สำนักหอสมุดกลาง พระจอมเกล้าลาดกระบัง

รถสำรวจติดกล้องไร้สายบังคับผ่านอินเตอร์เน็ต

OBSERVER CAR WITH WIRELESS CAMERA CONTROLLED VIA INTERNET

โดย

นายมุขพล วัฒนะเจริญ

นายศุภเดช เจริญอภิญโญ

นายสืบตระกูล เจียมจรรยา

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต

สาขาวิศวกรรมโทรคมนาคม

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2546

เลขหมู่.....
เลขทะเบียน.....54936
วัน,เดือน,ปี.....1 เม.ย. 2548

.b.....
.i.....

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รถสำรวจติดกล้องไร้สายบังคับผ่านอินเตอร์เน็ต

OBSERVER CAR WITH WIRELESS CAMERA CONTROLLED VIA INTERNET

โดย

นายมุขพล วัฒนะเจริญ 43010339

นายศุภเดช เจริญอภิญโญ 43010438

นายลือตระกูล เจียมจรรยา 43010472

อาจารย์ที่ปรึกษา

รศ.ดร. กอบชัย เดชหาญ

อ. นภัทร สระเอี่ยม

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต

สาขาวิศวกรรมโทรคมนาคม

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2546

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ปริญญานิพนธ์ปีการศึกษา 2546

ภาควิชาวิศวกรรมโทรคมนาคม

คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เรื่อง **รถสำรวจติดกล้องไร้สายบังคับผ่านอินเตอร์เน็ต**

OBSERVER CAR WITH WIRELESS CAMERA CONTROLLED VIA INTERNET

ผู้จัดทำ

1. นายมุขพล วัฒนะเจริญ **43010339**
2. นายศุภเดช เจริญอภิญโญ **43010438**
3. นายสืบตระกูล เจียมจรรยา **43010472**

อาจารย์ที่ปรึกษา

(รศ.ดร. กอบชัย เดชหาญ)

อาจารย์ที่ปรึกษา

(อ. นภัทร สระเอี่ยม)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รถสำรวจติดกล้องไร้สายบังคับผ่านอินเตอร์เน็ต

OBSERVER CAR WITH WIRELESS CAMERA CONTROLLED VIA INTERNET

โดย นายมุขพล วัฒนะเจริญ 43010339

นายศุภเดช เจริญอภิญโญ 43010438

นายสืบตระกูล เจียมจรรยา 43010472

อาจารย์ที่ปรึกษา รศ.ดร. กอบชัย เดชหาญ

อ. นภัทร สระเอี่ยม

บทคัดย่อ

ในปัจจุบัน ระบบสื่อสารได้พัฒนาอย่างรวดเร็วจนเข้าสู่ยุคของการสื่อสารมัลติมีเดีย ภาพ เสียง และข้อมูลข่าวสารสามารถส่งผ่านถึงกันได้ทั่วโลกในเวลาอันสั้นโดยใช้ระบบอินเตอร์เน็ต โครงงานนี้เป็นการประยุกต์การส่งข้อมูลผ่านระบบอินเตอร์เน็ต โดยทำการส่งข้อมูลภาพจากกล้องที่ติดอยู่กับตัวรถแบบไร้สายเข้ากับเครื่องรับส่ง จากนั้นทำการเชื่อมโยงข้อมูลเข้ากับคอมพิวเตอร์ที่ต่อระบบอินเตอร์เน็ต ข้อมูลภาพจะส่งผ่านไปยังเครื่องคอมพิวเตอร์แม่ข่าย จากนั้นจะถูกส่งผ่านระบบอินเตอร์เน็ตไปยังผู้ใช้งาน ผู้ใช้งานจะสามารถเห็นข้อมูลภาพและทำการควบคุมรถไปยังทิศทางต่างๆ ที่ต้องการ หรือควบคุมกล้องให้หมุนไปยังทิศทางต่างๆ ได้

ABSTRACT

At present, the communication systems have been rapid developed till enter into the generation of multimedia communication. Video, audio and information can be transmitted around the world in short time. This project is an application from internet communication by sending video signal from wireless camera that has been attached with observer car to transceiver, then interfacing video signal with computer that has been connected with internet system. Video signal will be transmitted to the server, then it will be transmitted via internet system to user. User can see the video signal and control the observer car in any direction, and user can turn the camera in any direction too.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ

หน้า

บทที่ 1 บทนำ 1

บทที่ 2 ทฤษฎีและหลักการ 3

2.1 การมอดูเลตทางขนาด 3

2.1.1 รูปร่างของสัญญาณเอเอ็ม 3

2.1.2 การมอดูเลตแอมปลิจูดแบบแถบข้างคู่ขจัดคลื่นพาห์ 5

2.1.3 การสร้างสัญญาณมอดูเลตทางขนาด 7

2.1.4 การดีมอดูเลตสัญญาณที่มีการมอดูเลตทางขนาด 11

2.1.5 กำลังส่งที่เสียไปกับคลื่นพาห์ในการส่งสัญญาณเอเอ็ม 14

2.2 องค์ประกอบของสัญญาณภาพ 16

2.2.1 สัญญาณลูมีแนนซ์ 16

2.2.2 สัญญาณควบคุมตำแหน่งภาพ 16

2.2.3 สัญญาณโครมา 17

2.3 มาตรฐาน RS232 (EIA232) 19

2.3.1 การจัดขาของอุปกรณ์ 19

2.3.2 การเชื่อมต่อแบบ Null modem 21

2.3.3 ระดับแรงดัน 21

2.3.4 ช่วงเวลาของสัญญาณ 22

2.4 เซอร์โวมอเตอร์ 22

2.5 TCP/IP, OSI Model กับระบบเครือข่าย 24

2.5.1 OSI Model 24

2.5.2 การอ้างอิงอุปกรณ์ในเครือข่าย 32

2.5.3 Data Packet 33

2.5.4 Encapsulation 34

2.5.5 IP Datagram 35

2.5.6 โปรโตคอล TCP 36

2.5.7 โปรโตคอล UDP 37

2.5.8 โปรโตคอล IP 38

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หน้า

บทที่ 3 การคำนวณและการสร้าง 39

3.1 ภาครับ - ส่งสัญญาณภาพ 39

3.2 ภาครับ - ส่งสัญญาณควบคุม 41

3.2.1 โปรแกรมในการติดต่อกับเครื่องส่งสัญญาณความถี่วิทยุ 41

3.2.2 เครื่องส่งสัญญาณควบคุม 42

3.2.3 เครื่องรับสัญญาณควบคุม 47

3.3 ส่วนโปรแกรมติดต่อผ่านอินเตอร์เน็ต 50

3.3.1 ส่วนส่งข้อมูลภาพ 50

3.3.2 ส่วนควบคุมผ่านอินเตอร์เน็ต 50

3.3.3 โปรแกรมในการติดต่อผ่านอินเตอร์เน็ต 51

3.4 ลักษณะของชิ้นงาน 56

บทที่ 4 การทดลองและผลการทดลอง 58

4.1 ภาควงจรเครื่องส่งภาพ 58

4.2 ภาคส่งสัญญาณควบคุม 61

4.3 ภาครับสัญญาณควบคุม 64

4.4 ภาคติดต่อกับเครือข่ายอินเตอร์เน็ต 65

บทที่ 5 บทวิจารณ์และบทสรุป 70

ภาคผนวก

หนังสืออ้างอิง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูปภาพ

หน้า

รูปที่ 1.1	Block Diagram ของโครงงาน	1
รูปที่ 2.1	ตัวอย่างแสดงสเปกตรัมของสัญญาณอนาลอกที่มีความถี่สูง	3
รูปที่ 2.2	แสดงการมอดูเลตที่ค่าแรงดันไฟตรงต่าง ๆ กัน	4
รูปที่ 2.3	ตัวอย่างของสัญญาณ DSBSC	5
รูปที่ 2.4	สเปกตรัมของสัญญาณข่าวสาร	6
รูปที่ 2.5	สเปกตรัมของสัญญาณ DSBSC	6
รูปที่ 2.6	ริงมอดูเลเตอร์ และวงจรสมมูล	7
รูปที่ 2.7	วงจรริงมอดูเลเตอร์เชิงปฏิบัติ	8
รูปที่ 2.8	วงจรไม่เชิงเส้น	9
รูปที่ 2.9	วงจรมอดูเลเตอร์แบบสมดุล	10
รูปที่ 2.10	ระบบแปลงความถี่ของสัญญาณที่ใช้วงจรคูณ	11
รูปที่ 2.11	ตัวอย่างวงจรเอ็นเวลโลปดีเทคชัน	12
รูปที่ 2.12	การดีเทคสัญญาณที่มีค่า RC ต่างๆกัน	12
รูปที่ 2.13	วงจรเครื่องรับแบบคอสแทส	13
รูปที่ 2.14	แสดงสัญญาณลูมิแนนซ์กับสัญญาณภาพสีรวม	16
รูปที่ 2.15	แสดงรูปร่างของสัญญาณซิงค์ที่อยู่ในสัญญาณโทรทัศน์	17
รูปที่ 2.16	ลักษณะการสร้างสัญญาณโครมา	18
รูปที่ 2.17	การจัดขาทางด้าน DTE	19
รูปที่ 2.18	การจัดขาทางด้าน DCE	20
รูปที่ 2.19	การต่อสาย null modem	21
รูปที่ 2.20	แสดงระดับแรงดันในมาตรฐาน RS232	21
รูปที่ 2.21	ตัวอย่างเซอร์โวมอเตอร์	22
รูปที่ 2.22	รายละเอียดภายในของเซอร์โวมอเตอร์	23
รูปที่ 2.23	ตัวอย่างการควบคุมองศาการหมุนของเซอร์โวมอเตอร์	24
รูปที่ 2.24	โครงสร้างของ OSI 7-Layer Reference Model	25
รูปที่ 2.25	การรับส่งข้อมูลของ OSI 7-Layer Model	26
รูปที่ 2.26	การแบ่งกลุ่มของ OSI 7-Layer Model	27
รูปที่ 2.27	การรับส่งข้อมูลแต่ละชั้นของ OSI 7-Layer Model	30
รูปที่ 2.28	โปรโตคอล TCP/IP เมื่อเทียบกับ OSI 7-Layer Model	31
รูปที่ 2.29	ตัวอย่าง Encapsulation ของข้อมูล FTP เทียบกับ TCP/IP Layer	34

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หน้า

รูปที่ 2.30 รูปของ IP Datagram ประกอบด้วยส่วน Header และ Payload 35
รูปที่ 2.31 รูปแบบของ TCP Packet ซึ่งมีฟิลด์ Acknowledgement Number และข้อมูล Checksum 37
รูปที่ 2.32 รูปแบบของ UDP Packet จะมีฟิลด์ข้อมูลส่วน Header น้อยมากและไม่มีส่วนตรวจสอบข้อมูล 38
รูปที่ 2.33 การติดต่อกันระหว่างโปรโตคอลในชั้นของ Host-to-Host Layer และ Internetwork Layer 38
รูปที่ 3.1 Block Diagram ของภาครับ – ส่งสัญญาณภาพ 39
รูปที่ 3.2 Block Diagram ของภาครับ – ส่งสัญญาณควบคุม 39
รูปที่ 3.3 รูปวงจรเครื่องส่งภาพไร้สาย 40
รูปที่ 3.4 Flow Chart ของโปรแกรมติดต่อกับเครื่องส่ง 41
รูปที่ 3.5 Block Diagram ของเครื่องส่งสัญญาณควบคุม 42
รูปที่ 3.6 การจัดเรียงขาของ IC MAX232 42
รูปที่ 3.7 วงจรใช้งานของ IC MAX232 43
รูปที่ 3.8 Flow Chart ของโปรแกรมแปลงรหัสสัญญาณ 44
รูปที่ 3.9 การจัดขาของ IC PT8A977P 45
รูปที่ 3.10 Block Diagram ของ IC PT8A977P 45
รูปที่ 3.11 แสดงการเข้ารหัสสัญญาณของ IC PT8A977P 46
รูปที่ 3.12 Block Diagram ของเครื่องรับสัญญาณควบคุม 47
รูปที่ 3.13 การจัดขาของ IC PT8A978P 47
รูปที่ 3.14 Block Diagram ของ IC PT8A978P 48
รูปที่ 3.15 Flow Chart ของโปรแกรมควบคุมมอเตอร์ (89C2051) 48
รูปที่ 3.16 การจัดขาของ IC L293D 49
รูปที่ 3.17 การต่อวงจรใช้งานของ IC L293D 49
รูปที่ 3.18 ขั้นตอนการเชื่อมต่อระหว่างเครื่อง Server และเครื่อง Client 50
รูปที่ 3.19 Flow Chart ของโปรแกรมฝั่ง Server 51
รูปที่ 3.20 Flow Chart ของโปรแกรมฝั่ง Client 52
รูปที่ 3.21 ลักษณะของหน้าต่างโปรแกรมฝั่ง Server 53
รูปที่ 3.22 ลักษณะของหน้าต่างโปรแกรมฝั่ง Client 53
รูปที่ 3.23 วงจรรวมเครื่องส่งสัญญาณควบคุม 54
รูปที่ 3.24 วงจรรวมเครื่องรับสัญญาณควบคุม 55
รูปที่ 3.25 ลักษณะชิ้นงานภายในกล่องของเครื่องส่งสัญญาณควบคุม 56
รูปที่ 3.26 ลักษณะชิ้นงานภายในกล่องของเครื่องรับสัญญาณควบคุม 56

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หน้า

รูปที่ 3.27 ลักษณะชิ้นงานส่วนวงจรส่งสัญญาณภาพไร้สาย 56
รูปที่ 3.28 ลักษณะชิ้นงานส่วนวงจรไมโครคอนโทรลเลอร์ และวงจรขับมอเตอร์ 57
รูปที่ 3.29 ลักษณะชิ้นงานส่วนชุดควบคุมกล้อง 57
รูปที่ 3.30 ลักษณะชิ้นงานรวม 57
รูปที่ 4.1 สัญญาณภาพที่ได้จากกล้อง 58
รูปที่ 4.2 สัญญาณภาพที่ผ่านการขยายแล้ว 58
รูปที่ 4.3 สัญญาณคลื่นพาห์ 66.667 MHz 59
รูปที่ 4.4 สเปกตรัมของสัญญาณคลื่นพาห์ 66.667 MHz 59
รูปที่ 4.5 สเปกตรัมของสัญญาณ AM เมื่ออินพุตเป็นสัญญาณภาพจากกล้อง 60
รูปที่ 4.6 ภาพที่รับได้ที่เครื่องคอมพิวเตอร์ที่ระยะ 5 เมตร 60
รูปที่ 4.7 สัญญาณจากโมดูลตัวส่ง เมื่อทำการกดปุ่มหมายเลข 1 61
รูปที่ 4.8 สัญญาณจากโมดูลตัวส่ง เมื่อทำการกดปุ่มหมายเลข 2 61
รูปที่ 4.9 สัญญาณจากโมดูลตัวส่ง เมื่อทำการกดปุ่มหมายเลข 3 62
รูปที่ 4.10 สัญญาณจากโมดูลตัวส่ง เมื่อทำการกดปุ่มหมายเลข 4 62
รูปที่ 4.11 สัญญาณจากโมดูลตัวส่ง เมื่อทำการกดปุ่มหมายเลข 6 62
รูปที่ 4.12 สัญญาณจากโมดูลตัวส่ง เมื่อทำการกดปุ่มหมายเลข 7 63
รูปที่ 4.13 สัญญาณจากโมดูลตัวส่ง เมื่อทำการกดปุ่มหมายเลข 8 63
รูปที่ 4.14 สัญญาณจากโมดูลตัวส่ง เมื่อทำการกดปุ่มหมายเลข 9 63
รูปที่ 4.15 สัญญาณจากโมดูลตัวส่ง เมื่อทำการกดปุ่มหมายเลข 5 หรือ ปุ่มอื่นๆ 64
รูปที่ 4.16 โปรแกรมฝั่ง Server ขณะรอรับการติดต่อ 66
รูปที่ 4.17 โปรแกรมฝั่ง Client 66
รูปที่ 4.18 กล่องข้อความแสดงการเชื่อมต่อของ Winsock Control 67
รูปที่ 4.19 กล่องข้อความสำหรับใส่รหัสผ่าน 67
รูปที่ 4.20 โปรแกรมฝั่ง Client หลังจากเชื่อมต่อระหว่างเครื่องได้แล้ว 67
รูปที่ 4.21 กล่องข้อความแสดงการเปิดพอร์ต 67
รูปที่ 4.22 โปรแกรมฝั่ง Client หลังจากเปิดพอร์ตและรับภาพได้แล้ว 68
รูปที่ 4.23 โปรแกรมฝั่ง Server หลังจากเชื่อมต่อและเปิดพอร์ตแล้ว 68
รูปที่ 4.24 โปรแกรมฝั่ง Server ขณะทำการเก็บภาพ 69
รูปที่ 4.25 โปรแกรมฝั่ง Client ขณะทำการเก็บภาพ 69
รูปที่ 4.26 โปรแกรมขณะทำการบันทึกเก็บภาพ 69

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญตาราง

หน้า

ตารางที่ 2.1 การระบุโปรโตคอลที่ทำงานใน Layer ข้างบนซึ่งผนึกลงมาใน IP Datagram 36
ตารางที่ 3.1 รหัสสัญญาณต่างๆ ในการเข้ารหัสของ IC PT8A977P 46
ตารางที่ 4.1 การทำงานของรถและมอเตอร์หมุนกล้องเมื่อกดปุ่มจากคีย์บอร์ด 64

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่1
บทนำ

ในปัจจุบันการใช้งานเทคโนโลยีมีบทบาทมากขึ้นในชีวิตประจำวัน ในแต่ละวันเราจำเป็นต้องเรียนรู้ให้ทันกับเทคโนโลยีใหม่ๆ อยู่ตลอดเวลา อีกทั้งปัจจุบันยังเป็นโลกแห่งข้อมูลข่าวสารที่กว้างใหญ่ไม่มีที่สิ้นสุด ดังนั้นการที่เราจะใช้ชีวิตอยู่ได้นั้นจึงจำเป็นที่จะต้องเรียนรู้เพื่อที่จะก้าวตามเทคโนโลยีให้ทัน ไม่เพียงแค่นั้น ถ้าเราสามารถพัฒนาความรู้ความสามารถให้มากขึ้น สักวันหนึ่งเราอาจเป็นผู้สร้างเทคโนโลยีเพื่ออำนวยความสะดวก และพัฒนาศักยภาพในด้านต่างๆ ที่จะเป็นประโยชน์ต่อไป

หุ่นยนต์ก็เป็นเทคโนโลยีหนึ่งที่พัฒนาอย่างรวดเร็วในช่วง 2 - 3 ปีที่ผ่านมา เทคโนโลยีหุ่นยนต์ได้เข้ามาแทนมนุษย์ในหลายๆ ด้าน ทั้งในด้านอุตสาหกรรมการผลิต การเกษตร การประกอบรถยนต์ แม้กระทั่งเป็นเครื่องใช้ในบ้าน เช่น เครื่องดูดฝุ่นอัจฉริยะ ตู้เย็นอัจฉริยะ และในด้านการบันเทิง เช่น หุ่นยนต์ไอโบ้ (สุนัขอิเล็กทรอนิกส์) เป็นต้น

ในโครงงานนี้เป็นการนำเสนอรูปแบบหนึ่งในการใช้เทคโนโลยีมาอำนวยความสะดวกให้กับมนุษย์ โดยการใช้รถที่ติดกล้องเพื่อจับภาพ ส่งมายังคอมพิวเตอร์แบบไร้สาย เราสามารถควบคุมรถได้จากการกดปุ่มควบคุมที่คีย์บอร์ด และสามารถหมุนกล้องเพื่อดูภาพในทิศทางต่างๆ ได้จากการกดปุ่มควบคุมของคีย์บอร์ดเช่นกัน โดยสามารถควบคุมได้ทั้งทางด้าน Server และทางด้าน Client ซึ่งมีการเชื่อมต่อผ่านระบบอินเตอร์เน็ต ซึ่งบล็อกไดอะแกรมของโครงงานนี้แสดงดังรูปที่ 1.1

รูปที่ 1.1 Block Diagram ของโครงงาน

สัญญาณภาพจากรถจะถูกส่งแบบไร้สายไปยังเครื่องรับสัญญาณภาพที่ต่ออยู่กับเครื่องคอมพิวเตอร์แม่ข่าย จากนั้นจะถูกส่งไปยังผู้ใช้โดยผ่านเครือข่ายอินเตอร์เน็ต ส่วนทางผู้ใช้ก็จะสามารถควบคุมทิศทางของรถและกล้องได้ โดยสัญญาณควบคุมจากผู้ใช้จะผ่านกลับมาในทิศทางที่กลับกัน โดยการส่งสัญญาณแบบไร้สายนี้จะใช้การมอดูเลตสัญญาณแบบ Amplitude Modulation : AM ซึ่งจากการที่เราสามารถเห็นภาพจากกล้องที่คอมพิวเตอร์และสามารถควบคุมรถได้นั้น เราจึงสามารถประยุกต์แนวคิดของโครงงานนี้ไปสร้างเป็นรถถ่ายทอดโทรทัศน์บังคับระยะไกล หรือใช้ประโยชน์ในงานสำรวจต่อไปได้ในอนาคต

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาครับ – ส่งสัญญาณภาพ

- ภาครับสัญญาณภาพ ใช้การ์ด TV Tuner เป็นตัวรับสัญญาณ และแสดงผลที่หน้าจอของเครื่องคอมพิวเตอร์
- ภาคส่งสัญญาณภาพ ใช้วงจรขยายสัญญาณภาพจากกล้องแล้วจึงนำสัญญาณมามอดูเลตแบบ AM โดยใช้ความถี่คลื่นพาห์ที่ 66.667 MHz

ภาครับ – ส่งสัญญาณควบคุม

- ภาครับ – ส่งสัญญาณควบคุม ใช้โมดูลตัวรับ และโมดูลตัวส่ง ของรถกระป๋อง โดยในภาคส่งสัญญาณควบคุมจะมีการต่ออินเตอร์เฟสกับเครื่องคอมพิวเตอร์ เพื่อให้สามารถควบคุมได้โดยตรงจากโปรแกรมที่สร้างขึ้น

ภาคควบคุมรถและกล้อง

- วงจรไมโครคอนโทรลเลอร์ ทำหน้าที่รับสัญญาณมาจากโมดูลตัวรับ แล้วแปลงรหัสสัญญาณไปเป็นสัญญาณในการควบคุมมอเตอร์
- ภาคควบคุมรถ จะใช้ IC Drive เบอร์ L293D เป็นตัวขับ DC Motor ให้รถเคลื่อนที่ไปในทิศทางต่างๆได้
- ภาคควบคุมกล้อง จะใช้ Servo Motor เป็นตัวหมุนทิศทางของกล้อง โดยกล้องจะทำการถ่ายภาพเป็นแบบ Real-time

ภาคติดต่อกับเครือข่ายอินเตอร์เน็ตและพอร์ตอนุกรม

- โปรแกรมส่วนที่ติดต่อกับพอร์ตอนุกรมถูกสร้างขึ้นโดยใช้ Visual Basic 6 ซึ่งใช้ Control ที่มีชื่อว่า Mscomm โดยใช้อัตราบอด (Baud Rate) เท่ากับ 9600 และมีข้อมูล 8 บิต 1 บิตเริ่มต้น 1 บิตสิ้นสุด และไม่ใช้บิตพาริตี้ (Parity Bit) ซึ่งทำการส่งข้อมูลอนุกรมออกทางพอร์ต Com 1
- โปรแกรมส่วนที่ติดต่อกับเครือข่ายอินเตอร์เน็ตจะใช้ Winsock Control ในการเชื่อมต่อและรับ – ส่งข้อมูลผ่านเครือข่ายอินเตอร์เน็ต โดยอ้างอิงตำแหน่งเครื่องคอมพิวเตอร์ผ่านทางหมายเลข IP Address
- ส่วนแสดงภาพจากกล้องผ่านเครือข่ายอินเตอร์เน็ต จะใช้ Control NDNM3 เป็นตัวจัดการ โดยตัว Control นี้สามารถดึงเอาทรัพยากรของโปรแกรม Windows Netmeeting ในการส่งข้อมูลภาพแบบ Streaming มาใช้ได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 2
ทฤษฎีและหลักการ

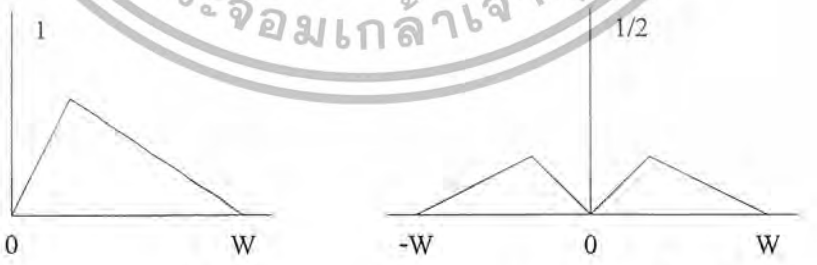
ในการสื่อสาร ถ้าสัญญาณถูกส่งไปยังเครื่องรับโดยไม่ผ่านการมอดูเลต (Modulate) กล่าวคือ ไม่มีการเคลื่อนย้ายความถี่ของข้อมูลไปจากเดิมเลย การสื่อสารในลักษณะเช่นนี้จะเรียกว่า การสื่อสารในเบสแบนด์ (Baseband communication) ซึ่งหมายถึงการสื่อสารที่มีย่านความถี่อยู่ในช่วงความถี่ของสัญญาณเดิมดังกล่าว แต่ถ้าการส่งสัญญาณนั้นผ่านการมอดูเลตสัญญาณแล้ว เราจะเรียกการสื่อสารชนิดนั้นว่า การสื่อสารโดยคลื่นพาห์ (Carrier communication)

เนื่องจากสัญญาณในเบสแบนด์นั้นขาดความเหมาะสมที่จะใช้ส่งผ่านช่องการสื่อสารไปยังเครื่องรับ ด้วยสาเหตุสำคัญคือ สัญญาณนั้นมีสเปกตรัมของความถี่อยู่ในย่านที่ต่างจากย่านความถี่ของช่องการสื่อสารที่มีอยู่ ดังนั้นเพื่อที่จะสามารถส่งสัญญาณนั้นผ่านช่องการสื่อสารได้ จึงต้องมีการอาศัยสัญญาณหนึ่ง ซึ่งเรียกว่า คลื่นพาห์ (Carrier) เป็นสื่อช่วยเคลื่อนย้ายสเปกตรัมของสัญญาณนั้น และกระบวนการที่ทำการเคลื่อนย้ายสเปกตรัมนี้เรียกว่า การมอดูเลต ซึ่งในที่นี้จะกล่าวถึงเฉพาะ การมอดูเลตทางขนาด (Amplitude Modulation : AM) เท่านั้น

2.1 การมอดูเลตทางขนาด (Amplitude Modulation)

2.1.1 รูปร่างของสัญญาณเอเอ็ม

โดยทั่วไปแล้วสัญญาณอนาลอกเบสแบนด์ที่ทำการส่งนั้นจะมีค่าความถี่สูงที่มีค่าจำกัด เช่น สัญญาณเสียงที่หูคนเรารับฟังได้อาจจะสูงสุดเพียง 16 kHz และสัญญาณภาพในระบบโทรทัศน์มีความถี่สูงสุดประมาณ 5 MHz ดังนั้นถ้าเรากำหนดให้ความถี่สูงสุดของสัญญาณมีค่าเป็น W Hz เมื่อเราเขียนสเปกตรัมของสัญญาณนั้นแบบแสดงความถี่ด้านบวกด้านเดียว และแสดงความถี่ทั้ง 2 ด้าน จะได้ดังรูปที่ 2.1 ก และ ข ตามลำดับ



(ก) แสดงความถี่ด้านเดียว (ข) แสดงความถี่สองด้าน

รูปที่ 2.1 ตัวอย่างแสดงสเปกตรัมของสัญญาณอนาลอกที่มีความถี่สูง

ในกรณีแสดงความถี่ทั้งสองด้านนั้นสเปกตรัมทางด้านบวกและทางด้านลบจะสมมาตรกัน และมีขนาดเป็นครึ่งหนึ่งของสเปกตรัมที่แสดงความถี่บวกเพียงด้านเดียว การมอดูเลตแบบ AM ที่ใช้ใน

เอกสารนี้เป็นลิขสิทธิ์ของมหาวิทยาลัยเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

วิทยุ AM โดยทั่วไปจะเป็นการสร้างสัญญาณที่มีลักษณะเหมือนกับสัญญาณเบสแบนด์ (Baseband) กล่าวคือเหมือนกับการนำสัญญาณเบสแบนด์ไปขี่อยู่บนสัญญาณคลื่นพาห์ ทำให้ยอดคลื่นของคลื่นพาห์หรือเอ็นเวลโลป (Envelope) เปลี่ยนไปตามสัญญาณเบสแบนด์นั้น ในสภาพดังกล่าวนี้เราจะสามารถดีเทค (Detect) สัญญาณกลับมาได้โดยใช้วงจรดีเทคแบบง่าย ๆ ซึ่งการดีเทคยอดของคลื่นพาห์นี้ เราเรียกว่า เอ็นเวลโลปดีเทคชั่น (envelope detection) ซึ่งเป็นวิธีการดีเทคสัญญาณ AM วิธีหนึ่งสัญญาณ AM เมื่อเขียนเป็นฟังก์ชันของเวลาจะเป็นดังนี้

$$V_c(t) = [A_c + m(t)] \cos(\omega_c t) \qquad (2.1)$$

โดย A_c เป็นระดับแรงดันไฟตรง และ $m(t)$ เป็นสัญญาณเบสแบนด์ซึ่ง

$$m(t) = A_m \cos(\omega_m t) \qquad (2.2)$$

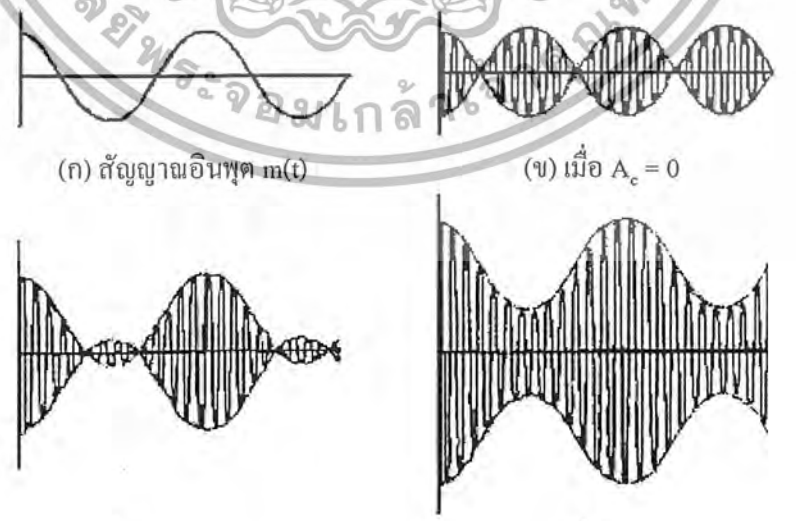
และจากสมการที่ (2.1) เราจะนิยมเขียนแสดงได้เป็น

$$V_c(t) = A_c\{1 + \mu\cos(\omega_m t)\}\cos(\omega_c t) \qquad (2.3)$$

โดย μ จะถูกเรียกว่า มอดูเลชันอินเด็กซ์ของสัญญาณ AM (modulation index) ซึ่งเป็นตัวบ่งชี้ว่าสัญญาณคลื่นพาห์ถูกทำให้มีการเปลี่ยนระดับไปอย่างไร สามารถหาได้โดย

$$\mu = \frac{A_m}{A_c} \times 100\% = \frac{V_{max} - V_{min}}{V_{max} + V_{min}} \times 100\% \quad ; 0 < \mu < 100 \qquad (2.4)$$

$$\text{หรือ} \quad \mu = \frac{A_m}{A_c} = \frac{V_{max} - V_{min}}{V_{max} + V_{min}} \quad ; 0 < \mu < 1 \qquad (2.5)$$



(ก) สัญญาณอินพุต $m(t)$ (ข) เมื่อ $A_c = 0$

(ค) เมื่อ $0 < A_c < A_m$ (ค) เมื่อ $A_c > A_m$

รูปที่ 2.2 แสดงการมอดูเลตที่ค่าแรงดันไฟตรงต่าง ๆ กัน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

แต่อย่างไรก็ตามในทางปฏิบัตินั้น คำว่า มอดูเลชันอินเด็กซ์นั้นมักจะหมายถึงค่า m(t) ที่มีขนาดสูงที่สุด ซึ่งสัญญาณ AM ที่จะสามารถดีเทคได้ด้วยวงจรตรวจจับกรอบสัญญาณ(วิธีเอ็นเวลโลปดีเทคชัน) จะต้องมี $\mu < 1$ เสมอ ซึ่งหากค่า μ มีค่าสูงกว่า 1 ก็จะเกิดการโอเวอร์มอดูเลชัน (Overmodulation) ทำให้วงจรตรวจจับกรอบสัญญาณจะดีเทคได้แต่เพียงเอ็นเวลโลปด้านบวกด้านเดียว ซึ่งจะทำให้ไม่สามารถดีเทคสัญญาณเบสแบนด์เดิมกลับมาได้อย่างสมบูรณ์

2.1.2 การมอดูเลตแอมปลิจูดแบบแถบข้างคู่ขจัดคลื่นพาห์

การมอดูเลตแบบแถบข้างคู่ขจัดคลื่นพาห์ หรือเรียกว่าการมอดูเลตแบบ DSBSC (Double-sideband suppress carrier modulation) ตามความหมายก็คือ การมอดูเลตที่ให้สัญญาณที่มีไซด์แบนด์ทั้งสองด้าน และสกัดเอาคลื่นพาห์ออกไป ซึ่งวัตถุประสงค์หลักของการมอดูเลตแบบนี้ ก็คือ ต้องการลดกำลังส่งในส่วนที่เป็นสัญญาณคลื่นพาห์ออกจากสัญญาณ AM เพื่อเพิ่มประสิทธิภาพด้านกำลังส่ง เมื่อทำการสกัดสัญญาณคลื่นพาห์ออก สัญญาณ DSBSC ที่ได้จะเขียนได้ดังนี้

$$V_{DSBSC}(t) = m(t)\cos(\omega_c t) \qquad (2.6)$$

ตามลักษณะของสมการ (2.6) ค่ายอดคลื่นของสัญญาณหรือเอ็นเวลโลปที่ได้จะเป็น $|m(t)|$ แทนที่จะเป็นไปตามรูปแบบของ m(t) ซึ่งในสภาพเช่นนี้จะไม่สามารถใช้วิธีเอ็นเวลโลปดีเทคชันได้ รูปที่ 2.3 แสดงตัวอย่างของสัญญาณ DSBSC นี้

(ก) คลื่นพาห์ (ข) สัญญาณเบสแบนด์ (ค) สัญญาณ DSBSC

รูปที่ 2.3 ตัวอย่างของสัญญาณ DSBSC

เพื่อที่จะทำความเข้าใจถึงความเปลี่ยนแปลงที่เกิดขึ้นในโดเมนความถี่ จะสมมุติให้ $M(\omega)$ คือฟังก์ชันสเปกตรัมของ m(t) กล่าวคือ

$$m(t) \leftrightarrow M(\omega) \qquad (2.7)$$

โดยอาศัยคุณสมบัติของการแปลงฟูเรียร์ จะได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

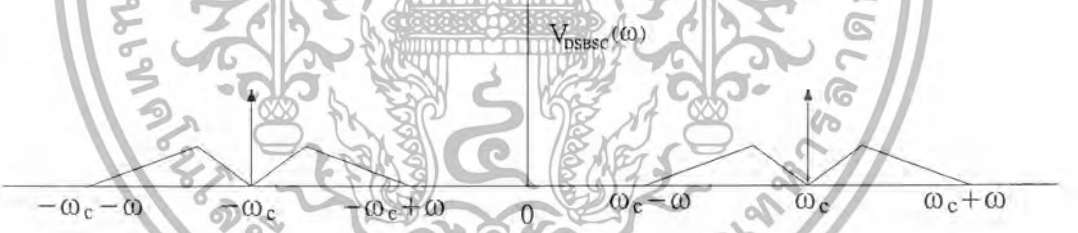
$$m(t)\cos(\omega_c t) \leftrightarrow \frac{1}{2}\left[M(\omega_c+\omega)+M(\omega_c-\omega)\right] \qquad (2.8)$$

นั่นคือสเปกตรัมของสัญญาณ DSBSC ก็จะมีค่าดังนี้

$$V_{DSBSC}(\omega) = \frac{1}{2}\left[M(\omega_c+\omega)+M(\omega_c-\omega)\right] \qquad (2.9)$$

ซึ่งรูปร่างของสเปกตรัมของสัญญาณ AM แบบ DSBSC นี้ ก็จะได้ดังที่แสดงในรูปที่ 2.4

รูปที่ 2.4 สเปกตรัมของสัญญาณข่าวสาร



รูปที่ 2.5 สเปกตรัมของสัญญาณ DSBSC

ถ้าแบนด์วิดท์ของ M(ω) มีค่าเป็น W เฮิรตซ์ หรือ ω เรเดียนต่อวินาที ($\omega = 2\pi W$) เราก็จะกล่าวได้ว่า $V_{DSB\text{-}SC}(\omega)$ มีแบนด์วิทด์เป็น 2ω เรเดียนต่อวินาที ดังแสดงในรูปที่ 2.5 และเป็นที่น่าสังเกตว่าค่าความถี่กึ่งกลางแบนด์ของสัญญาณเอเอ็มนี้อยู่ที่ความถี่คลื่นพาห์ คือ ω_c

จะพบว่าแกนตั้งที่ ω_c จะมีสเปกตรัมสองส่วนที่สมมาตรกัน ส่วนของสเปกตรัมที่อยู่ที่ความถี่สูงกว่า และอยู่ต่ำกว่าความถี่ ω_c นั้นจะมีชื่อว่า แถบข้างส่วนบน หรือ ไซด์แบนด์ด้านบน (Upper sideband : USB) และแถบข้างส่วนล่าง หรือ ไซด์แบนด์ด้านล่าง (Lower sideband : LSB) ตามลำดับ และในทำนองเดียวกันบนแกนความถี่ด้านลบ ค่าความถี่ $-\omega_c$ ก็จะทำให้เกิดไซด์แบนด์ด้านบน และไซด์แบนด์ด้านล่างขึ้นเช่นเดียวกัน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

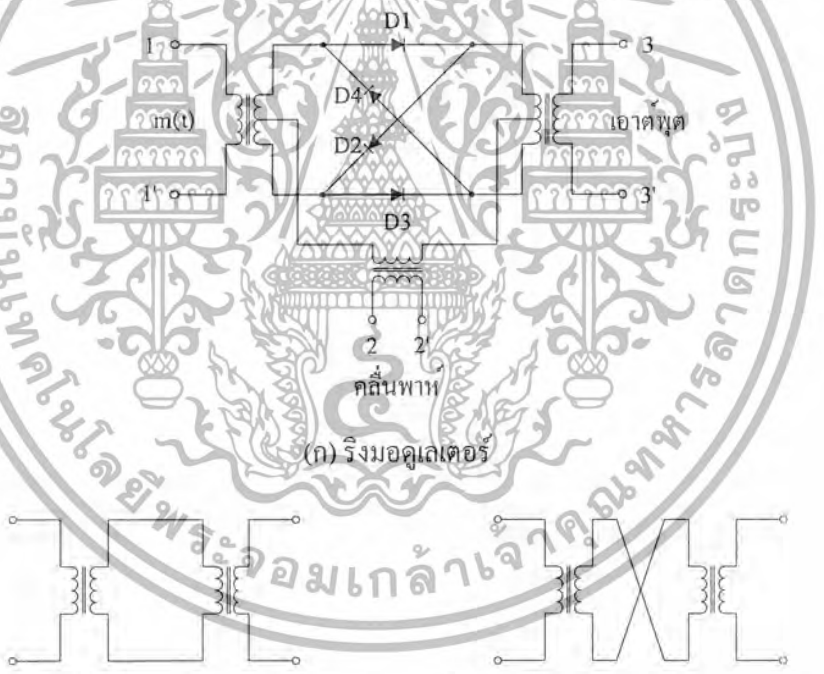
ควรสังเกตอีกด้วยว่า ถ้าสัญญาณ m(t) นั้นไม่มีส่วนประกอบเชิงความถี่ที่เป็นไฟตรง ไซด์แบนด์ทั้งสองข้างที่เกิดจากการมอดูเลตก็จะอยู่ห่างกันอย่างชัดเจน และสเปกตรัมของสัญญาณเอเอ็ม ก็จะไม่มีองค์ประกอบความถี่ที่ $\omega = \pm\omega_c$ ปรากฏ กล่าวคือ ส่วนประกอบสัญญาณที่มีความถี่ตรงกับความถี่ของคลื่นพาห์จะถูกขจัดออกไป ด้วยเหตุนี้เองทำให้การมอดูเลตแบบนี้จึงได้ชื่อว่า การมอดูเลตแอมปลิจูดแบบแถบข้างคู่ขจัดคลื่นพาห์ ซึ่งเรียกย่อว่า DSBSC นั่นเอง

2.1.3 การสร้างสัญญาณมอดูเลตทางขนาด

เนื่องจากสัญญาณเอเอ็มมีอยู่หลายรูปแบบ ดังนั้นในที่นี้จะกล่าวถึงแต่เฉพาะวิธีการสร้างสัญญาณ DSBSC เท่านั้น ซึ่งในทางปฏิบัตินั้นสามารถทำได้หลายวิธี วงจรที่ทำหน้าที่นี้คือ วงจรมอดูเลเตอร์ ซึ่งวงจรมูลฐานที่น่าสนใจมีดังนี้

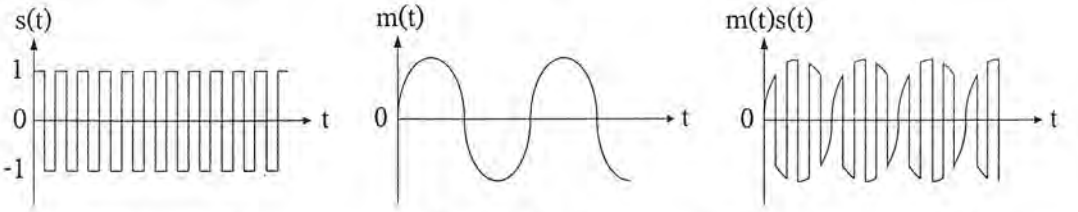
1) ริงมอดูเลเตอร์

ริงมอดูเลเตอร์ (Ring modulator) เป็นมอดูเลเตอร์แบบสวิตซ์ ซึ่งมีไดโอดต่อกัน ดังแสดงในรูปที่ 2.6



(ก) ริงมอดูเลเตอร์

(ข) วงสมมูลเมื่อคลื่นพาห์เป็นบวก (ค) วงจรสมมูลเมื่อคลื่นพาห์เป็นลบ



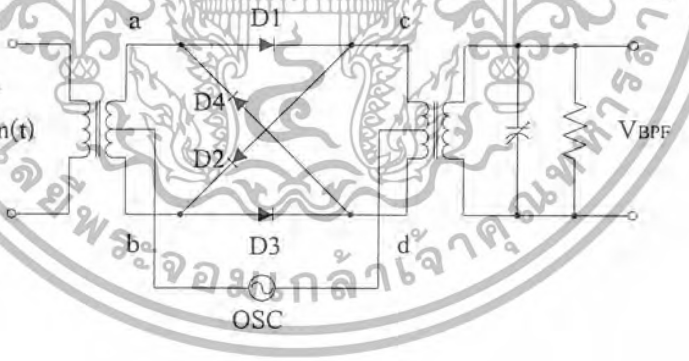
(ง) สัญญาณสวิตซ์ (จ) สัญญาณอินพุต (ฉ) สัญญาณเอาท์พุตที่ได้

รูปที่ 2.6 ริงมอดูเลเตอร์ และวงจรสมมูล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การทำงานของวงจรอธิบายได้ดังนี้ คือ เมื่อสมมุติว่าไม่มีสัญญาณ m(t) เข้ามาชั่วขณะ และให้คลื่นพาห์มีขนาดแรงพอควรถูกป้อนเข้ามาระหว่างขั้ว 2 – 2′ จะพบว่าในช่วงเวลาที่คลื่นพาห์มีสถานะเป็นบวก กล่าวคือแรงดันไฟฟ้าที่ขั้ว 2 สูงกว่าแรงดันไฟฟ้าที่ขั้ว 2′ จะพบว่าไดโอด D_1 และ D_3 จะนำกระแส แต่ D_2 และ D_4 จะตัดกระแส ดังนั้นวงจรสมมูลในสถานะเช่นนี้ วงจรสมมูลจะเป็นดังแสดงในรูปที่ 2.6 ข ครั้นพอถึงช่วงเวลาที่สถานะของคลื่นพาห์เปลี่ยนเป็นลบ กล่าวคือแรงดันไฟฟ้าที่ขั้ว 2 มีค่าต่ำกว่าแรงดันไฟฟ้าที่ขั้ว 2′ ไดโอด D_1 และ D_3 จะตัดกระแสแต่ D_2 และ D_4 จะนำกระแสทำให้วงจรสมมูลในสถานะนี้เป็นดังแสดงในรูปที่ 2.6 ค ดังนั้นเมื่อพิจารณาโดยรวม จะเห็นว่าคลื่นพาห์จะเป็นตัวคอยควบคุมการตัดต่อวงจรให้สัญญาณจากอินพุตไปปรากฏเป็นบวก หรือสลับกันที่เอาต์พุตตามสถานะของคลื่นพาห์ กล่าวคือสถานะภาพของการตัดต่อสวิตช์ จะเป็นฟังก์ชันของเวลา คือ s(t) ดังแสดงในรูปที่ 2.6 ง โดย s(t) จะมีค่าเท่ากับ +1 และ -1 เมื่อคลื่นพาห์มีสถานะของสัญญาณเป็นค่าบวกและค่าลบตามลำดับ เพราะฉะนั้นเมื่อมีสัญญาณ m(t) ตามรูปที่ 2.6 จ เข้ามา ก็จะทำให้ได้สัญญาณเอาต์พุตออกมาเป็นดังรูปที่ 2.6 ฉ สัญญาณตามรูปที่ 2.6 ฉ นี้ เมื่อผ่านวงจรกรองแถบความถี่ผ่านก็จะกลายเป็นสัญญาณ ดีเอสบี เอสซี ตามต้องการ

ในทางปฏิบัติ วงจรกรองแถบความถี่ผ่าน มักจะสร้างขึ้นด้วยวงจรแทงค์รีโซแนนซ์ (Tank resonance) ซึ่งประกอบด้วยความเหนี่ยวนำของขดลวดเอาต์พุตของหม้อแปลงไฟฟ้าขนานกับตัวเก็บประจุที่ทำให้เกิดการรีโซแนนซ์ ที่ความถี่ ω_c และมีตัวความต้านทานเป็นตัวที่ช่วยจัดค่าตัวประกอบคุณสมบัติ (Quality factor) ของวงจรให้เหมาะสม ดังนั้นวงจรที่สมบูรณ์จึงมีลักษณะง่ายๆ ดังในรูปที่ 2.7

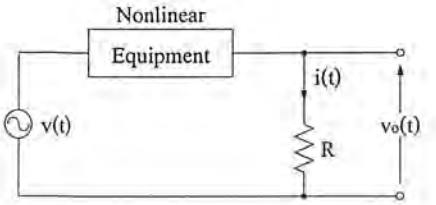


รูปที่ 2.7 วงจรริงมอดูเลเตอร์เชิงปฏิบัติ

2) มอดูเลเตอร์ไม่เชิงเส้น

วิธีการมอดูเลเตอร์ที่น่าสนใจอีกวิธีหนึ่ง คือ การใช้คุณสมบัติของอุปกรณ์ไม่เชิงเส้น (Nonlinear element) เช่น ไดโอด หรือทรานซิสเตอร์ มาต่อร่วมกับอุปกรณ์ไฟฟ้าอื่น เราจะเริ่มทำความเข้าใจด้วยการพิจารณาวงจรไม่เชิงเส้นดังแสดงในรูปที่ 2.8

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.8 วงจรไม่เชิงเส้น

กระแสของวงจรไม่เชิงเส้นทั่วไป เขียนในรูปอนุกรมกำลังของแรงดันอินพุต ได้ v(t) ดังนี้

$$i(t) = a_1 v(t) + a_2 v^2(t) + a_3 v^3(t) + \ldots \qquad (2.10)$$

โดย $a_i (i = 1,2\ldots)$ คือ ค่าคงที่ขึ้นกับความไม่เป็นเชิงเส้นของอุปกรณ์ในวงจร

ในการสร้างมอดูเลเตอร์ทำได้ โดยการเลือกอุปกรณ์ไม่เชิงเส้นที่ทำให้ค่าพจน์ที่มีกำลังสูงกว่า 2 ใน (2.10) นั้นมีค่าน้อยมาก นั่นคือจะเลือกอุปกรณ์และส่วนประกอบของวงจรในรูปที่ 2.8 ที่ให้ความ สัมพันธ์

$$i(t) \approx a_1 v(t) + a_2 v^2(t) \qquad (2.11)$$

ซึ่งปกติ (2.11) จะใช้ประมาณค่าของ (2.10) ได้เสมอ เมื่อ v(t) มีค่าน้อยกว่า 1 มาก

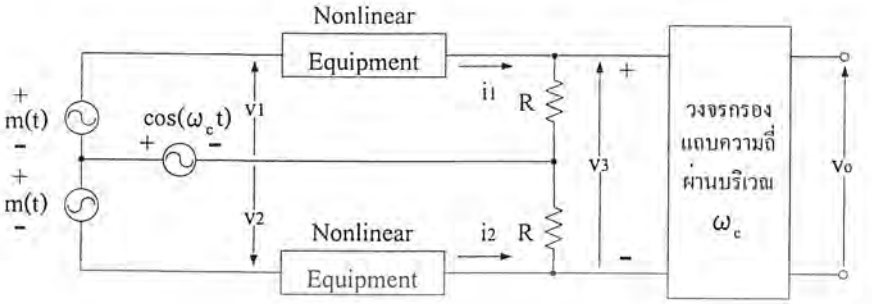
เมื่อใช้สัญญาณข่าวสาร m(t) รวมกับคลื่นพาห์ $\cos(\omega_c t)$ เป็นสัญญาณอินพุต v(t) จะได้เอาต์พุต $v_o(t)$ ของวงจรเป็น

$$\begin{aligned} v_o(t) &= R\, i(t) \\ &= R\,[\, a_1 \{m(t) + \cos(\omega_c t)\} + a_2 \{m(t) + \cos(\omega_c t)\}^2\,] \\ &= R\,[\, a_1 \{m(t) + \cos(\omega_c t)\} + a_2 \{m^2(t) + 2m(t)\cos(\omega_c t) + \cos^2(\omega_c t)\}\,] \\ &= a_1 R\, m(t) + a_2 R\, m^2(t) + a_1 R \cos(\omega_c t) + 2a_2 R\, m(t)\cos(\omega_c t) \\ &\quad + \frac{a_2 R}{2} + \frac{a_2 R}{2}\cos(2\omega_c t) \end{aligned} \qquad (2.12)$$

จะเห็นได้ว่าใน (2.12) นั้นพจน์ที่ 4 ทางด้านขวามือของสมการคือสัญญาณ ดีเอสบี เอสซีตามต้องการ แต่การจะแยกเอาเฉพาะสัญญาณส่วนนี้ โดยใช้วงจรกรองแถบความถี่ผ่านนั้น จะทำให้สัญญาณตามพจน์ที่ 3 ลอดผ่านออกไปได้ด้วย ทั้งนี้เพราะสัญญาณตามพจน์ที่ 3 และ พจน์ที่ 4 อยู่ในระหว่างช่วงความถี่ผ่านของวงจรกรองสัญญาณเหมือนกัน ดังนั้นเราจึงไม่สามารถที่จะแยกเฉพาะสัญญาณ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ดีเอสบี เอสซี ออกมาได้ตามต้องการ โดยใช้แต่เพียงวงจรกรองแถบความถี่ผ่าน เพื่อขจัดปัญหานี้ จะทำได้โดยการต่อวงจรให้มีลักษณะดังแสดงในรูปที่ 2.9 ตามรูปนี้เราจะพบว่า



รูปที่ 2.9 วงจรมอดูเลเตอร์แบบสมดุล

$$v_1(t) = \cos(\omega_c t) + m(t) \tag{2.13}$$

และ

$$v_2(t) = \cos(\omega_c t) - m(t) \tag{2.14}$$

โดยสมมุติให้ความสัมพันธ์ระหว่างกระแส และแรงดันไฟฟ้าของวงจรไม่เชิงเส้นเป็นไปตาม (2.11) จะได้

$$i_1(t) = a_1 [\cos(\omega_c t) + m(t)] + a_2 [\cos(\omega_c t) + m(t)]^2 \tag{2.15}$$

และ

$$i_2(t) = a_1 [\cos(\omega_c t) - m(t)] + a_2 [\cos(\omega_c t) - m(t)]^2 \tag{2.16}$$

จากรูปที่ 2.9 จะพบว่า

$$\begin{aligned} v_3(t) &= i_1 R - i_2 R \\ &= (i_1 - i_2) R \end{aligned} \tag{2.17}$$

เมื่อนำค่ากระแสตาม (2.15) และ (2.16) แทนใน (2.17) และจัดรูปสมการใหม่จะได้

$$v_3(t) = 4a_2 R \left[m(t)\cos(\omega_c t) + \frac{a_1}{2a_2} m(t) \right] \tag{2.18}$$

จะพบว่าในวงจรนี้ให้ค่าแรงดัน $v_3(t)$ ที่ไม่ปรากฏคลื่นพาห์ $\cos(\omega_c t)$ รวมอยู่ด้วย ดังนั้นเมื่อเราเอาวงจรกรองแถบความถี่ผ่านมากรองสัญญาณตาม (2.18) เราก็จะได้สัญญาณ ดีเอสบี เอสซีตามต้องการ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

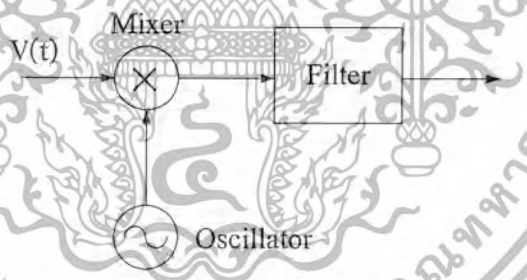
2.1.4 การดีมอดูเลตสัญญาณที่มีการมอดูเลตทางขนาด

สัญญาณที่ ได้รับการมอดูเลต ทางขนาดนั้น จะเป็นสัญญาณเบสแบนด์ ที่ถูกเลื่อนความถี่ให้สูงขึ้นดังที่กล่าวมา ดังนั้น การดีมอดูเลตสัญญาณ ก็จะเป็นกระบวนการในการเลื่อนความถี่นั้น ให้กลับมาอยู่ที่ความถี่เบสแบนด์อย่างเดิม ซึ่งอาจจะทำได้หลายวิธี ดังต่อไปนี้

