สำนักหอสมุดกลาง พระจอมเกล้าลาดกระบัง

รถบังคับอินฟราเรด

Infrared Remote Control Car

โดย

นายปิติพงษ์ ประภามณฑล

นายพิสิษฐ ไชยรังสินันท์

T042652

อาจารย์ที่ปรึกษา

รศ.ดร.มนัส สังวรศิลป์

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งการศึกษาตามหลักสูตรวิศวกรรมศาสตร์บัณฑิต

สาขาวิศวกรรมอิเล็กทรอนิกส์

คณะวิศวกรรมศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ประจำภาคเรียนที่ 2 ปีการศึกษา 2543

เลขหมู่.....
เลขทะเบียน..... 42652
วัน, เดือน, ปี..... 5 มิ.ย. 2545

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

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รายงานเรื่อง	รถบังคับอินฟราเรด		
	Infrared Car		
จัดทำโดย	นายปิติพงษ์	ประภามณฑล	รหัส 40010460
	นายพิสิษฐ	ไชยรังสินันท์	รหัส 40010523
อาจารย์ที่ปรึกษา	รศ.ดร.มนัส	สังวรศิลป์	

รายงานฉบับนี้ได้ผ่านการตรวจสอบโดยอาจารย์ที่ปรึกษาแล้ว

ลงชื่อ..... อาจารย์ที่ปรึกษา

(รศ.ดร.มนัส สังวรศิลป์)

วันที่ 20 / มี.ค. / 2544

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กิตติกรรมประกาศ

ผู้จัดทำรายงานขอขอบพระคุณอาจารย์ที่ปรึกษาและอาจารย์ในภาควิชาอิเล็กทรอนิกส์ทุกท่านที่ได้ให้คำปรึกษาในการทำโครงงานชิ้นนี้ตลอดจนข้อคิดเห็นแนวทางในการแก้ไขปัญหาต่างๆ จนสำเร็จมาได้ด้วยดี รวมถึงอาจารย์ผู้สอนวิชา Optoelectronics ซึ่งผู้จัดทำได้นำความรู้ที่ได้จากการเรียนวิชานี้มาประยุกต์ใช้จนสามารถทำโครงงานนี้ได้สำเร็จ

สุดท้ายนี้ผู้จัดทำรายงานใคร่ขอขอบคุณบุคคลที่ได้กล่าวถึงเป็นอย่างสูง ผู้จัดทำรายงานขอระลึกถึงด้วยความขอบคุณเป็นอย่างยิ่ง

ปีติพงษ์ ประภามณฑล

ปีติพงษ์ ประภามณฑล

พิสิษฐ ไชยรังสินันท์

พิสิษฐ ไชยรังสินันท์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รถบังคับอินฟราเรด

นายปิติพงษ์ ประภามณฑล
นายพิสิษฐ ไชยรังสินันท์
รศ.ดร.มนัส สังวรศิลป์
(อาจารย์ที่ปรึกษา)
ภาคการศึกษาที่ 2 ปีการศึกษา 2543

บทคัดย่อ

โครงงานนี้เป็นรูปแบบการสื่อสารดิจิตอลซึ่งใช้การส่งข้อมูลโดยใช้แสงอินฟาเรดเพื่อไปใช้ในการติดต่อระหว่างเครื่องคอมพิวเตอร์ส่วนบุคคล (PC) กับไมโครคอนโทรลเลอร์เบอร์ 8051 โดยมีตัวส่งและตัวรับอินฟราเรดเป็นตัวกลางในการส่งข้อมูลดิจิตอล ซึ่งจะนำไปประยุกต์ใช้กับการควบคุมรถให้เคลื่อนที่และสามารถทำการควบคุมผ่านทางคอมพิวเตอร์โดยมีกล้องวีดีโอจับภาพแล้วนำมาแสดงที่จอคอมพิวเตอร์ สามารถบังคับรถได้ในรัศมีสูงสุด 5 เมตร

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Infrared Remote Control Car

Mr.Pitipong Prapamontol
Mr.Pisit Chaiyarangsinan
Mr.Manus Sungvornsil (Advisor)
2nd semister Educational 2543

Abstract

This project is the digital communication that sends data by infrared for communicating between Personal Computer (PC) and Microcontroller (8051). The car is controlled by operator via a video camera on the monitor.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ

	หน้า
กิตติกรรมประกาศ	I
บทคัดย่อ	II
Abstract	III
สารบัญ	IV
สารบัญรูป	V
บทที่ 1 บทนำ	
1.1 หลักการทำงานของโครงงาน	1
บทที่ 2 รูปแบบการส่งข้อมูลทางคอมพิวเตอร์	
2.1 การทำงานของวงจรอินเตอร์เฟส	2
2.2 ลักษณะสมบัติทางไฟฟ้าของสัญญาณ RS-232	
2.2.1 สถานะ	5
บทที่ 3 การส่งผ่านข้อมูลแบบดิจิตอล	
3.1 การส่งข้อมูลระยะไกล	
3.1.1 ระบบของสัญญาณควบคุม	6
1) ระบบใช้สาย	6
2) ระบบไร้สาย	6
3.1.2 การใช้แสงอินฟาเรด	6
3.2 ตัวส่งสัญญาณอินฟราเรด	
3.2.1 หลักการทำงาน	8
3.2.2 การควบคุมด้วยสัญญาณโทนเบริสต์	8
3.2.3 วงจรตัวส่งข้อมูลด้วยอินฟราเรด	10
3.2.4 การออกแบบสัญญาณแคเรียร์ 38 kHz	11
3.2.6 รหัสสัญญาณจากตัวส่ง	12

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ (ต่อ)

	หน้า
บทที่ 4 ตัวรับข้อมูลแบบอินฟาเรด	
4.1 การออกแบบวงจรภาครับ	
4.1.1 อุปกรณ์รับแสงอินฟาเรดและดีมอดูเลทสัญญาณ	13
4.1.2 หลักการทำงาน	14
4.2 วงจรภาคขับมอร์เตอร์และรีเลย์	15
บทที่ 5 การทดลองและผลการทดลอง	17
บทที่ 6 สรุปและวิจารณ์การทดลอง	20
ภาคผนวก	21
บรรณานุกรม	26

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูป

หน้า

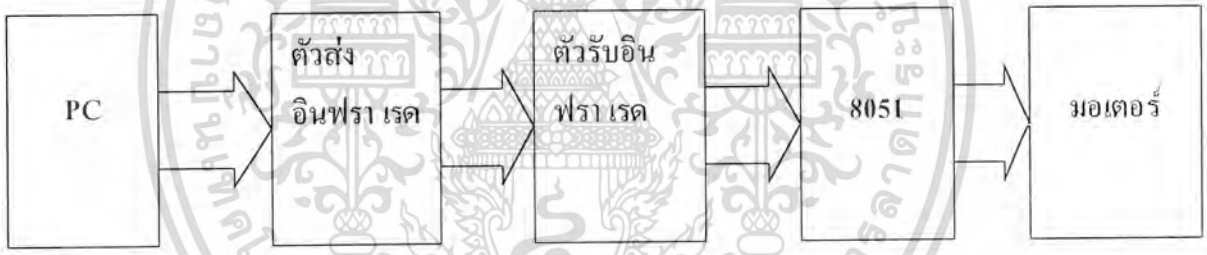
รูปที่ 1.1 บล็อกไดอะแกรมแสดงการทำงานของโครงงาน 1
รูปที่ 2.1 พอร์ท RS-232 2
รูปที่ 2.2 การต่อสายสัญญาณแต่ละขาของ RS-232 4
รูปที่ 2.3 การต่อสายสัญญาณของ RS-232 โดยไม่คำนึงถึงสัญญาณ
Hand Shaking 4
รูปที่ 2.4 การต่อสายสัญญาณของ RS-232 ในโครงงาน 5
รูปที่ 2.5 แสดงย่านแรงดันไฟฟ้าที่ใช้ในสัญญาณ RS-232C 5
รูปที่ 3.1 การไบอัสอินฟาเรด 7
รูปที่ 3.2 รูปแบบการส่งข้อมูลแบบอนุกรม 7
รูปที่ 3.3 บล็อกไดอะแกรมแสดงการทำงานของส่วนส่งสัญญาณอินฟราเรด 8
รูปที่ 3.4 ลักษณะสัญญาณแบบโทนเบิรสต์ 9
รูปที่ 3.5 แสดงขั้นตอนการเกิดสัญญาณโทนเบิรสต์ 9
รูปที่ 3.6 วงจรตัวส่งข้อมูลอินฟาเรด 10
รูปที่ 3.7 สัญญาณที่ได้จากเอาท์พุทของอะสเตเบิลมัลติไวเบรเตอร์ 11
รูปที่ 3.8 แสดงวงจรอะสเตเบิลมัลติไวเบรเตอร์ 38 kHz 11
รูปที่ 4.1 วงจรตัวรับข้อมูลอินฟาเรด 13
รูปที่ 4.2 วงจรไมโครคอนโทรลเลอร์ 14
รูปที่ 4.3 วงจรภาคขับรีเลย์และมอเตอร์ 15

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 1
บทนำ

1.1 หลักการทำงานของโครงงาน

ในโครงงานนี้มีจุดประสงค์เพื่อการสื่อสารข้อมูลดิจิตอลไร้สายจากคอมพิวเตอร์ (PC) กับไมโครคอนโทรลเลอร์โดยผ่านตัวส่งและตัวรับอินฟราเรด ซึ่งหลักการทำงานแสดงได้ดังบล็อกไดอะแกรมรูปที่ 1.1 โดย PC จะส่งข้อมูลไปที่ตัวส่งอินฟาเรดโดยผ่านพอร์ท RS-232 แสงอินฟาเรดจะถูกส่งไปยังตัวรับอินฟาเรด ซึ่งข้อมูลที่ใช้ส่งและรับนั้นเป็นสัญญาณดิจิตอล ต่อจากนั้นเอาท์พุทจากตัวรับจะต่อไปที่ไมโครคอนโทรลเลอร์ 8051 โดยที่ 8051 จะให้แรงดันเอาท์พุทอยู่ในระดับ TTL (5V) ที่พอร์ทเอาท์พุทของ 8051 จะต่อไปยังชุดควบคุมรีเลย์และภาคขับมอเตอร์เพื่อให้รถเคลื่อนที่



รูปที่ 1.1 บล็อกไดอะแกรมแสดงการทำงานของโครงงาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

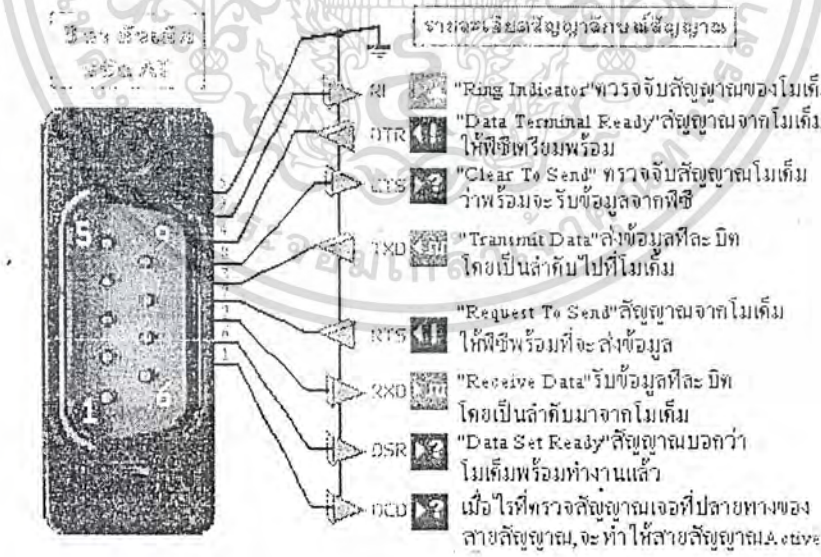
บทที่ 2

รูปแบบการส่งข้อมูลทางคอมพิวเตอร์

2.1 PORT RS-232C

Port RS-232C นี้ทำหน้าที่รับและส่งข้อมูลในแบบอนุกรม เรียกว่า Universal Asynchronous Adapter เนื่องจาก EIAได้กำหนดมาตรฐานของอุปกรณ์แบบอนุกรมเอาไว้ภายใต้ชื่อว่า RS-232C คำว่า RS ย่อมาจาก Recommend Standard ส่วน 232 เป็นหมายเลขบ่งบอกมาตรฐานตัวนี้ C เป็นหมายเลขของฉบับสุดท้ายของมาตรฐานนี้ ความเร็วในการเชื่อมต่อ RS-232C สามารถถ่ายเทข้อมูลได้ในช่วง 0-20,000 บิทต่อวินาที และระยะทางในการส่งสัญญาณตามมาตรฐานของ RS-232C ไม่ควรเกิน 50 ฟุต ซึ่งพอเพียงสำหรับการสื่อสารระหว่าง DTE กับ DCE

RS-232C เป็นมาตรฐาน มีชื่อเต็มว่า Interface Between Data Terminal Equipment and Data Communication Equipment employing serial binary interchange ซึ่งแปลตรงๆ ก็คือ เป็นมาตรฐานสำหรับการอินเตอร์เฟสอุปกรณ์ Data Terminal (DTE) เข้ากับอุปกรณ์ Data Communication (DCE) โดยอาศัยวิธีการส่งข้อมูลดิจิตอลแบบอนุกรมซึ่งมีตำแหน่งขาดังรูปที่ 2.1



รูปที่ 2.1 พอร์ท RS-232

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ซึ่งจะอธิบายขาได้ดังนี้

1. DCD หรือ Carrier Detector ขานี้จะแอคทีฟเมื่อตรวจพบสัญญาณที่อุปกรณ์ปลายทาง

2. RXD หรือ Receive Data เป็นขาอินพุตเข้าคอมพิวเตอร์ รับข้อมูลทีละบิตจากอุปกรณ์ที่อยู่ปลายทาง

3. TXD หรือ Transmit Data เป็นขาเอาท์พุทออกจากคอมพิวเตอร์ ส่งข้อมูลทีละบิตไปยังอุปกรณ์ปลายทาง

4. DTR หรือ Data Terminal Ready เป็นขารับสัญญาณจากอุปกรณ์ปลายทางเพื่อให้คอมพิวเตอร์พร้อมที่จะรับข้อมูล

5. GND หรือ Signal Ground

6. DSR หรือ Data Set Ready ขานี้จะบอกว่าอุปกรณ์ปลายทางพร้อมที่จะทำงานแล้ว

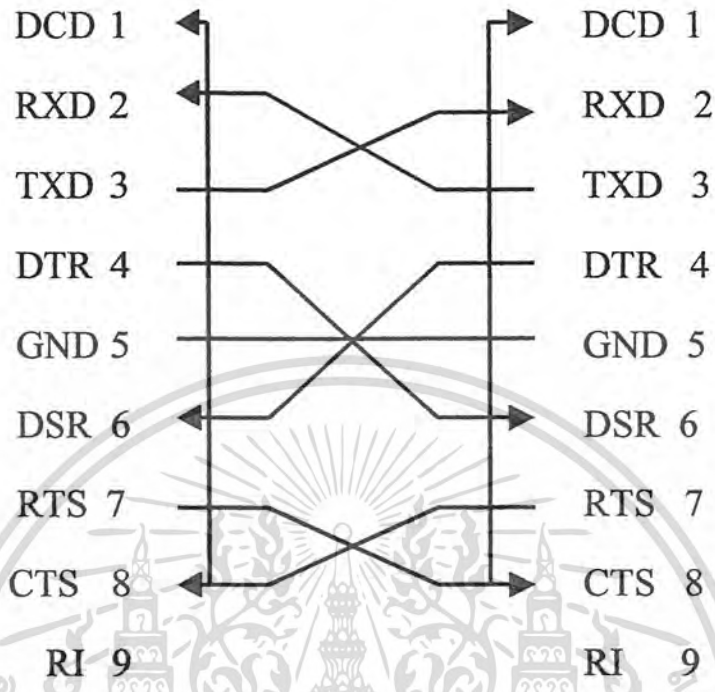
7. RTS หรือ Request To Send ขานี้เป็นสัญญาณจากอุปกรณ์ปลายทางเพื่อให้คอมพิวเตอร์พร้อมที่จะส่งข้อมูล

8. CTS หรือ Clear To Send เป็นขาสัญญาณตรวจจับสัญญาณจากอุปกรณ์ปลายทางว่าพร้อมจะรับข้อมูลหรือยัง

9. RI หรือ Ring Indicator ขานี้จะตรวจจับสัญญาณของอุปกรณ์ปลายทาง

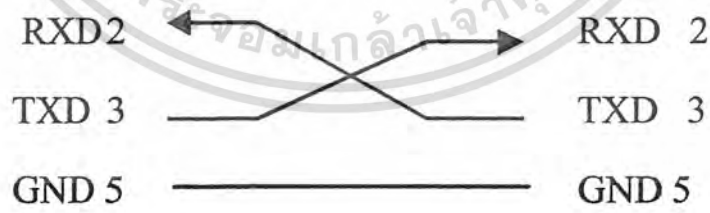
จะเห็นว่าขาทั้งหมดของพอร์ตอนุกรมมีถึง 9 ขา ซึ่งหากมีการติดต่อระหว่างคอมพิวเตอร์กับอุปกรณ์ใดๆ จะต้องมีการต่อสายสัญญาณของแต่ละขาเข้าด้วยกันดังนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.2 การต่อสายสัญญาณแต่ละขาของ RS-232

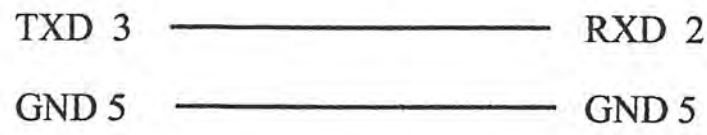
หากเราไม่คำนึงถึงสัญญาณ HandShaking จะทำให้ลดสายสัญญาณลงไปได้มากจะเหลือเพียง RXD TXD และ GND เท่านั้นดังรูป 2.3



รูปที่ 2.3 การต่อสายสัญญาณของ RS-232 โดยไม่คำนึงถึงสัญญาณ Hand Shaking

ในโครงงานนี้จะมีเพียงการส่งข้อมูลออกจากคอมพิวเตอร์เพียงอย่างเดียวดังนั้นจึงต่อสายเพียง 2 เส้นเท่านั้น ดังรูปที่ 2.4

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.4 การต่อสายสัญญาณของ RS-232 ในโครงงาน

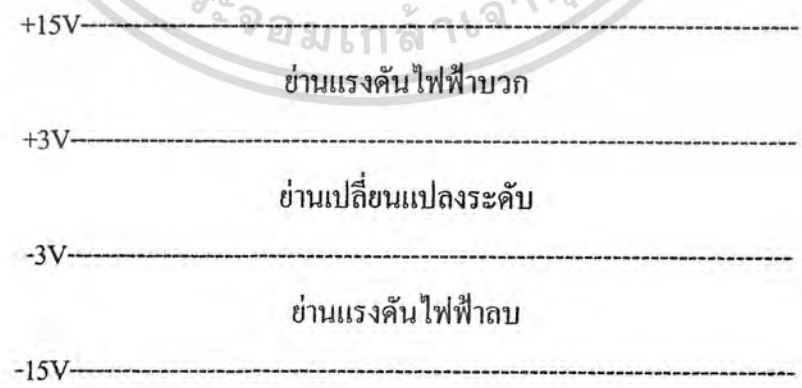
2.2 ลักษณะสมบัติทางไฟฟ้าของสัญญาณ RS-232C

2.2.1 สถานะ

สำหรับมาตรฐานของการใช้แรงดันไฟฟ้าจะกำหนดไว้ดังตารางมาตรฐานแรงดันไฟฟ้ารูปที่ 2.5 แรงดันสูงสุดที่วงจรใน DTE และ DCE ไม่ควรเกิน 25 V และ open voltage ต้องไม่เกิน 2 V (วัดเทียบกับ signal ground)

ตารางมาตรฐานของการใช้แรงดันไฟฟ้า

แรงดันไฟฟ้า	ลอจิก	สัญญาณ	ฟังก์ชันในการควบคุม
บวก	0	Space	ON
ลบ	1	Mark	OFF



รูปที่ 2.5 แสดงย่านแรงดันไฟฟ้าที่ใช้ในสัญญาณ RS-232C

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 3
การส่งผ่านข้อมูลแบบดิจิตอล

3.1 การส่งข้อมูลระยะไกล

3.1.1 ระบบของสัญญาณควบคุม

การส่งข้อมูลระยะไกลสามารถจำแนกออกได้เป็น 2 ประเภทตามลักษณะของการส่งผ่านสัญญาณ

1. ระบบใช้สาย

เป็นระบบควบคุมที่ต้องมีอุปกรณ์นำสัญญาณจากตัวส่งไปยังตัวรับ อุปกรณ์นำสัญญาณนั้นขึ้นอยู่กับชนิดของสัญญาณพาหะ ซึ่งได้แก่ สัญญาณไฟฟ้าและสัญญาณแสง เป็นต้น

ในกรณีที่เป็นสัญญาณไฟฟ้า อุปกรณ์นำสัญญาณอาจได้แก่ สายไฟ, สายโทรศัพท์, สายโคแอกเซียล เป็นต้น แต่ถ้าในกรณีที่เป็นสัญญาณแสง อุปกรณ์นำสัญญาณจะเป็นเส้นใยแก้วนำแสง หรือเรียกว่า ไฟเบอร์ออปติค

2. ระบบไร้สาย

เป็นระบบควบคุมที่ไม่ต้องมีอุปกรณ์ใดๆ เป็นตัวนำสัญญาณ โดยสัญญาณควบคุมจะเดินทางผ่านไปในอากาศ ซึ่งอาจอยู่ในรูปของสัญญาณเสียง สัญญาณแสง และคลื่นวิทยุ

ในปัจจุบัน สัญญาณแสง และคลื่นวิทยุกำลังเป็นที่นิยม คลื่นวิทยุจะนิยมใช้ทำเครื่องเล่นประเภทวิทยุบังคับ ส่วนสัญญาณแสง ได้แก่ ระบบอินฟาเรด

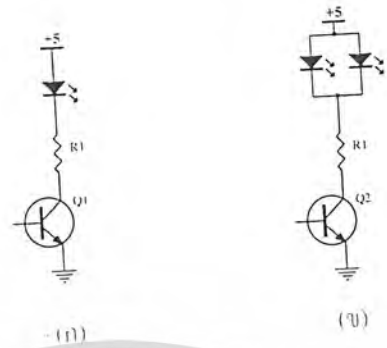
3.1.2 การใช้แสงอินฟาเรด

การใช้แสงอินฟาเรดเป็นสัญญาณควบคุมสามารถตัดปัญหาการรบกวนของแสงภายนอกอื่นๆ ลงไปได้โดยเด็ดขาด ยิ่งไปกว่านั้นวงจรใช้งานของระบบอินฟาเรดยังเป็นวงจรที่ง่ายและไม่ซับซ้อนมีความเชื่อถือได้สูงในการใช้งาน

การส่งสัญญาณแสงอินฟาเรดสามารถทำได้ด้วยวงจรง่ายๆ ดังรูปที่ 3.1 (ก) ซึ่งประกอบด้วย LED ที่เปล่งแสงในย่านอินฟาเรดต่อเข้ากับแหล่งจ่ายไฟ โดยมีความต้านทาน R1 ทำหน้าที่จำกัดกระแส

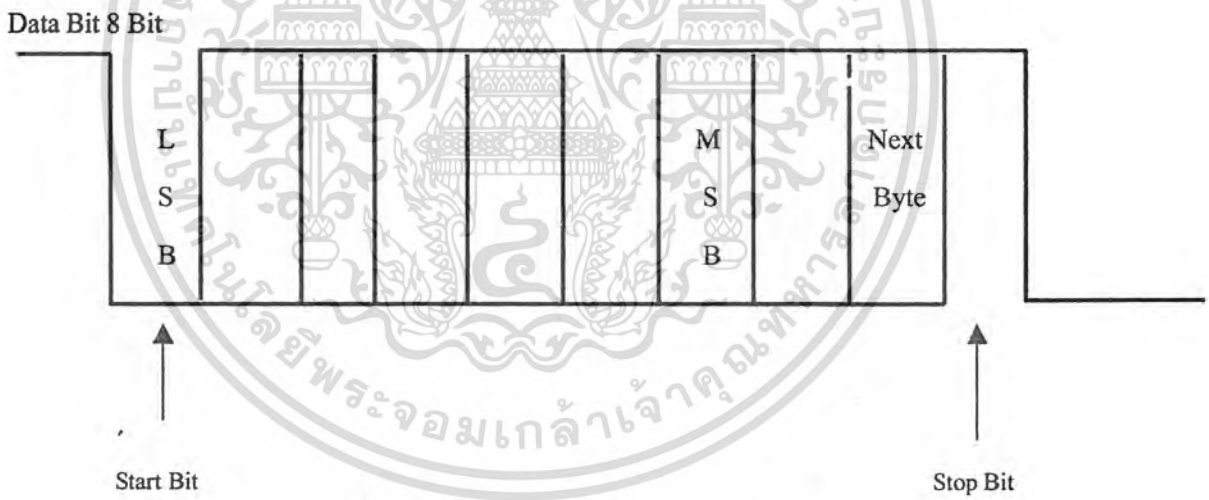
สัญญาณแสงที่ส่งออกโดย LED เพียงตัวเดียวจะเหมาะกับการใช้งานในระยะไม่กี่เมตรเท่านั้น การเพิ่มกำลังส่งของอินฟาเรดให้ไปได้ไกลขึ้นทำได้โดยใช้ LED หลายตัวต่อขนานกันดังรูปที่ 3.1 (ข) โดยที่ R1 จะต้องมีค่าลดลงจากเดิมเพราะต้องขับกระแสมากขึ้น ปัจจุบันมี LED ย่านอินฟาเรดรุ่นใหม่ที่ให้กำลังส่งหรือความเข้มแสงสูงช่วยให้ส่งสัญญาณได้ไกลกว่าเดิมมาก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.1 การไบอัสอินฟาเรด

