2.14 แมกเนติกคอนแทกเตอร์ (Magnetic Contactors)	34
รูปที่ 2.15 หลิวรีเลย์ (Reed Relay)	34
รูปที่ 2.16 โครงสร้างพื้นฐานของสเต็ปมิ่งมอเตอร์แบบยูนิโพลาร์ทั้งแบบ 5 และ 6 สาย	35
รูปที่ 3.1 แผนผังการทำงานของระบบเตือนภัยผู้บุกรุกสำนักงานผ่านเครือข่ายอินเตอร์เน็ต	38
รูปที่ 3.2 แผงวงจรควบคุมการทำงานหลัก	39
รูปที่ 3.3 วงจรแปลงสัญญาณมาตรฐาน RS-232 เป็นมาตรฐานทีทีแอล	40
รูปที่ 3.4 วงจรบัฟเฟอร์และส่วนแสดงผล	41
รูปที่ 3.5 วงจรแปลงระบบบัส I <sup>2</sup> C	42
รูปที่ 3.6 วงจรขยายพอร์ตอินพุตเอาต์พุต 16 บิต	43
รูปที่ 3.7 วงจรขับสเต็ปมิ่งมอเตอร์	44
รูปที่ 3.8 วงจรแหล่งจ่ายไฟ	45
รูปที่ 3.9 โครงสร้างส่วนยึดติดกล่องกับสเต็ปมิ่งมอเตอร์	45
รูปที่ 3.10 โครงสร้างของชุดควบคุมการหมุนของกล่อง	46

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## สารบัญรูป (ต่อ)

รูป	หน้า
รูปที่ 3.11 โครงสร้างส่วนยึดติดชุดคล้องกับเพดาน	46
รูปที่ 3.12 โครงสร้างของชุดควบคุมคล้องที่สามารถยึดติดกับเพดาน	47
รูปที่ 3.13 ผังงานของโปรแกรมควบคุมสแต็ปปีงมอเตอร์จากเครื่องเซิร์ฟเวอร์	48
รูปที่ 3.14 ผังงานของโปรแกรมย่อยควบคุมสแต็ปปีงมอเตอร์หมุนซ้าย	49
รูปที่ 3.15 ผังงานของโปรแกรมย่อยควบคุมสแต็ปปีงมอเตอร์หมุนขวา	50
รูปที่ 3.16 ผังงานของโปรแกรมย่อยควบคุมสแต็ปปีงมอเตอร์หมุนบน	51
รูปที่ 3.17 ผังงานของโปรแกรมย่อยควบคุมสแต็ปปีงมอเตอร์หมุนล่าง	52
รูปที่ 3.18 ผังงานของโปรแกรมตรวจสอบสัญญาณนิรภัย	53
รูปที่ 3.19 ผังงานของโปรแกรมโทรศัพท์แจ้งเตือนภัย	54
รูปที่ 3.20 ผังการทำงานการควบคุมคล้องผ่านอินเตอร์เน็ต	54
รูปที่ 3.21 ผังการทำงานการส่งสัญญาณภาพผ่านเครือข่ายอินเตอร์เน็ต	55
รูปที่ 4.1 การทดลองบอร์ดควบคุมการทำงานหลัก	57
รูปที่ 4.2 การทดลองบอร์ดขยายอินพุตเอาต์พุต	58
รูปที่ 4.3 ชุดสถานะการทำงานของสแต็ปปีงมอเตอร์	59
รูปที่ 4.4 การทดลองบอร์ดขับสแต็ปปีงมอเตอร์	61
รูปที่ 4.5 การเรียกโปรแกรม Windows Media Encoder	62
รูปที่ 4.6 การเลือกลักษณะการส่งออกของสัญญาณภาพ	62
รูปที่ 4.7 การเลือกความเร็วชนิดของข้อมูลและแบนด์วิดธ์	63
รูปที่ 4.8 การเลือกที่มาของแหล่งข้อมูล	63
รูปที่ 4.9 รายละเอียดโคดรเวอร์ที่คอมพิวเตอร์มองเห็น	64
รูปที่ 4.10 การเลือกวิธีการส่ง	64
รูปที่ 4.11 จบการเลือกเพื่อเข้าสู่การเอน โคคเตอร์	65
รูปที่ 4.12 การเอน โคคเตอร์จากแหล่งข้อมูลที่เตรียมไว้	65
รูปที่ 4.13 ตัวอย่างสัญญาณภาพที่ผ่านการเอน โคคเตอร์	66
รูปที่ 4.14 เว็บเพจสำหรับผู้ใช้งาน	67

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## สารบัญรูป (ต่อ)

รูป	หน้า
รูปที่ 4.15 หน้าหลักของเว็บเพจ	67
รูปที่ 4.16 การควบคุมสแต็ปปีงมอเตอร์	68
รูปที่ 4.17 ข้อมูลของตัวตรวจจับผู้บุกรุก	68
รูปที่ 4.18 ข้อมูลองศาการหมุนของสแต็ปปีงมอเตอร์	69
รูปที่ 4.19 ผลการรัน โปรแกรมวิน โดว์มีเคอร์รี่เพื่อดูสัญญาณภาพผ่านเครือข่าย	70
รูปที่ 4.20 ฐานข้อมูลของส่วนควบคุมกล้อง	71
รูปที่ 4.21 ฐานข้อมูลของส่วนติดต่อกับสัญญาณตรวจจับผู้บุกรุก	71
รูปที่ 4.22 ฐานข้อมูลเบอร์โทรศัพท์	71
รูปที่ 4.23 ฐานข้อมูลสำหรับตรวจเช็คห้องสาขาของสแต็ปปีงมอเตอร์	72
รูปที่ 4.24 เมนูการป้อนรหัสผู้ใช้	72
รูปที่ 2.25 ฟอรัมหลักของ โปรแกรม	73
รูปที่ 4.26 โปรแกรมสั่งงานควบคุมการหมุนของสแต็ปปีงมอเตอร์	74
รูปที่ 4.27 การเลือกทิศทางการหมุนและการลดค่าของมุมมอง	74
รูปที่ 4.28 ฐานข้อมูล Microsoft Access ชื่อฐานข้อมูล STEP ตาราง STEP	75
รูปที่ 4.29 โปรแกรมสั่งงานควบคุมการหมุนของสแต็ปปีงมอเตอร์ โดยใช้ ค่าองศาจากฐานข้อมูล	76
รูปที่ 4.30 โปรแกรมสั่งงานควบคุมการหมุนของสแต็ปปีงมอเตอร์ โดยแสดง ค่าองศาจากฐานข้อมูลและแสดงการลดค่าองศา	76
รูปที่ 4.31 การป้อนค่าองศาและการเลือกทิศทางการหมุน	77
รูปที่ 4.32 ผลค่าองศาการหมุนของสแต็ปปีงมอเตอร์ที่อยู่ในตารางฐานข้อมูล	77
รูปที่ 4.33 ผลค่าองศาการหมุนของสแต็ปปีงมอเตอร์ที่อยู่ในฐานข้อมูล Microsoft Access	78
รูปที่ 4.34 โปรแกรมตรวจสอบการทำงานของอุปกรณ์เซนเซอร์	78
รูปที่ 4.35 ผลของโปรแกรมตรวจสอบการทำงานของอุปกรณ์เซนเซอร์	79
รูปที่ 4.36 โปรแกรมสั่งงาน โมเด็ม	79
รูปที่ 4.37 การทำงานของโปรแกรมเมื่อมีการป้อนหมายเลขและกดปุ่ม Dial	80

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## บทที่ 1

### บทนำ

#### 1.1 ความเป็นมาและความสำคัญของปริญญานิพนธ์

ในปัจจุบันความสถานที่ต่างๆ ได้มีการคิดตั้งอุปกรณ์ป้องกันภัยประเภทต่างๆเอาไว้แต่ อุปกรณ์เหล่านั้นยังไม่สามารถแจ้งเตือนภัยไปยังเจ้าของได้โดยตรง กล่าวคือ หากเจ้าของไม่ได้อยู่ ก็ จะไม่สามารถรับทราบเหตุการณ์ต่างๆ ที่เกิดขึ้นได้ จึงไม่สามารถทราบว่าจะเกิดเหตุขึ้นได้มี อะไรเกิดขึ้นบ้างเนื่องจากไม่มีอุปกรณ์ตัวใดที่สามารถแจ้งเตือนและส่งข้อมูลไปยังเจ้าได้รับทราบ ด้วยเหตุนี้จึงต้องมีอุปกรณ์ที่ใช้ในการแจ้งเตือนขณะที่ไม่อยู่เพื่อที่จะได้สามารถรับทราบและควบคุมเหตุการณ์ที่เกิดขึ้นได้

#### 1.2 ขีดความสามารถของโครงการ

โครงการนี้มีขีดความสามารถดังนี้

- 1) สามารถรับสัญญาณภาพได้ 1 ช่องสัญญาณ
- 2) สามารถเลือกการควบคุมการเคลื่อนที่ของกล้องได้ 4 ทิศทาง
- 3) สามารถดูและสำรวจบริเวณทั่วไป โดยดูจากคอมพิวเตอร์ซึ่งสัญญาณภาพจะถูกส่งผ่านระบบเครือข่ายอินเทอร์เน็ต
- 4) สามารถโทรแจ้งเตือนผ่านทางโทรศัพท์มือถือผู้ดูแล

#### 1.3 เนื้อหาโดยสังเขป

เนื้อหาภายในปริญญานิพนธ์ฉบับนี้แบ่งออกเป็นบทต่างๆ เพื่อสะดวกต่อการศึกษาและทำความเข้าใจ ในแต่ละบทจะประกอบด้วยเนื้อหาที่สำคัญดังนี้

บทที่ 2 ทฤษฎีและหลักการ ประกอบด้วยเนื้อหาในทางทฤษฎีที่เกี่ยวข้อง ซึ่งทำให้ผู้อ่านได้ มีความเข้าใจที่เป็นพื้นฐานเสียก่อน อันจะเป็นประโยชน์ต่อการทำความเข้าใจกับวงจรที่ใช้งานจริงต่อไป

บทที่ 3 การออกแบบ การสร้าง และการทำงาน จะกล่าวถึงการสร้าง การออกแบบ ฮาร์ดแวร์และซอฟต์แวร์ รวมทั้งหลักการทำงานต่างๆ ซึ่งจะทำให้ผู้อ่านได้เข้าใจการทำงาน โดยรวมของโครงการนี้

บทที่ 4 การทดลอง และผลการทดลอง กล่าวถึงขั้นตอนในการทดลอง และการทดสอบประสิทธิภาพในการทำงานของฮาร์ดแวร์ และซอฟต์แวร์ของโครงการนี้ เพื่อตรวจสอบว่าโครงการนี้สามารถทำงานได้ตรงตามวัตถุประสงค์หรือไม่

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 5 บทสรุป ปัญหาแนวทางการแก้ไขและพัฒนา ขั้นการสรุปผลในการจัดทำ โครงการงาน ปัญหาที่เกิดขึ้นและได้เสนอแนะแนวในการแก้ไข รวมทั้งแนวทางในการพัฒนาให้มีประสิทธิภาพ และการใช้งานได้อย่างกว้างขวางมากขึ้น



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 2.1 พื้นฐานการสื่อสารผ่านทางพอร์ตสื่อสารแบบอนุกรมโดยวิซวลเบสิก

ในการพัฒนาแอปพลิเคชันที่สามารถทำการสื่อสารและแลกเปลี่ยนข้อมูลผ่านทางพอร์ตสื่อสารแบบอนุกรม (Serial Port) ได้นั้นจำเป็นที่จะต้องมีความรู้พื้นฐานในเรื่องการสื่อสารข้อมูล แต่ในการพัฒนาแอปพลิเคชัน เพื่อใช้ในการสื่อสารผ่านทางพอร์ตสื่อสารอนุกรมโดยใช้โปรแกรมวิซวลเบสิกนั้น ผู้พัฒนาสามารถใช้คอนโทรลเลอร์สื่อสารที่วิซวลเบสิกได้เตรียมไว้ให้เพื่อใช้ในการพัฒนาแอปพลิเคชันในการสื่อสารข้อมูล ซึ่งคอนโทรลเลอร์ที่จะสามารถใช้ในการพัฒนา คือ คอนโทรลเลอร์ MSComm ซึ่งเป็นคอนโทรลเลอร์ที่มีความสามารถในการติดต่อสื่อสารข้อมูลผ่านทางพอร์ตอนุกรมได้เป็นอย่างดี มีฟังก์ชันที่ใช้ในการติดต่อสื่อสารกับพอร์ตอนุกรมอย่างครบถ้วน สามารถใช้ในการควบคุมการสื่อสาร การตรวจจับเหตุการณ์ที่เกิดขึ้นในระหว่างการสื่อสารผ่านทางพอร์ตอนุกรมเพื่อพัฒนาแอปพลิเคชันที่จะใช้ในการควบคุมเหตุการณ์นั้น ๆ

พื้นฐานในการสื่อสาร โดยผ่านทางพอร์ตอนุกรม คือ เครื่องคอมพิวเตอร์จะมีพอร์ตอนุกรมที่ใช้ในการเชื่อมต่อกับอุปกรณ์ภายนอก และหัวใจสำคัญในการสื่อสารข้อมูลโดยใช้พอร์ตอนุกรมก็คือ กระบวนการในการเปลี่ยนแปลงข้อมูลหรือการจัดเรียงข้อมูล เมื่อมีการแลกเปลี่ยนข้อมูลระหว่างข้อมูลที่รับหรือส่งผ่านทางพอร์ตอนุกรมกับ CPU กล่าวคือ ในการส่งข้อมูลจาก CPU มายังพอร์ตอนุกรมนี้ จะทำการส่งมาในรูปแบบของข้อมูลที่มีหน่วยเป็นไบนารี แต่เมื่อมาถึงพอร์ตอนุกรมแล้วจะต้องมีกระบวนการในการแปลงข้อมูลนั้นให้อยู่ในรูปของบิต ซึ่งในการส่งข้อมูลออกหรือว่าการรับข้อมูลเข้าของพอร์ตอนุกรมนั้น จะทำการส่งข้อมูลในรูปแบบที่ละบิตเป็นลำดับเรียงกันไป ในการรับข้อมูลก็เช่นเดียวกัน จะต้องมีกระบวนการในการแปลงลำดับของบิตที่รับเข้ามาผ่านทางพอร์ตอนุกรมให้เป็นไบนารีเพื่อส่งไปให้กับ CPU ต่อไป

ในการพัฒนาแอปพลิเคชันที่ใช้ในการสื่อสารผ่านทางพอร์ตอนุกรมโดยใช้คอนโทรลเลอร์ MSComm นี้ จะมีความสะดวกเป็นอย่างมาก เพราะว่ามันขั้นตอนการพัฒนาแอปพลิเคชัน ผู้พัฒนาไม่จำเป็นต้องสนใจหรือต้องทำการพัฒนาส่วนการแปลงข้อมูลหรือการจัดเรียงข้อมูลเพื่อใช้ในการติดต่อสื่อสารผ่านทางพอร์ตอนุกรม เพราะว่าการทำงานในส่วนนี้จะจัดการโดยคอนโทรลเลอร์ MSComm ทั้งหมด สิ่งที่ผู้พัฒนาจะต้องสนใจและทำการพัฒนาคือ จะทำการติดต่อกับคอนโทรลเลอร์ MSComm นี้อย่างไร คอนโทรลเลอร์นี้มีฟังก์ชันอะไรให้ใช้งานได้บ้าง

ในการตั้งค่าพื้นฐานเพื่อใช้ในการเปิดช่องทางการติดต่อกับพอร์ตอนุกรมของคอนโทรลเลอร์ MSComm จะต้องมีการตั้งค่าพื้นฐานสามส่วนด้วยกัน คือ

## ตารางที่ 2.1 ค่าพื้นฐานในการติดต่อพอร์ตอนุกรมของคอนโทรล MSComm

คุณสมบัติ	คำอธิบาย
Comport	เป็นการตั้งค่าเพื่อเลือกหมายเลขของพอร์ตอนุกรมที่จะใช้งาน โดยทั่วไปแล้วจะมีค่าเป็น 1, 2, 3 และ 4 ซึ่งหมายถึง com1, com2, com 3 และ com4 ตามลำดับ
Settings	เป็นการตั้งค่าเพื่อจะกำหนดค่าพื้นฐานต่างๆ ที่จะใช้เป็นมาตรฐานในการสื่อสารข้อมูลผ่านทางพอร์ตอนุกรม เช่น ค่าความเร็วของช่องทางเดินของข้อมูลการเซ็ทบิตเพื่อใช้เช็คความถูกต้องของข้อมูลจำนวนชุดของบิต ที่จะส่งการเลือกใช้บิตหยุด เป็นต้น โดยจะทำการส่งค่าเป็นแบบข้อความ เช่น "9600,N,8,1" เป็นต้น
PortOpen	เป็นการตั้งค่าในการเปิดหรือการปิดพอร์ตอนุกรม ที่จะใช้ในการสื่อสารข้อมูล

ค่าต่างๆ ที่ใช้ในการตั้งค่าให้กับพรอพเพอร์ตี้ Settings จะมีอยู่ด้วยกันสี่ส่วน ได้แก่

- 1) Baud Rate
- 2) Parity Values
- 3) Valid Data Bit Values
- 4) Stop Bit Values

ซึ่งสามารถตั้งค่าต่างๆ ที่เป็นไปได้ของพรอพเพอร์ตี้แต่ละตัวดังนี้ คือ

- 1) Baud Rate จะมีค่าได้ดังนี้ คือ 110, 300, 600, 1200, 2400, 9600 (เป็นค่ามาตรฐาน), 14400, 28800, 38400, 56000, 128000, 256000
- 2) Parity Values จะมีค่าได้ดังนี้ คือ

ตารางที่ 2.2 ค่าพาริตีในการตั้งค่าพรอพเพอร์ตี้

คำย่อ	ความหมาย
E	Even
M	Mark
N	None (เป็นค่ามาตรฐาน)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## ตารางที่ 2.2 (ต่อ) ค่าพารามิเตอร์ในการตั้งค่าพอร์ทพอร์ต

คำย่อ	ความหมาย
O	Odd
S	Space

3) Valid Data Bit Values จะมีค่าได้ดังนี้ คือ 4, 5, 6, 7, 8 (เป็นค่ามาตรฐาน)

4) Stop Bit Values จะมีค่าได้ดังนี้ คือ 1 (เป็นค่ามาตรฐาน), 1.5, 2

ถ้าต้องการเปิดพอร์ตอนุกรมเพื่อใช้ในการสื่อสารข้อมูลผ่านทางพอร์ท com 2 และต้องการให้มี Baud Rate เป็น 9600 ไม่มีการเช็คค่า Parity Bit มี Valid Data Bit เป็น 8 และ Stop Bit เป็น 1 จะต้องทำการตั้งค่าพอร์ทพอร์ตต่าง ๆ ของคอนโทรล MSComm ดังนี้ คือ

```
'Open the Serial Port (Com 2)
```

```
MSComm1.CommPort = 2
```

```
MSComm1.Settings = "9600,8,1"
```

```
MSComm1.PortOpen = True
```

เมื่อทำการเปิดพอร์ตอนุกรมเรียบร้อยแล้วก็จะสามารถทำการส่งหรือรับข้อมูลผ่านทางพอร์ตอนุกรมที่เปิดได้ โดยการเชื่อมต่อในการรับส่งข้อมูลที่คอนโทรล MSComm ได้จัดเตรียมไว้

### 2.1.1 การรับและส่งข้อมูลผ่านพอร์ตอนุกรมโดยคอนโทรล MSComm

ในการรับหรือส่งข้อมูลผ่านทางพอร์ตอนุกรมนี้สามารถที่จะทำได้ก็ต่อเมื่อได้มีการเปิดการเชื่อมต่อกับพอร์ตอนุกรมเป็นที่เรียบร้อยแล้วเท่านั้น โดยในการรับหรือส่งข้อมูลโดยการใช้อินโทรล MSComm นี้จะรับส่งผ่านบัฟเฟอร์ ซึ่งคอนโทรลสื่อสารนี้ได้เตรียมไว้ให้ โดยจะแยกการรับและการส่งข้อมูลผ่านทางบัฟเฟอร์ของแต่ละส่วน โดยจะประกอบด้วย

1) บัฟเฟอร์ที่ใช้ในการส่งข้อมูล (The Transmit Buffer)

2) บัฟเฟอร์ที่ใช้ในการรับข้อมูล (The Receive Buffer)

ในการรับหรือส่งข้อมูลผ่านทางบัฟเฟอร์นี้จะสามารถรับหรือส่งได้ทั้งข้อมูลที่เป็นข้อความหรือข้อมูลที่เป็นไบนารี เช่น รูปภาพหรือไฟล์วีดีโอ เป็นต้น โดยในการรับหรือส่งข้อมูลโดยใช้อินโทรล MSComm จะมีลักษณะการใช้งานดังนี้

การรับข้อมูลเพื่อใช้แสดงในคอนโทรลเท็กซ์ จะใช้เมธอด Input ดังนี้

```
TextDisplay.Text = MSComm1.Input
```

การส่งข้อมูลออกไปยังพอร์ตอนุกรม จะใช้เมธอด Output ดังนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่นับผูกมัดให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

MSCComm1.Output = "This is a text string"

เมื่อทราบการตั้งค่าต่างๆ เพื่อใช้ในงานคอนโทรลแล้ว ในการพัฒนาแอปพลิเคชันเพื่อใช้ในการติดต่อกับพอร์ตอนุกรมก็จะสามารถทำได้ง่าย สิ่งที่ทำาการพัฒนาแอปพลิเคชันสื่อสารข้อมูลต้องสนใจคือ การจัดการรับข้อมูลเข้าผ่านพอร์ตอนุกรมเพื่อจะนำมาใช้ในการประมวลผลหรือส่งข้อมูลที่เป็นผลลัพธ์จากการประมวลผลผ่านทางพอร์ตอนุกรมเพื่อนำไปใช้งานต่อไปเท่านั้น

### 2.1.2 การใช้งานคอนโทรล MSCComm

คอนโทรล MSCComm เป็นคอนโทรลตัวหนึ่งซึ่งช่วยติดต่อกับพอร์ตอนุกรม (Serial Port) ซึ่งผู้อ่านทำการรับ-ส่งข้อมูลผ่านทางพอร์ตอนุกรมได้ด้วยคอนโทรลนี้ เช่น การติดต่อโดยตรงกับบอร์ดอิเล็กทรอนิกส์ เป็นต้น ซึ่งคอนโทรล MSCComm ที่มากับวิชาลเบสิกจะเป็นคอนโทรลที่ทำงานโดยมีการตอบสนองต่อเหตุการณ์แบบ even-driven นั่นก็คือคอนโทรลจะทำหน้าที่ตรวจสอบการเกิดขึ้นหรือการร้องขอให้เกิดเหตุการณ์ต่างๆ กับพอร์ตอนุกรมโดยอัตโนมัติ และจะมีการแจ้งเตือนให้ได้ทราบผ่านทางไพร์ซีเควนซ์เหตุการณ์ เช่นเดียวกับคอนโทรลทั่วไปของวิชาลเบสิกนั่นเอง

คอนโทรล MSCComm จะมีหน้าที่มาตรฐานหลักๆ สำหรับกรสื่อสารผ่านพอร์ตอนุกรม 3 ประการ ดังต่อไปนี้

- 1) หมุนหมายเลขติดต่อกับโทรศัพท์ปลายทางที่กำหนด
- 2) ตรวจสอบการเข้ามาของข้อมูลยังพอร์ตอนุกรมโดยอัตโนมัติ
- 3) ส่งข้อมูลตามที่กำหนดจากโปรแกรมไปยังพอร์ตอนุกรม

ในความเป็นจริงคอนโทรล MSCComm ไม่ได้ทำหน้าที่ติดต่อกับพอร์ตอนุกรมโดยตรง แต่มันจะทำหน้าที่เรียกใช้ฟังก์ชันวินโดวส์ API ซึ่งวินโดวส์จะทำการส่งหรือรับข้อมูลผ่านทางพอร์ตอนุกรมโดยอาศัยไดรเวอร์ Comm.drv จากนั้นก็จะส่งผ่านข้อมูลที่ถูกรูปแบบตามมาตรฐานการสื่อสาร (ทั้งนี้ขึ้นอยู่กับอุปกรณ์ที่ต่อเข้ากับพอร์ตอนุกรม) ให้กับดีไวซ์ไดรเวอร์อีกทอดหนึ่ง

### 2.2 ความรู้เบื้องต้นเกี่ยวกับพอร์ตอนุกรม

การเคลื่อนย้ายข้อมูลจากคอมพิวเตอร์ไปยังอุปกรณ์ต่อพ่วงภายนอกหรือคอมพิวเตอร์ด้วยกันมีกัน 2 รูปแบบคือ รับส่งข้อมูลแบบขนานและรับส่งข้อมูลแบบอนุกรม

การรับส่งข้อมูลแบบขนานเป็นการรับและส่งข้อมูลคราวละ 4 ถึง 8 บิตในเวลาเดียวกัน ทำให้การรับและส่งข้อมูลมีความเร็วสูง แต่จำนวนสายที่ใช้ในการถ่ายทอกข้อมูลมีมากเท่ากับจำนวนบิตของข้อมูลที่ทำาการถ่ายทอก นอกจากนั้นยังมีสายที่ใช้สำหรับควบคุมและตรวจสอบการรับส่งข้อมูลด้วย ซึ่งอาจต้องใช้อย่างมากเป็น 2 เท่าของจำนวนบิตข้อมูลก็ได้

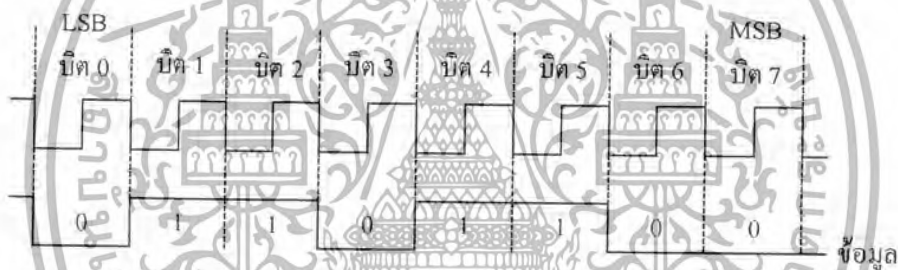
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ในขณะที่การรับส่งข้อมูลแบบอนุกรมจะเป็นการรับส่งข้อมูลครั้งละ 1 บิต โดยมีรูปแบบการรับส่งที่เป็นมาตรฐาน ต้องมีการตรวจสอบความพร้อมในการรับและส่งข้อมูลของตัวส่งและตัวรับ การรับส่งข้อมูลแบบอนุกรมมีข้อดีในเรื่องจำนวนสายสัญญาณที่น้อยมากและไม่แปรผันตามจำนวนบิตของข้อมูล ระยะทางในการรับส่งข้อมูลสูงกว่าแบบขนานมาก

การสื่อสารแบบอนุกรมแบ่งออกเป็น 2 แบบคือ การสื่อสารอนุกรมแบบซิงโครนัสและการสื่อสารอนุกรมแบบอะซิงโครนัส

### 2.2.1 การสื่อสารข้อมูลแบบซิงโครนัส

การสื่อสารแบบซิงโครนัสจะมีสัญญาณนาฬิกา ร่วมอยู่กับการรับและส่งสัญญาณด้วยตัวอย่างการส่งข้อมูลแบบซิงโครนัสก็คือคีย์บอร์ดของคอมพิวเตอร์ ซึ่งสายเส้นหนึ่งจะเป็นสายของสัญญาณนาฬิกา ส่วนสายอีกเส้นจะเป็นสายของข้อมูล ดังนั้นการติดต่อกันแบบซิงโครนัสนี้จะต้องใช้สายในการเชื่อมต่ออย่างน้อยที่สุด 3 เส้นคือ สัญญาณนาฬิกา, ข้อมูลและกราวด์ รูปที่ 2.1 แสดงให้เห็นถึง โค้ดแอมเวลตาของการสื่อสารข้อมูลแบบซิงโครนัส



รูปที่ 2.1 แผนผังการทำงานเวลาของการสื่อสารข้อมูลแบบซิงโครนัส

### 2.2.2 การสื่อสารข้อมูลแบบอะซิงโครนัส

การสื่อสารข้อมูลแบบอะซิงโครนัสคือการรับและส่งข้อมูลโดยไม่จำเป็นต้องมีสัญญาณนาฬิกา ร่วมด้วย แต่จะใช้การกำหนดค่าอัตราความเร็วในการรับและส่งข้อมูลให้มีค่าเท่ากัน ซึ่งเรียกอัตราเร็วนี้ว่า อัตราบอดหรือบอดเรต (Baud Rate) มีหน่วยเป็น บิตต่อวินาที

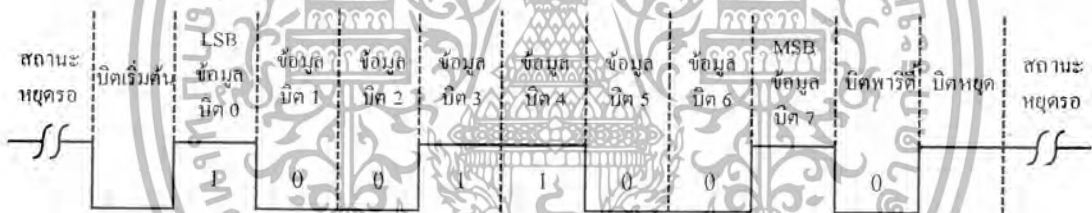
รูปแบบของข้อมูลที่ใช้ในการรับส่งแบบอะซิงโครนัสประกอบด้วย 4 ส่วนด้วยกันคือ

- 1) บิตเริ่มต้น
- 2) บิตข้อมูลแบบอนุกรม มีขนาด 5, 6, 7 หรือ 8 บิต
- 3) บิตตรวจสอบพาริตี (Parity Bit) มีขนาด 1 บิต หรือไม่มีบิต
- 4) บิตปิดท้ายหรือบิตหยุด (Stop Bit) มีขนาด 1, 1.5, หรือ 2 บิต

รูปที่ 2.2 แสดงรูปแบบของข้อมูลอนุกรมแบบอะซิงโครนัส เมื่อไม่มีการส่งข้อมูลหา DATA จะมีสถานะลอจิก "1" เรียกสถานะนี้ว่า สถานะหยุดรอ (Waiting Stage) การเริ่มต้นส่งข้อมูลเอกสารนี้เป็นเอกสารที่ส่งวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จะเริ่มจากการให้ขา DATA มีลอจิก "0" ด้วยช่วงระยะเวลา 1 บิต เรียกบิตนี้ว่าบิตเริ่มต้น (Start Bit) จากนั้นบิตข้อมูลจะถูกส่งออกไป โดยเริ่มจากบิตที่มีนัยสำคัญต่ำสุดหรือบิต LSB ก่อน ซึ่งข้อมูลที่ต้องการส่งอาจมีจำนวน 5, 6, 7 หรือ 8 บิตก็ได้ จากนั้นตามด้วยบิตพาริตี (Parity Bit) ซึ่งใช้ในการตรวจสอบความผิดพลาดที่เกิดขึ้นจากการส่งข้อมูล บิตสุดท้ายที่จะส่งก็คือ บิตปิดท้ายหรือบิตหยุด (Stop Bit) โดยจะเป็นการทำให้ขา DATA มีสถานะลอจิก "1" อีกครั้งด้วยระยะเวลาอย่างน้อย 1 บิต 1.5 บิต หรือ 2 บิต เพื่อเป็นการแสดงว่าสิ้นสุดข้อมูลแล้ว

อัตราความเร็วในการรับและส่งข้อมูลของการรับส่งข้อมูลแบบอะซิงโครนัสหรืออัตราบอดหรือบอดเรตที่ใช้สำหรับพอร์ตอนุกรม RS-232 มีด้วยกันหลายค่า ได้แก่ 110, 150, 300, 600, 1200, 2400, 4800, 9600 และ 19200 บิตต่อวินาที โดยมีค่ามากขึ้นตามเทคโนโลยีของคอมพิวเตอร์ เนื่องจากบอดเรตคือค่าของจำนวนบิตที่สามารถส่งได้ใน 1 วินาที สมมติว่า ข้อมูลอนุกรมมีขนาด 8 บิต ไม่มีการตรวจสอบพาริตี มีบิตเริ่มต้น 1 บิต และบิตปิดท้าย 1 บิต ความยาวของข้อมูล 1 ไบต์ จะมีความยาวเท่ากับ 10 บิต ถ้าใช้บอดเรตในการส่งข้อมูลเท่ากับ 9.600 บิตต่อวินาที ก็จะสามารถส่งข้อมูล ได้ด้วยความเร็ว 960 ไบต์ต่อวินาที



รูปที่ 2.2 รูปแบบของข้อมูลแบบอะซิงโครนัส

การตรวจสอบพาริตีสามารถกำหนดเป็นแบบคี่ (Odd), แบบคู่ (Even) หรือไม่มีการตรวจสอบพาริตีก็ได้ พาริตีคี่หรือพาริตีคู่แสดงถึงจำนวนลอจิก "1" ทั้งหมดภายในข้อมูลที่ส่งไป 1 ไบต์ รวมพาริตีว่ามีจำนวนเป็นเลขคู่หรือเลขคี่ ยกตัวอย่างข้อมูลที่จะทำการส่งมีขนาด 8 บิต มีค่าเท่ากับ 99H หรือ 10011001B จะเห็นว่าข้อมูลในไบต์มีจำนวนลอจิก "1" จำนวน 4 ตัวซึ่งเป็นเลขคู่ ดังนั้นถ้ากำหนดค่าพาริตีเป็นคู่ ค่าของพาริตีบิตจะต้องมีลอจิกเป็น "0" แต่ถ้ากำหนดพาริตีเป็นคี่ ค่าของบิตพาริตีจะต้องเป็น "1" เพื่อให้ข้อมูล 1 ไบต์รวมทั้งบิตพาริตีเป็นคี่

บิตพาริตีถูกสร้างขึ้นจากภาคส่งข้อมูลของ UART (Universal Asynchronous Receiver Transmitter) ซึ่งทางภาครับจะต้องกำหนดคุณสมบัติการตรวจสอบพาริตีที่ตรงกันเอาไว้ว่าจะตรวจสอบพาริตีคี่หรือพาริตีคู่ โดยการนับจำนวนลอจิก "1" ทั้งหมดรวมทั้งบิตพาริตีด้วย ถ้ากำหนดพาริตีไว้เป็นคู่แต่อ่านค่าตัวเลขในการนับออกมาได้ตัวเลขเป็นคี่ ทางภาครับจะแสดงข้อผิดพลาดออกมา เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ให้ผู้ใช้งานทราบ กระบวนการดังกล่าวเป็นวิธีการตรวจสอบความผิดพลาดที่เกิดขึ้นในการรับส่งข้อมูลที่ง่ายที่สุด แต่สามารถตรวจสอบได้เมื่อมีบิตข้อมูลที่ทำการรับส่งผิดพลาดเพียงบิตเดียวเท่านั้น ถ้าข้อมูลที่ทำการส่งมีบิตที่ผิดพลาดมากกว่า 1 บิต การตรวจสอบด้วยวิธีนี้จะไม่ได้ผล สำหรับการตั้งพาริตีบิตเป็น NONE นั้นทั้งภาครับและภาคส่ง จะไม่มีการตรวจสอบพาริตี

คอมพิวเตอร์ในรุ่น AT เกือบทั้งหมดจะใช้ไอซี UART เบอร์ 16450 และ 16550 ส่วนคอมพิวเตอร์ในรุ่น XT ใช้ไอซี UART เบอร์ 8250 ไอซี UART เหล่านี้มีระดับแรงดันของลอจิกเป็นแบบทีทีแอล (+5V) แต่เพื่อให้แรงดันเป็นไปตามมาตรฐาน RS-232 และเพื่อให้การรับส่งข้อมูลสามารถทำได้ในระยะไกลมากยิ่งขึ้น ระดับแรงดันทีทีแอลจะถูกแปลงเป็นระดับแรงดันที่สูงขึ้น โดยลอจิก "0" จะมีระดับแรงดัน -3V ถึง -12V และลอจิก "1" มีระดับแรงดัน +3V ถึง +12V

### 2.2.3 มาตรฐานพอร์ตอนุกรมแบบ RS-232

มาตรฐานการเชื่อมต่อแบบอนุกรม RS-232 เป็นมาตรฐานอุตสาหกรรมที่ออกแบบมาเพื่อใช้ในการส่งข้อมูลอนุกรมแบบอะซิงโครนัส 2 ทิศทาง โดยมาตรฐาน RS-232 ในอดีตนั้นออกแบบมาเพื่อการส่งผ่านข้อมูลจากคอมพิวเตอร์ไปยัง โมเด็มเพียงอย่างเดียวเพื่อที่จะนำข้อมูลจาก โมเด็มนี้ส่งผ่านสายโทรศัพท์ไปยังคอมพิวเตอร์อีกชุดซึ่งอยู่ห่างไกลโดยสมาคมอุตสาหกรรมอิเล็กทรอนิกส์ (Electronic Industries Association : EIA) ได้วางมาตรฐานที่มีชื่อเรียกกันว่า EIA RS-232 มาตรฐานนี้ในช่วงแรกจะใช้คอนเน็กเตอร์เป็นแบบ DB-25 โดยกำหนดความยาวสูงสุดของสายสัญญาณไว้ที่ 50 ฟุต มีระดับสัญญาณตั้งแต่ -3V จนถึง -12V แสดงว่ามีข้อมูล (mark) และ +3V จนถึง +12V แสดงว่าเป็นช่องว่าง (space)

มาตรฐาน RS-232 ถูกใช้ในการกำหนดรูปแบบการสื่อสารข้อมูลกันระหว่างอุปกรณ์เชื่อมต่อข้อมูล (Data Terminal Equipment : DTE) กับวงจรขั้วสุดท้ายทาง (Data Circuit Terminating : DCE) อุปกรณ์ DTE จะต้องเป็นอุปกรณ์ที่มีการประมวลผลในตัวเช่น ไมโครคอนโทรลเลอร์หรือไมโครคอมพิวเตอร์ซึ่งมีความสามารถในการสร้างบิตข้อมูลแบบอนุกรมได้ ส่วนอุปกรณ์ DCE ทำหน้าที่เป็นเพียงตัวรับข้อมูลที่ส่งมาจาก DTE เท่านั้น

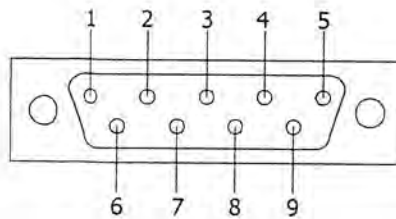
ข้อแตกต่างของอุปกรณ์ DTE และอุปกรณ์ DCE อย่างหนึ่งที่ได้ชี้ชัดคือ คอนเน็กเตอร์ของ DTE จะเป็นตัวผู้ ส่วนคอนเน็กเตอร์ของ DCE จะเป็นตัวเมีย ซึ่งพอร์ตอนุกรมของคอมพิวเตอร์ที่ใช้กันอยู่ทั่วไปจะเป็นแบบ DTE ส่วนคอนเน็กเตอร์ที่อยู่ทีโมเด็มจะเป็นแบบ DCE

สำหรับการใช้งานคอมพิวเตอร์ พอร์ตอนุกรม RS-232 ถูกใช้เพื่อเชื่อมต่อกับ โมเด็ม, เม้าส์ และเครื่องพิมพ์ที่สามารถติดต่อทางพอร์ตอนุกรมได้

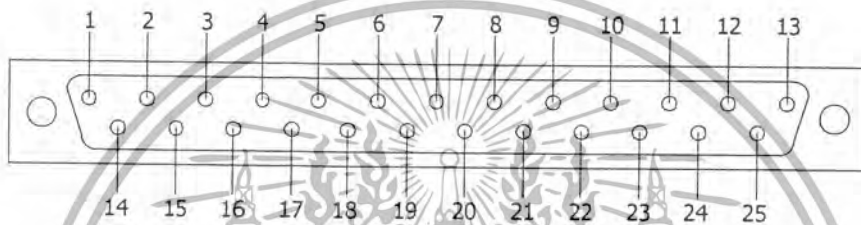
### 2.2.4 คอนเน็กเตอร์สำหรับพอร์ต RS-232 และการเชื่อมต่อ

มาตรฐานการเชื่อมต่อแบบ RS-232 จะใช้คอนเน็กเตอร์แบบ DB-25 ตัวผู้ หรือ DB-9 ตัวผู้ ซึ่งคอนเน็กเตอร์แบบ DB-25 จะมีขาต่อใช้งานเพียง 9 เส้น เช่นเดียวกับคอนเน็กเตอร์แบบ DB-9 เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เนื่องจากขาอื่นๆ ที่เคยมีใช้งานมาในอดีตไม่ค่อยมีความสำคัญมากนักจึงถูกยกเลิกไป โดยแสดงรูปร่างและตำแหน่งขาในรูปที่ 2.3



(ก) คอนเน็กเตอร์อนุกรม 9 ขา หรือแบบ DB-9 (มองจากด้านหลังคอมพิวเตอร์)



(ข) คอนเน็กเตอร์อนุกรม 25 ขา หรือแบบ DB-25 (มองจากด้านหลังคอมพิวเตอร์)

รูปที่ 2.3 คอนเน็กเตอร์อนุกรม

ตารางที่ 2.3 การจัดขาสัญญาณของพอร์ตอนุกรมในแบบต่างๆ และหน้าที่การทำงาน

คอนเน็กเตอร์ DB-9	คอนเน็กเตอร์ DB-25	ชื่อของสายสัญญาณ	ชนิดของสายสัญญาณ
1	8	Data Carrier Detect :DCD	อินพุต
2	3	Received Data :RxD	อินพุต
3	2	Transmitted Data :TxD	เอาต์พุต
4	20	Data Terminal Ready :DTR	เอาต์พุต

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 2.3 (ต่อ) การจัดขาสัญญาณของพอร์ตอนุกรมในแบบต่างๆ และหน้าที่การทำงาน

คอนเน็กเตอร์ DB-9	คอนเน็กเตอร์ DB-25	ชื่อของสายสัญญาณ	ชนิดของสายสัญญาณ
5	7	Single Ground : GND	-
6	6	Data Set Ready : DSR	อินพุต
7	4	Request To Send : RTS	เอาต์พุต
8	5	Clear To Send : CTS	อินพุต
9	22	Ring Indicator : RI	อินพุต

ขา Data Carrier Detect :DCD หรืออาจเรียกว่า Carrier Detect :CD ขานี้จะแอกติฟเมื่อมีการส่งสัญญาณพาห้จากอุปกรณ์สื่อสารข้อมูล เช่น โมเด็ม สำหรับใช้งานปกติ ขานี้จะไม่ถูกนำมาใช้งานมากนัก

ขา Receive Data : RD หรือ RxD ขานี้ใช้เพื่อรับสัญญาณอนุกรมเข้ามายังคอมพิวเตอร์ โดยจะนำข้อมูลที่อ่านได้ไปเก็บไว้ในรีจิสเตอร์บัฟเฟอร์

ขา Transmitted Data : TD หรือ TxD ขานี้ใช้เพื่อส่งข้อมูลอนุกรมออกจากคอมพิวเตอร์ โดยการนำข้อมูลที่เก็บอยู่ในบัฟเฟอร์สำหรับส่งข้อมูลส่งออกไป

ขา Data Terminal Ready : DTR เป็นขาเอาต์พุตที่ใช้สำหรับ ส่งสัญญาณ ออกจากคอมพิวเตอร์เพื่อให้อุปกรณ์ปลายทางรับรู้ว่าการติดต่อกับอุปกรณ์ปลายทาง โดยขา DTR นี้จะต้องเชื่อมต่อกับขา DSR ของอุปกรณ์ปลายทาง และขา DTR ของอุปกรณ์ปลายทางจะต้องเชื่อมต่อกับขา DSR ของคอมพิวเตอร์และถ้าใช้การเชื่อมต่อแบบ 3 สาย ต้องเชื่อมต่อกับขา DTR และ DSR ของพอร์ตอนุกรมเข้าด้วยกัน และจะต้องต่อเชื่อมเข้ากับขา DCD ด้วยในกรณีที่โปรแกรมสื่อสารที่ใช้มีการตรวจจับสัญญาณพาห้

ขา Signal Ground : GND เป็นขากราวด์ของสัญญาณ

ขา Data Set Ready : DSR ขานี้จะใช้ควบคู่กับขา DTR เพื่อตรวจสอบการเชื่อมต่อกันระหว่างคอมพิวเตอร์กับอุปกรณ์ปลายทาง ซึ่งขา DSR นี้จะเป็นขาสำหรับรับข้อมูลจากภายนอก

ขา Request To Send : RTS เป็นขาเอาต์พุตสำหรับส่งสัญญาณร้องขอให้อุปกรณ์ปลายทางส่งข้อมูลมาให้คอมพิวเตอร์โดยขาที่รับสัญญาณ RTS ก็คือขา CTS ซึ่งในกรณีที่มีการเชื่อมต่อแบบ 3 สาย จะต้องเชื่อมต่อกับขา RTS และ CTS เข้าด้วยกัน เพื่อให้การรับและส่งข้อมูลสามารถเกิดขึ้นได้ตลอดเวลา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ขา Clear To Send : CTS เป็นขาอินพุตทำหน้าที่รอรับสัญญาณที่ส่งเข้ามา เมื่อมีการส่งสัญญาณเข้ามาที่ขา RXD ข้อมูลที่ขา TXD จะถูกส่งออกไป ขานี้จะใช้เพื่อตรวจสอบอุปกรณ์ต่อพ่วงว่าพร้อมที่จะรับข้อมูลแล้วหรือยัง

ขา Ring Indicator :RI ใช้แสดงสถานะสัญญาณเรียกจากสายโทรศัพท์ ปกติในการสื่อสารโดยทั่วไปสายนี้จะไม่ถูกใช้งาน จะใช้งานก็ต่อเมื่อมีการเชื่อมต่อกับ โมเด็มแล้วยังมีความต้องการตรวจสอบสัญญาณเรียกจากสายโทรศัพท์

สำหรับการเชื่อมต่อสายระหว่างคอมพิวเตอร์กับอุปกรณ์ภายนอกแสดงดังในรูปที่ 2.4 ลูกศรในรูป แสดงถึงทิศทางของข้อมูล การเชื่อมต่อในรูปที่ 2.4 (ก) เป็นการเชื่อมต่อแบบ NULL MODEM หรือการเชื่อมต่อโดยตรงโดยไม่ต้องผ่าน โมเด็ม ส่วนการเชื่อมต่อในรูปที่ 2.4 (ข) เป็นการเชื่อมต่อโดยใช้สัญญาณน้อยที่สุดเพียง 3 เส้น โดยเส้นหนึ่งสำหรับส่งข้อมูล อีกเส้นสำหรับรับข้อมูล และเส้นสุดท้ายเป็นกราวด์



(ข) การต่ออุปกรณ์ภายนอกเข้ากับคอมพิวเตอร์แบบ RS-232  
ในลักษณะที่ใช้สายสัญญาณน้อยที่สุดเพียง 3 เส้น

**รูปที่ 2.4** การต่ออุปกรณ์ภายนอกเข้ากับคอมพิวเตอร์ผ่านทางพอร์ตอนุกรมในรูปแบบต่างๆ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

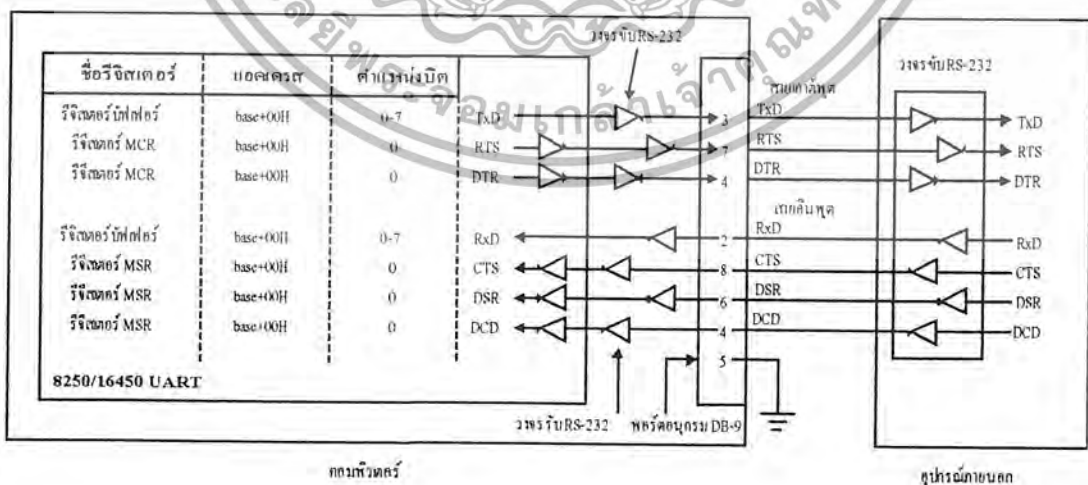
2.2.5 มาตรฐานการรับส่งข้อมูลแบบอะซิงโครนัส (UART)

UART มาจากคำว่า Universal Asynchronous Receiver Transmitter ซึ่งหมายถึงอุปกรณ์ที่ทำหน้ารับและส่งข้อมูลแบบอะซิงโครนัสนั่นเอง สำหรับการสื่อสารอนุกรมบนคอมพิวเตอร์แล้ว UART ถือว่าเป็นหัวใจสำคัญของการสื่อสารอนุกรม