1) การแปลงความถี่สัญญาณ

ดังได้กล่าวมาแล้วว่าในการสร้างสัญญาณ AM หรือสัญญาณ DSBSC นั้น เราอาจทำได้โดยการใช้วงจรคูณ เพราะวงจรคูณจะทำการคูณสัญญาณเบสแบนด์นั้นกับสัญญาณคลื่นพาห์ทำให้เกิดการเลื่อนความถี่สูงขึ้นได้ และในทำนองเดียวกันถ้ามีสัญญาณ AM เข้ามา เราก็สามารถใช้การเลื่อนความถี่ของสัญญาณนั้นให้ไปอยู่ที่ความถี่เดิมได้ รูปที่ 2.10 แสดงบล็อกไดอะแกรมของระบบแปลงความถี่ของสัญญาณที่กล่าวถึงนี้ ถ้าให้สัญญาณเข้าเป็นสัญญาณ AM ที่เขียนได้เป็น $V_{AM} = m(t)\cos(\omega_c t)$ เมื่อเราคูณสัญญาณออกจากออสซิลเลเตอร์ที่มีความถี่เป็น ω_c (ความถี่เดียวกันกับคลื่นพาห์) และมีขนาดเป็น A_o เราก็จะได้สัญญาณผลคูณที่เขียนได้ ดังต่อไปนี้

$$\begin{aligned} m(t)A_o\cos(\omega_c t) &= m(t)A_o\cos(\omega_c t)\cos(\omega_c t) \\ &= [m(t)A_o/2]\cos(\omega_c+\omega_c)t + [m(t)A_o/2]\cos(\omega_c-\omega_c)t \\ &= [m(t)A_o/2] + [m(t)A_o/2]\cos(2\omega_c t) \end{aligned} \quad (2.19)$$



รูปที่ 2.10 ระบบแปลงความถี่ของสัญญาณที่ใช้วงจรคูณ

พบว่าจากสมการที่ (2.19) เมื่อเรานำไปผ่านวงจรกรองความถี่ต่ำก็จะสามารถกรองเอาสัญญาณเบสแบนด์ m(t) ออกมาได้ ซึ่งการเปลี่ยนความถี่ที่กล่าวมานี้นอกจากใช้ประโยชน์ในการดีมอดูเลตสัญญาณสื่อสารแล้ว ยังสามารถใช้ประโยชน์ทางด้านอื่นอีกมากมาย เช่น ใช้ในเครื่องวิเคราะห์ความถี่ ใช้ในการสแกรมเบิล สัญญาณเสียง ใช้ในการสร้างออสซิลเลเตอร์ เป็นต้น

2) วงจรตรวจจับกรอบสัญญาณ

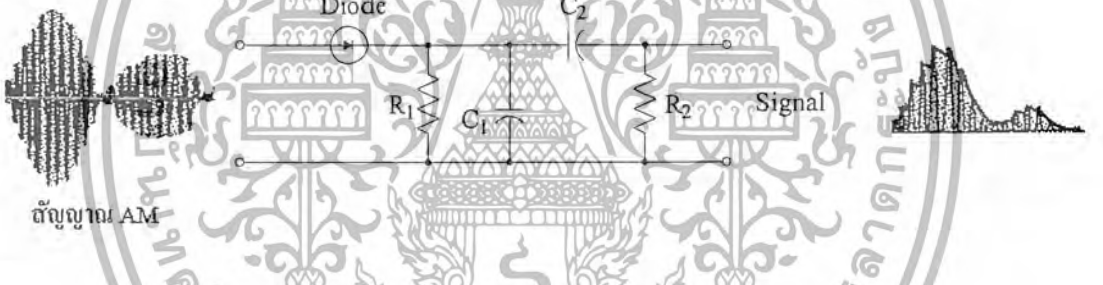
วงจรตรวจจับกรอบสัญญาณ คือวงจรที่ให้เอาต์พุตมีลักษณะเป็นไปตามกรอบ หรือแนวทางเดินของยอดคลื่นของสัญญาณอินพุต วงจรตรวจจับกรอบสัญญาณแบบง่ายจะเป็นวงจรชนิดไม่เชิงเส้น โดยตัวอย่างวงจรที่ทำหน้าที่นี้คือ วงจรที่แสดงในรูปที่ 2.11 ขณะที่สัญญาณมีค่าเพิ่มขึ้นในทางบวก ไดโอดในวงจรจะกรองสัญญาณด้านบวกให้ผ่านไปได้ ในขณะที่ R_1C_1 ซึ่งจะทำงานได้ดีนั้นค่า

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

คงตัวเชิงเวลา (Time constant) ของวงจรหรือค่า R_1C_1 นั้น เมื่อเทียบกับคาบเวลาของสัญญาณคลื่นพาห์จะต้องมีค่าที่เหมาะสม ซึ่งถ้าค่าคงตัวเชิงเวลาของวงจรตรวจจับกรอบสัญญาณมีค่ามากเมื่อเทียบกับค่าคาบเวลาที่สั้นที่สุดของสัญญาณที่ประกอบอยู่ใน m(t) และก็จะทำให้เอาต์พุตของวงจรเกิดลดตามยอดคลื่นของสัญญาณ AM ไม่ทัน และจะทำให้เกิดการผิดเพี้ยนของสัญญาณเอาต์พุต และในกรณีที่เกิดมีการใช้ค่า RC มีค่าน้อยมาก กล่าวคือค่าผลคูณของ RC มีค่าน้อย เมื่อเทียบกับค่าคาบเวลาของคลื่นพาห์ก็จะทำให้เกิดการกระเพื่อม (Ripple) บนสัญญาณเอาต์พุตมาก ซึ่งเงื่อนไขสำหรับการเลือกค่า RC ที่เหมาะสมสามารถสรุปได้ในรูปของอสมการดังต่อไปนี้คือ

$$f_m \leq 1/R_1C_1 \leq f_c \qquad (2.20)$$

โดยในที่นี้ f_c และ f_m คือ ความถี่ของคลื่นพาห์ และค่าความถี่สูงสุดที่ประกอบอยู่ในสัญญาณข่าวสาร m(t) ตามลำดับ



รูปที่ 2.11 ตัวอย่างวงจรเอ็นเวลโลปดีเทคชัน

(ก) ค่า RC ที่เหมาะสม (ข) ค่า RC ที่มากเกิน (ค) ค่า RC ที่น้อยเกิน

รูปที่ 2.12 การดีเทคสัญญาณที่มีค่า RC ต่างๆกัน

สำหรับ R_2C_2 นั้นจะทำหน้าที่สกัดส่วนประกอบกระแสตรงออก และมักจะสกัดความถี่ต่ำ ๆ ออกด้วย ซึ่งอาจจะไม่เหมาะที่จะนำไปใช้งานที่ส่วนประกอบความถี่มีความหมายมาก

ตามที่อธิบายนั้น จะเห็นว่าวงจรตรวจจับกรอบสัญญาณเป็นวงจรที่มีส่วนประกอบน้อยมาก ดังนั้นถ้าเกิดมีความจำเป็นที่จะต้องสร้างเครื่องรับจำนวนมาก การใช้วิธีการตรวจจับกรอบสัญญาณนี้ก็จะ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ช่วยให้ประหยัดได้มาก ด้วยข้อดีดังกล่าวนี้ ในการส่งวิทยุกระจายเสียงแบบเอเอ็มทั่วไป จึงใช้การส่งสัญญาณดีเอสบี เอสซี พร้อมคลื่นพาห์ (AM with carrier) ทั้งสิ้น ทั้งนี้เพื่อที่จะได้ประหยัดอุปกรณ์ในการสร้างเครื่องรับ ดังนั้นเมื่อกล่าวถึงการส่งวิทยุเอเอ็มแบบธรรมดาแล้ว ก็จะหมายถึง การส่งสัญญาณดีเอสบี-เอสซี พร้อมคลื่นพาห์

3) เครื่องรับแบบคอสแทส

ในการดีมอดูเลตสัญญาณดีเอสบี เอสซี นั้น ความถี่และเฟสของคลื่นพาห์ที่เครื่องรับสร้างขึ้นมา เพื่อใช้ในการดีมอดูเลตแบบสัมพัทธ์นั้น ถ้าหากว่ามีค่าต่างไปจากค่าความถี่และเฟสของคลื่นพาห์ที่รับมาจากเครื่องส่งก็จะทำให้เกิดการลดทอน และเกิดความผิดเพี้ยนของสัญญาณข่าวสารที่ตรวจจับออกมาได้ การแก้ปัญหานี้วิธีหนึ่งทำได้โดย การจัดแยกส่งคลื่นนำร่อง (Pilot Signal) ที่มีข้อมูลของคลื่นพาห์แฝงอยู่ไปยังเครื่องรับพร้อมกับสัญญาณดีเอสบี เอสซี เพื่อให้เครื่องรับจะได้มีสื่ออ้างอิงในการผลิตคลื่นพาห์ขึ้นอย่างถูกต้อง วิธีการเช่นนี้มีจุดอ่อนที่มีความซับซ้อนในการที่จะต้องจัดไม่ให้คลื่นนำร่องนั้นไปรบกวนสัญญาณหลัก ยังมีวิธีการที่นิยมกันในการดีมอดูเลตสัญญาณ ดีเอสบี เอสซี อีกวิธีหนึ่งก็คือ การจัดวงจรในรูปคอสแทส (Costas's loop) ดังแสดงในรูป 2.13

รูปที่ 2.13 วงจรเครื่องรับแบบคอสแทส

หลักการทำงานของวงจรมีดังนี้ คือ เมื่อจะทำการดีมอดูเลตสัญญาณ ดีเอสบี เอสซี คือ $m(t)\cos(\omega_c t)$ นั้น จะใช้วงจรออสซิลเลเตอร์ควบคุมได้ด้วยแรงดัน (Voltage Controlled Oscillator) หรือที่เรียกย่อว่าวงจรวีซีโอ (VCO) มาทำหน้าที่เป็นวงจรออสซิลเลเตอร์ในเครื่องรับ โดยเริ่มต้นออสซิลเลตที่ความถี่ทำงานอิสระ (Free Running Frequency) ซึ่งปรับไว้ให้ใกล้เคียงกับความถี่ ω_c ของคลื่นพาห์ที่รับได้มากที่สุด โดยในที่นี้จะสมมุติว่าคลื่นพาห์ที่ได้จากวงจรวีซีโอนี้คือ $2\cos(\omega_c t + \theta(t))$ ซึ่ง $\theta(t)$ เป็นค่าผิดพลาดทางเฟสที่เกิดขึ้น (ส่วนที่เป็นความคลาดเคลื่อนของความถี่ก็คิดรวมอยู่ใน $\theta(t)$ นี้ด้วย) เพื่อปรับความถี่จากวีซีโอให้ถูกต้องสัญญาณส่วนหนึ่งจากวีซีโอจะถูกส่งตรงไปมอดูเลต (คูณ) กับสัญญาณอินพุต

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

$m(t)\cos(\omega_c t)$ และนำสัญญาณที่เกิดขึ้นไปผ่านวงจรกรองความถี่ต่ำผ่าน จะทำให้ได้สัญญาณ $m(t)\cos(\theta(t))$ ออกมา และสัญญาณอีกส่วนหนึ่งจากวีซีโอ จะถูกนำไปเลื่อนเฟส $-\frac{\pi}{2}$ เรเดียน เพื่อให้ได้คลื่น $2\sin(\omega_c t + \theta(t))$ ออกมา ซึ่งสัญญาณนี้จะถูกนำไปมอดูเลตกับสัญญาณ $m(t)\cos(\omega_c t)$ ที่รับได้ ซึ่งเมื่อนำสัญญาณที่ได้ไปผ่านวงจรกรองความถี่ต่ำก็จะทำให้ได้สัญญาณ $m(t)\sin(\theta(t))$ ออกมา เมื่อนำสัญญาณนี้ไปคูณกับสัญญาณ $m(t)\cos(\theta(t))$ จะได้ผลลัพธ์เป็น $\frac{1}{2}m^2(t)\sin(2\theta(t))$ ซึ่งสัญญาณนี้จะถูกส่งผ่านวงจรกรองความถี่ต่ำที่มีช่วงผ่านความถี่แคบก็จะได้สัญญาณความถี่ต่ำมากที่แปรผันตามค่า $\sin(2\theta(t))$ ออกมา กล่าวคือจะได้เอาต์พุตเป็น $k\sin(2\theta(t))$ โดย k เป็นค่าที่ขึ้นอยู่กับค่าเฉลี่ยของสัญญาณ $\frac{1}{2}m^2$ ซึ่งประมาณคงที่ และถ้า $\theta(t)$ มีค่าน้อย $k\sin(2\theta(t))$ จะมีค่าประมาณเท่ากับ $2k\theta(t)$ ซึ่งก็หมายความว่าค่าเอาต์พุตของวงจรกรองความถี่ต่ำที่มีช่วงผ่านความถี่แคบนั้น จะแปรผันตามค่า $\theta(t)$ ค่าเอาต์พุตนี้จะถูกส่งไปควบคุมวงจรวีซีโอ เพื่อให้ผลิตคลื่นความถี่ที่มีค่า $\theta(t)$ น้อยลง กระบวนการป้อนกลับนี้จะทำงานกันอย่างต่อเนื่องไปจน $\theta(t)$ ลดลงเป็นศูนย์ในที่สุด กล่าวคือคลื่นพาห์จากวีซีโอ จะถูกปรับตัวจนมีค่าความถี่ตรงกับค่าจากเครื่องส่ง ซึ่งก็จะทำให้วงจรสามารถดีมอดูเลตสัญญาณ $m(t)$ ออกมาได้อย่างถูกต้อง ซึ่งเราสามารถทำออกมาใช้ได้โดยตรงจากวงจรกรองความถี่ต่ำผ่านวงจรข้างบน

ในการใช้งานจริง ปรากฏการณ์ป้อนกลับนี้จะเกิดขึ้นอย่างรวดเร็วจนผู้รับฟังไม่รู้สึกได้ วิธีการแก้ความคลาดเคลื่อนของคลื่นพาห์โดยใช้วงจรคอสแทสแบบนี้เป็นที่นิยมมาก เครื่องรับที่ใช้หลักการนี้จะมีชื่อเรียกเฉพาะว่า เครื่องรับแบบคอสแทส (Costas's receiver)

ให้สังเกตจากรูปที่ 2.13 ว่า เมื่อ $\theta(t)$ เป็นศูนย์จะได้ $\cos(\theta(t)) = 1$ ดังนั้นเอาต์พุตของวงจรคือสัญญาณ $m(t)$ ที่ต้องการได้จากการดีมอดูเลตนั่นเอง

2.1.5 กำลังส่งที่เสียไปกับคลื่นพาห์ในการส่งสัญญาณเอเอ็ม

ดังที่ได้อธิบายแล้วว่า ในระบบวิทยุกระจายเสียงเอเอ็มนั้นจะใช้คลื่นพาห์เพิ่มส่งไปพร้อมกับสัญญาณดีเอสบี เอสซี เพื่อประโยชน์ที่จะดีมอดูเลตสัญญาณที่เครื่องรับได้ง่าย คือสามารถใช้วงจรตรวจจับยอดคลื่นได้ดังกล่าวมาแล้ว แต่ข้อเสียของวิธีการนี้ก็มีอยู่ กล่าวคือ จะมีการเสียกำลังส่งเพิ่มขึ้นในการส่งคลื่นพาห์ที่เพิ่มขึ้นนั้น โดยกำลังส่งที่เสียเพิ่มขึ้นนี้จะไม่ได้ช่วยขับกำลังของการส่งข่าวสาร $m(t)$ เลย ทั้งนี้เพราะข่าวสารต่าง ๆ จากสัญญาณ $m(t)$ นั้น จะรวมอยู่ในแถบข้างทั้งสองเท่านั้น (เรายอมสิ้นเปลืองกำลังในการส่งคลื่นพาห์เพิ่มขึ้น ก็เพื่อที่จะทำให้ทำการตรวจจับสัญญาณได้ง่ายขึ้นเท่านั้น)

เพื่อแสดงให้เห็นว่ามีการสูญเสียกำลังของเครื่องส่งในคลื่นพาห์มากน้อยอย่างไร เราจะพิจารณาเปรียบเทียบกำลังนั้น ในกรณีที่ $m(t)$ มีรูปแบบง่าย ๆ คือ $m(t) = A_m\cos\omega_m t$; $\omega_m \ll \omega_c$ โดยการกระจาย $\phi_{AM}(t) = A_c\{1+\mu\cos(\omega_m t)\}\cos(\omega_c t)$ จะได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

$$\phi_{AM}(t) = A_c \cos(\omega_c t) + A_c \mu \cos(\omega_m t)\cos(\omega_c t)$$

$$= A_c \cos(\omega_c t) + \frac{A_c \mu}{2}\cos\{(\omega_c - \omega_m)t\} + \frac{A_c \mu}{2}\cos\{(\omega_c + \omega_m)t\} \qquad (2.21)$$

พจน์ที่ 1, 2 และ 3 ทางขวามือของ (2.21) นั้นแสดงถึงส่วนประกอบของสัญญาณเอเอ็มที่เกิดขึ้น ซึ่งจะประกอบขึ้นด้วยส่วนที่เป็นคลื่นพาห์ สัญญาณแอลเอสบีและสัญญาณยูเอสบี ตามลำดับ

เนื่องจากค่ากำลังเฉลี่ยของคลื่นรูปไซน์ จะเท่ากับค่ากำลังสองของค่าอาร์เอ็มเอสของแอมปลิจูดของคลื่น ดังนั้นเราจะได้ค่ากำลังเฉลี่ยของคลื่นทั้งสามส่วนตาม (2.21) นั้น เท่ากับ $\frac{A_c^2}{2}$, $\frac{A_c^2\mu^2}{8}$ และ $\frac{A_c^2\mu^2}{8}$ ตามลำดับ เพราะฉะนั้นค่ากำลังเฉลี่ยรวมทั้งหมดของ $\phi_{AM}(t)$ จึงเท่ากับ $\frac{A_c^2}{2}(1+\frac{\mu^2}{2})$ และค่าอัตราส่วนของกำลังของคลื่นต่าง ๆ จะหาได้ดังนี้ คือ

$$\frac{\text{กำลังเฉลี่ยของคลื่นพาห์}}{\text{กำลังเฉลี่ยรวมของ } \phi_{AM}(t)} = \frac{2}{2+\mu^2} \qquad (2.22)$$

$$\frac{\text{กำลังเฉลี่ยของสัญญาณแถบข้าง}}{\text{กำลังเฉลี่ยรวมของ } \phi_{AM}(t)} = \frac{\mu^2}{2+\mu^2} \qquad (2.23)$$

จะพบว่าการส่งสัญญาณเอเอ็มในแบบนี้จะสิ้นเปลืองกำลังส่งไปกับการส่งคลื่นพาห์อย่างมาก แม้เมื่อกรณีที่มีการมอดูเลตเต็มที่ คือ $\mu = 100\%$ ความสูญเสียนี้จะมีสูงถึงประมาณ 67% ของกำลังส่งทั้งหมด และเนื่องจากในทางปฏิบัติจริงการส่งวิทยุแบบนี้จะต้องใช้ค่า μ ที่น้อยกว่า 100% คือประมาณ 75 – 85% เท่านั้น เพราะสัญญาณข่าวสาร m(t) นั้น จะมีขนาดที่เปลี่ยนแปลงไม่ แน่นอนตลอดเวลา ดังนั้นจึงต้องเตรียมค่าไว้เพื่อกันไม่ให้เกิดการมอดูเลตเกิน เมื่อสัญญาณมีขนาดมากขึ้นด้วย โดยมีแต่การส่งวิทยุกระจายเสียงเพียงเท่านั้นที่ใช้การมอดูเลตแบบนี้อยู่เป็นหลัก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.2 องค์ประกอบของสัญญาณภาพ

สัญญาณโทรทัศน์สีมีองค์ประกอบที่เป็นสัญญาณต่าง ๆ ที่มีหน้าที่ต่าง ๆ กันไป ซึ่งที่จะกล่าวถึงในที่นี้จะอ้างอิงสัญญาณภาพที่เป็นระบบ PAL Standard โดยจะอธิบายไว้เพียงสังเขปดังนี้

2.2.1 สัญญาณลูมิแนนซ์ (Luminance Signal)

สัญญาณลูมิแนนซ์ หรือสัญญาณส่องสว่างเป็นสัญญาณที่มีระดับแรงดันเปลี่ยนแปลงไปตามระดับความสว่างของภาพ โดยจะมีแบนด์วิดท์ 5 MHz สำหรับมาตรฐาน CCIR หรือ 4MHz สำหรับมาตรฐาน FCC ในการส่งสัญญาณโทรทัศน์สีนั้นจำเป็นจะต้องส่งสัญญาณความสว่างนี้อยู่เพื่อทำให้โทรทัศน์ขาวดำสามารถรับและสร้างภาพขาวดำให้เกิดขึ้นบนหน้าจอได้และส่งไปเข้าเครื่องรับโทรทัศน์สีเพื่อไปหักล้างกับสัญญาณความต่างสี R-Y B-Y และ G-Y ก็จะเหลือเฉพาะสัญญาณแม่สี คือ R B และ G ตามลำดับ ลักษณะของสัญญาณลูมิแนนซ์หรือสัญญาณส่องสว่างกับสัญญาณภาพสีรวมดังรูปที่ 2.14

รูปที่ 2.14 แสดงสัญญาณลูมิแนนซ์กับสัญญาณภาพสีรวม

2.2.2 สัญญาณควบคุมตำแหน่งภาพ

ประกอบด้วยสัญญาณต่าง ๆ ดังนี้

1) <u>สัญญาณซิงโครไนซ์</u> (Synchronized Signal) เป็นสัญญาณควบคุมลำอิเล็กตรอนเพื่อให้การสแกนทั้งทางด้านส่งและทางด้านรับสัญญาณเป็นไปอย่างถูกต้องทั้งแนวตั้งและแนวนอน แยกออกเป็น

- สัญญาณซิงโครไนซ์ตามแนวนอน (Horizontal synchronize) มีความถี่ 15,625 ในระบบ CCIR และ 15,725 ในระบบ FCC ถ้าไม่มีสัญญาณนี้จะทำให้ภาพที่ปรากฏทางจอโทรทัศน์ล้มได้
- สัญญาณซิงโครไนซ์ทางแนวตั้ง (Vertical synchronize) มีความถี่ 50 Hz ในระบบ CCIR และ 60 Hz ในระบบ FCC ถ้าไม่มีสัญญาณนี้จะทำให้ภาพเลื่อน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สำนักหอสมุดกลาง พระจอมเกล้าลาดกระบัง

2) สัญญาณแบลงกิ้ง (Blanking signal) เมื่อลำอิเล็กตรอนทำการสแกนภาพจนครบ 1 เส้น ตามแนวนอน จะเกิดการสะบัดกลับตามแนวนอนเรียกว่าการ Retrace ซึ่งจะทำให้เกิดเป็นเส้นขึ้นที่ไม่ใช่ส่วนประกอบของภาพ ดังนั้น จึงต้องมีการลบทิ้ง โดยเครื่องส่งจะต้องส่งสัญญาณแบลงกิ้งนี้ออกมาพร้อมสัญญาณภาพ เพื่อเป็นการบังคับให้ลำอิเล็กตรอนให้ทางจอภาพรู้ว่าตรงช่วงไหนไม่ต้องทำการยิงอิเล็กตรอนก็จะคัทออฟการทำงานของปืนยิงอิเล็กตรอนช่วงเวลานั้น เพราะฉะนั้นก็จะไม่เกิดเส้นภาพในขณะนั้น Retrace แล้วในระบบ CCIR เวลาที่ใช้ในการสะบัดกลับตามแนวนอนจะเท่ากับ 12 μsec ส่วนเวลาในการสะบัดกลับทางแนวตั้งจะใช้เวลา 1600 μsec

3) สัญญาณอิควอไลซิ่ง (Equalizing signal) เป็นสัญญาณบังคับรูปร่างของสัญญาณซิงโครไนเซชั่นทางแนวตั้ง เพื่อให้คงรูปได้ถูกต้อง และยังช่วยการสแกนแบบสลับเส้นให้เป็นไปอย่างถูกต้องด้วย สามารถขึ้นเส้นสแกนเลขคี่ เลขคู่ได้อย่างถูกต้อง ทั้งยังส่งผลทางอ้อมให้สัญญาณซิงโครไนซ์ทางแนวนอนไม่ขาดหายไปในระหว่างการส่งสัญญาณทางแนวตั้งด้วย

รูปที่ 2.15 แสดงรูปร่างของสัญญาณซิงค์ที่อยู่ในสัญญาณโทรทัศน์

2.2.3 สัญญาณโครมา (Chroma Signal)

เพื่อให้ได้ภาพที่ปรากฏทางจอภาพของเครื่องรับโทรทัศน์มีสีเหมือนกับทางสถานีส่ง จึงต้องส่งรายละเอียดและความเข้มของสัญญาณสีที่ออกอากาศ โดยปกติสีที่ได้จากหลอดภาพโทรทัศน์จะประกอบไปด้วยแม่สีทางแสง คือ แดง เขียว น้ำเงิน แต่ถ้าจะส่งสัญญาณสีทั้ง 3 ออกไปในระบบการส่งสัญญาณโดยตรงนั้น จะต้องใช้คลื่นพาห์ในการส่งถึง 3 ความถี่ด้วยกัน ซึ่งจะทำให้แบนด์วิดท์ของสัญญาณของระบบมากเกินไป

ดังนั้น จึงต้องใช้สัญญาณก่อรูปความต่างสี ซึ่งจะทำหน้าที่เปลี่ยนสัญญาณแม่สีทั้งสามให้เหลือเพียงสัญญาณความต่างสีสองสัญญาณเท่านั้น คือ สัญญาณความต่างสี R-Y และ B-Y โดยมีอัตราส่วนการผสมดังนี้

$$R\text{-}Y = 0.70R - 0.59G - 0.11B$$

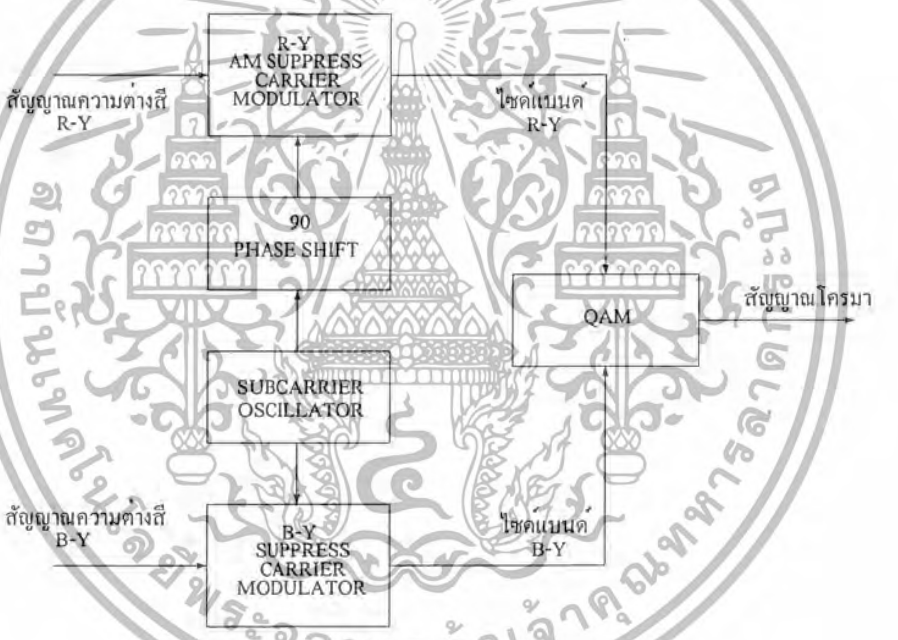
$$B\text{-}Y = 0.89B - 0.59G - 0.3R$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้
54936

ส่วนของสัญญาณสีเขียวที่เราจะไม่ส่งออกอากาศโดยตรงแต่จะฝากร่วมไปกับสัญญาณความต่างสีทั้งโดยจะอยู่ในรูปของสัญญาณ Y เนื่องจากมีแม่สีทั้งสามประกอบอยู่ร่วมกันในอัตราส่วน

$$Y = 0.3R - 0.59G - 0.11B$$

ข้อกำหนดในการส่งสัญญาณโทรทัศน์สีมาตรฐาน PAL ได้กำหนดให้ส่งสัญญาณความเข้มสีที่ได้ปนออกมากับสัญญาณความสว่างที่มีอยู่แล้ว โดยให้มอดูเลตแบบ DSB-SC ด้วยซับแคเรียร์ 4.43 MHz เพียงความถี่เดียว แต่สัญญาณความต่างสีมีอยู่ถึง 2 สัญญาณ ดังนั้น เพื่อให้สัญญาณความต่างสีทั้งสองสามารถส่งออกไปกับซับแคเรียร์ความถี่เดียวกันได้จึงมีวิธีการผสมสัญญาณแบบ QAM (Quadrature Amplitude Modulation) โดยทำการ Shift Phase ของซับแคเรียร์ ไป 90 องศา สัญญาณที่ได้จากการผสมแบบ QAM นี้เองเรียกว่า สัญญาณโครมา



รูปที่ 2.16 ลักษณะการสร้างสัญญาณโครมา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

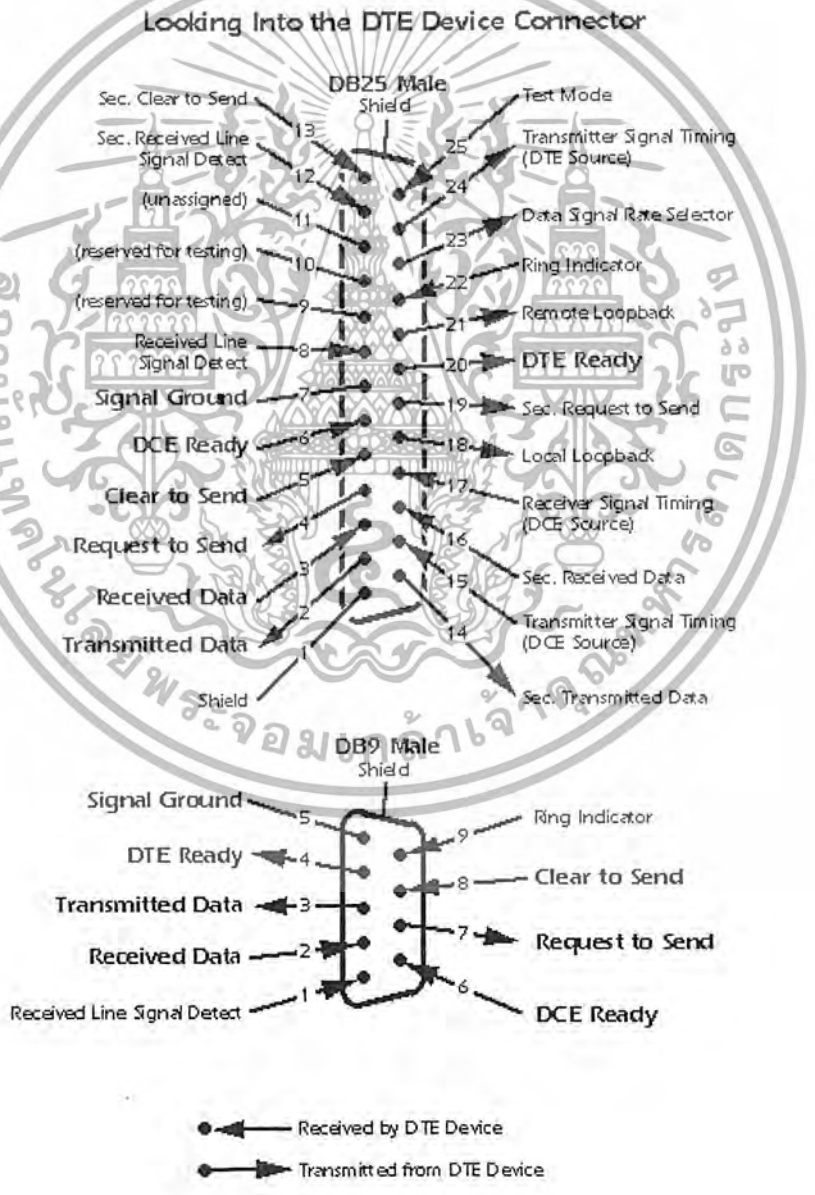
2.3 มาตรฐาน RS232 (EIA232)

เป็นมาตรฐานที่ออกแบบมาสำหรับการติดต่อสื่อสารระหว่างอุปกรณ์สื่อสารข้อมูลต่างๆ โดยมีรายละเอียดทั้งในเรื่อง ระดับสัญญาณ เวลาของสัญญาณ รูปแบบโปรโตคอลที่ใช้ และตลอดจนลักษณะทางกายภาพของอุปกรณ์ โดยมีรายละเอียดดังนี้

2.3.1 การจัดขาของอุปกรณ์

แบ่งเป็น 2 ลักษณะ คือการจัดขาทางด้าน DTE :Data Terminal Equipment (computer) และการจัดขาทางด้าน DCE :Data Circuit-terminating Equipment (อุปกรณ์ต่อพ่วง เช่น Modem) ดังนี้

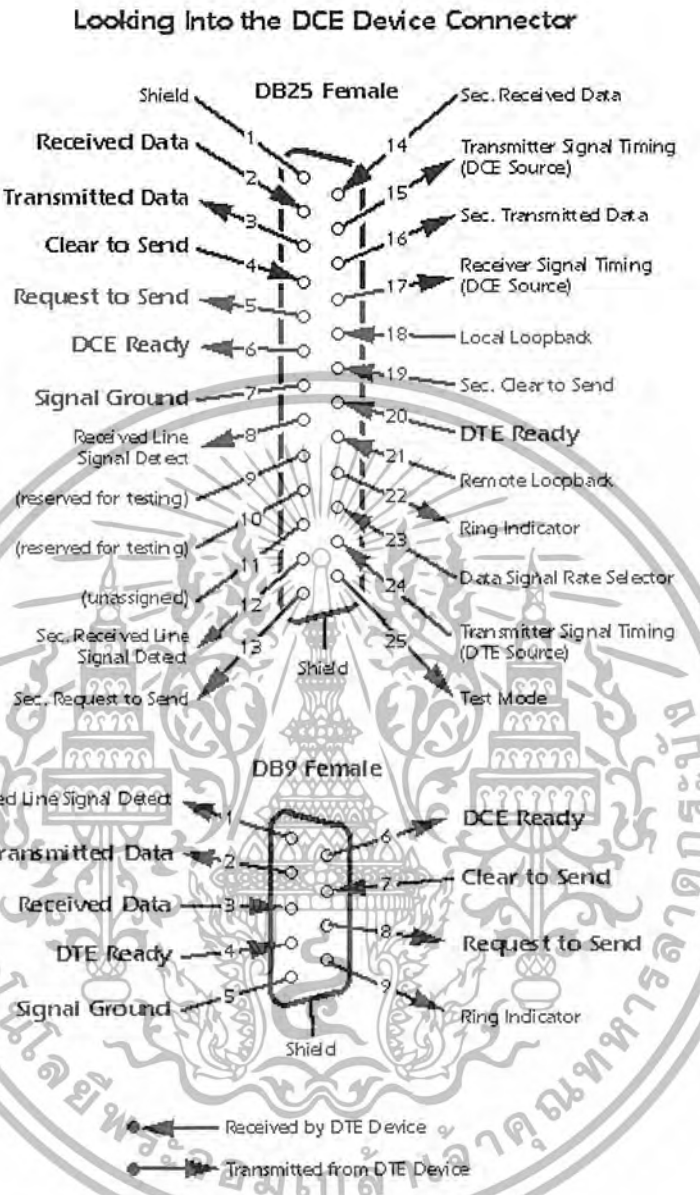
1) การจัดขาทางด้าน DTE



รูปที่ 2.17 การจัดขาทางด้าน DTE

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2) การจัดขาทางด้าน DCE



รูปที่ 2.18 การจัดขาทางด้าน DCE

ต่อไปเราจะกล่าวเฉพาะการจัดขาแบบ 9 pin เนื่องจากได้นำมาใช้ในโครงงานนี้ ความหมายของชื่อ pin ต่างๆ(แบบ 9 pin)

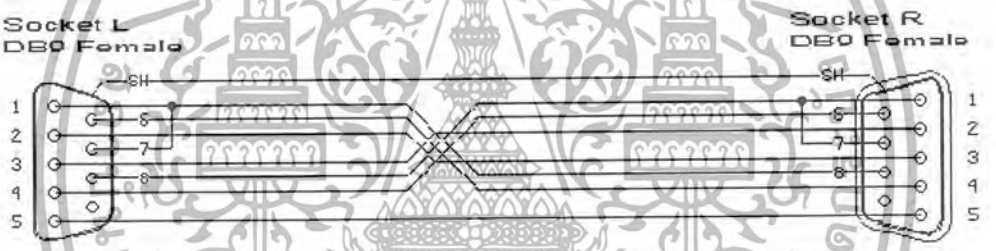
1. Signal Ground เป็น Ground ของสัญญาณ
2. Transmitted Data (TxD) เป็นขาที่ใช้ส่งข้อมูลแบบอนุกรมออกไปยังอุปกรณ์อื่นๆ
3. Received Data (RxD) เป็นขาที่ใช้รับข้อมูลอนุกรมจากอุปกรณ์ภายนอกข้ามา
4. Request to Send (RTS) เป็นขาที่ใช้ส่งสัญญาณจากอุปกรณ์ DTE ให้อุปกรณ์ DCE เตรียมตัวรับข้อมูลได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5. Clear to Send (CTS) เป็นขาที่อุปกรณ์ DCE จะส่งสัญญาณตอบกลับอุปกรณ์ DTE ว่า การส่งข้อมูลจะเกิดขึ้นได้
6. DCE Ready (DSR) เป็นขาที่ใช้บอกว่าอุปกรณ์ DCE ต่ออยู่อย่างถูกต้อง
7. DTE Ready (DTR) เป็นขาที่ใช้บอกว่าอุปกรณ์ DTE ต้องการเปิดช่องสัญญาณในการสื่อสาร
8. Received line Signal Detector (CD) บางทีก็เรียกว่า Carrier Detector เป็นขาที่ใช้ในการติดต่อ ในกรณีที่อุปกรณ์ DCE เป็นโมเด็ม ขานี้จะรับสัญญาณจากการตรวจสอบ Answer tone ของโมเด็ม
9. Ring Indicator (RI) เป็นขาที่ใช้ในการตรวจสอบสัญญาณกระดิ่ง เมื่ออุปกรณ์ DCE เป็นโมเด็ม

2.3.2 การเชื่อมต่อแบบ Null modem

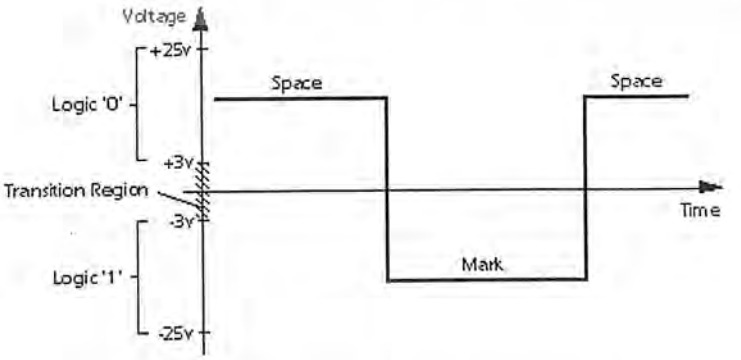
เป็นการเชื่อมต่อคอมพิวเตอร์ด้วยกันโดยใช้ Port อนุกรม โดยไม่ต้องใช้โมเด็ม การนำไปใช้งานต้องมีการต่อสายให้ถูกต้อง ดังนี้



รูปที่ 2.19 การต่อสาย null modem

2.3.3 ระดับแรงดัน

ระดับแรงดันตามมาตรฐาน RS232 จะไม่เหมือนกับ logic แบบ TTL เนื่องจากในการส่งผ่านสายสัญญาณ จะมีสัญญาณรบกวนเกิดขึ้นและจะรบกวนข้อมูลของเราได้ มาตรฐานระดับแรงดัน logic 0 และ 1 แสดงในรูปที่ 2.20



รูปที่ 2.20 แสดงระดับแรงดันในมาตรฐาน RS232

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากรูปที่ 2.20 จะเห็นได้ว่าช่วง logic 0 จะมีระดับแรงดันเป็น +3 ถึง +25 V ส่วนช่วง logic 1 จะมีระดับแรงดันเป็น –3 ถึง – 25 V แต่โดยทั่วไปช่วงlogic 0 จะมีระดับแรงดันเป็น +8 ถึง +14 V ส่วนช่วง logic 1 จะมีระดับแรงดันเป็น –8 ถึง – 14 V

2.3.4 ช่วงเวลาของสัญญาณ

ในมาตรฐาน RS232 กำหนดช่วงเวลาของสัญญาณไว้ดังนี้

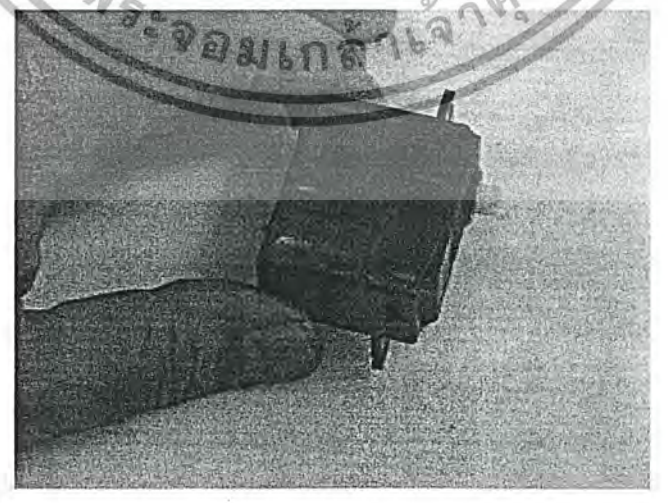
ในการใช้งานปกติมักจะใช้ความเร็วบอด 300, 1200, 2400, 9600, 19200 แต่ไม่ได้กำหนดความเร็วบอดแน่นอน

ในการเปลี่ยนแปลงของสัญญาณจากระดับ logic หนึ่งไปยังอีกระดับ logic หนึ่ง มีข้อกำหนดดังนี้

1. สัญญาณที่ผ่านช่วง transition region (ดูในรูปที่ 2.20) จะต้องเปลี่ยนไปยังอีกช่วงระดับ logic หนึ่งโดยไม่มีการย้อนกลับ
2. สัญญาณควบคุมต้องผ่านช่วง transition region โดยใช้เวลาน้อยกว่า 1 ms
3. สัญญาณข้อมูล หรือสัญญาณ timing ถ้าช่วงกว้างบิตมากกว่า 25 ms ให้ใช้เวลาเปลี่ยนผ่านช่วง transition region น้อยกว่า 1 ms แต่ถ้าช่วงกว้างบิตอยู่ระหว่าง 125 μs จนถึง 25 ms ให้ใช้เวลาเปลี่ยน 4% ของช่วงกว้างบิต และถ้าช่วงกว้างบิตน้อยกว่า 125 μs ให้ใช้น้อยกว่า 5 μs
4. Slope ควรน้อยกว่า 30 V/μs

2.4 เซอร์โวมอเตอร์

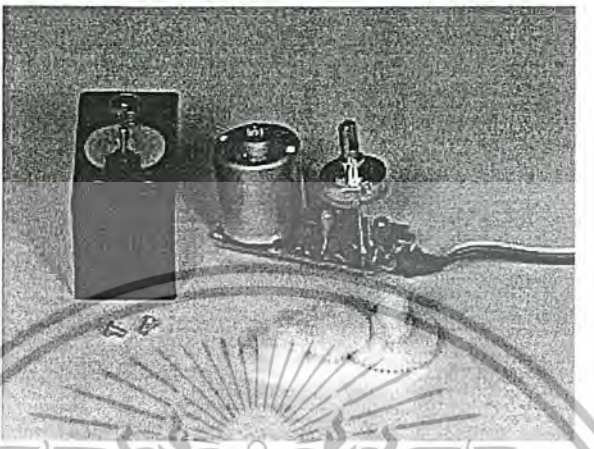
เซอร์โวมอเตอร์ ก็คือมอเตอร์ไฟตรงธรรมดาที่สามารถปรับองศาของแกนได้โดยอาศัยการส่งรหัสให้กับเซอร์โว ถ้ารหัสยิ่งนานมอเตอร์ก็จะยิ่งหมุนไปมากทำให้องศาเพิ่มขึ้น เมื่อรหัสเปลี่ยนองศาของมอเตอร์ก็จะเปลี่ยน ใช้มากในงานควบคุมต่างๆ เช่น ใน รถวิทยุบังคับ และหุ่นยนต์ เป็นต้น



รูปที่ 2.21 ตัวอย่างเซอร์โวมอเตอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เซอร์โวมอเตอร์จะใช้การป้อนกลับจากค่าความต้านทานปรับค่าได้ที่ติด(หรือทดเฟือง) กับแกนมอเตอร์ ผ่านวงจรควบคุม เซอร์โวมอเตอร์บางตัวจะมีภาค Drive อยู่ในตัวด้วย ซึ่งทำให้ง่านต่อการนำไปใช้งาน โดยใช้สาย 3 เส้นคือ ไฟเลี้ยงบวก ไฟเลี้ยงลบ สายสัญญาณควบคุม

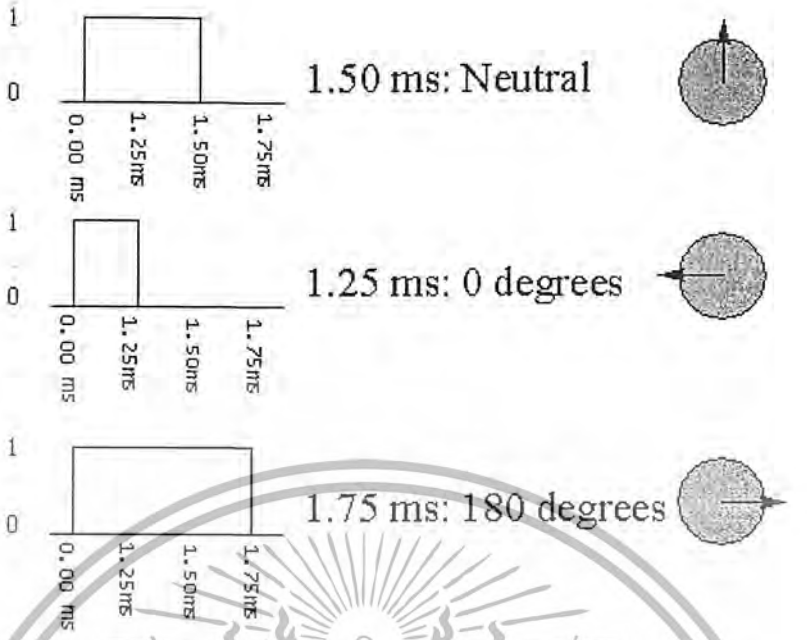


รูปที่ 2.22 รายละเอียดภายในของเซอร์โวมอเตอร์

การทำงานของเซอร์โวมอเตอร์จะใช้การตรวจจับค่าความเปลี่ยนแปลงโดยผ่านทางตัวต้านทานปรับค่าได้ เมื่อเซอร์โวมอเตอร์หมุนได้มุมที่ถูกต้อง มันจะหยุดหมุนเนื่องจากค่าความต้านทานก็ได้เปลี่ยนไปจนกระทั่งค่าความต่างระหว่างสัญญาณภายในที่ใช้ในการเปรียบเทียบเป็นศูนย์ นั่นคือเซอร์โวมอเตอร์ก็จะหยุดหมุน และจะหมุนอีกครั้งเมื่อเราส่งสัญญาณควบคุมค่าใหม่เข้าไป โดยปกติเซอร์โวมอเตอร์มักจะหมุนได้ในช่วง180 องศา แต่มีบางรุ่นที่หมุนได้ถึง 210 องศา

การควบคุมเซอร์โวมอเตอร์เราจำเป็นต้องจ่ายสัญญาณควบคุมเข้าไปที่ขาควบคุมของเซอร์โวมอเตอร์ โดยเราจะควบคุมแบบใช้ความกว้างพัลส์ หรือ pulse width modulation ดังแสดงในรูปที่ 2.23 ซึ่งจะพบว่ายิ่งความกว้างพัลส์มีค่ามากก็จะทำให้เซอร์โวมอเตอร์หมุนไปมากตามด้วย โดยปกติเซอร์โวมอเตอร์ทั่วไปจะใช้ค่าความกว้างพัลส์เท่ากับ 1.5 ms ในการทำให้หมุนไปที่ตำแหน่ง 90 องศา หรือตำแหน่งกลาง แต่ถ้าจ่ายค่าความกว้างพัลส์น้อยกว่า 1.5 ms เซอร์โวมอเตอร์ก็จะหมุนไปในตำแหน่ง 0-90 องศา แต่ถ้าจ่ายค่าความกว้างพัลส์มากกว่า 1.5 ms เซอร์โวมอเตอร์ก็จะหมุนไปในตำแหน่งที่มากกว่า 90 องศา นั่นก็คือ 90-180 องศา แต่อย่างไรก็ตามเซอร์โวมอเตอร์บางตัวอาจจะใช้ค่าความกว้างพัลส์ในการควบคุมต่างไปจากนี้ซึ่งขึ้นอยู่กับผู้ผลิต

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.23 ตัวอย่างการควบคุมองศาการหมุนของเซอร์โวมอเตอร์

2.5 TCP/IP, OSI Model กับระบบเครือข่าย

2.5.1 OSI Model

เมื่อคอมพิวเตอร์ของเรามีการรับส่งข้อมูลกับคอมพิวเตอร์เครื่องอื่น ๆ การเชื่อมต่อคอมพิวเตอร์หลาย ๆ เครื่องเข้าด้วยกันเป็นระบบเครือข่ายก็เกิดขึ้น อย่างไรก็ตามการเชื่อมต่อคอมพิวเตอร์คนละระบบหรือคนละยี่ห้อเป็นสิ่งที่ทำได้ยากในยุคแรก ๆ ของการสื่อสารข้อมูล เนื่องจากขาดมาตรฐานส่วนกลางที่จำเป็นต้องใช้ในการรับส่งข้อมูล เพราะแต่ละยี่ห้อก็จะมีมาตรฐานของตนเอง ซึ่งเข้ากันไม่ได้กับยี่ห้ออื่น ทำให้ผู้ใช้ต้องผูกติดกับผู้ผลิตแต่ละยี่ห้อ และเป็นข้อจำกัดในการเชื่อมต่อทำให้ไม่สามารถรับส่งข้อมูลระหว่างคอมพิวเตอร์คนละชนิดได้ ระบบคอมพิวเตอร์ในยุคนั้นจึงเป็น ระบบเปิด (Closed System)

ปัญหานี้ทำให้เกิดหน่วยงานกำหนดมาตรฐานสากลขึ้น คือ International Standards Organization (ISO) เพื่อจัดการกำหนดโครงสร้างทั้งหมดที่จำเป็นต้องใช้ในการสื่อสารข้อมูลจากคอมพิวเตอร์ระบบหนึ่งไปยังคอมพิวเตอร์อีกระบบหนึ่ง จุดมุ่งหมายก็เพื่อเปิดช่องทางให้กับข้อมูลที่ถูกเก็บอยู่ในระบบคอมพิวเตอร์หนึ่ง ๆ รับส่งไปยังคอมพิวเตอร์ที่เป็นระบบเดียวกันหรือต่างระบบได้อย่างอิสระ โดยไม่ขึ้นกับผู้ผลิตอย่างที่เป็นอยู่ในอดีต ซึ่งเป็นการทำงานแบบที่เรียกว่า ระบบเปิด (Open System) เราเรียกโครงสร้างของมาตรฐานการรับส่งข้อมูลนี้ว่า Open System Interconnection หรือ OSI ซึ่งจัดทำขึ้นราวกลางปี ศ.ศ. 1970 และใช้อ้างอิงมาจนถึงยุคปัจจุบัน

OSI ได้กำหนดให้การสื่อสารข้อมูลจากระบบคอมพิวเตอร์หนึ่งไปยังอีกระบบหนึ่งแบ่งออกเป็น 7 ขั้นตอนย่อย ๆ ซึ่งคอมพิวเตอร์ทั้งสองระบบจะต้องมีขั้นตอนทั้ง 7 นี้เหมือนกัน เราเรียกการสื่อสารข้อมูลนี้ว่า OSI 7-Layer Reference Model ดังแสดงในรูปที่ 2.24

เอกสารนี้สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

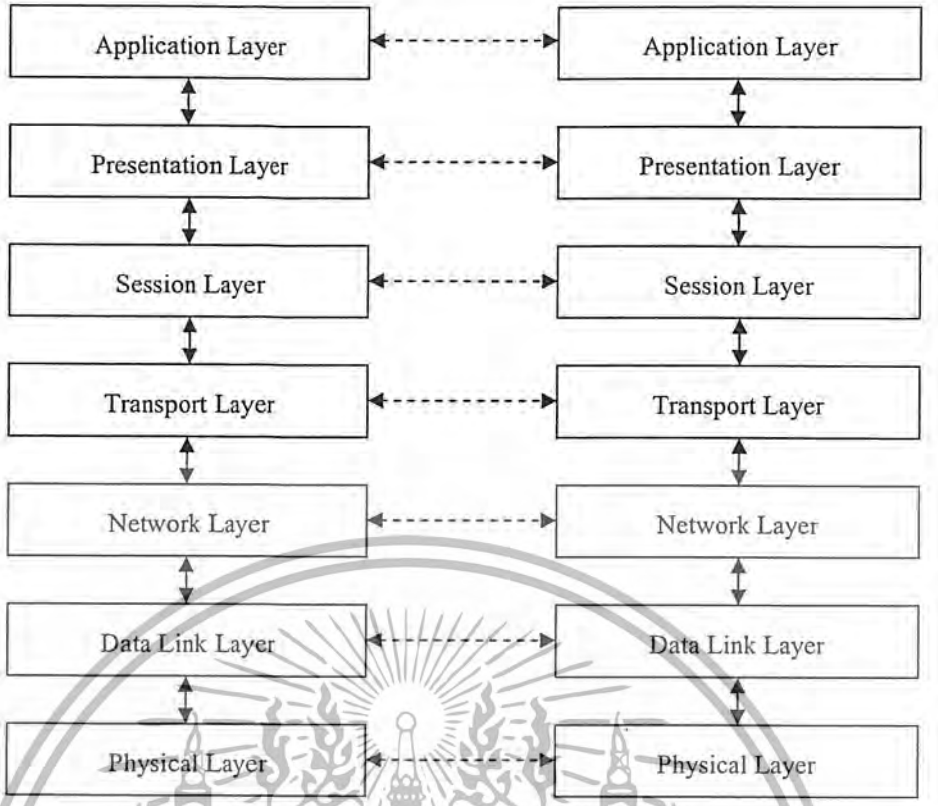
Application Layer
Presentation Layer
Session Layer
Transport Layer
Network Layer
Data Link Layer
Physical Layer

รูปที่ 2.24 โครงสร้างของ OSI 7-Layer Reference Model

สำหรับโครงสร้างของการสื่อสารข้อมูลที่กำหนดขึ้นมีคุณสมบัติดังนี้ คือ แต่ละขั้นของการสื่อสารข้อมูลเราเรียกว่า Layer หรือ ชั้น ซึ่งในการสื่อสารข้อมูลจะประกอบด้วยชั้นย่อย ๆ 7 ชั้น ในแต่ละชั้นหรือแต่ละ Layer จะเสมือนเชื่อมต่ออยู่กับชั้นที่เทียบเท่ากันของคอมพิวเตอร์อีกด้านหนึ่ง ส่วนการเชื่อมต่อกันจริง ๆ จะมีเพียงชั้นที่ 1 หรือ Layer 1 ซึ่งเป็นชั้นล่างสุดเท่านั้นที่มีการรับส่งข้อมูลเกิดขึ้นผ่านสายส่งข้อมูลระหว่างคอมพิวเตอร์ทั้ง 2 ระบบ ส่วนชั้นอื่น ๆ จะไม่ได้เชื่อมต่อกันจริง เพียงแต่ทำงานเสมือนกับว่ามีการติดต่อรับส่งข้อมูลกับกลไกในชั้นเดียวกันของคอมพิวเตอร์อีกด้านหนึ่งเท่านั้น

