

หุ่นยนต์รักษาความปลอดภัยภายในบ้าน

HOME SECURITY ROBOT



เลขหมู่.....  
เลขทะเบียน 62167  
วัน,เดือน,ปี 31 ก.ค. 2549

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรวิศวกรรมศาสตรบัณฑิต  
ภาควิชาวิศวกรรมสารสนเทศ  
คณะวิศวกรรมศาสตร์  
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง  
ปีการศึกษา 2547

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

# HOME SECURITY ROBOT

BY

MR. PONGSATORN TONGLAIROUM

MR. PORNNATEE MALEEWAT

MR. POLLAWIT KOWSAARD



A THESIS SUBMITTED IN PARTIAL FULFILLMENT OF  
THE REQUIREMENT FOR THE DEGREE OF  
BACHELOR IN DEPARTMENT OF INFORMATION ENGINEERING  
FACULTY OF ENGINEERING  
KING MONGKUT'S INSTITUTE OF TECHNOLOGY LADKRABANG

2004

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ชื่อโครงการภาษาไทย

หุ่นยนต์รักษาความปลอดภัยภายในบ้าน

THESIS TITLE

HOME SECURITY ROBOT

โดย

นายพงศธร ทองไทรรวม รหัสนักศึกษา 44010311

นายพรนที มาลีวัตร รหัสนักศึกษา 44010318

นายพลวิษญ์ ขาวสะอาด รหัสนักศึกษา 44010322

อาจารย์ที่ปรึกษา

ผศ. มยุรี เลิศเวชกุล

ภาควิชา

วิศวกรรมสารสนเทศ

ปีการศึกษา

2547

ปริญญานิพนธ์ฉบับนี้ได้รับการอนุมัติให้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตร  
วิศวกรรมศาสตรบัณฑิต คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหาร  
ลาดกระบัง



( ผศ. มยุรี เลิศเวชกุล )

อาจารย์ผู้ควบคุมปริญญานิพนธ์

ลิขสิทธิ์ของคณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หัวข้อปริญญานิพนธ์	หุ่นยนต์รักษาความปลอดภัยภายในบ้าน	
ชื่อนักศึกษา	นายพงศธร ทองไหลรวม	รหัสประจำตัว 44010311
	นายพรนที มาลีวัตร	รหัสประจำตัว 44010318
	นายพลวิษณุ ขาวสะอาด	รหัสประจำตัว 44010322
อาจารย์ที่ปรึกษา	ผศ. มยุรี เลิศเวชกุล	
ระดับการศึกษา	ปริญญาตรี วิศวกรรมศาสตรบัณฑิต	
	สาขาวิศวกรรมสารสนเทศ	
ภาควิชา	วิศวกรรมสารสนเทศ	
ปีการศึกษา	2547	

### บทคัดย่อ

โครงการหุ่นยนต์รักษาความปลอดภัยภายในบ้านนี้เป็นการพัฒนาหุ่นยนต์และระบบการควบคุมหุ่นยนต์เพื่อใช้ในการรักษาความปลอดภัยภายในบ้านเมื่อไม่มีคนอยู่ โดยอาศัยเทคโนโลยีเครือข่ายอินเทอร์เน็ตและเครือข่ายโทรศัพท์เคลื่อนที่ ซึ่งในปัจจุบันนี้มีความทันสมัยและสะดวกสบายในการใช้งาน ทำให้สามารถเข้าถึงระบบได้จากที่ใดก็ได้ที่มีเครือข่ายอินเทอร์เน็ตหรือเครือข่ายโทรศัพท์เคลื่อนที่

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

**Thesis Title** Home Security Robot  
**Student** Mr. Pongsatorn Tonglairoum ID. 44010311  
Mr. Pornnatee Maleewat ID. 44010318  
Mr. Pollawit Kowsaard ID. 44010322  
**Advisor** Asst.Prof. Mayuree Lertvechakul  
**Graduate Level** Bachelor Degree of Information Engineering  
**Academic Year** 2004

### Abstract

This project develops an indoor robot that could be used within a house area. The robot automatically goes around the house, detects a movement, temperature and could alert to the house owner if irregular event happened. The picture of the detected area will be capture and send via MMS and e-mail. Moreover user can control the direction and action of the robot via mobile interface or web application through the access server.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## กิตติกรรมประกาศ

ขอขอบคุณท่านอาจารย์ที่ปรึกษาอาจารย์มยุรี เลิศเวชกุล พี่ๆ เพื่อนๆ ทุกคนที่ได้ให้คำปรึกษาอันเป็นประโยชน์ ขอขอบคุณลูกท้อและหนึ่งที่เอื้อเพื่อช่วยเหลือด้านอุปกรณ์การทดลองที่เป็นประโยชน์อย่างมาก ขอขอบคุณผู้ปกครองที่สนับสนุนด้านเงินทุนและกำลังใจอันมีค่า ขอขอบคุณเพื่อนร่วมกลุ่มที่มีความรับผิดชอบช่วยเหลือซึ่งกันและกันเป็นอย่างดีทำให้งานที่ยากบรรเทาลงเป็นงานที่ง่ายและสนุกสนานในการทำงานและสุดท้ายขอขอบคุณทุกท่านที่ได้ให้ความช่วยเหลือที่ไม่ได้กล่าวถึงในที่นี้



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## สารบัญ

เรื่อง	หน้า
บทคัดย่อภาษาไทย	ก
บทคัดย่อภาษาอังกฤษ	ข
กิตติกรรมประกาศ	ค
สารบัญ	ง
สารบัญตาราง	ช
สารบัญภาพ	ซ
บทที่ 1 บทนำ	
1.1 ความสำคัญและที่มา	1
1.2 วัตถุประสงค์ของโครงการ	1
1.3 ขอบเขตของโครงการ	1
1.4 สถาปัตยกรรมโดยรวม	2
1.5 ขั้นตอนการดำเนินงาน	4
บทที่ 2 ทฤษฎีและหลักการที่เกี่ยวข้อง	5
2.1 WAP	5
2.1.1 WAP Programming Model	5
2.1.2 Wireless Application Environment (WAE)	6
2.1.3 WAP Protocol Stack	7
2.1.4 ระดับชั้นของโปรโตคอลแบบเก่า	7
2.1.5 ระดับชั้นของโปรโตคอลแบบใหม่	10
2.2 Java Technology	11
2.2.1 เซิร์ฟเล็ต (Servlets)	12
2.2.2 Java Media Framework (JMF)	14
2.3 สื่อแบบสตรีมมิ่ง (Streaming)	20
2.3.1 Real-Time Transport Protocol	21

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## สารบัญ

(ต่อ)

เรื่อง	หน้า
2.4 BARCODE หรือ รหัสแท่ง	22
2.4.1 ส่วนประกอบของบาร์โค้ด	22
2.4.2 หลักการทำงาน	23
2.4.3 ตัวอย่างรหัสบาร์โค้ด (Code39)	23
2.5 อินฟราเรดเซ็นเซอร์ (Infrared Sensor)	25
2.6 ระบบสื่อสารข้อมูลอนุกรมแบบหนึ่งสาย (1-Wire™ Serial Bus)	25
2.7 ไอซีตรวจจับอุณหภูมิ DS1820	26
2.8 ไมโครคอนโทรลเลอร์	28
2.8.1 โครงสร้างของพอร์ต	28
2.8.2 การจัดหน่วยความจำของ MCS-51	29
2.8.3 รีจิสเตอร์เลือกโหมดการทำงานของไทเมอร์/เคาน์เตอร์ (TMOD)	29
2.8.4 การรับส่งข้อมูลอนุกรม ใน MCS-51	30
2.9 เซอร์โวมอเตอร์ (Servo motor)	36
2.9.1 หลักการทำงานของเซอร์โวมอเตอร์	37
2.9.2 การปรับแต่งของเซอร์โวมอเตอร์	38
บทที่ 3 โครงสร้างและการออกแบบ	39
3.1 ฮาร์ดแวร์	39
3.1.1 ส่วนหุ่นยนต์	39
3.1.2 ส่วนรับส่งข้อมูลระหว่างเซิร์ฟเวอร์กับหุ่นยนต์	48
3.2 ซอฟต์แวร์	49
3.2.1 โปรแกรมควบคุมหุ่นยนต์	49
3.2.2 โปรแกรมควบคุมการรับส่งข้อมูลระหว่างเซิร์ฟเวอร์กับหุ่นยนต์	52
3.2.3 โปรแกรมควบคุมที่เซิร์ฟเวอร์	54
3.2.4 โปรแกรมติดต่อระหว่างเซิร์ฟเวอร์กับโทรศัพท์เคลื่อนที่	59
3.3 โปรแกรมติดต่อระหว่างเซิร์ฟเวอร์ โคลเอนต์ผ่านเครือข่ายอินเทอร์เน็ต	60

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## สารบัญ

(ต่อ)

เรื่อง	หน้า
บทที่ 4 การทดสอบและวิเคราะห์ผล	62
4.1 ทดสอบการรับภาพจากกล้องผ่านคลื่นวิทยุ	62
4.2 ทดสอบการวัดอุณหภูมิ	63
4.3 การทดลองการแจ้งเตือนเมื่อสามารถตรวจจับการเคลื่อนไหว	64
4.4 การทดลองการอ่านรหัสบาร์โค้ด	65
4.5 การทดลองติดต่อผ่านโทรศัพท์เคลื่อนที่	68
4.6 การทดลองติดต่อผ่านอินเทอร์เน็ต	71
4.7 การทดลองโปรแกรมควบคุมที่เซิร์ฟเวอร์	74
บทที่ 5 สรุปผลและวิจารณ์	78
5.1 สรุปผล	78
5.2 ปัญหาที่พบ	79
5.3 แนวทางในการพัฒนาต่อ	79
บรรณานุกรม	81
ภาคผนวก	82

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## สารบัญตาราง

	หน้า
ตารางที่ 2.1 ตารางแสดงการเข้ารหัสความกว้างและรหัสไบนารีของรหัส ASCII	24
ตารางที่ 2.2 การแปลงค่าออคทามิเป็นเลขฐานสิบ และฐานสิบหก	27
ตารางที่ 2.3 การเลือกการทำงานของ Timer/Counter	30
ตารางที่ 3.1 การป้อนสัญญาณควบคุมไอซี TA7279P	41
ตารางที่ 3.2 การรับแสงสะท้อนของตัวรับแสงอินฟราเรด	47
ตารางที่ 4.1 ผลการทดลองการอ่านรหัสบาร์โค้ดสัญลักษณ์ที่ 1	65
ตารางที่ 4.2 ผลการทดลองการอ่านรหัสบาร์โค้ดสัญลักษณ์ที่ 2	66



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## สารบัญภาพ

	หน้า
รูปที่ 1.1 ภาพรวมของระบบ	2
รูปที่ 2.1 WAP Programming Model	6
รูปที่ 2.2 การใช้ WAP Proxy เพื่อเพิ่มประสิทธิภาพการใช้งานเครือข่าย	6
รูปที่ 2.3 ตัวอย่าง WAP Gateway แบบเก่า	9
รูปที่ 2.4 ตัวอย่าง WAP Gateway แบบใหม่	10
รูปที่ 2.5 WAP Stack แบบคู่	11
รูปที่ 2.6 Servlet Engine	13
รูปที่ 2.7 ระบบการบันทึก ประมวลผล และการแสดงผลของข้อมูลสื่อเชิงเวลา	15
รูปที่ 2.8 รูปแบบการประมวลผลข้อมูลสื่อ	16
รูปที่ 2.9 รูปแบบของ Player ของ JMF	17
รูปที่ 2.10 สถานะของ Player	18
รูปที่ 2.11 รูปแบบของ Processor ของ JMF	18
รูปที่ 2.12 สถานะของ Processor	19
รูปที่ 2.13 สถาปัตยกรรมของ RTP	21
รูปที่ 2.14 ส่วนประกอบของบาร์โค้ด	22
รูปที่ 2.15 ตัวอย่างรหัส code39 แทน ABC	23
รูปที่ 2.16 ไอซี DS1820	27
รูปที่ 2.17 การจัดสรรพื้นที่ของสแควร์แพด	27
รูปที่ 2.18 โครงสร้างพอร์ตทั้ง 4 ของ MCS-51	29
รูปที่ 2.19 รีจิสเตอร์เลือกโหมดการทำงานของ ไทมเมอร์/คาน์เตอร์	30
รูปที่ 2.20 รีจิสเตอร์ควบคุมพอร์ตอนุกรม SCON	31
รูปที่ 2.21 สัญญาณการรับและส่งข้อมูลในโหมด 0	33
รูปที่ 2.22 การรับและส่งข้อมูลในโหมด 1	33
รูปที่ 2.23 การรับและส่งข้อมูลในโหมด 2 และ 3	34
รูปที่ 2.24 รีจิสเตอร์ PCON	34
รูปที่ 2.25 ส่วนประกอบต่าง ๆ ของ Servo Motor	36
รูปที่ 2.26 สัญญาณความกว้างพัลส์	37

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## สารบัญญภาพ (ต่อ)

	หน้า
รูปที่ 2.27 การป้อนสัญญาณความกว้างพัลส์ให้กับมอเตอร์	37
รูปที่ 3.1 แสดงส่วนต่างๆของหุ่นยนต์	39
รูปที่ 3.2 วงจรควบคุมการหมุนของดีซีมอเตอร์	40
รูปที่ 3.3 วงจรการใช้งานเซอร์โวมอเตอร์	42
รูปที่ 3.4 แสดงวงจรเปรียบเทียบแรงดันที่ใช้	44
รูปที่ 3.5 แสดงการวางตำแหน่งของเซนเซอร์พื้น	45
รูปที่ 3.6 วงจรเซนเซอร์อินฟราเรด	46
รูปที่ 3.7 ตำแหน่งเซนเซอร์อินฟราเรด	47
รูปที่ 3.8 ข้อมูลอินพุท/เอาต์พุทที่ไมโครคอนโทรลเลอร์ของตัวหุ่นยนต์	48
รูปที่ 3.9 แสดงโครงสร้างของส่วนรับส่งข้อมูลระหว่างเซิร์ฟเวอร์ กับหุ่นยนต์	48
รูปที่ 3.10 โฟลว์ชาร์ตการทำงานขั้นพื้นฐานของหุ่นยนต์	50
รูปที่ 3.11 โฟลว์ชาร์ตการทำงานแบบอัตโนมัติของหุ่นยนต์	51
รูปที่ 3.12 โฟลว์ชาร์ตการส่งข้อมูลของภาคส่งข้อมูลควบคุมรถ	52
รูปที่ 3.13 โฟลว์ชาร์ตการรับส่งข้อมูลของหุ่นยนต์	53
รูปที่ 3.14 ภาพแสดงโครงสร้างของโปรแกรมควบคุมที่เซิร์ฟเวอร์	54
รูปที่ 3.15 โฟลว์ชาร์ตแสดงการทำงานของโปรแกรมส่วนหลัก	55
รูปที่ 3.16 โฟลว์ชาร์ตแสดงการทำงานในแต่ละคำสั่งของโปรแกรมหลัก	56
รูปที่ 3.17 แสดงการนำภาพจากกล้องมาใช้	57
รูปที่ 3.18 แสดงโครงสร้างในการติดต่อระหว่างเซิร์ฟเวอร์ กับโทรศัพท์เคลื่อนที่	59
รูปที่ 3.19 แสดงการรับส่งข้อมูลระหว่างไคลเอ็นต์กับเซิร์ฟเวอร์	60
รูปที่ 4.1 การทดสอบการรับส่งข้อมูลภาพที่ระยะต่าง ๆ	62
รูปที่ 4.2 การทดลองเซนเซอร์อุณหภูมิในสภาวะอุณหภูมิห้อง	63
รูปที่ 4.3 การทดลองเซนเซอร์อุณหภูมิในสภาวะที่มีความร้อนผิดปกติ	63
รูปที่ 4.4 การทดลองเซนเซอร์อุณหภูมิในสภาวะที่มีความเย็นผิดปกติ	64
รูปที่ 4.5 หน้าจอการตรวจจับความเคลื่อนไหว	64
รูปที่ 4.6 การส่ง MMS แจ้งเตือนความเคลื่อนไหว	65
รูปที่ 4.7 อินพุทและและผลลัพธ์จากโปรแกรมพีลเตอร์	66

เมื่อแถบบาร์โค้ดอยู่ห่างจากกล้องเป็นแนวตรงที่ ระยะ 4 นิ้ว  
 เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
 ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## สารบัญภาพ (ต่อ)

	หน้า
รูปที่ 4.8 อินพุทและและผลลัพธ์จากโปรแกรมฟิเตอร์	
เมื่อแถบบาร์โค้ดอยู่ห่างจากกล้องเป็นแนวตรงที่ ระยะ 9 นิ้ว	66
รูปที่ 4.9 อินพุทและและผลลัพธ์จากโปรแกรมฟิเตอร์แถบบาร์โค้ดที่ 2	
เมื่อแถบบาร์โค้ดอยู่ห่าง จากกล้องเป็นแนวตรงที่ ระยะ 4 นิ้ว	67
รูปที่ 4.10 อินพุทและและผลลัพธ์จากโปรแกรมฟิเตอร์แถบบาร์โค้ดที่ 2	
เมื่อแถบบาร์โค้ดอยู่ห่างจากกล้องเป็นแนวตรงที่ ระยะ 9 นิ้ว	67
รูปที่ 4.11 หน้าจอแสดงการ log in ผ่านทางโทรศัพท์เคลื่อนที่	68
รูปที่ 4.12 หน้าจอแสดงฟังก์ชันการใช้งานระบบผ่านทางโทรศัพท์เคลื่อนที่	68
รูปที่ 4.13 หน้าจอการควบคุมการเคลื่อนที่ของหุ่นยนต์ผ่านทางโทรศัพท์เคลื่อนที่	69
รูปที่ 4.14 หน้าจอสำหรับการสั่งงานเกี่ยวกับภาพที่ติดอยู่ที่ตัวหุ่นยนต์	69
รูปที่ 4.15 หน้าจอแสดงไฟล์ภาพที่ได้รับจากเซิร์ฟเวอร์	70
รูปที่ 4.16 หน้าจอแสดงข้อความออกสู่ระบบ	70
รูปที่ 4.17 หน้าจอเข้าใช้งานระบบผ่านอินเทอร์เน็ต	71
รูปที่ 4.18 หน้าจอแสดงข้อความแจ้งเตือนหากกรอกชื่อผู้ใช้หรือรหัสผ่านผิด	71
รูปที่ 4.19 หน้าจอแสดงข้อความแจ้งเตือนเมื่อมีผู้ใช้งานระบบอยู่	72
รูปที่ 4.20 หน้าจอการเชื่อมต่อ เพื่อเข้าใช้งานระบบผ่านอินเทอร์เน็ต	72
รูปที่ 4.21 หน้าจอแสดงฟังก์ชันการใช้งานระบบผ่านอินเทอร์เน็ต	73
รูปที่ 4.22 หน้าจอแสดงฟังก์ชันการใช้งานที่เซิร์ฟเวอร์	74
รูปที่ 4.23 หน้าจอแสดงฟังก์ชันการใช้งานการควบคุมหุ่นยนต์ที่เซิร์ฟเวอร์	75
รูปที่ 4.24 หน้าจอการเลือกคู่มือวิดีโอที่ได้ทำการบันทึกไว้	76
รูปที่ 4.25 หน้าจอแสดงการตั้งค่าพารามิเตอร์ต่างๆจากโทรศัพท์เคลื่อนที่	77

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## บทที่ 1

### บทนำ

#### 1.1 ความสำคัญและที่มา

ในปัจจุบันเทคโนโลยีหุ่นยนต์เป็นเทคโนโลยีที่ได้รับความสนใจและมีแนวโน้มที่จะเข้ามาเป็นส่วนหนึ่งในชีวิตของเรามากยิ่งขึ้นในอนาคต ประกอบกับเทคโนโลยีการสื่อสารที่ได้มีการพัฒนาเป็นอย่างมากในด้านประสิทธิภาพ ความรวดเร็วและความสะดวกในการใช้งานทั้งเครือข่ายแบบมีสายและไร้สาย

ผู้จัดทำได้เห็นถึงความสำคัญของทั้งสองเทคโนโลยีนี้จึงมีแนวคิดในการนำเอาทั้งสองเทคโนโลยีมาประยุกต์ใช้ร่วมกันเพื่อเป็นการศึกษาและใช้ประโยชน์จากเทคโนโลยีได้อย่างคุ้มค่าและมีประสิทธิภาพมากยิ่งขึ้น โดยเฉพาะทุกวันนี้สมาชิกในครอบครัวมักจะออกไปทำงานหรือไปเรียนนอกบ้านทำให้ไม่มีคนอยู่ดูแลบ้านทำให้เกิดแนวคิดในการจัดทำโครงงานหุ่นยนต์รักษาความปลอดภัยภายในบ้านที่สามารถควบคุมผ่านเครือข่ายขึ้นมา

#### 1.2 วัตถุประสงค์ของโครงการ

- เพื่อศึกษาโครงสร้างและเขียน โปรแกรมเพื่อรับส่งข้อมูลผ่านเครือข่าย โทรศัพท์เคลื่อนที่
- เพื่อศึกษาโครงสร้างและการเขียน โปรแกรมเพื่อรับส่งข้อมูลผ่านเครือข่ายอินเทอร์เน็ต
- เพื่อศึกษาการสร้างและการเขียน โปรแกรมเพื่อควบคุมหุ่นยนต์
- เพื่อศึกษาการทำงานและประยุกต์ใช้อุปกรณ์เซนเซอร์
- เพื่อประยุกต์ใช้โครงงานในการดูแลและตรวจสอบสภาพภายในบ้านผ่าน โทรศัพท์เคลื่อนที่และเครือข่ายอินเทอร์เน็ต

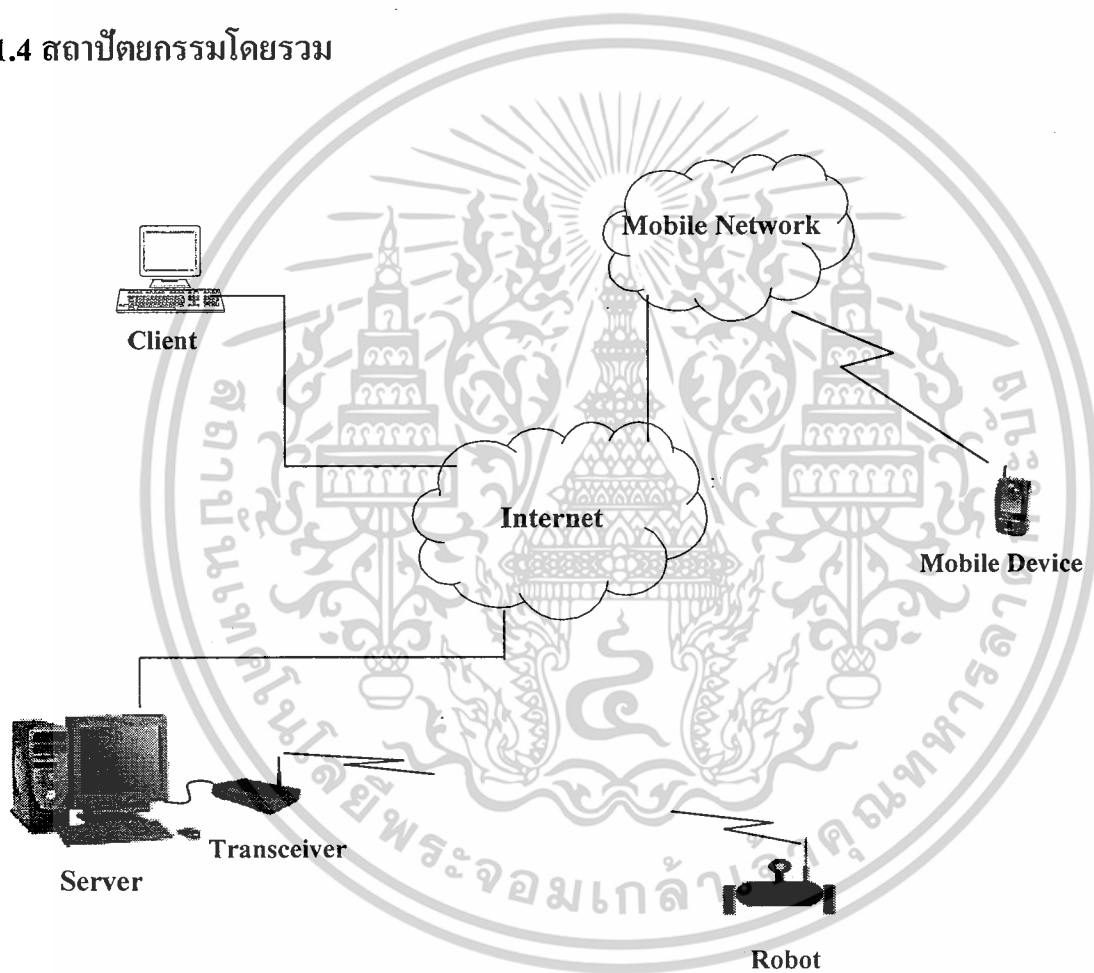
#### 1.3 ขอบเขตโครงการ

- สามารถควบคุมหุ่นยนต์ให้เคลื่อนที่ได้ในแบบอัตโนมัติและแบบบังคับด้วยมือ (manual)
- ในการเคลื่อนที่แบบอัตโนมัติ หุ่นยนต์จะเคลื่อนที่ไปชิดไปตามกำแพงและจะมีจุดบอกตำแหน่งซึ่งจะใช้สัญลักษณ์บาร์โค้ด (barcode) ติดอยู่ที่ผนังห้อง
- สามารถควบคุมหุ่นยนต์ได้จากโทรศัพท์เคลื่อนที่และควบคุมผ่านเครือข่ายอินเทอร์เน็ต

เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์ การใช้งานเพื่อการค้าโดยไม่ได้รับอนุญาตถือว่าผิดกฎหมาย การนำเอกสารนี้ไปใช้ในการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- การควบคุมผ่านเครือข่ายอินเทอร์เน็ตสามารถดูภาพเคลื่อนไหวจากกล้องและควบคุมแบบ บังคับด้วยมือได้
- การสั่งงานหุ่นยนต์จะผ่านคอมพิวเตอร์เซิร์ฟเวอร์โดยใช้คลื่นวิทยุ
- สามารถตรวจจับสิ่งผิดปกติที่เกิดขึ้นได้ เช่น ความร้อนหรือการเคลื่อนไหวของวัตถุและ สามารถแจ้งเตือนไปยังโทรศัพท์เคลื่อนที่ได้

#### 1.4 สถาปัตยกรรมโดยรวม



รูปที่ 1.1 ภาพรวมของระบบ

การทำงานในภาพรวมของระบบจะมีคอมพิวเตอร์เซิร์ฟเวอร์เป็นศูนย์กลางในการเชื่อมต่อ เพื่อควบคุมหุ่นยนต์ โดยที่เซิร์ฟเวอร์จะทำการประมวลผลข้อมูลคำสั่งและทำการส่งข้อมูลเพื่อ สั่งงานหุ่นยนต์โดยผ่านอุปกรณ์รับส่งข้อมูล (Transceiver) ซึ่งจะทำการส่งข้อมูลไปยังหุ่นยนต์ ผ่านคลื่นวิทยุ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ในการควบคุมการเคลื่อนที่ของหุ่นยนต์จะมีอยู่ 2 โหมดคือ โหมดอัตโนมัติและโหมดบังคับด้วยมือ

### โหมดอัตโนมัติ

ในโหมดนี้สามารถสั่งงานหุ่นยนต์ได้ดังนี้

- เคลื่อนที่ไปรอบๆบ้านแล้วทำการเก็บภาพนิ่งในช่วงเวลาหรือเก็บภาพวิดีโอ
- เคลื่อนที่ไปที่จุดที่ต้องการแล้วทำการตรวจจับการเคลื่อนไหวหรือเก็บภาพนิ่งหรือหยุดนิ่ง (จุดที่ต้องการคือจุดที่ได้ติดแถบสีดำไปที่พื้นและเครื่องหมายบาร์โค้ดไว้ที่ผนังห้อง)

### โหมดบังคับด้วยมือ

ในโหมดนี้จะสามารถสั่งงานได้ดังนี้

- ให้อุ่นยนต์เคลื่อนที่ (เดินหน้า, ถอยหลัง, หมุนซ้าย, หมุนขวา) ตามระยะที่กำหนด
- หมุนกล้องที่ติดอยู่ที่หุ่นยนต์ได้

ในการเชื่อมต่อเข้ามาควบคุมหุ่นยนต์สามารถทำได้ 3 วิธี โดยที่ในเวลาเดียวกันจะสามารถมีผู้ควบคุมหุ่นยนต์ได้ผู้เดียวเท่านั้น ไม่สามารถควบคุมพร้อมกันจากหลายทางหรือจากหลายผู้ใช้ได้ ดังนั้นเมื่อมีผู้ใดเข้ามาควบคุมระบบก็จะทำการยึดครองระบบไม่ให้ผู้อื่นสามารถเข้ามาใช้งานได้ จนกว่าจะทำการออกจากระบบ โดยแต่ละวิธีมีดังนี้

#### ควบคุมจากเซิร์ฟเวอร์โดยตรง

วิธีนี้จะเป็นการควบคุมหุ่นยนต์โดยตรงจากเซิร์ฟเวอร์

#### ควบคุมผ่านอุปกรณ์โทรศัพท์เคลื่อนที่

วิธีนี้จะเป็นการควบคุมจากระยะไกลผ่านอุปกรณ์โทรศัพท์เคลื่อนที่ โดยในการเชื่อมต่อจะอาศัยเทคโนโลยีเว็บ (WAP) ในการรับส่งข้อมูลผ่านเครือข่ายโทรศัพท์เคลื่อนที่ซึ่งเชื่อมต่อกับเครือข่ายอินเทอร์เน็ต โดยในการควบคุมหุ่นยนต์จากโทรศัพท์เคลื่อนที่จะสามารถทำได้ดังนี้

- เก็บภาพนิ่งจากกล้อง ณ เวลานั้นและดูภาพนั้นได้ทันที
- เริ่มหรือหยุดการเก็บภาพวิดีโอได้
- ดูภาพนิ่งที่เก็บไว้ก่อนหน้าได้
- สั่งให้หุ่นยนต์เคลื่อนที่ในโหมดอัตโนมัติได้ ถ้าหุ่นยนต์อยู่ในสถานะหยุดนิ่ง
- โทรศัพท์เคลื่อนที่สามารถรับข้อมูลการแจ้งเตือนเมื่อมีการเคลื่อนไหวเกิดขึ้นเมื่อมีการสั่งให้มีการตรวจจับการเคลื่อนไหว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### ควบคุมผ่านเครือข่ายอินเทอร์เน็ต

วิธีนี้จะเป็นการควบคุมจากเครื่องคอมพิวเตอร์ใดๆก็ได้ที่อยู่ในเครือข่ายอินเทอร์เน็ต โดยจะมีความสามารถดังนี้

- ควบคุมเคลื่อนไหวแบบเรียลไทม์จากกล้องที่ติดตั้งอยู่ที่หุ่นยนต์
- ควบคุมการเคลื่อนที่ของหุ่นยนต์และกล้องใน โหมคบังคับด้วยมือ

### 1.5 ขั้นตอนการดำเนินงาน

แบ่งออกเป็น 2 ส่วน

- ส่วนติดต่อระหว่างคอมพิวเตอร์กับหุ่นยนต์
  - ศึกษาการทำงานและการเขียน โปรแกรมเพื่อควบคุมไมโครคอนโทรลเลอร์ตระกูล 8051
  - ศึกษาการทำงานของดีซีมอเตอร์ (DC motor) และเซอร์โวมอเตอร์ (servo motor)
  - ศึกษาการรับส่งข้อมูลผ่านคลื่นความถี่วิทยุ
  - ศึกษาการทำงานของเซนเซอร์ (เช่น เซอร์อูธโทมิ , เซนเซอร์อินฟราเรด, เซนเซอร์แสง)
  - นำข้อมูลที่ได้ศึกษามาสร้างตัวหุ่นยนต์และเขียน โปรแกรมควบคุมและสั่งงานผ่านคลื่นวิทยุ
- ส่วนของ โปรแกรมควบคุมจากโทรศัพท์เคลื่อนที่และผ่านเครือข่ายอินเทอร์เน็ต
  - ศึกษาการติดต่อระหว่างคอมพิวเตอร์กับ โทรศัพท์เคลื่อนที่
  - ศึกษาการเชื่อมต่อระหว่างคอมพิวเตอร์ผ่านเครือข่ายอินเทอร์เน็ต
  - ศึกษาการรับ แสดงและประมวลผลภาพจากกล้อง
  - ทดลองเขียน โปรแกรมรับส่งข้อมูลระหว่างคอมพิวเตอร์และ โทรศัพท์เคลื่อนที่
  - ทดลองเขียน โปรแกรมรับส่งข้อมูลระหว่างคอมพิวเตอร์ผ่านเครือข่ายอินเทอร์เน็ต
  - ทดลองเขียน โปรแกรมรับ แสดงและการประมวลผลภาพจากกล้อง
  - นำข้อมูลทั้งหมดมาเขียนโปรแกรมเพื่อควบคุมหุ่นยนต์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## บทที่ 2

### ทฤษฎีและหลักการที่เกี่ยวข้อง

#### 2.1 แวร์พ (WAP)

แวร์พ (WAP: Wireless Application Protocol) เป็นตัวกลางสำคัญในการเชื่อมต่อโลกของอุปกรณ์สื่อสารไร้สายเข้ากับเครือข่ายอินเทอร์เน็ต ซึ่งในบางกรณีก็สามารถประยุกต์ใช้กับเครือข่ายอินเทอร์เน็ตภายในองค์กรจุดประสงค์เพื่อขยายขอบเขตในการให้บริการเสริมให้กับบริการเครือข่ายไร้สายโดยไม่คำนึงถึงประเภทของเครือข่ายบริการและเครื่องลูกข่ายแต่อย่างใด ผู้ใช้บริการสามารถดึงข้อมูลจากเครือข่ายอินเทอร์เน็ตได้จากอุปกรณ์สื่อสารไร้สายของตน โดยตัวเครื่องอาจมีขนาดเท่าฝ่ามือแต่มีขีดความสามารถในการทำงานเทียบเท่ากับเครื่องคอมพิวเตอร์ส่วนบุคคล

แวร์พเป็นเทคโนโลยีเกิดขึ้นมาด้วยความพยายามที่จะทำให้เกิดมาตรฐานสำหรับเทคโนโลยีไร้สายโดยในปี 1994 Openwave หรือบริษัท อันไวร์ แพลนเน็ต ได้เสนอเทคโนโลยีที่จะนำอินเทอร์เน็ตไปสู่อุปกรณ์สื่อสารไร้สาย ภาษาที่ใช้เป็นภาษาสำหรับอุปกรณ์ไร้สายเรียกว่า HDML (Handheld Device Mark-up Language) ซึ่งมีพื้นฐานมาจากภาษา HTML โดย HDML ประกอบด้วยรูปแบบเพิ่มเติมในการส่วนติดต่อกับผู้ใช้ (soft keys, ปุ่มตัวเลข) และการใช้ทรัพยากรของระบบ (ตัวแปลทางด้านเครื่องลูกข่าย, โครงสร้างแบบ deck/card) เพื่อให้เหมาะสมกับอุปกรณ์สื่อสารไร้สายที่มีข้อจำกัดในการประมวลผล การแสดงผลและแบนวิดธ์ของเครือข่ายสื่อสาร

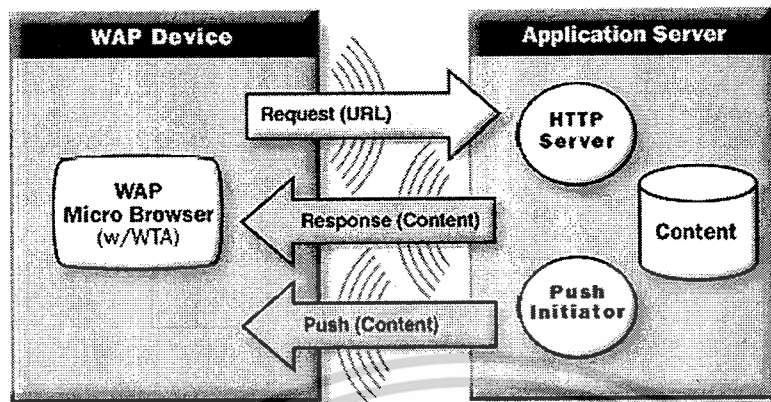
ต่อมา Openwave และบริษัทที่เกี่ยวข้องในเทคโนโลยีไร้สาย (Nokia, Ericsson และ Motorola) พร้อมทั้งผู้ให้บริการเครือข่ายไร้สายรายใหญ่โดยเฉพาะ AT&T ได้มีแนวความคิดในการมีมาตรฐานสากลของเทคโนโลยี ซึ่งนำไปสู่การเกิดขึ้นของ WAP และ WAP Forum และทำให้เกิดภาษา WML (Wireless Markup Language) ซึ่งเป็นภาษามาตรฐานที่ใช้ใน WAP เวอร์ชันแรก

ในปี 2001 แวร์พได้พัฒนาไปเป็นแวร์พเวอร์ชัน 2 (WAP 2.0) ซึ่งในเวอร์ชันนี้จะทำให้มาตรฐานของแวร์พเข้าใกล้กับมาตรฐานของอินเทอร์เน็ตที่มีอยู่มากยิ่งขึ้นและยังคงเข้ากันได้กับ WAP 1.0