หน้าที่หลักของ UART คือแปลงข้อมูลที่อยู่ในรูปแบบขนานจากซีพียูให้อยู่ในรูปแบบอนุกรมแบบอะซิงโครนัส แล้วทำการส่งออกไปและแปลงสัญญาณอนุกรมแบบอะซิงโครนัสที่ป้อนเข้ามายัง UART ให้เป็นแบบขนานก่อนที่จะส่งเข้าสู่ซีพียู ซึ่งนอกจาก UART จะส่งข้อมูลไปยังซีพียูแล้วยังแจ้งรายละเอียดอื่นๆ ของข้อมูล ให้คอมพิวเตอร์รับทราบด้วย เช่น อัตราความเร็วในการรับส่งข้อมูลหรือบอดเรต, รูปแบบการส่งข้อมูล, ความผิดพลาดที่เกิดขึ้นระหว่างการส่งข้อมูล เช่น ผิดพลาดจากพาริตี, เฟรมข้อมูล, โอเวอร์รัน เป็นต้น

ภายใน UART จะมีวงจรสร้างบอดเรตโปรแกรมได้ (Programmable Baudrate Generator) โดยการกำหนดค่าตัวหารให้กับสัญญาณนาฬิกาของ UART โดยตัวหารนี้จะมีขนาด 16 บิต ดังนั้นจะสามารถกำหนดตัวหารอยู่ในช่วง 1-65,535

มาตรฐานการรับส่งข้อมูลแบบอะซิงโครนัสที่ใช้ในคอมพิวเตอร์ทั่วไปจะมี UART ที่ใช้งานกันอยู่ 2 เบอร์ คือ เบอร์ 8250 และ 16550 สำหรับ UART เบอร์ 8250 เป็น UART มาตรฐานที่มีใช้กันมายาวนาน UART เบอร์นี้มีบัฟเฟอร์สำหรับรับและส่งข้อมูลเป็นตำแหน่งเดียวกัน ทำให้การรับและส่งข้อมูลถูกจำกัดความเร็วอยู่ที่ 57.6 กิโลบิตต่อวินาที สำหรับ UART เบอร์ 16550 จะเพิ่มส่วนของชิพที่รีจิสเตอร์แบบ FIFO (First In First Out) ขนาด 16 ไบต์เข้าไปด้วย ทำให้สามารถสนับสนุนความเร็วในการรับส่งข้อมูลที่ระดับ 256 กิโลบิตต่อวินาทีได้



รูปที่ 2.5 ผังการทำงานภายในของขาสัญญาณต่างๆ ของพอร์ตอนุกรม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

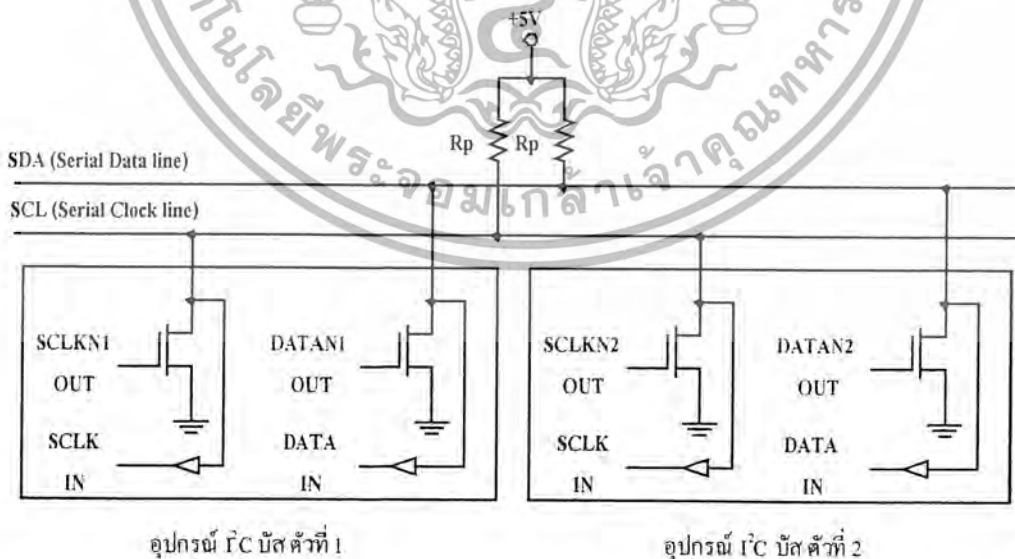
### 2.3 การเชื่อมต่อพอร์ตอนุกรมกับระบบบัส I<sup>2</sup>C

I<sup>2</sup>C ย่อมาจาก Inter – IC Communication หมายถึง การติดต่อสื่อสารระหว่างไอซีโดยบัส I<sup>2</sup>C ได้รับการพัฒนาขึ้นโดยฟิลิปส์ (Philips) ด้วยจุดมุ่งหมายหลักคือ ต้องการให้ไอซีหรือโมดูลสามารถติดต่อ ตั้งงานและควบคุมภายใต้สายสัญญาณเพียง 2 เส้น เส้นหนึ่งคือ สายข้อมูล อีกเส้นหนึ่งคือ สายสัญญาณนาฬิกาที่ใช้ในการกำหนดจังหวะการทำงาน การต่อร่วมกันของอุปกรณ์บนบัส I<sup>2</sup>C ทำได้ง่ายมาก เพียงต่อสายข้อมูลและสายสัญญาณนาฬิกาของอุปกรณ์แต่ละตัวขนาน หรือพ่วงกันไป ส่วนการกำหนดแอดเดรสหรือตำแหน่งสำหรับติดต่ออุปกรณ์แต่ละตัว จะใช้รหัสข้อมูลและการกำหนดสภาวะลอจิกที่ขาแอดเดรสของอุปกรณ์แต่ละตัว

สายข้อมูลบนบัส I<sup>2</sup>C มีชื่อเรียกอย่างเป็นทางการว่า สายข้อมูลอนุกรม หรือ SDA (Serial Data line) ส่วนสายสัญญาณนาฬิกามีชื่อเรียกว่า สายสัญญาณนาฬิกาอนุกรม หรือ SCL (Serial Clock line) ในการอธิบายต่อไปนี้จะเรียกสายสัญญาณทั้งสองว่า สาย SDA และ SCL

#### 2.3.1 คุณสมบัติโดยทั่วไปของบัส I<sup>2</sup>C

สาย SDA และ SCL เป็นสายสัญญาณ 2 ทิศทาง (Bi – Directional Line) ต้องมีการต่อตัวต้านทานพูลอัพกับแรงดัน +5V ไว้ตลอดเวลา เพื่อให้สายมีสถานะลอจิกสูงในขณะที่ไม่มีการติดต่อใช้งาน ทั้งยังช่วยในการป้องกันสัญญาณรบกวนที่อาจมีเข้ามาในสายสัญญาณทั้งสองวงจรเอาต์พุตของอุปกรณ์ที่ต่ออยู่บนบัส I<sup>2</sup>C ต้องมีลักษณะเป็นวงจรทรานซิสเตอร์เปิด (Open – Drain) หรือคอลเล็กเตอร์เปิด (Open – Collector) ดังแสดงรายละเอียดในรูปที่ 2.6

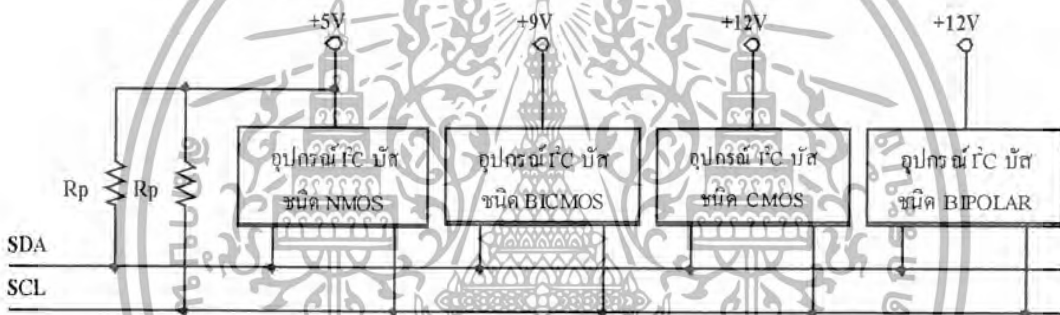


รูปที่ 2.6 โครงสร้างวงจรเอาต์พุตของอุปกรณ์ที่ใช้การเชื่อมต่อนระบบบัส I<sup>2</sup>C

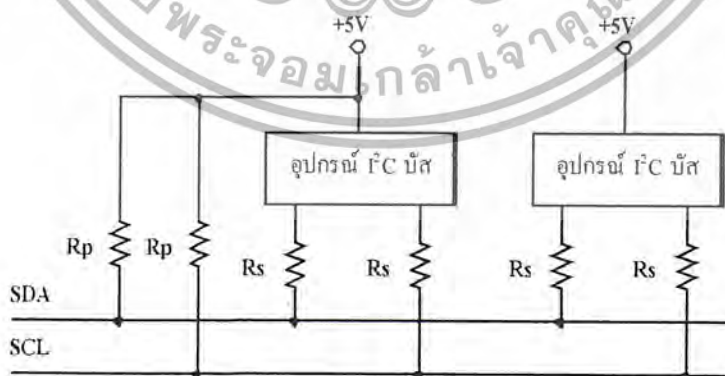
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

อัตราการถ่ายข้อมูลบนบัส I<sup>2</sup>C สูงถึง 100 กิโลบิตต่อวินาทีในโหมดปกติ (Standard Mode) และสูงถึง 400 กิโลบิตต่อวินาที ในโหมดความเร็วสูง (Fast Mode) อุปกรณ์ที่ต่ออยู่บนบัส I<sup>2</sup>C จะต้องมีค่าความจุไฟฟ้ารวมที่เกิดขึ้นระหว่างสาย SDA และ SCL ไม่เกิน 400 pF การเข้าถึงอุปกรณ์บนบัส I<sup>2</sup>C ใช้ข้อมูลสำหรับการเข้าถึง 2 ค่า คือ 7 บิต (7 – Bit Addressing) หรือ 10 บิต (10 – Bit Addressing)

ข้อเด่นอีกประการหนึ่งของบัส I<sup>2</sup>C คือ สามารถเชื่อมต่ออุปกรณ์ที่ใช้ไฟเลี้ยงไม่เท่ากันให้สามารถติดต่อสื่อสารกันได้โดยอุปกรณ์บนบัส I<sup>2</sup>C ตัวหนึ่งอาจใช้ไฟเลี้ยง +5V ในขณะที่อีกตัวหนึ่งใช้ไฟเลี้ยง +12V การต่อร่วมกันบนบัส I<sup>2</sup>C สามารถกระทำได้ในลักษณะเดียวกับกรณีที่อุปกรณ์ทั้งสองใช้ไฟเลี้ยงเท่ากัน กล่าวคือ ให้ต่อสาย SDA และ SCL ของอุปกรณ์แต่ละตัวเข้าด้วยกัน และต้องต่อตัวต้านทานพูลอัพ (Rp) เข้ากับแรงดัน +5V ไว้ด้วยเสมอ ดังแสดงในรูปที่ 2.7



รูปที่ 2.7 การเชื่อมต่ออุปกรณ์บนระบบบัส I<sup>2</sup>C ที่ใช้ไฟเลี้ยงไม่เท่ากัน



รูปที่ 2.8 การต่อตัวต้านทานเพื่อป้องกันแรงดันกระชากที่อาจปะปนเข้ามาในไฟเลี้ยงของ อุปกรณ์ในระบบบัส I<sup>2</sup>C

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ในกรณีที่อาจมีแรงดันไฟกระชากขนาดใหญ่ปะปนเข้ามาในบัส I<sup>2</sup>C ที่ขา SDA และ SCL ของอุปกรณ์แต่ละตัวต้องต่อตัวต้านทานอนุกรมกับขา SDA และ SCL เรียกว่า R<sub>s</sub> ก่อนต่อเข้าสู่บัส I<sup>2</sup>C ดังแสดงในรูปที่ 2.8

### 2.3.2 หลักการของบัส I<sup>2</sup>C

บัส I<sup>2</sup>C ประกอบด้วยสายสัญญาณ 2 เส้น ดังที่ได้กล่าวมาแล้วคือ SDA และ SCL อุปกรณ์ที่ต่อพ่วงบนบัสสามารถมีได้มากมาย ดังนั้น จึงต้องมีการกำหนดรูปแบบของการติดต่อบนบัส หรือเรียกว่า โพรโตคอล (Protocol) เพื่อให้ผู้ใช้งานทราบว่า ขณะนี้อุปกรณ์ใดติดต่อกันอยู่ และอุปกรณ์ตัวใดเป็นตัวรับหรือตัวส่ง ต่อไปนี้จะขออธิบายลักษณะ หน้าที่ และนิยามของอุปกรณ์ที่ต่ออยู่บนบัส I<sup>2</sup>C เพื่อเป็นข้อตกลงพื้นฐานก่อนที่จะอธิบายการทำงานของบัส I<sup>2</sup>C ต่อไป

อุปกรณ์ที่เป็นผู้สร้างข้อมูลหรือส่งข้อมูล เรียกว่า ตัวส่ง (Transmitter)

อุปกรณ์ที่เป็นผู้รับข้อมูล เรียกว่า ตัวรับ (Receiver)

ในอุปกรณ์บนบัส I<sup>2</sup>C สามารถเป็นได้ทั้งตัวรับและตัวส่ง บางอุปกรณ์ทำหน้าที่เป็นตัวรับเพียงอย่างเดียว จะไม่มีอุปกรณ์ใดบนบัส I<sup>2</sup>C ที่ทำหน้าที่เป็นตัวส่งเพียงอย่างเดียว

อุปกรณ์ที่ทำหน้าที่ควบคุมจังหวะการทำงานหรือการติดต่อบนบัส I<sup>2</sup>C เรียกว่า มาสเตอร์ (Master)

อุปกรณ์ที่ถูกควบคุมหรืออุปกรณ์ที่ต่อพ่วงเข้าไปบนบัส I<sup>2</sup>C เรียกว่า สเลฟ (Slave)

ข้อกำหนด 2 ประการ สำคัญของการติดต่อบนบัส I<sup>2</sup>C คือ

- 1) การถ่ายข้อมูลจะเกิดขึ้นได้เมื่อมีแสงเท่านั้น
- 2) ในระหว่างการถ่ายทอกข้อมูล เมื่อใดก็ตามที่สาย SCL มีสถานะเป็นลอจิกสูง สายข้อมูลต้องรักษาข้อมูลไว้ อย่าให้เกิดการเปลี่ยนแปลงขึ้นเด็ดขาด มิฉะนั้น สัญญาณที่เกิดขึ้นจะได้รับการแปลความหมายเป็นสัญญาณควบคุมแทน

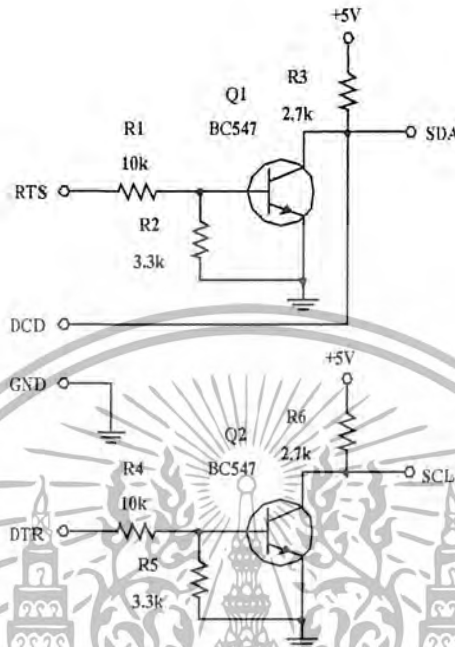
### 2.3.3 วงจรแปลงสัญญาณพอร์ตอนุกรม เพื่อเชื่อมต่อกับระบบบัส I<sup>2</sup>C

เพื่อให้สายสัญญาณของพอร์ตอนุกรมจากคอมพิวเตอร์สามารถติดต่อกับอุปกรณ์ที่มีการเชื่อมต่อแบบบัส I<sup>2</sup>C ต้องมีวงจรเพื่อทำหน้าที่สร้างสายสัญญาณ SDA และ SCL ขึ้น จากสัญญาณของพอร์ตอนุกรม ดังมีตัวอย่างวงจรมารูปที่ 2.9 โดยในการเชื่อมต่อคอมพิวเตอร์ เข้าสู่บัส I<sup>2</sup>C นี้ ต้องกำหนดให้คอมพิวเตอร์เป็นอุปกรณ์มาสเตอร์เท่านั้น

ทรานซิสเตอร์ Q<sub>1</sub> และ Q<sub>2</sub> ได้รับการจัดวงจรให้มีลักษณะเป็นวงจรบัฟเฟอร์แบบคอลเล็กเตอร์เปิดตามข้อกำหนดของวงจรเอาต์พุตของบัส I<sup>2</sup>C โดย Q<sub>1</sub> ใช้สำหรับถ่ายทอกสัญญาณของสาย SDA ในขณะที่ Q<sub>2</sub> ทำหน้าที่ถ่ายทอกสัญญาณ ของสาย SCL ไฟเลี้ยงของวงจรคือ +5V จึงสามารถใช้ทรานซิสเตอร์ชนิดเอ็นพีเอ็นเบอร์มาตรฐานเบอร์ใดก็ได้ที่สามารถตอบสนองความถี่ได้สูงถึง 100 kHz ข้อมูลจากคอมพิวเตอร์จะถ่ายทอกสู่สาย SDA ของ บัส I<sup>2</sup>C ผ่านทางขา RTS และรับข้อมูล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เข้าทางขา DCD ส่วนสายสัญญาณพิก้าที่ออกจากพอร์ตอนุกรมจะถูกส่งออกมาทางขา DTR เพื่อใช้เป็นสาย SCL สำหรับบัส I<sup>2</sup>C



รูปที่ 2.9 วงจรสร้างสายสัญญาณเพื่อเชื่อมต่อกับระบบบัส I<sup>2</sup>C ของพอร์ตอนุกรม

### 2.3.4 การเขียนโปรแกรมติดต่อบัส I<sup>2</sup>C ด้วยวิชาวาลเปติก

จากส่วนของ ฮาร์ดแวร์จะเห็นว่า ขาที่ใช้งานในการติดต่ออุปกรณ์บนระบบบัส I<sup>2</sup>C นั้น ใช้ขาของพอร์ตอนุกรมทั้งหมด 3 ขา คือ DTR, RTS และ DCD ขาทั้งสามนี้สามารถควบคุมด้วยคอนโทรล MSCComm1 ของวิชาวาลเปติก ซึ่งมีรูปแบบการใช้งานดังนี้

การกำหนดให้ขา SDA เป็น “1” ต้องเขียนคำสั่งเป็น MSCComm1.RTSEnable = True

การกำหนดให้ขา SDA เป็น “0” ต้องเขียนคำสั่งเป็น MSCComm1.RTSEnable = False

การกำหนดให้ขา SCL เป็น “1” ต้องเขียนคำสั่งเป็น MSCComm1.DTREnable = True

การกำหนดให้ขา SCL เป็น “0” ต้องเขียนคำสั่งเป็น MSCComm1.DTREnable = False

การอ่านค่าอินพุตของระบบบัส I<sup>2</sup>C นั้น จะใช้คำสั่งเพื่อตรวจสอบค่าจาก CDHolding โดยจะต้องตรวจสอบว่าค่าที่ได้เป็น True หรือ False

## 2.4 หลักในการพัฒนาแอปพลิเคชันด้วย ASP

ASP (Active Server Pages) นับเป็น Interpreter ที่ใช้ในการตีความซอร์สโค้ด ที่ถูกฝังอยู่ในเว็บเพจ หรือ เอกสาร html โดยใช้ไวยากรณ์ หรือ Syntax ของภาษา VBScript (ซึ่ง VBScript ก็อาศัยโครงสร้างของภาษาวิซวลเบสิก) เพื่อสร้างเอกสาร HTML ซึ่งเป็นเว็บเพจผลลัพธ์ขึ้นมา จากนั้นจะทำการส่งไปให้เว็บเซิร์ฟเวอร์ เพื่อที่จะได้ทำการส่งต่อไปยังเบราว์เซอร์ที่ได้ทำการเรียกไฟล์

ASP จะ เป็น ความสามารถ ที่ จะ ต้อง ถูก ติด ตั้ง เข้า ไป ใน เว็บเซิร์ฟเวอร์ เอง นอก จาก จะ เป็น พีเจอร် ของ เว็บเซิร์ฟเวอร์ อื่น นั้น ดั่ง นั้น เว็บเซิร์ฟเวอร์ รุ่ น เก่า จะ ไม่ มี ความ สามารถ อื่น นี้ ใน ปัจจุบัน เรา สามารถ ใช้ เว็บเซิร์ฟเวอร์ ที่ เป็น ผล ติ ก ณ ์ ของ บริษัท ไมโครซอฟท์ อัน ได้แก่ IIS (Internet Information Sever) สำหรับ Windows NT Server, PWS (Personal Web Server) สำหรับ Windows 95/98, Peer Web Server สำหรับ Windows NT Workstation นอกจากนั้น ยังมีผลิตภัณฑ์ที่มาจากบริษัทอื่นๆ อัน ได้แก่ ChillilASP ของ ChillilSoft, FastTrack Servers, LotusDomino , GoServers และอื่นๆ

### 2.4.1 หลักการทำงานของ ASP

การทำงานของ ASP จะทำการอ่านไฟล์หรือเว็บเพจที่ถูกกระบวนนามสกุลเป็น .asp แล้วทำงานตามสคริปต์ที่เขียนเอาไว้ สคริปต์ดังกล่าวจะเรียกว่าเป็น Server Side Script หรือสคริปต์ที่จะถูกเรียกขึ้นมาทำงานที่เครื่องเซิร์ฟเวอร์ ส่วนสคริปต์อีกประเภทเราจะพบเห็นทั่วไป เช่น JavaScript เป็นต้น ที่ซึ่งจะถูกเรียกทำงานจากเบราว์เซอร์เราจะเรียกว่าเป็น Client Side Script ซึ่งการเขียน Slient Side Script จะต้องคำนึงถึง เบราวเซอร์ เป็นหลัก เนื่องจากว่าสคริปต์จะต้องทำงานที่เบราว์เซอร์นั้นคือหากเบราว์เซอร์ไม่มีความสามารถรองรับสคริปต์นั้นได้ ก็จะไม่สามารถทำงานได้ เช่นการเขียน VBScript แบบ Client Side Script ก็ไม่สามารถนำมาใช้งานที่เบราว์เซอร์ที่เป็น Netscape ได้ เป็นต้น ทำนองเดียวกันตัว Server Side Script ก็จะต้องเป็นไปตามความต้องการของตัว Interpreter เช่น ASP มีความสามารถในการใช้สคริปต์ได้ 2 ภาษา คือ VBScript และ Jscript

### 2.4.2 หลักในการเขียน ASP

การเขียน ASP Script จะทำ โดยการฝัง หรือ Embedded ส่วนที่เป็น สคริปต์ ลงไป ในเว็บเพจ กล่าวคือ หาก ไม่มีการฝัง ASP Script เลย เว็บเพจนั้น ก็จะเป็น เอกสารเว็บเพจธรรมดาทั่วไป นั่นเอง การตีความโดย ASP ก็ จะ ทำ การ ตี ความ ได้ ลง ไป จาก ดั น ไฟล์ ไป ที ละ บรรทัด บรรทัดใดมีส่วนของ ASP Script อยู่ จะ ทำ การ ตี ความ ก่อน แล้ว จึง ค่อย ทยอย ส่ง ผล ติ ก ณ ์ ออก มา เรื่อย ๆ หาก เกิด ข้อ ผิด พลาด ที่ รุนแรง ก็ จะ หยุด การ ทำงาน ไว้ ส่วนที่เป็น ASP Script จะ ขึ้น ดั น หรือ เปิด ด้วย Tag โดยใช้ เครื่องหมายเปิดด้วย <% และ ปิดท้ายด้วย %>

ยกตัวอย่างการเขียน ASP Script เช่นหากเราต้องการแสดงวัน – เวลาปัจจุบัน ก็จะใช้ฟังก์ชัน Now ก็จะเขียนได้ดังนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

วัน – เวลา ขณะนี้ คือ

```
<%=now%>
```

โดยที่บรรทัดที่ 1 ก็จะแสดงข้อความตามปกติ ด้วยภาษา HTML แต่บรรทัดที่ 2 จะมีการนำค่า วันและเวลาปัจจุบัน ด้วย ASP การจะเขียน ASP Script เพื่อทำงานอะไร ก็ต้องศึกษารูปแบบคำสั่งของภาษา VBScript หรือ Jscript และต้องมีพื้นฐานของการเขียนภาษา HTML ด้วย เพราะจะต้องใช้ร่วมกัน ดังตัวอย่างที่ผ่านมา หากจะให้มีการแสดงเป็นตัวเข้ม ขนาดเท่ากับ 2 และแสดงข้อความเป็นสีแดง ทั้งหมดที่ว่ามาจะเป็นกำหนดที่ ภาษา HTML ดังนี้ ดังนั้นเราพอจะเข้าใจได้ว่า การทำงานที่เป็นลักษณะ Process จะทำงานบน ASP แต่สำหรับการกำหนดส่วนแสดงผล ต่างๆ นั้น จะต้องเขียนด้วยภาษา HTML ตามปกติ

```
<b><font size = 2 color = "ff0000">วัน-เวลา ขณะนี้ คือ
<%=now%></font></b>
```

### 2.4.3 การใช้โปรแกรมพัฒนาและเขียน ASP

การเขียนเว็บเพจให้เป็น .asp นั้น ไม่มีความจำเป็นต้องใช้โปรแกรมอะไรเป็นพิเศษ แต่ก็มีแอปพลิเคชันมากมายที่ถูกสร้างไว้สำหรับช่วยให้การออกแบบ ASP ทำงานได้ง่ายขึ้น อันได้แก่ MS Visual Interdev, MS Visual Basics เป็นต้น (บนแอปพลิเคชันเหล่านี้ จะมีลักษณะการใช้งานสำหรับออกแบบ ไฟล์ ASP เป็นคู่ที่ถูกออกแบบและสร้างขึ้นพร้อมกัน อีกทั้งมี Object Server Side เตรียมไว้ให้สำหรับการออกแบบด้วยการคลิกและลาก (ไม่ก่ครั้ง) เพราะไฟล์ ASP นั้นมีลักษณะเป็นไฟล์ข้อความธรรมดาทั่วไป สามารถใช้โปรแกรมเขียนไฟล์ข้อความทั่วไป เช่น Notepad หรือ Edit Plus ในการเขียนได้ สำหรับการเขียน .asp ใดๆ ก็ตาม การบันทึก ให้บันทึกไฟล์เป็นนามสกุล .asp เช่น "First.asp" เป็นต้น

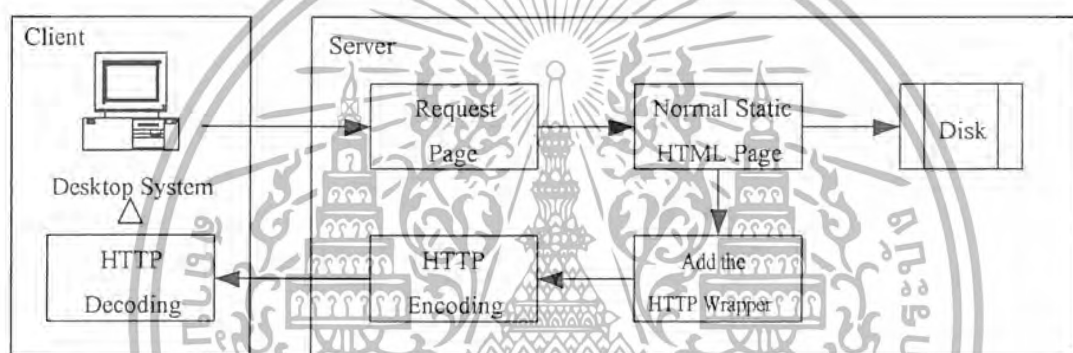
### 2.4.4 ความสามารถของ ASP

เว็บเซิร์ฟเวอร์เมื่อไม่มีความสามารถทางด้าน ASP ก็จะเป็นเว็บเซิร์ฟเวอร์ที่ให้บริการผู้ใช้กับไฟล์ HTML ธรรมดาเมื่อเบราว์เซอร์ ทำการร้องขอหน้าเอกสาร นั้นๆ เว็บเซิร์ฟเวอร์ จะทำการดึงข้อมูลหน้าเอกสาร (Page) ที่ต้องการจากฐานข้อมูล ไฟล์ที่เก็บไว้หรือจากหน่วยความจำ แล้วทำการส่งไปให้เบราว์เซอร์ของผู้ใช้ ดังนั้นเว็บเซิร์ฟเวอร์จึงมีหน้าที่หลักเพียงเป็นตัวกลางระหว่างเบราว์เซอร์และไฟล์ที่อยู่ในดิสก์บนเครื่องเซิร์ฟเวอร์เท่านั้น สคริปต์ต่างๆที่ฝังอยู่ในเอกสาร HTML จะทำการประมวลผลที่ฝั่ง Client ด้วยตัวเอง

สรุปขั้นตอนการทำงานของเว็บเซิร์ฟเวอร์ กับไฟล์ HTML ได้ดังนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- 1) ผู้ใช้ใส่ Internet Address ของไฟล์ HTML ที่แอดเดรสบาร์ของเว็บเบราว์เซอร์ และกด Enter เพื่อร้องขอหน้าเอกสาร (page) ที่ต้องการ ตัวอย่างเช่น <http://161.246.48.109/index.htm>
- 2) ที่เบราว์เซอร์ จะส่งคำร้องขอสำหรับเอกสารนั้นๆ ให้แก่เว็บเซิร์ฟเวอร์
- 3) ที่เว็บเซิร์ฟเวอร์ เมื่อได้รับการร้องขอเอกสาร และตรวจชนิดของเอกสารที่ร้องขอเป็นไฟล์ชนิดใด ในกรณีนี้ ไฟล์ที่ร้องขอมาจะมีส่วนขยาย (นามสกุลไฟล์) เป็น .htm หรือ .html
- 4) ที่เว็บเซิร์ฟเวอร์ จะทำการตรวจสอบระบบต่างๆ และค้นหาไฟล์ดังกล่าว จากไดเรกทอรีที่กำหนดไปสู่ที่สำรองไฟล์ไว้ หรือ หน่วยความจำ แล้วทำการส่งไปยังเบราว์เซอร์ที่ร้องขอมา
- 5) ตัวไฟล์ HTML ที่ส่งมาจะทำการ ประมวลผล โดยเบราว์เซอร์ของบุคคลที่ร้องขอหน้าเอกสารนั่นเอง แล้วแสดงผลบนหน้าจอให้เห็นบนเบราว์เซอร์



รูปที่ 2.10 ผังการทำงานของเว็บเซิร์ฟเวอร์ กับไฟล์ชนิด HTML

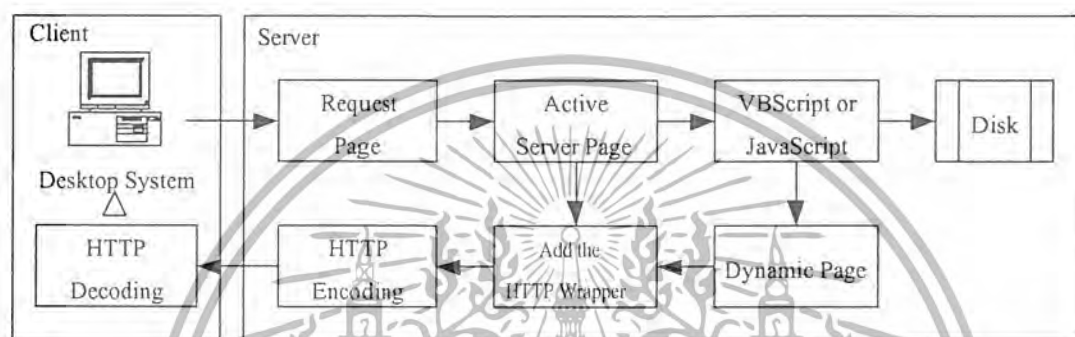
แต่สำหรับไฟล์ที่เป็นชนิด Active Server Pages นอกจากที่เว็บเซิร์ฟเวอร์ สามารถบริการเอกสาร ที่เป็นไฟล์ HTML ปกติได้แล้ว จะสามารถทำการ ประมวลผล ไฟล์ชนิด ASP ได้อีกด้วย โดยส่วนที่อยู่ใน ไฟล์ ASP ที่เขียนด้วยภาษา HTML ก็จะทำงานปกติ ส่วนที่เป็น ASP ก็จะนำซอร์สโค้ดมาทำการประมวลผล

สรุปขั้นตอนการทำงานของเว็บเซิร์ฟเวอร์ สำหรับไฟล์ .asp ได้ดังนี้

- 1) ผู้ใช้ใส่ Internet Address ของไฟล์ ASP ที่ แอดเดรสบาร์ของเว็บเบราว์เซอร์ และกด Enter เพื่อร้องขอหน้าเอกสาร (Page) ที่ต้องการ ตัวอย่างเช่น <http://161.246.48.109/main.asp>
- 2) ตัวเบราว์เซอร์ จะส่งคำร้องขอหน้าเอกสาร Active Server Pages ไปยังเว็บเซิร์ฟเวอร์
- 3) ตัวเว็บเซิร์ฟเวอร์ จะรับคำร้องขอ และจดจำได้ว่าเป็นการร้องขอไฟล์ ASP เพราะว่าเป็นการร้องขอไฟล์ที่มี ส่วนขยาย (นามสกุลไฟล์) เป็น .asp
- 4) ตัวเว็บเซิร์ฟเวอร์ จะทำการเรียกไฟล์ Active Server Pages ที่เหมาะสมจากไดเรกทอรีหลักหรือจากหน่วยความจำ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่นิยมนำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- 5) เว็บเซิร์ฟเวอร์ จะทำการส่งไฟล์นั้นไปยังแอปพลิเคชัน ActiveX DLL ที่มีชื่อว่า ASP.dll
- 6) ไฟล์ Active Server Pages คำสั่งข้างใจจะถูกอ่านจาก ส่วนบนลงในส่วนล่าง (Top to Bottom) ทีละคำสั่ง เพื่อประมวลผล โดยผลลัพธ์ จะถูกส่งออกไปในรูปของไฟล์ HTML
- 7) ไฟล์ HTML ที่ได้จะถูกส่งกลับไปยังบราวเซอร์ ที่ร้องขอ
- 8) ไฟล์ HTML จะถูกประมวลผลโดยบราวเซอร์ของผู้ร้องขอหน้าเอกสารนั้น และแสดงผลที่บนหน้าจอ



รูปที่ 2.11 ฟังก์ชันการทำงานของเว็บเซิร์ฟเวอร์ กับไฟล์ชนิด ASP

ถ้าสำหรับมุมมองของเว็บเซิร์ฟเวอร์ แล้ว Active Server Pages จะมีความแตกต่างกับหน้าเอกสาร ปกติอย่างมากเนื่องจากเอกสารปกติ (Normal HTML Page, Static HTML Page) จะถูกส่งให้บราวเซอร์โดยไม่มีการประมวลผลใดๆ เลย แต่สำหรับ Active Server Pages แล้ว ซอร์สโคดที่บรรทัดคำสั่ง (Command) ใน Active Server Pages จะต้องถูกประมวลผล (Execute) เพื่อที่จะสร้างผลลัพธ์ที่เป็นเอกสาร HTML ซึ่งเป็นสิ่งที่ทำให้ ASP ทำงานได้เหมือน แอปพลิเคชันบนอินเทอร์เน็ต

ส่วนมุมมองของบราวเซอร์กล่าวได้ว่า ASP นั้นจะเหมือนกับหน้าเอกสารธรรมดา มีสิ่งเดียวที่แตกต่างกันก็คือ Active Server Pages จะมีส่วนนามสกุลเป็น .asp แทนที่จะเป็น .htm หรือ .html ที่เป็นเช่นนี้เพราะว่า เมื่อมีการร้องขอ หน้าเอกสารที่เป็น ASP จะทำการประมวลผล และผลลัพธ์ ที่ส่งกลับคืนมานั้นก็คือ เอกสารที่เป็นไฟล์ HTML ธรรมดา การทำงานในลักษณะเช่นนี้ทำให้ ASP สามารถนำไปใช้ได้กับบราวเซอร์ทุกชนิดที่มีอยู่ในปัจจุบัน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## 2.5 มัลติมีเดียบนอินเทอร์เน็ต

เรื่องนี้จะได้อธิบายรายละเอียดทางเทคนิคว่าระบบมัลติมีเดียบนอินเทอร์เน็ตถูกนำมาใช้งานบนอินเทอร์เน็ตได้อย่างไร โดยมีจุดมุ่งหมายเพื่ออธิบายมาตรฐานและผลิตภัณฑ์ที่เป็นสิทธิเฉพาะของผู้ผลิตบางราย รวมทั้งเทคนิคที่ใช้

ตารางที่ 2.4 แอปพลิเคชันมัลติมีเดียและแบนด์วิดท์ที่ต้องการ

แอปพลิเคชัน	แบนด์วิดท์ที่ต้องการ (โดยไม่มีกำบัง)	เทคโนโลยีการส่งข้อมูลที่เป็นไปได้
การสนทนาทางโทรศัพท์	64 Kbps	ISDN (Intergrated Service Digital Network)
ระบบเสียงคุณภาพเทียบเท่าซีดี เล็กน้อย	มากกว่า 700 Kbps	Ethernet, xDSL (Digital Subscriber Line)
โทรทัศน์ความคมชัดสูง (HDTV : High Definition Television )	2 Gbps	ATM ( Asynchronous Transfer Mode )

จากแบนด์วิดท์จำกัดของอินเทอร์เน็ต (โดยเฉพาะเมื่อถูกแบ่งย่อยออกมาเป็นของผู้ใช้แต่ละคน) การบีบอัดข้อมูลจึงเข้ามามีบทบาทความสำคัญมาก วิธีการขนส่งข้อมูลมัลติมีเดียบนอินเทอร์เน็ต ทำได้วิธีเดียวคือต้องอาศัยการบีบอัดข้อมูลนี้ แต่เพียงเทคนิคการบีบอัดข้อมูลแบบใดแบบหนึ่งเท่านั้นยังไม่พอ ยังมีความต้องการอื่นๆ อีกซึ่งจะได้อธิบายไว้ในที่นี้โดยย่อ

อัลกอริทึมบีบอัดข้อมูลต้องดำเนินงานในแบบเรียลไทม์ (Real-Time) เพื่อให้สามารถประยุกต์ใช้กับงานอย่างเช่น การประชุมระยะไกล (Video Teleconference) และงานถ่ายทอดสด เช่น ข่าวหรือการแข่งขันกีฬา เป็นต้น ทั้งการบีบอัดและขยายข้อมูลจะต้องสามารถทำได้ในทันที นั่นคือเวลาที่ใช้บีบอัด และขยายข้อมูลจะต้องสั้นมาก (อาจมีกรณีพิเศษ เช่น การกระจายข่าวหรือภาพยนตร์ ซึ่งได้บันทึกและบีบอัดไว้แล้วล่วงหน้า โดยใช้อัลกอริทึมบีบอัดข้อมูลที่ต้องใช้เวลาของ CPU อย่างมากแต่ว่าการขยายข้อมูลออกมาจะทำได้ง่ายมาก)

อัลกอริทึมบีบอัดข้อมูลต้องมีความคงทนต่อความผิดพลาด เนื่องจากอาจเกิดการสูญหายของแพ็กเก็ตของข้อมูลบนอินเทอร์เน็ตได้ อัลกอริทึมบีบอัดข้อมูลต้องสามารถทำงานได้ ไม่ว่าแพ็กเก็ตข้อมูลที่ได้รับมาก่อนจะมาถึงหลังสลับกัน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

อัลกอริทึมต้องมีความคงทนต่อความผิดพลาด เนื่องจากอาจเกิดการสูญหายของแพ็กเก็ตของข้อมูลบนอินเทอร์เน็ตได้ อัลกอริทึมบีบอัดข้อมูลต้องสามารถทำงานได้ ไม่ว่าแพ็กเก็ตข้อมูลที่ได้รับจะมาก่อนหรือมาหลังสลับกัน

อัลกอริทึมต้องสามารถรองรับการ “สูญหาย” ได้ ในบางอัลกอริทึมสามารถละทิ้งข้อมูลบางส่วนที่ไม่สำคัญทิ้งไป ทำให้ข้อมูลที่ได้ขยายออกมาไม่เหมือนข้อมูลต้นฉบับอย่างสมบูรณ์แบบแม้ว่าการสูญเสียข้อมูลบางส่วนเป็นสิ่งที่พอยอมรับได้แต่ก็ต้องเป็นไปอย่างจำกัดในกรณีที่ส่งข้อมูลวิธีนี้ออกมาแล้วสูญเสียข้อมูลมากเกินไป จะส่งผลให้ได้รับภาพที่มีคุณภาพต่ำมาก อัลกอริทึมที่ใช้บีบอัดและขยายข้อมูลนี้มักจะเรียกกันว่า Codec (Compression /Decompression)

เพื่อแสดงให้เห็นประสิทธิภาพของ Codec ในตารางที่ 2.5 จะแสดงแบนด์วิดท์ที่ต้องการเมื่อใช้ Codec กับแอปพลิเคชันเดียวกับที่ได้แสดงไว้ในตารางที่ 2.4

ตารางที่ 2.5 แอปพลิเคชันทางด้านมัลติมีเดียและแบนด์วิดท์ที่ต้องการ

แอปพลิเคชัน	แบนด์วิดท์ที่ต้องการ (โดยไม่มีบีบอัด)	เทคโนโลยีการส่งข้อมูลที่เป็นไปได้
การสนทนาทางโทรศัพท์	น้อยกว่า 5 Kbps	โมเด็ม
ระบบเสียงคุณภาพเทียบเท่าซีดี	64 Kbps	ISDN
โทรทัศน์ความคมชัดสูง (HDTV : High Definition Television)	30 Mbps	FDDI ( Fiber Distributed Data Interface ), ATM

Codec สามารถสร้างขึ้นเป็นซอฟต์แวร์หรือฮาร์ดแวร์ก็ได้ โดยฮาร์ดแวร์จะทำงานได้เร็วกว่า และไม่สร้างภาระให้ CPU มากนัก แต่ข้อด้อยคือมีราคาแพงและการปรับเปลี่ยนใดๆ ทำได้ยาก แต่สำหรับซอฟต์แวร์แล้วจะมีข้อดีคือปรับเปลี่ยนได้ง่าย และสามารถใช้ Codec ได้หลายแบบอีกด้วย นอกจากนี้ซอฟต์แวร์ยังมีค่าใช้จ่ายที่ต่ำกว่า แต่ก็ทำให้เกิดข้อด้อยตามมาคือความล่าช้า และเป็นภาระให้กับ CPU ด้วย

### 2.5.1 การบีบอัดสัญญาณภาพ (Streaming)

เทคนิค Streaming เป็นเทคนิคที่สำคัญอย่างหนึ่งซึ่งทำให้สามารถใช้มัลติมีเดียบนอินเทอร์เน็ตได้ เว็บจะเริ่มต้นทำงานเหมือนกับเครื่องมือแบบ TeX-Mode ซึ่งไฟล์ต่างๆ จะถูกดาวน์โหลดจนครบทั้งไฟล์ก่อนจึงสามารถนำมาแสดงผลได้ การทำงานเช่นนี้ก็สมเหตุผลสำหรับไฟล์ที่มีขนาดเล็กไม่ใหญ่มากนัก แต่ไฟล์มัลติมีเดียมักจะมีขนาดใหญ่มาก เทคนิค Streaming จึงอนุญาตให้ออกสารนี้เป็นเอกสารที่ส่งวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไมอนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โคลเอนต์ดาวน์โหลดข้อมูลเพียงบางส่วนของไฟล์ แล้วนำข้อมูลที่ได้ออกมาแสดง (ที่เป็นภาพ/เสียง) ออกมาได้ทันที ไม่ต้องรอให้ดาวน์โหลดจนครบทั้งไฟล์ก่อน โดยข้อมูลจะถูกนำไปเก็บไว้ในบัฟเฟอร์ (Buffer) ก่อนนำมาแสดงผล ส่วนข้อมูลที่เหลือก็จะดาวน์โหลดต่อไปเรื่อยๆ พร้อมกับแสดงผลออกมาด้วย ข้อมูลในส่วนที่ทำการแสดงผลนั้นอาจล่าช้าไปบ้าง เพราะในขณะนั้น โคลเอนต์ยังไม่มีข้อมูลครบทั้งไฟล์

### 2.5.2 พื้นฐานการบีบอัดสัญญาณภาพ

การส่งข้อมูลมัลติมีเดียประกอบไปด้วยการส่งข้อมูลและการแสดงชุดของภาพ เมื่อภาพไม่ได้เป็น “ภาพเดี่ยว” แต่เป็นส่วนหนึ่งของภาพยนตร์ ภาพแต่ละภาพเหล่านี้จะเรียกว่า เฟรม (Frame) การบีบอัดสัญญาณภาพวิดีโอเป็นเรื่องจำเป็น อย่างน้อยก็เพื่อลดจำนวนของข้อมูลวิดีโอให้อยู่ในขนาดที่สามารถจัดการได้ การบีบอัดทางเทคนิคสามารถบีบอัดข้อมูลในเฟรมเดียว โดยไม่ต้องอ้างอิงถึงเฟรมอื่นๆ ซึ่งเรียกว่า Intraframe Coding การบีบอัดข้อมูลวิดีโอที่อาศัยการลัดเลาะข้ามเฟรมในภาพวิดีโอ ซึ่งการข้ามเฟรมสามารถเกิดขึ้นได้ในสามแบบคือ

1) Spatial Redundancy พิกเซล (Pixel) ต่างๆ ภายในจะมีความสัมพันธ์ในเชิงระยะซึ่งกันและกัน ตัวอย่างเช่น พิกเซลหนึ่งอยู่ติดกับอีกพิกเซลหนึ่ง การข้ามเฟรมในเชิงระยะนี้เกิดขึ้นเมื่อพิกเซลที่อยู่ใกล้กันมีสีเหมือนกัน และบางครั้งอาจมีค่าความเข้มของสีเท่ากันอีกด้วยสิ่งนี้เกิดขึ้นค่อนข้างบ่อย แต่ด้วย Spatial Redundancy จะสามารถทำให้ลดลงโดยใช้เทคนิคแบบ Intraframe Coding ที่มีชื่อว่า DCT (Discrete Cosine Transform) ซึ่งจะได้อธิบายต่อไป

2) Spectral Redundancy ประกอบด้วยความเข้มของแสงและสี ซึ่ง Spectral Redundancy ในเฟรมภาพยนตร์จะเกิดขึ้นจากพิกเซลที่มีแนวโน้มจะมีความสว่างในทุกๆ สี แทนที่จะเป็นความสว่างที่เกิดขึ้นกับสีใดสีหนึ่งเป็นการเฉพาะ นอกจากนี้สายตาของคนทั่วไปนั้นจะไวกับความสว่างมากกว่าสี หมายความว่าสายตาของเราจะสังเกตเห็นความสว่างที่ผิดปกติไปได้ดีกว่าสีที่ผิดเพี้ยนไป

3) Temporal Redundancy ภาพยนตร์จะแสดงภาพด้วยอัตราความเร็วประมาณ 30 เฟรมต่อวินาที แม้แต่ในภาพยนตร์แบบ “แอ็คชั่น” บางส่วนของเฟรมเท่านั้นจึงจะเปลี่ยนแปลงไปจากเฟรมก่อนหน้า เมื่อแสดงเฟรมด้วยความเร็วนี้แล้วสุมตัวอย่างขึ้นมา เทคนิคหนึ่งที่ใช้ในการแก้ไขปัญหาคือ Temporal redundancy คือ Motion Vector Encoding ซึ่งมีแนวคิดที่จะแบ่งพื้นที่ในเฟรมออกเป็นบล็อกย่อย สามารถกล่าวได้ว่าในการทำงานแบบนี้ “ทุกสิ่งจะเหมือนกับเฟรมก่อนหน้า ยกเว้นมีการเคลื่อนไหวเกิดขึ้นในบล็อกใดๆ” (ตัวอย่างเช่น ในภาพนิ่งที่มีการกระพริบตา จะมีการเปลี่ยนแปลงเฉพาะในบล็อกที่แสดงภาพตาเท่านั้น)

เฟรมแบบง่ายๆ ทั่วไปที่แสดงภาพอันสมบูรณ์จะไม่ได้รับประโยชน์จาก Temporal Redundancy ในการจัดการกับความซ้ำซ้อนแบบต่างๆ จึงได้มีการกำหนดการใช้เฟรมสามแบบ เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไมอนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

แบบแรกเป็นเฟรมต้นฉบับที่แสดงภาพอันสมบูรณ์ซึ่งเรียกว่า I-frame หรือ Intraframe เพื่อสื่อว่าเฟรมแบบนี้สามารถใช้แสดงผลได้โดยไม่ต้องอ้างอิงถึงเฟรมอื่นใด เฟรมแบบที่สองคือ P-frame หรือ Predicted Frame เฟรมแบบนี้จะสร้างขึ้นได้โดยอาศัย B-frame ที่อยู่ก่อนหน้า เวกเตอร์ที่เคลื่อนไหว และภาพที่ผิดพลาดที่ใช้ปรับแก้ความผิดพลาดในเวกเตอร์การเคลื่อนไหว ส่วนเฟรมประเภทสุดท้ายคือ B-frame หรือ Bi-directional Predicted Frame โดยจะเป็นเฟรมที่ประกอบด้วยเวกเตอร์ภาพเคลื่อนไหว แบบเดินหน้า, เวกเตอร์ภาพเคลื่อนไหว แบบถอยหลัง และข้อผิดพลาด เฟรมแบบ B-frame จะถูกใช้เมื่อ P-frame มีวัตถุภาพเกิดขึ้นใหม่