คุณสมบัติข้อที่สองของ OSI 7-Layer Model ก็คือ แต่ละชั้นที่ทำหน้าที่รับส่งข้อมูลจะมีการติดต่อรับส่งข้อมูลข้ามกระโดดไปชั้นอื่น ๆ ในคอมพิวเตอร์ของตัวเองไม่ได้ การส่งข้อมูลจะไล่ลำดับชั้นลงมาจนถึงชั้นที่ 1 ซึ่งจะเป็นชั้นเดียวที่เชื่อมต่อจริงเข้ากับคอมพิวเตอร์ด้านรับข้อมูลผ่านสายส่งข้อมูล และจะทำการรับข้อมูลจากชั้นที่ 1 ไล่ขึ้นไปจนถึงชั้นที่ 7 ตามลำดับ ลำดับชั้นของการส่งข้อมูลชั้นที่ 7 จะเสมือนเชื่อมต่อเข้ากับลำดับลำดับชั้นการรับข้อมูลในชั้นที่ 7 ของคอมพิวเตอร์อีกด้านหนึ่ง ดังแสดงในรูปที่ 2.25

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

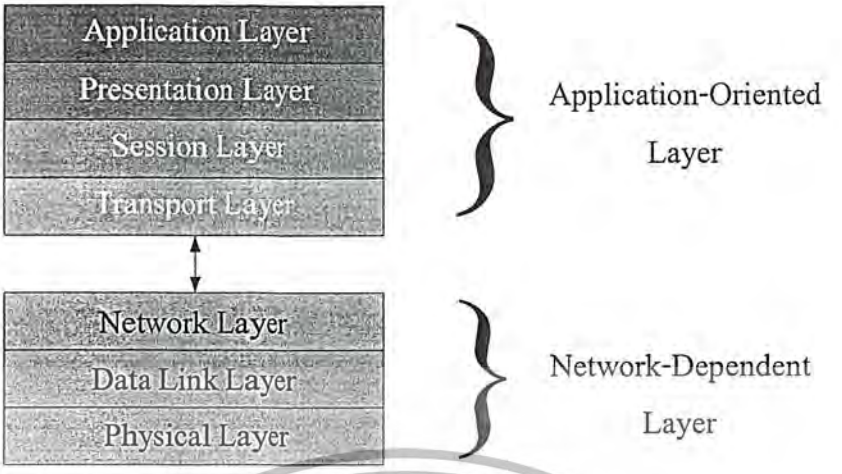


รูปที่ 2.25 การรับส่งข้อมูลของ OSI 7-Layer Model

ผู้ใช้ หรือ User จะติดต่อรับส่งข้อมูลผ่านทางชั้นที่ 7 ซึ่งอยู่ด้านบนสุดของ OSI 7-Layer Model เท่านั้น ในทางทฤษฎีแล้วแต่ละชั้นของการรับส่งข้อมูลจะมีฟังก์ชั่นการทำงานที่แน่นอนและจากกันอย่างเด็ดขาด สามารถที่จะนำแต่ละชั้นของแต่ละบริษัทมาเชื่อมต่อกันได้อย่างไม่มีขีดจำกัด แต่ในทางปฏิบัตินั้น OSI 7-Layer Model จะแบ่งออกเป็น 2 กลุ่มใหญ่ ๆ คือ กลุ่มแรก ได้แก่ 4 ชั้นบน คือ ชั้นที่ 7, 6, 5, 4 ทำหน้าที่เชื่อมต่อรับส่งข้อมูลระหว่างผู้ใช้กับโปรแกรมประยุกต์ ให้รับส่งข้อมูลกับฮาร์ดแวร์ที่อยู่ชั้นล่างได้ถูกต้อง เรียกว่า Application – Oriented Layers ซึ่งจะเกี่ยวข้องกับซอฟต์แวร์เป็นหลัก โดย 4 ชั้นด้านบนนี้มักจะเป็นซอฟต์แวร์ของบริษัทใดบริษัทหนึ่งรวมอยู่อย่างเบ็ดเสร็จในโปรแกรมเดียว จะแยกออกจากกันเป็นชั้น ๆ เพื่อใช้โปรแกรมของบริษัทอื่นได้ลำบาก หรือในบางกรณีอาจทำไม่ได้เลย

กลุ่มที่ 2 จะเป็นชั้นล่าง ได้แก่ ชันที่ 3, 2, 1 ทำหน้าที่เกี่ยวกับการรับส่งข้อมูลผ่านสายส่ง และควบคุมการรับส่งข้อมูล, ตรวจสอบความผิดพลาด รวมทั้งเลือกเส้นทางที่ใช้ในการรับส่งข้อมูล ซึ่งจะเกี่ยวข้องกับฮาร์ดแวร์เป็นหลักเรียกว่า Network – Dependent Layers ดังแสดงในรูปที่ 2.26

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.26 การแบ่งกลุ่มของ OSI 7-Layer Model

สำหรับในส่วนของ 3 ชั้นล่างสุด หรือชั้นที่ 1, 2, 3 นั้น เนื่องจากเกี่ยวข้องกับฮาร์ดแวร์และโปรแกรมควบคุมฮาร์ดแวร์เป็นหลัก ทำให้สามารถแยกแต่ละชั้นออกจากกันได้ง่ายและใช้ผลิตภัณฑ์ของต่างบริษัทกันในแต่ละชั้นได้อย่างไม่เป็นปัญหา

OSI 7-Layer Model ที่แบ่งการรับส่งข้อมูลระหว่างคอมพิวเตอร์สองระบบออกเป็น 7 ชั้นนั้น แต่ละชั้นมีชื่อเรียกและหน้าที่การทำงาน ดังนี้

ชั้นที่ 7 Application Layer

เป็นชั้นที่อยู่บนสุดของกระบวนการรับส่งข้อมูล ทำหน้าที่เชื่อมต่อผู้ใช้เข้ากับระบบคอมพิวเตอร์ โดยรับคำสั่งต่าง ๆ จากผู้ใช้ให้ระบบคอมพิวเตอร์แปลความหมาย และทำงานตามคำสั่งที่ได้รับในระดับโปรแกรมประยุกต์ เช่น แปลความหมายของการกดปุ่มบนเมาส์ให้เป็นคำสั่งในการคัดลอกแฟ้มข้อมูลหรือดึงข้อมูลมาแสดงผลบนจอภาพ เป็นต้น ซึ่งการแปลคำสั่งจากผู้ใช้ส่งให้กับคอมพิวเตอร์รับไปทำงานนี้จะต้องแปลออกมาถูกต้องตามกฎ (Syntax) ที่ใช้ในระบบปฏิบัติการของคอมพิวเตอร์นั้น ๆ ตัวอย่างเช่น ถ้ามีการคัดลอกแฟ้มข้อมูลเกิดขึ้นในระบบ คำสั่งที่ใช้จะต้องสร้างแฟ้มได้อย่างถูกต้อง มีชื่อแฟ้มยาวไม่เกินจำนวนที่ระบบปฏิบัติการใช้อยู่ และชื่อแฟ้มจะต้องประกอบด้วยตัวอักษรตามที่กำหนด ไม่มีตัวอักษรต้องห้าม เป็นต้น สิ่งต่าง ๆ เหล่านี้จะเกิดขึ้นในชั้นที่ 7 ของการสื่อสารข้อมูล รวมทั้งฟังก์ชั่นในการเชื่อมต่อรับส่งข้อมูลระหว่างชั้นที่ 7 กับชั้นที่ 6 ด้วย

ชั้นที่ 6 Presentation Layer

เป็นชั้นที่ทำหน้าที่ตกลงกับคอมพิวเตอร์อีกด้านหนึ่งว่า การรับส่งข้อมูลในระดับโปรแกรมประยุกต์จะมีขั้นตอนและข้อบังคับอย่างไร ข้อมูลที่ทำการรับส่งกันในชั้นที่ 6 นี้ จะอยู่ในรูปแบบของข้อมูลชั้นสูง ซึ่งอยู่ในรูปแบบของคำสั่งที่มีกฎ (Syntax) บังคับอย่างแน่นอน เช่น ในการคัดลอกแฟ้มข้อมูลก็จะมีขั้นตอนย่อยประกอบกัน คือ สร้างแฟ้มที่กำหนดขึ้นมาเสียก่อน จากนั้นจึงเปิดแฟ้มแล้วทำการรับข้อมูลจากปลายทางมาเก็บลงในแฟ้มดังกล่าว โดยเนื้อหาของข้อมูลที่ทำการรับส่งระหว่างกัน ก็

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

คือ คำสั่งของขั้นตอนย่อย ๆ ข้างต้นนั่นเอง คำสั่งเหล่านี้จะต้องหมายถึงว่าจะให้ทำอะไรบ้างและถูกต้องตามกฎด้วย นอกจากนี้ในชั้นที่ 6 ยังทำหน้าที่แปลความหมายของคำสั่งที่ได้รับจากชั้นที่ 7 ให้เป็นคำสั่งระดับปฏิบัติการเพื่อส่งให้ชั้นที่ 5 ต่อไปอีกด้วย

ชั้นที่ 5 Session Layer

ทำหน้าที่ควบคุมจังหวะในการรับส่งข้อมูลของคอมพิวเตอร์ทั้ง 2 ด้านที่รับส่งแลกเปลี่ยนข้อมูลกันให้มีความสอดคล้องกัน (Synchronization) และกำหนดวิธีที่ใช้รับส่งข้อมูล เช่น อาจจะเป็นในลักษณะสลับกันส่ง (Half Duplex) หรือรับส่งข้อมูลพร้อมกันทั้ง 2 ด้าน (Full Duplex) ซึ่งในชั้นที่ 5 นี้จะเป็นชั้นที่ใช้ควบคุมการรับส่งข้อมูลในลักษณะดังกล่าว ข้อมูลที่รับส่งกันในชั้นที่ 5 นี้ จะอยู่ในรูปของ Dialog หรือประโยคของข้อมูลที่สนทนาโต้ตอบกันระหว่างด้านรับและด้านส่ง โดยไม่ได้มองเป็นคำสั่งอย่างในชั้นที่ 6 เช่น เมื่อผู้รับได้รับข้อมูลส่วนแรกจากผู้ส่ง ก็จะโต้ตอบกลับไปให้ผู้ส่งรู้ว่าได้รับข้อมูลส่วนแรกเรียบร้อยแล้ว และพร้อมที่จะรับข้อมูลส่วนที่สองต่อไป คล้ายกับการสนทนาโต้ตอบกันระหว่างผู้รับและผู้ส่งนั่นเอง

ชั้นที่ 4 Transport Layer

ทำหน้าที่เชื่อมต่อการรับส่งข้อมูลระดับสูงของชั้นที่ 5 (ซึ่งมองข้อมูลอยู่ในรูปของ Dialog) มาเป็นข้อมูลที่รับส่งกันในระดับฮาร์ดแวร์ เช่น แปลงค่าหรือชื่อของคอมพิวเตอร์ในเครือข่ายให้เป็น Network Address พร้อมทั้งเป็นชั้นที่ควบคุมการรับส่งข้อมูลจากปลายด้านส่งถึงปลายด้านรับข้อมูล ให้ข้อมูลมีการไหลตลอดเส้นทางตามจังหวัดที่ควบคุมจากชั้นที่ 5 โดยในชั้นที่ 4 นี้ จะเป็นรอยต่อระหว่างการรับส่งข้อมูลของซอฟต์แวร์และฮาร์ดแวร์ การรับส่งข้อมูลของระดับสูงจะถูกแยกจากฮาร์ดแวร์ที่ใช้รับส่งข้อมูลที่ชั้นที่ 4 นี้ และจะไม่มีส่วนใดผูกติดกับฮาร์ดแวร์ที่ใช้รับส่งข้อมูลในระดับล่าง ดังนั้น ฮาร์ดแวร์และซอฟต์แวร์ที่ใช้ควบคุมการรับส่งข้อมูลในระดับล่างลงไปจากชั้นที่ 4 จึงสามารถสับเปลี่ยนและใช้ข้ามไปมากับซอฟต์แวร์รับส่งข้อมูลในระดับสูงที่อยู่ข้างบน (ตั้งแต่ชั้นที่ 4 ขึ้นไปถึงชั้นที่ 7) ได้ง่าย หน้าที่อีกประการของชั้นที่ 4 คือ การควบคุมคุณภาพของการรับส่งข้อมูลให้มีมาตรฐานในระดับที่ตกลงไว้ของทั้งสองฝ่าย และการตัดข้อมูลออกเป็นส่วนย่อย ๆ ให้เหมาะสมกับลักษณะการทำงานของฮาร์ดแวร์ที่ใช้ในเครือข่าย เช่น หากชั้นที่ 5 ต้องการส่งข้อมูลที่มีความยาวมากเกินกว่าที่ระบบเครือข่ายจะส่งได้ ชั้นที่ 4 ก็จะทำหน้าที่ตัดข้อมูลออกเป็นส่วนย่อย ๆ แล้วส่งไปให้ผู้รับ ข้อมูลที่ได้รับปลายทางก็จะถูกนำมาต่อกันที่ชั้นที่ 4 ของด้านผู้รับแล้วส่งให้ชั้น 5 ต่อไป

ชั้นที่ 3 Network Layer

ทำหน้าที่เชื่อมต่อคอมพิวเตอร์ของด้านรับและด้านส่งเข้าหากันผ่านระบบเครือข่าย พร้อมทั้งเลือกหรือกำหนดเส้นทางที่ใช้ในการรับส่งข้อมูลระหว่างกัน และส่งผ่านข้อมูลที่ได้รับไปยังอุปกรณ์ในเครือข่ายต่าง ๆ จนกระทั่งถึงปลายทาง ในชั้นที่ 3 นี้ข้อมูลที่รับส่งกันจะอยู่ในรูปแบบของกลุ่มข้อมูลที่เรียกว่า Packet หรือ Frame ซึ่งข้อมูลที่ชั้น 4, 5, 6, 7 มองเห็นเป็นคำสั่งและ Dialog ต่าง ๆ นั้นจะถูกแปลงและผนึกรวมอยู่ในรูปของ Packet หรือ Frame ที่มีเพียงที่อยู่ของผู้รับ, ผู้ส่ง, ลำดับการรับส่งและส่วนของข้อมูลเท่านั้น ตัวเนื้อหาของข้อมูลจะไม่มีผลใด ๆ ในการรับส่งข้อมูลเลย ไม่ว่าข้อมูลในระดับสูงจะเป็นวิดีโอ, ภาพ, เสียง หรือข้อมูลใดก็ตาม แต่ในชั้นที่ 3 จะมองข้อมูลทั้งหมดเป็น Packet หรือ Frame เท่านั้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หน้าที่อีกประการหนึ่งของชั้นที่ 3 นี้ คือการทำ Call Setup หรือเรียกติดต่อคอมพิวเตอร์ปลายทางก่อนการรับส่งข้อมูล และทำการ Call Clearing หรือยกเลิกการติดต่อเมื่อรับส่งข้อมูลจบลงแล้ว ในกรณีที่การรับส่งข้อมูลนั้นต้องมีการติดต่อกันก่อน

ชั้นที่ 2 Data Link Layer

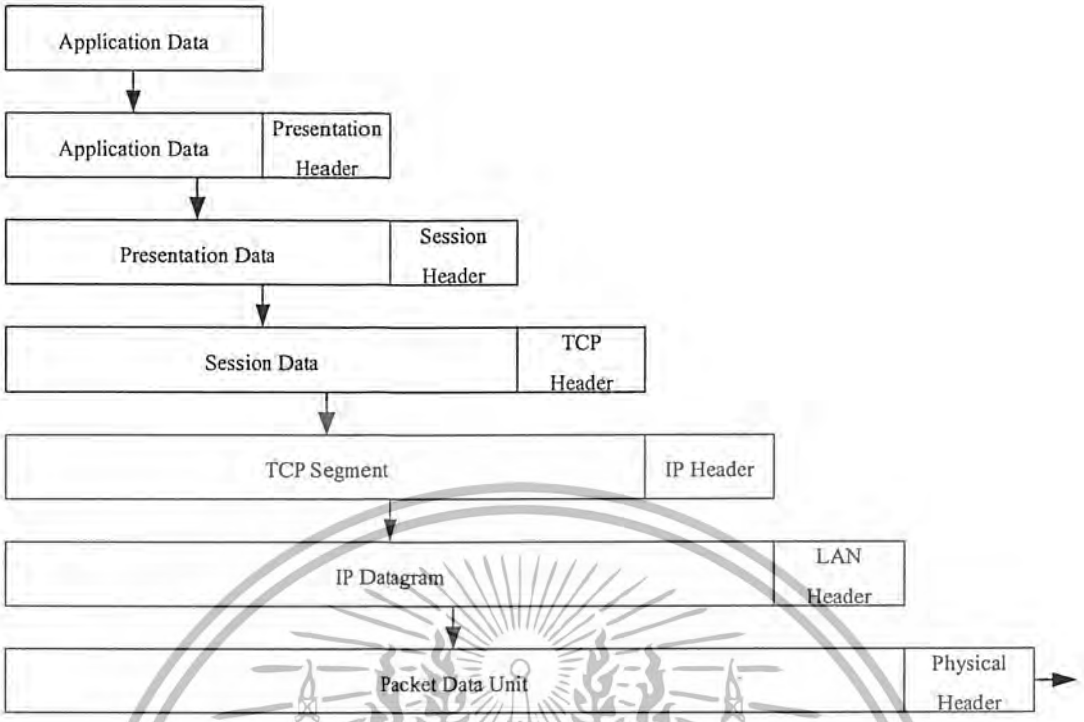
เป็นชั้นที่ทำหน้าที่เชื่อมต่อการรับส่งข้อมูลในระดับฮาร์ดแวร์ โดยเมื่อมีการสั่งให้รับข้อมูลจากในชั้นที่ 3 ลงมา ชั้นที่ 2 จะทำหน้าที่แปลคำสั่งนั้นให้เป็นคำสั่งควบคุมฮาร์ดแวร์ที่ใช้รับส่งข้อมูล ทำการตรวจสอบข้อผิดพลาดในการรับส่งข้อมูลระดับฮาร์ดแวร์ และแก้ไขข้อผิดพลาดที่ตรวจพบนั้น ข้อมูลที่อยู่ในชั้นที่ 2 นี้จะอยู่ในรูปของ Frame คือ กลุ่มของข้อมูลที่มีรูปร่างตามข้อบังคับของฮาร์ดแวร์ที่ใช้ในการรับส่งข้อมูล เช่น ถ้าฮาร์ดแวร์ที่ใช้เป็น Ethernet LAN ข้อมูลก็จะมีรูปร่างของ Frame ตามที่ระบุไว้ในมาตรฐานของ Ethernet หากว่าฮาร์ดแวร์ที่ใช้รับส่งข้อมูลเป็นชนิดอื่น เช่น Token Ring LAN หรือ Fiber Distributed Data Interface (FDDI) รูปร่างของ Frame ที่ใช้ในการรับส่งข้อมูลก็จะเปลี่ยนไปตามมาตรฐานนั้น ๆ

ชั้นที่ 1 Physical Layer

เป็นชั้นล่างสุดของขั้นตอนในการรับส่งข้อมูลของ OSI 7-Layer Reference Model ซึ่งเป็นชั้นเดียวที่มีการเชื่อมต่อกันทางกายภาพระหว่างคอมพิวเตอร์ 2 ระบบที่ทำการรับส่งข้อมูลกัน ในชั้นที่ 1 นี้จะกำหนดคุณสมบัติทางกายภาพของฮาร์ดแวร์ที่ใช้เชื่อมต่อระหว่างคอมพิวเตอร์ทั้ง 2 ระบบ เช่น สายที่ใช้รับส่งข้อมูลจะเป็นแบบไหน, ข้อต่อหรือปลั๊กที่ใช้การรับส่งข้อมูลมีมาตรฐานอย่างไร, ใช้ไฟกี่โวลต์, ความเร็วในการรับส่งข้อมูลเป็นเท่าใด, สัญญาณที่ใช้รับส่งข้อมูลในสายมีรูปร่างอย่างไร ข้อมูลในชั้นที่ 1 นี้จะมองเห็นเป็นการรับส่งข้อมูลที่ละบิตเรียงต่อกันไป โดยไม่มีการพิจารณาเรื่องความหมายของข้อมูลเลย การรับส่งจะส่งข้อมูล "0" หรือ "1" ไปให้คอมพิวเตอร์ด้านรับข้อมูลในระดับฮาร์ดแวร์เท่านั้น หากการรับส่งข้อมูลมีปัญหาเนื่องจากฮาร์ดแวร์ เช่น สายสัญญาณที่ใช้รับส่งข้อมูลขาด, อุปกรณ์เสียหาย ก็จะเป็นหน้าที่ของชั้นที่ 1 นี้เช่นกันที่จะตรวจสอบและแจ้งข้อผิดพลาดนั้นให้ชั้นอื่น ๆ ที่อยู่เหนือขึ้นไปทราบ

ในการรับส่งข้อมูลใน OSI 7-Layer Model นั้น ข้อมูลจากชั้นบนสุด คือ ชั้นที่ 7 เมื่อถูกส่งลงไปในชั้นถัดลงไป ข้อมูลเดิมก็จะถูกผนวกรวมกับข้อมูลที่ใช้ควบคุมของแต่ละชั้นซ้อน ๆ กันเป็นลำดับเท่ากับจำนวนชั้นที่ผ่านลงไป ตัวอย่างเช่น Application Data เมื่อถูกส่งลงไปยังชั้นถัดไปก็จะถูกผนึกด้วย Application Header และทั้ง Application Header และ Application Data จะรวมกันเป็นข้อมูลของชั้นที่อยู่ถัดลงไปอีก ซึ่งชั้นที่อยู่ถัดลงไปก็จะผนึกข้อมูลนี้ด้วย Header ของมันอีก และทำไปเรื่อย ๆ จนกระทั่งถึงชั้นล่างสุดซึ่งเป็น Physical Layer ซึ่งเมื่อข้อมูลถูกส่งไปถึงปลายทาง ข้อมูลที่ได้รับจะถูกแยก Header ที่เพิ่มเข้ามานี้ออกทีละชั้น ซึ่งเป็นกระบวนการย้อนกลับกับด้านส่ง จนกระทั่งถึงชั้นบนสุด จึงจะเป็นข้อมูลของ Application Data ให้ผู้รับตามต้องการ ดังแสดงในรูปที่ 2.27

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.27 การรับส่งข้อมูลแต่ละชั้นของ OSI 7-Layer Model

เนื่องจาก OSI เกิดขึ้นมาหลังจากที่ TCP/IP หรือ Transmission Control Protocol / Internet Protocol ได้มีการใช้งานกันอย่างแพร่หลายไปแล้ว และการที่ TCP/IP เป็นโปรโตคอลชนิดที่ให้ใช้ได้ฟรี ไม่ต้องจ่ายค่าลิขสิทธิ์ การใช้งาน TCP/IP ก็ยิ่งมีจำนวนผู้ใช้เพิ่มมากขึ้นไปอีกจนถือเป็นมาตรฐานที่มีผู้ใช้รับส่งข้อมูลมากที่สุดในปัจจุบัน

เมื่อ TCP/IP เป็นมาตรฐานที่เกิดขึ้นก่อน OSI 7-Layer Model มาตรฐาน TCP/IP จึงไม่ใช่มาตรฐานเดียวกันกับของ OSI โดย TCP/IP จะมีการแบ่งจำนวนขั้นตอนที่ใช้รับส่งข้อมูลระหว่างเครื่องคอมพิวเตอร์สองระบบออกเป็น 4 ชั้นเท่านั้น หรือเรียกว่าเป็น TCP/IP Stack โดยมีชื่อเรียก ดังนี้

ชั้นที่ 4 Process Layer

จะเป็น Application Protocol ที่ทำหน้าที่เชื่อมต่อกับผู้ใช้และให้บริการต่าง ๆ เช่น FTP, Telnet, SNMP เป็นต้น

ชั้นที่ 3 Host – to – Host Layer

จะเป็น TCP หรือ UDP ที่ทำหน้าที่คล้ายกับชั้นที่ 4 ของ OSI 7-Layer Model คือ ควบคุมการรับส่งข้อมูลจากปลายด้านส่งถึงปลายด้านรับข้อมูล และตัดข้อมูลออกเป็นส่วนย่อยให้เหมาะกับเครือข่ายที่ใช้รับส่งข้อมูล รวมทั้งประกอบข้อมูลส่วนย่อย ๆ นี้เข้าด้วยกันเมื่อถึงปลายทาง

ชั้นที่ 2 Internetwork Layer

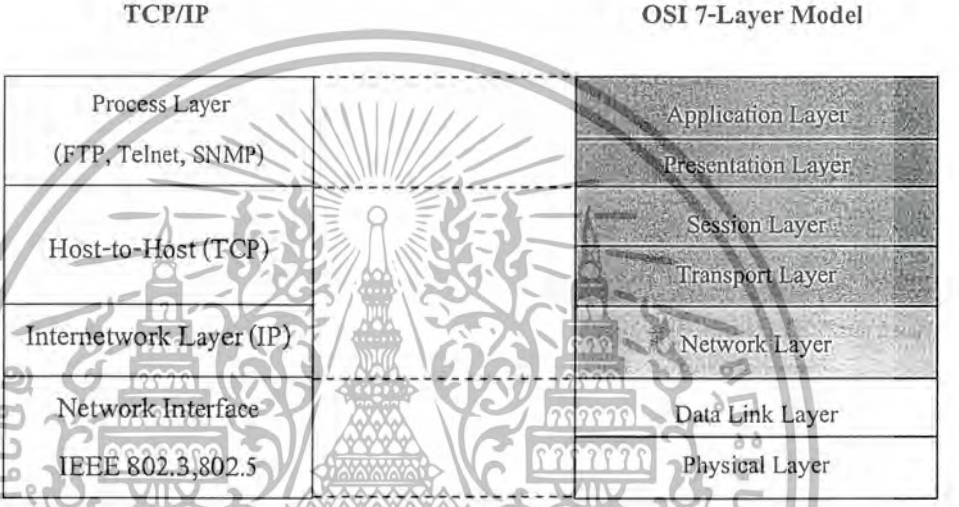
ได้แก่ ส่วนของ IP Protocol ซึ่งทำหน้าที่คล้ายกับชั้นที่ 3 ของ OSI 7-Layer Model คือ เชื่อมต่อคอมพิวเตอร์เข้ากับระบบเครือข่ายที่อยู่ชั้นล่างลงไป และทำหน้าที่เลือกเส้นทางการรับส่งข้อมูล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ผ่านอุปกรณ์เครือข่ายต่าง ๆ จนไปถึงผู้รับข้อมูล ในชั้นนี้จะจัดการกับกลุ่มข้อมูลในลักษณะที่เรียกว่า Frame ในรูปแบบของ TCP/IP ที่เรารู้จักกัน

ชั้นที่ 1 Network Interface

คือ ชั้นที่ควบคุมฮาร์ดแวร์การรับส่งข้อมูลผ่านเครือข่าย ซึ่งเทียบได้กับชั้นที่ 1 และ 2 ของ OSI 7-Layer Model ในชั้นนี้จะทำหน้าที่เชื่อมต่อกับฮาร์ดแวร์และควบคุมการรับส่งข้อมูลในระดับฮาร์ดแวร์ของเครือข่าย ซึ่งที่ใช้กันอยู่จะเป็นไปตามมาตรฐานของ IEEE เช่น IEEE802.3 ซึ่งเป็นการเชื่อมต่อผ่าน LAN แบบ Ethernet LAN เป็นต้น



รูปที่ 2.28 โปรโตคอล TCP/IP เมื่อเทียบกับ OSI 7-Layer Model

จากข้างต้นเราจะเห็นได้ว่าที่จริงแล้วโปรโตคอล TCP/IP นั้นแบ่งออกเป็น 2 โปรโตคอลซ้อนกันอยู่ คือ TCP อยู่ชั้นบนและ IP อยู่ในชั้นถัดลงมา นั่นคือ TCP/IP ไม่ได้เป็นโปรโตคอลเดียวกันทั้งหมดและไม่ได้เชื่อมติดกันเป็นชิ้นเดียว แต่ว่า TCP ก็มีมาตรฐานของ Frame ที่ใช้รับส่งข้อมูลของมันเอง และมีหน้าที่ในการรับส่งข้อมูลแตกต่างไปจาก IP ซึ่งในการรับส่งข้อมูลนั้น Frame ของ TCP ที่อยู่ชั้นบนทั้งหมดจะถูกผนึกอยู่ในส่วนที่เป็นข้อมูลของ IP เหมือนกับที่แต่ละชั้นของ OSI 7-Layer Model ผนึกข้อมูลอยู่ในชั้นถัดไปนั่นเอง

ถึงแม้ว่า TCP/IP จะไม่ได้มีการแบ่งชั้นของการสื่อสารข้อมูลตรงตาม OSI 7-Layer Model และไม่ได้เป็นมาตรฐานเดียวกัน แต่ OSI ก็ออกแบบมาให้เปิดกว้างและเข้ากันได้ดีกับ TCP/IP โดย TCP จะเทียบได้กับประมาณชั้นที่ 4 ของ OSI และ IP จะเทียบได้กับประมาณชั้นที่ 3 ของ OSI แม้จะไม่ลงตัวตรงกันพอดีนัก แต่ก็สามารถเชื่อมต่อทำงานด้วยกันได้ ทำให้มาตรฐานของ OSI สามารถนำ TCP/IP มาใช้งานร่วมกันได้เป็นอย่างดี เมื่อเรากลับไปมองมาตรฐานของ OSI 7-Layer Model ที่เปิดกว้างให้เราเลือกใช้มาตรฐานต่าง ๆ ของแต่ละชั้นมาใช้งานร่วมกันแล้ว จะพบว่าการกำหนดมาตรฐานของ OSI ได้บรรลุวัตถุประสงค์เป็นอย่างดี คือ เราสามารถเลือกใช้อุปกรณ์ฮาร์ดแวร์เครือข่ายและโปรแกรมควบคุมในชั้นที่

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1 และ 2 จากบริษัทใดก็ได้มาเชื่อมต่อเข้าด้วยกัน แล้วนำ TCP/IP ซึ่งมีใช้งานกันอย่างแพร่หลายมาใช้ในชั้นที่ 4 และ 3 ตามลำดับ ส่วนชั้นที่ 5 ถึงชั้นที่ 7 จะเป็นแอพพลิเคชั่นที่ต้องการ

2.5.2 การอ้างอิงอุปกรณ์ในเครือข่าย

แนวความคิดหลักของระบบเครือข่ายคอมพิวเตอร์ก็คือ การเชื่อมโยงอุปกรณ์เข้าด้วยกัน ไม่ว่าจะเป็นเครื่อง Server (หรือที่เรียกว่า Host) และอุปกรณ์ในเครือข่ายอื่น ๆ เช่น Router, Printer เพื่อให้สามารถใช้อุปกรณ์ร่วมกันได้ หรือสามารถส่งข้อมูลไปมาระหว่างกันได้ถูกต้อง เมื่อมีการเชื่อมต่อกันแล้วก็จำเป็นต้องมีการกำหนดหรือระบุเลขหมายของอุปกรณ์ทุกชิ้นทุกชนิดในเครือข่าย เพื่อให้อ้างอิงได้โดยไม่ซ้ำกัน เพราะถ้าซ้ำกันแล้วการรับส่งข้อมูลอาจจะไม่ถึงมือผู้รับปลายทางได้อย่างถูกต้อง เลขหมายดังกล่าวจะเรียกว่า แอดเดรส (Address) หรือเลขหมายประจำตัวที่มีข้อกำหนดเป็นมาตรฐาน ซึ่งในการใช้งานโปรโตคอล TCP/IP ที่เชื่อมโยงเครือข่ายอินเตอร์เน็ตนี้ เลขหมายที่ใช้อ้างถึงกันจะใช้เป็นตัวเลขที่เรียกว่า IP Address หรือหมายเลข IP

หมายเลข IP ถูกกำหนดขึ้นมาให้เป็นหมายเลขอ้างอิงประจำตัวของอุปกรณ์ต่าง ๆ ที่เชื่อมต่ออยู่ในเครือข่ายอินเตอร์เน็ต โดยการกำหนดหมายเลข IP ให้แต่ละเครื่องหรือแต่ละอุปกรณ์นี้จะต้องไม่ซ้ำกัน ซึ่งหมายเลข IP นี้ จะไม่ผูกติดกับตัวฮาร์ดแวร์แต่อย่างใด จึงสามารถกำหนดใหม่หรือแก้ไขเปลี่ยนแปลงได้ เมื่อมีการเปลี่ยนตัวฮาร์ดแวร์ ทั้งนี้เนื่องจากเป็นการกำหนดด้วยซอฟต์แวร์ แตกต่างกับหมายเลข MAC Address (Media Access Control Address) ซึ่งเป็นหมายเลขประจำตัวของอุปกรณ์ที่ต่ออยู่ในเครือข่ายค่า MAC Address จะถูกกำหนดจากบริษัทผู้ผลิตอุปกรณ์ตั้งแต่เริ่มผลิต เช่น อุปกรณ์ Network Interface Card (NIC) จะมีค่า MAC Address ประจำตัวที่ไม่ซ้ำกันและไม่สามารถแก้ไขได้ เป็นการระบุค่าอ้างอิงของอุปกรณ์ฮาร์ดแวร์ในระดับล่างสุด (Physical Layer) ของกลไกการรับส่งข้อมูลภายในเครือข่าย ถ้าใช้หมายเลข MAC Address สำหรับระบุอ้างอิงกันในเครือข่ายแล้วจะเกิดปัญหามากเมื่อมีการเปลี่ยนหรือเคลื่อนย้ายต้องทำการกำหนดระบบเครือข่ายใหม่ การที่หมายเลข IP ถูกใช้อ้างอิงในการติดต่อกันด้วยโปรโตคอล TCP/IP เพราะการใช้หมายเลข IP จะยืดหยุ่นและคล่องตัวกว่า

การทำงานของโปรโตคอล IP จำเป็นต้องอาศัยหมายเลข IP นี้เพื่อระบุและอ้างถึงอุปกรณ์ต่าง ๆ ที่ต่ออยู่ในเครือข่ายว่าจะเป็น Web Server, Mail Server, Router หรืออื่น ๆ หมายเลข IP จะเป็นค่าตัวเลขขนาด 32 บิต ถูกแบ่งออกเป็น 4 ส่วน ส่วนละ 8 บิต และคั่นแต่ละส่วนด้วยเครื่องหมายจุด ดังนั้นในแต่ละส่วนจะมีค่าตั้งแต่ 0 – 255

ค่าของหมายเลข IP จะถูกแบ่งเป็น 2 ความหมาย คือ ค่าของหมายเลขอุปกรณ์ในเครือข่าย (Host Address) และค่าของหมายเลขเครือข่าย (Network Address) ตัวอย่างเช่น มีเครื่อง Server เชื่อมต่ออยู่ในเครือข่าย 2 เครื่อง โดยแต่ละเครื่องมี IP Address คือ 205.144.78.2 และ 205.144.78.3 เครื่องทั้งสองมีหมายเลขเครือข่ายเหมือนกัน คือ 205.144.78 แสดงว่าเครื่องทั้งสองต่อเชื่อมอยู่ในเครือข่ายเดียวกัน แต่มีหมายเลขเครื่องที่แตกต่างกันคือ 2 และ 3 ตามลำดับ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เพื่อไม่ให้การกำหนดแจกจ่ายหมายเลข IP ซ้ำซ้อนกัน จึงมีหน่วยงานกลางทำหน้าที่กำหนดหมายเลข IP และแจกจ่ายให้แต่ละองค์กรได้ใช้งาน คือ หน่วยงาน InterNIC บริษัทหรือองค์กรที่ต้องการหมายเลข IP

2.5.3 Data Packet

เมื่อมีการรับหรือส่งข้อมูลกันในระบบเครือข่ายอินเตอร์เน็ตนั้น ตัวข้อมูลจะถูกทำให้มีขนาดเล็กลงโดยแบ่งออกเป็นส่วนย่อย ๆ เรียกว่า Data Packet หรือ Datagram ซึ่งการจัดแบ่งข้อมูลให้เป็นส่วนย่อยลงนี้มีประโยชน์ คือ ทำให้เครือข่ายนั้นสามารถรองรับการติดต่อและรับส่งข้อมูลกันได้อย่างราบรื่นไม่ติดขัด หรือพบปัญหาเครือข่ายทำงานช้าเมื่อมีการรับส่งข้อมูลขนาดใหญ่ เนื่องจากสายสัญญาณเชื่อมโยงเป็นสื่อที่ต้องแบ่งกันใช้งาน นอกจากนี้การแบ่งข้อมูลออกเป็นส่วนย่อย ๆ ยังทำให้สามารถเพิ่มกระบวนการตรวจทานความถูกต้องของข้อมูลที่ปลายทาง และแก้ไขเมื่อข้อมูลผิดพลาดหรือตกหล่นได้โดยง่ายอีกด้วย

จากที่กล่าวมาแล้วว่าสายเชื่อมโยงอุปกรณ์นั้นเป็นสิ่งที่ต้องใช้งานร่วมกัน เมื่อมีอุปกรณ์ใดต้องการส่งข้อมูล อุปกรณ์อื่น ๆ ก็ต้องรอให้การส่งข้อมูลนั้นเสร็จสิ้นก่อนจึงจะสามารถส่งข้อมูลของตนได้ โดยเฉพาะอย่างยิ่งในกรณีที่มีการส่งข้อมูลขนาดใหญ่ ถ้าไม่มีการแบ่งข้อมูลให้เป็นส่วนเล็กลงเพื่อทยอยส่งไปยังปลายทางซึ่งเป็นการแบ่งเวลาให้กับอุปกรณ์อื่น ๆ ได้ใช้สายสัญญาณแล้ว เครือข่ายนั้นก็อาจจะเกิดปัญหาติดขัดได้ ทั้งนี้เมื่อ Datagram ถูกส่งไปยังปลายทางแล้ว ก็จะมีกระบวนการร่วมข้อมูลย่อยเหล่านี้ให้กลับคืนสู่สภาพเดิมได้ต่อไป

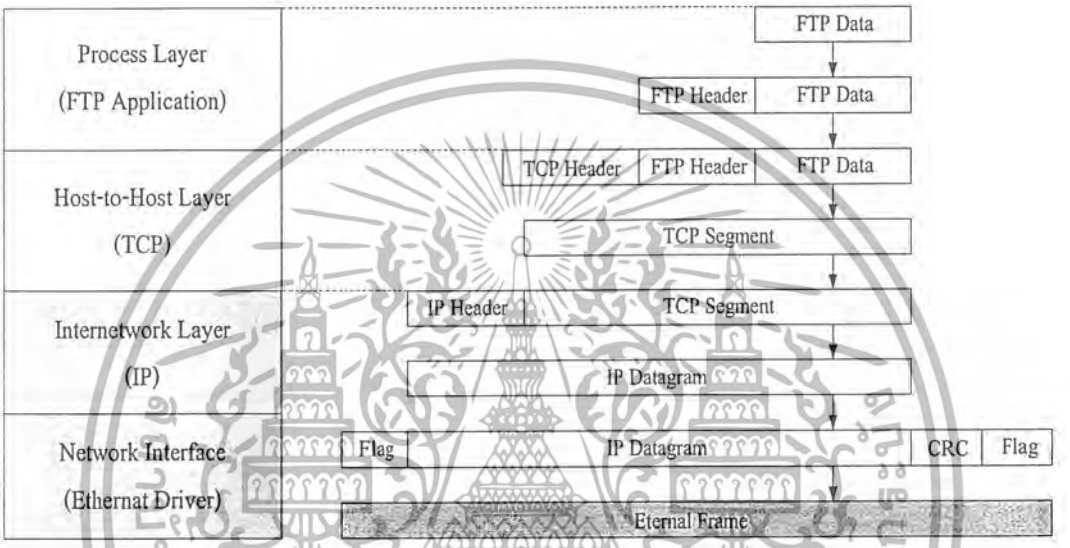
ประโยชน์อีกประการหนึ่งในการแยกข้อมูลให้เป็นส่วนย่อย ๆ คือ การแก้ไขและตรวจสอบข้อมูลที่เสียหายในการส่งข้อมูล จะสามารถทำได้อย่างมีประสิทธิภาพ ซึ่งการส่งข้อมูลผ่านสายสัญญาณต่าง ๆ เรามักจะพบปัญหาสัญญาณรบกวนหรือสัญญาณขาดหายระหว่างการส่งอยู่บ่อย ๆ ทำให้ข้อมูลที่ส่งไปยังผู้รับไม่ถูกต้องครบถ้วน ซึ่งปัญหานี้สามารถแก้ไขได้ เนื่องจากข้อมูลที่ถูกแบ่งเป็น Datagram จะมีขนาดเล็กลง ทำให้สามารถเพิ่มการตรวจสอบการรับส่งข้อมูลนั้นได้ดีขึ้น เช่น เทคนิคการคำนวณ (Check-Sum) จะคำนวณค่าของข้อมูลที่ส่งไปและได้รับ ถ้าตรงกันก็แสดงว่าการรับส่ง Datagram นั้นถูกต้อง แต่ถ้าผลการคำนวณไม่ตรงกัน ด้านผู้รับข้อมูลก็จะส่งสัญญาณมาให้เพื่อส่งเฉพาะ Datagram นั้นใหม่อีกครั้ง โดยไม่ต้องส่งข้อมูลทั้งหมดมาอีก ทำให้สามารถแก้ไขข้อมูลที่ผิดพลาดได้อย่างรวดเร็ว

ตัวข้อมูลที่ถูกแยกออกเป็น Data Packet หรือ Datagram นี้จะมีลักษณะเป็นข้อมูลแบบต่อเนื่อง (Stream Byte) คือ มีการกำหนดลำดับก่อนหลังของข้อมูลเพื่อให้ประกอบข้อมูลย่อยคืนสู่สภาพเดิมได้อย่างถูกต้อง และมีรูปแบบหรือฟอร์แมต (Format) ที่แน่นอน คือ Datagram จะประกอบด้วยส่วนของ Header และส่วนของตัวข้อมูล (Body) โดยในส่วนของ Header จะมีข้อมูลต่าง ๆ ที่ระบุที่อยู่ปลายทางที่ต้องส่งข้อมูลไป, หมายเลขต้นทางที่ส่งข้อมูลมา, ค่าบอกขนาดความยาวของ Datagram นี้และข้อมูลอื่น ๆ สำหรับในส่วนของ Body อาจจะเป็นเนื้อหาข้อมูลใด ๆ เช่น ข้อความใน E-Mail, แฟ้มข้อมูลบางส่วน หรืออาจจะเป็น Datagram ของข้อมูลในรูปแบบอื่น ๆ ที่ถูกผนึก (Encapsulation) มาด้วย เป็นต้น ซึ่ง Datagram ที่ใช้ในเครือข่ายอินเตอร์เน็ตนี้จะเรียกว่า IP Datagram

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.5.4 Encapsulation

ก่อนที่ข้อมูลใดจะถูกส่งผ่านไปในเครือข่ายอินเตอร์เน็ตได้ ก็จะต้องถูกแยกเป็นส่วนย่อย ๆ ที่เรียกว่า Datagram และถูกผนึกหรือทำ Encapsulation เข้าไปกับโปรโตคอล IP หรือเรียกว่าเป็น IP Datagram ก่อนจึงจะส่งผ่านไปในเครือข่ายอินเตอร์เน็ตได้ เนื่องจากโปรโตคอล IP จะมีข้อมูลในการระบุเส้นทางการส่งผ่านข้อมูลไปยังปลายทางได้นั่นเอง การผนึกข้อมูลหนึ่งไปเป็นข้อมูลในอีกรูปแบบหนึ่งนี้เป็นกลไกที่สำคัญของการใช้งานโปรโตคอล TCP/IP มาก โดยขบวนการที่ใช้จะมีขั้นตอนคร่าว ๆ ดังรูปที่ 2.29



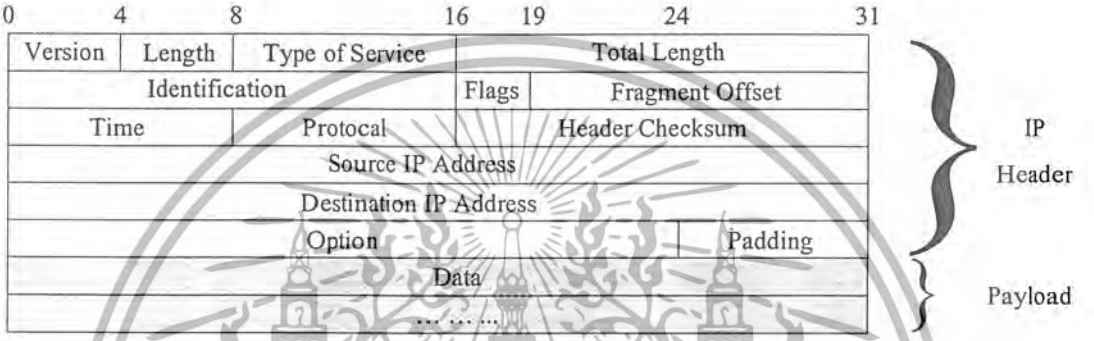
รูปที่ 2.29 ตัวอย่าง Encapsulation ของข้อมูล FTP เทียบกับ TCP/IP Layer

จากรูปที่ 2.29 เริ่มต้นมีการใช้งานโปรแกรมรับส่งข้อมูล เช่น เมื่อเรียกโปรแกรม FTP โปรแกรมแอพพลิเคชั่นจะเตรียมข้อมูลเพื่อส่งผ่านไปในเครือข่ายอินเตอร์เน็ตหลังจากประมวลผล FTP เตรียมข้อมูลและแยกส่วนเป็น FTP Data หรือ FTP Datagram แล้วจะมีส่วนของ FTP Header เพิ่มเข้าไปในส่วนของข้อมูล เมื่อมาถึงชั้น Transport หรือ Host-to-Host Layer ซึ่งโปรโตคอล TCP เป็นผู้รับผิดชอบจะมีการสร้าง TCP Segment โดยการเพิ่มส่วนของ TCP Header เข้าไปและมีการผนึกส่วนของ FTP Datagram รวมกัน จากนั้น TCP Segment นี้จะถูกส่งต่อไปยัง Layer ระดับล่างลงไปคือ Internetwork Layer ในชั้นนี้โปรโตคอล IP จะทำงานโดยการเพิ่มส่วน IP Header รวมกันกับ TCP Segment เข้าไป เรียกว่าเป็น IP Datagram ก็เป็นอันเสร็จสิ้นขึ้นตอนการผนึกหรือ Encapsulation ข้อมูลจากระดับบนสุดลงมา เพื่อให้ส่งผ่าน IP Datagram นี้ไปยังเครือข่ายอินเตอร์เน็ตได้ และในขั้นตอนสุดท้ายก่อนที่จะส่ง Datagram ออกไปยังสายสัญญาณ ในชั้น Network Interface จะมีการแปลงข้อมูลและเพิ่มส่วน Error Correction และ Flag เพื่อให้การส่งข้อมูลนั้นไม่ผิดพลาด จากนั้นก็แปลงข้อมูลเป็นสัญญาณไฟฟ้าส่งผ่านสายสัญญาณที่เชื่อมโยงอยู่ต่อไป ซึ่งจากตัวอย่างนี้มีการส่งผ่านข้อมูลไปในเครือข่ายแบบ Ethernet ดังนั้นในชั้นสุดท้ายข้อมูลก็จะถูกแปลงเป็น Ethernet Frame เสียก่อน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.5.5 IP Datagram

จากกระบวนการ Encapsulation นั้น เราทราบว่าข้อมูลในการติดต่อกันไม่ว่าจะเป็นเนื้อความในอีเมล์หรือไฟล์ที่ส่งไปมา จะถูกผนึกข้อมูลหรือ Encapsulated ไปเป็นรูปของ IP Datagram และสุดท้ายก็จะถูกแปลงเป็น Ethernet Frame หรือ Frame ข้อมูลในรูปแบบอื่น ๆ ตามลักษณะการเชื่อมต่อทางกายภาพ เช่น Ethernet หรือ Token Ring เป็นต้น เพื่อให้สามารถส่งข้อมูลออกสู่เครือข่ายและข้ามเครือข่ายไปสู่อินเตอร์เน็ตได้ ตัวข้อมูลที่ถูกแปลงมาเป็น IP Datagram นี้จะประกอบด้วย 2 ส่วน คือ ส่วน IP Header ที่มีขนาด 32 ไบต์ และส่วนเนื้อข้อมูลที่เรียกว่า Payload ขนาดของ IP Datagram มีขนาดไม่แน่นอน และมีลักษณะตามรูปที่ 2.30



รูปที่ 2.30 รูปของ IP Datagram ประกอบด้วยส่วน Header และ Payload

ส่วนของ IP Header มีการแบ่งย่อยเพื่อระบุตัวแปรในการใช้งานต่าง ๆ ดังนี้

Version มีขนาด 4 บิต ถูกกำหนดค่าเป็น 4 ในกรณีที่ใช้หมายเลข IP เป็น IPv4 ในอนาคตจะมีค่าเป็น 6 เมื่อมีการเปลี่ยนมาใช้ IPv6

Length มีขนาด 4 บิต ซึ่งเป็นค่าความยาวของ IP Header นี้

Type of Service เป็นฟิลด์ข้อมูลขนาด 8 บิต เพื่อบอกให้ทราบว่าจะดำเนินการกับข้อมูลนี้อย่างไร เช่น Low Delay, High Throughput เป็นต้น แต่ในการใช้งานจริงอุปกรณ์ Router ที่ส่งผ่านข้อมูลจะไม่สนใจข้อมูลนี้

Total Length มีขนาด 16 บิต เก็บข้อมูลแสดงค่าความยาวสุทธิของ IP Datagram นี้เป็นจำนวนไบต์ ดังนั้นขนาดของ IP Datagram จะมีความยาวได้ไม่เกิน 2^{16} หรือ 65,535 ไบต์ ซึ่งในส่วนของ IP Header จะมีขนาดอย่างน้อย 20 ไบต์ ดังนั้นเนื้อหาข้อมูลหรือ Payload ของ IP Datagram ใด ๆ จะมีขนาดไม่เกิน 65,515 ไบต์ และในการส่งผ่านข้อมูลกันในอินเตอร์เน็ต ตัว IP Datagram จะมีขนาดเล็กที่สุดที่ 576 ไบต์ ซึ่งเป็นขนาดเล็กที่สุดซึ่งไม่สามารถแยกย่อยลงไปกว่านี้ได้อีก

Identification เป็นข้อมูลที่บอกให้ทราบว่า IP Datagram นั้นมาจากที่ใด โดยเฉพาะกรณีที่ข้อมูลถูกแยกเป็นส่วนย่อย ๆ แล้ว

Flags, Fragment Offset เป็นส่วนข้อมูลที่ใช้ระบุการแยกและรวมข้อมูล เพื่อให้ข้อมูลที่ถูกแยกเป็นข้อมูลย่อย (Fragment) สามารถกลับมารวมกันใหม่ตามลำดับได้ถูกต้อง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Time หรือ Time to Live เป็นข้อมูลแสดงจำนวนเวลามากที่สุดของ IP Datagram นี้ที่จะสามารถส่งผ่านเครือข่ายไปยังปลายทางได้ โดยมีหน่วยเป็นวินาที และตามปกติจะมีค่าเป็น 32 โดยในระหว่างที่ข้อมูล IP Datagram ถูกส่งผ่าน Router ตัว Router ก็จะลดค่า Time to Live ลง 1 ค่าเสมอ ทำให้สามารถนำค่า Time นี้ไปใช้นับจำนวนเครือข่ายที่ IP Datagram นี้ถูกส่งผ่านไปยังปลายทางได้ ซึ่งเรียกว่า Hop Count

Protocol เป็นข้อมูลการระบุโปรโตคอลที่ทำงานใน Layer ข้างบนซึ่งผนึกลงมาใน IP Datagram ซึ่งตัวอย่างของโปรโตคอลในชั้นบนที่ถูกผนึกมาให้ IP นี้ก็ได้แก่ โปรโตคอล ICMP, TCP, UDP เป็นต้น ส่วนค่าที่อยู่ในฟิลด์นี้จะเป็นตัวเลขตามตารางที่ 2.1

โปรโตคอล	ค่าที่กำหนดในฟิลด์	คำอธิบาย
ICMP	1	Internet Control Message Protocol
TCP	6	Transmission Control Protocol
BGP	8	Border Gateway Protocol
UDP	17	User Datagram Protocol
OSPF	89	Open Shortest Path First

ตารางที่ 2.1 การระบุโปรโตคอลที่ทำงานใน Layer ข้างบนซึ่งผนึกลงมาใน IP Datagram

Header Checksum เป็นส่วนของข้อมูลที่ใช้ตรวจสอบความถูกต้องของข้อมูล เฉพาะในส่วนของ IP Header โดยไม่เกี่ยวกับส่วนของ Payload ซึ่งการตรวจสอบความถูกต้องของข้อมูลนี้โปรโตคอล IP จะทำหน้าที่ในการคำนวณและตรวจสอบ โดยกรณีที่เกิดความผิดพลาดของข้อมูล IP Datagram นั้นจะถูกยกเลิกหรือไม่รับข้อมูลเข้ามาใช้งาน

Source IP Address เป็นส่วนเก็บข้อมูลหมายเลข IP ต้นทางที่ IP Datagram นี้ถูกส่งมา

Destination IP Address เป็นส่วนเก็บข้อมูลของหมายเลข IP ปลายทางที่เป็นผู้รับข้อมูล IP Datagram นี้

Options เป็นฟิลด์เก็บข้อมูลที่มีขนาดไม่แน่นอน ใช้สำหรับกำหนดค่าตัวแปรส่วนประกอบปลีกย่อย ซึ่งส่วนใหญ่ไม่มีการนำไปใช้งาน

Padding ทำหน้าที่เป็นส่วนข้อมูลเติมเต็มเพื่อให้ IP Header เต็มครบ 32 ไบต์ ซึ่งเป็นผลมาจาก Options ที่มีขนาดไม่แน่นอนนั่นเอง

2.5.6 โปรโตคอล TCP

โปรโตคอล TCP (Transmission Control Protocol) เป็นโปรโตคอลที่มีการรับส่งข้อมูลแบบ Stream – Oriented Protocol หมายความว่า การรับส่งข้อมูลจะไม่คำนึงถึงปริมาณข้อมูลที่จะส่งไป แต่จะแบ่งข้อมูลออกเป็นส่วนย่อย ๆ ก่อน แล้วจึงจะส่งไปยังปลายทางอย่างต่อเนื่องเป็นลำดับข้อมูล ในกรณีที่

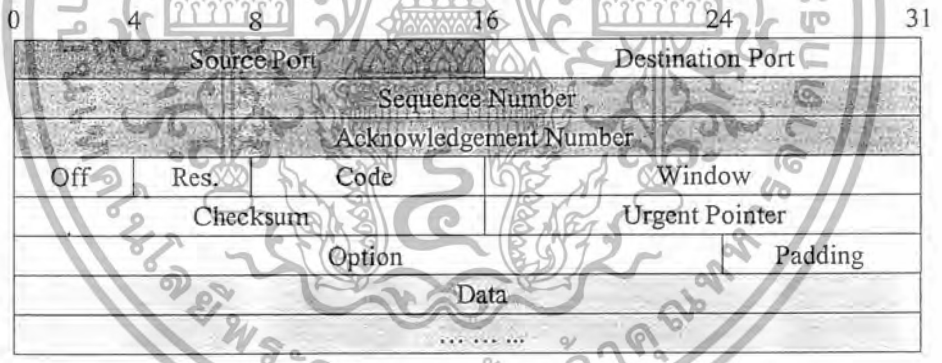
เอกสารนี้สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ข้อมูลส่วนใดส่วนหนึ่งสูญหายไป ก็จะส่งข้อมูลส่วนนั้นใหม่อีกครั้ง สำหรับปลายทางก็จะทำหน้าที่จัดเรียงส่วนของข้อมูล Datagram ใหม่ให้ต่อเนื่องกันและประกอบกับเป็นข้อมูลทั้งหมดได้ ซึ่งจะแยกข้อมูลที่ไม่ถูกต้องออก ดังนั้นแอพพลิเคชั่นหรือกระบวนการใดที่อาศัยการส่งผ่านข้อมูลด้วยโปรโตคอล TCP จะต้องใช้หน่วยความจำและขนาดของช่องสัญญาณมากกว่า UDP