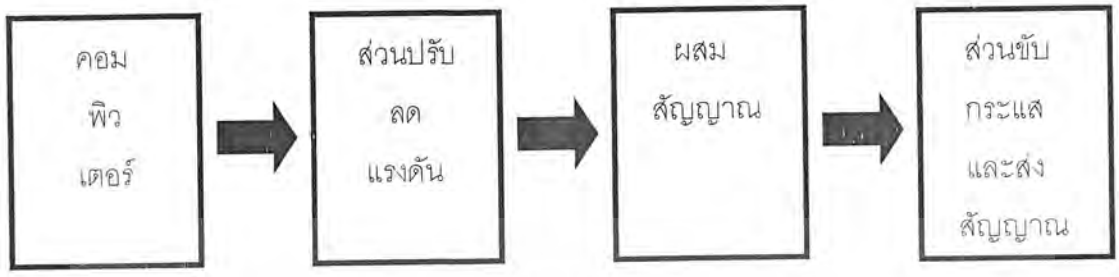
ข้อมูลที่จะส่งออกจากคอมพิวเตอร์ จะอยู่ในรูปข้อมูลอนุกรมซึ่งในการส่ง 1 ครั้งจะมีข้อมูลทั้งหมด 10 บิต คือ บิตเริ่มต้น (Start Bit) 1 บิต บิตข้อมูล (Data Bit) 8 บิต และบิตสุดท้าย (Stop Bit) 1 บิต ซึ่งจะแสดงดังรูปที่ 3.2



รูปที่ 3.2 รูปแบบการส่งข้อมูลแบบอนุกรม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.2 ตัวส่งสัญญาณอินฟราเรด



รูปที่ 3.3 บล๊อกไดอะแกรมแสดงการทำงานของส่วนส่งสัญญาณอินฟราเรด

3.2.1 หลักการทำงาน

ส่วนส่งสัญญาณจะเป็นส่วนของการส่งสัญญาณอินฟราเรดออกไป ซึ่งจะมีการทำงานหลักๆ ดังบล๊อกไดอะแกรมรูปที่ 3.3 คอมพิวเตอร์จะเป็นส่วนอินเทอร์เฟส(Interface) กับผู้ใช้งาน (Users) เพื่อกำหนดลักษณะของสัญญาณที่จะส่ง ซึ่งสัญญาณจะถูกส่งออกจากคอมพิวเตอร์ในลักษณะของข้อมูลอนุกรมเพื่อให้ง่ายต่อการรับส่ง(ใช้สายสัญญาณเพียงสายเดียว) สัญญาณจะส่งผ่านพอร์ตอนุกรม (Serial Port)ในมาตรฐาน RS-232 ขนาดสัญญาณเอาท์พุทของพอร์ตอนุกรมจะมีแรงดันจากยอดถึงยอด (Peak to Peak) ประมาณ 24 โวลท์ หรือ ±12 โวลท์ มี I/O Address ที่ 3F8H-3FFH ของพอร์ต COM1 และที่ 2F8H-2FFH ของพอร์ต COM2

สัญญาณเมื่อออกจากคอมพิวเตอร์ทางพอร์ตอนุกรมแล้วจะไปเข้าส่วนปรับลดแรงดัน เนื่องจากสัญญาณที่มาจากพอร์ตอนุกรมมีขนาดยอดถึงยอด 24 โวลท์ หากนำมาจ่ายโดยตรงให้กับ IC ประเภท TTL ก็อาจจะส่งผลให้ IC นั้นเสียได้จึงต้องมีการปรับลดแรงดันลงจนเหลือ 0-5 โวลท์ซึ่งเหมาะสมที่จะใช้กับ IC TTL ซึ่งการปรับลดแรงดันจะใช้ IC เบอร์ MAX232E เป็นส่วนลดแรงดันลง สัญญาณที่ได้จากการลดแรงดันจะนำไปผสมกับสัญญาณที่ได้จากออสซิลเลเตอร์ เพื่อให้เกิดเป็นสัญญาณโทนเบริสต์(จะกล่าวในหัวข้อถัดไป) สัญญาณจากออสซิลเลเตอร์จะใช้สัญญาณคลื่นสี่เหลี่ยม (Square Wave) ที่มีความถี่ประมาณ 38 kHz เพื่อให้ตัวรับรับสัญญาณได้พอดี ซึ่งจะใช้ IC555 เป็นตัวกำเนิดสัญญาณนี้สัญญาณโทนเบริสต์ที่ได้จะถูกส่งไปยังส่วนส่งสัญญาณและขับกระแสให้สูงขึ้นเพื่อที่จะทำให้ส่งได้เป็นระยะทางไกลๆ

3.2.2 การควบคุมด้วยสัญญาณโทนเบิรสต์

เป็นการดัดแปลงรูปแบบของสัญญาณให้มีลักษณะคล้ายกับการส่งคลื่นวิทยุเพื่อป้องกันการรบกวนจากสิ่งแวดล้อมหรือจากเครื่องส่งอื่นที่อยู่บริเวณใกล้เคียงหากมีรูปแบบการเข้ารหัสเหมือนกัน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ลักษณะสัญญาณแบบโทนเบิรสต์แสดงดังรูปที่ 3.4 ซึ่งสร้างจากสัญญาณพัลส์ที่มีสภาวะสูงของสัญญาณควบคุมแบบธรรมดา สัญญาณโทนเบิรสต์ประกอบด้วยพัลส์ความถี่สูงแบบต่อเนื่องตลอดช่วงความกว้างของบิทที่เป็น 1 ในขณะที่บิทข้อมูลอยู่ในสภาวะต่ำสัญญาณจะคงเดิมไม่มีการเปลี่ยนแปลงแต่อย่างใด ขั้นตอนของการเกิดสัญญาณโทนเบิรสต์แสดงดังรูป 3.5

ลักษณะของสัญญาณโทนเบริสต์แสดงดังรูปที่ 3.4

รูปที่ 3.4 ลักษณะสัญญาณแบบโทนเบิรสต์

ซึ่งการกำเนิดสัญญาณโทนเบริสต์ทำได้ดังนี้

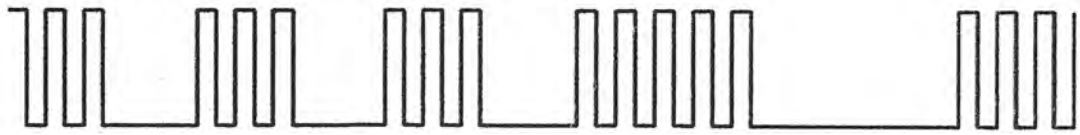
สมมุติสัญญาณที่ออกจาก TXD เป็น 35H

(ก)

สัญญาณจากออสซิลเลเตอร์ความถี่ประมาณ 38 kHz

(ข)

เมื่อนำสัญญาณทั้งคู่มารวมกันจะได้เป็นสัญญาณโทนเบริสต์



(ค)

รูปที่ 3.5 แสดงขั้นตอนการเกิดสัญญาณโทนเบิรสต์

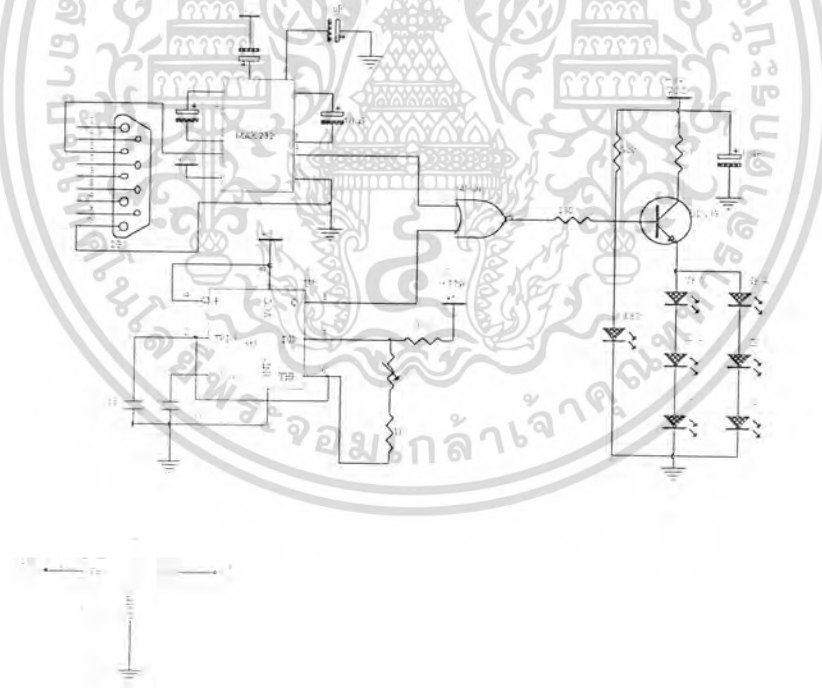
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สำหรับวงจรตัวรับต้องทำการออกแบบให้มีการตอบสนองต่อสัญญาณโทนเบิรสต์ในช่วงความถี่ที่เหมาะสมกันกับวงจรตัวส่ง

ข้อดีอีกประการของการส่งสัญญาณแบบโทนเบิรสต์ ก็คือสามารถขจัดสัญญาณรบกวนจากภายนอกได้ดีมาก ตัวอย่างเช่น ระบบควบคุมที่ใช้สัญญาณแสง หรือแสงอินฟาเรดที่ส่งด้วยพัลส์ธรรมดาอาจมีแสงจากภายนอก แสงจากหลอดไฟ หรือแม้กระทั่งแสงอาทิตย์ที่เข้าไปรบกวนที่เครื่องรับจนอาจเกิดข้อผิดพลาดในการรับสัญญาณควบคุมได้ แต่ถ้าหากมีการใช้สัญญาณโทนเบิรสต์แล้ว สิ่งรบกวนต่างๆ เหล่านี้จะถูกขจัดออกไปโดยสิ้นเชิง

3.2.3 วงจรตัวส่งข้อมูลอินฟาเรด

วงจรตัวส่งแสดงไว้ในรูปที่ 3.6

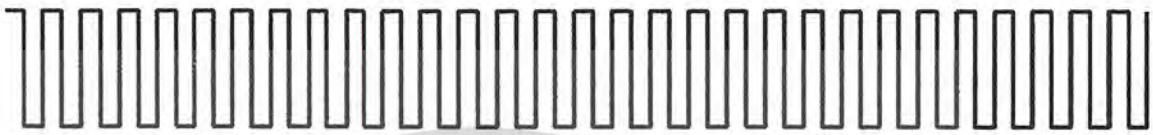


รูปที่ 3.6 วงจรตัวส่งอินฟาเรด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

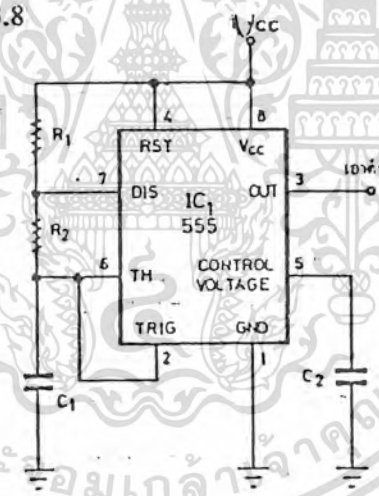
3.2.4 การออกแบบสัญญาณแคเรียร์ 38 kHz

เราใช้ไอซีเบอร์ 555 ซึ่งต่อเป็นวงจรอะสเตเบิลมัลติไวเบรเตอร์เพื่อสร้างเป็นสัญญาณแคเรียร์ 38 kHz ซึ่งรูปแบบของความถี่นั้นจะอยู่ในรูปของสัญญาณพัลส์สี่เหลี่ยม ที่มีความกว้างของพัลส์ดังรูปที่ 3.7



รูปที่ 3.7 สัญญาณที่ได้จากเอาท์พุทของอะสเตเบิลมัลติไวเบรเตอร์

จากรูปจะเห็นว่าช่วงเวลาของ t_1 และ t_2 เราสามารถที่จะกำหนดได้ว่าให้มีความกว้างเป็นเท่าไร โดยคิดเป็นเปอร์เซ็นต์ เช่นเราให้ความกว้างของ $t_1 = t_2$ นั่นคือ ดิวตี้ไซเคิ่ลเท่ากับ 50% การออกแบบเราสามารถออกแบบได้ดังรูปที่ 3.8



รูปที่ 3.8 แสดงวงจรอะสเตเบิลมัลติไวเบรเตอร์ 38 kHz

จากสมการ $t_1 + t_2 = T = 0.693 (R_1 + 2R_2) C$

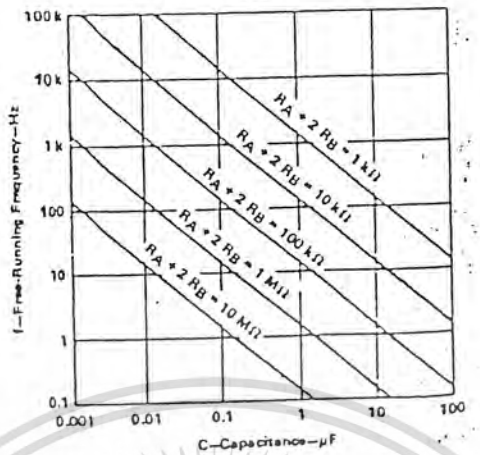
และ $F = 1.44 / (R_1 + 2R_2) C$

แทนค่า $F = 38$ kHz จะได้

$38 \times 10^3 = 1.44 / (R_1 + 2R_2) C$

จากสมการข้างต้นเรามี ตัวแปร 2 ตัวคือ R และ C การออกแบบเราจะกำหนดค่าของ C ก่อน เพราะมีค่าเฉพาะและหาได้จากตารางความสัมพันธ์ตารางที่ 3.1

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



ตารางที่ 3.1 ตารางแสดงความสัมพันธ์ของค่า R และ C และความถี่

จากตารางเลือกค่า $C = 0.01\mu F$

จะได้ $38 \times 10^{3} = 1.44 / (R_1 + 2R_2) \times 0.01 \times 10^{-6}$

$R_1 + 2R_2 = 1.44 / (38 \times 10^{3} \times 0.01 \times 10^{-6})$

$= 3.8\ k\Omega$

ให้ $R_1 = 220\ \Omega$

$220 + 2R_2 = 3.8 \times 10^{3}$

$R_2 = (3.8 \times 10^{3} - 220) / 2$

$= 1.79\ k\Omega$

ค่าดิวตี้ไซเคิล $D = (R_1 + R_2) / (R_1 + 2R_2)$

$= (220 + 1.79 \times 10^{3}) / (220.2 \times 1.79 \times 10^{3})$

$= 53\ \%$

3.2.6 รหัสสัญญาณจากตัวส่ง

สัญญาณจากภาคส่งซึ่งจะส่งไปยังภาครับเพื่อนำไปควบคุมรถให้วิ่งไปในทิศทางที่กำหนด ในโครงงานนี้จะใช้สัญญาณ 4 สัญญาณ คือ

$(255)_{10}$ หรือ $(11111111)_2$ ทำให้รถวิ่งเดินหน้า

$(15)_{10}$ หรือ $(00001111)_2$ ทำให้รถวิ่งถอยหลัง

$(63)_{10}$ หรือ $(00111111)_2$ ทำให้รถเลี้ยวซ้าย

(3)10 หรือ $(00000011)_2$ ทำให้รถเลี้ยวขวา

$(240)_{10}$ หรือ $(11110000)_2$ ทำให้รถหยุด

สัญญาณที่ถูกส่งจากตัวส่งจะมีบอดเรต เป็น 1200 ซึ่งจะส่งและรับได้ดีที่สุด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 4
ตัวรับข้อมูลแบบอินฟาเรด

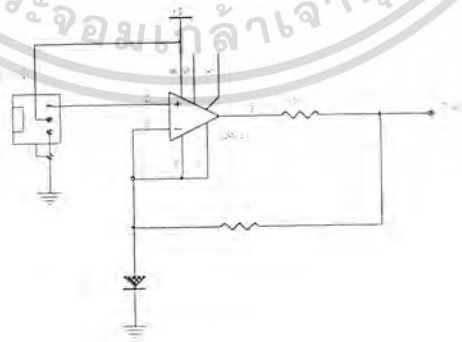
4.1 การออกแบบวงจรภาครับ

วงจรภาครับแสดงดังรูปที่ 4.1 ในการออกแบบวงจรภาครับก็เหมือนกับการออกแบบวงจรภาคส่ง คือจะต้องมีส่วนของ ตัวรับแสงอินฟาเรด และแปลงเป็นสัญญาณไฟฟ้า การดีมอดูเลทสัญญาณ และการออกแบบวงจรภาครับจะต้องให้สัมพันธ์กับภาคส่ง คือ สามารถรับข้อมูลจากแสงอินฟาเรดที่ส่งมาให้มากที่สุด

4.1.1 อุปกรณ์รับแสงอินฟาเรดและดีมอดูเลทสัญญาณ

ในเรื่องของการส่งและรับแสงอินฟาเรดให้ได้ระยะทางไกลๆ และถูกต้องนั้นอยู่ที่ตัวอุปกรณ์ของตัวส่งและรับแสงอินฟาเรดให้ได้ไกลๆ และมีคุณสมบัติตรงกัน

เนื่องจากเราส่งสัญญาณข้อมูลมาพร้อมกับสัญญาณแคเรียร์ 38 kHz ดังนั้นการดีมอดูเลทสัญญาณ จะต้องดีมอดูเลทที่ 38 kHz ด้วยเช่นกัน เพื่อให้การสื่อสารเป็นไปอย่างถูกต้อง และส่งได้ในระยะไกล และมีการรบกวนจากสิ่งแวดล้อมภายนอกให้น้อยที่สุด

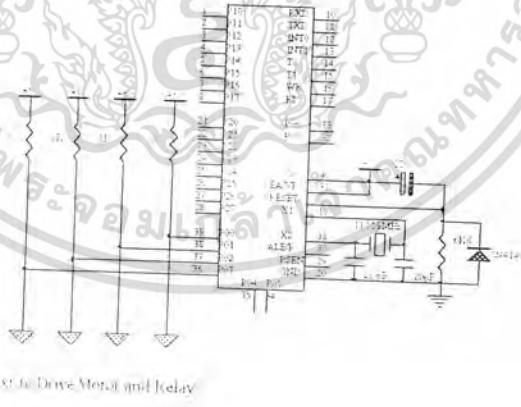


รูปที่ 4.1 วงจรตัวรับอินฟาเรด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.1.2 หลักการทำงาน

การทำงานของวงจรภาครับที่สำคัญคือโมดูลตัวรับหรือ Receiver Module เป็นส่วนรับสัญญาณอินฟราเรดที่สำเร็จรูป มีความสามารถหลายอย่างคือ มีวงจรขยายสัญญาณอยู่ภายในตัว มีแบนด์พาสฟิลเตอร์ ดีมอดดูเลเตอร์ เป็นต้น โมดูลตัวรับจะรับสัญญาณดิจิตอลที่ถูกมอดดูเลทด้วยความถี่ประมาณ 38 kHz เท่านั้น ซึ่งจะไม่รับสัญญาณจากแหล่งอื่นที่มีความถี่ไม่ตรงกับ 38 kHz หลังจากได้สัญญาณจากโมดูลตัวรับเรียบร้อยแล้วจึงนำไปผ่านตัวเปรียบเทียบสัญญาณ (Comparator) เบอร์ LM311 ซึ่งให้ แรงดันอ้างอิง (Vref) เป็น 0.7 โวลต์ ซึ่งจะทำให้ได้สัญญาณดิจิตอลขนาด 0 และ 5 โวลต์ ตามที่ต้องการ หลังจากนั้นสัญญาณที่ได้ก็นำเข้าสู่ไมโครคอนโทรลเลอร์ 8051 ที่ขา 10 ซึ่งเป็นขารับส่งข้อมูลอนุกรม เมื่อไมโครคอนโทรลเลอร์รับสัญญาณเข้ามาแล้วก็จะมาตรวจสอบสัญญาณว่าเป็นสัญญาณควบคุมที่ส่งมาหรือไม่ ถ้าตรงกับสัญญาณควบคุมที่ตั้งไว้ก็จะส่งสัญญาณควบคุมไปที่รีเลย์ผ่านทางพอร์ต P0.6,P0.7 และวงจรไดรว์มอร์เตอร์ผ่านทางพอร์ต P0.4,P0.5

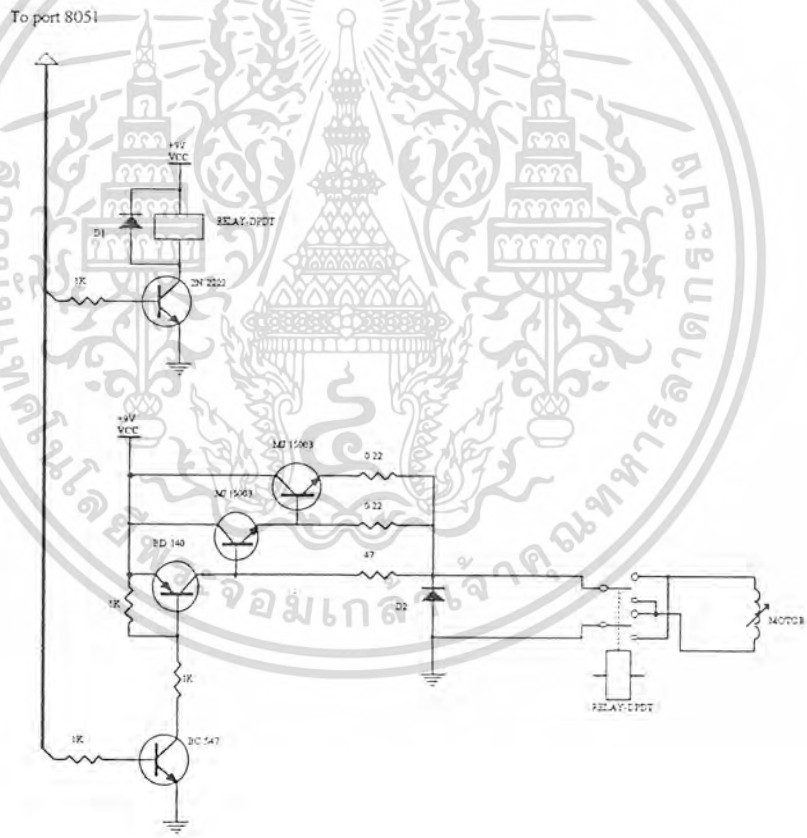


รูปที่ 4.2 วงจรไมโครคอนโทรลเลอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.2 วงจรภาคขับมอร์เตอร์และรีเลย์

จากการที่ออกแบบโครงสร้างของตัวรถให้มีระบบขับเคลื่อนโดยใช้รีเลย์เป็นตัวควบคุมให้เกิดการเดินหน้า ถอยหลัง เลี้ยวซ้ายและเลี้ยวขวา และเพื่อให้สามารถประยุกต์ใช้งานกับโปรแกรมในการควบคุมระบบการเคลื่อนที่ของตัวรถ จึงต้องอาศัยข้อมูลขนาด 4 บิตเพื่อควบคุมการทำงานของตัวรถ