เทคนิค Discrete Cosine Transform การบีบอัดสามารถใช้สมการแทนที่ข้อมูลแทนที่จะเป็นแบบสุ่มตัวอย่าง (Sampling) อย่างซ้ำๆ แล้วส่งค่าจริงออกไป ให้พิจารณากรณีที่ท่านต้องวาดเส้นตรง เริ่มจากวิชาพีชคณิตที่สอนไว้ว่าเราสามารถจะวาดเส้นตรงได้โดยใช้สมการ  $y = mx + c$  นั้นหมายความว่าสิ่งที่เราต้องการก็คือค่าของ  $x$  และ  $c$  เพียงสองค่าเท่านั้น

เห็นได้ชัดว่าการบีบอัดนั้นไม่ยาก เติล็ดล็บอยู่ที่การหาค่าสมการที่ใช้แสดงแทนข้อมูลที่สนใจ และปกติสมการจะมีความซับซ้อนมากกว่าสมการเส้นตรง เทคนิคหนึ่งที่น่าสนใจในการบีบอัดข้อมูลก็คือ DCT (Discrete Cosine Transform) โดยพยายามที่จะแสดงข้อมูลออกมาในชุดของ Cosine Wave และ Cosine Wave บางส่วนเท่านั้น (ที่สอดคล้องกับความถี่หลัก) ซึ่งจัดเก็บไว้ ส่วนที่ทิ้งไปก็ทำให้เกิดการสูญเสียของข้อมูล

DCT นั้นใช้ในมาตรฐาน MPEG (Moving Picture Experts Group) ซึ่งจะได้อธิบายต่อไป โดยถือว่า DCT เป็นเทคนิคแบบ Intraframe Coding เนื่องจากการทำงานจะดำเนินไปครั้งละเฟรมเดียวเท่านั้น

### 2.5.3 มาตรฐานวิดีโอ

มาตรฐานในการเข้ารหัสวิดีโอ และ Video Conferencing มีอยู่หลายแบบ โดยจะอธิบายถึงมาตรฐานสำคัญๆ โดยย่อ ดังนี้

1) MPEG (Moving Picture Experts Group) เป็นชื่ออย่างไม่เป็นทางการที่ตั้งให้กับคณะทำงานที่ได้กำหนดมาตรฐานมัลติมีเดีย ชื่อที่ตั้งอย่างเป็นทางการคือ ISO/IEC JTC1 SC29 WG11 ซึ่งย่อมาจาก "International Organization for Standardization/International Electro-Technical Commission, Joint Technical Committee 1, Subcommittee 29, Work Group 11" คณะทำงานของ MPEG นั้นจะมีการประชุมปีละครั้งเพื่อวางแผนกิจกรรมการทำงานและการดำเนินการ

MPEG มีมาตรฐานที่ได้นำมาใช้งานแล้วสามมาตรฐาน ซึ่งที่รู้จักกันในชื่อ MPEG-1, MPEG-2 และ MPEG-3 ปัจจุบันคณะทำงานกำลังในระหว่างการพัฒนามาตรฐาน MPEG-4 และ MPEG-7

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2) MPEG-1 ถูกใช้กันเป็นหลักในการบีบอัด ภาพวิดีโอแบบซีดีรอมมีรายละเอียดอธิบาย อยู่ใน ISO/IEC Standard 11172 -2

MPEG-1 อาศัยพื้นฐานจากหลักการของความซ้ำซ้อนแบบ Spatial, Spectral และ Temporal ดังที่ได้อธิบายไว้ก่อนหน้านี้ พื้นที่ของเฟรมจะถูกแบ่งเป็นส่วนย่อยที่เรียกว่า Macroblock ขนาด  $16 \times 16$  พิกเซล ข้อมูลที่ไม่ได้ถูกบีบอัดของแต่ละ Macroblock ประกอบด้วย 16 บล็อก โดยแต่ละกลุ่มขนาด 4 บล็อกใช้เก็บข้อมูลของสีเขียว, สีแดง, สีน้ำเงิน และความสว่างของแต่ละพิกเซลเรียงต่อกัน โดย MPEG-1 จะลด 16 บล็อกนี้ไปเป็น 4 บล็อกโดย 2 บล็อกเพื่อแทนค่าความสว่าง(Chrominance) และแต่ละบล็อกที่เหลือเก็บสีแดงและสีน้ำเงิน การบีบอัดข้อมูล MPEG-1 ใช้ DCT โดยใช้เวกเตอร์ การเคลื่อนไหว และเฟรมทั้ง 3 แบบคือ I-frame, P-frame และ B-frame

ข้อมูลของ MPEG-1 จะใช้การคำนวณในช่วงบีบอัดข้อมูลสูงกว่าในช่วงขยายข้อมูลอย่างมาก อัตราการบีบอัดข้อมูลของ MPEG-1 ปกติจะอยู่ในช่วง 50:1 แม้ว่าในทางทฤษฎีจะเป็นไปได้ที่จะทำได้ถึง 200:1 ก็ตาม ในปัจจุบัน MPEG-1 ได้รับความนิยมมาก แม้ว่าต่อมาคณะกรรมการ MPEG ได้กำหนด MPEG-2 ขึ้นมาและอยู่ในระหว่างการพัฒนามาตรฐาน MPEG ตัวใหม่ขึ้นมาอีกสาเหตุหนึ่งนั้นอาจเป็นเพราะ MPEG-2 ไม่ได้รับความนิยมมากเท่าที่คาดหวังไว้ อีกเหตุผลหนึ่งคือ MPEG-1 ก้าวล้ำหน้าความสามารถในการคำนวณของเครื่องคอมพิวเตอร์อยู่เล็กน้อย จำต้องใช้เครื่องในยุคถัดมาจึงจะสามารถรองรับความต้องการของ MPEG-1 ได้ดี

3) MPEG-2 ได้นำ MPEG-1 ในส่วนที่เหมาะสมมาใช้กับโทรทัศน์แบบดิจิทัล โดยถือว่า MPEG-2 เป็น Superset ของ MPEG-1 ในลักษณะเด่นที่เห็นชัดเจนของ MPEG-2 คือสนับสนุนเฟรมแบบ Interlaced (เฟรมที่แสดงภาพเพียงครึ่งเดียว) สนับสนุนกรรมวิธีการผลิตสัญญาณวิดีโอและอัลกอริทึมบีบอัดพื้นที่ส่วนที่มีความสว่างและแปลงแสงออกมาแบบต่างๆ

การสนับสนุนมาตรฐาน MPEG-2 ยังไม่แน่ชัดนัก แต่อาจมีการเปลี่ยนไปในภายหลัง เนื่องจากพลังในการคำนวณที่สูงขึ้นมาก อย่างเช่น การพัฒนาของ MMX ในชิปของ Intel

4) MPEG-3 มีจุดมุ่งหมายในการปรับปรุง MPEG-2 จากเดิมที่เป็นมาตรฐานบีบอัดข้อมูลสำหรับโทรทัศน์แบบดิจิทัลให้สามารถนำไปใช้ได้กับโทรทัศน์ความคมชัดสูง (HDTV) ต่อมาก็พบว่าหน้าที่ต่างๆ ที่กำหนดไว้ใน MPEG-3 นั้นสามารถดำเนินการได้โดยใช้ MPEG-2 ดังนั้น MPEG-3 จึงกลายเป็นเพียงเชิงอรรถหนึ่งในประวัติศาสตร์ของเทคโนโลยีมัลติมีเดีย

5) MPEG-4 การทำงานยังคงดำเนินต่อไปสำหรับ MPEG-4 ที่อาจจะกลายเป็นมาตรฐานสากลในช่วงปลายปี 1998 ปัจจุบันข้อเสนอในคุณลักษณะของ MPEG-4 ประกอบไปด้วยข้อเสนอสนับสนุนการสื่อสารข้อมูลวิดีโอแบบสองทาง ข้อเสนอสนับสนุนในการดำเนินการสำหรับสายที่มี

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

แบนด์วิดท์ต่ำ และสนับสนุนให้ผู้ใช้สามารถโต้ตอบเพื่อเลือกเนื้อหาบางส่วนบางตอน และข้ามบางตอนไปได้

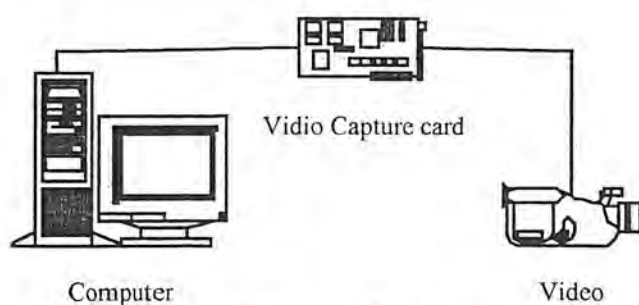
6) MPEG-7 ยังไม่พบคำอธิบายว่าเกิดอะไรขึ้นกับ MPEG-5 และ MPEG-6 ได้มีการพัฒนา MPEG-7 กันต่อไป โดยเป้าหมายอย่างหนึ่งคือ ต้องการสร้างมาตรฐานสำหรับ Search Engine สำหรับงานมัลติมีเดีย

#### 2.5.4 Windows Media Player MPEG-4

การแสดงภาพวิดีโอ หรือภาพเคลื่อนไหวพร้อมเสียงบรรยายประกอบในโฮมเพจด้วยบราวเซอร์ ในรูปแบบของ HTML ผ่านเครือข่ายอินเทอร์เน็ต แบบแสดงรายการสดในลักษณะ Real Time หรือ Live เราจะเรียกว่าเป็น Streaming Files ที่จะส่งจากเว็บเซิร์ฟเวอร์ ผ่านเครือข่ายอินเทอร์เน็ตสู่บราวเซอร์ของผู้ใช้งาน ในความเป็นจริงรูปแบบของ Streamig Files มีอยู่ด้วยกันหลายรูปแบบ เช่น Real Player หรือ Windows Media Player

โปรแกรม Windows Media Player เป็น โปรแกรมของบริษัทไมโครซอฟท์ที่มีติดตั้งมาพร้อมกับ OS Windows ตั้งแต่ Version Windows 98 ขึ้นไป เป็นโปรแกรมที่สามารถแสดงข้อมูล Multimedia ในรูปของวิดีโอ หรือข้อมูลภาพนิ่งพร้อมสัญญาณเสียง ที่มีการบีบอัดข้อมูลในลักษณะ MPEG-4 สามารถเลือกกำหนดอัตราความเร็วของการส่งข้อมูลจากเว็บเซิร์ฟเวอร์ ได้หลายอัตราความเร็ว เช่นตั้งแต่ 28.8 Kb/s จนถึง 100 Mb/s จึงทำให้สามารถเข้าถึงข้อมูล Multimedia ในเว็บเซิร์ฟเวอร์ ได้ตามความต้องการความเร็วของผู้ใช้

ดังนั้นหากเราคิดแปลงวิธีการควบคุมภาพนิ่ง เสียงบรรยายประกอบ และวิดีโอ ในลักษณะของ Streaming Files โดยให้ปรากฏผ่านโฮมเพจด้วยบราวเซอร์ ของ IE และใช้คำสั่ง Java ให้ผู้ใช้ที่เข้าผ่านอินเทอร์เน็ตสามารถควบคุมภาพที่ปรากฏ ตลอดจนทำให้เกิดการตอบโต้ระหว่างผู้ใช้กับโฮมเพจได้แล้ว เราจะสามารถออกแบบและสร้างโฮมเพจที่ใช้ประกอบการเรียนการสอน การบรรยาย ตลอดจนเป็นระบบ Web Based Instruction สำหรับการพัฒนาบทเรียนผ่านเครือข่ายอินเทอร์เน็ต หรือจะใช้เป็นลักษณะแผ่น CD-ROM สำหรับเรียนแบบ Off-Line ก็ได้



รูปที่ 2.12 การเชื่อมต่อสัญญาณวิดีโอจากกล้องถ่ายภาพกับการ์ดวิดีโอ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## 2.6 ActiveX

ActiveX กำหนดกรอบการทำงานในการเพิ่มความสามารถให้ทั้งเว็บเซิร์ฟเวอร์และไคลเอนต์ในแบบไดนามิก ซึ่งที่จริงแล้ว โปรแกรม IE เวอร์ชันใหม่ๆ ต่างพึ่งพาความสามารถของ ActiveX Control เพิ่มขึ้นเรื่อยๆ ตัวอย่างเช่น IE เวอร์ชัน 3.0 และเวอร์ชันต่อมาได้นำ ActiveX มาใช้แสดงโค้ดของ HTML นอกจากนี้ข้อบกพร่องของ ActiveX ยังสามารถใช้งานโดยแอปพลิเคชันอื่นที่ไม่ได้เกี่ยวเนื่องกันเลยก็ได้ อย่างเช่น โปรแกรมทางการเงิน หรือโปรแกรมให้บริการอ้างอิงข้อมูลจากตลาดหุ้น เป็นต้น การสร้างข้อบกพร่องของ ActiveX ไม่ได้อาศัยการเขียนโปรแกรมภาษาใดเป็นการเฉพาะ สามารถนำไปใช้ โดยใช้งาน โดยใช้ภาษา Visual C++, Visual Basic หรือ Java ยิ่งไปกว่านั้นยังสนับสนุน ActiveX บนแพลตฟอร์มอื่นๆ นอกจาก Windows เช่น Macintosh และ UNIX

ActiveX อาศัยพื้นฐานของเทคโนโลยีมาจากไมโครซอฟท์ที่เรียกว่า COM (Component Object Model) ซึ่งได้ผ่านการพัฒนามาหลายเวอร์ชัน โดยเริ่มจากการเป็น OLE (Object Linking and Embedding OLE) มาก่อน ในเวอร์ชัน 1.0 นั้นได้นำความสามารถ OLE Compound Document มาให้แต่ในเวอร์ชัน 2 ได้เป็นการนำ COM มาให้คำว่า ActiveX ก็เป็นคำที่มาใช้แทน OLE2 แม้ว่าจะมีข้อแตกต่างกันในทางเทคโนโลยีระหว่างทั้งสอง (ซึ่งจะได้อธิบายต่อไปในบทนี้)

COM สนับสนุนการสื่อสารระหว่างสองอ็อบเจกต์ที่อยู่บนคอมพิวเตอร์เครื่องเดียวกัน ทั้งสองอ็อบเจกต์อาจจะอยู่ในโปรเซสเดียวกันหรือคนละโปรเซสก็ได้ ส่วน DCOM (Distributed COM) สนับสนุนการทำงานแบบเดียวกัน แต่อ็อบเจกต์ทั้งสองอาจจะอยู่บนคอมพิวเตอร์คนละเครื่องก็ได้ อ็อบเจกต์จะยังคงใช้อินเตอร์เฟสเดียวกันในการสื่อสารกับอ็อบเจกต์อื่นๆ ไม่ว่าจะอยู่บนเครื่องเดียวกันหรือบนเครื่องอื่นก็ตาม

ส่วนที่สำคัญส่วนหนึ่งของ OLE คือแนวคิดในเรื่องของ Moniker ซึ่งเป็นอ็อบเจกต์หนึ่งของ OLE ที่เข้าใจวิธีในการทำ Initialize อ็อบเจกต์ของ OLE อื่นๆ ส่วนตัวมันเองจะต้องถูก Initialize โดยอ็อบเจกต์ที่สร้างตัวมัน อาจจะเป็นการง่ายกว่าหากจะสร้าง Moniker แทนที่จะสร้างอ็อบเจกต์ของ OLE ขึ้นมาในครั้งแรก ทั้งนี้ขึ้นอยู่กับตัว Moniker (และการทำงาน)

ก่อนจะถึงยุคของ ActiveX นั้น OCX (OLE Control) ถูกใช้งานเฉพาะอย่าง เช่น แสดง HTML เป็นต้น ActiveX Control มีความคล้ายคลึงกับ OLE Control ส่วนสำคัญที่แตกต่างกันคือ

- 1) ActiveX Control มีขนาดเล็กกว่า ทั้งนี้เนื่องจาก ActiveX Control ไม่จำเป็นต้องจัดการอินเตอร์เฟสเองทั้งหมดดังที่ OCX ต้องทำ ซึ่งทำให้ ActiveX สิ้นเปลืองการสื่อสาร น้อยกว่า OCX
- 2) ActiveX อนุญาตให้ใช้ Moniker แบบอะซิงโครนัสเพื่อเพิ่มประสิทธิภาพในการทำงาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3) ActiveX Control อนุญาตให้มีระบบรักษาความปลอดภัย และการสร้างลายเซ็นลงในโค้ด ActiveX ประกอบไปด้วยเทคโนโลยีพื้นฐานมากมาย บางส่วนก็เป็นเทคโนโลยีเกี่ยวกับเว็บไคลเอนต์โดยเฉพาะ บางส่วนก็เป็นเทคโนโลยีเกี่ยวกับเรื่องของเว็บเซิร์ฟเวอร์ หรือบางส่วนก็เป็นระบบสนับสนุนทั้งสองด้าน เนื้อหาต่อไปนี้จะได้กล่าวถึงเทคโนโลยีนี้เพิ่มเติมในรายละเอียด

### 2.6.1 ActiveX Control

ActiveX Control ประกอบด้วยองค์ประกอบย่อยๆ ของ ActiveX เมื่อมีคนกล่าวถึง ActiveX ปกติจะหมายถึงความถึง ActiveX Control ซึ่งเป็นซอฟต์แวร์โมดูลที่ไม่สามารถทำงานได้ด้วยตนเองจะทำงานได้ต้องอยู่ภายใต้ ActiveX Control อย่างเช่น เว็บเบราว์เซอร์, เวิร์ดโปรเซสเซอร์ หรือ สเปรดชีต ActiveX Container จะมีหน้าที่การทำงานเป็นการเฉพาะหน้าที่การทำงานของ ActiveX Control สามารถเป็นอะไรก็ได้ตามแต่ผู้สร้างสรรค์ต้องการจะทำให้ เช่น การเข้าถึงข้อมูลในฐานข้อมูล, การเข้าถึงข้อมูลในไฟล์, การอ่านค่าเวลา, การเชื่อมต่อกับผู้ใช้ และอื่นๆ ActiveX Control สามารถจะสื่อสารกับ ActiveX Control อื่นๆ, ActiveX Control หรือระบบปฏิบัติการ

ActiveX Control ได้มีการสร้างไว้นานอยู่แล้ว มีให้ใช้มากมายหลายอย่าง ตัวอย่างหนึ่งคือ FastNet ActiveX ของ NetMaster (เดิมชื่อ Internet ActiveX Control Pack) ซึ่งประกอบไปด้วย ActiveX Control เกี่ยวกับอินเทอร์เน็ต เช่น TCP, UDP, FTP และ SMTP (Simple Mail Transfer Protocol) ใช้สำหรับทำให้แอปพลิเคชันสามารถสื่อสารโดยใช้เทคโนโลยีเหล่านี้ โดยไม่จำเป็นต้องรู้รายละเอียดในการทำงานเหล่านั้น

### 2.6.2 ActiveX Scripting

ActiveX Scripting อนุญาตให้เก็บโค้ดของภาษาไว้ในไฟล์ HTML แทนที่จะคอมไพล์ไว้ล่วงหน้าแล้วฝังไว้ในไฟล์ HTML ซึ่งโค้ดนี้จะถูกแปลและดำเนินการ

ActiveX Scripting สนับสนุนภาษาสคริปต์ทั้ง JavaScript และ VBScript ซึ่งเป็นส่วนย่อยสำหรับใช้เขียนสคริปต์ของภาษา Visual Basic โดยสร้างขึ้นให้คงคุณสมบัติรักษาความปลอดภัยไว้ นอกจากนี้ ActiveX Scripting ยังสนับสนุน ActiveX Scripting API ที่สามารถย้ายแอปพลิเคชันจาก OCX Container ที่มีอยู่ไปยัง ActiveX Scripting Container ได้โดยง่าย

หากกล่าวให้ชัดเจนลงไปแล้ว ActiveX Scripting ประกอบด้วยสองเทคโนโลยีหลักคือ

- 1) ActiveX Scripting Host ซึ่งอาจจะเป็นเว็บเบราว์เซอร์ เซิร์ฟเวอร์ หรือแอปพลิเคชันอื่นๆ อย่างเช่น โปรแกรมสร้างเว็บเพจ หรือสเปรดชีต เป็นต้น
- 2) ActiveX Scripting Engine ทำหน้าที่แปลสคริปต์ สนับสนุนการทำงานของสคริปต์หลายอย่าง เช่น Perl, VBScript, JavaScript, Lisp และอื่น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 2.6.3 ActiveX Document

ActiveX Document ได้อาศัยพื้นฐานมาจาก OLE DocObjects เพื่อใช้แสดงผลและแก้ไขภายใน ActiveX Document แต่ไม่ได้เป็นส่วนสำคัญของเทคโนโลยี ActiveX

### 2.6.4 ActiveX Security

ActiveX ใช้เทคโนโลยีหลายอย่างจัดการทางด้านความปลอดภัย อย่างเช่น ใช้ Authenticode และเข้ารหัสลับโดยใช้คีย์สาธารณะเพื่อความมั่นใจในคุณภาพและผู้สร้างโปรแกรมที่ได้มีการดาวน์โหลดมาทำงานอย่างไดนามิก ActiveX Control สามารถเซ็นรับรองโดยบริษัทที่สร้างคอนโทรลนั้นทั้งนี้บริษัทดังกล่าวจะต้องได้ใบรับรอง X.509 จาก VeriSign และเซ็น ActiveX Control โดยใช้มาตรฐานเข้ารหัสลับด้วยคีย์ส่วนตัว และคีย์สาธารณะแบบ PKSC07 และ 10 เมื่อ ActiveX Container ได้ดาวน์โหลดคอนโทรลมาแล้ว จะทำการตรวจสอบที่เรียกว่า Authenticode เพื่อตรวจสอบลายเซ็นผ่านการตรวจสอบจึงจะเรียกใช้งานคอนโทรลนั้นได้ อย่างไรก็ตามผู้ใช้สิทธิที่จะตั้งห้ามการตรวจสอบนี้ แต่ผลก็คือจะมีความเสี่ยงทางด้านความปลอดภัยแทน

### 2.6.5 ActiveX และ Java

ActiveX Control สามารถเขียนขึ้นมาโดยใช้ภาษาต่างๆ หลายภาษารวมทั้ง Java ดังนั้น Java Applet อาจจะถูกกลายเป็น ActiveX Control, Java Applet และ JavaBean ต่างสามารถถูกดาวน์โหลดและตั้งให้ทำงานได้ในแบบไดนามิก ซึ่ง ActiveX จะใช้รูปแบบที่ทำการตรวจสอบ ActiveX Control ก่อนจะให้มีการทำงานส่วน Java นั้นชอบที่จะใช้แบบของ "กล่องทราย (Sandbox)" ที่ซึ่งองค์ประกอบของ Java Applet และ JavaBean สามารถ "เล่น" ได้โดยที่ตัวกล่องทรายนี้จะจำกัดการทำงานของแอปพลิเคชันว่าจะสามารถทำงานอะไรได้บ้าง เช่น ไม่สามารถเข้าถึงหรือเขียนดิสก์ได้

## 2.7 สวิตช์ลำแสง

สวิตช์ลำแสง คือ อุปกรณ์ที่ใช้ตรวจจับวัตถุ ประกอบด้วยส่วนสำคัญ คือ ตัวรับแสงและตัวส่งแสง ลักษณะของการตรวจจับนั้น เกิดจากการที่ลำแสงจากตัวส่งถูกส่งไปสะท้อนกับวัตถุหรือขวางด้วยวัตถุ หรือ ถูกดูดซับด้วยวัตถุ และมีผลให้ตัวรับแสงรับรู้สภาวะที่เกิดขึ้น โดยการเปลี่ยนแปลงสภาวะของสัญญาณเอาต์พุตเพื่อไปใช้งาน

คุณลักษณะเด่นของสวิตช์ลำแสง

- 1) ตรวจจับวัตถุได้โดยไม่ต้องสัมผัส
- 2) ระยะเวลาตรวจจับใกล้
- 3) สามารถตรวจจับวัตถุได้เกือบทุกชนิด
- 4) ความเร็วในการตรวจจับสูง
- 5) มีรุ่นที่สามารถแยกความแตกต่างของสีได้
- 6) ความแม่นยำในการตรวจจับสูง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 2.7.1 ชนิดของตัวรับแสงและตัวกำเนิดแสงในสวิตช์ลำแสง

1) ตัวตรวจจับแสง จะใช้โฟโตทรานซิสเตอร์ (Photo Transister) หรือ โฟโตไดโอด (Photo Diode) ทำหน้าที่ตรวจจับแสงและแปลงเป็นสัญญาณไฟฟ้า

2) ตัวกำเนิดแสง มีหลายประเภทด้วยกัน คือ

2.1) หลอดแบบมิใช่ เป็นชนิดที่ใช้ในสวิตช์ลำแสงรุ่นเก่า มีข้อเสียตรงที่ไส้จะขาดง่าย และมีขนาดที่ค่อนข้างใหญ่ แต่ในปัจจุบันก็ยังมีใช้อยู่เพื่อใช้ในงานเฉพาะแบบ

2.2) หลอด LED (Light Emitting Diode) หลอด LED เป็นอุปกรณ์ที่กำเนิดลำแสง และมีขนาดเล็ก มีความทนทานสูงนิยมใช้กันมากที่สุดในสวิตช์ลำแสงรุ่นใหม่ ชนิดของหลอด LED แบ่งตามชนิดของลำแสงที่เปล่งออกมาได้ดังนี้

2.2.1) LED แบบแสงอินฟราเรด จะสามารถแบ่งเป็นแสง ที่มีความยาวคลื่นอยู่ในช่วงระหว่าง 910-950 mm. มองด้วยตาเปล่าไม่เห็นให้ความเข้มของแสงสูงจึงส่งไปได้ในระยะไกล และสามารถส่งทะลุวัตถุบางชนิดได้ แต่ไม่สามารถแยกสีของวัตถุได้

2.2.2) LED แบบสีแดง จะเป็นแสงที่มองเห็น มีความยาวคลื่นประมาณ 660 mm. ให้ความเข้มของแสงปานกลางสวิตช์ลำแสงที่ใช้ LED แบบสีแดง จะสามารถตรวจจับสีดำ สีน้ำเงิน หรือสีเขียวบนพื้นสีขาวได้

2.2.3) LED แบบสีเขียว เป็นแสงที่มองเห็น มีความยาวคลื่นประมาณ 560 mm. ให้ความเข้มของแสงต่ำ สวิตช์ลำแสงที่ใช้ LED สีเขียวจะมีระยะการตรวจจับใกล้แต่สามารถตรวจจับสีแดงบนพื้นสีขาวได้

2.2.4) LED 3 สี (แดง, เขียว, น้ำเงิน) สวิตช์ลำแสงที่ใช้ LED 3 สีนั้นจะให้กำเนิดแสงขาวซึ่งเกิดจากแม่สีทั้ง 3 สี คือ แดง เขียว และน้ำเงิน โดยลำแสงที่เกิดเป็นแสงสีขาวนั้นจะสามารถตรวจแยกความแตกต่างของสีได้

2.3) หลอดเลเซอร์ (Laser) เป็นอุปกรณ์ที่เริ่มมีใช้ในสวิตช์ลำแสง แต่ยังไม่ค่อยนิยมมากนักเนื่องจากมีราคาสูง ส่วนจุดเด่นของหลอดเลเซอร์ คือ มีความเข้มของแสงสูง, ขนาดกะทัดรัด และสามารถส่งแสงออกเป็นเส้นตรงโดยไม่มีการกระจาย ทำให้มีระยะการตรวจจับไกลสามารถตรวจจับวัตถุที่มีขนาดเล็กได้ในระยะไกลและพื้นที่แคบๆ มีจุดของลำแสงที่มองเห็นช่วยให้สามารถปรับตั้งได้

### 2.7.2 เทคนิคในการรับส่งลำแสง

เทคนิคในการรับส่งลำแสง มีอยู่ 2 วิธี คือ

1) วิธีการรับส่งลำแสงทั่วไป เป็นวิธีที่ตัวส่งแสงจะส่งลำแสงไปอย่างต่อเนื่องเป็นปกติ เหมือนกับแสงตามธรรมชาติซึ่งจะมีการตรวจจับไม่ไกลนักและอาจจะถูกแสงภายนอกมารบกวนได้ง่าย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2) วิธีรับส่งลำแสงแบบ Pulse Modulation เป็นวิธีที่ตัวส่งจะส่งลำแสงไปเป็นจังหวะที่สม่ำเสมอด้วยอัตราความถี่สูงและที่รับแสงก็จะถูกออกแบบมาสำหรับรับสัญญาณแสงนี้โดยเฉพาะ ด้วยวิธีนี้จะทำให้มีระยะเวลาตรวจจับทำได้ไกลและด้านทานต่อแสงรบกวนจากภายนอกได้

### 2.7.3 หลักการเลือกใช้สวิทช์ลำแสงให้เหมาะสมกับงาน

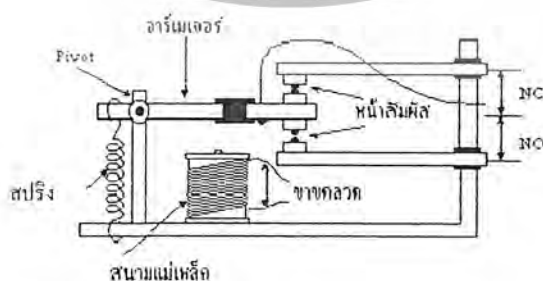
เนื่องจากสวิทช์ลำแสงนั้นมีหลายชนิดด้วยกัน ในการเลือกใช้สวิทช์ลำแสงจึงจำเป็นต้องรู้รายละเอียดคุณสมบัติบางอย่างของสวิทช์ลำแสงตลอดลักษณะการใช้งาน เพื่อที่จะเลือกใช้ใช้งานได้อย่างถูกต้องและเหมาะสมมากที่สุด ดังนั้นการเลือกใช้จึงต้องคำนึงถึง

- 1) ขนาดของวัตถุที่ตรวจจับ
- 2) ชนิดของวัตถุที่ตรวจจับ
- 3) สีและผิวของวัตถุตรวจจับ
- 4) ความแม่นยำของตำแหน่งที่ตรวจจับ
- 5) ความเร็วในการเคลื่อนที่ของวัตถุที่ตรวจจับ
- 6) สภาพแวดล้อมที่ใช้สวิทช์ลำแสง
- 7) ระยะห่างระหว่างสวิทช์ลำแสงที่อยู่ใกล้กัน
- 8) ประเภทและชนิดของแสง

### 2.8 สวิตช์แม่เหล็กไฟฟ้าหรือรีเลย์

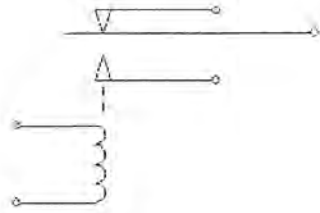
รีเลย์เป็นสวิตช์ที่ทำงานด้วยไฟฟ้า และมีการใช้งานอย่างกว้างขวางในอุตสาหกรรม รีเลย์นี้อาจแบ่งออกเป็น 2 ประเภทตามการใช้งาน คือ รีเลย์ควบคุม (Control Relays) และรีเลย์กำลัง (Power Relays)

รีเลย์ควบคุมเป็นรีเลย์ที่ใช้ในการควบคุมวงจร ไฟฟ้ากำลังต่ำ หรือ รีเลย์ตัวอื่นในวงจรการทำงานแบบอัตโนมัติ ส่วนรีเลย์กำลัง บางครั้งเรียกว่า “คอนแทกเตอร์” (Contactors) เป็นรีเลย์ที่ใช้กับระบบไฟฟ้ากำลังสูง เช่น การควบคุมมอเตอร์ เป็นต้น



(ก) โครงสร้าง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

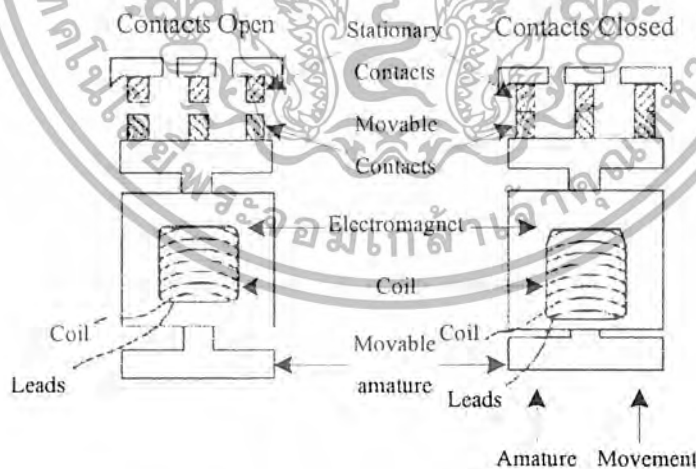


(ข) สัญลักษณ์

รูปที่ 2.13 รีเลย์ (Relays)

จากรูปที่ 2.13 รีเลย์ประกอบด้วยขดลวดที่พันอยู่บนแกนแท่งเหล็ก หน้าสัมผัสที่อยู่กับที่ (Stationary Contact) ซึ่งยึดติดกับอาร์เมเจอร์ (Armature)

การทำงานของรีเลย์ เมื่อมีกระแสไฟฟ้าไหลผ่านขดลวดที่พันรอบแท่งเหล็กอันทำให้เกิดมีอำนาจแม่เหล็กดึงอาร์เมเจอร์ ซึ่งเป็นแท่งเหล็กอ่อนเข้ามาหาตัวมัน และทำให้หน้าสัมผัสทั้งสองแตะกันวงจรทางด้านโหลดจะปิดวงจร ขณะเมื่อไม่มีกระแสไฟฟ้าไหลผ่านขดลวด แท่งเหล็กจะหมดอำนาจแม่เหล็กสปริงก็จะดันอาร์เมเจอร์กลับไปที่ตำแหน่งปกติตามเดิม อันทำให้หน้าสัมผัสทั้งสองแยกออกจากกันและวงจรทางด้านโหลดจะเปิดวงจร



(ก) ขั้วสัมผัสเปิดวงจร

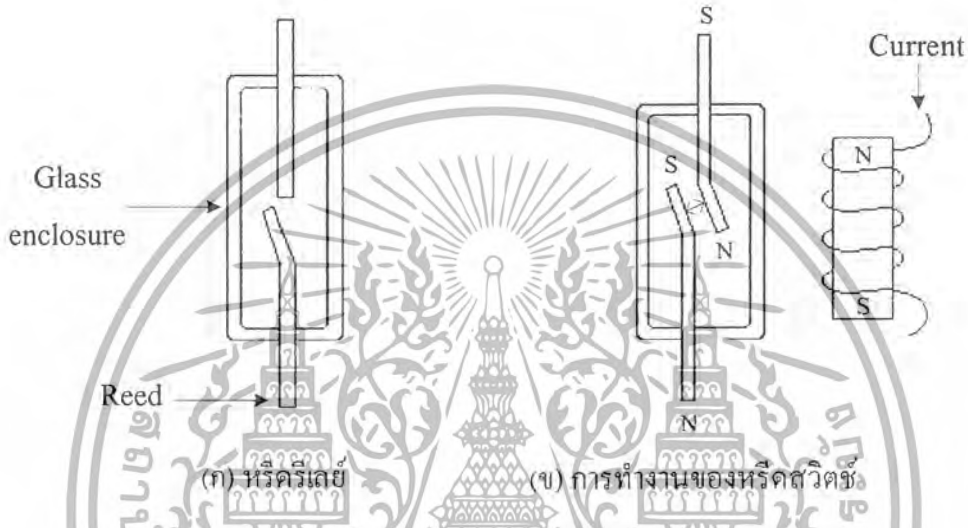
(ข) ขั้วสัมผัสปิดวงจร

รูปที่ 2.14 แมกเนติกคอนแทกเตอร์ (Magnetic Contactors)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากรูปที่ 2.14 เป็นรีเลย์กำลังที่ใช้ในการควบคุมมอเตอร์ในงานอุตสาหกรรม ที่เรียกว่า “แมกเนติก คอนแทกเตอร์” นอกจากนี้ยังมีรีเลย์อีกชนิดหนึ่งที่นิยมใช้งานกันอย่างมากคือ หรีดรีเลย์ (Reed Relay)

หรีดรีเลย์ เป็นรีเลย์ที่ประกอบด้วยขั้วสัมผัสของโลหะแผ่นบางๆ 2 ชั้น ที่มีความไวต่อสนามแม่เหล็ก ซึ่งบรรจุอยู่ในหลอดแก้ว หน้าสัมผัสนี้เรียกว่า “หรีดสวิตช์” (Reed Switch) และภายนอกหลอดแก้วจะมีขดลวดพันอยู่รอบตัวมัน ดังแสดงในรูปที่ 2.15



รูปที่ 2.15 หรีดรีเลย์ (Reed Relay)

การทำงานของหรีดรีเลย์จะเหมือนกับรีเลย์ทั่วไป คือ เมื่อมีกระแสไฟฟ้าไหลผ่านขดลวดรีเลย์จะทำให้เกิดสนามแม่เหล็กขึ้น และมีผลทำให้หรีดสวิตช์ภายในหลอดแก้วแต่ละอันได้ ดังแสดงในรูปที่ 2.15 (ข) ขณะที่หรีดสวิตช์แต่ละอันยอมทำให้เกิดการอาร์กที่ขั้วสัมผัสทั้งสอง และมีผลทำให้ขั้วสัมผัสความเสียหาย การป้องกันการอาร์กของขั้วสัมผัสทำได้โดยการบรรจุโลหะเหลวที่เป็นปรอทไว้ภายในหลอดแก้ว เพื่อช่วยขจัดปัญหาที่เกิดขึ้น

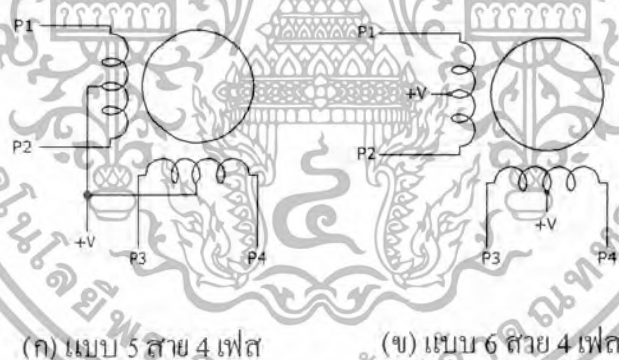
## 2.9 ความรู้เบื้องต้นเกี่ยวกับสตีปิ้งมอเตอร์

สำหรับสตีปิ้งมอเตอร์เป็นมอเตอร์ที่มีลักษณะการทำงานแตกต่างจากมอเตอร์ทั่วไป เพราะจะต้องป้อนสัญญาณเป็นพัลส์ให้แก่ขดลวดของมอเตอร์เป็นจังหวะอย่างเหมาะสม และการหมุนของมอเตอร์ชนิดนี้จะหมุนตามจังหวะตามพัลส์ที่ป้อนเข้ามา ไม่หมุนต่อเนื่องเหมือนกับมอเตอร์ธรรมดา ทำให้ผู้ควบคุมสามารถเลือกตำแหน่งที่ต้องการให้มอเตอร์หยุดหมุนได้ จังหวะการหมุนของสตีปิ้งมอเตอร์เรียกว่า สตีป (Step) ความละเอียดของมอเตอร์กำหนดเป็นองศาที่หมุนไปในหนึ่งสตีป หากมอเตอร์มีจำนวนองศาต่อสตีปมาก หมายความว่ามอเตอร์ตัวนี้มีความละเอียดของการหมุนต่ำ ยกตัวอย่าง การหมุนครบ 1 รอบเท่ากับ 360 องศา หากมอเตอร์มีสตีปการหมุนเท่าเอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กับ 7.5 องศาต่อสเต็ป มอเตอร์ตัวนี้มีความละเอียดของการหมุนเท่ากับ 48 ตำแหน่ง แต่ถ้าหากมีสเต็ปการหมุนเท่ากับ 1.8 องศาต่อสเต็ป ความละเอียดของการหมุนเท่ากับ 200 ตำแหน่ง จะเห็นได้ว่า มอเตอร์ตัวหลังมีความละเอียดสูงกว่าตัวแรกมาก ทำให้นำมาใช้ในงานที่ต้องการกำหนดตำแหน่งได้ดีกว่า แม่นยำกว่า ผนวกเข้ากับวงจรขับเคลื่อนฮาล์ฟสเต็ป ความละเอียดของการหมุนจะเพิ่มขึ้นอีก 2 เท่า ทำให้มีความละเอียดของการหมุนเป็น 400 ตำแหน่ง

ขนาดของสเต็ปปิ้งมอเตอร์ที่มีการผลิตและจำหน่ายในท้องตลาด มีตั้งแต่ขนาดแรงดันต่ำ 3V ไปจนถึง 24V ส่วนขนาดของกระแสมีตั้งแต่ไม่กี่สิบลิมิลลิแอมป์อันเป็นสเต็ปปิ้งมอเตอร์ตัวเล็ก ไปจนถึงเป็นสิบแอมป์ซึ่งขนาดของมอเตอร์ก็จะใหญ่โตขึ้นตามลำดับ ราคาอยู่ในหลักเป็นร้อยบาทขึ้นไปสำหรับของใหม่

สเต็ปมอเตอร์ได้รับการพัฒนาอย่างต่อเนื่อง จนในปัจจุบันสเต็ปปิ้งมอเตอร์ที่นิยมใช้กันอย่างแพร่หลายมากที่สุด และหาได้ง่ายคือ สเต็ปปิ้งมอเตอร์แบบยูนิโพลาร์ (Uni-Polar Stepper Motor) มีลักษณะการพันขดลวดของมอเตอร์แสดงในรูปที่ 2.16



(ก) แบบ 5 สาย 4 เฟส

(ข) แบบ 6 สาย 4 เฟส

รูปที่ 2.16 โครงสร้างพื้นฐานของสเต็ปปิ้งมอเตอร์แบบยูนิโพลาร์ทั้งแบบ 5 และ 6 สาย

สเต็ปปิ้งมอเตอร์แบบนี้มีการพันขดลวด 2 ขดบนแต่ละขั้วแม่เหล็กของสเตเตอร์ แต่ละขดแบ่งเป็น 2 เฟส รวมมอเตอร์ทั้งตัวจะมี 4 เฟส คือ เฟส 1, 2, 3 และ 4 มีการต่อสายออกมาจากขดลวดแต่ละขดเพื่อจ่ายไฟเลี้ยงทำให้สเต็ปปิ้งมอเตอร์แบบนี้มีทั้งแบบ 5 สาย และ 6 สาย ถ้าเป็นแบบ 5 สาย จะเป็นการนำสายไฟเลี้ยงของขดลวดทั้งสองมาต่อรวมกันเป็นสายเดียว

การกระตุ้นและควบคุมการหมุนของมอเตอร์ให้เคลื่อนที่ไปแต่ละสเต็ปทำได้โดยจ่ายกำลังไฟฟ้าไปยังขดลวดแต่ละขดบนสเตเตอร์ซึ่งต้องป้อนเป็นแบบซีควเอนเชียลในรูปแบบที่ถูกต้องด้วย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สามารถแบ่งได้เป็น 3 รูปแบบ คือ แบบฟูลสเต็ป 1 เฟส (full step 1 phase), แบบฟูลสเต็ป 2 เฟส และแบบฮาล์ฟสเต็ป (half step)

แบบฟูลสเต็ป 1 เฟส เป็นการกระตุ้นที่มีรูปแบบง่ายที่สุด โดยทำการกระตุ้นขดลวดทีละขด ในเวลาหนึ่งไต่เรียงติดกันไป เช่น เริ่มต้นที่ขดที่ 1, 2, 3, 4 แล้ววนกลับมาขดที่ 1 วนไปเรื่อยๆ หรือ เริ่มที่ขดที่ 1 แล้วย้อนไปยังขดที่ 4, 3, 2 แล้วกลับมาขดที่ 1 อีกครั้ง ซึ่งทำให้ทิศทางของการหมุนสวนกัน ในการกระตุ้นรูปแบบนี้จึงมีขดลวดเพียงขดเดียวในเวลาหนึ่งที่ถูกกระตุ้นเท่านั้น วงจรกระตุ้นแบบเวฟจึงมีราคาถูกและง่าย ขั้นตอนการทำงานต่างๆ แสดงดังตารางที่ 2.6

ตารางที่ 2.6 รูปแบบการขับสเต็ปปั้งมอเตอร์แบบฟูลสเต็ป 1 เฟส

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

แบบฟูลสเต็ป 2 เฟส เป็นการกระตุ้นซึ่งคล้ายกับแบบ 1 เฟส แต่การกระตุ้นแบบนี้จะทำการกระตุ้นแบบนี้จะทำการกระตุ้น โดยจ่ายกำลังไฟฟ้าไปที่ขดลวด 2 ขด ที่อยู่ใกล้กันในเวลาเดียวกัน และเรียงติดกันไปเช่นเดียวกับแบบเวฟ ดังตัวอย่าง ขดลวดขดแรกที่ถูกกระตุ้นจะเป็นขดที่ 1 และ 2 ตามด้วยการกระตุ้นขดที่ 2 และ 3 ต่อไปเป็นขดที่ 3 และ 4 ถัดไปเป็นขดที่ 4 และ 1 แล้วกลับมาที่ขดที่ 1 และ 2 วนไปตามลำดับเช่นนี้ หรือเริ่มที่ขด 1 และ 4 ตามด้วยขดที่ 4 และ 3 ถัดไปเป็นขดที่ 3 และ 2 ต่อไปเป็นขดที่ 2 และ 1 แล้ววนกลับมาที่ขดที่ 1 และ 4 ทิศทางการหมุนจะสวนทางกัน การกระตุ้นสเต็ปปั้งมอเตอร์แบบนี้สามารถเพิ่มแรงบิดได้มากกว่าแบบฟูลสเต็ป 1 เฟส โรเตอร์จะเคลื่อนที่ด้วยแรงดึงอย่างเต็มแรงจาก 2 ขดลวดที่ถูกกระตุ้นพร้อมกัน และต่อไปด้วยแรงดึงจากอีก 2 ขดลวดถัดไปสำหรับข้อเสียคือการกระตุ้นแบบนี้ต้องใช้กำลังไฟฟ้ามามากขึ้น ขั้นตอนการทำงานต่างๆ แสดงดังตารางที่ 2.7

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 2.7 รูปแบบการขับสตีปี่งมอเตอร์แบบฟูลสตีป 2 เฟส

สตีปี่งที่	เฟสที่ 1	เฟสที่ 2	เฟสที่ 3	เฟสที่ 4
1	ทำงาน	ทำงาน	-	-
2	-	ทำงาน	ทำงาน	-
3	-	-	ทำงาน	ทำงาน
4	ทำงาน	-	-	ทำงาน

แบบฮาล์ฟสตีปี่งเป็นรูปแบบที่ผสมผสานระหว่างการกระตุ้นแบบฟูลสตีป 1 และ 2 เฟส เพื่อเพิ่มจำนวนของสตีปี่งต่อรอบอีกเท่าตัวหนึ่ง ในระบบนี้จะทำการกระตุ้นขดลวดเรียงกันไปเป็นลำดับดังนี้ เริ่มจากขดลวดที่ 1, 1 และ 2, 2, 2 และ 3, 3, 3 และ 4, 4, 4 และ 1 แล้ววนกลับมายังขดลวดที่ 1 แรงบิดที่ได้จากการกระตุ้นแบบนี้จะเพิ่มมากขึ้นอีก เพราะช่วงสตีปี่งมีระยะที่สั้นลง แต่สตีปี่งเกิดแรงดึงจากขดลวด 2 ขดที่ถูกกระตุ้นพร้อมกัน ความถูกต้องของตำแหน่งมีเพิ่มมากขึ้น แต่ต้องพึงระวังไว้อีกประการหนึ่งว่า เมื่อกระตุ้นให้ทำงานในรูปแบบนี้จะต้องทำการหมุนถึง 2 สตีปี่ง จึงจะได้เท่ากับระยะเท่ากับ 1 สตีปี่งเต็มของการควบคุมใน 2 แบบแรก สำหรับแหล่งจ่ายกำลังไฟฟ้าต้องใช้ขนาดเท่ากับแบบ 2 เฟสเป็นอย่างน้อย จึงจะเพียงพอ ขั้นตอนการทำงานต่างๆ แสดงดังตารางที่ 2.8

ตารางที่ 2.8 รูปแบบการขับสตีปี่งมอเตอร์แบบฮาล์ฟสตีปี่ง

สตีปี่งที่	เฟสที่ 1	เฟสที่ 2	เฟสที่ 3	เฟสที่ 4
1	ทำงาน	-	-	-
2	ทำงาน	ทำงาน	-	-
3	-	ทำงาน	-	-
4	-	ทำงาน	ทำงาน	-
5	-	-	ทำงาน	-
6	-	-	ทำงาน	ทำงาน
7	-	-	-	ทำงาน
8	ทำงาน	-	-	ทำงาน

