

สำนักหอสมุดกลาง พระจอมเกล้าลาดกระบัง

การพัฒนาเครื่องมือวัดโดยใช้เครื่องพ็อกเก็ต พีซี

DEVELOPMENT OF A PORTABLE MEASUREMENT DEVICE USING POCKET PC



นายกอบเกียรติ หุตะโกวิท
นายศักดิ์ยศ สิทธิกรกุล
นายสุทรินันท์ ไม้คู่แผ่นดิน

เลขหมู่.....
เลขทะเบียน **61982**
วัน,เดือน,ปี **2 5 ก.ค. 2549**

บ.....
ว.....

ปฏิญญาฉบับนี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต
ภาควิชาวิศวกรรมเครื่องกล
คณะวิศวกรรมศาสตร์
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
ปีการศึกษา 2547

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การพัฒนาเครื่องมือวัดโดยใช้เครื่องพ็อกเก็ต พีซี

DEVELOPMENT OF A PORTABLE MEASUREMENT DEVICE USING POCKET PC



โดย

นายกอบเกียรติ หุตะโกวิท
นายศักดิ์ยศ สิทธิกรกุล
นายสุทรินทร์ ไม้คู่แผ่นดิน

อาจารย์ที่ปรึกษา
อ.ชนาธิป ชัยฉลภพัฒน์กุล

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต

ภาควิชาวิศวกรรมเครื่องกล

คณะวิศวกรรมศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2547

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ปริญญาานิพนธ์ปีการศึกษา 2547

ภาควิชา วิศวกรรมเครื่องกล

คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เรื่อง การพัฒนาเครื่องมือวัดโดยใช้เครื่องพ็อคเก็ต พีซี

Development of a Portable Measurement device using Pocket PC

ผู้จัดทำ

1. นาย กอบเกียรติ หุตะโกวิท รหัสประจำตัว 44010636
2. นาย สักคยศ สิทธิกรกุล รหัสประจำตัว 44010859
3. นาย สุทธินันท์ ไม้คู่แผ่นดิน รหัสประจำตัว 44010869



อาจารย์ที่ปรึกษา

(อ.ชนาริพ ชัยติลลพัฒน์กุล)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การพัฒนาเครื่องมือวัดโดยใช้เครื่องพ็อคเก็ต พีซี

นาย กอบเกียรติ	หุตะโกวิท	44010636
นาย สักคยศ	สิทธิกรกุล	44010859
นาย สุทินันท์	ไม้คู่แผ่นดิน	44010869

อ.ชนาธิป ชัยดิลกพัฒนกุล อาจารย์ที่ปรึกษา
ปีการศึกษา 2547

บทคัดย่อ

การสั้นสะเทือนเป็นสัญญาณอย่างหนึ่งของความเสียหายของเครื่องจักร ดังนั้นถ้าเราทำการวิเคราะห์การสั้นสะเทือนก็จะทำให้ทราบถึงสาเหตุและแก้ไขได้ทันก่อนที่เครื่องจักรจะเกิดความเสียหาย แต่เครื่องวิเคราะห์ความสั้นสะเทือนที่มีอยู่ในปัจจุบันนั้นมีราคาค่อนข้างสูง ดังนั้นปริญญาานิพนธ์นี้จึงเป็นการศึกษาและพัฒนาเครื่องวิเคราะห์การสั้นสะเทือนให้สามารถทำการประมวลผลสัญญาณแล้วนำไปแสดงผลบนหน้าจอพ็อคเก็ต พีซี ซึ่งมีราคาถูกกว่าที่มีอยู่ในปัจจุบัน โดยมีหลักการทํางาน คือ รับสัญญาณอินพุตจากเครื่องกำเนิดสัญญาณ หรือที่เรียกว่า ฟังก์ชันเจนเนอเรเตอร์ ซึ่งเครื่องดังกล่าวจะทำการผลิตสัญญาณที่มีความต่อเนื่องในรูปแบบของสัญญาณแอมพลิจูดเชิงเวลา โดยสัญญาณอินพุตจะถูกส่งเข้าในส่วนวงจรประมวลผล เพื่อทำการแปลงสัญญาณที่มีความต่อเนื่องหรือสัญญาณอนาลอกให้กลายเป็นสัญญาณดิจิทัล โดยอาศัยทฤษฎีการสุ่มข้อมูล จากสัญญาณที่ถูกแปลงเป็นสัญญาณดิจิทัลแล้วจะถูกส่งเข้าเครื่องพ็อคเก็ต พีซี ผ่านทางพอร์ตอนุกรม (อาร์เอส-232) เพื่อทำการแปลงสัญญาณที่อยู่ในรูปของแอมพลิจูดเชิงเวลาให้อยู่ในรูปแอมพลิจูดเชิงความถี่ โดยใช้วิธีคำนวณฟูเรียร์อย่างรวดเร็ว จากนั้นผลที่คำนวณได้จะถูกนำมาแสดงบนหน้าจอของเครื่องพ็อคเก็ต พีซี ในรูปแบบของกราฟสเปกตรัม พร้อมทั้งบอกค่าความถี่สูงสุด โดยโครงการนี้ได้ศึกษาและพัฒนาอุปกรณ์ที่ใช้ในการวิเคราะห์การสั้นสะเทือนเพียงอย่างเดียวเท่านั้น โดยคาดหวังว่าโครงการนี้จะเป็พื้นฐานเพื่อนำไปพัฒนาเครื่องมือวัดต่างๆ ได้อย่างมีประสิทธิภาพและมีความหลากหลายเชิงวิศวกรรม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Development of a Portable Measurement device using Pocket PC

Korbkiet Hutakovida

Sakkayod Sittsikornkul

Suthinun Maikoopandin

Chanatip Chaidilokpatanakul Advisor

Abstract

Vibration is the cause of damage to machines. Vibration analysis can be used to help maintain machines before damages occur. But vibration analyzer is costly. So this project is to study and develop a low-priced vibration analyzer that can process signals and display them on a Pocket PC. The Principle is to obtain input signal from a function generator that produces time domain signals and then convert them from analog to digital. Applying sampling theorem to send the signal a Pocket PC via serial port (RS-232), the signal from time domain is transferred to frequency domain using Fast Fourier Transform (FFT). A graph spectrum and the peak frequency are shown on the Pocket PC. This project focuses on studying and developing a vibration analyzer with hopes that it will provide a basic guideline for the development of other measurement devices.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กิตติกรรมประกาศ

ปริญญาานิพนธ์ฉบับนี้สามารถสำเร็จลุล่วงไปได้ด้วยดี เนื่องด้วยได้รับความอนุเคราะห์และการช่วยเหลือในทุกๆด้านจากอาจารย์ที่ปรึกษา จึงขอขอบพระคุณ อ.ชนาธิป ชัยดิลกพัฒนกุล รวมทั้งอาจารย์ภาควิชาวิศวกรรมเครื่องกลทุกท่านที่ประสิทธิ์ประสาทวิชาให้ความรู้และคำแนะนำอันเป็นประโยชน์ต่อการทำปริญญาานิพนธ์ฉบับนี้

ข้าพเจ้าขอกราบขอบพระคุณ คุณพ่อ คุณแม่ที่สนับสนุนและเป็นกำลังใจให้ข้าพเจ้าเสมอมา สิ่งศักดิ์สิทธิ์ที่ข้าพเจ้าเคารพนับถือที่คอยปกป้องดูแลรักษาข้าพเจ้า

ขอขอบคุณคณะอาจารย์ทุกท่าน รุ่นพี่ เพื่อนๆและรุ่นน้องทุกคนที่ให้ความช่วยเหลือและข้อเสนอแนะในการทำงาน ข้อเสนอแนะเหล่านี้ทำให้งานวิจัยมีความสมบูรณ์มากขึ้น

คุณค่าและประโยชน์อันพึงมีจากปริญญาานิพนธ์ฉบับนี้ ผู้วิจัยขอมอบให้ผู้มีพระคุณทุกท่าน

นายกอบเกียรติ หุตะโกวิท

นายสัตกยศ สิริกรกุล

นายสุทธินันท์ ไม้คู่แผ่นดิน



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ

	หน้าที่
บทคัดย่อภาษาไทย	I
บทคัดย่อภาษาอังกฤษ	II
กิตติกรรมประกาศ	III
สารบัญ	IV
สารบัญตาราง	VII
สารบัญรูปภาพ	VIII
บทที่ 1 บทนำ	1
1.1 ความเป็นมาของโครงการ	1
1.2 วัตถุประสงค์ของโครงการ	1
1.3 ขอบเขตของการศึกษา	2
1.4 วิธีการดำเนินงาน	2
บทที่ 2 ทฤษฎีและหลักการที่เกี่ยวข้อง	3
2.1 การสื่อสารข้อมูลแบบดิจิทัล	3
2.1.1 รูปแบบการสื่อสารข้อมูลแบบอนุกรม	3
2.1.2 การส่งข้อมูลแบบซิงโครนัส (Synchronous Transmission)	5
2.1.3 การส่งข้อมูลแบบอะซิงโครนัส (Asynchronous Transmission)	6
2.1.4 อัตราความเร็วในการรับและส่งข้อมูลของการรับส่งข้อมูลแบบอะซิงโครนัสหรือบอดเรต	7
2.2 การอินเตอร์เฟสโดยพอร์ต RS-232	7
2.2.1 มาตรฐาน RS-232	8
2.2.2 การอินเตอร์เฟสตามมาตรฐาน RS-232	8
2.3 หลักการทำงานของ Analog to Digital Converter	10
2.3.1 หลักการเบื้องต้นของการแปลงสัญญาณอนาล็อกเป็นสัญญาณดิจิทัล	10
2.3.2 ค่าความละเอียดของ ADC	10
2.3.3 ชนิดของ เอ/ดี คอนเวอร์เตอร์	11
2.3.4 การแปลงสัญญาณอนาล็อกเป็นดิจิทัล	13
2.3.5 การจัดระดับสัญญาณ (Quantizing)	15
2.3.6 การเข้ารหัส (Coding)	16
2.4 Microcontroller MCS-51	17
2.4.1 คุณสมบัติของ MCS-51	17

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.4.2	โครงสร้างภายนอกของMCS-51	17
2.4.3	โครงสร้างภายในของ MCS-51	20
2.4.3.1	หน่วยความจำสำหรับเก็บ โปรแกรม	21
2.4.3.2	หน่วยความจำสำหรับเก็บข้อมูล	21
2.4.3.3	รีจิสเตอร์ใช้งานเฉพาะ	23
2.4.3.4	รีจิสเตอร์สำหรับใช้งานทั่วไป	23
2.4.4	โครงสร้างพอร์ต	23
2.4.5	ไทม์เมอร์และเคาน์เตอร์	24
2.4.6	พอร์ตสื่อสารแบบอนุกรม	24
2.4.7	โครงสร้างการอินเตอร์รัปต์	24
2.4.8	รีจิสเตอร์ใช้งานเฉพาะ IE (Interupt Enable - Register)	25
2.4.9	รีจิสเตอร์ใช้งานเฉพาะ IP (Interupt Priority Register)	26
2.4.10	กลุ่มคำสั่งใน MCS-51	27
2.4.10.1	วิธีการเข้าถึงข้อมูลโดยตรง(Direct addressing)	27
2.4.10.2	วิธีการเข้าถึงข้อมูลโดยอ้อม(Indirect addressing)	27
2.4.10.3	วิธีการเข้าถึงข้อมูลในรีจิสเตอร์ใช้งานทั่วไป(Register instruction)	28
2.4.10.4	วิธีการเข้าถึงข้อมูลในรีจิสเตอร์เฉพาะของตัวคำสั่ง(Register-specific instruction)	28
2.4.10.5	วิธีการเข้าถึงข้อมูลที่กำหนดเองโดยตรง(Immediate constant)	28
2.4.10.6	วิธีการเข้าถึงข้อมูลที่มีตัวชี้อ้างอิง(Indexed addressing)	28
2.5	ทฤษฎีสเปกตรัมแถบความถี่	29
2.5.1	ฟังก์ชันเป็นคาบ (Periodic Function)	29
2.5.2	อนุกรมฟูเรียร์ตรีโกณมิติ (Trigonometric Fourier Series)	30
2.5.3	อนุกรมฟูเรียร์เชิงซ้อน (Complex Form of Fourier Series)	31
2.5.4	สเปกตรัมความถี่เชิงซ้อน (Complex Frequency Spectrum)	33
2.5.5	สเปกตรัมเชิงกำลัง (Power Spectrum) ของฟังก์ชันคาบ	35
2.6	ทฤษฎีการสุ่มตัวอย่าง (Sampling)	38
2.7	ทฤษฎีฟาสต์ฟูเรียร์ทรานฟอร์ม (Fast Fourier Transform)	39
2.7.1	ทฤษฎีดิสครีทฟูเรียร์ทรานฟอร์ม (The Discrete Fourier Transform)	39
2.7.2	ทฤษฎีฟาสต์ฟูเรียร์ทรานฟอร์ม (FFT) และวิธีการแบบบัตเตอร์ฟลาย (Butterfly)	40
2.7.3	การคำนวณโดยใช้คอมพิวเตอร์	41
บทที่ 3	การออกแบบและโครงสร้าง	44
3.1	โครงสร้างโดยรวมของโครงการงาน	44
3.2	วงจร (Circuit)	44
3.2.1	วงจรตามแรงดัน (Voltage Follower)	44

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.2.2	วงจรการแบ่งแรงดันที่ไม่มีโหลด	45
3.2.3	วงจรแหล่งจ่ายแรงดัน	47
3.2.4	ไมโครคอนโทรลเลอร์	48
3.2.5	การเชื่อมต่อกับพอร์ตอนุกรมของคอมพิวเตอร์	50
3.3	Pocket PC	53
3.3.1	ข้อเปรียบเทียบระหว่าง Palm และ Pocket PC	53
3.3.2	สาเหตุที่เลือกใช้ Pocket PC	54
3.4	ขั้นตอนการทำงาน	55
บทที่	4 การทดลองและผลการทดลอง	57
บทที่	5 สรุปผลการทดลอง	62
	5.1 สรุปผลการทดลอง	62
	5.2 ปัญหา	62
	5.3 ข้อเสนอแนะ	63
บทที่	6 Application การใช้งาน	64
	6.1 การสันสะเทือน	64
	6.2 เครื่องวิเคราะห์การสันสะเทือน (Vibration Analyzer)	64
	6.3 การวิเคราะห์แบบสเปกตรัม	65
ภาคผนวก ก	วงจร	67
ภาคผนวก ข	โปรแกรมบนวงจร และ Pocket PC	119
บรรณานุกรม		148

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญตาราง

	หน้าที่
ตารางที่ 2.1 รายละเอียดคอนเน็กเตอร์แบบ DB9 ตามมาตรฐาน RS – 232	9
ตารางที่ 2.2 แสดงการเข้ารหัสสัญญาณ	16
ตารางที่ 2.3 แสดงการเปรียบเทียบจำนวนการคำนวณระหว่างดีสคริปทิวรีเวิร์ทรานฟอร์มและ ฟาสต์ฟูรีเวิร์ทรานฟอร์ม	42
ตารางที่ 4.1 ตารางผลการทดลองการวัด sampling rate ของวงจร	57
ตารางที่ 4.2 ตารางผลการทดลองการทดสอบค่าเปอร์เซ็นต์ error ของค่าความถี่ที่ Pocket PC วิเคราะห์ได้	61
ตารางที่ 5.1 ข้อแตกต่างระหว่างเครื่องวิเคราะห์การสั่นสะเทือนของโรงงานกับที่มีอยู่ทั่วไป	60



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูปภาพ

	หน้าที่
รูปที่ 2.1 แสดงการส่งข้อมูลแบบขนาน	4
รูปที่ 2.2 แสดงการส่งข้อมูลแบบอนุกรม	4
รูปที่ 2.3 แสดงรูปแบบการสื่อสารข้อมูลแบบข้อมูลอนุกรม	4
รูปที่ 2.4 แสดงการส่งข้อมูลแบบซิงโครนัส	5
รูปที่ 2.5 แสดงตัวอย่างของการหาอักขระซิงโครนัส	6
รูปที่ 2.6 แสดงรูปแบบของข้อมูลอนุกรมแบบอะซิงโครนัส	6
รูปที่ 2.7 แสดงตัวอย่างพอร์ตอนุกรม RS-232 แบบ DB9	8
รูปที่ 2.8 แสดงกราฟคุณสมบัติของเอ็ดจ์ขนาด 3 บิต	10
รูปที่ 2.9 แสดงบล็อกของวงจร เอ็ดจ์ ขึ้นบันไดหรือตามรอย	11
รูปที่ 2.10 แสดงรูปคลื่นวงจร เอ็ดจ์ขึ้นบันได	12
รูปที่ 2.11 แสดงรูปคลื่นของวงจรเอ็ดจ์ตามรอย	12
รูปที่ 2.12 แสดงบล็อกของวงจร เอ็ดจ์ที่ใช้การประมาณค่าโดยลำดับ	13
รูปที่ 2.13 แสดงรูปคลื่น	13
รูปที่ 2.14 การสุ่มสัญญาณ	14
รูปที่ 2.15 การเกิดความถี่ Aliasing	15
รูปที่ 2.16 แสดงตำแหน่งขาของชิป Microcontroller ตระกูล MCS-51 เบอร์ 8051	18
รูปที่ 2.17 โครงสร้างหน่วยความจำภายใน MCS-51	20
รูปที่ 2.18 หน่วยความจำสำหรับเก็บ โปรแกรม	21
รูปที่ 2.19 แผนภาพแสดงหน่วยความจำสำหรับเก็บข้อมูลภายในชิป MCS-51	22
รูปที่ 2.20 แสดงหน่วยความจำสำหรับเก็บข้อมูลทั่วไปภายในชิปทั้งสองส่วน	22
รูปที่ 2.21 แสดงตำแหน่งหน่วยความจำของ โปรแกรมบริการอินเตอร์รัปต์แต่ละชนิดใน MCS-51	25
รูปที่ 2.22 รีจิสเตอร์ใช้งานเฉพาะ IE	25
รูปที่ 2.23 รีจิสเตอร์ใช้งานเฉพาะ IP	26
รูปที่ 2.24 แสดงฟังก์ชันเป็นคาบ	29
รูปที่ 2.25 สัญญาณ (ฟังก์ชัน) ไซน์และโคไซน์ที่มีคาบครบรอบเท่ากับ	29
รูปที่ 2.26 สัญญาณไซน์ที่ฮาร์โมนิก $n = 1, 2$ และ 3	30
รูปที่ 2.27 แสดงการแทนฟังก์ชันรูปสี่เหลี่ยมด้วยฟังก์ชัน ไซน์และโคไซน์	31
รูปที่ 2.28 สัญญาณเรคติฟาย์ ไซน์เต็มคลื่น	34
รูปที่ 2.29 ไนส์สเปคตรัมของสัญญาณเรคติฟาย์ ไซน์เต็มคลื่น	34
รูปที่ 2.30 สัญญาณพัลส์รูปสี่เหลี่ยมผืนผ้า	34

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 2.31	สเปกตรัมเชิงแอมพลิจูด	34
รูปที่ 2.32	สเปกตรัมเชิงแอมปริจูด	35
รูปที่ 2.33	สัญญาณเรกติฟาย์ไซน์เต็มคลื่น	37
รูปที่ 2.34	สเปกตรัมเชิงกำลังของสัญญาณเรกติฟาย์ไซน์เต็มคลื่น	37
รูปที่ 2.35	สัญญาณพัลส์รูปสี่เหลี่ยมผืนผ้า	37
รูปที่ 2.36	สเปกตรัมเชิงกำลังของสัญญาณพัลส์รูปสี่เหลี่ยมผืนผ้า	37
รูปที่ 2.37	การประมวลผลเชิงอนุมาณ โดยใช้ระบบการประมวลผลของสัญญาณเชิงเลข	38
รูปที่ 2.38	แสดงตัวอย่างของความถี่ในการสุ่มสัญญาณ $T_s = 1/(4f_0)$	38
รูปที่ 2.39	ดิสกรีทฟูเรียร์ทรานฟอร์ม	39
รูปที่ 2.40	การหาค่า 8 จุดฟาสต์ฟูเรียร์ทรานฟอร์ม	41
รูปที่ 2.41	การหาค่า 8 จุดฟาสต์ฟูเรียร์ทรานฟอร์ม	41
รูปที่ 2.42	หลักการคำนวณโดยใช้บิตเตอร์ฟลายเดียว	42
รูปที่ 2.43	อินพุท 16 จุด ฟาสต์ฟูเรียร์ทรานฟอร์ม	42
รูปที่ 2.44	เอาต์พุทของ 16 จุด ฟาสต์ฟูเรียร์ทรานฟอร์ม	43
รูปที่ 3.1	แสดงโครงสร้างโดยรวมของโครงการ	44
รูปที่ 3.2	วงจรตามแรงดัน	44
รูปที่ 3.3	แสดงรูป IC เบอร์ LM324N	45
รูปที่ 3.4	วงจรการแบ่งแรงดันที่ไม่มีโหลด	45
รูปที่ 3.5	การเชื่อมต่อวงจร Unloaded Voltage Divider กับวงจร Voltage Follower เพื่อส่งสัญญาณเข้าไมโครคอนโทรลเลอร์	46
รูปที่ 3.6	ลักษณะกราฟของ v-t ที่ออกจาก Function Generator	46
รูปที่ 3.7	ลักษณะกราฟของ v-t เมื่อผ่าน Voltage Follower	47
รูปที่ 3.8	ลักษณะกราฟของ v-t เมื่อผ่าน แหล่งจ่ายแรงดันกระแสตรง +1.5	47
รูปที่ 3.9	วงจรแหล่งจ่ายแรงดัน	47
รูปที่ 3.10	แสดงขาของไมโครคอนโทรลเลอร์	48
รูปที่ 3.11	แสดงวงจรของไมโครคอนโทรลเลอร์	49
รูปที่ 3.12	วงจรเชื่อมต่อ MAX232 เข้ากับพอร์ตอนุกรมของคอมพิวเตอร์ และ MCS-51	51
รูปที่ 3.13	แสดงลายวงจรที่ออกแบบด้วยโปรแกรม Portel	51
รูปที่ 3.14	แสดงส่วนประกอบของวงจรทั้งหมด	52
รูปที่ 3.15	Pocket PC รุ่น iPAQ 1710	54
รูปที่ 3.16	แสดงการรับสัญญาณอินพุต	55
รูปที่ 3.17	แสดงการหยุดรับสัญญาณอินพุต	55
รูปที่ 3.18	แสดงกราฟ Time Domain	55
รูปที่ 3.19	แสดงกราฟ Spectrum	56

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 4.1 แสดงระยะเวลาที่ใช้ต่อสัญญาณ 1 คาบ เท่ากับ 208 μ s	58
รูปที่ 4.2 แสดงระยะเวลาที่ใช้ต่อสัญญาณ 1 คาบ เท่ากับ 210 μ s	58
รูปที่ 6.1 Vibration Analyzer	64
รูปที่ 6.2 สัญญาณที่อยู่ในรูปโดเมนของเวลา	65
รูปที่ 6.3 สัญญาณที่อยู่ในรูปโดเมนของความถี่	65
รูปที่ 6.4 แสดงแถบความถี่ของเครื่องจักรที่มีความเสียหายของชิ้นงานที่เชิงศูนย์	65
รูปที่ 6.5 แสดงแถบความถี่ของเครื่องจักรที่มีความเสียหายของชิ้นงานที่ไม่สมดุล	65



บทที่ 1

บทนำ

1.1 ความเป็นมาของโครงการ

ในปัจจุบันเทคโนโลยีได้มีความเจริญก้าวหน้าไปอย่างมาก โดยได้มีการผลิตอุปกรณ์อิเล็กทรอนิกส์ที่มีขนาดเล็กแต่เปี่ยมไปด้วยประสิทธิภาพในการทำงาน อย่างเช่น คอมพิวเตอร์พกพา หรือ พ็อกเก็ตพีซี (Pocket PC) เป็นต้น ซึ่งคอมพิวเตอร์พกพานี้ในปัจจุบันได้มีการใช้งานอย่างแพร่หลาย และมีแนวโน้มที่ผู้คนจะหันมาสนใจใช้งานเพิ่มมากขึ้น เนื่องจากมีขนาดเล็กกะทัดรัดจึงสามารถพกพาไปในสถานที่ต่างๆ ได้สะดวก และมีราคาไม่สูงมากนัก อีกทั้งยังสามารถใช้งานได้หลากหลาย ใช้งานง่าย ไม่ซับซ้อน และยังสามารพัฒนาเพิ่มประสิทธิภาพของการทำงานให้กับคอมพิวเตอร์ขนาดพกพานี้ได้โดยการสร้างรูปแบบการทำงานใหม่ๆ คือนำมาประยุกต์ใช้เป็นเครื่องมือวัดค่าต่างๆ ทั้งนี้ทั้งนั้นขึ้นอยู่กับลักษณะการใช้งาน ตัวอย่างเช่น นำมาประยุกต์ใช้งานเป็นอุปกรณ์วัดค่าต่างๆทางวิศวกรรม เป็นต้น