##### 2.1.1 WAP Programming Model

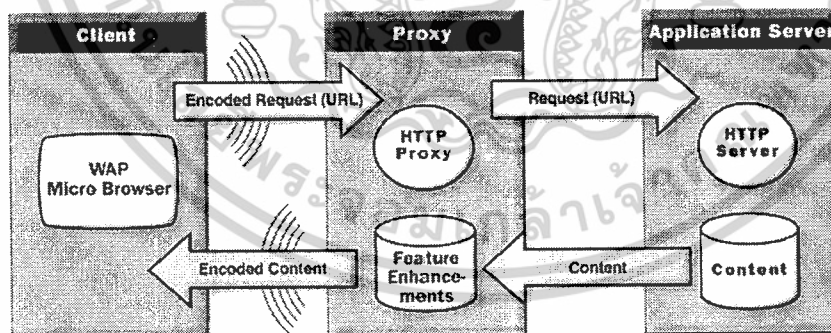
WAP Programming Model มีความใกล้เคียงกับ Web Programming Model คือใช้ Pull Model (เครื่องลูกข่ายทำการร้องขอข้อมูลจากเครื่องแม่ข่าย) ซึ่งจะต่างกันตรงที่แวร์พจะมีโครงสร้างที่เพิ่มเติมจากเว็บคือส่วนของการสื่อสารและการรองรับ Push Model ซึ่งเครื่องแม่ข่ายสามารถเป็นผู้เริ่มต้นส่งข้อมูลไปยังเครื่องลูกข่ายได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.1 WAP Programming Model

ใน WAP 1.0 จะใช้ WAP proxy หรือ WAP gateway ในการเป็นตัวกลางเชื่อมต่อระหว่างเครื่องลูกข่ายกับเครื่องแม่ข่าย WAP proxy จะใช้เว็บโปรโตคอลในการสื่อสารกับเครื่องลูกข่ายและใช้อินเตอร์เน็ตโปรโตคอลในการเชื่อมต่อกับเครื่องแม่ข่าย WAP2.0 ไม่ต้องการ WAP proxy เพราะว่าการสื่อสารระหว่างเครื่องลูกข่ายและเครื่องแม่ข่ายสามารถใช้โปรโตคอล HTTP/1.1 อย่งไรก็ตามการใช้ WAP proxy จะเป็นการเพิ่มความสามารถในกระบวนการสื่อสารและการให้บริการอย่างเช่น การระบุพื้นที่บริการ, นโยบายความเป็นส่วนตัว, การยืนยันตัวตนและมีความจำเป็นในการใช้งานแบบ Push Model



รูปที่ 2.2 การใช้ WAP Proxy เพื่อเพิ่มประสิทธิภาพการใช้งานเครื่องข่าย

### 2.1.2 Wireless Application Environment (WAE)

สภาวะแวดล้อมระดับแอปพลิเคชันของเว็บ (WAE) ให้บริการสำหรับการเชื่อมต่อระหว่างแอปพลิเคชันของเว็บกับเว็บของอุปกรณ์ที่มีเว็บไมโครเบราเซอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ไมโครเบราเซอร์ใน WAP2.0 จะสนับสนุนภาษา XHTML Mobile Profile markup language (XHTMLMP) สำหรับข้อมูลรูปแบบใหม่และ WML เพื่อสนับสนุนเนื้อหาของ WAP1.0 ภาษาเหล่านี้มีหน้าที่ในการแสดงผลสำหรับอุปกรณ์สื่อสารไร้สาย และภาษาสคริปต์สำหรับเครื่องลูกข่ายคือ WMLScript เป็นการเพิ่มความสามารถให้มากขึ้น

WAE ยังสนับสนุนรูปแบบเนื้อหาอื่นๆอีก เช่น ภาพ WBMP vCard และ vCalendar

ลักษณะของ WAE ในเว็บเวอร์ชัน 2 มีดังนี้

- ภาษาพื้นฐานที่ใช้คือ XHTMLMP ซึ่งเพิ่มเติมมาจาก XHTML ซึ่งกำหนดโดย W3C โดยเพิ่มเติมบางส่วนเพื่อเพิ่มประสิทธิภาพในการใช้งาน เนื้อหาที่เขียนด้วยภาษา XHTML สามารถเข้ากันได้กับ XHTMLMP เบราเซอร์ได้อย่างสมบูรณ์
- สนับสนุนการทำงานของสไตลชีทซึ่งมีพื้นฐานมาจาก CSS จาก W3C ทั้งแบบอยู่ภายในเอกสารและภายนอกเช่นเดียวกับอินเทอร์เน็ตเบราว์เซอร์
- เข้ากันได้กับ WML1 โดย WAE ในเว็บเวอร์ชัน 2 จะสนับสนุนทั้ง WML1 และ XHTMLMP ตั้งแต่ต้นหรือแปลงจาก WML1 ไปเป็น WML2 ซึ่ง WML2 เป็นส่วนเพิ่มเติมจาก XHTMLMP เพื่อความเข้ากันได้กับ WML1

### 2.1.3 WAP Protocol Stack

ส่วนสำคัญของ WAP 2.0 ก็คือการสนับสนุนอินเทอร์เน็ตโพรโทคอลแม่อุปกรณ์สื่อสารไร้สายใช้ไอพีในการเชื่อมต่อแต่ยังคงต้องมีการสนับสนุน WAP Stack แบบเดิมเพื่อใช้กับเครือข่ายสื่อสารที่ไม่สนับสนุนไอพีซึ่งทั้งสองรูปแบบมีอยู่ใน WAP2.0 และให้บริการในระดับแอปพลิเคชันที่ไม่มีความแตกต่างกัน

### 2.1.4 ระดับชั้นของโพรโทคอลแบบเก่า

WAP 2.0 จะยังคงสนับสนุน WAP 1 Stack ซึ่งเป็นการเพิ่มประสิทธิภาพสำหรับการเชื่อมต่อของเครือข่ายแบบวิเศษต์ต่ำ โดยในแต่ละระดับชั้นมีดังนี้

#### - Wireless Session Protocol (WSP)

WSP ประกอบด้วยอินเทอร์เน็ตเฟสสำหรับ 2 ส่วนของการบริการ ส่วนแรกคือบริการการเชื่อมต่อแบบ Connection-Oriented ที่ทำงานอยู่บนระดับชั้น Transaction Protocol (WTP) ส่วนที่ 2 คือ บริการการเชื่อมต่อแบบ Connectionless ที่ทำงานอยู่บนบริการแบบดาต้าแกรม (WDP)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

WSP ประกอบด้วยบริการที่ร้องขอจากบราวเซอร์แอปพลิเคชัน (WSP/B) ซึ่ง (WSP/B) จะประกอบด้วยฟังก์ชันต่อไปนี้

- ฟังก์ชันในการเข้ารหัสและแปลความหมายของ HTTP/1.1
- Long-lived session state
- การพัก session และกลับมาทำงาน (Session suspend and resume with session migration)
- ความสะดวกในการใส่ข้อมูล
- การทำงานข้ามโพรโตคอล

มาตรฐานใน WSP จะถูกลดขนาดลงเพื่อให้เหมาะสมกับการใช้งานบนเครือข่ายที่มีย่านความถี่แคบ และระยะเวลาในการส่งข้อมูลนาน WSP/B ถูกออกแบบให้ WAP proxy ติดต่อ WSP/B ของไคลเอนต์ กับ เซิร์ฟเวอร์ที่เป็น HTTP

#### - Wireless Transaction Protocol (WTP)

WTP เกี่ยวข้องกับการรับประกันความน่าเชื่อถือของการส่งข้อมูล WTP ทำงานอยู่บนชั้นบนสุดของการให้บริการแบบดาต้าแกรมและใช้ light-weight transaction-oriented protocol ที่เหมาะสมกับอุปกรณ์ไร้สาย WTP ทำงานได้อย่างมีประสิทธิภาพอยู่บนเครือข่ายแบบดาต้าแกรมแบบไร้สายทั้งที่มีระบบความปลอดภัยและไม่มี ซึ่งมีลักษณะสำคัญดังนี้

- มีคลาสในการให้บริการ 3 คลาส
  - Unreliable one-way requests
  - Reliable one-way requests
  - Reliable two-way request-reply transactions
- ผู้ใช้สามารถเลือกระดับความน่าเชื่อถือได้
- ผู้ใช้สามารถเลือกระดับข้อมูลผิดพลาดที่ยอมรับได้
- มีการหน่วงเวลาของ ACK เพื่อลดจำนวนข้อมูลที่ส่ง
- Asynchronous transaction

#### - Wireless Transaction Layer Security (WTLS)

จุดประสงค์ของ WTLS คือการสร้างความปลอดภัยในระดับชั้นการส่งข้อมูลระหว่างไคลเอนต์กับแอสเซมบลีซึ่ง WTLS อยู่บนพื้นฐานความปลอดภัยในชั้นการขนส่ง (TLS: Transport Layer Security) แต่ถูกลดขนาดลงเพื่อเหมาะสมกับช่องสัญญาณที่แคบ WTLS มีลักษณะสำคัญดังนี้

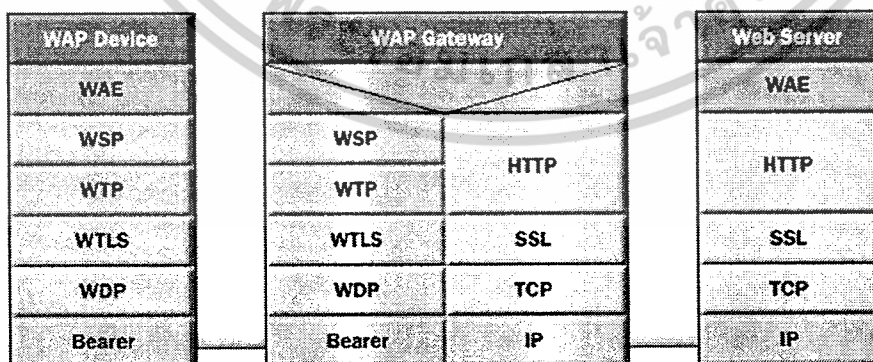
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- ความถูกต้องของข้อมูล (Data integrity) – รับประกันว่าข้อมูลที่ส่งระหว่างไคลเอนต์และเซิร์ฟเวอร์จะไม่ถูกเปลี่ยนแปลง
- ความเป็นส่วนตัว (Privacy) – รับประกันว่าข้อมูลที่จะไม่สามารถถูกอ่านโดยคนอื่นได้นอกจากเจ้าของเท่านั้น
- การพิสูจน์ตน (Authentication) – มีการพิสูจน์ตนระหว่างไคลเอนต์และเซิร์ฟเวอร์
- การป้องกัน Denial-of-service – มีการป้องกันการโจมตีแบบ Denial-of-service แอปพลิเคชันสามารถเลือกที่จะเอา WTLS หรือไม่เอาได้ขึ้นอยู่กับระดับความปลอดภัยที่ต้องการและลักษณะของเครือข่าย

#### - Wireless Datagram Protocol (WDP)

ระดับชั้นในการส่งข้อมูลของเว็บจะใช้เทคโนโลยีแบบค้ำแกรมซึ่งทำงานอยู่บนการให้บริการส่งข้อมูลโดยตัวกลาง (Bearer) ที่รองรับเครือข่ายแบบต่างๆ ถ้าเว็บถูกใช้บนตัวกลางที่สนับสนุน UDP (User Datagram Protocol) แล้ว WDP ก็ไม่จำเป็นต้องใช้ แต่บนตัวกลางอื่นๆ เช่น GSM, SMS นั้นฟังก์ชันค้ำแกรมถูกจัดเตรียมโดย WDP ดังนั้นเว็บจึงใช้ UDP หรือไม่ก็ WDP ในการให้บริการค้ำแกรมที่มีการซ่อนลักษณะของตัวกลางที่แตกต่างกันไว้และเตรียมฟังก์ชัน, port number และถ้าจำเป็น WDP ก็สามารถขยายฟังก์ชันสำหรับการแตก (fragment) และการรวม (reassemble) ของค้ำแกรมที่ใหญ่เกินไปสำหรับตัวกลางนั้นๆ

ตัวอย่างการทำงานระหว่างอุปกรณ์สื่อสารกับ WAP Gateway โดย WAP Gateway ทำหน้าที่ในการเชื่อมต่อบริการในชั้นโปรโตคอล WSP ไปยังโปรโตคอล HTTP เพื่อเข้าถึงข้อมูลในเครือข่ายอินเทอร์เน็ต



รูปที่ 2.3 ตัวอย่าง WAP Gateway แบบเก่า

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 2.1.5 ระดับชั้นของโปรโตคอลแบบใหม่

ส่วนเสริมที่สำคัญของ WAP2.0 ก็คืออินเทอร์เน็ตโปรโตคอล (IP) ในสภาวะแวดล้อมของเว็บ ซึ่งได้รับอิทธิพลมาจากการพัฒนาของเครือข่ายไร้สายความเร็วสูง (2.5G และ 3G) ซึ่งอุปกรณ์สื่อสารไร้สายสนับสนุนอินเทอร์เน็ตโปรโตคอลโดยตรง ซึ่งประกอบไปด้วยชั้นดังนี้

#### - Wireless Profiled HTTP (WP-HTTP)

WP-HTTP เป็นโครงร่างเฉพาะของ HTTP โปรโตคอลสำหรับการสื่อสารไร้สายและสามารถทำงานร่วมกับ HTTP/1.1 ได้อย่างสมบูรณ์ การทำงานพื้นฐานระหว่างอุปกรณ์เว็บกับ WAP Proxy/WAP Server คือการติดต่อในลักษณะร้องขอและตอบสนอง (Request/Response) WP-HTTP สนับสนุนการบีบอัดและการสร้างช่องทางที่ปลอดภัยพิเศษในการสื่อสารข้อมูล

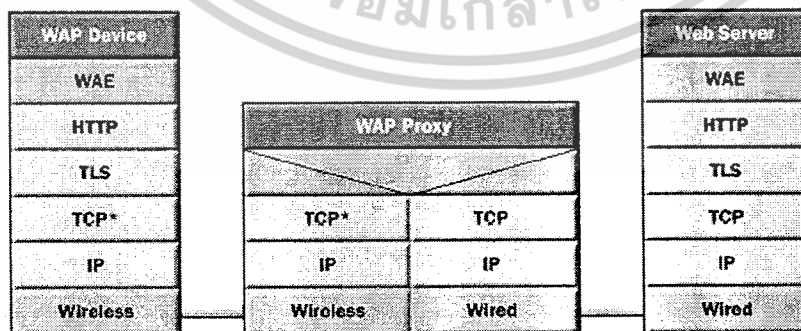
#### - Transport Layer Security (TLS)

โปรโตคอล TLS จะอนุญาตให้สามารถทำการติดต่อที่ปลอดภัย โดยจะรวมเอาชุดของการเข้ารหัส, รูปแบบของการรับรอง, วิธีการในการแสดงตัวตนและการใช้การยืนยันถิ่นสภาพการติดต่อ และยังมีกำหนดวิธีการทำช่องทางเชื่อมต่อพิเศษเพื่อการเชื่อมต่อระหว่างต้นทางและปลายทางที่ปลอดภัยในชั้นการนำส่งข้อมูล

#### - Wireless Profiled TCP (WP-TCP)

WP-TCP ให้บริการการเชื่อมต่อแบบเชื่อมต่อได้ (Connection-oriented) ซึ่งเป็นการเพิ่มประสิทธิภาพในการเชื่อมต่อแบบไร้สายและสามารถทำงานร่วมกับโปรโตคอล TCP มาตรฐานในอินเทอร์เน็ตได้อย่างสมบูรณ์

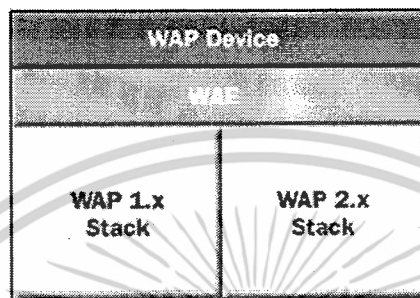
ตัวอย่างของการใช้โปรโตคอลเหล่านี้ในการเชื่อมต่อระหว่างอุปกรณ์ที่สนับสนุนเว็บกับเว็บเซิร์ฟเวอร์ การสร้างช่องทางเชื่อมต่อพิเศษ (Tunnel) ในชั้นโปรโตคอล TLS จะอาศัย WAP Proxy และการติดต่อจากต้นทางถึงปลายทางของโปรโตคอล HTTP TCP\* หมายถึง WP-TCP



รูปที่ 2.4 ตัวอย่าง WAP Gateway แบบใหม่

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เนื่องจากเว็บเวอร์ชัน 2 สนับสนุนทั้งสองรูปแบบของระดับชั้นโปรโตคอลซึ่งในการทำงานทั้งสองแบบจะทำงานเป็นอิสระต่อกัน นั่นคือจะไม่มีการทำงานร่วมกันหรือเชื่อมต่อกันของทั้งสองรูปแบบในการสื่อสาร



รูปที่ 2.5 WAP Stack แบบคู่

จากรูปแสดงให้เห็นว่า ในสถานะแวดล้อมในระดับแอปพลิเคชันสามารถที่จะทำงานได้ทั้งสองรูปแบบของระดับชั้นโปรโตคอลสำหรับอุปกรณ์ซึ่งสนับสนุนการทำงานของทั้งเครือข่ายแบบเก่าและแบบใหม่ การสลับการทำงานของระดับชั้นอาจเกิดขึ้นเมื่ออุปกรณ์เคลื่อนที่เข้าหรือออกพื้นที่ครอบคลุมของเครือข่ายที่ต่างกัน

## 2.2 Java Technology

เทคโนโลยีจาวาสามารถแบ่งออกเป็น 3 กลุ่ม เพื่อใช้งานกับอุปกรณ์ที่มีคุณสมบัติที่แตกต่างกันดังนี้คือ

1. Standard Edition (J2SE): เป็นเทคโนโลยีจาวาที่ออกแบบเพื่อนำมาใช้พัฒนางานบนเครื่องคอมพิวเตอร์เดสก์ท็อปและคอมพิวเตอร์เวิร์กสเตชัน
2. Enterprise Edition (J2EE): ถูกออกแบบมาเพื่อให้สามารถใช้งานบนระบบเครื่องเซิร์ฟเวอร์รองรับการทำงานร่วมกับ Servlets, JSP และ XML
3. Micro Edition (J2ME): มีเป้าหมายสำหรับพัฒนาแอปพลิเคชันที่ทำงานบนอุปกรณ์เคลื่อนที่ที่มีทรัพยากรจำกัดไม่ว่าจะเป็นขนาดของหน่วยความจำ ความสามารถในการประมวลผล เช่น โทรศัพท์มือถือ เพจเจอร์ ปาล์ม PDA เป็นต้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 2.2.1 เซิร์ฟเล็ต (Servlet)

แม้ว่าโลกของอินเทอร์เน็ตจะเพิ่งเกิดขึ้นเพียงไม่กี่ปีก็ตาม แต่เทคโนโลยีที่ใช้กับอินเทอร์เน็ตกลับมีการเปลี่ยนแปลงไปอย่างรวดเร็วมาก ในสมัยแรกๆ หน้าเพจต่างๆ ที่ในเว็บจะเป็นลักษณะของสแตติกเพจหรือเพจที่ไม่มีการเปลี่ยนแปลงเนื้อหาที่ไม่ว่าจะนานเท่าไร นอกเสียจากว่าผู้ดูแลเพจนั้นจะทำการอัปเดตเพจดังกล่าว เพจลักษณะนี้เป็นเพจที่นิยมใช้กันทั่วไปในอินเทอร์เน็ตสมัยแรกเพราะอินเทอร์เน็ตยังนิยมกันอยู่ในวงแคบ โดยกลุ่มผู้ใช้จะเป็นกลุ่มบุคคลที่อยู่ในวงการศึกษานั้น ต่อมาจากนั้นไม่นานทางผู้ผลิตเบราว์เซอร์ได้ทำการเพิ่มความสามารถให้กับเพจโดยอนุญาตให้เพจสามารถแทรกสคริปต์เล็กๆ ลงไปพร้อมกับส่วนที่เป็น HTML ได้ซึ่งจุดนี้ก็คือจุดเริ่มต้นของจาวาสคริปต์ทางฝั่งไคลเอนต์นั่นเองแม้ว่าเพจจะเริ่มมีความสามารถในการโต้ตอบกับผู้ใช้โดยอิงความสามารถจากจาวาสคริปต์แล้วก็ตาม ถ้าพูดถึงในแง่ของส่วนเนื้อหาของตัวเพจจริงๆ แล้วตัวเพจเองก็ยังคงเป็นสแตติกอยู่เช่นเดิม เมื่อกลุ่มผู้ใช้อินเทอร์เน็ตเริ่มมีมากขึ้นความต้องการที่จะทำให้เพจสามารถทำการรับส่งข้อมูลรวมไปถึงการเปลี่ยนแปลงเนื้อหาได้โดยอัตโนมัติก็เกิดขึ้น เทคโนโลยีที่เกิดขึ้นเพื่อรองรับความต้องการเหล่านี้ก็คือแอปพลิเคชันทางฝั่งเซิร์ฟเวอร์ (Server Side Application) นั่นเอง

#### ความหมายของเซิร์ฟเล็ต

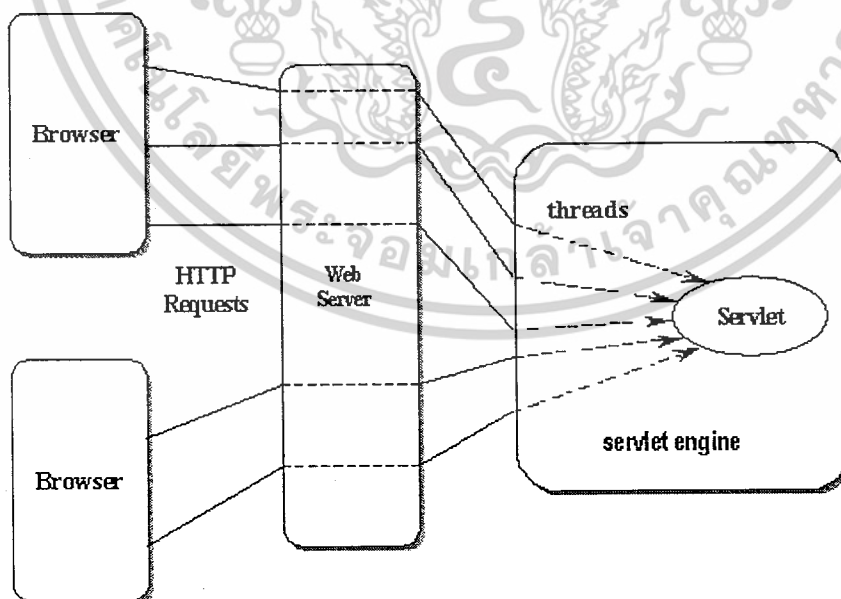
เซิร์ฟเล็ตเป็นแอปพลิเคชันที่อยู่ทางฝั่งเซิร์ฟเวอร์แบบหนึ่งซึ่งอ้างอิงคอนเซ็ปต์มาจาก CGI (Common Gateway Interface) ข้อดีของเซิร์ฟเล็ตที่อยู่เหนือ CGI อย่างแรกก็คือตัวภาษาที่ใช้เขียนซึ่งก็คือภาษาจาวานั้นเอง จาวาเป็นภาษาที่ใช้คอนเซ็ปต์ของออบเจกต์-โอเรียนเต็ด (Object-Oriented) การเขียนโปรแกรมแบบออบเจกต์-โอเรียนเต็ด สามารถลดความซับซ้อนของโครงสร้างโปรแกรมรวมไปถึงการอำนวยความสะดวกในการนำส่วนของโปรแกรมที่เขียนไว้แล้วนำกลับมาใช้ใหม่ได้ นอกจากนี้ จาวายังเป็นภาษาที่เป็นลักษณะแบบไม่ขึ้นกับแพลตฟอร์มซึ่งจะช่วยให้เราสามารถที่จะทำการพัฒนาระบบโดยใช้สภาพแวดล้อมอะไรก็ได้ซึ่งโดยทั่วไปมักนิยมใช้ Window โดยจะนำโปรแกรมที่เขียนเสร็จแล้วมารันบนยูนิคซ์เพื่อเพิ่มความเสถียรภาพของโปรแกรมแทน นอกจากนี้เซิร์ฟเล็ตยังมีความเร็วที่สูงกว่า CGI เพราะเซิร์ฟเล็ตใช้หลักการของเทรด (thread) โดยทำการสร้างหนึ่งเทรดต่อหนึ่งคำร้องขอที่มาจากไคลเอนต์ซึ่งในทางกลับกัน CGI จะทำการสร้างหนึ่งโพรเซส (process) ต่อหนึ่งคำร้องขอ ซึ่งจะทำให้เปลืองทรัพยากรมากกว่าและโพรเซสในการรันก็จะช้ากว่าด้วยท้ายที่สุดจุดเด่นที่สำคัญของเซิร์ฟเล็ตก็คือ API (Application Programming Interface) โดยระบบที่ทำการพัฒนาโดยใช้คอนเซ็ปต์ของเซิร์ฟเล็ตจะสามารถเรียกใช้ API ที่ทางจาวามีมาให้ซึ่งจะช่วยให้การพัฒนาพัฒนาดังกล่าว่ง่ายและเร็วยิ่งขึ้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เมื่อมีการขอเรียกใช้บริการเซิร์ฟเลตมาถึงยังเซิร์ฟเอนจินจะเรียกตัวโหลดคลาส (Class loader) ตัวโหลดคลาสจะทำการตรวจสอบว่าคลาสดังกล่าวถูกโหลดขึ้นมาเรียบร้อยแล้วหรือไม่ และตรวจสอบเวอร์ชันของคลาสที่ถูกโหลดว่ามีค่าใหม่ล่าสุด (time stamp) เท่ากับ .class ไฟล์ (.classfile) ที่อยู่บน (disk) หรือไม่ ถ้าคลาสของเซิร์ฟเลตยังไม่ถูกโหลดหรือมีเวอร์ชันเก่ากว่าไฟล์ที่อยู่บนดิสก์ เซิร์ฟเลตเอนจินจะทำการโหลดคลาสไปเก็บไว้ใน JVM และสร้างอินสแตนซ์ (instance) ของคลาสดังกล่าว หลังจากนั้นวงจรชีวิตของเซิร์ฟเลตจะเริ่มขึ้นใหม่

### เซิร์ฟเลตเอนจิน (Servlet engine)

ในการรันระบบที่เขียนขึ้นโดยใช้หลักการของเซิร์ฟเลต เราจะต้องนำระบบดังกล่าวมาบรรจุอยู่ในสิ่งๆ หนึ่งที่เรียกว่าเซิร์ฟเลตเอนจิน ให้นึกว่าเซิร์ฟเลตเอนจินคล้ายๆ กับกล่องๆ หนึ่งที่ใส่ลูกปิงปองไว้หลายลูก โดยลูกปิงปองแต่ละลูกก็คือระบบๆ หนึ่งนั่นเอง หลายคนอาจสงสัยทำไมถึงใช้คำว่าระบบ โดยทั่วไปแอปพลิเคชันทางฝั่งเซิร์ฟเวอร์ต่างๆ ที่ถูกเขียนขึ้นโดยใช้ API ของเซิร์ฟเลตจะถูกเรียกว่าเซิร์ฟเลตในหนึ่งระบบอาจประกอบด้วยเซิร์ฟเลตหลายอันยกตัวอย่างเช่น ระบบที่เกี่ยวข้องกับ Shopping Cart อาจประกอบด้วยเซิร์ฟเลตที่ทำหน้าที่ในการเช็คสต็อกสินค้า, เซิร์ฟเลตที่ทำหน้าที่ในการเก็บข้อมูลสินค้า, เซิร์ฟเลตที่ทำหน้าที่ในการส่งเมลล์กลับไปยังลูกค้าเพื่อบอกว่าได้ทำการส่งของไปให้แล้ว เป็นต้น ดังนั้นถ้ามองโดยรวมแล้วเซิร์ฟเลตเอนจินก็คือที่ รวมของระบบตั้งแต่หนึ่งระบบถึงหลายระบบ โดยแต่ละระบบจะประกอบด้วยเซิร์ฟเลตหนึ่งอันหรือมากกว่า ซึ่งระบบในที่นี้ อาจหมายถึง Zone (ApacheJserv) หรือ Web Application (Tomcat) ก็ได้



รูปที่ 2.6 Servlet Engine

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

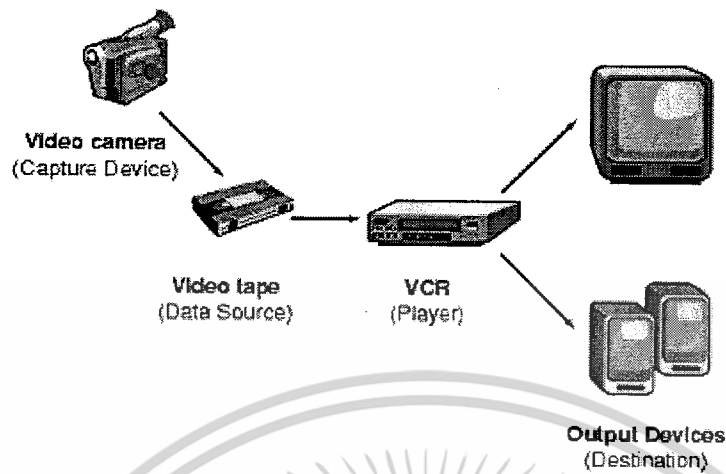
เซิร์ฟเวตเอนจินเป็นเพียงกล่องๆ หนึ่งที่ให้บริการและรันกลุ่มของเซิร์ฟเวตเท่านั้น ในการที่จะทำการติดต่อสื่อสารกับไคลเอนต์ ตัวเซิร์ฟเวตเอนจินนี้จะต้องทำงานร่วมกับเว็บเซิร์ฟเวอร์ ซึ่งเปรียบเสมือนฉากหน้าที่ติดต่อกับไคลเอนต์อีกทีหนึ่ง เมื่อใดก็ตามที่มีคำร้องขอส่งมาจากไคลเอนต์ ถ้าคำร้องขอนั้นจะจงมาที่ตัวเซิร์ฟเวต ทางเว็บเซิร์ฟเวอร์ก็จะทำการส่งคำร้องขอนั้นมาให้เซิร์ฟเวตเอนจินซึ่งทางเซิร์ฟเวตเอนจินก็จะทำการเรียกเซิร์ฟเวตที่ไคลเอนต์ต้องการขึ้นมาทำการประมวลผลคำร้องขอนั้น โดยท้ายสุดเซิร์ฟเวตจะส่งผลกลับไปให้เซิร์ฟเวตเอนจิน เซิร์ฟเวตเอนจินก็จะส่งผลที่ได้กลับไปให้เว็บเซิร์ฟเวอร์ซึ่งเว็บเซิร์ฟเวอร์ก็จะส่งผลกลับไปให้ไคลเอนต์ เซิร์ฟเวตเอนจินอาจจะเป็นส่วนที่ติดมากับเว็บเซิร์ฟเวอร์อยู่แล้วยกตัวอย่างเช่น เซิร์ฟเวตเอนจินที่อยู่ในเซิร์ฟเวอร์ของ Netscape Enterprise, IBM Web Sphere หรืออาจจะเป็นส่วนที่เป็นส่วนที่เพิ่มเติมให้กับเว็บเซิร์ฟเวอร์ก็ได้เช่น Apache Jserv, Tomcat, JRun หรือแม้กระทั่งเป็นส่วนหนึ่งที่อยู่ในเว็บแอปพลิเคชัน เช่น BEA Web logic เป็นต้น ทั้งนี้ การเลือกใช้เซิร์ฟเวตเอนจินแต่ละชนิดก็มักขึ้นอยู่กับปัจจัยหลายอย่างเช่น ความสะดวกในการรวมระบบที่จะสร้างขึ้นมาใหม่กับระบบที่มีอยู่แล้ว, งบประมาณที่มีอยู่สำหรับโครงการหรืออาจจะรวมไปถึงทักษะและประสบการณ์ส่วนตัวของนักพัฒนาแต่ละคน

### 2.2.2 Java Media Framework (JMF)

Java Media Framework (JMF) API ทำให้แอปพลิเคชันและแอปพेटที่ใช้เทคโนโลยีจาวาสามารถใช้ข้อมูลเสียง, วิดีโอและข้อมูลสื่อเชิงเวลาอื่นๆ ได้ แพคเกจเพิ่มเติมนี้สามารถเก็บ, เล่นกลับ, จัดส่งแบบสตรีมและการแปลงข้อมูลที่หลากหลายรูปแบบซึ่งเพิ่มเติมมาจาก Java 2 Platform Standard Edition (J2SE) สำหรับผู้พัฒนามัลติมีเดียโดยที่จัดเตรียมเครื่องมือประสิทธิภาพสูงเพื่อการพัฒนาที่มีความยืดหยุ่นและทำงานข้ามแพลตฟอร์ม

JMF ได้พัฒนามาถึงเวอร์ชันที่สองแล้ว JMF 1.0 API ทำให้ผู้พัฒนาสามารถพัฒนาโปรแกรมที่สามารถแสดงข้อมูลสื่อเชิงเวลาได้ ส่วน JMF 2.0 API ได้เพิ่มเติมความสามารถเพื่อสนับสนุนการเก็บและการบันทึกข้อมูลสื่อ, การควบคุมรูปแบบการประมวลผลในขณะที่กำลังแสดงผลและการประมวลผลตามความต้องการของข้อมูลสตรีมมิ่ง และยังได้กำหนด API ส่วนเสริมทำให้ผู้พัฒนาระดับสูงสามารถแก้ไขและเพิ่มเติมความสามารถได้ตามต้องการ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

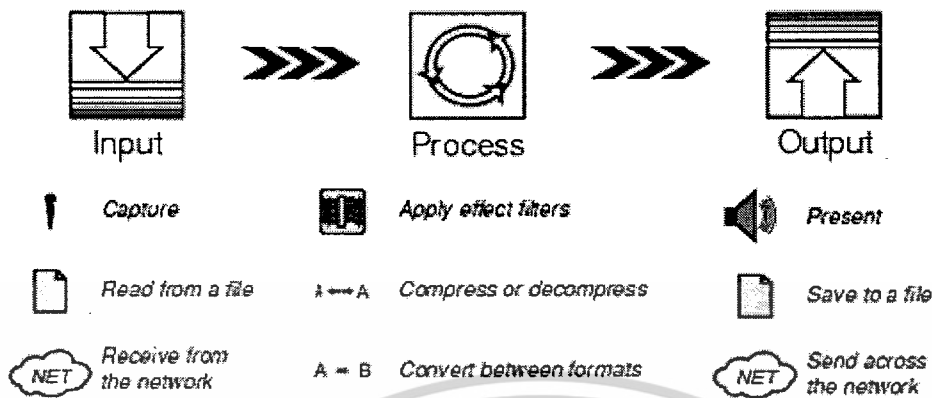


รูปที่ 2.7 ระบบการบันทึก ประมวลผล และการแสดงผลของข้อมูลสื่อเชิงเวลา

ข้อมูลสื่อเชิงเวลา (Time-based media) คือข้อมูลที่มีการเปลี่ยนแปลงโดยขึ้นอยู่กับเวลา อย่างเช่น เสียงและวิดีโอคลิป ลำดับของ MIDI และอะนิเมชัน JMF มีความสามารถที่หลากหลาย อย่างเช่น

- เล่นไฟล์มัลติมีเดียที่หลากหลายในจาวาแอปเพล็ตหรือจาวาแอปพลิเคชัน รูปแบบที่สนับสนุนเช่น AU , AVI, MIDI, MPEG, QuickTime, และ WAV
- เล่นข้อมูลสื่อแบบสตรีมมิ่งจากอินเทอร์เน็ตได้
- เก็บข้อมูลเสียงและภาพจากไมโครโฟนและกล้องวิดีโอ แล้วบันทึกข้อมูลในรูปแบบที่สนับสนุนได้
- ประมวลผลข้อมูลสื่อเชิงเวลาและสามารถเปลี่ยนประเภทของเนื้อหาได้
- จัดส่งข้อมูลภาพและเสียงผ่านอินเทอร์เน็ตแบบเรียลไทม์
- บรอดคาสต์วิทยุและโทรทัศน์แบบสดๆ ได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.8 รูปแบบการประมวลผลข้อมูลสื่อ

### สถาปัตยกรรมของ JMF

เพื่อความง่ายในการทำความเข้าใจสถาปัตยกรรมของ JMF ให้เปรียบเทียบกับระบบสเตอริโอ เมื่อเล่นซีดีเพลงด้วยเครื่องเล่นซีดี ในซีดีจะให้ข้อมูลเพลงกับระบบ ข้อมูลนี้จะถูกเก็บมาก่อนหน้านี้โดยใช้ไมโครโฟนและอุปกรณ์อื่นๆในห้องบันทึกเสียง ไมโครโฟนทำหน้าที่เป็นอุปกรณ์เก็บเสียง ตัวซีดีทำหน้าที่เป็นแหล่งข้อมูล (Data Source) สำหรับระบบสเตอริโอ

เครื่องเล่นซีดีจะให้สัญญาณเพลงกับลำโพง (อุปกรณ์เอาต์พุต) อย่างไรก็ตาม จากเครื่องเล่นซีดีเรายังสามารถเสียบหูฟังเข้าไปได้ ในกรณีนี้ หูฟังทำหน้าที่เป็นอุปกรณ์เอาต์พุต