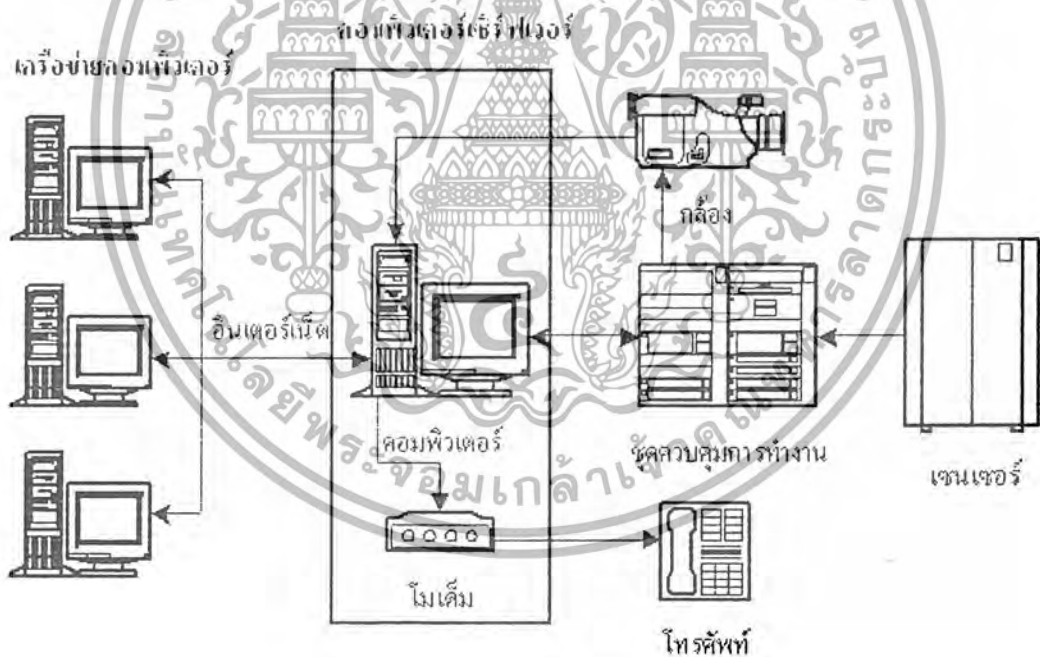
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### บทที่ 3

#### การออกแบบ การสร้าง และการทำงาน

##### 3.1 กล่าวนำ

ระบบตรวจสอบและเตือนภัยผู้บุกรุกโดยผ่านเครือข่ายอินเทอร์เน็ตมีส่วนประกอบ 2 ส่วนหลักๆ คือ ส่วนของฮาร์ดแวร์และส่วนของซอฟต์แวร์ ซึ่งส่วนของฮาร์ดแวร์นั้นจะประกอบด้วยวงจรต่างๆ คือ แผงวงจรควบคุมการทำงานหลัก (ชุดอินเทอร์เฟซ), วงจรขับสแตมป์มอเตอร์, วงจรขยายพอร์ตอินพุตเอาต์พุต (โดยใช้ไอซี PCF8574) และวงจรแหล่งจ่ายไฟ เพื่อใช้ในการควบคุมการทำงานของระบบเตือนภัย และส่วนของซอฟต์แวร์ซึ่งจะประกอบไปด้วยโปรแกรมควบคุมการทำงานส่วนต่างๆดังนี้ คือ โปรแกรมตรวจสอบสัญญาณนิรภัย (จากวงจขยายพอร์ตอินพุตเอาต์พุต), โปรแกรมโทรศัพท์แจ้งเตือนภัยและ โปรแกรมควบคุมสแตมป์มอเตอร์ สามารถแสดงการทำงานจากระบบตรวจสอบและเตือนภัยผู้บุกรุกโดยผ่านเครือข่ายอินเทอร์เน็ต ดังรูปที่ 3.1



รูปที่ 3.1 ผังการทำงานของระบบตรวจสอบและเตือนภัยผู้บุกรุกโดยผ่านเครือข่ายอินเทอร์เน็ต

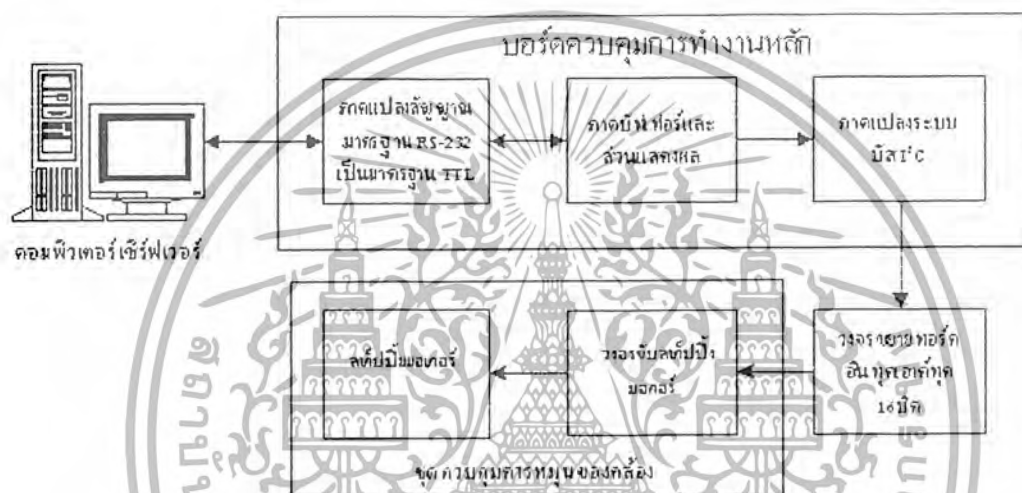
##### 3.2 การออกแบบด้านฮาร์ดแวร์

การออกแบบทางด้านฮาร์ดแวร์ของระบบตรวจสอบและเตือนภัยผู้บุกรุกโดยผ่านเครือข่ายอินเทอร์เน็ตมีส่วนประกอบดังต่อไปนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 3.2.1 แผงวงจรควบคุมการทำงานหลัก

แผงวงจรควบคุมการทำงานหลักใช้ในการควบคุมการทำงานของระบบตรวจสอบและเตือนภัยผู้บุกรุกโดยผ่านเครือข่ายอินเทอร์เน็ต โดยจะประกอบไปด้วยวงจรมหาภาคต่างๆ คือ ภาคแปลงสัญญาณมาตรฐาน RS-232 เป็นมาตรฐาน TTL, ภาคบัฟเฟอร์และส่วนแสดงผล และภาคแปลงระบบบัส I<sup>2</sup>C ซึ่งมีการเชื่อมต่อกับส่วนต่างๆ ดังนี้ วงจรขยายอินพุตเอาต์พุตและวงจรขับเตีปิ้งมอเตอร์



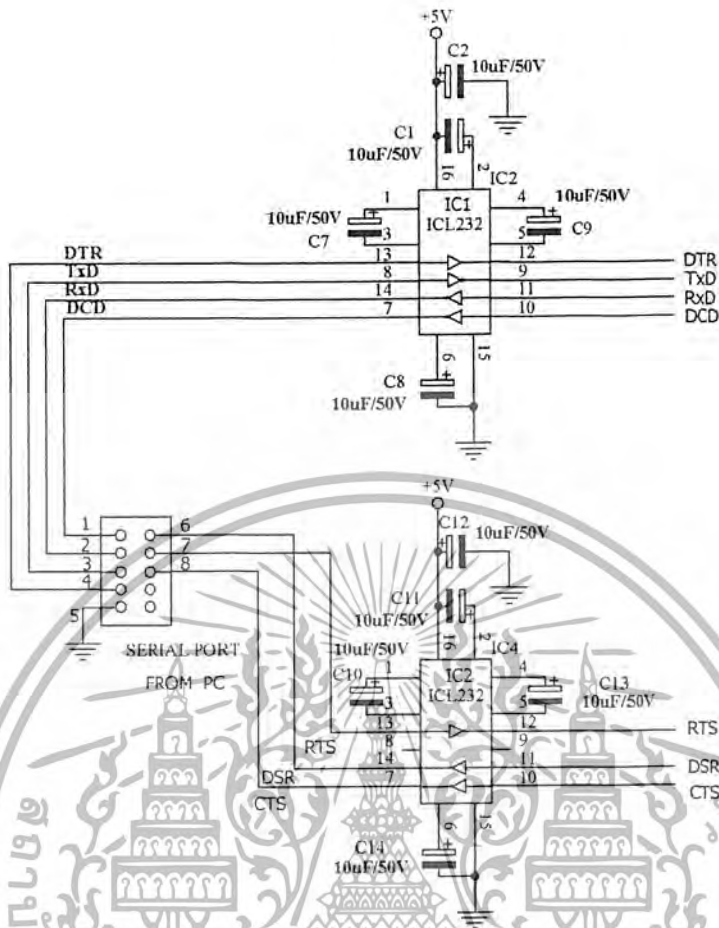
รูปที่ 3.2 แผงวงจรควบคุมการทำงานหลัก

#### 1) ภาคแปลงสัญญาณมาตรฐาน RS-232 เป็นมาตรฐานที่ทีแอล

สัญญาณจากพอร์ตอนุกรมของคอมพิวเตอร์ตามแบบมาตรฐาน RS-232 มีระดับแรงดันอยู่ที่ -12V สำหรับลอจิก "1" และ +12V สำหรับลอจิก "0" เมื่อถูกส่งผ่านบอร์ดจะได้รับการแปลงระดับให้มีแรงดัน +5V สำหรับลอจิก "1" และ 0V สำหรับลอจิก "0" ตามมาตรฐานของสัญญาณทีทีแอล โดยใช้ไอซี MAX232

ภายในไอซี MAX232 มีวงจรแปลงอยู่ภายในทั้งหมด 4 ชุดแบ่งออกเป็นวงจรแปลงระดับสัญญาณทีทีแอล เป็น RS-232 สองชุด และวงจรแปลงจาก RS-232 เป็นทีทีแอลอีกสองชุด ดังนั้นเมื่อต้องการใช้งานพอร์ตอนุกรมซึ่งมีขาอินพุตและขาเอาต์พุตมากกว่า 4 ขาต้องใช้ไอซี MAX232 สองตัวจึงจะเพียงพอ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.3 วงจรแปลงสัญญาณมาตรฐาน RS-232 เป็นมาตรฐานทีทีแอล

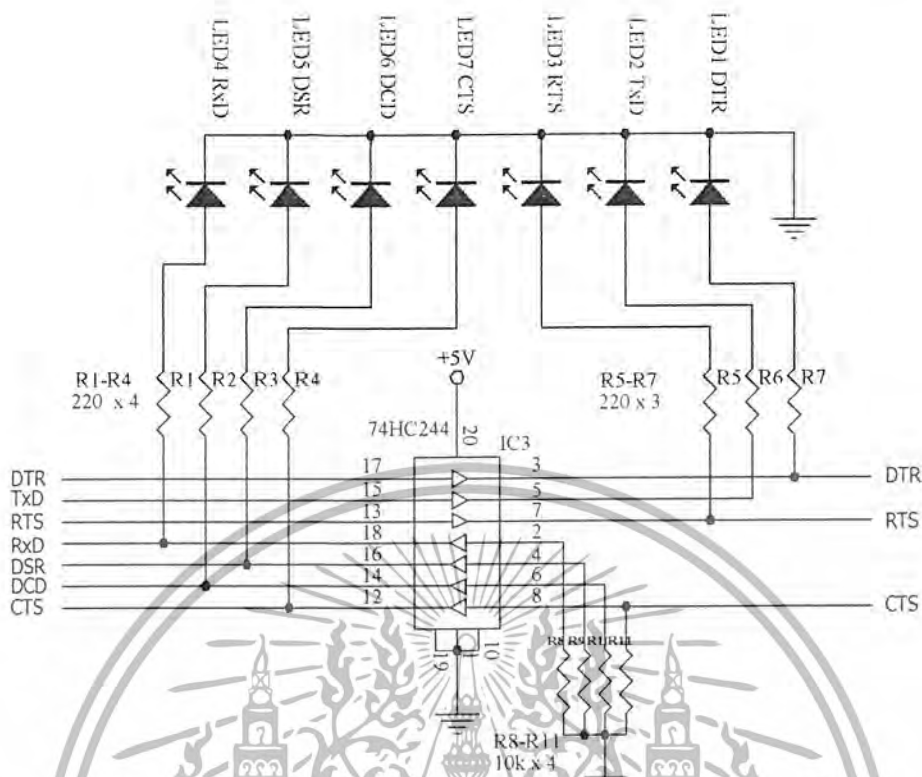
2) ภาควิชาฟิสิกส์และส่วนแสดงผล

สัญญาณเอาต์พุตของพอร์ตอนุกรมที่ผ่าน MAX232 ได้แก่ DTR, RTS และ TxD ถูกส่งไปยัง ไอซีบัพเฟอร์เบอร์ 74HC244 เพื่อขยายกระแสและป้องกันความเสียหายที่อาจเกิดกับพอร์ตอนุกรม ที่เอาต์พุตของ 74HC244 มี LED ค่อยเอาไว้เพื่อแสดงสถานะของบิตเอาต์พุตบิตต่างๆ ของพอร์ตอนุกรม

สัญญาณอินพุตของพอร์ตอนุกรม ได้แก่ CTS, DCD, DSR และ RxD จากบอร์ดก่อนจะป้อนเข้าสู่ MAX232 จะต้องผ่าน ไอซีบัพเฟอร์ 74HC244 ก่อนเพื่อขยายกระแส ที่เอาต์พุตของ 74HC244 มี LED ค่อยไว้เพื่อแสดงสถานะในขณะนั้นของขา CTS, DCD, DSR และ RxD

ทั้งสัญญาณอินพุตและเอาต์พุตของพอร์ตอนุกรมหลังจากผ่าน ไอซี 74HC244 แล้วจะถูกต่อเข้ากับคอนเน็คเตอร์ K3 เพื่อนำไปใช้งานต่อพ่วงกับอุปกรณ์ภายนอกต่อไป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.4 วงจรบัฟเฟอร์และส่วนแสดงผล

### 3) การแปลงระบบบัส I<sup>2</sup>C

สัญญาณจากพอร์ตอนุกรมก่อนที่จะเชื่อมต่อกับอุปกรณ์ I<sup>2</sup>C จะต้องมีการแปลงระดับแรงดันที่เหมาะสมให้แก่ตัวมันก่อน โดยสถานะปกติระบบ I<sup>2</sup>C บัสจะต้องมีลอจิกเป็น “1” ดังนั้นในวงจรจะเห็นว่าทรานซิสเตอร์ Q2 และ Q3 จะมีตัวต้านทาน R13 และ R17 ต่อลงกราวด์เอาไว้เพื่อว่าในสถานะปกติทรานซิสเตอร์จะไม่ทำงานเอาต์พุตของ I<sup>2</sup>C บัสจะมีลอจิก “1” จากตัวต้านทานพูลอัพ R12 และ R15 เมื่อต้องการให้บัสมีลอจิก “0” จะต้องป้อนลอจิก “1” (แรงดัน +5V) ให้กับขาเบสของทรานซิสเตอร์ เพื่อให้ทรานซิสเตอร์ทำงาน และต่อขาเอาต์พุตลงกราวด์ สำหรับการอ่านค่าข้อมูลจากบัส I<sup>2</sup>C จะต้องกำหนดให้ขา SDA มีลอจิกเป็น “1” ก่อนโดยการป้อนลอจิก “0” ให้กับทรานซิสเตอร์ Q2

เพื่อเป็นการป้องกันการอ่านหรือเขียนข้อมูลผิดพลาดไปยังอุปกรณ์ I<sup>2</sup>C บัส จึงจำเป็นต้องมีคิฟสวิทช์เพื่ออีน่าเบิลหรือดิสเอน่าเบิลสัญญาณที่ป้อนให้กับวงจรแปลงบัส I<sup>2</sup>C ถ้ากำหนดให้คิฟสวิทช์ I<sup>2</sup>C ENABLE ทุกตัว ON จะเป็นการอีน่าเบิลบัส I<sup>2</sup>C ถ้ากำหนดให้คิฟสวิทช์ทุกตัว OFF จะเป็นการดิสเอน่าเบิลบัส I<sup>2</sup>C

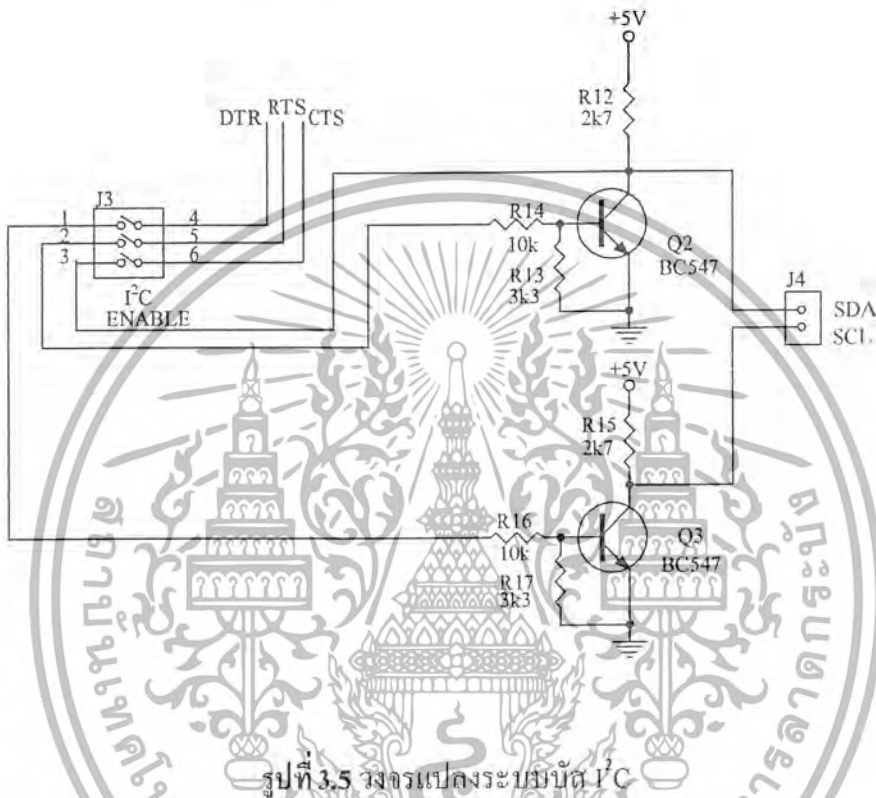
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ขาพอร์ตอนุกรมที่ต่อใช้งานกับระบบบัส I<sup>2</sup>C บัสนี้ดังนี้

ขา DTR ใช้เป็นขา SCL เพื่อป้อนสัญญาณนาฬิกา

ขา RTS ใช้เป็นขา SDA ในขณะที่ส่งข้อมูล

ขา CTS ใช้เป็นขา SDA ในขณะที่รับข้อมูล



รูปที่ 3.5 วงจรแปลงระบบบัส I<sup>2</sup>C

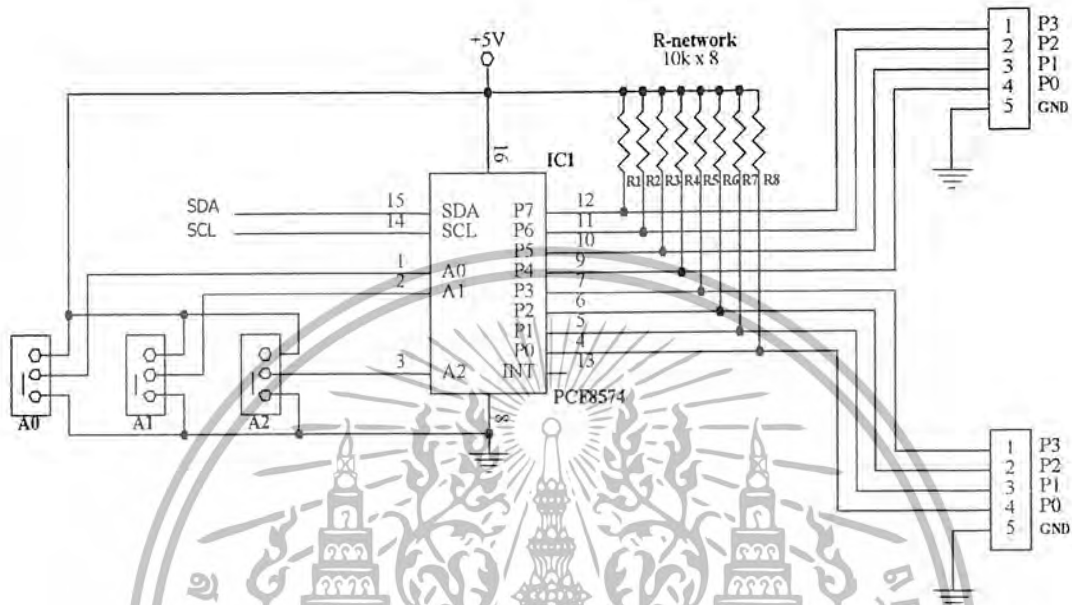
### 3.2.2 วงจรขยายพอร์ตอินพุตเอาต์พุต

วงจรสร้างสัญญาณบัส I<sup>2</sup>C จากบอร์ดควบคุมการทำงานหลักได้ทำการต่อพ่วงระบบบัสเข้ากับไอซี PCF8574 จำนวน 5 ตัว โดย PCF8574 ตัวหนึ่งใช้เป็นพอร์ตอินพุต 8 ช่อง และอีก 4 ตัวหนึ่งสำหรับใช้งานเป็นเอาต์พุต การสั่งงานให้ PCF 8574 ตัวใดตัวหนึ่งทำงานนั้นทำได้โดยการกำหนดแอดเดรส A0-A2 ซึ่งบนบอร์ดควบคุมการทำงานหลักนั้นจะมีจัมเปอร์สำหรับกำหนดค่าแอดเดรสไว้ให้แล้ว ถ้าต่อขาแอดเดรสลงกราวด์ ขาแอดเดรสนั้นจะมีลอจิก “0” ถ้าต่อเข้ากับแรงดัน +5V จะมีลอจิก “1”

วงจรเอาต์พุตของ PCF8574 เป็นแบบคอลเล็กเตอร์เปิด ดังนั้นที่ขาพอร์ตของ PCF8574 จะต้องต่อตัวต้านทานพูลอัปเอาไว้เพื่อกำหนดให้ในสภาวะปกติขาพอร์ตมีสถานะเป็น “1” คือพร้อมสำหรับส่งค่าออกเอาต์พุตหรือรับค่าอินพุตจากภายนอก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การกำหนดแอดเดรสของ PCF8574 ทั้ง 5 ตัว จะต้องกำหนดไม่ให้แอดเดรสตรงกัน ไม่งั้นนั้นจะเกิดความสับสนในการอ่านหรือเขียนข้อมูล



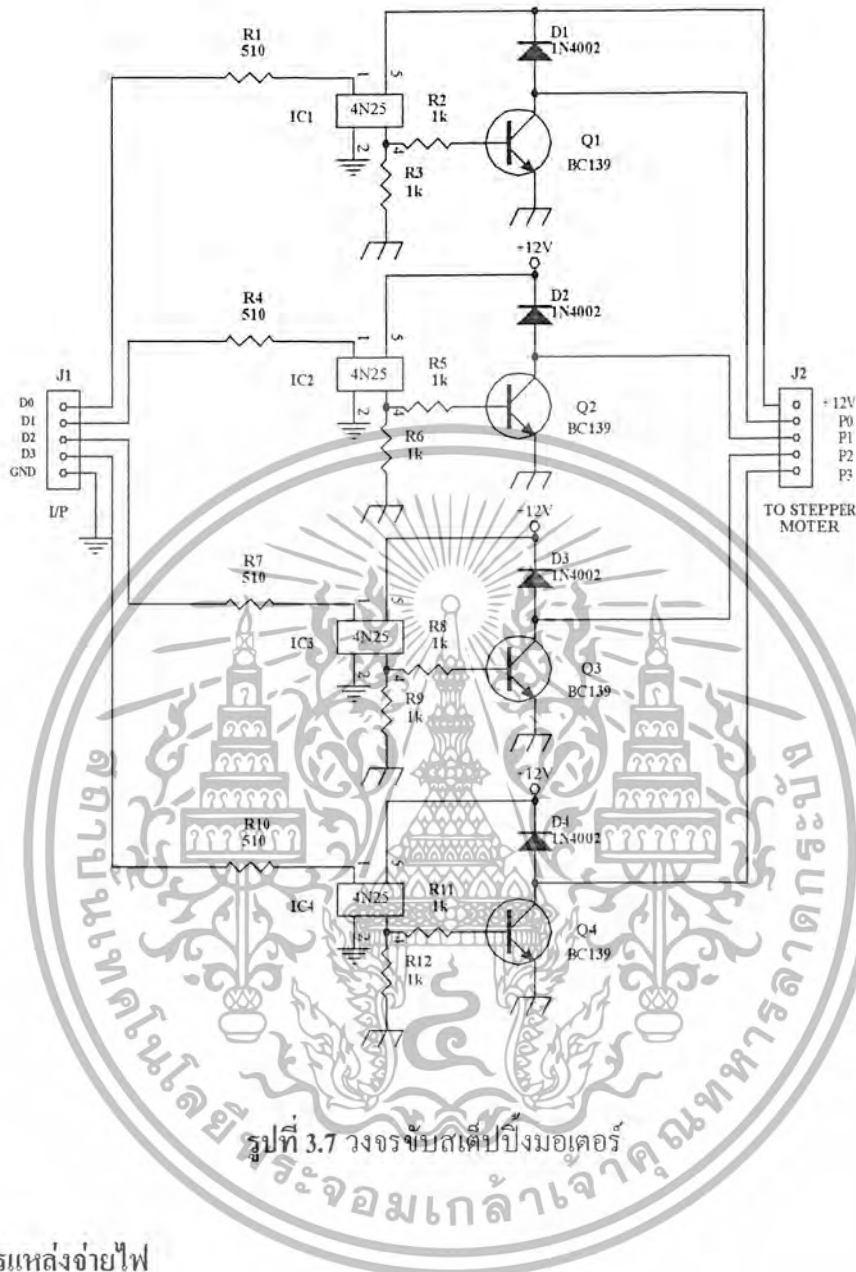
รูปที่ 3.6 วงจรขยายพอร์ตอินพุตเอาต์พุต 16 บิต

### 3.2.3 วงจรขับสเต็ปมอเตอร์

เนื่องจากสเต็ปมอเตอร์ต้องการแรงดัน 12V จึงจะทำงานแต่แรงดันที่ได้จากวงจรมอเตอร์อินพุตเอาต์พุตมีเพียง 5V ซึ่งไม่เพียงพอที่จะทำให้สเต็ปมอเตอร์ทำงานได้ ดังนั้นจึงต้องมีวงจรขับสเต็ปมอเตอร์เพิ่มเข้ามา

เป็นวงจรขับสเต็ปมอเตอร์แบบยูนิโพลาร์ที่มีการแยกส่วนของอินพุตและเอาต์พุตแยกออกจากกัน โดยใช้การเชื่อมต่อทางแสงหรือออปโตคัปเปอเรอร์ ทำให้ลดปัญหาสัญญาณรบกวนที่อาจผ่านมาจากกราวด์ของระบบ ทั้งยังสามารถช่วยลดความเสียหายที่เกิดขึ้นกับคอมพิวเตอร์ได้ สัญญาณอินพุตจะถูกป้อนเข้าที่ออปโตคัปเปอเรอร์เอาต์พุตของออปโตคัปเปอเรอร์จะต่อเข้ากับทรานซิสเตอร์ BD139 โดยเมื่อทำการป้อนลอจิก "1" ให้กับออปโตคัปเปอเรอร์ทำให้ทรานซิสเตอร์ภายในออปโตคัปเปอเรอร์ทำงาน ส่งผลให้เกิดกระแสไปไบแอสให้ทรานซิสเตอร์ BD139 ทำงาน ทำให้ขดลวดของสเต็ปมอเตอร์ที่ต่ออยู่ที่ขาคอลเล็กเตอร์ต่อลงกราวด์ เกิดกระแสไฟฟ้าไหลครบวงจรมอเตอร์จึงทำงาน เกิดการเคลื่อนที่ไปหนึ่งสเต็ป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

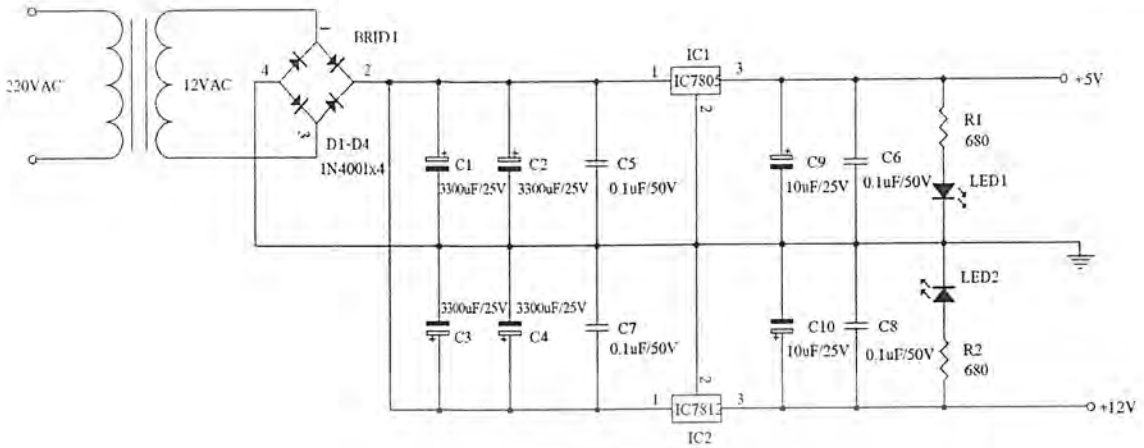


รูปที่ 3.7 วงจรขับสี่เฟสปั๊มมอเตอร์

### 3.2.4 วงจรแหล่งจ่ายไฟ

เป็นวงจรที่ทำหน้าที่แปลงแรงดันจากหม้อแปลง 12 VAC กระแส 2A ให้เป็นแรงดันไฟตรงจำนวน 2 ชุด คือชุดแรงดันไฟตรง 5V โดยใช้ ไอซีเบอร์ 7805 และชุดแรงดันไฟตรง 12V โดยใช้ไอซีเบอร์ 7812 โดยมีตัวเก็บประจุ C1, C2, C3 และ C4 ทำหน้าที่กรองแรงดันและกระแสให้คงที่ แล้วส่งต่อไปยังไอซีทั้ง 2 ตัว เพื่อรักษาระดับของแรงดัน ส่วน C9 และ C10 ทำหน้าที่กรองแรงดันอีกต่อหนึ่งก่อนจะส่งไปยังเอาต์พุต โดยที่ LED1 และ LED2 ทำหน้าที่แสดงสถานะการทำงานของแหล่งจ่ายไฟตรงทั้ง 2 ชุด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.8 วงจรแหล่งจ่ายไฟ

3.2.5 การออกแบบฐานควบคุมการหมุนของมอเตอร์

1) การออกแบบส่วนยึดติดคล่องกับสเต็ปปีงมอเตอร์

การออกแบบส่วนยึดติดคล่องกับสเต็ปปีงมอเตอร์คำนึงถึงการออกแบบ โครงสร้างที่สามารถควบคุมการหมุนของมอเตอร์ได้ 4 ทิศทาง คือ บน-ล่าง และ ซ้าย-ขวา โดยจะยึดติดกับแผ่นอลูมิเนียมฉากเพื่อให้มีน้ำหนักที่เบา ดังรูปที่ 3.9



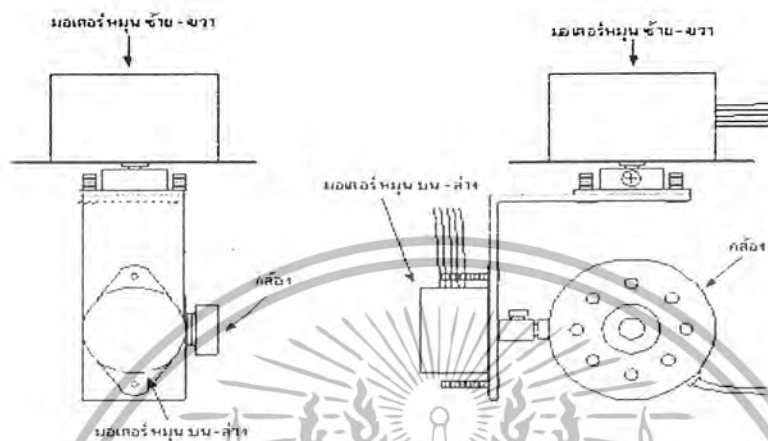
(ก) ด้านหน้า

(ข) ด้านข้าง

รูปที่ 3.9 โครงสร้างส่วนยึดติดคล่องกับสเต็ปปีงมอเตอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

นำมาประกอบกันได้ดังรูปที่ 3.10

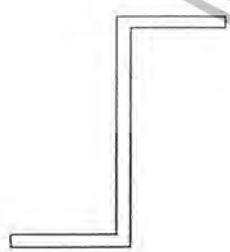


(ก) ด้านข้าง

(ข) ด้านหน้า

รูปที่ 3.10 โครงสร้างของชุดควบคุมการหมุนของกล้อ

2) การออกแบบส่วนยึดติดชุดกล้อกับเพดาน การออกแบบส่วนยึดติดชุดกล้อกับเพดานทำนึ่งถึงการออกแบบโครงสร้างที่สามารถยึดติดชุดกล้อกับเพดานได้ ดังรูปที่ 3.11



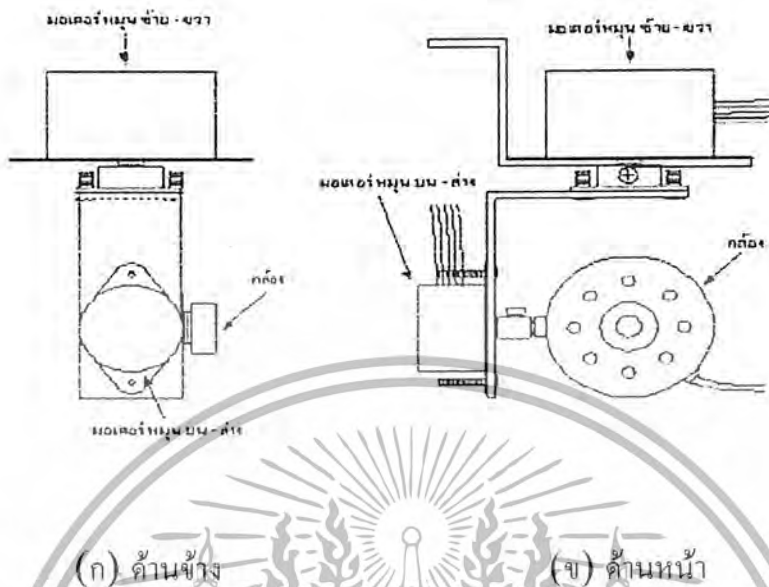
(ก) ด้านข้าง

(ข) ด้านหน้า

รูปที่ 3.11 โครงสร้างส่วนยึดติดชุดกล้อกับเพดาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไมออนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

นำส่วนยึดติดกับเพดานมาติดกับชุดกล่องได้ดังรูปที่ 3.12



รูปที่ 3.12 โครงสร้างของชุดควบคุมกล่องที่สามารถยึดติดกับเพดาน

### 3.3 การออกแบบด้านซอฟต์แวร์

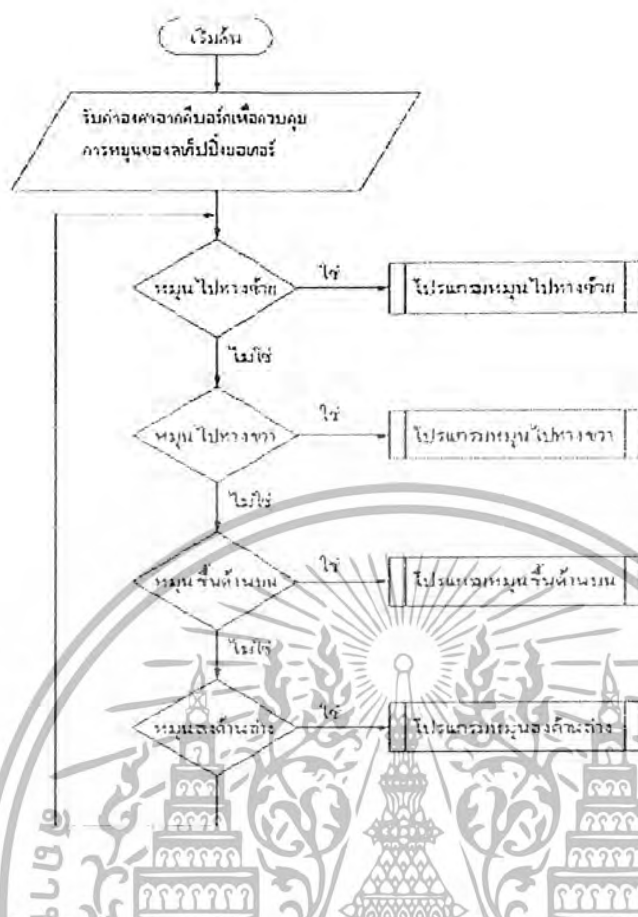
การออกแบบทางด้านซอฟต์แวร์ของระบบเตือนภัยผู้บุกรุกสำนักงานผ่านเครือข่ายอินเทอร์เน็ตจะแบ่งออกเป็น 4 ส่วนหลักๆ คือ การออกแบบ โปรแกรมควบคุมสตैंปิ้งมอเตอร์จากเครื่องเซิร์ฟเวอร์ การออกแบบ โปรแกรมควบคุมสตैंปิ้งมอเตอร์ผ่านเครือข่ายอินเทอร์เน็ต การออกแบบ โปรแกรมตรวจสอบสัญญาณนิรภัย และการออกแบบ โปรแกรมโทรศัพท์แจ้งเตือนภัย

#### 3.3.1 การออกแบบโปรแกรมควบคุมสตैंปิ้งมอเตอร์

##### 1) การออกแบบโปรแกรมควบคุมสตैंปิ้งมอเตอร์จากเครื่องเซิร์ฟเวอร์

โปรแกรมการควบคุมสตैंปิ้งมอเตอร์มีหลักการทำงานคือ ให้ผู้ใช้ทำการป้อนองศาของชุดควบคุมมอเตอร์ซึ่งมีอยู่ 4 ชุดด้วยกัน โดยเลือกป้อนของแต่ละชุดได้ทันที เมื่อป้อนองศาเสร็จจึ้นแล้วก็เลือกทิศทางการหมุนของสตैंปิ้งมอเตอร์ คือ หมุนซ้าย หมุนขวา หมุนขึ้นบน และหมุนลงล่าง เมื่อเลือกทิศทางการหมุนของสตैंปิ้งมอเตอร์แล้ว จะมีโปรแกรมน้อยในการคำนวณค่าองศาและควบคุมทิศทางการหมุนของที่เลือกไว้ ส่งค่าออกไปควบคุมมอเตอร์ผ่านทางพอร์ตอนุกรม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

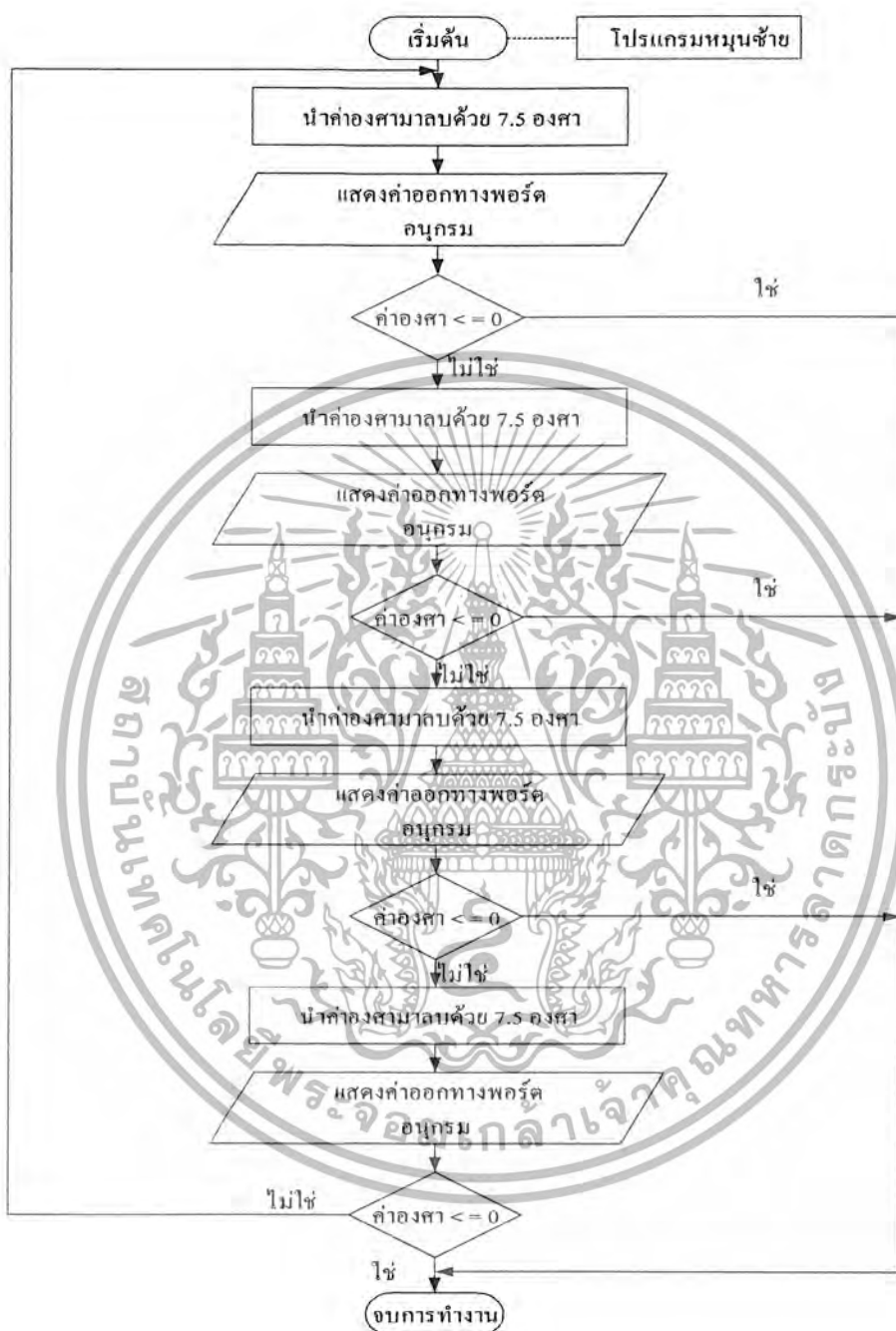


รูปที่ 3.13 ฟังก์ชันของโปรแกรมควบคุมสเต็ปมอเตอร์จากเครื่องเซิร์ฟเวอร์

## 2) การออกแบบโปรแกรมควบคุมสเต็ปมอเตอร์ผ่านเครื่องข่ายอินเทอร์เน็ต

มีหลักการทำงานคือให้ผู้ใช้งานในการควบคุมสเต็ปมอเตอร์ผ่านทางเว็บเพจ และนำข้อมูลที่ป้อนองศาขึ้นเก็บไว้ในฐานข้อมูลบนเครื่องเซิร์ฟเวอร์แล้วจึงใช้โปรแกรมบนเครื่องเซิร์ฟเวอร์ ดึงข้อมูลองศาจากฐานข้อมูลมาคำนวณว่ามีข้อมูลองศาของการหมุนแบบใด (หมุนซ้าย หมุนขวา หมุนขึ้นบน และหมุนลงล่าง) เมื่อได้รับค่าองศาของทิศทางการหมุนของสเต็ปมอเตอร์แล้ว จะมีโปรแกรมช่วยในการคำนวณค่าองศาและควบคุมการทิศหมุนของมอเตอร์ที่เลือกไว้ ส่งค่าออกไปควบคุมมอเตอร์ผ่านทางพอร์ตอนุกรม

## 2.1) โปรแกรมย่อยควบคุมสตีปิ้งมอเตอร์หมุนซ้าย



รูปที่ 3.14 ผังงานของ โปรแกรมย่อยควบคุมสตีปิ้งมอเตอร์หมุนซ้าย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## 2.2) โปรแกรมย่อยควบคุมสเต็มปี้งมอเตอร์หมุนขวา



รูปที่ 3.15 ผังงานของ โปรแกรมย่อยควบคุมสเต็มปี้งมอเตอร์หมุนขวา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.3) โปรแกรมย่อยควบคุมสตีปิ้งมอเตอร์หมุนบน



รูปที่ 3.16 ผลงานของ โปรแกรมย่อยควบคุมสตีปิ้งมอเตอร์หมุนบน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## 2.4) โปรแกรมย่อยควบคุมสเต็มปี้งมอเตอร์หมุนล่าง

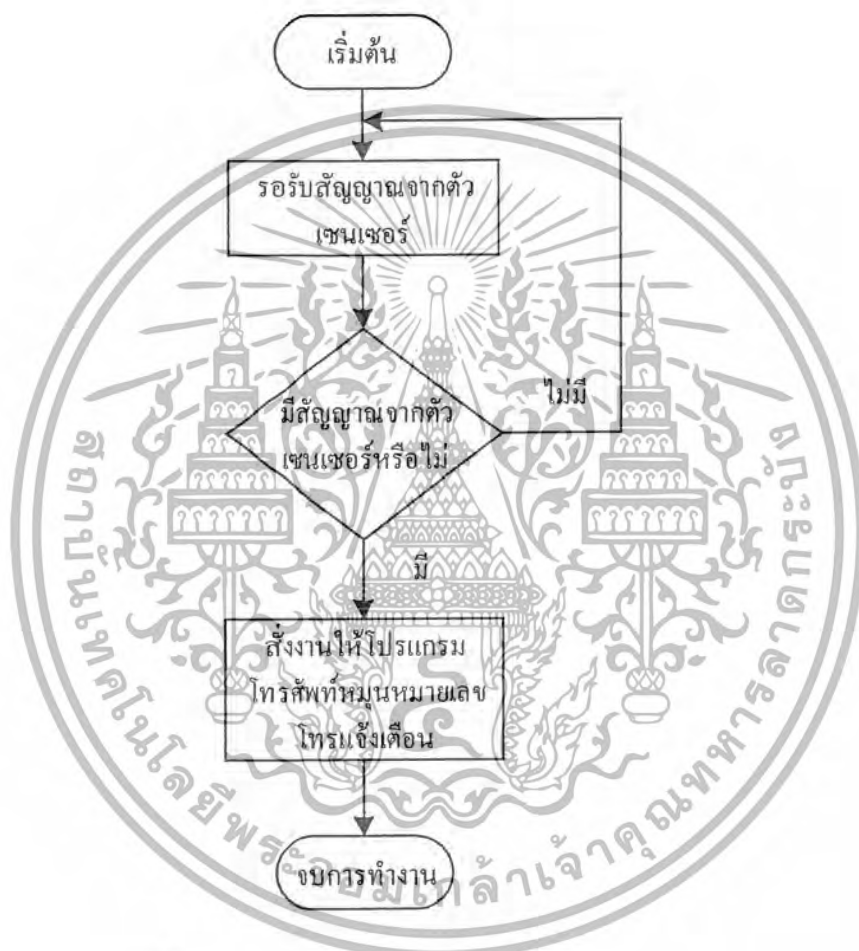


รูปที่ 3.17 ฟังงานของโปรแกรมย่อยควบคุมสเต็มปี้งมอเตอร์หมุนล่าง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 3.3.2 การออกแบบโปรแกรมตรวจสอบสัญญาณนิรภัย

หลักการการทำงานของ โปรแกรมคือคอยรับสัญญาณที่ได้จากตัวตรวจจับเข้ามายังคอมพิวเตอร์ เพื่อใช้ในการสั่งงานให้โมเด็มโทรแจ้งเตือนภัยตามหมายเลขที่ระบุไว้ โดยการรับสัญญาณนั้นจะรับผ่านพอร์ตอนุกรม คือเมื่อมีผู้บุกรุกเข้ามาและตัวตรวจจับสามารถตรวจจับได้ก็จะส่งสัญญาณมายังคอมพิวเตอร์โดยมีโปรแกรมเป็นตัวตรวจสอบสัญญาณที่เข้ามาแล้วสั่งงานให้โมเด็มเริ่มทำงาน

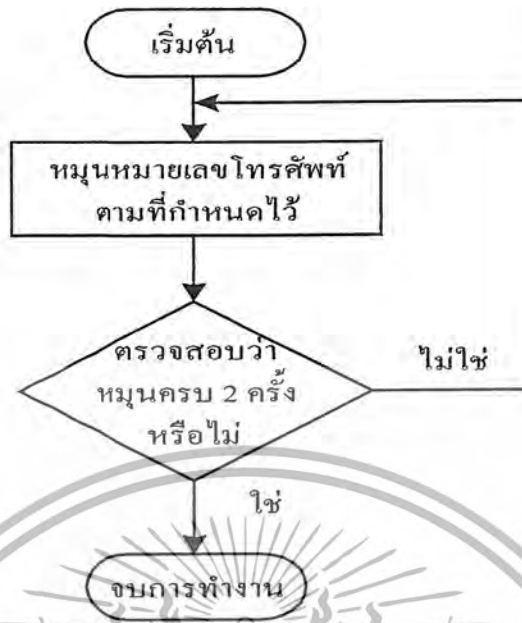


รูปที่ 3.18 ผังงานของ โปรแกรมตรวจสอบสัญญาณนิรภัย

### 3.3.3 การออกแบบโปรแกรมโทรศัพท์แจ้งเตือนภัย

หลักการการทำงานของ โปรแกรมนี้คือเมื่อได้รับคำสั่งจากโปรแกรมตรวจสอบสัญญาณนิรภัยว่าให้เริ่มทำงานได้ โปรแกรมโทรศัพท์แจ้งเตือนภัยก็จะทำงาน โดยจะโทรศัพท์ไปยังหมายเลขที่ระบุไว้ในตัวโปรแกรมตามลำดับที่ระบุไว้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.19 ผังงานของ โปรแกรมโทรศัพท์แจ้งเตือนภัย