ในปัจจุบันพบปัญหาที่เกิดจากการสั่นสะเทือนเพิ่มมากขึ้น เนื่องจากความซับซ้อนของโครงสร้างต่างๆ รวมทั้งผลิตภัณฑ์เครื่องจักรกลต่างๆ ถูกพัฒนาให้ทำงานเร็วขึ้นซับซ้อนขึ้น จึงมีโอกาสเกิดการสั่นมากขึ้น ทำให้เกิดเสียงดังมากขึ้นด้วย หากเครื่องจักรดังกล่าวถูกผลิตขึ้นด้วยคุณภาพที่ไม่สูงพอ โดยเฉพาะเครื่องจักรที่ใช้งานมานานก็จะมีโอกาสที่จะเกิดการสั่นสะเทือนมากยิ่งขึ้น

การสั่นสะเทือนนั้นมีผลกระทบต่อมนุษย์ทั้งกายภาพ จิตใจรวมถึงสังคมด้วย เสียงรบกวนมีผลในการทำลายการได้ยินของมนุษย์ รบกวนแทรกแซงการสื่อสาร ทำให้รู้สึกรำคาญ ก่อให้เกิดความเหนื่อยล้า และลดประสิทธิภาพในการทำงาน เป็นต้น นอกจากนี้การสั่นสะเทือนของเครื่องจักรยังถือได้ว่าเป็นการแสดงออกถึงอาการผิดปกติของเครื่องจักรอีกอย่างหนึ่ง ซึ่งถ้ามีการตรวจวัดถึงลักษณะการสั่นสะเทือนแล้วนำไปวิเคราะห์ ก็สามารถระบุถึงสาเหตุที่เกิดขึ้นได้ ทำให้สามารถแก้ไขได้ตรงตำแหน่งก่อนที่จะเกิดความเสียหายที่อาจจะรุนแรงจนถึงกับต้องหยุดการผลิต และเป็นการป้องกันการบาดเจ็บของบุคลากรที่ควบคุมเครื่องจักรและบุคลากรในบริเวณใกล้เคียง อีกทั้งยังสามารถนำไปเป็นข้อมูลสำหรับวางแผนการบำรุงรักษาได้

เครื่องวิเคราะห์การสั่นสะเทือน (Vibration Analyzer) เป็นอุปกรณ์ที่สามารถวิเคราะห์การสั่นสะเทือนได้อย่างมีประสิทธิภาพ โดยจะวัดลักษณะการสั่นสะเทือนแล้วนำไปประมวลผล จากนั้นจึงนำผลที่ประมวลได้ไปวิเคราะห์หาสาเหตุของการสั่นสะเทือนที่เกิดขึ้นได้ ด้วยเหตุนี้เครื่องวิเคราะห์การสั่นสะเทือน จึงมีประโยชน์กับงานอุตสาหกรรมในปัจจุบันซึ่งมีการแข่งขันกันสูง แต่ทว่าเครื่องวิเคราะห์การสั่นสะเทือน ที่มีอยู่ในปัจจุบันนั้นมีราคาค่อนข้างสูง ดังนั้นจึงเป็นที่มาของโครงการนี้

1.2 วัตถุประสงค์ของโครงการ

1.2.1 เพื่อศึกษาและออกแบบระบบระบบเครื่องมือวัด การประมวลผลพร้อมทั้งการอินเตอร์เฟซกับอุปกรณ์แสดงผลได้

1.2.2 เพื่อพัฒนาระบบเครื่องมือวัดให้สามารถแสดงผล และวิเคราะห์ผ่าน Pocket PC ได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1.3 ขอบเขตของการศึกษา

โครงการนี้เป็นกรออกแบบและพัฒนาอุปกรณ์วัดความสั่นสะเทือนที่สามารถแสดงผลได้บน Pocket PC โดยเราแบ่งส่วนศึกษาออกเป็น 2 ส่วน ดังนี้

1. ส่วนของ Hardware แบ่งเป็น

- ศึกษาการแปลงสัญญาณอนาล็อก (Analog) เป็นสัญญาณดิจิทัล (Digital)
- ศึกษาการส่งข้อมูลของพอร์ตอนุกรม อาร์เอส-232 (Serial Port RS-232)
- ศึกษาการทำงานของไมโครคอนโทรลเลอร์ MCS-51

2. ส่วนของ Software

- ศึกษาโปรแกรมแอสเซมบลี (Assembly) เพื่อควบคุมการทำงานของไมโครคอนโทรลเลอร์
- ศึกษาทฤษฎี และวิธีการแปลงฟูเรียร์อย่างรวดเร็ว (Fast Furier Transform)
- ศึกษาโปรแกรม Visual C# เพื่อสร้าง Software ที่สามารถประมวลผลและแสดงผลบนพีซี

เกิด พีซีได้

1.4 วิธีการดำเนินงาน

งานวิจัยในโครงการนี้จะเริ่มด้วยการศึกษาทฤษฎีพื้นฐานต่างๆที่เกี่ยวข้องกับงานวิจัย ซึ่งมีเรื่องหลักๆ อยู่ 4 เรื่องด้วยกัน คือ ทฤษฎีการแปลงฟูเรียร์อย่างรวดเร็ว, ทฤษฎีพื้นฐานของไมโครคอนโทรลเลอร์, ทฤษฎี Analog to Digital Converter และทฤษฎีการส่งข้อมูลของ RS-232 ซึ่งมีรายละเอียดดังในบทที่ 2 จากนั้นก็เอาความรู้ที่ได้ศึกษาทั้งหมดมาออกแบบโครงสร้างของวงจรซึ่งมีรายละเอียดในบทที่ 3

ในบทที่ 4 เป็นการทดลองในส่วนต่างๆของโครงการ ซึ่งแบ่งออกเป็น 2 ส่วน คือ ส่วนของวงจร และ ส่วนของโปรแกรม ในบทที่ 5 เป็นการสรุปการทำงาน ผลที่ได้จากงานวิจัยนี้และแนวทางในการพัฒนางานวิจัยนี้เพิ่มเติมและบทที่ 6 ซึ่งเป็นบทสุดท้าย เป็นแนวทางในการนำเครื่องวิเคราะห์การสั่นสะเทือนไปประยุกต์ใช้

บทที่ 2

ทฤษฎีและหลักการที่เกี่ยวข้อง

2.1 การสื่อสารข้อมูลแบบดิจิทัล

การสื่อสารข้อมูลแบบดิจิทัล นั้นข้อมูลจากแหล่งกำเนิดจะอยู่ในลักษณะสัญญาณดิจิทัลและข้อมูลที่รับได้ก็จะอยู่ในลักษณะดิจิทัลเช่นเดียวกัน ข่าวสารจากแหล่งกำเนิดอาจจะเป็นตัวรหัสของตัวอักษร ตัวเลข เครื่องหมายที่อยู่ในรูปเลขฐานสอง หรือสัญญาณอนาล็อกที่ทำการเปลี่ยนเป็นสัญญาณดิจิทัลแล้ว เป็นต้น

การถ่ายโอนข้อมูลจากที่หนึ่งไปยังอีกที่หนึ่งนั้นมีอยู่ 2 ประเภท

1. การถ่ายโอนข้อมูลแบบขนาน ลักษณะการส่งข้อมูลแบบขนาน ทำได้โดยการส่งข้อมูลออกมาทีละ 1 ไบต์จากอุปกรณ์ส่งไปยังอุปกรณ์รับ ตัวกลางระหว่าง อุปกรณ์รับ – ส่ง จะต้องมียช่องทางให้ข้อมูลเดินทางอย่างน้อย 8 ช่องทาง โดยมากจะเป็นสายขนานให้กระแสไฟฟ้าวิ่งมากกว่าจะเป็นตัวกลางชนิดอื่น ดังแสดงในรูปที่ 2.1 เนื่องจากมีการลดทอนของสัญญาณเนื่องจากความต้านทานของสาย ระยะทางระหว่าง อุปกรณ์รับ – ส่ง จึงไม่ควรเกิน 100 ฟุต ปัญหาที่เกิดขึ้นหากระยะทางของสายมากกว่านี้ คือ ระดับของกราวด์ในทางไฟฟ้าที่ถูกรับผิดไปจากจุดส่งทำให้เกิดการผิดพลาดในการรับสัญญาณทางด้านรับได้

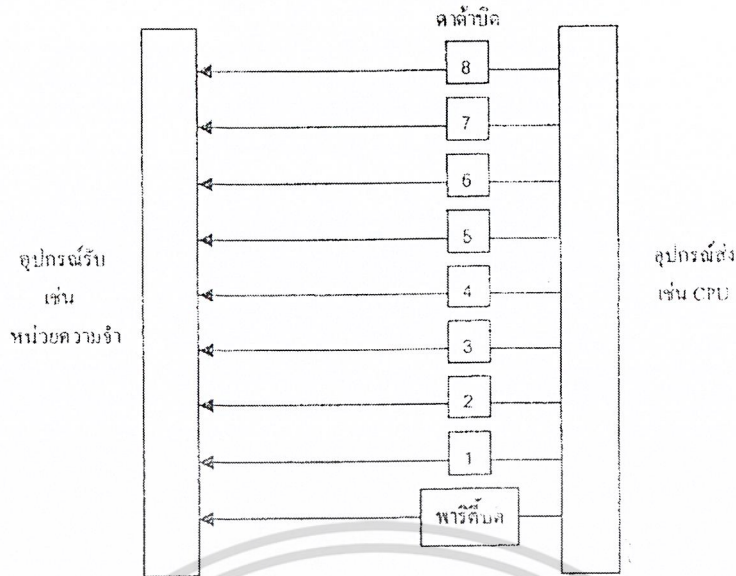
จะเห็นว่าการส่งแบบขนานส่วนมากจะทำในระยะทางใกล้ๆ เนื่องจากจะต้องมีช่องทางเดินของสัญญาณมากกว่า 8 สาย

2. การถ่ายโอนข้อมูลแบบอนุกรม ในถ่ายโอนข้อมูลแบบอนุกรม ข้อมูลจะถูกส่งออกมาทีละบิตระหว่างอุปกรณ์รับ – ส่ง จะเห็นว่าการส่งข้อมูลแบบนี้จู้จี้กว่าแบบขนานที่กล่าวมาแล้วอย่างแน่นอน แต่สาเหตุที่มีการนิยมใช้การส่งแบบนี้กันก็เพราะตัวกลางที่ใช้ในการสื่อสารแบบนี้ต้องการเพียงช่องสัญญาณเดียว ทำให้ค่าใช้จ่ายเรื่องของสื่อกลางถูกกว่าแบบขนาน สำหรับการส่งในระยะทางไกลๆ ดังแสดงในรูปที่ 2.2 จากรูปที่ 2.2 แสดงให้เห็นการส่งข้อมูลแบบอนุกรม ข้อมูลจากอุปกรณ์ส่งจะถูกเปลี่ยนให้เป็นอนุกรมเสียก่อนแล้วค่อยทยอยส่งออกไปยังอุปกรณ์รับ ที่อุปกรณ์รับจะต้องมีการเปลี่ยนข้อมูลที่ส่งมาทีละบิตให้เป็นสัญญาณแบบขนานซึ่งลงตัวพอดีนั่นคือ บิตที่ 1 ลงที่ ดาต้าบัส (data bus) เส้นที่ 1 พอดี การที่จะทำให้การแปลงสัญญาณจากอนุกรมทีละบิตให้ลงพอดีนั่นจำเป็นที่จะต้องมีการกลไกที่เหมาะสมเพื่อป้องกันการผิดพลาดในการรับ กลไกที่วันนี้แบ่งออกเป็น 2 แบบ คือ 1.การส่งข้อมูลแบบซิงโครนัส 2.การส่งข้อมูลแบบอะซิงโครนัส

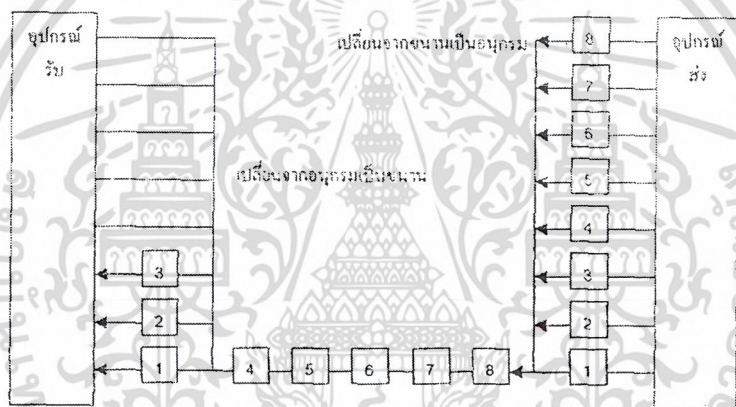
2.1.1 รูปแบบการสื่อสารข้อมูลแบบอนุกรม

การติดต่อสื่อสารแบบอนุกรมอาจแบ่งตามรูปได้ 3 แบบ

1. แบบซิมเพิล็กซ์ (Simplex) ข้อมูลที่ส่งได้ในทางเดียวเท่านั้น บางครั้งก็เรียกว่าการส่งทิศทางเดียว ในการสื่อสารแบบนี้อุปกรณ์สื่อสารด้านหนึ่ง จะส่งข้อมูลไปในช่องสัญญาณได้เท่านั้นแต่จะรับข้อมูลจากช่องสัญญาณไม่ได้ แต่อุปกรณ์สื่อสารอีกด้านหนึ่งจะรับข้อมูลจากช่องสัญญาณได้เท่านั้น และจะส่งข้อมูลไปในช่องสัญญาณไม่ได้ การใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



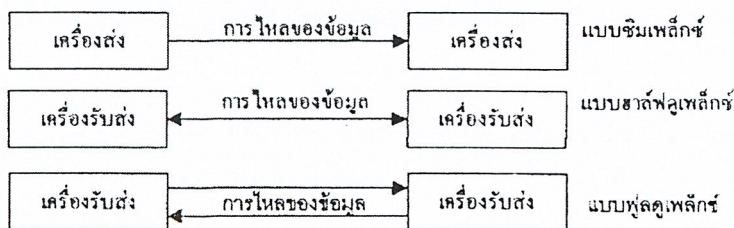
รูปที่ 2.1 แสดงการส่งข้อมูลแบบขนาน



รูปที่ 2.2 แสดงการส่งข้อมูลแบบอนุกรม

2. แบบฮาล์ฟดูเพล็กซ์ (Half Duplex) หมายถึงการสื่อสารข้อมูลใน 2 ทิศทางแต่ในช่วงเวลาหนึ่งได้เพียงทิศทางเดียวเท่านั้น อุปกรณ์สื่อสารทั้ง 2 ด้านจะผลัดกันรับ-ส่ง การสื่อสารแบบนี้ส่วนใหญ่แล้วจะใช้ระบบสาย 2 เส้น

3. แบบฟูลดูเพล็กซ์ (Full Duplex) หมายถึงการสื่อสารข้อมูล 2 ทิศทางพร้อมกัน การสื่อสารแบบนี้ใช้ได้ทั้งระบบสาย 2 เส้น และสาย 4 เส้น แต่ในระบบสาย 2 เส้น จะต้องอาศัยเทคนิคการแบ่งความถี่เข้าช่วย คือ จะส่งในความถี่ช่วงหนึ่งและจะรับในความถี่อีกช่วงหนึ่ง



รูปที่ 2.3 แสดงรูปแบบการสื่อสารข้อมูลแบบข้อมูลอนุกรม

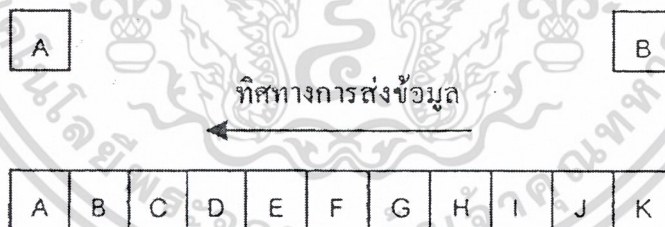
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.1.2 การส่งข้อมูลแบบซิงโครนัส (Synchronous Transmission)

การส่งข้อมูลแบบซิงโครนัส หมายถึง การที่ด้านรับอ่านข้อมูลเข้ามาในจังหวะเดียวกับด้านส่ง โดยใช้สัญญาณนาฬิกาเป็นตัวกำหนดจังหวะการทำงานของรีจิสเตอร์ทั้งสอง ให้ทำงานสัมพันธ์กันนอกจากนี้เมื่อจังหวะเวลาถูกตั้งให้ซิงโครนัสกับทางด้านรับได้แล้ว ข้อมูลจะถูกส่งไปบนทางติดต่อในแบบบิตต่อบิตต่อเนื่องกันไปอาศัยช่วงเวลาระหว่างบิตต่อบิตมีค่าเท่ากันโดยไม่ต้องมีบิตเริ่มส่งหรือบิตจบคอยกำกับ ทำให้ความเร็วในการส่งข้อมูลมีสูง ข้อเสียของการส่งแบบซิงโครนัส คือ การที่ต้องมีสัญญาณนาฬิกาขนานไปกับข้อมูล ทำให้ต้องการทางติดต่อช่องที่สองเพิ่มขึ้น โดยเฉพาะกรณีระยะทางไกล ๆ เป็นการยากมากที่จัดหาทางติดต่อแยกต่างหากสำหรับสัญญาณนาฬิกา นอกจากนี้ทางด้านรับต้องมีวงจรมอดูเลชันเฟสล็อก (PLL) เพิ่มทำหน้าที่รับข้อมูลจังหวะเวลาจากด้านส่ง และสร้างสัญญาณนาฬิกาขึ้นมาใหม่ทางด้านรับ เพื่อให้เกิดการซิงโครนัสขึ้น อาจกล่าวได้ว่าการส่งแบบซิงโครนัสมีค่าใช้จ่ายแพงกว่าแบบอะซิงโครนัส สิ่งที่ควรเพิ่มเติมในการส่งข้อมูลแบบซิงโครนัส (Synchronous Transmission) มีดังนี้

ข้อมูลในแบบซิงโครนัสจะถูกจัดการให้อยู่ในรูปของชุดข้อมูลที่มีลักษณะพิเศษ คือ ช่วงระยะเวลาระหว่างตัวอักษรด้วยกันจะไม่มี ทำให้การส่งข้อมูลเป็นไปอย่างต่อเนื่อง ซึ่งตัวอักษรจะแทนด้วยรหัสเลขฐานสอง เช่น รหัสแอสกี ดังรูปที่ 2.4

จากรูป ตัวอักษรจาก A -K ถูกส่งออกไปอย่างต่อเนื่องโดยที่ช่วงระหว่างตัวอักษรมีค่าเท่ากับศูนย์ทางด้านรับ (ด้านเอ) จะต้องรับทราบตำแหน่งแรกของตัวอักษรตัวแรกสุครวมทั้งขนาดของตัวอักษรและความเร็วในการส่งด้วย อย่างเช่นในกรณีตัวอักษรเป็นรหัสแอสกี พบว่าแต่ละตัวอักษรมีขนาด 8 บิต เมื่อถึงจุดนี้พบว่าปัญหาที่เกิดขึ้นคือ ทำอย่างไรจึงจะรู้ตำแหน่งบิตแรกของตัวอักษรตัวแรกได้ มีการแก้ปัญหาโดยการกำหนดตัวอักษรแบบพิเศษที่ใช้เฉพาะทำหน้าที่การซิงโครนัสเท่านั้นเรียกว่าอักขระการควบคุมซิงโครนัส



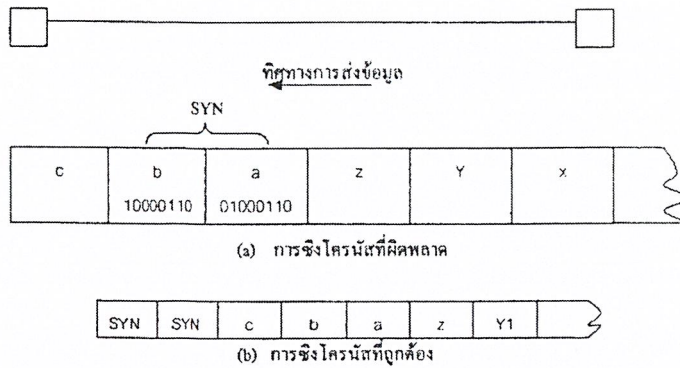
รูปที่ 2.4 แสดงการส่งข้อมูลแบบซิงโครนัส

อย่างเช่นทางด้านส่งรหัสแอสกีใช้ค่า TC 9 แทน SYN Character การที่มีอักขระ "SYN" เพื่อแจ้งให้ด้านรับทราบว่าข้อมูลหลังจากอักขระ SYN คือข่าวสารที่ต้องการติดต่อรับได้เลย

ส่วนทางด้านรับจะมีขั้นตอนการทำงานดังนี้

1. ด้านรับทำการตรวจสอบ อักขระ SYN ในสายข้อมูลให้พบก่อน โดยมีการกำหนดรูปแบบบิตของอักขระ SYN เป็น 00010110 (พาริตีคู่)
2. ตำแหน่งที่ด้านรับกำลังตรวจสอบ จะนำข้อมูลขณะนั้นไปเปรียบเทียบกับอักขระ SYN แล้วจะสามารถรับอักษรหลังจากนั้นไปได้อย่างต่อเนื่องอย่างไรก็ตามก็ยังมีปัญหาเกิดขึ้นตามมา ดังตัวอย่างในรูปที่ 2.5

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.5 แสดงตัวอย่างของการหาอักขระซิงโครไนส์

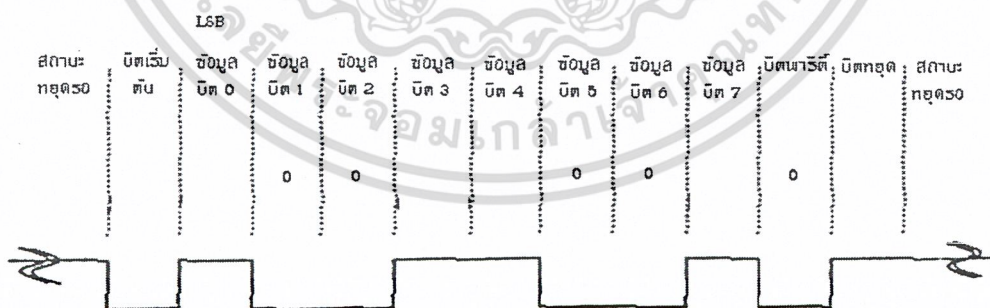
2.1.3 การส่งข้อมูลแบบอะซิงโครไนส์ (Asynchronous Transmission)

การส่งข้อมูลแบบอะซิงโครไนส์คือการรับและส่งข้อมูลโดยไม่จำเป็นต้องมีสัญญาณนาฬิกาพร้อมด้วย แต่จะใช้การกำหนดค่าอัตราเร็วในการรับและการส่งข้อมูลให้มีค่าเท่ากัน ซึ่งเรียกอัตรารเร็วนี้ว่า อัตราบอดเรต หรือ บอดเรต (Baud rate) มีหน่วยเป็น บิต ต่อ วินาที (bps)

Baud rate คือ อัตราการส่งข้อมูลแบบอนุกรม เทียบกับเวลา 1 วินาที เช่น ส่งข้อมูลแบบอนุกรม Baud rate 9600 หมายถึงการส่งข้อมูลทั้งหมด 9600 บิต ต่อ 1 วินาที ซึ่งในการส่งข้อมูลแบบอนุกรมจะส่ง Baud rate ได้ 110, 1200, 2400, 4800 และ 9600 เป็นต้น

รูปแบบของข้อมูลที่ใช้ในการรับส่งแบบอะซิงโครไนส์ประกอบด้วย 4 ส่วนด้วยกันคือ

1. บิตเริ่มต้น (start bit) มีขนาด 1 บิต
2. บิตข้อมูลแบบอนุกรมมีขนาด 8 บิต
3. บิตตรวจสอบพาริตี (parity bit) มีขนาด 1 บิตหรือไม่มี
4. บิตปิดท้ายหรือบิตหยุด (stop bit) มีขนาด 1 บิต