JMF ใช้รูปแบบเดียวกับตัวอย่างที่ด้านบน โดยมีคำที่เกี่ยวข้องอย่างเช่น

- Data source (แหล่งข้อมูล)
- Capture device (อุปกรณ์เก็บข้อมูล)
- Player (ตัวเล่น)
- Processor (ตัวประมวลผล)
- DataSink (ตัวรับข้อมูล)
- Format (รูปแบบข้อมูล)
- Manager (ตัวจัดการข้อมูล)

### Data source

ใน JMF DataSource หมายถึงข้อมูลเสียง, ข้อมูลภาพหรือข้อมูลทั้งภาพและเสียง อย่างเช่น ซีดีเพลง DataSource สามารถเป็นไฟล์หรือสตรีมจากอินเทอร์เน็ต ใน DataSource จะประกอบด้วย ข้อมูลที่อยู่ของสื่อและโปรโตคอลที่ใช้ในการจัดส่งสื่อ นั้น เมื่อสร้าง DataSource ขึ้นมาแล้วสามารถเอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ส่งให้ Player นำไปแสดงผลโดยไม่สนใจว่าที่อยู่ของ DataSource อยู่ที่ไหนหรือมีรูปแบบเป็นอย่างไร

ข้อมูลสื่อสามารถได้มาจากหลากหลายแหล่งอย่างเช่น ไฟล์ในเครื่องหรือในเน็ตเวิร์คหรือข้อมูลบรอดคาสตจากอินเทอร์เน็ต

#### Capture device

อุปกรณ์เก็บข้อมูลหมายถึงฮาร์ดแวร์ที่ใช้ในการเก็บข้อมูลอย่างเช่น ไมโครโฟน กล้องถ่ายภาพนิ่ง กล้องวิดีโอ ข้อมูลที่เก็บมาได้สามารถนำไปเข้าตัวเล่น (Player) ในการแสดงผลซึ่งสามารถแปลงข้อมูลให้อยู่ในรูปแบบอื่นหรือเก็บไว้ใช้ในอนาคตได้

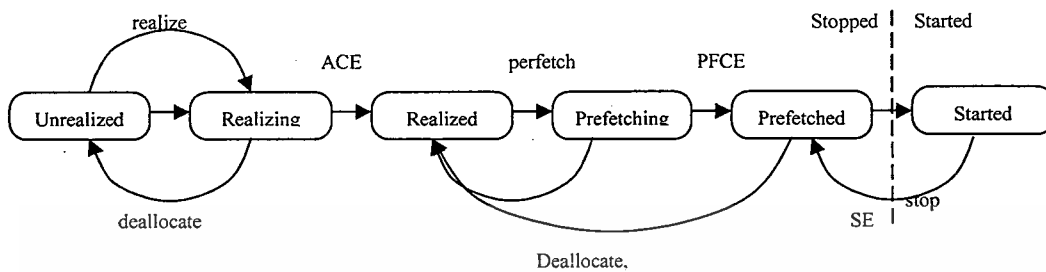


รูปที่ 2.9 รูปแบบของ Player ของ JMF

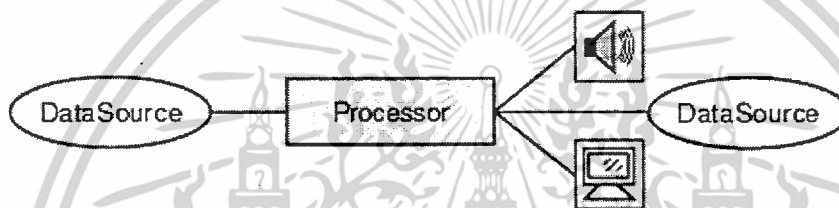
#### Player

Player ทำหน้าที่ในการนำข้อมูลอินพุตเข้ามาในลักษณะสตรีมของภาพและเสียงและทำการแสดงผลไปยังลำโพงหรือจอภาพ เหมือนกับเครื่องเล่นซีดีอ่านข้อมูลจากซีดีและเสียงเพลงไปยังลำโพง Player สามารถมีสถานะซึ่งเป็นธรรมดาเพราะ Player จำเป็นจะต้องเตรียมพร้อมตัวมันและแหล่งข้อมูลก่อนที่จะสามารถเริ่มเล่น เหมือนกับการใส่ซีดีไปในเครื่องเล่นซีดีและเล่นเพลงที่สี่ของซีดี เครื่องเล่นจะไม่สามารถเล่นเพลงได้ทันที มันจะต้องเริ่มค้นหาข้อมูลก่อนสักพักหนึ่งจึงจะสามารถได้ยินเสียง โดยมีสถานะต่างๆดังนี้

- Unrealized
- Realizing
- Realized
- Prefetching
- Prefetched
- Started



รูปที่ 2.10 สถานะของ Player



รูปที่ 2.11 รูปแบบของ Processor ของ JMF

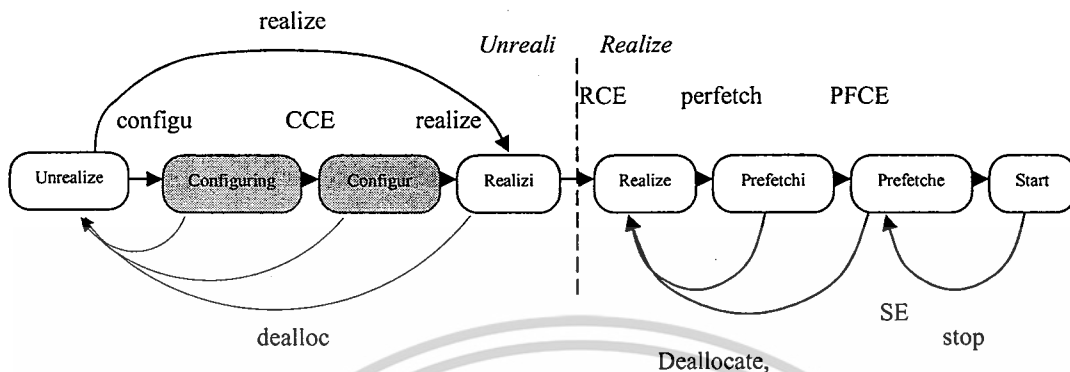
### Processor

Processor เป็นประเภทหนึ่งของ Player ใน JMF API Processor ขยายมาจาก Player อย่างเช่น Processor สนับสนุนการควบคุมการแสดงผลเช่นเดียวกับ Player สิ่งที่แตกต่างกันคือ Processor มีการประมวลผลข้อมูลอินพุตได้

นอกจากการแสดงผลข้อมูล Processor ยังสามารถให้เอาต์พุตข้อมูลผ่าน DataSource ดังนั้น มันจึงสามารถถูกแสดงผลโดย Player หรือ Processor ตัวอื่นได้

นอกจากสถานะหลักสถานะใน Player Processor ยังมีสถานะเพิ่มเติมอีกได้แก่

- Configuring
- Configured



รูปที่ 2.12 สถานะของ Processor

**DataSink**

DataSink เป็นตัวเชื่อมต่อสำหรับออปเจ็คที่อ่านข้อมูลมาจาก DataSource และแสดงผลข้อมูลไปยังปลายทาง ตัวอย่างเช่น ออปเจ็คที่ทำหน้าที่เขียนไฟล์ซึ่งทำหน้าที่บันทึกข้อมูลสื่อลงไฟล์

**Format**

ออปเจ็ค Format หมายถึงออปเจ็คของข้อมูลสื่อที่มีรูปแบบแน่นอน Format จะไม่มีข้อมูลเกี่ยวกับพารามิเตอร์ของการเข้ารหัสหรือข้อมูลทางเวลา แต่จะอธิบายถึงชื่อรูปแบบในการเข้ารหัสและประเภทของข้อมูลที่ต้องการ โดยประกอบด้วยคลาสย่อยคือ AudioFormat และ VideoFormat ใน VideoFormat จะประกอบด้วยคลาสย่อยได้แก่

- H261Format
- H263Format
- IndexedColorFormat
- JPEGFormat
- RGBFormat
- YUVFormat

## Manager

Manager เป็นออปเจ็คต์ตัวกลางที่สร้างมาจากคลาสอื่นๆที่มีอยู่ โดยที่ไม่มีอยู่ในความเป็นจริงเมื่อเทียบกับระบบสแตเรียโอ ซึ่งมีหน้าที่ในการเชื่อมต่อระหว่างสองออปเจ็คต์ อย่างเช่น Manager สามารถสร้าง Player จาก DataSource Manager มีอยู่ที่แบบคือ

- Manager
- PackageManager
- CaptureDeviceManager
- PlugInManager

## 2.3 สื่อแบบสตรีมมิ่ง (Streaming)

เมื่อข้อมูลถูกส่งแบบสตรีมมิ่งไปยังผู้รับตามเวลาจริง ผู้รับสามารถที่จะแสดงข้อมูลได้โดยไม่ต้องรอให้โหลดข้อมูลจนสมบูรณ์ก่อน ในความเป็นจริงสตรีมมิ่งอาจไม่มีแม้แต่การกำหนดเวลาไว้ล่วงหน้า การที่จะโหลดข้อมูลมาจนครบก่อนที่จะเล่นจึงเป็นไปได้ ข้อมูลสื่อแบบสตรีมมิ่งมักจะหมายถึงทั้งเทคนิคในการนำส่งข้อมูลผ่านเน็ตเวิร์คตามเวลาจริง และตัวข้อมูลสื่อที่ส่งแบบตามเวลาจริงนั้น

สื่อแบบสตรีมมิ่งมีอยู่ทั่วไปในอินเทอร์เน็ต การถ่ายทอดสดวิทยุ โทรทัศน์และคอนเสิร์ตต่างๆและในตอนนี้สามารถที่จะนำมาใช้กับการประชุมทางไกลผ่านอินเทอร์เน็ตได้แล้ว โพรโตคอลสำหรับสื่อแบบสตรีมมิ่ง

การส่งข้อมูลสื่อผ่านเครือข่ายในเวลาจริงต้องการทราฟฟิกของเน็ตเวิร์คที่สูง ซึ่งสามารถที่จะซัดเซยข้อมูลที่สูญหายได้ง่ายกว่าข้อมูลที่มิตีเลจสูงในการรับข้อมูล ซึ่งต่างจากข้อมูลอย่างข้อมูลอย่างไฟล์ที่การมาถึงของทุกข้อมูลมีความสำคัญมากกว่า เป็นผลให้โพรโตคอลที่ใช้สำหรับข้อมูลอย่างไฟล์ไม่สามารถนำมาใช้กับข้อมูลแบบสตรีมมิ่งได้

โพรโตคอล HTTP และ FTP มีพื้นฐานจาก Transmission Control Protocol (TCP) ซึ่งอยู่ในชั้นการนำส่งข้อมูลที่ออกแบบมาสำหรับการสื่อสารที่ต้องการความเชื่อถือได้ที่มีแบนวิดธ์ต่ำ ความผิดพลาดสูง เมื่อมีข้อมูลสูญหายหรือผิดพลาดจะต้องมีการส่งใหม่ ทำให้ต้องมีส่วนข้อมูลเพิ่มเติมเป็นผลให้การรับส่งข้อมูลล่าช้า

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

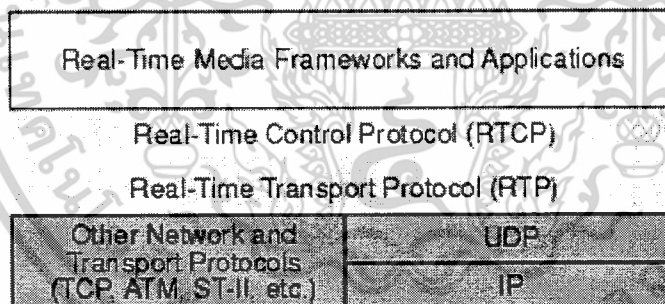
ด้วยเหตุนี้ ข้อมูลแบบสตรีมมิ่งจึงต้องการโปรโตคอลอื่นที่ไม่ใช่ TCP ซึ่งหนึ่งในนั้นก็คือ User Datagram Protocol (UDP) ซึ่งเป็นโปรโตคอลที่ไม่มีค่าน่าเชื่อถือ มันไม่มีการรับประกันว่าข้อมูลนั้นๆจะส่งถึงผู้รับ ไม่มีการรับประกันว่าข้อมูลที่ส่งไปจะเป็นไปตามลำดับ ที่ผู้รับจึงไม่สามารถแก้ไขการส่งข้อมูลที่สูญหาย ข้อมูลที่ซ้ำซ้อนหรือข้อมูลที่ผิดพลาดได้

เช่นเดียวกับ TCP โปรโตคอล UDP อยู่ในชั้นการนำส่งข้อมูลซึ่งอยู่ในระดับล่างของโปรโตคอลชั้นเน็ตเวิร์ค มาตรฐานของอินเทอร์เน็ตสำหรับการส่งข้อมูลแบบเวลาจริงอย่างเช่นเสียงและภาพก็คือ Real-Time Transport Protocol (RTP)

RTP ถูกกำหนดขึ้นโดย IETF RFC 1889 เป็นผลผลิตของกลุ่มงาน AVT ของ Internet Engineering Engineering Task Force (IETF)

### 2.3.1 Real-Time Transport Protocol

RTP ให้บริการในการนำส่งข้อมูลระหว่างจุดต่อจุดในเน็ตเวิร์คสำหรับการส่งข้อมูลแบบเวลาจริง RTP เป็นอิสระจากโปรโตคอลในชั้นเน็ตเวิร์คและชั้นนำส่งแม้ว่ามันจะใช้ UDP



รูปที่ 2.13 สถาปัตยกรรมของ RTP

RTP สามารถใช้ให้บริการได้ทั้งการส่งข้อมูลแบบยูนิคาส (unicast) และมัลติคาส (multicast) เมื่อใช้แบบยูนิคาสข้อมูลชุดเดียวกันถูกทำซ้ำและส่งแยกกันไปยังเป้าหมายที่แตกต่างกัน เมื่อใช้แบบมัลติคาสข้อมูลชุดเดียวกันจะถูกส่งไปเพียงครั้งเดียวไปยังเป้าหมายที่ต่างกัน การส่งข้อมูลแบบมัลติคาสมีประสิทธิภาพที่สูงกว่าสำหรับการนำไปใช้งานอย่างเช่น การประชุมทางผ่านวิดีโอ มาตรฐาน Internet Protocol (IP) สนับสนุนการส่งข้อมูลแบบมัลติคาส

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## 2.4 BARCODE หรือ รหัสแท่ง

คือ การแทนข้อมูลที่เป็นรหัสเลขฐานสอง (Binary codes) ในรูปแบบของแถบสีดำและขาวที่มีความกว้างของแถบที่ต่างกัน แถบที่มีสีและความกว้างที่แตกต่างกันนี้จะมีค่าเป็นตัวเลขที่แตกต่างกันและมาตรฐานสากลได้กำหนดค่าไว้

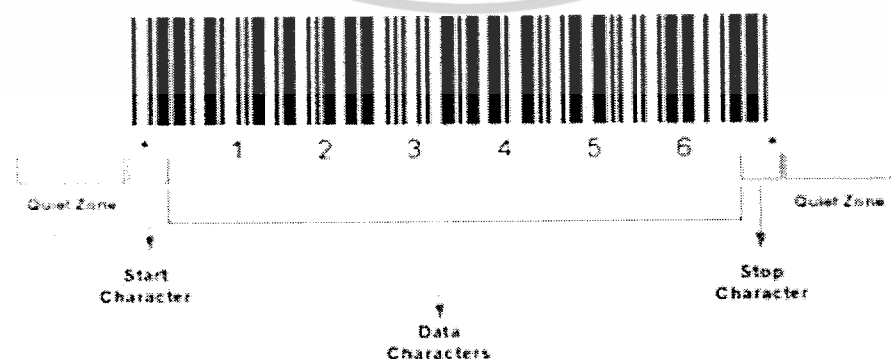
เทคโนโลยีบาร์โค้ดถูกนำมาใช้ทดแทนในส่วนการบันทึกข้อมูล (Data Entry) การบันทึกด้วยคีย์บอร์ดมีอัตราความผิดพลาดอยู่ประมาณ 1 ใน 100 หรือบันทึกข้อมูลผิดพลาด 1 ตัวอักษรในทุกๆ 100 ตัวอักษร และเมื่อเปลี่ยนมาใช้ระบบบาร์โค้ดแทนในขั้นตอนการบันทึกข้อมูล อัตราการเกิดความผิดพลาดจะลดลงเหลือเพียง 1 ใน 10,000,000 ตัวอักษร

### 2.4.1 ส่วนประกอบของบาร์โค้ด

สัญลักษณ์ของบาร์โค้ดที่ใช้กันมีการกำหนดขึ้นมาหลายรูปแบบ ตามมาตรฐานของแต่ละองค์กร และตามจุดประสงค์ของการใช้งาน แต่โดยทั่วไปแล้วบาร์โค้ดจะมีส่วนประกอบต่างๆ ดังต่อไปนี้

1. Quiet Zone เป็นบริเวณที่ว่างเปล่าไม่มีการพิมพ์ข้อความใดๆ โดยจะอยู่ก่อนและหลังบาร์โค้ด
2. Start/ Stop Character เป็นบริเวณแถบแท่งหรือช่องว่าง
3. Data เป็นบริเวณแถบแท่งหรือช่องว่างที่แทนข้อมูลต่างๆ ที่เราต้องการ
4. Check Digit เป็นบริเวณแถบแท่งที่ไว้สำหรับเก็บค่าตัวเลข เพื่อตรวจสอบในข้อมูลส่วน Data เพื่อเตรียมสั่งให้เซนเซอร์เริ่มต้นหรือหยุดบาร์โค้ด เพื่อให้มั่นใจว่าถูกต้องแม่นยำ

### Structure of a Width - Based Bar Code



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับรูปที่ 2.14 ส่วนประกอบของบาร์โค้ด อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## 2.4.2 หลักการทำงาน

1. ส่วนลายเส้น ซึ่งเป็นลายเส้นสีขาว (โปร่งแสง) และสีดำ มีขนาดความกว้างของลายเส้นตามมาตรฐานแต่ละชนิดของบาร์โค้ด

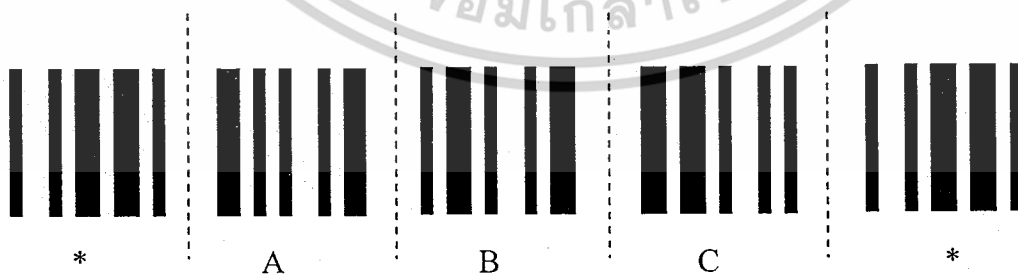
2. ส่วนข้อมูลตัวอักษร เป็นส่วนที่แสดงความหมายของชุดข้อมูลลายเส้นสำหรับให้อ่านเข้าใจ

3. ส่วนแถบว่าง เป็นส่วนที่เครื่องอ่านบาร์โค้ดใช้กำหนดขอบเขตของบาร์โค้ดและกำหนดค่าให้กับสีขาว (ความเข้มของการสะท้อนแสงในสีของพื้นผิวแต่ละชนิดที่ใช้แทนสีขาว) โดยทุกเส้นจะมีความยาวเท่ากันเรียงตามลำดับในแนวนอนจากซ้ายไปขวา แถบสีทั้งสีขาวและสีดำที่มีความกว้างจะแทนค่าเป็น 1 และแถบสีที่มีความแคบ (หรือมองด้วยตาเหมือนเป็นเส้นตรงเล็กๆ) ทั้งขาวและดำจะมีค่าเป็น 0 แถบขาวและดำที่มีลักษณะและชื่อที่ใช้คือ

- แถบสีดำที่มีความกว้างมากกว่าเรียกว่า Wide Bar ถ้ามีความกว้างน้อยกว่าเรียกว่า Narrow Bar
- ช่องว่างหรือแถบสีขาวที่มีความกว้างมากกว่าเรียกว่า Wide Space ถ้ามีความกว้างน้อยกว่าเรียกว่า Narrow Space

## 2.4.3 ตัวอย่างรหัสบาร์โค้ด (Code39)

Code39 เป็นมาตรฐานในการเข้ารหัสบาร์โค้ดแบบใช้ความกว้างในการแทนสัญลักษณ์ซึ่งประกอบด้วยโค้ดเริ่มต้นและโค้ดหยุดแทนด้วยสัญลักษณ์ (\*)



รูปที่ 2.15 ตัวอย่างรหัส code39 แทน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 2.1 ตารางแสดงการเข้ารหัสความกว้างและรหัสไบนารีของรหัส ASCII

ASCII Code	Width Encoding	Barcode Encoding	ASCII Code	Width Encoding	Barcode Encoding
0	NNNWWNWN	000110100	M	WNWNNNNWN	101000010
1	WNNWNNNNW	100100001	N	NNNNWNNWW	000010011
2	NNWNNNNNW	001100001	O	WNNNWNWN	100010010
3	WNWNNNNN	101100000	P	NNWNWNNWN	001010010
4	NNNWWNNNW	000110001	Q	NNNNNNWWW	000000111
5	WNNWNNNN	100110000	R	WNNNNNWWN	100000110
6	NNWNNNNN	001110000	S	NNWNNNWWN	001000110
7	NNNWNWNW	000100101	T	NNNNWNWWN	000010110
8	WNNWNNWN	100100100	U	WWNNNNNNW	110000001
9	NNWNNWN	001100100	V	NWNNNNNNW	011000001
A	NNWNNWN	001100100	W	WWWNNNNN	111000000
B	NNWNNWN	001001001	X	NWNNWNNW	010010001
C	WNWNNWN	101001000	Y	WWNNWNNN	110010000
D	NNNNWNNW	000011001	Z	NWNNWNNN	011010000
E	WNNNWN	100011000	-	NWNNNNWN	010000101
F	NNWNWN	001011000	.	WWNNNNWN	110000100
G	NNNNWNNW	000001101	SPACE	NWNNNNWN	011000100
H	WNNNNWN	100001100	\$	NWNWNWN	010101000
I	NNWNNWN	001001100	/	NWNWNWN	010100010
J	NNNNWNN	000011100	+	NWNNWNWN	010001010
K	WNNNNNW	100000011	%	NNNWNWN	000101010
L	NNWNNNW	001000011	*	NWNNWNWN	010010100

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## 2.5 อินฟราเรดเซ็นเซอร์ (Infrared Sensor)

อินฟราเรดเป็นแสงที่มีความยาวคลื่นอยู่ในช่วงประมาณ 780-3000 นาโนเมตร ซึ่งเป็นแสงที่ไม่สามารถมองเห็นได้ด้วยตาเปล่าจึงเป็นที่นิยมที่จะนำมาใช้ในการสื่อสารหรือตรวจจับสิ่งของต่างๆ เพราะปัญหาการรบกวนของสัญญาณของแสงอื่นๆมีน้อย อีกทั้งการสร้างวงจรที่ใช้ในระบบอินฟราเรดก็ง่ายไม่มีความซับซ้อนมากนักและความน่าเชื่อถือของสัญญาณที่ส่งก็มีความเชื่อถือที่สูงในการนำไปใช้งาน

โดยในระบบอินฟราเรดจะต้องมีเครื่องส่งและเครื่องรับ ซึ่งการสร้างเครื่องส่งนั้นก็เพียงแต่ให้มีการส่งแสงออกมาในช่วงความถี่ที่สูงกว่าความถี่ทั่วไปของแสงธรรมาคือ ต้องมากกว่า 20 KHz โดยจะใช้ IR LED เป็นตัวขับแสงอินฟราเรด ส่วนการสร้างเครื่องรับนั้นเราจะใช้โฟโตไดโอดหรือโฟโตทรานซิสเตอร์เป็นตัวรับแสง โดยที่ทั้งเครื่องรับและเครื่องส่งจะต้องมีความถี่เท่ากัน เพราะถ้าไม่เท่ากันจะทำให้การ Detect สัญญาณได้ไม่ตรง แต่สำหรับโครงการนี้จะใช้คุณสมบัติของตัวโฟโตทรานซิสเตอร์

## 2.6 ระบบสื่อสารข้อมูลอนุกรมแบบหนึ่งสาย (1-Wire™ Serial Bus)

ระบบการสื่อสารข้อมูลแบบนี้ผู้คิดคือ “ดัลลาสเซมิคอนดักเตอร์” ดังนั้นในบางครั้งจึงเรียกระบบสื่อสารข้อมูลแบบนี้ว่า “ระบบสื่อสารข้อมูลดัลลาสหนึ่งสาย” (The Dallas 1-Wire Bus) ระบบสื่อสารข้อมูลแบบนี้เป็นระบบที่มีความฉลาด และใช้จำนวนสายสัญญาณเพียง 1 เส้นเท่านั้น โดยไม่ต้องมีสายสัญญาณนำพิกามาควบคุมจังหวะการถ่ายทอดข้อมูลเหมือนกับระบบสื่อสารอนุกรมแบบอื่น ๆ เนื่องจากสายข้อมูลนั้นจะทำหน้าที่เสมือนหนึ่งเป็นสายสัญญาณนำพิกาในตัว ส่วนค่าของข้อมูลจะพิจารณาจากลักษณะของรูปสัญญาณที่ปรากฏบนสายสัญญาณ ในแต่ละช่องของเวลาซึ่งเรียกว่า “ไทม์สล็อต (Time-Slot)” โดยคาบเวลาดำสุดและสูงสุดของสถานะต่าง ๆ ในการสื่อสารข้อมูลในแต่ละไทม์สล็อต มีการกำหนดของเขตไว้อย่างชัดเจน การถ่ายทอดข้อมูลจะเกิดขึ้นในแต่ละไทม์สล็อตนั้น รูปแบบการถ่ายทอดข้อมูลจะเป็นแบบอะซิงโครนัสในระดับบิต ไม่มีการกำหนดความยาวของข้อมูลระดับไบต์ ระบบสื่อสารแบบนี้เหมาะที่จะใช้ในการสื่อสารข้อมูลระหว่างไอซีบนแผงวงจรเดียวกันหรือสร้างเป็นโครงข่ายสื่อสารแบบทวิสต์แพร์ก็ได้

สายสัญญาณบนระบบบัสแบบหนึ่งสายนี้จะป็นสายสัญญาณแบบสองทิศทางแต่ข้อมูลจะสามารถเดินทางได้ในทิศทางเดียวในช่วงเวลาหนึ่ง ๆ นั่นคือ มีลักษณะคล้ายกับระบบสื่อสารแบบฮาล์ฟดูเพล็กซ์ (Half Duplex) ตัวอย่างที่เห็นได้ชัดคือ การใช้งานวิทยุสื่อสารหรือวิทยุสมัครเล่น อุปกรณ์บนระบบบัสต้องมีการระบุอย่างชัดเจนว่าตัวใดเป็นอุปกรณ์มาสเตอร์ ตัวใดเป็นอุปกรณ์สเลฟ โดยส่วนใหญ่อุปกรณ์มาสเตอร์คือ ไมโครคอนโทรลเลอร์ ส่วนอุปกรณ์สเลฟ ได้แก่ ไอซีตรวจจับ

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

อุณหภูมิ, ไอซีหน่วยความจำแรม เป็นต้น อุปกรณ์มาสเตอร์จะเป็นตัวจัดเตรียมความพร้อมของ สายสัญญาณและควบคุมการถ่ายทอดข้อมูลบนสายสัญญาณนั้น ข้อมูลทั้งหมดไม่ว่าจะเป็นข้อมูล ควบคุมหรือข้อมูลใช้งานจะถูกส่งลงสู่สายสัญญาณที่มีเพียงเส้นเดียวนี้ทั้งหมด ในระหว่างการทำงาน อุปกรณ์มาสเตอร์และสเลฟสามารถเป็นได้ทั้งตัวส่งและตัวรับ ขึ้นอยู่กับเงื่อนไขของการทำงานใน ขณะนั้น ยกตัวอย่าง ถ้าหากมีการเขียนข้อมูลจากอุปกรณ์มาสเตอร์ไปยังอุปกรณ์สเลฟ ตัวส่งคือ อุปกรณ์มาสเตอร์ ตัวรับ คือ อุปกรณ์สเลฟ ในทางตรงกันข้าม หากเป็นการอ่านข้อมูลจากอุปกรณ์ส เลฟ ตัวส่งจะกลายเป็นอุปกรณ์ และตัวรับคืออุปกรณ์มาสเตอร์ ในระบบบัสหนึ่งระบบต้องมีอุปกรณ์ มาสเตอร์เพียงตัวเดียวเท่านั้น

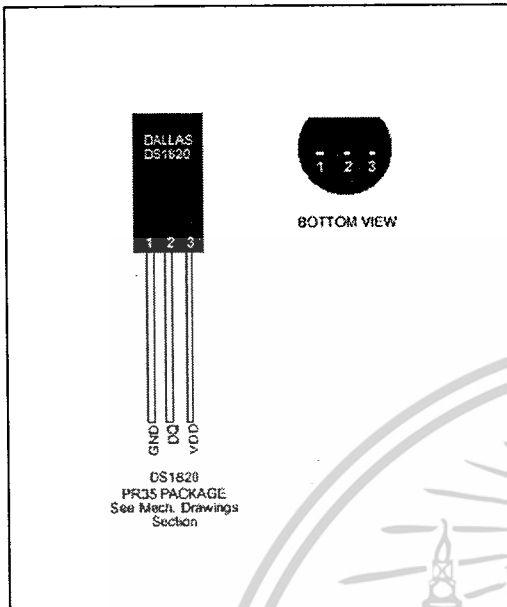
สายสัญญาณของระบบบัสนี้ต้องกำหนดสถานะปกติไว้ที่ลอจิกสูง สามารถทำได้โดยการต่อ ตัวต้านทานค่าประมาณ 4.7k โอห์ม พูลอัพกับไฟเลี้ยง +5V ดังนั้นอุปกรณ์ที่นำเข้ามาต่อบนระบบบัส นี้จึงต้องออกแบบให้ภาคเอาต์พุต ที่ต้องต่อกับสายสัญญาณมีลักษณะเป็นคอลเล็กเตอร์เปิดหรือครอน เปิด

## 2.7 ไอซีตรวจจับอุณหภูมิ DS1820

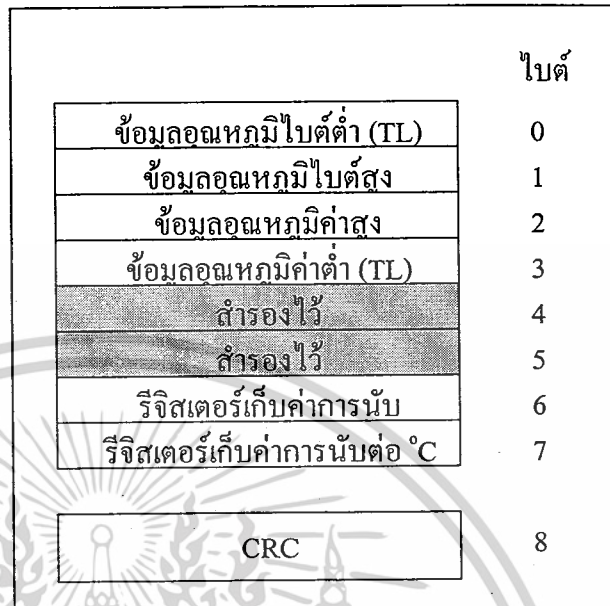
เป็นไอซีตรวจจับอุณหภูมิที่ใช้การติดต่อแบบระบบบัสหนึ่งสาย มีขาต่อใช้งานเพียง 3 ขา คือ DQ ซึ่งเป็นขาเชื่อมต่อกับระบบบัส, ขาต่อไฟเลี้ยงภายนอก และขากราวด์ ดังรูปที่ 2.10

หัวใจสำคัญของ DS1820 อยู่ที่ตัวตรวจจับอุณหภูมิ และหน่วยความจำความเร็วสูง ที่เรียกว่า “สแครตช์แพด (Scratchpad)” ดังรูปที่ 2.11 ซึ่งมีขนาด 9 ไบต์ เมื่อวัดอุณหภูมิได้ก็จะนำค่าที่ได้นี้มาเก็บ ไว้ในสแครตช์แพดที่ไบต์ 0 และ 1 ทั้งนี้เนื่องจากไอซี DS1820 สามารถให้ข้อมูลของอุณหภูมิได้ ละเอียดถึง 16 บิต เมื่อนำมาแปลงเป็นข้อมูลเลขฐานสิบจึงสามารถแสดงความละเอียดของค่าอุณหภูมิได้ ถึง 0.5 องศาเซลเซียสและ 0.9 องศาฟาเรนไฮต์ โดยมีย่านวัดอุณหภูมิ -55 ถึง +125 องศาเซลเซียส หรือ -67 ถึง +257 องศาฟาเรนไฮต์ โดยค่าขององศาฟาเรนไฮต์ต้องใช้ในการแปลงหน่วยเข้ามาช่วย ใช้เวลาใน การแปลงอุณหภูมิเป็นข้อมูลดิจิทัลประมาณ 200 มิลลิวินาที สามารถกำหนดขอบเขตของอุณหภูมิที่ทำการวัดได้ และให้แจ้งเตือนเมื่อค่าของอุณหภูมิสูงขึ้นหรือลดต่ำลงถึงค่าที่กำหนด โดยค่าที่ค่าที่กำหนดนี้ จะเก็บไว้ในสแครตช์แพดใน ไบต์ที่ 2 และ 3

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.16 ไอซี DS1820



รูปที่ 2.17 การจัดสรรพื้นที่ของสแควร์แพค

โดยตัวไอซี DS1820 จะให้ข้อมูลค่าอุณหภูมิมาเป็นเลขไบนารีขนาด 16 บิต โดยจะแบ่งเป็นข้อมูลอุณหภูมิไบต์ต่ำขนาด 8 บิต และข้อมูลอุณหภูมิไบต์สูงขนาด 8 บิต แล้วสามารถนำค่าอุณหภูมิที่ได้มาแปลงเป็นเลขฐานสิบได้ตั้งแต่ -55 ถึง +125 องศาเซลเซียส โดยดูตัวอย่างการแปลงข้อมูลเป็นเลขฐานสิบได้ดังตารางที่ 2.2

ตารางที่ 2.2 การแปลงค่าอุณหภูมิเป็นเลขฐานสิบ และฐานสิบหก

TEMPERATURE	DIGITAL OUTPUT (Binary)	DIGITAL OUTPUT (Hex)
+85.0°C*	0000 0000 1010 1010	00AAh
+25.0°C	0000 0000 0011 0010	0032h
+0.5°C	0000 0000 0000 0001	0001h
0°C	0000 0000 0000 0000	0000h
-0.5°C	1111 1111 1111 1111	FFFFh
-25.0°C	1111 1111 1100 1110	FFCEh
-55.0°C	1111 1111 1001 0010	FF92h

\*The power-on reset value of the temperature register is +85°C

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## 2.8 ไมโครคอนโทรลเลอร์

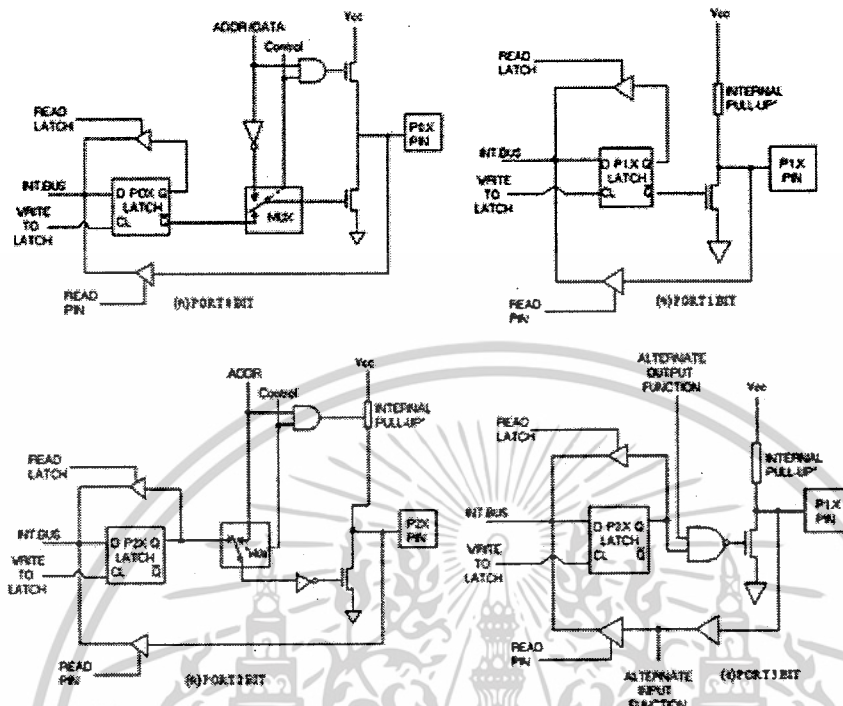
ไมโครคอนโทรลเลอร์ตระกูล MCS-51 ซึ่งมีหน่วยความจำเป็นแบบแฟลช (flash memory) อนุกรม AT89xx มีคุณสมบัติดังต่อไปนี้