### 3.3.4 การออกแบบการควบคุมกล้องผ่านอินเทอร์เน็ต

จากรูปผังการทำงานในส่วนของการควบคุมกล้องผ่านอินเทอร์เน็ตสามารถกระทำได้โดยการออกแบบเว็บเพจให้สามารถติดต่อกับฐานข้อมูล เมื่อมีการบันทึกค่าองศาที่ต้องการลงในฐานข้อมูลโดยการป้อนค่าผ่านทางเว็บเพจแล้วนั้น โปรแกรมควบคุมการหมุนของสเต็ปมอเตอร์ก็จะอ่านค่าองศาจากฐานข้อมูลออกมาคำนวณตลอดเวลา และตั้งงานควบคุมการหมุนของสเต็ปมอเตอร์ตามค่าองศาที่ปรากฏอยู่ในฐานข้อมูล



รูปที่ 3.20 ผังการทำงานการควบคุมกล้องผ่านอินเทอร์เน็ต

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 3.3.5 การออกแบบการส่งสัญญาณภาพผ่านเครือข่ายอินเทอร์เน็ต

จากผังการทำงานในส่วนของกล้องวิดีโอ นำกล้องวิดีโอต่อกับพอร์ต USB ในคอมพิวเตอร์ ในส่วนของคอมพิวเตอร์จะต้องมีโปรแกรมเอนโคเดออร์สัญญาณภาพ ถึงจะสามารถดูภาพผ่านพอร์ตUSBได้ (ในส่วนของทาง โปรแกรมวินโดวส์มีเดียเอนโคเดออร์นั้น จะสามารถหาดาวน์โหลดโปรแกรมได้ที่ [www.microsoft.com](http://www.microsoft.com)) เมื่อติดตั้งโปรแกรมวินโดวส์มีเดียเอนโคเดออร์เสร็จแล้วให้เปิดวินโดวส์มีเดียทำงานตลอดเวลาในส่วนของบล็อกเว็บเพจก็เขียนโปรแกรมภาษา HTML เพื่อที่จะดึงสัญญาณภาพที่ทำการเอนโคเดออร์แล้วมาดูผ่านเว็บเพจ



รูปที่ 3.21 ผังการทำงานการส่งสัญญาณภาพผ่านเครือข่ายอินเทอร์เน็ต



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## บทที่ 4

### การทดลอง และผลการทดลอง

#### 4.1 กล่าวนำ

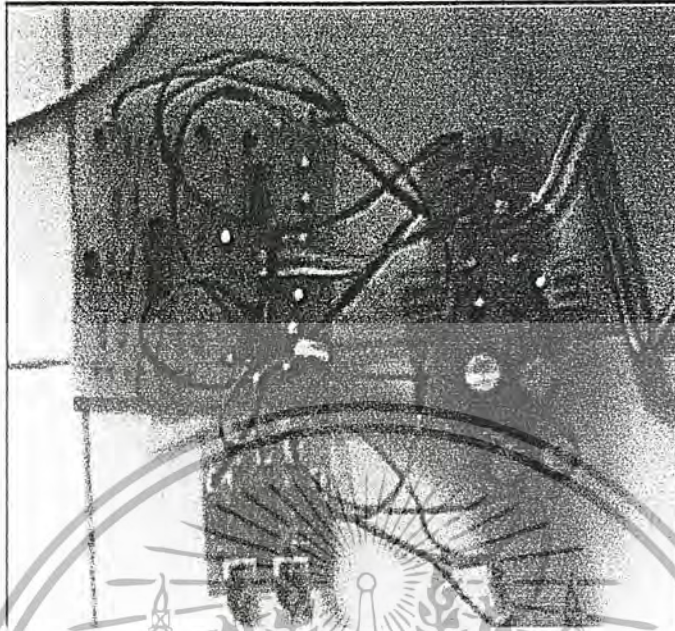
จากที่ได้กล่าวมาแล้วในบทที่ 3 เกี่ยวกับการสร้างและการออกแบบทั้งทางด้านฮาร์ดแวร์ และซอฟต์แวร์ ในบทนี้จะเป็นการทดลองและผลการทดลองตามที่ได้ออกแบบไว้ว่าในแต่ละวงจรของการทดลองนั้น มีผลการทดลองที่ถูกต้องหรือไม่ ดังนั้นในการทดลองจึงแยกออกเป็นการทดลองแต่ละวงจร ดังนี้

#### 4.2 การทดลองแผงวงจรควบคุมการทำงานหลัก

การรับส่งข้อมูลแบบอนุกรมผ่านพอร์ตอนุกรมของคอมพิวเตอร์ ตามมาตรฐาน RS-232 เป็นลักษณะการทำงานที่ใช้ในการรับส่งข้อมูลระหว่างเครื่องคอมพิวเตอร์กับแผงวงจรควบคุมการทำงานหลัก เนื่องจากว่าในเครื่องคอมพิวเตอร์มีส่วนของโปรแกรมควบคุมการหมุนของสเต็ปมอเตอร์ ดังนั้นเมื่อมีการป้อนองศาและกำหนดทิศทางการหมุนของสเต็ปมอเตอร์แล้วจึงจำเป็นต้องนำข้อมูลเหล่านั้นส่งออกไปควบคุมทิศทางการหมุนของสเต็ปมอเตอร์

##### 4.2.1 ลำดับขั้นการทดลอง

- 1) ทำการเชื่อมต่อแผงวงจรควบคุมการทำงานหลักเข้ากับเครื่องคอมพิวเตอร์ โดยการต่อเข้ากับพอร์ตอนุกรมซึ่งแสดงดังรูป 4.1
- 2) จ่ายแรงดันให้แก่แผงวงจรควบคุมการทำงานหลักจากวงจรภาคจ่ายไฟ
- 3) ทำการเรียกโปรแกรมควบคุมการหมุนของสเต็ปมอเตอร์ขึ้นมาทำงาน โปรแกรมนี้จะเขียนขึ้นจากภาษาวิซวลเบสิกโดยหน้าที่ของโปรแกรมนี้อีกคือ ทำการรับส่งข้อมูลผ่านพอร์ตอนุกรม RS-232 ส่งผ่านไปยังแผงวงจรควบคุมการทำงานหลักจากนั้นแผงวงจรควบคุมการทำงานหลักจะทำการส่งสัญญาณออกไปควบคุมอุปกรณ์ภายนอกโดยผ่านสายสัญญาณเพียงสองเส้นคือสายสัญญาณข้อมูลข้อมูลอนุกรม (Serial Data Line) : SDA และสายสัญญาณนาฬิกา (Serial Clock Line) : SCL
- 4) ทดลองป้อนองศาองในโปรแกรมควบคุมการหมุนของสเต็ปมอเตอร์แล้วคลิกเลือกว่าจะให้สเต็ปมอเตอร์หมุนไปทางด้านซ้าย ด้านขวา ด้านบนหรือด้านล่าง



รูปที่ 4.1 การทดลองแผงวงจรควบคุมการทำงานหลัก

#### 4.2.2 ผลการทดลอง

การทดลองแผงวงจรควบคุมการทำงานหลักให้ผลการทดลองที่ถูกต้องคือ เมื่อป้อนแรงดันไฟให้กับแผงวงจรควบคุมการทำงานหลักและเรียกโปรแกรมควบคุมการหมุนของสเต็ปปีงมอเตอร์ขึ้นมาใช้งานแล้ว ทำให้ไดโอดเปล่งแสงที่แสดงสถานะของขา CTS และ TxD ติดตลอดเวลา จากนั้นเมื่อทำการป้อนคำสั่งและทิศทางการหมุนของสเต็ปปีงมอเตอร์แล้ว ไดโอดเปล่งแสงที่แสดงสถานะการทำงานของขา DTR จะกระพริบตลอดเวลาที่มีการรับส่งข้อมูลระหว่างคอมพิวเตอร์กับแผงวงจรควบคุมการทำงานหลัก

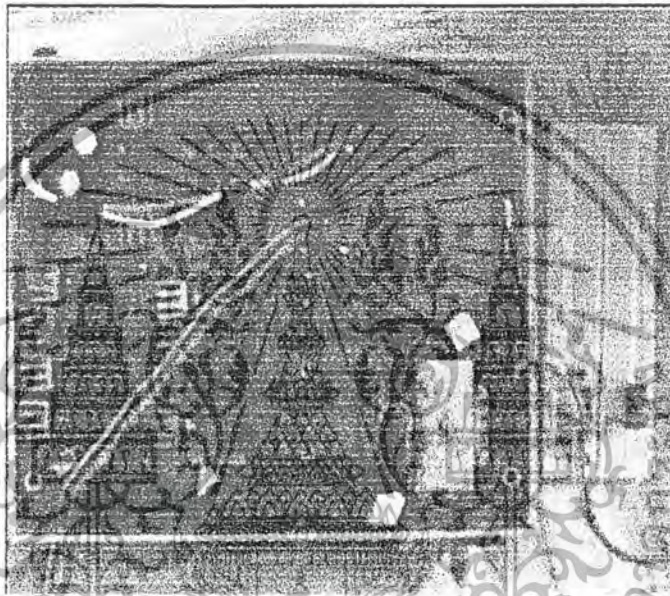
#### 4.3 การทดลองวงจรรขยายพอร์ตอินพุตเอาต์พุต

วงจรถูกประกอบด้วยไอซีขยายพอร์ตเบอร์ PCF8574 โดยไอซีเบอร์นี้สามารถขยายพอร์ตอินพุตเอาต์พุตได้ 8 ช่อง หรือ 8 บิต สามารถต่อพ่วงกันได้มากกว่าแปดตัวทำให้สามารถต่อพอร์ตอินพุตเอาต์พุตได้มากถึง 64 บิตรวมเป็น 128 บิต โดยการเชื่อมต่อนั้นต้องกระทำบนบัส I<sup>2</sup>C เท่านั้น การกำหนดให้ไอซี PCF8574 บนวงจรถูกนั้นทำได้โดยการกำหนดแอดเดรส (A0-A2) ซึ่งมีอยู่แล้วบนวงจรรขยายพอร์ตอินพุตเอาต์พุต แต่การกำหนดแอดเดรสของไอซี PCF8574 แต่ละตัวนั้นจะต้องกำหนดไม่ให้แอดเดรสแต่ละตัวมีค่าที่ตรงกัน ไม่เช่นนั้นจะเกิดความสับสนในการอ่านหรือเขียนข้อมูล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

#### 4.3.1 ลำดับขั้นตอนการทดลอง

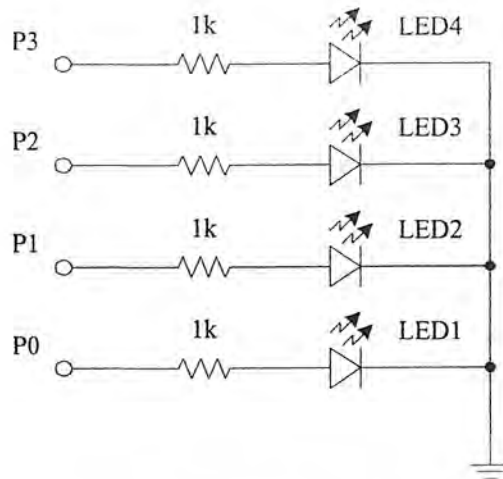
- 1) ทำการเชื่อมต่อแผงวงจรควบคุมการทำงานหลัก, เครื่องคอมพิวเตอร์ และวงจรถายพอร์ตอินพุตเอาต์พุตเข้าด้วยกันแสดงดังรูปที่ 4.2
- 2) จ่ายแรงดันให้กับแผงวงจรควบคุมการทำงานหลักและวงจรถายพอร์ตอินพุตเอาต์พุต จากวงจรภาคจ่ายไฟ



รูปที่ 4.2 การทดลองวงจรถายพอร์ตอินพุตเอาต์พุต

- 3) ต่อเอาต์พุตของวงจรถายพอร์ตอินพุตเอาต์พุต (P0-P7) เข้ากับตัวต้านทาน ที่มีค่าอยู่ที่ 1 กิโลโอห์ม ซึ่งต่ออนุกรมอยู่กับไดโอดเปล่งแสง ดังรูปที่ 4.3
- 4) ทำการเรียกโปรแกรมควบคุมการหมุนของสเต็ปปีงมอเตอร์ขึ้นมาทำงานบนเครื่องคอมพิวเตอร์ แล้วทดลองป้อนองศาลงในโปรแกรมควบคุมการหมุนของสเต็ปปีงมอเตอร์แล้วคลิกเลือกว่าจะให้สเต็ปปีงมอเตอร์หมุนไปทางด้านซ้าย ด้านขวา ด้านบนหรือด้านล่าง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.3 ชุดสถานะการทำงานการหมุนของสเต็ปมอเตอร์

#### 4.3.2 ผลการทดลอง

การทดลองการทำงานของวงจรขยายพอร์ตอินพุตเอาต์พุตให้ผลการทดลองที่ถูกต้อง โดยเมื่อเรียกโปรแกรมควบคุมการหมุนของสเต็ปมอเตอร์ขึ้นมาใช้งานบนคอมพิวเตอร์แล้ว ป้อนค่าองศาและเลือกทิศทางการหมุนจะทำให้ไดโอดเปล่งแสงติดและดับตามสเต็ปการหมุนของสเต็ปมอเตอร์ แล้วนำผลการทดลองมาบันทึกลงในตารางการทดลองได้ดังนี้

ตารางที่ 4.1 การทดลองวงจรขยายพอร์ตอินพุตเอาต์พุต

การหมุนของ สเต็ป	สถานะ LED1	สถานะ LED2	สถานะ LED3	สถานะ LED4
สถานะเริ่มต้น	-	-	-	-
สเต็ปที่ 1	ติด	-	-	-
สเต็ปที่ 2	-	ติด	-	-
สเต็ปที่ 3	-	-	ติด	-
สเต็ปที่ 4	-	-	-	ติด
สเต็ปที่ 5	ติด	-	-	-
สเต็ปที่ 6	-	ติด	-	-
สเต็ปที่ 7	-	-	ติด	-

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 4.1 (ต่อ) การทดลองวงจรขยายพอร์ตอินพุตเอาต์พุต

การหมุนของ สแต็ปป์	สถานะ LED1	สถานะ LED2	สถานะ LED3	สถานะ LED4
สแต็ปป์ที่ 8	-	-	-	ติด
สแต็ปป์ที่ 9	ติด	-	-	-
สแต็ปป์ที่ 10	-	ติด	-	-
สแต็ปป์ที่ 11	-	-	ติด	-
สแต็ปป์ที่ 12	-	-	-	ติด

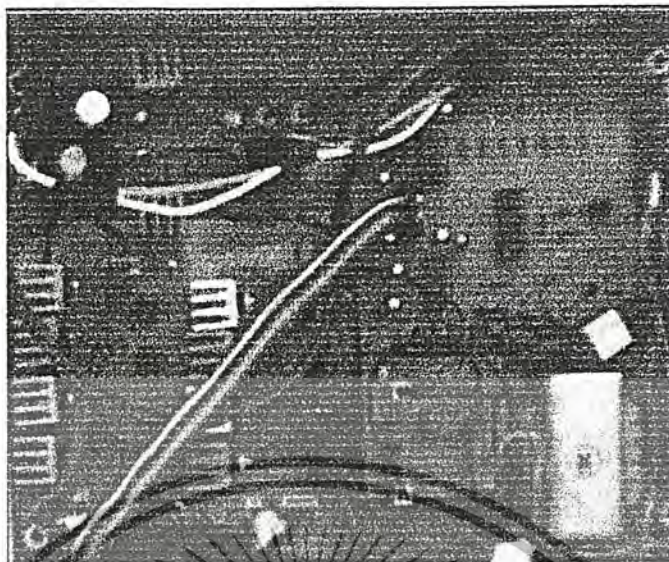
#### 4.4 การทดลองวงจรขับสแต็ปป์มอเตอร์

ในวงจรขับสแต็ปป์มอเตอร์ประกอบด้วยอุปกรณ์ที่ใช้ในการขับสแต็ปป์มอเตอร์แบบยูนิโพลาร์ ที่มีกระแสส่วนของอินพุตและเอาต์พุตออกจากกัน โดยใช้การเชื่อมต่อทางแสงหรือออปโตคัปเปลอร์ ทำให้ลดปัญหาสัญญาณรบกวนที่อาจผ่านมาจากกราวด์ของระบบและช่วยลดความเสี่ยงที่จะเกิดขึ้นกับคอมพิวเตอร์ โดยวงจรขับสแต็ปป์มอเตอร์จะทำงานร่วมกับแผงวงจรควบคุมการทำงานหลัก, วงจรขยายพอร์ตอินพุตเอาต์พุต และใช้โปรแกรมวิชวลเบสิกในการเขียนโปรแกรมเพื่อใช้ในการควบคุมการหมุนของสแต็ปป์มอเตอร์ซึ่งได้ทำการทดลองการทำงานดังนี้

##### 4.4.1 ลำดับขั้นตอนการทดลอง

- 1) ทำการเชื่อมแผงวงจรควบคุมการทำงานหลัก, เครื่องคอมพิวเตอร์, วงจรขยายพอร์ตอินพุตเอาต์พุตและวงจรขับสแต็ปป์มอเตอร์เข้าด้วยกันเพื่อทดลองการใช้งานวงจรขับสแต็ปป์มอเตอร์
- 2) จ่ายแรงดันไฟ +5V ให้กับแผงวงจรควบคุมการทำงานหลัก และวงจรขยายพอร์ตอินพุตเอาต์พุต ส่วนวงจรขับสแต็ปป์มอเตอร์ก็จ่ายแรงดันไฟ +12V จากวงจรภาคจ่ายไฟ
- 3) ต่อเอาต์พุตของวงจรขับสแต็ปป์มอเตอร์เข้ากับสแต็ปป์มอเตอร์แบบยูนิโพลาร์ แสดงดังรูปที่ 4.4
- 4) เรียกโปรแกรมควบคุมการหมุนของสแต็ปป์มอเตอร์ขึ้นมาทำงานบนเครื่องคอมพิวเตอร์แล้วทดลองป้อนองศาองในโปรแกรมควบคุมการหมุนของสแต็ปป์มอเตอร์แล้วคลิกเลือกว่าจะให้สแต็ปป์มอเตอร์หมุนไปทางด้านซ้าย ด้านขวา ด้านบนหรือด้านล่าง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.4 การทดลองวงจรขับสเต็ปปีงมอเตอร์

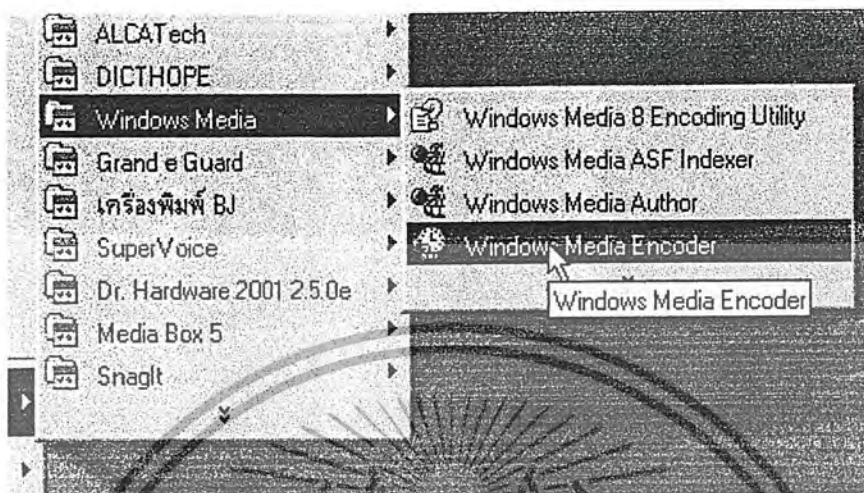
#### 4.4.2 ผลการทดลอง

การทดลองการทำงานของวงจรขับสเต็ปปีงมอเตอร์ให้ผลการทดลองถูกต้อง จากการทดลองป้อนค่าองศาและทิศทางการหมุนของสเต็ปปีงมอเตอร์ ลงในโปรแกรมควบคุมการหมุนของสเต็ปปีงมอเตอร์ ผลที่ได้คือ สเต็ปปีงมอเตอร์หมุนตามค่าองศาและทิศทางที่กำหนดลงไปในการทดลอง

#### 4.5 การทดลองการส่งสัญญาณภาพ

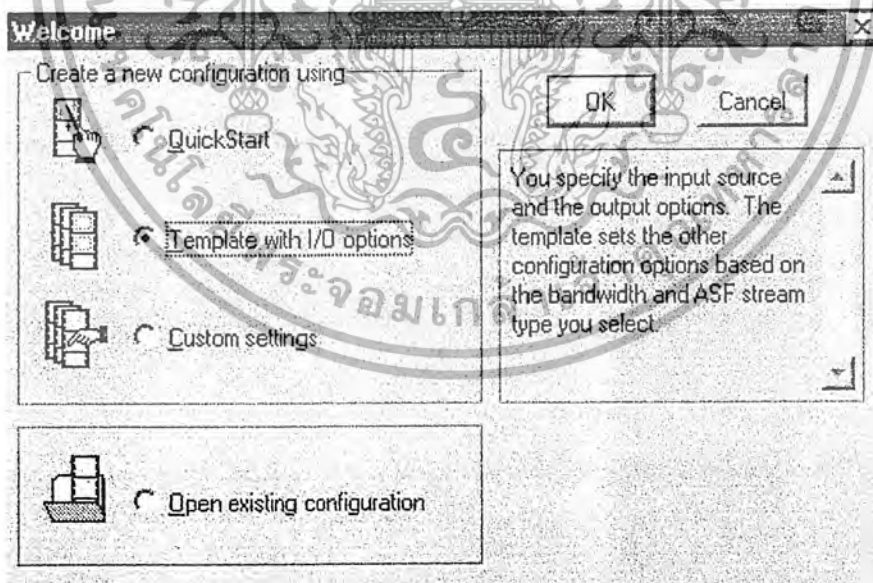
เมื่ออุปกรณ์ทางฮาร์ดแวร์พร้อมที่จะรับสัญญาณวิดีโอหรือเสียงแล้ว เราจะเริ่มทำการเข้ารหัสสัญญาณเหล่านั้นให้อยู่ในรูปของ Streaming File ในรูปแบบของ MPEG-4 โดยเรียกโปรแกรม Windows Media Encoder จาก Program Menu และเข้าสู่ขั้นตอนต่างๆ ดังนี้

### 1) เรียกโปรแกรม Windows Media Encoder



รูปที่ 4.5 การเรียกโปรแกรม Windows Media Encoder

### 2) ปรากฏภาพให้เลือก Template with I/O options แล้วกด OK

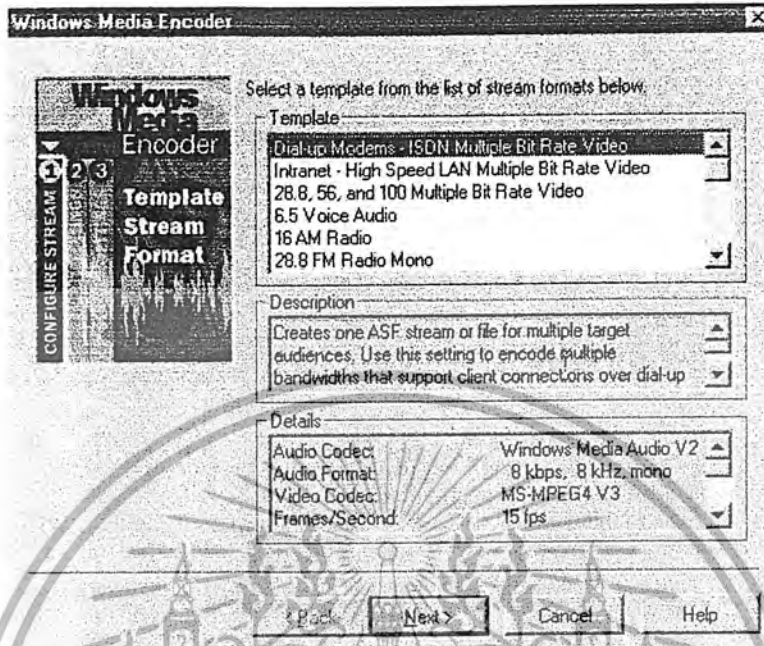


รูปที่ 4.6 การเลือกลักษณะการส่งออกของสัญญาณภาพ

### 3) ให้เลือกความเร็วชนิดของข้อมูลและแบนด์วิดท์ ที่ต้องการใช้ส่งผ่านอินเทอร์เน็ต แล้ว

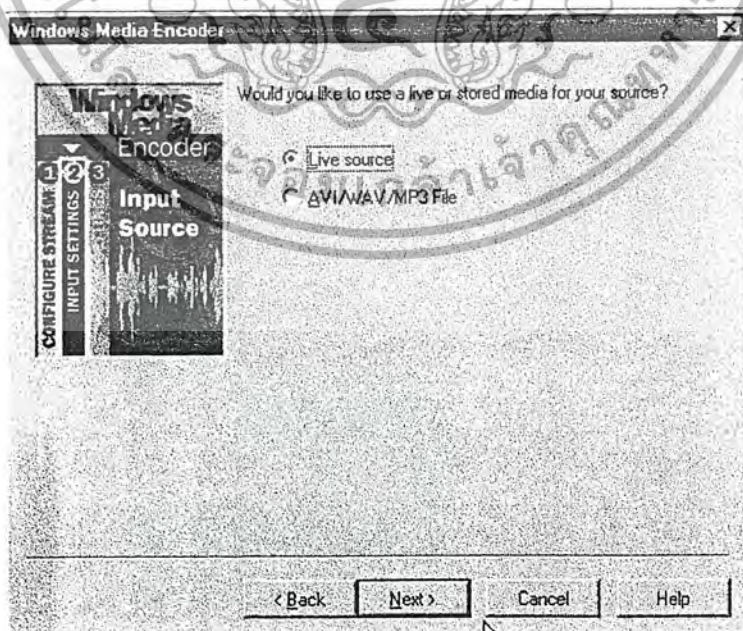
กด Next

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.7 การเลือกความเร็วขงข้อมูลและแบบคิวคัท

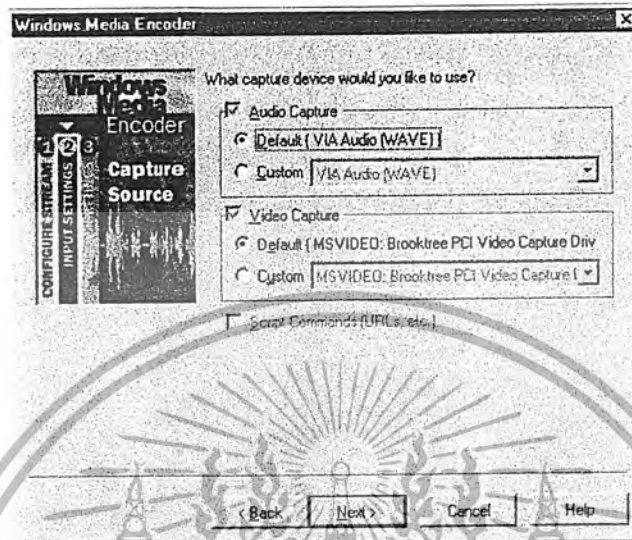
4) เลือกที่มาของแหล่งข้อมูล ถ้า Play Back จากเครื่องบันทึกเลือก Live Source



รูปที่ 4.8 การเลือกที่มาของแหล่งข้อมูล

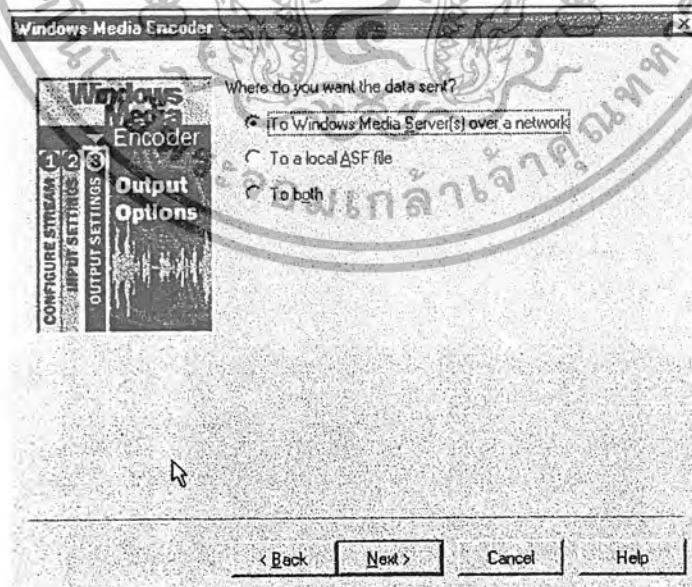
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไมอนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5) จะปรากฏไดรเวอร์ของฮาร์ดแวร์ที่เครื่องคอมพิวเตอร์มองเห็น Sound Card และ Video Card ให้กด Next ต่อไป



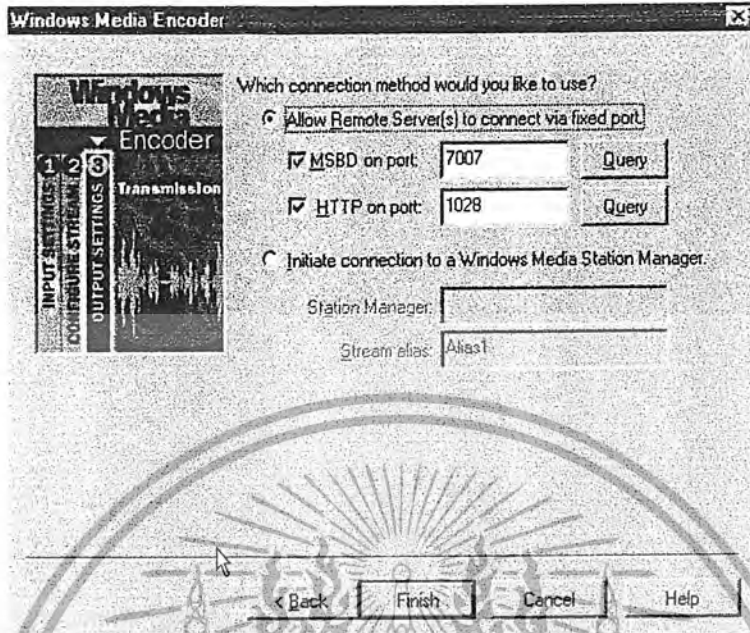
รูปที่ 4.9 รายละเอียดไดรเวอร์ที่คอมพิวเตอร์มองเห็น

6) ให้เลือกวิธีส่งควรเลือกเป็น To Both เพื่อให้สามารถใช้ได้ทั้ง Off-Line และ On-Line แล้วกด Next



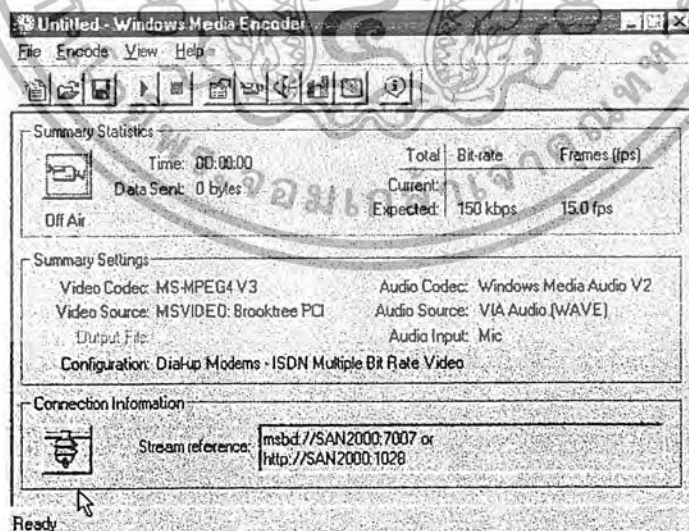
รูปที่ 4.10 การเลือกวิธีการส่ง

7) เมื่อมาถึงตรงนี้ก็กด Finish เพื่อเข้าสู่การเอนโคดเดอร์ ต่อไปได้ เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับบริการใช้งานเพื่อการศึกษาเท่านั้น ไม่นับญาติให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.11 จอการเลือกเพื่อเข้าสู่การเอน โคดเดอร์

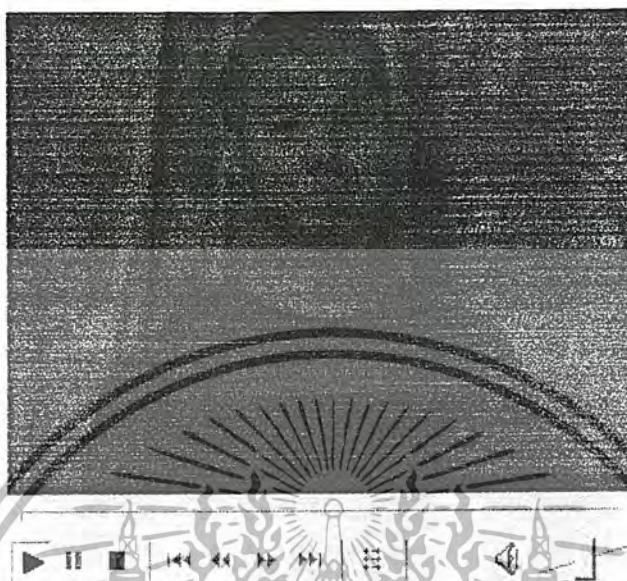
8) จากนั้นเข้าสู่โปรแกรมการเริ่มเอนโคดเดอร์จากแหล่งข้อมูลที่เตรียมไว้ เมื่อพร้อมที่จะเอนโคดเดอร์ให้คลิกปุ่ม  พร้อมกับเปิดเครื่องเล่นของแหล่งข้อมูล



รูปที่ 4.12 การเอนโคดเดอร์จากแหล่งข้อมูลที่เตรียมไว้

9) เมื่อทำการเอนโคดเดอร์ข้อมูลสัญญาณเสียงหรือสัญญาณวิดีโอแล้ว ให้ทำการบันทึก Sa ข้อมูลไว้ในโฟลเดอร์ที่ค้นหาได้ง่าย ข้อมูลที่ทำการบันทึกจะอยู่ในรูปแบบของ .ASF หรือ เอ็มสารันเป็นเอกสารที่ส่งบนเวปสำหรับการใช้งานเพื่อการศึกษาเท่านั้น เมื่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Advanced Streaming Format (ASF) เป็น Files Format ของ Multimedia ที่รวมเอา Video, Sound และ Script Commands ที่สามารถวิ่งผ่านเน็ตเวิร์คได้



รูปที่ 4.13 ตัวอย่างสัญญาณภาพที่ผ่านการเอนโคดเดอร์

#### 4.6 การทดลอง การใช้งานเว็บเพจติดต่อกับฐานข้อมูล และควบคุมสแต็ปมอเตอร์ผ่านเครือข่าย

เมื่อทำการทดลองวงจรควบคุมสแต็ปมอเตอร์เรียบร้อยแล้ว และเขียนเว็บเพจที่ใช้ในการสั่งงานการทำงานของสแต็ปมอเตอร์เสร็จเรียบร้อยแล้ว ก็จะต้องทำการทดลองการควบคุมสั่งงานสแต็ปมอเตอร์ โดยควบคุมผ่านทางเว็บเพจ โดยการติดต่อกับฐานข้อมูลที่ใช้ในการกำหนดทิศทางและองศาการหมุนของสแต็ปมอเตอร์ ต่อไป

##### 4.6.1 ลำดับขั้นการทดลอง

- 1) ทำการติดตั้งโปรแกรม PWS (Personal Web Server) เพื่อต้องการทำให้เครื่องคอมพิวเตอร์เป็น Server แบบจำลองทำให้เราสามารถใช้งานเหมือนกับ Server จริงๆ
- 2) ออกแบบหน้าเว็บเพจสำหรับผู้ใช้โดยใช้โปรแกรมภาษา Html
- 3) เขียนโปรแกรมภาษา ASP เพื่อใช้ในการติดต่อกับฐานข้อมูล ของส่วนควบคุมกล้อง, ฐานข้อมูลของตัวตรวจจับผู้บุกรุก, ฐานข้อมูลสำหรับตรวจเช็คองศาการหมุนของสแต็ปมอเตอร์ครั้งสุดท้าย และเขียนโปรแกรมเพื่อดึงสัญญาณภาพวีดีโอมายังเว็บเพจของผู้ใช้
- 4) เมื่อทำการเขียน โปรแกรม ติดต่อกับฐานข้อมูลเสร็จเรียบร้อยแล้ว ก็ทำการทดสอบการใช้งาน

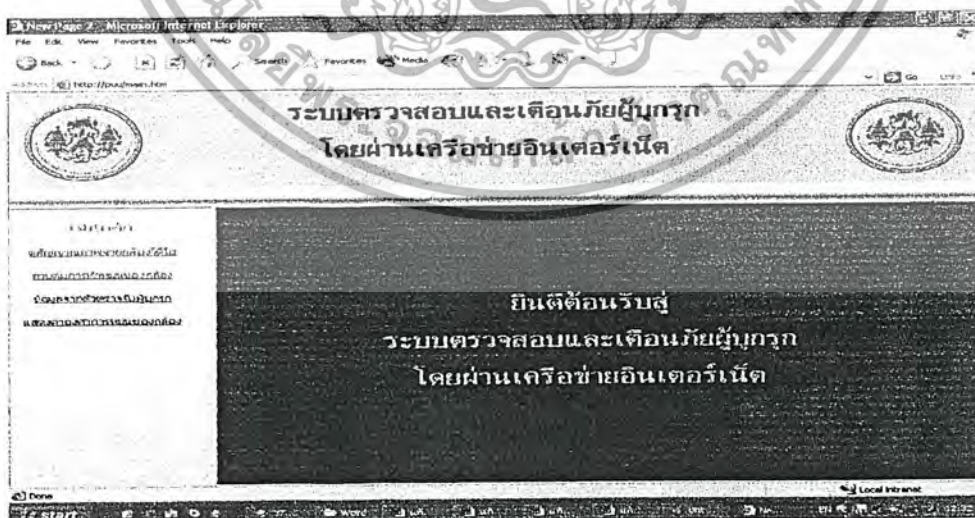
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

#### 4.6.2 ผลการทดลอง

1) ในการทดลองครั้งนี้สิ่งที่สำคัญไม่ควรลืมคือการติดตั้งโปรแกรม PWS ในเครื่องตนเอง และทำการเซ็ทระบบให้พร้อมสำหรับการใช้งานเสียก่อนจากนั้นทำการทดลองตามข้อที่ 2 ทำให้ได้รูปแบบเว็บเพจสำหรับผู้ใช้งานดังรูปที่ 4.14



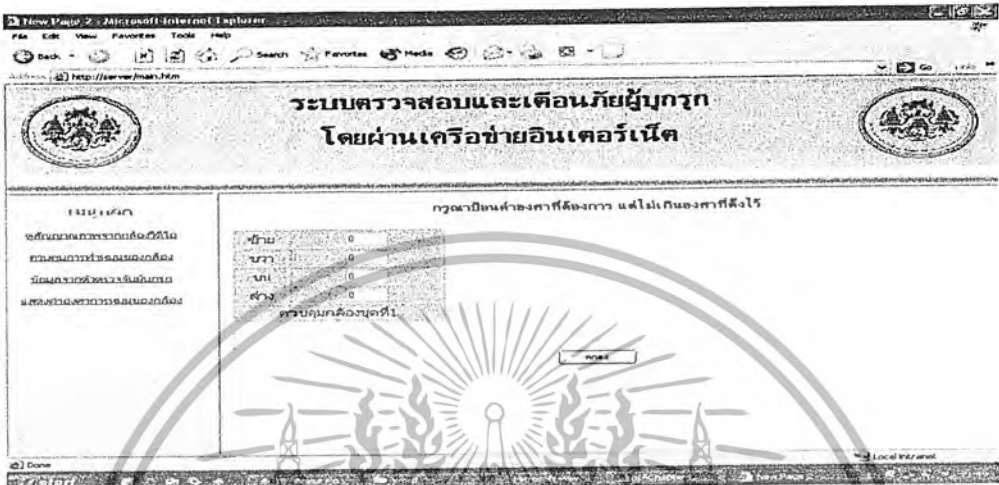
รูปที่ 4.14 เว็บเพจสำหรับผู้ใช้งาน



รูปที่ 4.15 หน้าหลักของเว็บเพจ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2) สำหรับการเขียนโปรแกรม ASP เมื่อเขียนเสร็จแล้ว ทำการทดสอบรันโปรแกรมผ่าน PWS ผลทำให้สามารถดึงข้อมูล และสามารถติดต่อกับฐานข้อมูลที่เราเขียนขึ้นมาได้ดังรูปที่ 4.16 รูปที่ 4.17 และรูปที่ 4.18 ตามลำดับ



รูปที่ 4.16 การควบคุมสติปิ้งมอเตอร์



รูปที่ 4.17 ข้อมูลของตัวตรวจจับผู้บุกรุก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่นอนุญาติให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ระบบตรวจสอบและเตือนภัยผู้บุกรุก  
โดยผ่านเครือข่ายอินเทอร์เน็ต

เมนูหลัก

- คลิกเพื่อดูภาพจากกล้องวิดีโอ
- สถานะการทำงานของกล้อง
- ข้อมูลจากตัวตรวจขโมยบุกรุก
- แสดงค่าองศาการหมุนของกล้อง

แสดงค่าองศาการหมุนของกล้อง	
ซ้าย	0
ขวา	0
บน	0
ล่าง	0
ควบคุมกล้องชุดที่ 1	

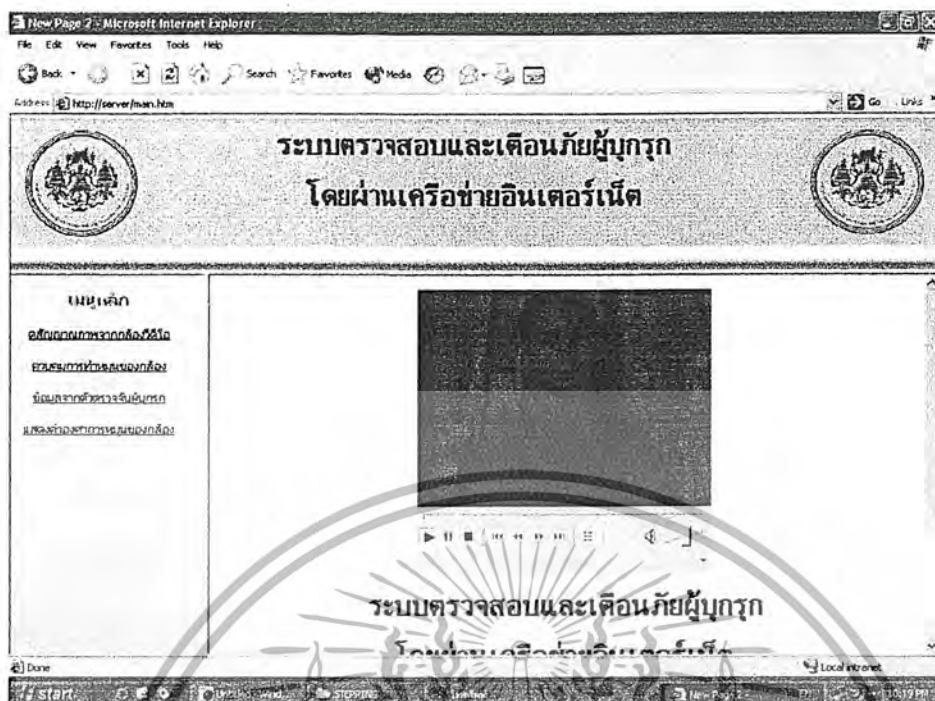
คลิกเพื่อดูภาพ  
กลับไปได้ออนไลน์

Local intranet

รูปที่ 4.18 ข้อมูลองศาการหมุนของสเต็ปปีงมอเตอร์

3) จากนั้นเป็นผลของการเขียนโปรแกรม เพื่อดึงภาพที่ผ่านการเอนโคคเตอร์ของโปรแกรม วินโดว์มีเดียมาร์นผ่านเว็บเพจได้ผลดังรูปที่ 4.19

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.19 ผลการรัน โปรแกรมวินโดวส์มีเดียเพื่อดูสัญญาณภาพผ่านเครือข่าย

4) ผลการทดลองการควบคุมสแต็ปปีงมอเตอร์ผ่านเครือข่ายจำลองคือ สามารถควบคุมการทำงานสแต็ปปีงมอเตอร์ให้หมุนซ้าย, ขวา, บน และล่าง ได้ โดยผ่านทางเว็บเพจของผู้ใช้

#### 4.7 การทดลองการเขียนฐานข้อมูลโดยใช้ โปรแกรม Microsoft Access

ในการสั่งงานให้กล้องทำงานนั้น จะต้องมีฐานข้อมูลเป็นส่วนประกอบเพื่อใช้เป็นข้อมูลในการทำงาน นอกจากนั้นยังใช้ในการติดต่อกับสัญญาณตรวจจับผู้บุกรุกและอื่นๆ โดยฐานข้อมูลนี้จะติดต่อโดยการ ใช้โปรแกรมวิซวลเบสิกในการติดต่อ

##### 4.7.1 ลำดับขั้นการทดลอง

1) ออกแบบและเขียนฐานข้อมูลสำหรับการควบคุมกล้อง เพื่อให้จะให้เหมาะสมกับการใช้งาน และสามารถเขียน โปรแกรมวิซวลเบสิก ติดต่อกับฐานข้อมูลได้

2) ออกแบบและเขียนฐานข้อมูล เพื่อติดต่อกับสัญญาณตรวจจับผู้บุกรุก และใช้โปรแกรมวิซวลเบสิก ติดต่อกับฐานข้อมูลเช่นกัน

3) ออกแบบและเขียนฐานข้อมูลของเบอร์โทรศัพท์ ที่ใช้ในการสั่งให้โมเด็มติดต่อโทรศัพท์อัตโนมัติ

4) ออกแบบและเขียนฐานข้อมูล สำหรับการตรวจเช็คกองสการหมุนของสแต็ปปีงมอเตอร์ครั้งสุดท้าย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

#### 4.7.2 ผลการทดลอง

1) เมื่อทำการเขียนฐานข้อมูลสำหรับส่วนควบคุมกล้องเสร็จแล้วจะได้รูปแบบฐานข้อมูลดังรูปที่ 4.20

CODE	L1	R1	T1	U1	L2	R2	T2	U2
	0	0	0	0	0	0	0	0
*	0	0	0	0	0	0	0	0

รูปที่ 4.20 ฐานข้อมูลของส่วนควบคุมกล้อง

2) เมื่อทำการเขียนฐานข้อมูลสำหรับส่วนติดต่อกับสัญญาณตรวจจับผู้บุกรุกเสร็จแล้วจะได้รูปแบบฐานข้อมูลดังรูปที่ 4.21

DANG	T1	T2	T3	T4	T5	T
	ไม่มีผู้บุกรุก ณ จุดที่	ไม่มีผู้บุกรุก ณ จุดที่	ไม่มีผู้บุกรุก ณ จุดที่	ไม่มีผู้บุกรุก ณ จุดที่	ไม่มีผู้บุกรุก ณ จุดที่	ไม่มีผู้บุกรุก
*						

รูปที่ 4.21 ฐานข้อมูลของส่วนติดต่อกับสัญญาณตรวจจับผู้บุกรุก

3) เมื่อเขียนฐานข้อมูลเบอร์โทรศัพท์เสร็จแล้วจะได้รูปแบบฐานข้อมูลดังรูปที่ 4.22

TEL1	TEL2	TEL3	TEL4	TEL5	TIMER	USED
027390982,2	1	2	4	5	30	1
*						

รูปที่ 4.22 ฐานข้อมูลเบอร์โทรศัพท์

4) เมื่อเขียนฐานข้อมูลตรวจเช็คกองสการหมุนของสเต็ปปีงมอเตอร์ จะได้รูปแบบฐานข้อมูลดังรูปที่ 4.23

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

CODE	L1	R1	T1	U1	L2	R2	T2	U2	L3	R3	T3	U3	L4	R4	T4	U4
1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

รูปที่ 4.23 ฐานข้อมูลสำหรับตรวจเช็คคองสาของสตีปีมอเตอร์

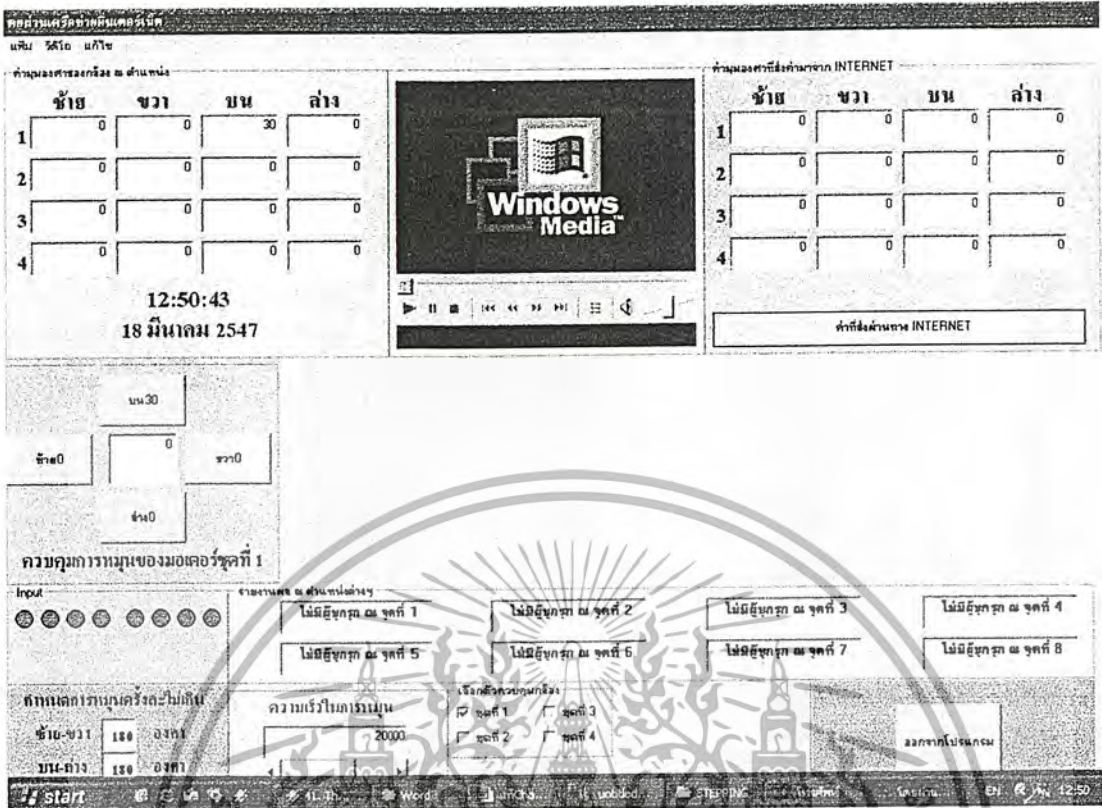
#### 4.8 การทดลองการใช้งาน โปรแกรมระบบตรวจสอบและเตือนภัยผู้บุกรุกโดยผ่าน เครือข่ายอินเทอร์เน็ต