รูป 4.3 แสดงวงจรภาคขับมอร์เตอร์และรีเลย์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากรูป 4.3 สามารถอธิบายการทำงานได้โดยเริ่มจากสัญญาณคำสั่งขนาด 4 บิตจะถูกส่งมาจากไมโครคอนโทรลเลอร์โดยใช้พอร์ตศูนย์ตำแหน่งบิตที่ P0.4 ถึง P0.7 โดยพอร์ต P0.4 และ P0.5 จะต่อไปยังภาคไดร์วมอร์เตอร์ซึ่งเมื่อมีสัญญาณเป็นลอจิก 1 ก็จะทำให้ภาคไดรว์มอร์เตอร์ทำงาน โดยทรานซิสเตอร์ BC 547 จะทำงาน (ON) ทำให้ทรานซิสเตอร์ BD140 ทำงานต่อจากนั้น ทรานซิสเตอร์ MJ15003 นำกระแส พร้อมที่จะขับมอร์เตอร์ ส่วนพอร์ต P0.6 และ P0.7 จะต่อไปยังชุดควบคุมรีเลย์ เมื่อมีสัญญาณลอจิก 1 ก็จะทำให้ทรานซิสเตอร์ 2N2222 นำกระแสซึ่งจะทำให้รีเลย์ทำงานควบคุมทิศทางการหมุนของมอร์เตอร์

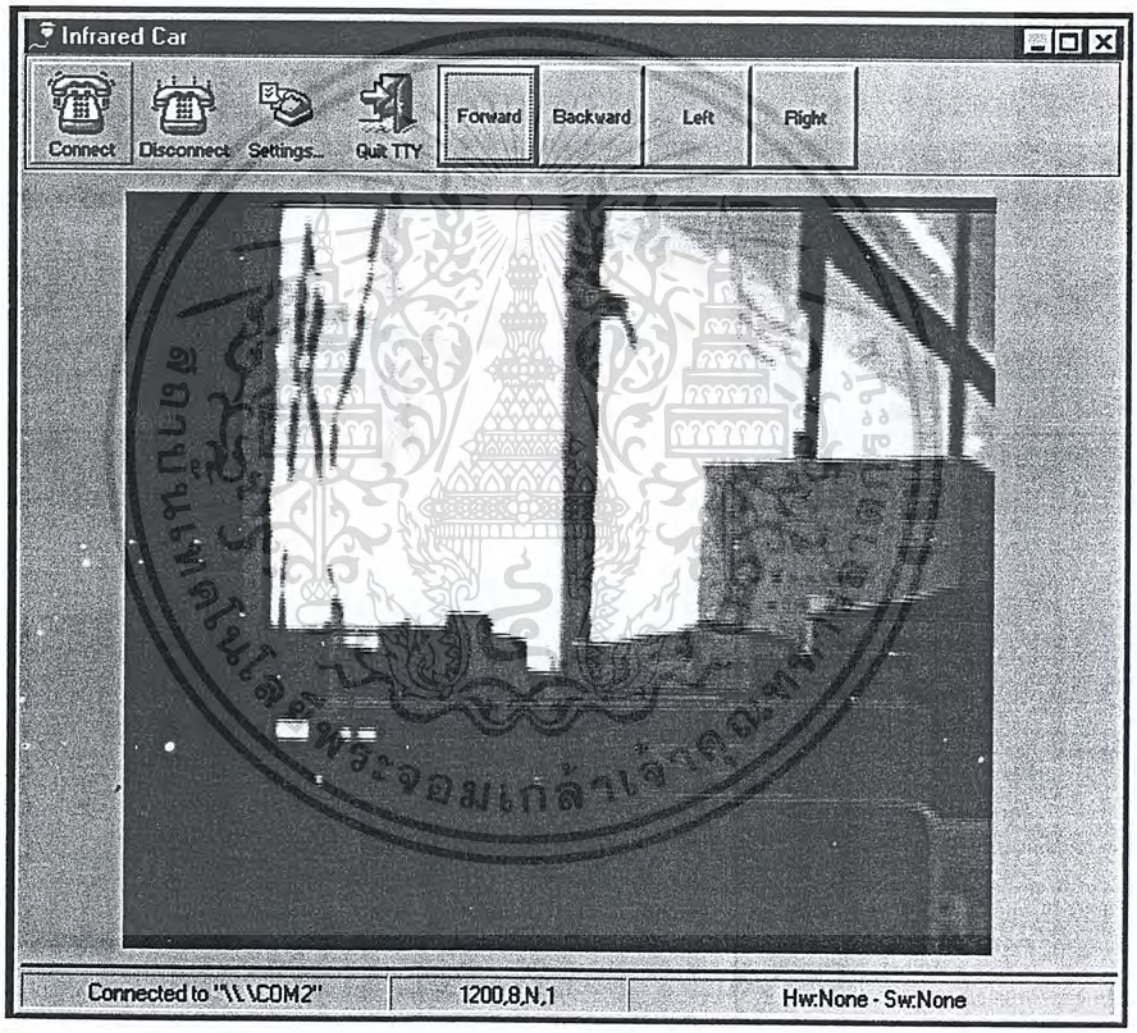
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สำนักหอสมุดกลาง พระจอมเกล้าลาดกระบัง

บทที่ 5

การทดลองและผลการทดลอง

ภาพหน้าจอขณะที่โปรแกรมกำลังทำงาน



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

42652

การทดลอง

1.หาช่วงระยะของการรับส่งข้อมูล

1.1 นำตัวส่งตั้งอยู่ที่ระยะคงที่ห่างจากคอมพิวเตอร์ 1 เมตร และตั้งตัวรับและไมโครคอนโทรลเลอร์ห่างจากตัวส่งเป็นระยะ 1/2 เมตร ทดสอบว่ารับข้อมูลได้ถูกต้องหรือไม่

1.2 ขยับตัวรับให้ห่างจากตัวส่งเพิ่มขึ้นครั้งละ 1/2 เมตรในแนวตรงไม่มีการเอียง แล้วทดสอบว่ารับข้อมูลได้ถูกต้องหรือไม่

2.หามุมการรับ

2.1 นำตัวส่งตั้งอยู่ที่ระยะคงที่ห่างจากคอมพิวเตอร์ 1 เมตร และตั้งตัวรับและไมโครคอนโทรลเลอร์ห่างจากตัวส่งเป็นระยะ 1 เมตร ทดสอบว่ารับข้อมูลได้ถูกต้องหรือไม่

2.2 ขยับตัวรับทำมุมกับตัวส่ง 10 องศา โดยไม่เพิ่มระยะห่างจากตัวส่ง แล้วทดสอบว่ายังรับข้อมูลได้อย่างถูกต้องหรือไม่

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ผลการทดลอง

1.หาช่วงระยะของการรับส่งข้อมูล

ระยะห่างจากตัวส่ง (เมตร)	การรับข้อมูล
0.5	รับข้อมูลได้อย่างถูกต้อง
1	รับข้อมูลได้อย่างถูกต้อง
1.5	รับข้อมูลได้อย่างถูกต้อง
2	รับข้อมูลได้อย่างถูกต้อง
2.5	รับข้อมูลได้อย่างถูกต้อง
3	รับข้อมูลได้อย่างถูกต้อง
3.5	รับข้อมูลได้อย่างถูกต้อง
4	รับข้อมูลได้อย่างถูกต้อง
4.5	รับข้อมูลมีความผิดพลาดบ้าง
5	รับข้อมูลผิดพลาด
5.5	ไม่สามารถรับข้อมูลได้เลย

2.หามุมการรับ

มุม	การรับข้อมูล
0	รับข้อมูลได้อย่างถูกต้อง
10	รับข้อมูลได้อย่างถูกต้อง
20	รับข้อมูลได้อย่างถูกต้อง
30	รับข้อมูลได้อย่างถูกต้อง
40	รับข้อมูลผิดพลาด
50	ไม่สามารถรับข้อมูลได้เลย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 6

สรุปและวิจารณ์การทดลอง

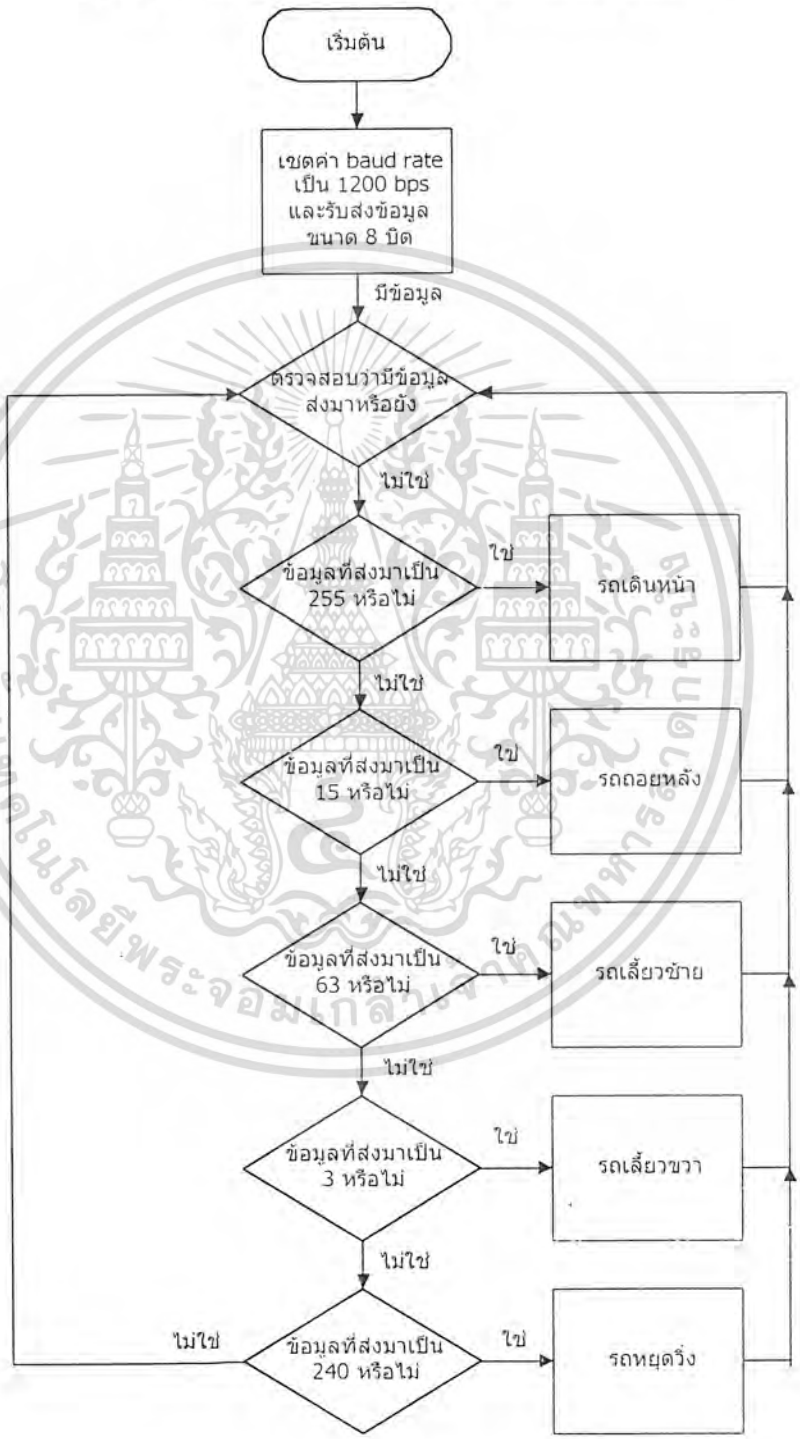
การส่งข้อมูลดิจิตอลไร้สายแบบอินฟราเรดจากคอมพิวเตอร์ผ่านพอร์ตอนุกรมมีเครื่องส่งและเครื่องรับเป็นหัวใจในการทำงาน การส่งจะส่งข้อมูลในลักษณะโทนเบริสต์ ซึ่งคล้ายกับการมอดดูเลทสัญญาณกับพาหะแล้วส่งไปยังตัวรับ เพื่อนำสัญญาณไปใช้ในการควบคุมรถผ่านไมโครคอนโทรลเลอร์ เบอร์ 8051

จากผลการทดลองจะเห็นได้ว่ารถจะสามารถเคลื่อนที่ได้อย่างถูกต้องในระยะประมาณ 5 เมตร ส่วนมุมในการรับส่งก็จะรับได้ไม่เกิน 45 องศา รวมทั้งการถ่ายทอดสัญญาณยังคงใช้สายจากกล้องต่อมายังเครื่องคอมพิวเตอร์ทำให้ไม่สะดวกเท่าที่ควร

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก

Flow Chat การทำงานของไมโครคอนโทรลเลอร์ 8051



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โปรแกรมไมโครคอนโทรลเลอร์ 8051

```
ORG 0000H
INITIAL:MOV SCON,#50H
    MOV TMOD,#20H
    MOV TH1,#0E8H
    SETB TR1
    MOV IE,#10001010B
    MOV P0,#000H   ;INITIAL PORT 0
    MOV P1,#0FFH   ;INITIAL PORT 1
    MOV P2,#0FFH   ;INITIAL PORT 2
;-------------MAIN PROGRAM----------------
MAIN:
    LCALL CHECKRX
    SJMP MAIN
CHECKRX:
    JBC RI,RECV
    RET
RECV:
    MOV A,SBUF
    CJNE A,#11111111B,REC2
    SJMP FORWARD
    RET
REC2:
    CJNE A,#00001111B,REC3
    SJMP BACKWARD
    RET
REC3:
    CJNE A,#00111111B,REC4
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    SJMP LEFT
    RET
REC4:
    CJNE A,#00000011B,REC5
    SJMP RIGHT
    RET
REC5:
    CJNE A,#11110000B,REC6
    SJMP STOP
REC6:  RET

;----------------SUBROUTINE---------

FORWARD:
    SETB P0.4
    SETB P0.5
    SETB P0.6
    SETB P0.7

    SETB P0.0
    SETB P0.1
    SETB P0.2
    SETB P0.3

    RET
BACKWARD:
    SETB P0.4
    SETB P0.5
    CLR P0.6
    CLR P0.7
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    SETB P0.0
    SETB P0.1
    CLR P0.2
    CLR P0.3

    RET
STOP:
    CLR P0.4
    CLR P0.5
    SETB P0.6
    SETB P0.7

    CLR P0.0
    CLR P0.1
    SETB P0.2
    SETB P0.3

    RET
LEFT:
    SETB P0.4
    SETB P0.5
    SETB P0.6
    CLR P0.7

    SETB P0.0
    SETB P0.1
    SETB P0.2
   CLR P0.3

    RET
```

เอกสารนี้เป็นการที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
RIGHT:
    SETB P0.4
    SETB P0.5
    CLR P0.6
    SETB P0.7

    SETB P0.0
    SETB P0.1
    CLR P0.2
    SETB P0.3

    RET
END
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โปรแกรม **Infrared remote control car** ซึ่งเขียนโดยเดลไฟล์

```
unit vfw;

interface

uses
  Windows, MMSystem, Messages;


const
    // ------------------------------------------------------------
    // Window Messages  WM_CAP... which can be sent to an AVICAP window
    // ------------------------------------------------------------

    // Defines start of the message range
    WM_CAP_START                = WM_USER;

    WM_CAP_GET_CAPSTREAMPTR     = (WM_CAP_START+ 1);
    WM_CAP_SET_CALLBACK_ERROR   = (WM_CAP_START+ 2);
    WM_CAP_SET_CALLBACK_STATUS  = (WM_CAP_START+ 3);
    WM_CAP_SET_CALLBACK_YIELD   = (WM_CAP_START+ 4);
    WM_CAP_SET_CALLBACK_FRAME   = (WM_CAP_START+ 5);
    WM_CAP_SET_CALLBACK_VIDEOSTREAM = (WM_CAP_START+ 6);
    WM_CAP_SET_CALLBACK_WAVESTREAM = (WM_CAP_START+ 7);
    WM_CAP_GET_USER_DATA        = (WM_CAP_START+ 8);
    WM_CAP_SET_USER_DATA        = (WM_CAP_START+ 9);

    WM_CAP_DRIVER_CONNECT       = (WM_CAP_START+ 10);
    WM_CAP_DRIVER_DISCONNECT    = (WM_CAP_START+ 11);
    WM_CAP_DRIVER_GET_NAME      = (WM_CAP_START+ 12);
    WM_CAP_DRIVER_GET_VERSION   = (WM_CAP_START+ 13);
    WM_CAP_DRIVER_GET_CAPS      = (WM_CAP_START+ 14);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
WM_CAP_FILE_SET_CAPTURE_FILE   =(WM_CAP_START+ 20);
WM_CAP_FILE_GET_CAPTURE_FILE   =(WM_CAP_START+ 21);
WM_CAP_FILE_ALLOCATE           =(WM_CAP_START+ 22);
WM_CAP_FILE_SAVEAS             =(WM_CAP_START+ 23);
WM_CAP_FILE_SET_INFOCHUNK      =(WM_CAP_START+ 24);
WM_CAP_FILE_SAVEDIB            =(WM_CAP_START+ 25);

WM_CAP_EDIT_COPY               =(WM_CAP_START+ 30);

WM_CAP_SET_AUDIOFORMAT         =(WM_CAP_START+ 35);
WM_CAP_GET_AUDIOFORMAT         =(WM_CAP_START+ 36);

WM_CAP_DLG_VIDEOFORMAT         =(WM_CAP_START+ 41);
WM_CAP_DLG_VIDEOSOURCE         =(WM_CAP_START+ 42);
WM_CAP_DLG_VIDEODISPLAY        =(WM_CAP_START+ 43);
WM_CAP_GET_VIDEOFORMAT         =(WM_CAP_START+ 44);
WM_CAP_SET_VIDEOFORMAT         =(WM_CAP_START+ 45);
WM_CAP_DLG_VIDEOCOMPRESSION    =(WM_CAP_START+ 46);

WM_CAP_SET_PREVIEW             =(WM_CAP_START+ 50);
WM_CAP_SET_OVERLAY             =(WM_CAP_START+ 51);
WM_CAP_SET_PREVIEWRATE         =(WM_CAP_START+ 52);
WM_CAP_SET_SCALE               =(WM_CAP_START+ 53);
WM_CAP_GET_STATUS              =(WM_CAP_START+ 54);
WM_CAP_SET_SCROLL              =(WM_CAP_START+ 55);

WM_CAP_GRAB_FRAME              =(WM_CAP_START+ 60);
WM_CAP_GRAB_FRAME_NOSTOP       =(WM_CAP_START+ 61);

WM_CAP_SEQUENCE                =(WM_CAP_START+ 62);
WM_CAP_SEQUENCE_NOFILE         =(WM_CAP_START+ 63);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        WM_CAP_SET_SEQUENCE_SETUP      = (WM_CAP_START+ 64);
        WM_CAP_GET_SEQUENCE_SETUP      = (WM_CAP_START+ 65);
        WM_CAP_SET_MCI_DEVICE          = (WM_CAP_START+ 66);
        WM_CAP_GET_MCI_DEVICE          = (WM_CAP_START+ 67);
        WM_CAP_STOP                    = (WM_CAP_START+ 68);
        WM_CAP_ABORT                   = (WM_CAP_START+ 69);

        WM_CAP_SINGLE_FRAME_OPEN       = (WM_CAP_START+ 70);
        WM_CAP_SINGLE_FRAME_CLOSE      = (WM_CAP_START+ 71);
        WM_CAP_SINGLE_FRAME            = (WM_CAP_START+ 72);

        WM_CAP_PAL_OPEN                = (WM_CAP_START+ 80);
        WM_CAP_PAL_SAVE                = (WM_CAP_START+ 81);
        WM_CAP_PAL_PASTE               = (WM_CAP_START+ 82);
        WM_CAP_PAL_AUTOCREATE          = (WM_CAP_START+ 83);
        WM_CAP_PAL_MANUALCREATE        = (WM_CAP_START+ 84);

                // Following added post VFW 1.1
        WM_CAP_SET_CALLBACK_CAPCONTROL = (WM_CAP_START+ 85);

        // Defines end of the message range
        WM_CAP_END                     = WM_CAP_SET_CALLBACK_CAPCONTROL;

// dwFlags field of TVIDEOHDR
VHDR_DONE =     $00000001;  // Done bit */
VHDR_PREPARED=  $00000002;  // Set if this header has been prepared */
VHDR_INQUEUE =  $00000004;  // Reserved for driver */
VHDR_KEYFRAME=  $00000008;  // Key Frame */

//------------------------------------------------------------------------------
//  Structures
//------------------------------------------------------------------------------
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
type
 PCapDriverCaps = ^TCapDriverCaps;
 TCapDriverCaps = record
   wDeviceIndex          :WORD;        // Driver index in system.ini
   fHasOverlay           :BOOL;        // Can device overlay?
   fHasDlgVideoSource    :BOOL;        // Has Video source dlg?
   fHasDlgVideoFormat    :BOOL;        // Has Format dlg?
   fHasDlgVideoDisplay   :BOOL;        // Has External out dlg?
   fCaptureInitialized   :BOOL;        // Driver ready to capture?
   fDriverSuppliesPalettes :BOOL;      // Can driver make palettes?
   hVideoIn              :THANDLE;     // Driver In channel
   hVideoOut             :THANDLE;     // Driver Out channel
   hVideoExtIn           :THANDLE;     // Driver Ext In channel
   hVideoExtOut          :THANDLE;     // Driver Ext Out channel
 end;


  pCapStatus = ^TCapStatus;
  TCapStatus = record
   uiImageWidth          :UINT;    // Width of the image
   uiImageHeight         :UINT;    // Height of the image
   fLiveWindow           :BOOL;    // Now Previewing video?
   fOverlayWindow        :BOOL;    // Now Overlaying video?
   fScale                :BOOL;    // Scale image to client?
   ptScroll              :TPOINT;  // Scroll position
   fUsingDefaultPalette  :BOOL;    // Using default driver palette?
   fAudioHardware        :BOOL;    // Audio hardware present?
   fCapFileExists        :BOOL;    // Does capture file exist?
   dwCurrentVideoFrame   :DWORD;   // # of video frames cap'td
   dwCurrentVideoFramesDropped :DWORD;  // # of video frames dropped
   dwCurrentWaveSamples  :DWORD;   // # of wave samples cap'td
```

เอกสารนี้มีลิขสิทธิ์สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
  dwCurrentTimeElapsedMS    :DWORD;    // Elapsed capture duration
  hPalCurrent               :HPALETTE; // Current palette in use
  fCapturingNow             :BOOL;     // Capture in progress?
  dwReturn                  :DWORD;    // Error value after any operation
  wNumVideoAllocated        :UINT;     // Actual number of video buffers
  wNumAudioAllocated        :UINT;     // Actual number of audio buffers
end;


pCaptureParms = ^TCaptureParms;
TCaptureParms = record                // Default values in parenthesis
  dwRequestMicroSecPerFrame :DWORD;   // Requested capture rate
  fMakeUserHitOKToCapture   :BOOL;    // Show "Hit OK to cap" dlg?
  wPercentDropForError      :UINT;    // Give error msg if > (10%)
  fYield                    :BOOL;    // Capture via background task?
  dwIndexSize               :DWORD;   // Max index size in frames (32K)
  wChunkGranularity         :UINT;    // Junk chunk granularity (2K)
  fUsingDOSMemory           :BOOL;    // Use DOS buffers?
  wNumVideoRequested        :UINT;    // # video buffers, If 0, autocalc
  fCaptureAudio             :BOOL;    // Capture audio?
  wNumAudioRequested        :UINT;    // # audio buffers, If 0, autocalc
  vKeyAbort                 :UINT;    // Virtual key causing abort
  fAbortLeftMouse           :BOOL;    // Abort on left mouse?
  fAbortRightMouse          :BOOL;    // Abort on right mouse?
  fLimitEnabled             :BOOL;    // Use wTimeLimit?
  wTimeLimit                :UINT;    // Seconds to capture
  fMCIControl               :BOOL;    // Use MCI video source?
  fStepMCIDevice            :BOOL;    // Step MCI device?
  dwMCIStartTime            :DWORD;   // Time to start in MS
  dwMCIStopTime             :DWORD;   // Time to stop in MS
  fStepCaptureAt2x          :BOOL;    // Perform spatial averaging 2x
  wStepCaptureAverageFrames :UINT;    // Temporal average n Frames
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
  dwAudioBufferSize       :DWORD;    // Size of audio bufs (0 = default)
  fDisableWriteCache      :BOOL;     // Attempt to disable write cache
  AVStreamMaster          :UINT;     // Which stream controls length?
end;

PCapInfoChunk = ^TCapInfoChunk;
TCapInfoChunk = record
  fccInfoID :FOURCC;              // Chunk ID, "ICOP" for copyright
  lpData    :Pointer;             // pointer to data
  cbData    :LongInt;             // size of lpData
end;

PVIDEOHDR = ^TVIDEOHDR;
TVIDEOHDR = record
  lpData:pByte;                   // pointer to locked data buffer
  dwBufferLength:DWORD;           // Length of data buffer
  dwBytesUsed:DWORD;              // Bytes actually used
  dwTimeCaptured:DWORD;           // Milliseconds from start of stream
  dwUser:DWORD;                   // for client's use
  dwFlags:DWORD;                  // assorted flags (see defines)
  dwReserved: array [0..4] of DWORD;   // reserved for driver
end;