- เป็นไมโครคอนโทรลเลอร์ที่ใช้ซีพียูขนาด 8 บิต
- ภายในมีหน่วยความจำโปรแกรมเป็นแบบแฟลชสามารถลบและเขียนใหม่ได้พันครั้ง
- หน่วยความจำข้อมูลพื้นฐานเป็นหน่วยความจำแบบแรม ในบางเบอร์จะมีหน่วยความจำแบบอีอีพรอมเพิ่มเติม
- ขาพอร์ตเป็นแบบสองทิศทาง สามารถใช้งานเป็นได้ทั้งอินพุตและเอาต์พุต
- มีวงจรสื่อสารอนุกรมแบบฟูลดูเพล็กซ์
- ไทมเมอร์ / เคาน์เตอร์ขนาด 16 บิตอย่างน้อย 2 ตัว
- สามารถรองรับแหล่งกำเนิดอินเทอร์รัปต์ได้ 6 ประเภท
- สามารถขยายหน่วยความจำภายนอกเพิ่มเติมได้สูงสุด 64 กิโลไบต์
- มีวงจรถ่ายสัญญาณอนาล็อกภายในชิป

### 2.8.1 โครงสร้างของพอร์ต

ไมโครคอนโทรลเลอร์ MCS-51 แบบแฟลชมีพอร์ตให้ใช้งานทั้งสิ้น 4 พอร์ตคือ พอร์ต 0 ถึงพอร์ต 3 แต่ละพอร์ตมีขนาด 8 บิต เป็นพอร์ตแบบ 2 ทิศทาง กล่าวคือ สามารถเป็นได้ทั้งอินพุตสำหรับรับสัญญาณข้อมูลเข้ามา และเอาต์พุตสำหรับส่งสัญญาณข้อมูลออกไป ขาของพอร์ตแสดงโครงสร้างภายในได้ ดังรูปที่ 2.18 โดยมีโครงสร้างเป็น Field-effect Transistor ต่อกับขาภายนอก และมีความต้านทานต่อ Pull-up อยู่สำหรับพอร์ต 1, 2, 3 เมื่อเป็นพอร์ต 0 ไม่มีตัวต้านทาน Pull-up ภายใน เพราะต้องใช้เป็นขาบัสดำเนินงาน และบัสดำเนินงาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.18 โครงสร้างพอร์ตทั้ง 4 ของ MCS-51

### 2.8.2 การจัดหน่วยความจำของ MCS-51

1. ชุดรีจิสเตอร์ 4 ชุดแต่ละชุดเรียกว่ารีจิสเตอร์ชุด ที่ตำแหน่ง 00H ถึง 1FH โดย แต่ละชุดประกอบด้วยรีจิสเตอร์ R0 ถึง R7
2. หน่วยความจำที่สามารถเข้าถึงข้อมูลระดับบิตได้ ตำแหน่ง 20H ถึง 2FH
3. หน่วยความจำใช้งานทั่วไปตำแหน่ง 30H ถึง 7FH
4. รีจิสเตอร์ฟังก์ชันพิเศษ ตำแหน่ง 80H ถึง FFH

แผนผังการจัดการหน่วยความจำข้อมูลภายใน แสดงได้ดังรูปที่ 5 จากแผนผังแสดงถึง การอ้างตำแหน่ง หน่วยความจำภายในซึ่งอ้างได้ 2 แบบ ได้แก่ การอ้างไปที่ตำแหน่งของไบต์ (เขียนหมายเลขด้านนอก) หรือการอ้างไปที่ตำแหน่งของบิต (เขียนหมายเลขตำแหน่งด้านใน) โดยตำแหน่งของหน่วยความจำ ที่อ้างเป็นแบบบิตที่แน่นอน

### 2.8.3 รีจิสเตอร์เลือกโหมดการทำงานของไทมเมอร์/เคาน์เตอร์ (TMOD)

เป็นรีจิสเตอร์สำหรับควบคุม Mode การทำงานของ Timer/Counter และไม่สามารถอ้างอิงระดับบิตได้ ซึ่งจะแบ่งเป็นสองส่วน โดยที่สี่บิตแรกของรีจิสเตอร์นี้ใช้สำหรับ Timer/Counter 0 และสี่บิตบนสำหรับ Timer/Counter 1 มีรายละเอียดดังรูปที่ 2.19

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

GATE	C/T	M1	M0	GATE	C/T	M1	M0
------	-----	----	----	------	-----	----	----

รูปที่ 2.19 รีจิสเตอร์เลือกโหมดการทำงานของไทเมอร์/เคาน์เตอร์

**GATE:** เป็นบิตใช้สำหรับการควบคุม Timer/Counter ถ้าเป็นลอจิก 1 Timer/Counter เมื่อขาสัญญาณที่ขา INTx เป็นลอจิก 1 เท่านั้น และบิต TRx ใน TCON ต้องเป็นลอจิก 1 ด้วย ซึ่งจะเป็นการควบคุมด้วยฮาร์ดแวร์ เมื่อเป็นลอจิก 0 การควบคุม Timer/Counter นั้นๆ จะเป็นการควบคุมด้วยโปรแกรมที่บิต TRx ของรีจิสเตอร์ TCON เท่านั้น

**C/T:** เป็นบิตสำหรับเลือกการทำงานของ Timer/Counter เป็น Timer หรือ Counter โดยที่ถ้าเป็นลอจิก 1 จะทำหน้าที่เป็น Counter โดยรับสัญญาณจากขา TX ของ 8051 และลอจิก 0 เป็น Timer โดยความถี่ที่นับได้จากความถี่ภายในของ 8051 M0, M1

**M0, M1:** เป็นบิตสำหรับเลือกการทำงานของ Timer/Counter ดังแสดงในตารางต่อที่ 2.2

ตารางที่ 2.3 การเลือกการทำงานของ Timer/Counter

M0	M1	การทำงาน
0	0	เป็น Timer ขนาด 13 บิต เมื่อการทำงานของตระกูล 48
0	0	เป็น Timer/Counter ขนาด 16 บิต
1	0	เป็น Timer/Counter ขนาด 8 บิต ที่สามารถตั้งค่าใหม่อัตโนมัติ
1	1	Timer/Counter 0 จะถูกแบ่งออกเป็น Timer/Counter 2 ชุด ขนาด 8 บิต โดยที่ TLO จะเป็น Timer/Counter ขนาด 8 บิต ที่ถูกควบคุมของ Timer/Counter 0 และ TH0 จะเป็น Timer/Counter ขนาด 8 บิต ที่ควบคุมโดยชุดควบคุมของ Timer/Counter 1
1	1	Timer/Counter 1 จะหยุดทำงาน

#### 2.8.4 การรับส่งข้อมูลอนุกรม ใน MCS-51

การสื่อสารข้อมูลแบบอนุกรมเป็นการรับ และส่งข้อมูลแบบทีละบิตข้อมูลเรียงกันไปโดยใช้สายสัญญาณเพียงสองหรือสามเส้น ต่างจากการสื่อสารแบบขนาน ซึ่งสามารถรับส่งข้อมูลได้เร็วกว่าแต่จะต้องใช้สายสัญญาณเท่ากับจำนวนบิตข้อมูลที่จะรับและส่งในแต่ละครั้ง ถ้าต้องการรับและส่งข้อมูลเป็นระยะทางไกล จะทำให้ต้องใช้สายสัญญาณเป็นจำนวนมาก และไม่ค่อยมีอุปกรณ์สนับสนุน การสื่อสารแบบอนุกรม จึงเป็นที่นิยมกว่า และถูกสร้างไว้เป็นมาตรฐานในไมโครคอนโทรลเลอร์หลายตระกูล ในตระกูล MCS-51 ก็เช่นกันที่มีพอร์ตที่ใช้ในการรับและส่งไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ข้อมูลแบบอนุกรม อยู่ภายในชิปอยู่แล้ว ในการใช้งานเราไม่จำเป็นต้องต่อชิปสนับสนุนเพิ่มเติมอีก (กรณีที่ติดต่อกันเอง) การทำงานเป็นแบบ full-duplex คือสามารถรับและส่งข้อมูลได้พร้อมๆกัน ในการรับข้อมูลจะมีบัฟเฟอร์ในการรับข้อมูลให้ด้วย ทำให้ MCS-51 สามารถรับข้อมูลไปตัดไปได้ โดยไม่ต้องรอให้อ่านข้อมูลไปต์แรกไปเก็บก่อน ทำให้ข้อมูลไม่สูญหาย เกิดความคล่องตัวในการรับข้อมูล

การรับและส่งข้อมูลจะอาศัยรีจิสเตอร์ขนาด 8 บิตสองตัว ซึ่งแยกกันทำงานสองอย่างคือใช้รับข้อมูลที่มาจากภายนอกและใช้สำหรับส่งข้อมูลออกไปภายนอก รีจิสเตอร์ทั้งสองมีตำแหน่งเดียวกันใน SFR (Special function register) มีชื่อเรียกว่า SBUF (ตำแหน่ง 99H) การเข้าถึงข้อมูลในรีจิสเตอร์ทั้งสองนั้น MCS-51 จะทราบเองว่าผู้ใช้ต้องการติดต่อกับรีจิสเตอร์ SBUF ที่ใช้รับหรือส่งจากรหัสคำสั่ง เพราะ SBUF ที่ใช้รับข้อมูลจะเป็น Operand ตัวแรกในคำสั่ง ส่วน SBUF ที่ใช้ส่งข้อมูลจะเป็น Operand ตัวที่สองในคำสั่ง การทำงานสามารถ กำหนดโหมดการทำงานได้ 4 แบบ โดยควบคุมจากค่าในรีจิสเตอร์ SCON

#### SCON

การใช้งานพอร์ตอนุกรมของ MCS-51 มีความคล่องตัวสูงมากทั้งนี้เนื่องจากผู้ใช้สามารถควบคุมการทำงานที่แตกต่างกันได้ถึง 4 ประเภท โดยควบคุมจากบิตต่างๆ ในรีจิสเตอร์ควบคุมพอร์ตอนุกรม SCON (ตำแหน่ง 99H) ซึ่งสามารถเข้าถึงได้ในระดับบิต

Bit	7	6	5	4	3	2	1	0
	SM0	SM1	SM2	REN	TB8	RB8	T1	R2

รูปที่ 2.20 รีจิสเตอร์ควบคุมพอร์ตอนุกรม SCON

*SM0* บิตเลือกการทำงานของพอร์ตสื่อสารข้อมูลแบบอนุกรมควบคุมได้โดยโปรแกรม

*SM1* บิตเลือกการทำงานของพอร์ตสื่อสารข้อมูลแบบอนุกรมควบคุมได้โดยโปรแกรม

*SM0*   *SM1*   Mode

0	0	0	ทำงานเป็น shift register อัตราเร็วในการรับหรือส่งข้อมูลเท่ากับ $OSC/12$
0	1	1	8 Bit UART อัตราเร็วในการรับหรือส่งข้อมูลกำหนดเองได้
1	0	2	9 Bit UART อัตราเร็วในการรับหรือส่งข้อมูล = $OSC/32$ หรือ $OSC/64$
1	1	3	9 Bit UART อัตราเร็วในการรับหรือส่งข้อมูลกำหนดเองได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

*SM2* บิตการเลือกใช้งานพอร์ตสื่อสารข้อมูลแบบอนุกรมในโหมด 2 หรือ 3 เพื่อใช้ติดต่อระหว่างซีพียูด้วยกันเอง ควบคุมโดยใช้โปรแกรม

1 ใช้พอร์ตสื่อสารข้อมูลแบบอนุกรมติดต่อระหว่างซีพียูด้วยกันเอง เมื่อข้อมูลบิตที่ 9 ที่ได้รับมีค่าเป็น 0 (ค่าต่ำไบต์) บิต RI จะไม่ถูกเซตแต่หากข้อมูลที่ 9 มีค่าเป็น 1 (แอดเดรสไบต์) บิต RI จะถูกเซต

0 ใช้พอร์ตสื่อสารข้อมูลแบบอนุกรมโหมด 1 และโหมด 3 ตามปกติในการทำงานของพอร์ตสื่อสารข้อมูลแบบอนุกรมโหมด 1 หากบิต SM2 ถูกเซต บิต RI จะไม่ถูกเซตจนกว่าบิตสิ้นสุดของข้อมูลจะถูกรับเข้ามา ในการทำงานของพอร์ตสื่อสารข้อมูลแบบอนุกรมโหมด 0 บิตนี้ควรถูกเคลียร์ให้เป็น 0

*REN* บิตควบคุมการอนุญาตให้มีการรับข้อมูลสามารถควบคุมได้โดยโปรแกรม

1 อนุญาตให้มีการรับข้อมูล

0 ไม่อนุญาตให้มีการรับข้อมูล

*TB8* ข้อมูลบิตที่ 9 ซึ่งถูกส่งออกไปในการทำงานของพอร์ตสื่อสารข้อมูลแบบอนุกรมโหมด 2 และ 3 ควบคุมได้ด้วยคำสั่งในโปรแกรมเท่านั้น

*RB8* ข้อมูลบิตที่ 9 ที่ได้รับเข้ามาจากภายนอก ในการทำงานของพอร์ตสื่อสารข้อมูลแบบอนุกรมโหมด 2 และ 3 ส่วนในการทำงานโหมด 1 ถ้าบิต SM2 = 0 บิตนี้จะเป็นบิตสิ้นสุดของข้อมูลที่รับเข้ามาได้และมาถูกกำหนดการใช้งานในโหมด 0

*TI* บิตบอกสถานะสัญญาณอินเตอร์รัพท์ที่เกิดจากการส่งข้อมูลถูกเซตโดยฮาร์ดแวร์ เมื่อข้อมูลเริ่มส่งบิตสิ้นสุดของข้อมูลออกไปและจะต้องเคลียร์โดยคำสั่งในโปรแกรมเท่านั้น

*RI* บิตบอกสถานะสัญญาณอินเตอร์รัพท์ที่เกิดจากการรับข้อมูล ถูกเซตโดยฮาร์ดแวร์เมื่อได้รับข้อมูลบิตที่ 8 ในการทำงานโหมด 0 หรือที่ถูกครึ่งทางของช่วงรับบิตสิ้นสุดของข้อมูลในการทำงานโหมดอื่นและจะต้องเคลียร์ในโปรแกรมเท่านั้น

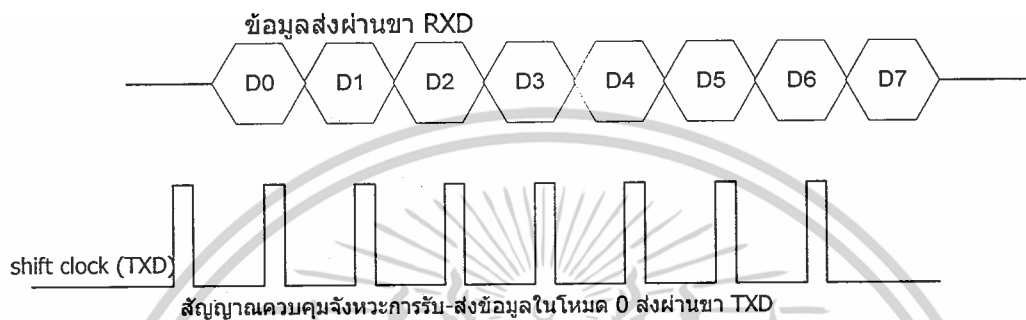
โหมดของการรับ-ส่งข้อมูลแบบอนุกรม

โหมด 0 (*Shift Register Mode*)

ขา P3.0 (RXD) ถูกใช้เป็นที่สำหรับรับและส่งข้อมูล ขา P3.1(TXD) ใช้สร้างสัญญาณ Shift Clock เพื่อกำหนดจังหวะในการรับและส่งข้อมูล การส่งข้อมูลจะเริ่มทันทีที่มีการใช้งาน รีจิสเตอร์ SBUF เป็นรีจิสเตอร์ปลายทาง (destination register) ข้อมูลจะถูกเลื่อนออกไปทีละ 1 บิต โดยเริ่มจากบิตต่ำสุด (LSB) ก่อนและจะมีบิต 0 เลื่อนเข้ามาแทนที่บิตสูงสุดที่ถูกเลื่อนไป สัญญาณที่ขา TXD จะมีความถี่เท่ากับ  $1/12$  ของความถี่ออสซิลเลเตอร์ ถูกส่งออกมาพร้อมๆ กับบิตข้อมูลตั้ง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปเผยแพร่โดยไม่ได้รับอนุญาต  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูป 000 การรับข้อมูลจะเกิดขึ้นได้ต่อเมื่อบิต  $REN = 1$  และ  $RI = 0$  โดยที่ขา TXD จะมีสัญญาณควบคุมการรับข้อมูลออกมาพร้อมกับการรับข้อมูลแต่ละบิต ดังรูปที่ 2.14 อัตราการรับส่งข้อมูลถูกกำหนดไว้ที่  $1/12$  ของความถี่ออสซิลเลเตอร์ที่ใช้

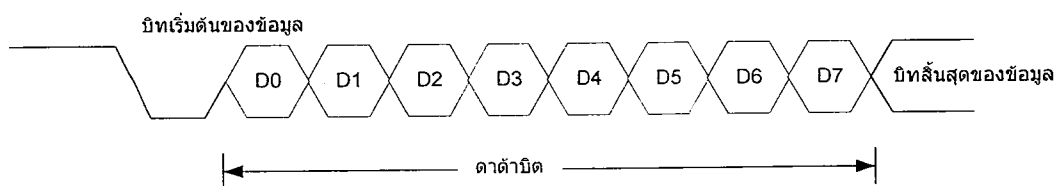


รูปที่ 2.21 สัญญาณการรับและส่งข้อมูลในโหมด 0

#### โหมด 1 (Standard 8 bit UART Mode)

ขา P3.0 (RXD) ถูกใช้เป็นการรับข้อมูล ขา P3.1 (TXD) ถูกใช้เป็นการส่งข้อมูล การรับ-ส่งข้อมูลจะเป็นแบบ 10 บิต ประกอบด้วยบิตเริ่มต้น 1 บิต (เป็น 0 เสมอ) บิตข้อมูล 8 บิต (รับและส่งบิตต่ำสุดก่อน) และบิตสิ้นสุดของข้อมูล 1 บิต (เป็น 1 เสมอ) ดังรูปที่ 2.15 ในการส่งข้อมูลจะเริ่มต้นโดยคำสั่งใดๆ ที่มีการใช้งานรีจิสเตอร์ SBUF เป็นรีจิสเตอร์ปลายทาง บิต TI จะเซตเมื่อข้อมูลถูกเลื่อนออกไปจนครบทั้ง 10 บิตแล้ว การรับข้อมูลจะเริ่มต้นทันทีที่ลอจิกที่ขา RXD เปลี่ยนสถานะจาก 1 เป็น 0 ข้อมูลจะถูกเลื่อนเข้ามาใน Buffer สำหรับการรับข้อมูล เมื่อข้อมูลเข้ามาครบทั้ง 8 บิตแล้ว จึงจะถูกโหลดไปไว้ที่รีจิสเตอร์ SBUF พร้อมกับบิตสิ้นสุดของข้อมูลถูกเก็บไว้ที่บิต RB8 ของรีจิสเตอร์ SCON บิต RI จะเซต เป็นการบอกให้รู้ว่ากระบวนการรับข้อมูลจำนวน 1 ไบท์ได้เสร็จสิ้นลงแล้ว และเริ่มรีเซตตัวเองเพื่อกลับไปรอรับการเปลี่ยนสถานะจาก 1 เป็น 0 ที่ขา RXD อีก ทั้งนี้กระบวนการรับข้อมูลจะเป็นจริงก็ต่อเมื่อ

- บิต  $RI = 0$  และ
- บิต  $SM2 = 0$  หรือถ้า  $SM2 = 1$  บิตสิ้นสุดของข้อมูลจะต้อง = 1 ด้วย



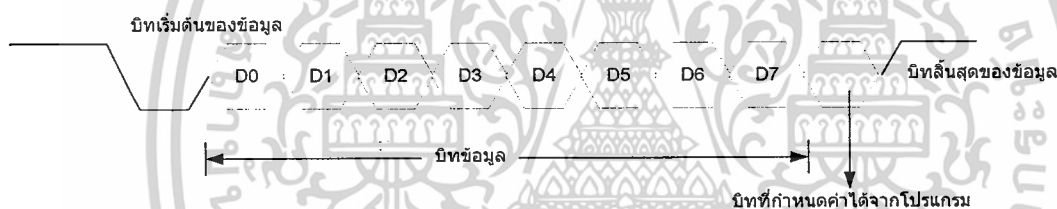
รูปที่ 2.22 การรับและส่งข้อมูลในโหมด 1

เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์ไว้เพื่อการศึกษาเท่านั้น เมื่ออนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### โหมด 2 และ 3 (Multiprocessor)

ขา P3.0 (RXD) ถูกใช้เป็นขาสำหรับรับข้อมูล ขา P3.1 (TXD) ถูกใช้เป็นขาสำหรับส่งข้อมูล การรับ-ส่งข้อมูลจะเป็นแบบ 11 บิต ประกอบด้วยบิตเริ่มต้น 1 บิต (เป็น 0 เสมอ) บิตข้อมูล 8 บิต (รับและส่งบิตต่ำสุดก่อน) บิตที่กำหนดได้ 1 บิต (โปรแกรมเองได้ว่าจะเป็น 1 หรือ 0) และบิตสิ้นสุด 1 บิต (เป็น 1 เสมอ) ดังรูปที่ 2.16

การส่งข้อมูลจะเริ่มขึ้นโดยคำสั่งใดๆ ที่มีการใช้รีจิสเตอร์ SBUF เป็นรีจิสเตอร์ปลายทาง บิต TI จะเซต เมื่อข้อมูลถูกเลื่อนออกไปจน ครบทั้ง 11 บิตแล้ว บิตที่ 9 จะได้จากบิต TB8 ในรีจิสเตอร์ SCON ส่วนในขณะรับข้อมูลบิตที่ 9 จะอยู่ในบิต RB8 ของรีจิสเตอร์ SCON โดยไม่สนใจบิตสิ้นสุดของข้อมูล การรับข้อมูลจะใช้การตรวจสอบการเปลี่ยนแปลงสถานะลอจิกจาก 1 เป็น 0 ที่ขา RXD และมีเงื่อนไขในการรับข้อมูลเช่นเดียวกับ โหมด 1 ในการทำงานโหมด 2 อัตราการรับและส่งข้อมูลสามารถเปลี่ยนแปลงได้



รูปที่ 2.23 การรับและส่งข้อมูลในโหมด 2 และ 3

### รีจิสเตอร์ PCON

เป็นรีจิสเตอร์สำหรับควบคุมการทำงานของ 8051 ซึ่งเป็นรีจิสเตอร์ที่ไม่สามารถอ้างอิงได้ดังมีรายละเอียดดังนี้

SMOD	-	-	-	GF1	GF0	PD	IDL
------	---	---	---	-----	-----	----	-----

รูปที่ 2.24 รีจิสเตอร์ PCON

SMOD เป็นบิตกำหนดอัตราความเร็วของการรับส่งข้อมูลแบบอนุกรมที่ใช้ Timer1 เช่นถ้าเวลาในการรับส่ง เมื่อ SMOD = 1 จะทำให้ความเร็วในการรับส่งแบบอนุกรมเป็น 2 เท่า เมื่อเลือกการใช้งานใน Mode 1, 2, 3 ของการรับส่งแบบอนุกรม

- เป็นบิตที่ไม่ได้ใช้งานซึ่งสงวนไว้สำหรับหน้าที่เพิ่มเติมในอนาคตของ 8051 ดังนั้นผู้ใช้จึงไม่ควรกำหนดในบิตนี้เป็น 1

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

*GF0 – GF1* เป็นบิตอเนกประสงค์ที่ผู้ใช้สามารถกำหนดการทำงานได้

*PD* เป็นบิตสำหรับให้ 8051 เข้าสู่การทำงานใน Power down Mode ซึ่งจะมีเฉพาะเบอร์ที่มีโครงสร้างภายในเป็น CMOS เท่านั้น ซึ่งจะทำงานเมื่อบิตนี้เป็น 1

*IDL* เป็นบิตสำหรับกำหนดให้ 8051 เข้าสู่การทำงานใน Idle Mode ซึ่งจะมีเฉพาะเบอร์ที่มีโครงสร้างภายในเป็น CMOS เท่านั้น ซึ่งจะทำงานเมื่อบิตนี้เป็น 1

### การคำนวณความเร็วการรับและส่งข้อมูลแบบอนุกรม (Generating Baud Rate)

การกำหนดความเร็วการรับส่งข้อมูลแบบอนุกรมสามารถแบ่งออกตาม Mode การทำงาน ดังนี้

#### Mode 0

ความเร็วในการรับส่งข้อมูลแบบอนุกรมใน Mode นี้จะกำหนดอัตราการรับส่งตายตัวเท่ากับ 1/12 ของความถี่ของชุดกำเนิดความอ้างอิงของ 8051 และจะไม่ใช้ Timer/Counter ดังนั้นเพียงกำหนดที่ รีจิสเตอร์ SCON ก็เพียงพอ จะได้

$$\text{Baud Rate} = \text{Osc. Freq} / 12$$

#### Mode 1

ในการกำหนดความเร็วการรับส่งข้อมูลแบบอนุกรมใน Mode 1 นี้จะใช้ Timer 1 เป็นฐานเวลาของการทำงานของ Timer 1 ใน Mode 2 (Auto-Reload) โดยสามารถคำนวณได้ดังนี้

$$\text{Baud Rate} = [K * \text{Osc. Freq}] / [32 * 12 * (256 - (TH1))]$$

K = 1 เมื่อ SMOD ในรีจิสเตอร์ PCON = 0

K = 2 เมื่อ SMOD ในรีจิสเตอร์ PCON = 1

ส่วนมากแล้ว ผู้ใช้จะทราบค่าของ Baud Rate ที่จะส่งได้ค่านั้น จะได้ค่าของ Time 1 สำหรับ Reload ได้เป็น

$$TH1 = 256 - [(K * \text{Osc. Freq}) / (384 * \text{Baud Rate})]$$

#### Mode 2

ความเร็วการรับส่งใน Mode นี้จะเป็นค่าคงที่ซึ่งมี 2 ค่า ขึ้นกับค่า SMOD ในรีจิสเตอร์ PCON ดังนี้

เมื่อ SMOD = 1       $\text{Baud Rate} = 1/32 \text{ Osc. Freq}$

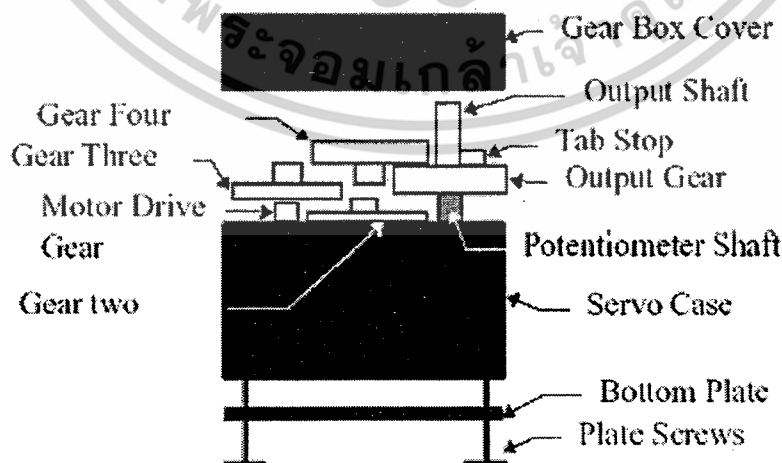
เมื่อ SMOD = 0       $\text{Baud Rate} = 1/64 \text{ Osc. Freq}$

#### Mode 3

การกำหนดความเร็วการรับส่งใน Mode 3 จะคิดเช่นเดียวกับการคิดใน Mode 1 เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## 2.9 เซอร์โวมอเตอร์ (Servo motor)

Servo motor คือ มอเตอร์ไฟฟ้ากระแสตรง (DC motor) ที่ถูกประกอบรวมกับ ชุดเกียร์ และ ส่วนควบคุม ต่างๆ ไว้ในโมดูลเดียวกัน หรือ ภายในกล่องพลาสติกเดียวกัน โดยมอเตอร์ชนิดนี้จะมี สายต่อใช้งานเพียง 3 เส้นเท่านั้น คือ VCC, GND และ สายสัญญาณควบคุม (Control Line) ซึ่งสามารถควบคุมให้มอเตอร์หมุนซ้ายหรือขวาได้ จากสายสัญญาณเพียงเส้นเดียว โดยสัญญาณที่ใช้ควบคุม นี้จะเป็นสัญญาณ พัลส์วิดมอด (PWM) แบบ TTL Level ระดับแรงดันที่จ่ายให้มอเตอร์นี้จะอยู่ใน ช่วงประมาณ 4 ถึง 6 โวลต์ ขึ้นอยู่กับคุณสมบัติของมอเตอร์แต่ละตัว ข้อดีของมอเตอร์ชนิดนี้ ก็คือ จะมีขนาดเล็กน้ำหนักเบา, ให้แรงบิดสูง, กินพลังงานน้อย และ สามารถควบคุม ด้วยแรงดันลอจิกที่เป็น TTL ได้โดยตรงไม่จำเป็นต้องต่อวงจรขับ (Driver) อื่นๆ เพราะมอเตอร์ชนิดนี้จะมีวงจรควบคุมบรรจุไว้ภายในอยู่แล้ว - ซึ่งมอเตอร์ชนิดนี้สามารถควบคุมให้หมุนไปในตำแหน่ง หรือ ทิศทาง องศาที่ต้องการได้ โดยอาศัยสัญญาณความกว้างพัลส์ที่ป้อนให้มอเตอร์ แต่เซอร์โวมอเตอร์นี้จะ หมุนได้แค่เพียงในช่วงประมาณ 180° หรือ ครึ่งรอบเท่านั้น หรือ บางรุ่นอาจหมุนได้ถึง 210° แต่จะ ไม่สามารถหมุนเป็นวงรอบได้เนื่องจาก โครงสร้างภายในจะประกอบด้วย ตัวต้านทานชนิดปรับค่า ได้ (VR) ที่ทำหน้าที่ตรวจสอบตำแหน่งการหมุนของมอเตอร์ และ ตัวต้านทานนี้จะถูกยึดติดกับแกน หมุนของมอเตอร์ ซึ่งจากการที่ตัวต้านทานปรับค่านี้ไม่สามารถหมุนเป็นวงรอบได้ ดังนั้น เซอร์โว มอเตอร์จึงถูกออกแบบให้หมุนได้เพียงแค่ประมาณ 180 องศา หรือ ครึ่งรอบเท่านั้น เพื่อป้องกัน ความเสียหายที่จะเกิดกับตัวต้านทานปรับค่าได้ แต่ถ้าหากเราต้องการให้มอเตอร์หมุนเป็นวงรอบ (360°) นั้นก็สามารถทำได้ โดยจะต้องทำการปรับแต่ง (Modify) คัดแปลงชิ้นส่วนบางอย่างของมอ ตอร์ ซึ่งวิธีการต่าง ๆ จะได้กล่าวไว้ในภายหลัง

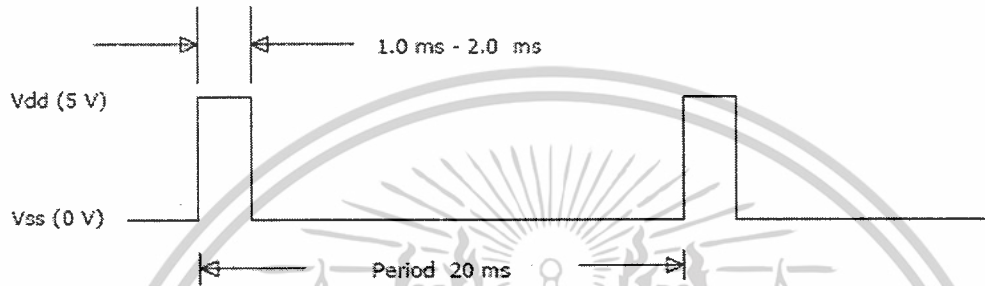


รูปที่ 2.25 ส่วนประกอบต่าง ๆ ของ Servo Motor

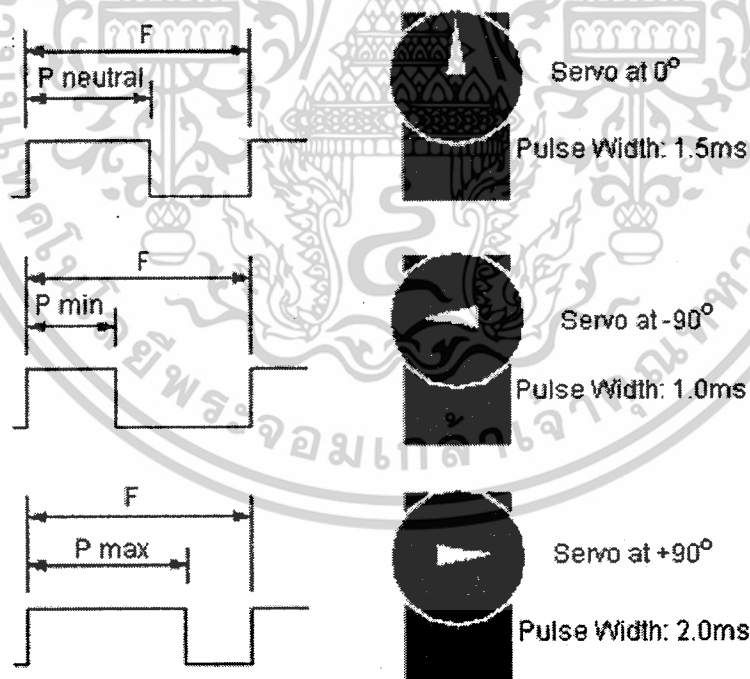
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 2.9.1 หลักการทำงานของเซอร์โวมอเตอร์

การควบคุมการทำงานของเซอร์โวมอเตอร์ทำได้โดยการป้อนสัญญาณความกว้างพัลส์ให้กับมอเตอร์ ซึ่งตำแหน่งและทิศทางการหมุนของมอเตอร์นี้จะขึ้นอยู่กับขนาดของความกว้างของพัลส์นั้นๆ โดยทั่วไปแล้วความกว้างของสัญญาณพัลส์จะมีจุดให้อ้างอิง 3 จุด ดังรูป คือ



รูปที่ 2.26 สัญญาณความกว้างพัลส์



รูปที่ 2.27 การป้อนสัญญาณความกว้างพัลส์ให้กับมอเตอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- สัญญาณความกว้างพัลส์ขนาด 1.5 ms จะควบคุมให้เซอร์โวมอเตอร์หมุนไปอยู่ที่ตำแหน่งมุม 0 องศา หรือ จุดกึ่งกลางของมอเตอร์
- สัญญาณความกว้างพัลส์ขนาด 1 ms จะควบคุมให้เซอร์โวมอเตอร์หมุนไปอยู่ที่ตำแหน่งมุม - 90 องศา หรือในทิศทางทวนเข็มนาฬิกา
- สัญญาณความกว้างพัลส์ขนาด 2 ms จะควบคุมให้เซอร์โวมอเตอร์หมุนไปอยู่ที่ตำแหน่งมุม + 90 องศา หรือในทิศทางตามเข็มนาฬิกา

ส่วนการที่จะควบคุมให้มอเตอร์หมุนเป็นมุมอื่นๆนั้นก็สามารถทำได้โดยการป้อนสัญญาณพัลส์เป็นระดับความกว้างต่าง ๆ โดยอ้างอิงจากจุด ทั้ง 3 จุดที่กล่าวมานี้ ตัวอย่างเช่น ถ้าต้องการให้มอเตอร์หมุนไปที่มุม - 45 องศา เราก็จะต้องป้อนสัญญาณพัลส์ที่มีความกว้าง 1.25 ms เป็นต้น และสัญญาณพัลส์นี้จะต้องจ่ายให้มอเตอร์ทุก ๆ 20 ms (Period) เพื่อรักษาสภาพตำแหน่งของมอเตอร์ไว้ โดยหลักการก็คือ จะอาศัยการเปรียบเทียบช่วงเวลาของความกว้างพัลส์ที่จ่ายให้กับมอเตอร์ทางขาสัญญาณควบคุมกับค่าเวลาของวงจร RC ภายในบอร์ดควบคุมในตัวของมอเตอร์ ซึ่งค่าเวลาของวงจร RC นี้จะมีการเปลี่ยนแปลงตามการหมุนของมอเตอร์ เนื่องจากตัวต้านทานปรับค่าจะถูกยึดติดอยู่กับแกนหมุนของมอเตอร์ ซึ่งการหมุนของมอเตอร์จะทำให้ค่าความต้านทานของตัวต้านทานปรับค่าได้ (VR) เปลี่ยนแปลงไป เป็นผลทำให้ค่าเวลาของวงจร RC เปลี่ยนแปลงตามไปด้วย โดยในขณะที่เราป้อนสัญญาณความกว้างพัลส์ให้กับมอเตอร์ทางขาสัญญาณควบคุม สัญญาณนี้จะถูกนำไปเปรียบเทียบกับค่าเวลาของวงจร RC หากค่าทั้ง 2 ไม่เท่ากันมอเตอร์ก็จะหมุนทำให้ค่าเวลาของวงจร RC เปลี่ยนแปลงจนกระทั่งค่าเวลาความกว้างพัลส์ของ วงจร RC เปลี่ยนแปลงจนเท่ากับสัญญาณพัลส์ทางขาควบคุม (Control line) มอเตอร์จึงจะหยุดหมุน