โปรแกรมระบบตรวจสอบและเตือนภัยผู้บุกรุกโดยผ่านเครือข่ายอินเทอร์เน็ต เป็นโปรแกรมที่ทำงานบนระบบปฏิบัติการวินโดวส์ เปิดโปรแกรมได้โดยการสั่งงาน โปรแกรมวิซวลเบสิก ซึ่งเราสามารถเข้าสู่โปรแกรมได้โดยการสั่งงานโปรแกรมที่ชื่อ STEPPING ก่อนจะเข้าฟอร์มหลักของโปรแกรม ต้องป้อนรหัสผู้ใช้ก่อนดังรูปที่ 4.24 จากนั้นก็จะเข้าสู่ฟอร์มหลักของโปรแกรม STEPPING ดังรูปที่ 4.25



รูปที่ 4.24 เมนูการป้อนรหัสผู้ใช้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.25 รูปแบบหลักของโปรแกรม

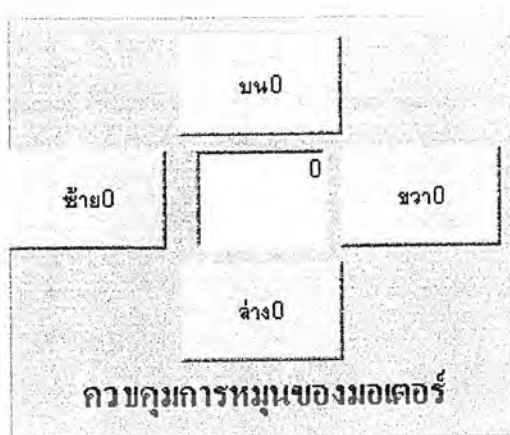
#### 4.9 การทดลอง โปรแกรมควบคุมการหมุนของสเต็ปปีงมอเตอร์

การทดลองโปรแกรมสั่งงานควบคุมการหมุนของสเต็ปปีงมอเตอร์ จะประกอบไปด้วย สเต็ปปีงมอเตอร์ 2 ตัวด้วยกันที่ใช้งานในการควบคุมการหมุนของกล้อง 1 ตัว คือสเต็ปปีงมอเตอร์ สำหรับหมุนด้านซ้าย - ขวา 1 ตัว หมุนบน - ล่าง 1 ตัว ซึ่งการสั่งงานโปรแกรมควบคุมการหมุนของสเต็ปปีงมอเตอร์ให้ทำงานในแต่ละชุดนั้นจะมีหลักการการทำงานเหมือนกัน ต่างกันแต่เพียงการตั้งค่าตำแหน่งของไอซี PCF8574 เท่านั้น

##### 4.9.1 ลำดับขั้นการทดลอง

- 1) เลือกโปรแกรมสั่งงานควบคุมการหมุนของสเต็ปปีงมอเตอร์ชุดที่ต้องการ
- 2) ทำการป้อนองศาการหมุนของกล้องไม่เกินค่าองศาตามที่ตั้งไว้
- 3) เลือกทิศทางการหมุนของกล้องหมุนซ้าย, ขวา, บน และล่าง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่นิยมนำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.26 โปรแกรมตั้งงานควบคุมการหมุนของสเต็ปปีงมอเตอร์

#### 4.9.2 ผลการทดลอง

จากการทดลองเมื่อป้อนค่าองศาและเลือกทิศทางการหมุนไปของกล้องไปในทิศทางที่ต้องการ ผลที่ได้คือ สเต็ปปีงมอเตอร์สามารถหมุนตามทิศทางที่เลือกและหมุนตามองศาที่ป้อนไว้ โดยโปรแกรมจะลดค่าองศาจากที่ป้อนไว้ลงครึ่งละ 7.5 องศาจนมีค่าเท่ากับศูนย์ สเต็ปปีงมอเตอร์ก็จะหยุดหมุน

สรุปว่าผลการทดลองโปรแกรมตั้งงานควบคุมการหมุนของสเต็ปปีงมอเตอร์สามารถนำไปใช้ในการควบคุมการหมุนของสเต็ปปีงมอเตอร์ได้



รูปที่ 4.27 การเลือกทิศทางการหมุนและการลดค่าของมุมองศา

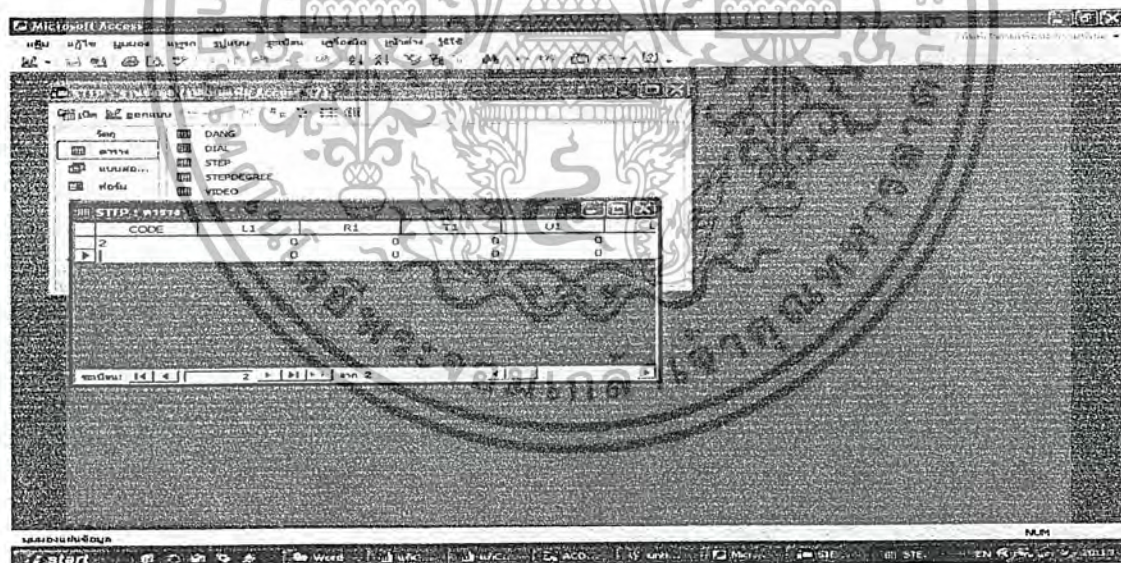
#### 4.10 การทดลอง โปรแกรมควบคุมการหมุนของสเต็ปปีงมอเตอร์โดยใช้ค่าองศาจากฐานข้อมูล

การทดลองโปรแกรมตั้งงานควบคุมการหมุนของสเต็ปปีงมอเตอร์โดยใช้ค่าองศาจากฐานข้อมูล ควบคุมการหมุนของกล้องจะประกอบไปด้วยสเต็ปปีงมอเตอร์ 2 ตัวด้วยกันที่ใช้ในการควบคุมการหมุนของกล้องคือ สเต็ปปีงมอเตอร์สำหรับหมุนด้านซ้ายขวา 1 ตัว หมุนบนล่าง 1 ตัว เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ซึ่งการสั่งงาน โปรแกรมควบคุมการหมุนของสเต็ปปีงมอเตอร์ให้ทำงานนั้นสามารถกำหนดตำแหน่งของ IC PCF8574 เท่านั้น โดยในการทดลองนี้ จะใช้ค่าองศาจากฐานข้อมูลซึ่งใช้ฐานข้อมูลเป็น Microsoft Access แทนการป้อนองศาโดยตรง ค่าขององศาที่ได้จากฐานข้อมูล Microsoft Access นั้นได้มาจากการส่งค่าผ่านระบบอินเตอร์เน็ตมาเก็บไว้ยังฐานข้อมูล

#### 4.10.1 ลำดับขั้นการทดลอง

- 1) ป้อนค่าลงในฐานข้อมูล Microsoft Access ชื่อฐานข้อมูล STEP ในตาราง STEP ป้อนค่าองศาที่ต้องการลงในเซลล์ของทิศทางการหมุนของมอเตอร์ดังรูปที่ 4.28
- 2) สังเกตการเปลี่ยนแปลงของโปรแกรมสั่งงานควบคุมการหมุนของสเต็ปปีงมอเตอร์โดยใช้ค่าองศาจากฐานข้อมูลว่าได้รับค่ามาจากฐานข้อมูลหรือไม่ ดังรูปที่ 4.29
- 3) สังเกตผลว่าหากโปรแกรมสั่งงานควบคุมการหมุนของสเต็ปปีงมอเตอร์ ได้รับค่าจากฐานข้อมูลแล้วมีการสั่งงานให้มอเตอร์หมุนตามค่าองศาที่ได้รับมาจากฐานข้อมูลหรือไม่
- 4) สังเกตผลโปรแกรมสั่งงานควบคุมการหมุนของสเต็ปปีงมอเตอร์ ว่ามีการลดค่าองศาที่ได้จากฐานข้อมูลลงจนเท่ากับศูนย์หรือไม่



รูปที่ 4.28 ฐานข้อมูล Microsoft Access ชื่อฐานข้อมูล STEP ตาราง STEP

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

INTERNET				
	ซ้าย	ขวา	บน	ล่าง
1	0	0	0	0
2	0	0	0	0
3	0	0	0	0
4	0	0	0	0

ค่าที่ส่งผ่านทาง INTERNET

รูปที่ 4.29 โปรแกรมตั้งงานควบคุมการหมุนของสเต็ปปีงมอเตอร์โดยใช้ค่าองศาจากฐานข้อมูล

#### 4.10.2 ผลการทดลอง

จากการทดลองเมื่อป้อนค่าองศาในฐานข้อมูล ผลที่ได้ปรากฏว่าโปรแกรมตั้งงานควบคุมการหมุนของสเต็ปปีงมอเตอร์โดยใช้ค่าองศาจากฐานข้อมูล สามารถใช้ค่าที่ป้อนลงไป ในฐานข้อมูลมาควบคุมการหมุนของสเต็ปปีงมอเตอร์ได้ คือเมื่อมีค่าองศาในฐานข้อมูลในทิศทางการหมุนใด โปรแกรมก็จะสั่งงานให้สเต็ปปีงมอเตอร์หมุนตามทิศทางที่ค่าองศาปรากฏอยู่โดยหมุนตามค่าขององศาที่อยู่ในฐานข้อมูล โดยจะทำการตั้งงานให้สเต็ปปีงมอเตอร์หมุนและลดค่าองศาที่อยู่ในฐานข้อมูลลงครั้งละ 7.5 องศาจนค่าองศามีค่าเท่ากับศูนย์สเต็ปปีงมอเตอร์จึงหยุดหมุน สรุปได้ว่าผลการทดลองโปรแกรมตั้งงานควบคุมการหมุนของสเต็ปปีงมอเตอร์โดยใช้ค่าองศาจากฐานข้อมูลสามารถนำไปใช้งานในการควบคุมการหมุนของสเต็ปปีงมอเตอร์ได้โดยใช้ค่าองศาการหมุนจากฐานข้อมูล

INTERNET				
	ซ้าย	ขวา	บน	ล่าง
1	315	0	0	0
2	0	0	0	0
3	0	0	0	0
4	0	0	0	0

ค่าที่ส่งผ่านทาง INTERNET

รูปที่ 4.30 โปรแกรมตั้งงานควบคุมการหมุนของสเต็ปปีงมอเตอร์โดยแสดงค่าองศาจากฐานข้อมูลและแสดงการลดค่าองศา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

#### 4.11 การทดลองการตรวจสอบค่าองศาการหมุนของสเต็ปปีงมอเตอร์

การทดลองการตรวจสอบค่าองศาการหมุนของสเต็ปปีงมอเตอร์เพื่อให้ทราบค่าองศาของทิศทางการหมุนของสเต็ปปีงมอเตอร์ว่าขณะนั้นสเต็ปปีงมอเตอร์หมุนอยู่ ณ ตำแหน่งทิศทางใดอยู่ที่องศาที่เท่าใด

##### 4.11.1 ลำดับขั้นการทดลอง

- 1) เลือกโปรแกรมสั่งงานควบคุมการหมุนของสเต็ปปีงมอเตอร์ ดังรูปที่ 4.31
- 2) ทำการป้อนองศาการหมุนของกล้องไม่เกินค่าองศาตามที่ตั้งไว้เลือกทิศทางการหมุนซ้าย
- 3) เลือกทิศทางการหมุนของกล้องหมุนซ้าย



##### 4.11.2 ผลการทดลอง

จากการทดลองป้อนค่าองศาและเลือกทิศทางการหมุนของสเต็ปปีงปรากฏว่าจะมีค่าองศาการหมุนของสเต็ปปีงมอเตอร์ปรากฏในตารางฐานข้อมูลในช่องทิศทางการหมุนทางซ้าย สรุปได้ว่าการตรวจสอบมุมมองศาที่ว่าจะขณะนั้นกล้องหมุนอยู่ ณ ตำแหน่งไหน อยู่ในทิศทางใด

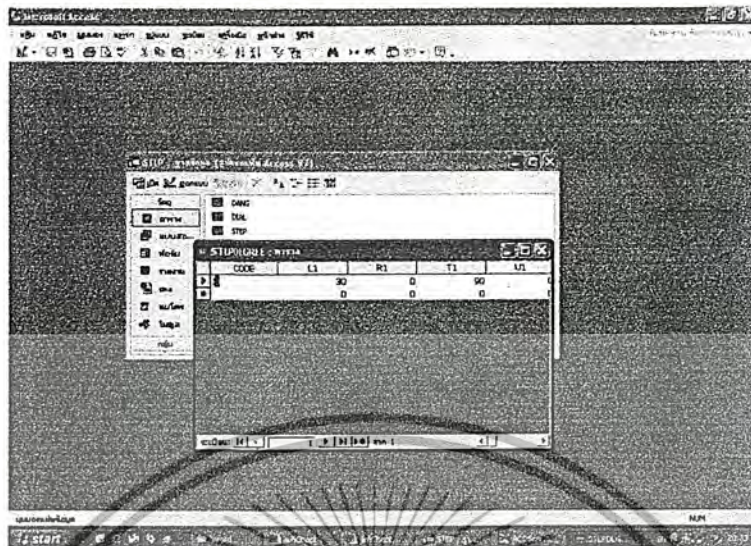
ค่ามุมองศาของกล้อง ณ ตำแหน่ง

	ซ้าย	ขวา	บน	ล่าง
1	0	45	0	60
2	0	0	0	0
3	0	0	0	0
4	0	0	0	0

3:09:11  
29 มีนาคม 2547

#### รูปที่ 4.32 ผลค่าองศาการหมุนของสเต็ปปีงมอเตอร์ที่อยู่ในตารางฐานข้อมูล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



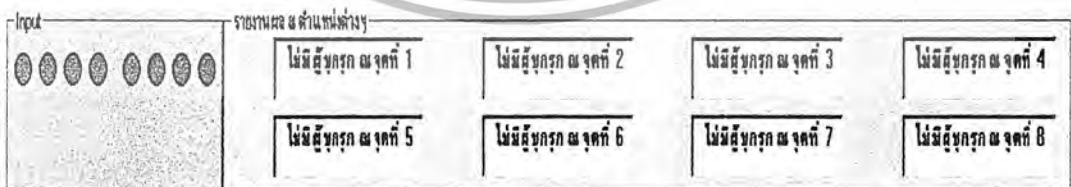
รูปที่ 4.33 ผลค่าองค์การหมุนของสตีปิงมอเตอร์ที่อยู่ในฐานข้อมูล Microsoft Access

#### 4.12 การทดสอบโปรแกรมตรวจสอบการทำงานของอุปกรณ์เซนเซอร์

การทดสอบโปรแกรมตรวจสอบการทำงานของอุปกรณ์เซนเซอร์เพื่อให้ทราบว่าสามารถรับค่าจากตัวอุปกรณ์เซนเซอร์เข้ามาได้หรือไม่

##### 4.12.1 ลำดับขั้นการทดสอบ

- 1) ต่อสัญญาณจากตัวเซนเซอร์กับวงจรชุดอินเตอร์เฟสเพื่อเชื่อมต่อเข้ากับคอมพิวเตอร์
- 2) เปิดโปรแกรมตรวจสอบการทำงานของอุปกรณ์เซนเซอร์
- 3) ทำให้ตัวเซนเซอร์ทำงาน
- 4) สังเกตการแสดงผลของโปรแกรม

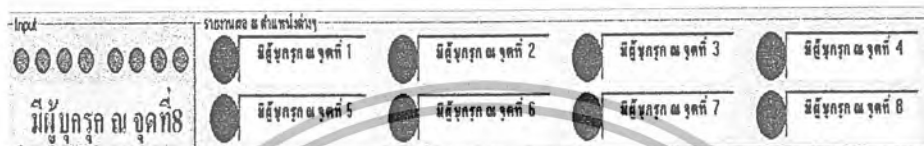


รูปที่ 4.34 โปรแกรมตรวจสอบการทำงานของอุปกรณ์เซนเซอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

#### 4.12.2 ผลการทดลอง

จากการทดลองผลที่ได้คือ เมื่อตัวเซนเซอร์ทำงาน โปรแกรมตรวจสอบการทำงานของอุปกรณ์เซนเซอร์จะมีการแสดงผลว่าตัวเซนเซอร์ตัวใดทำงานบ้าง โดยจะแสดงผลเป็นรูปวงกลมทึบจากสีเขียวอ่อนจะเปลี่ยนเป็นสีเขียวเข้มและแสดงรูปวงกลมสีแดงทึบขึ้นมา สรุปผลการทดลองได้ว่าโปรแกรมตรวจสอบการทำงานของอุปกรณ์เซนเซอร์สามารถทำการตรวจสอบสัญญาณที่มาจากตัวเซนเซอร์ได้



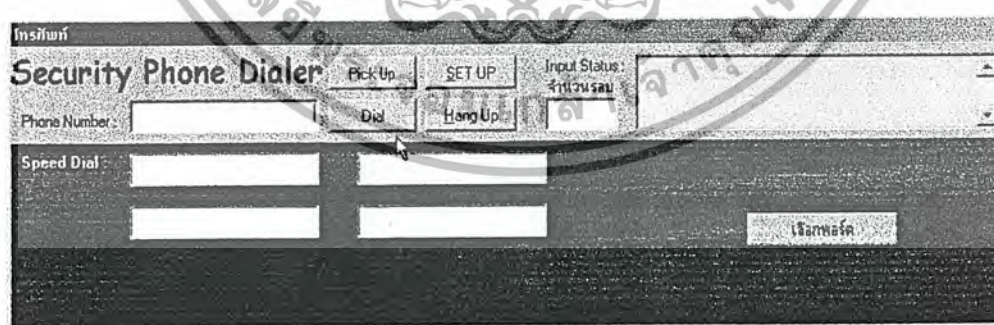
รูปที่ 4.35 ผลของโปรแกรมตรวจสอบการทำงานของอุปกรณ์เซนเซอร์

#### 4.13 การทดลองโปรแกรมสั่งงานโมเด็ม

การทดลองโปรแกรมสั่งงานโมเด็มเพื่อให้ทราบว่าโมเด็มสามารถหมุนหมายเลขโทรศัพท์ตามที่ตั้งไว้ได้หรือไม่

##### 4.13.1 ลำดับขั้นการทดลอง

- 1) ป้อนหมายเลข โทรศัพท์ในช่องหมายเลขโทรศัพท์
- 2) กดปุ่ม Dial
- 3) สังเกตผลที่ได้ว่ามีการหมุนหมายเลขโทรศัพท์หรือไม่

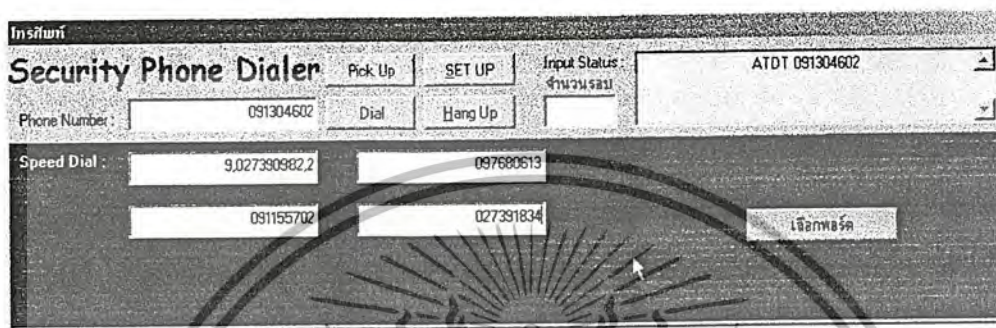


รูปที่ 4.36 โปรแกรมสั่งงาน โมเด็ม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

#### 4.13.2 ผลการทดลอง

จากการทดลองเมื่อป้อนเบอร์โทรศัพท์และกดปุ่ม Dial แล้วปรากฏว่าโปรแกรมจะทำการหมุนหมายเลขโทรศัพท์ตามเบอร์โทรศัพท์ที่ป้อนไว้ และแสดงผลการต่อสาย ATDT ตามด้วยหมายเลขที่ป้อนไว้ สรุปได้ว่าโปรแกรมสั่งงานโมเด็มสามารถสั่งงานโมเด็มให้หมุนโทรศัพท์ตามหมายเลขที่ป้อนไว้ได้



รูปที่ 4.37 การทำงานของโปรแกรมเมื่อมีการป้อนหมายเลขและกดปุ่ม Dial

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## บทที่ 5

### บทสรุป ปัญหา แนวทางแก้ไขและพัฒนา

#### 5.1 บทสรุป

ปริญญานิพนธ์ฉบับนี้เสนอผลงานเกี่ยวกับระบบตรวจสอบและเตือนภัยผู้บุกรุกโดยผ่านเครือข่ายอินเทอร์เน็ต โดยใช้การสื่อสารข้อมูลแบบอนุกรมผ่านพอร์ตอนุกรมของเครื่องคอมพิวเตอร์ตามมาตรฐานพอร์ตอนุกรม RS-232 ซึ่งเป็นมาตรฐานอุตสาหกรรมที่ออกแบบมาเพื่อใช้ในการส่งข้อมูลแบบอะซิงโครนัส 2 ทิศทาง โดยมีการใช้งานเชื่อมต่อกับอุปกรณ์บัส I<sup>2</sup>C ซึ่งมีการตั้งงานและควบคุมการทำงานภายใต้สายสัญญาณเพียงสองเส้น คือ สายข้อมูลและสายสัญญาณนาฬิกา ซึ่งอุปกรณ์ที่ทำหน้าที่ในการเชื่อมต่อกับอุปกรณ์ต่าง ๆ บนบัส I<sup>2</sup>C มีหลากหลาย ในปริญญานิพนธ์เล่มนี้จะใช้ไอซีขยายพอร์ตอินพุตเอาต์พุต เบอร์ PCF8574 ในการเชื่อมต่อกับอุปกรณ์ภายนอก ซึ่งระบบตรวจสอบและเตือนภัยผู้บุกรุกโดยผ่านเครือข่ายอินเทอร์เน็ตสามารถแจ้งเตือนเจ้าของผ่านโทรศัพท์เมื่อมีผู้บุกรุก และสามารถดูภาพเหตุการณ์ที่เกิดขึ้นในขณะนั้นได้ โดยสามารถดูผ่านระบบเครือข่ายอินเทอร์เน็ตและยังสามารถควบคุมทิศทางการหมุนของกล้องได้สลับโดยกล้องนั้นสามารถควบคุมทิศทางให้หมุนไปทางด้านซ้าย ด้านขวา ด้านบนและด้านล่างได้ ทำให้สามารถมองเห็นภาพได้ทุกทิศทางตามที่ต้องการหรือดูภาพเหตุการณ์ย้อนหลังในฮาร์ดดิสก์ที่มีการบันทึกเอาไว้แบบอัตโนมัติ จึงทำให้เจ้าของสามารถแก้ไขสถานการณ์ได้ทัน่วงทีและมีภาพบันทึกไว้เป็นหลักฐานในกรณีที่มีผู้ประสงค์ร้ายบุกรุกเพื่อขโมยทรัพย์สินที่มีค่าหรือก่อเหตุร้ายขึ้น นอกจากนี้ยังสามารถนำมาประยุกต์ใช้ภายในบ้านเรือนได้ เพราะอุปกรณ์ในการผลิตมีขายทั่วไปตามท้องตลาดราคาถูก ติดตั้งได้ง่าย ทำให้ระบบตรวจสอบและเตือนภัยผู้บุกรุกโดยผ่านเครือข่ายอินเทอร์เน็ตสามารถผลิตได้เองในประเทศไทยและเป็นประโยชน์มากสำหรับผู้ที่ต้องการความปลอดภัยในทรัพย์สินของท่านที่มีอยู่ในบ้านเรือนและสำนักงาน

#### 5.2 ปัญหาและแนวทางแก้ไข

จากการทดลองการทำงานของระบบตรวจสอบและเตือนภัยผู้บุกรุก โดยผ่านเครือข่ายอินเทอร์เน็ต สามารถทำงานได้ตามวัตถุประสงค์ที่กำหนดเอาไว้ แต่จากการทดลองทำให้ทราบปัญหาที่เกิดขึ้นซึ่งสามารถสรุปเป็นข้อๆ ได้ดังนี้

1) ปัญหา ในการทดลองภาพที่ได้จากกล้อง เมื่อวัตถุอยู่ในระยะไกลจะทำให้มองเห็นภาพได้ไม่ชัดเจน

แนวทางแก้ไข ใช้กล้องที่มีคุณภาพสูง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2) ปัญหา ในการทดลองภาพที่ได้จะมีการสั่นสะเทือน ภาพไม่นิ่ง  
แนวทางแก้ไข ใช้สเต็ปป์มอดเตอร์ที่มีสเต็ปการหมุน 1.8 องศาต่อสเต็ป ซึ่งมีความละเอียดของการหมุนเท่ากับ 200 ตำแหน่ง จะทำให้ภาพที่ได้มีคุณภาพมากขึ้น

3) ปัญหา การหน่วงเวลาของภาพ เมื่อดูผ่านเครื่องถ่ายอินเตอร์เน็ต จะมีการหน่วงเวลาของภาพประมาณ 15 – 20 วินาที ทำให้มองเห็นภาพที่เกิดขึ้นจริงในขณะนั้นช้ากว่าปกติ

แนวทางแก้ไข เปลี่ยนมาใช้ระบบการสื่อสารแบบ ISDN หรือ ใช้ระบบการส่งสัญญาณผ่านดาวเทียม

4) ปัญหา กระแสที่ขาเอาต์พุตของบอร์ดขยายอินพุตเอาต์พุต ไม่เพียงพอที่จะทำให้ออปโตคัปเปลอร์ในบอร์ดขับเคลื่อนสเต็ปป์มอดเตอร์ทำงาน ทำให้สเต็ปป์มอดเตอร์ไม่หมุน

แนวทางแก้ไข เปลี่ยนค่าตัวต้านทานพูลอัป ที่วงจรถยายพอร์ตอินพุตเอาต์พุตจาก 10 กิโลโอห์ม มาเป็น 5 กิโลโอห์ม

5) ปัญหา การเขียนโปรแกรมสั่งงาน โมเด็ม ไม่สามารถเขียนให้ส่งสัญญาณเสียงผ่านทางโมเด็มได้จึงทำให้เกิดปัญหาในการรับทราบข้อมูลเหตุการณ์ที่เกิดขึ้น

แนวทางแก้ไข สอบถามผู้รู้ ศึกษาจากหนังสือต่างๆ ค้นหาข้อมูลจากอินเตอร์เน็ตเรื่องการส่งสัญญาณเสียงผ่าน โมเด็ม

6) ปัญหา การเขียนโปรแกรมการสั่งงาน โมเด็ม ไม่สามารถเขียนให้หมุนเรียกหมายเลขโทรศัพท์ได้หลายเบอร์

แนวทางแก้ไข ศึกษาการเขียนโปรแกรมสั่งงาน โมเด็มจากหนังสือ สอบถามผู้รู้ หรือค้นหาข้อมูลจากอินเตอร์เน็ต

7) ปัญหา เมื่อดูสัญญาณภาพจากเครื่องหลักและสั่งงานให้กล้องหมุน ปรากฏว่าโปรแกรมสั่งงานสเต็ปป์มอดเตอร์จะไม่ทำงาน แต่เมื่อปิดการดูสัญญาณภาพแล้วปรากฏว่าโปรแกรมสั่งงานสเต็ปป์มอดเตอร์ทำงานตามปกติ

แนวทางแก้ไข ดูสัญญาณภาพโดยใช้คอมพิวเตอร์อีกเครื่องหนึ่ง โดยให้เครื่องหลักทำงานเฉพาะสั่งงานควบคุมทิศทางการหมุนของกล้องและรับสัญญาณจากอุปกรณ์เซนเซอร์ หรือให้สั่งงานหมุนองศาของกล้องก่อนแล้วจึงมาดูสัญญาณภาพที่ปรากฏ

### 5.3 แนวทางการพัฒนาโครงการ

โครงการที่สร้างขึ้นนี้สามารถทำงานได้ตามขีดความสามารถที่กำหนดเอาไว้แล้ว แต่อย่างไรก็ตามยังสามารถที่จะเพิ่มขีดความสามารถในการทำงานของโครงการนี้อีกดังต่อไปนี้

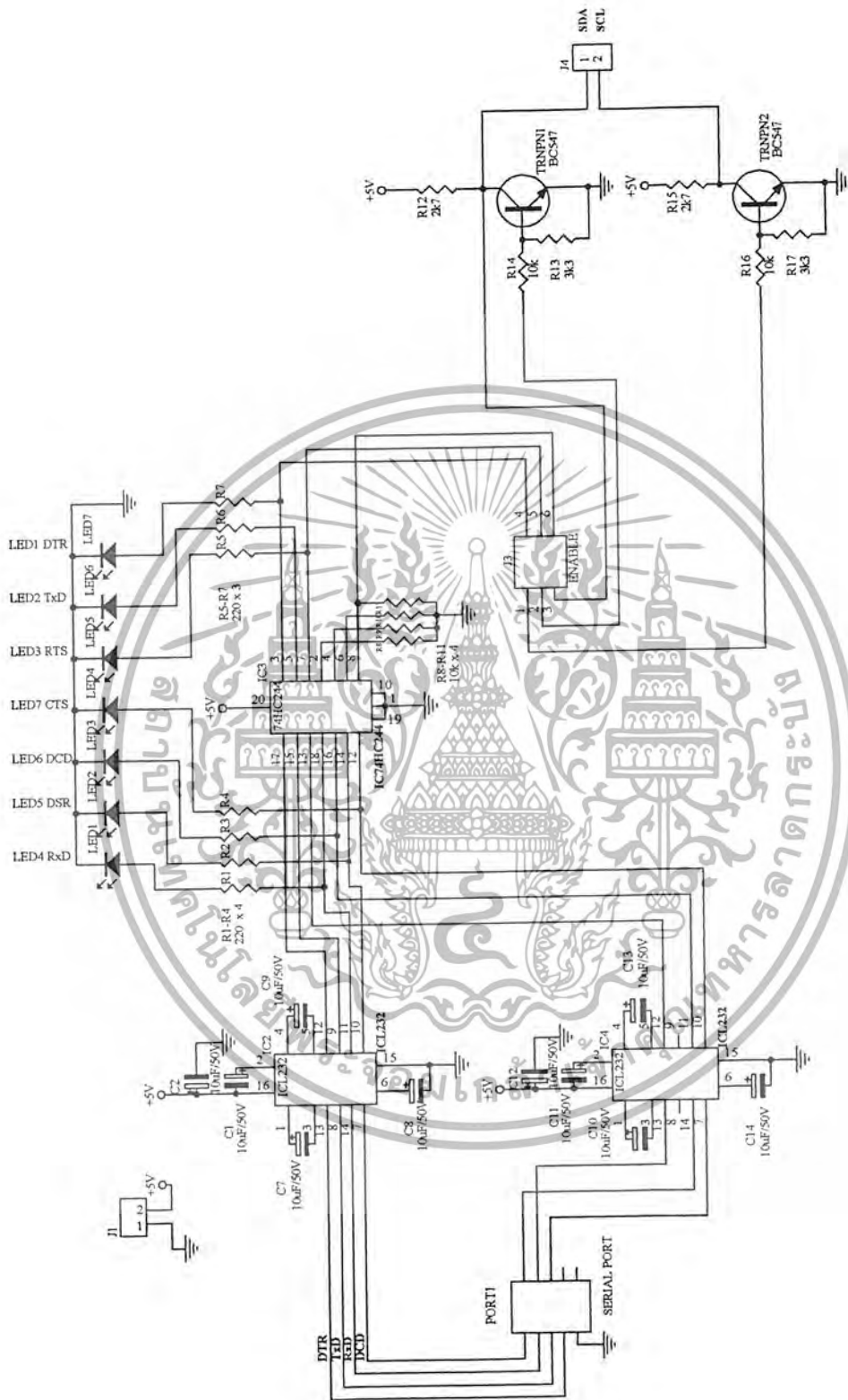
- 1) สามารถต่อใช้งานกับกล้องได้เพิ่มขึ้นมากกว่า 1 ตัว
- 2) สามารถสั่งให้โทรศัพท์ โทรส่งเสียงที่บันทึกเอาไว้ เพื่อแจ้งเตือนให้เจ้าของทราบว่าขณะนี้ มีผู้บุกรุกแทนการแสดงผลเลข โทรศัพท์เพียงอย่างเดียว

จากแนวคิดที่กลุ่มผู้จัดทำนำเสนอให้พิจารณา หากมีผู้ใดสนใจและต้องการศึกษาระบบการทำงานของโครงการนี้คือ ไปทางกลุ่มผู้จัดทำมีความยินดีเป็นอย่างยิ่งที่จะช่วยเหลือและแนะนำด้วยดีและหวังว่าโครงการนี้คงเป็นชิ้นงานที่ทำงานได้อย่างมีประสิทธิภาพมากขึ้นหากได้รับการพัฒนา



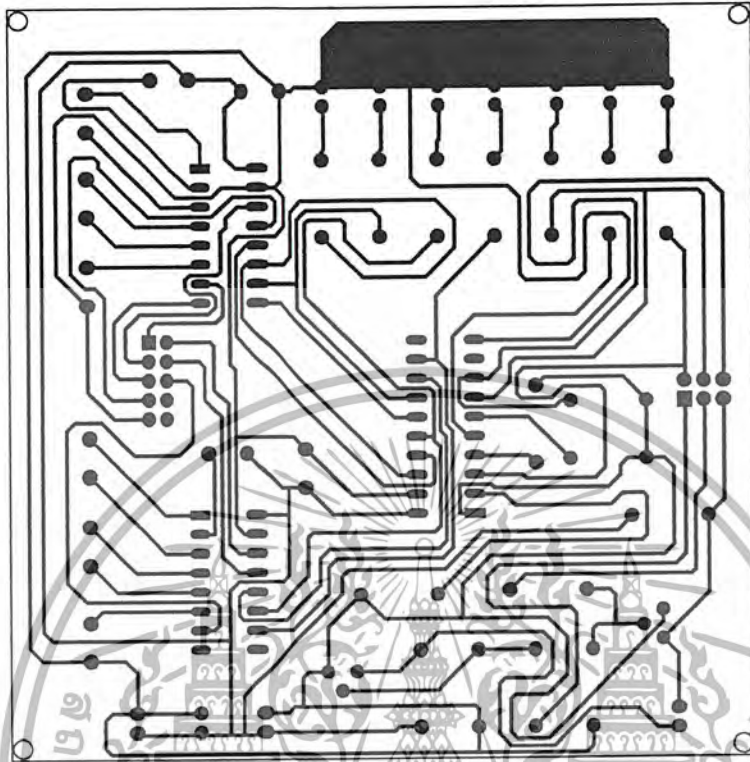


เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

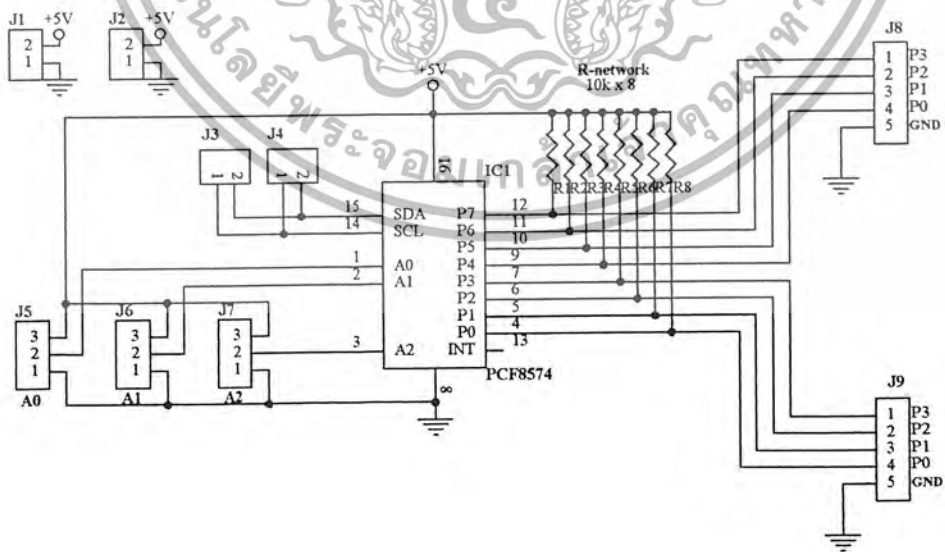


รูป วงจรควบคุมการทำงานหลัก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

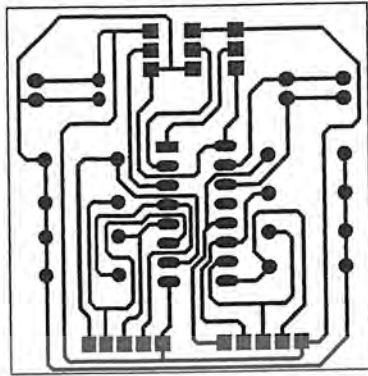


รูปแผ่นวงจรพิมพ์วงจรควบคุมการทำงานหลัก

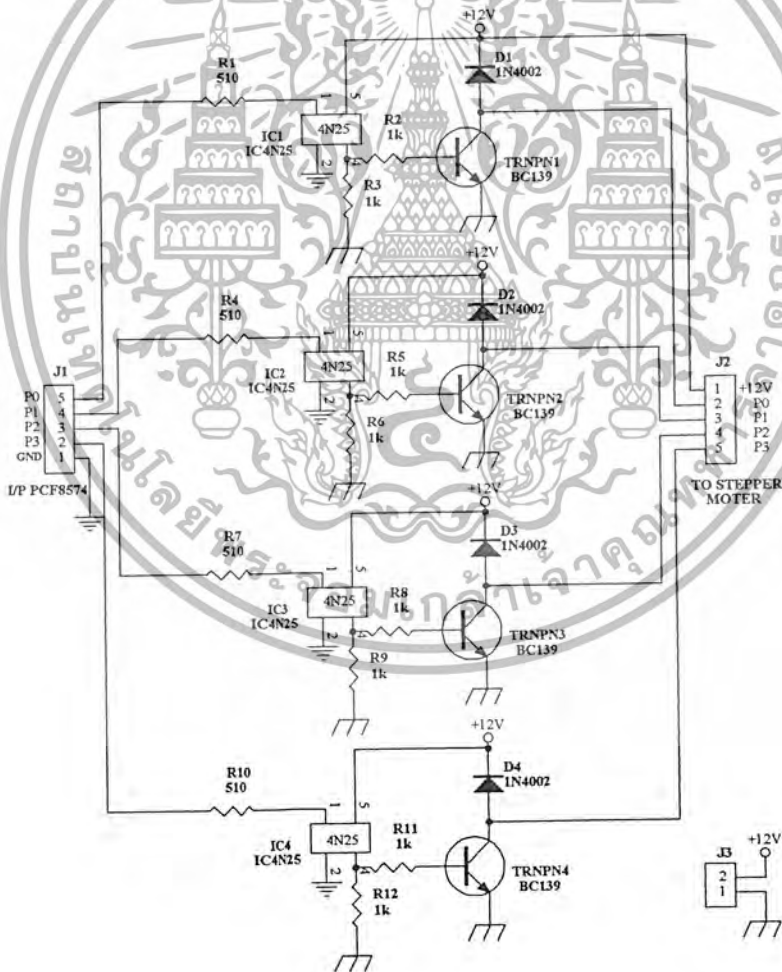


รูปวงจรมายจ่ายพอร์ตอินพุตเอาต์พุต

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

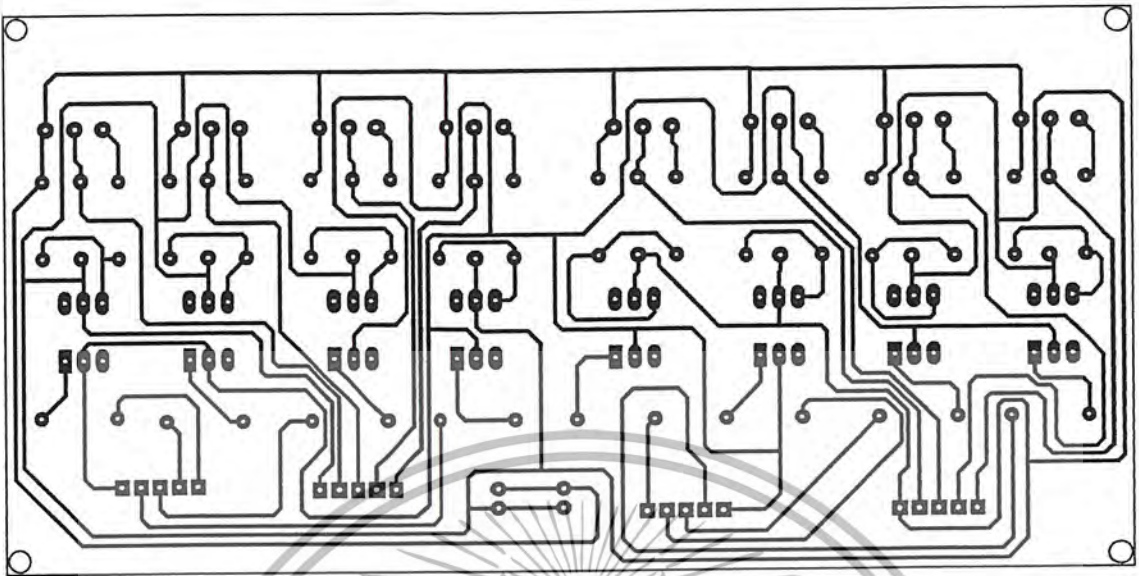


รูปแผ่นวงจรพิมพ์วงจรขยายพอร์ตอินพุตเอาต์

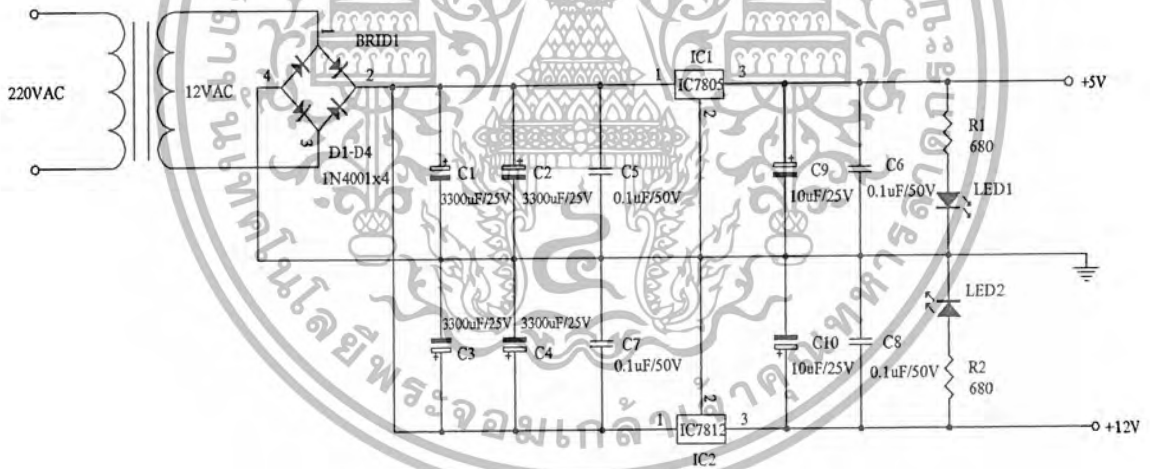


รูป วงจรขับสเต็ปิ่งมอเตอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

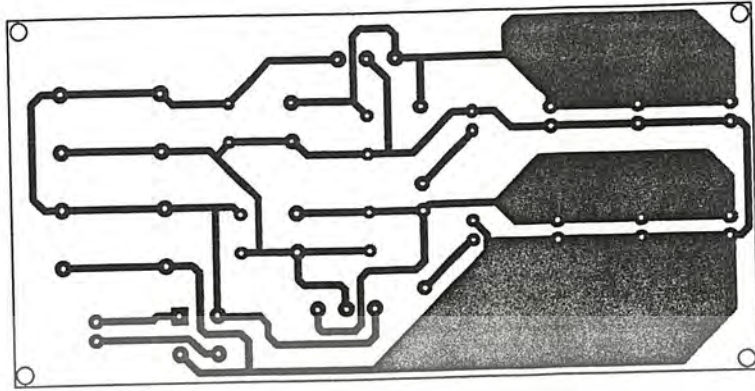


รูปที่ ก.6 แผงวงจรพิมพ์วงจรจับสแต็ปิงมอเตอร์



รูปวงจรแหล่งจ่ายไฟ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปแผ่นวงจรพิมพ์วงจรแหล่งจ่ายไฟ



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

Dim K, DA, PAT, Chip As Integer
Dim A, B, C, D, E, F, G As Integer
Dim XX As Integer
Dim AA, BB, CC As Integer
Dim TXT As String
Dim LENGTH As Integer
Dim CHECK As String
Dim BBA, BBB, BBC, BBD As Integer
Private Sub Cha4()
Dim DB, DC, DD, DE, DF, DG, DH, DI, DJ, DK, DL, DM, DN As Double
DJ = Val(KA.Text)
DB = Val(T41.Text)
DC = Val(T42.Text)
DD = Val(T43.Text)
DE = Val(T44.Text)
If DB >= DC Then
DF = DB - DC
DL4.Text = DF
C13.Caption = "ซ้าย" & DF
C14.Caption = "ขวา" & 0
ElseIf DC >= DB Then
DG = DC - DB
DR4.Text = DG
C14.Caption = "ขวา" & DG
C13.Caption = "ซ้าย" & 0
End If
If DD >= DE Then
DH = DD - DE
DT4.Text = DH
C15.Caption = "บน" & DH
C16.Caption = "ล่าง" & 0
ElseIf DE >= DD Then
DI = DE - DD
DU4.Text = DI
C16.Caption = "ล่าง" & DI
C15.Caption = "บน" & 0
End If

If DF >= DJ Then
C13.Enabled = False
Else
C13.Enabled = True
End If
If DG >= DJ Then
C14.Enabled = False
Else
C14.Enabled = True
End If
If DH >= DJ Then
C15.Enabled = False
Else

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

C15.Enabled = True
End If
If DI >= DJ Then
C16.Enabled = False
Else
C16.Enabled = True
End If
End Sub

```

```

Sub delay()
For I = 1 To HScroll11.Value
DoEvents
Next I
End Sub

```

```

Private Sub C10_Click()
Chip = &H74
If T32.Text = "" Then
T32.Text = Val(T30.Text)
Else
AA = Val(T32.Text)
BB = Val(T30.Text)
T32.Text = Val(AA + BB)
End If

```