รูปที่ 2.6 แสดงรูปแบบของข้อมูลอนุกรมแบบอะซิงโครไนส์

เมื่อไม่มีการส่งข้อมูลหา DATA จะมีสถานะลอจิก "1" เรียกสถานะนี้ว่า สถานะหยุดรอ (wait state) การเริ่มต้นการส่งข้อมูลจะเริ่มจากการให้หา DATA มีลอจิก "0" ด้วยช่วงระยะเวลา 1 บิต เรียกบิตนี้ว่า บิตเริ่มต้น (start bit) จากนั้นข้อมูลจะถูกส่งออกไป โดยเริ่มจากบิตน้อยสำคัญต่ำสุดหรือ LSB ก่อนซึ่งข้อมูลที่ต้องการส่งมีจำนวน 8 บิต จากนั้นตามด้วย บิตพาริตี (parity bit) ซึ่งใช้ตรวจสอบความผิดพลาดที่

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เกิดอาจจะเกิดขึ้นจากการส่งข้อมูลบิตสุดท้ายที่จะส่งคือ บิตปิดท้ายหรือบิตหยุด (stop bit) โดยจะเป็นการทำให้ขา DATA มีลอจิก “1” อีกครั้งด้วยระยะเวลาอย่างน้อย 1 บิต 1.5 บิต หรือ 2 บิต เพื่อเป็นการแสดงว่าสิ้นสุดข้อมูลแล้ว

2.1.4 อัตราความเร็วในการรับและส่งข้อมูลของการรับส่งข้อมูลแบบอะซิงโครนัส หรือบอดเรต

สำหรับพอร์ตอนุกรม RS-232 มีด้วยกันหลายค่าได้แก่ 110 , 150 , 300 , 600 , 1200 , 2400 , 4800 , 9600 , 19200 จนถึง 115200 บิตต่อวินาที โดยมีค่าเพิ่มมากขึ้นตามเทคโนโลยีของคอมพิวเตอร์ เนื่องจากอัตราคือค่าของจำนวนบิตที่ส่งได้ภายใน 1 วินาที สมมติว่าข้อมูลอนุกรมมีขนาด 8 บิต ไม่มีการตรวจสอบพาริตี มีบิตเริ่มต้น 1 บิต และบิตปิดท้าย 1 บิต ความยาวของข้อมูล 1 ไบต์จะมีขนาดเท่ากับ 10 บิต ถ้าใช้บอดเรตในการส่งข้อมูลเท่ากับ 9600 บิตต่อวินาทีก็จะสามารถรับส่งข้อมูลได้ด้วยความเร็ว 960 ไบต์ต่อวินาที การตรวจสอบพาริตีสามารถกำหนดให้เป็นแบบคี่ (odd) แบบคู่ (even) หรือไม่มีการตรวจสอบพาริตีก็ได้ พาริตีคี่หรือคู่ แสดงถึงจำนวนลอจิก “1” ทั้งหมดภายในข้อมูลที่ส่งไป 1 ไบต์รวมบิตพาริตีว่ามีจำนวนเลขคี่หรือเลขคู่ ยกตัวอย่างข้อมูลที่จะทำการส่งมีขนาด 8 บิต มีค่าเท่ากับ 99h หรือ 10011001 จะเห็นว่าข้อมูลในไบต์นี้มีจำนวนลอจิก “1” จำนวน 4 ตัวซึ่งเป็นเลขคู่ ดังนั้นถ้ากำหนดค่าพาริตีเป็นคู่ ค่าของบิตพาริตีจะต้องมีลอจิกเป็น “0” แต่ถ้ากำหนดพาริตีเป็นคี่ ค่าของพาริตีจะต้องเป็น “1” เพื่อให้ข้อมูล 1 ไบต์ รวมทั้งบิตพาริตีเป็นคี่

บิตพาริตีถูกสร้างขึ้นจากภาคส่งข้อมูลของ UART (Universal Asynchronous Receiver Transmitter): เป็นอุปกรณ์ที่ใช้ในการรับส่งข้อมูลอนุกรม ซึ่งทางภาครับจะต้องกำหนดคุณสมบัติการตรวจสอบพาริตีที่ตรงกันเอาไว้ว่าจะตรวจสอบพาริตีคี่หรือพาริตีคู่ จากนั้นภาครับของ UART จะทำการตรวจสอบค่าพาริตีที่เกิดขึ้นว่าเป็นคู่หรือคี่ โดยการนับจำนวนลอจิก 1 ทั้งหมดรวมทั้งบิตพาริตีด้วย ถ้ากำหนดพาริตีไว้เป็นคู่แต่อ่านค่าตัวเลขในการนับค่าออกมาได้เป็นที่ทางภาครับจะแสดงผลผิดพลาดออกมาให้ผู้ใช้งานทราบ กระบวนการดังกล่าวเป็นวิธีการตรวจสอบได้เมื่อมีบิตข้อมูลที่ทำการรับส่งข้อมูลผิดพลาดเพียงบิตเดียวเท่านั้น ถ้าข้อมูลที่ส่งมีบิตผิดพลาดมากกว่าหนึ่งบิตการตรวจสอบด้วยวิธีนี้จะไม่เหมาะสำหรับการตั้งพาริตีบิตเป็น NONE ทั้งทางภาครับและส่งไม่มีการตรวจสอบพาริตี

2.2 การอินเตอร์เฟสโดยพอร์ต RS-232

ลักษณะของการส่งข้อมูลแบบอนุกรมนั้น ข้อมูลจะถูกส่งออกมาทีละบิต จากตัวอุปกรณ์ส่งไปยังอุปกรณ์รับ ช่องสัญญาณในการส่งข้อมูลอาจจะใช้เพียง 1 หรือ 2 ช่องสัญญาณเท่านั้น ทำให้ค่าใช้จ่ายในการสื่อสารถูกกว่าแบบขนาน แต่อัตราการส่งข้อมูลจะช้ากว่าการส่งแบบขนาน ในการส่งข้อมูลแบบอนุกรมข้อมูลที่ต้องการส่งจะอยู่ในลักษณะเป็นไบต์จะถูกทยอยส่งทีละบิตทางอุปกรณ์รับซึ่งจะต้องรับข้อมูลเข้ามาทีละบิต แล้วมารวมกันเป็นไบต์ซึ่งทางด้านอุปกรณ์รับจะต้องคอยตรวจสอบว่าบิตใดเป็นบิตเริ่มแรกของไบต์นั้น การตรวจสอบขึ้นอยู่กับรูปแบบรหัสของบิตที่ใช้ในการรับส่งข้อมูลแบบอนุกรม ระหว่างไมโครคอมพิวเตอร์กับอุปกรณ์ภายนอกนั้น จำเป็นจะต้องมีมาตรฐานในการรับส่ง ซึ่งมาตรฐานที่นิยมมากที่สุดคือมาตรฐาน RS-232

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.2.1 มาตรฐาน RS-232

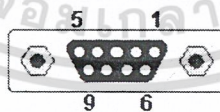
เพื่อให้ให้อุปกรณ์จากผู้ผลิตต่างกันทำงานร่วมกันได้ มาตรฐานหลายชนิดจึงได้รับการออกแบบขึ้น มาตรฐานที่ใช้กันอย่างกว้างขวางที่สุดคือ RS-232 ซึ่งโดยปกติไมโครคอมพิวเตอร์จะมีพอร์ตที่เป็นแบบอนุกรมอยู่ในตัวอยู่แล้ว และจะทำหน้าที่รับส่งข้อมูลในแบบอนุกรมตามจุดประสงค์ของมาตรฐาน RS-232 เพื่อจะสามารถเชื่อมต่อกันระหว่างอุปกรณ์รับส่งข้อมูลปลายทาง (Data Terminal Equipment : DTE) เช่น พอร์ตของคอมพิวเตอร์หลักหรืออุปกรณ์ปลายทางกับอุปกรณ์สื่อสารข้อมูล (Data Communication Equipment : DCE) หมายถึงอุปกรณ์ที่สามารถแปลงคลื่นรูปดิจิทัล ไปเป็นสัญญาณที่เหมาะสมสำหรับการส่งผ่านสายโทรศัพท์ หรือตัวกลางสื่อสารอื่นๆโดยกระบวนการผสมสัญญาณ (Modulation)

2.2.2 การอินเตอร์เฟสตามมาตรฐาน RS-232

มาตรฐาน RS-232 ใช้สัญญาณเพียงเส้นเดียวในการส่งสัญญาณ โดยสัญญาณที่ส่งไปได้ทิศทางเดียว สำหรับความเร็วและระยะทางของการเชื่อมต่อ RS-232 สามารถเชื่อมต่อถ่ายโอนข้อมูลได้ถึง 200 Kbps. ซึ่งเพียงพอสำหรับไมโครคอนโทรลเลอร์ที่มี Baud rate เท่ากับ 110-115200 bit/s ความยาวของสายเชื่อมสัญญาณระหว่าง DTE และ DCE โดยสัญญาณตามมาตรฐานของ RS-232 จำกัดอยู่ที่ 50 ฟุต หรือประมาณ 15 เมตร ซึ่งเพียงพอสำหรับการสื่อสารไมโครคอมพิวเตอร์กับอุปกรณ์ภายนอก สำหรับแรงดันของระดับสัญญาณจะถูกกำหนดลงใน 2 บริเวณ คือ

- แรงดันไฟบวก (สถานะspace) อยู่ระหว่าง +5 ถึง +15 โวลต์สำหรับเอาต์พุต และระหว่าง +3 ถึง +15 โวลต์ สำหรับอินพุตความแตกต่างมีไว้เพื่อกรณีที่แรงดันไฟฟ้าสูญหายจากความยาวของสายสัญญาณ
- แรงดันไฟลบ (สถานะmark) ถูกกำหนดไว้ระหว่าง -5 ถึง 15 โวลต์สำหรับเอาต์พุต และระหว่าง -3 ถึง -15 โวลต์สำหรับอินพุต

ซึ่งจะเห็นได้ว่าถ้าให้สายยาวเกินไป ระดับแรงดันไฟฟ้าจะตกลงเกินขอบเขตที่กำหนด นอกจากนี้ค่าความจุไฟฟ้าเกิดขึ้น จะมีผลกับคุณภาพของสัญญาณ โดยทำการเปลี่ยนสถานะจากแรงดันไฟบวกเป็นแรงดันไฟลบนั้นไม่ชัดเจน เนื่องจาก RS-232 ไม่ได้ออกแบบให้นำไปใช้กับระยะทางไกล ดังนั้นถ้าอุปกรณ์อยู่ห่างกันมาก อาจจำเป็นต้องใช้โมเด็มหรือวิธีการอื่นแทน



รูปที่ 2.7 แสดงตัวอย่างพอร์ตอนุกรม RS-232 แบบ DB9

หมายเลขขาสัญญาณ	ชื่อของสายสัญญาณ
1	Data carrier detect
2	Received data
3	Transmitted data
4	Data terminal ready
5	Signal common
6	Data set ready
7	Request to send
8	Clear to send
9	Ring indicator

ตารางที่ 2.1 รายละเอียดคอนเน็กเตอร์แบบ DB9 ตามมาตรฐาน RS – 232

รายละเอียดของขาสัญญาณ ตามมาตรฐาน RS – 232 มีดังนี้

ขา 1 Data Carrier Detect : DCD หรืออาจเรียกว่า Carrier Detect : CD ขานี้จะแอกทีฟเมื่อมีการส่งสัญญาณพาห้จากอุปกรณ์สื่อสารข้อมูล เช่น โมเด็ม สำหรับการใช้งานปกติขานี้จะไม่ได้ถูกใช้งานมากนัก

ขา 2 Receive Data : RD หรือ RxD ขานี้ใช้เพื่อรับสัญญาณอนุกรมเข้ามายังคอมพิวเตอร์โดยนำข้อมูลที่อ่านได้เก็บไว้ในรีจิสเตอร์ บัฟเฟอร์

ขา 3 Transmitted Data : TD หรือ TxD ขานี้ใช้เพื่อส่งข้อมูลออกจากคอมพิวเตอร์โดยนำข้อมูลที่เก็บอยู่ในบัฟเฟอร์สำหรับส่งข้อมูลออกไป

ขา 4 Data Terminal Ready : DTR เป็นขาสัญญาณที่ส่งออกจากคอมพิวเตอร์เพื่อให้อุปกรณ์ปลายทางรับรู้ว่าการติดต่อด้วยโดยขา DTR นี้จะต้องเชื่อมต่อกับขา DSR ของอุปกรณ์ปลายทางและขา DTR ของอุปกรณ์ปลายทางจะต้องเชื่อมต่อกับขา DSR ของคอมพิวเตอร์ ถ้าใช้การเชื่อมต่อเป็นแบบ Null Modem ซึ่งใช้สายในการเชื่อมต่อเพียง 3 เส้น จะต้องต่อขา DTR และ DSR ของตัวมันเองเข้าด้วยกันและต้องต่อกับขา DCD ด้วยในกรณีที่โปรแกรมสื่อสารที่ใช้มีการตรวจจับสัญญาณพาห้

ขา 5 Signal Ground : GND ขากราวด์ของระบบ

ขา 6 Data Set Ready : DSR ขานี้จะใช้คู่กับขา DTR เพื่อตรวจสอบการเชื่อมต่อระหว่างคอมพิวเตอร์กับอุปกรณ์ปลายทางซึ่งขา DSR นี้จะเป็นขาสำหรับรับข้อมูลที่ส่งจากภายนอกซึ่งถูกส่งมาจากขา DTR

ขา 7 Request To Send : RTS เป็นขาสำหรับส่งสัญญาณร้องขอให้ทางอุปกรณ์ปลายทางส่งข้อมูลกลับมายังคอมพิวเตอร์โดยขาที่รับสัญญาณ RTS ก็คือขา CTS ในกรณีที่ใช้การเชื่อมต่อแบบ Null modem 3 สาย จะต้องเชื่อมต่อกับขา RTS และ CTS ของตัวมันเองเข้าด้วยกัน เพื่อจะให้การรับและส่งข้อมูลสามารถเกิดขึ้นได้ตลอดเวลา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ขา 8 Clear To Send : CTS ขานี้จะคอยรับสัญญาณจากขา RTS เมื่อนับสัญญาณได้ข้อมูลที่ขา TXD จะถูกส่งออกไป ดังนั้นขานี้จึงถูกใช้เพื่อตรวจสอบอุปกรณ์ต่อพ่วงว่าพร้อมที่จะรับข้อมูลหรือไม่

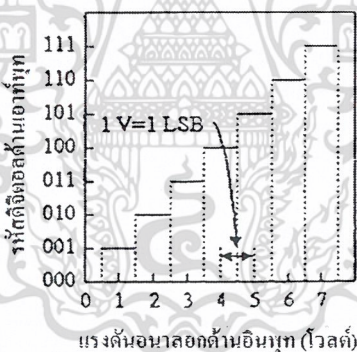
ขา 9 Ring Indicator : RI ใช้แสดงสถานะสัญญาณเรียกจากสายโทรศัพท์ ปกติในการสื่อสารโดยทั่วไปสายนี้จะไม่ถูกใช้งานจะใช้งานก็ต่อเมื่อมีการเชื่อมต่อกับโมเด็ม และโปรแกรมมีการตรวจสอบสัญญาณนี้เท่านั้นผิดพลาดที่เกิดขึ้นระหว่างการถ่ายทอดข้อมูล (ผิดพลาดจากพริตตีเฟรมข้อมูล โอเวอร์รัน) เป็นต้น

2.3 หลักการทำงานของ Analog to Digital Converter

กระบวนการต่างๆที่เกิดขึ้นตามธรรมชาติส่วนใหญ่ หากนำมาแปรค่าเป็นสัญญาณทางไฟฟ้ามักเป็นสัญญาณที่อยู่ในรูปของแรงดันหรือกระแส หรือไม่ก็เป็นลักษณะของค่าความต้านทานลักษณะที่ได้จะเป็นสัญญาณอนาล็อก ซึ่งไม่สามารถนำไปใช้กับคอมพิวเตอร์โดยตรงได้จึงจำเป็นต้องมีวงจรแปลงสัญญาณอนาล็อกเป็นสัญญาณดิจิทัล เราเรียกว่าวงจร “Analog to Digital Converter (ADC)”

2.3.1 หลักการเบื้องต้นของการแปลงสัญญาณอนาล็อกเป็นสัญญาณดิจิทัล

หากนำเอา Analog to Digital Converter (ADC) ขนาด 3 บิต มาเขียนกราฟคุณสมบัติระหว่างสัญญาณอินพุตกับเอาต์พุต สมมติว่าแรงดันอินพุต V_i เปลี่ยนค่าจาก 0-7 โวลต์ และได้สัญญาณเอาต์พุตที่เป็นสัญญาณดิจิทัลจาก 000-111 ดังแสดงในรูปที่ 2.8



รูปที่ 2.8 แสดงกราฟคุณสมบัติของเอดีซีขนาด 3 บิต

2.3.2 ค่าความละเอียดของ ADC

ค่าความละเอียดของ ADC หาได้จากการเปลี่ยนแปลงค่าแรงดันอินพุตแล้วทำให้สัญญาณดิจิทัลเปลี่ยนค่าบิตนัยสำคัญต่ำสุดไป

$$\text{ความละเอียด} = \text{ค่าแรงดันอินพุตต่อบิต} = \text{ค่าเต็มสเกลหารด้วย } 2^n - 1$$

หรือถ้าอ้างถึงเรื่อง A/D จะได้ว่า

$$\text{ความละเอียด} = 2^n$$

โดยที่ n คือจำนวนบิตของวงจร

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ถ้าสมมติว่ามีเอาต์พุต 8 เส้น โดยเอาต์พุตแต่ละเส้นแสดงสถานะทางลอจิกเป็น 0 หรือ 1 จะมีความแตกต่างทางรหัสไบนารี (Binary) ทั้งหมด 28 หรือ 256 เช่น ถ้าตัวแปลงสัญญาณมีความแตกต่างทางรหัสเอาต์พุต 256 ระดับสัญญาณอินพุตถูกแทนเป็นไบนารีจาก 00000000 ถึง 11111111 ถ้าย่านอินพุตเริ่มจาก 0-5 โวลต์ ดังนั้นค่าความละเอียดเท่ากับ $5/256 = 0.0195$ โวลต์

2.3.3 ชนิดของ เอ/ดี คอนเวอร์เตอร์

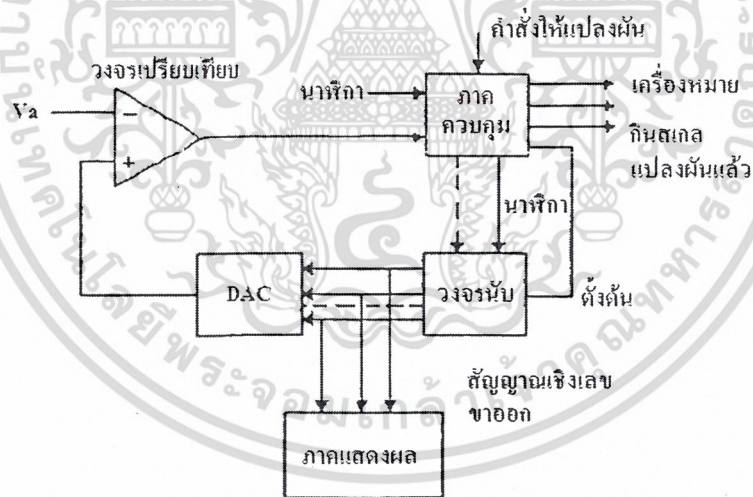
วงจร เอ/ดี คอนเวอร์เตอร์ อาจจำแนกได้ดังนี้

1. เอ/ดี ชนิด ป้อนกลับขนาน
2. เอ/ดี ชนิด อินทิเกรต
3. เอ/ดี ชนิดพร้อมกัน
4. เอ/ดี ชนิดป้อนกลับเรียงลำดับ

ในที่นี้จะกล่าวถึงวงจรเอ/ดีชนิดป้อนกลับแบบขนานเท่านั้น ซึ่งเป็นชนิดที่ใช้ใน โครงการงานเอ/ดีชนิดป้อนกลับขนาน

หลักการคือ การใช้วงจร ดีเอซี ในระบบป้อนกลับ วงจรเอ/ดี ที่เป็นวงจรประมวลผลสำเร็จรูปส่วนใหญ่จะใช้หลักการนี้ เอ/ดี ชนิดป้อนกลับขนานยังแบ่งเป็นเอ/ดี ชั้นบันได

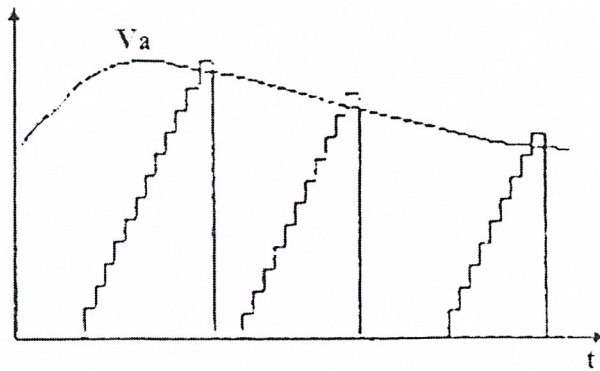
- เอ/ดี ตามรอย (Tracking)
- เอ/ดี ที่ใช้การประมาณค่าโดยลำดับ (successive approximation)



รูปที่ 2.9 แสดงบล็อกของวงจร เอ/ดี ชั้นบันไดหรือตามรอย

จากรูปที่ 2.9 แสดงแผนภาพของวงจร เอ/ดี ชั้นบันไดหรือตามรอย แล้วแต่ว่าวงจรนับเป็นแบบธรรมดาหรือแบบนับขึ้นลง

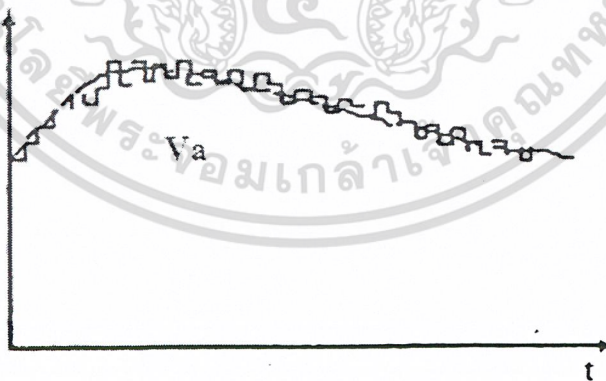
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.10 แสดงรูปคลื่นวงจร เอดีซีขึ้นบันได

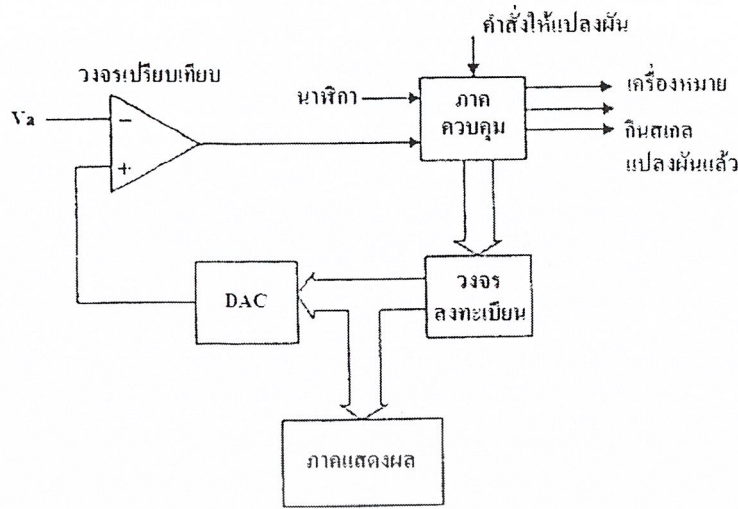
ในกรณีของวงจร เอดีซีขึ้นบันได เมื่อมีคำสั่งให้แปรผัน วงจรนับจะตั้งคั้งใหม่ (reset) สัญญาณนาฬิกาจะทำให้วงจรถับ นับขึ้นไปเรื่อยๆ ผลก็คือ แรงดันออก ดีเอซี จะเป็นรูปขึ้นบันได (ดูรูปที่ 2.10) จนกระทั่งขึ้นสูงกว่า V_{th} วงจรเปรียบเทียบจะบอกให้วงจรถวลคุมหยุดสัญญาณนาฬิกาที่ส่งให้วงจรถับจนกว่าจะมีคำสั่งให้แปรผันใหม่ หรือไม่มีวงจรถวลคุมจะบอกให้วงจรถวลคุมตั้งคั้งใหม่ และวงจรถวลคุมจะเริ่มคั้งกระบวนการแปรผันใหม่ทันที