// ------------------------------------------------------------------
// Callback Definitions
// ------------------------------------------------------------------
type
    TCAPSTATUSCALLBACK = function(hWnd:HWND; nID:Integer;
lpsz:PChar):LongInt; stdcall;
    TCAPYIELDCALLBACK = function(hWnd:HWND):LongInt; stdcall;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        TCAPERRORCALLBACK   = function(hWnd:HWND; nID:Integer;
lpsz:Pchar):LongInt; stdcall;
        TCAPVIDEOSTREAMCALLBACK = function(hWnd:HWND;
lpVHdr:PVIDEOHDR):LongInt; stdcall;
        TCAPWAVESTREAMCALLBACK   = function(hWnd:HWND;
lpWHdr:PWAVEHDR):LongInt; stdcall;
        TCAPCONTROLCALLBACK = function(hWnd:HWND; nState:Integer):LongInt;
stdcall;
        // ------------------------------------------------------------------
        //  CapControlCallback states
        // ------------------------------------------------------------------
Const
        CONTROLCALLBACK_PREROLL     = 1;    // Waiting to start capture
        CONTROLCALLBACK_CAPTURING   = 2;   // Now capturing




        // ------------------------------------------------------------------
        //  Message crackers for above
        // ------------------------------------------------------------------
function capSetCallbackOnError (hwnd : THandle; fpProc:TCAPERRORCALLBACK):LongInt;
function capSetCallbackOnStatus(hwnd : THandle;
fpProc:TCAPSTATUSCALLBACK):LongInt;
function capSetCallbackOnYield (hwnd : THandle; fpProc:TCAPYIELDCALLBACK):LongInt;
function capSetCallbackOnFrame (hwnd : THandle;
fpProc:TCAPVIDEOSTREAMCALLBACK):LongInt;  // Hier ist der Type der Callbackfunktion
nicht klar !
function capSetCallbackOnVideoStream(hwnd:THandle;
fpProc:TCAPVIDEOSTREAMCALLBACK):LongInt;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

function capSetCallbackOnWaveStream (hwnd:THandle; fpProc:
TCAPWAVESTREAMCALLBACK):LongInt;
function capSetCallbackOnCapControl (hwnd:THandle;
fpProc:TCAPCONTROLCALLBACK):LongInt;

function capSetUserData(hwnd:THandle; lUser:LongInt):LongInt;
function capGetUserData(hwnd:THandle):LongInt;

function capDriverConnect(hwnd:THandle; I: Word) : boolean;
function capDriverDisconnect(hwnd:THandle):boolean;

function capDriverGetName(hwnd:THandle; szName:PChar; wSize:Word):boolean;
function capDriverGetVersion(hwnd:THandle; szVer:PChar; wSize:Word):Boolean;
function capDriverGetCaps(hwnd:THandle; s:PCapDriverCaps; wSize:Word):boolean;

function capFileSetCaptureFile(hwnd:THandle; szName:PChar):boolean;
function capFileGetCaptureFile(hwnd:THandle; szName:PChar; wSize:Word):boolean;
function capFileAlloc(hwnd:THandle; dwSize:DWORD):boolean;
function capFileSaveAs(hwnd:THandle; szName:Pchar):boolean;
function capFileSetInfoChunk(hwnd:THandle; lpInfoChunk:pCapInfoChunk):boolean ;
function capFileSaveDIB(hwnd:THandle; szName:Pchar):boolean;

function capEditCopy(hwnd : THandle):boolean;

function capSetAudioFormat(hwnd:THandle; s:PWaveFormatEx; wSize:Word):Boolean;
function capGetAudioFormat(hwnd:THandle; s:PWaveFormatEx; wSize:Word):DWORD;
function capGetAudioFormatSize(hwnd:THandle):DWORD;

function capDlgVideoFormat(hwnd:THandle):boolean;
function capDlgVideoSource(hwnd:THandle):boolean;
function capDlgVideoDisplay(hwnd:THandle):boolean;

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

function capDlgVideoCompression(hwnd:THandle):boolean;

function capGetVideoFormat(hwnd:THandle; s:pBitmapInfo; wSize:Word):DWord;

function capGetVideoFormatSize(hwnd:THandle):DWORD;

function capSetVideoFormat(hwnd:THandle; s:pBitmapInfo; wSize:Word):boolean;

function capPreview(hwnd:THandle; f:boolean):boolean;

function capPreviewRate(hwnd:THandle; wMS:Word):boolean;

function capOverlay(hwnd:THandle; f:boolean):boolean;

function capPreviewScale(hwnd:THandle; f:boolean):boolean;

function capGetStatus(hwnd:THandle; s:pCapStatus; wSize:Word):boolean;

function capSetScrollPos(hwnd:THandle; lpP:pPoint):boolean;

function capGrabFrame(hwnd:THandle):boolean;

function capGrabFrameNoStop(hwnd:THandle):boolean;

function capCaptureSequence(hwnd:THandle):Boolean;

function capCaptureSequenceNoFile(hwnd:THandle):Boolean;

function capCaptureStop(hwnd:THandle):boolean;

function capCaptureAbort(hwnd:THandle):boolean;

function capCaptureSingleFrameOpen(hwnd:THandle):boolean;

function capCaptureSingleFrameClose(hwnd:THandle):boolean;

function capCaptureSingleFrame(hwnd:THandle):boolean;

function capCaptureGetSetup(hwnd:THandle; s:pCaptureParms; wSize:Word):boolean;

function capCaptureSetSetup(hwnd:THandle; s:pCaptureParms; wSize:Word):boolean;

function capSetMCIDeviceName(hwnd:THandle; szName:PChar):boolean;

function capGetMCIDeviceName(hwnd:THandle; szName:PChar; wSize:Word):boolean;

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
function capPaletteOpen(hwnd:THandle; szName:PChar):boolean;
function capPaletteSave(hwnd:THandle; szName:PChar):boolean;
function capPalettePaste(hwnd:THandle):Boolean;
function capPaletteAuto(hwnd:THandle; iFrames:Word; iColors:word):boolean;
function capPaletteManual(hwnd:THandle; fGrab:Word; iColors:word):boolean;

    // ------------------------------------------------------------
    // The only exported functions from AVICAP.DLL
    // ------------------------------------------------------------
function capCreateCaptureWindow (
                lpszWindowName : PChar;
            dwStyle    : DWord;
                x, y       : Integer;
                nWidth, nHeight : Integer;
                hwndParent     : THandle;
            nID        : Integer ) : THandle; stdcall;

function capGetDriverDescription (
                    wDriverIndex : DWord;
                lpszName     : PChar;
                cbName       : Integer;
                lpszVer      : PChar;
                cbVer        : Integer ) : Boolean; stdcall;

    // ------------------------------------------------------------
    // New Information chunk IDs
    // ------------------------------------------------------------
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
(*
        infotypeDIGITIZATION_TIME = mmioStringToFOURCC(PChar('IDIT'),
MMIO_TOUPPER);
        infotypeSMPTE_TIME      = mmioStringToFOURCC(PChar('ISMP'),
MMIO_TOUPPER);
*)

        // --------------------------------------------------------------
        // String IDs from status and error callbacks
        // --------------------------------------------------------------
Const
        IDS_CAP_BEGIN           = 300; (* "Capture Start" *)
        IDS_CAP_END             = 301; (* "Capture End" *)

        IDS_CAP_INFO            = 401; (* "%s" *)
        IDS_CAP_OUTOFMEM        = 402; (* "Out of memory" *)
        IDS_CAP_FILEEXISTS      = 403; (* "File '%s' exists -- overwrite it?" *)
        IDS_CAP_ERRORPALOPEN    = 404; (* "Error opening palette '%s'" *)
        IDS_CAP_ERRORPALSAVE    = 405; (* "Error saving palette '%s'" *)
        IDS_CAP_ERRORDIBSAVE    = 406; (* "Error saving frame '%s'" *)
        IDS_CAP_DEFAVIEXT       = 407; (* "avi" *)
        IDS_CAP_DEFPALEXT       = 408; (* "pal" *)
        IDS_CAP_CANTOPEN        = 409; (* "Cannot open '%s'" *)
        IDS_CAP_SEQ_MSGSTART    = 410; (* "Select OK to start capture\nof video
sequence\nto %s." *)
        IDS_CAP_SEQ_MSGSTOP     = 411; (* "Hit ESCAPE or click to end capture" *)

        IDS_CAP_VIDEDITERR      = 412; (* "An error occurred while trying to run
VidEdit." *)
        IDS_CAP_READONLYFILE    = 413; (* "The file '%s' is a read-only file." *)
        IDS_CAP_WRITEERROR      = 414; (* "Unable to write to file '%s'.\nDisk may be
full." *)
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

IDS_CAP_NODISKSPACE = 415; (* "There is no space to create a capture file on the specified device." *)

IDS_CAP_SETFILESIZE = 416; (* "Set File Size" *)

IDS_CAP_SAVEASPERCENT = 417; (* "SaveAs: %2ld%% Hit Escape to abort." *)

IDS_CAP_DRIVER_ERROR = 418; (* Driver specific error message *)

IDS_CAP_WAVE_OPEN_ERROR = 419; (* "Error: Cannot open the wave input device.\nCheck sample size, frequency, and channels." *)

IDS_CAP_WAVE_ALLOC_ERROR = 420; (* "Error: Out of memory for wave buffers." *)

IDS_CAP_WAVE_PREPARE_ERROR = 421; (* "Error: Cannot prepare wave buffers." *)

IDS_CAP_WAVE_ADD_ERROR = 422; (* "Error: Cannot add wave buffers." *)

IDS_CAP_WAVE_SIZE_ERROR = 423; (* "Error: Bad wave size." *)

IDS_CAP_VIDEO_OPEN_ERROR = 424; (* "Error: Cannot open the video input device." *)

IDS_CAP_VIDEO_ALLOC_ERROR = 425; (* "Error: Out of memory for video buffers." *)

IDS_CAP_VIDEO_PREPARE_ERROR = 426; (* "Error: Cannot prepare video buffers." *)

IDS_CAP_VIDEO_ADD_ERROR = 427; (* "Error: Cannot add video buffers." *)

IDS_CAP_VIDEO_SIZE_ERROR = 428; (* "Error: Bad video size." *)

IDS_CAP_FILE_OPEN_ERROR = 429; (* "Error: Cannot open capture file." *)

IDS_CAP_FILE_WRITE_ERROR = 430; (* "Error: Cannot write to capture file. Disk may be full." *)

IDS_CAP_RECORDING_ERROR = 431; (* "Error: Cannot write to capture file. Data rate too high or disk full." *)

IDS_CAP_RECORDING_ERROR2 = 432; (* "Error while recording" *)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

IDS_CAP_AVI_INIT_ERROR = 433; (* "Error: Unable to initialize for capture." *)

IDS_CAP_NO_FRAME_CAP_ERROR = 434; (* "Warning: No frames captured.\nConfirm that vertical sync interrupts\nare configured and enabled." *)

IDS_CAP_NO_PALETTE_WARN = 435; (* "Warning: Using default palette." *)

IDS_CAP_MCI_CONTROL_ERROR = 436; (* "Error: Unable to access MCI device." *)

IDS_CAP_MCI_CANT_STEP_ERROR = 437; (* "Error: Unable to step MCI device." *)

IDS_CAP_NO_AUDIO_CAP_ERROR = 438; (* "Error: No audio data captured.\nCheck audio card settings." *)

IDS_CAP_AVI_DRAWDIB_ERROR = 439; (* "Error: Unable to draw this data format." *)

IDS_CAP_COMPRESSOR_ERROR = 440; (* "Error: Unable to initialize compressor." *)

IDS_CAP_AUDIO_DROP_ERROR = 441; (* "Error: Audio data was lost during capture, reduce capture rate." *)

(* status string IDs *)

IDS_CAP_STAT_LIVE_MODE = 500; (* "Live window" *)

IDS_CAP_STAT_OVERLAY_MODE = 501; (* "Overlay window" *)

IDS_CAP_STAT_CAP_INIT = 502; (* "Setting up for capture - Please wait" *)

IDS_CAP_STAT_CAP_FINI = 503; (* "Finished capture, now writing frame %ld" *)

IDS_CAP_STAT_PALETTE_BUILD = 504; (* "Building palette map" *)

IDS_CAP_STAT_OPTPAL_BUILD = 505; (* "Computing optimal palette" *)

IDS_CAP_STAT_I_FRAMES = 506; (* "%d frames" *)

IDS_CAP_STAT_L_FRAMES = 507; (* "%ld frames" *)

IDS_CAP_STAT_CAP_L_FRAMES = 508; (* "Captured %ld frames" *)

IDS_CAP_STAT_CAP_AUDIO = 509; (* "Capturing audio" *)

IDS_CAP_STAT_VIDEOCURRENT = 510; (* "Captured %ld frames (%ld dropped) %d.%03d sec." *)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        IDS_CAP_STAT_VIDEOAUDIO    = 511; (* "Captured %d.%03d sec.  %ld frames
(%ld dropped) (%d.%03d fps).  %ld audio bytes (%d,%03d sps)" *)
        IDS_CAP_STAT_VIDEOONLY     = 512; (* "Captured %d.%03d sec.  %ld frames
(%ld dropped) (%d.%03d fps)" *)
        IDS_CAP_STAT_FRAMESDROPPED = 513; (* "Dropped %ld of %ld frames
(%d.%02d%%) during capture." *)


{== DRAWDIB - Routines for drawing to the display ==========================}

type
  HDRAWDIB              = THandle; // hdd


{ == DrawDib Flags
============================================================================}

const
  DDF_UPDATE            = $0002;  // re-draw the last DIB
  DDF_SAME_HDC          = $0004;  // HDC same as last call (all setup)
  DDF_SAME_DRAW         = $0008;  // draw params are the same
  DDF_DONTDRAW          = $0010;  // dont draw frame, just decompress
  DDF_ANIMATE           = $0020;  // allow palette animation
  DDF_BUFFER            = $0040;  // always buffer image
  DDF_JUSTDRAWIT        = $0080;  // just draw it with GDI
  DDF_FULLSCREEN        = $0100;  // use DisplayDib
  DDF_BACKGROUNDPAL     = $0200;  // Realize palette in background
  DDF_NOTKEYFRAME       = $0400;  // this is a partial frame update, hint
  DDF_HURRYUP           = $0800;  // hurry up please!
  DDF_HALFTONE          = $1000;  // always halftone
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
  DDF_PREROLL            = DDF_DONTDRAW; // Builing up a non-keyframe
  DDF_SAME_DIB           = DDF_SAME_DRAW;
  DDF_SAME_SIZE          = DDF_SAME_DRAW;

{== DrawDib functions
=============================================================}

{-- DrawDibOpen() ---------------------------------------------------------}

function   DrawDibOpen: HDRAWDIB; stdcall;

{-- DrawDibClose() --------------------------------------------------------}

function   DrawDibClose(hdd: HDRAWDIB): BOOL; stdcall;

{-- DrawDibGetBuffer() ----------------------------------------------------}

function   DrawDibGetBuffer(hdd: HDRAWDIB; lpbi: PBITMAPINFOHEADER; dwSize:
DWORD; dwFlags: DWORD): Pointer;stdcall;

{-- DrawDibGetPalette() - get the palette used for drawing DIBs -----------}

function   DrawDibGetPalette(hdd: HDRAWDIB): HPALETTE; stdcall;

{-- DrawDibSetPalette() - set the palette used for drawing DIBs -----------}

function   DrawDibSetPalette(hdd: HDRAWDIB; hpal: HPALETTE): BOOL; stdcall;

{-- DrawDibChangePalette() ------------------------------------------------}

function   DrawDibChangePalette(hdd: HDRAWDIB; iStart, iLen: integer; lppe:
PPALETTEENTRY): BOOL; stdcall;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
{-- DrawDibRealize() - realize the palette in a HDD ----------}

function   DrawDibRealize(hdd: HDRAWDIB; hdc: HDC; fBackground: BOOL): UINT; stdcall;

{-- DrawDibStart() - start of streaming playback ----------}

function   DrawDibStart(hdd: HDRAWDIB; rate: DWORD): BOOL; stdcall;

{-- DrawDibStop() - start of streaming playback ----------}

function   DrawDibStop(hdd: HDRAWDIB): BOOL; stdcall;

{-- DrawDibBegin() - prepare to draw ----------}

function   DrawDibBegin(
  hdd       : HDRAWDIB;
  hdc       : HDC;
  dxDst     : integer;
  dyDst     : integer;
  lpbi      : PBITMAPINFOHEADER;
  dxSrc     : integer;
  dySrc     : integer;
  wFlags    : UINT
  ): BOOL; stdcall;

{-- DrawDibDraw() - actually draw a DIB to the screen ----------}

function   DrawDibDraw(
  hdd       : HDRAWDIB;
  hdc       : HDC;
  xDst      : integer;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
  yDst      : integer;
  dxDst     : integer;
  dyDst     : integer;
  lpbi      : PBITMAPINFOHEADER;
  lpBits    : Pointer;
  xSrc      : integer;
  ySrc      : integer;
  dxSrc     : integer;
  dySrc     : integer;
  wFlags    : UINT
  ): BOOL; stdcall;

{-- DrawDibUpdate() - redraw last image (may only be valid with DDF_BUFFER) --}

//function   DrawDibUpdate(hdd: HDRAWDIB; hdc: HDC; x, y: integer): BOOL;stdcall;

{-- DrawDibEnd() ------------------------------------------------------------}

function   DrawDibEnd(hdd: HDRAWDIB): BOOL; stdcall;

{-- DrawDibTime() - for debugging purposes only -----------------------------}

type
  PDRAWDIBTIME      = ^TDRAWDIBTIME;
  TDRAWDIBTIME      = record
    timeCount     : DWORD;
    timeDraw      : DWORD;
    timeDecompress  : DWORD;
    timeDither    : DWORD;
    timeStretch   : DWORD;
    timeBlt       : DWORD;
    timeSetDIBits   : DWORD;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
  end;

function  DrawDibTime(hdd: HDRAWDIB; lpddtime: PDRAWDIBTIME): BOOL; stdcall;

{-- Display profiling -------------------------------------------------------}

const
  PD_CAN_DRAW_DIB          = $0001;  // if you can draw at all
  PD_CAN_STRETCHDIB        = $0002;  // basicly RC_STRETCHDIB
  PD_STRETCHDIB_1_1_OK     = $0004;  // is it fast?
  PD_STRETCHDIB_1_2_OK     = $0008;  // ...
  PD_STRETCHDIB_1_N_OK     = $0010;  // ...

function  DrawDibProfileDisplay(lpbi: PBITMAPINFOHEADER): DWORD; stdcall;



implementation

// Dlls

const
  AVICAP32    = 'AVICAP32.dll';
  VFWDLL      = 'MSVFW32.DLL';



(* Externals from AVICAP.DLL *)
function capGetDriverDescription; external AVICAP32 name 'capGetDriverDescriptionA';
function capCreateCaptureWindow;  external AVICAP32 name 'capCreateCaptureWindowA';
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
(* Message crackers for above *)
function capSetCallbackOnError(hwnd : THandle; fpProc:TCAPERRORCALLBACK) :
LongInt;
begin
        Result := SendMessage(hwnd, WM_CAP_SET_CALLBACK_ERROR, 0,LPARAM
((@fpProc));
end;

function capSetCallbackOnStatus(hwnd : THandle;
fpProc:TCAPSTATUSCALLBACK):LongInt;
begin
        Result := SendMessage(hwnd, WM_CAP_SET_CALLBACK_STATUS, 0, LPARAM
((@fpProc));
end;

function capSetCallbackOnYield (hwnd : THandle; fpProc:TCAPYIELDCALLBACK):LongInt;
begin
        Result := SendMessage(hwnd, WM_CAP_SET_CALLBACK_YIELD, 0, LPARAM
((@fpProc));
end;

function capSetCallbackOnFrame (hwnd : THandle;
fpProc:TCAPVIDEOSTREAMCALLBACK):LongInt;
begin
        Result := SendMessage(hwnd, WM_CAP_SET_CALLBACK_FRAME, 0,LPARAM(
@fpProc));
end;

function capSetCallbackOnVideoStream(hwnd:THandle;
fpProc:TCAPVIDEOSTREAMCALLBACK):LongInt;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
begin
        Result := SendMessage(hwnd, WM_CAP_SET_CALLBACK_VIDEOSTREAM, 0,
LPARAM(@fpProc));
end;

function capSetCallbackOnWaveStream (hwnd:THandle;
fpProc:TCAPWAVESTREAMCALLBACK):LongInt;
begin
        Result := SendMessage(hwnd, WM_CAP_SET_CALLBACK_WAVESTREAM, 0,
LPARAM(@fpProc));
end;

function capSetCallbackOnCapControl (hwnd:THandle;
fpProc:TCAPCONTROLCALLBACK):longint;
begin
        Result := SendMessage(hwnd, WM_CAP_SET_CALLBACK_CAPCONTROL, 0,
LPARAM(@fpProc));
end;

function capSetUserData(hwnd:THandle; lUser:LongInt):LongInt;
begin
        Result := SendMessage(hwnd, WM_CAP_SET_USER_DATA, 0, lUser);
end;

function capGetUserData(hwnd:THandle):LongInt;
begin
        Result := SendMessage(hwnd, WM_CAP_GET_USER_DATA, 0, 0);
end;

function capDriverConnect(hwnd:THandle; I: Word) : boolean;
begin
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        Result :=boolean( SendMessage(hwnd, WM_CAP_DRIVER_CONNECT, WPARAM
(1), 0));
end;

function capDriverDisconnect(hwnd:THandle):Boolean;
begin
        Result := boolean(SendMessage(hwnd, WM_CAP_DRIVER_DISCONNECT, 0, 0));
end;

function capDriverGetName(hwnd:THandle; szName:PChar; wSize:Word):boolean;
begin
        Result :=boolean( SendMessage(hwnd, WM_CAP_DRIVER_GET_NAME, WPARAM
(wSize), LPARAM( szName)));
end;

function capDriverGetVersion(hwnd:THandle; szVer:PChar; wSize:Word):boolean;
begin
        Result :=boolean( SendMessage(hwnd, WM_CAP_DRIVER_GET_VERSION,
WPARAM(wSize),LPARAM( szVer)));
end;

function capDriverGetCaps(hwnd:THandle; s:pCapDriverCaps; wSize:Word):boolean;
begin
        Result := boolean(SendMessage(hwnd, WM_CAP_DRIVER_GET_CAPS, WPARAM
(wSize),LPARAM(s)));
end;

function capFileSetCaptureFile(hwnd:THandle; szName:PChar):boolean;
begin
        Result := Boolean(SendMessage(hwnd, WM_CAP_FILE_SET_CAPTURE_FILE, 0,
LPARAM(szName)));
end;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
function capFileGetCaptureFile(hwnd:THandle; szName:PChar; wSize:Word):Boolean;
begin
        Result := Boolean(SendMessage(hwnd, WM_CAP_FILE_GET_CAPTURE_FILE,
wSize, LPARAM(szName)));
end;

function capFileAlloc(hwnd:THandle; dwSize:DWord):boolean;
begin
        Result := Boolean(SendMessage(hwnd, WM_CAP_FILE_ALLOCATE, 0, LPARAM
(dwSize)));
end;

function capFileSaveAs(hwnd:THandle; szName:Pchar):Boolean;
begin
        Result := Boolean(SendMessage(hwnd, WM_CAP_FILE_SAVEAS, 0,LPARAM
(szName)));
end;

function capFileSetInfoChunk(hwnd:THandle; lpInfoChunk:pCapInfoChunk):boolean;
begin
        Result := Boolean(SendMessage(hwnd, WM_CAP_FILE_SET_INFOCHUNK, 0,
LPARAM(lpInfoChunk)));
end;

function capFileSaveDIB(hwnd:THandle; szName:Pchar):Boolean;
begin
        Result :=Boolean(SendMessage(hwnd, WM_CAP_FILE_SAVEDIB, 0,LPARAM
(szName)));
end;

function capEditCopy(hwnd : THandle):Boolean;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
begin
        Result := Boolean(SendMessage(hwnd, WM_CAP_EDIT_COPY, 0, 0));
end;

function capSetAudioFormat(hwnd:THandle; s:PWaveFormatEx; wSize:Word):boolean;
begin
        Result := Boolean(SendMessage(hwnd, WM_CAP_SET_AUDIOFORMAT, WPARAM
(wSize),LPARAM(s)));
end;

function capGetAudioFormat(hwnd:THandle; s:PWaveFormatEx; wSize:Word):DWORD;
begin
        Result :=DWORD( SendMessage(hwnd, WM_CAP_GET_AUDIOFORMAT,
WPARAM(wSize),LPARAM(s)));
end;

function capGetAudioFormatSize(hwnd:THandle):DWORD;
begin
        Result := DWORD(SendMessage(hwnd, WM_CAP_GET_AUDIOFORMAT, 0, 0));
end;

function capDlgVideoFormat(hwnd:THandle):boolean;
begin
        Result :=boolean(SendMessage(hwnd, WM_CAP_DLG_VIDEOFORMAT, 0, 0));
end;