### 2.9.2 การปรับแต่งของเซอร์โวมอเตอร์

จากคุณสมบัติของ Servo motor ที่ผลิตออกมาจาก โรงงานจะสามารถหมุนได้แค่เพียงประมาณ 180 องศา หรือประมาณครึ่งรอบเท่านั้น หากเราต้องการนำเอา Servo motor ไปใช้งานในลักษณะที่หมุนเป็นวงรอบนั้นก็สมารถทำได้ แต่ก็จะมีข้อเสียการควบคุมในเรื่องของการสั่งให้มอเตอร์หมุนไปในตำแหน่ง หรือ มุมที่ต้องการ ไปด้วย จะทำได้ก็เพียงในเรื่องของการสั่งให้หมุนซ้าย, ขวา และหยุด เท่านั้น โดยการทำให้มอเตอร์สามารถหมุนเป็นวงรอบได้นั้น จะต้องทำการปรับแต่ง หรือแก้ไขโครงสร้างภายในบางส่วนของมอเตอร์ ซึ่ง ได้แก่

- การต่อตัวต้านทานคงที่ 2 ตัวอนุกรม แทนตัวต้านทานปรับค่าได้
- ตัดชิ้นส่วนของแกนเฟืองที่ทำหน้าที่ที่หยุดมอเตอร์ (TAB STOP) ออก
- การตัดแปลงตัวต้านทานปรับค่าได้ (VR) ให้สามารถหมุนได้รอบทิศทาง (360°)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## บทที่ 3

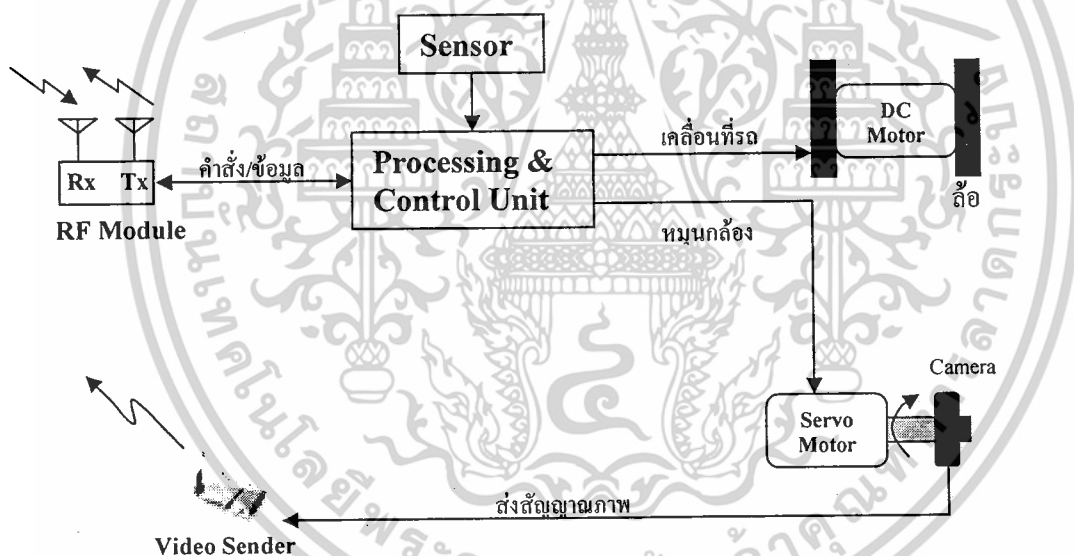
### โครงสร้างและการออกแบบ

การออกแบบสามารถแบ่งออกได้เป็น

#### 3.1 ฮาร์ดแวร์

##### 3.1.1 ส่วนหุ่นยนต์

ส่วนหุ่นยนต์จะประกอบด้วยส่วนต่างๆ คือ ส่วนการควบคุมการเคลื่อนที่ของหุ่นยนต์ ส่วนควบคุมการหมุนของกล้อง ส่วนการรับส่งข้อมูลผ่านคลื่นวิทยุ ส่วนการส่งภาพผ่านคลื่นวิทยุ ส่วนของอุปกรณ์เซนเซอร์ มาทำงานร่วมกัน โดยมีส่วนประมวลผลและสั่งงานเป็นศูนย์กลางดังแสดงในรูปที่ 3.1



รูปที่ 3.1 แสดงส่วนต่างๆของหุ่นยนต์

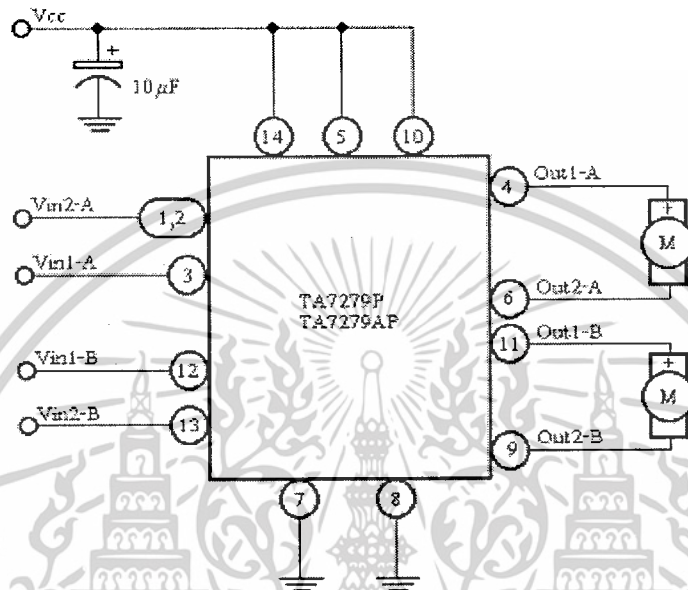
ส่วนควบคุมการเคลื่อนที่ของหุ่นยนต์

ในส่วนนี้จะแบ่งออกเป็น 2 ส่วน คือ

ในการเคลื่อนที่ของตัวหุ่นยนต์จะเคลื่อนที่โดยใช้ดีซีมอเตอร์ 12 โวลต์ 100 rpm แบบมีเฟืองทดช่วยในการขับเคลื่อนซึ่งจะมีอยู่ด้วยกัน 2 ชุด (ซ้าย-ขวา) โดยไมโครคอนโทรลเลอร์จะส่งค่าออกมาทางพอร์ต P2.0 – P2.3 ไปให้กับไอซี TA7279P เพื่อควบคุมดีซีมอเตอร์จำนวน 2 ตัว

ไอซี TA7279P มีตัวถังแบบ HDIP14-P-500-2.54A ลักษณะภายในไอซีจะประกอบด้วยวงจร H-Bridge SW จำนวน 2 ชุด สามารถควบคุมมอเตอร์ได้ 2 ตัว แต่ละชุดมีคุณสมบัติดังนี้ เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- รับแรงดันได้ตั้งแต่ 0 ~ 18 V.
  - เอาท์พุทจ่ายกระแสได้ถึง 1.0A (AVE) , 3.0A (Peak)
- ซึ่งไอซี TA7279P จะมีวงจรการใช้งานเป็นดังรูปที่ 3.2



รูปที่ 3.2 วงจรควบคุมการหมุนของดีซีมอเตอร์

การนำไอซี TA7279P ไปใช้งานนั้น สามารถทำงานด้วยตัวของไอซีเองได้ จากรูปที่ 3.2 เป็นวงจรการทำงาน จะเห็นว่านอกจากไอซีแล้ว วงจรต้องการเพียงตัวเก็บประจุ ค่าประมาณ 10  $\mu\text{F}$  อีกเพียง 1 ตัวเท่านั้น ตัวเก็บประจุนี้ไว้เพื่อทำหน้าที่ในการขยพาสสัญญาณรบกวนหากไม่ใส่ตัวเก็บประจุอาจส่งผลให้การทำงานผิดพลาดเมื่อต่อร่วมกับวงจรไมโครคอนโทรลเลอร์

#### การควบคุมดีซีมอเตอร์

เนื่องจากวงจรการควบคุมทิศทางการหมุนของดีซีมอเตอร์ได้ใช้ไอซี TA7279P ซึ่งเป็นไอซีสำเร็จรูป เพราะฉะนั้นเมื่อต่อวงจรใช้งานตามวงจรรูปที่ 3.2 การควบคุมนั้นสามารถทำได้โดยการป้อนสัญญาณลอจิก "1" หรือสัญญาณลอจิก "0" การป้อนจะป้อนสัญญาณลักษณะเป็นความกว้างของลูกคลื่น (Pulse Width Mod) เพื่อที่จะให้สามารถควบคุมระยะการหมุนของมอเตอร์ได้เข้าที่ด้านอินพุต 1, 2 เท่านั้น วงจรก็สามารถทำงานได้ ส่วนมอเตอร์จะหมุน Forward หรือ Reward หรือหยุดหมุนขึ้นอยู่กับสัญญาณควบคุมที่ป้อนให้แก่ตัวของไอซี TA7279P ดังแสดงรายละเอียดดังตารางที่ 3.1

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 3.1 การป้อนสัญญาณควบคุมไอซี TA7279P

IN		OUT		MODE
1	2	1	2	
0	0	High – Impedance		Stop
0	1	L	H	CW/CWW
1	0	H	L	CCW/CW
1	1	H	H	Break

จากตารางที่ 3.1 จะมีการทำงานที่น่าสนใจอยู่ 2 โหมด คือ Stop และ Break ซึ่ง 2 โหมด นี้มีการทำงานต่างกันอยู่เล็กน้อยดังนี้

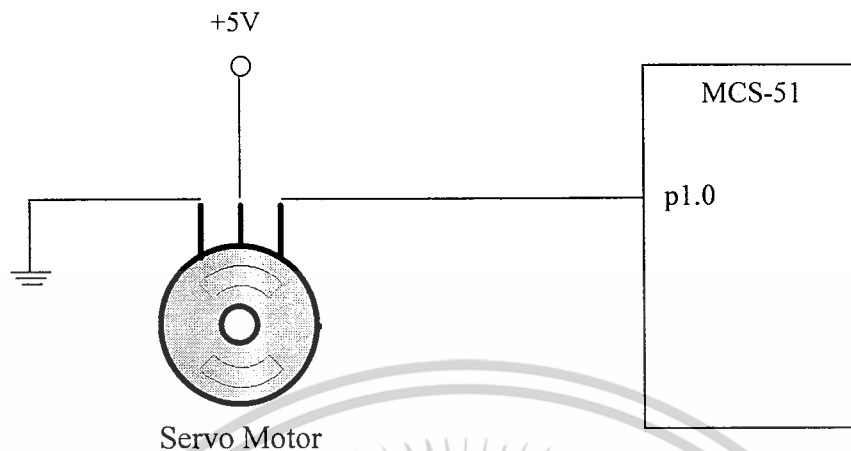
**Stop:** จะทำงานเปรียบเสมือนการปลดมอเตอร์ออกจากแหล่งจ่าย คือถ้าป้อนสัญญาณให้แก่มอเตอร์แล้ว มอเตอร์จะค่อย ๆ หมุนช้าลงเรื่อย ๆ จนหยุดนิ่งสนิท

**Break:** จะทำงานเปรียบเสมือน การล๊อคมอเตอร์เอาไว้ คือถ้าป้อนสัญญาณให้แก่มอเตอร์แล้ว มอเตอร์จะหยุดในทันทีทันใด

#### ส่วนควบคุมการหมุนของกล่อง

ส่วนของการควบคุมทิศทางการหมุนของกล่องจะใช้เซอร์โวมอเตอร์ช่วยในการหมุนกล่อง โดยตัวเซอร์โวมอเตอร์สามารถหมุนกล่องไปในทิศทางตามเข็มนาฬิกา  $90^\circ$  และทวนเข็มนาฬิกา  $90^\circ$  ในการเลือกทิศทางการหมุนของกล่องจะทำได้โดยการป้อนสัญญาณความกว้างพัลส์ให้กับเซอร์โวมอเตอร์ ซึ่งตำแหน่งและทิศทางการหมุนของมอเตอร์นี้จะขึ้นอยู่กับขนาดความกว้างของพัลส์นั้น

ในการควบคุมมอเตอร์จะทำการต่อสายสัญญาณของมอเตอร์เข้ากับพอร์ตของไมโครคอนโทรลเลอร์เพียง 1 บิต ดังรูปที่ 3.3 แล้วให้ไมโครคอนโทรลเลอร์ส่งสัญญาณพัลส์ออกมาเพื่อให้มอเตอร์หมุนไปยังตำแหน่งที่ต้องการ



รูปที่ 3.3 วงจรการใช้งานเซอร์โวมอเตอร์

#### ส่วนรับส่งข้อมูลผ่านคลื่นวิทยุ

ในการรับส่งข้อมูลระหว่างหุ่นยนต์กับเซิร์ฟเวอร์จะใช้วิธีส่งผ่านทางคลื่นวิทยุ โดยจะมีชุดส่งและรับสัญญาณคลื่นวิทยุ (RF-Module) โดยจะใช้ไอซีเบอร์ TLP434A และ RLP434A ตามลำดับ ซึ่งใช้การมอดูเลตสัญญาณแบบ ASK และใช้ย่านความถี่ 434 MHz อยู่ 2 ชุดด้วยกัน ชุดแรกจะอยู่ที่ฝั่งของคอมพิวเตอร์เซิร์ฟเวอร์ ส่วนอีกชุดหนึ่งจะอยู่ที่ตัวหุ่นยนต์ โดยตัวหุ่นยนต์สามารถรับและส่งข้อมูลกลับไปยังฝั่งคอมพิวเตอร์เซิร์ฟเวอร์ได้

การรับและส่งข้อมูลระหว่างไมโครคอนโทรลเลอร์กับชุดรับส่งสัญญาณคลื่นวิทยุจะใช้รูปแบบในการรับและส่งข้อมูลอนุกรมแบบอะซิงโครนัส โดยสัญญาณอินพุตและเอาต์พุตจะต่ออยู่กับพอร์ท TxD/P3.1 และ RxD/P3.0 ตามลำดับ แต่เนื่องจากการส่งผ่านคลื่นวิทยุจึงต้องมีรูปแบบของเฟรมในการส่งเพื่อประสิทธิภาพในการรับส่ง โดยมีรูปแบบเฟรมดังนี้

5 ไบต์	1 ไบต์	1 ไบต์	1 ไบต์	2 ไบต์
0101010101010101 (preamble)	ไบต์ 'j'	ไบต์ 'U'	ลำดับเฟรม	ไบต์ข้อมูล

#### ลำดับขั้นตอนในการส่งข้อมูล

- เริ่มต้นด้วยการส่ง Preamble เป็นข้อมูล 0 และ 1 สลับกันเพื่อให้ทางด้านรับได้รู้ว่ากำลังจะมีการส่งข้อมูลซึ่งจะส่งทีละ 1 ไบต์
- ตามด้วยข้อมูลไบต์ j และ U เพื่อใช้ในการตรวจสอบว่าข้อมูลไบต์ต่อไปเป็นไบต์ข้อมูล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- ไบต์ลำดับเฟรม (frame sequence) ใช้บอกว่าเป็นข้อมูลชุดใด
- ชุดท้ายเป็นไบต์ข้อมูล โดยสามารถส่งต่อเนื่องหลายๆไบต์ต่อกันได้

#### ลำดับขั้นตอนในการรับข้อมูล

- ทำการตรวจสอบข้อมูลเมื่อเกิดการอินเทอร์รัปต์ที่พอร์ทอนุกรม โดยจะทำการรับเข้ามาครั้งละ 1 ไบต์
- ถ้าข้อมูลที่ได้รับเข้ามาเป็นไบต์  $j$  ก็จะทำกรรับข้อมูลไบต์ต่อไป ถ้าข้อมูลที่ได้รับเป็นไบต์  $U$  ก็จะทำกรรับข้อมูลไบต์ต่อไปซึ่งเป็นไบต์ Sequence ตรวจสอบว่าเป็นข้อมูลชุดใด ถ้าถูกแสดงว่าเป็นข้อมูลที่ต้องการ ถ้าไม่ใช่ก็จะกลับไปรอเกิดอินเทอร์รัปต์ใหม่

#### ส่วนส่งสัญญาณภาพผ่านคลื่นวิทยุ

ในการรับข้อมูลภาพจะใช้ชุดรับส่งข้อมูลภาพไร้สายรุ่น CCTV - 1500RS ซึ่งใช้ย่านความถี่ UHF (Ultra High Frequency) โดยที่ภากรับจะนำสัญญาณภาพวิดีโออนาลอกที่ได้ต่อเข้ากับพอร์ตวิดีโออินพุตของคอมพิวเตอร์เซิร์ฟเวอร์และเขียนโปรแกรมเพื่อรับสัญญาณภาพเพื่อนำไปบันทึกหรือประมวลผลต่อไป

#### ส่วนเซนเซอร์

- เซนเซอร์อุณหภูมิ

ส่วนของเซนเซอร์อุณหภูมินี้ จะใช้ในการตรวจสอบความผิดปกติของอุณหภูมิภายในบ้าน โดยจะใช้ไอซี DS1820 ในการตรวจจับอุณหภูมิ ซึ่งสามารถวัดค่าอุณหภูมิได้ตั้งแต่ -55 ถึง 125 องศาเซลเซียส

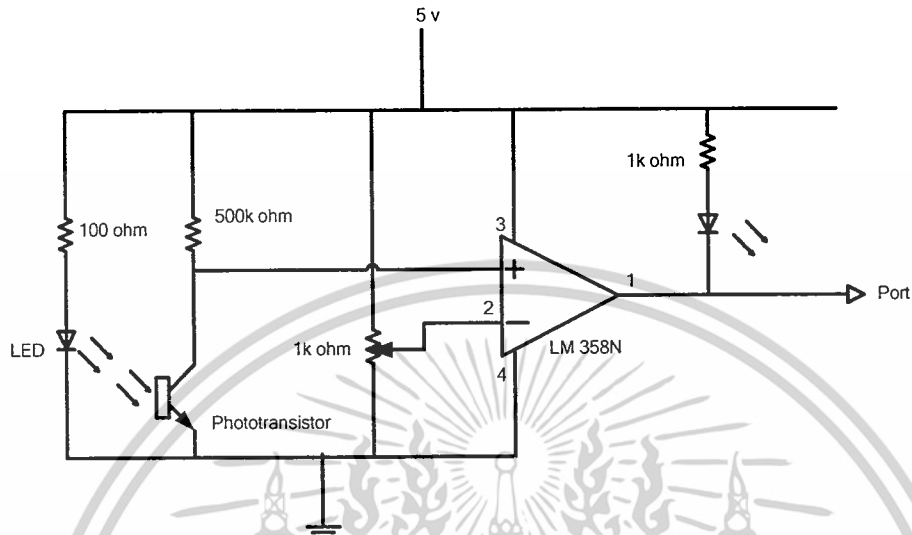
การเชื่อมต่อกับไมโครคอนโทรลเลอร์ที่ตัวหุ่นยนต์จะใช้พอร์ทเพียง 1 พอร์ทสำหรับการเชื่อมต่อกับ DS1820 โดยต้องมีตัวต้านทานค่า 4.7K โอห์ม ต่อพูลอัพกับไฟเลี้ยง +5V จากนั้นจึงทำการเขียนโปรแกรมติดต่อตามมาตรฐานระบบบัสหนึ่งสาย

- เซนเซอร์ที่พื้น

ส่วนของเซนเซอร์ที่พื้นจะใช้เพื่อให้หุ่นยนต์สามารถหยุดในจุดที่ต้องการ ในห้องต่างๆภายในบ้าน เพื่อใช้ในการทำงานในโหมดที่ให้หุ่นยนต์เดินไปที่ห้องที่ต้องการ โดยมีหลักการทำงานคือใช้หลอด LED ในการปล่อยแสงและใช้โฟโตทรานซิสเตอร์ ในการรับแสงสะท้อนจากพื้น ในสถานะปกติที่ยังไม่เจอจุดหยุดโฟโตทรานซิสเตอร์จะสามารถรับแสงสะท้อนจากพื้นได้ เมื่อเจอจุดที่ต้องการให้หุ่นยนต์หยุดจะใช้แถบสีดำติดไว้ที่พื้นเพื่อที่จะให้โฟโตทรานซิสเตอร์ไม่

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สามารถรับแสงสะท้อนได้เพราะว่าแถบสีดำจะมีการสะท้อนของแสงน้อยที่สุด โดยวงจรที่ใช้เป็น  
 ดังรูปที่ 3.4



รูปที่ 3.4 แสดงวงจรเปรียบเทียบแรงดันที่ใช้

เนื่องจากออปแอมป์ในอุดมคตินั้นจะมีค่าอัตราขยายเป็นอนันต์ แต่ในทางปฏิบัติอัตราขยายอาจมีค่าสูงสุดเพียง 10,000 เท่าหรือ 1,000,000 เท่า เท่านั้น ซึ่งเรียกว่าอัตราขยายขณะเปิดคูล (Av) ในขณะที่เกิดความแตกต่างของแรงดันเพียงเล็กน้อยระหว่างอินพุททั้งสองข้างของออปแอมป์ด้วย แต่เอาท์พุทจะไม่สามารถมีค่าสูงสุดเท่ากับแรงดันจากแหล่งจ่ายไฟเลี้ยงได้จริง เนื่องจากเกิดแรงดันตกคร่อมภายใน ดังนั้นแรงดันเอาท์พุทสูงสุดอาจมีค่าประมาณ 90% ของแรงดันจากแหล่งจ่ายไฟเลี้ยงเท่านั้น จากคุณสมบัติข้างต้น เราสามารถนำออปแอมป์ในขณะเปิดคูลไปใช้เป็นคอมพาราเตอร์ (Comparator) หรือวงจรเปรียบเทียบแรงดันได้ โดยเอาท์พุทจะเปลี่ยนทันทีเมื่อมีความแตกต่างของแรงดันเกิดขึ้นระหว่างขั้วอินพุททั้งสองของออปแอมป์

ถ้าขั้วลบ มีศักดาเป็นบวกสูงกว่าขั้วบวก เอาท์พุทเป็น ศักดาต่ำ

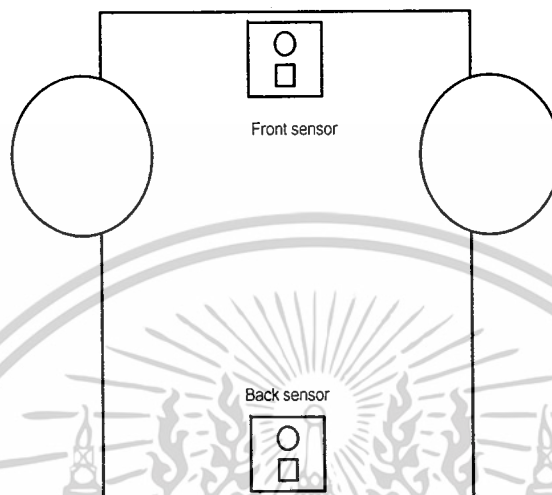
ถ้าขั้วลบ มีศักดาเป็นบวกต่ำกว่าขั้วบวก เอาท์พุทเป็น ศักดาสูง

ในส่วนนี้ได้ใช้ไอซีเบอร์ LM358 เพื่อทำการเปรียบเทียบแรงดันที่ได้รับจากอินพุท เพื่อทำการตรวจเช็คค่าเซนเซอร์อยู่ในสถานะสะท้อนเส้นสีอะไร

หลักการสะท้อนของแสงคือเมื่อแสงตกกระทบบเส้นสีขาวแล้ว จะเกิดการสะท้อนกลับของแสงและเมื่อแสงตกกระทบบเส้นสีดำ แสงจะถูกดูดกลืน โดยเมื่อแสงสะท้อนเส้นสีขาว แรงดันที่ขาอินเวอร์ตติ้ง (Inverting) ของออปแอมป์จะเป็น 0 ทำให้เอาท์พุทที่ออกมาเป็น บวก คือ LED จะดับ และเมื่อแสงตกกระทบบเส้นสีดำ แสงจะถูกดูดกลืนไว้ และไม่มีการสะท้อนกลับของ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

แสง ทำให้ที่ขาอินเวอร์ตของออปแอมป์จะเป็น 1 ทำให้เอาต์พุตที่ออกมาเป็น ลบ คือ LED จะติด โดยมีการวางตำแหน่งของเซนเซอร์ดังรูปที่ 3.5

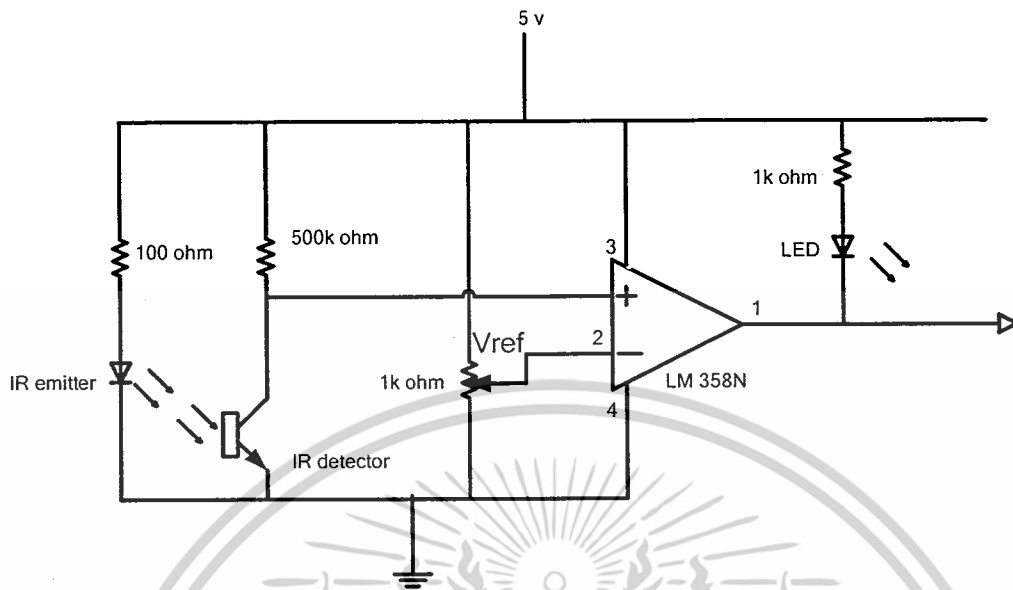


รูปที่ 3.5 แสดงการวางตำแหน่งของเซนเซอร์พิน

โดยเซนเซอร์ทั้งสองชุดจะต่อเอาต์พุตจากส่วนตรวจสอบจุดหยุดไปเข้า P1.1 และ P1.2 ของ MCS51

#### - เซนเซอร์แสง (Infrared Sensor)

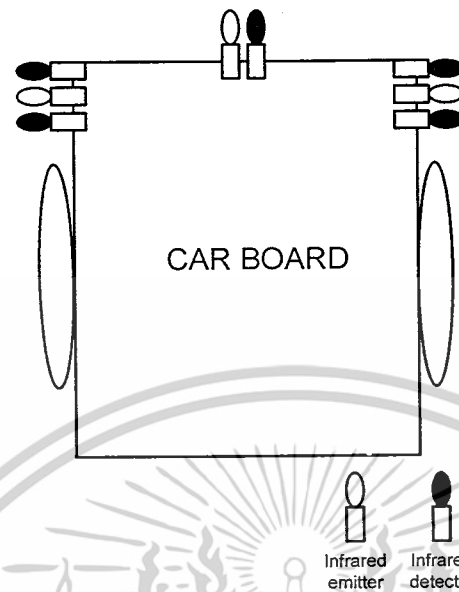
เป็นส่วนวงจรเซนเซอร์ที่ใช้ในการเดินของตัวหุ่นยนต์ โดยใช้วิธีการเดินไปตาม กำแพงบ้าน ซึ่งจะใช้วงจรเซนเซอร์อินฟราเรดในการตรวจสอบสถานะระหว่างตัวหุ่นยนต์กับ กำแพงว่าอยู่ในสถานะใด โดยที่ตัวหุ่นยนต์จะติดเซนเซอร์อินฟราเรดไว้ 3 ชุด ที่ด้านหน้าและ ด้านข้างทั้ง 2 ด้านของตัวหุ่นยนต์ โดยมีวงจรเป็นดังรูปที่ 3.6



รูปที่ 3.6 วงจรเซนเซอร์อินฟราเรด

จากวงจรจะเป็นวงจรเดียวกันกับวงจรเซนเซอร์ที่เปลี่ยนจากหลอด LED กับโฟโตทรานซิสเตอร์เป็นตัวส่งอินฟราเรดกับตัวรับแสงอินฟราเรด โดยที่เซนเซอร์ด้านข้างของตัวหุ่นยนต์แต่ละชุดจะใช้ตัวรับแสงอินฟราเรด 2 ตัวต่อตัวส่ง 1 ตัว เพื่อที่จะให้สามารถตรวจสอบระยะระหว่างตัวหุ่นยนต์กับกำแพงได้ และจะได้รู้สถานะว่าเป็นสถานะใด โดยเราจะปรับค่า  $V_{ref}$  ในวงจรให้ตัวรับแสงอินฟราเรดแต่ละตัวมีระยะการรับแสงได้ที่ต่างกัน ส่วนเซนเซอร์อินฟราเรดตัวที่อยู่ด้านหน้าของตัวหุ่นยนต์จะใช้ตัวรับแสงอินฟราเรดแค่ 1 ตัว เพราะจะตรวจสอบแค่ว่าตัวหุ่นยนต์พบกำแพงที่อยู่ด้านหน้าหุ่นยนต์หรือไม่ ถ้าปรับค่า  $R$  ปรับค่าได้ให้มีค่ามากทำให้แรงดันไฟที่ขาลบของ LM 358N มีค่ามาก เมื่อตัวรับแสงอินฟราเรดรับแสงที่สะท้อนกลับมาได้ ทำให้แรงดันไฟที่ขาบวกของ LM 358N มีค่าลดลง เมื่อแรงดันที่ขาบวกน้อยกว่าที่ขาลบทำให้เอาท์พุทที่ออกทางขา 1 จะเป็นลอจิก 0 หลอด LED จะติด (มีกระแสไหลผ่าน) แสดงสถานะว่าตัวรับ รับแสงสะท้อนได้

ถ้าตัวรับอินฟราเรดรับแสงสะท้อนไม่ได้ แรงดันไฟที่ขาบวกของ LM 358N จะมีค่ามาก (ประมาณ 5 โวลต์) ซึ่งจะมีค่ามากกว่าแรงดันไฟที่ขาลบ ทำให้เอาท์พุทที่ได้เป็นสัญญาณลอจิก 1 หลอด LED จะดับแสดงสถานะว่าตัวรับ รับแสงสะท้อนไม่ได้ โดยระยะที่ต้องการในการให้หุ่นยนต์ตามกำแพงอยู่ที่ประมาณ 10 ซม. ตำแหน่งของเซนเซอร์เป็นดังรูปที่ 3.7



รูปที่ 3.7 ตำแหน่งเซนเซอร์อินฟราเรด

โดยที่สถานะต่างๆ ของเซนเซอร์ที่เป็นไปได้ขณะเคลื่อนที่ตามผนังห้องเป็นดัง

ตารางที่ 3.2

ตารางที่ 3.2 การรับแสงสะท้อนของตัวรับแสงอินฟราเรด

L1/R1	L2/R2	ระยะระหว่างหุ่นกับ กำแพง	สถานะ
รับแสงสะท้อนไม่ได้	รับแสงสะท้อนไม่ได้	มากกว่า 10 ซม.	สถานะที่1
รับแสงสะท้อนได้	รับแสงสะท้อนไม่ได้	ประมาณ 10 ซม.	สถานะที่2
รับแสงสะท้อนได้	รับแสงสะท้อนได้	น้อยกว่า 10 ซม.	สถานะที่3

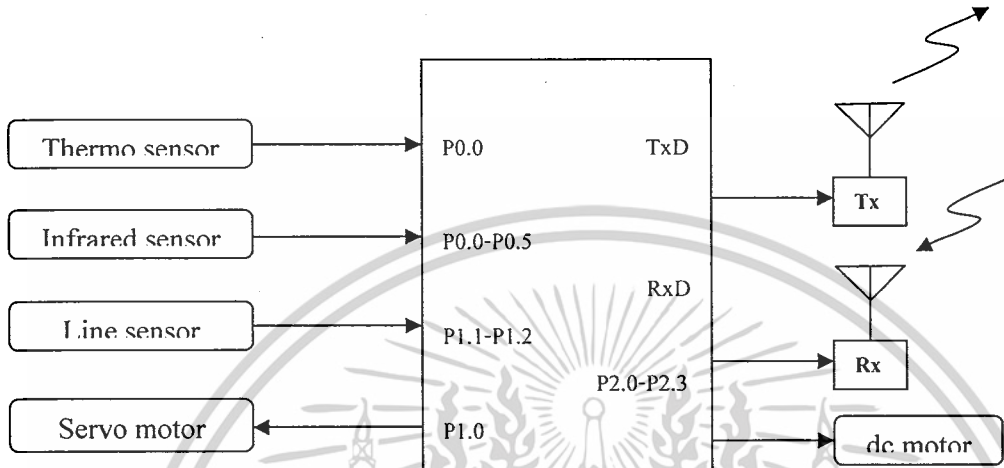
(L1 = left near detector L2 = left far detector F = front detector R1 = right near detector R2 = right far detector)

### ส่วนประมวลผลและสั่งงาน

ในส่วนนี้จะใช้ไมโครคอนโทรลเลอร์ เป็นตัวประมวลผลและควบคุมการทำงานของอุปกรณ์ต่างๆ บนตัวรถ เมื่อภาครับของชุดรับส่งสื่อสารไร้สาย (RF-Module) ที่ตัวหุ่นยนต์ได้รับสัญญาณควบคุมมาจากภาคส่งของส่วนควบคุมหุ่นยนต์ ก็จะส่งข้อมูลที่ได้นี้ให้กับไมโครคอนโทรลเลอร์ ซึ่งเป็นการส่งข้อมูลแบบอนุกรมเข้าที่ขาพอร์ท RxD เมื่อไมโครคอนโทรลเลอร์ได้รับข้อมูลมาก็จะนำไปใช้ในการควบคุมอุปกรณ์ต่างๆ ของหุ่นยนต์ได้แก่ การควบคุมการเคลื่อนที่ของ

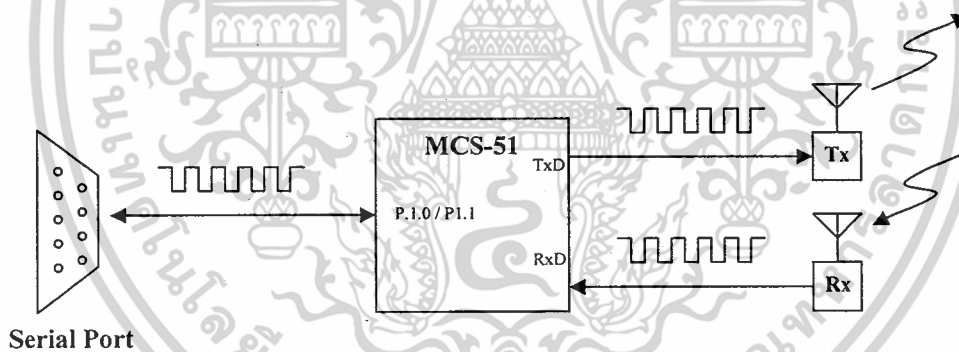
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หุ่นยนต์ (ควบคุมดีซีมอเตอร์) และการควบคุมทิศทางการหมุนของล้อ (ควบคุมเซอร์โวมอเตอร์) และจะรับข้อมูลจากเซนเซอร์ต่างๆ ได้แก่ เซนเซอร์อินฟราเรด, เซนเซอร์อุณหภูมิ, เซนเซอร์พื้น เข้ามาทางพอร์ตต่างๆดังรูปที่ 3.8



รูปที่ 3.8 ข้อมูลอินพุต/เอาต์พุตที่ไมโครคอนโทรลเลอร์ของตัวหุ่นยนต์

### 3.1.2 ส่วนรับส่งข้อมูลระหว่างเซิร์ฟเวอร์กับหุ่นยนต์



รูปที่ 3.9 แสดงโครงสร้างของส่วนรับส่งข้อมูลระหว่างเซิร์ฟเวอร์ กับหุ่นยนต์

#### ส่วนรับส่งข้อมูลกับเซิร์ฟเวอร์

ในส่วนนี้จะเป็นการติดต่อรับส่งข้อมูลกันระหว่างคอมพิวเตอร์เซิร์ฟเวอร์กับไมโครคอนโทรลเลอร์ โดยจะติดต่อผ่านทางพอร์ตอนุกรมของตัวคอมพิวเตอร์เซิร์ฟเวอร์กับพอร์ต P1.0 และ P1.1 ของไมโครคอนโทรลเลอร์ดังรูปที่ 3.9 การรับส่งข้อมูลระหว่างไมโครคอนโทรลเลอร์กับพอร์ตอนุกรม จะใช้รูปแบบการรับส่งข้อมูลอนุกรมแบบอะซิงโครนัส โดยให้หนึ่งบิตเริ่มและหนึ่งบิตจบโดยสัญญาณอินพุตและเอาต์พุตจะต่ออยู่กับพอร์ต P1.0 และ P1.1 ของไมโครคอนโทรลเลอร์ตามลำดับ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### ส่วนประมวลผล