```

G = 0
A = Val(T30.Text)
Do
DoEvents
A = A
If A > 0 Then
Call Sendout(7)
Call delay
B = A - 7.5
T30.Text = B
Else
B = 0
A = 0
End If
B = B
If B > 0 Then
Call Sendout(11)
Call delay
D = B - 7.5
T30.Text = D
Else
D = 0
A = 0
End If
D = D
If D > 0 Then
Call Sendout(13)
Call delay

```



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

E = D - 7.5
T30.Text = E
Else
E = 0
A = 0
End If
E = E
If E > 0 Then
Call Sendout(14)
Call delay
F = E - 7.5
A = F
T30.Text = F
G = G + 1
Text3.Text = G
Else
E = 0
A = 0
End If
A = A
Loop Until A < 0
End Sub

Private Sub C11_Click()
Chip = &H74
If T33.Text = "" Then
T33.Text = Val(T30.Text)
Else
AA = Val(T33.Text)
BB = Val(T30.Text)
T33.Text = Val(AA + BB)
End If

G = 0
A = Val(T30.Text)
Do
DoEvents
A = A
If A > 0 Then
Call Sendout(224)
Call delay
B = A - 7.5
T30.Text = B
Else
B = 0
A = 0
End If
B = B
If B > 0 Then
Call Sendout(208)
Call delay
D = B - 7.5
T30.Text = D

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

Else
D = 0
A = 0
End If
D = D
If D > 0 Then
Call Sendout(176)
Call delay
E = D - 7.5
T30.Text = E
Else
E = 0
A = 0
End If
E = E
If E > 0 Then
Call Sendout(112)
Call delay
F = E - 7.5
A = F
T30.Text = F
G = G + 1
Text3.Text = G
Else
E = 0
A = 0
End If
A = A
Loop Until A < 0
End Sub

Private Sub C12_Click()
Chip = &H74
If T34.Text = "0" Then
T34.Text = Val(T30.Text)
Else
AA = Val(T34.Text)
BB = Val(T30.Text)
T34.Text = Val(AA + BB)
End If

G = 0
A = Val(T30.Text)
Do
DoEvents
A = A
If A > 0 Then
Call Sendout(112)
Call delay
B = A - 7.5

T30.Text = B
Else

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

B = 0
A = 0
End If
B = B
If B > 0 Then
Call Sendout (176)
Call delay
D = B - 7.5
T30.Text = D
Else
D = 0
A = 0
End If
D = D
If D > 0 Then
Call Sendout (208)
Call delay
E = D - 7.5
T30.Text = E
Else
E = 0
A = 0
End If
E = E
If E > 0 Then
Call Sendout (224)
Call delay
F = E - 7.5
A = F
T30.Text = F
G = G + 1
Text3.Text = G
Else
E = 0
A = 0
End If
A = A
Loop Until A < 0
End Sub

```

```

Private Sub C13_Click()
Chip = &H76
If T41.Text = "" Then
T41.Text = Val(T40.Text)
Else
AA = Val(T41.Text)
BB = Val(T40.Text)
T41.Text = Val(AA + BB)
End If

G = 0
A = Val(T40.Text)
Do
DoEvents

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

A = A
If A > 0 Then
Call Sendout (14)
Call delay
B = A - 7.5
T40.Text = B
Else
B = 0

```

```

A = 0
End If
B = B
If B > 0 Then
Call Sendout (13)

```

```

Call delay
D = B - 7.5
T40.Text = D
Else
D = 0
A = 0
End If
D = D

```

```

If D > 0 Then
Call Sendout (11)
Call delay
E = D - 7.5
T40.Text = E
Else
E = 0
A = 0
End If
E = E

```

```

If E > 0 Then
Call Sendout (7)
Call delay
F = E - 7.5
A = F
T40.Text = F
G = G + 1
Text3.Text = G
Else
E = 0
A = 0
End If
A = A

```

```

Loop Until A < 0
End Sub

```

```

Private Sub C14_Click()
Chip = &H76
If T42.Text = "" Then
T42.Text = Val(T40.Text)

```



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

Else
AA = Val(T42.Text)
BB = Val(T40.Text)
T42.Text = Val(AA + BB)
End If

```

```

G = 0
A = Val(T40.Text)

```

```

Do
DoEvents
A = A
If A > 0 Then
Call Sendout(7)
Call delay
B = A - 7.5
T40.Text = B

```

```

Else
B = 0
A = 0
End If
B = B
If B > 0 Then
Call Sendout(11)
Call delay
D = B - 7.5
T40.Text = D

```

```

Else
D = 0
A = 0
End If
D = D

```

```

If D > 0 Then
Call Sendout(13)
Call delay
E = D - 7.5
T40.Text = E

```

```

Else
E = 0
A = 0
End If
E = E
If E > 0 Then
Call Sendout(14)
Call delay

```

```

F = E - 7.5
A = F
T40.Text = F
G = G + 1
Text3.Text = G

```

```

Else
E = 0

```



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
A = 0
End If
A = A
Loop Until A < 0
End Sub
```

```
Private Sub C15_Click()
Chip = &H76
If T43.Text = "" Then
T43.Text = Val(T40.Text)
Else
AA = Val(T43.Text)
BB = Val(T40.Text)
T43.Text = Val(AA + BB)
End If
```

```
G = 0
A = Val(T40.Text)
Do
DoEvents
A = A
If A > 0 Then
Call Sendout(224)
Call delay
B = A - 7.5
T40.Text = B
Else
B = 0
A = 0
End If
B = B
If B > 0 Then
Call Sendout(208)
Call delay
D = B - 7.5
T40.Text = D
Else
D = 0
A = 0
End If
D = D
If D > 0 Then
Call Sendout(176)
Call delay

E = D - 7.5
T40.Text = E
Else
E = 0
A = 0
```

```
End If
E = E
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

If E > 0 Then
Call Sendout(112)
Call delay
F = E - 7.5
A = F
T40.Text = F
G = G + 1
Text3.Text = G
Else
E = 0
A = 0
End If
A = A
Loop Until A < 0
End Sub

```

```

Private Sub C16_Click()
Chip = &H76
If T44.Text = "" Then
T44.Text = Val(T40.Text)
Else
AA = Val(T44.Text)
BB = Val(T40.Text)
T44.Text = Val(AA + BB)
End If

```

```

G = 0
A = Val(T40.Text)
Do
DoEvents
A = A
If A > 0 Then
Call Sendout(112)
Call delay
B = A - 7.5
T40.Text = B
Else
B = 0
A = 0
End If
B = B
If B > 0 Then
Call Sendout(176)
Call delay
D = B - 7.5
T40.Text = D
Else
D = 0
A = 0
End If
D = D
If D > 0 Then
Call Sendout(208)
Call delay

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

E = D - 7.5
T40.Text = E
Else
E = 0
A = 0
End If
E = E
If E > 0 Then

Call Sendout (224)
Call delay
F = E - 7.5
A = F
T40.Text = F

G = G + 1
Text3.Text = G
Else
E = 0
A = 0
End If
A = A
Loop Until A < 0
End Sub

Private Sub C5_Click()
Chip = &H72
If T21.Text = "" Then
T21.Text = Val (T20.Text)
Else
AA = Val (T21.Text)
BB = Val (T20.Text)
T21.Text = Val (AA + BB)
End If

G = 0
A = Val (T20.Text)
Do
DoEvents
A = A
If A > 0 Then
Call Sendout (14)
Call delay
B = A - 7.5
T20.Text = B
Else
B = 0
A = 0
End If
B = B
If B > 0 Then
Call Sendout (13)
Call delay

```



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

D = B - 7.5
T20.Text = D
Else
D = 0
A = 0
End If
D = D
If D > 0 Then
Call Sendout(11)
Call delay
E = D - 7.5
T20.Text = E
Else
E = 0
A = 0
End If
E = E
If E > 0 Then
Call Sendout(7)
Call delay
F = E - 7.5
A = F
T20.Text = F
G = G + 1
Text3.Text = G
Else
E = 0
A = 0
End If
A = A
Loop Until A = 0

End Sub

Private Sub C6_Click()
Chip = &H72
If T22.Text = "" Then
T22.Text = Val(T20.Text)
Else
AA = Val(T22.Text)
BB = Val(T20.Text)
T22.Text = Val(AA + BB)
End If

G = 0
A = Val(T20.Text)
Do
DoEvents
A = A
If A > 0 Then
Call Sendout(7)
Call delay

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

B = A - 7.5
T20.Text = B
Else
B = 0
A = 0
End If
B = B
If B > 0 Then
Call Sendout(11)
Call delay
D = B - 7.5
T20.Text = D
Else
D = 0
A = 0
End If
D = D
If D > 0 Then
Call Sendout(13)
Call delay
E = D - 7.5
T20.Text = E
Else
E = 0
A = 0
End If
E = E
If E > 0 Then
Call Sendout(14)
Call delay
F = E - 7.5
A = F
T20.Text = F
G = G + 1
Text3.Text = G
Else
E = 0
A = 0
End If
A = A
A = A
Loop Until A < 0
End Sub

```

```

Private Sub C7_Click()
Chip = &H72
If T23.Text = "" Then
T23.Text = Val(T20.Text)
Else
AA = Val(T23.Text)

```

```

BB = Val(T20.Text)
T23.Text = Val(AA + BB)

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

End If

G = 0
A = Val(T20.Text)
Do
DoEvents
A = A
If A > 0 Then
Call Sendout(224)
Call delay
B = A - 7.5
T20.Text = B
Else
B = 0
A = 0
End If
B = B
If B > 0 Then
Call Sendout(208)
Call delay
D = B - 7.5
T20.Text = D
Else
D = 0
A = 0
End If
D = D
If D > 0 Then
Call Sendout(176)
Call delay
E = D - 7.5
T20.Text = E
Else
E = 0
A = 0
End If
E = E
If E > 0 Then
Call Sendout(112)
Call delay
F = E - 7.5
A = F
T20.Text = F
G = G + 1
Text3.Text = G
Else
E = 0
A = 0
End If
A = A
Loop Until A < 0
End Sub

```

```
Private Sub C8_Click()
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไมอนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

Chip = &H72
If T24.Text = "" Then
T24.Text = Val(T20.Text)
Else
AA = Val(T24.Text)
BB = Val(T20.Text)
T24.Text = Val(AA + BB)

```

```
End If
```

```
G = 0
```

```
A = Val(T20.Text)
```

```
Do
```

```
DoEvents
```

```
A = A
```

```
If A > 0 Then
```

```
Call Sendout(112)
```

```
Call delay
```

```
B = A - 7.5
```

```
T20.Text = B
```

```
Else
```

```
B = 0
```

```
A = 0
```

```
End If
```

```
B = B
```

```
If B > 0 Then
```

```
Call Sendout(176)
```

```
Call delay
```

```
D = B - 7.5
```

```
T20.Text = D
```

```
Else
```

```
D = 0
```

```
A = 0
```

```
End If
```

```
D = D
```

```
If D > 0 Then
```

```
Call Sendout(208)
```

```
Call delay
```

```
E = D - 7.5
```

```
T20.Text = E
```

```
Else
```

```
E = 0
```

```
A = 0
```

```
End If
```

```
E = E
```

```
If E > 0 Then
```

```
Call Sendout(224)
```

```
Call delay
```

```
F = E - 7.5
```

```
A = F
```

```
T20.Text = F
```

```
G = G + 1
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

Text3.Text = G
Else
E = 0
A = 0
End If
A = A
Loop Until A < 0
End Sub
Private Sub C9_Click()
Chip = &H74
If T31.Text = "" Then
T31.Text = Val(T30.Text)
Else
AA = Val(T31.Text)
BB = Val(T30.Text)
T31.Text = Val(AA + BB)
End If

G = 0
A = Val(T30.Text)
Do
DoEvents
A = A
If A > 0 Then
Call Sendout(14)

Call delay
B = A - 7.5
T30.Text = B
Else
B = 0
A = 0

End If
B = B
If B > 0 Then
Call Sendout(13)
Call delay
D = B - 7.5
T30.Text = D
Else
D = 0
A = 0
End If
D = D
If D > 0 Then
Call Sendout(11)
Call delay
E = D - 7.5
T30.Text = E
Else
E = 0
A = 0

```



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

End If
E = E
If E > 0 Then
Call Sendout (7)
Call delay
F = E - 7.5
A = F
T30.Text = F
G = G + 1
Text3.Text = G
Else
E = 0
A = 0
End If
A = A
Loop Until A < 0
End Sub

Private Sub Combol_Click()
File1.Pattern = Mid(Combol.Text, Len(Combol.Text) - 5, 5)
File1.Refresh
If Combol.Text = "All File (*.*)" Then
File1.Pattern = "*.*)"
End If
End Sub

Private Sub Command1_Click()
Chip = &H70
If T11.Text = "" Then
T11.Text = Val(Text1.Text)
Else
AA = Val(T11.Text)
BB = Val(Text1.Text)
T11.Text = Val(AA + BB)
End If

G = 0
A = Val(Text1.Text)
Do
DoEvents
A = A
If A > 0 Then
Call Sendout (14)

Call delay
B = A - 7.5
Text1.Text = B
Else
B = 0
A = 0

End If
B = B

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

If B > 0 Then
Call Sendout(13)
Call delay
D = B - 7.5
Text1.Text = D
Else
D = 0
A = 0
End If
D = D
If D > 0 Then
Call Sendout(11)
Call delay
E = D - 7.5
Text1.Text = E
Else
E = 0
A = 0
End If
E = E
If E > 0 Then
Call Sendout(7)
Call delay
F = E - 7.5
A = F
Text1.Text = F
G = G + 1
Text3.Text = G
Else
E = 0
A = 0
End If
A = A
Loop Until A < 0
End Sub

```

```

Private Sub Command2_Click()
Chip = &H70
If T12.Text = "" Then
T12.Text = Val(Text1.Text)
Else
AA = Val(T12.Text)
BB = Val(Text1.Text)
T12.Text = Val(AA + BB)
End If

G = 0
A = Val(Text1.Text)
Do
DoEvents
A = A
If A > 0 Then
Call Sendout(7)
Call delay

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

B = A - 7.5
Text1.Text = B
Else
B = 0
A = 0
End If
B = B

If B > 0 Then
Call Sendout (11)
Call delay
D = B - 7.5
Text1.Text = D
Else

D = 0
A = 0
End If
D = D
If D > 0 Then
Call Sendout (13)
Call delay
E = D - 7.5
Text1.Text = E
Else
E = 0
A = 0
End If
E = E
If E > 0 Then
Call Sendout (14)
Call delay
F = E - 7.5
A = F
Text1.Text = F
G = G + 1
Text3.Text = G
Else
E = 0
A = 0
End If
A = A
Loop Until A < 0
End Sub

Private Sub Command3_Click()
  Chip = &H70
  If T14.Text = "" Then
    T14.Text = Val(Text1.Text)
  Else
    AA = Val(T14.Text)
    BB = Val(Text1.Text)
    T14.Text = Val(AA + BB)
  End If
End Sub

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

End If

G = 0
A = Val(Text1.Text)
Do
DoEvents
A = A
If A > 0 Then
Call Sendout(112)
Call delay
B = A - 7.5
Text1.Text = B
Else
B = 0
A = 0
End If
B = B
If B > 0 Then
Call Sendout(176)
Call delay
D = B - 7.5
Text1.Text = D
Else
D = 0
A = 0

End If
D = D
If D > 0 Then
Call Sendout(208)
Call delay
E = D - 7.5

Text1.Text = E
Else
E = 0
A = 0
End If
E = E
If E > 0 Then
Call Sendout(224)
Call delay
F = E - 7.5
A = F
Text1.Text = F
G = G + 1
Text3.Text = G
Else
E = 0
A = 0
End If
A = A
Loop Until A < 0

```



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

End Sub

Private Sub Command4_Click()
    Chip = &H70
    If T13.Text = "" Then
    T13.Text = Val(Text1.Text)
    Else
    AA = Val(T13.Text)
    BB = Val(Text1.Text)
    T13.Text = Val(AA + BB)
    End If

    G = 0
    A = Val(Text1.Text)
    Do
    DoEvents
    A = A
    If A > 0 Then
    Call Sendout(224)
    Call delay
    B = A - 7.5
    Text1.Text = B
    Else
    B = 0
    A = 0
    End If
    B = B
    If B > 0 Then
    Call Sendout(208)
    Call delay
    D = B - 7.5
    Text1.Text = D
    Else
    D = 0
    A = 0
    End If
    D = D
    If D > 0 Then
    Call Sendout(176)
    Call delay
    E = D - 7.5
    Text1.Text = E
    Else

    E = 0
    A = 0
    End If
    E = E
    If E > 0 Then
    Call Sendout(112)

    Call delay
    F = E - 7.5

```



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

A = F
Text1.Text = F
G = G + 1
Text3.Text = G
Else
E = 0
A = 0
End If
A = A
Loop Until A < 0
End Sub

Private Sub Command5_Click()
Frame10.Visible = False
End Sub

Private Sub Command6_Click()
Text4.Text = ""
Frame10.Visible = False
End Sub

Private Sub Command7_Click()
FF3.Visible = False
End Sub

Private Sub Command8_Click()
BBA = 1
BBB = 2
BBC = 3
BBD = 4
If Option1.Value = Checked Then
MSComm1.CommPort = BBA
If Option2.Value = Checked Then
MSComm1.CommPort = BBB
If Option3.Value = Checked Then
MSComm1.CommPort = BBC
If Option4.Value = Checked Then
MSComm1.CommPort = BBD
If Not MSComm1.PortOpen Then 'Open Port
MSComm1.PortOpen = True
End If
End If
End If
End If
Call Sendout(0)
FF3.Visible = False
End Sub

Private Sub Dir1_Change()
File1.FileName = Dir1.Path
End Sub

Private Sub Drive1_Change()

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

On Error Resume Next
Dir1.Path = Drive1.Drive
File1.FileName = Dir1.Path
End Sub

Private Sub EDITPASSWORD_Click()
Form4.Visible = True

End Sub

Private Sub EDITPORT_Click()
FF3.Visible = True
End Sub

Private Sub EDITVIDEO_Click()
Frame10.Visible = True
End Sub

Private Sub EX1_Click()
End
End Sub

Private Sub EX2_Click()
FF1.Visible = False
XXX.Enabled = True
End Sub

Private Sub EXIT_Click()
FF1.Visible = True
End Sub

Private Sub File1_Click()
Text4.Text = File1.FileName
End Sub

Private Sub Form_Load()
If Not MSComm1.PortOpen Then 'Open Port
    MSComm1.PortOpen = True
End If
    Call Sendout(0)
Form2.Visible = True
TXT = "ระบบเตือนภัยผู้บุกรุกสำนักงานผ่านเครือข่ายอินเทอร์เน็ต (Web Based Office Security)" +
Space(250)
Me.Caption = TXT
LENGTH = Len(TXT) - 1
Combo1.AddItem "DAT File(*.DAT) "
Combo1.AddItem "AVI File(*.AVI) "
Combo1.AddItem "WAV File(*.WAV) "
Combo1.AddItem "MP3 File(*.MP3) "
Combo1.AddItem "MID File(*.MID) "
End Sub

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

Private Sub Form_Unload(Cancel As Integer)
    If MSComm1.PortOpen Then      'Close Port
        MSComm1.PortOpen = False
    End If
End Sub
Private Sub Ack()
    MSComm1.RTSEnable = True      'SDA=1
    MSComm1.DTREnable = True      'SCL=1
    MSComm1.DTREnable = False     'SCL=0
End Sub
Private Function DAT()
    For I = 7 To 0 Step -1
        MSComm1.RTSEnable = True      'SDA=1
        MSComm1.DTREnable = True      'SCL=1
        If Not MSComm1.CTSHolding Then 'Read SDA
            DAT1 = 2 ^ I Or DAT1
        End If
        MSComm1.DTREnable = False     'SCL=0
    Next I
    DAT = DAT1                      'Data 8 Bit
End Function
Private Sub I2CStart()
    MSComm1.RTSEnable = True      'SDA=1
    MSComm1.DTREnable = True      'SCL=1
    MSComm1.RTSEnable = False     'SDA=0
    MSComm1.DTREnable = False     'SCL=0
End Sub
Private Sub I2CStop()
    MSComm1.RTSEnable = False     'SDA=0
    MSComm1.DTREnable = True      'SCL=1
    MSComm1.RTSEnable = True      'SDA=1
End Sub
Private Sub send0()
    MSComm1.RTSEnable = False     'SDA=0
    MSComm1.DTREnable = True      'SCL=1
    MSComm1.DTREnable = False     'SCL=0
End Sub
Private Sub Send1()
    MSComm1.RTSEnable = True      'SDA=1
    MSComm1.DTREnable = True      'SCL=1
    MSComm1.DTREnable = False     'SCL=0
End Sub
Private Sub Send8BIT(Add As Integer)
    For I = 7 To 0 Step -1
        If (Add And 2 ^ I) = 2 ^ I Then 'Test Bit 0 OR 1
            Call Send1
        Else
            Call send0
        End If
    Next I

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

End Sub

Private Sub Sendout (B As Integer)
    Call I2CStart           'Start
    Call Send8BIT(Chip)     'Send Control Word
    Call Ack                 'Acknowledge
    Call Send8BIT(B)        'Send Data
    Call Ack                 'Acknowledge
    Call I2CStop            'Stop
End Sub

```

```

Private Sub TC_Timer()
    If Val(L1.Text) > 0 Then
        TL1.Enabled = True
        L1.Text = L1.Text
    ElseIf Val(L1.Text) <= 0 Then
        L1.Text = 0
        TL1.Enabled = False
    End If
    If Val(R1.Text) > 0 Then
        TR1.Enabled = True
        R1.Text = R1.Text
    ElseIf Val(R1.Text) <= 0 Then
        R1.Text = 0
        TR1.Enabled = False
    End If
    If Val(T1.Text) > 0 Then
        TT1.Enabled = True
        T1.Text = T1.Text
    ElseIf Val(T1.Text) <= 0 Then
        T1.Text = 0
        TT1.Enabled = False
    End If
    If Val(U1.Text) > 0 Then
        TU1.Enabled = True
        U1.Text = U1.Text
    ElseIf Val(U1.Text) <= 0 Then
        U1.Text = 0
        TU1.Enabled = False
    End If

```

```

    If Val(L2.Text) > 0 Then
        TL2.Enabled = True

        L2.Text = L2.Text
    ElseIf Val(L2.Text) <= 0 Then
        L2.Text = 0
        TL2.Enabled = False
    End If

```

```

    If Val(R2.Text) > 0 Then
        TR2.Enabled = True
        R2.Text = R2.Text

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
ElseIf Val(R2.Text) <= 0 Then
R2.Text = 0
TR2.Enabled = False
End If
```

```
If Val(T2.Text) > 0 Then
TT2.Enabled = True
T2.Text = T2.Text
ElseIf Val(T2.Text) <= 0 Then
T2.Text = 0
TT2.Enabled = False
End If
```

```
If Val(U2.Text) > 0 Then
TU2.Enabled = True
U2.Text = U2.Text
ElseIf Val(U2.Text) <= 0 Then
U2.Text = 0
TU2.Enabled = False
End If
```

```
If Val(L3.Text) > 0 Then
TL3.Enabled = True
L3.Text = L3.Text
ElseIf Val(L3.Text) <= 0 Then
L3.Text = 0
TL3.Enabled = False
End If
```

```
If Val(R3.Text) > 0 Then
TR3.Enabled = True
R3.Text = R3.Text
ElseIf Val(R3.Text) <= 0 Then
R3.Text = 0
TR3.Enabled = False
End If
```

```
If Val(T3.Text) > 0 Then
TT3.Enabled = True
T3.Text = T3.Text
ElseIf Val(T3.Text) <= 0 Then
T3.Text = 0
TT3.Enabled = False
End If
```

```
If Val(U3.Text) > 0 Then
TU3.Enabled = True
U3.Text = U3.Text
ElseIf Val(U3.Text) <= 0 Then
U3.Text = 0
TU3.Enabled = False
End If
```

```
If Val(L4.Text) > 0 Then
TL4.Enabled = True
L4.Text = L4.Text
ElseIf Val(L4.Text) <= 0 Then
L4.Text = 0
TL4.Enabled = False
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

End If
If Val(R4.Text) > 0 Then
TR4.Enabled = True
R4.Text = R4.Text
ElseIf Val(R4.Text) <= 0 Then
R4.Text = 0
TR4.Enabled = False

```

```

End If
If Val(T4.Text) > 0 Then
TT4.Enabled = True
T4.Text = T4.Text
ElseIf Val(T4.Text) <= 0 Then

```

```

T4.Text = 0
TT4.Enabled = False
End If
If Val(U4.Text) > 0 Then
TU4.Enabled = True
U4.Text = U4.Text
ElseIf Val(U4.Text) <= 0 Then
U4.Text = 0
TU4.Enabled = False
End If
End Sub

```

```

Private Sub Text4_Change()
MediaPlayer2.FileName = Dir1.Path & "\" & Text4.Text
End Sub

```

```

Private Sub Timer1_Timer()
TC.Enabled = True
If Val(Text1.Text) < 0 Then
Text1.Text = 0
Else
Text1.Text = Text1.Text
End If
If Val(T20.Text) < 0 Then
T20.Text = 0
Else
T20.Text = T20.Text
End If
If Val(T30.Text) < 0 Then
T30.Text = 0
Else
T30.Text = T30.Text
End If
If Val(T40.Text) < 0 Then
T40.Text = 0
Else
T40.Text = T40.Text
End If
End Sub

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

Private Sub Timer2_Timer()
Call Cha1
Call Cha2
Call Cha3
Call Cha4
Data1.Refresh
Data2.Refresh
Data3.Refresh
Text2.Text = HScroll1.Value
Label10.Caption = Format(Time, "LONG TIME")
Label24.Caption = Format(Date, "LONG DATE")
If TC.Enabled = True Then
Shape3.Visible = True
Else
Shape3.Visible = False
End If
If TL1.Enabled = True Or TL2.Enabled = True Or TL3.Enabled = True
Or TL4.Enabled = True Then
Shape4.Visible = True
Else
Shape4.Visible = False
End If
If TR1.Enabled = True Or TR2.Enabled = True Or TR3.Enabled = True
Or TR4.Enabled = True Then
Shape5.Visible = True
Else
Shape5.Visible = False
End If
If TT1.Enabled = True Or TT2.Enabled = True Or TT3.Enabled = True
Or
TT4.Enabled = True Then
Shape6.Visible = True
Else
Shape6.Visible = False
End If
If TU1.Enabled = True Or TU2.Enabled = True Or TU3.Enabled = True
Or TU4.Enabled = True Then
Shape7.Visible = True
Else
Shape7.Visible = False
End If
End Sub

Private Sub Chal()
Dim AB, AC, AD, AE, AF, AG, AH, AI, AJ, AK, AL, AM, AN As Double
AJ = Val(KA.Text)
AB = Val(T11.Text)
AC = Val(T12.Text)
AD = Val(T13.Text)
AE = Val(T14.Text)

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

If AB >= AC Then
AF = AB - AC
DL1.Text = AF
Command1.Caption = "ซ้าย" & AF
Command2.Caption = "ขวา" & 0
ElseIf AC >= AB Then
AG = AC - AB
DR1.Text = AG
Command2.Caption = "ขวา" & AG
Command1.Caption = "ซ้าย" & 0
End If

```

```

If AD >= AE Then
AH = AD - AE
DT1.Text = AH
Command4.Caption = "บน" & AH
Command3.Caption = "ล่าง" & 0
ElseIf AE >= AD Then
AI = AE - AD
DU1.Text = AI
Command3.Caption = "ล่าง" & AI
Command4.Caption = "บน" & 0
End If

```

```

If AF >= AJ Then
Command1.Enabled = False
Else
Command1.Enabled = True
End If

```

```

If AG >= AJ Then
Command2.Enabled = False
Else
Command2.Enabled = True
End If

```

```

If AH >= AJ Then
Command4.Enabled = False
Else
Command4.Enabled = True
End If

```

```

If AI >= AJ Then

```

```

Command3.Enabled = False
Else
Command3.Enabled = True
End If
End Sub

```

```

Private Sub Cha2()
Dim BB, BC, BD, BE, BF, BG, BH, BI, BJ, BK, BL, BM, BN As Double
BJ = Val(KA.Text)
BB = Val(T21.Text)
BC = Val(T22.Text)

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

BD = Val (T23.Text)
BE = Val (T24.Text)
If BB >= BC Then
BF = BB - BC
DL2.Text = BF
C5.Caption = "ซ้าย" & BF
C6.Caption = "ขวา" & 0
ElseIf BC >= BB Then
BG = BC - BB
DR2.Text = BG
C6.Caption = "ขวา" & BG
C5.Caption = "ซ้าย" & 0
End If
If BD >= BE Then
BH = BD - BE
DT2.Text = BH
C7.Caption = "บน" & BH
C8.Caption = "ล่าง" & 0
ElseIf BE >= BD Then
BI = BE - ED
DU2.Text = BI
C8.Caption = "ล่าง" & BI
C7.Caption = "บน" & 0
End If

If BF >= BJ Then
C5.Enabled = False
Else
C5.Enabled = True
End If
If BG >= BJ Then
C6.Enabled = False
Else
C6.Enabled = True
End If
If BH >= BJ Then
C7.Enabled = False
Else
C7.Enabled = True
End If
If BI >= BJ Then
C8.Enabled = False
Else
C8.Enabled = True
End If
End Sub
Private Sub Cha3()
Dim CB, CC, CD, CE, CF, CG, CH, CI, CJ, CK, CL, CM, CN As Double
CJ = Val (KA.Text)
CB = Val (T31.Text)
CC = Val (T32.Text)

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
CD = Val(T33.Text)
CE = Val(T34.Text)

If CB >= CC Then
CF = CB - CC
DL3.Text = CF
C9.Caption = "ซ้าย" & CF
```

```
C10.Caption = "ขวา" & 0
ElseIf CC >= CB Then
CG = CC - CB
DR3.Text = CG
C10.Caption = "ขวา" & CG
C9.Caption = "ซ้าย" & 0
End If
```

```
If CD >= CE Then
CH = CD - CE
DT3.Text = CH
C11.Caption = "บน" & CH
C12.Caption = "ล่าง" & 0
ElseIf CE >= CD Then
CI = CE - CD
DU3.Text = CI
C12.Caption = "ล่าง" & CI
C11.Caption = "บน" & 0
End If
```

```
If CF >= CJ Then
C9.Enabled = False
Else
C9.Enabled = True
End If
If CG >= CJ Then
C10.Enabled = False
Else
C10.Enabled = True
End If
If CH >= CJ Then
C11.Enabled = False
Else
C11.Enabled = True
End If
If CI >= CJ Then
C12.Enabled = False
Else
C12.Enabled = True
End If
End Sub
```

```
Private Sub Timer3_Timer()
Dim TMP2 As Integer
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

Call I2CStart 'Start
Call Send8BIT(&H4F) 'Control Word Read
Call Ack 'Acknowledge
TMP2 = DAT 'Read Data
Call Ack 'Acknowledge
Call I2CStop 'Stop
For I = 0 To 7 '-----
    If (TMP2 And 2 ^ I) = 2 ^ I Then 'Show Data On Display
        Shape2(I).BackColor = &HFF00&
    Else
        Shape2(I).BackColor = &H8000&
    End If
Next I
End Sub

Private Sub Timer4_Timer()
    For I = 0 To 7 '-----
        If Shape2(I).BackColor = &H8000& Then 'Show Data On Display
            ' Form2.Command6.Value = True
            Label1.Visible = True
            Label1.Caption = "สัญญาณที่" & I + 1

            Timer4.Enabled = False
        Else
            Timer4.Enabled = True
        End If
    Next I
End Sub

Private Sub Timer5_Timer()
    TXT = Left(Right(TXT, LENGTH) & LEFT(TXT, 1)
    FORM1.Caption = TXT
    Label26.Caption = "ไฟล์เล่น" & Dir1.Path & "\" & Text4.Text
End Sub

Private Sub Timer6_Timer()
    If Shape2(0).BackColor = &H8000& Then 'Show Data On Display
        S1.Visible = True
    End If

    If Shape2(1).BackColor = &H8000& Then 'Show Data On Display
        S2.Visible = True
    End If

    If Shape2(2).BackColor = &H8000& Then 'Show Data On Display
        S3.Visible = True
    End If

    If Shape2(3).BackColor = &H8000& Then 'Show Data On Display
        S4.Visible = True
    End If

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไมอนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

If Shape2(4).BackColor = &H8000& Then 'Show Data On Display
S5.Visible = True
End If

If Shape2(5).BackColor = &H8000& Then 'Show Data On Display
S6.Visible = True
End If

If Shape2(6).BackColor = &H8000& Then 'Show Data On Display
S7.Visible = True
End If

If Shape2(7).BackColor = &H8000& Then 'Show Data On Display
S8.Visible = True
End If

If S1.Visible = True Then
Text5.Text = "มีผู้บุกรุก ณ จุดที่ 1"
Else
Text5.Text = "ไม่มีผู้บุกรุก ณ จุดที่ 1"
End If
If S2.Visible = True Then
Text6.Text = "มีผู้บุกรุก ณ จุดที่ 2"
Else
Text6.Text = "ไม่มีผู้บุกรุก ณ จุดที่ 2"
End If
If S3.Visible = True Then
Text7.Text = "มีผู้บุกรุก ณ จุดที่ 3"
Else
Text7.Text = "ไม่มีผู้บุกรุก ณ จุดที่ 3"
End If
If S4.Visible = True Then
Text8.Text = "มีผู้บุกรุก ณ จุดที่ 4"
Else
Text8.Text = "ไม่มีผู้บุกรุก ณ จุดที่ 4"
End If
If S5.Visible = True Then
Text9.Text = "มีผู้บุกรุก ณ จุดที่ 5"
Else
Text9.Text = "ไม่มีผู้บุกรุก ณ จุดที่ 5"
End If
If S6.Visible = True Then
Text10.Text = "มีผู้บุกรุก ณ จุดที่ 6"
Else
Text10.Text = "ไม่มีผู้บุกรุก ณ จุดที่ 6"
End If
If S7.Visible = True Then
Text11.Text = "มีผู้บุกรุก ณ จุดที่ 7"

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

Else
    Text11.Text = "ไม่มีผู้บุกรุก ณ จุดที่ 7"
End If
If S8.Visible = True Then
    Text12.Text = "มีผู้บุกรุก ณ จุดที่ 8"
Else
    Text12.Text = "ไม่มีผู้บุกรุก ณ จุดที่ 8"
End If
End Sub

```

```

Private Sub TL1_Timer()
If Val(L1.Text) > 0 Then
    Chip = &H70
    If T11.Text = "" Then
        T11.Text = Val(L1.Text)
    Else
        AA = Val(T11.Text)
        BB = Val(L1.Text)
        T11.Text = Val(AA + BB)
    End If
    G = 0
    Do
        A = Val(L1.Text)
        DoEvents
        A = A
        If A > 0 Then
            Call Sendout(14)
            Call delay
            B = A - 7.5
            L1.Text = B
        Else
            B = 0
            A = 0
        End If
        B = B
        If B > 0 Then
            Call Sendout(13)
            Call delay
            D = B - 7.5
            L1.Text = D
        Else
            D = 0
            A = 0
        End If
        D = D

        If D > 0 Then
            Call Sendout(11)
            Call delay
            E = D - 7.5

            L1.Text = E

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

Else
E = 0
A = 0
End If
E = E
If E > 0 Then
Call Sendout(7)
Call delay
F = E - 7.5
A = F
L1.Text = F
G = G + 1
Text3.Text = G
Else
E = 0
A = 0
End If
A = A
Loop Until A <= 0
Else
L1.Text = 0
End If
End Sub

Private Sub TL2_Timer()
If Val(L2.Text) > 0 Then
Chip = &H72
If T21.Text = "" Then
T21.Text = Val(L2.Text)
Else
AA = Val(T21.Text)
BB = Val(L2.Text)
T21.Text = Val(AA + BB)
End If
G = 0
Do
A = Val(L2.Text)
DoEvents
A = A
If A > 0 Then
Call Sendout(14)
Call delay
B = A - 7.5
L2.Text = B
Else
B = 0
A = 0
End If
B = B
If B > 0 Then
Call Sendout(13)
Call delay
D = B - 7.5
L2.Text = 0

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

Else
D = 0
A = 0
End If
D = D
If D > 0 Then
Call Sendout(11)
Call delay
E = D - 7.5
L2.Text = E

Else
E = 0
A = 0
End If

E = E
If E > 0 Then
Call Sendout(7)
Call delay
F = E - 7.5
A = F
L2.Text = F
G = G + 1
Text3.Text = G
Else
E = 0
A = 0
End If
A = A
Loop Until A <= 0
Else
L2.Text = 0
End If
End Sub

Private Sub TL3_Timer()
If Val(L3.Text) > 0 Then
Chip = &H74
If T31.Text = "" Then
T31.Text = Val(L3.Text)
Else
AA = Val(T31.Text)
BB = Val(L3.Text)
T31.Text = Val(AA + BB)
End If
G = 0
Do
A = Val(L3.Text)
DoEvents
A = A
If A > 0 Then
Call Sendout(14)

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

Call delay
B = A - 7.5
L3.Text = B
Else
B = 0
A = 0
End If
B = B
If B > 0 Then
Call Sendout(13)
Call delay
D = B - 7.5
L3.Text = D
Else
D = 0
A = 0
End If
D = D
If D > 0 Then
Call Sendout(11)
Call delay
E = D - 7.5
L3.Text = E
Else
E = 0
A = 0
End If
E = E
If E > 0 Then
Call Sendout(7)
Call delay
F = E - 7.5

A = F
L3.Text = F
G = G + 1
Text3.Text = G
Else
E = 0
A = 0
End If
A = A
Loop Until A <= 0
Else
L3.Text = 0
End If
End Sub

```

```

Private Sub TL4_Timer()
If Val(L4.Text) > 0 Then
Chip = &H76
If T41.Text = "" Then

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

T41.Text = Val(L4.Text)
Else
AA = Val(T41.Text)
BB = Val(L4.Text)
T41.Text = Val(AA + BB)
End If
G = 0
Do
A = Val(L4.Text)
DoEvents
A = A
If A > 0 Then
Call Sendout(14)
Call delay
B = A - 7.5
L4.Text = B
Else
B = 0
A = 0
End If
B = B
If B > 0 Then
Call Sendout(13)
Call delay
D = B - 7.5
L4.Text = D
Else
D = 0
A = 0
End If
D = D
If D > 0 Then
Call Sendout(11)
Call delay
E = D - 7.5
L4.Text = E
Else
E = 0
A = 0
End If
E = E
If E > 0 Then
Call Sendout(7)
Call delay
F = E - 7.5
A = F

L4.Text = F
G = G + 1
Text3.Text = G
Else

```

```

E = 0

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

A = 0
End If
A = A
Loop Until A <= 0
Else
L4.Text = 0
End If
End Sub

Private Sub TMI1_Timer()
Label11.Visible = True
TMI2.Enabled = True
TMI1.Enabled = False
End Sub

Private Sub TMI2_Timer()
Label11.Visible = False
TMI1.Enabled = True
TMI2.Enabled = False
End Sub

Private Sub TR1_Timer()
If Val(R1.Text) > 0 Then
    Chip = &H70
    If T12.Text = "" Then
        T12.Text = Val(R1.Text)
    Else
        AA = Val(T12.Text)
        BB = Val(R1.Text)
        T12.Text = Val(AA + BB)
    End If
    G = 0
    Do
        A = Val(R1.Text)
        DoEvents
        A = A
        If A > 0 Then
            Call Sendout(7)
            Call delay
            B = A - 7.5
            R1.Text = B
        Else
            B = 0
            A = 0
        End If
        B = B
        If B > 0 Then
            Call Sendout(11)
            Call delay
            D = B - 7.5
            R1.Text = D
        Else
            D = 0
            A = 0

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

End If
D = D
If D > 0 Then
Call Sendout(13)
Call delay
E = D - 7.5
R1.Text = E
Else
E = 0
A = 0

End If
E = E
If E > 0 Then
Call Sendout(14)

Call delay
F = E - 7.5
A = F
R1.Text = F
G = G + 1
Text3.Text = G
Else
E = 0
A = 0
End If
A = A
Loop Until A <= 0
Else
R1.Text = 0
End If
End Sub

Private Sub TR2_Timer()
If Val(R2.Text) > 0 Then
Chip = &H72
If T22.Text = "" Then
T22.Text = Val(R2.Text)
Else
AA = Val(T22.Text)
BB = Val(R2.Text)
T22.Text = Val(AA + BB)
End If
G = 0
Do
A = Val(R2.Text)
DoEvents
A = A
If A > 0 Then
Call Sendout(7)
Call delay
B = A - 7.5
R2.Text = B

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

Else
B = 0
A = 0
End If
B = B
If B > 0 Then
Call Sendout(11)
Call delay
D = B - 7.5
R2.Text = D
Else
D = 0
A = 0
End If
D = D
If D > 0 Then
Call Sendout(13)
Call delay
E = D - 7.5
R2.Text = E
Else
E = 0
A = 0
End If
E = E
If E > 0 Then
Call Sendout(14)
Call delay
F = E - 7.5
A = F
R2.Text = F
G = G + 1

Text3.Text = G
Else
E = 0
A = 0
End If
A = A
Loop Until A <= 0
Else
R2.Text = 0
End If
End Sub

Private Sub TR3_Timer()
If Val(R3.Text) > 0 Then
Chip = &H74
If T32.Text = "" Then
T32.Text = Val(R3.Text)
Else
AA = Val(T32.Text)

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

BB = Val (R3.Text)
T32.Text = Val (AA + BB)
End If
G = 0
Do
A = Val (R3.Text)
DoEvents
A = A
If A > 0 Then
Call Sendout (7)
Call delay
B = A - 7.5
R3.Text = B
Else
B = 0
A = 0
End If
B = B
If B > 0 Then
Call Sendout (11)
Call delay
D = B - 7.5
R3.Text = D
Else
D = 0
A = 0
End If
D = D
If D > 0 Then
Call Sendout (13)
Call delay
E = D - 7.5
R3.Text = E
Else
E = 0
A = 0
End If
E = E
If E > 0 Then
Call Sendout (14)
Call delay
F = E - 7.5
A = F
R3.Text = F
G = G + 1
Text3.Text = G

Else
E = 0
A = 0
End If

A = A

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

Loop Until A <= 0
Else
R3.Text = 0
End If
End Sub

Private Sub TR4_Timer()
If Val(R4.Text) > 0 Then
    Chip = &H76
    If T42.Text = "" Then
        T42.Text = Val(R4.Text)
    Else
        AA = Val(T42.Text)
        BB = Val(R4.Text)
        T42.Text = Val(AA + BB)
    End If
    G = 0
    Do
        A = Val(R4.Text)
        DoEvents
        A = A
        If A > 0 Then
            Call Sendout(7)
            Call delay
            B = A - 7.5
            R4.Text = B
        Else
            B = 0
            A = 0
        End If
        B = B
        If B > 0 Then
            Call Sendout(11)
            Call delay
            D = B - 7.5
            R4.Text = D
        Else
            D = 0
            A = 0
        End If
        D = D
        If D > 0 Then
            Call Sendout(13)
            Call delay
            E = D - 7.5
            R4.Text = E
        Else
            E = 0
            A = 0
        End If
        E = E
        If E > 0 Then
            Call Sendout(14)
            Call delay

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

F = E - 7.5
A = F
R4.Text = F
G = G + 1
Text3.Text = G
Else
E = 0
A = 0
End If
A = A

Loop Until A <= 0
Else
R4.Text = 0
End If

```

```
End Sub
```

```

Private Sub TT1_Timer()
If Val(T1.Text) > 0 Then
Chip = &H70
If T13.Text = "" Then
T13.Text = Val(T1.Text)
Else
AA = Val(T13.Text)
BB = Val(T1.Text)
T13.Text = Val(AA + BB)
End If
G = 0
Do
A = Val(T13.Text)
DoEvents
A = A
If A > 0 Then
Call Sendout(224)
Call delay
B = A - 7.5
T1.Text = B
Else
B = 0
A = 0
End If
B = B
If B > 0 Then
Call Sendout(208)
Call delay
D = B - 7.5
T1.Text = D
Else
D = 0
A = 0
End If
D = D

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

If D > 0 Then
Call Sendout (176)
Call delay
E = D - 7.5
T1.Text = E
Else
E = 0
A = 0
End If
E = E
If E > 0 Then
Call Sendout (112)
Call delay
F = E - 7.5
A = F
T1.Text = F
G = G + 1
Text3.Text = G
Else
E = 0
A = 0
End If
A = A
Loop Until A <= 0
Else
T1.Text = 0
End If
End Sub

Private Sub TT2_Timer()
If Val (T2.Text) > 0 Then
Chip = &H72
If T23.Text = "" Then

T23.Text = Val (T2.Text)
Else
AA = Val (T23.Text)
BB = Val (T2.Text)
T23.Text = Val (AA + BB)
End If
G = 0
Do
A = Val (T2.Text)
DoEvents
A = A
If A > 0 Then
Call Sendout (224)
Call delay
B = A - 7.5
T2.Text = B
Else
B = 0
A = 0

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

End If
B = B
If B > 0 Then
Call Sendout(208)
Call delay
D = B - 7.5
T2.Text = D
Else
D = 0
A = 0
End If
D = D
If D > 0 Then
Call Sendout(176)
Call delay
E = D - 7.5
T2.Text = E
Else
E = 0
A = 0
End If
E = E
If E > 0 Then
Call Sendout(112)
Call delay
F = E - 7.5
A = F
T2.Text = F
G = G + 1
Text3.Text = G
Else
E = 0
A = 0
End If
A = A
Loop Until A <= 0
Else
T2.Text = 0
End If
End Sub

Private Sub TT3_Timer()
If Val(T3.Text) > 0 Then
Chip = &H74
If T33.Text = "" Then
T33.Text = Val(T3.Text)

Else
AA = Val(T33.Text)
BB = Val(T3.Text)
T33.Text = Val(AA + BB)

End If

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

G = 0
Do
A = Val(T3.Text)
DoEvents
A = A
If A > 0 Then
Call Sendout(224)
Call delay
B = A - 7.5
T3.Text = B
Else
B = 0
A = 0
End If
B = B
If B > 0 Then
Call Sendout(208)
Call delay
D = B - 7.5
T3.Text = D
Else
D = 0
A = 0
End If
D = D
If D > 0 Then
Call Sendout(176)
Call delay
E = D - 7.5
T3.Text = E
Else
E = 0
A = 0
End If
E = E
If E > 0 Then
Call Sendout(112)
Call delay
F = E - 7.5
A = F
T3.Text = F
G = G + 1
Text3.Text = G
Else
E = 0
A = 0
End If
A = A
Loop Until A <= 0
Else
T3.Text = 0
End If
End Sub

```



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

Private Sub TT4_Timer()
If Val(T4.Text) > 0 Then
    Chip = &H76
    If T43.Text = "" Then
        T43.Text = Val(T4.Text)
    Else
        AA = Val(T43.Text)
        BB = Val(T4.Text)
        T43.Text = Val(AA + BB)
    End If

```

```

    G = 0
    Do
        A = Val(T4.Text)
        DoEvents

```

```

    A = A
    If A > 0 Then
        Call Sendout(224)
        Call delay
        B = A - 7.5
        T4.Text = B
    Else
        B = 0
        A = 0
    End If
    B = B
    If B > 0 Then
        Call Sendout(208)
        Call delay
        D = B - 7.5
        T4.Text = D
    Else
        D = 0
        A = 0
    End If
    D = D
    If D > 0 Then
        Call Sendout(176)
        Call delay
        E = D - 7.5
        T4.Text = E
    Else
        E = 0
        A = 0
    End If
    E = E
    If E > 0 Then
        Call Sendout(112)
        Call delay
        F = E - 7.5
        A = F
        T4.Text = F

```



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

G = G + 1
Text3.Text = G
Else
E = 0
A = 0
End If
A = A
Loop Until A <= 0
Else
T4.Text = 0
End If
End Sub

Private Sub TU1_Timer()
If Val(U1.Text) > 0 Then
    Chip = &H70
    If T14.Text = "" Then
        T14.Text = Val(U1.Text)
    Else
        AA = Val(T14.Text)
        BB = Val(U1.Text)
        T14.Text = Val(AA + BB)
    End If
    G = 0
    Do
        A = Val(U1.Text)
        DoEvents
        A = A
        If A > 0 Then
            Call Sendout(112)
            Call delay
            B = A - 7.5
            U1.Text = B
        Else
            B = 0
            A = 0
            End If
            B = B
            If B > 0 Then
                Call Sendout(176)
                Call delay
                D = B - 7.5
                U1.Text = D
            Else
                D = 0
                A = 0
                End If
                D = D
                If D > 0 Then
                    Call Sendout(208)
                    Call delay
                    E = D - 7.5

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

U1.Text = E
Else
E = 0
A = 0
End If
E = E
If E > 0 Then
Call Sendout (224)
Call delay
F = E - 7.5
A = F
U1.Text = F
G = G + 1
Text3.Text = G
Else
E = 0
A = 0
End If
A = A
Loop Until A <= 0
Else
U1.Text = 0
End If
End Sub

Private Sub TU2_Timer ()
If Val (U2.Text) > 0 Then
Chip = &H72
If T24.Text = "" Then
T24.Text = Val (U2.Text)
Else
AA = Val (T24.Text)
BB = Val (U2.Text)
T24.Text = Val (AA + BB)
End If
G = 0
Do
A = Val (U2.Text)
DoEvents
A = A
If A > 0 Then
Call Sendout (112)
Call delay
B = A - 7.5
U2.Text = B

Else
B = 0
A = 0
End If
B = B

If B > 0 Then

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

Call Sendout(176)
Call delay
D = B - 7.5
U2.Text = D
Else
D = 0
A = 0
End If
D = D
If D > 0 Then
Call Sendout(208)
Call delay
E = D - 7.5
U2.Text = E
Else
E = 0
A = 0
End If
E = E
If E > 0 Then
Call Sendout(224)
Call delay
F = E - 7.5
A = F
U2.Text = F
G = G + 1
Text3.Text = G
Else
E = 0
A = 0
End If
A = A
Loop Until A <= 0
Else
U2.Text = 0
End If
End Sub

Private Sub TU3_Timer()
If Val(U3.Text) > 0 Then
Chip = &H74
If T34.Text = "" Then
T34.Text = Val(U3.Text)
Else
AA = Val(T34.Text)
BB = Val(U3.Text)
T34.Text = Val(AA + BB)
End If
G = 0
Do
A = Val(U3.Text)
DoEvents
A = A
If A > 0 Then

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

Call Sendout (112)
Call delay
B = A - 7.5
U3.Text = B
Else
B = 0
A = 0
End If
B = B