ถ้าเราใช้วงจรถับขึ้น-ลง และให้วงจรถวลคุมทำหน้าที่ตัดคั้งว่าจะนับขึ้นหรือนับลง ถ้าแรงดันออกของ ดีเอซีต่ำกว่า V_{th} ให้นับขึ้น แรงดันออกของ ดีเอซี จะลดลงเช่นกัน สังเกตได้ว่ากรณีที่ V_{th} มีค่าคงตัวหรือเกือบคงตัว แรงดันออกของ ดีเอซีจะขึ้นลงทุกๆจังหวะของสัญญาณนาฬิกา ทำให้เกิดความไม่แน่นอนในบิตต่ำสุดของสัญญาณเชิงเลข (digital signal) ขาออก (ดูรูปที่ 2.11) อย่างไรก็ตามวงจร เอดีซีตามรอยมีข้อที่น่าสนใจคือ เวลาการแปรผัน (conversion time) จะสั้นและสัญญาณเชิงเลขที่ได้จะแทนรูปคลื่นของสัญญาณเชิงอุปมาน (analog signal) ได้ดีกว่า



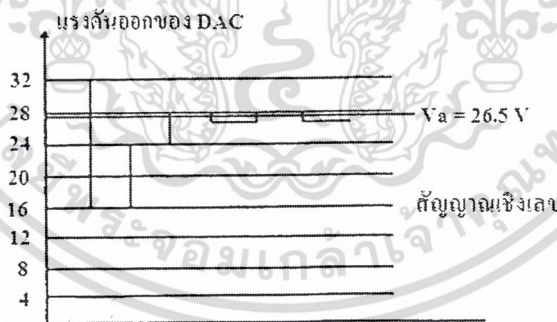
รูปที่ 2.11 แสดงรูปคลื่นของวงจรเอดีซีตามรอย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.12 แสดงบล็อกของวงจร เอดีซีที่ใช้การประมาณค่าโดยลำดับ

จากรูปที่ 2.12 แสดงแผนภาพบล็อกของวงจร เอดีซีที่ใช้การประมาณค่าโดยลำดับ จะสังเกตว่าเราใช้ วงจรทะเบียน (Resister) แทนวงจรนับ การควบคุมการทำงานของวงจรจะมีขั้นตอนที่ยุ่ยากกว่า เอดีซีตาม รอย ขั้นตอนการทำงานนี้แสดงอยู่ในแผนภาพในรูปที่ 2.11 ตัวอย่างเช่น เราต้องการแปลงผัน $V_u = 26.5$ โวลต์ เป็นสัญญาณเชิงเลข 6 บิต สมมติว่าบิตนัยสำคัญต่ำสุด (LSB) สมพันธ์กับ 1 โวลต์ ดังนั้นเลข 100000 จะ สมพันธ์กับ 32 โวลต์ซึ่งเปรียบเทียบแล้วจะใหญ่กว่า V_u ดังนั้นบิตนัยสำคัญสูงสุดเท่ากับ 0 (MSB = 0) ให้บิตที่มี นัยสำคัญรองลงไปเป็น 1 เลข 010000 จะสมพันธ์กับ 24 โวลต์เทียบ 24 โวลต์ V_u และเช่นนี้ไปเรื่อยๆไปจนครบ ทุกบิต ดังรูปคลื่นในรูปที่ 2.13

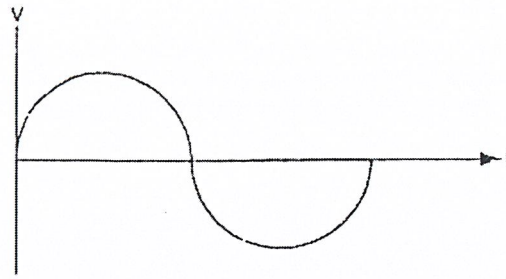


รูปที่ 2.13 แสดงรูปคลื่น

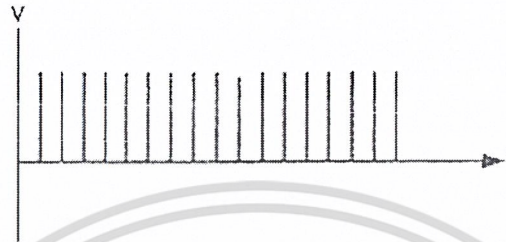
2.3.4 การแปลงสัญญาณอนาลอกเป็นดิจิทัล

ในการแปลงสัญญาณอนาลอกเป็นดิจิทัลนั้น สามารถทำได้โดยสร้างขบวนพัลส์เพื่อที่จะนำมาสุ่ม สัญญาณ โดยพัลส์นั้นจะมีความถี่ที่ค่าหนึ่งที่เราเรียกว่า ความถี่ในการส่งสัญญาณ (Sampling Frequency : f_s) เมื่อทำการมอดูเลท (Modulate) ระหว่างขบวนพัลส์กับสัญญาณอนาลอก โดยเสมือนว่าสัญญาณอนาลอกจะ ซ้อนทับมาบนขบวนพัลส์ ถ้าหากสัญญาณอนาลอกที่ถูกสุ่มถูก hold จนกว่าสัญญาณค่าใหม่จะถูกสุ่มเข้ามา จะๆได้ลักษณะของเอาท์พุทที่แสดงในรูปที่ 2.14 ง.

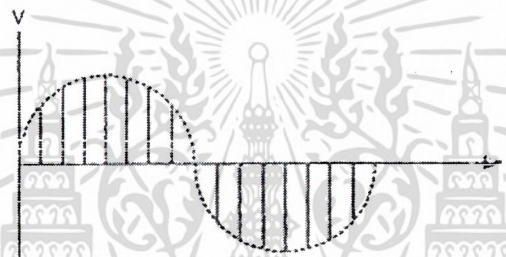
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



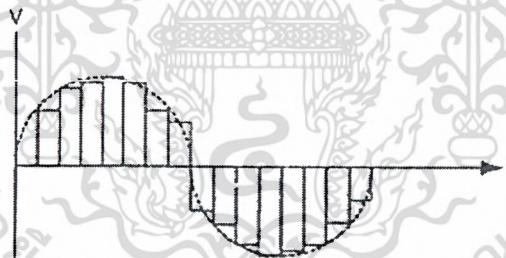
ก. สัญญาณอนาล็อกอินพุต



ข. พัลส์ที่นำมาสุ่มสัญญาณ



ค. สัญญาณอนาล็อกหลังการสุ่ม



ง. สัญญาณอนาล็อกหลังการสุ่มและ hold ไว้

รูปที่ 2.14 การสุ่มสัญญาณ

ถ้าสัญญาณต่อเนื่องซึ่งมีความถี่และฮาร์โมนิกส์ไม่เกิน f_c ถูกสุ่มด้วยอัตราการสุ่มเท่ากับ f_s ซึ่งมีค่าไม่น้อยกว่า $2f_c$ แล้วสัญญาณดังกล่าวจะสามารถเปลี่ยนกลับมาได้อย่างเดิมโดยไม่สูญเสียรายละเอียดหรือผิดเพี้ยนไป

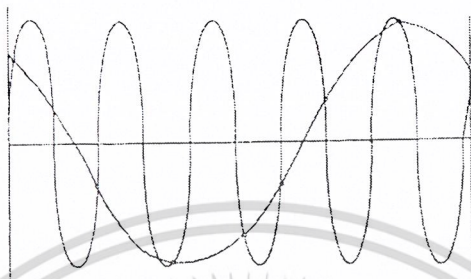
ผลของการใช้อัตราการสุ่มที่ไม่เหมาะสมจะทำให้เกิดสัญญาณรบกวนความถี่ต่ำที่เรียกว่า Aliasing frequency เมื่อสัญญาณถูกเปลี่ยนกลับมาเช่นเดิมหลังจากถูกสุ่มแล้วดังรูปที่ 2.15 จะเห็นว่าความถี่ Aliasing จะแตกต่างจากความถี่เดิมไปมาก Anti aliasing filter เป็นวงจรกรองผ่านความถี่ต่ำที่จะช่วยลดสัญญาณในแถบความถี่ที่ทำให้เกิด Aliasing frequency ในขณะที่ไม่ต้องการให้เกิดความผิดพลาดของสัญญาณในแบนด์ที่ใช้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

งานและไม่ลดความแม่นยำในการวัดโดยรวมอีกด้วย นอกจากการใช้ฟิลเตอร์ฮาร์โมนิกที่สูงๆแล้วพยายามให้ การสุ่มเป็นไปอย่างรวดเร็วมากที่สุด ซึ่งปกติแล้วจะสูงกว่าความถี่ต่ำสุดตามทฤษฎี Sampling คือ $2 f_c$

การใช้ Anti aliasing filter ขึ้นอยู่กับ

- ความถี่สูงสุดที่สนใจ
- อัตราการสุ่ม
- ความละเอียดของการแปลงสัญญาณ



รูปที่ 2.15 การเกิดความถี่ Aliasing

จากทฤษฎีการสุ่มที่ว่าให้ f_s มากกว่า $2 f_c$ นั่นก็เพื่อจัดการซ้อนกันของสเปกตรัม (Aliasing Effect) เพื่อจำกัดแบนด์วิดธ์ของสัญญาณที่ถูกแปลงไม่ให้เกินไปกว่า $f_s/2$ ซึ่งในทางปฏิบัติแล้วจะยังคงเกิด frequency folding ได้เสมอจากส่วนฮาร์โมนิกของสัญญาณและสเปกตรัมของสัญญาณรอบวงที่ยังคงมีอยู่แม้ว่าจำทำการ ฟิลเตอร์แล้วก็ตาม

2.3.5 การจัดระดับสัญญาณ (Quantizing)

อีกขั้นตอนหนึ่งที่สำคัญหลังจากที่สัญญาณผ่านการสุ่มมาแล้วก็คือ ขั้นตอนการจัดระดับของสัญญาณ โดยสัญญาณที่ผ่านการสุ่มมาแล้วจะมีสัญญาณรอบวงปะปนมาด้วยเสมอ จึงจำเป็นต้องจำกัดสัญญาณเหล่านี้ ออกไปเพื่อไม่ให้เกิดความผิดพลาดในการประมวลผลแบบดิจิทัล และสามารถนำไปเข้ารหัสได้ง่าย

เมื่อผ่านการสุ่มและการจัดระดับ แล้วข้อมูลที่สัญญาณอนาลอกจะถูกเปลี่ยนเป็นระดับสัญญาณที่แน่นอนและมีลักษณะไม่ต่อเนื่องเป็นระดับต่างๆกัน ในแต่ละสถานะของสัญญาณดิจิทัล เอาท์พุทจะแทน ขนาดของสัญญาณอนาลอกค่าใดค่าหนึ่งในช่วงแคบๆระหว่างจุดแบ่งระดับ เรียกช่วงเล็กๆนี้ว่า Analog Quantization หรือหนึ่งควอนตัม (Quantum) หรือ 1 LSB (Least Significant Bit) ของการเปลี่ยนแปลงสัญญาณ คลาดเคลื่อน (Error) ในการแทนค่าสัญญาณอนาลอกในช่วงนั้นๆอยู่ด้วย ค่าความคลาดเคลื่อนนี้เป็นธรรมชาติ ของ Quantizing ซึ่งทำการแก้ไขไม่ได้นอกจากการเพิ่มจำนวนบิตของ Quantizer ให้มากขึ้นและอีกอย่างหนึ่งที่สำคัญมากก็คือความละเอียด (Resolution) ของตัวแปลงว่ามีบิตของตัวแปลงจะบอกถึงจำนวนระดับสัญญาณ คือ

$$\text{จำนวนระดับสัญญาณ (Q)} = 2^n : n \text{ คือจำนวนบิต}$$

ถ้าเราใช้ตัวแปลงขนาด 8 บิต กับขนาดสัญญาณอนาลอกที่สูงสุด 10 โวลต์ จะได้จำนวนระดับ สัญญาณเท่ากับ 256 ระดับ ดังนั้นค่าของบิตขวาสุด (Least Significant Bit : LSB) จะมีค่าเป็น

$$\text{ค่าหนึ่งบิตด้านต่ำ} = \text{แรงดันสัญญาณด้านสูง} / \text{จำนวนระดับสัญญาณ}$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ในบางครั้งเราเรียก LSB ว่า “Step Size” โดยใช้สัญญาณจากตัวแปลง 8 บิต และขนาดสัญญาณ 10 โวลต์ จะมีค่า $10/2^8 = 10/256 = 0.039 \text{ V}$ แล้วอาจเขียนสมการในการหาค่า (Δ) ใหม่ ได้ว่า

$$(\Delta) = \text{FSR}/Q$$

เมื่อ Q คือ จำนวนระดับสัญญาณ

FSR คือ ค่าวงเต็มสเกลของแรงดันอนาล็อก (Full Scale Range)

Δ คือ ค่าความผิดพลาดจากการจัดระดับสัญญาณ

จะเห็นว่าจำนวนบิตยิ่งมากค่า Δ จะยิ่งลดลงจะทำให้ค่าความผิดพลาดลดลงไปด้วย โดยค่าความผิดพลาดจะอยู่ระหว่าง 0 ถึง $\Delta / 2$ ซึ่งมีค่าความผิดพลาดอาจจะเป็นศูนย์ถ้าสัญญาณอนาล็อกมีค่ากึ่งกลางของควอนตัมพอดี

2.3.6 การเข้ารหัส (Coding)

ในการเข้ารหัสสัญญาณที่ผ่านการสุ่ม และการจัดระดับมาแล้วนั้นส่วนใหญ่จะแปลงให้อยู่ในรูปของรหัสตัวเลขฐานสอง (Binary Code) แล้วเปลี่ยนจากข้อมูลแบบขนานให้เป็นแบบอนุกรมเพื่อสามารถลดช่องสัญญาณในการส่งข้อมูลให้เหลือเพียงช่องสัญญาณเดียวได้

จากตารางที่ 2.2 จะแสดงการเข้ารหัสของระดับแรงดันจากการสุ่มสัญญาณขนาด 0 ถึง 5 โวลต์ และผ่านการจัดระดับสัญญาณมาแล้ว ซึ่งที่ระดับแรงดันต่ำที่สุดก็จะมีขนาด 8 บิต เลขฐานสองเป็น 0000 0000 ส่วนระดับแรงดันสูงสุดก็จะมีรหัสเลขฐานสองเป็น 1111 1111 เป็นต้น

Vin	Binary Code
0	0000 0000
0.5	0001 1101
1.0	0011 0011
1.5	0100 1011
2.0	0110 0111
2.5	1000 0000
3.0	1001 1100
3.5	1011 0010
4.0	1100 1111
4.5	1110 1001
5.0	1111 1111

ตารางที่ 2.2 แสดงการเข้ารหัสสัญญาณ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากขั้นตอนทั้งหมดของการแปลงสัญญาณอนาลอกเป็นดิจิทัลนั้น เราสามารถกำหนดความสามารถหรือคุณภาพในการแปลงได้ด้วยปัจจัยหนึ่งที่สำคัญมากคือความละเอียด (Resolution) ซึ่งขึ้นอยู่กับจำนวนบิตนั่นเอง แต่ในการใช้งานจริงแล้วยังมีเรื่องความเร็วในการแปลงอีกด้วย

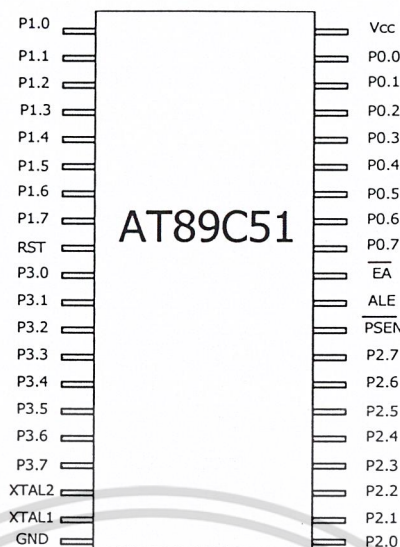
2.4 Microcontroller MCS-51

2.4.1 คุณสมบัติของ MCS-51

1. ต้องการแหล่งจ่ายไฟ 5 Volt เพียงชุดเดียว
2. มีหน่วยความจำสำหรับการเก็บโปรแกรมควบคุมการทำงานอยู่ภายในชิปจำนวน 4 กิโลไบต์
3. มีหน่วยความจำสำหรับเก็บข้อมูลทั่วไป(RAM) อยู่ภายในชิปจำนวน 128 ไบต์
4. สามารถใช้หน่วยความจำสำหรับ โปรแกรมข้อมูลที่อยู่ภายนอกชิปได้อย่างละ 64 กิโลไบต์แยกจากกัน
5. คำสั่งส่วนใหญ่ใช้เวลาการทำงาน 1 ไมโครวินาที เมื่อใช้ คริสตัลความถี่ 12 เมกะเฮิร์ตซ์
6. มีพอร์ตที่สามารถรับส่งข้อมูลได้ 2 ทิศทางจำนวน 4 พอร์ตๆ ละ 8 บิต หรือสามารถใช้งานเป็น พอร์ตขนาด 1 บิตแยกออกจากกัน
7. รับและส่งข้อมูลแบบอนุกรมได้ในตัว โดยสามารถกำหนดอัตราเร็วในการรับและส่งข้อมูล (Baud rate) ได้ตั้งแต่ 300-375 กิโลบิตต่อวินาที
8. จัดลำดับความสำคัญของสัญญาณ Interrupt ได้ 2 ระดับ
9. มี Register สำหรับใช้งานเป็น Timer หรือ Counter เพื่อนับจำนวนสัญญาณนาฬิกาภายในชิปหรือนับการเปลี่ยนแปลงสถานะของสัญญาณภายนอกขนาด 16 บิตจำนวน 2 ตัวเพื่อใช้สำหรับนับจำนวน Pulse วัดความกว้างของ Pulse หรือใช้วัดช่วงเวลา
10. หน่วยความจำสำหรับเก็บข้อมูลภายในบางส่วนสามารถเข้าถึงข้อมูลได้ทั้งระดับ ไบต์และระดับ บิต เพื่อให้การออกแบบโปรแกรมและการควบคุมระบบทำได้ง่ายขึ้น
11. มีคำสั่ง คูณและหาร เลขขนาด 8 บิตในตัวเอง
12. สามารถประมวลผลแบบบูลีนเพื่อใช้ในงานควบคุมโดยเฉพาะ
13. ใช้โปรแกรมของ Microcontroller ตระกูล MCS-48 (Upwardly compatible) ได้

2.4.2 โครงสร้างภายนอกของ MCS-51

Microcontroller ตระกูล MCS-51 ทุกเบอร์จะมีตำแหน่งขาพื้นฐานที่เหมือนกัน ดังรูปที่ 2.16



รูปที่ 2.16 แสดงตำแหน่งขาของชิป Microcontroller ตระกูล MCS-51 เบอร์ 8051

หน้าที่การใช้งานแต่ละขาของชิป Microcontroller ตระกูล MCS-51 มีดังนี้

* ขา Vss (ขา 20) สำหรับต่อลงกราวด์

* ขา Vcc (ขา 40) สำหรับต่อแหล่งจ่ายแรงดันไฟตรงขนาด 5 Volt

* ขาพอร์ต 0 (ขา 32-39) มี 8 ขา ใช้เป็นขาสำหรับพอร์ต 0 ขนาด 8 บิต (P0.0-P0.7) แบบ Open Drain Bidirectional พอร์ตนี้สามารถใช้งานเป็น Input Output พอร์ต ทั่วไปได้ โดยหากใช้งานเป็น Input พอร์ตต้อง โหลดค่า 1 ไปยังแต่ละบิตของพอร์ตนี้ เพื่อบังคับให้ขาอยู่ในสถานะถูกปล่อยลอย (มีสถานะ High impedance) นอกจากใช้งานเป็น Input Output พอร์ต แล้ว พอร์ต 0 ยังใช้ในการติดต่อหน่วยความจำสำหรับเก็บโปรแกรม และข้อมูลภายนอกชิปด้วย โดยส่งค่า แอดเดรสไบต์ค่า (A0-A7) และมัลติเพล็กซ์กับการรับส่งข้อมูล (D0-D7) จากหน่วยความจำภายนอก ในระหว่างการอ่านหรือเขียนข้อมูล โดยมีวงจรพูลอัพภายใน

* ขาพอร์ต 1 (ขา 1-8) มี 8 ขา ใช้เป็นขาสำหรับพอร์ต 1 (P1.0-P1.7) สามารถใช้งานเป็นอินพุตหรือ เอาท์พุตพอร์ตทั่วไปได้ หากต้องการใช้งานเป็นอินพุตพอร์ตต้องโหลดค่า 1 ไปยังแต่ละบิตของพอร์ตนี้เพื่อให้ มีสถานะ High impedance โดยมีวงจรพูลอัพภายใน

* ขาพอร์ต 2 (ขา 21-28) มี 8 ขา ใช้เป็นขาสำหรับพอร์ต 2 (P2.0-P2.7) ขนาด 8 บิต แบบ Open Drain Bidirectional สามารถใช้งานเป็นอินพุตหรือเอาท์พุตพอร์ตทั่วไปได้ หากต้องการใช้งาน เป็นอินพุตพอร์ต ต้อง โหลดค่า 1 ไปยังแต่ละบิตของพอร์ตนี้เพื่อให้มีสถานะ High impedance นอกจากจะใช้งานเป็นอินพุตเอาท์พุต พอร์ตทั่วไปแล้ว พอร์ต 2 ยังใช้ในการติดต่อหน่วยความจำสำหรับจัดเก็บ โปรแกรมและข้อมูลภายนอกด้วย โดยใช้สำหรับส่งค่าแอดเดรสไบต์สูง (A8-A15) และมีวงจรพูลอัพภายใน

* ขาพอร์ต 3 (ขา 10-17) มี 8 ขา ใช้เป็นขาสำหรับพอร์ต 3 (P3.0-P3.7) สามารถใช้งานเป็นอินพุตหรือ เอาท์พุตพอร์ตทั่วไปได้ หากต้องการใช้งาน เป็นอินพุตพอร์ต ต้องโหลดค่า 1 ไปยังแต่ละบิตของพอร์ตนี้ เพื่อให้มีสถานะ High impedance โดยใช้วงจรพูลอัพภายใน

นอกจากนี้ยังใช้งาน ยังใช้งานในหน้าที่พิเศษต่างๆอีกหลายอย่างดังนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ขา P3.0 ใช้รับส่งข้อมูลจากภายนอกแบบอนุกรม

ขา P3.1 ใช้ส่งข้อมูลออกไปภายนอกแบบอนุกรม

ขา P3.2 ใช้เป็นอินพุตเพื่อรับสัญญาณอินเตอร์รัปต์ชนิดที่ 0

ขา P3.3 ใช้เป็นอินพุตเพื่อรับสัญญาณอินเตอร์รัปต์ชนิดที่ 1

ขา P3.4 สัญญาณอินพุต ให้เคาเตอร์ของไทม์เมอร์ 0

ขา P3.5 สัญญาณอินพุต ให้เคาเตอร์ของไทม์เมอร์ 1

ขา P3.6 ใช้เป็นสัญญาณควบคุมการเขียนข้อมูลไปยังหน่วยความจำสำหรับการจัดเก็บข้อมูลภายนอกชิป

ขา P3.7 ใช้เป็นสัญญาณควบคุมการอ่านข้อมูล จากหน่วยความจำสำหรับการจัดเก็บข้อมูลภายนอกชิป

การใช้งาน 3 พอร์ตในหน้าที่พิเศษดังกล่าวนี้ จะต้องโหลดค่า 1 ไปยังแต่ละบิตที่ต้องการใช้งานก่อนทุกครั้ง

* ขา RST(ขา 9) ใช้สำหรับการรีเซ็ตวงจรทุกอย่างภายในชิป เพื่อเริ่มต้นการทำงานใหม่ การรีเซ็ตใช้เมื่อเริ่มจ่ายพลังงานหรือเมื่อโปรแกรมเกิดการทำงานผิดพลาด เมื่อต้องการรีเซ็ตชิป MCS-51 ขานี้ต้องมีสถานะ 1 อย่างน้อย 2 Machine cycle ระหว่าง ที่ออสซิลเลเตอร์ยังทำงานอยู่ โดยต้องต่อตัวต้านทานค่า 8.2 กิโลโอห์มเพื่อทำหน้าที่พูลดาวน์ (รักษาระดับแรงดัน ไฟฟ้าให้มีสถานะเป็นกราวด์) และเพื่อให้ตัวชิปรีเซ็ตเองเมื่อเริ่มจ่ายพลังงานให้ต่อตัวเก็บประจุขนาด 10 ไมโครฟารัด ครอบระหว่างขา RST กับ Vcc