function capDlgVideoSource(hwnd:THandle):boolean;
begin
        Result :=boolean (SendMessage(hwnd, WM_CAP_DLG_VIDEOSOURCE, 0, 0));
end;

function capDlgVideoDisplay(hwnd:THandle):boolean;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
begin
        Result := boolean(SendMessage(hwnd, WM_CAP_DLG_VIDEODISPLAY, 0, 0));
end;

function capDlgVideoCompression(hwnd:THandle):boolean;
begin
        Result := boolean(SendMessage(hwnd, WM_CAP_DLG_VIDEOCOMPRESSION, 0,
0));
end;

function capGetVideoFormat(hwnd:THandle; s:pBitmapInfo; wSize:Word):DWord;
begin
        Result := DWord(SendMessage(hwnd, WM_CAP_GET_VIDEOFORMAT, Wparam
(wSize), LPARAM(s)));
end;

function capGetVideoFormatSize(hwnd:THandle):DWord;
begin
        Result := DWord(SendMessage(hwnd, WM_CAP_GET_VIDEOFORMAT, 0, 0));
end;

function capSetVideoFormat(hwnd:THandle; s:PBitmapInfo; wSize:Word):Boolean;
begin
        Result := Boolean(SendMessage(hwnd, WM_CAP_SET_VIDEOFORMAT, WPARAM
(wSize), LPARAM(s)));
end;

function capPreview(hwnd:THandle; f:boolean):boolean;
begin
        Result := Boolean(SendMessage(hwnd, WM_CAP_SET_PREVIEW, WPARAM(f), 0));
end;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
function capPreviewRate(hwnd:THandle; wMS:Word):boolean;
begin
        Result := Boolean(SendMessage(hwnd, WM_CAP_SET_PREVIEWRATE, WPARAM
(wMS), 0));
end;

function capOverlay(hwnd:THandle; f:boolean):boolean;
begin
        Result := Boolean(SendMessage(hwnd, WM_CAP_SET_OVERLAY, WPARAM(f), 0));
end;

function capPreviewScale(hwnd:THandle; f:boolean):Boolean;
begin
        Result := Boolean(SendMessage(hwnd, WM_CAP_SET_SCALE,WPARAM(f), 0));
end;

function capGetStatus(hwnd:THandle; s:PCapStatus; wSize:Word):boolean;
begin
        Result := Boolean(SendMessage(hwnd, WM_CAP_GET_STATUS, WPARAM
(wSize),LPARAM(s)));
end;

function capSetScrollPos(hwnd:THandle; lpP:pPoint):Boolean;
begin
        Result := Boolean(SendMessage(hwnd, WM_CAP_SET_SCROLL, 0, LParam(lpP)));
end;

function capGrabFrame(hwnd:THandle):boolean;
begin
   Result := Boolean(SendMessage(hwnd, WM_CAP_GRAB_FRAME, 0, 0));
end;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
function capGrabFrameNoStop(hwnd:THandle):boolean;
begin
        Result := Boolean(SendMessage(hwnd, WM_CAP_GRAB_FRAME_NOSTOP, 0, 0));
end;

function capCaptureSequence(hwnd:THandle):boolean;
begin
        Result := Boolean(SendMessage(hwnd, WM_CAP_SEQUENCE, 0, 0));
end;

function capCaptureSequenceNoFile(hwnd:THandle):boolean;
begin
        Result :=Boolean(SendMessage(hwnd, WM_CAP_SEQUENCE_NOFILE, 0, 0));
end;

function capCaptureStop(hwnd:THandle):boolean;
begin
        Result := Boolean(SendMessage(hwnd, WM_CAP_STOP, 0, 0));
end;

function capCaptureAbort(hwnd:THandle):Boolean;
begin
        Result := Boolean(SendMessage(hwnd, WM_CAP_ABORT, 0, 0));
end;

function capCaptureSingleFrameOpen(hwnd:THandle):boolean;
begin
        Result := Boolean(SendMessage(hwnd, WM_CAP_SINGLE_FRAME_OPEN, 0, 0));
end;

function capCaptureSingleFrameClose(hwnd:THandle):boolean ;
begin
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        Result := Boolean(SendMessage(hwnd, WM_CAP_SINGLE_FRAME_CLOSE, 0, 0));
end;

function capCaptureSingleFrame(hwnd:THandle):boolean;
begin
        Result :=Boolean(SendMessage(hwnd, WM_CAP_SINGLE_FRAME, 0, 0));
end;

function capCaptureGetSetup(hwnd:THandle; s:pCaptureParms; wSize:Word):boolean;
begin
        Result :=Boolean( SendMessage(hwnd, WM_CAP_GET_SEQUENCE_SETUP,
WPARAM(wSize),LPARAM(s)));
end;

function capCaptureSetSetup(hwnd:THandle; s:pCaptureParms; wSize:Word):boolean;
begin
        Result := Boolean (SendMessage(hwnd, WM_CAP_SET_SEQUENCE_SETUP,
WParam(wSize),LParam(s)));
end;

function capSetMCIDeviceName(hwnd:THandle; szName:Pchar):Boolean;
begin
        Result :=Boolean( SendMessage(hwnd, WM_CAP_SET_MCI_DEVICE, 0, LParam
(szName)));
end;

function capGetMCIDeviceName(hwnd:THandle; szName:Pchar; wSize:Word):Boolean;
begin
        Result := Boolean(SendMessage(hwnd, WM_CAP_GET_MCI_DEVICE, Wparam
(wSize), LPARAM(szName)));
end;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
function capPaletteOpen(hwnd:THandle; szName:PChar):Boolean;
begin
        Result := Boolean(SendMessage(hwnd, WM_CAP_PAL_OPEN, 0, LParam(szName)));
end;

function capPaletteSave(hwnd:THandle; szName:PChar):boolean;
begin
        Result := Boolean(SendMessage(hwnd, WM_CAP_PAL_SAVE, 0,LParam(szName)));
end;

function capPalettePaste(hwnd:THandle):Boolean;
begin
        Result :=Boolean( SendMessage(hwnd, WM_CAP_PAL_PASTE, 0, 0));
end;

function capPaletteAuto(hwnd:THandle; iFrames:Word; iColors:word):Boolean;
begin
        Result :=Boolean( SendMessage(hwnd, WM_CAP_PAL_AUTOCREATE, WPARAM
(iFrames),LPARAM(iColors)));
end;

function capPaletteManual(hwnd:THandle; fGrab:Word; iColors:word):Boolean;
begin
        Result := Boolean(SendMessage(hwnd, WM_CAP_PAL_MANUALCREATE,
WPARAM(fGrab),LPARAM(iColors)));
end;


{== DrawDib functions
==========================================================}

{-- DrawDibOpen() ----------------------------------------------------}
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

function DrawDibOpen: HDRAWDIB; stdcall; external VFWDLL;

{-- DrawDibClose() ----------}

function DrawDibClose(hdd: HDRAWDIB): BOOL; stdcall; external VFWDLL;

{-- DrawDibGetBuffer() ----------}

function DrawDibGetBuffer(hdd: HDRAWDIB; lpbi: PBITMAPINFOHEADER; dwSize: DWORD; dwFlags: DWORD): Pointer; stdcall; external VFWDLL;

{-- DrawDibGetPalette() - get the palette used for drawing DIBs ----------}

function DrawDibGetPalette(hdd: HDRAWDIB): HPALETTE; stdcall; external VFWDLL;

{-- DrawDibSetPalette() - set the palette used for drawing DIBs ----------}

function DrawDibSetPalette(hdd: HDRAWDIB; hpal: HPALETTE): BOOL; stdcall; external VFWDLL;

{-- DrawDibChangePalette() ----------}

function DrawDibChangePalette(hdd: HDRAWDIB; iStart, iLen: integer; lppe: PPALETTEENTRY): BOOL; stdcall; external VFWDLL;

{-- DrawDibRealize() - realize the palette in a HDD ----------}

function DrawDibRealize(hdd: HDRAWDIB; hdc: HDC; fBackground: BOOL): UINT; stdcall; external VFWDLL;

{-- DrawDibStart() - start of streaming playback ----------}

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
function  DrawDibStart(hdd: HDRAWDIB; rate: DWORD): BOOL; stdcall; external VFWDLL;

{-- DrawDibStop() - start of streaming playback ----------------------------}

function  DrawDibStop(hdd: HDRAWDIB): BOOL; stdcall; external VFWDLL;

{-- DrawDibBegin() - prepare to draw ---------------------------------------}

function  DrawDibBegin(
  hdd      : HDRAWDIB;
  hdc      : HDC;
  dxDst    : integer;
  dyDst    : integer;
  lpbi     : PBITMAPINFOHEADER;
  dxSrc    : integer;
  dySrc    : integer;
  wFlags   : UINT
  ): BOOL; stdcall; external VFWDLL;

{-- DrawDibDraw() - actually draw a DIB to the screen ----------------------}

function  DrawDibDraw(
  hdd      : HDRAWDIB;
  hdc      : HDC;
  xDst     : integer;
  yDst     : integer;
  dxDst    : integer;
  dyDst    : integer;
  lpbi     : PBITMAPINFOHEADER;
  lpBits   : Pointer;
  xSrc     : integer;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
  ySrc      : integer;
  dxSrc     : integer;
  dySrc     : integer;
  wFlags    : UINT
  ): BOOL; stdcall; external VFWDLL;

{-- DrawDibEnd() ----------------------------------------------------------}

function  DrawDibEnd(hdd: HDRAWDIB): BOOL; stdcall; external VFWDLL;

{-- DrawDibTime() - for debugging purposes only ---------------------------}

function  DrawDibTime(hdd: HDRAWDIB; lpddtime: PDRAWDIBTIME): BOOL; stdcall;
external VFWDLL;

{-- Display profiling -----------------------------------------------------}

function  DrawDibProfileDisplay(lpbi: PBITMAPINFOHEADER): DWORD; stdcall; external
VFWDLL;


end.

unit MnForm1;

interface

uses
  // Delphi units
  Windows, Messages, SysUtils, Classes, Graphics, Controls, Forms, Dialogs,
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
Menus, ComCtrls, ExtCtrls, StdCtrls, ToolWin, ImgList, ClipBrd,
// ComDrv32 units
CPDrv,
// TTY units
AboutTTY,
SettingsDlg, Psock, NMFtp, Video, Buttons;

type
 TMainForm = class(TForm)
   cpDrv: TCommPortDriver;
   Splitter1: TSplitter;
   Panel1: TPanel;
   ToolBar1: TToolBar;
   ConnectToolButton: TToolButton;
   DisconnectToolButton: TToolButton;
   SettingsToolButton: TToolButton;
   E_ImageList: TImageList;
   QuitTTYToolButton: TToolButton;
   Panel4: TPanel;
   StatusPanel: TPanel;
   FrameSettingsPanel: TPanel;
   FlowSettingsPanel: TPanel;
   NMFTP1: TNMFTP;
   VideoCap1: TVideoCap;
   Label1: TLabel;
   Long: TEdit;
   Forward: TSpeedButton;
   Left: TSpeedButton;
   Backward: TSpeedButton;
   Right: TSpeedButton;
   Stop: TSpeedButton;
   lable2: TLabel;
```

เอกสารนี้สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
Turn: TEdit;
procedure SettingsToolButtonClick(Sender: TObject);
procedure FormCreate(Sender: TObject);
procedure ConnectToolButtonClick(Sender: TObject);
procedure DisconnectToolButtonClick(Sender: TObject);
procedure OutgoingRichEditKeyDown(Sender: TObject; var Key: Word;
  Shift: TShiftState);

procedure OutgoingRichEditKeyPress(Sender: TObject; var Key: Char);
procedure cpDrvReceiveData(Sender: TObject; DataPtr: Pointer;
  DataSize: Cardinal);

procedure QuitTTYToolButtonClick(Sender: TObject);
procedure HelpAboutCmdClick(Sender: TObject);
procedure OT_PasteCmdClick(Sender: TObject);
procedure VideoCap1Status(Sender: TObject);
procedure ForwardClick(Sender: TObject);
procedure BackwardClick(Sender: TObject);
procedure LeftClick(Sender: TObject);
procedure RightClick(Sender: TObject);
procedure StopClick(Sender: TObject);


private
  // Startup about-box (splash screen)
  FAboutBox: TAboutBoxForm;
  FAboutBoxShownTime: DWORD;

  // Called when the message queue gets empty.
  procedure IdleProc( Sender: TObject; var Done: boolean );
  // Updates the panels on bottom of this window.
  procedure UpdateStatusPanels;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
  // Displays an error box informing the user we can't send data
  procedure CannotSendError;
 public
 end;

var
 MainForm: TMainForm;

implementation

{$R *.DFM}

// Form setup
procedure TMainForm.FormCreate(Sender: TObject);
begin
 // Redirect OnIdle
 Application.OnIdle := IdleProc;
 // Display the splash screen
 {FAboutBox := TAboutBoxForm.Create( nil, false );
 Enabled := false;
 FAboutBoxShownTime := GetTickCount();
 FAboutBox.Show;
 FAboutBox.Update;}
 //VideoCap1.Videopreview:= true;
end;

 // Lets the user to customize I/O settings
 procedure TMainForm.SettingsToolButtonClick(Sender: TObject);
 var dlg: TSettingsForm;
 begin
  // Tell the user we cannot change settings while a connection is active
  if cpDrv.Connected then
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
begin
 if Application.MessageBox( 'Could not change settings while a connection is active.'#13#10+
                  'Close the connection and continue ?',
                  'Confirm',
                  MB_OKCANCEL or MB_ICONQUESTION ) <> ID_OK then

  exit;
 cpDrv.Disconnect;
end;
// Let the user to customize settings
dlg := nil;
try
  dlg := TSettingsForm.Create( Self, cpDrv );
  dlg.ShowModal;
finally
  dlg.Free;
end;
end;

// Called when the message queue gets empty.
procedure TMainForm.IdleProc( Sender: TObject; var Done: boolean );
var elapsedTime: DWORD;
begin
  Done := false;
  // Hides the splash-screen
  if FAboutBox <> nil then
  begin
    elapsedTime := GetTickCount - FAboutBoxShownTime;
    if elapsedTime < 400 then
      SetForegroundWindow( FAboutBox.Handle );
    if (elapsedTime > 5000) or FAboutBox.ReqToClose then
    begin
      FAboutBox.Free;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    FAboutBox := nil;
    Enabled := true;
   end;
  end;
  // Updates status panels
  UpdateStatusPanels;
end;

// Updates the panels on bottom of this window.
procedure TMainForm.UpdateStatusPanels;
const _databits: array[TDataBits] of string = ('5','6','7','8');
      _parity: array[TParity] of string = ('N','E','O','M','S');
      _stopbits: array[TStopBits] of string = ('1','1.5','2');
      _hwflow: array[THwFlowControl] of string = ('None','None+DTR on','RTS/CTS');
      _swflow: array[TSwFlowControl] of string = ('None','XON/XOFF');
 var s: string;
 begin
   // Updates the connection status
   if cpDrv.Connected then
    s := 'Connected to "' + cpDrv.PortName + '"'
   else
    s := 'Not connected';
   StatusPanel.Caption := s;
   // Show current frame settings
   s := IntToStr( cpDrv.BaudRateValue ) + ',' +
        _databits[ cpDrv.DataBits ] + ',' +
        _parity[ cpDrv.Parity ] + ',' +
        _stopbits[ cpDrv.StopBits ];
    FrameSettingsPanel.Caption := s;
    // Show current flow control settings
    s := 'Hw:' + _hwflow[ cpDrv.HwFlow ] + ' - Sw:' + _swflow[ cpDrv.SwFlow ];
    FlowSettingsPanel.Caption := s;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
end;

// Connect
procedure TMainForm.ConnectToolButtonClick(Sender: TObject);
begin
 // Do nothing if already connected
 if cpDrv.Connected then
   exit;
 // Try connecting
 if not cpDrv.Connect then
 begin
  Application.MessageBox( 'Could not connect to serial port.'#13#10+
                'Please, check settings and try again.',
                'Error',
                MB_OK or MB_ICONERROR );
   exit;
 end;
 videocap1.Width:=465;
 videocap1.Height:=400;
 VideoCap1.Videopreview:= true;

end;

 // Disconnect
 procedure TMainForm.DisconnectToolButtonClick(Sender: TObject);
 begin
  // Do nothing if not connected
  if not cpDrv.Connected then
    exit;
  // Disconnect
  cpDrv.Disconnect;
  VideoCap1.Videopreview:= false;
```

เอกสารนี้สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
 VideoCap1.CleanupInstance;


end;


// If user is trying to send text but the connection is not active then
// automatically bring it on.
procedure TMainForm.OutgoingRichEditKeyDown(Sender: TObject; var Key: Word;
  Shift: TShiftState);
begin
  if not cpDrv.Connected then
  begin
    ConnectToolButtonClick( nil );
    if not cpDrv.Connected then
      Key := 0;
  end;
end;




// Displays an error box informing the user we can't send data
procedure TMainForm.CannotSendError;
begin
  if cpDrv.CheckLineStatus then
    Application.MessageBox( 'Could not send data.'#13#10+
                  'No device connected to serial port or device is off. Please, turn it on.'#13#10 +
                  'Try setting Device Check to Off in the settings dialog box.'#13#10 +
                  'Also, replace your two wires serial cable with a full wires cable.',
                  'Warning',
                  MB_OK or MB_ICONINFORMATION )
  else
    Application.MessageBox( 'Could not send data.'#13#10+
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                'Please, check connections and try again.'#13#10 +
                'Turn serial device on or, try setting Device Check to On in the settings dialog
box.'#13#10 +
                'Also, disable Hardware Flow Control if you are using a two wires cable.',
                'Warning',
                MB_OK or MB_ICONINFORMATION )
end;

procedure TMainForm.OutgoingRichEditKeyPress(Sender: TObject;
 var Key: Char);
begin
 // Do nothing if not connected
 if not cpDrv.Connected then
  exit;
 // Send the character
 if not cpDrv.SendChar( Key ) then
  CannotSendError;
end;

// Handles incoming data
procedure TMainForm.cpDrvReceiveData(Sender: TObject; DataPtr: Pointer;
 DataSize: Cardinal);
var //iLastLine,
  i: integer;
  s: string;
begin
 // Convert incoming data into a string
 s := StringOfChar( ' ', DataSize );
 move( DataPtr^, pchar(s)^, DataSize );
 // Exit if s is empty. This usually occurs when one or more NULL characters
 // (chr(0)) are received.
 while pos( #0, s ) > 0 do
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
  delete( s, pos( #0, s ), 1 );
 if s = '' then
  exit;
 // Remove line feeds
 i := pos( #10, s );
 while i <> 0 do
 begin
  delete( s, i, 1 );
  i := pos( #10, s );
 end;

 // Don't redraw the rich edit control until we finished updating it
 //IncomingRichEdit.Lines.BeginUpdate;
 // Get last line index

end;


// Quits TTY
procedure TMainForm.QuitTTYToolButtonClick(Sender: TObject);
begin
 PostQuitMessage( Handle );
end;

procedure TMainForm.HelpAboutCmdClick(Sender: TObject);
var dlg: TAboutBoxForm;
begin
 dlg := nil;
 try
  dlg := TAboutBoxForm.Create( Self, true );
  dlg.ShowModal;
 finally
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
      dlg.Free;
    end;
  end;


procedure TMainForm.OT_PasteCmdClick(Sender: TObject);
var clp: TClipboard;
  s, ss: string;
  iLastLine, i: integer;
begin
 // Get the clipboard object
 clp := Clipboard;
 // If the clipboard contains some text...
 if clp.HasFormat( CF_TEXT ) then
 begin
  // Automatically connect
  if not cpDrv.Connected then
  begin
   ConnectToolButtonClick( nil );
   if not cpDrv.Connected then
    exit;
  end;
  // Get the text
  s := clp.AsText;
  // Remove line feeds
  i := pos( #10, s );
 end;
end;

procedure TMainForm.VideoCap1Status(Sender: TObject);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
begin
 VideoCap1.SetDriverName('iCOM-H.32x Capture Driver');
   VideoCap1.DriverName:='iCOM-H.32x Capture Driver';
   VideoCap1.DriverOpen:=true;
end;


procedure TMainForm.ForwardClick(Sender: TObject);
var i,lng:integer;
begin
 // Do nothing if not connected
 if not cpDrv.Connected then
   exit;
 // Send the character
 lng:=strtoint(Long.text);
 for i:=1 to lng do
    if not cpDrv.SendChar(#255) then
       CannotSendError;
 for i:=1 to 10 do
    if not cpDrv.SendChar(#240) then
       CannotSendError;
end;

procedure TMainForm.BackwardClick(Sender: TObject);
var i,lng:integer;
begin
   // Do nothing if not connected
  if not cpDrv.Connected then
    exit;
  // Send the character
   lng:=strtoint(Long.text);
   for i:=1 to lng do
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    if not cpDrv.SendChar(#15) then
      CannotSendError;
  for i:=1 to 10 do
    if not cpDrv.SendChar(#240) then
      CannotSendError;
end;

procedure TMainForm.LeftClick(Sender: TObject);
var i,long:integer;
begin
    // Do nothing if not connected
  if not cpDrv.Connected then
    exit;
  // Send the character
  long:=strtoint(Turn.Text);
  for i:=1 to long do
    if not cpDrv.SendChar(#63) then
      CannotSendError;
  for i:=1 to 10 do
    if not cpDrv.SendChar(#240) then
      CannotSendError;
end;

procedure TMainForm.RightClick(Sender: TObject);
var i,long:integer;
begin
      // Do nothing if not connected
  if not cpDrv.Connected then
    exit;
  // Send the character
  long:=strtoint(Turn.text);
  for i:=1 to long do
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    if not cpDrv.SendChar(#3) then
        CannotSendError;
  for i:=1 to 10 do
    if not cpDrv.SendChar(#240) then
        CannotSendError;
end;

procedure TMainForm.StopClick(Sender: TObject);
var i:integer;
begin
    if not cpDrv.Connected then
   exit;
  // Send the character
  for i:=1 to 100 do
     if not cpDrv.SendChar(#240) then
         CannotSendError;
 end;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

19-0175; Rev 2; 5/96

MAXIM

±15kV ESD-Protected, +5V RS-232 Transceivers

General Description

The MAX202E–MAX213E, MAX232E/MAX241E line drivers/receivers are designed for RS-232 and V.28 communications in harsh environments. Each transmitter output and receiver input is protected against ±15kV electrostatic discharge (ESD) shocks, without latchup. The various combinations of features are outlined in the *Selection Guide*. The drivers and receivers for all ten devices meet all EIA/TIA-232E and CCITT V.28 specifications at data rates up to 120kbps, when loaded in accordance with the EIA/TIA-232E specification.

The MAX211E/MAX213E/MAX241E are available in 28-pin SO packages, as well as a 28-pin SSOP that uses 60% less board space. The MAX202E/MAX232E come in 16-pin narrow SO, wide SO, and DIP packages. The MAX203E comes in a 20-pin DIP/SO package, and needs no external charge-pump capacitors. The MAX205E comes in a 24-pin wide DIP package, and also eliminates external charge-pump capacitors. The MAX206E/MAX207E/MAX208E come in 24-pin SO, SSOP, and narrow DIP packages. The MAX232E/MAX241E operate with four 1µF capacitors, while the MAX202E/MAX206E/MAX207E/MAX208E/MAX211E/MAX213E operate with four 0.1µF capacitors, further reducing cost and board space.

Applications

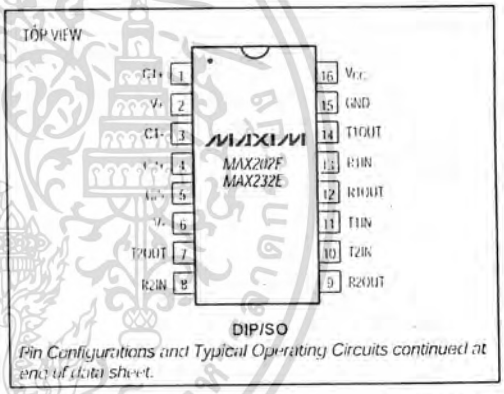
Notebook, Subnotebook, and Palmtop Computers
Battery-Powered Equipment
Hand-Held Equipment

Ordering Information appears at end of data sheet.

Features

- ESD Protection for RS-232 I/O Pins:
  ±15kV—Human Body Model
  ±8kV—IEC1000-4-2, Contact Discharge
  ±15kV—IEC1000-4-2, Air-Gap Discharge
- Latchup Free (unlike bipolar equivalents)
- Guaranteed 120kbps Data Rate—LapLink™ Compatible
- Guaranteed 3V/µs Min Slew Rate
- Operate from a Single +5V Power Supply

Pin Configurations



DIP/SO

Pin Configurations and Typical Operating Circuits continued at end of data sheet.