การติดต่อระหว่างกันจะต้องเป็นแบบ Connection – Oriented คือ ต้องมีการสร้างการติดต่อกันเป็น Session ทั้ง 2 ด้านเสียก่อน แล้วจึงจะรับส่งข้อมูลไปได้พร้อมกัน (Full Duplex) เหมือนกับการใช้โทรศัพท์ติดต่อกัน ซึ่งในระหว่างการติดต่อกันนั้น แม้ว่าฝ่ายใดฝ่ายหนึ่งหรือทั้งสองฝ่ายจะเงียบไป คือ ไม่พูดอะไรเป็นเวลานาน ๆ แต่การเชื่อมโยงระหว่างทั้งสองด้านยังคงมีอยู่ ไม่ขาดไปจนกว่าฝ่ายใดฝ่ายหนึ่งจะวางสายเช่นเดียวกับการติดต่อกันด้วยกลไกโปรโตคอล TCP เมื่อแอพพลิเคชั่นต้องการส่งผ่านข้อมูลจะใช้โปรโตคอลที่เหมาะสมในชั้น Process Layer ติดต่อไปและมีการสร้างช่องส่งข้อมูลผ่าน Port ที่กำหนดขึ้น เพื่อส่งผ่านข้อมูลไปยังโปรโตคอล TCP

ในระหว่างการรับส่งข้อมูลนี้ ตัวโปรโตคอล TCP จะทำการเพิ่มขบวนการสอบทานข้อมูล (Acknowledgement) เพื่อให้ข้อมูลมีความถูกต้องไม่ผิดพลาดไปจากเดิม โดยการส่งสัญญาณสอบทานข้อมูลและส่งข้อมูลให้ใหม่อีกครั้งถ้าปลายทางไม่ได้รับหรือเกิดความผิดพลาดขึ้น

ความน่าเชื่อถือของการส่งผ่านข้อมูลโดยโปรโตคอล TCP จะมีมากกว่า แต่ก็ต้องอาศัยทรัพยากรของระบบมากกว่าในการทำงานเช่นกัน



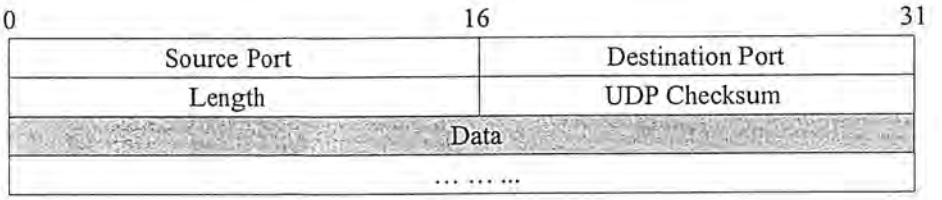
รูปที่ 2.31 รูปแบบของ TCP Packet ซึ่งมีฟิลด์ Acknowledgement Number และข้อมูล Checksum

2.5.7 โปรโตคอล UDP

ใน Host-to-Host Layer นอกจากจะมีโปรโตคอล TCP ทำงานแล้ว ก็ยังมีโปรโตคอล UDP (User Datagram Protocol) ที่มีคุณสมบัติแตกต่างกันอยู่แล้ว ในการรับส่งข้อมูลผ่านโปรโตคอล UDP จะเป็นแบบที่ทั้งสองด้านไม่จำเป็นต้องอาศัยการสร้างช่องทางเชื่อมต่อกัน (Connectionless) ระหว่างเครื่อง Server กับเครื่องที่ขอใช้บริการ (Client) โดยไม่ต้องแจ้งให้ฝ่ายรับข้อมูลเตรียมรับข้อมูลเหมือนโปรโตคอล TCP และไม่มีการตรวจสอบความถูกต้องครบถ้วนในการรับส่งข้อมูลนั้น ๆ ด้วย เนื่องจากโปรโตคอล UDP ไม่มีสัญญาณสอบทานข้อมูลในการส่งข้อมูลแต่ละครั้ง และไม่มีการส่งข้อมูลใหม่อีกใน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

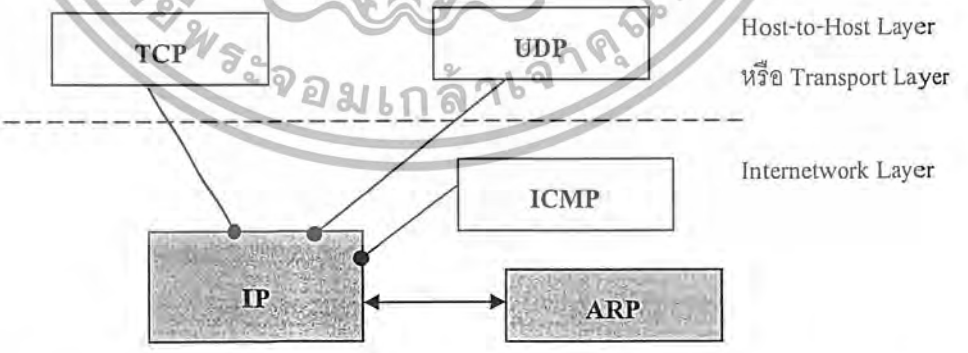
กรณีที่เกิดความผิดพลาดของการส่งข้อมูล เมื่อเป็นเช่นนี้แอพพลิเคชั่นหรือกระบวนใดที่ต้องอาศัยโปรโตคอล UDP ในการส่งผ่านข้อมูลก็อาจจะต้องสร้างกระบวนการตรวจสอบข้อมูลขึ้นมาเอง



รูปที่ 2.32 รูปแบบของ UDP Packet จะมีฟิลด์ข้อมูลส่วน Header น้อยมากและไม่มีส่วนตรวจสอบข้อมูล

2.5.8 โปรโตคอล IP

โปรโตคอล IP ทำหน้าที่ให้บริการส่งผ่านข้อมูลที่มาจาก Host-to-Host Layer เพื่อส่งข้ามไปยังเครือข่ายใด ๆ ได้อย่างถูกต้อง แม้ว่าจะมีเครือข่ายเชื่อมต่อกันอยู่ในอินเตอร์เน็ตเป็นล้าน ๆ เครือข่ายก็ตาม เนื่องจากโปรโตคอล IP มีข้อมูลตำแหน่ง IP ปลายทางที่จะส่งข้อมูลไปให้ โดยทำงานร่วมกับอุปกรณ์ Router เพื่อส่งข้อมูลข้ามเครือข่ายออกไป ตัวโปรโตคอล IP ทำงานแบบ Packet Switching คือ มีการส่งข้อมูลผ่านสวิตช์ไปยังปลายทาง โดยข้อมูลจะเดินทางไปยังเครือข่ายต่าง ๆ ผ่านสวิตช์นี้ไปเรื่อย ๆ จนกว่าจะถึงปลายทาง ตัววงจรผ่านหรือสวิตช์นี้อาจจะเป็น Gateway หรือ Router ในระบบเครือข่ายก็ได้ ซึ่งในข้อมูลของโปรโตคอล IP จะมีข้อมูลของหมายเลข IP ปลายทางที่จะส่งข้อมูลไปและเมื่อถึงเครือข่ายปลายทางแล้วจะมีกลไกแปลหมายเลข IP ให้เป็นหมายเลขฮาร์ดแวร์ประจำเครื่องที่ถูกต้องอีกทีหนึ่งด้วยโปรโตคอล ARP ตามรูปที่ 2.33 ที่จะแสดงการติดต่อกันระหว่างโปรโตคอลในชั้นของ Host-to-Host Layer และ Internetwork Layer

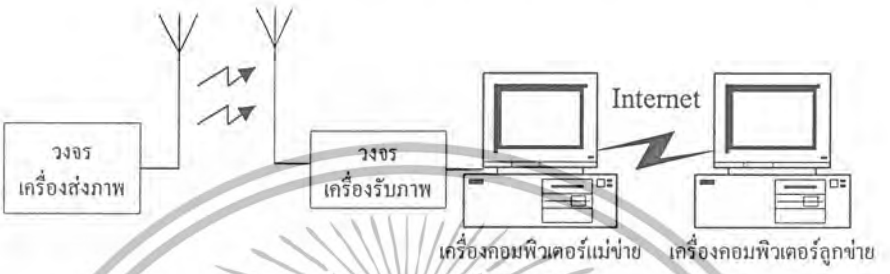


รูปที่ 2.33 การติดต่อกันระหว่างโปรโตคอลในชั้นของ Host-to-Host Layer และ Internetwork Layer

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 3
การคำนวณและการสร้าง

ในโครงงานนี้สามารถแบ่งออกเป็นส่วนหลักๆได้ 2 ส่วนด้วยกัน ได้แก่ ส่วนของภาครับ-ส่งสัญญาณภาพ กับภาครับ-ส่งสัญญาณควบคุม ซึ่งแสดงบล็อกไดอะแกรมดังรูปที่ 3.1 และ 3.2 ตามลำดับ



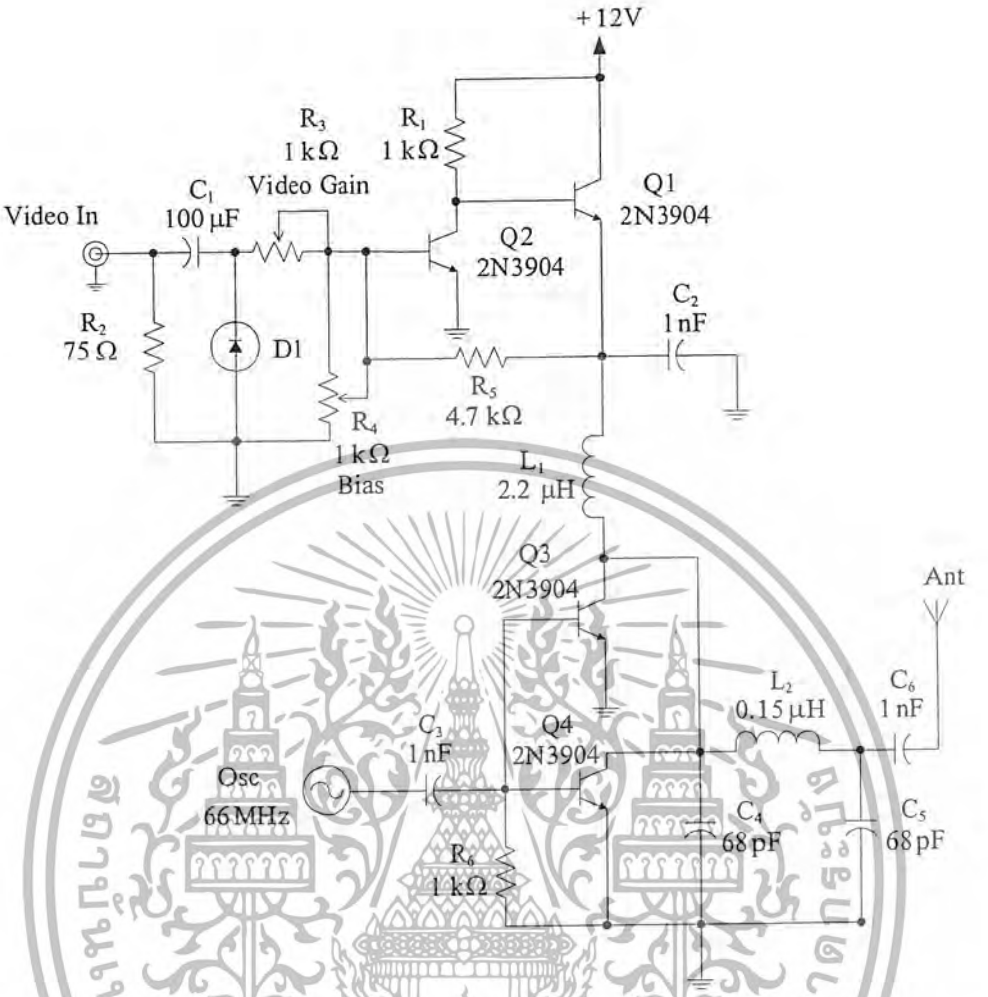
รูปที่ 3.1 Block Diagram ของภาครับ – ส่งสัญญาณภาพ

รูปที่ 3.2 Block Diagram ของภาครับ - ส่งสัญญาณควบคุม

3.1 ภาครับ - ส่งสัญญาณภาพ

ในส่วนนี้สามารถแบ่งการทำงานออกเป็น 2 ภาคหลักๆ คือ ภาคส่งสัญญาณ และภาครับสัญญาณ ซึ่งในภาครับสัญญาณนั้นจะใช้การ์ด TV Tuner ซึ่งเป็นอุปกรณ์ที่ติดตั้งเสริมให้กับเครื่องคอมพิวเตอร์ และในส่วนของภาคส่งสัญญาณภาพไร้สายนั้น จะใช้วงจรที่แสดงดังรูปที่ 3.3

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.3 รูปวงจรเครื่องส่งภาพไร้สาย

ในส่วนของภาคส่งสัญญาณภาพนั้นจะใช้วงจรซึ่งแสดงดังรูปที่ 3.3 โดยมีหลักการทำงาน คือ เมื่อสัญญาณภาพเข้ามาทางขั้วอินพุต จะผ่านเข้าสู่ทรานซิสเตอร์ Q2 ซึ่งเป็นวงจรขยายสัญญาณแบบ Common Emitter โดยมีตัวต้านทานปรับค่า R_3 เป็นตัวปรับอัตราการขยายสัญญาณ (เป็นตัวปรับสีของภาพ) ส่วนตัวต้านทานปรับค่า R_4 จะเป็นตัวปรับแรงดันไบอัสให้แก่ Q2 (เป็นตัวปรับความสว่างของภาพ) หลังจากนั้นสัญญาณที่ได้จะไปผ่านทรานซิสเตอร์ Q1 ซึ่งทำหน้าที่เป็นบัฟเฟอร์

จากนั้นสัญญาณที่ได้จะถูกส่งผ่านขดลวด L_1 ซึ่งทำหน้าที่เป็น Radio Frequency Choke (RFC) ลงมาเพื่อทำการมอดูเลตกับสัญญาณคลื่นพาห์ ซึ่งสัญญาณคลื่นพาห์นั้นจะถูกขยายด้วยทรานซิสเตอร์ Q3 และ Q4 โดยจะมีการขยายเปลี่ยนแปลงไปตามสัญญาณภาพที่ผ่านการขยายมาแล้ว ซึ่งจะพบว่าเป็นการมอดูเลตแบบ Amplitude Modulation และสัญญาณที่ได้ก็จะไปผ่านวงจรแมตชิ่งอิมพีแดนซ์ แล้วจึงส่งต่อไปยังสายอากาศ ตามลำดับ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.2 ภาครับ - ส่งสัญญาณควบคุม

ในที่นี้จะกล่าวถึงเฉพาะส่วนควบคุมการเคลื่อนที่ของรถ และส่วนควบคุมการหมุนกล้อง โดยแต่ละส่วนจะควบคุมจากคีย์บอร์ดของคอมพิวเตอร์ ผ่านโปรแกรมในการติดต่อกับพอร์ต ส่งค่าให้กับไมโครคอนโทรลเลอร์ 89C2051 ในการเข้ารหัสส่งให้กับโมดูลตัวส่งซึ่งใช้ความถี่ 27 MHz ส่วนทางด้านรับก็จะทำการถอดรหัสจากโมดูลตัวรับ และส่งค่าเข้าสู่ไมโครคอนโทรลเลอร์ 89C2051 เช่นกัน โดยจะแยกเป็นการควบคุม DC Motor เพื่อบังคับทิศทางรถผ่านทาง IC Drive เบอร์ L293D และการควบคุมการหมุนของกล้องด้วย Servo Motor ในทั้งนี้เราสามารถควบคุมทิศทางของรถและการหมุนของกล้องได้ทั้งจากเครื่องคอมพิวเตอร์แม่ข่าย (Server) และเครื่องคอมพิวเตอร์ลูกข่าย (Client) ได้ สำหรับเครื่องคอมพิวเตอร์ลูกข่ายจะควบคุมโดยผ่านทางอินเตอร์เน็ต โดยใช้ Winsock control ของ Visual Basic6 เป็นตัวจัดการการเชื่อมต่อและส่งสัญญาณระหว่างเครื่องคอมพิวเตอร์ทั้งสองฝั่ง

3.2.1 โปรแกรมในการติดต่อกับเครื่องส่งสัญญาณความถี่วิทยุ

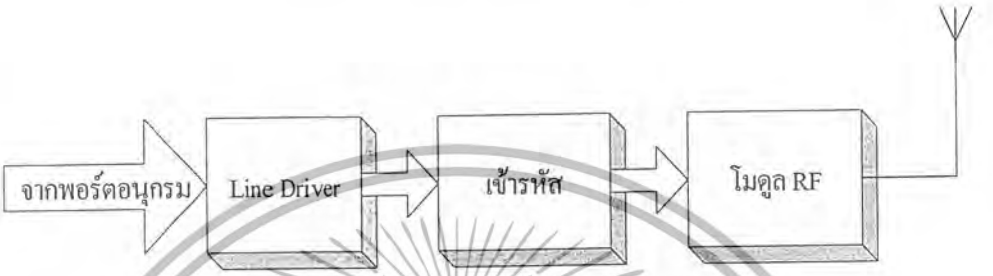
ใช้โปรแกรมที่เขียนโดยภาษา Visual Basic 6 โดยตัวโปรแกรมจะทำหน้าที่รับข้อมูลจากคีย์บอร์ดและส่งรหัส ASCII ของปุ่มที่เรากดออกไปยัง Serial Port ที่เราได้กำหนดไว้ (COM1) และยังมีหน้าต่างแสดงผลภาพที่ได้จาก TV Tuner Card อีกด้วยโดยใช้ คอนโทรลเพิ่มเติมชื่อว่า NDNM3.ocx ซึ่งแสดง Flow Chart ของโปรแกรมดังรูปที่ 3.4

รูปที่ 3.4 Flow Chart ของโปรแกรมติดต่อกับเครื่องส่ง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.2.2 เครื่องส่งสัญญาณควบคุม

ทำหน้าที่เปลี่ยนการเข้ารหัสจากพอร์ตของคอมพิวเตอร์มาเป็นรหัสที่ชุดส่งความถี่วิทยุรู้จัก โดยใช้ IC เบอร์ MAX232 ในการเปลี่ยนระดับ Logic จากมาตรฐาน RS232 เป็นระดับ logic แบบ TTL แล้วส่งต่อให้กับ Microcontroller เบอร์ 89C2051 ในการเข้ารหัส จากนั้นจึงส่งรหัสให้กับชุดโมดูลส่งสัญญาณ RF เพื่อทำการส่งออกอากาศต่อไป โดยจะประกอบไปด้วยส่วนย่อยๆ ดังนี้



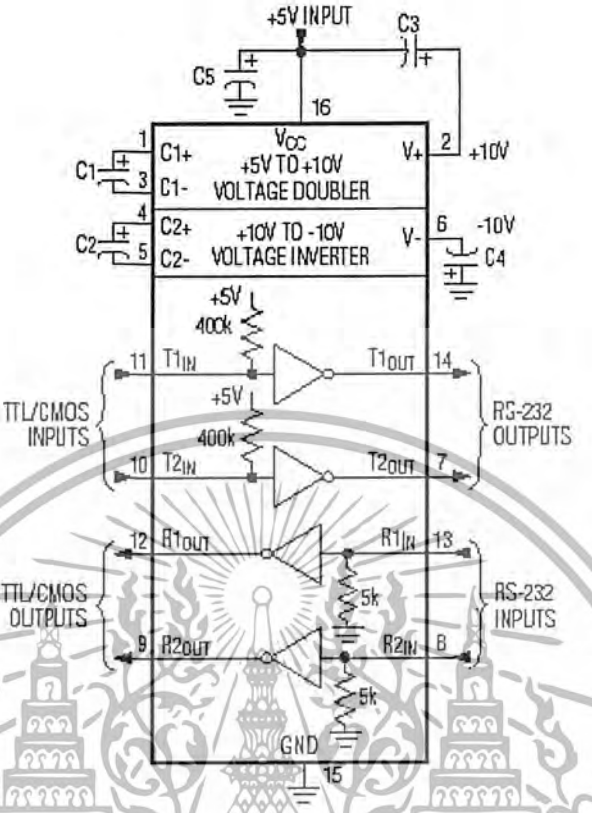
รูปที่ 3.5 Block Diagram ของเครื่องส่งสัญญาณควบคุม

1) ภาคแปลงระดับ Logic (Line Driver)

ในโครงงานนี้จะใช้ IC เบอร์ MAX232 ในการแปลงระดับลอจิกจากมาตรฐาน RS232 เป็นระดับ logic แบบ TTL ดังแสดงในรูปที่ 3.6

รูปที่ 3.6 การจัดเรียงขาของ IC MAX232

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.7 วงจรใช้งานของ IC MAX232

2) ภาคเข้ารหัสสัญญาณ

ใช้การเขียนโปรแกรมลงบนไมโครคอนโทรลเลอร์เบอร์ 89C2051 เพื่อแปลงรหัสจากรหัส ASCII เป็นรหัสที่โมดูลส่งสัญญาณวิทยุรู้จัก นอกจากนี้ยังใช้เป็นตัวส่งสัญญาณตอบกลับบอกไปยังคอมพิวเตอร์ด้วยว่าตอนนี้เครื่องส่งสัญญาณควบคุมนี้ต่ออยู่กับคอมพิวเตอร์ และทำการส่งสัญญาณอยู่ ซึ่งแสดง Flow Chart ดังรูปที่ 3.8

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

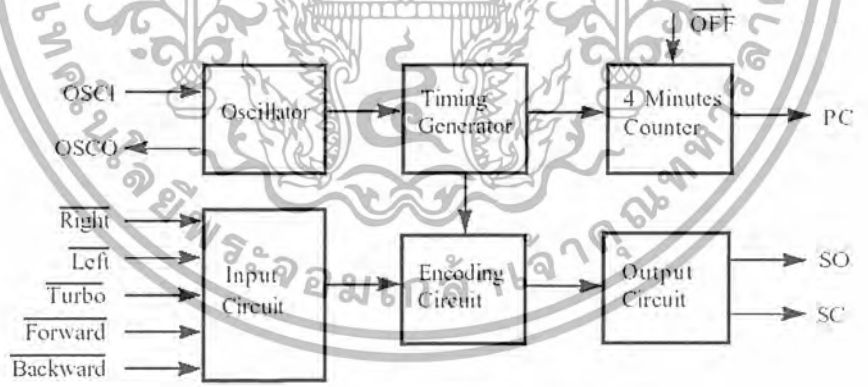
รูปที่ 3.8 Flow Chart ของโปรแกรมแปลงรหัสสัญญาณ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3) ภาคส่งสัญญาณความถี่วิทยุ

ใช้โมดูลตัวส่งที่ใช้ความถี่ 27MHz ในการส่งสัญญาณ โดยโมดูลตัวส่งนี้จะมีชุดกำเนิดความถี่แบบใช้คริสตัล และมีส่วนมอดูเลตสัญญาณ เพื่อที่จะส่งสัญญาณออกอากาศ จะใช้ IC เบอร์ TX-2 หรือ PT8A977P ดังรูปที่ 3.9 ซึ่งเป็น IC ที่ใช้ในงาน Radio Control โดยเฉพาะ ในการเข้ารหัสสัญญาณ logic เป็นสัญญาณรูปแบบต่างๆกันดังแสดงในรูปที่ 3.12 เพื่อที่จะนำไปมอดูเลตแบบ AM โดยจะใช้คู่กับตัวรับเบอร์ RX-2 หรือ PT8A978P

รูปที่ 3.9 การจัดขาของ IC PT8A977P

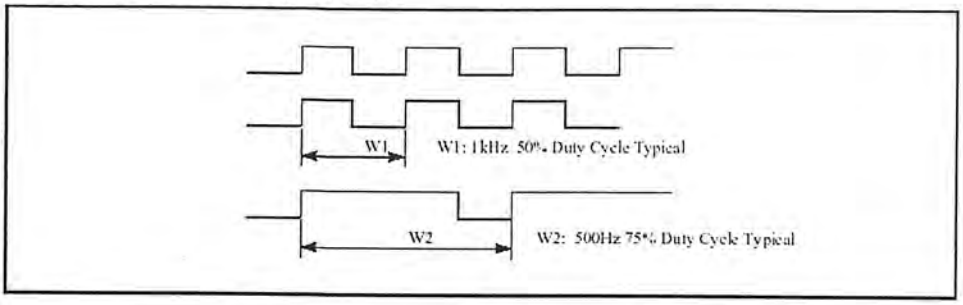


รูปที่ 3.10 Block Diagram ของ IC PT8A977P

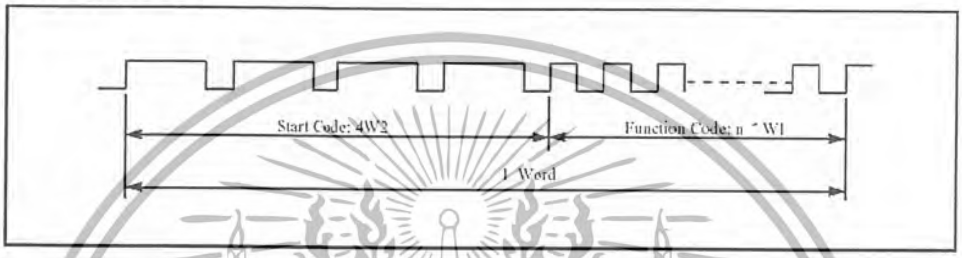
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Code Format

(W1 is used for function codes, W2 for start codes)



Word Format



รูปที่ 3.11 แสดงการเข้ารหัสสัญญาณของ IC PT8A977P

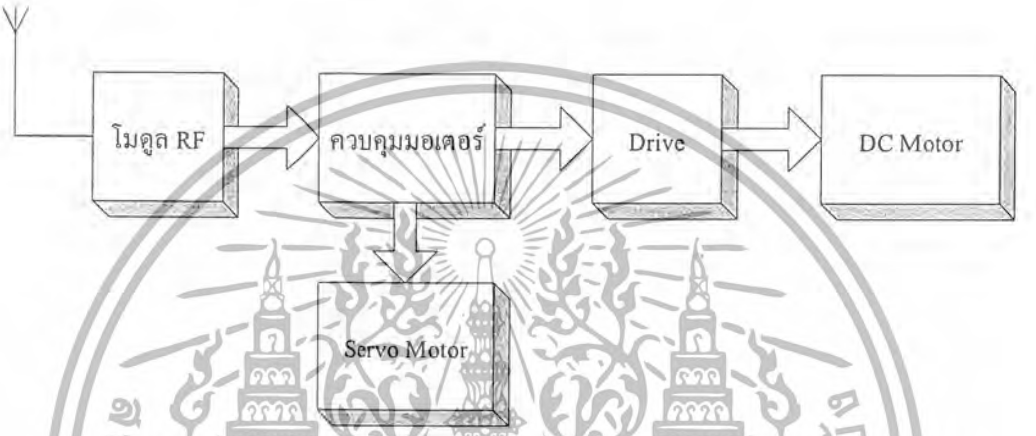
จำนวนของ Function Code W1	ผลการ Decoding
4	หยุด
10	เดินหน้า(พัลซ์)
16	เดินหน้า(ระดับสัญญาณสูง)
22	เทอร์โบ
28	เดินหน้า(ระดับสัญญาณสูง)+เลี้ยวซ้าย
34	เดินหน้า(ระดับสัญญาณสูง)+เลี้ยวขวา
40	ถอยหลัง
46	ถอยขวา
52	ถอยซ้าย
58	เลี้ยวซ้าย
64	เลี้ยวขวา

ตารางที่ 3.1 รหัสสัญญาณต่างๆ ในการเข้ารหัสของ IC PT8A977P

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.2.3 เครื่องรับสัญญาณควบคุม

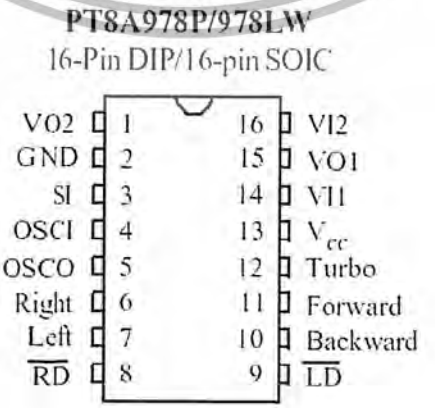
ตัวเครื่องรับสัญญาณควบคุมจะทำหน้าที่รับสัญญาณวิทยุจากเครื่องส่ง ด้วยโมดูลตัวรับ แล้วส่งสัญญาณให้กับไมโครคอนโทรลเลอร์เบอร์ 89C2051 ในการแปลงสัญญาณที่ได้รับเพื่อนำไปควบคุมมอเตอร์ ส่วนควบคุมมอเตอร์แบ่งเป็น 2 ส่วน คือส่วนควบคุมล้อ จะใช้ DC Motor และควบคุมการกลับทางหมุนโดย IC DRIVE เบอร์ L293D และในส่วนควบคุมกล้องจะใช้ Servo Motor เป็นตัวหมุน โดยจะเป็นไปตามโปรแกรมควบคุมที่ได้เขียนไว้ในไมโครคอนโทรลเลอร์ ซึ่งในเครื่องรับสัญญาณควบคุมจะประกอบไปด้วยส่วนย่อยๆดังรูป



รูปที่ 3.12 Block Diagram ของเครื่องรับสัญญาณควบคุม

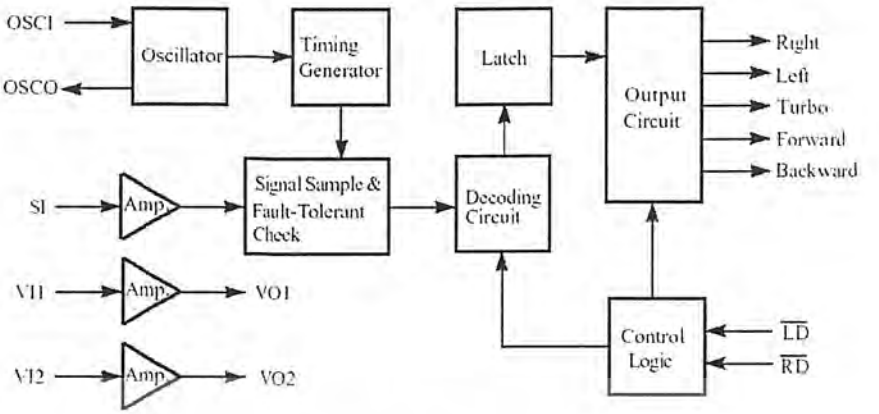
1) ภาครับสัญญาณวิทยุ

ส่วนนี้จะติดอยู่ที่รถ และทำหน้าที่รับสัญญาณวิทยุที่ถูกส่งมาจากโมดูลตัวส่ง แล้วจึงนำสัญญาณมาทำการถอดรหัสโดยไอซี RX-2 หรือ PT8A978P ดังแสดงในรูปที่ 3.13 แล้วจึงส่งสัญญาณให้กับไมโครคอนโทรลเลอร์ 89C2051 เพื่อทำการควบคุมมอเตอร์แต่ละตัว ทั้ง DC Motor และ Servo Motor



รูปที่ 3.13 การจัดขาของ IC PT8A978P

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.14 Block Diagram ของ IC PT8A978P

2) ภาคควบคุมมอเตอร์

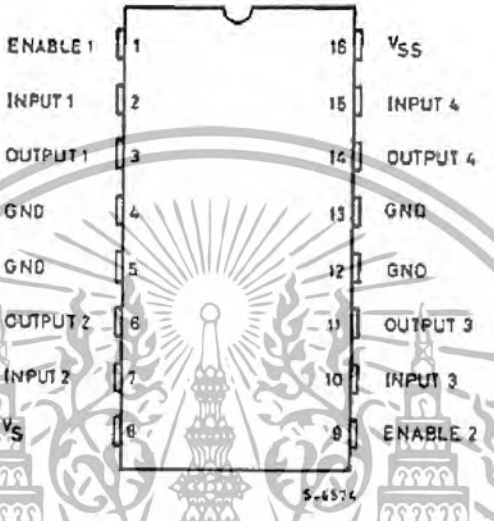
จะใช้ไมโครคอนโทรลเลอร์ เบอร์ 89C2051 ในการรับสัญญาณจากโมดูลตัวรับสัญญาณวิทยุ แล้วแปลงเป็นสัญญาณในการควบคุมมอเตอร์ โดยการควบคุม DC Motor จะผ่าน IC Drive เบอร์ L293D ส่วนการควบคุม Servo Motor จะใช้การเขียนโปรแกรมควบคุมในตัวไมโครคอนโทรลเลอร์ ซึ่งแสดง Flow Chart ของโปรแกรมได้ดังรูปที่ 3.15

รูปที่ 3.15 Flow Chart ของโปรแกรมควบคุมมอเตอร์ (89C2051)

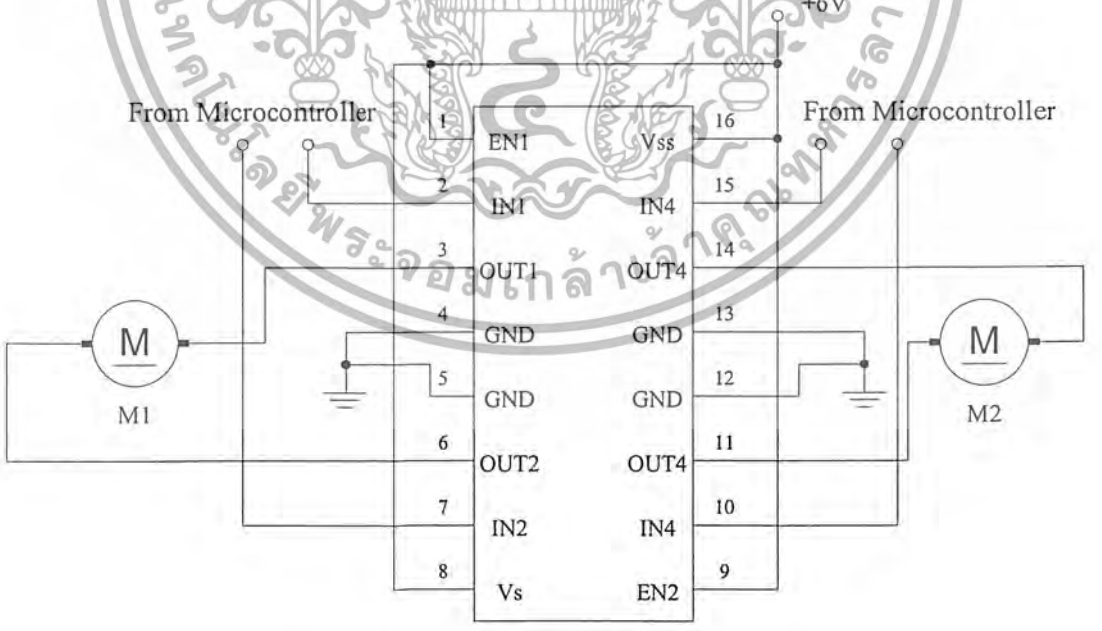
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3) ภาคขับมอเตอร์

ใช้ IC Drive เบอร์ L293D ในการขับมอเตอร์กระแสตรง ซึ่งไอซี 1 ตัว จะสามารถขับมอเตอร์ 2 ทางได้ 2 ตัว โดยใช้ IC Buffer เบอร์ 74HCT245 เป็นตัวช่วยขับกระแสจากไมโครคอนโทรลเลอร์ ซึ่งแสดงลักษณะของ IC L293D และการต่อวงจรใช้งานได้ดังรูปที่ 3.16 และ 3.17 ตามลำดับ



รูปที่ 3.16 การจัดขาของ IC L293D



รูปที่ 3.17 การต่อวงจรใช้งานของ IC L293D

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.3 ส่วนโปรแกรมติดต่อผ่านอินเตอร์เน็ต

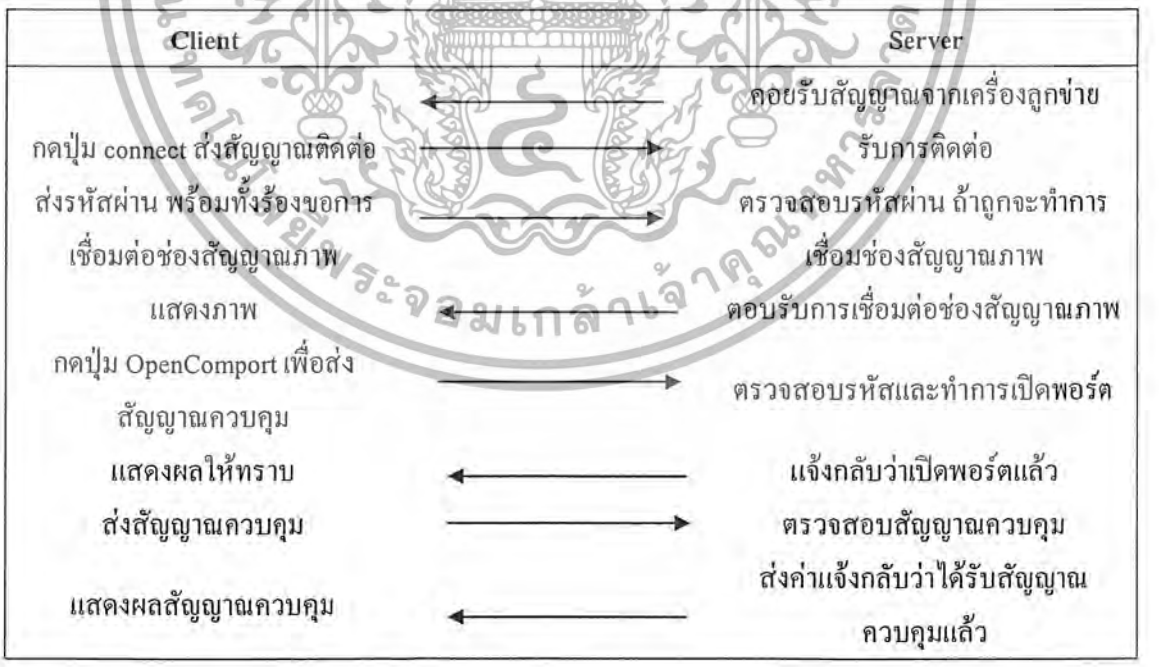
ใช้โปรแกรม Visual Basic 6 ในการเขียนโปรแกรมติดต่อระหว่างเครื่อง Sever และเครื่อง Client โดยใช้โปรโตคอล TCP/IP ในการรับ-ส่งข้อมูล สามารถแบ่งย่อยได้เป็น 2 ส่วน คือส่วนส่งข้อมูลภาพ และส่วนควบคุมผ่านอินเตอร์เน็ต

3.3.1 ส่วนส่งข้อมูลภาพ

ในส่วนส่งข้อมูลภาพนี้เราจะใช้คอนโทรลสำเร็จรูปมาช่วยในการส่งข้อมูล streaming ผ่านอินเตอร์เน็ต โดยเราเลือกใช้คอนโทรล NDNM3.ocx ซึ่งนอกจากจะสามารถแสดงข้อมูลภาพได้อย่างถูกต้องแล้ว ยังสามารถติดต่อโดยใช้ทรัพยากรของโปรแกรม Windows Netmeeting ได้อีกด้วย โดยเราจะต้องติดตั้งโปรแกรม Windows Netmeeting ไว้ก่อนแล้ว นอกจากนี้เรายังสามารถ Capture ข้อมูลภาพและจัดเก็บเป็นไฟล์ภาพชนิด BMP ได้โดยใช้คอนโทรล capture.ocx ซึ่งสามารถ Capture ข้อมูลภาพได้อย่างถูกต้อง และสามารถทำได้ทั้งฝั่ง Sever และฝั่ง Client

3.3.2 ส่วนควบคุมผ่านอินเตอร์เน็ต

ใช้ Winsock Control ในการส่งรหัส ASCII ที่ใช้เป็นรหัสในการควบคุมจากเครื่อง Client มายังเครื่อง Server โดยผู้ใช้จำเป็นต้องใส่รหัสผ่านในการเข้าใช้งานโปรแกรม จากนั้นโปรแกรมจะทำการเชื่อมต่อระหว่างเครื่องทั้งสอง จากนั้นการส่งสัญญาณควบคุมจะถูกกระทำโดยส่งจากเครื่อง Client กลับมายังเครื่อง Server และส่งต่อไปยังเครื่องส่งสัญญาณควบคุมต่อไป ซึ่งแสดงขั้นตอนการเชื่อมต่อระหว่างเครื่อง Server และเครื่อง Client ได้ดังรูปที่ 3.18

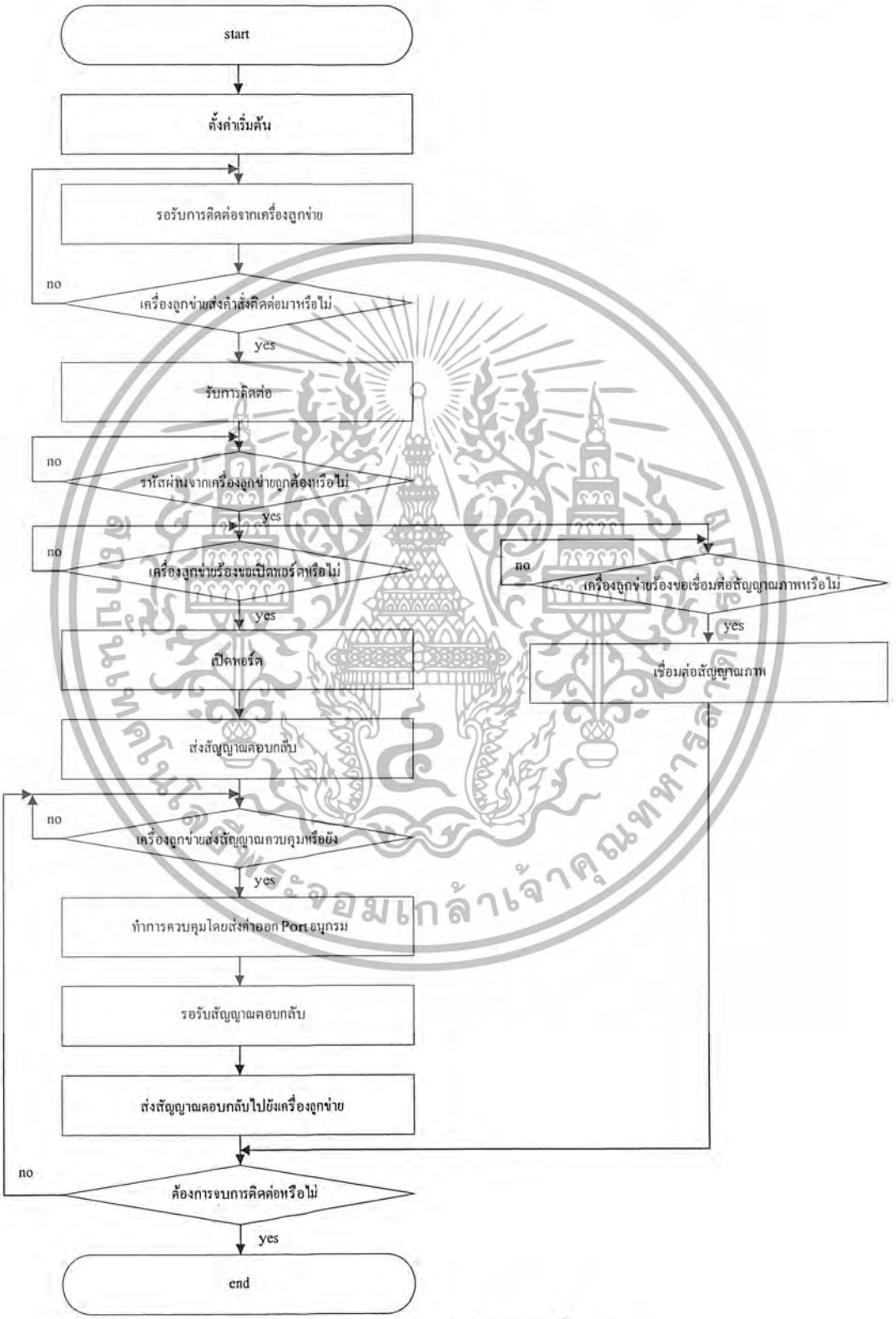


รูปที่ 3.18 ขั้นตอนการเชื่อมต่อระหว่างเครื่อง Server และเครื่อง Client

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

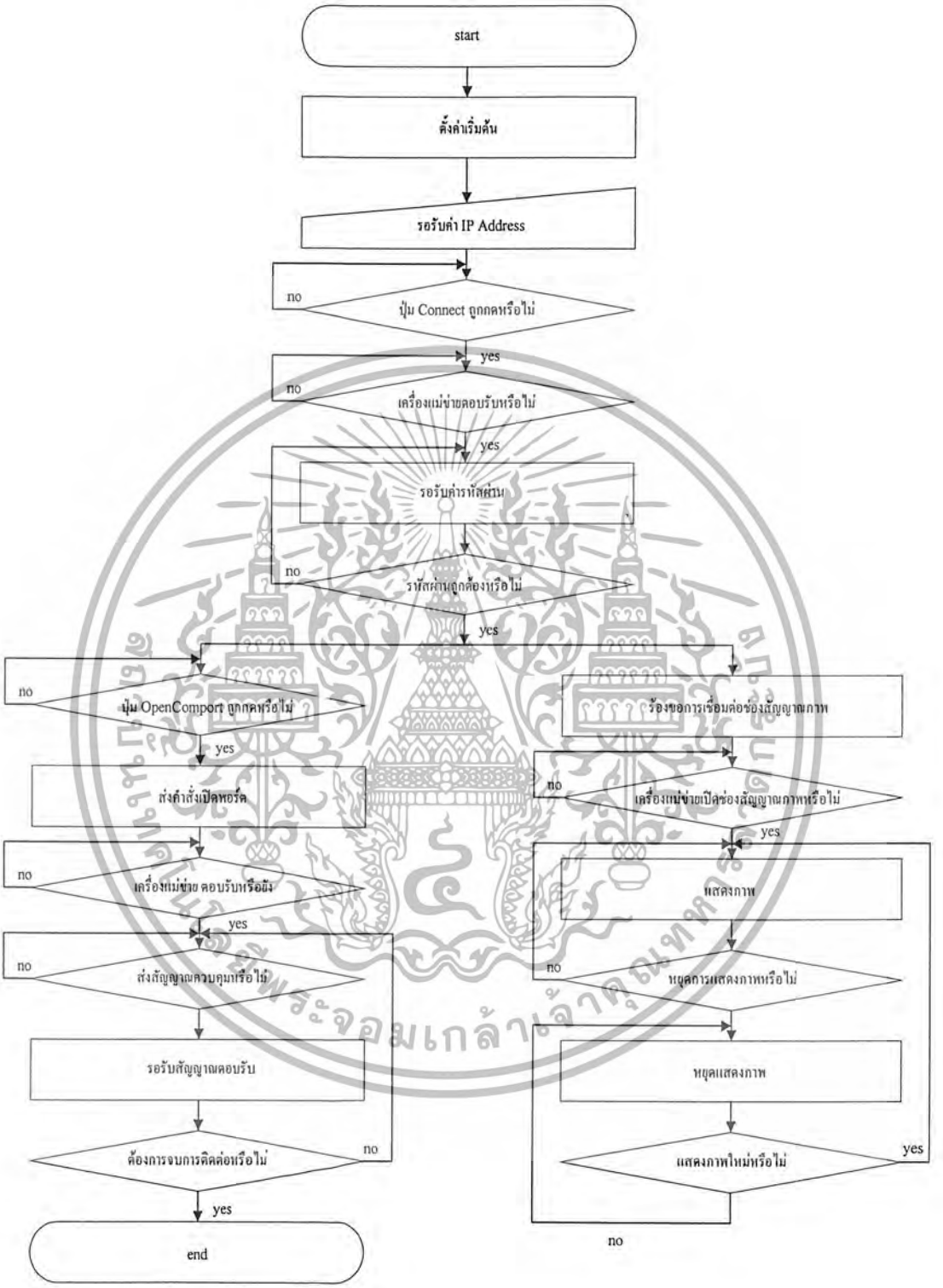
3.3.3 โปรแกรมในการติดต่อผ่านอินเตอร์เน็ต

เราจะแบ่งโปรแกรมออกเป็น 2 ส่วนใหญ่ๆ คือโปรแกรมทางฝั่งเครื่อง Server และโปรแกรมทางฝั่งเครื่อง Client โดยมีขั้นตอนการทำงานจะเป็นไปตามรูปที่ 3.19 และ 3.20 ตามลำดับ



รูปที่ 3.19 Flow Chart ของโปรแกรมฝั่ง Server

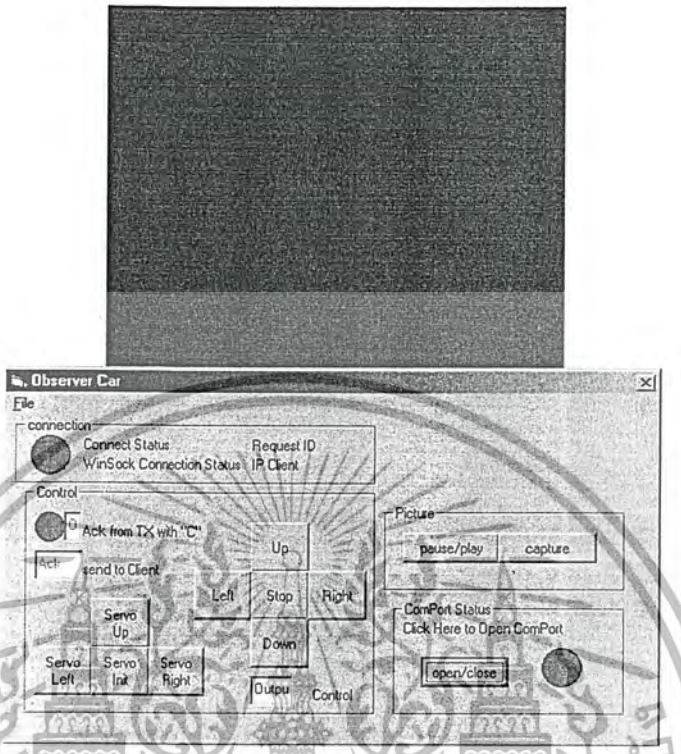
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



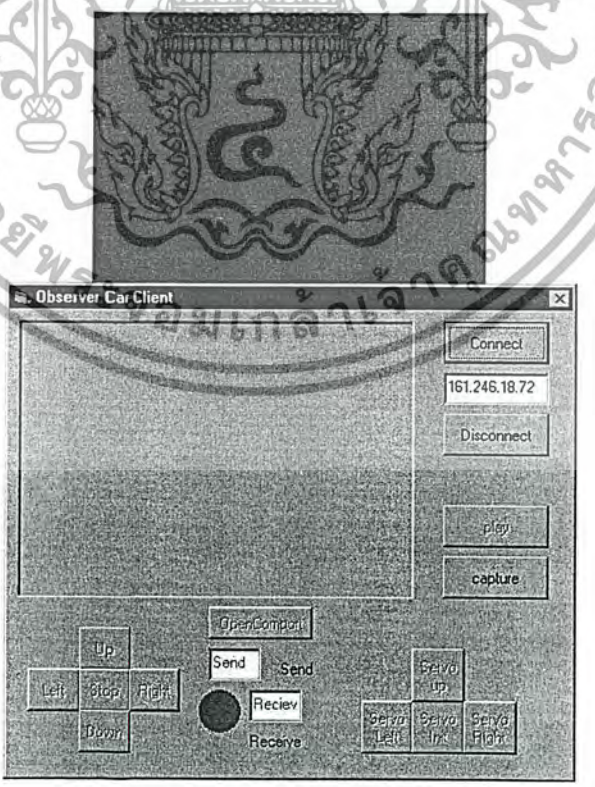
รูปที่ 3.20 Flow Chart ของโปรแกรมฝั่ง Client

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ลักษณะหน้าตาโปรแกรมฝั่ง Server และฝั่ง Client จะเป็นดังรูปที่ 3.21 และ 3.22 ตามลำดับ



รูปที่ 3.21 ลักษณะของหน้าต่างโปรแกรมฝั่ง Server

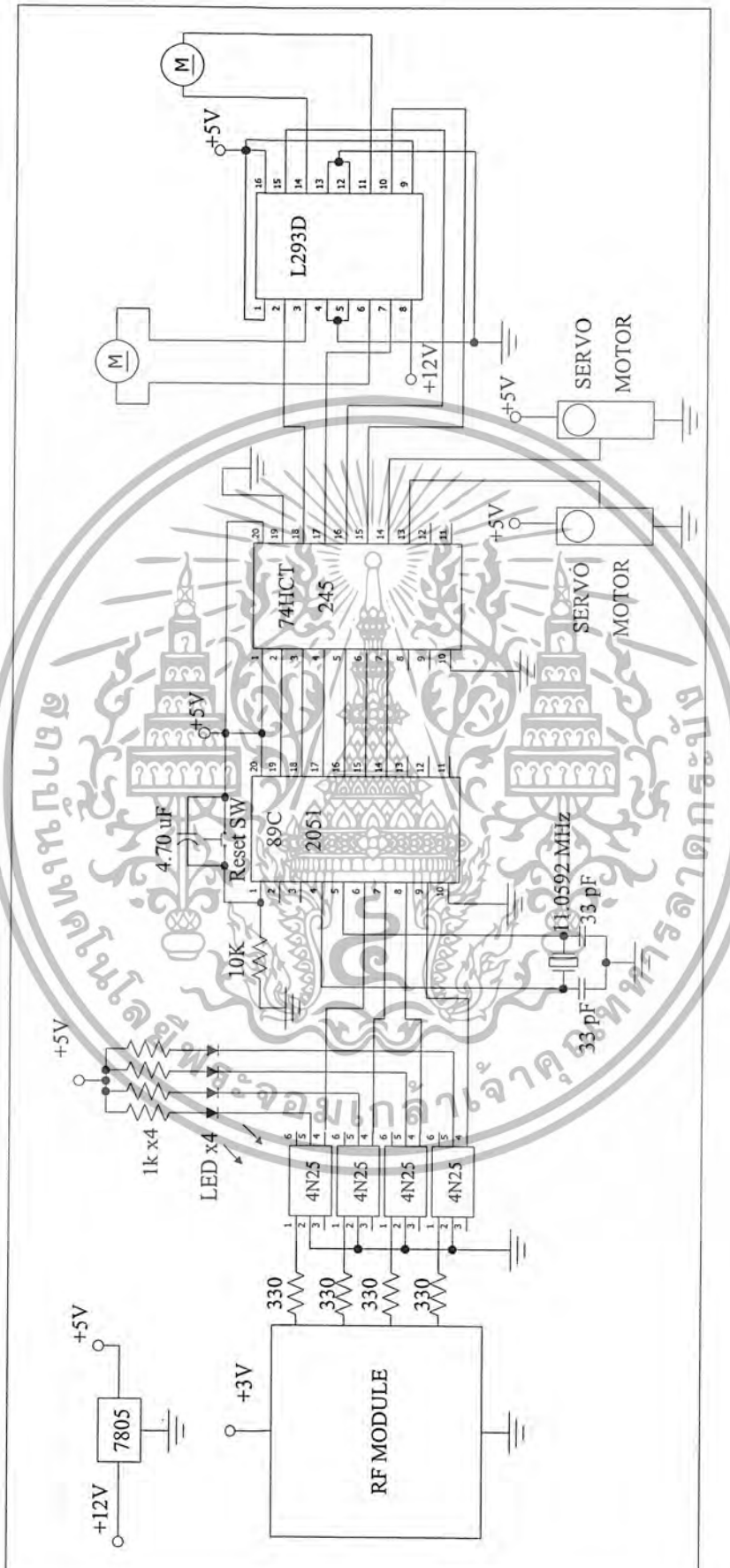


รูปที่ 3.22 ลักษณะของหน้าต่างโปรแกรมฝั่ง Client

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 3.23 วงจรรวมเครื่องส่งสัญญาณควบคุม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