ในส่วนนี้จะใช้ไมโครคอนโทรลเลอร์ทำหน้าที่รับคำสั่งที่ได้รับมาจากคอมพิวเตอร์เซิร์ฟเวอร์แล้วทำการประมวลผล จากนั้นก็จะทำการส่งข้อมูลออกที่พอร์ต TxD ไปที่ชุดส่งสัญญาณคลื่นวิทยุ เพื่อส่งข้อมูลไปยังตัวรับที่อยู่ที่ตัวหุ่นยนต์ และรอรับข้อมูลที่ถูกส่งมาจากตัวหุ่นยนต์ผ่านชุดรับสัญญาณคลื่นวิทยุเข้าที่พอร์ต RxD ของไมโครคอนโทรลเลอร์ แล้วทำการส่งข้อมูลกลับไปยังคอมพิวเตอร์เซิร์ฟเวอร์ผ่านพอร์ตอนุกรม

#### ส่วนรับส่งข้อมูลผ่านคลื่นวิทยุ

ในส่วนนี้จะมีลักษณะเช่นเดียวกับส่วนรับส่งข้อมูลผ่านคลื่นวิทยุของส่วนหุ่นยนต์

## 3.2 ซอฟแวร์

### 3.2.1 โปรแกรมควบคุมหุ่นยนต์

การทำงานของหุ่นยนต์สามารถแบ่งออกได้เป็น 2 แบบคือ แบบบังคับด้วยมือและแบบอัตโนมัติ

แบบบังคับด้วยมือ ได้แก่

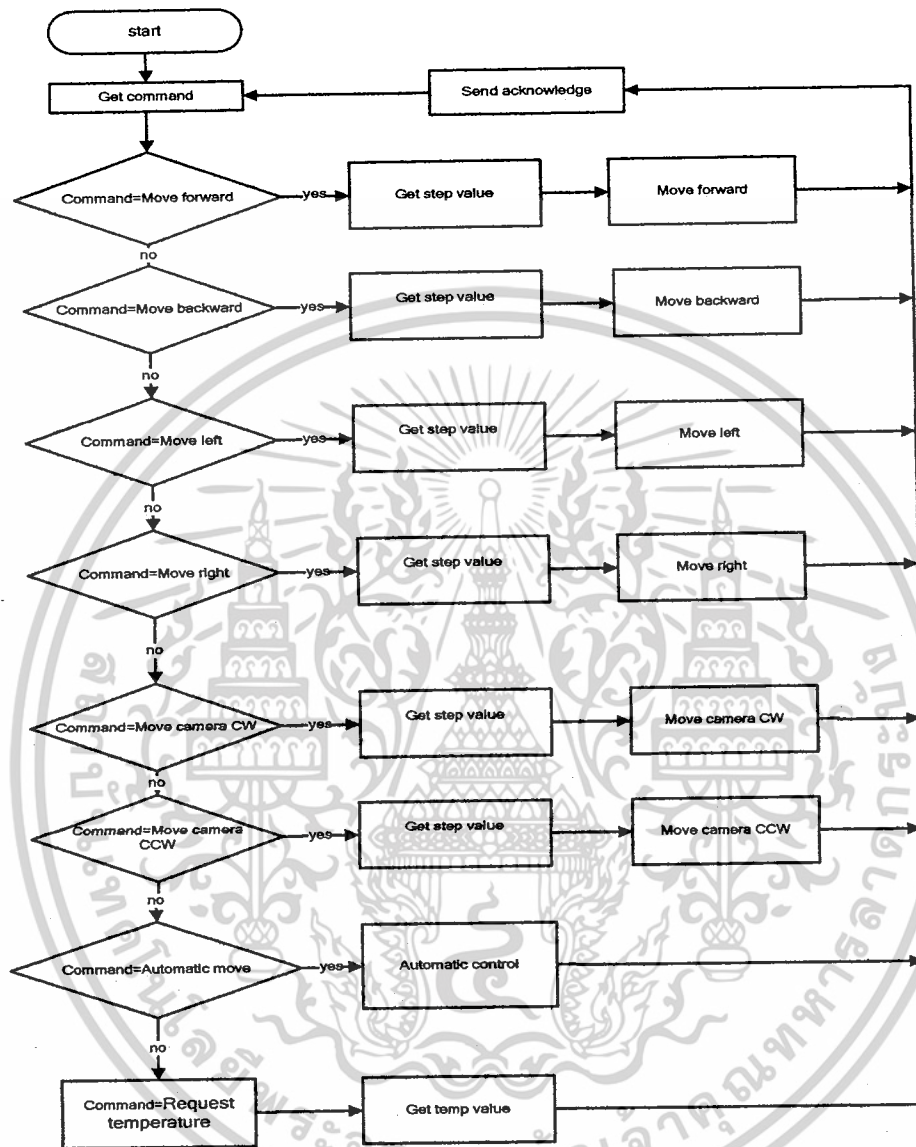
- การเคลื่อนที่ (เดินหน้า ถอยหลัง หมุนซ้าย หมุนขวา) ตามระยะที่กำหนด
- การหมุนกลิ้งไปในทิศทางตามเข็มนาฬิกาและทวนเข็มนาฬิกาโดยมีความกว้างทั้งหมด  $180^{\circ}$
- การวัดอุณหภูมิห้อง
- การรับส่งข้อมูลผ่านคลื่นวิทยุ

การทำงานแบบอัตโนมัติ ได้แก่

- การเคลื่อนที่ไปตามแนวกำแพงและหยุดเมื่อเจอจุดหยุดและส่งข้อมูลกลับไปยังเซิร์ฟเวอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

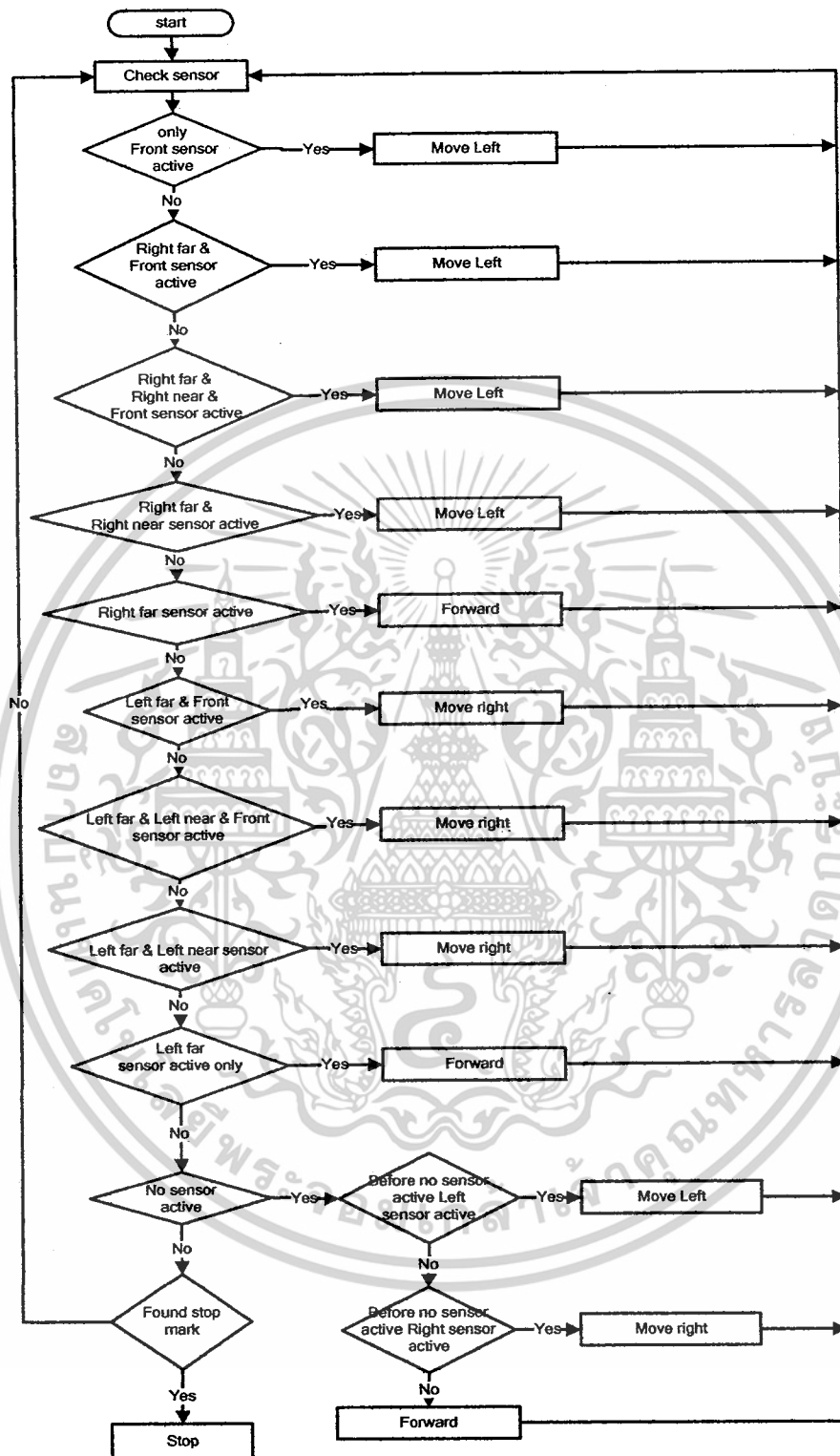
โดยที่ตัวหุ่นยนต์จะมี Flowchart การทำงานในส่วนต่างๆ เป็นดังรูปที่ 3.10



รูปที่ 3.10 โพล์ชาร์ตการทำงานขั้นพื้นฐานของหุ่นยนต์

ในโมดคนี่เมื่อหุ่นยนต์ได้รับคำสั่งก็จะมาตรวจสอบว่าเป็นคำสั่งประเภทใด เช่นการสั่งให้ หุ่นเลี้ยวซ้าย, เลี้ยวขวา, เดินหน้า, ถอยหลัง, การหมุนกล้อง, การขอข้อมูลอุณหภูมิ แล้วจะไป ทำงานตามฟังก์ชันที่ได้รับคำสั่งมา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



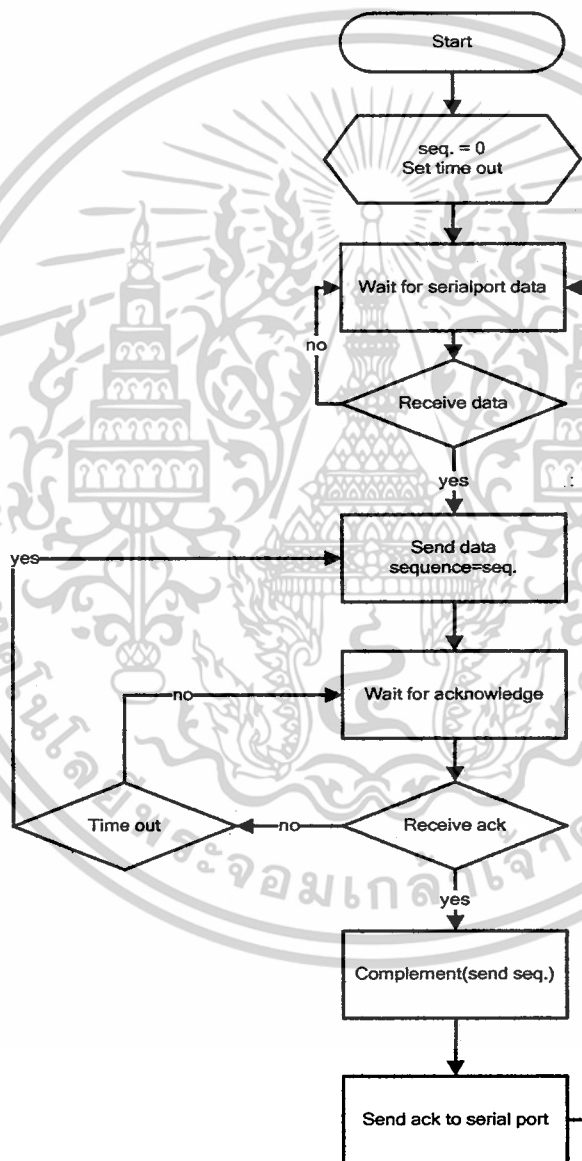
รูปที่ 3.11 โฟลว์ชาร์ตการทำงานแบบอัตโนมัติของหุ่นยนต์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การทำงานคังโพล์ชาร์ตรูปที่ 3.11 ในโหมคนี้เมื่อหุ่นยนต์ได้รับคำสั่งมาแล้ว ก็จะไปทำงานตามอัลกอริทึมที่ใช้ในการเคลื่อนที่ไปตามแนวกำแพง โดยการตรวจสอบสถานะของเซนเซอร์อินฟราเรดทั้ง 3 ซุคว่าอยู่ในสถานะใด และหยุดเมื่อเจอจุดที่ให้หยุดและส่งข้อมูลกลับไปยังเครื่องเซิร์ฟเวอร์

### 3.2.2 โปรแกรมควบคุมการรับส่งข้อมูลระหว่างเซิร์ฟเวอร์กับหุ่นยนต์

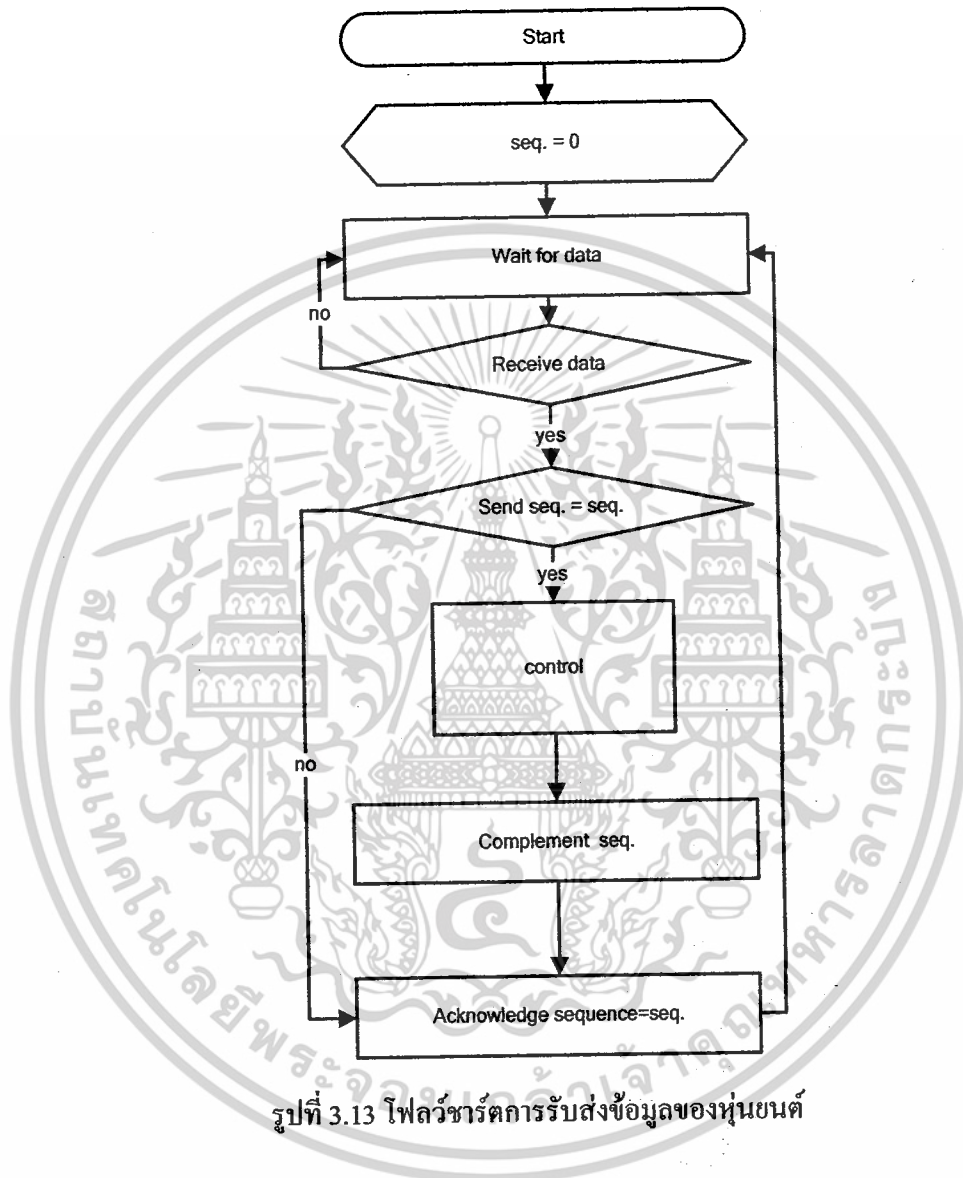
มีรูปแบบการรับส่งข้อมูลเป็นแบบ ARQ มีโพล์ชาร์ตการทำงานเป็นดังนี้



รูปที่ 3.12 โพล์ชาร์ตการส่งข้อมูลของภาคส่งข้อมูลควบคุมรถ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โดยที่เซิร์ฟเวอร์จะมีการส่งลำดับชุดข้อมูลไปด้วยเพื่อที่จะให้ตรวจสอบได้ว่าหุ่นยนต์ได้รับคำสั่งแล้วหรือยัง ถ้ายังไม่ได้รับข้อมูลนานเกินกว่าค่าที่ตั้งไว้ (Time out) ก็จะส่งข้อมูลชุดนั้นไปใหม่อีกครั้ง การส่งข้อมูลจะสมบูรณ์ได้ต่อเมื่อภาคส่งได้รับการตอบกลับ (Acknowledge) กลับมา

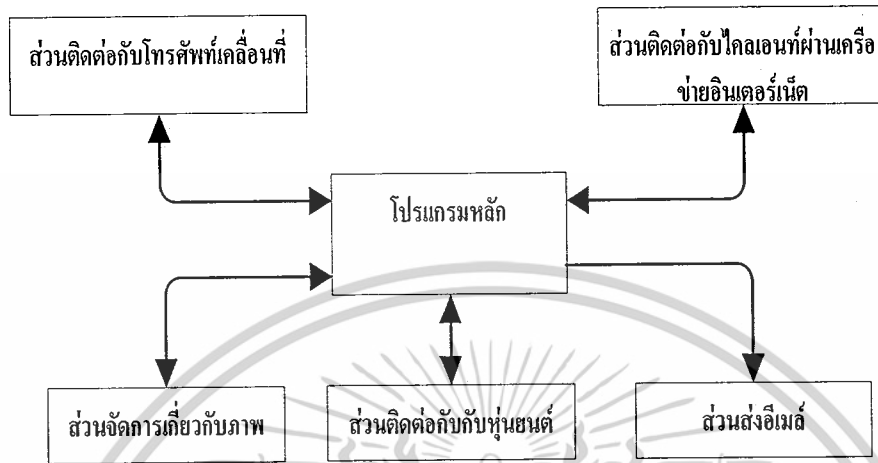


รูปที่ 3.13 โพลีชาร์ตการรับส่งข้อมูลของหุ่นยนต์

โดยที่ตัวหุ่นยนต์จะรอรับข้อมูลที่ส่งมาจากเซิร์ฟเวอร์ โดยจะตรวจสอบจากค่าลำดับของเฟรม (seq) ว่าตรงกับค่า seq ที่ส่งมาจากเซิร์ฟเวอร์หรือไม่ ถ้าตรงกันก็จะทำตามคำสั่งที่ได้รับมา เมื่อทำเสร็จงานแล้วก็จะกลับค่า seq แล้วทำการตอบกลับ (acknowledge)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 3.2.3 โปรแกรมควบคุมที่เซิร์ฟเวอร์



รูปที่ 3.14 ภาพแสดงโครงสร้างของโปรแกรมควบคุมที่เซิร์ฟเวอร์

โปรแกรมในส่วนนี้จะทำหน้าที่เป็นศูนย์กลางในการเชื่อมต่อโปรแกรมส่วนต่างๆ ทั้งหมดเข้าด้วยกัน รวมถึงการทำหน้าที่ในการติดต่อกับโทรศัพท์เคลื่อนที่และการเชื่อมต่อกับไคลเอนต์ผ่านเครือข่ายอินเทอร์เน็ตโดยใช้เทคโนโลยีจาวาแอปพลิเคชันในการสร้างโปรแกรมดังรูปที่ 3.14 โดยจะประกอบไปด้วยส่วนหลักๆ ดังนี้

#### ส่วนโปรแกรมหลัก

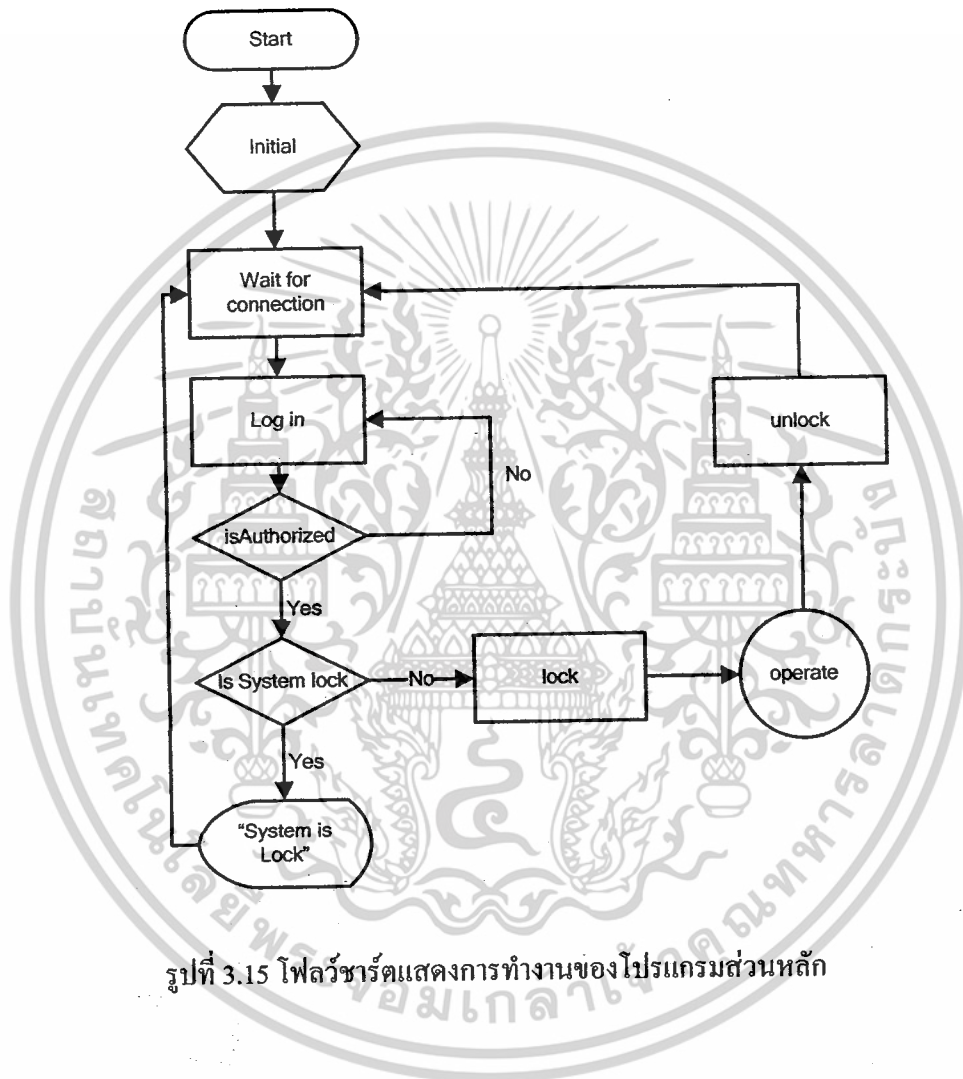
โปรแกรมในส่วนนี้จะทำหน้าที่ในการเชื่อมโยงโปรแกรมในส่วนต่างๆ เข้าด้วยกันทั้งหมด และมีหน้าที่ในการประมวลผลคำสั่งและข้อมูลที่เข้ามาเพื่อใช้ในการตัดสินใจในการปฏิบัติตามคำสั่งนั้นๆ

การทำงานหลักๆ ของโปรแกรมในส่วนนี้คือ

- เตรียมความพร้อมโดยการกำหนดค่าเริ่มต้นต่างๆ รวมถึงการสั่งให้โปรแกรมในส่วนอื่นๆ เตรียมความพร้อมในการทำงาน
- รอกการเชื่อมต่อซึ่งอาจเป็น โทรศัพท์เคลื่อนที่ คอมพิวเตอร์ หรือจากเซิร์ฟเวอร์เอง
- ทำการตรวจสอบสิทธิ์ของผู้ใช้ที่ติดต่อเข้ามา
- ถ้าเป็นผู้มีสิทธิ์ใช้งานก็จะตรวจสอบว่ามีผู้ใช้งานอยู่ก่อนแล้วหรือไม่ ถ้ามีผู้ใช้งานอยู่ก่อนก็จะทำการส่งข้อความเพื่อแจ้งว่ามีผู้ใช้งานอยู่ก่อนแล้ว

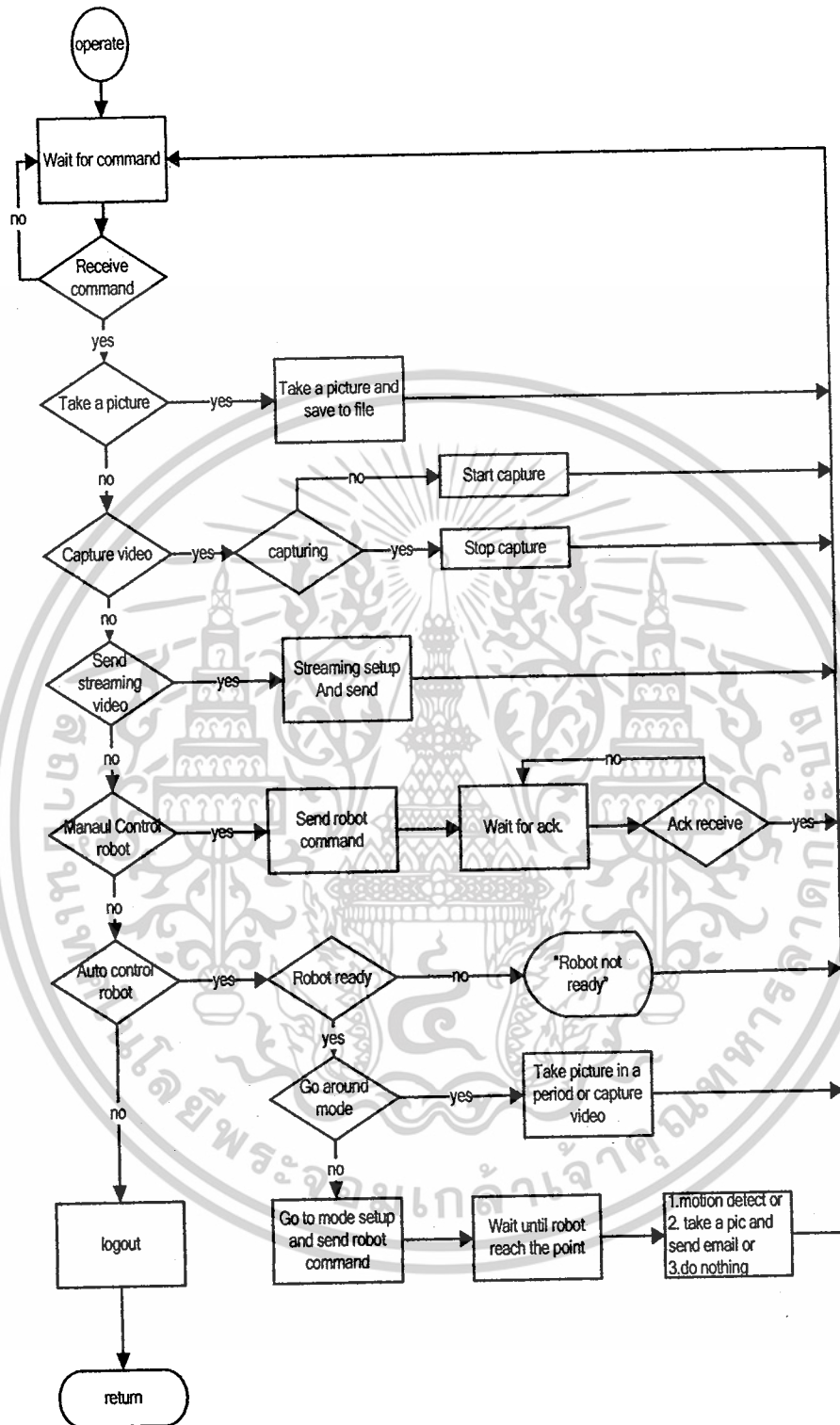
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- ถ้าไม่มีผู้ใช้งานอยู่ก็จะได้เข้าใช้งานซึ่งจะขึ้นอยู่กับว่าผู้ที่ติดต่อเข้ามาเชื่อมต่อโดยวิธีใด ถ้ามาก็จะเข้าสู่โปรแกรมการทำงานของส่วนนั้นๆ
- การเชื่อมต่อถัดไปจะได้เข้าใช้ระบบก็ต่อเมื่อผู้ใช้ปัจจุบันทำการยกเลิกการเชื่อมต่อ



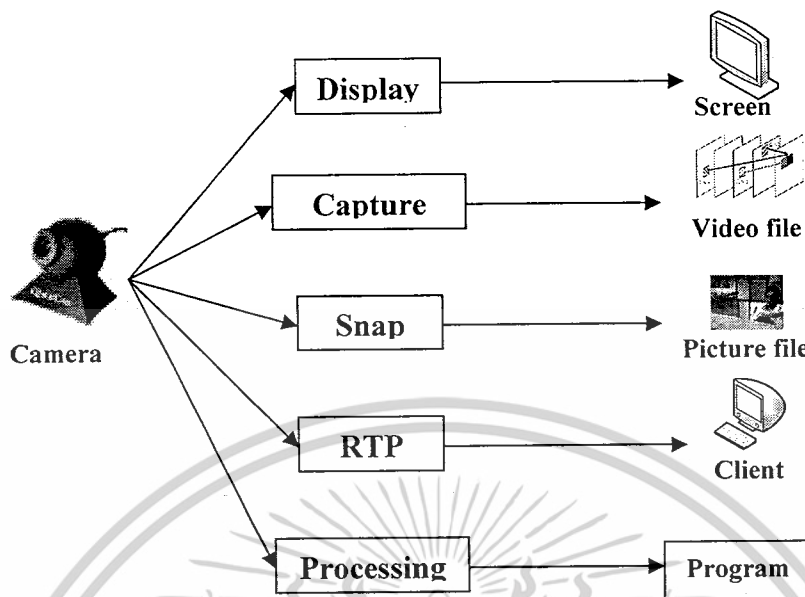
รูปที่ 3.15 โฟลว์ชาร์ตแสดงการทำงานของโปรแกรมส่วนหลัก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.16 โฟลว์ชาร์ตแสดงการทำงานในแต่ละคำสั่งของโปรแกรมหลัก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.17 แสดงการนำภาพจากกล้องมาใช้

#### ส่วนจัดการเกี่ยวกับภาพ

โปรแกรมในส่วนนี้มีหน้าที่ในการจัดการเกี่ยวกับภาพดังรูปที่ 3.17 โดยทั้งหมดจะใช้เทคโนโลยี Java Media Framework (JMF) API ในการติดต่อเพื่อรับภาพเข้ามาจากกล้องที่ติดตั้งอยู่ที่หุ่นยนต์แล้วนำมาใช้งานต่างๆ ดังต่อไปนี้

- แสดงผลภาพ (Display) ใช้แสดงผลภาพออกที่หน้าจอเครื่องเซิร์ฟเวอร์
- บันทึกภาพลงไฟล์ (Capture) ใช้สำหรับการบันทึกภาพเคลื่อนไหวจัดเก็บลงไฟล์โดยจะทำการจัดเก็บในรูปแบบของไฟล์ AVI (Audio Video Interface) และมีการเข้ารหัสวิดีโอด้วยตัวเข้ารหัส Intel Indeo
- เก็บเป็นภาพนิ่ง (Still picture) สำหรับการบันทึกเป็นภาพนิ่งโดยจะอยู่ในรูปแบบของไฟล์ JPEG (Joint Picture Expert Group)
- จัดส่งผ่านเครือข่ายอินเทอร์เน็ต (RTP) ใช้สำหรับจัดส่งข้อมูลภาพในรูปแบบเรียลไทม์ผ่านเครือข่ายไปยังเครื่องไคลเอนต์ที่ติดต่อเข้ามาโดยจะใช้ Real-time Transfer Protocol (RTP) และรูปแบบของภาพที่ส่งจะอยู่ในรูปแบบ JPEG
- ประมวลผลภาพ (Image Processing) ทำการนำภาพที่เข้ามาประมวลผลโดยที่สามารถกำหนดโหมดในการประมวลผลจะมีอยู่ 2 โหมดคือ

#### โหมดตรวจจับการเคลื่อนไหว (Motion Detection)

ในโหมดนี้จะทำการตรวจสอบว่ามีการเคลื่อนไหวเกิดขึ้นหรือไม่โดยการนำภาพ

ก่อนหน้ามาทำการเปรียบเทียบกับภาพปัจจุบันว่ามีความแตกต่างกันและมากกว่าค่าที่กำหนด เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หรือไม่ ถ้ามีความแตกต่างกันมากกว่าค่าที่กำหนดจะถือว่ามีการเคลื่อนไหวเกิดขึ้นและจะทำการแจ้งไปยังโปรแกรมหลักเพื่อทำการแจ้งเตือนต่อไป โดยในโมดนี้จำเป็นที่หุ่นยนต์จะต้องอยู่ในสถานะหยุดนิ่ง

ค่าที่ใช้กำหนดระดับของการเคลื่อนไหวจะใช้จำนวนช่องย่อยในภาพตามขนาดที่กำหนดที่มีการเปลี่ยนแปลงค่าเมื่อเทียบกับภาพก่อนหน้า

#### โมดค้นหาบาร์โค้ด (Barcode Detection)

ในโมดนี้จะทำการค้นหาและประมวลผลภาพบาร์โค้ดเพื่อใช้ในการระบุตำแหน่งของหุ่นยนต์

การที่เลือกใช้บาร์โค้ดในการระบุตำแหน่งที่ต้องการ โดยการประมวลผลภาพเนื่องจากบาร์โค้ดเป็นรหัสที่แทนค่าสัญลักษณ์ด้วยรหัสแท่งสีขาวและสีดำสลับกันทำให้คุณสมบัติที่สามารถแยกจากสิ่งแวดล้อมได้ง่ายและสามารถแทนค่าสัญลักษณ์ได้เป็นจำนวนมาก โดยในที่นี้จะใช้รหัสของบาร์โค้ดแบบ code39

#### ส่วนติดต่อกับหุ่นยนต์

โปรแกรมในส่วนนี้จะทำหน้าที่ในการรับหรือส่งข้อมูลระหว่างเซิร์ฟเวอร์กับหุ่นยนต์ผ่านพอร์ตอนุกรมโดยใช้ Java Communication API ในการสร้างโปรแกรมติดต่อกับพอร์ตอนุกรม โดยในการส่งและคำสั่งไปยังหุ่นยนต์จะมีการรอการตอบกลับเมื่อหุ่นยนต์ทำตามคำสั่งเสร็จสิ้นหรือเกิดข้อผิดพลาดในการส่งข้อมูล

#### ส่วนส่งอีเมลล์

ในส่วนนี้มีหน้าที่ในการส่งอีเมลล์เพื่อใช้ในการส่งข้อมูลภาพไปยังอุปกรณ์สื่อสารไร้สายของผู้ใช้โดยจะใช้ในการแจ้งเตือนเมื่อมีการตรวจจับการเคลื่อนไหวเมื่ออยู่ในโมดการตรวจจับการเคลื่อนไหวและใช้ในการส่งภาพไปยังผู้ใช้เมื่อต้องการ

ในการทำงานโปรแกรมจะทำการนำข้อมูลภาพส่งในรูปแบบอีเมลล์ผ่านผู้ให้บริการเครือข่ายไร้สายแล้วจึงส่งต่อไปยังผู้ใช้ในลักษณะของ MMS (Media Messaging Service)

#### ส่วนติดต่อกับโทรศัพท์เคลื่อนที่

ในส่วนนี้มีหน้าที่ในการรับคำสั่งจากโทรศัพท์เคลื่อนที่ปฏิบัติตามคำสั่งและส่งผลที่ได้กลับไปโดยอาศัยเทคโนโลยีเว็บซึ่งจะกล่าวถึงรายละเอียดต่อไป

#### ส่วนติดต่อกับไคลเอนต์ผ่านเครือข่ายอินเทอร์เน็ต

ในส่วนนี้มีหน้าที่ในการเชื่อมต่อและรับคำสั่งกับไคลเอนต์ที่ติดต่อเข้ามาผ่านเครือข่ายอินเทอร์เน็ตโดยประกอบด้วยการส่งข้อมูลภาพแบบเรียลไทม์และการรับส่งข้อมูลผ่านเครือข่ายอินเทอร์เน็ตซึ่งจะกล่าวถึงรายละเอียดต่อไป