Selection Guide

PART	No. of RS-232 DRIVERS	No. of RS-232 RECEIVERS	RECEIVERS ACTIVE IN SHUTDOWN	No. of EXTERNAL CAPACITORS	LOW-POWER SHUTDOWN	TTL THREE-STATE
MAX202E	2	2	0	4 (0.1µF)	No	No
MAX203E	2	2	0	None	No	No
MAX205E	5	5	0	None	Yes	Yes
MAX206E	4	3	0	4 (0.1µF)	Yes	Yes
MAX207E	5	3	0	4 (0.1µF)	No	No
MAX208E	4	4	0	4 (0.1µF)	No	No
MAX211E	4	5	0	4 (0.1µF)	Yes	Yes
MAX213E	4	5	2	4 (0.1µF)	Yes	Yes
MAX232E	2	2	0	4 (1µF)	No	No
MAX241E	4	5	0	4 (1µF)	Yes	Yes

LapLink is a registered trademark of Traveling Software, Inc.

For free samples & the latest literature: http://www.maxim-ic.com, or phone 1-800-998-8800

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

±15kV ESD-Protected, +5V RS-232 Transceivers

ABSOLUTE MAXIMUM RATINGS

V_{CC} ..... -0.3V to +6V
$V+$ ..... $(V_{CC}$ - 0.3V) to +14V
$V-$ ..... -14V to +0.3V
Input Voltages
T_IN ..... -0.3V to (V+ + 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)
16-Pin Plastic DIP (derate 10.53mW/°C above +70°C) 842mW
16-Pin Narrow SO (derate 8.70mW/°C above +70°C) 696mW
16-Pin Wide SO (derate 9.52mW/°C above +70°C) 762mW
20-Pin Plastic DIP (derate 11.11mW/°C above +70°C) 889mW
20-Pin SO (derate 10.00mW/°C above +70°C) 800mW
24-Pin Narrow Plastic DIP
(derate 13.33mW/°C above +70°C) 1.07W
24-Pin Wide Plastic DIP
(derate 14.29mW/°C above +70°C) 1.14W
24-Pin SO (derate 11.76mW/°C above +70°C) 941mW
24-Pin SSOP (derate 8.00mW/°C above +70°C) 640mW
28-Pin SO (derate 12.50mW/°C above +70°C) 1W
28-Pin SSOP (derate 9.52mW/°C above +70°C) 762mW
Operating Temperature Ranges
MAX2__EC_ _ ..... 0°C to +70°C
MAX2__EE_ _ ..... -40°C to +85°C
Storage Temperature Range ..... -65°C to +165°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

(V_{CC} = +5V ±10% for MAX202E/206E/208E/211E/213E/232E/241E; V_{CC} = +5V ±5% for MAX203E/205E/207E; C1–C4 = 0.1µF for MAX202E/206E/207E/208E/211E/213E; C1–C4 = 1µF for MAX232E/241E; T_A = T_{MIN} to T_{MAX}, unless otherwise noted. Typical values are at T_A = +25°C.)

PARAMETER	SYMBOL	CONDITIONS		MIN	TYP	MAX	UNITS
DC CHARACTERISTICS							
V_{CC} Supply Current	I_{CC}	No load, T_A = +25°C	MAX202E/203E		8	15	mA
			MAX205E–208E		11	20	
			MAX211E/213E		14	20	
			MAX232E		5	10	
			MAX241E		7	15	
Shutdown Supply Current		T_A = +25°C, Figure 1	MAX205E/206E		1	10	µA
			MAX211E/241E		1	10	
			MAX213E		15	50	
LOGIC							
Input Pull-Up Current		T_IN = 0V (MAX205E–208E/211E/213E/241E)			15	200	µA
Input Leakage Current		T_IN = 0V to V_{CC} (MAX202E/203E/232E)				±10	µA
Input Threshold Low	V_{IL}	T_IN, EN, SHDN (MAX213E) or EN, SHDN (MAX205E–208E/211E/241E)				0.8	V
Input Threshold High	V_{IH}	T_IN		2.0			V
		EN, SHDN (MAX213E) or EN, SHDN (MAX205E–208E/211E/241E)		2.4			
Output Voltage Low	V_{OL}	R_OUT; I_{OUT} = 3.2mA (MAX202E/203E/232E) or I_{OUT} = 1.6mA (MAX205E/208E/211E/213E/241E)				0.4	V
Output Voltage High	V_{OH}	R_OUT; I_{OUT} = -1.0mA		3.5	V_{CC} - 0.4		V
Output Leakage Current		EN = V_{CC}, EN = 0V, 0V ≤ R_{OUT} ≤ V_{CC}, MAX205E–208E/211E/213E/241E outputs disabled			±0.05	±10	µA

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ELECTRICAL CHARACTERISTICS (continued)

(V_{CC} = +5V ±10% for MAX202E/206E/208E/211E/213E/232E/241E; V_{CC} = +5V ±5% for MAX203E/205E/207E; C1–C4 = 0.1µF for MAX202E/206E/207E/208E/211E/213E; C1–C4 = 1µF for MAX232E/241E; T_A = T_{MIN} to T_{MAX}; unless otherwise noted. Typical values are at T_A = +25°C.)

PARAMETER	SYMBOL	CONDITIONS		MIN	TYP	MAX	UNITS
EIA/TIA-232E RECEIVER INPUTS							
Input Voltage Range				-30		30	V
Input Threshold Low		T_A = +25°C, V_{CC} = 5V	All parts, normal operation	0.8	1.2		V
			MAX213E, $\overline{SHDN}$ = 0V, EN = V_{CC}	0.6	1.5		
Input Threshold High		T_A = +25°C, V_{CC} = 5V	All parts, normal operation		1.7	2.4	V
			MAX213E (R4, R5), $\overline{SHDN}$ = 0V, EN = V_{CC}		1.5	2.4	
Input Hysteresis		V_{CC} = 5V, no hysteresis in shutdown		0.2	0.5	1.0	V
Input Resistance		T_A = +25°C, V_{CC} = 5V		3	5	7	kΩ
EIA/TIA-232E TRANSMITTER OUTPUTS							
Output Voltage Swing		All drivers loaded with 3kΩ to ground (Note 1)		±5	±9		V
Output Resistance		V_{CC} = V+ = V- = 0V, V_{OUT} = ±2V		300			Ω
Output Short-Circuit Current					±10	±60	mA
TIMING CHARACTERISTICS							
Maximum Data Rate		R_L = 3kΩ to 7kΩ, C_L = 50pF to 1000pF, one transmitter switching		120			kbps
Receiver Propagation Delay	t_{PLHR}, t_{PHLR}	C_L = 150pF	All parts, normal operation		0.5	10	µs
			MAX213E (R4, R5), $\overline{SHDN}$ = 0V, EN = V_{CC}		4	40	
Receiver Output Enable Time		MAX205E/206E/211E/213E/241E normal operation, Figure 2			600		ns
Receiver Output Disable Time		MAX205E/206E/211E/213E/241E normal operation, Figure 2			200		ns
Transmitter Propagation Delay	t_{PLHT}, t_{PHLT}	R_L = 3kΩ, C_L = 2500pF, all transmitters loaded			2		µs
Transition-Region Slew Rate		T_A = +25°C, V_{CC} = 5V, R_L = 3kΩ to 7kΩ, C_L = 50pF to 1000pF, measured from -3V to +3V or +3V to -3V, Figure 3		3	6	30	V/µs
ESD PERFORMANCE: TRANSMITTER OUTPUTS, RECEIVER INPUTS							
ESD-Protection Voltage		Human Body Model			±15		kV
		IEC1000-4-2, Contact Discharge			±8		
		IEC1000-4-2, Air-Gap Discharge			±15		

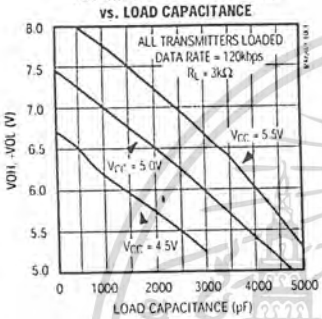
Note 1: MAX211EE_ _ tested with V_{CC} = +5V ±5%.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

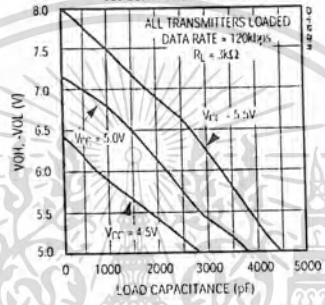
Typical Operating Characteristics

(Typical Operating Circuits, V_{CC} = +5V, T_A = +25°C, unless otherwise noted.)

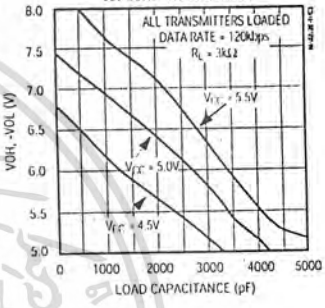
MAX232E
TRANSMITTER OUTPUT VOLTAGE
vs. LOAD CAPACITANCE



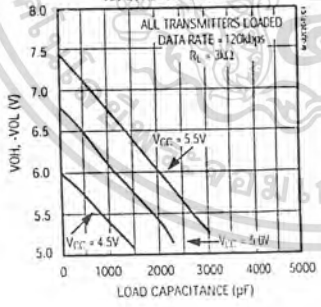
MAX202E/MAX203E
TRANSMITTER OUTPUT VOLTAGE
vs. LOAD CAPACITANCE



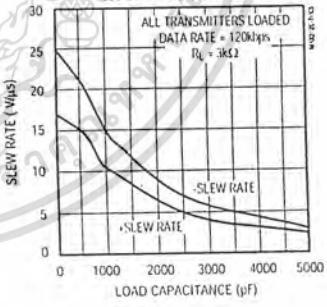
MAX241E
TRANSMITTER OUTPUT VOLTAGE
vs. LOAD CAPACITANCE



MAX211E/MAX213E
TRANSMITTER OUTPUT VOLTAGE
vs. LOAD CAPACITANCE



MAX211E/MAX213E/MAX241E
TRANSMITTER SLEW RATE
vs. LOAD CAPACITANCE

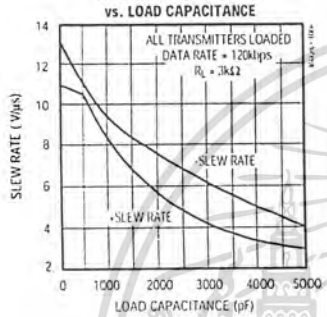


เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

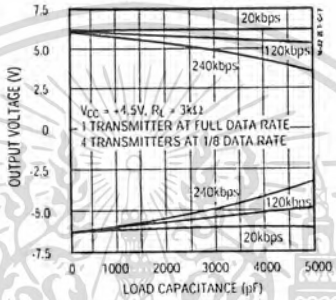
Typical Operating Characteristics (continued)

(Typical Operating Circuits, V_{CC} = +5V, T_A = +25°C, unless otherwise noted.)

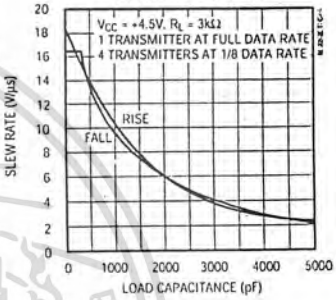
MAX202E/MAX203E/MAX232E
TRANSMITTER SLEW RATE
vs. LOAD CAPACITANCE



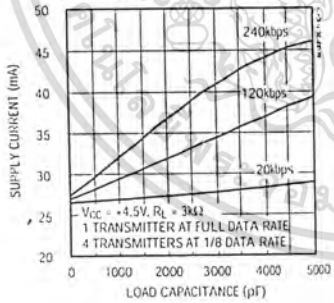
MAX205E–MAX208E
TRANSMITTER OUTPUT VOLTAGE
vs. LOAD CAPACITANCE



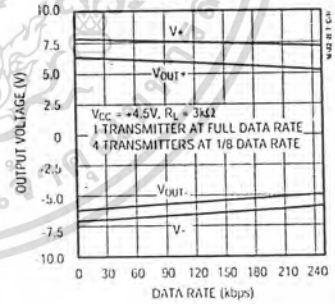
MAX205E–MAX208E
TRANSMITTER SLEW RATE
vs. LOAD CAPACITANCE



MAX205E–MAX208E
SUPPLY CURRENT
vs. LOAD CAPACITANCE



MAX205E–MAX208E
OUTPUT VOLTAGE vs. DATA RATE



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

±15kV ESD-Protected, +5V RS-232 Transceivers

Pin Descriptions

MAX202E/MAX232E

PIN		NAME	FUNCTION
DIP/SO	LCC		
1, 3	2, 4	C1+, C1-	Terminals for positive charge-pump capacitor
2	3	V+	+2V_{CC} voltage generated by the charge pump
4, 5	5, 7	C2+, C2-	Terminals for negative charge-pump capacitor
6	8	V-	-2V_{CC} voltage generated by the charge pump
7, 14	9, 18	T_OUT	RS-232 Driver Outputs
8, 13	10, 17	R_IN	RS-232 Receiver Outputs
9, 12	12, 15	R_OUT	RS-232 Receiver Outputs
10, 11	13, 14	T_IN	RS-232 Driver Inputs
15	19	GND	Ground
16	20	V_{CC}	+4.5V to +5.5V Supply-Voltage Input
—	1, 6, 11, 16	N.C.	No Connect—not internally connected

MAX203E

PIN		NAME	FUNCTION
DIP	SO		
1, 2	1, 2	T_IN	RS-232 Driver Inputs
3, 20	3, 20	R_OUT	RS-232 Receiver Outputs
4, 19	4, 19	R_IN	RS-232 Receiver Inputs
5, 18	5, 18	T_OUT	RS-232 Transmitter Outputs
6, 9	6, 9	GND	Ground
7	7	V_{CC}	+4.5V to +5.5V Supply-Voltage Input
8	13	C1+	Make no connection to this pin.
10, 16	11, 16	C2-	Connect pins together.
12, 17	10, 17	V-	-2V_{CC} voltage generated by the charge pump. Connect pins together.
13	14	C1-	Make no connection to this pin.
14	8	V+	+2V_{CC} voltage generated by the charge pump
11, 15	12, 15	C2+	Connect pins together.

MAX205E

PIN	NAME	FUNCTION
1–4, 19	T_OUT	RS-232 Driver Outputs
5, 10, 13, 18, 24	R_IN	RS-232 Receiver Inputs
6, 9, 14, 17, 23	R_OUT	TTL/CMOS Receiver Outputs. All receivers are inactive in shutdown.
7, 8, 15, 16, 22	T_IN	TTL/CMOS Driver Inputs. Internal pull-ups to V_{CC}.
11	GND	Ground
12	V_{CC}	+4.75V to +5.25V Supply Voltage
20	$\overline{EN}$	Receiver Enable—active low
21	SHDN	Shutdown Control—active high

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Pin Descriptions (continued)

MAX206E

PIN	NAME	FUNCTION
1, 2, 3, 24	T_OUT	RS-232 Driver Outputs
4, 16, 23	R_IN	RS-232 Receiver Inputs
5, 17, 22	R_OUT	TTL/CMOS Receiver Outputs. All receivers are inactive in shutdown.
6, 7, 18, 19	T_IN	TTL/CMOS Driver Inputs. Internal pull-ups to V_{CC}.
8	GND	Ground
9	V_{CC}	+4.5V to +5.5V Supply Voltage
10, 12	C1+, C1-	Terminals for positive charge-pump capacitor
11	V+	$+2V_{CC}$ generated by the charge pump
13, 14	C2+, C2-	Terminals for negative charge-pump capacitor
15	V-	$-2V_{CC}$ generated by the charge pump
20	$\overline{EN}$	Receiver Enable—active low
21	SHDN	Shutdown Control—active high

MAX207E

PIN	NAME	FUNCTION
1, 2, 3, 20, 24	T_OUT	RS-232 Driver Outputs
4, 16, 23	R_IN	RS-232 Receiver Inputs
5, 17, 22	R_OUT	TTL/CMOS Receiver Outputs. All receivers are inactive in shutdown.
6, 7, 18, 19, 21	T_IN	TTL/CMOS Driver Inputs. Internal pull-ups to V_{CC}.
8	GND	Ground
9	V_{CC}	+4.75V to +5.25V Supply Voltage
10, 12	C1+, C1-	Terminals for positive charge-pump capacitor
11	V+	$+2V_{CC}$ generated by the charge pump
13, 14	C2+, C2-	Terminals for negative charge-pump capacitor
15	V-	$-2V_{CC}$ generated by the charge pump

MAX208E

PIN	NAME	FUNCTION
1, 2, 20, 24	T_OUT	RS-232 Driver Outputs
3, 7, 16, 23	R_IN	RS-232 Receiver Inputs
4, 6, 17, 22	R_OUT	TTL/CMOS Receiver Outputs. All receivers are inactive in shutdown.
5, 18, 19, 21	T_IN	TTL/CMOS Driver Inputs. Internal pull-ups to V_{CC}.
8	GND	Ground
9	V_{CC}	+4.5V to +5.5V Supply Voltage
10, 12	C1+, C1-	Terminals for positive charge-pump capacitor
11	V+	$+2V_{CC}$ generated by the charge pump
13, 14	C2+, C2-	Terminals for negative charge-pump capacitor
15	V-	$-2V_{CC}$ generated by the charge pump

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Applications Information

Capacitor Selection

The capacitor type used for C1–C4 is not critical for proper operation. The MAX202E, MAX206–MAX208E, MAX211E, and MAX213E require 0.1µF capacitors, and the MAX232E and MAX241E require 1µF capacitors, although in all cases capacitors up to 10µF can be used without harm. Ceramic, aluminum-electrolytic, or tantalum capacitors are suggested for the 1µF capacitors, and ceramic dielectrics are suggested for the 0.1µF capacitors. When using the minimum recommended capacitor values, make sure the capacitance value does not degrade excessively as the operating temperature varies. If in doubt, use capacitors with a larger (e.g., 2x) nominal value. The capacitors' effective series resistance (ESR), which usually rises at low temperatures, influences the amount of ripple on V+ and V-.

Use larger capacitors (up to 10µF) to reduce the output impedance at V+ and V-. This can be useful when "stealing" power from V+ or from V-. The MAX203E and MAX205E have internal charge-pump capacitors.

Bypass VCC to ground with at least 0.1µF. In applications sensitive to power-supply noise generated by the charge pumps, decouple VCC to ground with a capacitor the same size as (or larger than) the charge-pump capacitors (C1–C4).

V+ and V- as Power Supplies

A small amount of power can be drawn from V+ and V-, although this will reduce both driver output swing and noise margins. Increasing the value of the charge-pump capacitors (up to 10µF) helps maintain performance when power is drawn from V+ or V-.

Driving Multiple Receivers

Each transmitter is designed to drive a single receiver. Transmitters can be paralleled to drive multiple receivers.

Driver Outputs when Exiting Shutdown

The driver outputs display no ringing or undesirable transients as they come out of shutdown.

High Data Rates

These transceivers maintain the RS-232 ±5.0V minimum driver output voltages at data rates of over 120kbps. For data rates above 120kbps, refer to the Transmitter Output Voltage vs. Load Capacitance graphs in the *Typical Operating Characteristics*. Communication at these high rates is easier if the capacitive loads on the transmitters are small; i.e., short cables are best.

Table 2. Summary of EIA/TIA-232E, V.28 Specifications

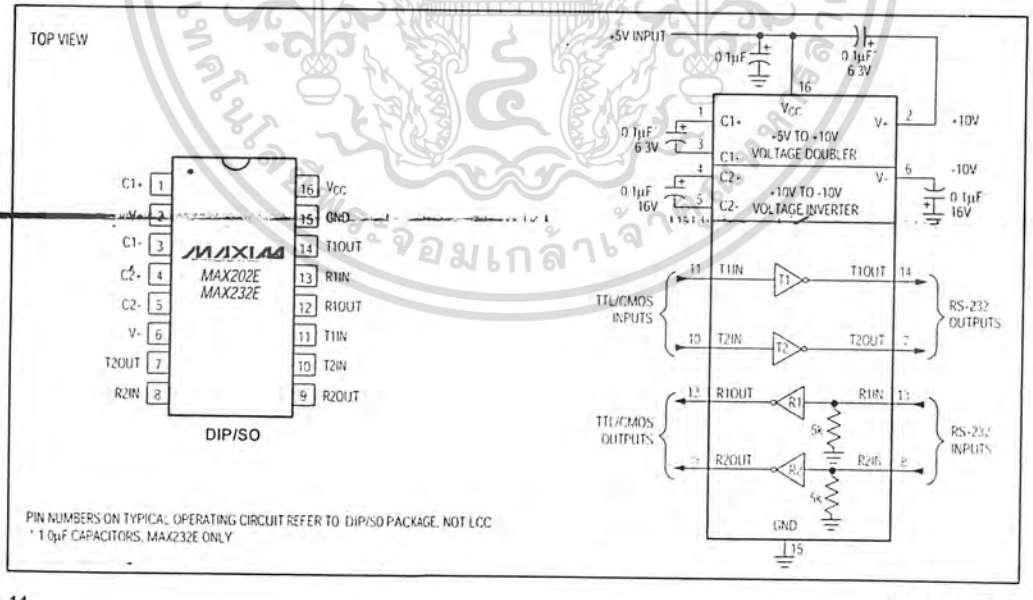
PARAMETER		CONDITIONS	EIA/TIA-232E, V.28 SPECIFICATIONS
Driver Output Voltage	0 Level	3kΩ to 7kΩ load	-5V to +15V
	1 Level	3kΩ to 7kΩ load	+5V to -15V
Driver Output Level, Max		No load	±25V
Data Rate		3kΩ ≤ R_L ≤ 7kΩ, C_L ≤ 2500pF	Up to 20kbps
Receiver Input Voltage	0 Level		-3V to +15V
	1 Level		+3V to -15V
Receiver Input Level			±25V
Instantaneous Slew Rate, Max		3kΩ ≤ R_L ≤ 7kΩ, C_L ≤ 2500pF	30V/µs
Driver Output Short-Circuit Current, Max			100mA
Transition Rate on Driver Output		V.28	1ms or 3% of the period
		EIA/TIA-232E	4% of the period
Driver Output Resistance		-2V < V_{OUT} < +2V	300Ω

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Table 3. DB9 Cable Connections Commonly Used for EIA/TIAE-232E and V.24 Asynchronous Interfaces

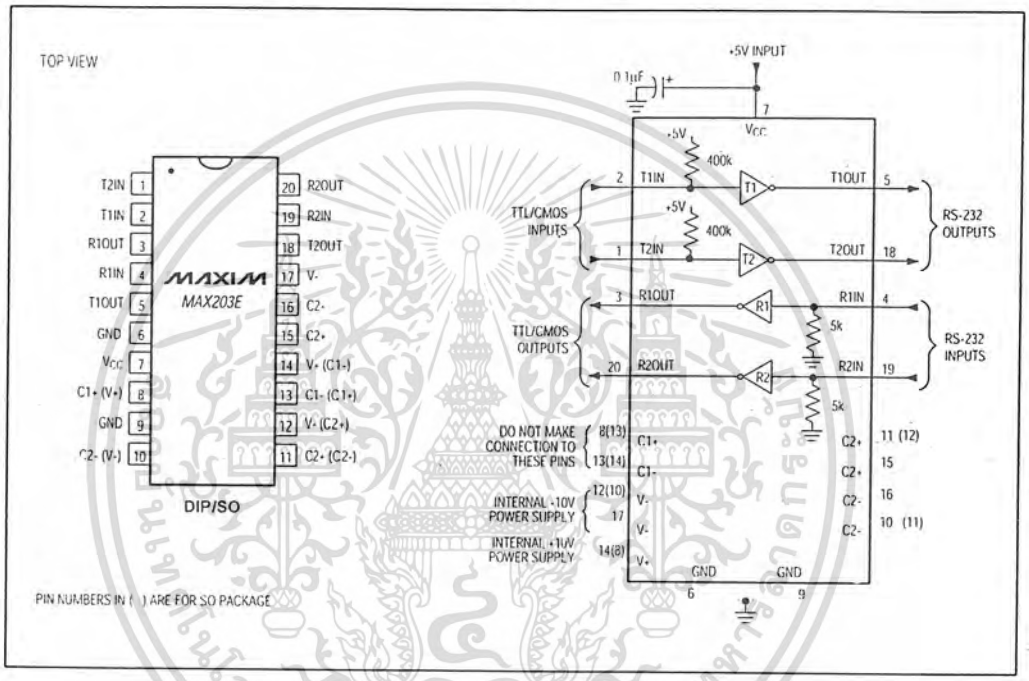
PIN	CONNECTION	
1	Received Line Signal Detector (sometimes called Carrier Detect, DCD)	Handshake from DCE
2	Receive Data (RD)	Data from DCE
3	Transmit Data (TD)	Data from DTE
4	Data Terminal Ready	Handshake from DTE
5	Signal Ground	Reference point for signals
6	Data Set Ready (DSR)	Handshake from DCE
7	Request to Send (RTS)	Handshake from DTE
8	Clear to Send (CTS)	Handshake from DCE
9	Ring Indicator	Handshake from DCE