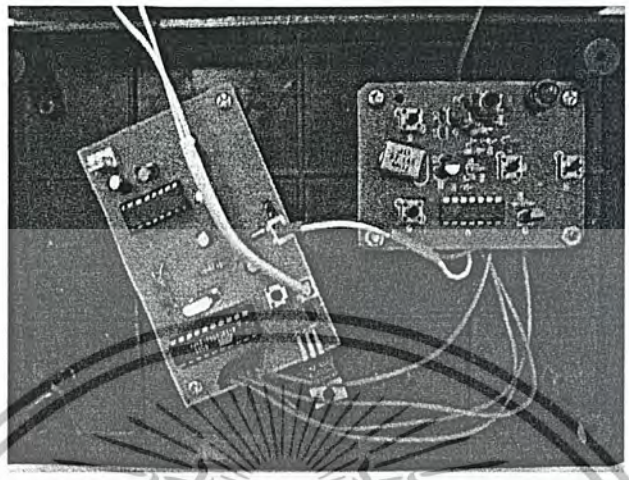


รูปที่ 3.24 วงจรรวมเครื่องรับสัญญาณควบคุม

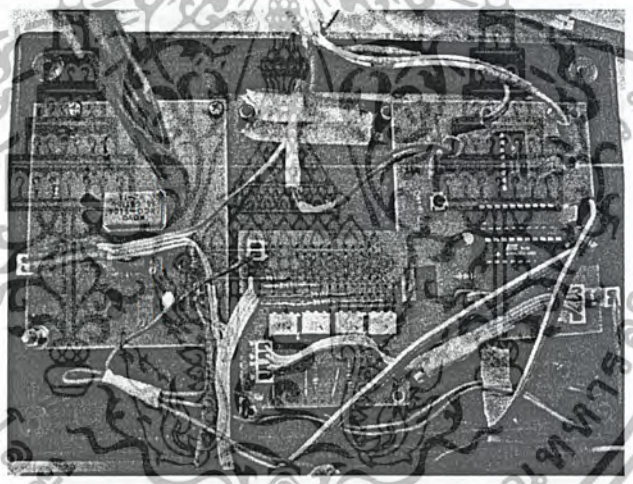
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.4 ลักษณะของชิ้นงาน

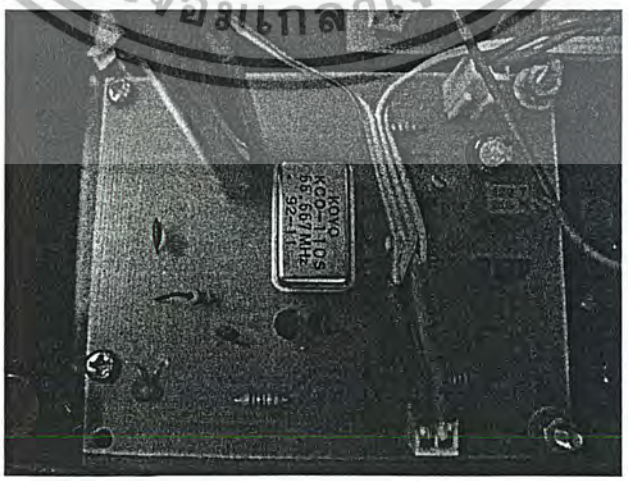
ชิ้นงานส่วนฮาร์ดแวร์แบ่งออกเป็น 3 ส่วน คือ ส่วนเครื่องส่งสัญญาณควบคุม, ส่วนรับสัญญาณควบคุม และส่วนชุดควบคุมรถและกล้อง ซึ่งมีลักษณะชิ้นงานดังนี้



รูปที่ 3.25 ลักษณะชิ้นงานภายในกล่องของเครื่องส่งสัญญาณควบคุม

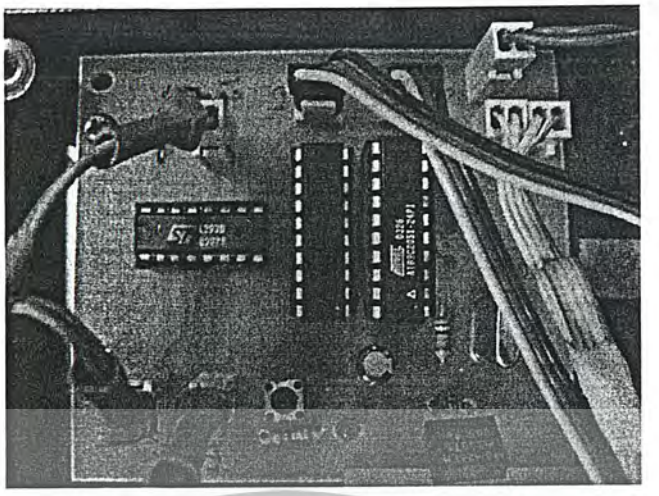


รูปที่ 3.26 ลักษณะชิ้นงานภายในกล่องของเครื่องรับสัญญาณควบคุม



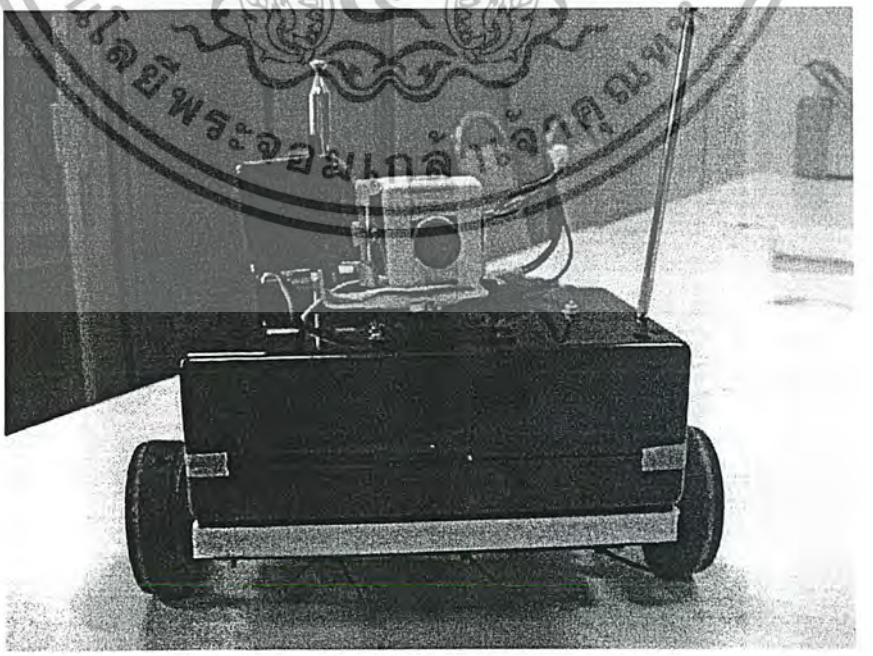
รูปที่ 3.27 ลักษณะชิ้นงานส่วนวงจรส่งสัญญาณภาพไร้สาย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.28 ลักษณะชิ้นงานส่วนวงจรไมโครคอนโทรลเลอร์ และวงจรขับมอเตอร์

รูปที่ 3.29 ลักษณะชิ้นงานส่วนชุดควบคุมกล้อง



รูปที่ 3.30 ลักษณะชิ้นงานรวม

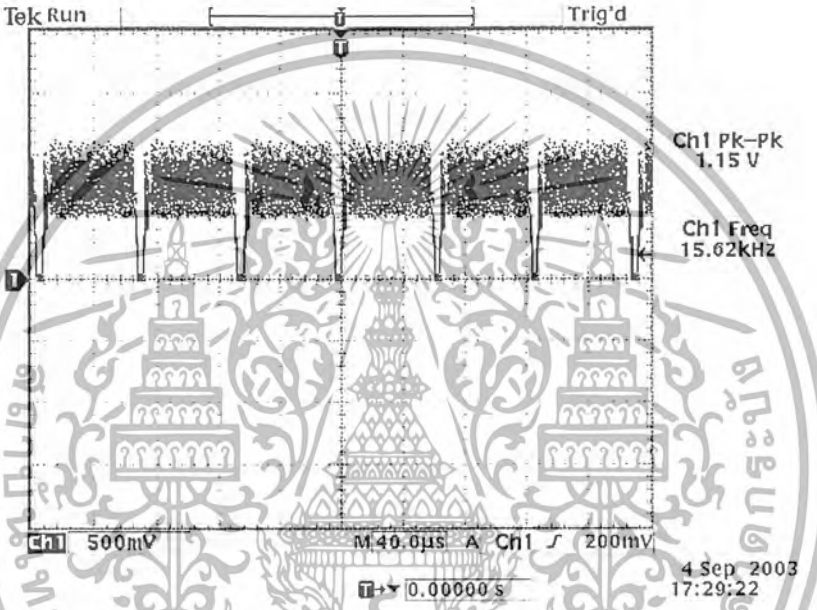
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 4
การทดลองและผลการทดลอง

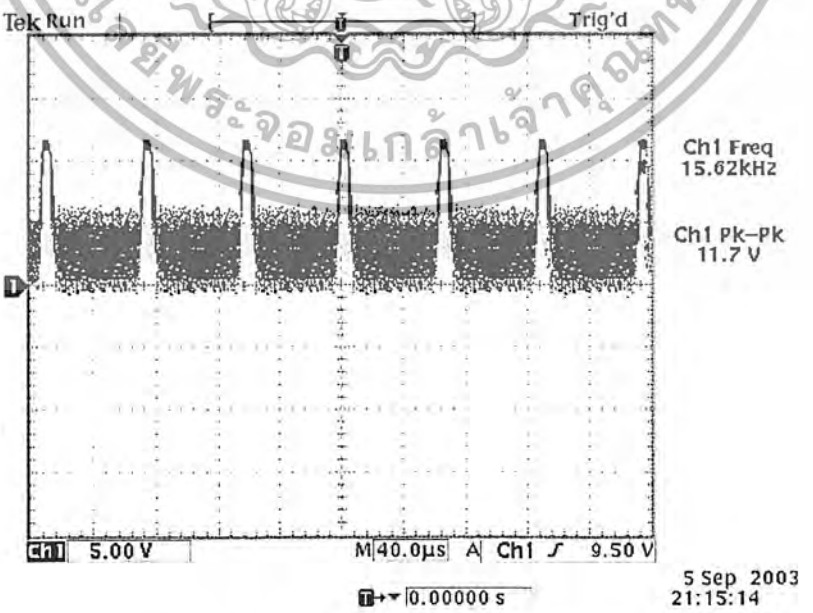
4.1 ภาควงจรเครื่องส่งภาพ

ทำการทดลองโดยการต่อวงจรเครื่องส่งภาพ ดังรูปที่ 3.3 แล้วจึงทดลองตามขั้นตอนต่างๆ

1. ทำการวัดสัญญาณภาพจากกล้อง จะได้สัญญาณดังรูปที่ 4.1 หลังจากนั้นจึงต่อเข้ากับขาอินพุตของวงจร ทำการวัดสัญญาณที่ขา Emitter ของทรานซิสเตอร์ Q1 จะได้สัญญาณที่ทำการขยายแล้ว ซึ่งจะเป็นสัญญาณที่จะนำไปมอดูเลต ดังรูปที่ 4.2 ตามลำดับ



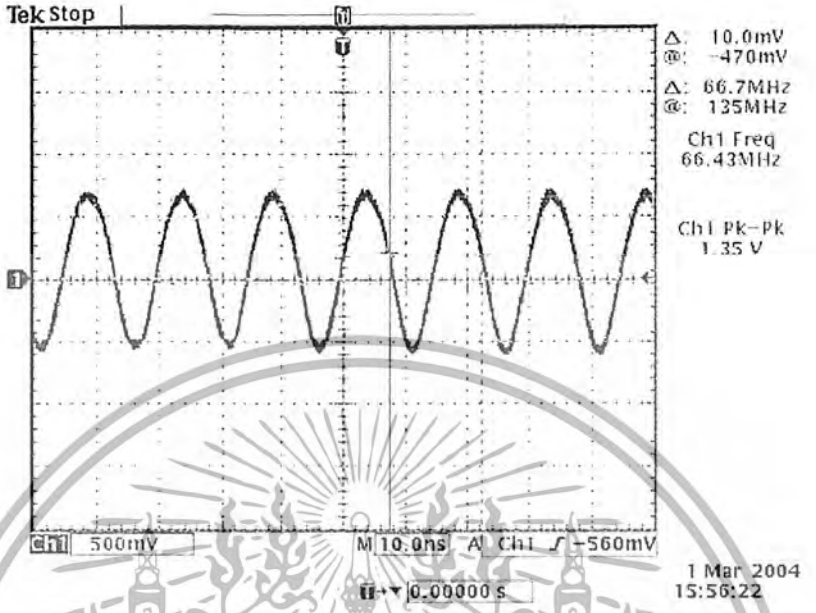
รูปที่ 4.1 สัญญาณภาพที่ได้จากกล้อง



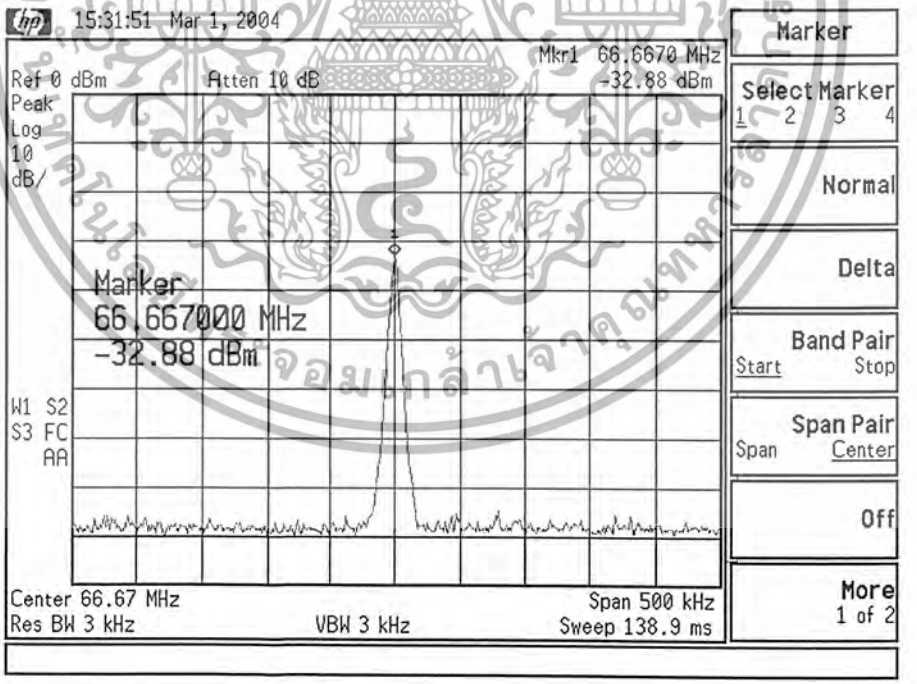
รูปที่ 4.2 สัญญาณภาพที่ผ่านการขยายแล้ว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2. วัดรูปสัญญาณที่ได้จาก Oscillator 66.667 MHz จะได้สัญญาณดังรูปที่ 4.3 และทำการวัดสเปกตรัมของสัญญาณจะได้ดังรูปที่ 4.4



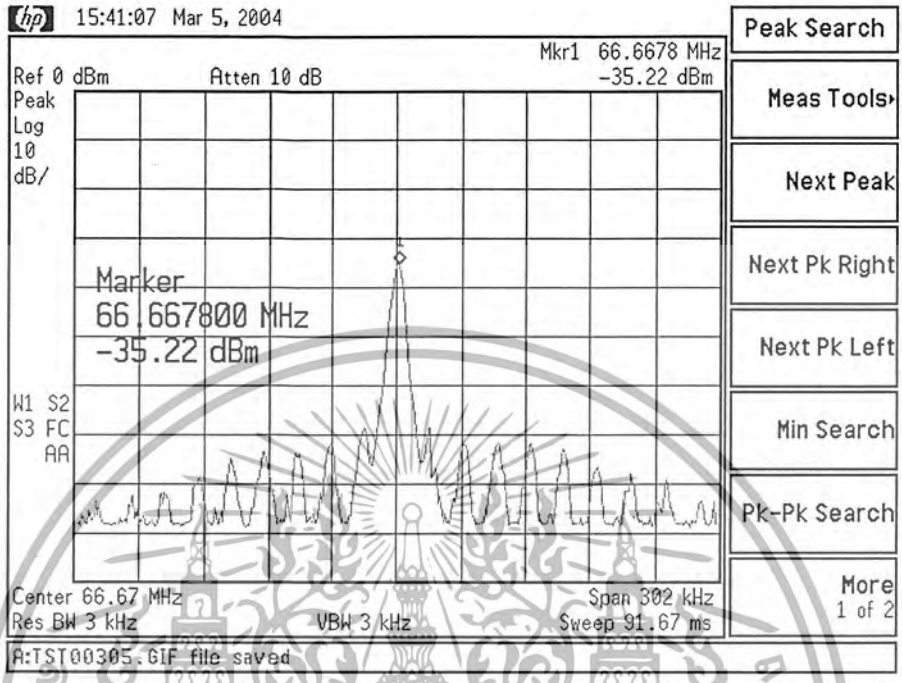
รูปที่ 4.3 สัญญาณคลื่นพาห์ 66.667 MHz



รูปที่ 4.4 สเปกตรัมของสัญญาณคลื่นพาห์ 66.667 MHz

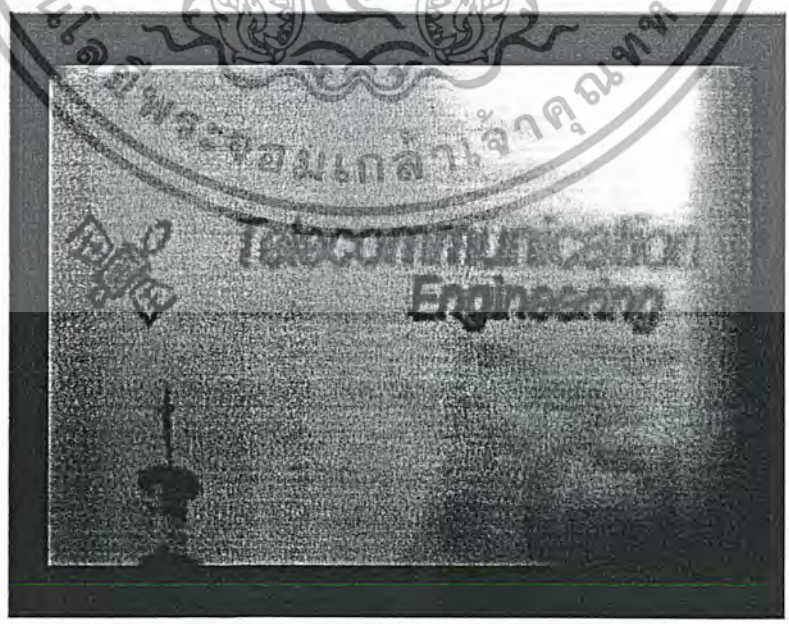
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3. ต่อสัญญาณภาพจากกล้องเข้าที่อินพุตของวงจรในรูปที่ 3.3 แล้วจึงทำการวัดสเปกตรัมของสัญญาณเอาท์พุตที่ทำการมอดูเลตแล้ว (ที่สายอากาศ) ได้ดังรูปที่ 4.5



รูปที่ 4.5 สเปกตรัมของสัญญาณ AM เมื่ออินพุตเป็นสัญญาณภาพจากกล้อง

4. เปิดโปรแกรมที่คอมพิวเตอร์ที่มีการติดตั้งการ์ดทีวีอยู่ เพื่อทำการรับสัญญาณภาพ ได้ภาพดังรูปที่ 4.6

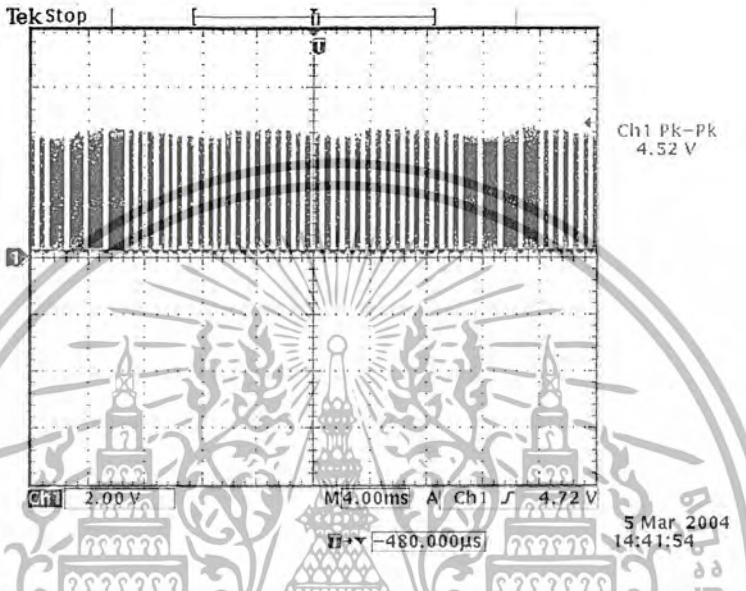


รูปที่ 4.6 ภาพที่รับได้ที่เครื่องคอมพิวเตอร์ที่ระยะ 5 เมตร

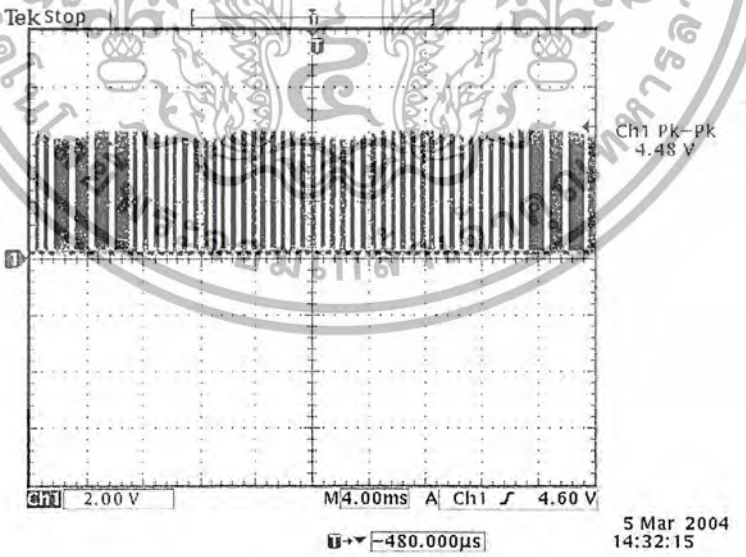
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.2 ภาคส่งสัญญาณควบคุม

เนื่องจากในการควบคุมทิศทางของรถและกล้องนั้น มีการนำโมดูลตัวส่งและตัวรับของรถกระป๋องมาใช้งาน โดยมีการต่อวงจรเพื่อทำการอินเตอร์เฟสกับเครื่องคอมพิวเตอร์ดังรูปที่ 3.23 เพื่อให้สามารถควบคุมจากปุ่มต่างๆที่เครื่องคอมพิวเตอร์นั้นๆได้ ซึ่งส่งผลให้เมื่อกดปุ่มต่างๆแล้ว ที่โมดูลตัวส่งของรถกระป๋องมีการเข้ารหัสสัญญาณและมอดูเลตสัญญาณออกมาต่างๆกันดังรูป

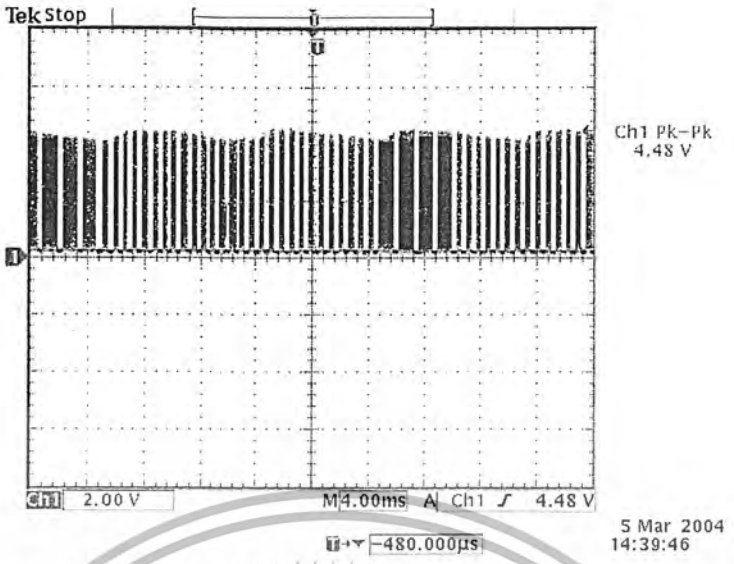


รูปที่ 4.7 สัญญาณจากโมดูลตัวส่ง เมื่อทำการกดปุ่มหมายเลข1

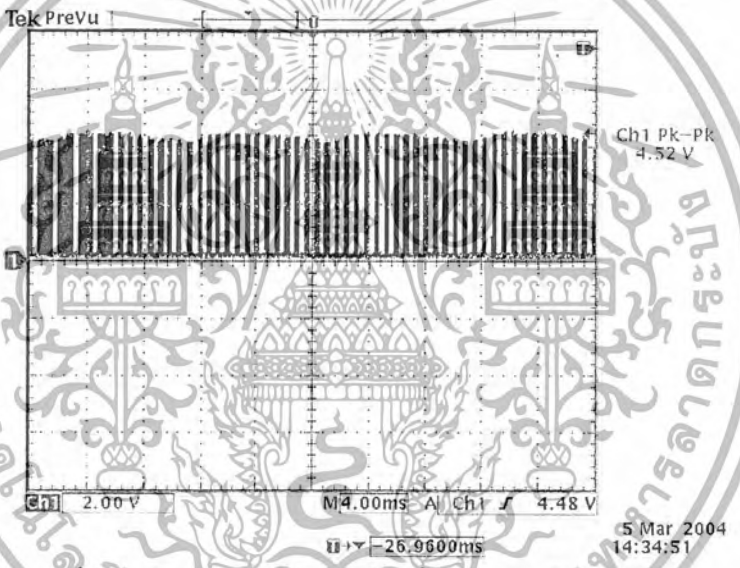


รูปที่ 4.8 สัญญาณจากโมดูลตัวส่ง เมื่อทำการกดปุ่มหมายเลข 2

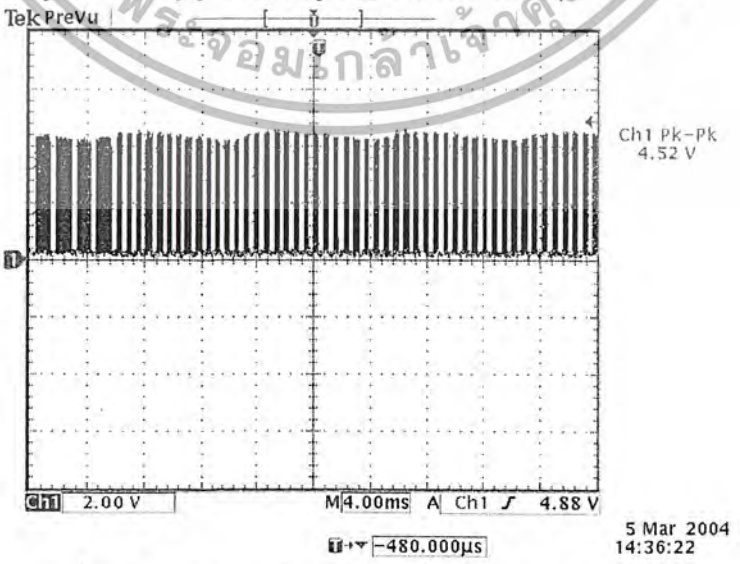
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.9 สัญญาณจากโมดูลตัวส่ง เมื่อทำการกดปุ่มหมายเลข 3

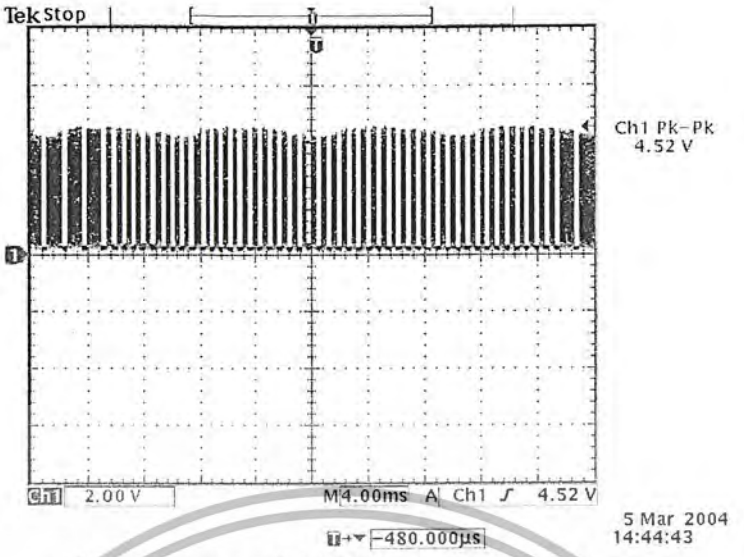


รูปที่ 4.10 สัญญาณจากโมดูลตัวส่ง เมื่อทำการกดปุ่มหมายเลข 4

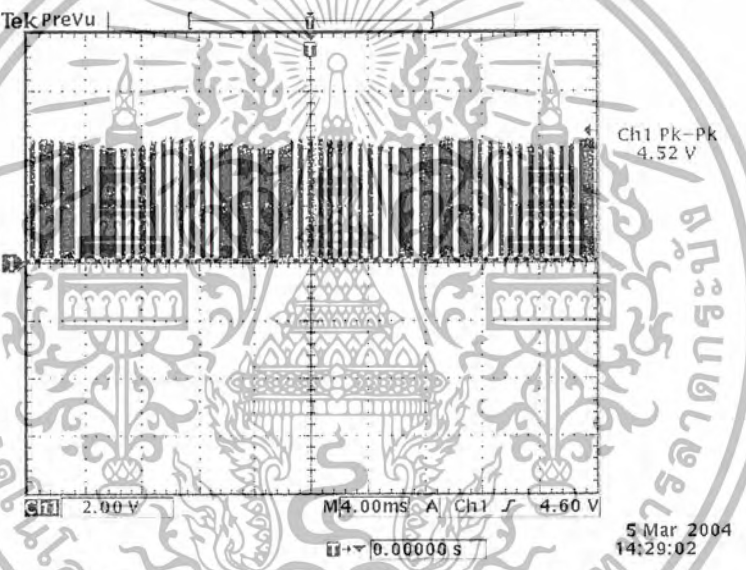


รูปที่ 4.11 สัญญาณจากโมดูลตัวส่ง เมื่อทำการกดปุ่มหมายเลข 6

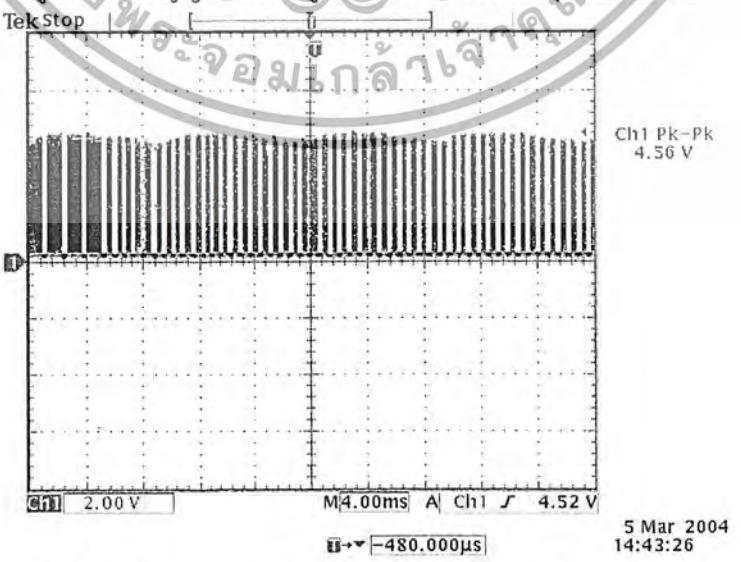
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.12 สัญญาณจากโมดูลตัวส่ง เมื่อทำการกดปุ่มหมายเลข 7

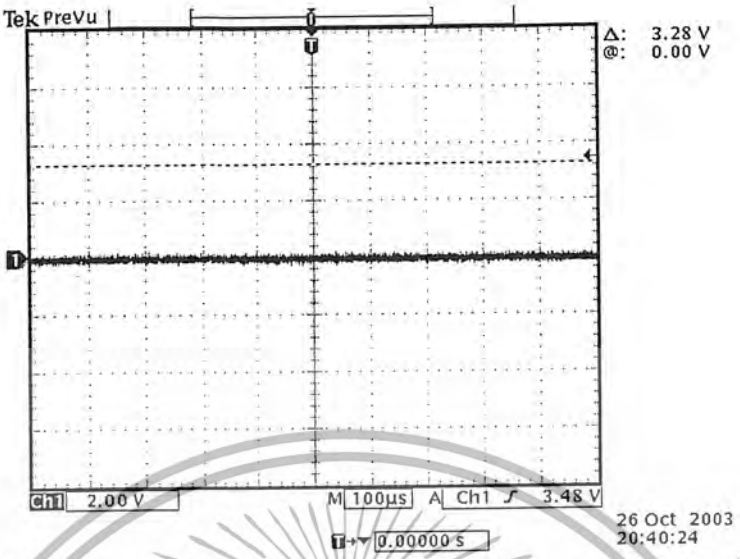


รูปที่ 4.13 สัญญาณจากโมดูลตัวส่ง เมื่อทำการกดปุ่มหมายเลข 8



รูปที่ 4.14 สัญญาณจากโมดูลตัวส่ง เมื่อทำการกดปุ่มหมายเลข 9

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.15 สัญญาณจากโมดูลตัวส่ง เมื่อทำการกดปุ่มหมายเลข 5 หรือ ปุ่มอื่นๆ

4.3 ภาครับสัญญาณควบคุม

ทำการทดลองโดยต่อ โมดูลตัวรับเข้ากับวงจรดังรูปที่ 3.24 สังเกตและบันทึกการทำงานของรถและกล้องเมื่อทำการกดปุ่มต่างๆกันได้ดังตารางที่ 4.1

ปุ่ม	ผล
หมายเลข 8	รถเดินหน้า
หมายเลข 4	รถเลี้ยวซ้าย
หมายเลข 6	รถเลี้ยวขวา
หมายเลข 2	รถถอยหลัง
หมายเลข 7	กล้องอยู่ตรงกลาง
หมายเลข 9	กล้องหันซ้าย
หมายเลข 1	กล้องเงยขึ้น
หมายเลข 3	กล้องหันขวา
หมายเลข 5 และปุ่มอื่นๆ	รถหยุด

ตารางที่ 4.1 การทำงานของรถและมอเตอร์หมุนกล้องเมื่อกดปุ่มจากคีย์บอร์ด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.4 ภาคติดต่อกับเครือข่ายอินเตอร์เน็ต

ทำการทดลองโดยการติดตั้งรถและกล้องที่ฝั่ง Server และเปิดโปรแกรมควบคุมที่เครื่อง Server เพื่อรอรับการติดต่อจากเครื่อง Client และทำการเปิดโปรแกรมควบคุมที่เครื่อง Client เพื่อเตรียมที่จะทำการเชื่อมต่อกับฝั่ง Server โดยทั้งสองเครื่องจะต้องทำการติดตั้งโปรแกรม Windows Netmeeting, Active X Control NDNM3.ocx และ Capture.ocx ก่อน โดยมีขั้นตอนการใช้งานดังนี้

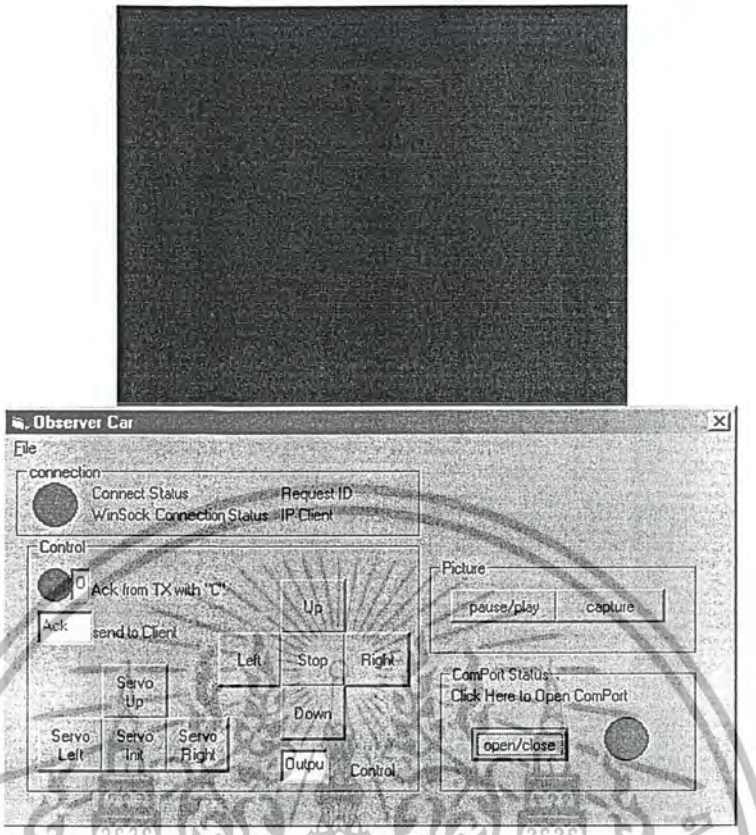
โปรแกรมควบคุมฝั่ง Server

1) เปิดโปรแกรมควบคุม เพื่อรอรับการติดต่อจากฝั่ง Client ดังรูปที่ 4.16
2) เมื่อฝั่ง Client ทำการเชื่อมต่อและเปิดพอร์ตได้แล้ว ฝั่ง Server จะเป็นดังรูปที่ 4.23
3) เมื่อกดปุ่ม Capture ที่โปรแกรมฝั่ง Server เพื่อทำการเก็บภาพ จะเป็นดังรูปที่ 4.24
4) เมื่อทำการกดภายในภาพ ดังแสดงในรูปที่ 4.24 จะปรากฏหน้าต่างเพื่อทำการบันทึกดังรูปที่ 4.26
5) เมื่อฝั่ง Client ทำการยกเลิกการเชื่อมต่อแล้ว โปรแกรมฝั่ง Server จะกลับมาเหมือนตอนเปิดโปรแกรม ได้เป็นดังรูปที่ 4.16

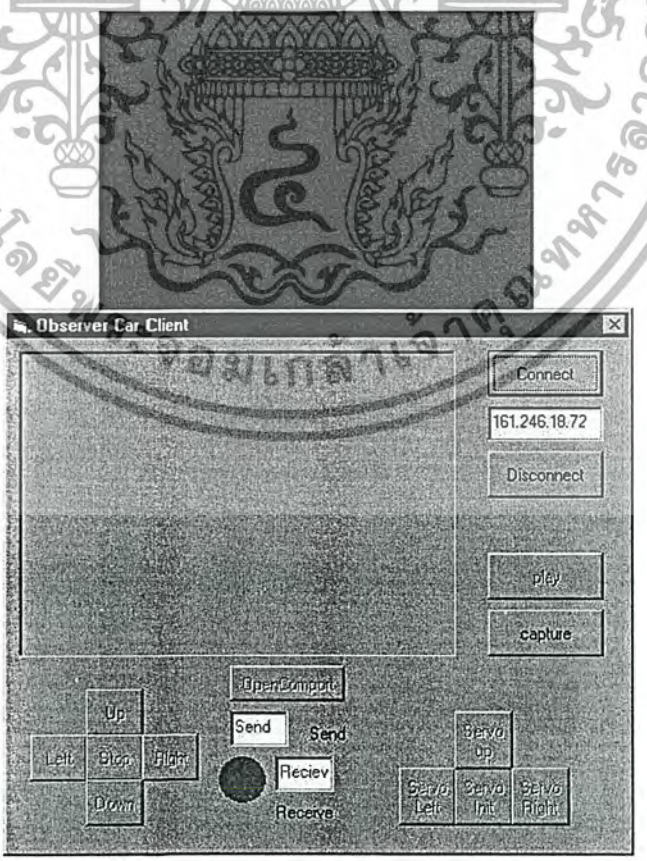
โปรแกรมควบคุมฝั่ง Client

1) เปิดโปรแกรมควบคุม จะปรากฏหน้าต่างดังรูปที่ 4.17
2) เมื่อทำการใส่หมายเลข IP Address และกดปุ่ม Connect เพื่อทำการเชื่อมต่อ ถ้าโปรแกรมสามารถเชื่อมต่อได้แล้ว จะปรากฏกล่องข้อความดังรูปที่ 4.18
3) เมื่อทำการกดปุ่ม OK ที่กล่องข้อความรูปที่ 4.18 แล้ว จะปรากฏกล่องข้อความให้ใส่รหัสผ่าน ดังรูปที่ 4.19 เพื่อป้องกันการใช้งานของผู้อื่น
4) หลังจากใส่รหัสผ่านเรียบร้อยแล้ว จะเป็นดังรูปที่ 4.20
5) เมื่อเราต้องการรับภาพที่ได้จากกล้องที่อยู่ทางฝั่ง Server ให้ทำการกดปุ่ม Play บนโปรแกรม และหากต้องการที่จะควบคุมรถและกล้องให้สามารถไปในทิศทางต่างๆได้นั้น ให้ทำการกดปุ่ม OpenComport ซึ่งจะปรากฏกล่องข้อความดังรูปที่ 4.21 แสดงว่ามีการเปิดพอร์ตอนุกรมทางฝั่ง Server เรียบร้อยแล้ว จากนั้นโปรแกรมจะมีลักษณะดังรูปที่ 4.22 ซึ่งเราจะสามารถควบคุมทิศทางการวิ่งของรถหรือทิศทางการหมุนของกล้องได้จากปุ่มต่างๆบนโปรแกรม (เพื่อความสะดวก อาจจะใช้ปุ่มตัวเลขเพื่อทำการควบคุมก็ได้)
6) เมื่อกดปุ่ม Capture ที่โปรแกรมฝั่ง Client จะมีการแสดงภาพที่ได้ดังรูปที่ 4.25
7) เมื่อทำการกดภายในภาพ ดังแสดงในรูปที่ 4.25 จะปรากฏหน้าต่างเพื่อทำการบันทึกดังรูปที่ 4.26
8) เมื่อทำการกดปุ่ม Disconnect โปรแกรมจะยกเลิกการเชื่อมต่อและทำการปิดหน้าต่าง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



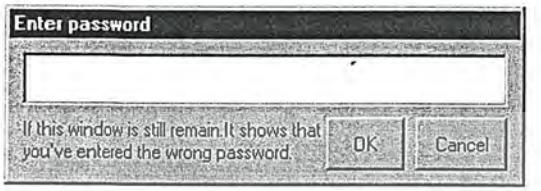
รูปที่ 4.16 โปรแกรมฝั่ง Server ขณะรอรับการติดต่อ



รูปที่ 4.17 โปรแกรมฝั่ง Client

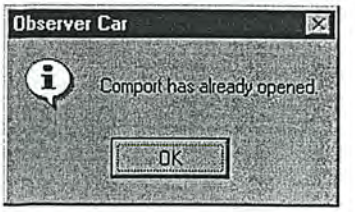
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 4.18 กล่องข้อความแสดงการเชื่อมต่อของ Winsock Control



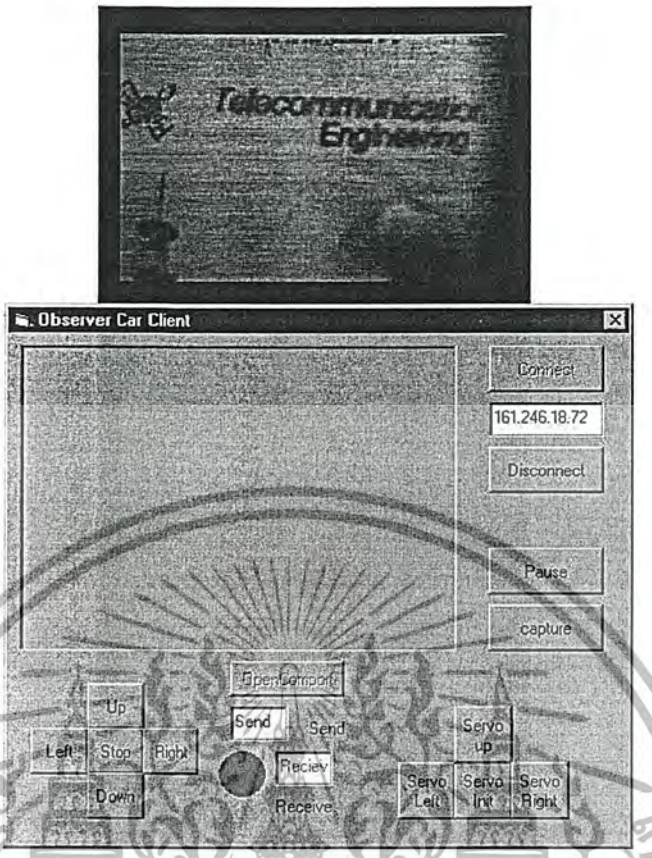
รูปที่ 4.19 กล่องข้อความสำหรับใส่รหัสผ่าน

รูปที่ 4.20 โปรแกรมฝั่ง Client หลังจากเชื่อมต่อระหว่างเครื่องได้แล้ว

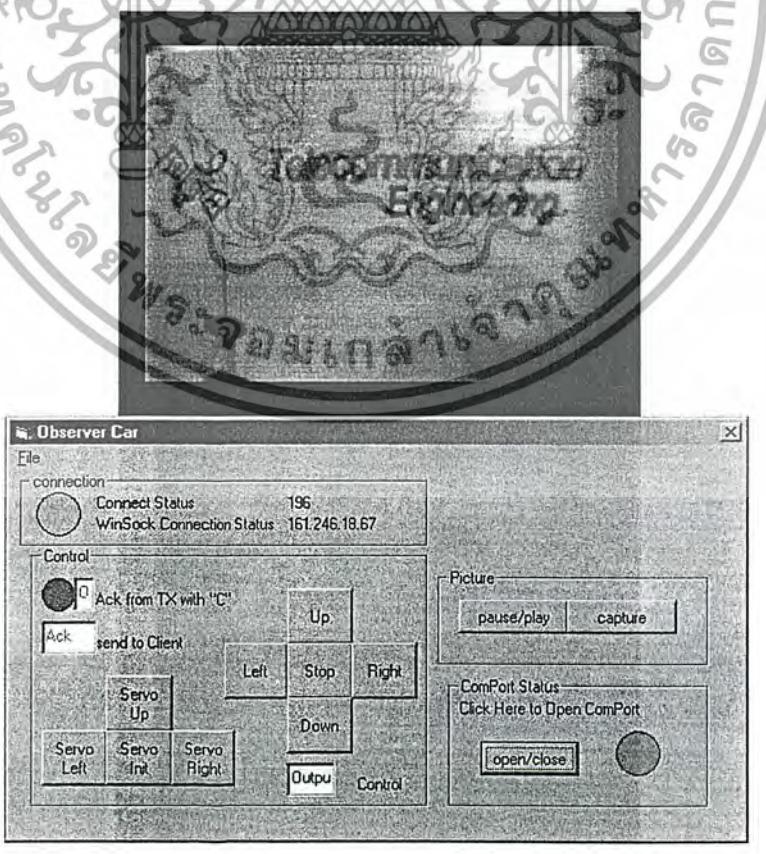


รูปที่ 4.21 กล่องข้อความแสดงการเปิดพอร์ต

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

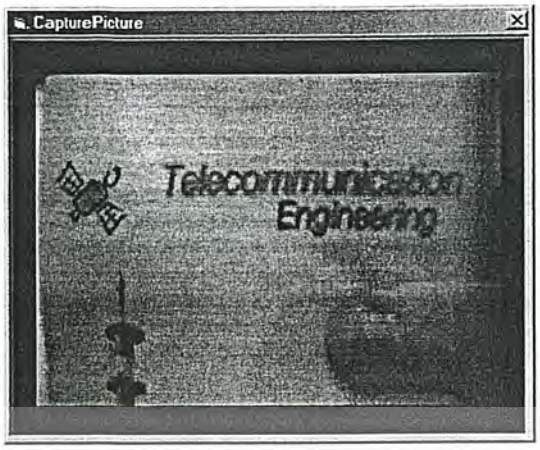


รูปที่ 4.22 โปรแกรมฝั่ง Client หลังจากเปิดพอร์ตและรับภาพได้แล้ว

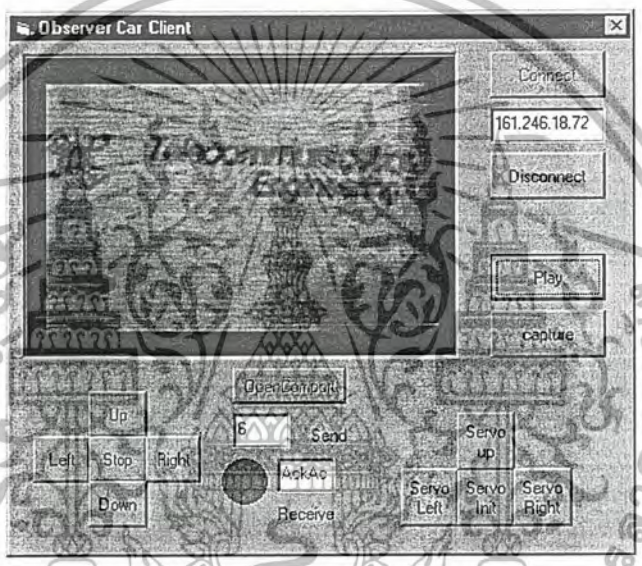


รูปที่ 4.23 โปรแกรมฝั่ง Server หลังจากเชื่อมต่อและเปิดพอร์ตแล้ว

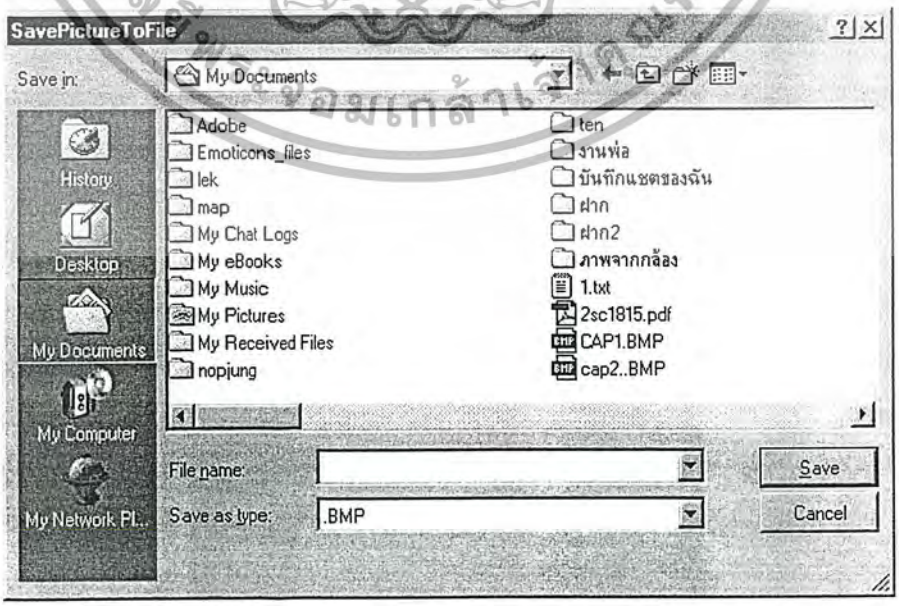
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.24 โปรแกรมฝั่ง Server ขณะทำการเก็บภาพ



รูปที่ 4.25 โปรแกรมฝั่ง Client ขณะทำการเก็บภาพ



รูปที่ 4.26 โปรแกรมขณะทำการบันทึกเก็บภาพ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 5
บทวิจารณ์และบทสรุป

จะเห็นได้ว่าในการปฏิบัติงานของทั้งภาครับ–ส่งสัญญาณควบคุมและภาครับ–ส่งสัญญาณภาพต่างก็จำเป็นที่จะต้องทำงานสัมพันธ์กันกับภาคอื่นๆ ทั้งภาคอินเตอร์เฟสกับคอมพิวเตอร์ ภาคเข้ารหัส ภาคถอดรหัส ภาคขับมอเตอร์ และประกอบกับการใช้งานอย่างถูกต้องของผู้ใช้อีกด้วย

จากการทดลองพบว่าเกิดปัญหาต่างๆหลายปัญหา ซึ่งแบ่งออกเป็นส่วนใหญ่ๆ ดังนี้

ภาครับ – ส่งสัญญาณควบคุม

ในภาคส่งสัญญาณควบคุมที่ใช้โมดูลตัวส่งของรถกระป๋องนั้น สามารถส่งสัญญาณควบคุมได้ดีในระดับหนึ่ง แต่ในระยะไกลจะเกิดปัญหากำลังในการส่งไม่เพียงพอ และเนื่องจากการเข้ารหัสนั้นมีค่าความยาวที่ไม่เท่ากัน ทำให้ในการส่งสัญญาณบางรหัสที่มีความยาวมาก เกิดการผิดพลาดมากกว่ารหัสที่มีความยาวสั้น

ในภาครับสัญญาณควบคุมที่ใช้โมดูลตัวรับของรถกระป๋องนั้น สามารถรับสัญญาณและถอดรหัสได้อย่างถูกต้อง แต่ในบางจุดจะมีจุดอับของสัญญาณ ทำให้การรับสัญญาณและการถอดรหัสมีความผิดพลาดเกิดขึ้น

ภาครับ – ส่งสัญญาณภาพ

ในภาคส่งสัญญาณภาพ สามารถส่งสัญญาณได้ไกล ส่วนในภาครับสัญญาณภาพที่ใช้การ์ด TV Tuner นั้นสามารถรับสัญญาณและแสดงผลได้ แต่มีปัญหาคือ สายอากาศที่ใช้รับสัญญาณนั้นต้องมีขนาดที่ใหญ่จึงจะสามารถรับสัญญาณได้ชัดเจน

ภาคควบคุมรถและกล้อง

สามารถรับสัญญาณมาและเคลื่อนที่ไปในทิศทางต่างๆได้อย่างถูกต้องตามโปรแกรม

ภาคติดต่อกับเครือข่ายอินเตอร์เน็ตและพอร์ตอนุกรม

การสั่งงานจากเครื่อง Client ไปยังเครื่อง Server สามารถทำงานได้อย่างถูกต้อง โดยสามารถรับสัญญาณภาพวิดีโอจากฝั่ง Server และสามารถส่งสัญญาณควบคุมไปยังฝั่ง Server ได้ถูกต้องเช่นเดียวกัน แต่สัญญาณภาพที่ทางฝั่ง Client รับได้นั้นจะเกิดการล่าช้าของสัญญาณมาก ทำให้ภาพที่ได้รับเกิดอาการกระตุกอย่างเห็นได้ชัด