* ขา ALE/PROG (ขา30) เป็นขาสำหรับใช้ส่งสัญญาณออกไปภายนอก เพื่อควบคุมการแลตช์ค่าแอดเดรสไปตค่า (address latch enable) จากพอร์ต 0 ในระหว่างการติดต่อหน่วยความจำสำหรับเก็บโปรแกรมหรือข้อมูลภายนอก ปกติเมื่อไม่มีการติดต่อหน่วยความจำภายนอก ขานี้จะส่ง pulse ออกมาด้วยความถี่ 1/8 ของความถี่ออสซิลเลเตอร์ที่ใช้ตลอดเวลา ดังนั้นเราสามารถใช้เวลาที่ได้ออกมาได้จากขานี้ไปใช้งานอย่างอื่นได้ แต่ความถี่ที่ขานี้จะลดลงครึ่งหนึ่งในระหว่างติดต่อกับหน่วยความจำสำหรับเก็บข้อมูลที่อยู่ภายนอกชิปนอกจากนี้ ขา ALE สำหรับควบคุมการเขียนโปรแกรมลงไปใน EPROM สำหรับ MCS-51 เบอร์ที่มีหน่วยความจำสำหรับเก็บโปรแกรมภายในชิปเป็น EPROM

* ขา PSEN (ขา29) ใช้ส่งสัญญาณสโตรบ เพื่ออ่านคำสั่งจากโปรแกรมที่เก็บไว้ในหน่วยความจำภายนอกชิป (Program strobe enable) เมื่อชิปทำงานด้วยโปรแกรมจากภายนอกขานี้จะส่งสัญญาณสโตรบ 2 ครั้งในแต่ละแมชชีน-ไซเคิล แต่ในช่วงการเขียนหรืออ่านข้อมูลกับหน่วยความจำภายนอกหรือเมื่อใช้โปรแกรมจากหน่วยความจำสำหรับเก็บโปรแกรมภายในชิปจะไม่มีสัญญาณออกมาจากขานี้

* ขา EA/Vpp (ขา 31) เป็นขาสำหรับใช้เลือกให้ MCS-51 ทำงานจากโปรแกรมที่อยู่ภายในหรือภายนอกชิป โดยหากขานี้มีสถานะเป็น 0 หมายถึงให้ใช้โปรแกรมจากหน่วยความจำที่เก็บโปรแกรมจากภายนอก หากขานี้มีสถานะเป็น 1 หมายถึงบังคับให้ MCS-51 ใช้โปรแกรมจากหน่วยความจำสำหรับเก็บโปรแกรมภายในชิป และสำหรับ MCS-51 ที่มีหน่วยความจำสำหรับเก็บโปรแกรมภายในชิป สามารถเลือกให้ทำงานได้จากโปรแกรมที่เก็บไว้ในหน่วยความจำภายนอกชิป ด้วยการต่อขา EA กับไฟเลี้ยงหรือ กราวด์ตามลำดับ ส่วนใน MCS-51 ที่ไม่มีหน่วยความจำสำหรับจัดเก็บโปรแกรมภายในชิป ให้ต่อขานี้ลงกราวด์เสมอ

* ขา XTAL 1(ขา 19) ใช้ต่อคริสตอลภายนอก โดยเป็นอินพุตเข้าสู่วงจรออสซิลเลเตอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

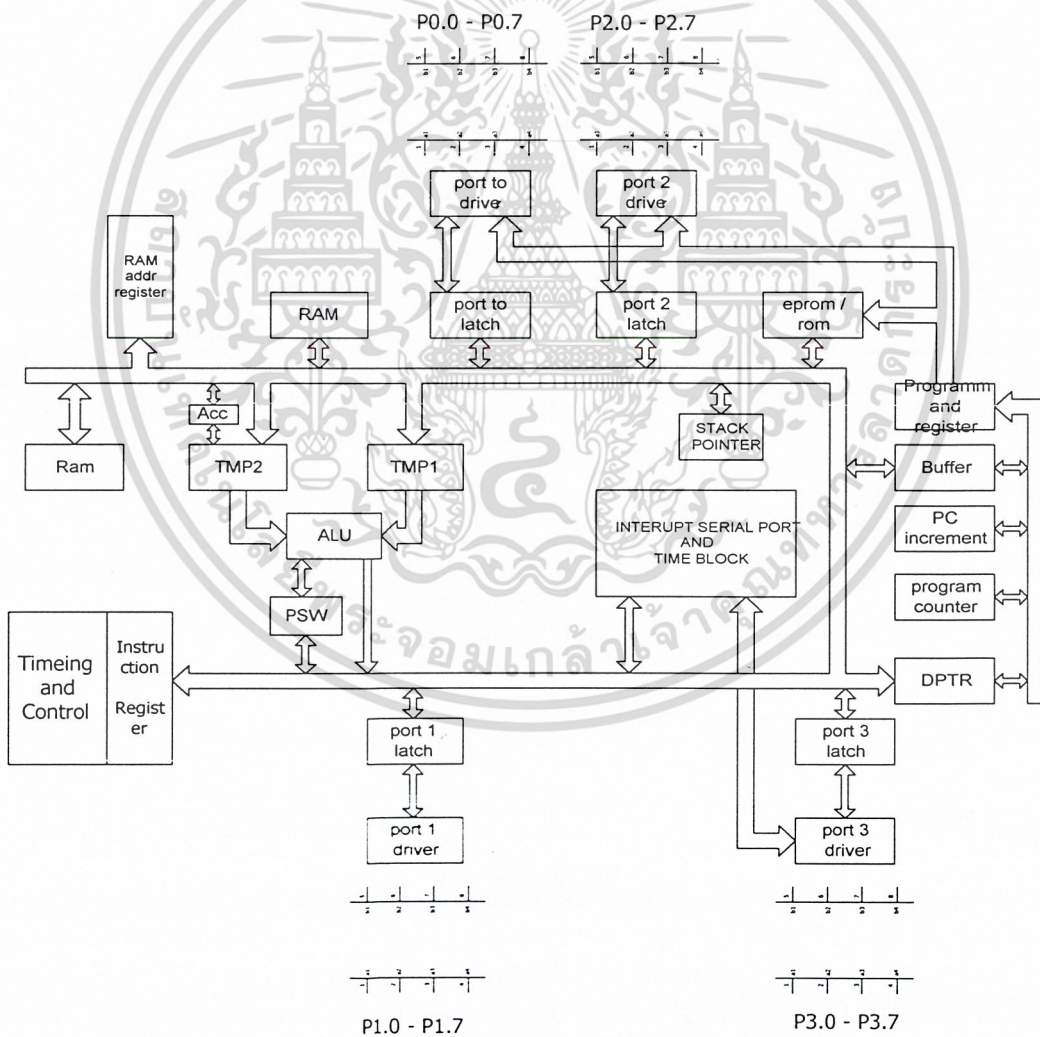
* ขา XTAL 1(ขา 19) ใช้ต่อคริสตัลภายนอก โดยเป็นเอาต์พุตออกจากวงจรออสซิลเลเตอร์

2.4.3 โครงสร้างภายในของ MCS-51

โครงสร้างภายในของ Microcontroller MCS-51 Series แสดงดังรูป 2.17 ในไมโครคอนโทรลเลอร์ตระกูล MCS-51 แบ่งชนิดหรือหน้าที่ของหน่วยความจำออกเป็น 2 ส่วนคือ

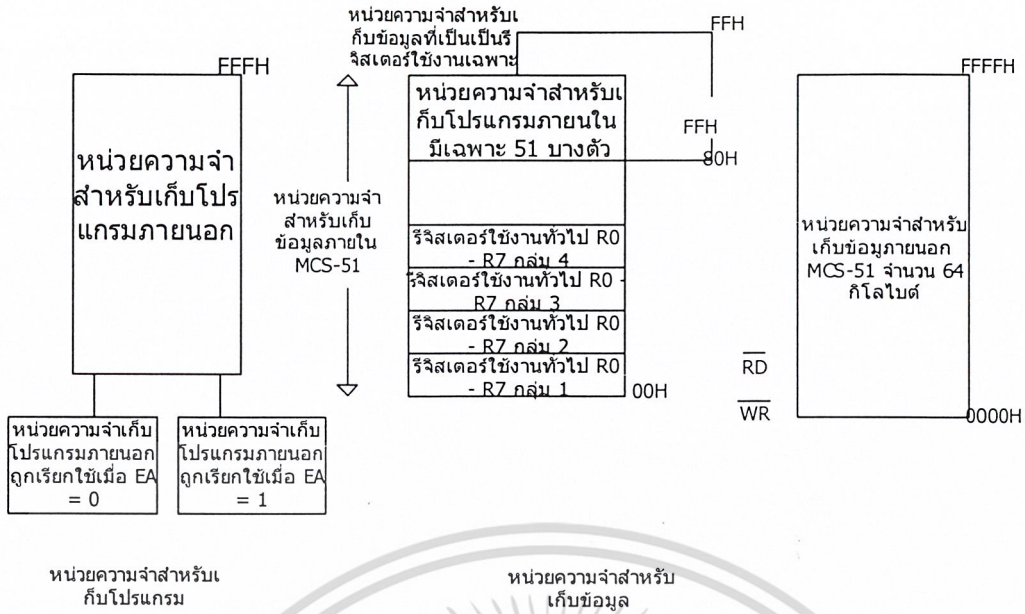
- หน่วยความจำโปรแกรม (Program memory)
- หน่วยความจำข้อมูล (Data memory)

หน่วยความจำสำหรับเก็บโปรแกรมจะใช้เก็บโปรแกรมควบคุมการทำงานของชิป MCS-51 บางเบอร์จะมีหน่วยความจำส่วนนี้อยู่ภายในชิป แต่บางเบอร์จะไม่มี ทำให้ต้องเก็บโปรแกรมไว้ในหน่วยความจำภายนอกทั้งหมด ส่วนหน่วยความจำส่วนที่สอง คือหน่วยความจำสำหรับเก็บข้อมูลซึ่งใช้สำหรับเก็บข้อมูลระหว่างการทำงาน MCS-51 จะมีหน่วยความจำส่วนนี้อยู่ภายในชิปจำนวนหนึ่ง แต่จะมีมากหรือน้อยขึ้นอยู่กับเบอร์ของชิป โครงสร้างของหน่วยความจำทั้งหมดของ MCS-51 มักแสดงในรูปที่ 2.17



รูปที่ 2.17 โครงสร้างหน่วยความจำภายใน MCS-51

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.18 หน่วยความจำสำหรับเก็บ โปรแกรม

2.4.3.1 หน่วยความจำสำหรับเก็บโปรแกรม

หน่วยความจำสำหรับเก็บ โปรแกรมใน MCS-51 จะแบ่งออกเป็น 2 ส่วนคือ หน่วยความจำสำหรับเก็บโปรแกรมภายในชิป (Internal program memory) และหน่วยความจำสำหรับเก็บโปรแกรมภายนอกชิป (External program memory) ขนาดของหน่วยความจำสำหรับเก็บ โปรแกรมภายในชิปมีได้ตั้งแต่ 0,4,8,16 กิโลไบต์ ขึ้นอยู่กับเบอร์ของชิป

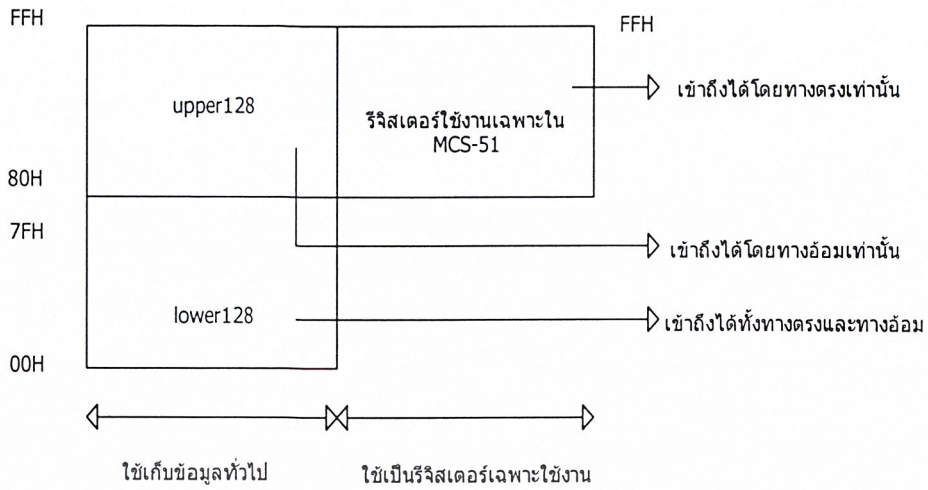
2.4.3.2 หน่วยความจำสำหรับเก็บข้อมูล

หน่วยความจำสำหรับเก็บข้อมูลของ MCS-51 จะแบ่งออกเป็น 2 ส่วนคือ หน่วยความจำสำหรับเก็บข้อมูลภายในชิปของ MCS-51 ยังแบ่งออกเป็น 2 ส่วนย่อยดังนี้

- ส่วนที่ใช้เก็บข้อมูลทั่วไป (Internal ram)
- ส่วนที่ใช้เป็นรีจิสเตอร์ใช้งานเฉพาะ (Special function register)

หน่วยความจำที่ใช้เก็บข้อมูลทั่วไปภายในชิป เป็นหน่วยความจำสำหรับเก็บข้อมูลที่มีอยู่ภายใน MCS-51 หน่วยความจำนี้มีไว้สำหรับเก็บข้อมูลขณะทำงาน ส่วนหน่วยความจำสำหรับเก็บข้อมูลที่มีอยู่ภายในชิปที่ใช้เป็นรีจิสเตอร์ใช้งานเฉพาะเป็นหน่วยความจำสำหรับเก็บข้อมูลที่มีอยู่ภายใน MCS-51 ซึ่งถูกกำหนดให้เป็นรีจิสเตอร์ใช้งานเฉพาะเพื่อควบคุมการทำงานและบอกสถานะการทำงานของชิพ แผนภาพแสดง

หน่วยความจำสำหรับเก็บข้อมูลภายในชิปทั้งสองบริเวณมีดังในรูปที่ 2.19



รูปที่ 2.19 แผนภาพแสดงหน่วยความจำสำหรับเก็บข้อมูลภายในชิป MCS-51

MCS-51 ทุกเบอร์จะมีหน่วยความจำสำหรับจัดเก็บข้อมูลทั่วไป ภายในชิปอย่างน้อย 128 ไบต์ ไปจนถึง 256 ไบต์ ทั้งนี้ขึ้นกับเบอร์ของชิป หน่วยความจำสำหรับเก็บข้อมูลทั่วไปภายในชิปบริเวณ 128 ไบต์แรก มีชื่อเรียกว่า lower 128 และในบริเวณ 128 ไบต์หลังที่มีเพิ่มในบางเบอร์มีชื่อเรียกว่า upper 128 ดังแสดงในรูปที่ 2.20

FFH	หน่วยความจำสำหรับเก็บ โปรแกรมภายใน มีเฉพาะ 51 บางตัว
80H 7FH	
2FH 20H	บริเวณหน่วยความจำที่ใช้ได้ถึง ระดับบิต จำนวน 16 ไบต์ * 8 = 128
18H	รีจิสเตอร์ใช้งานทั่วไป R0 - R7 กลุ่ม 4
10H	รีจิสเตอร์ใช้งานทั่วไป R0 - R7 กลุ่ม 3
08H	รีจิสเตอร์ใช้งานทั่วไป R0 - R7 กลุ่ม 2
00H	รีจิสเตอร์ใช้งานทั่วไป R0 - R7 กลุ่ม 1

รูปที่ 2.20 แสดงหน่วยความจำสำหรับเก็บข้อมูลทั่วไปภายในชิปทั้งสองส่วน

หน่วยความจำสำหรับเก็บข้อมูลทั่วไปภายในชิปบริเวณ 128 ไบต์หลัง(ตำแหน่ง 80 H ขึ้นไป) จะมีตำแหน่งตรงกับหน่วยความจำสำหรับเก็บข้อมูลภายในชิปที่ใช้เป็นรีจิสเตอร์ใช้งานเฉพาะ(ตำแหน่ง 80 H ขึ้นไปเช่นกัน)โดยมีวิธีเข้าถึงข้อมูลในหน่วยความจำทั้งสองส่วนไม่เหมือนกัน

2.4.3.3 รีจิสเตอร์ใช้งานเฉพาะ

เนื่องจาก MCS-51 ถูกออกแบบไว้สำหรับใช้ควบคุมระบบโดยเฉพาะ จึงทำให้มีความสามารถเฉพาะตัวหลายอย่างซึ่งจำเป็นต้องอาศัยวงจรภายในชิปที่มีเพิ่มขึ้นจากไมโครโปรเซสเซอร์ทั่วไป การควบคุมการทำงานของวงจรภายในไมโครคอนโทรลเลอร์จะกระทำผ่านรีจิสเตอร์ที่ถูกกำหนดหน้าที่ไว้แล้ว ดังนั้นการต้องการใช้ MCS-51 ให้มีประสิทธิภาพจำเป็นต้องทราบหน้าที่การทำงานของรีจิสเตอร์ใช้งานเฉพาะแต่ละตัวให้ละเอียด รีจิสเตอร์ใช้งานเฉพาะทั้งหมดจะอยู่ในหน่วยความจำสำหรับเก็บข้อมูลภายในชิปบริเวณที่ใช้เป็นรีจิสเตอร์ใช้งานเฉพาะ

ในส่วนของหน่วยความจำสำหรับเก็บ โปรแกรมและหน่วยความจำสำหรับเก็บข้อมูลที่อยู่ภายนอกชิปจะเป็นหน่วยความจำที่อยู่ภายนอกชิป MCS-51 ที่ผู้ผลิตต้องคิดติดตั้งเพิ่มเอง การติดต่อระหว่างหน่วยความจำทั้งสอง กับ MCS-51 จะใช้ขา 32- 39 (พอร์ต 0) เป็นตัวส่งค่าแอดเดรสไปต่อกับ (A0-A7) และใช้รับส่งข้อมูลกับหน่วยความจำด้วย(ใช้เป็นดาต้าบัส) ส่วนค่าแอดเดรสไปต่อกับ (A8-A15) จะใช้ขา 21-28 (พอร์ต2) ดังนั้น เมื่อพอร์ต0 และพอร์ต2 ถูกใช้ในการติดต่อกับหน่วยความจำภายนอก (ทั้งหน่วยความจำสำหรับจัดเก็บโปรแกรมและหน่วยความจำสำหรับจัดเก็บข้อมูล) จะทำให้เหลือพอร์ตสำหรับใช้ทำงานอย่างอื่นน้อยลง

2.4.3.4 รีจิสเตอร์สำหรับใช้งานทั่วไป

รีจิสเตอร์สำหรับใช้งานทั่วไป มีไว้สำหรับให้ผู้เขียน โปรแกรมสามารถนำข้อมูลไปพักไว้ชั่วคราวหรือใช้งานทั่วไปได้ตามที่ต้องการ ซึ่งรีจิสเตอร์ใช้งานทั่วไปมีอยู่ 8 ตัว คือ R0-R7 โดยรีจิสเตอร์ทั้ง 8 ตัวถูกจัดให้อยู่รวมกันและมีให้เลือกใช้ถึง 4 แบนก์ (bank) นั่นคือ รีจิสเตอร์ใช้งานทั่วไป 32 ตัวให้ใช้งาน เพียงแต่การเลือกใช้รีจิสเตอร์ R0-R7 ในแบนก์ใดแบนก์หนึ่งจะถูกกำหนดจากบิต RS0,RS1 ในรีจิสเตอร์หน้าที่พิเศษ PSW ดังนั้นการเลือกใช้ จึงเลือกใช้ได้เพียงแบนก์เดียวในขณะใดขณะหนึ่ง อย่างไรก็ตามค่าข้อมูลที่เก็บไว้ในรีจิสเตอร์แบนก์ใดก็ตามที่มีชื่อเดียวกันแต่อยู่คนละแบนก์ จะไม่มีผลซึ่งกันและกันเลย ทำให้ผู้เขียนโปรแกรมใช้งานรีจิสเตอร์ทั่วไปนี้ได้ทั้ง 32 ตัว อย่างเต็มที่และไม่ยุ่งยากในการเขียน โปรแกรม

2.4.4 โครงสร้างพอร์ต

MCS-51 ทุกเบอร์จะมีพอร์ตขนาด 8 บิตจำนวน 4 พอร์ต(P0,P1,P2,P3) โดยสามารถกำหนดให้ทำงานแบบพอร์ตขนาด 8 บิต 4 พอร์ต หรือจะใช้เป็นแบบขนาด 1 บิต 32 พอร์ต ทั้งนี้ผู้ใช้สามารถกำหนดให้แต่ละพอร์ตใช้งานเป็นอินพุตพอร์ตหรือเอาต์พุตพอร์ตได้อย่างใดอย่างหนึ่งได้อย่างอิสระ

ในกรณีที่ผู้ออกแบบต้องการใช้หน่วยความจำภายนอก ไม่ว่าจะเป็นหน่วยความจำสำหรับเก็บข้อมูลหรือสำหรับโปรแกรม พอร์ต 0 จะถูกกำหนดการใช้งานเป็นตัวส่งค่าแอดเดรสไปต่อกับ และบางส่วนของพอร์ต 3 จะถูกใช้ส่งสัญญาณควบคุม หรือคอนโทรลบัส (สัญญาณที่ใช้ควบคุมการอ่านหรือการเขียนข้อมูล) แต่หากหน่วยความจำที่ใช้ภายนอกต้องการไม่ถึง 64 กิโลไบต์ พอร์ต 2 ที่ใช้เป็นแอดเดรสไปต่อกับจะไม่ถูกนำมาใช้ทั้งหมดแต่พอร์ต 0

จะถูกใช้ทั้งหมด 8 เส้นเพราะต้องใช้เป็นดาต้าบัส ส่วนพอร์ต 3 จะนำมาใช้ติดต่อกับหน่วยความจำด้วยหรือไม่ (ขึ้นอยู่กับหน่วยความจำภายนอกว่ามีหน่วยความจำส่วนที่ใช้เก็บข้อมูลด้วยหรือไม่(ต้องการสัญญาณควบคุมการอ่านหรือเขียนข้อมูลด้วยหรือไม่นั่นเอง) ดังนั้นในการออกแบบระบบ หากต้องการใช้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หน่วยความจำภายนอกมากขึ้นเพียงใด ก็จะทำให้เหลือจำนวนพอร์ตที่นำมาใช้งานน้อยลง ในการออกแบบจริงจึงต้องพยายามลดขนาดหน่วยความจำภายนอกให้เหลือน้อยที่สุด

พอร์ต 3 ซึ่งมีขนาด 8 บิต นอกจากจะใช้ส่งสัญญาณสำหรับการอ่านหรือการเขียนข้อมูลในการติดต่อกันหน่วยความจำสำหรับเก็บข้อมูลภายนอกชิปแล้ว มันยังถูกใช้เป็นตัวรับสัญญาณอินเทอร์พรีต (INT0,INT1) สัญญาณอินพุตที่ต้องการนับสำหรับเคาน์เตอร์(T0,T1) รวมทั้งใช้ในการติดต่อสื่อสารข้อมูลแบบอนุกรมกับอุปกรณ์ภายนอก (รับและส่งข้อมูลผ่านขา RXD,TXD) อีกด้วย

ภายในแต่ละพอร์ตที่ใช้เป็นอินพุตพอร์ตหรือเอาต์พุตพอร์ต ผู้ใช้สามารถกำหนดให้ทำงานเป็นอินพุตหรือเอาต์พุตพอร์ตโดยอิสระ โดยอาศัยการควบคุมจากโปรแกรม ซึ่งสามารถควบคุมให้แต่ละพอร์ตถูกใช้เป็นอินพุตในช่วงเวลาหนึ่งและเป็นเอาต์พุตในอีกช่วงเวลาหนึ่งได้