Pin Configurations and Typical Operating Circuits (continued)



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Pin Configurations and Typical Operating Circuits (continued)



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Ordering Information

PART	TEMP. RANGE	PIN-PACKAGE
MAX202ECPE	0°C to +70°C	16 Plastic DIP
MAX202ECSE	0°C to +70°C	16 Narrow SO
MAX202ECWE	0°C to +70°C	16 Wide SO
MAX202EC/D	0°C to +70°C	Dice*
MAX202EEPE	-40°C to +85°C	16 Plastic DIP
MAX202EESE	-40°C to +85°C	16 Narrow SO
MAX202EEWE	-40°C to +85°C	16 Wide SO
MAX203ECPP	0°C to +70°C	20 Plastic DIP
MAX203ECWP	0°C to +70°C	20 SO
MAX203EEPP	-40°C to +85°C	20 Plastic DIP
MAX203EEWP	-40°C to +85°C	20 SO
MAX205ECPG	0°C to +70°C	24 Wide Plastic DIP
MAX205EEPG	-40°C to +85°C	24 Wide Plastic DIP
MAX206ECNG	0°C to +70°C	24 Narrow Plastic DIP
MAX206ECWG	0°C to +70°C	24 SO
MAX206ECAG	0°C to +70°C	24 SSOP
MAX206EENG	-40°C to +85°C	24 Narrow Plastic DIP
MAX206EEWG	-40°C to +85°C	24 SO
MAX206EEAG	-40°C to +85°C	24 SSOP
MAX207ECNG	0°C to +70°C	24 Narrow Plastic DIP
MAX207ECWG	0°C to +70°C	24 SO
MAX207ECAG	0°C to +70°C	24 SSOP
MAX207EENG	-40°C to +85°C	24 Narrow Plastic DIP
MAX207EEWG	-40°C to +85°C	24 SO
MAX207EEAG	-40°C to +85°C	24 SSOP

PART	TEMP. RANGE	PIN-PACKAGE
MAX208ECNG	0°C to +70°C	24 Narrow Plastic DIP
MAX208ECWG	0°C to +70°C	24 SO
MAX208ECAG	0°C to +70°C	24 SSOP
MAX208EENG	-40°C to +85°C	24 Narrow Plastic DIP
MAX208EEWG	-40°C to +85°C	24 SO
MAX208EEAG	-40°C to +85°C	24 SSOP
MAX211ECWI	0°C to +70°C	28 SO
MAX211ECAI	0°C to +70°C	28 SSOP
MAX211EEWI	-40°C to +85°C	28 SO
MAX211EEAI	-40°C to +85°C	28 SSOP
MAX213ECWI	0°C to +70°C	28 SO
MAX213ECAI	0°C to +70°C	28 SSOP
MAX213EEWI	-40°C to +85°C	28 SO
MAX213EEAI	-40°C to +85°C	28 SSOP
MAX232ECPE	0°C to +70°C	16 Plastic DIP
MAX232ECSE	0°C to +70°C	16 Narrow SO
MAX232ECWE	0°C to +70°C	16 Wide SO
MAX232EC/D	0°C to +70°C	Dice*
MAX232EEPE	-40°C to +85°C	16 Plastic DIP
MAX232EESE	-40°C to +85°C	16 Narrow SO
MAX232EEWE	-40°C to +85°C	16 Wide SO
MAX241ECWI	0°C to +70°C	28 SO
MAX241ECAI	0°C to +70°C	28 SSOP
MAX241EEWI	-40°C to +85°C	28 SO
MAX241EEAI	-40°C to +85°C	28 SSOP

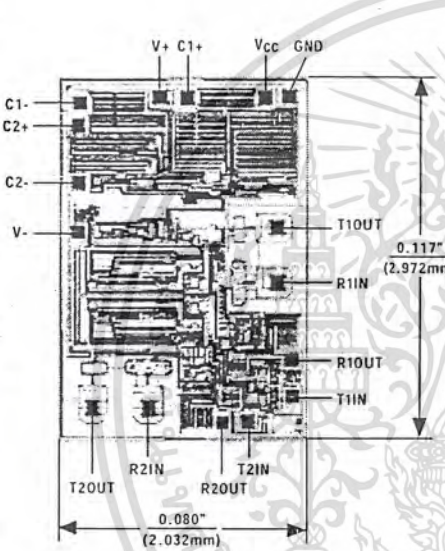
*Dice are specified at T_A = +25°C.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

±15kV ESD-Protected, +5V RS-232 Transceivers

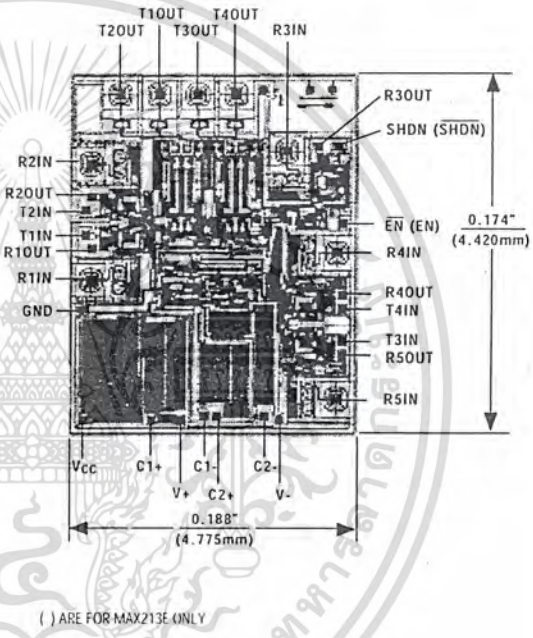
Chip Topographies

MAX202E/MAX232E



TRANSISTOR COUNT: 123
SUBSTRATE CONNECTED TO GND

MAX211E/MAX213E/MAX241E



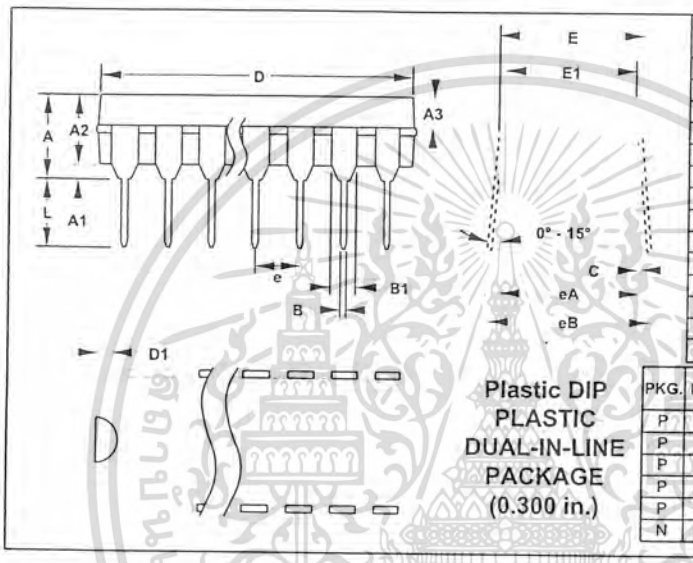
TRANSISTOR COUNT: 542
SUBSTRATE CONNECTED TO GND

Chip Information

MAX205E/MAX206E/MAX207E/MAX208E
TRANSISTOR COUNT: 328
SUBSTRATE CONNECTED TO GND

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Package Information

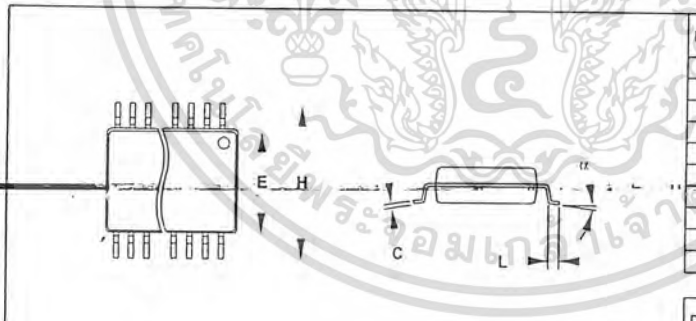


Plastic DIP
PLASTIC
DUAL-IN-LINE
PACKAGE
(0.300 in.)

DIM	INCHES		MILLIMETERS	
	MIN	MAX	MIN	MAX
A	–	0.200	–	5.08
A1	0.015	–	0.38	–
A2	0.125	0.175	3.18	4.45
A3	0.055	0.080	1.40	2.03
B	0.016	0.022	0.41	0.56
B1	0.045	0.065	1.14	1.65
C	0.008	0.012	0.20	0.30
D1	0.005	0.080	0.13	2.03
E	0.300	0.325	7.62	8.26
E1	0.240	0.310	6.10	7.87
e	0.100	–	2.54	–
eA	0.300	–	7.62	–
eB	–	0.400	–	10.16
L	0.115	0.150	2.92	3.81

PKG.	DIM	PINS	INCHES		MILLIMETERS	
			MIN	MAX	MIN	MAX
P	D	8	0.348	0.390	8.84	9.91
P	D	14	0.735	0.765	18.67	19.43
P	D	16	0.745	0.765	18.92	19.43
P	D	18	0.885	0.915	22.48	23.24
P	D	20	1.015	1.045	25.78	26.54
N	D	24	1.14	1.265	28.96	32.13

21-0043A



SSOP
SHRINK
SMALL-OUTLINE
PACKAGE

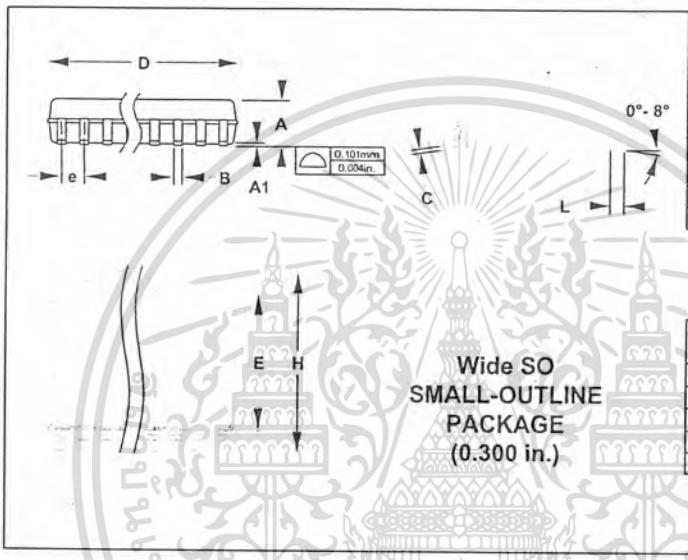
DIM	INCHES		MILLIMETERS	
	MIN	MAX	MIN	MAX
A	0.068	0.078	1.73	1.99
A1	0.002	0.008	0.05	0.21
B	0.010	0.015	0.25	0.38
C	0.004	0.008	0.09	0.20
D	SEE VARIATIONS			
E	0.205	0.209	5.20	5.38
e	0.0256 BSC		0.65 BSC	
H	0.301	0.311	7.65	7.90
L	0.025	0.037	0.63	0.95
α	0°	8°	0°	8°

DIM	PINS	INCHES		MILLIMETERS	
		MIN	MAX	MIN	MAX
D	14	0.239	0.249	6.07	6.33
D	16	0.239	0.249	6.07	6.33
D	20	0.278	0.289	7.07	7.33
D	24	0.317	0.328	8.07	8.33
D	28	0.397	0.407	10.07	10.33

21-0056A

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Package Information (continued)



DIM	INCHES		MILLIMETERS	
	MIN	MAX	MIN	MAX
A	0.093	0.104	2.35	2.65
A1	0.004	0.012	0.10	0.30
B	0.014	0.019	0.35	0.49
C	0.009	0.013	0.23	0.32
E	0.291	0.299	7.40	7.60
e	0.050		1.27	
H	0.394	0.419	10.00	10.65
L	0.016	0.050	0.40	1.27

DIM	PINS	INCHES		MILLIMETERS	
		MIN	MAX	MIN	MAX
D	16	0.398	0.413	10.10	10.50
D	18	0.447	0.463	11.35	11.75
D	20	0.496	0.512	12.60	13.00
D	24	0.598	0.614	15.20	15.60
D	28	0.697	0.713	17.70	18.10

21-0042A

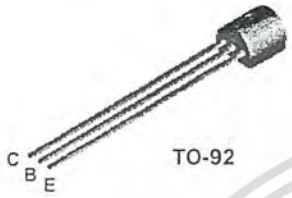
Maxim cannot assume responsibility for use of any circuitry other than circuitry entirely embodied in a Maxim product. No circuit patent licenses are implied. Maxim reserves the right to change the circuitry and specifications without notice at any time.

© 1996 Maxim Integrated Products Printed USA **MAXIM** is a registered trademark of Maxim Integrated Products.

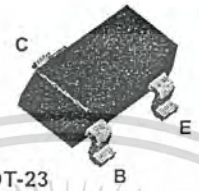
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

FAIRCHILD
SEMICONDUCTOR™

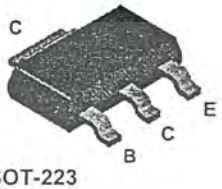
PN2222A MMBT2222A PZT2222A



TO-92

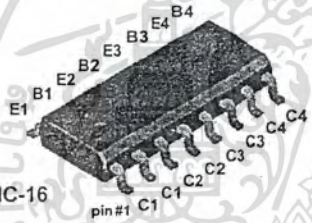


SOT-23
Mark: 1P

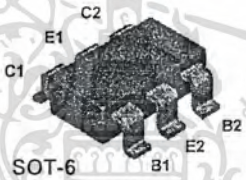


SOT-223

MMPQ2222 NMT2222



SOIC-16



SOT-6
Mark: .1B

NPN General Purpose Amplifier

This device is for use as a medium power amplifier and switch requiring collector currents up to 500 mA. Sourced from Process 19.

Absolute Maximum Ratings*

TA = 25°C unless otherwise noted

Symbol	Parameter	Value	Units
V_{CEO}	Collector-Emitter Voltage	40	V
V_{CBO}	Collector-Base Voltage	75	V
V_{EBO}	Emitter-Base Voltage	6.0	V
I_C	Collector Current - Continuous	1.0	A
T_J, T_{stg}	Operating and Storage Junction Temperature Range	-55 to +150	°C

*These ratings are limiting values above which the serviceability of any semiconductor device may be impaired.

NOTES:
1) These ratings are based on a maximum junction temperature of 150 degrees C.
2) These are steady state limits. The factory should be consulted on applications involving pulsed or low duty cycle operations.

©1997 Fairchild Semiconductor Corporation

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

NPN General Purpose Amplifier
(continued).

Electrical Characteristics
TA = 25°C unless otherwise noted

Symbol	Parameter	Test Conditions	Min	Max	Units
OFF CHARACTERISTICS					
$V_{(BR)CEO}$	Collector-Emitter Breakdown Voltage*	I_C = 10 mA, I_B = 0	40		V
$V_{(BR)CBO}$	Collector-Base Breakdown Voltage	I_C = 10 µA, I_E = 0	75		V
$V_{(BR)EBO}$	Emitter-Base Breakdown Voltage	I_E = 10 µA, I_C = 0	6.0		V
I_{CEX}	Collector Cutoff Current	V_{CE} = 60 V, $V_{EB(OFF)}$ = 3.0 V		10	nA
I_{CBO}	Collector Cutoff Current	V_{CB} = 60 V, I_E = 0 V_{CB} = 60 V, I_E = 0, T_A = 150°C		0.01 10	µA µA
I_{EBO}	Emitter Cutoff Current	V_{EB} = 3.0 V, I_C = 0		10	nA
I_{BL}	Base Cutoff Current	V_{CE} = 60 V, $V_{EB(OFF)}$ = 3.0 V		20	nA
ON CHARACTERISTICS					
h_{FE}	DC Current Gain	I_C = 0.1 mA, V_{CE} = 10 V I_C = 1.0 mA, V_{CE} = 10 V I_C = 10 mA, V_{CE} = 10 V I_C = 10 mA, V_{CE} = 10 V, T_A = -55°C I_C = 150 mA, V_{CE} = 10 V* I_C = 150 mA, V_{CE} = 1.0 V* I_C = 500 mA, V_{CE} = 10 V*	35 50 75 35 100 50 40	300	
$V_{CE(sat)}$	Collector-Emitter Saturation Voltage*	I_C = 150 mA, I_B = 15 mA I_C = 500 mA, I_B = 50 mA		0.3 1.0	V V
$V_{BE(sat)}$	Base-Emitter Saturation Voltage*	I_C = 150 mA, I_B = 15 mA I_C = 500 mA, I_B = 50 mA	0.6	1.2 2.0	V V
SMALL SIGNAL CHARACTERISTICS (except MMPQ2222 and NMT2222)					
f_T	Current Gain - Bandwidth Product	I_C = 20 mA, V_{CE} = 20 V, f = 100 MHz	300		MHz
C_{obo}	Output Capacitance	V_{CB} = 10 V, I_E = 0, f = 100 kHz		8.0	pF
C_{ibo}	Input Capacitance	V_{EB} = 0.5 V, I_C = 0, f = 100 kHz		25	pF
rb'C_C	Collector Base Time Constant	I_C = 20 mA, V_{CB} = 20 V, f = 31.8 MHz		150	pS
NF	Noise Figure	I_C = 100 µA, V_{CE} = 10 V, R_S = 1.0 kΩ, f = 1.0 kHz		4.0	dB
Re(h_{ie})	Real Part of Common-Emitter High Frequency Input Impedance	I_C = 20 mA, V_{CE} = 20 V, f = 300 MHz		60	Ω
SWITCHING CHARACTERISTICS (except MMPQ2222 and NMT2222)					
t_d	Delay Time	V_{CC} = 30 V, $V_{BE(OFF)}$ = 0.5 V,		10	ns
t_r	Rise Time	I_C = 150 mA, I_{B1} = 15 mA		25	ns
t_s	Storage Time	V_{CC} = 30 V, I_C = 150 mA,		225	ns
t_f	Fall Time	I_{B1} = I_{B2} = 15 mA		60	ns

*Pulse Test: Pulse Width ≤ 300 µs, Duty Cycle ≤ 2.0%

Spice Model

NPN (Is=14.34f Xti=3 Eg=1.11 Vaf=74.03 Bf=255.9 Ne=1.307 Ise=14.34f Ikf=.2847 Xtb=1.5 Br=6.092 Nc=2 Isc=0 Ikr=0 Rc=1 Cjc=7.306p Mjc=.3416 Vjc=.75 Fc=.5 Cje=22.01p Mje=.377 Vje=.75 Tr=46.91n Tf=411.1p Itf=.6 Vtf=1.7 Xtf=3 Rb=10)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

NPN General Purpose Amplifier
(continued)

Thermal Characteristics
TA = 25°C unless otherwise noted

Symbol	Characteristic	Max		Units
		PN2222A	*PZT2222A	
P_D	Total Device Dissipation Derate above 25°C	625 5.0	1,000 8.0	mW mW/°C
$R_{\theta JC}$	Thermal Resistance, Junction to Case	83.3		°C/W
$R_{\theta JA}$	Thermal Resistance, Junction to Ambient	200	125	°C/W

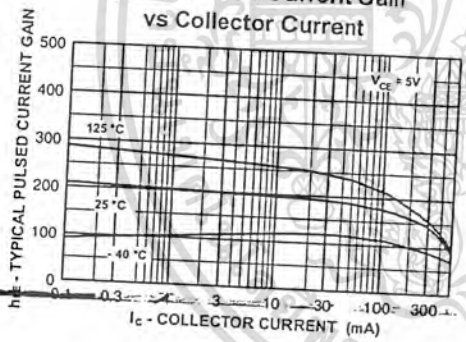
Symbol	Characteristic	Max		Units
		**MMBT2222A	MMPQ2222	
P_D	Total Device Dissipation Derate above 25°C	350 2.8	1,000 8.0	mW mW/°C
$R_{\theta JA}$	Thermal Resistance, Junction to Ambient Effective 4 Die Each Die	357	125 240	°C/W °C/W °C/W

*Device mounted on FR-4 PCB 36 mm X 18 mm X 1.5 mm; mounting pad for the collector lead min. 6 cm².

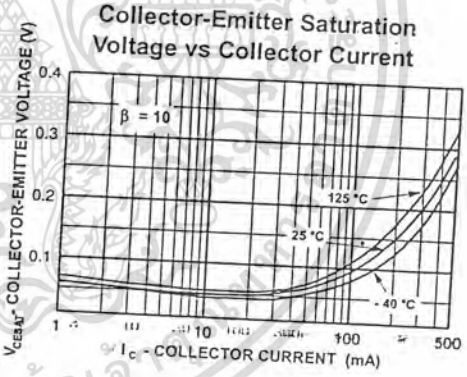
**Device mounted on FR-4 PCB 1.6" X 1.6" X 0.06."

Typical Characteristics

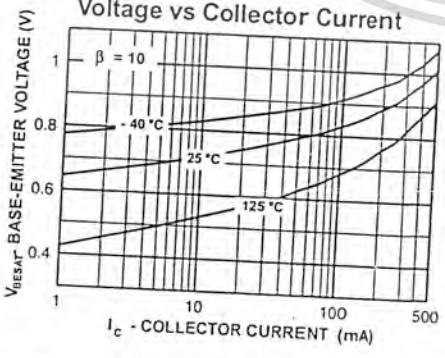
Typical Pulsed Current Gain vs Collector Current



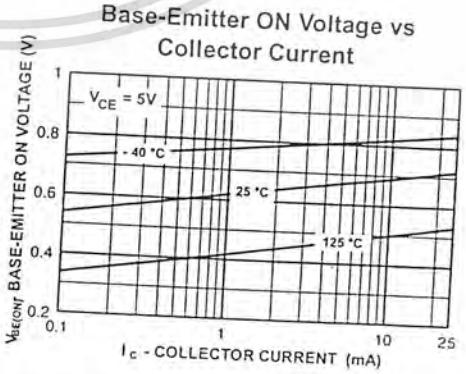
Collector-Emitter Saturation Voltage vs Collector Current



Base-Emitter Saturation Voltage vs Collector Current



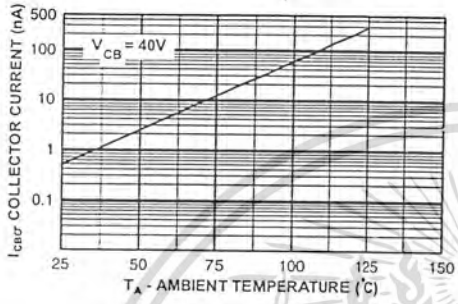
Base-Emitter ON Voltage vs Collector Current



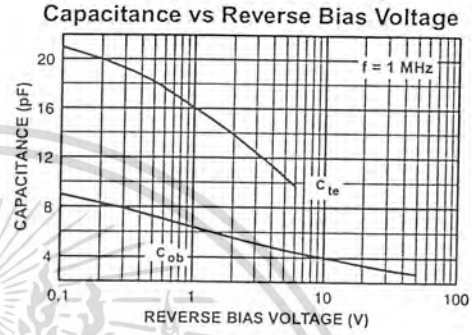
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Typical Characteristics (continued)

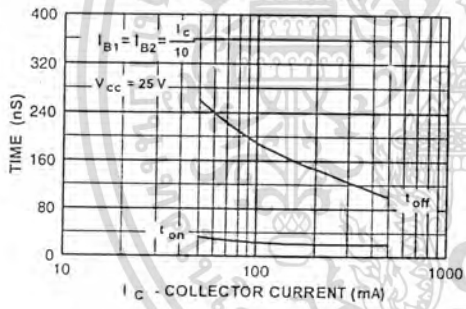
Collector-Cutoff Current vs Ambient Temperature



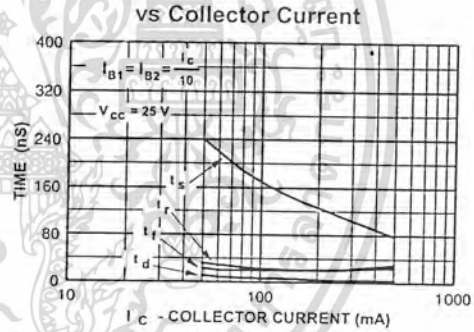
Emitter Transition and Output Capacitance vs Reverse Bias Voltage



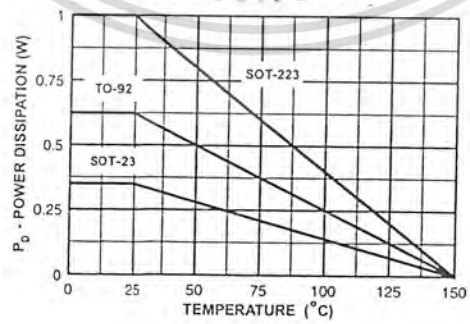
Turn On and Turn Off Times vs Collector Current



Switching Times vs Collector Current



Power Dissipation vs Ambient Temperature



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Test Circuits

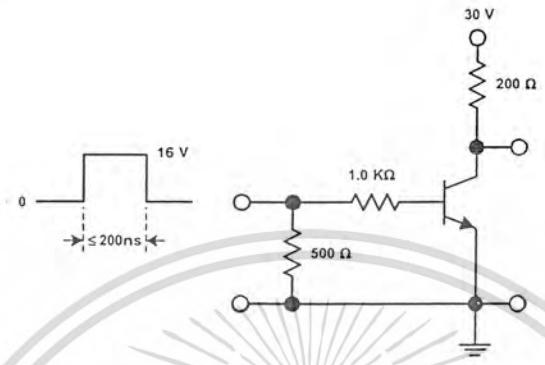


FIGURE 1: Saturated Turn-On Switching Time

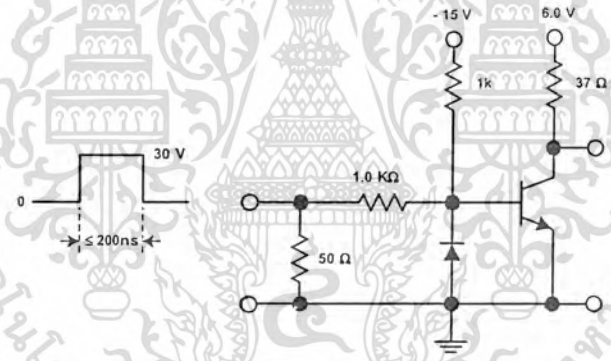


FIGURE 2: Saturated Turn-Off Switching Time

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

TRADEMARKS

The following are registered and unregistered trademarks Fairchild Semiconductor owns or is authorized to use and is not intended to be an exhaustive list of all such trademarks.