แนวทางการแก้ไขปัญหา คือ แก้ไขในภาคส่งสัญญาณควบคุมให้มีการเข้ารหัสที่มีความยาวสั้นลง และเพิ่มกำลังในการส่งที่ให้เพียงพอสำหรับการควบคุมในระยะไกล

แนวทางในการพัฒนาโครงงาน คือ พัฒนาให้เป็นระบบเครือข่ายแบบโทรศัพท์เคลื่อนที่ โดยจัดรูปแบบของเครื่องส่งที่อยู่ตาม Server ต่างๆให้อยู่ในลักษณะของ Cell Site เพื่อที่จะขยายขอบเขตของการควบคุมให้ได้มากขึ้น ซึ่งจะเป็นแนวทางในการนำไปใช้สำรวจ หรือ ดูแลรักษาความปลอดภัย ในอนาคต

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Source Code ของวงจรไมโครคอนโทรลเลอร์ทางเครื่องส่งสัญญาณควบคุม

```
        ORG     0000H

; กำหนดตั้งค่าให้กับ Timer1 เพื่อใช้สร้าง Baud Rate ในการรับส่งข้อมูล
        MOV     IE,#00000000B   ;ไม่ให้ interrupt
        MOV     TMOD,#00100000B ;mode 1
        MOV     TL1,#0FDH       ;9600 baud
        MOV     TH1,#0FDH

; กำหนดค่าให้กับ SCON
        MOV     SCON,#01010000B ;mode 1,REN = 1
        SETB    TR1             ;timer 1 start
        MOV     P1,#00000000B

MAIN:   LCALL   RCD
        LCALL   SEL
        LCALL   TD
        SJMP    MAIN

;รับข้อมูลจาก Com1
RCD:    JNB     RI,$            ;รอข้อมูลเข้า
        CLR     RI
        MOV     A,SBUF
        MOV     P1,#00H
        RET

;เข้ารหัสในการส่งออก port1
SEL:    CJNE    A,#38H,SEL2
        MOV     P1,#01111111B
        RET

SEL2:   CJNE    A,#36H,SEL3
        MOV     P1,#11101111B
        RET

SEL3:   CJNE    A,#34H,SEL4
        MOV     P1,#11011111B
        RET

SEL4:   CJNE    A,#32H,SEL5
        MOV     P1,#10111111B
        RET

SEL5:   CJNE    A,#31H,SEL6
        MOV     P1,#01101111B
        RET

SEL6:   CJNE    A,#33H,SEL7
        MOV     P1,#01011111B
        RET

SEL7:   CJNE    A,#37H,SEL8
        MOV     P1,#10101111B
        RET
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
SEL8:   CJNE    A,#39H,SEL9
        MOV     P1,#10011111B
        RET

SEL9:   MOV     P1,#0FFH
        RET

; ส่งข้อมูลออกไป com1
TD:     MOV     SBUF,#43H
        JNB     TI,$
        CLR     TI
        RET

        END
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Source Code ของวงจรไมโครคอนโทรลเลอร์ทางเครื่องรับสัญญาณควบคุม

```
        ORG     0000H
        LJMP    MAIN
        ORG     000BH
        LJMP    SERVODRV2
        ORG     001BH
        LJMP    SERVODRV1

        ORG     0030H

MAIN:   MOV     P3,#0FFH
        MOV     TMOD,#11H
        MOV     IE,#8AH
        SETB    TR0
        SETB    TR1
        ACALL   SELXX

RX:     MOV     P3,#0FFH
        MOV     A,P3
        ANL     A,#00111100B
        ACALL   SELRX
        SJMP    RX

; decode ค่าที่ได้จากการรับ
SELRX:  CJNE    A,#00111000B,SELRX2
        ACALL   SELXX
        SETB    P1.6
        CLR     P1.7
        SETB    P1.5
        CLR     P1.4
        RET

SELRX2: CJNE    A,#00110100B,SELRX3
        ACALL   SELXX
        SETB    P1.7
        CLR     P1.6
        SETB    P1.4
        CLR     P1.5
        RET

SELRX3: CJNE    A,#00101100B,SELRX4
        ACALL   SELXX
        SETB    P1.7
        CLR     P1.6
        SETB    P1.5
        CLR     P1.4
        RET

SELRX4: CJNE    A,#00011100B,SELRX5
        ACALL   SELXX
        SETB    P1.6
        CLR     P1.7
        SETB    P1.4
        CLR     P1.5
        RET

SELRX5: ;#7
        CJNE    A,#00100100B,SELRX6
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
            ACALL       SELRX9
            MOV         R1,#020H
            MOV         R2,#020H
            MOV         R3,#0FAH
            MOV         R4,#0FAH
            RET

SELRX6:     ;#1
            CJNE        A,#00010100B,SELRX7
            ACALL       SELRX9
            MOV         R1,#040H
            MOV         R3,#0F7H
            RET

SELRX7:     ;#9
            CJNE        A,#00101000B,SELRX8
            ACALL       SELRX9
            MOV         R1,#020H
            MOV         R2,#0FCH
            MOV         R3,#0FAH
            MOV         R4,#0FFH
            RET

SELRX8:     ;#3
            CJNE        A,#00011000B,SELRX9
            ACALL       SELRX9
            MOV         R1,#020H
            MOV         R2,#0E0H
            MOV         R3,#0FAH
            MOV         R4,#0F5H
            RET

SELRX9:     CLR         P1.6
            CLR         P1.7
            CLR         P1.5
            CLR         P1.4
            RET

SELXX:      MOV         R1,#020H
            MOV         R2,#020H
            MOV         R3,#0FAH
            MOV         R4,#0FAH
            RET

SERVODRV1:  CLR         TR1
            MOV         TH1,R3
            MOV         TL1,R1
            SETB        TR1
            CPL         P1.3
            RETI

SERVODRV2:  CLR         TR0
            MOV         TH0,R4
            MOV         TL0,R2
            SETB        TR0
            CPL         P1.2
            RETI


            END
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Source Code ของโปรแกรมทางฝั่ง Server

```
############################Form FrmServer#############################

Option Explicit
Private WithEvents ConferenceManager As NDNM3Ctl.ConferenceManager
Private WithEvents m_Call As NDNM3Ctl.Call
Private WithEvents m_Conference As NDNM3Ctl.Conference
Private WithEvents m_MyVideo As NDNM3Ctl.VideoChannel
Private WithEvents m_TheirVideo As NDNM3Ctl.VideoChannel
 Dim ControlAccept As Boolean
 Dim tpData As String

'Our application ID. Use something different for your own custom
application
Const cstrAppGuid = "{DAA6C931-A99F-4bc3-B975-8B1C05CA3217}"

Private Sub cmdcapture_Click()
       Load FrmCapture

             ' capture button

              FrmVideo.SetFocus
              FrmVideo.Capture1.CaptureType = vbClient

               FrmCapture.Width = FrmVideo.Width + 200
               FrmCapture.Height = FrmVideo.Height + 400
               FrmCapture.Picture1.Width = FrmVideo.Width + 100
               FrmCapture.Picture1.Height = FrmVideo.Height
             FrmVideo.Capture1.Capture
             FrmCapture.Picture1.Cls
             FrmCapture.Picture1.Picture = FrmVideo.Capture1.Picture
            FrmCapture.Show
           FrmCapture.Refresh


           frmServer.SetFocus
End Sub


Private Sub CmdDown_Click()
    txtOutput.Text = "2"
End Sub


Private Sub cmdleft_Click()
 txtOutput.Text = "4"
End Sub

Private Sub cmdlocalpause_Click()
    FrmVideo.LocalVideo1.Paused = Not FrmVideo.LocalVideo1.Paused

End Sub

Private Sub cmdRight_Click()
    txtOutput.Text = "6"
End Sub

Private Sub cmdServoInit_Click()
txtOutput.Text = "7"
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
End Sub

Private Sub cmdServoL_Click()
txtOutput.Text = "3"
End Sub

Private Sub cmdServoR_Click()
txtOutput.Text = "9"
End Sub

Private Sub cmdServoUp_Click()
txtOutput.Text = "1"
End Sub

Private Sub cmdStop_Click()
txtOutput.Text = "5"
End Sub

Private Sub cmdUP_Click()
    txtOutput.Text = "8"
End Sub


Private Sub Form_Activate()
    FrmVideo.Move frmServer.Left + 1250, frmServer.Top -
FrmVideo.Height
    If tcpServer.State = sckClosed Then
    tcpServer.LocalPort = 1001
    tcpServer.Listen
    End If
    End Sub

Private Sub Form_Deactivate()
  FrmVideo.Move frmServer.Left + 1250, frmServer.Top -
FrmVideo.Height
End Sub

Private Sub Form_Initialize()
    'You can use either syntax:
    'Set ConferenceManager = New NDNM3Ctl.ConferenceManager
    'or
    Set ConferenceManager = CreateObject("NDNM3.ConferenceManager")

End Sub
Private Sub Form_load()
Dim ThisSystem As NDNM3Ctl.LocalSystem
   Dim ThisConference As NDNM3Ctl.Conference

   Set ThisSystem = ConferenceManager.LocalSystem
    If ThisSystem.IsInstalled Then
      Debug.Print "NetMeeting is installed."
    Else
        MsgBox "NetMeeting is not installed on this system."
    End If
    If ThisSystem.IsRunning Then
        Debug.Print "NetMeeting is running."
          End If

   'Setting Async to True allows you to perform other
    'tasks while waiting for NetMeeting to initialize
    ConferenceManager.Initialize Async:=True
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    Me.Move 4000, 6000

      'Load the form containing the video windows

      Load FrmVideo
      FrmVideo.Show

     Unload FrmCapture
      ControlAccept = False

      '--------------------------- Winsock comment----------------------
 -----------------
      ' Set the LocalPort property to an integer.
      ' Then invoke the Listen method.
      tcpServer.LocalPort = 1001
      tcpServer.Listen
      Debug.Print "portListen"
      MSComm1.CommPort = 1
         MSComm1.RThreshold = 1    ' Use Event from Input
      MSComm1.InputLen = 1      ' read buffer of  MSComm 1 char


     'frmClient.Show ' Show the client form.
 End Sub
 Private Sub Form_MouseMove(Button As Integer, Shift As Integer, X As
 Single, Y As Single)
     FrmVideo.Move frmServer.Left + 1250, frmServer.Top -
 FrmVideo.Height

 End Sub

 Private Sub Form_QueryUnload(Cancel As Integer, UnloadMode As
 Integer)
   ' If FrmCapture.Appearance <> vbHide Then
       FrmCapture.Hide
   '    End If
    Unload FrmCapture

 End Sub
 Private Sub Form_Unload(Cancel As Integer)
         FrmVideo.LocalVideo1.Paused = True
           If Not (CurrentCall Is Nothing) Then
         CurrentCall.Cancel
     End If
     If Not (CurrentConference Is Nothing) Then
         On Error Resume Next
         CurrentConference.Leave
         If Err.Number <> 0 Then
            Debug.Print "Error encountered leaving the conference."
         End If
         Set CurrentConference = Nothing
     End If
     FrmVideo.LocalVideo1.Paused = True
     FrmVideo.Hide
     Unload FrmVideo


 End Sub
 Private Sub mnucapture_Click(Index As Integer)
           Load FrmCapture
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
            ' capture menu

              FrmVideo.SetFocus
              FrmVideo.Capture1.CaptureType = vbClient

                FrmCapture.Width = FrmVideo.Width + 200
                FrmCapture.Height = FrmVideo.Height + 400
                FrmCapture.Picture1.Width = FrmVideo.Width + 100
                FrmCapture.Picture1.Height = FrmVideo.Height
             FrmVideo.Capture1.Capture
             FrmCapture.Picture1.Cls
             FrmCapture.Picture1.Picture = FrmVideo.Capture1.Picture
           FrmCapture.Show
         FrmCapture.Refresh

             frmServer.SetFocus

End Sub

Private Sub MnuExit_Click()
    Me.Hide
 Unload Me
End Sub

Private Sub tcpServer_Close()

        Label1.Caption = "requestID"
    Shape1.FillColor = vbRed
    Label2.Caption = "RemoteHostIP"

    Me.Hide
    Unload Me
    Load Me
   Me.Show
 Me.Hide
    Unload Me
    Load Me
   Me.Show

 End Sub

Private Sub tcpServer_ConnectionRequest _
(ByVal requestID As Long)

    ' Check if the control's State is closed. If not,
    ' close the connection before accepting the new
    ' connection.
    If tcpServer.State <> sckClosed Then
    tcpServer.Close
    ' Accept the request with the requestID
    ' parameter.

    End If
    tcpServer.Accept requestID
    Label1.Caption = requestID
    Shape1.FillColor = &H80FF80
    Label2.Caption = tcpServer.RemoteHostIP


End Sub
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
Private Sub txtOutput_Change()
  Dim tpData As String
If MSComm1.PortOpen = False Then
MsgBox "Please Open ComPort", vbOKOnly, "Observer Car"
Else

    Select Case txtOutput.Text
        Case "8"
     tpData = Hex(56)
           Case "4"
        tpData = Hex(54)
           Case "6"
     tpData = Hex(52)
           Case "2"
       tpData = Hex(50)
           Case "7"
        tpData = Hex(55)
            Case "9"
     tpData = Hex(51)
        Debug.Print "Send 9"
           Case "1"
        tpData = Hex(49)
           Case "3"
         tpData = Hex(57)
        Debug.Print "Send 3"
         Case Else
    tpData = Hex(53)
        End Select


MSComm1.Output = tpData    ' send data out of serial
'Label5 = Str(Asc(txtOutput.Text))
't xtOutput.Text = ""
Shape3.FillColor = vbRed
End If
End Sub
Private Sub txtOutput_KeyDown(KeyCode As Integer, Shift As Integer)
txtOutput.Text = ""
End Sub

Private Sub txtSendData_Change()
    ' The TextBox control named txtSendData
    ' contains the data to be sent. Whenever the user
    ' types into the  textbox, the  string is sent
    ' using the SendData method.

    tcpServer.SendData txtSendData.Text 'ส่งค่ากลับไปยัง Client ตามค่าที่อยู่ใน textbox


End Sub

Private Sub tcpServer_DataArrival _
(ByVal bytesTotal As Long)
    ' Declare a variable for the incoming data.
    ' Invoke the GetData method and set the Text
    ' property of a TextBox named txtOutput to
    ' the data.
    Dim strData As String
    Dim strTemp As String
    tcpServer.GetData strData
    If strData = "mukpumppop" Then 'Password
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
            tcpServer.SendData "Accept"
            'CurrentCall.Accept
            ControlAccept = True
            Exit Sub
    ElseIf ControlAccept = True Then
        If strData = "OpenSesame" Then
            If MSComm1.PortOpen = False Then
                MSComm1.PortOpen = True
                Shape2.FillColor = vbGreen
                tcpServer.SendData "ComportOpened"
                Exit Sub
          Else
                tcpServer.SendData "ComportOpened"
                Exit Sub
            End If
        End If
  '  Debug.Print strData

    Label3.Caption = strData

    txtOutput.Text = strData
    End If
End Sub

Private Sub Command1_Click()
    ' open comport
    On Error Resume Next
    If MSComm1.PortOpen = False Then
    MSComm1.Settings = "9600,n,8,1" ' Setting
   MSComm1.CommPort = 1    ' Com port set
  '  MSComm1.RThreshold = 1
   MSComm1.PortOpen = True    ' open Com port
   Shape2.FillColor = &HFF00&
  ' MSComm1.InputLen = 1
    Else
    MSComm1.PortOpen = False
    Shape2.FillColor = &HFF&

    End If

End Sub
Private Sub MSComm1_OnComm()
    Select Case MSComm1.CommEvent
      Case comEvReceive    ' Recieve state
      Dim Buffer As Variant
          Text1.Text = ""
           Buffer = MSComm1.Input

      Text1.Text = Buffer

     If Buffer = "C" Then
      Shape3.FillColor = vbGreen
      End If

         If tcpServer.State = sckConnected Then
        tcpServer.SendData txtSendData.Text
      'Else
      ' MsgBox "Please Open Connection", vbOKOnly, "Observer Car"
     End If

    End Select
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
End Sub
'-------------------------------------------------------
'
'Conference manager events
'
'-------------------------------------------------------
Private Sub ConferenceManager_CallCreated(ByVal NewCall As
NDNM3Ctl.Call)
    Debug.Print "ConferenceManager_CallCreated"

    Set CurrentCall = NewCall
    If CurrentCall.IsIncoming Then
        Debug.Print "The call is incoming."
    Else
        Debug.Print "The call is outgoing."
    End If
    If Not (CurrentCall.Conference Is Nothing) Then

        Debug.Print "CallCreated " &
CurrentCall.CurrentConference.ChannelCount
    End If
End Sub
Private Sub ConferenceManager_ConferenceCreated(ByVal NewConference
As NDNM3Ctl.Conference)
    Debug.Print "ConferenceManager_ConferenceCreated"
    Set CurrentConference = NewConference
   Debug.Print "Channelcount =" & CurrentConference.ChannelCount
   Debug.Print "CurrentConference.State " & CurrentConference.State
End Sub

Private Sub ConferenceManager_Initialization(ByVal Initialized As
Boolean)
    Debug.Print "ConferenceManager_Initialization"

End Sub
Private Sub ConferenceManager_NetMeetingStarted()
    Debug.Print "ConferenceManager_NetMeetingStarted"
End Sub

'
'Call events
'
'-------------------------------------------------------
Private Sub m_Call_Accepted(ByVal Conference As NDNM3Ctl.Conference)
    Debug.Print "Call_Accepted"
End Sub

Private Sub m_Call_Failed(ByVal CallError As NDNM3Ctl.ECallError)
    Debug.Print "Call_Failed: Error # " & CallError
End Sub

Private Sub m_Call_Information(ByVal Information As
NDNM3Ctl.EInformationEvent)
    Debug.Print "Call_Information: " & Information
    Select Case Information
    Case EInformationEvent.einfocallFailed
        Debug.Print "The call failed."
    Case EInformationEvent.einfocallIgnored
        Debug.Print "The call was ignored (user selected ""Ignore""
or the pop-up dialog timed out)."
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    Case EInformationEvent.einfocallIncompatible
        Debug.Print "The remote version of NetMeeting is not
compatible with the local system."
    Case EInformationEvent.einfocallInConference
        Debug.Print "The remote system is already in a conference."
    Case EInformationEvent.einfocallNewer
        Debug.Print "The remote system is a later version of
NetMeeting than the local one."
    Case EInformationEvent.einfocallOlder
        Debug.Print "The remote system is an earlier version of
NetMeeting (for example, NetMeeting 2.1 is calling NetMeeting 2.0 or
1.0)."
    End Select
End Sub

Private Sub m_Call_StateChanged(ByVal CallState As
NDNM3Ctl.ECallState)
    Dim mbreply As VbMsgBoxResult

    Debug.Print "Call_StateChanged: " & CallState
    Select Case CallState
    Case ECallState.csInvalid
        Debug.Print "The call is invalid."

    Case ECallState.csInit
        Debug.Print "The call is being initialized."

    Case ECallState.csRing
        Debug.Print "The incoming call is waiting for a response."
        'Notify the user of an incoming call.

            CurrentCall.Accept
            If FrmVideo.LocalVideo1.Paused Then
            FrmVideo.LocalVideo1.Paused = Not
FrmVideo.LocalVideo1.Paused
            End If


    Case ECallState.csSearch
        Debug.Print "The outgoing call is in the process of finding
the destination address."
    Case ECallState.csWait
        Debug.Print "The outgoing call is waiting for a response from
the remote system."
    Case ECallState.csAccepted
        Debug.Print "The call has been accepted."
        'Notify the user we have connected

        Set CurrentCall = Nothing
    Case ECallState.csRejected
        Debug.Print "The call was rejected or ignored."


    Case ECallState.csCanceled
        Debug.Print "The call has been canceled."


        Set CurrentCall = Nothing
    End Select
End Sub
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
'-------------------------------------------------------
'
'Video channel events
'
'-------------------------------------------------------
Private Sub m_MyVideo_Information(ByVal Information As
NDNM3Ctl.EInformationEvent)
    'This event never fires - it is only included because the
VideoChannel object
    'implements the IChannel interface
End Sub
Private Sub m_TheirVideo_Information(ByVal Information As
NDNM3Ctl.EInformationEvent)
    'This event never fires - it is only included because the
VideoChannel object
    'implements the IChannel interface
End Sub
Private Sub m_Conference_VideoChannelAdded(ByVal AddedChannel As
NDNM3Ctl.VideoChannel)

    'Make sure the video windows form is loaded

End Sub

Private Property Get CurrentCall() As NDNM3Ctl.Call
    Set CurrentCall = m_Call
End Property

Private Property Set CurrentCall(value As NDNM3Ctl.Call)

  Set m_Call = value
End Property



Private Sub DoConnection()
    m_Conference.CreateDataChannel "{DAA6C931-A99F-4bc3-B975-
8B1C05CA3217}"
End Sub


'-------------------------------------------------------
'
'Conference events
'
'-------------------------------------------------------
Private Sub m_Conference_Information(ByVal Information As
NDNM3Ctl.EInformationEvent)
    'This event never fires - it is only included because the
Conference object
    'implements the IChannel interface
End Sub

Private Sub m_Conference_MemberAdded(ByVal ChangedMember As
NDNM3Ctl.Member)
    Debug.Print "Added " & ChangedMember.Name
End Sub
Private Sub m_Conference_MemberUpdated(ByVal ChangedMember As
NDNM3Ctl.Member)
    Debug.Print "Updated " & ChangedMember.Name
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
End Sub
Private Sub m_Conference_MemberRemoved(ByVal ChangedMember As
NDNM3Ctl.Member)
    Debug.Print "Removed " & ChangedMember.Name

End Sub


Private Sub m_Conference_StateChanged(ByVal ConferenceState As
NDNM3Ctl.EConferenceState)
    Debug.Print "Conference_StateChanged : " & ConferenceState
    Select Case ConferenceState
    Case EConferenceState.csActive
        Debug.Print "The conference is active. Most operations on a
conference require it to be in this state."

    Case EConferenceState.csIdle
        Debug.Print "The conference is not active. It contains no
members or channels."

        Set CurrentConference = Nothing
    Case EConferenceState.csInitializing
        Debug.Print "The conference is transitioning between the idle
and active states. The application should wait for a notification
that the state has changed to either waiting or active before
attempting to perform any other conference operations."

    Case EConferenceState.csWaiting
        Debug.Print "A hosted conference with only the local Member
participating."

    End Select
End Sub

Private Property Get CurrentConference() As NDNM3Ctl.Conference
    Set CurrentConference = m_Conference
End Property

Private Property Set CurrentConference(value As NDNM3Ctl.Conference)
    Set m_Conference = value

End Property

##########################Form FrmVideo##########################

Private Sub LocalVideo1_Initialization(ByVal Success As Boolean)
    Debug.Print "LocalVideo1_Initialization"
'   LocalVideo1.Paused = False
 '  If LocalVideo1.Paused Then
   '     cmdPauseLocal.Caption = "Play"
 '   Else
   '     cmdPauseLocal.Caption = "Pause"
'   End If
End Sub
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
############################Form FrmCapture#############################
Private Sub Form_load()
       ' Capture button
                ' SavePicture LocalVideo1.Picture, "OBSERV1.BMP"
                FrmVideo.SetFocus
                FrmVideo.Capture1.CaptureType = vbClient
                'FrmVideo.Capture1.ScaleMode = vbPixels
                 'Capture1.ScaleHeight = FrmVideo.Height - 400
                  Picture1.Width = FrmVideo.Width
                  Picture1.Height = FrmVideo.Height
               FrmVideo.Capture1.Capture
               Picture1.Picture = FrmVideo.Capture1.Picture

           '    Picture1.Height = FrmVideo.Height - 200

'              frmServer.SetFocus
End Sub

Private Sub Picture1_Click()
On Error GoTo ErrHandler
     CommonDialog1.ShowSave
      SavePicture Picture1.Picture, CommonDialog1.FileName


ErrHandler:
' User pressed Cancel button.
   Exit Sub

End Sub
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Source Code ของโปรแกรมทางฝั่ง Client

```
############################Form FrmClient#############################

Option Explicit
Private WithEvents ConferenceManager As NDNM3Ctl.ConferenceManager
Private WithEvents m_Call As NDNM3Ctl.Call
Private WithEvents m_Conference As NDNM3Ctl.Conference

Private WithEvents m_MyVideo As NDNM3Ctl.VideoChannel
Private WithEvents m_TheirVideo As NDNM3Ctl.VideoChannel
'Our application ID. Use something different for your own custom
application
Const cstrAppGuid = "{DAA6C931-A99F-4bc3-B975-8B1C05CA3217}"

Private Sub cmdCapture_Click()
    FrmVideo.SetFocus
    FrmVideo.Capture1.CaptureType = vbClient
    Picture1.Width = FrmVideo.RemoteVideo1.Width
    Picture1.Height = FrmVideo.RemoteVideo1.Height
    FrmVideo.Capture1.Capture
    Picture1.Picture = FrmVideo.Capture1.Picture
    frmClient.SetFocus
End Sub

Private Sub cmdDisconnect_Click()

    Me.Hide
    Unload Me

End Sub

Private Sub cmdDown_Click()
 txtSend.Text = "2"
End Sub

Private Sub cmdLeft_Click()
 txtSend.Text = "4"
End Sub

Private Sub CmdOpenComm_Click()
    tcpClient.SendData "OpenSesame"
    CmdOpenComm.Enabled = False
End Sub

Private Sub cmdPauseRemote_Click()
    FrmVideo.RemoteVideo1.Paused = Not FrmVideo.RemoteVideo1.Paused
    If FrmVideo.RemoteVideo1.Paused Then
        cmdPauseRemote.Caption = "Play"
    Else
        cmdPauseRemote.Caption = "Pause"
    End If
End Sub

Private Sub cmdRight_Click()
 txtSend.Text = "6"
End Sub

Private Sub cmdServoInit_Click()
 txtSend.Text = "7"
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
End Sub

Private Sub cmdServoLeft_Click()
 txtSend.Text = "3"
End Sub

Private Sub cmdServoRight_Click()
 txtSend.Text = "9"
End Sub

Private Sub cmdServoUp_Click()
 txtSend.Text = "1"
End Sub

Private Sub cmdStop_Click()
 txtSend.Text = "5"
End Sub

Private Sub cmdUp_Click()
        txtSend.Text = "8"
End Sub

Private Sub Form_Activate()
        FrmVideo.Move frmClient.Left + 1250, frmClient.Top -
FrmVideo.Height
End Sub

Private Sub Form_Initialize()
    'You can use either syntax:
    'Set ConferenceManager = New NDNM3Ctl.ConferenceManager
    'or
    Set ConferenceManager = CreateObject("NDNM3.ConferenceManager")
End Sub
Private Sub Form_Load()
Dim ThisSystem As NDNM3Ctl.LocalSystem
    Dim ThisConference As NDNM3Ctl.Conference

    Set ThisSystem = ConferenceManager.LocalSystem
    If ThisSystem.IsInstalled Then
        Debug.Print "NetMeeting is installed."
    Else
        MsgBox "NetMeeting is not installed on this system."
    End If
    If ThisSystem.IsRunning Then
        Debug.Print "NetMeeting is running."
    End If

    'Setting Async to True allows you to perform other
    'tasks while waiting for NetMeeting to initialize
    ConferenceManager.Initialize Async:=True
    'Load the form containing the video windows

    Load FrmVideo
    FrmVideo.Show
     Picture1.Width = FrmVideo.RemoteVideo1.Width
    Picture1.Height = FrmVideo.RemoteVideo1.Height
    If tcpClient.State = sckConnected Then
        txtSend.Enabled = True
    End If
    Debug.Print "TCP State" & tcpClient.State
End Sub
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
Private Sub Form_MouseMove(Button As Integer, Shift As Integer, X As
Single, Y As Single)
    FrmVideo.Move frmClient.Left + 1250, frmClient.Top -
FrmVideo.Height
End Sub

Private Sub Form_Unload(Cancel As Integer)


      If Not (CurrentCall Is Nothing) Then
        CurrentCall.Cancel
    End If
    If Not (CurrentConference Is Nothing) Then
        On Error Resume Next
        CurrentConference.Leave
         FrmVideo.RemoteVideo1.Paused = True
         Debug.Print "video already pause"
        If Err.Number <> 0 Then
            Debug.Print "Error encountered leaving the conference."
        End If
        Set CurrentConference = Nothing
    End If
    'If tcpClient.State = sckConnected Then
        tcpClient.Close
    ' End If


    FrmVideo.Hide
    Unload FrmVideo
End Sub

Private Sub cmdConnect_Click()



    ' Invoke the Connect method to initiate a
    ' connection.
    '------------------winsock comment
    ' The name of the Winsock control is tcpClient.
    ' Note: to specify a remote host, you can use
    ' either the IP address (ex: "121.111.1.1") or
    ' the computer's "friendly" name, as shown here.

    tcpClient.RemoteHost = txtEnterIP.Text

    tcpClient.RemotePort = 1001
    tcpClient.Connect
' txtSend.SetFocus
    txtEnterIP.Text = tcpClient.RemoteHost
    cmdConnect.Enabled = False
End Sub

Private Sub Picture1_Click()

    On Error GoTo errorhandler
    CommonDialog1.ShowSave
    SavePicture Picture1.Picture, CommonDialog1.FileName
errorhandler:
   Exit Sub
End Sub
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
Private Sub tcpClient_Close()
    If  Not (CurrentCall Is Nothing) Then
        CurrentCall.Cancel
    End If
    If  Not (CurrentConference Is Nothing) Then
      ' On Error Resume Next
      ' CurrentConference.Leave
      ' If Err.Number <> 0 Then
      ' Debug.Print "Error encountered leaving the conference."
      ' End If
        Set CurrentConference = Nothing
    End If

    Me.Hide
    Unload Me
    Load Me
    Me.Show
    ConferenceManager.Initialize Async:=True
End Sub
Private Sub tcpClient_Connect()


    MsgBox "Winsock Connection Establish", vbOKOnly, "Observer car"
    Load frmPasswd
    frmPasswd.Show
    cmdConnect.Enabled = False


End Sub


Private Sub txtSend_Change()
    tcpClient.SendData txtSend.Text
    Shape1.FillColor = vbRed

    End Sub

Private Sub tcpClient_DataArrival _
(ByVal bytesTotal As Long)
    Dim strData As String
     Dim TempCall As NDNM3Ctl.Call

    tcpClient.GetData strData
    If strData = "Accept" Then
        CmdOpenComm.Enabled = True
         frmPasswd.Hide
         Unload frmPasswd
         '-------------------------------------------------------------
------------------

'
    If CurrentConference Is Nothing Then
        Set CurrentCall = ConferenceManager.CreateCall(ctDefault,
atIP, txtEnterIP.Text)
    Else
        Set CurrentCall = ConferenceManager.CreateCall(ctDefault,
atIP, txtEnterIP.Text, CurrentConference)
    End If
    Set TempCall = CurrentCall
    Set TempCall = Nothing
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    '-------------------------------------------------------------
---------------------

    '-------------------------------------------------------------
-------

    ElseIf strData = "ComportOpened" Then
    MsgBox "Comport has already opened.", vbOKOnly + vbInformation,
"Observer Car"
    CmdOpenComm.Enabled = False
     txtSend.Enabled = True
         cmdUp.Enabled = True
         cmdDown.Enabled = True
         cmdRight.Enabled = True
         cmdLeft.Enabled = True
         cmdServoUp.Enabled = True
          cmdServoInit.Enabled = True
          cmdServoLeft.Enabled = True
          cmdServoRight.Enabled = True
         cmdStop.Enabled = True

         cmdConnect.Enabled = False


    Else 'If strData = "Ack" Then
     Shape1.FillColor = vbGreen
     txtOutput.Text = strData

     txtSend.SetFocus
    ' txtSend.Text = ""
    End If
    '-------------------------------------------------------------
---------------------


    '-------------------------------------------------------------
---------------


End Sub

'-------------------------------------------------------
'
'Conference manager events
'
'-------------------------------------------------------
Private Sub ConferenceManager_CallCreated(ByVal NewCall As
NDNM3Ctl.Call)
    Debug.Print "ConferenceManager_CallCreated"

    Set CurrentCall = NewCall
    If CurrentCall.IsIncoming Then
        Debug.Print "The call is incoming."
    Else
        Debug.Print "The call is outgoing."
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    End If
    If Not (CurrentCall.Conference Is Nothing) Then
        Debug.Print CurrentCall.CurrentConference.ChannelCount
    End If
End Sub
Private Sub ConferenceManager_ConferenceCreated(ByVal NewConference
As NDNM3Ctl.Conference)


    Debug.Print "ConferenceManager_ConferenceCreated"
    Set CurrentConference = NewConference
    Debug.Print CurrentConference.ChannelCount
    Debug.Print CurrentConference.State
    End Sub

Private Sub ConferenceManager_Initialization(ByVal Initialized As
Boolean)
    Debug.Print "ConferenceManager_Initialization"

End Sub
Private Sub ConferenceManager_NetMeetingStarted()
    Debug.Print "ConferenceManager_NetMeetingStarted"
End Sub

'
'Call events
'
'---------------------------------------------------------
Private Sub m_Call_Accepted(ByVal Conference As NDNM3Ctl.Conference)
    Debug.Print "Call_Accepted"

End Sub

Private Sub m_Call_Failed(ByVal CallError As NDNM3Ctl.ECallError)
    Debug.Print "Call_Failed: Error # " & CallError
End Sub

Private Sub m_Call_Information(ByVal Information As
NDNM3Ctl.EInformationEvent)
    Debug.Print "Call_Information: " & Information
    Select Case Information
    Case EInformationEvent.einfocallFailed
        Debug.Print "The call failed."
    Case EInformationEvent.einfocallIgnored
        Debug.Print "The call was ignored (user selected ""Ignore""
or the pop-up dialog timed out)."
    Case EInformationEvent.einfocallIncompatible
        Debug.Print "The remote version of NetMeeting is not
compatible with the local system."
    Case EInformationEvent.einfocallInConference
        Debug.Print "The remote system is already in a conference."
    Case EInformationEvent.einfocallNewer
        Debug.Print "The remote system is a later version of
NetMeeting than the local one."
    Case EInformationEvent.einfocallOlder
        Debug.Print "The remote system is an earlier version of
NetMeeting (for example, NetMeeting 2.1 is calling NetMeeting 2.0 or
1.0)."
    End Select
End Sub
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
Private Sub m_Call_StateChanged(ByVal CallState As
NDNM3Ctl.ECallState)
    Dim mbReply As VbMsgBoxResult

    Debug.Print "Call_StateChanged: " & CallState
    Select Case CallState
    Case ECallState.csInvalid
        Debug.Print "The call is invalid."

    Case ECallState.csInit
        Debug.Print "The call is being initialized."

    Case ECallState.csRing
        Debug.Print "The incoming call is waiting for a response."

        mbReply = MsgBox(CurrentCall.Name & " is requesting a call."
& vbNewLine & "Would you like to accept it?", vbInformation +
vbYesNo, "NDNM3 Test Harness")
        Select Case mbReply
        Case vbYes
            CurrentCall.Accept
        Case vbNo
            CurrentCall.Cancel
        End Select
    Case ECallState.csSearch
        Debug.Print "The outgoing call is in the process of finding
the destination address."

    Case ECallState.csWait
        Debug.Print "The outgoing call is waiting for a response from
the remote system."

    Case ECallState.csAccepted
        Debug.Print "The call has been accepted."
        cmdPauseRemote.Enabled = True

        Set CurrentCall = Nothing
    Case ECallState.csRejected
        Debug.Print "The call was rejected or ignored."
        MsgBox "The call was rejected or ignored.", vbOKOnly,
"Observer Car"

        Set CurrentCall = Nothing
    Case ECallState.csCanceled
        Debug.Print "The call has been canceled."

        Set CurrentCall = Nothing
    End Select
End Sub

'-------------------------------------------------------
'
'Video channel events
'
'-------------------------------------------------------
Private Sub m_MyVideo_Information(ByVal Information As
NDNM3Ctl.EInformationEvent)
    'This event never fires - it is only included because the
VideoChannel object
    'implements the IChannel interface
End Sub
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
Private Sub m_TheirVideo_Information(ByVal Information As
NDNM3Ctl.EInformationEvent)
    'This event never fires - it is only included because the
VideoChannel object
    'implements the IChannel interface
End Sub


Private Property Get CurrentCall() As NDNM3Ctl.Call
    Set CurrentCall = m_Call
End Property

Private Property Set CurrentCall(value As NDNM3Ctl.Call)
    Set m_Call = value

End Property


Private Sub DoConnection()
    m_Conference.CreateDataChannel "{DAA6C931-A99F-4bc3-B975-
8B1C05CA3215}"
End Sub
Private Sub txtEnterIP_KeyPress(KeyAscii As Integer)
    If KeyAscii = 13 Then
        If cmdConnect.Enabled Then cmdConnect_Click
        'Cancel the keystroke - we handled it
        KeyAscii = 0
    End If
End Sub
'--------------------------------------------------------
'
'Conference events
'
'--------------------------------------------------------
Private Sub m_Conference_Information(ByVal Information As
NDNM3Ctl.EInformationEvent)
    'This event never fires - it is only included because the
Conference object
    'implements the IChannel interface
End Sub

Private Sub m_Conference_MemberAdded(ByVal ChangedMember As
NDNM3Ctl.Member)
    Debug.Print "Added " & ChangedMember.Name

End Sub
Private Sub m_Conference_MemberUpdated(ByVal ChangedMember As
NDNM3Ctl.Member)
    Debug.Print "Updated " & ChangedMember.Name

End Sub
Private Sub m_Conference_MemberRemoved(ByVal ChangedMember As
NDNM3Ctl.Member)
    Debug.Print "Removed " & ChangedMember.Name

End Sub
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
Private Sub m_Conference_StateChanged(ByVal ConferenceState As
NDNM3Ctl.EConferenceState)
    Debug.Print "Conference_StateChanged : " & ConferenceState
    Select Case ConferenceState
    Case EConferenceState.csActive
        Debug.Print "The conference is active. Most operations on a
conference require it to be in this state."

    Case EConferenceState.csIdle
        Debug.Print "The conference is not active. It contains no
members or channels."

        Set CurrentConference = Nothing
    Case EConferenceState.csInitializing
        Debug.Print "The conference is transitioning between the idle
and active states. The application should wait for a notification
that the state has changed to either waiting or active before
attempting to perform any other conference operations."

    Case EConferenceState.csWaiting
        Debug.Print "A hosted conference with only the local Member
participating."

    End Select
End Sub

Private Sub m_Conference_VideoChannelAdded(ByVal AddedChannel As
NDNM3Ctl.VideoChannel)

End Sub

Private Property Get CurrentConference() As NDNM3Ctl.Conference
    Set CurrentConference = m_Conference
End Property


Private Property Set CurrentConference(value As NDNM3Ctl.Conference)
    Set m_Conference = value


End Property

Private Sub txtSend_KeyDown(KeyCode As Integer, Shift As Integer)
    txtSend.Text = ""
End Sub
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
############################Form FrmPasswd#############################

Private Sub cmdCancel_Click()
    txtPasswd.Text = ""
    frmPasswd.Hide
    Unload frmPasswd
End Sub

Private Sub cmdOK_Click()

    frmClient.tcpClient.SendData txtPasswd.Text
     txtPasswd.Text = ""

End Sub
Private Sub txtPasswd_KeyPress(KeyAscii As Integer)
    If KeyAscii = 13 Then
        If cmdOK.Enabled Then cmdOK_Click
        'Cancel the keystroke - we handled it
        KeyAscii = 0
    End If
End Sub


############################Form FrmVideo#############################

Private Sub RemoteVideo1_Initialization(ByVal Success As Boolean)
    Debug.Print "RemoteVideo1_Initialization"
    'RemoteVideo1.Paused = False
End Sub
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

CCIR TV Channel Table

Channel Number	Vision Carrier MHz	Sound Carrier MHz	Upper Limit MHz	Lower Limit MHz
Band I				
2	48.25	53.75	47	54
3	55.25	60.75	54	61
4	62.25	67.75	61	68
Lower Hyperband USB				
S2	112.25	117.75	111	118
S3	119.25	124.75	118	125
S4	126.25	131.75	125	132
S5	133.25	138.75	132	139
S6	140.25	145.75	139	146
S7	147.25	152.75	146	153
S8	154.25	159.75	153	160
S9	161.25	166.75	160	167
S10	168.25	173.75	167	174
Band III				
5	175.25	180.75	174	181
6	182.25	187.75	181	188
7	189.25	194.75	188	195
8	196.25	201.75	195	202
9	203.25	208.75	202	209
10	210.25	215.75	209	216
11	217.25	222.75	216	223
12	224.25	229.75	223	230
Upper Hyperband (OSB)				
S11	231.25	236.75	230	237
S12	238.25	243.75	237	244
S13	245.25	250.75	244	251
S14	252.25	257.75	251	258
S15	259.25	264.75	258	265
S16	266.25	271.75	265	272
S17	273.25	278.75	272	279
S18	280.25	285.75	279	286
S19	287.25	292.75	286	293
S20	294.25	299.75	293	300
Extended Hyperband (ESB)				
S21	303.25	308.75	302	310
S22	311.25	316.75	310	318
S23	319.25	324.75	318	326
S24	327.25	332.75	326	334
S25	335.25	340.75	334	342
S26	343.25	348.75	342	350
S27	351.25	356.75	350	358
S28	359.25	364.75	358	366
S29	367.25	372.75	366	374
S30	375.25	380.75	374	382
S31	383.25	388.75	382	390
S32	391.25	396.75	390	398
S33	399.25	404.75	398	406
S34	407.25	412.75	406	414
S35	415.25	420.75	414	422
S36	423.25	428.75	422	430
S37	431.25	436.75	430	438
S38	439.25	444.75	438	446

Channel Number	Vision Carrier MHz	Sound Carrier MHz	Upper Limit MHz	Lower Limit MHz
Band IV				
21	471.25	476.75	470	478
22	479.25	484.75	478	486
23	487.25	492.75	486	494
24	495.25	500.75	494	502
25	503.25	508.75	502	510
26	511.25	516.75	510	518
27	519.25	524.75	518	526
28	527.25	532.75	526	534
29	535.25	540.75	534	542
30	543.25	548.75	542	550
31	551.25	556.75	550	558
32	559.25	564.75	558	566
33	567.25	572.75	566	574
34	575.25	580.75	574	582
35	583.25	588.75	582	590
36	591.25	596.75	590	598
37	599.25	604.75	598	606
Band V				
38	607.25	612.75	606	614
39	615.25	620.75	614	622
40	623.25	628.75	622	630
41	631.25	636.75	630	638
42	639.25	644.75	638	646
43	647.25	652.75	646	654
44	655.25	660.75	654	662
45	663.25	668.75	662	670
46	671.25	676.75	670	678
47	679.25	684.75	678	686
48	687.25	692.75	686	694
49	695.25	700.75	694	702
50	703.25	708.75	702	710
51	711.25	716.75	710	718
52	719.25	724.75	718	726
53	727.25	732.75	726	734
54	735.25	740.75	734	742
55	743.25	748.75	742	750
56	751.25	756.75	750	758
57	759.25	764.75	758	766
58	767.25	772.75	766	774
59	775.25	780.75	774	782
60	783.25	788.75	782	790
61	791.25	796.75	790	798
62	799.25	804.75	798	806
63	807.25	812.75	806	814
64	815.25	820.75	814	822
65	823.25	828.75	822	830
66	831.25	836.75	830	838
67	839.25	844.75	838	846
68	847.25	852.75	846	854
69	855.25	860.75	854	862

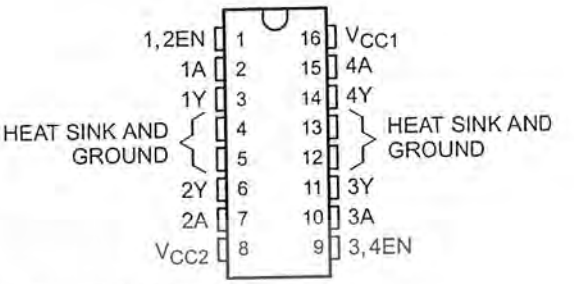
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

L293D
QUADRUPLE HALF-H DRIVER

SLRS008A – SEPTEMBER 1986 – REVISED MAY 1990

- 600-mA Output Current Capability Per Driver
- Pulsed Current 1.2-A Per Driver
- Output Clamp Diodes for Inductive Transient Suppression
- Wide Supply Voltage Range 4.5 V to 36 V
- Separate Input-Logic Supply
- Thermal Shutdown
- Internal ESD Protection
- High-Noise-Immunity Inputs
- Functional Replacement for SGS L293D

NE PACKAGE
(TOP VIEW)



description

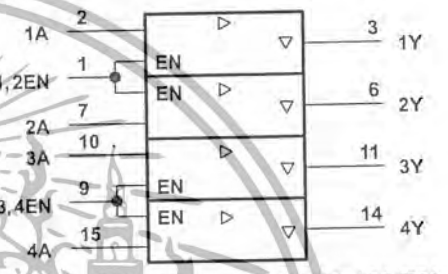
The L293D is a quadruple high-current half-H driver designed to provide bidirectional drive currents of up to 600-mA at voltages from 4.5 V to 36 V. It is designed to drive inductive loads such as relays, solenoids, dc and bipolar stepping motors, as well as other high-current/high-voltage loads in positive-supply applications.

All inputs are TTL-compatible. Each output is a complete totem-pole drive circuit with a Darlington transistor sink and a pseudo-Darlington source. Drivers are enabled in pairs with drivers 1 and 2 enabled by 1,2EN and drivers 3 and 4 enabled by 3,4EN. When an enable input is high, the associated drivers are enabled, and their outputs are active and in phase with their inputs. External high-speed output clamp diodes should be used for inductive transient suppression. When the enable input is low, those drivers are disabled, and their outputs are off and in a high-impedance state. With the proper data inputs, each pair of drivers form a full-H (or bridge) reversible drive suitable for solenoid or motor applications.

A V_{CC1} terminal, separate from V_{CC2}, is provided for the logic inputs to minimize device power dissipation.

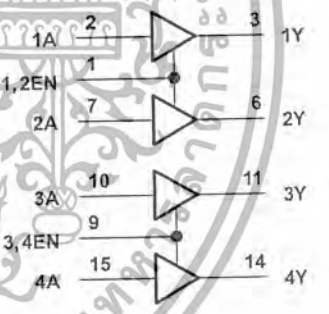
The L293D is designed for operation from 0°C to 70°C.

logic symbol†



† This symbol is in accordance with ANSI/IEEE Std 91-1984 and IEC Publication 617-12.

logic diagram



FUNCTION TABLE
(each driver)

INPUTS‡		OUTPUT
A	EN	Y
H	H	H
L	H	L
X	L	Z

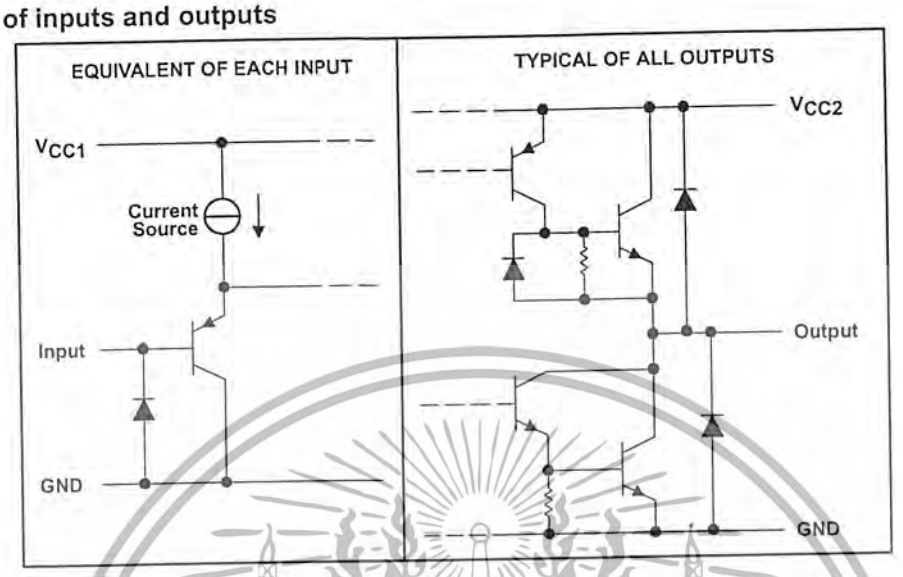
H = high-level, L = low level,
X = irrelevant, Z = high-impedance (off)
‡ In the thermal shutdown mode, the output is in the high-impedance state regardless of the input levels.

PRODUCTION DATA information is current as of publication date. Products conform to specifications per the terms of Texas Instruments standard warranty. Production processing does not necessarily include testing of all parameters.

Copyright © 1990, Texas Instruments Incorporated

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

schematics of inputs and outputs



absolute maximum ratings over operating free-air temperature range (unless otherwise noted)

Logic supply voltage range, V_{CC1} (see Note 1) . . . . . 36 V
Output supply voltage range, V_{CC2} . . . . . 36 V
Input voltage range, V_I . . . . . 7 V
Output voltage range, V_O . . . . . −3 V to V_{CC2} + 3 V
Peak output current (nonrepetitive, $t \leq 100\ \mu s$) . . . . . ±1.2 A
Continuous output current, I_O . . . . . ±600 mA
Continuous total dissipation at (or below) 25°C free-air temperature (see Notes 2 and 3) . . . . . 2075 mW
Continuous total dissipation at 80°C case temperature (see Note 3) . . . . . 5000 mW
Operating case or virtual junction temperature range, T_J . . . . . −40°C to 150°C
Storage temperature range, T_{stg} . . . . . −65°C to 150°C
Lead temperature 1,6 mm (1/16 inch) from case for 10 seconds . . . . . 260°C

NOTES: 1. All voltage values are with respect to the network ground terminal.
2. For operation above 25°C free-air temperature, derate linearly at the rate of 16.6 mW/°C.
3. For operation above 25°C case temperature, derate linearly at the rate of 71.4 mW/°C. Due to variations in individual device electrical characteristics and thermal resistance, the built-in thermal overload protection may be activated at power levels slightly above or below the rated dissipation.

recommended operating conditions

		MIN	MAX	UNIT
Logic supply voltage, V_{CC1}		4.5	7	V
Output supply voltage, V_{CC2}		V_{CC1}	36	V
High-level input voltage, V_{IH}	$V_{CC1} \leq 7$ V	2.3	V_{CC1}	V
	$V_{CC1} \geq 7$ V	2.3	7	
Low-level input voltage, V_{IL}		−0.3†	1.5	V
Operating free-air temperature, T_A		0	70	°C

† The algebraic convention, in which the least positive (most negative) value is designated minimum, is used in this data sheet for logic voltage levels.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

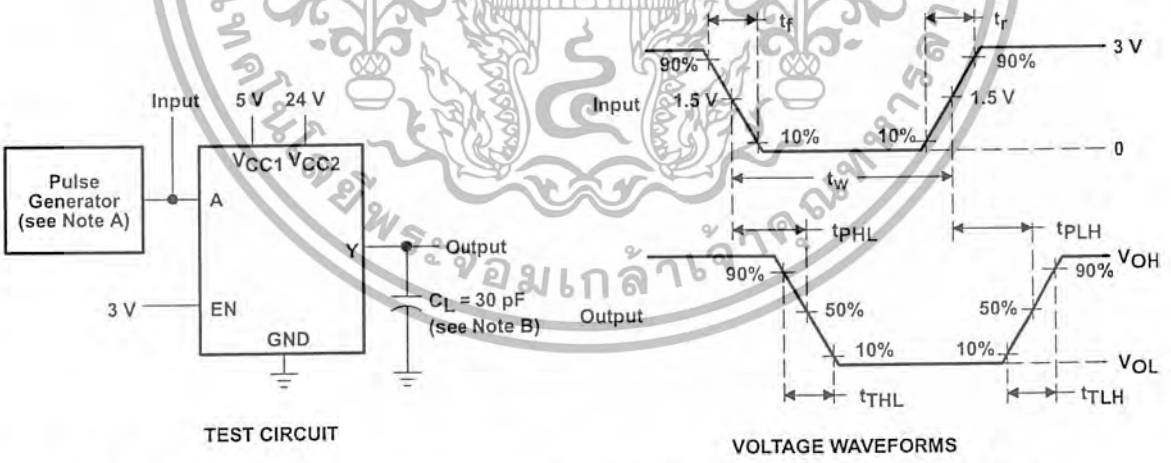
electrical characteristics, V_{CC1} = 5 V, V_{CC2} = 24 V, T_A = 25°C

PARAMETER			TEST CONDITIONS		MIN	TYP	MAX	UNIT
V_{OH}	High-level output voltage		I_{OH} = –0.6 A		V_{CC2}–1.8	V_{CC2}–1.4		V
V_{OL}	Low-level output voltage		I_{OL} = 0.6 A			1.2	1.8	V
V_{OKH}	High-level output clamp voltage		I_{OK} = –0.6 A			V_{CC2}+1.3		V
V_{OKL}	Low-level output clamp voltage		I_{OK} = –0.6 A			1.3		V
I_{IH}	High-level input current	A	V_I = 7 V			0.2	100	µA
		EN				0.2	±10	
I_{IL}	Low-level input current	A	V_I = 0			–3	–10	µA
		EN				–2	–100	
I_{CC1}	Logic supply current		I_O = 0	All outputs at high level		13	22	mA
				All outputs at low level		35	60	
				All outputs at high impedance		8	24	
I_{CC2}	Output supply current		I_O = 0	All outputs at high level		14	24	mA
				All outputs at low level		2	6	
				All outputs at high impedance		2	4	

switching characteristics, V_{CC1} = 5 V, V_{CC2} = 24 V, T_A = 25°C

PARAMETER		TEST CONDITIONS	MIN	TYP	MAX	UNIT
t_{PLH}	Propagation delay time, low-to-high-level output from A input	C_L = 30 pF, See Figure 1		800		ns
t_{PHL}	Propagation delay time, high-to-low-level output from A input			400		ns
t_{TLH}	Transition time, low-to-high-level output			300		ns
t_{THL}	Transition time, high-to-low-level output			300		ns

PARAMETER MEASUREMENT INFORMATION



NOTES: A. The pulse generator has the following characteristics: $t_r \le 10$ ns, $t_f \le 10$ ns, t_w = 10 µs, PRR = 5 kHz, Z_O = 50 Ω.
B. C_L includes probe and jig capacitance.