2.4.5 ไทม์เมอร์และเคาน์เตอร์

ใน MCS-51 มีรีจิสเตอร์ใช้งานเฉพาะที่สามารถนับจำนวนสัญญาณนาฬิกาหรือแมชชีน ไซเคิลของวงจรรอสซิงเคลเตอร์ภายใน (ทำงานเป็นไทม์เมอร์) หรือนับจำนวนครั้งของการเปลี่ยนสัญญาณภายนอก (นับจำนวนพัลส์ภายนอก) ที่ขา T0,T1 ของพอร์ต 3 ทำงานเป็นเคาน์เตอร์ รีจิสเตอร์ที่ใช้เป็นไทม์เมอร์หรือเคาน์เตอร์มีขนาด 16 บิต จำนวน 2 ตัว คือรีจิสเตอร์ไทม์เมอร์ 0 หรือรีจิสเตอร์ไทม์เมอร์ 1 ตามลำดับ (ในเบอร์ 8052 มีรีจิสเตอร์ไทม์เมอร์ 2 ตัว เพิ่มให้อีก 1 ตัว) เมื่อต้องการใช้ไทม์เมอร์ 0 หรือไทม์เมอร์ 1 จะต้องโหลดค่าที่ต้องการนับไปไว้ในรีจิสเตอร์ไทม์เมอร์ 0 หรือรีจิสเตอร์ไทม์เมอร์ 1 และเมื่อนับได้ครบจำนวนที่ตั้งไว้ จะมีสัญญาณอินเทอร์พรีตเพื่อบอกให้ซีพียูทราบ

การควบคุมการทำงาน ไทม์เมอร์หรือเคาน์เตอร์ สามารถควบคุมได้จากวงจรรภายนอก(ควบคุมด้วยสัญญาณขา INT0,INT1) หรือควบคุมจากคำสั่งใน โปรแกรม ดังนั้นรีจิสเตอร์ที่ใช้เป็นไทม์เมอร์ใน MCS-51 จะสามารถวัดช่วงห่างของเวลา วัดความกว้างของพัลส์ หรือนับจำนวนครั้งของเหตุการณ์ที่เกิดขึ้นภายนอกที่เปลี่ยนให้อยู่ในรูปของสัญญาณ ไฟฟ้าแล้ว รวมทั้งใช้กำเนิดสัญญาณอินเทอร์พรีตที่มีคาบเวลาแน่นอน

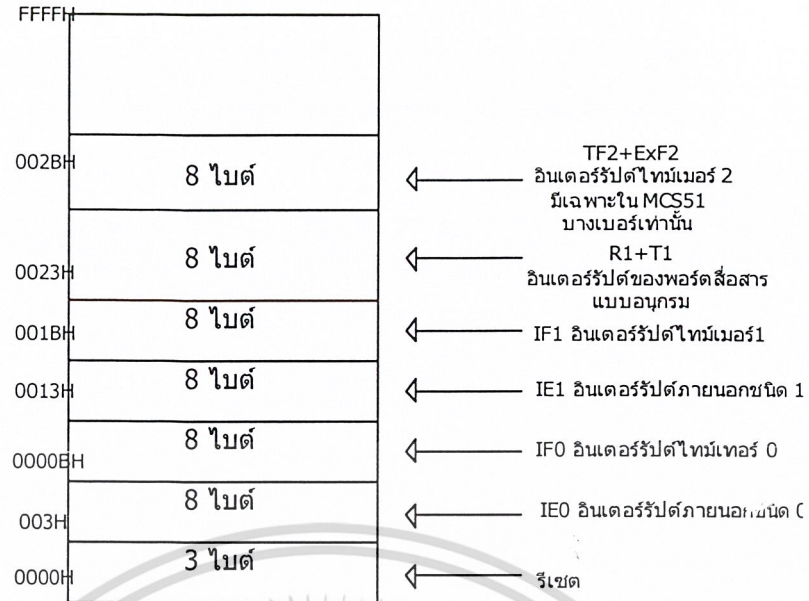
2.4.6 พอร์ตสื่อสารแบบอนุกรม

MCS-51 สามารถรับและส่งข้อมูลแบบอนุกรมได้โดยไม่ต้องพึ่งอุปกรณ์ภายนอกแบบอื่นๆแต่อย่างใด ในด้านอัตราเร็วของการรับและส่งข้อมูลก็สามารถกำหนดค่าได้ตามความต้องการของผู้ใช้โดยสามารถเลือกอัตราเร็วในการรับส่งข้อมูล(band rate)มาตรฐานได้ตั้งแต่ 110 , 1.2K , 2.4K,4.8K , 9.6K , 19.2K , 375K ตามมาตรฐานของ UART นอกจากนี้ยังสามารถกำหนดการทำงานที่แตกต่างกันได้ถึง 4 รูปแบบ ตามความเหมาะสมในแต่ละงาน

2.4.7 โครงสร้างการอินเทอร์พรีต

MCS-51 สามารถรับส่งสัญญาณอินเทอร์พรีตได้ถึง 5 ชนิด โดยจะเป็นสัญญาณอินเทอร์พรีตที่เกิดจากภายนอก 2 ชนิด และที่เกิดจากภายในชิปอีก 3 ชนิด เมื่อมีสัญญาณอินเทอร์พรีตเกิดขึ้น MCS-51 จะละการทำงานของโปรแกรมที่กำลังทำอยู่และข้ามไปทำงานโปรแกรมบริการอินเทอร์พรีต (interrupt service routine) ที่อยู่ในหน่วยความจำตำแหน่งต่างๆ ขึ้นอยู่กับชนิดของสัญญาณอินเทอร์พรีต ดังแสดงในรูปที่ 2.21

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

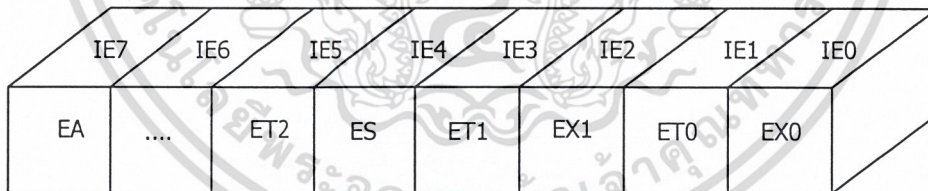


รูปที่ 2.21 แสดงตำแหน่งหน่วยความจำของโปรแกรมบริการอินเตอร์รัปต์แต่ละชนิดใน MCS-51

เราสามารถเลือกให้ซีพียูใน MCS-51 ถูกอินเตอร์รัปต์โดยสัญญาณอินเตอร์รัปต์ที่เกิดขึ้นได้ โดยการกำหนดค่ารีจิสเตอร์ใช้งานเฉพาะ IE นอกจากนี้ยังสามารถควบคุมลำดับความสำคัญในการตอบสนองต่อสัญญาณอินเตอร์รัปต์ของ MCS-51 ได้ด้วยรีจิสเตอร์ใช้งานเฉพาะ IP

2.4.8 รีจิสเตอร์ใช้งานเฉพาะ IE (Interrupt Enable - Register)

เข้าถึงข้อมูลใดในระดับบิต รายละเอียดแสดงดังรูปที่ 2.22



รูปที่ 2.22 รีจิสเตอร์ใช้งานเฉพาะ IE

บิต ชื่อบิต

IE1 EA ใช้ควบคุมการตอบสนองต่อสัญญาณอินเตอร์รัปต์ทั้งหมด
 0 : MCS-51 จะไม่ตอบสนองต่อสัญญาณอินเตอร์รัปต์ใดๆทั้งสิ้น
 1 : การตอบสนองต่อสัญญาณอินเตอร์รัปต์แต่ละชนิด จะถูกควบคุมโดยตรงจากบิตที่ทำ

หน้า

ที่ควบคุมการตอบสนองต่อสัญญาณอินเตอร์รัปต์ ซึ่งอยู่ในรีจิสเตอร์เช่นกัน

IE6 -- ไม่ถูกกำหนดใช้งาน(สำรองไว้ใช้ใน MCS-51 ใหม่ๆในอนาคต)

IE5 ET2 ควบคุมการตอบสนองต่อสัญญาณอินเตอร์รัปต์ของไทม์เมอร์ 2 เมื่อเกิด overflow (มีเฉพาะ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
 ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

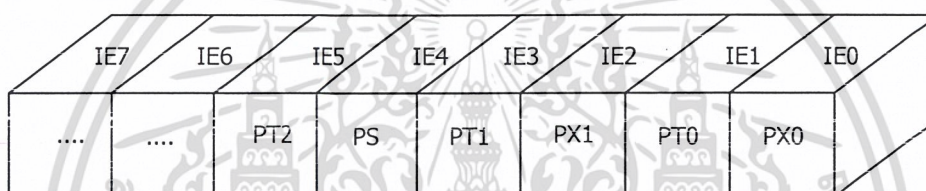
MCS-51 บางเบอร์ที่มีไทม์เมอร์ 2 เช่น 8052)

IE4	ES	ควบคุมการตอบสนองต่อสัญญาณอินเทอร์รัปต์ของพอร์ตสื่อสารอนุกรม
IE3	ET1	ควบคุมการตอบสนองต่อสัญญาณอินเทอร์รัปต์ของไทม์เมอร์ 1 เมื่อเกิด overflow
IE2	EX1	ควบคุมการตอบสนองต่อสัญญาณอินเทอร์รัปต์ภายนอกชนิด 1
IE1	ET0	ควบคุมการตอบสนองต่อสัญญาณอินเทอร์รัปต์ของไทม์เมอร์ 0 เมื่อเกิด overflow
IE0	EX0	ควบคุมการตอบสนองต่อสัญญาณอินเทอร์รัปต์ภายนอกชนิด 0

* หมายเหตุ ถ้าบิตที่ควบคุมการตอบสนองต่อสัญญาณอินเทอร์รัปต์แต่ละบิตมีค่าเป็น 1 หมายถึงอนุญาตให้ MCS-51 ตอบสนองต่อสัญญาณอินเทอร์รัปต์ได้ หากมีค่าเป็น 0 หมายถึงไม่อนุญาตให้ MCS-51 ตอบสนองต่อสัญญาณอินเทอร์รัปต์ได้

2.4.9 รีจิสเตอร์ใช้งานเฉพาะ IP (Interrupt Priority Register)

เข้าถึงข้อมูลได้ในระดับบิต ดังแสดงในรูปที่ 2.23



รูปที่ 2.23 รีจิสเตอร์ใช้งานเฉพาะ IP

บิต	ชื่อบิต	
IP7	--	ไม่ถูกกำหนดใช้งาน(สำรองไว้ใช้ใน MCS-51 ใหม่ๆในอนาคต)
IP6	--	ไม่ถูกกำหนดใช้งาน(สำรองไว้ใช้ใน MCS-51 ใหม่ๆในอนาคต)
IP5	PT2	กำหนดลำดับความสำคัญในการตอบสนองต่อสัญญาณอินเทอร์รัปต์ของไทม์เมอร์ 2
IP4	PS	กำหนดลำดับความสำคัญในการตอบสนองต่อสัญญาณอินเทอร์รัปต์ของพอร์ตสื่อสารอนุกรม
IP3	PT1	กำหนดลำดับความสำคัญในการตอบสนองต่อสัญญาณอินเทอร์รัปต์ของไทม์เมอร์ 1
IP2	PX1	กำหนดลำดับความสำคัญในการตอบสนองต่อสัญญาณอินเทอร์รัปต์ของไทม์เมอร์ 1
IP1	PT0	กำหนดลำดับความสำคัญในการตอบสนองต่อสัญญาณอินเทอร์รัปต์ของไทม์เมอร์ 0
IP0	PX0	กำหนดลำดับความสำคัญในการตอบสนองต่อสัญญาณอินเทอร์รัปต์ของไทม์เมอร์ 0

* หมายเหตุ ค่าในบิตที่กำหนดลำดับความสำคัญของสัญญาณอินเทอร์รัปต์แต่ละชนิด หากเป็น 1 หมายถึงกำหนดให้มีลำดับความสำคัญสูง หากเป็น 0 กำหนดให้มีความสำคัญต่ำ

2.4.10 กลุ่มคำสั่งใน MCS-51

คำสั่งที่ใช้ควบคุมการทำงานของ MCS-51 มี 2 ประเภทคือ คำสั่งที่ต้องการข้อมูลมาดำเนินการเพื่อให้ได้ผลลัพธ์ และคำสั่งที่ไม่ต้องการข้อมูลดำเนินการ คำสั่งที่ต้องการข้อมูลจะต้องมีวิธีในการเข้าถึงข้อมูลได้หลายวิธีดังนี้

วิธีการเข้าถึงข้อมูลในคำสั่ง

- วิธีการเข้าถึงข้อมูลโดยตรง(Direct addressing)
- วิธีการเข้าถึงข้อมูลโดยอ้อม(Indirect addressing)
- วิธีการเข้าถึงข้อมูลในรีจิสเตอร์ใช้งานทั่วไป(Register instruction)
- วิธีการเข้าถึงข้อมูลในรีจิสเตอร์เฉพาะของตัวคำสั่ง(Register-specific instruction)
- วิธีการเข้าถึงข้อมูลที่กำหนดเองโดยตรง(Immediate constant)
- วิธีการเข้าถึงข้อมูลที่มีตัวชี้อ้างอิง(Indexed addressing)

คำสั่งแต่ละคำสั่งที่ต้องการข้อมูลหรือโอเปอเรนด์(operand) อาจมีวิธีในการเข้าถึงข้อมูลในโอเปอเรนด์ได้วิธีเดียวหรือหลายวิธีขึ้นกับคำสั่งแต่ละคำสั่ง รายละเอียดของวิธีการเข้าถึงข้อมูลของโอเปอเรนด์แต่ละวิธีมีดังนี้

2.4.10.1 วิธีการเข้าถึงข้อมูลโดยตรง(Direct addressing)

เป็นวิธีกำหนดตำแหน่งหน่วยความจำโดยตรงในคำสั่งบริเวณหน่วยความจำที่สามารถอ้างอิงได้ โดยวิธีนี้จะป็นหน่วยความจำสำหรับจัดเก็บข้อมูลทั่วไปภายในชิปบริเวณ 128 ไบต์แรก และหน่วยความจำสำหรับจัดเก็บข้อมูลที่อยู่ภายนอกชิป รวมทั้งหน่วยความจำสำหรับเก็บข้อมูลที่อยู่ภายในชิปที่ใช้เป็นรีจิสเตอร์ที่ใช้งานเฉพาะ

2.4.10.2 วิธีการเข้าถึงข้อมูลโดยอ้อม(Indirect addressing)

เป็นวิธีการเข้าถึงข้อมูลโดยทางอ้อม โดยค่าตำแหน่งหน่วยความจำจะอยู่ในรีจิสเตอร์เฉพาะบางตัว นั่นคือวิธีนี้จะใช้ค่าในรีจิสเตอร์เป็นตัวชี้ตำแหน่งหน่วยความจำ หน่วยความจำที่สามารถใช้วิธีการเข้าถึงแบบนี้ได้คือ หน่วยความจำสำหรับใช้เก็บข้อมูลทั่วไปบริเวณ 128 ไบต์ล่าง และ 128 ไบต์บน และทั้งหน่วยความจำสำหรับเก็บข้อมูลที่อยู่ภายนอกชิป รีจิสเตอร์ที่สามารถนำมาใช้เป็นตัวชี้ตำแหน่งของหน่วยความจำมีดังต่อไปนี้

- รีจิสเตอร์ใช้งานทั่วไป R0.R1 ของแต่ละกลุ่ม
- รีจิสเตอร์ใช้งานเฉพาะ SP (Stack pointer)
- รีจิสเตอร์ใช้งานเฉพาะ DPTR (Data pointer)

การใช้วิธีเข้าถึงข้อมูลโดยอ้อมนี้ รีจิสเตอร์ที่เก็บค่าตำแหน่งหน่วยความจำ จะต้องระบุเครื่องหมาย @ ไว้ข้างหน้า ดังตัวอย่าง

```
MOV A,@R0
```

```
MOV X@DPTR,A
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.4.10.3 วิธีการเข้าถึงข้อมูลในรีจิสเตอร์ใช้งานทั่วไป(Register instruction)

ข้อมูลที่ต้องการจะอยู่ในรีจิสเตอร์ใช้งานทั่วไป R0-R7 ของแต่ละกลุ่มรีจิสเตอร์ที่ถูกเลือกใช้งานในขณะนั้น โดยในการทำงานจริงๆ ซีพียูจะตรวจสอบกลุ่มรีจิสเตอร์ที่ถูกเลือกใช้งานจากบิต RS0,RS1

2.4.10.4 วิธีการเข้าถึงข้อมูลในรีจิสเตอร์เฉพาะของตัวคำสั่ง(Register-specific instruction)

คำสั่งบางคำสั่งของ MCS-51 จะระบุไว้แล้วว่าต้องดำเนินการกับข้อมูลในรีจิสเตอร์ตัวใด เช่น ACCUMULATOR

DPSR , SP ดังนั้นในรหัสคำสั่ง(opcode) ของคำสั่งประเภทนี้ MCS-51 จะรู้เองได้ว่าต้องประมวลผลกับรีจิสเตอร์ตัวใด ด้วยเหตุนี้คำสั่งในกลุ่มนี้จึงไม่ต้องบอกตำแหน่งของรีจิสเตอร์ที่ใช้งานเฉพาะคำสั่งนี้แต่อย่างใด เช่น

```
MOV A,#data
```

```
MOVE DPTR,#data
```

จากตัวอย่างนี้ เราไม่จำเป็นต้องระบุตำแหน่งขาของรีจิสเตอร์ A,DPTR ในรหัสคำสั่งของคำสั่งทั้งสองแต่อย่างใด เพราะ MCS-51 จะทราบเองว่าเป็นรีจิสเตอร์ทั้งสองจากรหัสของคำสั่ง

2.4.10.5 วิธีการเข้าถึงข้อมูลที่กำหนดเองโดยตรง(Immediate constant)

เป็นการกำหนดค่าข้อมูลให้กับคำสั่งโดยตรง ข้อมูลที่นำมาประมวลผลในคำสั่ง จะอยู่หลังรหัสคำสั่ง ทั้งนี้จะต้องใช้เครื่องหมาย # ระบุหน้าข้อมูลที่ต้องการ เช่น

```
MOV A,#100
```

2.4.10.6 วิธีการเข้าถึงข้อมูลที่มีตัวชี้อ้างอิง(Indexed addressing)

ข้อมูลวิธีการอ้างอิงแบบนี้จะเป็นข้อมูลที่อยู่ในหน่วยความจำ สำหรับเก็บโปรแกรมภายใน หรือภายนอกชิปเท่านั้น จุดประสงค์ของการอ้างอิงข้อมูลแบบนี้ มีไว้เพื่อใช้ในการเปิดหาค่าข้อมูลที่เก็บไว้ในหน่วยความจำสำหรับ โปรแกรม ซึ่งข้อมูลจะไม่สูญหาย แม้จะไม่มีพลังงาน ในการทำงานของคำสั่งที่ใช้การอ้างอิงวิธีนี้ จะใช้ค่าของรีจิสเตอร์ใช้งานเฉพาะ DPTR หรือ PC มารวมกับค่าในรีจิสเตอร์ A เพื่อชี้ไปยังตำแหน่งของหน่วยความจำสำหรับเก็บโปรแกรมซึ่งเก็บข้อมูลไว้ ดังนั้นค่ารีจิสเตอร์ในการใช้งานเฉพาะ DPTR,PC จะต้องมีค่าเท่ากับตำแหน่งต้นของหน่วยความจำส่วนที่เก็บข้อมูลที่ต้องการ ส่วนค่าของรีจิสเตอร์ A จะเป็นตัวเลือกข้อมูลที่อยู่ในหน่วยความจำ เช่น

```
MOVC A,@A+DPTR
```

```
MOVC A,@A+PC
```

ความจริงการอ้างอิงข้อมูลวิธีนี้ยังมีใช้ในคำสั่งควบคุมลำดับการทำงานของโปรแกรม โดยการบังคับให้โปรแกรมกระโดดข้ามไปทำงานที่ตำแหน่งใดๆ ในหน่วยความจำซึ่งมีค่าตำแหน่งเท่ากับผลรวมของค่าในรีจิสเตอร์ใช้งานเฉพาะ DPTR กับรีจิสเตอร์ A ดังตัวอย่างต่อไปนี้

```
JMP @A+DPTR
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.5 ทฤษฎีสเปกตรัมแถบความถี่

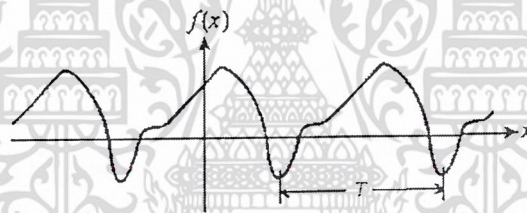
สัญญาณ (Signal) และ/หรือ ฟังก์ชัน (Function) ต่าง ๆ ที่เกิดขึ้นในปัญหาทางกายภาพ (Physical problem) และปัญหาทางวิศวกรรม (Engineering problem) มักจะเป็นฟังก์ชันเป็นคาบ (Periodic function) ซึ่งฟังก์ชันเหล่านี้สามารถแทนให้อยู่ในเทอมหรือพจน์ของฟังก์ชันคาบของสัญญาณไซน์และโคไซน์ ทั้งนี้แล้วสัญญาณทั้งสองจะมีความสำคัญมากเพราะเป็นสัญญาณพื้นฐานสำหรับการพิจารณาสัญญาณอื่นๆ และทั้งนี้ในการแทนแต่ละเทอมของอนุกรม (Series) ด้วยฟังก์ชันไซน์และโคไซน์เกิดขึ้นโดยแนวความคิดของนักฟิสิกส์และนักคณิตศาสตร์ชาวฝรั่งเศสชื่อ Joseph Fourier ดังนั้นเพื่อเป็นเกียรติแก่เขาจึงเรียกฟังก์ชันดังกล่าวนี้ว่า อนุกรมฟูเรียร์ (Fourier series) ซึ่งอนุกรมฟูเรียร์เป็นเหมือนเครื่องมืออันหนึ่งที่น่ามาใช้แก้ปัญหาต่าง ๆ เช่น แก้ปัญหาเชิงอนุพันธ์สามัญ (Ordinary differential equation) และสมการเชิงอนุพันธ์ย่อย (Partial differential equation) และสมการเชิงอนุพันธ์ย่อย (Partial differential equation)

2.5.1 ฟังก์ชันเป็นคาบ (Periodic Function)

ฟังก์ชันเป็นคาบ $f(x)$ เรียกว่าฟังก์ชันที่มีคาบครบรอบ T ถ้า T เป็นค่าจริงใดและไม่เท่ากับศูนย์ ดังนั้น

$$f(x+T) = f(x) \quad \text{ตลอดทุกค่า } x \quad (1)$$

ซึ่ง T คือคาบครบรอบของฟังก์ชัน $f(x)$ เช่น $\sin x$ และ $\cos x$ ที่มีคาบครบรอบเท่ากับ 2π

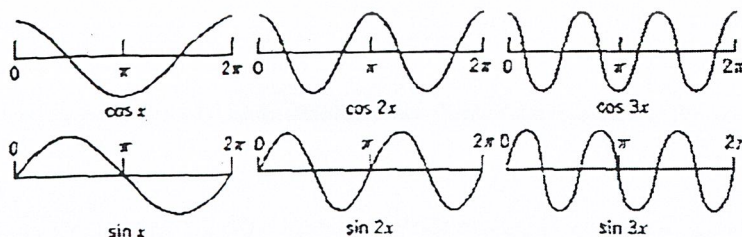


รูปที่ 2.24 แสดงฟังก์ชันเป็นคาบ

สมการที่ (1) และรูปที่ 2.24 จะเห็นได้ว่าสัญญาณที่มีคาบครบรอบเท่า T จะเกิดขึ้นซ้ำกันในทุก ๆ ค่าของ T ดังนั้นถ้า n เป็นตัวเลขจำนวนเต็มสามารถกำหนด :

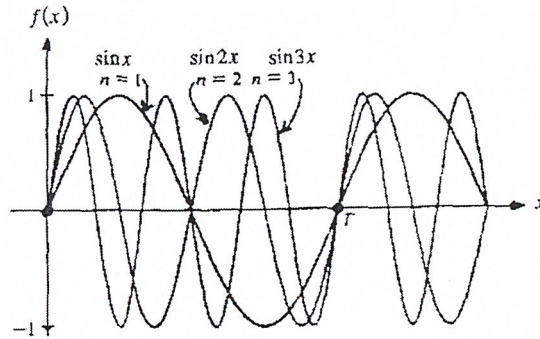
$$f(x+nT) = f(x) \quad \text{ตลอดทุกค่า } x \quad (2)$$

ดังนั้นถ้า $nT = 2T, 3T, 4T, \dots$ เป็นคาบของฟังก์ชัน ซึ่งสัญญาณจะเกิดขึ้นซ้ำกันตามจำนวนรอบหรือค่าของ