```

```

If B > 0 Then
Call Sendout (176)
Call delay
D = B - 7.5
U3.Text = D

```

```

Else
D = 0
A = 0
End If
D = D

```

```

If D > 0 Then
Call Sendout (208)
Call delay
E = D - 7.5
U3.Text = E

```

```

Else
E = 0
A = 0
End If
E = E

```

```

If E > 0 Then
Call Sendout (224)
Call delay
F = E - 7.5
A = F

```

```

U3.Text = F
G = G + 1
Text3.Text = G
Else
E = 0
A = 0

```

```

End If
A = A
Loop Until A <= 0

```

```

Else
U3.Text = 0
End If
End Sub

```

```

Private Sub TU4_Timer()
If Val(U4.Text) > 0 Then
Chip = &H76

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

If T44.Text = "" Then
  T44.Text = Val(U4.Text)
Else
  AA = Val(T44.Text)
  BB = Val(U4.Text)
  T44.Text = Val(AA + BB)
End If
  G = 0
  Do
    A = Val(U4.Text)
    DoEvents
    A = A
    If A > 0 Then
      Call Sendout(112)
      Call delay
      B = A - 7.5
      U4.Text = B
    Else
      B = 0
      A = 0
    End If
    B = B
    If B > 0 Then
      Call Sendout(176)
      Call delay
      D = B - 7.5
      U4.Text = D
    Else
      D = 0
      A = 0
    End If
    D = D

    If D > 0 Then
      Call Sendout(208)
      Call delay
      E = D - 7.5
      U4.Text = E
    Else
      E = 0
      A = 0
    End If
    E = E
    If E > 0 Then
      Call Sendout(224)
      Call delay
      F = E - 7.5
      A = F
      U4.Text = F
      G = G + 1
      Text3.Text = G
    Else

```



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

E = 0
A = 0
End If
A = A
Loop Until A <= 0
Else
U4.Text = 0
End If
End Sub

Private Sub XXX_Click()
FF1.Visible = True
XXX.Enabled = False
End Sub

```

รูปที่ 10 โปรแกรมควบคุมสแต็ปเปอร์มอเตอร์และตรวจสอบสัญญาณเซนเซอร์

```

Dim A, B, C, D As Integer
Private Sub Command1_Click()
FF3.Visible = True
End Sub

Private Sub Command4_Click()
Dim BBA, BBB, BBC, BBD As Integer
BBA = 1
BBB = 2
BBC = 3
BBD = 4
If Option1.Value = Checked Then
MSComm1.CommPort = BBA
If Option2.Value = Checked Then
MSComm1.CommPort = BBB
If Option3.Value = Checked Then
MSComm1.CommPort = BBC
If Option4.Value = Checked Then
MSComm1.CommPort = BBD
If Not MSComm1.PortOpen Then      'Open Port
MSComm1.PortOpen = True
End If

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

FF3.Visible = False
End Sub

Private Sub Command5_Click()
FF3.Visible = False
End Sub

Private Sub Command2_Click()
A = Val(Text8.Text)
B = 1000
C = A * B
Timer2.Interval = C
TM3.Interval = C
TM4.Interval = C
TM5.Interval = C
TM6.Interval = C
Frame1.Visible = False
End Sub

Private Sub Command3_Click()
Frame1.Visible = False
End Sub

Private Sub Command6_Click()
Dial.Text2.Text
TM3.Enabled = True
Label5.Visible = True
Text7.Visible = True
Text7.Text = Text2.Text
If Text10.Text = "" Then
Text10.Text = 1
Else
Text10.Text = Val(Text10.Text) + 1
End If
End Sub

Private Sub hag()
On Error Resume Next
If MSComm2.PortOpen = True Then
MSComm2.PortOpen = False
AddStatus "Port Closed"
End If
End Sub

Private Sub Command7_Click()
On Error Resume Next
If MSComm2.PortOpen = True Then
MSComm2.PortOpen = False
AddStatus "Port Closed"
End If
End Sub

Private Sub Dial(Number As String)

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

If MSComm2.PortOpen = True Then
    MSComm2.PortOpen = False
Else
    MSComm2.PortOpen = True
    MSComm2.Output = "ATDT " & Number & vbCrLf
End If
End Sub

```

```

Private Sub AddStatus(Text As String)
Text1.Text = Text1.Text & Text & vbCrLf
End Sub

```

```

Private Sub Command8_Click()
Dim micah As String
    MSComm2.PortOpen = True
    MSComm2.Output = "atdt" & vbCrLf
    MSComm2.InputMode = comInputModeText
    MSComm2.NullDiscard = False
    MSComm2.ParityReplace = "Micah Osborne"
    MSComm2.Tag = "micah"
micah = MSComm2.Input
Text1.Text = micah
Timer1_Timer
End Sub

```

```

Private Sub Command9_Click()
Frame1.Visible = True
'If FORM1.Height = 1095 Then
'FORM1.Height = 3480
'Exit Sub
'End If
'If FORM1.Height = 3480 Then
'FORM1.Height = 1095
'Exit Sub
'End If
End Sub

```

```

Private Sub Form_Load()
Timer1_Timer
A = Val(Text8.Text)
B = 1000
C = A * B
Timer2.Interval = C
TM3.Interval = C
TM4.Interval = C
TM5.Interval = C
TM6.Interval = C
Frame1.Visible = False
End Sub

```

```

Private Sub Slider1_Click()
If MSComm2.PortOpen = False Then
    MSComm2.PortOpen = True

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

End If
  MSComm2.Output = "AT " & "Ln" & vbCrLf
Timer1_Timer
End Sub

Private Sub Timer1_Timer()
Dim micah As String
Timer1.Interval = Timer1.Interval + 1
If MSComm2.PortOpen = True Then
micah = MSComm2.Input
If micah = "BUSY" Then
  MsgBox "The Phone line is busy."
  MSComm2.PortOpen = False
End If
If micah = "NO DIALTONE" Then
MsgBox "NO dial tone"
End If
Text1.Text = Text1.Text & micah
End If
End Sub

Private Sub Timer22_Timer()
Dim micah As String
Timer1.Interval = Timer1.Interval + 1
micah = MSComm2.Input
  MSComm2.PortOpen = False
Text1.Text = Text1.Text & micah
End Sub

Private Sub Timer2_Timer()
On Error Resume Next
If MSComm2.PortOpen = True Then
  MSComm2.PortOpen = False
AddStatus "Port Closed"
End If
Timer6.Enabled = False
TM6.Enabled = False
Label5.Visible = False

Text7.Visible = False
If Val(Text10.Text) < Val(Text9.Text) Then
Command6.Value = True

End If
Timer2.Enabled = False
End Sub

Private Sub Timer3_Timer()
Dial Text3.Text
TM4.Enabled = True
Text7.Text = Text3.Text
Timer3.Enabled = False
End Sub

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

Private Sub Timer4_Timer()
Dial Text4.Text
TM5.Enabled = True
Text7.Text = Text4.Text
Timer4.Enabled = False
End Sub

Private Sub Timer5_Timer()
Dial Text5.Text
TM6.Enabled = True
Text7.Text = Text5.Text
Timer5.Enabled = False
End Sub

Private Sub Timer6_Timer()
Dial Text6.Text
Timer2.Enabled = True
Text7.Text = Text6.Text
Timer6.Enabled = False
End Sub

Private Sub Timer7_Timer()
On Error Resume Next
If MSComm2.PortOpen = True Then
    MSComm2.PortOpen = False
    AddStatus "Port Closed"
End If
TM3.Enabled = False
Timer3.Enabled = False
Timer7.Enabled = False
Label5.Visible = False
Text7.Visible = False
End Sub

Private Sub Timer8_Timer()
A = Val(Text8.Text)
B = 1000
C = A * B
Timer2.Interval = C
TM3.Interval = C
TM4.Interval = C
TM5.Interval = C
TM6.Interval = C
Frame1.Visible = False
Timer8.Enabled = False
End Sub

Private Sub TM3_Timer()
On Error Resume Next
If MSComm2.PortOpen = True Then

    MSComm2.PortOpen = False
    AddStatus "Port Closed"

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

Private Sub Timer4_Timer()
Dial Text4.Text
TM5.Enabled = True
Text7.Text = Text4.Text
Timer4.Enabled = False
End Sub

Private Sub Timer5_Timer()
Dial Text5.Text
TM6.Enabled = True
Text7.Text = Text5.Text
Timer5.Enabled = False
End Sub

Private Sub Timer6_Timer()
Dial Text6.Text
Timer2.Enabled = True
Text7.Text = Text6.Text
Timer6.Enabled = False
End Sub

Private Sub Timer7_Timer()
On Error Resume Next
If MSComm2.PortOpen = True Then
    MSComm2.PortOpen = False
    AddStatus "Port Closed"
End If
TM3.Enabled = False
Timer3.Enabled = False
Timer7.Enabled = False
Label5.Visible = False
Text7.Visible = False
End Sub

Private Sub Timer8_Timer()
A = Val(Text8.Text)
B = 1000
C = A * B
Timer2.Interval = C
TM3.Interval = C
TM4.Interval = C
TM5.Interval = C
TM6.Interval = C
Frame1.Visible = False
Timer8.Enabled = False
End Sub

Private Sub TM3_Timer()
On Error Resume Next
If MSComm2.PortOpen = True Then

    MSComm2.PortOpen = False
    AddStatus "Port Closed"

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

End If
Timer3.Enabled = True
TM3.Enabled = False
End Sub

Private Sub TM4_Timer()

On Error Resume Next
If MSComm2.PortOpen = True Then
  MSComm2.PortOpen = False
AddStatus "Port Closed"
End If
Timer4.Enabled = True
TM4.Enabled = False
End Sub

Private Sub TM5_Timer()
On Error Resume Next
If MSComm2.PortOpen = True Then
  MSComm2.PortOpen = False
AddStatus "Port Closed"
End If
Timer5.Enabled = True
TM5.Enabled = False
End Sub

Private Sub TM6_Timer()
On Error Resume Next
If MSComm2.PortOpen = True Then
  MSComm2.PortOpen = False
AddStatus "Port Closed"
End If
Timer6.Enabled = True
TM6.Enabled = False
End Sub

```

รูปที่ 11 โปรแกรมทำงานโมเด็ม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

Private Sub Command1_Click()
If Val(Text1.Text) = Val(Text2.Text) Then
FORM1.Visible = True
Form2.Visible = True
Form3.Visible = False
Else
Frame1.Visible = True
End If
End Sub

```

```

Private Sub Command2_Click()
Form3.Visible = False
End
End Sub

```

```

Private Sub Command3_Click()
Frame1.Visible = False
Text1.Text = ""
End Sub

```

รูปที่ 12 โปรแกรมตรวจสอบรหัสผ่าน

```

Private Sub Command1_Click()
If Val(Text1.Text) = Val(Text4.Text) Then
Frame3.Visible = True
Frame2.Enabled = False
Command2.Visible = True

Command1.Enabled = False
Text1.Enabled = False
Text2.Text = ""
Text3.Text = ""
Else
Frame1.Visible = True
Frame1.Enabled = True
Command1.Enabled = False
End If
End Sub

```

```

Private Sub Command2_Click()
If Val(Text2.Text) = Val(Text3.Text) Then
Data1.Refresh
Form4.Visible = False
Frame3.Visible = False
Frame2.Enabled = True
Text1.Enabled = True
Text1.Text = ""
Command1.Enabled = True

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้





## รูปที่ 14 โปรแกรมป้อนรหัสผ่านบนเว็บเพจ

```
Main.html
<html>
<head>
<title>New Page 2</title>
<meta name="GENERATOR" content="Microsoft FrontPage 4.0">
<meta name="ProgId" content="FrontPage.Editor.Document">
</head>
<frameset rows="173,*">
  <frame name="banner" scrolling="no" noresize target="contents"
src="TITLE.htm">
  <frameset cols="216,*">
    <frame name="contents" target="main" src="TITLE2.htm"
scrolling="auto" noresize>
    <frame name="main" src="TITLE3.htm" target="_self"
scrolling="auto" noresize>
  </frameset>
</noframes>
<body>
<p>This page uses frames, but your browser doesn't support them.
</p>
</body>
</noframes>
</frameset>
</html>
title.html
<html>
<head>
<meta http-equiv="Content-Type" content="text/html;
charset=windows-1252">
<meta name="GENERATOR" content="Microsoft FrontPage 4.0">
<meta name="ProgId" content="FrontPage.Editor.Document">
<title>&#3619;&#3632;&#3610;&#3610;&#3648;&#3605;&#3639;&#3629;&#3
609;&#3616;&#3633;&#3618;&#3612;&#3641;&#3657;&#3610;&#3640;&#3585
;&#3619;&#3640;&#3585;&#3626;&#3635;&#3609;&#3633;&#3585;&#3591;&#
3634;&#3609;&#3612;&#3656;&#3634;&#3609;&#3648;&#3588;&#3619;&#363
9;&#3629;&#3586;&#3656;&#3634;&#3618;&#3629;&#3636;&#3609;&#3648;&#
3605;&#3629;&#3619;&#3660;&#3648;&#3609;&#3655;&#3605;</title>
<base target="contents">
</head>
<body background="bgbody1.gif">
<p align="left"></p>
<p align="center"><b><font face="MS Sans Serif" color="#000080"
size="6">&#3619;&#3632;&#3610;&#3610;&#3648;&#3605;&#3639;&#3629;&#
3609;&#3616;&#3633;&#3618;&#3612;&#3641;&#3657;&#3610;&#3640;&#35
85;&#3619;&#3640;&#3585;&#3626;&#3635;&#3609;&#3633;&#3585;&#3591;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

&#3634;&#3609;&#3612;&#3656;&#3634;&#3609;&#3648;&#3588;&#3619;&#3
639;&#3629;&#3586;&#3656;&#3634;&#3618;&#3629;&#3636;&#3609;&#3648
;&#3605;&#3629;&#3619;&#3660;&#3648;&#3609;&#3655;&#3605;</font>
</b></p>
<p align="center"><b><font face="Ms Sans Serif" color="#000080"
size="6">(Web
Based Office Security)</font></b></p>
</body>
</html>
title2.html
<html>
<head>
<meta http-equiv="Content-Type" content="text/html;
charset=windows-874">
<meta name="GENERATOR" content="Microsoft FrontPage 4.0">
<meta name="ProgId" content="FrontPage.Editor.Document">
<title>New Page 3</title>
<base target="main">
</head>
<body bgcolor="#808080" background="Ascent.jpg">
<p align="center"><font color="#FF00FF"
size="4"><b>&#3648;&#3617;&#3609;&#3641;&#3627;&#3621;&#3633;&#358
5;</b></font></p>
<p align="center"><a target="main" href="TEMP20.asp"><blink><font
color="#000080" size="2">ดูสัญญาณภาพจากกล้องวีซีดี</font></blink></a></p>
<p align="center"><blink><u><a target="main" href="ex10_04.asp">
<font color="#000080"
size="2">&#3588;&#3623;&#3610;&#3588;&#3640;&#3617;&#3585;&#3634;&#
3619;&#3607;&#3635;&#3627;&#3617;&#3640;&#3609;&#3586;&#3629;&#35
91;&#3585;&#3621;&#3657;&#3629;&#3591;</font></a></u></blink>
<p align="center"><a href="degree.asp" target="main"><font
color="#000080"
size="2">&#3649;&#3626;&#3604;&#3591;&#3588;&#3656;&#3634;&#3629;&#
3591;&#3624;&#3634;&#3585;&#3634;&#3619;&#3627;&#3617;&#3640;&#36
09;&#3586;&#3629;&#3591;&#3585;&#3621;&#3657;&#3629;&#3591;</font>
</a></p>
<p align="center">
<a href="TEMP11.ASP" target="main"><font color="#000080" size="2">
ข้อมูลจากตัวตรวจจับผู้บุกรุก</font></a>
</p>
<p align="center"><a href="changpass.asp" target="main"><font
color="#000080"
size="2">&#3648;&#3611;&#3621;&#3637;&#3656;&#3618;&#3609;&#3649;&#
3611;&#3621;&#3591;&#3619;&#3627;&#3633;&#3626;&#3612;&#3656;&#36
34;&#3609;</font></a></p>
</body>
</html>
title3.html
<html>
<head>

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

<meta http-equiv="Content-Type" content="text/html;
charset=windows-1252">
<meta name="GENERATOR" content="Microsoft FrontPage 4.0">
<meta name="ProgId" content="FrontPage.Editor.Document">
<title>#3618; #3636; #3609; #3604; #3637; #3605; #3657; #3629; #3
609; #3619; #3633; #3610; #3626; #3641; #3656; </title>
<base target="_self">
</head>
<body bgcolor="#000000">
<p align="center">&nbsp;</p>
<p align="center">&nbsp;</p>
<p align="center">&nbsp;</p>
<p align="center"><b><font face="MS Sans Serif" size="6"
color="#00FFFF">#3618; #3636; #3609; #3604; #3637; #3605; #3657; #
3629; #3609; #3619; #3633; #3610; #3626; #3641; #3656; </font></b>
</p>
<p align="center"><b><font face="MS Sans Serif" color="#00FFFF"
size="6">#3619; #3632; #3610; #3610; #3648; #3605; #3639; #3629; #
3609; #3616; #3633; #3618; #3612; #3641; #3657; #3610; #3640; #35
85; #3619; #3640; #3585; #3626; #3635; #3609; #3633; #3585; #3591;
#3634; #3609; #3612; #3656; #3634; #3609; #3648; #3588; #3619; #3
639; #3629; #3586; #3656; #3634; #3618; #3629; #3636; #3609; #3648
; #3605; #3629; #3619; #3660; #3648; #3609; #3655; #3605; </font>
</b></p>
<p align="center">&nbsp;</p>
</body>
</html>

```

### รูปที่ 15 โปรแกรมหน้าหลักของเว็บเพจ

```

Temp20.asp
<% ถ้าคลิกที่ปุ่ม Search และกรอกหมายเลข Barcode เรียบร้อยแล้ว ให้แสดงผลการสืบค้นข้อมูล %>
<%
Set ObjDB=Server.CreateObject("ADODB.Connection")
ObjDB.Open "STEP"
คำสั่ง SQL ให้แสดงข้อมูลทุกฟิลด์ในตาราง Book ของฐานข้อมูล
โดยแสดงเฉพาะเรคคอร์ดที่มีหมายเลข Barcode ตามที่กรอกเข้ามา
Sql="Select * From VIDEO Where VIDEO Like "%&Request.Form("VIDEO")&%"
ตั้งเอ็กซิควิวด์ Sql
Set ObjRS = ObjDB.Execute (Sql)
IF Session("Uname")="" THEN
Response.Redirect ("Index.ASP")
ELSE
END IF
If ObjRS.EOF Then

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ถ้าไม่พบข้อมูลให้ แสดงผลคำว่า ไม่พบข้อมูล

%>

<%

Response.End

End If

%>

<head>

<link rel="stylesheet" href="styles01.css" type="text/css">

</head>

<Center>

<Font Size=4 Face="Ms Sans Serif"><B>กรุณาเลือกกล่อง 1-4</B></Font>

<P>

<Table Bordercolor="#C0c0c0" Bgcolor="#Fcf9d8" Style="Ms Sans Serif" >

<%

Do While Not Objrs.EOF

%>

<%แสดงผลการสืบค้นข้อมูลจากไฟล์ต่างๆ ในตาราง แล้วแสดงผลลงในตาราง โดยมีปุ่ม Edit ต่อท้ายเพื่อแก้ไขข้อมูล

%>

<Form Action=TEMP21.asp Method=Post>

<select name=SELEC>

<option value=1 selected>กล่องที่ 1</option>

<option value=2>กล่องที่ 2</option>

<option value=3>กล่องที่ 3</option>

<option value=4>กล่องที่ 4</option>

</select>

<input Type=Submit Value=เลือกกล่อง id=Submit1 name=Submit1></td></tr>

</Form>

<%ฝังค่าตัวแปรที่ได้จากการ Search ไว้ใน Value เป็นตัวแปรคงที่ เพื่อส่งค่าไปให้ Ex10\_05.Asp เพื่อแก้ไข %>

<%

Objrs.MoveNext

Loop

%>

</Center>

</Table>

<Center><FONT SIZE="1" COLOR="#FFFFFF">.</Center></FONT>

<Center><FONT SIZE="1" COLOR="#FFFFFF">.</Center></FONT>

<Center><FONT SIZE="1" COLOR="#FFFFFF">.</Center></FONT>

<Center><FONT SIZE="1" COLOR="#FFFFFF">.</Center></FONT>

<Center><a href="title3.HTM"></a> </Center>

</Html>

temp21.asp

<%

Set ObjRS = Server.CreateObject("ADODB.Recordset")

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



```

<%
IF Session("Uname")="" THEN
  Response.Redirect ("index.ASP")
ELSE
  END IF
%>
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0 Transitional//EN">
<HTML>
<HEAD>
<TITLE> New Document </TITLE>
<META NAME="Generator" CONTENT="Microsoft FrontPage 4.0">
<META NAME="Author" CONTENT="">
<META NAME="Keywords" CONTENT="">
<META NAME="Description" CONTENT="">
</HEAD>
<BODY>
<p align="center">
<OBJECT id=NSPlay
codeBase=http://activex.microsoft.com/activex/controls/mplayer/en/
nsmpt2inf.cab#Version=5,1,52,701
classid=CLSID:22D6F312-B0F6-11D0-94AB-0080C74C7B95 width=320
height=240
type=application/x-oleobject
standby="Loading Microsoft Windows Media Player components..."
<PARAM NAME="AudioStream" VALUE="-1"><PARAM NAME="AutoSize"
VALUE="-1"><PARAM NAME="AutoStart" VALUE="-1"><PARAM
NAME="AnimationAtStart" VALUE="-1"><PARAM
NAME="AllowScan" VALUE="-1"><PARAM NAME="AllowChangeDisplaySize"
VALUE="-1"><PARAM NAME="AutoRewind" VALUE="0"><PARAM
NAME="Balance" VALUE="0"><PARAM NAME="BaseURL" VALUE=""><PARAM
NAME="BufferingTime" VALUE="5"><PARAM NAME="CaptioningID"
VALUE=""><PARAM NAME="ClickToPlay" VALUE="-1"><PARAM
NAME="CursorType" VALUE="0"><PARAM NAME="CurrentPosition" VALUE="-
1"><PARAM NAME="CurrentMarker" VALUE="0"><PARAM
NAME="DefaultFrame" VALUE=""><PARAM NAME="DisplayBackColor"
VALUE="0"><PARAM NAME="DisplayForeColor" VALUE="16777215"><PARAM
NAME="DisplayMode" VALUE="0"><PARAM NAME="DisplaySize" VALUE="0">
<PARAM NAME="Enabled" VALUE="-1"><PARAM NAME="EnableContextMenu"
VALUE="-1"><PARAM NAME="EnablePositionControls" VALUE="-1"><PARAM
NAME="EnableFullScreenControls" VALUE="0"><PARAM
NAME="EnableTracker" VALUE="-1"><PARAM NAME="Filename"
VALUE="http://san2000:8080"><PARAM NAME="InvokeURLs" VALUE="-1">
<PARAM NAME="Language" VALUE="-1"><PARAM NAME="Mute" VALUE="0">
<PARAM NAME="PlayCount" VALUE="1"><PARAM NAME="PreviewMode"
VALUE="0"><PARAM NAME="Rate" VALUE="1"><PARAM NAME="SAMILang"
VALUE=""><PARAM NAME="SAMISStyle" VALUE=""><PARAM
NAME="SAMIFilename" VALUE=""><PARAM NAME="SelectionStart" VALUE="-
1"><PARAM NAME="SelectionEnd" VALUE="-1"><PARAM

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไมอนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

NAME="SendOpenStateChangeEvents" VALUE="-1"><PARAM
NAME="SendWarningEvents" VALUE="-1"><PARAM NAME="SendErrorEvents"
VALUE="-1"><PARAM NAME="SendKeyboardEvents" VALUE="0"><PARAM
NAME="SendMouseClickEvents" VALUE="0"><PARAM
NAME="SendMouseMoveEvents" VALUE="0"><PARAM
NAME="SendPlayStateChangeEvents" VALUE="-1"><PARAM
NAME="ShowCaptioning" VALUE="0"><PARAM NAME="ShowControls"
VALUE="-1"><PARAM NAME="ShowAudioControls" VALUE="-1"><PARAM
NAME="ShowDisplay" VALUE="0"><PARAM NAME="ShowGotoBar" VALUE="-1">
<PARAM NAME="ShowPositionControls" VALUE="-1"><PARAM
NAME="ShowStatusBar" VALUE="0"><PARAM NAME="ShowTracker" VALUE="-
1">
<PARAM NAME="TransparentAtStart" VALUE="0"><PARAM
NAME="VideoBorderWidth" VALUE="0"><PARAM NAME="VideoBorderColor"
VALUE="0"><PARAM NAME="VideoBorder3D" VALUE="0"><PARAM
NAME="Volume" VALUE="0"><PARAM NAME="WindowlessVideo" VALUE="0">
<!--PARAM NAME="ShowControls" VALUE="0"--> <!--PARAM
NAME="ShowDisplay" VALUE="0"--> </OBJECT>
<p align="center"><b><font color="#0000ff" size="5">ระบบเตือนภัยผู้บุกรุก
สำนักงานผ่านเครือข่ายอินเทอร์เน็ต</font></b></p>
<p align="center"><b><a href="ex10_04.asp"><font size="3"
color="#ff0000"></font></a></b>&nbsp;</p>
<p align="center"><b><a href="temp20.asp"><font size="3"
color="#ff00ff"></font></a></b>&nbsp;</p>
<p align="right"><a href="main2.HTML"></a></p>
</BODY>
</HTML>

```

รูปที่ 17 โปรแกรมคู่มือสัญญาณภาพวีดีโอผ่านเว็บเพจ

```

<% ' ถ้าคลิกที่ปุ่ม Search และกรอกรหัสเลข CODE เรียบร้อยแล้ว ให้แสดงผลการสืบค้นข้อมูล %>
<%
Set ObjDB=Server.CreateObject("ADODB.Connection")
ObjDB.Open "STEP"
' คำสั่ง SQL ให้แสดงข้อมูลทุกฟิลด์ในตาราง STEP ของฐานข้อมูล
' โดยแสดงเฉพาะเรคคอร์ดที่มีหมายเลข CODE ตามที่กรอกเข้ามา
Sql="Select * From STEP Where CODE like '%" & Request.Form
("CODE") & "%'"
' สั่งเอ็กคิวต์ Sql
Set ObjRS = ObjDB.Execute (Sql)
IF Session("Uname")="" THEN
Response.Redirect ("index.ASP")
ELSE
END IF
If ObjRS.EOF Then

' ถ้าไม่พบข้อมูลให้ แสดงผลคำว่า ไม่พบข้อมูล
%>

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

<Html>
  <Body background=file://C:\TESTASP\midgard.jpg>
    <Font Size=4 Face="Ms Sans Serif"><B>ไม่พบข้อมูลรหัสผ่าน
    <Br><Br><A Href="Ex10_04.Asp">กลับไปป้อนรหัสผ่านใหม่</A>
  </Font></B></Center>

  </Body>
</Html>

<%
Response.End
End If
%>
<Center>
  <Font Size=4 Face="Ms Sans Serif"><B>กรุณาป้อนค่าองศาที่ต้องการ แต่ไม่เกินองศาที่ตั้งไว้</B></Font>
  <P>
  <Table Bordercolor="#C0c0c0" Style="Ms Sans Serif" >
    <%
Do While Not Objrs.EOF
%>
  <%' แสดงผลการสืบค้นข้อมูลจากที่ใดก็ตาม ๆ ในตาราง STEP แล้วแสดงผลลงในตาราง โดยมีปุ่ม Edit
ต่อท้ายเพื่อแก้ไขข้อมูล %>
  <Form Action=Ex10_05.Asp Method=Post id=form1 name=form1>
  <%' <Tr><Td>CODE : </Td><Td><Input Type=Hidden Name=CODE
Value=<%=Objrs("CODE")%><%'><%=Objrs("CODE")%>
  <td width="180">
  <table border="1" width="100%" bgcolor="#99CCFF">
  <tr>
  <Td width="50" height="23">ชื่อ</Td><Td width="30"
height="23"><Input Type=Text Name=L1 Value="<%=Objrs("L1")%>"
Size=4></Td>
  </tr>
  <tr>
  <Td width="50">นาม</Td><Td width="30"><Input Type=Text
Name=R1 Value="<%=Objrs("R1")%>" Size=4></Td>
  </tr>
  <tr>
  <Td width="50">นาม</Td><Td width="30"><Input Type=Text
Name=T1 Value="<%=Objrs("T1")%>" Size=4></Td>
  </tr>
  <tr>
  <Td width="50">ตำแหน่ง</Td><Td width="30"><Input Type=Text
Name=U1 Value="<%=Objrs("U1")%>" Size=4></Td>
  </tr>
  <Center>
  <Td width="168" colspan="2"> <Center>ควบคุมกล้องชุดที่ 1
</center></Td>
  </table>
</center>
</td>

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านธุรกิจ

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

<td width="50">
    &nbsp;
</td>
<td width="180">
    <table border="1" width="100%" bgcolor="#66FF99">
        <tr>
            <Td width="50">เข้า</Td><Td width="30"><Input Type=Text
Name=L2 Value="<%=Objrs("L2")%" Size=4></Td>
        </tr>
        <tr>
            <Td width="50">หา</Td><Td width="30"><Input Type=Text Name=R2
Value="<%=Objrs("R2")%" Size=4></Td>
        </tr>
        <tr>
            <Td width="50">บ</Td><Td width="30"><Input Type=Text
Name=T2 Value="<%=Objrs("T2")%" Size=4></Td>
        </tr>
        <tr>
            <Td width="50">ล</Td><Td width="30"><Input Type=Text
Name=U2
Value="<%=Objrs("U2")%" Size=4></Td>
        </tr>
        <Center>
            <Td width="168" colspan="2"> <Center>ตามแบบกล่องชุดที่2
</center></Td>
        </table>
    </center>
</td>
<td width="50">
    &nbsp;
</td>
<td width="180">
    <table border="1" width="100%" bgcolor="#CCCCFF">
        <tr>
            <Td width="50">เข้า</Td><Td width="30"><Input Type=Text
Name=L3 Value="<%=Objrs("L3")%" Size=4></Td>
        </tr>
        <tr>
            <Td width="50">หา</Td><Td width="30"><Input Type=Text
Name=R3 Value="<%=Objrs("R3")%" Size=4></Td>
        </tr>
        <tr>
            <Td width="50">บ</Td><Td width="30"><Input Type=Text
Name=T3 Value="<%=Objrs("T3")%" Size=4></Td>
        </tr>
        <tr>
            <Td width="50">ล</Td><Td width="30"><Input Type=Text
Name=U3 Value="<%=Objrs("U3")%" Size=4></Td>
        </tr>
    </table>

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

<Center>
    <Td width="168" colspan="2"> <Center>ถามคุณกตัญญูครั้งที่3
</center></Td>
</table>
</center>
</td>
<td width="50">
    &nbsp;
</td>
<td width="180">
    <table border="1" width="100%" bgcolor="#CCFFCC">
        <tr>
            <Td width="50">จำ</Td><Td width="30"><Input Type=Text
Name=L4 Value="<%=Objrs("L4")%" Size=4></Td>
        </tr>
        <tr>
            <Td width="50">พ</Td><Td width="30"><Input Type=Text
Name=R4 Value="<%=Objrs("R4")%" Size=4></Td>
        </tr>
        <tr>
            <Td width="50">ท</Td><Td width="30"><Input Type=Text
Name=T4 Value="<%=Objrs("T4")%" Size=4></Td>
        </tr>
        <tr>
            <Td width="50">ด</Td><Td width="30"><Input Type=Text Name=U4
Value="<%=Objrs("U4")%" Size=4></Td>
        </tr>
    <Center>
        <Td width="168" colspan="2"> <Center>ถามคุณกตัญญูครั้งที่4
</center></Td>
</table>
</td>
<tr>
<Td width="920" colspan="7"> . </center></Td>
</tr>
<tr>
    <Td width="920" colspan="7"><Center><Input Type=Submit
Value=" คกลง " id=Submit1 name=Submit1></Center></TD>
</tr>
</Form>
    <% ฝั่งคำตัวแปรที่ได้จากการ Search ไว้ใน Value เป็นตัวแปรคงที่ เพื่อส่งค่าไปให้
Ex10_05.Asp เพื่อแก้ไข %>
    <%
ObjRS.Movenext
Loop
%>

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
</Center>
</Table>
</Body>
</Html>
```

### รูปที่ 18 โปรแกรมป้อนค่าองศาผ่านเว็บเพจ

```
<%
Set ObjRS = Server.CreateObject("ADODB.Recordset")
SQL="Select * From STEP Where CODE Like '%" & Request.Form
("CODE") & "%'"
ObjRS.Open SQL, "STEP" ,1,3
ObjRS("L1") = Request.Form("L1")
ObjRS("R1") = Request.Form("R1")
ObjRS("T1") = Request.Form("T1")
ObjRS("U1") = Request.Form("U1")

ObjRS("L2") = Request.Form("L2")
ObjRS("R2") = Request.Form("R2")
ObjRS("T2") = Request.Form("T2")
ObjRS("U2") = Request.Form("U2")

ObjRS("L3") = Request.Form("L3")
ObjRS("R3") = Request.Form("R3")
ObjRS("T3") = Request.Form("T3")
ObjRS("U3") = Request.Form("U3")

ObjRS("L4") = Request.Form("L4")
ObjRS("R4") = Request.Form("R4")
ObjRS("T4") = Request.Form("T4")
ObjRS("U4") = Request.Form("U4")
'New
Set ObjRS2 = Server.CreateObject("ADODB.Recordset")
SQL="Select * From STEPDEGREE"
ObjRS2.Open SQL, "STEP" ,1,3
IF (ObjRS("L1") > (ObjRS2("MAXDEGREE") - ObjRS2("L1"))) then
ObjRS("L1") = (ObjRS2("MAXDEGREE") - ObjRS2("L1"))
else
end if
IF (ObjRS("R1") > (ObjRS2("MAXDEGREE") - ObjRS2("R1"))) then
ObjRS("R1") = (ObjRS2("MAXDEGREE") - ObjRS2("R1"))
else
end if
IF (ObjRS("T1") > (ObjRS2("MAXDEGREET") - ObjRS2("T1"))) then
ObjRS("T1") = (ObjRS2("MAXDEGREET") - ObjRS2("T1"))
Else
end if
IF (ObjRS("U1") > (ObjRS2("MAXDEGREET") - ObjRS2("U1"))) then
ObjRS("U1") = (ObjRS2("MAXDEGREET") - ObjRS2("U1"))
Else
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

end if
IF (ObjRS("L2") > (ObjRS2("MAXDEGREE") - ObjRS2("L2"))) then
ObjRS("L2") = (ObjRS2("MAXDEGREE") - ObjRS2("L2"))
else
end if
IF (ObjRS("R2") > (ObjRS2("MAXDEGREE") - ObjRS2("R2"))) then
ObjRS("R2") = (ObjRS2("MAXDEGREE") - ObjRS2("R2"))
Else
end if
IF (ObjRS("T2") > (ObjRS2("MAXDEGREET") - ObjRS2("T2"))) then
ObjRS("T2") = (ObjRS2("MAXDEGREET") - ObjRS2("T2"))
else
end if
IF (ObjRS("U2") > (ObjRS2("MAXDEGREET") - ObjRS2("U2"))) then
ObjRS("U2") = (ObjRS2("MAXDEGREET") - ObjRS2("U2"))
else
end if
IF (ObjRS("L3") > (ObjRS2("MAXDEGREE") - ObjRS2("L3"))) then
ObjRS("L3") = (ObjRS2("MAXDEGREE") - ObjRS2("L3"))
else
end if
IF (ObjRS("R3") > (ObjRS2("MAXDEGREE") - ObjRS2("R3"))) then
ObjRS("R3") = (ObjRS2("MAXDEGREE") - ObjRS2("R3"))
else
end if
IF (ObjRS("T3") > (ObjRS2("MAXDEGREET") - ObjRS2("T3"))) then
ObjRS("T3") = (ObjRS2("MAXDEGREET") - ObjRS2("T3"))
else
end if
IF (ObjRS("U3") > (ObjRS2("MAXDEGREET") - ObjRS2("U3"))) then
ObjRS("U3") = (ObjRS2("MAXDEGREET") - ObjRS2("U3"))
else
end if
IF (ObjRS("L4") > (ObjRS2("MAXDEGREE") - ObjRS2("L4"))) then
ObjRS("L4") = (ObjRS2("MAXDEGREE") - ObjRS2("L4"))
else
end if
IF (ObjRS("R4") > (ObjRS2("MAXDEGREE") - ObjRS2("R4"))) then
ObjRS("R4") = (ObjRS2("MAXDEGREE") - ObjRS2("R4"))
else
end if
IF (ObjRS("T4") > (ObjRS2("MAXDEGREET") - ObjRS2("T4"))) then
ObjRS("T4") = (ObjRS2("MAXDEGREET") - ObjRS2("T4"))
else
end if
IF (ObjRS("U4") > (ObjRS2("MAXDEGREET") - ObjRS2("U4"))) then
ObjRS("U4") = (ObjRS2("MAXDEGREET") - ObjRS2("U4"))
else
end if
'End New
ObjRS.Update

```

'เปิดฐานข้อมูลเพื่อสืบค้นจนพบแล้วแก้ไข

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

%>
<BODY>
  <HTML>
    <CENTER>
      <FONT SIZE=4 FACE="MS Sans Serif"><B>ส่งข้อมูล เรียบร้อยแล้ว
      <Br><Br><A Href=degree.asp>ดูองศาการหมุนของกล้อง</A></Font></B>
    </CENTER>
  </BODY>
</HTML>

```

### รูปที่ 19 โปรแกรมบันทึกข้อมูลองศาจากเว็บเพจลงในฐานข้อมูล

```

<% 'ถ้าคลิกปุ่ม Search และกรอกหมายเลข CODE เรียบร้อยแล้ว ให้แสดงผลการสืบค้นข้อมูล %>
<%
Set ObjDB=Server.CreateObject("ADODB.Connection")
ObjDB.Open "STEP"
'คำสั่ง SQL ให้แสดงข้อมูลทุกฟิลด์ในตาราง STEP ของฐานข้อมูล
'โดยแสดงเฉพาะเรกคอร์ดที่มีหมายเลข CODE ตามที่กรอกเข้ามา
Sql="Select * From STEPDEGREE Where CODE Like '%"&Request.Form
("CODE")&"%"
' ตั้งเอ็กซีกิวต์ Sql
Set ObjRS = ObjDB.Execute Sql)
IF Session("Uname")="" THEN
Response.Redirect ("index.ASP")
ELSE
END IF
If ObjRS.Eof Then
' ถ้าไม่พบข้อมูลให้ แสดงผลคำว่า ไม่พบข้อมูล
%>
<Html>
<Body background=file:///C:/TESTASP/midgard.jpg>
<Font Size=4 Face="Ms Sans Serif"><B>ไม่พบข้อมูลรหัสผ่าน
<Br><Br><A Href="INDEX.Asp">กลับไปป้อนรหัสผ่านใหม่</A></Font></B>
</Center>
</Body>
</Html>
<%
Response.End
End If
%>
<Center>
<Font Size=4 Face="Ms Sans Serif"><B>แสดงค่าองศาที่การหมุนของกล้อง</B>
</Font>
<P>
<Table Bordercolor="#C0c0c0" Style="Ms Sans Serif" >
<%
Do While Not Objrs.Eof

```

เอกสารนี้เป็นเอกสารสงวนลิขสิทธิ์สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

%>
<% 'แสดงผลการสืบค้นข้อมูลจากฟิลด์ต่าง ๆ ในตาราง STEP แล้วแสดงผลลงในตาราง โดยมีปุ่ม Edit
ต่อท้ายเพื่อแก้ไขข้อมูล %>
<Form Action=DEGREE.Asp Method=Post id=form1 name=form1>
<% ' <Tr><Td>CODE : </Td><Td><Input Type=Hidden Name=CODE
Value=<%=Objrs("CODE") %><% ' ><%=Objrs("CODE") %>
<td width="180">

<table border="1" width="100%" bgcolor="#99CCFF">
<tr>
<Td width="50" height="23">ชื่อ</Td><Td width="30"
bgcolor="#FF99FF" height="23"><%=Objrs("L1") %></Td>
</tr>
<tr>
<Td width="50">นาม</Td><Td width="30"
bgcolor="#FF99FF"><%=Objrs("R1") %></Td>
</tr>
<tr>
<Td width="50">นาม</Td><Td width="30"
bgcolor="#FF99FF"><%=Objrs("T1") %></Td>
</tr>
<tr>
<Td width="50">นาม</Td><Td width="30"
bgcolor="#FF99FF"><%=Objrs("U1") %></Td>
</tr>
<Center>
<Td width="168" bgcolor="#C0C0C0" colspan="2">
<Center>ควบคุมห้องชุดที่</center></Td>
</table>
</center>
</td>
<td width="50">
&nbsp;
</td>
<td width="180">
<table border="1" width="100%" bgcolor="#66FF99">
<tr>
<Td width="50">ชื่อ</Td><Td width="30"
bgcolor="#FF99FF"><%=Objrs("L2") %></Td>
</tr>
<tr>
<Td width="50">นาม</Td><Td width="30"
bgcolor="#FF99FF"><%=Objrs("R2") %></Td>
</tr>
<tr>
<Td width="50">นาม</Td><Td width="30"
bgcolor="#FF99FF"><%=Objrs("T2") %></Td>
</tr>

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

<tr>
  <Td width="50">ล่าง</Td><Td width="30"
bgcolor="#FF99FF"><%=Objrs("U2") %></Td>
</tr>
<Center>
  <Td width="168" bgcolor="#C0C0C0" colspan="2">
<Center>ควบคุมกล่องชุดที่2</center></Td>
</table>
</center>
</td>
<td width="50">
  &nbsp;
</td>
<td width="180">
  <table border="1" width="100%" bgcolor="#CCCCFF">
    <tr>
      <Td width="50">ซ้าย</Td><Td width="30"
bgcolor="#FF99FF"><%=Objrs("L3") %></Td>
</tr>
<tr>
<Td width="50">ขวา</Td><Td width="30" bgcolor="#FF99FF"><%=Objrs
("R3") %></Td>
</tr>
<tr>
  <Td width="50">บน</Td><Td width="30"
bgcolor="#FF99FF"><%=Objrs("T3") %></Td>
</tr>
<tr>
  <Td width="50">ล่าง</Td><Td width="30"
bgcolor="#FF99FF"><%=Objrs("U3") %></Td>
</tr>
<Center>
  <Td width="168" bgcolor="#C0C0C0" colspan="2">
<Center>ควบคุมกล่องชุดที่3</center></Td>
</table>
</center>
</td>
<td width="50">

  &nbsp;
</td>
<td width="180">
  <table border="1" width="100%" bgcolor="#CCFFCC">
    <tr>
      <Td width="50">ซ้าย</Td><Td width="30"
bgcolor="#FF99FF"><%=Objrs("L4") %></Td>
</tr>
<tr>
      <Td width="50">ขวา</Td><Td width="30"
bgcolor="#FF99FF"><%=Objrs("R4") %></Td>

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

        </tr>
        <tr>
            <Td width="50">บน</Td><Td width="30"
bgcolor="#FF99FF"><%=Objrs ("T4") %></Td>
        </tr>
        <tr>
            <Td width="50">ล่าง</Td><Td width="30"
bgcolor="#FF99FF"><%=Objrs ("U4") %></Td>
        </tr>
    <Center>
        <Td width="168" bgcolor="#C0C0C0" colspan="2">
<Center>ควบคุมกล่องชุดที่4</center></Td>
    </table>
</td>
<tr>

</Form>
<% 'ฝั่งค่าตัวแปรที่ได้จากการ Search ไว้ใน Value เป็นตัวแปรคงที่ เพื่อส่งค่าไปให้
Ex10_05.Asp เพื่อแก้ไข %>
<%
ObjRS.Movenext
Loop
%>
</Center>
</Table>
<p align="center">&nbsp;</p>
<p align="center"><b><font size="4" color="#FF0000"><a
href="temp20.ASP">ดูลักษณะภาพ</a></font></b></p>
<p align="center"><b><font size="4" color="#FF0000"><a
href="ex10_04.asp">
กลับไปป้อนองศา</a></font></b></p>
<p align="right"><a href="title3.HTM"></a></p>
</Body>
</Html>

```

รูปที่ 20 โปรแกรมแสดงองศาการหมุนของสเต็ปเปอร์มอเตอร์ผ่านเว็บเพจ

```

<% 'ถ้าคลิกที่ปุ่ม Search และกรอกหมายเลข CODE เรียบร้อยแล้ว ให้แสดงผลการสืบค้นข้อมูล %>
<%
Set ObjDB=Server.CreateObject("ADODB.Connection")
ObjDB.Open "STEP"

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

'คำสั่ง SQL ให้แสดงข้อมูลทุกฟิลด์ในตาราง STEP ของฐานข้อมูล

' โดยแสดงเฉพาะเรคคอร์ดที่มีหมายเลข CODE ตามที่กรอกเข้ามา

```
Sql="Select * From DANG Where DANG like '%" & Request.Form  
("DANG") & "%'"
```

' ตั้งเอ็กซิคิวต์ Sql

```
Set ObjRS = ObjDB.Execute (Sql)
```

```
IF Session("Uname")="" THEN
```

```
Response.Redirect ("index.ASP")
```

```
ELSE
```

```
END IF
```

```
If ObjRS.EOF Then
```

' ถ้าไม่พบข้อมูลให้ แสดงผลคำว่า ไม่พบข้อมูล

```
%>
```

```
<html>
```

```
<body background=file://C:\TESTASP\midgard.jpg>
```

```
<font size=4 face="Ms Sans Serif"><b>ไม่พบข้อมูลที่ค้นหา
```

```
<br><br><a href="INDEX.Asp">กลับไปอ่านที่หน้าใหม่</a></font></b>
```

```
</center>
```

```
</body>
```

```
</html>
```

```
<%
```

```
Response.End
```

```
End If
```

```
%>
```

```
<center>
```

```
<font size=4 face="Ms Sans Serif"><b>แสดงข้อมูลจากตัวตารางจับคู่กรอก</b>
```

```
</font>
```

```
<p>
```

```
<table bordercolor="#C0C0C0" style="Ms Sans Serif" >
```

```
<%
```

```
Do While Not ObjRS.EOF
```

```
%>
```

<% ' แสดงผลการสืบค้นข้อมูลจากไฟล์ต่าง ๆ ในตาราง STEP แล้วแสดงผลลงในตาราง โดยมีปุ่ม Edit ค่อย้ายเพื่อแก้ไขข้อมูล %>

```
<form action=DEGREE.Asp method=Post id=form1 name=form1>
```

```
<% ' <tr><td>CODE : </td><td><input type=hidden name=CODE value=
```

```
<%=ObjRS ("CODE") %><% ' ><%=ObjRS ("CODE") %>
```

```
<td width="320">
```

```
<table border="1" width="100%" bgcolor="#99CCFF">
```

```
<tr>
```

```
<td width="120" height="23">เซ็นเซอร์ตัวที่ 1</td><td width="200" bgcolor="#FF99FF" height="23"><%=ObjRS ("T1") %></td>
```

```
</tr>
```

```
<tr>
```

```
<td width="120">เซ็นเซอร์ตัวที่ 2</td><td width="200" bgcolor="#FF99FF">
```

```
<%=ObjRS
```

```
("T2") %></td>
```

```
</tr>
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

<tr>
<Td width="120">เซ็นเซอร์ตัวที่ 3</Td><Td width="200" bgcolor="#FF99FF">
<%=Objrs ("T3") %></Td>
</tr>
<tr>
<Td width="120">เซ็นเซอร์ตัวที่ 4</Td><Td width="200" bgcolor="#FF99FF">
<%=Objrs ("T4") %></Td>
</tr>
<tr>
<Td width="120">เซ็นเซอร์ตัวที่ 5</Td><Td width="200" bgcolor="#FF99FF">
<%=Objrs ("T5") %></Td>
</tr>
<tr>
<Td width="120">เซ็นเซอร์ตัวที่ 6</Td><Td width="200" bgcolor="#FF99FF">
<%=Objrs ("T6") %></Td>
</tr>
<tr>
<Td width="120">เซ็นเซอร์ตัวที่ 7</Td><Td width="200" bgcolor="#FF99FF">
<%=Objrs ("T7") %></Td>
</tr>
<tr>
<Td width="120">เซ็นเซอร์ตัวที่ 8</Td><Td width="200" bgcolor="#FF99FF">
<%=Objrs ("T8") %></Td>
</tr>
<Center>
<Td width="320" bgcolor="#C0C0C0" colspan="2"> <Center>ถาวรทุกกล่องจุดที่1
</center></Td>
</table>
</center>
<td width="50">
&ampnbsp
</td>
</Form>
<% ' ฝั่งค่าตัวแปรที่ได้จากการ Search ไว้ใน value เป็นตัวแปรคงที่ เพื่อส่งค่าไปให้ Ex10_05.Asp เพื่อ
แก้ไข %>
<%
ObjRS.Movenext
Loop
%>
</Center>
</Table>
<p align="center">&nbsp;</p>
<p align="center"><b><font size="4" color="#FF0000"><a
href="temp20.ASP">ดูสัญญาณภาพ</a></font></b></p>
<p align="center"><b><font size="4" color="#FF0000"><a
href="ex10_04.asp">กลับไปป้อนองศา</a></font></b></p>
<p align="right"><a href="title3.HTM"></a></p>

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับบริการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาติให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

</Body>  
</Html>

## รูปที่ 21 โปรแกรมแสดงข้อมูลผู้บุกรุกผ่านเว็บเพจ



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้