Figure 1. Test Circuit and Voltage Waveforms

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

APPLICATION INFORMATION

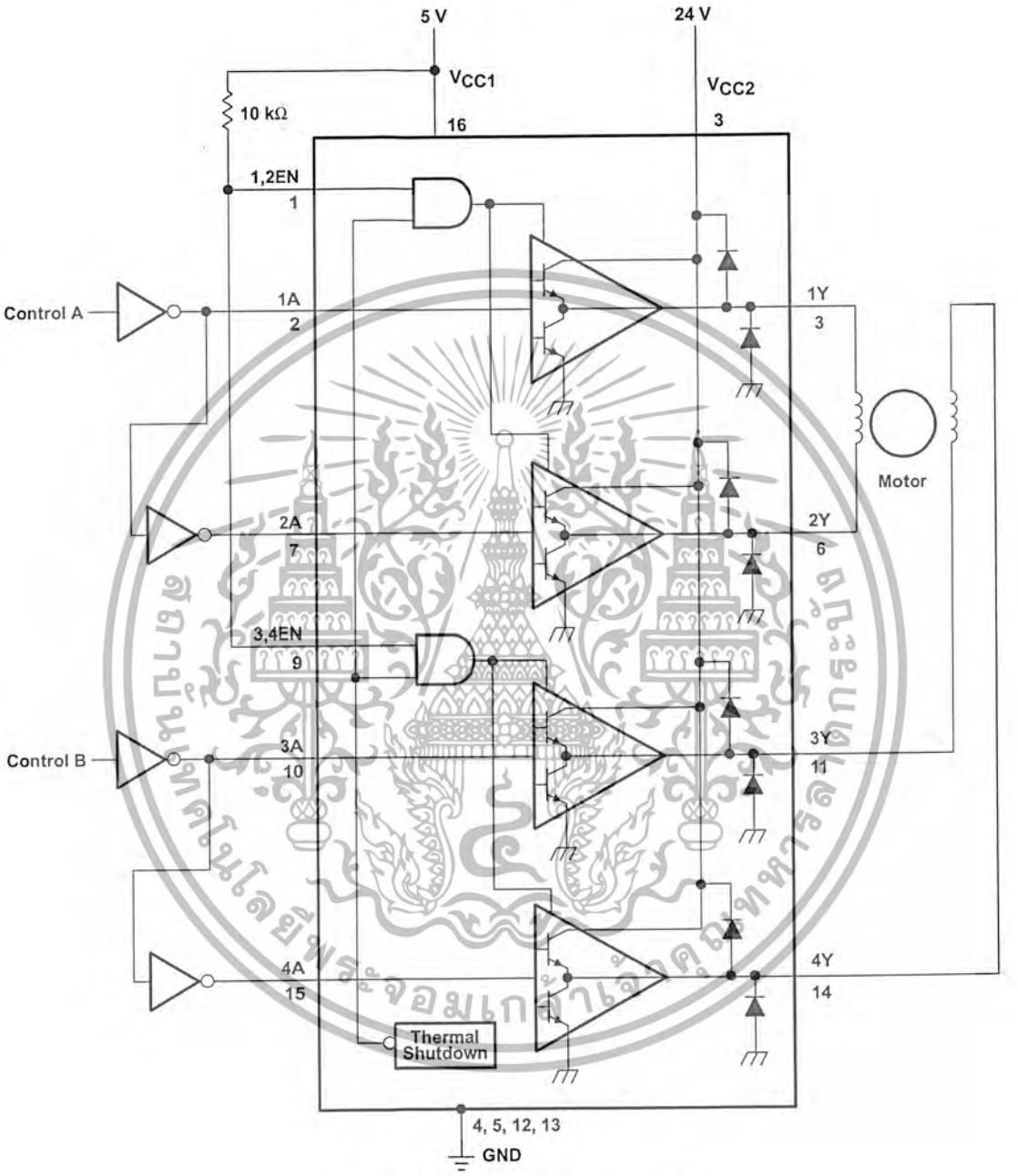


Figure 2. Two-Phase Motor Driver

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Features

- PT8A977P/977W works as encoder and PT8A978P/978LW works as decoder
- Five pins for five control functions
- Operating power-supply voltage: 2.5V to 5.0V (977P, 977W, 978P), 2.0V to 5.0V (978LW)
- Auto-power-off if no press on any button in 8s or continuously press on any button over 4 minues
- Press on any button as wake up (977P, 977W)
- Manual-power-off with OFF button
- One output pin used for external power control (977P, 977W)
- On-chip oscillator with an external resistor
- On-chip reversing amplifiers (978P, 978LW)
- Low operating current
- Few external components needed
- Package: 14-pin DIP, 14-pin SOIC, 16-pin DIP and 16-pin SOIC

Ordering Information

Part No.	Package
PT8A977P	14-pin DIP
PT8A977W	14-pin SOIC
PT8A978P	16-pin DIP
PT8A978LW	16-pin SOIC

General Description

The PTA977P (or 977W) and PT8A978P (or 978LW) provide a complete control functions to the remote-controlled toy. The PT8A977P/977W has five input pins corresponding with the five function buttons i.e, forward, backward, rightward, leftward and turbo. The encoding circuit in the PT8A977P/977W sends digital codes to the two output pins SO and SC. The digital codes correspond to the definite function buttons or their combinations. The SO and SC outputs are used in wireless and infra-red applications respectively.

The PT8A978P (or 978LW) has five output pins corresponding with the five actions. The received signals are amplified by the three-stage amplifier, and then the appropriate amplified signals are sampled, fault-tolerantly checked and decoded to control the actions of the remote-controlled toy.

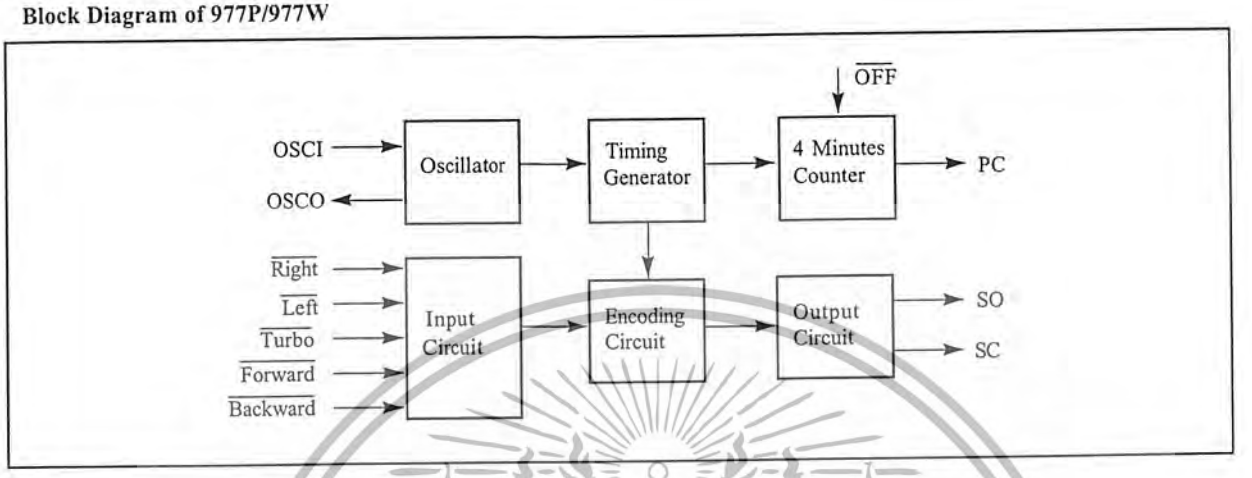
There is an internal oscillator in the PT8A977P/977W and 978P/978LW respectively. By adding an external resistor conveniently, the oscillator will be constructed. The oscillator frequency can be adjusted by the external resistor. The relative error between the frequencies of the two on-chip oscillators in the PT8A977P/977W and PT8A978P/978LW must be less than ±25%.

The auto-power-off function is achieved by an internal counter. The PC output is used to control on/off state of the external power supply. Pressing OFF button can also shut down the power supply. Press on any function button will wake up the chip promptly.

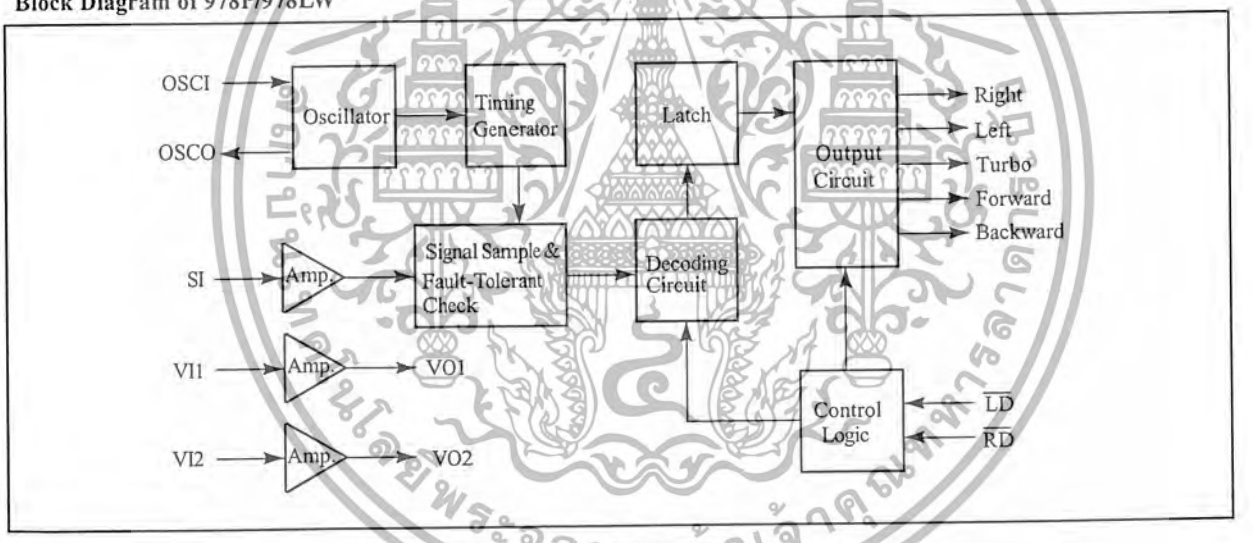
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Block Diagram

Block Diagram of 977P/977W

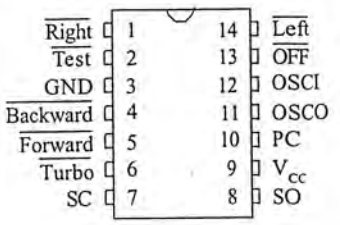


Block Diagram of 978P/978LW

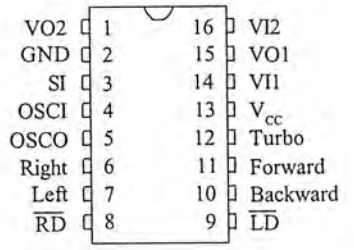


Pin Configuration

PT8A977P/977W
14-Pin DIP/14-Pin SOIC



PT8A978P/978LW
16-Pin DIP/16-pin SOIC



Top View

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Pin Description

Pin Description of 977P/977W

Pin No.	Pin Name	Description
1	$\overline{\text{Right}}$	With Pull-up resistor, rightward function selected if this pin connected to GND.
2	$\overline{\text{Test}}$	With Pull-up resistor, this pin is used for testing mode.
3	GND	Negative power supply
4	$\overline{\text{Backward}}$	With Pull-up resistor, backward function selected if this pin connected to GND.
5	$\overline{\text{Forward}}$	With Pull-up resistor, forward function selected if this pin connected to GND.
6	$\overline{\text{Turbo}}$	With Pull-up resistor, turbo function selected if this pin connected to GND.
7	SC	Output pin of the encoding signal with carrier frequency
8	SO	Output pin of the encoding signal without carrier frequency
9	V_{cc}	Positive power supply
10	PC	Power control output pin
11	OSCO	Oscillator output pin
12	OSCI	Oscillator input pin
13	$\overline{\text{OFF}}$	With Pull-up resistor, this pin is used to shut down the external power supply.
14	$\overline{\text{Left}}$	With Pull-up resistor, leftward function selected if this pin connected to GND.

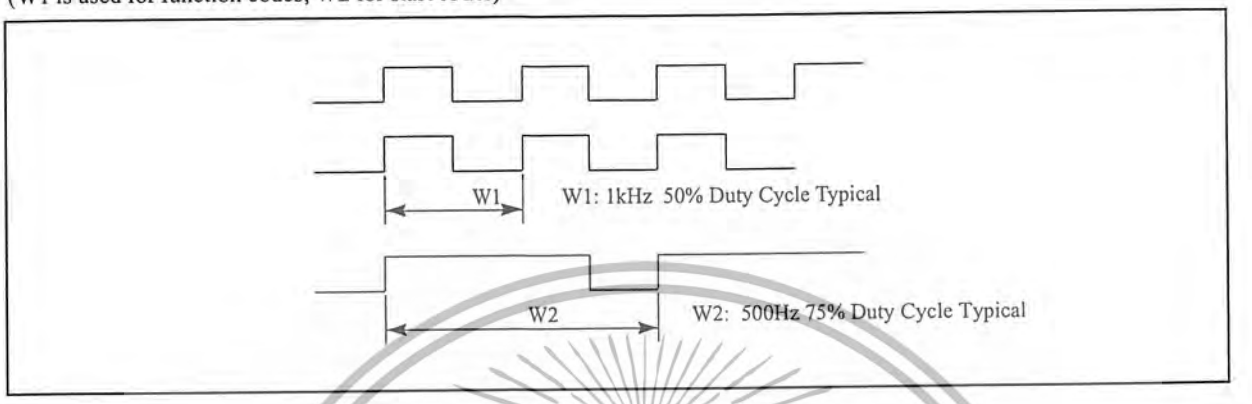
Pin Description of 978P/978LW

Pin No.	Pin Name	Description
1	VO2	Output pin for the amplifier 2
2	GND	Negative power supply
3	SI	Input pin of the encoding signal
4	OSCI	Oscillator input pin
5	OSCO	Oscillator output pin
6	Right	Rightward output pin
7	Left	Leftward output pin
8	$\overline{\text{RD}}$	With Pull-up resistor, rightward function disabled if this pin connected to GND.
9	$\overline{\text{LD}}$	With Pull-up resistor, leftward function disabled if this pin connected to GND.
10	Backward	Backward output pin
11	Forward	Forward output pin
12	Turbo	Turbo output pin
13	V_{cc}	Positive power supply
14	VI1	Input pin for the amplifier 1
15	VO1	Output pin for the amplifier 1
16	VI2	Input pin for the amplifier 2

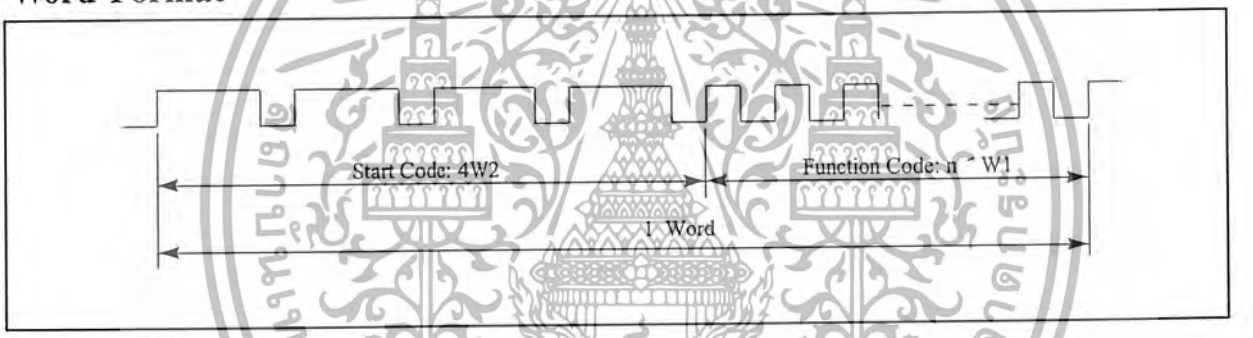
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Code Format

(W1 is used for function codes, W2 for start codes)

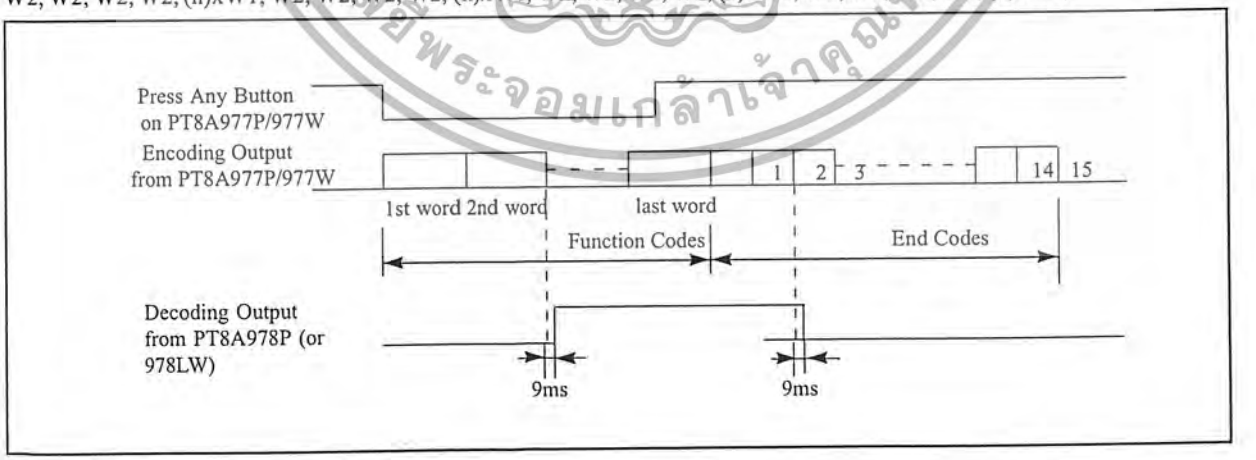


Word Format



Encoding and Decoding Timing

W2, W2, W2, W2, (n)xW1, W2, W2, W2, W2, (n)xW1, W2, W2, W2, W2, (n)xW1, W2, W2, W2, W2, (n)xW1,

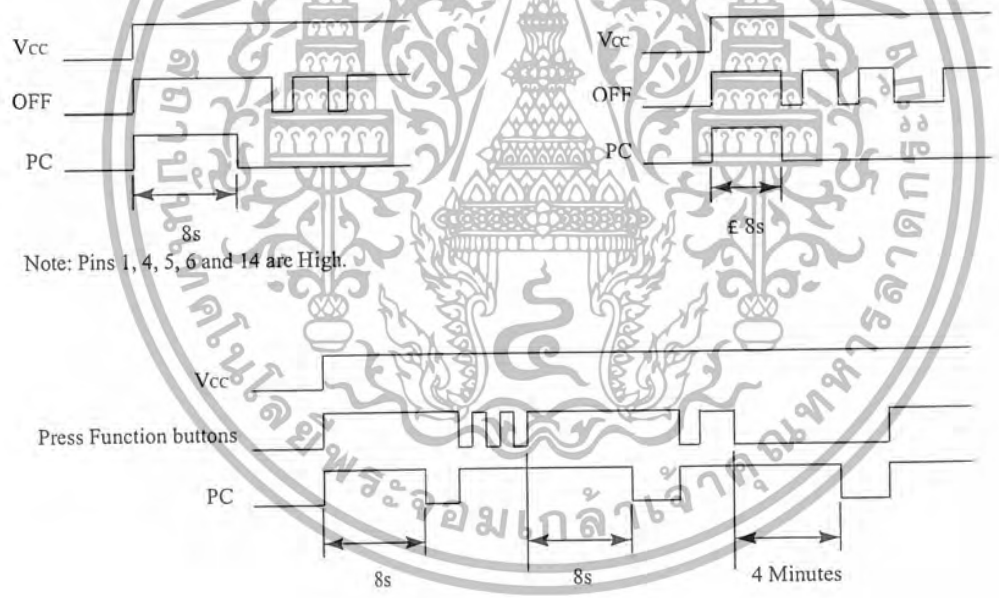


เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Encoding Input and Decoding Result

Number of Function Codes (n) W1	Decoding Results
4	End Code
10	Forward (Pulse)
16	Forward (High level)
22	Turbo
28	Forward (High level) & Left
34	Forward (High level) & Right
40	Backward
46	Backward & Right
52	Backward & Left
58	Left
64	Right

PC Output Timing of 977P/977W



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Maximum Ratings

(Above which the useful life may be impaired. For user guidelines, not tested)

Storage Temperature	-25°C to +85°C
Ambient Temperature with Power Applied	-10°C to +40°C
Supply Voltage to Ground Potential (Inputs & V_{CC} Only)	-0.5 to +6.0V
Supply Voltage to Ground Potential (Outputs & D/O Only)	-0.5 to +6.0V
DC Input Voltage	-0.5 to +6.0V
DC Output Current	20mA
Power Dissipation	500mW

Note:
Stresses greater than those listed under MAXIMUM RATINGS may cause permanent damage to the device. This is a stress rating only and functional operation of the device at these or any other conditions above those indicated in the operational sections of this specification is not implied. Exposure to absolute maximum rating conditions for extended periods may affect reliability.

DC Electrical Charateristics

DC Electrical Charateristics of 977P/977W

Parameters	Description		Test Condition	Min.	Type	Max.	Units
Vcc	Operating Voltage			2.5	4.0	5.0	V
Icc	Supply Current		Output unloaded			100	uA
Istb	Stand-by Current		OFF State			5	uA
Vil	Input Low Voltage		Guaranteed Logic LOW level			0.8	V
Vih	Input High Voltage		Guaranteed Logic HIGH level	3.0			V
Iil	Input Low Current	Pin 1, 4, 5, 6, 13, 14	V_{IL} = 0V, ON state			-60	uA
Iih	Input High Current	Pin 1, 4, 5, 6, 13, 14	V_{IH} = 4V, ON state			10	uA
Ii	Input Current	Pin 12	V_{IH} = 0 ~ 4V, ON state			±10	uA
Iol	Output Low Current		Vout = 0.5 V	150			uA
Ioh	Output High Current	Pin 7, 8, 10	Vout = 3.5 V	-1.0			mA
		Pin 11	Vout = 3.5 V	-200		- 800	uA

Note: Over the Operating Rating, 0°C ≤ T_A ≤ 70°C, V_{CC} = 4V

DC Electrical Charateristics of 978P/978LW

Parameters	Description		Test Condition	Min.	Type	Max.	Units
Vcc	Operating Voltage - 978P			2.5	4.0	5.0	V
	Operating Voltage - 978LW			2.0		5.0	V
Icc	Supply Current		Output unloaded			1	mA
Istb	Stand-by Current		OFF State			10	uA
Vil	Input Low Voltage		Guaranteed Logic LOW level			0.8	V
Vih	Input High Voltage		Guaranteed Logic HIGH level	3.0			V
Iil	Input Low Current	Pin 3, 8, 9	V_{IL} = 0V, ON state			-60	uA
Iih	Input High Current	Pin 3	V_{IH} = 4V, ON state			60	uA
	Input High Current	Pin 8, 9	V_{IH} = 4V, ON state			10	uA
Ii	Input Current	Pin 14, 16	V_{IH} = 0 ~ 4V, ON state			±10	uA
Iol	Output Low Current	Pin 1, 5, 15	Vout = 0.5 V	200		850	uA
		Pin 6, 7, 10, 11, 12	Vout = 0.5 V	2			mA
Ioh	Output High Current	Pin 1, 5, 15	Vout = 3.5 V	-200		- 850	uA
		Pin 6, 7, 10, 11, 12	Vout = 3.5 V	-500			uA

Note: Over the Operating Rating, 0°C ≤ T_A ≤ 70°C, V_{CC} = 4V

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

AC Electrical Characteristics

AC Electrical Characteristics of 977P/977W

Parameters	Description	Test Condition	Min.	Type	Max.	Units
fosc	Oscillator Frequency *	T_A= 25°C, R = 200 kW	102	128	154	kHz
f_{max}-f_{min}	Oscillator Frequency Fluctuation - 977P/977W	T_A=25°C, V_{CC}= 2.5 ~ 5V			15	kHz
tFUN	Cycle Time of Function Code	fosc = 102 to 154 kHz	0.8	1	1.2	ms
tSTA	Cycle Time of Start Code	fosc = 102 to 154 kHz	1.6	2	2.4	ms
fcsc	Carrier Frequency of SC Pin	fosc = 102 to 154 kHz	51	64	77	kHz
tOFF	Time of Auto-Power-Off **	Pins 1, 4, 5, 6 and 14 are High.	6.4	8	9.6	s
		Any of pins 1, 4, 5, 6 or 14 is Low.	3.2	4	4.8	min.

Note: Over the Operating Rating, 0°C £ T_A £ 70°C, V_{CC} = 4V

* The relative error between the frequencies of the two on-chip oscillators in the PT8A977P (or 977W) and PT8A978P (or 978LW) must be less than ±25%.

** When adjust the external oscillator resistor, the auto-power-off time will vary relevantly.

AC Electrical Characteristics of 978P/978LW

Parameters	Description	Test Condition	Min.	Type	Max.	Units
fosc	Oscillator Frequency *	T_A= 25°C, R = 200 kW	102	128	154	kHz
f_{max}-f_{min}	Oscillator Frequency Fluctuation - 978P	T_A=25°C, V_{CC}=2.5 ~ 5V			15	kHz
	Oscillator Frequency Fluctuation - 978LW	T_A=25°C, V_{CC}=2.0 ~ 5V			30	kHz
VSI	SI Pin Receive Sensitivity (V_{PP})	Guaranteed Effective Decoding	300			mV
tFUN	Cycle Time of Function Code	fosc = 128 kHz	0.75	1	1.25	ms
tSTA	Cycle Time of Start Code	fosc = 128 kHz	1.5	2	2.5	ms

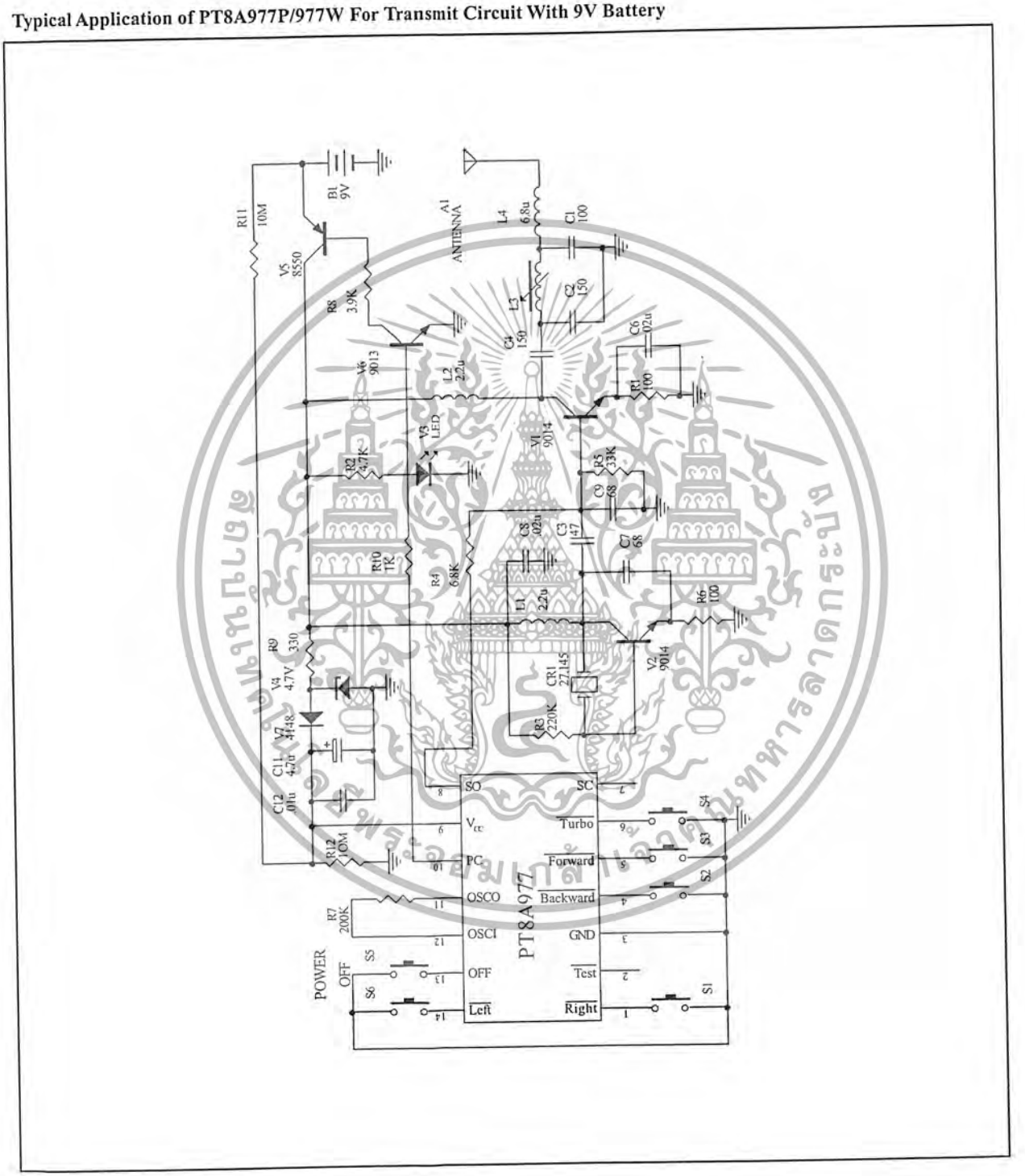
Note: Over the Operating Rating, 0°C £ T_A £ 70°C, V_{CC} = 4V

* The relative error between the frequencies of the two on-chip oscillators in the PT8A977P (or 977W) and PT8A978P (or 978LW) must be less than ±25%.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

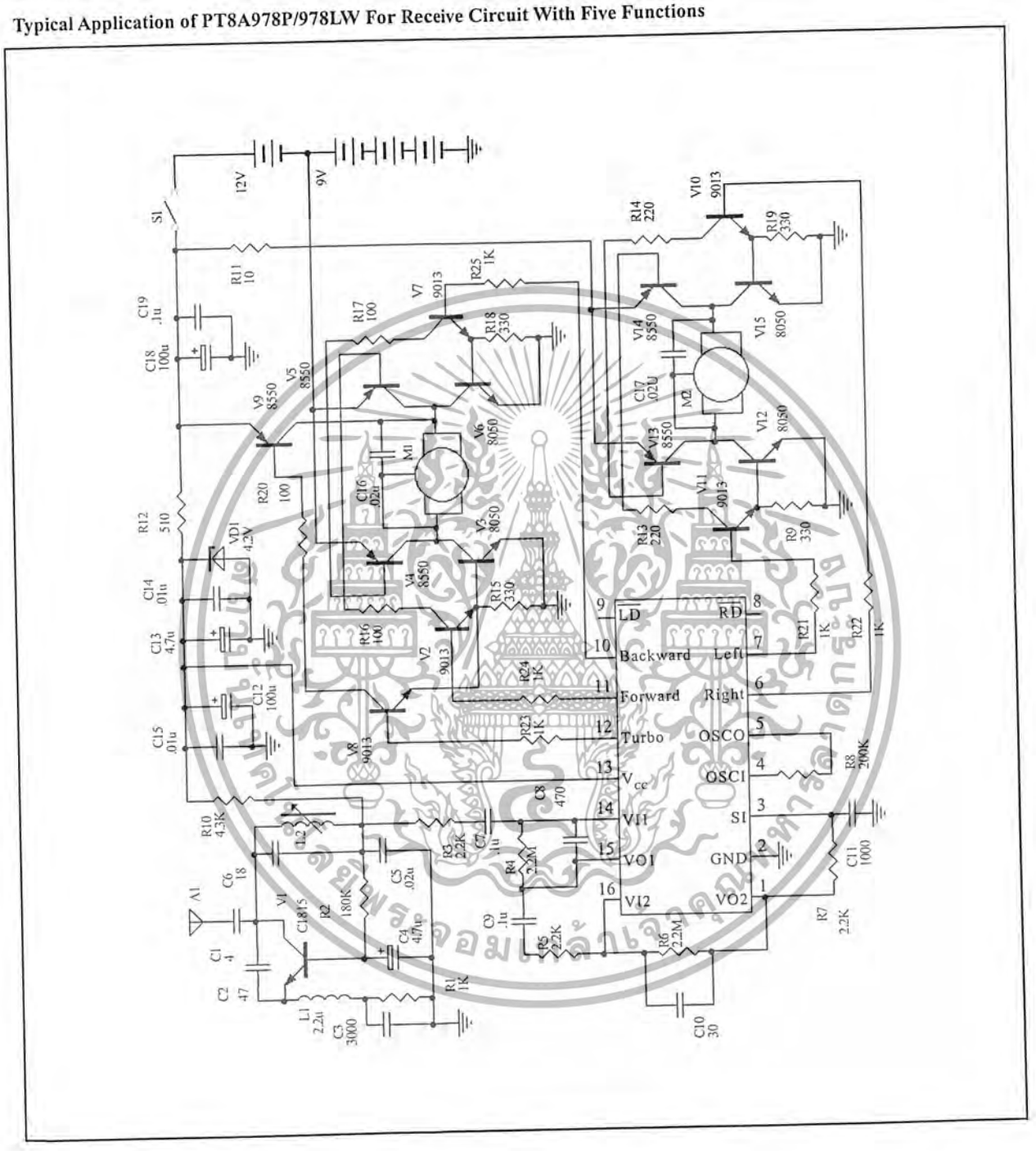
Application Circuits

Typical Application of PT8A977P/977W For Transmit Circuit With 9V Battery



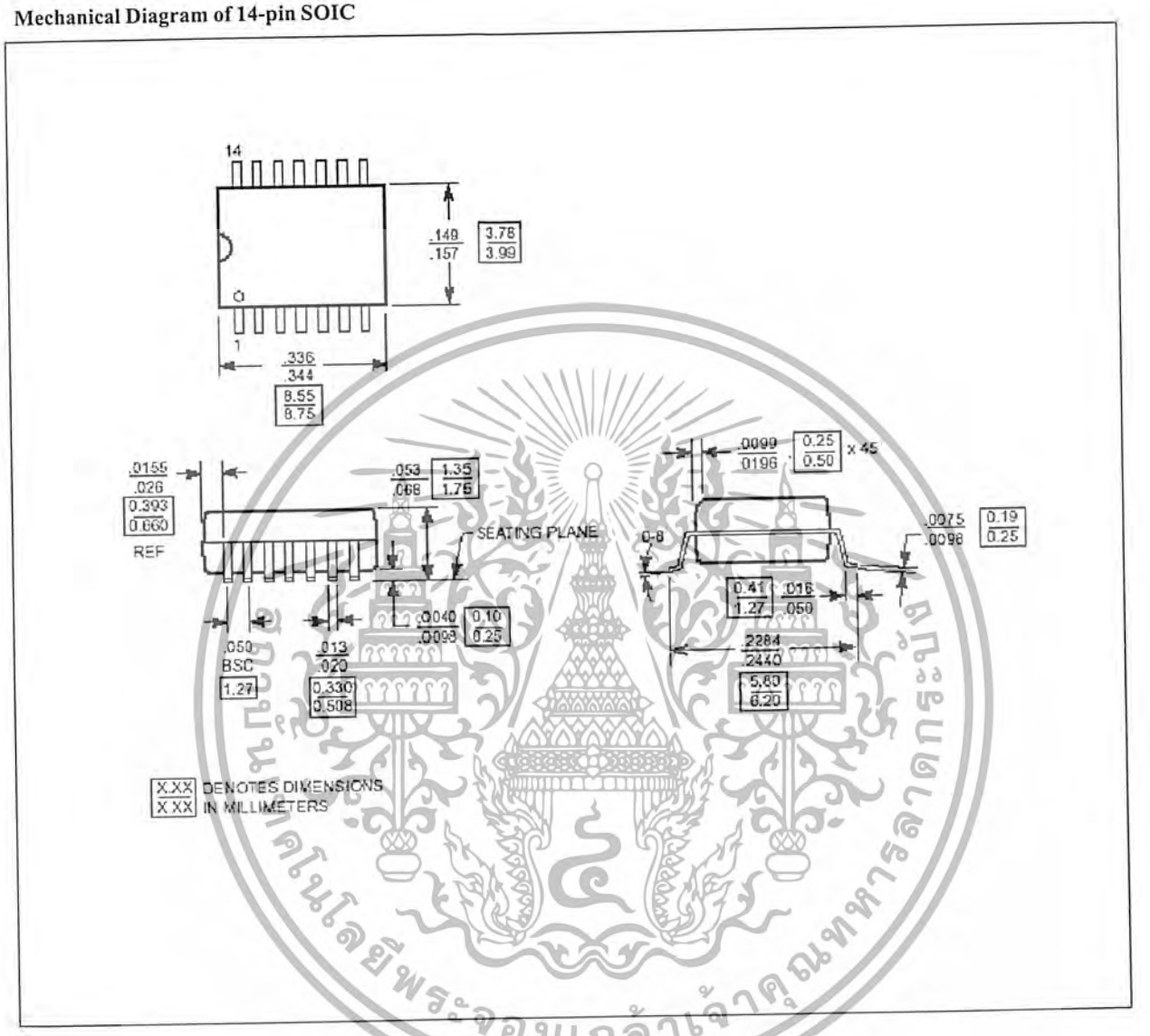
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Typical Application of PT8A978P/978LW For Receive Circuit With Five Functions



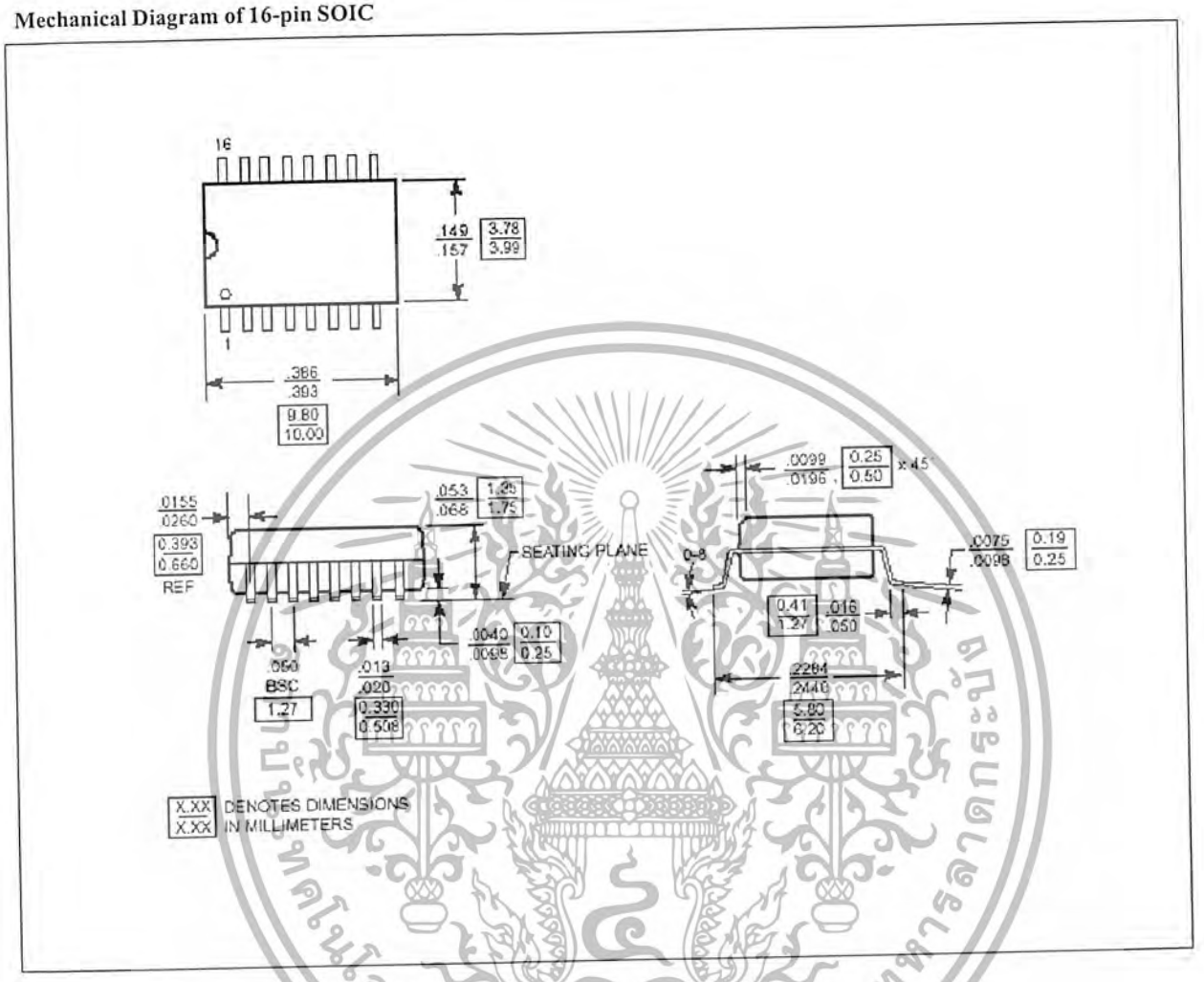
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Mechanical Diagram of 14-pin SOIC



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Mechanical Diagram of 16-pin SOIC



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Octal bus transceiver; 3-state

74HC/HCT245

FEATURES

- Octal bidirectional bus interface
- Non-inverting 3-state outputs
- Output capability: bus driver
- I_{CC} category: MSI

GENERAL DESCRIPTIONS

The 74HC/HCT245 are high-speed Si-gate CMOS devices and are pin compatible with low power Schottky TTL (LSTTL). They are specified in compliance with JEDEC standard no. 7A.

The 74HC/HCT245 are octal transceivers featuring non-inverting 3-state bus compatible outputs in both send and receive directions. The "245" features an output enable ($\overline{OE}$) input for easy cascading and a send/receive (DIR) for direction control. $\overline{OE}$ controls the outputs so that the buses are effectively isolated. The "245" is similar to the "640" but has true (non-inverting) outputs.

QUICK REFERENCE DATA

GND = 0 V; T_{amb} = 25 °C; $t_r = t_f$ = 6 ns

SYMBOL	PARAMETER	CONDITIONS	TYPICAL		UNIT
			HC	HCT	
t_{PHL}/t_{PLH}	propagation delay A_n to B_n; B_n to A_n	C_L = 15 pF; V_{CC} = 5 V	7	10	ns
C_I	input capacitance		3.5	3.5	pF
$C_{I/O}$	input/output capacitance		10	10	pF
C_{PD}	power dissipation capacitance per transceiver	notes 1 and 2	30	30	pF

Notes

1. C_{PD} is used to determine the dynamic power dissipation (P_D in µW):
   $P_D = C_{PD} \times V_{CC}^2 \times f_i + \sum (C_L \times V_{CC}^2 \times f_o)$ where:
   f_i = input frequency in MHz
   f_o = output frequency in MHz
   $\sum (C_L \times V_{CC}^2 \times f_o)$ = sum of outputs
   C_L = output load capacitance in pF
   V_{CC} = supply voltage in V
2. For HC the condition is V_I = GND to V_{CC}
   For HCT the condition is V_I = GND to V_{CC} − 1.5 V

ORDERING INFORMATION

See *"74HC/HCT/HCU/HCMOS Logic Package Information"*.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

PIN DESCRIPTION

PIN NO.	SYMBOL	NAME AND FUNCTION
1	DIR	direction control
2, 3, 4, 5, 6, 7, 8, 9	A_0 to A_7	data inputs/outputs
10	GND	ground (0 V)
18, 17, 16, 15, 14, 13, 12, 11	B_0 to B_7	data inputs/outputs
19	$\overline{OE}$	output enable input (active LOW)
20	V_{CC}	positive supply voltage

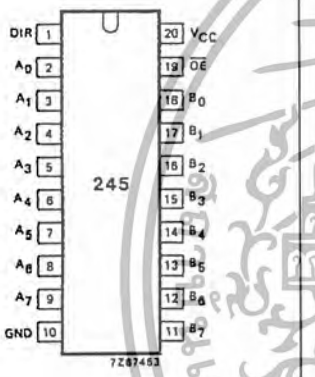


Fig.1 Pin configuration.

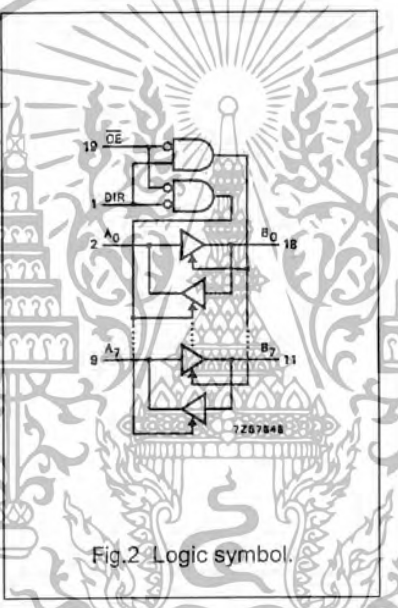


Fig.2 Logic symbol.

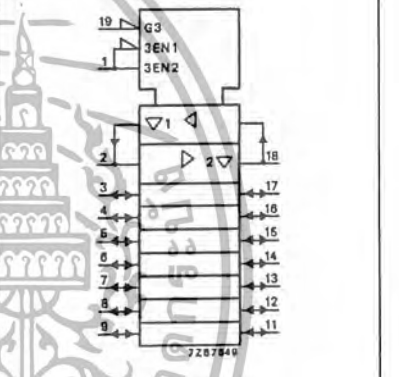


Fig.3 IEC logic symbol.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

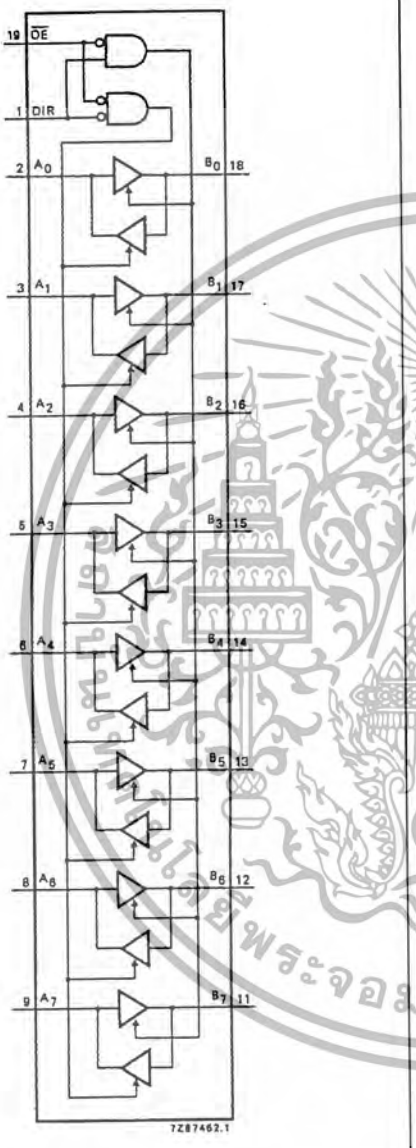


Fig.4 Functional diagram.

FUNCTION TABLE

INPUTS		INPUTS/OUTPUTS	
$\overline{OE}$	DIR	A_n	B_n
L	L	A = B	inputs
L	H	inputs	B = A
H	X	Z	Z

Notes

1. H = HIGH voltage level
   L = LOW voltage level
   X = don't care
   Z = high impedance OFF-state

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

DC CHARACTERISTICS FOR 74HC

For the DC characteristics see *"74HC/HCT/HCU/HCMOS Logic Family Specifications"*.

Output capability: bus driver
I_{CC} category: MSI

AC CHARACTERISTICS FOR 74HC

GND = 0 V; $t_r = t_f$ = 6 ns; C_L = 50 pF

SYMBOL	PARAMETER	T_{amb} (°C)							UNIT	TEST CONDITIONS	
		74HC								V_{CC} (V)	WAVEFORMS
		+25			−40 to +85		−40 to +125				
		min.	typ.	max.	min.	max.	min.	max.			
t_{PHL}/ t_{PLH}	propagation delay A_n to B_n; B_n to A_n		25 9 7	90 18 15		115 23 20		135 27 23	ns	2.0 4.5 6.0	Fig.5
t_{PZH}/ t_{PZL}	3-state output enable time $\overline{OE}$ to A_n; $\overline{OE}$ to B_n signalname DIR		30 11 9	150 30 26		190 38 33		225 45 38	ns	2.0 4.5 6.0	Fig.6
t_{PHZ}/ t_{PLZ}	3-state output disable time $\overline{OE}$ to A_n; $\overline{OE}$ to B_n signalname DIR		41 15 12	150 30 26		190 38 33		225 45 38	ns	2.0 4.5 6.0	Fig.6
t_{THL}/ t_{TLH}	output transition time		14 5 4	60 12 10		75 15 13		90 18 15	ns	2.0 4.5 6.0	Fig.5

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

DC CHARACTERISTICS FOR 74HCT

For the DC characteristics see *"74HC/HCT/HCU/HCMOS Logic Family Specifications"*.

Output capability: bus driver
I_{CC} category: MSI

Note to HCT types

The value of additional quiescent supply current (ΔI_{CC}) for a unit load of 1 is given in the family specifications. To determine ΔI_{CC} per input, multiply this value by the unit load coefficient shown in the table below.

INPUT	UNIT LOAD COEFFICIENT
A_n	0.40
B_n	0.40
$\overline{OE}$	1.50
DIR	0.90

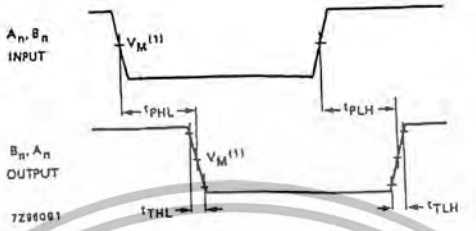
AC CHARACTERISTICS FOR 74HCT

GND = 0 V; $t_r = t_f$ = 6 ns; C_L = 50 pF

SYMBOL	PARAMETER	T_{amb} (°C)							UNIT	TEST CONDITIONS	
		74HCT								V_{CC} (V)	WAVEFORMS
		+25			−40 to +85		−40 to +125				
		min.	typ.	max.	min.	max.	min.	max.			
t_{PHL}/ t_{PLH}	propagation delay A_n to B_n; B_n to A_n		12	22		28		33	ns	4.5	Fig.5
t_{PZH}/ t_{PZL}	3-state output enable time $\overline{OE}$ to A_n; $\overline{OE}$ to B_n signalname DIR		16	30		38		45	ns	4.5	Fig.6
t_{PHZ}/ t_{PLZ}	3-state output disable time $\overline{OE}$ to A_n; $\overline{OE}$ to B_n signalname DIR		16	30		38		45	ns	4.5	Fig.6
t_{THL}/ t_{TLH}	output transition time		5	12		15		18	ns	4.5	Fig.5

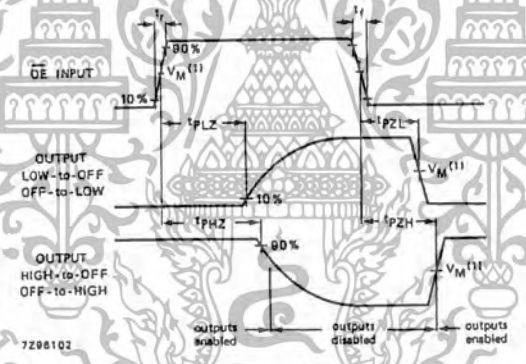
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

AC WAVEFORMS



(1) HC : V_M = 50%; V_I = GND to V_{CC}.
HCT : V_M = 1.3 V; V_I = GND to 3 V.

Fig.5 Waveforms showing the input (A_n, B_n) to output (B_n, A_n) propagation delays and the output transition times.



(1) HC : V_M = 50%; V_I = GND to V_{CC}.
HCT : V_M = 1.3 V; V_I = GND to 3 V.

Fig.6 Waveforms showing the 3-state enable and disable times.

PACKAGE OUTLINES

See *"74HC/HCT/HCU/HCMOS Logic Package Outlines"*.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Optocoupler with Phototransistor Output

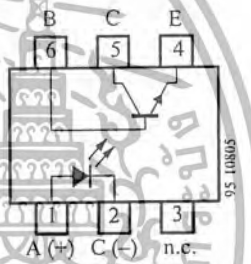
Description

The 4N25/ 4N26/ 4N27/ 4N28 consist of a phototransistor optically coupled to a gallium arsenide infrared-emitting diode in a 6-lead plastic dual-inline package.
The elements are mounted on one leadframe using a **coplanar technique**, providing a fixed distance between input and output for highest safety requirements.