รูปที่ 2.25 สัญญาณ (ฟังก์ชัน) ไซน์และโคไซน์ที่มีคาบครบรอบเท่ากับ 2π

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 2.26 สัญญาณไซน์ที่ฮาร์โมนิก $n=1,2$ และ 3

2.5.2 อนุกรมฟูรีเยร์ตรีโกณมิติ (Trigonometric Fourier Series)

จากในหัวข้อที่ 1.2 เราได้ทราบว่า $\sin nx$ และ $\cos nx$ มีค่าต่างๆตลอดช่วงทั้งหมด $(x_0, x_0 + T)$ ซึ่งเมื่อ $n=0$ ค่า $\sin nx = 0$ แต่ $\cos nx = 1$ เพราะฉะนั้นถ้า $f(x)$ เป็นฟังก์ชันใดๆ จึงสามารถแทนให้อยู่ในเทอมของฟังก์ชันทั้งหมดที่เกิดขึ้นตลอดช่วง $(x_0, x_0 + T)$ ได้ดังนี้คือ

$$f(x) = a_0 + a_1 \cos x + a_2 \cos 2x + \dots + a_n \cos nx + \dots + b_1 \sin x + b_2 \sin 2x + \dots + b_n \sin nx + \dots \quad (1)$$

เมื่อ $(x_0 < x < x_0 + T)$

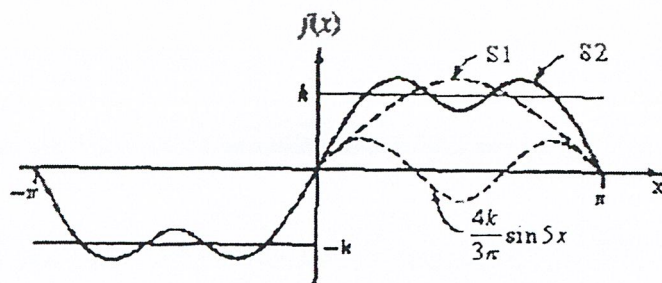
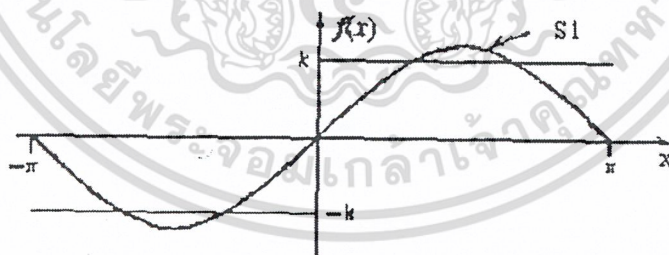
สมการที่ (3) สามารถเขียนให้อยู่ในรูปผลบวกได้ดังนี้คือ

$$f(x) = a_0 + \sum_{n=1}^{\infty} (a_n \cos nx + b_n \sin nx) \quad (2)$$

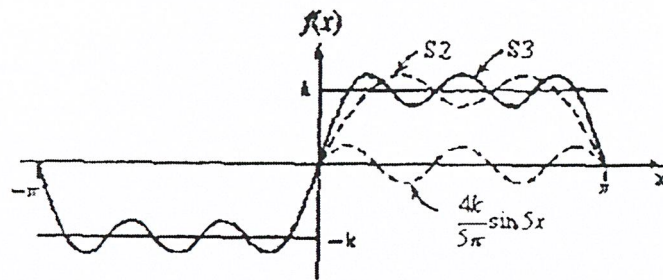
$$a_0 = \frac{1}{2\pi} \int_{-\pi}^{\pi} f(x) dx \quad (3)$$

$$a_n = \frac{1}{\pi} \int_{-\pi}^{\pi} f(x) \cos nx dx \quad (4)$$

$$b_n = \frac{1}{\pi} \int_{-\pi}^{\pi} f(x) \sin nx dx \quad (5)$$



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.27 แสดงการแทนฟังก์ชันรูปสี่เหลี่ยมด้วยฟังก์ชันไซน์และโคไซน์

อนุกรมฟูเรียร์นำไปใช้ในการแก้ปัญหาในทางวิศวกรรม โดยทั่วไปทฤษฎีของอนุกรมฟูเรียร์จะค่อนข้างยุ่งยากซับซ้อน แต่จะง่ายต่อการนำมาใช้ ซึ่งอนุกรมฟูเรียร์นำไปใช้ได้ทั่ว ๆ ไปกว้างกว่าอนุกรมเทเลอร์ (Taylor series) ทั้งนี้เพราะบางครั้งในทางปฏิบัติจะเกี่ยวข้องกับฟังก์ชันไม่เป็นคาบ (Nonperiodic function) และ/หรือฟังก์ชันไม่ต่อเนื่อง (Discontinue function) ซึ่งอนุกรมฟูเรียร์สามารถนำไปใช้ได้ทั้งสองกรณี

2.5.3 อนุกรมฟูเรียร์เชิงซ้อน (Complex Form of Fourier Series)

ในการนำอนุกรมฟูเรียร์มาใช้งานส่วนใหญ่จะเป็นการง่ายและสะดวกเมื่อแทนอนุกรมฟูเรียร์ให้อยู่ในเทอมของเอกซ์โพเนนเชียลเชิงซ้อน (complex exponential) ดังนั้นจึงอาศัยการพิจารณาจากอนุกรมฟูเรียร์ที่มีคาบครบรอบ $f(x)$ ใดๆ จากสมการที่ (2) ในหัวข้อที่ 2.5.2 ได้

$$f(t) = \frac{a_0}{2} + \sum_{n=1}^{\infty} [a_n \cos(n\omega_0 t) + b_n \sin(n\omega_0 t)] \quad (1)$$

เพื่อไม่ให้เข้าใจสับสนในการพิจารณาต่อไปนี้จึงใช้แทน ω ในสมการที่ (1) ด้วย $\omega_0 = 2\pi/T$ ซึ่งในสมการที่ (1) เทอมของโคไซน์และไซน์สามารถเขียนในรูปของเอกซ์โพเนนเชียลได้จาก

$$\cos(n\omega_0 t) = \frac{1}{2}(e^{jn\omega_0 t} + e^{-jn\omega_0 t}) \quad (2)$$

และ

$$\sin nx(n\omega_0 t) = \frac{1}{2j}(e^{jn\omega_0 t} - e^{-jn\omega_0 t}) \quad (3)$$

แทนสมการที่ (2) และ (3) ในสมการที่ (1) จะได้

$$f(t) = \frac{a_0}{2} + \sum_{n=1}^{\infty} [a_n \frac{1}{2}(e^{jn\omega_0 t} + e^{-jn\omega_0 t}) + b_n \frac{1}{2j}(e^{jn\omega_0 t} - e^{-jn\omega_0 t})] \quad (4)$$

$$f(t) = \frac{a_0}{2} + \sum_{n=1}^{\infty} [\frac{1}{2}(a_n - jb_n)e^{jn\omega_0 t} + \frac{1}{2}(a_n + jb_n)e^{-jn\omega_0 t}] \quad (5)$$

ให้

$$F_0 = \frac{a_0}{2}, F_n = \frac{1}{2}(a_n - jb_n), F_{-n} = \frac{1}{2}(a_n + jb_n) \quad (6)$$

แทนสมการที่ (6) ในสมการที่ (5)

$$f(t) = F_0 + \sum_{n=1}^{\infty} (F_n e^{jn\omega_0 t} + F_{-n} e^{-jn\omega_0 t})$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

$$f(t) = F_0 + \sum_{n=1}^{\infty} F_n e^{jn\omega_0 t} + \sum_{n=1}^{\infty} F_n e^{jn\omega_0 t}$$

$$f(t) = \sum_{n=-\infty}^{\infty} F_n e^{jn\omega_0 t} \quad (7)$$

สมการที่ (7) เรียกว่า รูปแบบเชิงซ้อน (complex form) ของอนุกรมฟูเรียร์ $f(x)$ หรือเรียกว่าอนุกรมเชิงซ้อน (Complex Fourier series) ของ $f(x)$ สัมประสิทธิ์ F_n สามารถคำนวณได้ง่าย ๆ ในเทอมของ a_n และ b_n และจากสมการที่ (3) ในหัวข้อที่ 2.5.2 คือ

$$F_0 = \frac{a_0}{2} = \frac{1}{2} \cdot \frac{2}{T} \int_{-T/2}^{T/2} f(t) dt$$

$$F_0 = \frac{a_0}{2} = \frac{1}{T} \int_{-T/2}^{T/2} f(t) dt \quad (8)$$

จากสมการที่ (6) และสมการที่ (4), (5) ในหัวข้อที่ 2.5.2 จะได้

$$F_n = \frac{1}{T} (a_n - jb_n)$$

$$= \frac{1}{2} \left[\frac{2}{T} \int_{-T/2}^{T/2} f(t) \cos(n\omega_0 t) dt - j \frac{2}{T} \int_{-T/2}^{T/2} f(t) \sin(n\omega_0 t) dt \right]$$

$$= \frac{1}{2} \left\{ \int_{-T/2}^{T/2} f(t) [\cos(n\omega_0 t) dt - j \sin(n\omega_0 t)] dt \right\}$$

$$F_n = \frac{1}{T} \int_{-T/2}^{T/2} f(t) e^{-jn\omega_0 t} dt \quad (9)$$

และ

$$F_{-n} = \frac{1}{2} (a_n + jb_n) = \frac{1}{T} \int_{-T/2}^{T/2} f(t) e^{-jn\omega_0 t} dt \quad (10)$$

ถ้าเป็นฟังก์ชันจำนวนจริง ดังนั้น

$$F_{-n} = F_n^* \quad (11)$$

ซึ่ง * แสดงถึงสังยุคเชิงซ้อน (complex conjugate)

จากสมการที่ (8), (9), (10) สามารถเขียนรวมอยู่ในสูตรเดียวได้คือ

$$F_n = \frac{1}{T} \int_{-T/2}^{T/2} f(t) e^{-jn\omega_0 t} dt \quad ; n = 0, \pm 1, \pm 2, \dots \quad (12)$$

เนื่องจาก $f(t)e^{-jn\omega_0 t}$ เป็นคาบครบรอบด้วยคาบ T จากการพิจารณาที่ผ่านมาสามารถกำหนดได้ว่า

$$F_n = \frac{1}{T} \int_0^T f(t) e^{-jn\omega_0 t} dt \quad (13)$$

กำหนดให้

$$F_n = |F_n| e^{j\theta_n}, F_{-n} = F_n^* = |F_n| e^{-j\theta_n} \quad (14)$$

ซึ่ง

$$|F_n| = \frac{1}{2} (a_n^2 + b_n^2)^{1/2} \quad (15)$$

$$\theta_n = \tan^{-1} \left(-\frac{b_n}{a_n} \right) \quad \text{สำหรับทุกค่าของ } n \text{ ยกเว้นที่ } n = 0 \quad (16)$$

ในกรณีนี้ F_0 เป็นค่าจริงนั้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

$$F_0 = \frac{a_0}{2} \quad (17)$$

2.5.4 สเปกตรัมความถี่เชิงซ้อน (Complex Frequency Spectrum)

การกระจายอนุกรมฟูเรียร์ของฟังก์ชันคาบดังกล่าวข้างต้นนั้น ฟังก์ชันจะอยู่ในเทอมของส่วนประกอบความถี่ต่าง ๆ ฟังก์ชันคาบที่มีคาบเท่ากับ T จะมีส่วนประกอบของความถี่เชิงมุม $\omega_0, 2\omega_0, 3\omega_0, \dots, n\omega_0$ ฯลฯ ดังนั้นฟังก์ชันคาบ $f(t)$ ก็จะอยู่ในสเปกตรัมของความถี่เหล่านั้น ถ้าเรากำหนด $f(t)$ ใด ๆ ขึ้นมาเราก็สามารถหาสเปกตรัมของมันได้ ในทางกลับกัน ถ้าเราทราบสเปกตรัมก็สามารถหาฟังก์ชันคาบนี้ได้ เพราะฉะนั้นจึงสามารถกำหนดฟังก์ชันคาบได้ 2 วิธีคือ โทมโดเมน (time domain) และฟรีแควนซีโดเมน (frequency domain)

1. โทมโดเมน (Time domain) คือ $f(t)$ แสดงอยู่ในรูปฟังก์ชันของเวลา

2. ฟรีแควนซีโดเมน (Frequency domain) คือ $f(t)$ จะกำหนดได้จากส่วนประกอบของความถี่ต่าง ๆ ทั้งนี้สเปกตรัมจะเกิดขึ้นที่ $\omega = \omega_0, 2\omega_0, 3\omega_0, \dots, n\omega_0$ ดังนั้นสเปกตรัมจะเป็นเคิร์ปที่ไม่ต่อเนื่อง ซึ่งจะเกิดขึ้นตามค่าดิสครีต (discrete value) ของ ω ดังนั้นจึงเรียกว่า สเปกตรัมแบบดิสครีต (discrete value) เราอาจจะแสดงสเปกตรัมได้ด้วยเส้นกราฟโดยกำหนดให้แนวนอนเป็นค่า ω และแนวตั้งเป็นแอมพลิจูด (amplitude) หรือระดับของสัญญาณที่เป็นสัดส่วนของส่วนประกอบต่าง ๆ

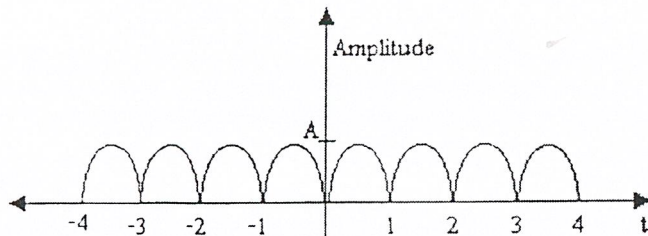
เราสามารถใช้ออนุกรมเอกซ์โพเนนเชียลหรืออนุกรมตรีโกณมิติแสดงสเปกตรัมได้การใช้รูปแบบเอกซ์โพเนนเชียลจะแสดงความหมายได้ดีกว่าและใช้ได้ทั่ว ๆ ไป ดังพิจารณาต่อไปนี้ ซึ่งสำหรับฟังก์ชันคาบที่มีคาบเท่ากับ T อนุกรมเอกซ์โพเนนเชียลจากสมการที่ (7) ในหัวข้อที่ 2.5.3 คือ

$$f(t) = \sum_{n=-\infty}^{\infty} F_n e^{jn\omega_0 t} \quad (1)$$

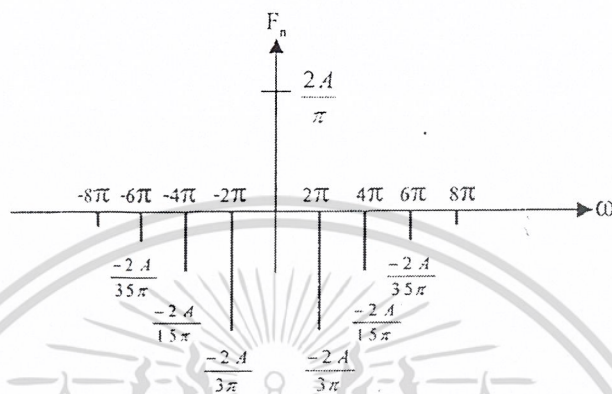
$f(t) = F_0 + F_1 e^{j\omega_0 t} + F_2 e^{j2\omega_0 t} + \dots + F_n e^{jn\omega_0 t} + \dots + F_{-1} e^{-j\omega_0 t} + F_{-2} e^{-j2\omega_0 t} + \dots + F_{-n} e^{-jn\omega_0 t} + \dots$ ซึ่งสมการที่ 1 จะประกอบด้วยความถี่ต่างๆคือ $0, \omega_0, -\omega_0, 2\omega_0, -2\omega_0, \dots, n\omega_0, -n\omega_0$ และแอมพลิจูดคือ $F_0, F_1, F_{-1}, F_2, F_{-2}, \dots, F_n, F_{-n}, \dots$

โดยทั่วไปแล้ว F_n เป็นปริมาณเชิงซ้อนจึงสามารถนำไปอธิบายสเปกตรัมเชิงแอมปริจูดของฟังก์ชันคาบ $f(t)$ จะได้จากการกำหนดขนาดของ F_n ในอนุกรมสมการที่ 1 ตามแกน y กับความถี่เชิงมุม (angular frequency) ตามแกน x และสเปกตรัมเชิงเฟส (phase spectrum) ของ $f(t)$ จะได้จากการกำหนดมุมเฟส (phase angle) θ_n ของ F_n ตามแกน y กับค่าความถี่เชิงมุมตามแกน x

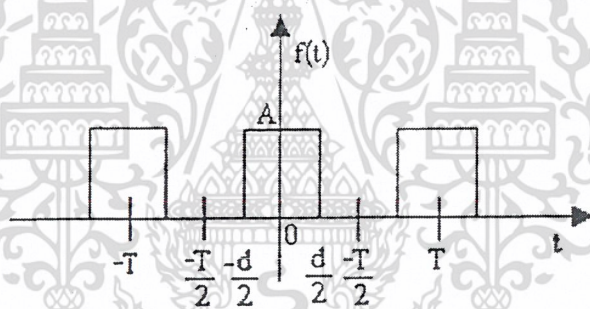
โดยทั่วไปแล้วสเปกตรัมจะเกิดขึ้นที่ $\omega = 0, \pm 2\pi, \pm 4\pi, \pm 6\pi, \dots$ และขนาด (magnitude) จะเป็นไปตามค่า $-2A/\pi, -2A/3\pi, -2A/5\pi, -2A/7\pi, \dots$ จาก $f(t)$ ซึ่งแอมปริจูดทั้งหมดเป็นค่าจริงจึงสามารถนำไปกำหนดได้เพียงสเปกตรัมเดียวเท่านั้น ซึ่งสเปกตรัมความถี่ (frequency spectrum) ได้แสดงไว้ในรูปที่ 2.28



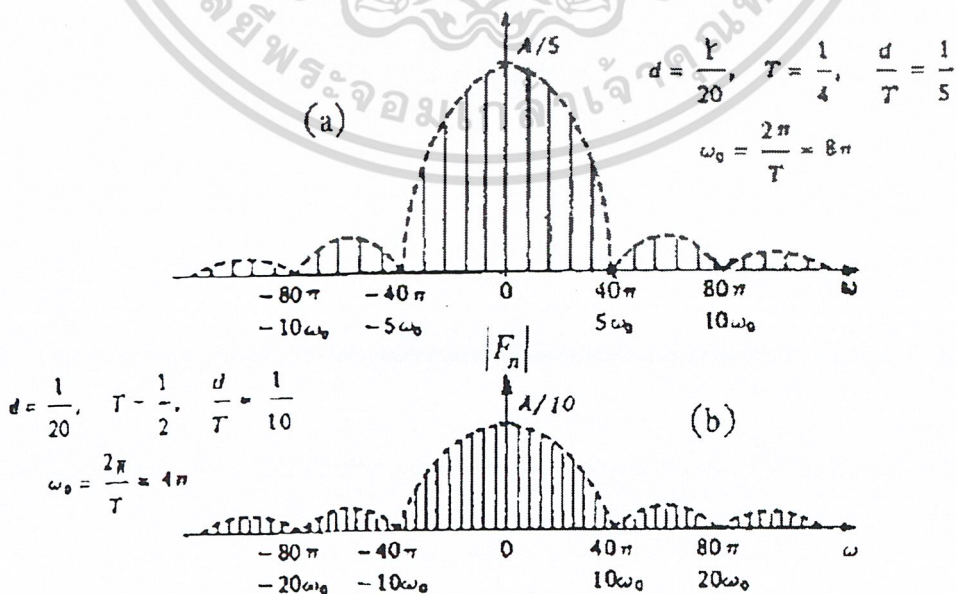
รูปที่ 2.28 สัญญาณเรกติฟาย์ไซน์เต็มคลื่น



รูปที่ 2.29 โลนีสเปคตรัมของสัญญาณเรกติฟาย์ไซน์เต็มคลื่น



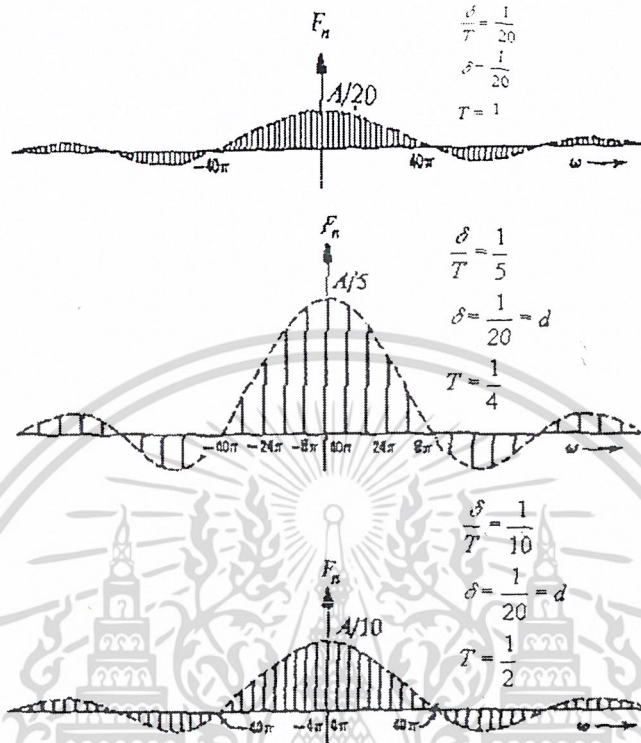
รูปที่ 2.30 สัญญาณพัลส์รูปสี่เหลี่ยมผืนผ้า



รูปที่ 2.31 สเปคตรัมเชิงแอมพลิจูด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
 ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากรูปที่ 2.31 จะเห็นได้ว่าเมื่อเวลามากขึ้นสเปกตรัมจะหนาแน่นมากขึ้น (แน่นขึ้น) แต่ขนาดหรือแอมพลิจูดจะลดลง แต่รูปร่างยังคงเหมือนเดิม เปรียบเทียบให้ชัดเจนยิ่งขึ้นดังรูปที่ 2.32



รูปที่ 2.32 สเปกตรัมเชิงแอมพลิจูด

2.5.5 สเปกตรัมเชิงกำลัง (Power Spectrum) ของฟังก์ชันคาบ

สมมติว่าฟังก์ชันคาบ $f(t)$ คือแรงดันหรือกระแส ดังนั้นอินทิกรัลคือ

$$\frac{1}{T} \int_{-T/2}^{T/2} f^2(t) dt \tag{1}$$

ซึ่งอินทิกรัล (1) แสดงถึงกำลังที่ได้จาก $f(t)$ ที่ป้อนให้กับความต้านทาน 1 โอห์ม และอินทิกรัลนี้เรียกว่ากำลังของสัญญาณคาบ $f(t)$ สเปกตรัมเชิงความถี่ของ $f(t)$ จะแสดงถึงแอมพลิจูดของส่วนประกอบความถี่ต่างๆของ $f(t)$ ซึ่งจะมีกำลังรวมอยู่ในแต่ละส่วนประกอบ ดังนั้นดังที่กล่าวไปแล้วในส่วนของสเปกตรัมเชิงแอมพลิจูด ซึ่งสามารถแทนอยู่ในรูปของความถี่ต่างๆเราสามารถแทนสเปกตรัมเชิงกำลังที่เป็นไปตามกำลังที่รวมอยู่กับส่วนประกอบของความถี่ต่างๆใน $f(t)$ ได้

พิจารณาอนุกรมฟูเรียร์ในรูปเอกซ์โพเนนเชียลจากสมการที่ 7 ในหัวข้อที่ 2.5.3 คือ

$$f(t) = \sum_{n=-\infty}^{\infty} F_n e^{jn\omega_0 t} \tag{2}$$

แทนสมการ 2 ลงในสมการ 1 ดังนั้น

$$\frac{1}{T} \int_{-T/2}^{T/2} f^2(t) dt = \frac{1}{T} \int_{-T/2}^{T/2} f(t) \sum_{n=-\infty}^{\infty} F_n e^{jn\omega_0 t} dt \tag{3}$$

เครื่องหมายอินทิเกรตและผลบวกทางด้านขวามือของสมการที่ (3) สามารถสลับที่กันได้ ดังนั้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

$$\frac{1}{T} \int_{-T/2}^{T/2} f^2(t) dt = \frac{1}{T} \int_{-T/2}^{T/2} F_n \sum_{n=-\infty}^{\infty} F_n e^{jn\omega_0 t} dt \quad (4)$$

ซึ่งจะเป็นว่าอินทิกรัลทางด้านขวามือในสมการที่ (14) มีค่าเท่ากับ TF_n ดังนั้น

$$\frac{1}{T} \int_{-T/2}^{T/2} f^2(t) dt = \sum_{n=-\infty}^{\infty} F_n F_{-n} \quad (5)$$