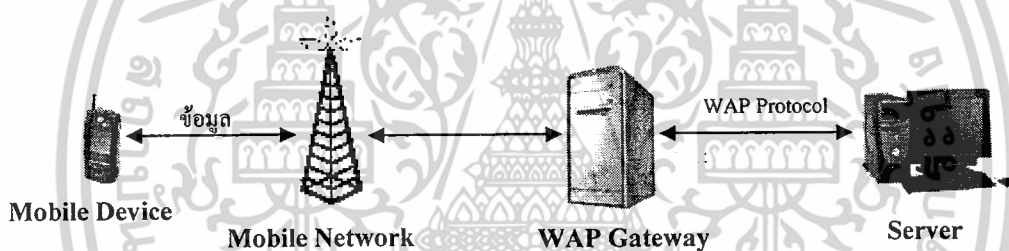
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โปรแกรมในส่วนต่างๆแบ่งออกเป็นคลาสย่อยหลายๆคลาสเพื่อความสะดวกในการเขียนโปรแกรม

### 3.2.4 โปรแกรมติดต่อระหว่างเซิร์ฟเวอร์กับโทรศัพท์เคลื่อนที่

โปรแกรมในส่วนนี้จะมีหน้าที่ในการเชื่อมต่อรับส่งข้อมูลระหว่างเซิร์ฟเวอร์กับโทรศัพท์เคลื่อนที่ผ่านเครือข่ายโทรศัพท์เคลื่อนที่ โดยในการเชื่อมต่อจะอาศัยเทคโนโลยี WAP (Wireless Application Protocol) ซึ่งในโครงสร้างจะประกอบด้วยไคลเอนต์ (Client), แวกเกตเวย์ (WAP Gateway) และเซิร์ฟเวอร์ดังรูปที่ 3.18

- ไคลเอนต์หมายถึง โทรศัพท์เคลื่อนที่
- WAP Gateway ทำหน้าที่ในการเชื่อมต่อระหว่างไคลเอนต์กับเซิร์ฟเวอร์
- เซิร์ฟเวอร์ทำหน้าที่ในการให้บริการเนื้อหาแก่ไคลเอนต์โดยในที่นี้จะใช้เทคโนโลยี Java Servlet



รูปที่ 3.18 แสดงโครงสร้างในการติดต่อระหว่างเซิร์ฟเวอร์ กับ โทรศัพท์เคลื่อนที่

เมื่อไคลเอนต์ส่งคำสั่งร้องขอข้อมูลพร้อมกับส่งค่าพารามิเตอร์มายังเซิร์ฟเวอร์ผ่านแวกเกตเวย์ ที่เซิร์ฟเวอร์จะทำการตรวจสอบค่าพารามิเตอร์ที่ส่งมาแล้วส่งข้อมูลกลับไปยังไคลเอนต์โดยจะประกอบไปด้วยหน้าต่างๆ ดังนี้

#### หน้า Login

ใช้สำหรับตรวจสอบว่าเป็นผู้ที่มีสิทธิ์เข้ามาใช้งานระบบหรือไม่โดยการให้กรอกชื่อผู้ใช้และรหัสผ่าน

#### หน้า Menu

เป็นหน้าหลักในการใช้งาน โดยจะแบ่งเป็นเมนูย่อยเพื่อเชื่อมต่อไปยังหน้าต่อไป โดยเมนูจะประกอบด้วยลิงค์ไปยังหน้าต่อไปนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### หน้า Robot Control

ใช้สำหรับการควบคุมการเคลื่อนที่ของหุ่นยนต์ในโหมดอัตโนมัติโดยจะสามารถสั่งงานได้ดังนี้

- สั่งให้หุ่นยนต์เคลื่อนที่ไปรอบๆ ภายในบ้านและทำการเก็บภาพนิ่งทุกๆ หนึ่งช่วงเวลาหรือบันทึกเป็นไฟล์วิดีโอ
- สั่งให้หุ่นยนต์เคลื่อนที่ไปยังจุดที่ต้องการ โดยสามารถเลือกได้ว่าต้องการให้ไปถึงแล้วทำการตรวจจับการเคลื่อนไหวหรือเก็บภาพนิ่งแล้วส่งภาพกลับมา

### หน้า Video Control

ใช้สำหรับสั่งงานเกี่ยวกับภาพจากกล้องที่ติดตั้งที่หุ่นยนต์ โดยมีเมนูย่อยดังนี้

- เก็บภาพนิ่งจากกล้องที่หุ่นยนต์ในขณะนั้นและแสดงภาพให้ดู
- เริ่มหรือหยุดบันทึกภาพวิดีโอเก็บไว้ที่เซิร์ฟเวอร์

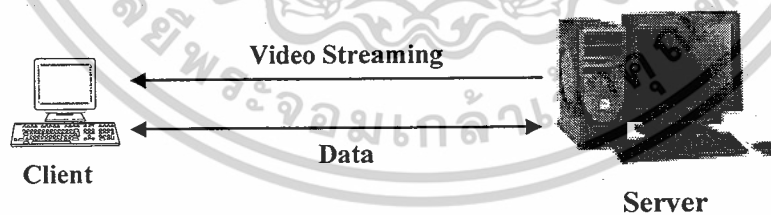
### หน้า View Picture

ใช้สำหรับดูภาพนิ่งที่ได้ถ่ายเก็บไว้ก่อนหน้าจากเซิร์ฟเวอร์

### หน้า Result

ใช้สำหรับแสดงผลการทำงานว่าสำเร็จหรือมีข้อผิดพลาดเกิดขึ้น

## 3.3 โปรแกรมติดต่อระหว่างเซิร์ฟเวอร์ ไคลเอนต์ผ่านเครือข่ายอินเทอร์เน็ต



รูปที่ 3.19 แสดงการรับส่งข้อมูลระหว่างไคลเอนต์กับเซิร์ฟเวอร์

ในส่วนนี้จะเป็นการติดต่อระหว่างไคลเอนต์ซึ่งเป็นคอมพิวเตอร์ใดๆ ที่เชื่อมต่อกับเครือข่ายอินเทอร์เน็ตกับเซิร์ฟเวอร์ ซึ่งเชื่อมต่อกับเครือข่ายอินเทอร์เน็ตเช่นเดียวกัน ดังรูปที่ 3.19 โดยที่ไคลเอนต์จะต้องทำการโหลดโปรแกรมสำหรับเชื่อมต่อมาจากเซิร์ฟเวอร์ก่อน โดยโปรแกรม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ทำให้สามารถดูภาพจากกล้องที่หุ่นยนต์ในแบบเรียลไทม์ที่ส่งมาจากเซิร์ฟเวอร์และสามารถรับและส่งข้อมูลเพื่อใช้ในการควบคุมหุ่นยนต์ไปยังเซิร์ฟเวอร์ได้โดยมีขั้นตอนในการเชื่อมต่อดังนี้

1. เซิร์ฟเวอร์จะทำการเปิดชอกเกตเพื่อรอการเชื่อมต่อจากไคลเอ็นต์
2. ไคลเอ็นต์ทำการกำหนดค่า IP ที่อยู่ของเซิร์ฟเวอร์และทำการเชื่อมต่อไปยังเซิร์ฟเวอร์
3. ไคลเอ็นต์ทำการล็อกอินถ้าที่อยู่ของเซิร์ฟเวอร์ถูกต้อง ถ้ามีผู้อื่นกำลังใช้ระบบอยู่จะไม่สามารถเข้าใช้ระบบได้
4. ถ้าข้อมูล username และ password ถูกต้องและไม่มีผู้ใช้ระบบอยู่เซิร์ฟเวอร์ จะทำการส่งข้อมูลภาพแบบเรียลไทม์จากกล้องที่ติดอยู่ที่ตัวหุ่นยนต์ผ่านเทคโนโลยี Java Media Framework (JMF) โดยอาศัย Real-time Transfer Protocol (RTP) ในการนำส่งข้อมูลภาพแบบเรียลไทม์
5. ไคลเอ็นต์สามารถส่งข้อมูลในการควบคุมการเคลื่อนที่และกล้องของหุ่นยนต์ในแบบบังคับด้วยมือผ่านชอกเกตที่เชื่อมต่อกับเซิร์ฟเวอร์เมื่อไคลเอ็นต์ ส่งข้อมูลไปยังเซิร์ฟเวอร์แล้วจะต้องรอให้เซิร์ฟเวอร์ส่งข้อมูลการตอบกลับมา จึงจะสามารถส่งข้อมูลต่อไปได้ เพื่อเป็นการป้องกันไม่ให้มีการส่งคำสั่งใหม่ไปที่หุ่นยนต์ขณะที่คำสั่งก่อนหน้ายังไม่เสร็จสิ้น

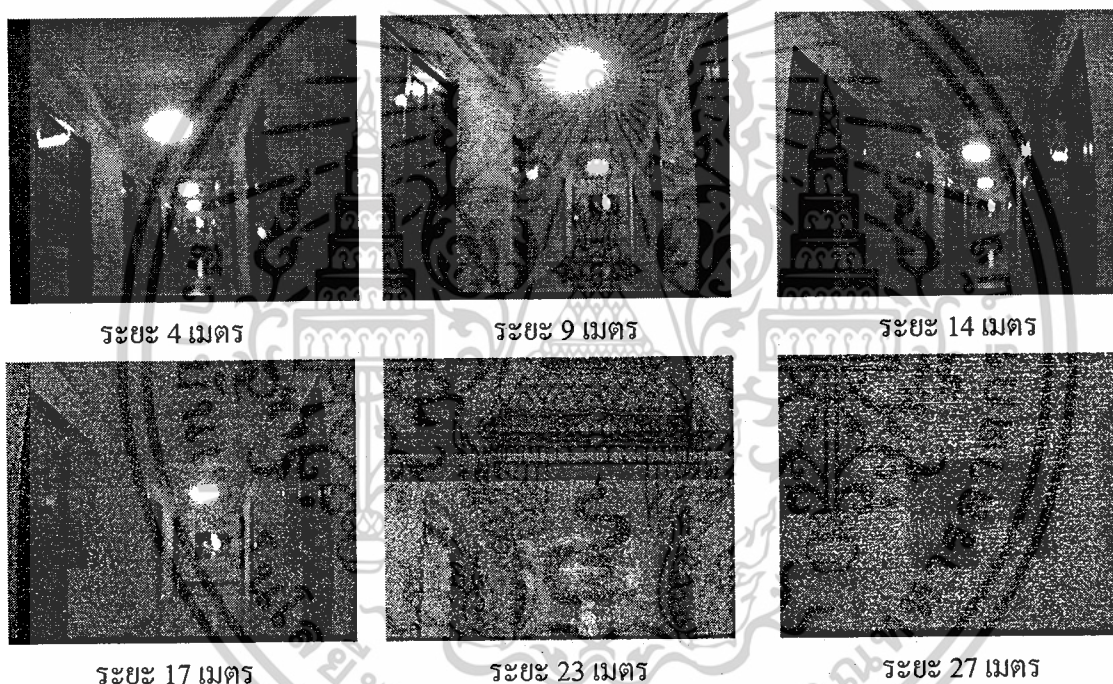
ในการออกแบบโปรแกรมฝั่งไคลเอ็นต์จะใช้เทคโนโลยีจาวาแอปพลิเคชัน(J2SE) และ Java Media Framework (JMF) ในการแสดงผลข้อมูลภาพแบบเรียลไทม์สตรีมมิ่ง

## บทที่ 4

### การทดสอบและวิเคราะห์ผล

#### 4.1 ทดสอบการรับภาพจากกล้องผ่านคลื่นวิทยุ

ตัวกล้องจะใช้แหล่งจ่ายไฟเป็นถ่าน 9 โวลต์ สภาวะที่ใช้ทดลอง จะเป็นการส่งสัญญาณภาพผ่านผนังห้อง โดยที่ตัวรับสัญญาณภาพจะอยู่ภายในห้อง ตัวกล้องจะอยู่นอกห้อง ผลที่ได้จากการทดสอบที่ระยะต่างๆเป็นดังนี้



รูปที่ 4.1 การทดสอบการรับส่งข้อมูลภาพที่ระยะต่าง ๆ

#### สรุปผลการทดลอง

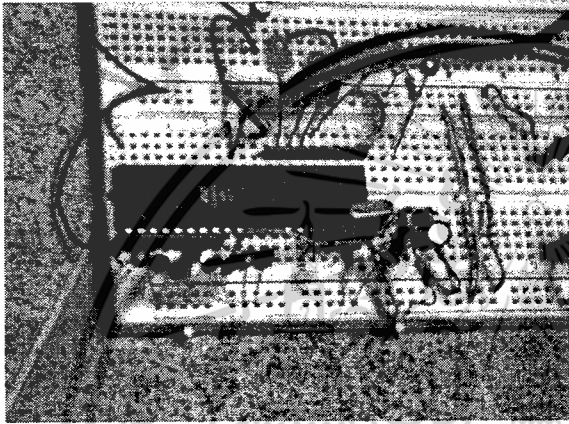
จากการทดลอง การรับสัญญาณภาพจะยังมีความชัดเจนถึงระยะประมาณ 15 เมตร หลังจากนั้นจะเริ่มมีสัญญาณรบกวน โดยที่ระยะ 17 เมตร มีสัญญาณรบกวนเล็กน้อย ระยะ 23 เมตร มีสัญญาณรบกวนมาก ที่ระยะ 27 เมตร เริ่มมองไม่เห็นภาพ นอกจากนี้พลังงานที่ให้กล้องเมื่อพลังงานเริ่มตก จะทำให้ภาพเกิดสัญญาณรบกวนขึ้นด้วย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

#### 4.2 ทดสอบการวัดอุณหภูมิ

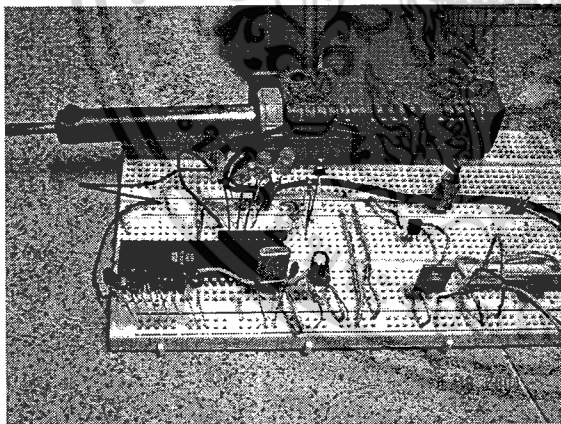
เซนเซอร์อุณหภูมิที่ใช้เป็นไอซี DS-1820 ซึ่งผลการทดลองที่อุณหภูมิต่างๆ จะแสดงออกทางหลอดไฟ LED ซึ่งมี 8 บิตเรียงจากบิตสูงไปต่ำ (P1.7 – P1.0)

โดยทำการทดลองที่อุณหภูมิต่างๆกัน ได้แก่ ในสภาวะอุณหภูมิห้อง, ในสภาวะที่มีความร้อนผิดปกติและในสภาวะที่มีความเย็นผิดปกติ โดยผลการทดลองได้เป็นดังนี้



- 1.) วัดในสภาวะอุณหภูมิห้อง  
ค่าที่อ่านได้เป็นเลขไบนารี  
 $= 01000001 = 32.5^{\circ} \text{C}$

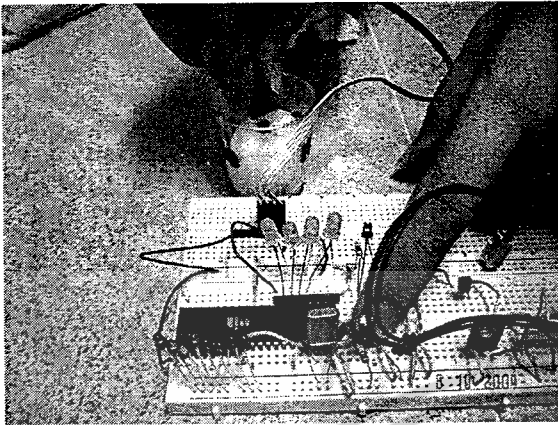
รูปที่ 4.2 การทดลองเซนเซอร์อุณหภูมิวัดที่สภาวะอุณหภูมิห้อง



- 2.) วัดในสภาวะที่มีความร้อนผิดปกติ  
ค่าที่อ่านได้เป็นเลขไบนารี  
 $= 10000010 = 65.0^{\circ} \text{C}$

รูปที่ 4.3 การทดลองเซนเซอร์วัดในสภาวะที่มีความร้อนผิดปกติ

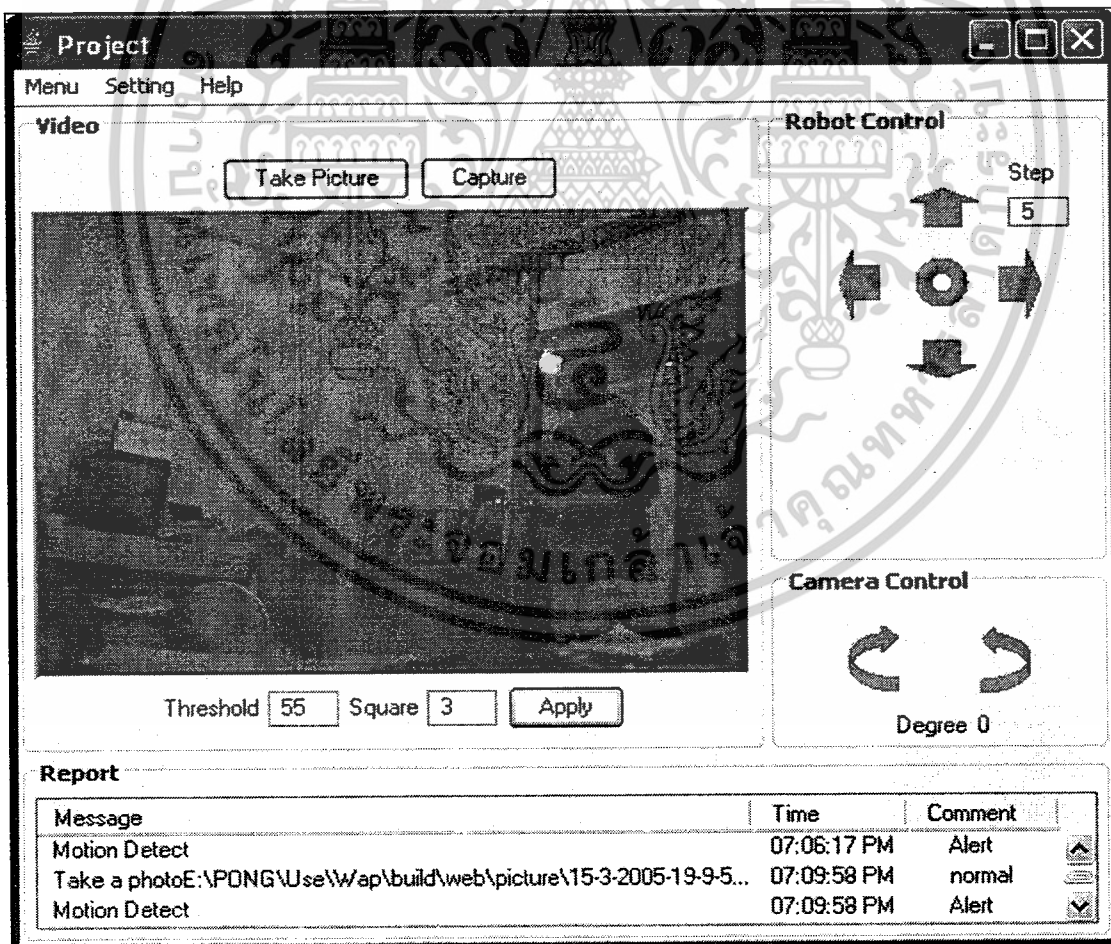
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



- 3.) วัดในสภาวะที่มีความเย็นผิดปกติ  
ค่าที่อ่านได้เป็นเลขไบนารี  
= 00001111 = 7.5° C

รูปที่ 4.4 การทดลองเซนเซอร์อุณหภูมิวัดในสภาวะที่มีความเย็นผิดปกติ

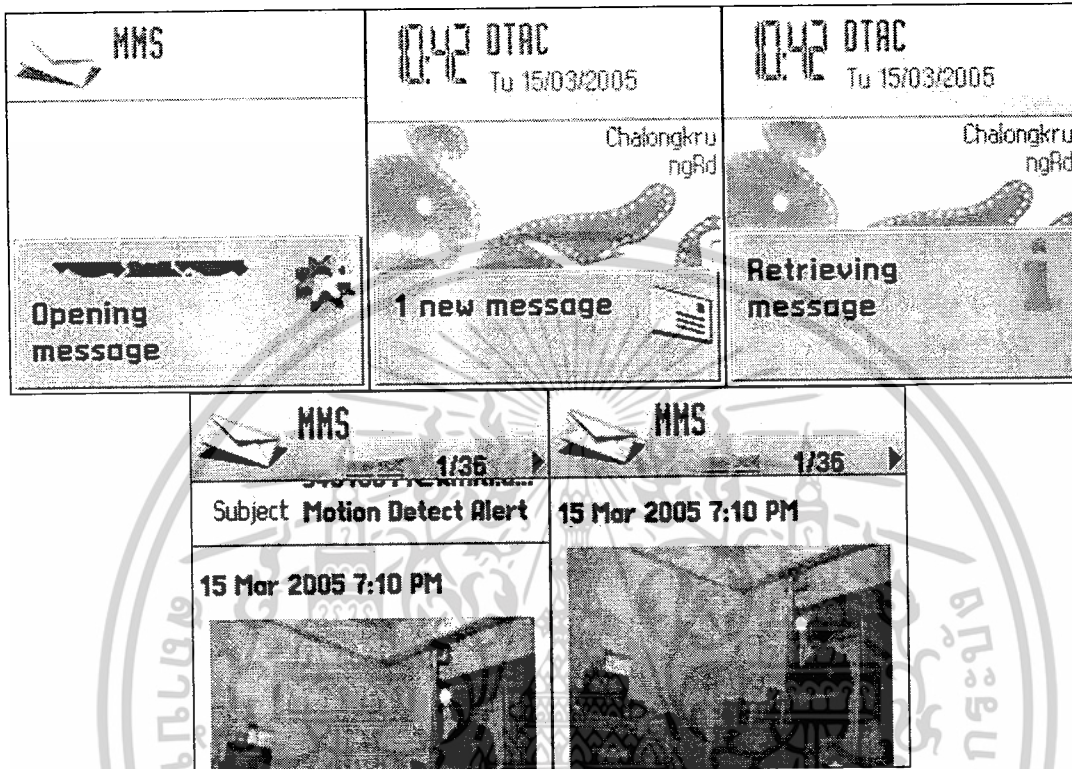
#### 4.3 การทดลองการแจ้งเตือนเมื่อสามารถตรวจจับการเคลื่อนไหว



รูปที่ 4.5 หน้าจอการตรวจจับความเคลื่อนไหว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เมื่อเกิดความเคลื่อนไหวขึ้นมากกว่าค่าที่ตั้งไว้(Threshold) ที่ตั้งไว้ระบบจะทำการบันทึกภาพความเคลื่อนไหวนั้นไว้และทำการแจ้งเตือนโดยส่วน MMS ให้ผู้ใช้



รูปที่ 4.6 การส่ง MMS แจ้งเตือนความเคลื่อนไหว

#### 4.4 การทดลองการอ่านรหัสบาร์โค้ด

เป็นการทดลองการอ่านรหัสบาร์โค้ดซึ่งใช้ในการระบุตำแหน่งภายในห้องต่างๆ โดยการประมวลผลภาพ ซึ่งจะจำลองสภาพของภาพที่รับได้ในรูปแบบต่างๆ ที่อาจจะเกิดขึ้นกับหุ่นยนต์ ได้แก่ ระยะห่างระหว่างกล้องกับบาร์โค้ด ภาพบาร์โค้ดในลักษณะเอียง

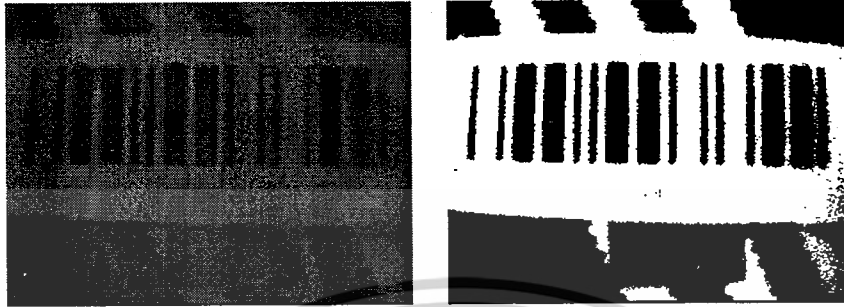
√ = อ่านรหัสได้      × = อ่านรหัสไม่ได้

ตารางที่ 4.1 ผลการทดลองการอ่านรหัสบาร์โค้ดสัญลักษณ์ที่ 1

ระยะห่าง(นิ้ว)	ตรง	เอียงซ้าย30องศา	เอียงขวา30องศา
4	√	√	√
6	√	√	√
9	√	√	×

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตัวอย่างรูปผลการทดลอง



รูปที่ 4.7 อินพุตและและผลลัพธ์จากโปรแกรมฟิลเตอร์เมื่อแถบบาร์โค้ดอยู่ห่างจากกล้องเป็นแนวตรงที่ ระยะ 4 นิ้ว



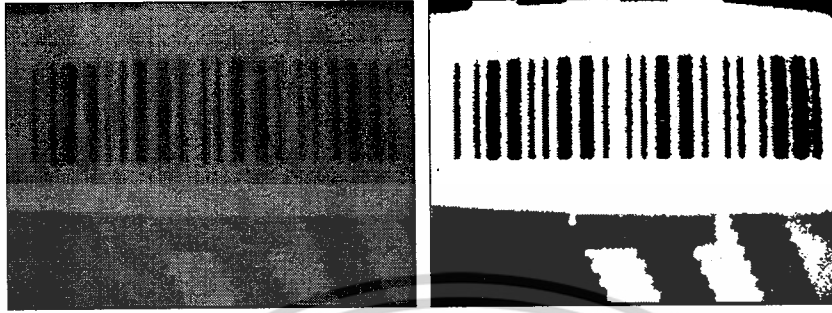
รูปที่ 4.8 อินพุตและและผลลัพธ์จากโปรแกรมฟิลเตอร์เมื่อแถบบาร์โค้ดอยู่ห่างจากกล้องเป็นแนวตรงที่ ระยะ 9 นิ้ว

ตารางที่ 4.2 ผลการทดลองการอ่านรหัสบาร์โค้ดสัญลักษณ์ที่ 2

ระยะห่าง(นิ้ว)	ตรง	เอียงซ้าย30องศา	เอียงขวา30องศา
4	√	×	√
6	√	√	√
9	√	×	×

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### ตัวอย่างรูปผลการทดลอง



รูปที่ 4.9 อินพุตและและผลลัพธ์จากโปรแกรมฟิลเตอร์แถบบาร์โค้ดที่ 2 เมื่อแถบบาร์โค้ดอยู่ห่างจากกล้องเป็นแนวตรงที่ ระยะ 4 นิ้ว



รูปที่ 4.10 อินพุตและและผลลัพธ์จากโปรแกรมฟิลเตอร์แถบบาร์โค้ดที่ 2 เมื่อแถบบาร์โค้ดอยู่ห่างจากกล้องเป็นแนวตรงที่ ระยะ 9 นิ้ว

### สรุปผลการทดลอง

ระยะที่จะอ่านรหัสบาร์โค้ดได้อยู่ที่ระยะ 4-9 นิ้ว และมีมุมเอียงที่ไม่ควรเกิน 30 องศา จากการทดลองระยะที่มีความเหมาะสมที่สุดอยู่ที่ 6 นิ้ว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

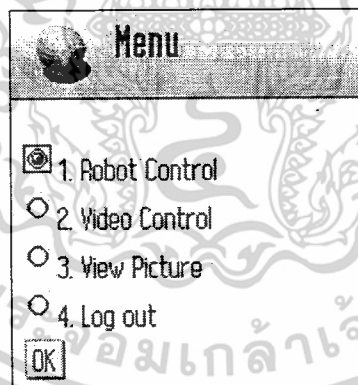
#### 4.5 การทดลองติดต่อผ่านโทรศัพท์เคลื่อนที่

ทำการทดลองโดยการเชื่อมต่อจากโทรศัพท์เคลื่อนที่เข้ามายังเซิร์ฟเวอร์ผ่านเครือข่ายจีพีอาร์เอส (GPRS) และใช้โปรแกรมเบราเซอร์ของโทรศัพท์เคลื่อนที่ซึ่งรองรับเว็บเวอร์ชัน 2.0

Login	Result	Login
Username: <input type="text" value="pong"/> Password: <input type="password" value="****"/> <input type="button" value="login"/> <i>security robot v. 0.1</i>	<b>Message:</b> Your username or password are incorrect! <input type="button" value="login"/> <i>security robot v. 0.1</i>	<b>Message:</b> System lock. <input type="button" value="login"/> <i>security robot v. 0.1</i>

รูปที่ 4.11 หน้าจอแสดงการ log in ผ่านทางโทรศัพท์เคลื่อนที่

การเข้าใช้งานระบบโดยการป้อนชื่อและรหัสผ่าน ถ้าชื่อหรือรหัสผ่านที่ป้อนไม่ถูกต้องระบบจะล็อก



รูปที่ 4.12 หน้าจอแสดงฟังก์ชันการใช้งานระบบผ่านทางโทรศัพท์เคลื่อนที่

เป็นหน้าหลักในการใช้งาน โดยจะแบ่งเป็นเมนูย่อยเพื่อเชื่อมต่อไปยังหน้าต่อไป ในรูปข้างบนทำการเลือกเมนู Robot Control

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Robot Control	Robot Control	Robot Control
<input checked="" type="radio"/> Go to: living → mode: motion detect → <input type="radio"/> Go Around: take pic → [OK]	<input checked="" type="radio"/> Go to: living → mode: motion detect → <b>Select</b> <input checked="" type="radio"/> living <input type="radio"/> project	<input checked="" type="radio"/> Go to: living → mode: motion detect → <b>Select</b> <input checked="" type="radio"/> take pic <input type="radio"/> cap video

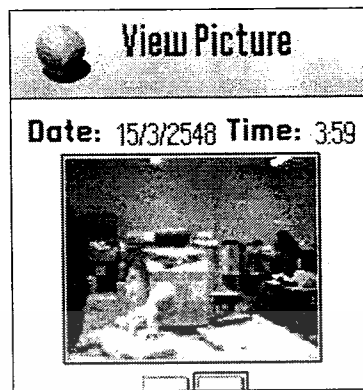
รูปที่ 4.13 หน้าจอการควบคุมการเคลื่อนที่ของหุ่นยนต์ผ่านทางโทรศัพท์เคลื่อนที่

ในหน้า Robot Control จะมีโหมดให้เลือก คือ โหมดที่ตั้งให้หุ่นไปห้องที่ต้องการโดยสามารถเลือกได้ว่าเมื่อไปถึงแล้วให้ตรวจจับความเคลื่อนไหวหรือให้ถ่ายภาพกลับมา และโหมดเดินรอบบ้านโดยสามารถเลือกได้ว่าให้ถ่ายรูปไปด้วยหรือให้ถ่ายวิดีโอ

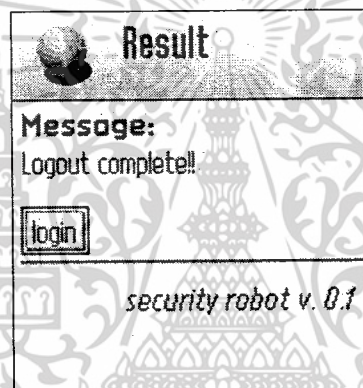
Video Control	Result
<input type="radio"/> 1. Take Picture <input checked="" type="radio"/> 2. Start Capture Video <input type="button" value="OK"/> <input type="button" value="MENU"/> <i>security robot v. 0.1</i>	<b>Message:</b> Capturing has been Start <input type="button" value="menu"/> <i>security robot v. 0.1</i>

รูปที่ 4.14 หน้าจอสำหรับการสั่งงานเกี่ยวกับภาพที่ติดอยู่ที่ตัวหุ่นยนต์

สามารถเลือกได้ว่าให้ถ่ายรูปหรือให้ถ่ายวิดีโอเก็บไว้ ในรูปด้านบนทำการเลือกเมนูถ่ายวิดีโอ จะปรากฏข้อความว่าเริ่มอัดไฟล้วิดีโอแล้ว



รูปที่ 4.15 หน้าจอแสดงไฟล์ภาพที่ได้รับจากเซิร์ฟเวอร์

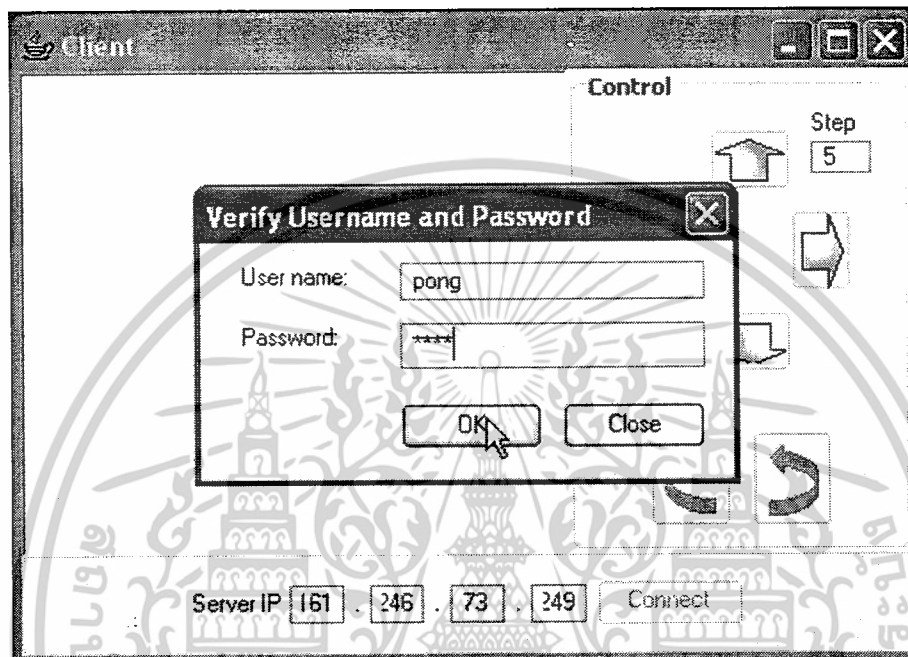


รูปที่ 4.16 หน้าจอแสดงข้อความออกสู่ระบบ

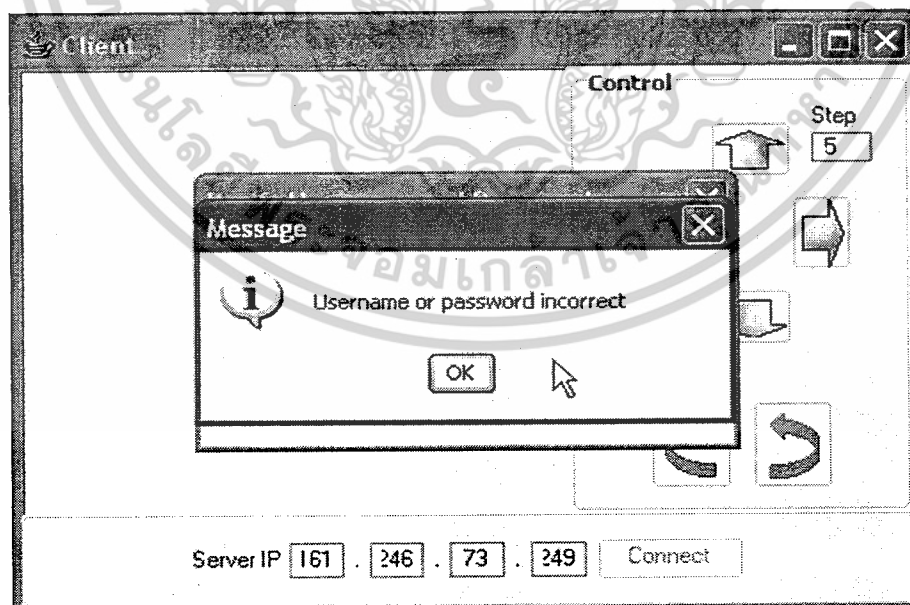
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

#### 4.6 การทดลองติดต่อผ่านอินเทอร์เน็ต

ทำการทดลองโดยการเชื่อมต่อจากโปรแกรมไคลเอนท์ผ่านเครือข่ายอินเทอร์เน็ตมายังเซิร์ฟเวอร์ โดยที่ทั้งสองฝั่งเชื่อมต่อเข้ากับอินเทอร์เน็ตผ่านเครือข่ายแลนความเร็ว 100 Mbps