95 10532

Applications

Galvanically separated circuits for general purposes



Features

- Isolation test voltage (RMS) 3.75 kV
- Underwriters Laboratory (UL) 1577 recognized, file number E-76222
- Low coupling capacity of typical 1 pF
- Current Transfer Ratio (CTR) of typical 100%
- Low temperature coefficient of CTR

Order Instruction

Ordering Code	CTR Ranking	Remarks
4N25/ 4N26	>20%	
4N27/ 4N28	>10%	

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Absolute Maximum Ratings

Input (Emitter)

Parameter	Test Conditions	Symbol	Value	Unit
Reverse voltage		V_R	5	V
Forward current		I_F	60	mA
Forward surge current	$t_p \leq 10\ \mu s$	I_{FSM}	3	A
Power dissipation	$T_{amb} \leq 25°C$	P_V	100	mW
Junction temperature		T_j	125	°C

Output (Detector)

Parameter	Test Conditions	Symbol	Value	Unit
Collector base voltage		V_{CBO}	70	V
Collector emitter voltage		V_{CEO}	30	V
Emitter collector voltage		V_{ECO}	7	V
Collector current		I_C	50	mA
Peak collector current	$t_p/T = 0.5$, $t_p \leq 10$ ms	I_{CM}	100	mA
Power dissipation	$T_{amb} \leq 25°C$	P_V	150	mW
Junction temperature		T_j	125	°C

Coupler

Parameter	Test Conditions	Symbol	Value	Unit
Isolation test voltage (RMS)		V_{IO} [1]	3.75	kV
Total power dissipation	$T_{amb} \leq 25°C$	P_{tot}	250	mW
Ambient temperature range		T_{amb}	–55 to +100	°C
Storage temperature range		T_{stg}	–55 to +125	°C
Soldering temperature	2 mm from case, $t \leq 10$ s	T_{sd}	260	°C

[1] Related to standard climate 23/50 DIN 50014

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Electrical Characteristics (T_{amb} = 25°C)

Input (Emitter)

Parameter	Test Conditions	Symbol	Min.	Typ.	Max.	Unit
Forward voltage	I_F = 50 mA	V_F		1.25	1.5	V
Junction capacitance	V_R = 0, f = 1 MHz	C_j		50		pF

Output (Detector)

Parameter	Test Conditions	Symbol	Min.	Typ.	Max.	Unit
Collector base voltage	I_C = 100 µA	V_{CBO}	70			V
Collector emitter voltage	I_E = 1 mA	V_{CEO}	30			V
Emitter collector voltage	I_E = 100 µA	V_{ECO}	7			V
Collector dark current	V_{CB} = 10 V	I_{CBO}		0.1	20	nA
Collector dark current	V_{CE} = 10 V	I_{CEO}		3.5	50	nA

Coupler

Parameter	Test Conditions	Symbol	Min.	Typ.	Max.	Unit
Isolation test voltage (RMS)	f = 50 Hz, t = 2 s	V_{IO} [1]	3.75			kV
Isolation resistance	V_{IO} = 1 kV, 40% relative humidity	R_{IO} [1]		10^{12}		Ω
Collector emitter saturation voltage	I_F = 50 mA, I_C = 2 mA	V_{CEsat}			0.5	V
Cut-off frequency	V_{CE} = 5 V, I_F = 10 mA, R_L = 100 Ω	f_c		110		kHz
Coupling capacitance	f = 1 MHz	C_k		1		pF

[1] Related to standard climate 23/50 DIN 50014

Current Transfer Ratio (CTR)

Parameter	Test Conditions	Type	Symbol	Min.	Typ.	Max.	Unit
I_C/I_F	V_{CE} = 10 V, I_F = 10 mA	4N25, 4N26	CTR	0.2	1		
		4N27, 4N28	CTR	0.1	1		

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Switching Characteristics

Parameter	Test Conditions	Symbol	Typ.	Unit
Turn-on time	V_{CE} = 10 V, I_C = 10 mA, R_L = 100 Ω (see figure 1)	t_{on}	4.0	µs
Turn-off time		t_{off}	3.0	µs
Turn-on time	V_{CE} = 5 V, I_F = 10 mA, R_L = 1 kΩ (see figure 2)	t_{on}	9.0	µs
Turn-off time		t_{off}	18.0	µs

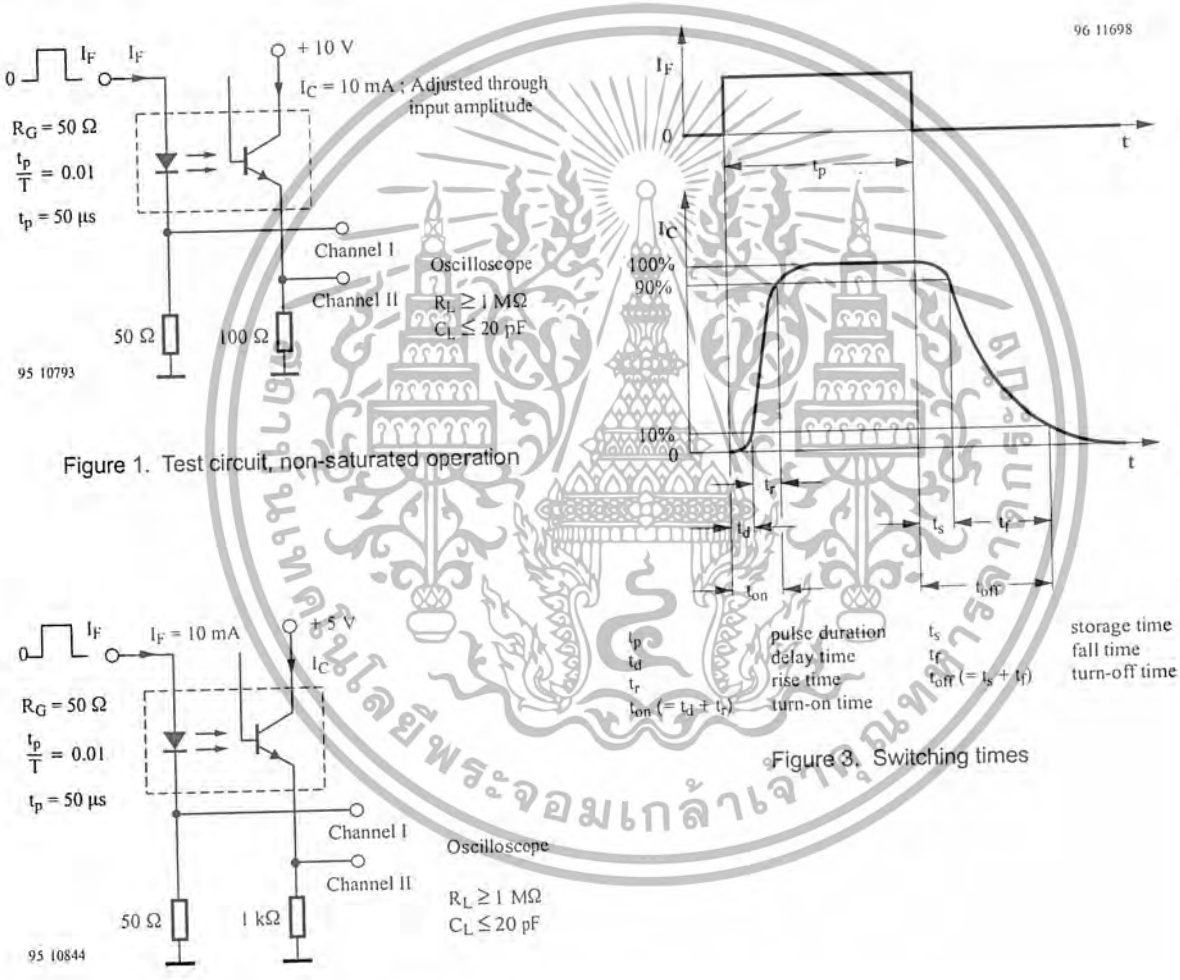


Figure 1. Test circuit, non-saturated operation

Figure 3. Switching times

Figure 2. Test circuit, saturated operation

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Typical Characteristics (T_{amb} = 25°C, unless otherwise specified)

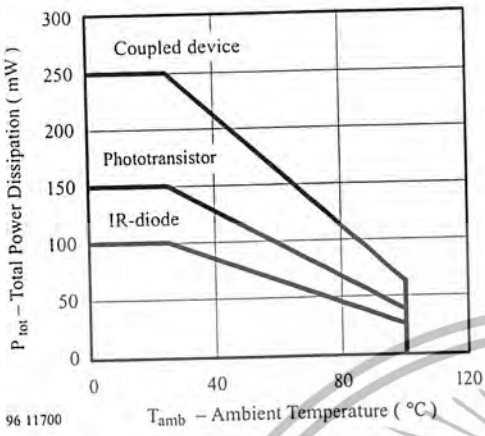


Figure 4. Total Power Dissipation vs. Ambient Temperature

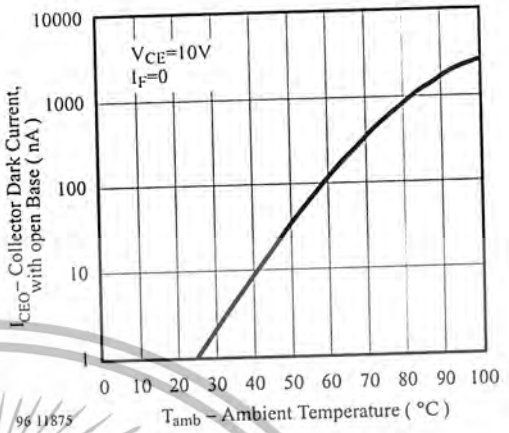


Figure 7. Collector Dark Current vs. Ambient Temperature

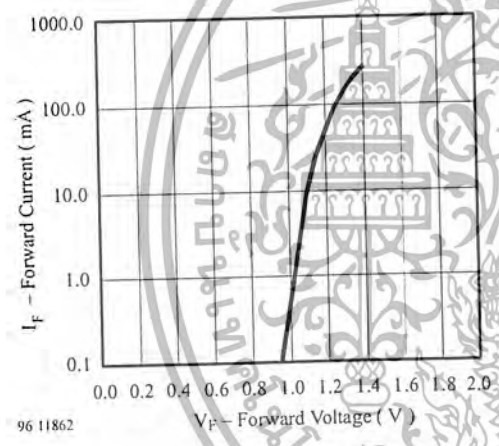


Figure 5. Forward Current vs. Forward Voltage

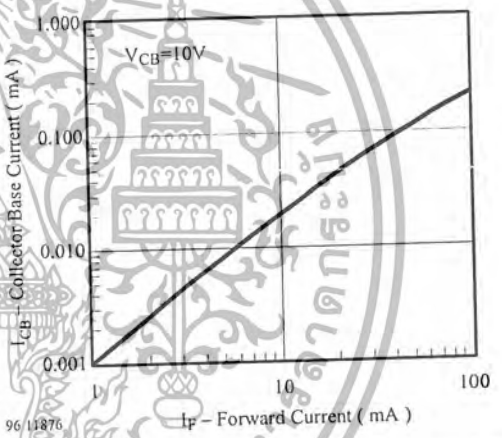


Figure 8. Collector Base Current vs. Forward Current

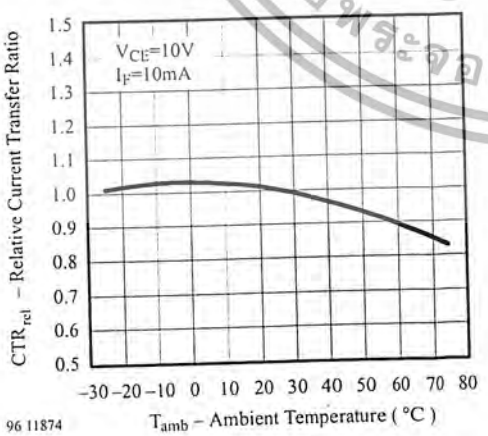


Figure 6. Relative Current Transfer Ratio vs. Ambient Temperature

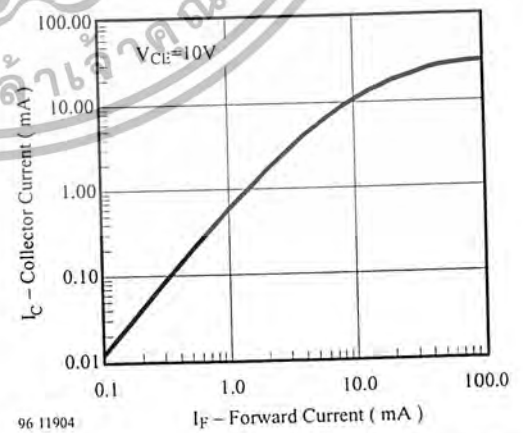


Figure 9. Collector Current vs. Forward Current

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

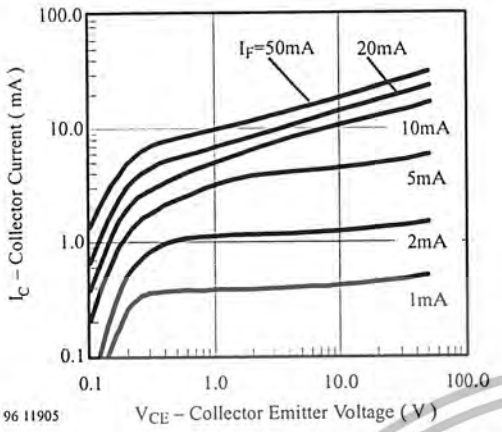


Figure 10. Collector Current vs. Collector Emitter Voltage

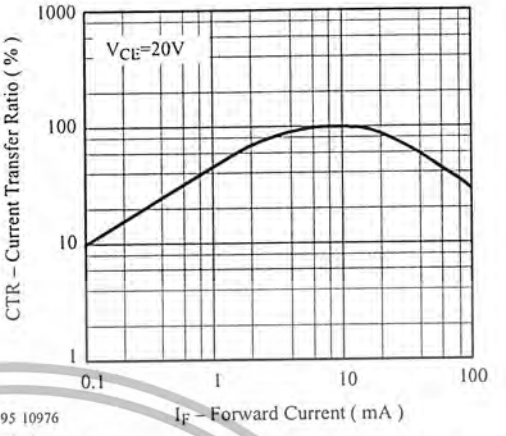


Figure 13. Current Transfer Ratio vs. Forward Current

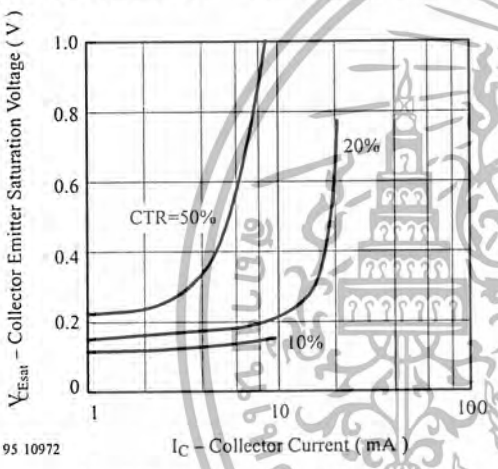


Figure 11. Collector Emitter Saturation Voltage vs. Collector Current

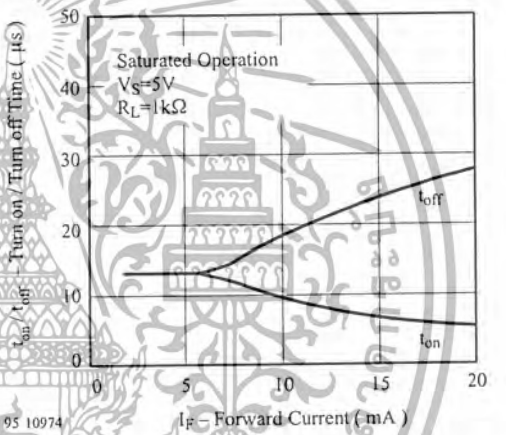


Figure 14. Turn on / off Time vs. Forward Current

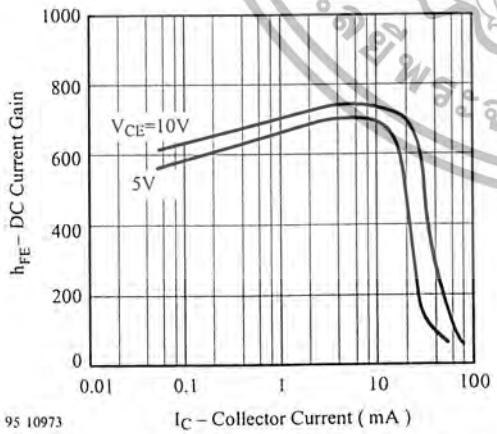


Figure 12. DC Current Gain vs. Collector Current

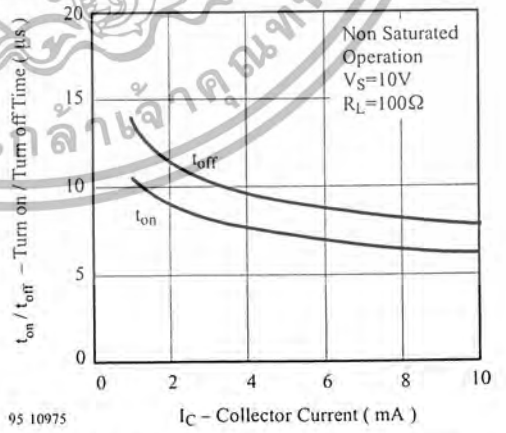


Figure 15. Turn on / off Time vs. Collector Current

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

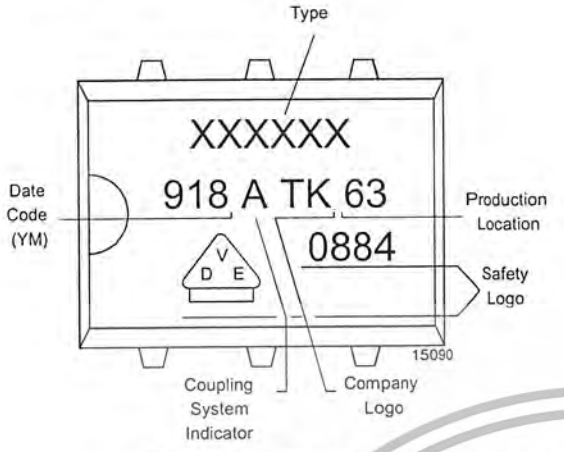
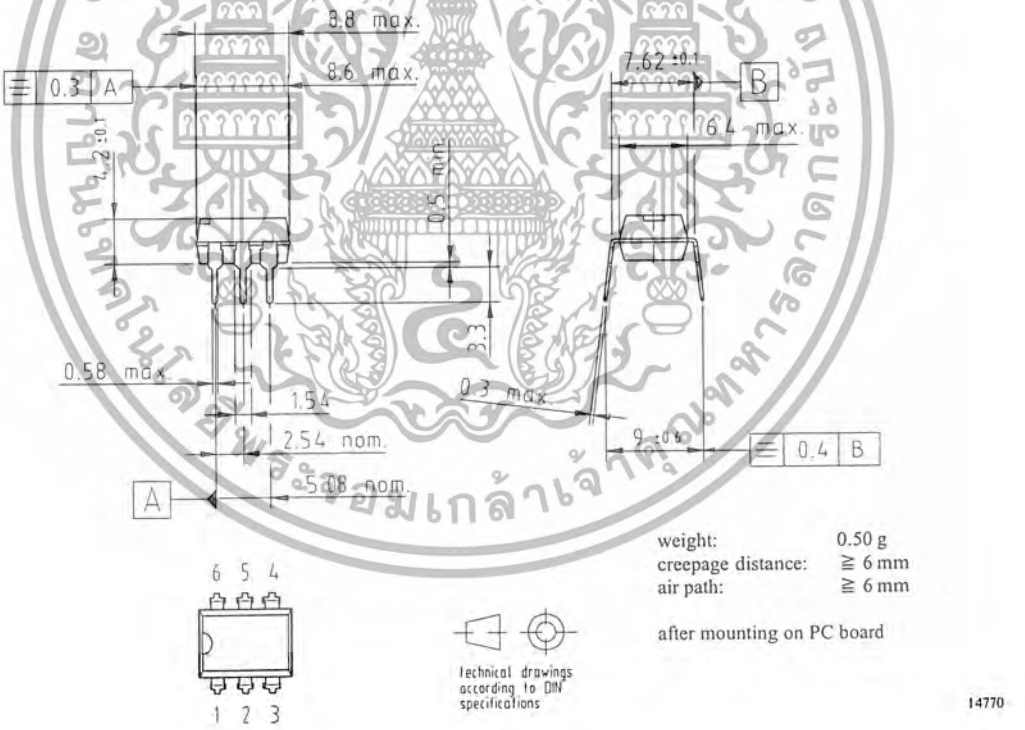


Figure 16. Marking example

Dimensions in mm



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

19-4323; Rev 10; 8/01

MAXIM

+5V-Powered, Multichannel RS-232 Drivers/Receivers

General Description

The MAX220–MAX249 family of line drivers/receivers is intended for all EIA/TIA-232E and V.28/V.24 communications interfaces, particularly applications where ±12V is not available.

These parts are especially useful in battery-powered systems, since their low-power shutdown mode reduces power dissipation to less than 5µW. The MAX225, MAX233, MAX235, and MAX245/MAX246/MAX247 use no external components and are recommended for applications where printed circuit board space is critical.

Applications

Portable Computers
Low-Power Modems
Interface Translation
Battery-Powered RS-232 Systems
Multidrop RS-232 Networks

Features

Superior to Bipolar

- ♦ Operate from Single +5V Power Supply (+5V and +12V—MAX231/MAX239)
- ♦ Low-Power Receive Mode in Shutdown (MAX223/MAX242)
- ♦ Meet All EIA/TIA-232E and V.28 Specifications
- ♦ Multiple Drivers and Receivers
- ♦ 3-State Driver and Receiver Outputs
- ♦ Open-Line Detection (MAX243)

Ordering Information

PART	TEMP. RANGE	PIN-PACKAGE
MAX220CPE	0°C to +70°C	16 Plastic DIP
MAX220CSE	0°C to +70°C	16 Narrow SO
MAX220CWE	0°C to +70°C	16 Wide SO
MAX220C/D	0°C to +70°C	Dice*
MAX220EPE	-40°C to +85°C	16 Plastic DIP
MAX220ESE	-40°C to +85°C	16 Narrow SO
MAX220EWE	-40°C to +85°C	16 Wide SO
MAX220EJE	-40°C to +85°C	16 CERDIP
MAX220MJE	-55°C to +125°C	16 CERDIP

Ordering Information continued at end of data sheet.

*Contact factory for dice specifications.

Selection Table

Part Number	Power Supply (V)	No. of RS-232 Drivers/Rx	No. of Ext. Caps	Nominal Cap. Value (µF)	SHDN & Three-State	Rx Active in SHDN	Data Rate (kbps)	Features
MAX220	+5	2/2	4	0.1	No	—	120	Ultra-low-power, industry-standard pinout
MAX222	+5	2/2	4	0.1	Yes	—	200	Low-power shutdown
MAX223 (MAX213)	+5	4/5	4	1.0 (0.1)	Yes	✔	120	MAX241 and receivers active in shutdown
MAX225	+5	5/5	0	—	Yes	✔	120	Available in SO
MAX230 (MAX200)	+5	5/0	4	1.0 (0.1)	Yes	—	120	5 drivers with shutdown
MAX231 (MAX201)	+5 and +7.5 to +13.2	2/2	2	1.0 (0.1)	No	—	120	Standard +5/+12V or battery supplies; same functions as MAX232
MAX232 (MAX202)	+5	2/2	4	1.0 (0.1)	No	—	120 (64)	Industry standard
MAX232A	+5	2/2	4	0.1	No	—	200	Higher slew rate, small caps
MAX233 (MAX203)	+5	2/2	0	—	No	—	120	No external caps
MAX233A	+5	2/2	0	—	No	—	200	No external caps, high slew rate
MAX234 (MAX204)	+5	4/0	4	1.0 (0.1)	No	—	120	Replaces 1488
MAX235 (MAX205)	+5	5/5	0	—	Yes	—	120	No external caps
MAX236 (MAX206)	+5	4/3	4	1.0 (0.1)	Yes	—	120	Shutdown, three state
MAX237 (MAX207)	+5	5/3	4	1.0 (0.1)	No	—	120	Complements IBM PC serial port
MAX238 (MAX208)	+5	4/4	4	1.0 (0.1)	No	—	120	Replaces 1488 and 1489
MAX239 (MAX209)	+5 and +7.5 to +13.2	3/5	2	1.0 (0.1)	No	—	120	Standard +5/+12V or battery supplies; single-package solution for IBM PC serial port
MAX240	+5	5/5	4	1.0	Yes	—	120	DIP or flatpack package
MAX241 (MAX211)	+5	4/5	4	1.0 (0.1)	Yes	—	120	Complete IBM PC serial port
MAX242	+5	2/2	4	0.1	Yes	✔	200	Separate shutdown and enable
MAX243	+5	2/2	4	0.1	No	—	200	Open-line detection simplifies cabling
MAX244	+5	8/10	4	1.0	No	—	120	High slew rate
MAX245	+5	8/10	0	—	Yes	✔	120	High slew rate, int. caps, two shutdown modes
MAX246	+5	8/10	0	—	Yes	✔	120	High slew rate, int. caps, three shutdown modes
MAX247	+5	8/9	0	—	Yes	✔	120	High slew rate, int. caps, nine operating modes
MAX248	+5	8/8	4	1.0	Yes	✔	120	High slew rate, selective half-chip enables
MAX249	+5	6/10	4	1.0	Yes	✔	120	Available in quad flatpack package

For pricing, delivery, and ordering information, please contact Maxim/Dallas Direct! at 1-888-629-4642, or visit Maxim's website at www.maxim-ic.com.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

+5V-Powered, Multichannel RS-232 Drivers/Receivers

ABSOLUTE MAXIMUM RATINGS—MAX220/222/232A/233A/242/243

Supply Voltage (V_{CC}) ..... -0.3V to +6V
Input Voltages
T_{IN} ..... -0.3V to (V_{CC} - 0.3V)
R_{IN} (Except MAX220) ..... ±30V
R_{IN} (MAX220) ..... ±25V
T_{OUT} (Except MAX220) (Note 1) ..... ±15V
T_{OUT} (MAX220) ..... ±13.2V
Output Voltages
T_{OUT} ..... ±15V
R_{OUT} ..... -0.3V to (V_{CC} + 0.3V)
Driver/Receiver Output Short Circuited to GND ..... Continuous
Continuous Power Dissipation (T_A = +70°C)
16-Pin Plastic DIP (derate 10.53mW/°C above +70°C) 842mW
18-Pin Plastic DIP (derate 11.11mW/°C above +70°C) 889mW
20-Pin Plastic DIP (derate 8.00mW/°C above +70°C) 440mW
16-Pin Narrow SO (derate 8.70mW/°C above +70°C) 696mW
16-Pin Wide SO (derate 9.52mW/°C above +70°C) 762mW
18-Pin Wide SO (derate 9.52mW/°C above +70°C) 762mW
20-Pin Wide SO (derate 10.00mW/°C above +70°C) 800mW
20-Pin SSOP (derate 8.00mW/°C above +70°C) 640mW
16-Pin CERDIP (derate 10.00mW/°C above +70°C) 800mW
18-Pin CERDIP (derate 10.53mW/°C above +70°C) 842mW
Operating Temperature Ranges
MAX2_ _AC_ _, MAX2_ _C_ _ ..... 0°C to +70°C
MAX2_ _AE_ _, MAX2_ _E_ _ ..... -40°C to +85°C
MAX2_ _AM_ _, MAX2_ _M_ _ ..... -55°C to +125°C
Storage Temperature Range ..... -65°C to +160°C
Lead Temperature (soldering, 10sec) ..... +300°C

Note 1: Input voltage measured with T_{OUT} in high-impedance state, $\overline{SHDN}$ or V_{CC} = 0V.
Note 2: For the MAX220, V+ and V- can have a maximum magnitude of 7V, but their absolute difference cannot exceed 13V.

Stresses beyond those listed under "Absolute Maximum Ratings" may cause permanent damage to the device. These are stress ratings only, and functional operation of the device at these or any other conditions beyond those indicated in the operational sections of the specifications is not implied. Exposure to absolute maximum rating conditions for extended periods may affect device reliability.

ELECTRICAL CHARACTERISTICS—MAX220/222/232A/233A/242/243

(V_{CC} = +5V ±10%, C1–C4 = 0.1µF, MAX220, C1 = 0.047µF, C2–C4 = 0.33µF, T_A = T_{MIN} to T_{MAX}, unless otherwise noted.)

PARAMETER	CONDITIONS		MIN	TYP	MAX	UNITS
RS-232 TRANSMITTERS						
Output Voltage Swing	All transmitter outputs loaded with 3kΩ to GND		±5	±8		V
Input Logic Threshold Low				1.4	0.8	V
Input Logic Threshold High	All devices except MAX220		2	1.4		V
	MAX220: V_{CC} = 5.0V		2.4			
Logic Pull-Up/Input Current	All except MAX220, normal operation			5	40	µA
	$\overline{SHDN}$ = 0V, MAX222/242, shutdown, MAX220			±0.01	±1	
Output Leakage Current	V_{CC} = 5.5V, $\overline{SHDN}$ = 0V, V_{OUT} = ±15V, MAX222/242			±0.01	±10	µA
	V_{CC} = $\overline{SHDN}$ = 0V, V_{OUT} = ±15V			±0.01	±10	
Data Rate				200	116	kb/s
Transmitter Output Resistance	V_{CC} = V+ = V- = 0V, V_{OUT} = ±2V		300	10M		Ω
Output Short-Circuit Current	V_{OUT} = 0V			±7	±22	mA
RS-232 RECEIVERS						
RS-232 Input Voltage Operating Range					±30	V
RS-232 Input Threshold Low	V_{CC} = 5V	All except MAX243 $R2_{IN}$	0.8	1.3		V
		MAX243 $R2_{IN}$ (Note 2)	-3			
RS-232 Input Threshold High	V_{CC} = 5V	All except MAX243 $R2_{IN}$		1.8	2.4	V
		MAX243 $R2_{IN}$ (Note 2)		-0.5	-0.1	
RS-232 Input Hysteresis	All except MAX243, V_{CC} = 5V, no hysteresis in shdn.		0.2	0.5	1	V
	MAX243			1		
RS-232 Input Resistance			3	5	7	kΩ
TTL/CMOS Output Voltage Low	I_{OUT} = 3.2mA			0.2	0.4	V
TTL/CMOS Output Voltage High	I_{OUT} = -1.0mA		3.5	V_{CC} - 0.2		V
TTL/CMOS Output Short-Circuit Current	Sourcing V_{OUT} = GND		-2	-10		mA
	Shrinking V_{OUT} = V_{CC}		10	30		

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ELECTRICAL CHARACTERISTICS—MAX220/222/232A/233A/242/243 (continued)

(V_{CC} = +5V ±10%, C1–C4 = 0.1µF, MAX220, C1 = 0.047µF, C2–C4 = 0.33µF, T_A = T_{MIN} to T_{MAX}, unless otherwise noted.)

PARAMETER	CONDITIONS		MIN	TYP	MAX	UNITS
TTL/CMOS Output Leakage Current	$\overline{SHDN}$ = V_{CC} or $\overline{EN}$ = V_{CC} ($\overline{SHDN}$ = 0V for MAX222), 0V ≤ V_{OUT} ≤ V_{CC}			±0.05	±10	µA
$\overline{EN}$ Input Threshold Low	MAX242			1.4	0.8	V
$\overline{EN}$ Input Threshold High	MAX242		2.0	1.4		V
Operating Supply Voltage			4.5		5.5	V
V_{CC} Supply Current ($\overline{SHDN}$ = V_{CC}), Figures 5, 6, 11, 19	No load	MAX220		0.5	2	mA
		MAX222/232A/233A/242/243		4	10	
	3kΩ load both inputs	MAX220		12		
		MAX222/232A/233A/242/243		15		
Shutdown Supply Current	MAX222/242	T_A = +25°C		0.1	10	µA
		T_A = 0°C to +70°C		2	50	
		T_A = -40°C to +85°C		2	50	
		T_A = -55°C to +125°C		35	100	
$\overline{SHDN}$ Input Leakage Current	MAX222/242				±1	µA
$\overline{SHDN}$ Threshold Low	MAX222/242			1.4	0.8	V
$\overline{SHDN}$ Threshold High	MAX222/242		2.0	1.4		V
Transition Slew Rate	C_L = 50pF to 2500pF, R_L = 3kΩ to 7kΩ, V_{CC} = 5V, T_A = +25°C, measured from +3V to -3V or -3V to +3V	MAX222/232A/233A/242/243	6	12	30	V/µs
		MAX220	1.5	3	30	
Transmitter Propagation Delay TLL to RS-232 (normal operation), Figure 1	t_{PHLT}	MAX222/232A/233A/242/243		1.3	3.5	µs
		MAX220		4	10	
	t_{PLHT}	MAX222/232A/233A/242/243		1.5	3.5	
		MAX220		5	10	
Receiver Propagation Delay RS-232 to TLL (normal operation), Figure 2	t_{PHLR}	MAX222/232A/233A/242/243		0.5	1	µs
		MAX220		0.6	3	
	t_{PLHR}	MAX222/232A/233A/242/243		0.6	1	
		MAX220		0.8	3	
Receiver Propagation Delay RS-232 to TLL (shutdown), Figure 2	t_{PHLS}	MAX242		0.5	10	µs
	t_{PLHS}	MAX242		2.5	10	
Receiver-Output Enable Time, Figure 3	t_{ER}	MAX242		125	500	ns
Receiver-Output Disable Time, Figure 3	t_{DR}	MAX242		160	500	ns
Transmitter-Output Enable Time ($\overline{SHDN}$ goes high), Figure 4	t_{ET}	MAX222/242, 0.1µF caps (includes charge-pump start-up)		250		µs
Transmitter-Output Disable Time ($\overline{SHDN}$ goes low), Figure 4	t_{DT}	MAX222/242, 0.1µF caps		600		ns
Transmitter + to - Propagation Delay Difference (normal operation)	t_{PHLT} - t_{PLHT}	MAX222/232A/233A/242/243		300		ns
		MAX220		2000		
Receiver + to - Propagation Delay Difference (normal operation)	t_{PHLR} - t_{PLHR}	MAX222/232A/233A/242/243		100		ns
		MAX220		225		

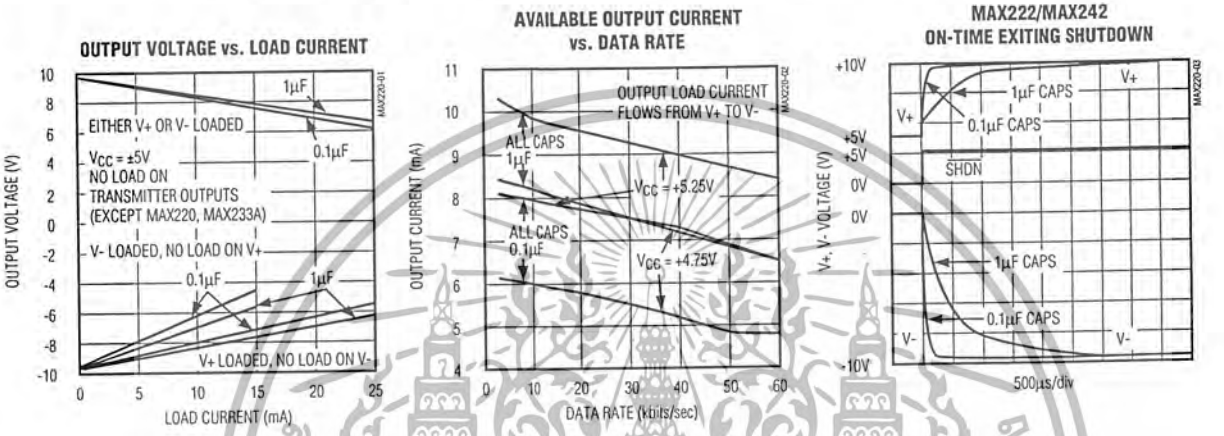
Note 3: MAX243 $R2_{OUT}$ is guaranteed to be low when $R2_{IN}$ is ≥ 0V or is floating.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

+5V-Powered, Multichannel RS-232 Drivers/Receivers

Typical Operating Characteristics

MAX220/MAX222/MAX232A/MAX233A/MAX242/MAX243



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ABSOLUTE MAXIMUM RATINGS—MAX223/MAX230–MAX241

V_{CC} ..... -0.3V to +6V
V+ ..... (V_{CC} - 0.3V) to +14V
V- ..... +0.3V to -14V
Input Voltages
T_{IN} ..... -0.3V to (V_{CC} + 0.3V)
R_{IN} ..... ±30V
Output Voltages
T_{OUT} ..... (V+ + 0.3V) to (V- - 0.3V)
R_{OUT} ..... -0.3V to (V_{CC} + 0.3V)
Short-Circuit Duration, T_{OUT} ..... Continuous
Continuous Power Dissipation (T_A = +70°C)
14-Pin Plastic DIP (derate 10.00mW/°C above +70°C) 800mW
16-Pin Plastic DIP (derate 10.53mW/°C above +70°C) 842mW
20-Pin Plastic DIP (derate 11.11mW/°C above +70°C) 889mW
24-Pin Narrow Plastic DIP
(derate 13.33mW/°C above +70°C) 1.07W
24-Pin Plastic DIP (derate 9.09mW/°C above +70°C) 500mW
16-Pin Wide SO (derate 9.52mW/°C above +70°C) 762mW
20-Pin Wide SO (derate 10 00mW/°C above +70°C) 800mW
24-Pin Wide SO (derate 11.76mW/°C above +70°C) 941mW
28-Pin Wide SO (derate 12.50mW/°C above +70°C) 1W
44-Pin Plastic FP (derate 11.11mW/°C above +70°C) 889mW
14-Pin CERDIP (derate 9.09mW/°C above +70°C) 727mW
16-Pin CERDIP (derate 10.00mW/°C above +70°C) 800mW
20-Pin CERDIP (derate 11.11mW/°C above +70°C) 889mW
24-Pin Narrow CERDIP
(derate 12.50mW/°C above +70°C) 1W
24-Pin Sidebraze (derate 20.0mW/°C above +70°C) 1.6W
28-Pin SSOP (derate 9.52mW/°C above +70°C) 762mW
Operating Temperature Ranges
MAX2_ _C_ _ ..... 0°C to +70°C
MAX2_ _E_ _ ..... -40°C to +85°C
MAX2_ _M_ _ ..... -55°C to +125°C
Storage Temperature Range ..... -65°C to +160°C
Lead Temperature (soldering, 10sec) ..... +300°C

Stresses beyond those listed under "Absolute Maximum Ratings" may cause permanent damage to the device. These are stress ratings only, and functional operation of the device at these or any other conditions beyond those indicated in the operational sections of the specifications is not implied. Exposure to absolute maximum rating conditions for extended periods may affect device reliability.

ELECTRICAL CHARACTERISTICS—MAX223/MAX230–MAX241

(MAX223/230/232/234/236/237/238/240/241, V_{CC} = +5V ±10; MAX233/MAX235, V_{CC} = 5V ±5%, C1–C4 = 1.0µF; MAX231/MAX239, V_{CC} = 5V ±10%; V+ = 7.5V to 13.2V; T_A = T_{MIN} to T_{MAX}; unless otherwise noted.)

PARAMETER	CONDITIONS		MIN	TYP	MAX	UNITS
Output Voltage Swing	All transmitter outputs loaded with 3kΩ to ground		±5.0	±7.3		V
V_{CC} Power-Supply Current	No load, T_A = +25°C	MAX232/233		5	10	mA
		MAX223/230/234–238/240/241		7	15	
		MAX231/239		0.4	1	
V+ Power-Supply Current		MAX231		1.8	5	mA
		MAX239		5	15	
Shutdown Supply Current	T_A = +25°C	MAX223		15	50	µA
		MAX230/235/236/240/241		1	10	
Input Logic Threshold Low	T_{IN}; EN, SHDN (MAX233); $\overline{EN}$, SHDN (MAX230/235–241)				0.8	V
Input Logic Threshold High	T_{IN}		2.0			V
	EN, $\overline{SHDN}$ (MAX223); $\overline{EN}$, SHDN (MAX230/235/236/240/241)		2.4			
Logic Pull-Up Current	T_{IN} = 0V			1.5	200	µA
Receiver Input Voltage Operating Range			-30		30	V

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

+5V-Powered, Multichannel RS-232 Drivers/Receivers

ELECTRICAL CHARACTERISTICS—MAX223/MAX230–MAX241 (continued)

(MAX223/230/232/234/236/237/238/240/241, V_{CC} = +5V ±10; MAX233/MAX235, V_{CC} = 5V ±5%, C1–C4 = 1.0µF; MAX231/MAX239, V_{CC} = 5V ±10%; V+ = 7.5V to 13.2V; T_A = T_{MIN} to T_{MAX}; unless otherwise noted.)

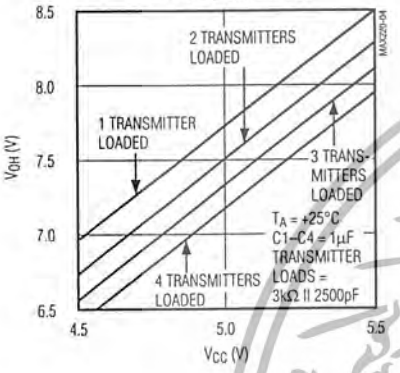
PARAMETER	CONDITIONS			MIN	TYP	MAX	UNITS
RS-232 Input Threshold Low	T_A = +25°C, V_{CC} = 5V	Normal operation $\overline{SHDN}$ = 5V (MAX223) SHDN = 0V (MAX235/236/240/241)		0.8	1.2		V
		Shutdown (MAX223) $\overline{SHDN}$ = 0V, $\overline{EN}$ = 5V ($R4_{IN}$, $R5_{IN}$)		0.6	1.5		
RS-232 Input Threshold High	T_A = +25°C, V_{CC} = 5V	Normal operation $\overline{SHDN}$ = 5V (MAX223) SHDN = 0V (MAX235/236/240/241)			1.7	2.4	V
		Shutdown (MAX223) $\overline{SHDN}$ = 0V, $\overline{EN}$ = 5V ($R4_{IN}$, $R5_{IN}$)			1.5	2.4	
RS-232 Input Hysteresis	V_{CC} = 5V, no hysteresis in shutdown			0.2	0.5	1.0	V
RS-232 Input Resistance	T_A = +25°C, V_{CC} = 5V			3	5	7	kΩ
TTL/CMOS Output Voltage Low	I_{OUT} = 1.6mA (MAX231/232/233, I_{OUT} = 3.2mA)					0.4	V
TTL/CMOS Output Voltage High	I_{OUT} = -1mA			3.5	V_{CC} - 0.4		V
TTL/CMOS Output Leakage Current	0V ≤ R_{OUT} ≤ V_{CC}; EN = 0V (MAX223); $\overline{EN}$ = V_{CC} (MAX235–241)				0.05	±10	µA
Receiver Output Enable Time	Normal operation	MAX223			600		ns
		MAX235/236/239/240/241			400		
Receiver Output Disable Time	Normal operation	MAX223			900		ns
		MAX235/236/239/240/241			250		
Propagation Delay	RS-232 IN to TTL/CMOS OUT, C_L = 150pF	Normal operation			0.5	10	µs
		SHDN = 0V (MAX223)	t_{PHLS}		4	40	
			t_{PLHS}		6	40	
Transition Region Slew Rate	MAX223/MAX230/MAX234–241, T_A = +25°C, V_{CC} = 5V, R_L = 3kΩ to 7kΩ, C_L = 50pF to 2500pF, measured from +3V to -3V or -3V to +3V			3	5.1	30	V/µs
	MAX231/MAX232/MAX233, T_A = +25°C, V_{CC} = 5V, R_L = 3kΩ to 7kΩ, C_L = 50pF to 2500pF, measured from +3V to -3V or -3V to +3V				4	30	
Transmitter Output Resistance	V_{CC} = V+ = V- = 0V, V_{OUT} = ±2V			300			Ω
Transmitter Output Short-Circuit Current					±10		mA

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

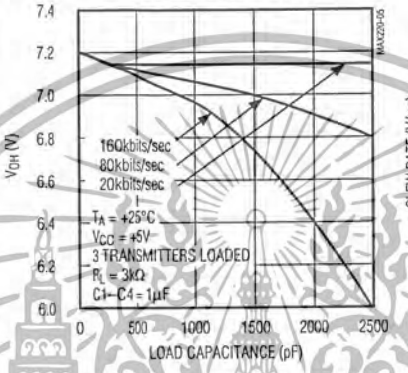
Typical Operating Characteristics

MAX223/MAX230–MAX241

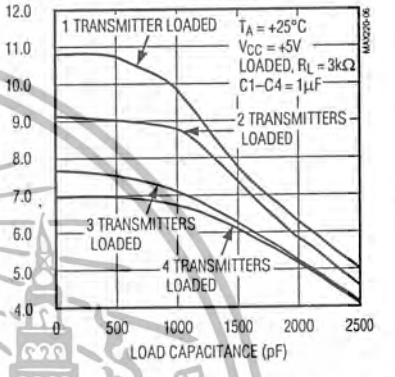
TRANSMITTER OUTPUT VOLTAGE (V_{OH}) vs. V_{CC}



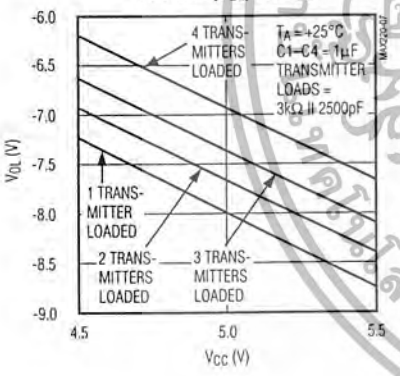
TRANSMITTER OUTPUT VOLTAGE (V_{OH}) vs. LOAD CAPACITANCE AT DIFFERENT DATA RATES



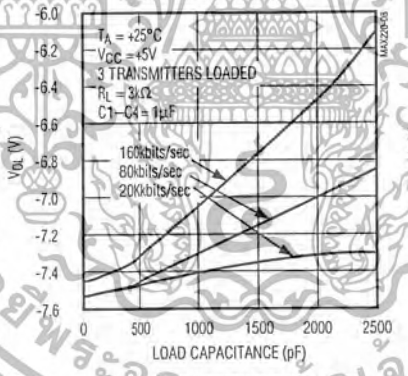
TRANSMITTER SLEW RATE vs. LOAD CAPACITANCE



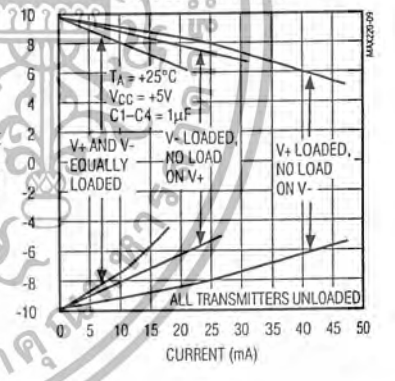
TRANSMITTER OUTPUT VOLTAGE (V_{OL}) vs. V_{CC}



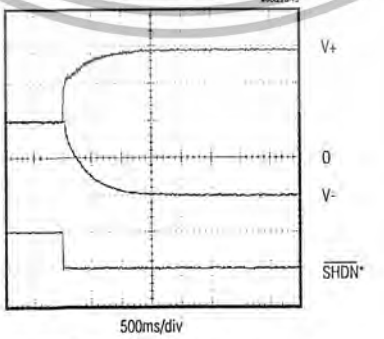
TRANSMITTER OUTPUT VOLTAGE (V_{OL}) vs. LOAD CAPACITANCE AT DIFFERENT DATA RATES



TRANSMITTER OUTPUT VOLTAGE (V+, V-) vs. LOAD CURRENT



V+, V- WHEN EXITING SHUTDOWN (1µF CAPACITORS)



*SHUTDOWN POLARITY IS REVERSED FOR NON MAX241 PARTS

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

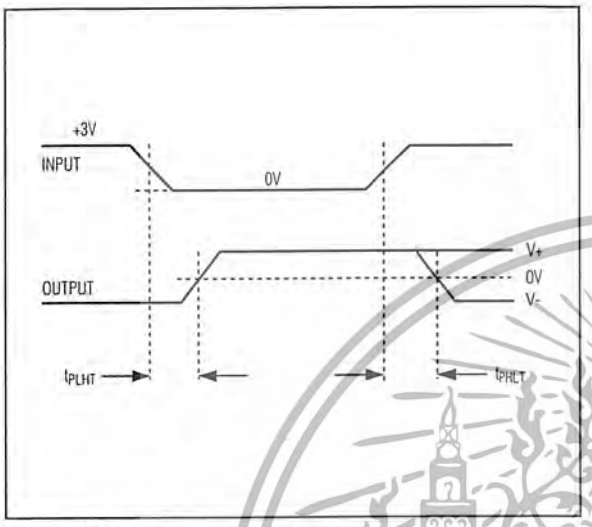


Figure 1. Transmitter Propagation-Delay Timing

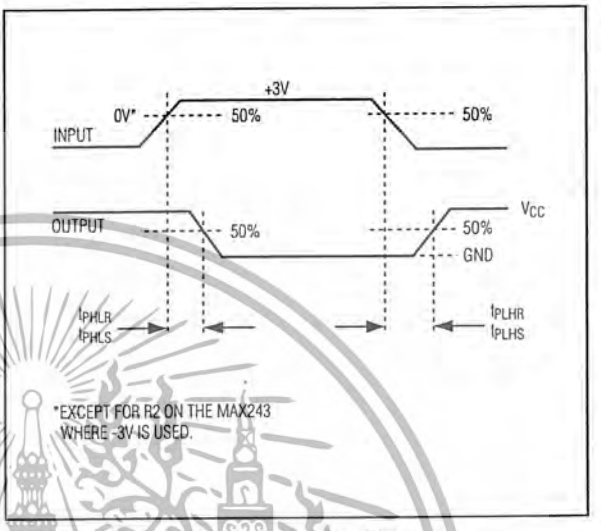


Figure 2. Receiver Propagation-Delay Timing

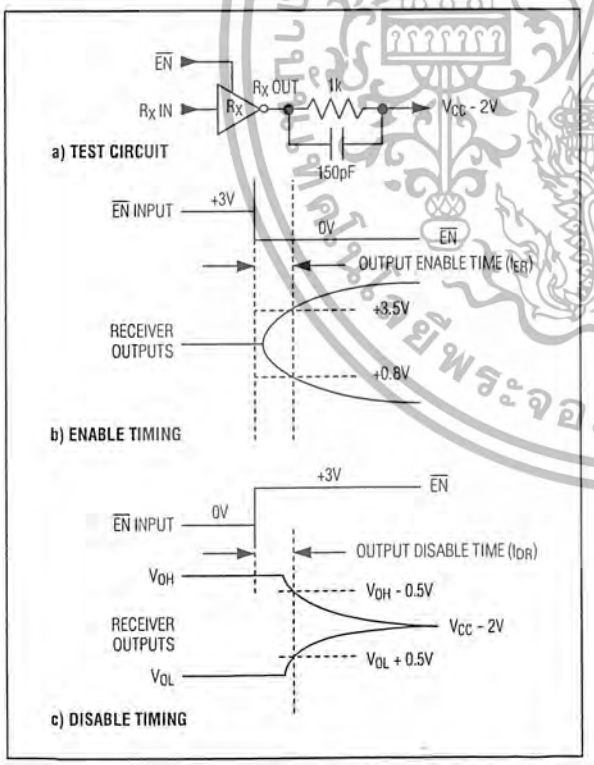


Figure 3. Receiver-Output Enable and Disable Timing

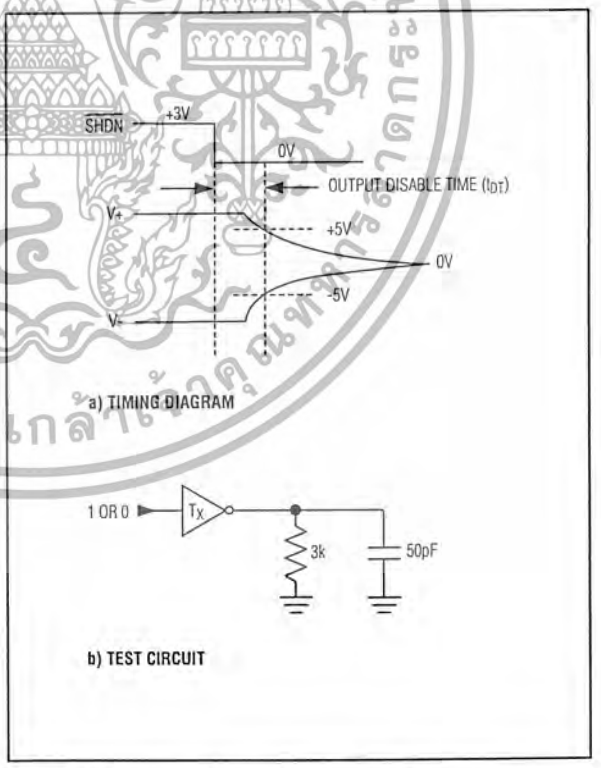
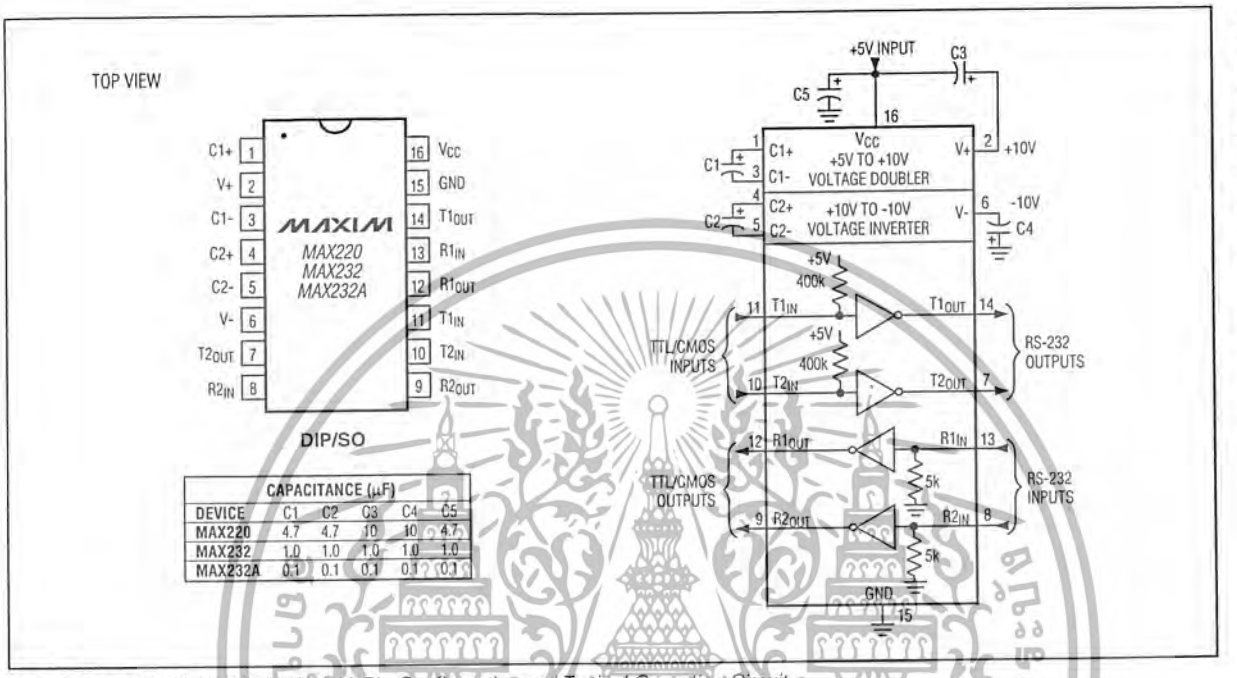


Figure 4. Transmitter-Output Disable Timing

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



CAPACITANCE (µF)					
DEVICE	C1	C2	C3	C4	C5
MAX220	4.7	4.7	10	10	4.7
MAX232	1.0	1.0	1.0	1.0	1.0
MAX232A	0.1	0.1	0.1	0.1	0.1

Figure 5. MAX220/MAX232/MAX232A Pin Configuration and Typical Operating Circuit

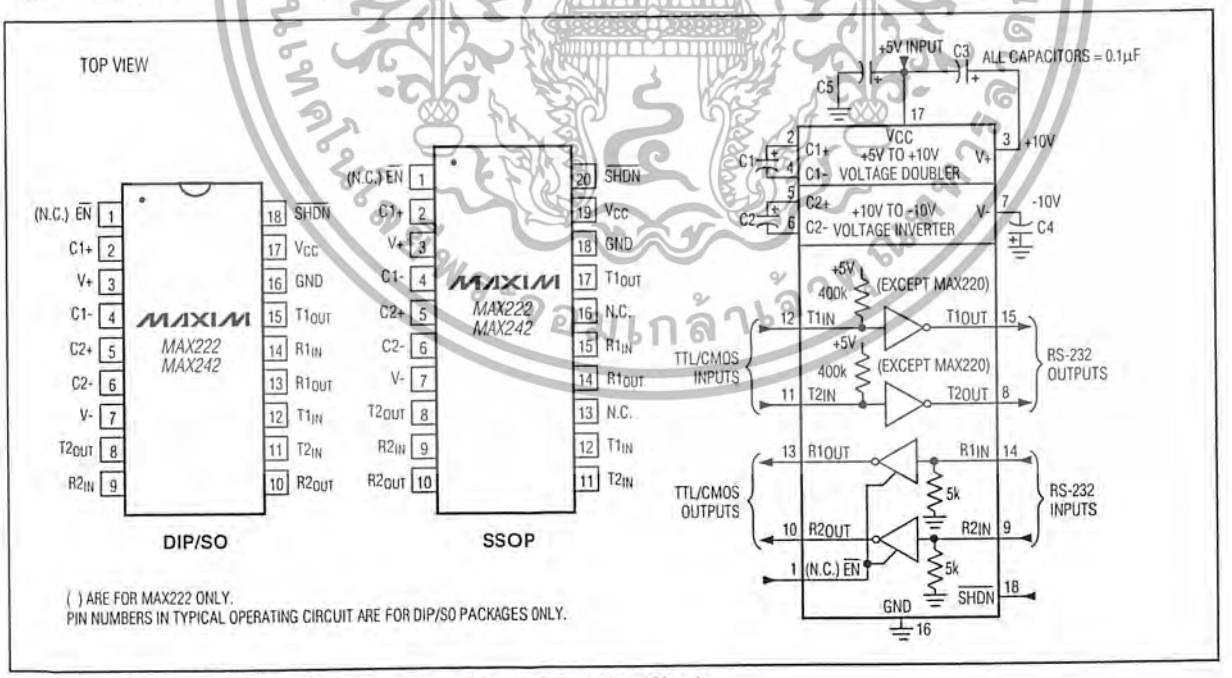


Figure 6. MAX222/MAX242 Pin Configurations and Typical Operating Circuit

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หนังสืออ้างอิง

1. สมยศ จุณณะปิยะ,รศ.,"การประยุกต์ใช้ไมโครคอนโทรลเลอร์ ตระกูล MCS-51", คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง, 2541.
2. วิวัฒน์ กิรานนท์,ศ.ดร.,"วิศวกรรมการสื่อสาร",คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง, 2542.
3. สุวิพล สิทธิชีวภาค,ผศ.ดร.,"เทคโนโลยีการสื่อสารระบบดิจิตอล",คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง, 2539.
4. ธีรวัฒน์ ประกอบผล,"การประยุกต์ใช้งานไมโครคอนโทรลเลอร์",สำนักพิมพ์สมาคมส่งเสริมเทคโนโลยี(ไทย-ญี่ปุ่น),2543.
5. ลุงเอี๋ยว,"รถกระป๋อง (2) เทอร์โบ แชนเนลที่หายไปของรถกระป๋อง",วารสารฮอบบี้อิเล็กทรอนิกส์,ฉบับที่117,2545,หน้า 92 – 99.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้