ACEx™
Bottomless™
CoolFET™
CROSSVOLT™
E²CMOS™
FACT™
FACT Quiet Series™
FAST®
FASTr™
GTO™
HiSeC™
ISOPLANAR™
MICROWIRE™
POP™
PowerTrench®
QFET™
QS™
Quiet Series™
SuperSOT™-3
SuperSOT™-6
SuperSOT™-8
SyncFET™
TinyLogic™
UHC™
VCX™

DISCLAIMER

FAIRCHILD SEMICONDUCTOR RESERVES THE RIGHT TO MAKE CHANGES WITHOUT FURTHER NOTICE TO ANY PRODUCTS HEREIN TO IMPROVE RELIABILITY, FUNCTION OR DESIGN. FAIRCHILD DOES NOT ASSUME ANY LIABILITY ARISING OUT OF THE APPLICATION OR USE OF ANY PRODUCT OR CIRCUIT DESCRIBED HEREIN; NEITHER DOES IT CONVEY ANY LICENSE UNDER ITS PATENT RIGHTS, NOR THE RIGHTS OF OTHERS.

LIFE SUPPORT POLICY

FAIRCHILD'S PRODUCTS ARE NOT AUTHORIZED FOR USE AS CRITICAL COMPONENTS IN LIFE SUPPORT DEVICES OR SYSTEMS WITHOUT THE EXPRESS WRITTEN APPROVAL OF FAIRCHILD SEMICONDUCTOR CORPORATION.

As used herein:

1. Life support devices or systems are devices or systems which, (a) are intended for surgical implant into the body, or (b) support or sustain life, or (c) whose failure to perform when properly used in accordance with instructions for use provided in the labeling, can be reasonably expected to result in significant injury to the user.

2. A critical component is any component of a life support device or system whose failure to perform can be reasonably expected to cause the failure of the life support device or system, or to affect its safety or effectiveness.

PRODUCT STATUS DEFINITIONS

Definition of Terms

Datasheet Identification	Product Status	Definition
Advance Information	Formative or In Design	This datasheet contains the design specifications for product development. Specifications may change in any manner without notice.
Preliminary	First Production	This datasheet contains preliminary data, and supplementary data will be published at a later date. Fairchild Semiconductor reserves the right to make changes at any time without notice in order to improve design.
No Identification Needed	Full Production	This datasheet contains final specifications. Fairchild Semiconductor reserves the right to make changes at any time without notice in order to improve design.
Obsolete	Not In Production	This datasheet contains specifications on a product that has been discontinued by Fairchild semiconductor. The datasheet is printed for reference information only.

Rev. E

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

FAIRCHILD
SEMICONDUCTOR TM

April 1995

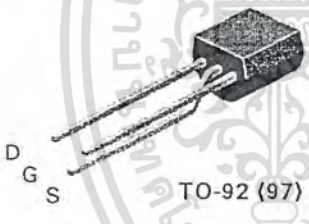
BS170 / MMBF170
N-Channel Enhancement Mode Field Effect Transistor

General Description

These N-Channel enhancement mode field effect transistors are produced using Fairchild's proprietary, high cell density, DMOS technology. These products have been designed to minimize on-state resistance while provide rugged, reliable, and fast switching performance. They can be used in most applications requiring up to 500mA DC. These products are particularly suited for low voltage, low current applications such as small servo motor control, power MOSFET gate drivers, and other switching applications.

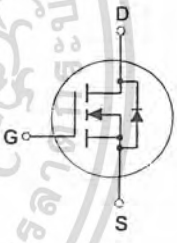
Features

- High density cell design for low $R_{DS(ON)}$.
- Voltage controlled small signal switch.
- Rugged and reliable.
- High saturation current capability.



TO-92 (97)

SOT-23



Absolute Maximum Ratings T_A=25°C unless otherwise noted

Symbol	Parameter	BS170	MMBF170	Units
V_{DSS}	Drain-Source Voltage	60		V
V_{DGR}	Drain-Gate Voltage ($R_{GS} \leq 1M\Omega$)	60		V
V_{GSS}	Gate-Source Voltage	±20		V
I_D	Drain Current - Continuous	500	500	mA
	- Pulsed	1200	800	
P_D	Maximum Power Dissipation	830	300	mW
	Derate Above 25 °C	6.6	2.4	mW/°C
T_J, T_{STG}	Operating and Storage Temperature Range	-55 to 150		°C
T_L	Maximum Lead Temperature for Soldering Purposes, 1/16" from Case for 10 Seconds	300		°C
THERMAL CHARACTERISTICS				
$R_{\theta JA}$	Thermal Resistacne, Junction-to-Ambient	150	417	°C/W

© 1997 Fairchild Semiconductor Corporation

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Electrical Characteristics (T_A = 25°C unless otherwise noted)

Symbol	Parameter	Conditions	Type	Min	Typ	Max	Units
OFF CHARACTERISTICS							
BV_{DSS}	Drain-Source Breakdown Voltage	V_{GS} = 0 V, I_D = 100 µA	All	60			V
I_{DSS}	Zero Gate Voltage Drain Current	V_{DS} = 25 V, V_{GS} = 0 V	All			0.5	µA
I_{GSSF}	Gate - Body Leakage, Forward	V_{GS} = 15 V, V_{DS} = 0 V	All			10	nA
ON CHARACTERISTICS (Note 1)							
$V_{GS(th)}$	Gate Threshold Voltage	V_{DS} = V_{GS}, I_D = 1 mA	All	0.8	2.1	3	V
$R_{DS(ON)}$	Static Drain-Source On-Resistance	V_{GS} = 10 V, I_D = 200 mA	All		1.2	5	Ω
g_{FS}	Forward Transconductance	V_{DS} = 10 V, I_D = 200 mA	BS170		320		mS
		$V_{DS} \geq 2\ V_{DS(on)}$, I_D = 200 mA	MMBF170		320		
DYNAMIC CHARACTERISTICS							
C_{iss}	Input Capacitance	V_{DS} = 10 V, V_{GS} = 0 V, f = 1.0 MHz	All		24	40	pF
C_{oss}	Output Capacitance		All		17	30	pF
C_{rss}	Reverse Transfer Capacitance		All		7	10	pF
SWITCHING CHARACTERISTICS (Note 1)							
t_{on}	Turn-On Time	V_{DD} = 25 V, I_D = 200 m A, V_{GS} = 10 V, R_{GEN} = 25 Ω	BS170			10	ns
		V_{DD} = 25 V, I_D = 500 mA, V_{GS} = 10 V, R_{GEN} = 50 Ω	MMBF170			10	
t_{off}	Turn-Off Time	V_{DD} = 25 V, I_D = 200 m A, V_{GS} = 10 V, R_{GEN} = 25 Ω	BS170			10	ns
		V_{DD} = 25 V, I_D = 500 mA, V_{GS} = 10 V, R_{GEN} = 50 Ω	MMBF170			10	

Note:
1. Pulse Test: Pulse Width ≤ 300us, Duty Cycle ≤ 2.0%.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Typical Electrical Characteristics

BS170 / MMBF170

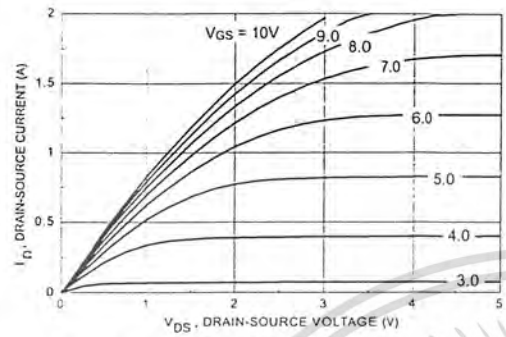


Figure 1. On-Region Characteristics

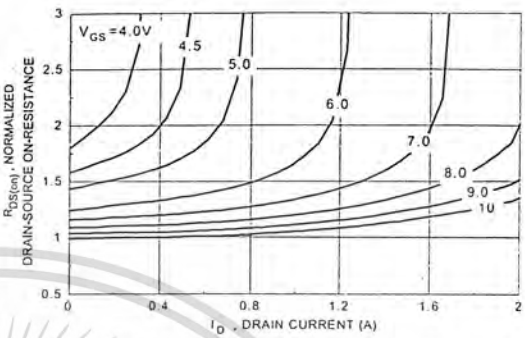


Figure 2. On-Resistance Variation with Gate Voltage and Drain Current.

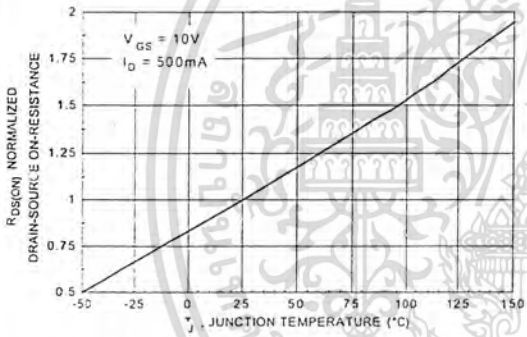


Figure 3. On-Resistance Variation with Temperature.

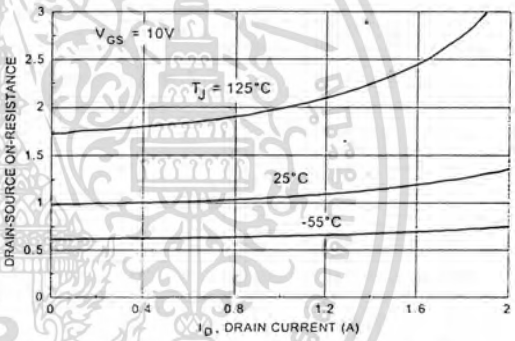


Figure 4. On-Resistance Variation with Drain Current and Temperature.

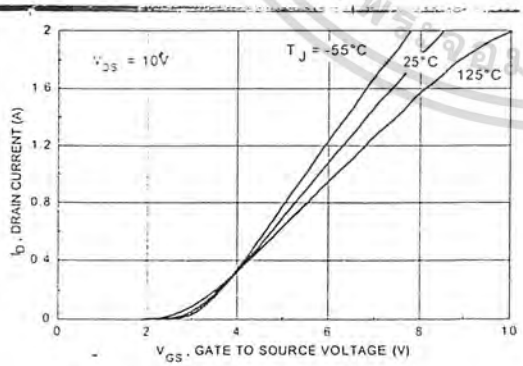


Figure 5. Transfer Characteristics.

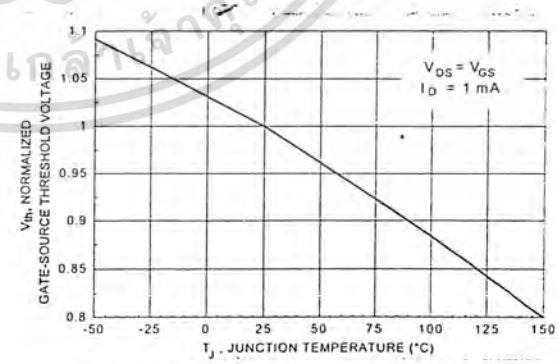


Figure 6. Gate Threshold Variation with Temperature.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Typical Electrical Characteristics (continued)

BS170 / MMBF170

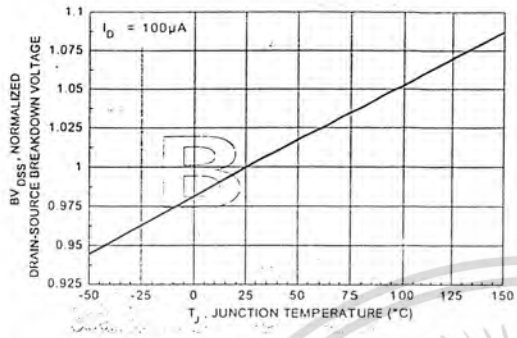


Figure 7. Breakdown Voltage Variation with Temperature.

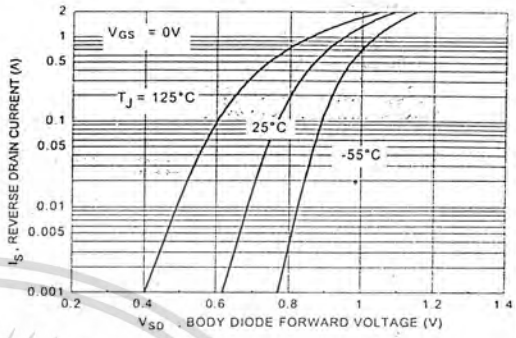


Figure 8. Body Diode Forward Voltage Variation with Current and Temperature.

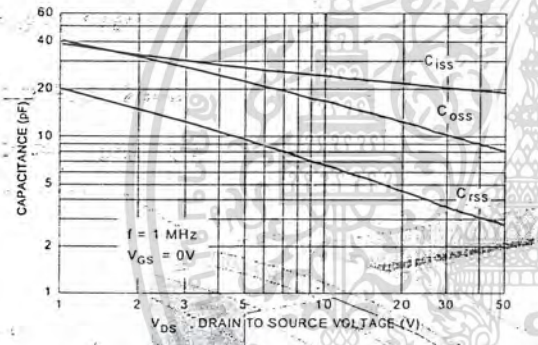


Figure 9. Capacitance Characteristics.

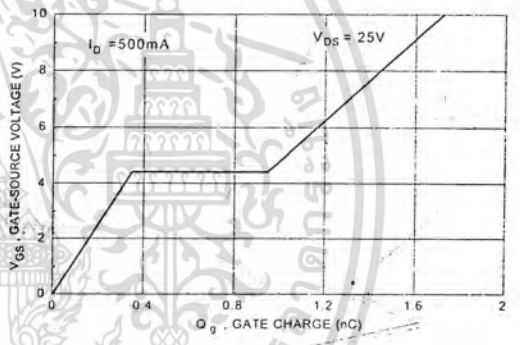


Figure 10. Gate Charge Characteristics.

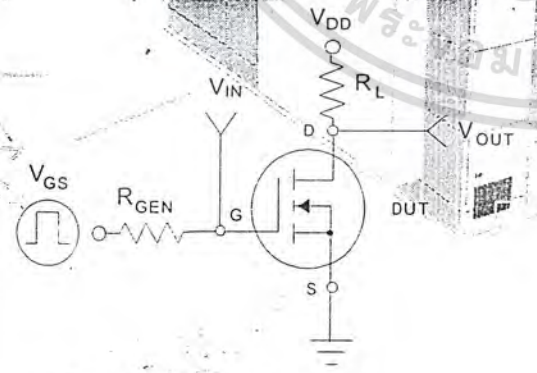


Figure 11. Switching Test Circuit

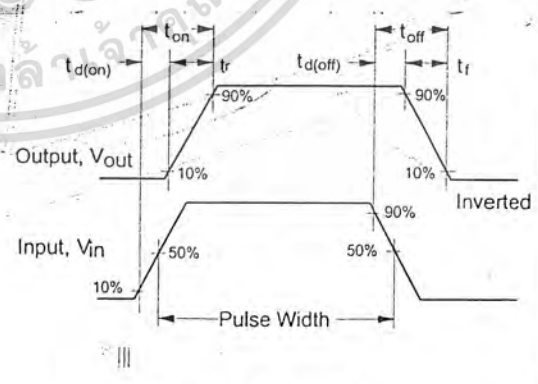


Figure 12. Switching Waveforms.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Typical Electrical Characteristics (continued)

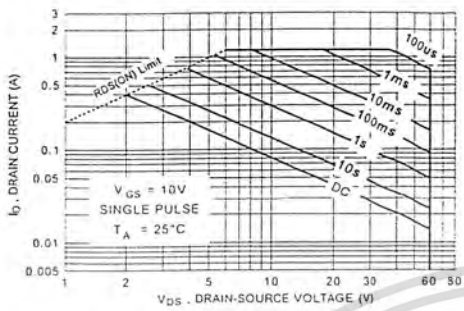


Figure 13. BS170 Maximum Safe Operating Area

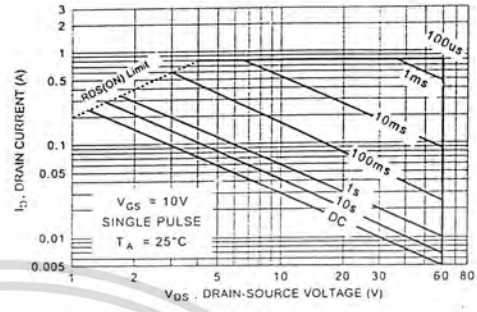


Figure 14. MMBF170 Maximum Safe Operating Area.

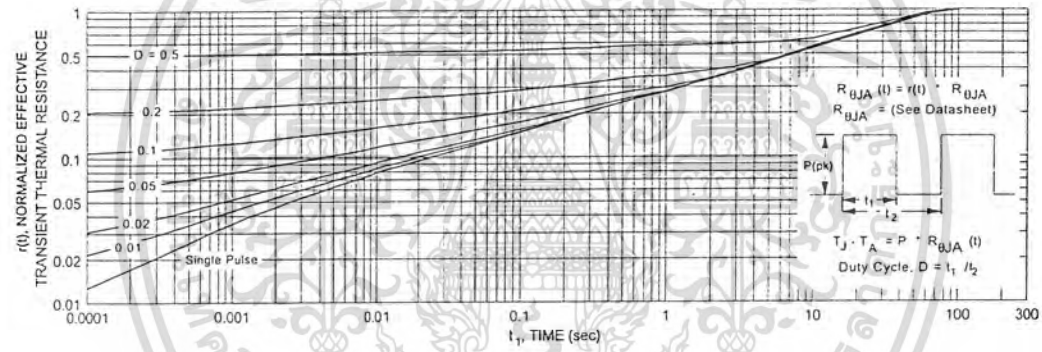


Figure 15. TO-92, BS170 Transient Thermal Response Curve.

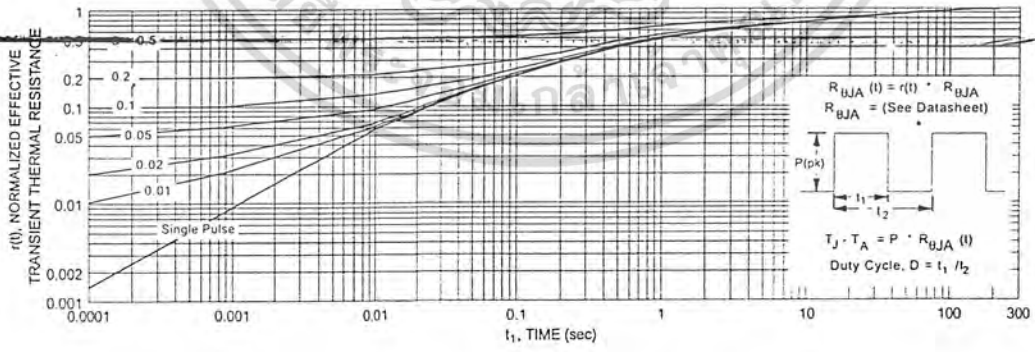


Figure 16. SOT-23, MMBF170 Transient Thermal Response Curve

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

FAIRCHILD
SEMICONDUCTOR™

August 1986
Revised March 2000

DM74LS04
Hex Inverting Gates

General Description

This device contains six independent gates each of which performs the logic INVERT function.

Ordering Code:

Order Number	Package Number	Package Description
DM74LS04M	M14A	14-Lead Small Outline Integrated Circuit (SOIC), JEDEC MS-120, 0.150 Narrow
DM74LS04SJ	M14D	14-Lead Small Outline Package (SOP), EIAJ TYPE II, 5.3mm Wide
DM74LS04N	N14A	14-Lead Plastic Dual-In-Line Package (PDIP), JEDEC MS-001, 0.300 Wide

Devices also available in Tape and Reel. Specify by appending the suffix letter "X" to the ordering code.

Connection Diagram

Function Table

$Y = \overline{A}$

Input	Output
A	Y
L	H
H	L

H = HIGH Logic Level
L = LOW Logic Level

© 2000 Fairchild Semiconductor Corporation

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

SINGLE COMPARATOR

VOLTAGE COMPARATOR

The LM311 series is a monolithic, low input current voltage comparator.
The device is also designed to operate from dual or single supplies voltage

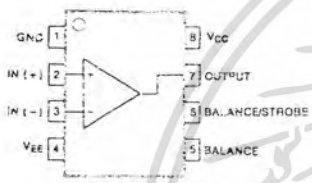
8 DIP

8 SOP

FEATURE

- Low input bias current : 250nA (Max)
- Low input offset current : 50nA (Max)
- Differential Input Voltage : ±30V.
- Power supply voltage : single 5.0V supply to ±15V.
- Offset voltage null capability.
- Strobe capability.

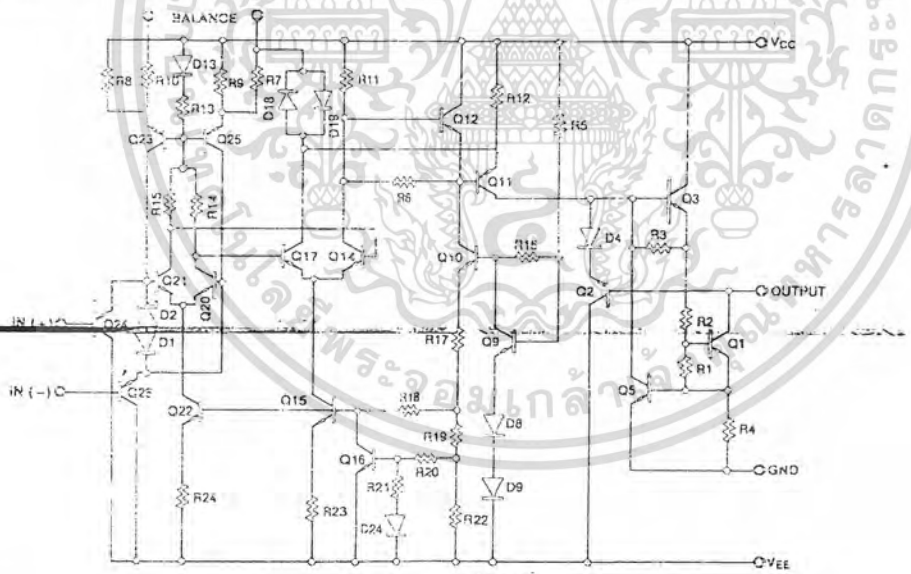
BLOCK DIAGRAM



ORDERING INFORMATION

Device	Package	Operating Temperature
LM311N	8 DIP	0 ~ +70°C
LM311M	8 SOP	

SCHEMATIC DIAGRAM



Rev. B

FAIRCHILD
SEMICONDUCTOR IM

©1999 Fairchild Semiconductor Corporation

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บรรณานุกรม

รีโมตเครื่องควบคุมไร้สาย, แผนกหนังสือพิเศษด้านอิเล็กทรอนิกส์ บริษัทซีเอ็ดยูเคชั่น รวบรวมจากบทความและโครงงานที่เคยตีพิมพ์ในวารสาร เซมิคอนดักเตอร์อิเล็กทรอนิกส์

การส่งข้อมูลด้วยอินฟาเรด, กฤต ศิริกุล 2541

ไมโครโปรเซสเซอร์, สถาบันเทคโนโลยีพระจอมเกล้าพระนครเหนือ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้