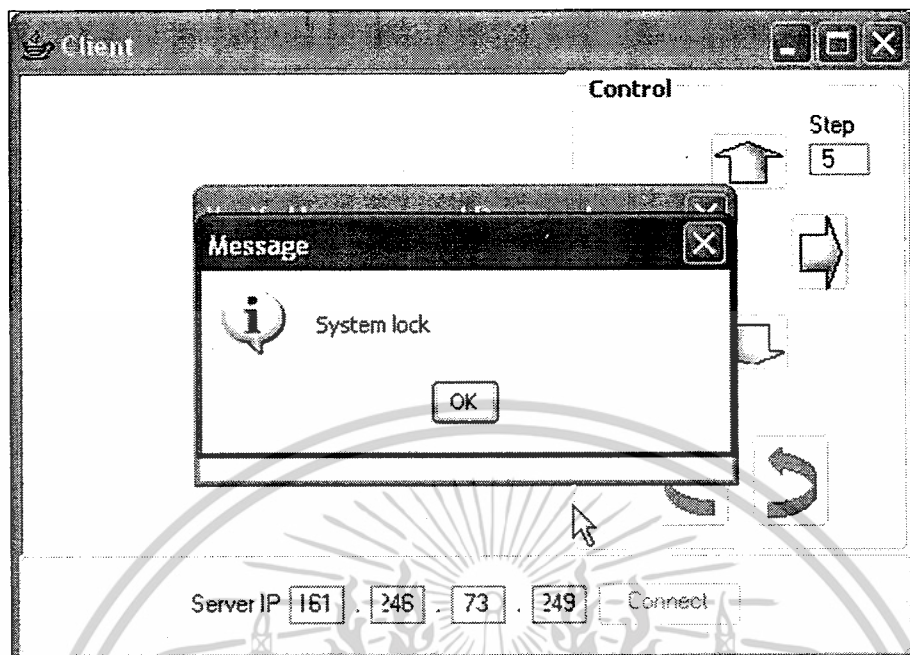


รูปที่ 4.17 หน้าจอใช้งานระบบผ่านอินเทอร์เน็ต



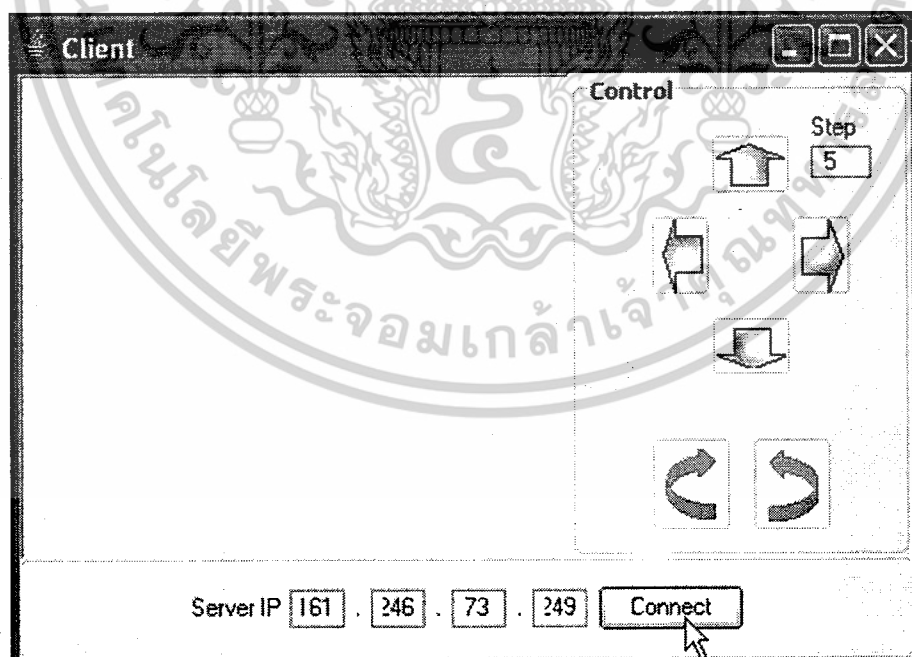
รูปที่ 4.18 หน้าจอแสดงข้อความแจ้งเตือนหากกรอกชื่อผู้ใช้หรือรหัสผ่านผิด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



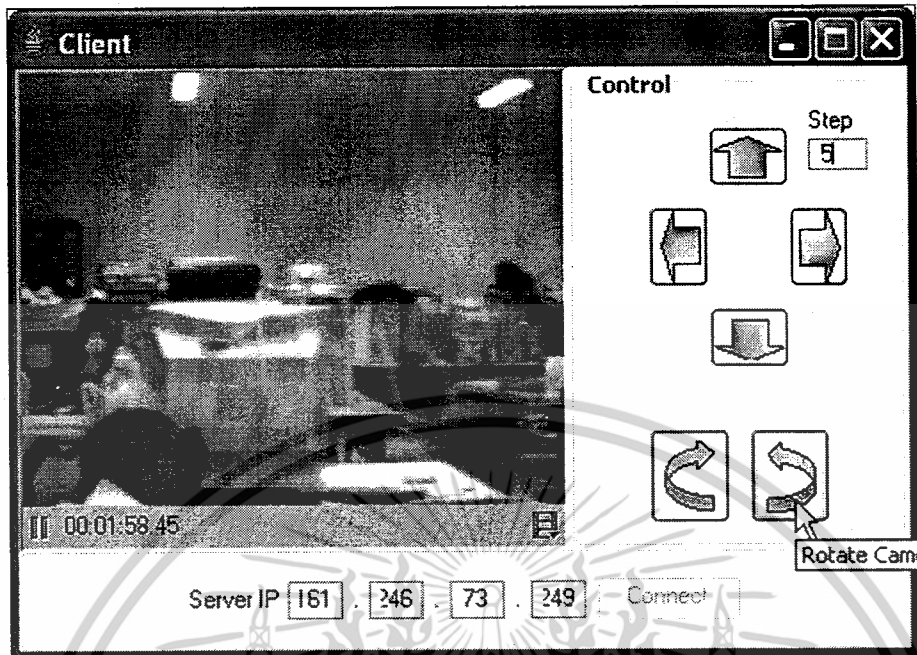
รูปที่ 4.19 หน้าจอแสดงข้อความแจ้งเตือนเมื่อมีผู้ใช้งานระบบอยู่

ถ้าหากระบบในขณะนั้นมีผู้ใช้อยู่ถึงแม้ผู้ใช้คนอื่นจะล็อกอินเข้ามาได้แต่จะไม่สามารถใช้งานได้โดยจะมีข้อความแจ้งเตือน



รูปที่ 4.20 หน้าจอการเชื่อมต่อ เพื่อเข้าใช้งานระบบผ่านอินเทอร์เน็ต

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

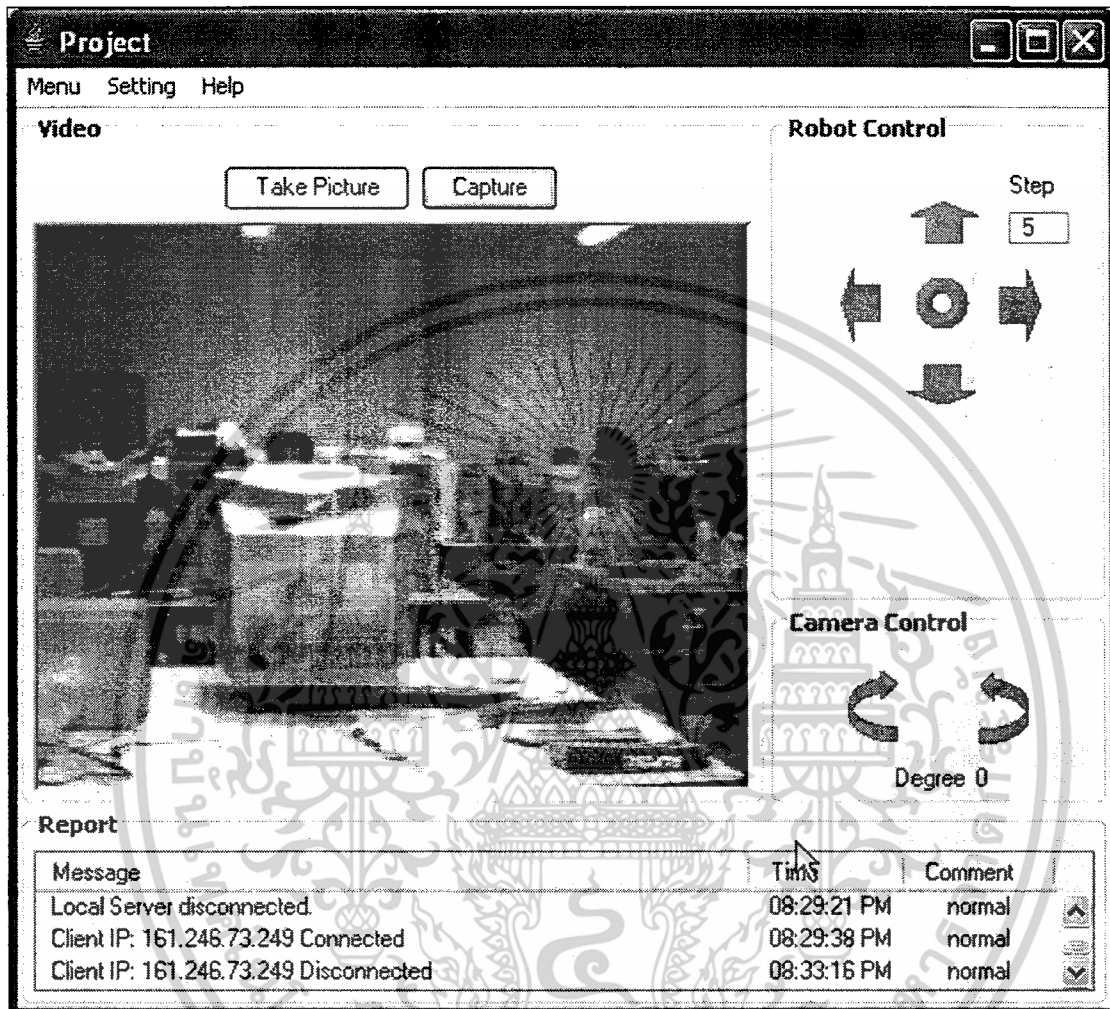


รูปที่ 4.21 หน้าจอแสดงฟังก์ชันการใช้งานระบบผ่านอินเทอร์เน็ต

เมื่อทำการเชื่อมต่อเข้าสู่ระบบได้แล้ว จะสามารถบังคับหุ่นยนต์ให้เคลื่อนที่เดินหน้า, ถอยหลัง, เลี้ยวซ้าย, เลี้ยวขวา ไปตามระยะที่ต้องการได้ และสามารถบังคับการหมุนของกล้องได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

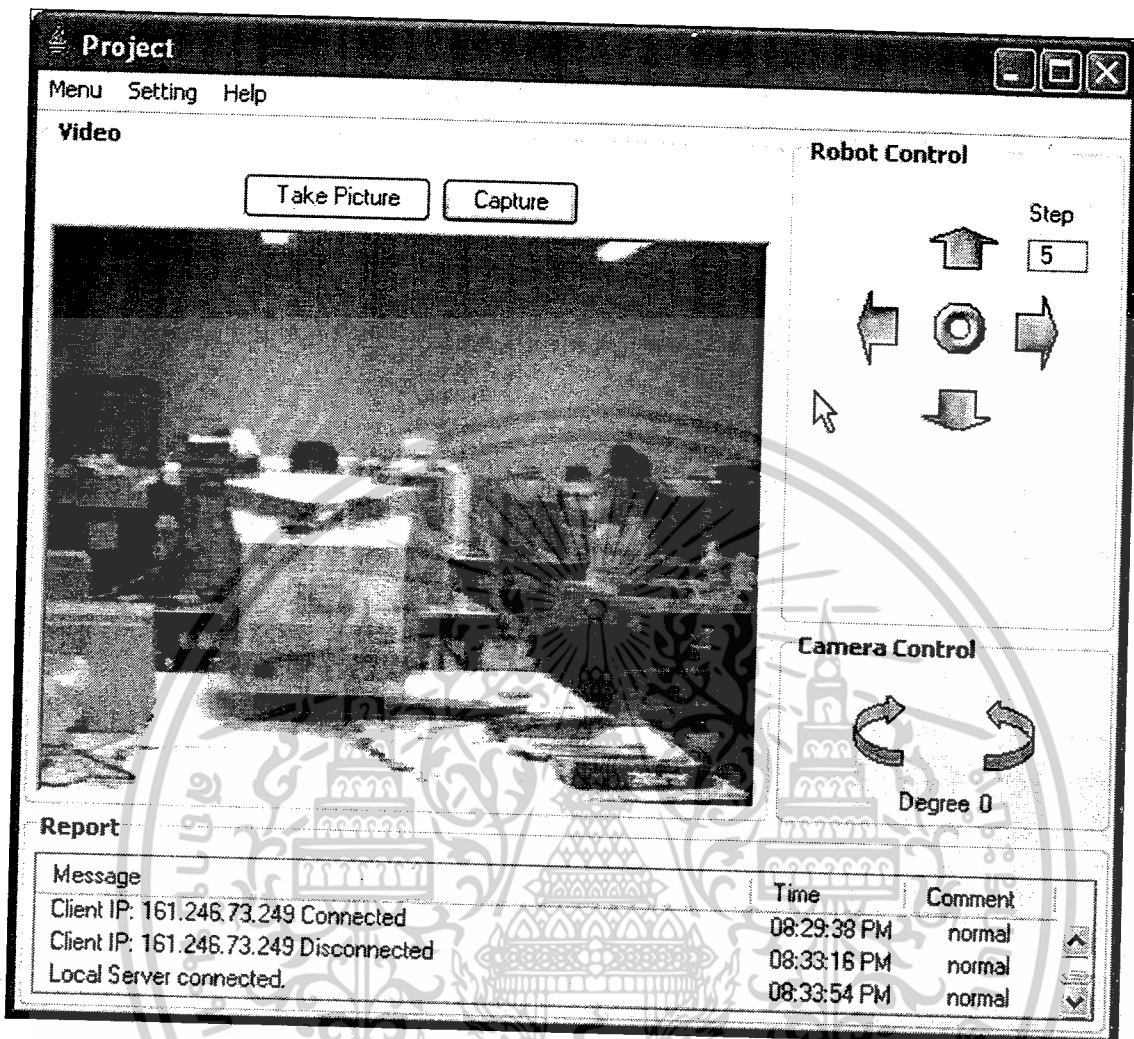
#### 4.7 การทดลองโปรแกรมควบคุมที่เซิร์ฟเวอร์



รูปที่ 4.22 หน้าจอแสดงฟังก์ชันการใช้งานที่เซิร์ฟเวอร์

โดยที่เซิร์ฟเวอร์จะสามารถทำการถ่ายภาพ, ถ่ายวิดีโอ, ควบคุมหุ่นยนต์, ควบคุมการหมุนกลิ้ง โดยในรูปแบบบนจะเป็นการเลือกถ่ายวิดีโอ

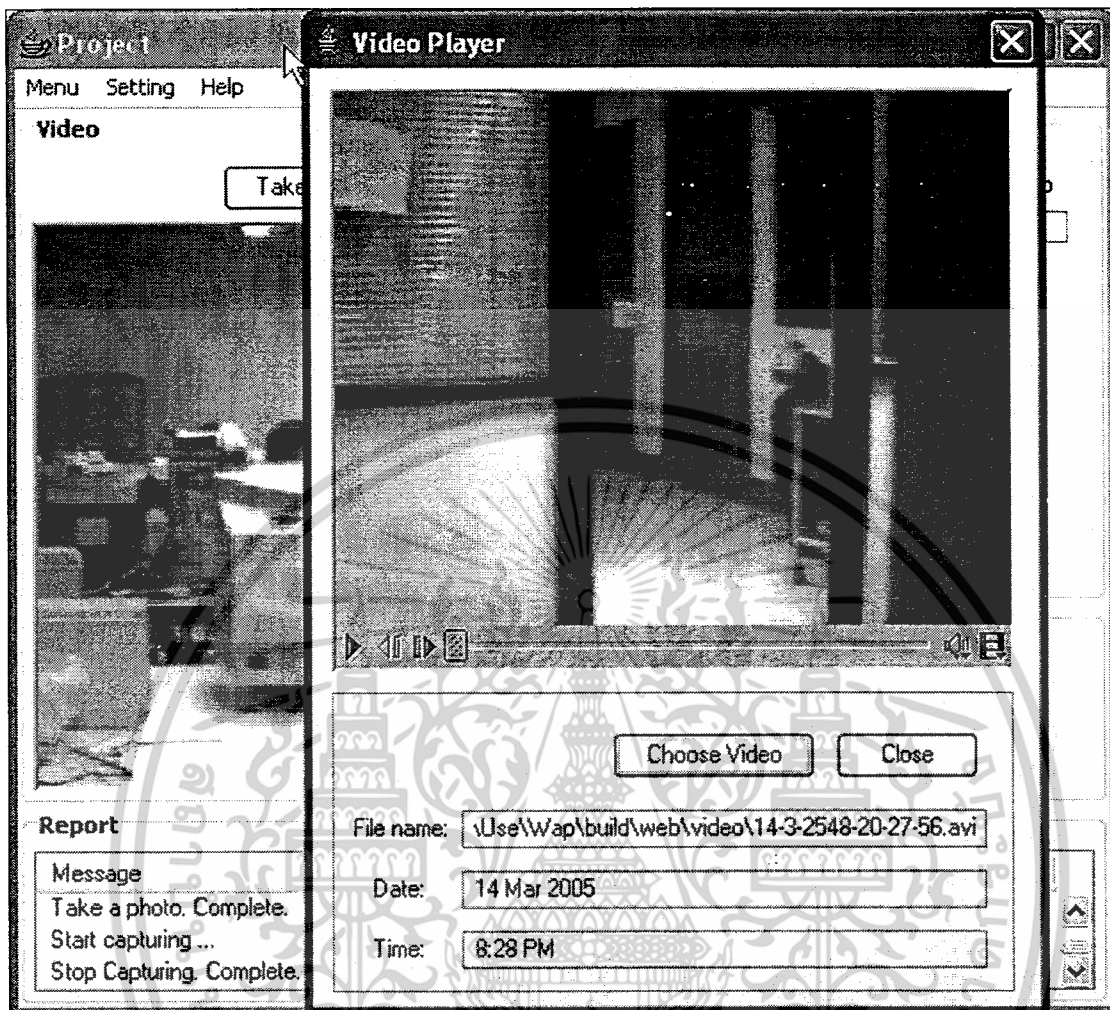
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.23 หน้าจอแสดงฟังก์ชันการใช้งานการควบคุมหุ่นยนต์ที่เซิร์ฟเวอร์

ที่เซิร์ฟเวอร์จะเลือกทำการควบคุมหุ่นยนต์แบบแมนนวลได้โดยสามารถกำหนดได้ว่าให้เคลื่อนที่ไปที่สตีปปีจากรูปด้านบนจะกำหนดที่ 5 สตีปปี และกำหนดการหมุนของกล้องได้ว่าให้หมุนไปที่องศา (0-180องศา)

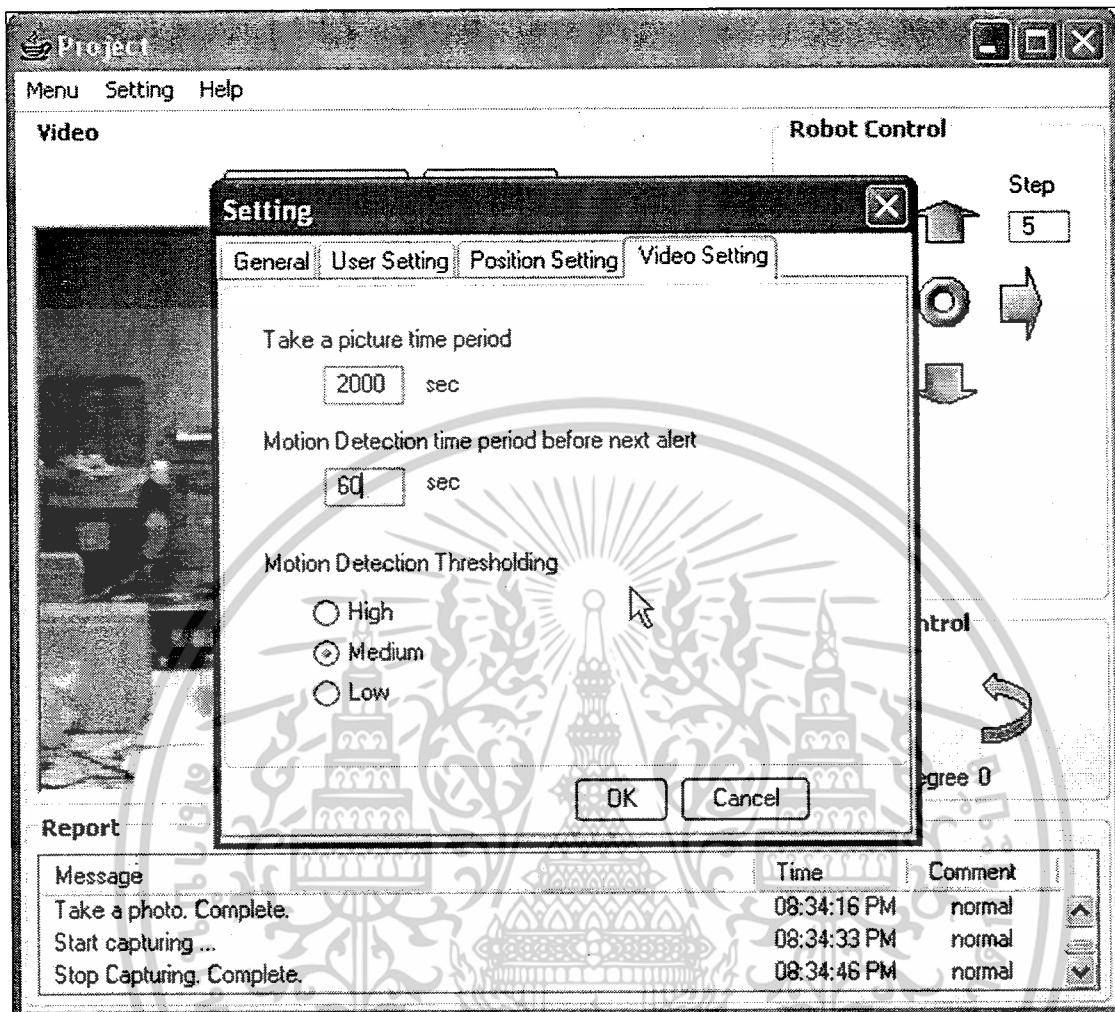
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.24 หน้าจอการเลือกดูไฟล์วิดีโอที่ได้ทำการบันทึกไว้

ที่เซิร์ฟเวอร์จะสามารถเลือกดูไฟล์วิดีโอที่ได้ทำการบันทึกเอาไว้ จากรูปด้านบน จะเป็นการเล่นไฟล์วิดีโอที่ผู้ใช้เลือกดู โดยจะแสดงชื่อของไฟล์, วันที่ได้ถ่ายเก็บไว้, เวลาที่เริ่มถ่าย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.25 หน้าจอแสดงการตั้งค่าพารามิเตอร์ต่างๆจากโทรศัพท์เคลื่อนที่

โดยในการใช้งานระบบผ่านทางโทรศัพท์เคลื่อนที่เราสามารถตั้งค่าพารามิเตอร์ที่ต้องการได้ ในรูปด้านบนจะเป็นการตั้งค่าเกี่ยวกับวิดีโอ โดยได้ตั้งค่าให้ถ่ายรูปในทุกๆ 2000 วินาที ช่วงเวลาในการตรวจจับความเคลื่อนไหวคือ 60 วินาที, ค่าที่ตั้งไว้ (Threshold) อยู่ที่ปานกลาง (medium)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



## 5.2 ปัญหาที่พบ

ปัญหาที่พบจากการพัฒนาโครงการนี้ส่วนมากจะเป็นปัญหาที่เกิดจากองค์ความรู้ที่มีอยู่เดิมมีไม่เพียงพอทำให้ต้องใช้เวลาในการศึกษาทำให้เกิดการล่าช้า พร้อมกับการออกแบบที่ไม่สมบูรณ์จึงต้องมีการแก้ไขสิ่งที่ได้ทำไปแล้ว ปัญหาในเชิงเทคนิคที่เกิดขึ้น ได้แก่

1. ในการควบคุมจากโทรศัพท์เคลื่อนที่ซึ่งจะรับส่งข้อมูลผ่านเครือข่ายโทรศัพท์ซึ่งยังมีความน่าเชื่อถือและความเร็วที่ต่ำจึงทำให้เกิดปัญหาในการรับส่งข้อมูลที่ล่าช้าหรือเกิดความผิดพลาดเกิดขึ้นบ่อยครั้ง
2. ส่วนการรับส่งข้อมูลผ่านทางคลื่นวิทยุ (RF) อาจกล่าวได้ว่าเป็นปัญหาที่ใหญ่ที่สุดในการทำโครงการนี้ เพราะการรับส่งข้อมูลแบบไร้สายจะเกิดปัญหามากไม่ว่าจะเป็นเรื่องระยะทางที่รับส่งได้, ความไม่แน่นอนของข้อมูลที่รับส่ง, การที่ต้องใช้การสื่อสารแบบสองทางคือมีทั้งตัวรับและตัวส่งในวงจรเดียวกันทำให้ระยะทางที่รับส่งได้สั้นลงและต้องให้แต่ละตัวทำงานคนละเวลากัน
3. ส่วนวงจรเซนเซอร์อินฟราเรดมีปัญหาในเรื่องความไม่เสถียรเนื่องจากสภาวะแวดล้อมที่ต่าง ๆ กัน เช่น สภาพแสงที่มีผลทำให้ตัวรับของอินฟราเรดทำงานผิดพลาดได้
4. การรับส่งภาพมาแสดงที่คอมพิวเตอร์ จะเกิดปัญหาทางด้านระยะทางที่ทำงานได้ เนื่องจากมีข้อจำกัดทางด้านพลังงานที่ให้ซึ่งใช้ถ่าน 9 โวลต์ ระยะเวลาที่ทำงานได้ยังไม่นานเท่าที่ควรเพราะพลังงานของถ่านลดลงค่อนข้างเร็ว ทำให้ภาพที่ได้มีสัญญาณรบกวนไม่ชัด

## 5.3 แนวทางในการพัฒนาต่อ

แนวทางในการพัฒนาต่อจะเป็นการพัฒนาแบบค่อยเป็นค่อยไปสามารถแบ่งออกได้เป็น 2 ช่วง โดยแต่ละช่วงจะเป็นการพัฒนาต่อจากระบบเดิมเพื่อให้มีประสิทธิภาพและสามารถนำไปใช้งานได้จริงมากยิ่งขึ้น

ช่วงที่ 1 ในช่วงนี้จะเน้นไปที่ความสามารถในการระบุตำแหน่งของหุ่นยนต์โดยใช้การประมวลผลภาพจากกล้องที่ติดอยู่ที่ตัวหุ่นยนต์เพื่อให้หุ่นยนต์สามารถที่จะระบุตำแหน่งของตัวเองได้ทุกที่และสามารถนำไปใช้ประโยชน์ในด้านอื่นๆ อีกมากมาย อย่างเช่น ใช้เป็นหุ่นยนต์ทำงานบ้าน และอีกส่วนคือส่วนการรับส่งข้อมูลระหว่างเซิร์ฟเวอร์กับหุ่นยนต์ผ่านคลื่นวิทยุ จะต้องพัฒนาให้มีความรวดเร็วและความถูกต้องในการรับส่งข้อมูล มีระยะที่รับส่งข้อมูลได้ไกลยิ่งขึ้น และเป็น การรับส่งข้อมูลแบบพูลดิวเพิล็กซ์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ช่วงที่ 2 ในช่วงนี้จะเน้นการพัฒนาให้หุ่นยนต์สามารถทำงานได้ด้วยตัวเองโดยไม่มี การสั่งงานผ่านคอมพิวเตอร์ โดยในการสั่งงานหุ่นยนต์จากระยะไกลจะทำผ่านเครือข่ายข้อมูลไร้สาย เช่น เครือข่ายท้องถิ่นไร้สาย (Wireless LAN) หรือเครือข่ายโทรศัพท์เคลื่อนที่ ส่วนประมวลผลและควบคุมอาจใช้เป็นระบบคอมพิวเตอร์ฝังตัว



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## บรรณานุกรม

- [1] Wap Forum Wireless Application Protocol WAP 2.0 Technical White Paper, Wireless Application Protocol Forum Ltd.
- [2] Java™ Media Framework API Guide November 19, 1999MF 2.0 FCS, 1998-99 Sun Microsystems, Inc.
- [3] ชัยวัฒน์ ลิ้มพรจิตรวิไล, วรพจน์ กรแก้ววัฒนกุล, “เรียนรู้และปฏิบัติการไมโครคอนโทรลเลอร์ MCS-51”, กรุงเทพฯ, 2538

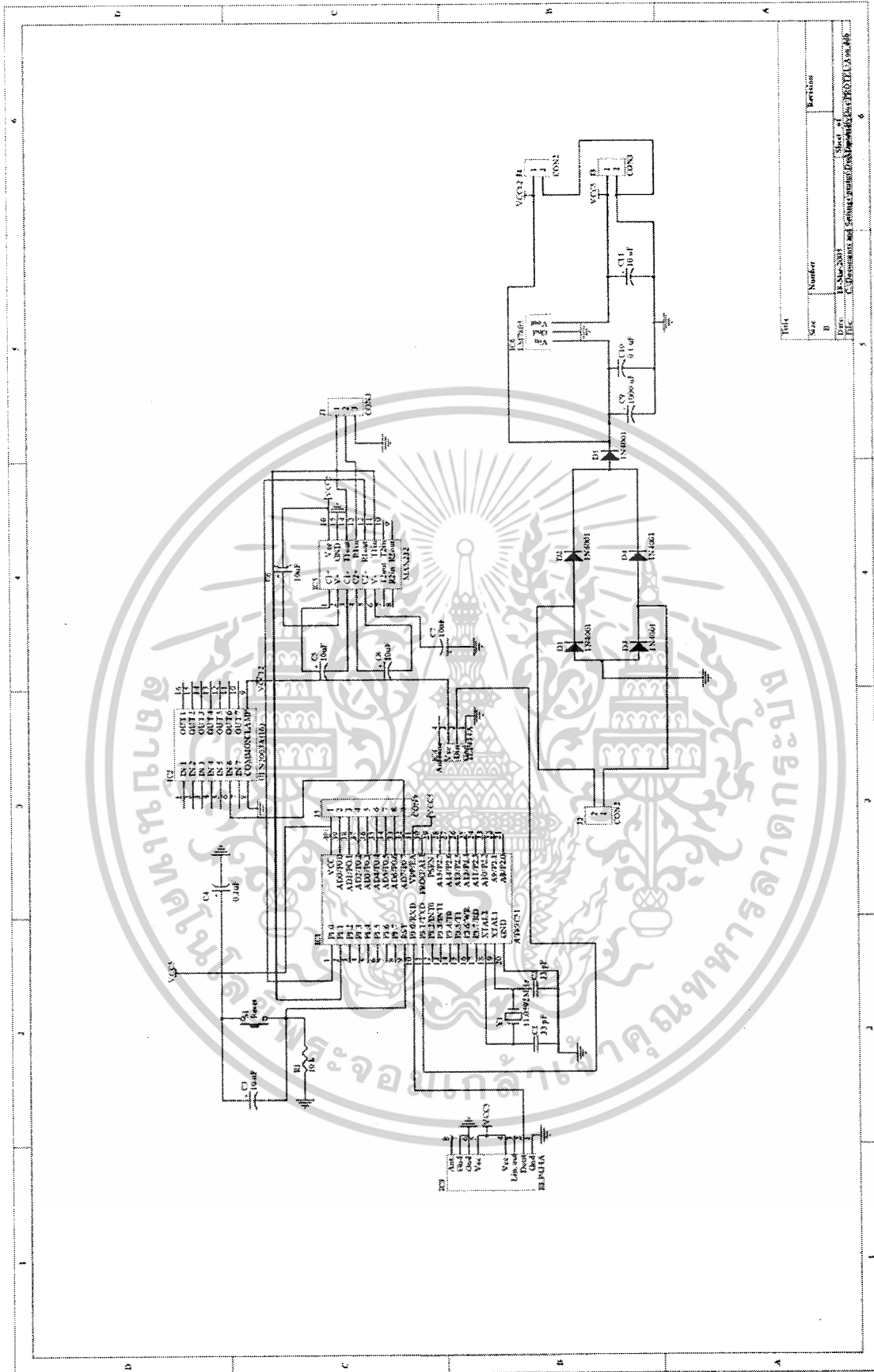


เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

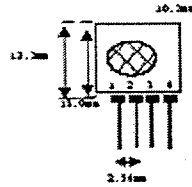
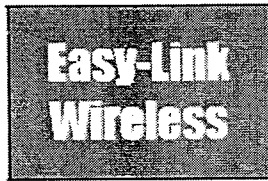




วงจรส่วนส่งสัญญาณควบคุมหุ่นยนต์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## TLP434A Ultra Small Transmitter



pin 1 : GND  
pin 2 : Data In  
pin 3 : Vcc  
pin 4 : Antenna ( RF output )

**Frequency 315, 418 and 433.92 Mhz**

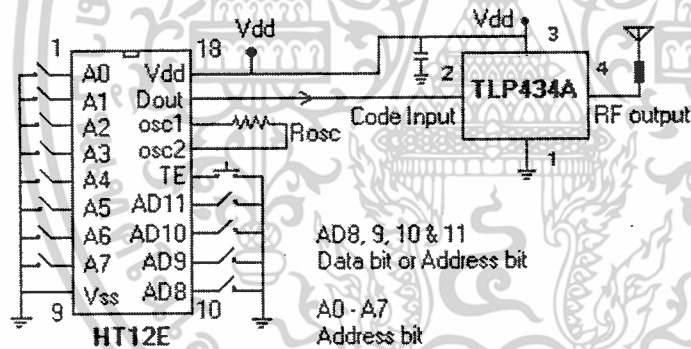
Modulation : ASK  
Operation Voltage : 2 - 12 VDC

Symbol	Parameter	Conditions	Min	Typ	Max	Unit
Vcc	Operating supply voltage		2.0	-	12.0	V
Icc 1	Peak Current (2V)		-	-	1.64	mA
Icc 2	Peak Current (12V)		-	-	19.4	mA
Vh	Input High Voltage	I <sub>data</sub> = 100µA (High)	V <sub>cc</sub> - 0.5	V <sub>cc</sub>	V <sub>cc</sub> - 0.5	V
VI	Input Low Voltage	I <sub>data</sub> = 0 µA (Low)	-	-	0.3	V
FO	Absolute Frequency	315Mhz module	314.8	315	315.2	MHz
PO	RF Output Power- 50ohm	V <sub>cc</sub> = 9V-12V	-	16	-	dBm
		V <sub>cc</sub> = 5V-6V	-	14	-	dBm
DR	Data Rate	External Encoding	512	4.8K	200K	bps

Notes : ( Case Temperature = 25°C ~ 2°C , Test Load Impedance = 50 ohm )

### Application Circuit :

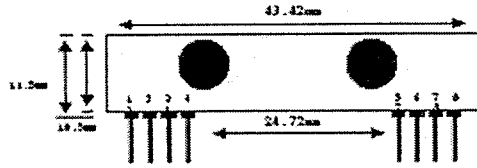
Typical Key-chain Transmitter using HT12E-18DIP, a Binary 12 bit Encoder from Holtek Semiconductor Inc.



ดาวน์โหลด (Datasheet) ตัวส่งสัญญาณความถี่วิทยุไอซี TLP 434A

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### RLP434A SAW Based Receiver



- pin 1 : Gnd
- pin 2 : Digital Data Output
- pin 3 : Linear Output /Test
- pin 4 : Vcc
- pin 5 : Vcc
- pin 6 : Gnd
- pin 7 : Gnd
- pin 8 : Antenna

**Frequency 315, 418 and 433.92 Mhz**

Modulation : ASK  
 Supply Voltage : 3.3 - 6.0 VDC  
 Output : Digital & Linear

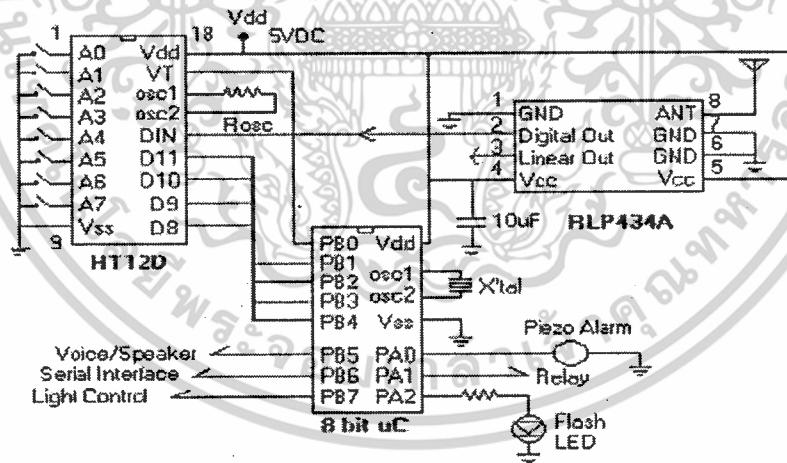
Symbol	Parameter	Conditions	Min	Typ	Max	
Vcc	Operating supply voltage		3.3	5.0V	6.0	V
I <sub>ot</sub>	Operating Current		-	4.5		mA
V <sub>data</sub>	Data Out	I <sub>data</sub> = +200 uA ( High )	Vcc-0.5	-	Vcc	V
		I <sub>data</sub> = -10 uA ( Low )	-	-	0.3	V

Characteristics	SYM	Min	Typ	Max	Unit
Operation Radio Frequency	FC	315, 418 and 433.92			MHz
Sensitivity	Pr <sub>ef</sub>		-110		dBm
Channel Width			~500		Khz
Noise Equivalent BW			4		Khz
Receiver Turn On Time			5		ms
Operation Temperature	Top	-20	-	80	C
Baseboard Data Rate			4.8		KHz

**Application Circuit :**

Typical RF Receiver using HT12D-1SDIP, a Binary 12 bit Decoder with 8 bit uC HT48RXX from Holtek Semiconductor Inc.



ดาตาชีท (Datasheet) ตัวรับสัญญาณความถี่วิทยุไอซี TLP 434A

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้