จากสมการที่ (11) ในหัวข้อ 2.5.3 จะได้

$$\frac{1}{T} \int_{-T/2}^{T/2} f^2(t) dt = \sum_{n=-\infty}^{\infty} |F_n|^2 \quad (6)$$

$$= F_0^2 + |F_1|^2 + |F_2|^2 + |F_n|^2 + \dots + |F_{-1}|^2 + |F_{-2}|^2 + |F_{-n}|^2 + \dots \quad (7)$$

สมการที่ (16) จะเป็นเงื่อนไขตามทฤษฎีของ ปาร์เซวาล (Parseval) ซึ่งนำไปประยุกต์ใช้กับอนุกรมฟูเรียร์ในรูปเอกซ์โพเนนเชียล และสมการที่ (17) แสดงให้เห็นว่ากำลังรวมอยู่ในแต่ละส่วนประกอบของความถี่และขนาดของกำลังที่บรรจุอยู่ ส่วนประกอบของความถี่ $n\omega_0$ คือ $|F_n|^2$ และในส่วนของ $-n\omega_0$ คือ $|F_{-n}|^2$ แต่เนื่องจาก

$$F_{-n} = F_n^*$$

จะได้

$$|F_{-n}|^2 = |F_n|^2 \quad (8)$$

ดังนั้นกำลังที่รวบรวมอยู่ในส่วนประกอบของความถี่ $n\omega_0$ จะเป็นเช่นเดียวกับที่ความถี่ $-n\omega_0$ เนื่องจากเราทราบว่าสัญญาณ $e^{jn\omega_0 t}$ และ $e^{-jn\omega_0 t}$ ทั้งสองแสดงถึงสัญญาณของความถี่เดียวกันคือ $n\omega_0$ เพราะฉะนั้นกำลังทั้งหมดที่รวมอยู่กับความถี่ $n\omega_0$ ในสัญญาณจะมีค่าเท่ากับ $|F_n|^2$ ซึ่งจะแสดงให้เห็นดังข้างล่างนี้ และส่วนประกอบทั้งสองของ $F_n e^{jn\omega_0 t}$ จะให้ฟังก์ชันจำนวนจริง (real function) ของเวลา $f(t)$ กล่าวคือ

$$f_n(t) = F_n e^{jn\omega_0 t} + F_{-n} e^{-jn\omega_0 t}$$

$$F_n = |F_n| e^{j\theta_n}$$

$$F_{-n} = |F_n| e^{-j\theta_n}$$

เพราะฉะนั้น

$$\begin{aligned} f_n(t) &= |F_n| e^{jn\omega_0 t} + |F_n| e^{-jn\omega_0 t} \\ &= 2|F_n| \cos(n\omega_0 t + \theta_n) \end{aligned} \quad (9)$$

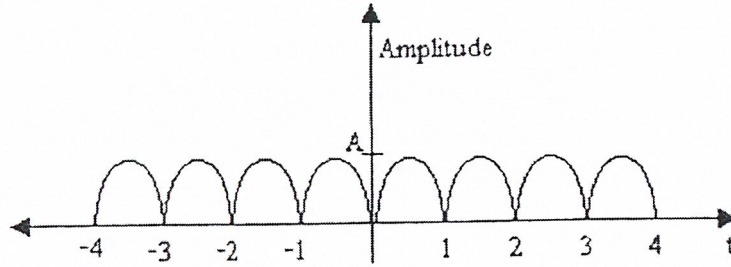
กำลังที่บรรจุในสัญญาณใดๆที่มีแอมพลิจูด A คือ $A^2/2$ ดังนั้นกำลังที่บรรจุอยู่ใน $= 2|F_n| \cos(n\omega_0 t + \theta_n)$ คือ $2|F_n|^2$ เราสามารถรวมกำลังนี้ครั้งหนึ่งกับส่วนประกอบ ** เช่นเดียวกันนี้ กำลังที่รวมกับ *** ก็คือ $|F_n|^2$ เพราะฉะนั้นสมการที่ (7) อาจจะแสดงความหมายได้ว่า กำลังในสัญญาณคาบ (periodic signal) จะมีค่าเท่ากับผลรวมของกำลังในแต่ละส่วนประกอบ

เราสามารถกำหนดสเปกตรัมกำลังที่รวมอยู่กับแต่ละส่วนประกอบของความถี่ได้ ซึ่งเห็นได้อย่างชัดเจนว่าสเปกตรัมจะเกิดขึ้นเป็นช่วงๆไม่ต่อเนื่อง (discrete) หรือไม่ติดต่อกัน ทั้งนี้จะเกิดตามค่าความถี่ต่างๆ คือที่ $\omega = 0, \pm\omega_0, \pm 2\omega_0, \dots$ ฯลฯ และจะสมมาตรทั้งซ้ายและขวากับแนวตั้งดังนั้น ถ้าเราให้ P เป็นกำลังของสัญญาณคาบ $f(t)$ จะได้

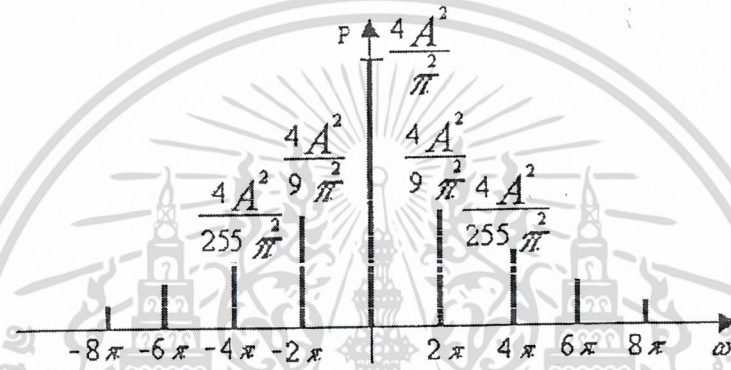
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

$$P = \frac{1}{T} \int_{-T/2}^{T/2} f^2(t) dt$$

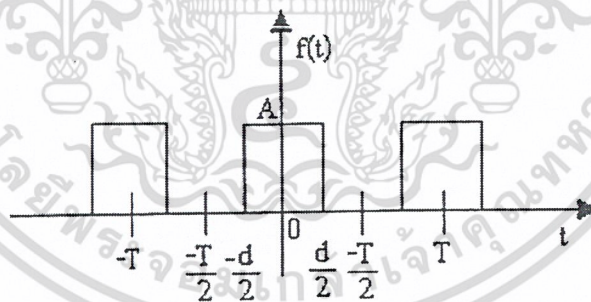
$$P = \sum_{n=-\infty}^{\infty} |F_n|^2$$



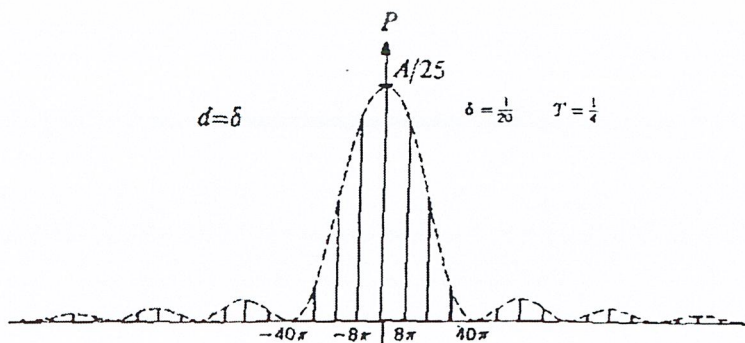
รูปที่ 2.33 สัญญาณเรกติฟาย์ไซน์เต็มคลื่น



รูปที่ 2.34 สเปกตรัมเชิงกำลังของสัญญาณเรกติฟาย์ไซน์เต็มคลื่น



รูปที่ 2.35 สัญญาณพัลส์รูปสี่เหลี่ยมผืนผ้า

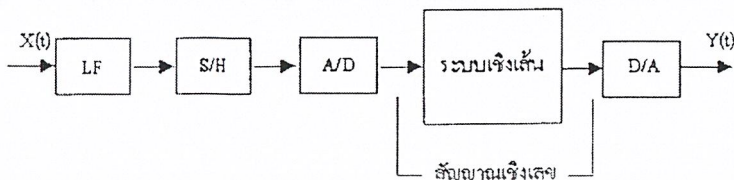


รูปที่ 2.36 สเปกตรัมเชิงกำลังของสัญญาณพัลส์รูปสี่เหลี่ยมผืนผ้า

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.6 ทฤษฎีการสุ่มตัวอย่าง (Sampling)

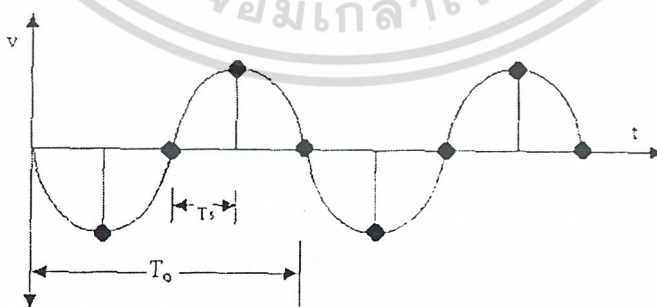
อาจกล่าวได้ว่าสัญญาณในธรรมชาติส่วนมาก เช่น สัญญาณเสียง สัญญาณการสั่นสะเทือนของพื้นโลก หรือการแปลค่าของอนุกรมหมุมิ เหล่านี้เป็นไปในลักษณะต่อเนื่องกับพิสัยเวลา หรืออาจกล่าวได้ว่า เป็นสัญญาณเชิงอนุমান การนำสัญญาณเหล่านี้ไปทำการประมวลผลในเชิงเลข หรือ การประมวลผลเชิงหน่วย (Discrete Signal Processing) ได้จำเป็นต้องใช้ระบบการประมวลผลตามรูปที่ 2.37



รูปที่ 2.37 การประมวลผลเชิงอนุমানโดยใช้ระบบการประมวลผลของสัญญาณเชิงเลข

ตามรูป วงจร S/H เป็นวงจรสุ่มและค่าสัญญาณไว้เพื่อให้วงจรแปลงอนาล็อกเป็นดิจิตอลทำการแปลงเป็นตัวเลขอีกทีหนึ่ง สังเกตว่าระบบการประมวลผลเชิงตัวเลขเป็นส่วนของระบบที่อยู่ระหว่างจุด A และ B โดยที่สัญญาณที่จุดนี้เป็นสัญญาณที่ถูกแปลงเป็นตัวเลขแล้ว อย่างไรก็ตามการแปลงสัญญาณเชิงอนุমানเป็นสัญญาณเชิงเลขนั้น ในเบื้องแรกต้องมีการสุ่มตัวอย่างก่อน ซึ่งความถี่ในการสุ่มตัวอย่างโดยที่ไม่ทำให้สัญญาณสูญเสียข้อมูลเดิมไป โดยทั่วไปแล้วเราอาจสุ่มตัวอย่างโดยใช้ค่าความถี่ $f_{SN} = 2f_0$ พอดี ค่าความถี่นี้มีชื่อเรียกว่า ความถี่ไนซ์ควิสต์ (Nyquist Frequency) และคาบเวลา $T_n = 1/(f_0)$ เรียกว่าช่วงเวลาสุ่มตัวอย่างไนซ์ควิสต์ (Interval) ในทางปฏิบัติเพื่อหลีกเลี่ยงผลของปรากฏการณ์ที่ไม่เป็นเชิงเส้น (Non Linearity) ที่อาจเกิดจากการสุ่มตัวอย่าง เรามักใช้ความถี่ในการสุ่มตัวอย่าง $2f_S$ มากกว่าค่า ความถี่ไนซ์ควิสต์ หรือ f_{SN} ขึ้นไป ส่วนมีค่ามากกว่าเป็นเท่าใดนั้นขึ้นอยู่กับลักษณะของงาน ไม่ได้มีการกำหนดค่าที่แน่นอน

ตามทฤษฎีการสุ่มตัวอย่างของแซนนอนนั้นจะเห็นได้ว่าการสุ่มตัวอย่างได้ถูกต้องนั้นก็ต่อเมื่อรู้ค่าแถบความถี่ที่ปฏิบัติงานของสัญญาณหรือพูดอีกนัยหนึ่ง สัญญาณต้องรู้แถบความถี่ที่ปฏิบัติงานจึงจะสามารถทำการสุ่มตัวอย่างได้ ดังนั้นเพื่อให้มั่นใจว่าสัญญาณที่ทำการประมวลผล ถูกประมวลผลอย่างถูกต้อง ในภาคแรกของการประมวลผลเชิงเต็มหน่วยและการประมวลผลเชิงเลข ดังแสดงในรูป 2.38



รูปที่ 2.38 แสดงตัวอย่างของความถี่ในการสุ่มสัญญาณ $T_s = 1/(4f_0)$

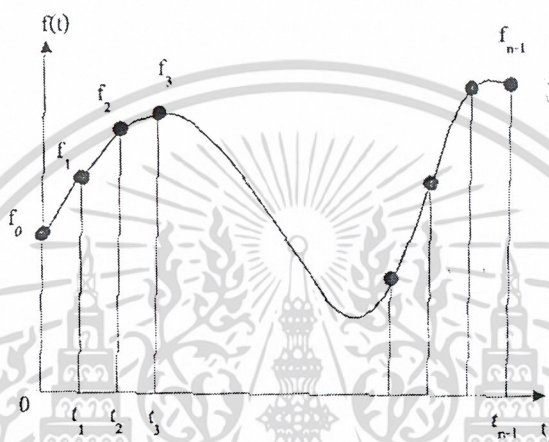
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.7 ทฤษฎีฟาสต์ฟูเรียร์ทรานฟอร์ม (Fast Fourier Transform)

ทฤษฎีฟาสต์ฟูเรียร์ทรานฟอร์มเป็นการคำนวณฟูเรียร์เพื่อสามารถคำนวณได้เร็วขึ้น โดยใช้คอมพิวเตอร์เป็นตัวควบคุมการคำนวณทั้งหมด ในการที่จะทำความเข้าใจเกี่ยวกับฟาสต์ฟูเรียร์ทรานฟอร์ม จะต้องศึกษาทฤษฎีดิสครีตฟูเรียร์เสียก่อน

2.7.1 ทฤษฎีดิสครีตฟูเรียร์ทรานฟอร์ม (The Discrete Fourier Transform)

ดิสครีตฟูเรียร์ทรานฟอร์มเป็นการแปลงฟูเรียร์แบบไม่ต่อเนื่อง โดยข้อมูลที่เราร่วมเข้ามาจะต้องเริ่มจากค่า “0” จนถึงค่า “T” ดังรูปที่ 1 โดยให้ค่า N เป็นจำนวนการสุ่มข้อมูลทั้งหมด โดยมีค่า t_n เป็นเวลาในการสุ่มข้อมูล และค่า f_n เป็นค่าของฟังก์ชันต่อเนื่องตามการสุ่มของ t_n ใดๆ



รูปที่ 2.39 ดิสครีตฟูเรียร์ทรานฟอร์ม

โดยข้อมูลที่อยู่ที่ตำแหน่ง $n = 0, 1, 2, 3, \dots, N-1$ สามารถเขียนสมการดิสครีตฟูเรียร์ทรานฟอร์มได้ดังนี้

$$F_k = \sum_{n=0}^{N-1} f_n e^{-jk\omega_0 n} \quad \text{โดยค่า } k = 0 \text{ ถึง } N-1 \quad (1)$$

ดังนั้นจะได้ค่าอินเวิรตฟูเรียร์ทรานฟอร์มดังนี้

$$f_n = 1/N \sum_{k=0}^{N-1} F_k e^{jk\omega_0 n} \quad \text{โดยค่า } n = 0 \text{ ถึง } N-1 \quad (2)$$

โดยค่า $\omega_0 = 2\pi/N$ และการแปลงค่าดิสครีตฟูเรียร์ทรานฟอร์มจากสมการ (1) ต้องทำการคำนวณเป็นจำนวน N^2 โดยข้อมูลต้องอยู่ในรูปของจำนวนเชิงซ้อนเพื่อให้คอมพิวเตอร์คำนวณได้ง่าย จากสมการ (1) สามารถใช้ทฤษฎีของออยเลอร์ (Euler's) มาคำนวณได้จากสูตรดังนี้

$$e^{-ia} = \cos a \pm i \sin a$$

แทนสูตรของออยเลอร์ลงในสมการ (1) และ (2) จะได้

$$F_k = \sum_{n=0}^{N-1} [f_n \cos(k\omega_0 n) - j f_n \sin(k\omega_0 n)] \quad (3)$$

$$f_n = \sum_{k=0}^{N-1} [F_k \cos(k\omega_0 n) - j F_k \sin(k\omega_0 n)] \quad (4)$$

จากสมการ (3) และ (4) สามารถนำไปเขียน โปรแกรมเพื่อใช้ในการคำนวณหาค่าฟูเรียร์ได้ต่อไป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.7.2 ทฤษฎีฟาสต์ฟูเรียร์ทรานฟอร์ม (FFT) และวิธีการแบบบัตเตอร์ฟลาย (Butterfly)

ฟาสต์ฟูเรียร์ทรานฟอร์ม (FFT) เป็นการพัฒนามาจากดิสครีทฟูเรียร์ทรานฟอร์ม (DFT) เพื่อให้มีการคำนวณเร็วขึ้น ซึ่งดิสครีทฟูเรียร์ทรานฟอร์มต้องการข้อมูลทีในการคำนวณเป็นจำนวน N^2 แต่ฟาสต์ฟูเรียร์ทรานฟอร์มจะใช้ข้อมูลในการคำนวณประมาณ $N \log_2 N$

ฟาสต์ฟูเรียร์ทรานฟอร์มจะคำนวณเร็วกว่าดิสครีทฟูเรียร์ทรานฟอร์มประมาณ 10 เท่า โดยจะใช้วิธีการแบบบัตเตอร์ฟลายหรือวิธีการของ คูลีย์-ทูกีย์ (Cooley-Tukey algorithm)

โดยสมมติให้ $N = 2^M$ โดย M เป็นค่าจริงใดๆ จะได้

$$F_k = X(k), f_n = x(n) \quad F_k = X(k), f_n = x(n) \quad \text{จะได้สมการใหม่ดังนี้}$$

$$X(k) = \sum_{n=0}^{N-1} x(n)e^{-j(2\pi/N)nk} \quad \text{โดย } k = 0, 1, 2, 3, \dots, N-1 \quad (5)$$

ถ้าให้ $e^{-j(2\pi/N)nk} = W^{nk}$ ได้สมการใหม่ดังนี้

$$X(k) = \sum_{n=0}^{N-1} x(n)W^{nk} \quad (6)$$

แบ่งสมการ (6) ได้เป็น 2 ส่วนดังนี้

$$X(k) = \sum_{n=0}^{N/2-1} x(2n)W_N^{nk} + \sum_{n=0}^{N/2-1} x(2n+1)W_N^{(2n+1)k} \quad (7)$$

N แทนขนาดความยาวของการสุ่มตัวอย่างและแทนสมการทั้งสองด้วยสมการคู่ และสมการที่ได้สมการใหม่คือ

$$X(k) = \sum_{n=0}^{N/2-1} x_{ev}(n)W_{N/2}^{nk} + \sum_{n=0}^{N/2-1} x_{od}(n)W_{N/2}^{nk} \quad (8)$$

$$\text{หรือ } X(k) = X_{ev}(k) + W_{N/2}^k x_{od}(k) \quad (9)$$

ในการนี้จะใช้ $N/2$ จุดเท่านั้นเพื่อคำนวณหาค่า $X(k)$ โดย k ต้องมีค่าตั้งแต่ 0 ถึง $N-1$ อย่างไรก็ตามคุณสมบัติของสมการคู่และสมการคี่หาได้จาก

$$X_{od}(k) = X_{ev}(k - N/2) \rightarrow \quad \text{โดย } N/2 \leq k \leq (N-1) \quad (10)$$

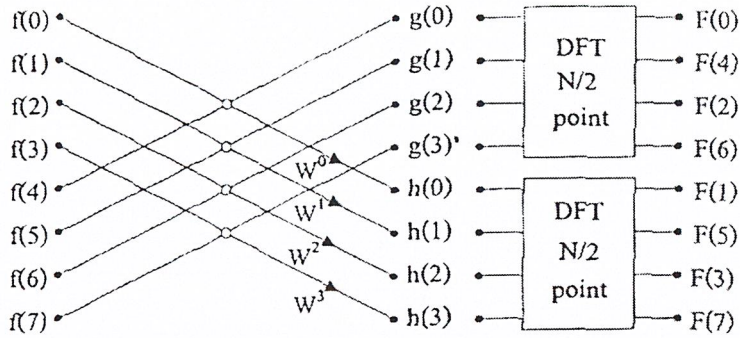
ผลลัพธ์ของการแปลงฟูเรียร์แบบไม่ต่อเนื่องสามารถทำซ้ำๆจนกระทั่งเหลือข้อมูลแค่ตัวเดียว โดยต้องมีข้อมูลอินพุต 2 ตัวเพื่อใช้ในการคำนวณตลอดเวลา

$$\Lambda(k) = \lambda(0) + \lambda(1)e^{-j2\pi k/2}$$

$$= \lambda(0) + \lambda(1) \quad \text{for all } k, \text{ for } k \text{ even}, \text{ for } k \text{ odd. ตามลำดับ}$$

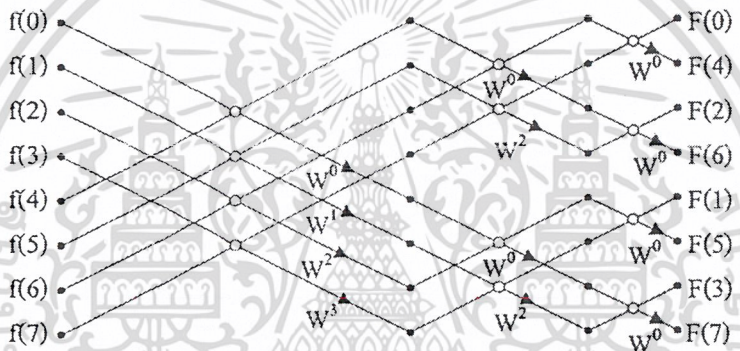
$$= \lambda(0) - \lambda(1)$$

สำหรับการหาค่าฟาสต์ฟูเรียร์ทรานฟอร์มจะใช้แค่ 2 จุด ในการคำนวณหาค่าของฟาสต์ฟูเรียร์ทรานฟอร์มที่สมบูรณ์ต้องมีการคูณโดยใช้แฟกเตอร์ที่เหมาะสมคือค่า "W" โดยค่า W จะต้องเริ่มจาก W^0 ถึง $W^{N/2-1}$ ในรูปที่ 2 แสดงกราฟของฟาสต์ฟูเรียร์ทรานฟอร์ม 8 จุด โดยต้องคำนวณหาค่าต่างๆตามวิธีการของบัตเตอร์ฟลายให้เหมาะสม



รูปที่ 2.40 การหาค่า 8 จุดฟาสต์ฟูเรียร์ทรานฟอร์ม

สำหรับการเริ่มต้นคำนวณต้องทราบค่าของ N ที่จะใช้ในการคำนวณ จากรูปที่ 2.40 จะใช้ค่า $N = 8$ และจะได้ค่า g_n และ h_n ที่คำนวณได้เป็นจำนวน $N/2$ จุด โดยมีค่า W_n มาคูณกับค่า h_n ดังรูปที่ 1 และจะคำนวณในลำดับต่อไปดังรูปที่ 2.40



รูปที่ 2.41 การหาค่า 8 จุดฟาสต์ฟูเรียร์ทรานฟอร์ม

จากรูปที่ 2.41 เมื่อคำนวณช่วงแรกเสร็จจะคำนวณช่วงที่ 2 ต่อโดยมีวิธีการคำนวณดังรูป และจะใช้ข้อมูลในการคำนวณทั้งสิ้นประมาณ $N \log_2(N)$ สามารถหาจำนวนของบิตเตอร์ฟลายได้ดังนี้

$$\text{จำนวนของบิตเตอร์ฟลาย} = N \log_2(N)$$

โดย $N/2 =$ จำนวนแถวของบิตเตอร์ฟลาย (มี 2 อินพุต)

$$\log_2(N) = \text{จำนวนหลักของบิตเตอร์ฟลาย}$$

แต่ละบิตเตอร์ฟลายต้องมี 2 อินพุตเท่านั้น

2.7.3 การคำนวณโดยใช้คอมพิวเตอร์

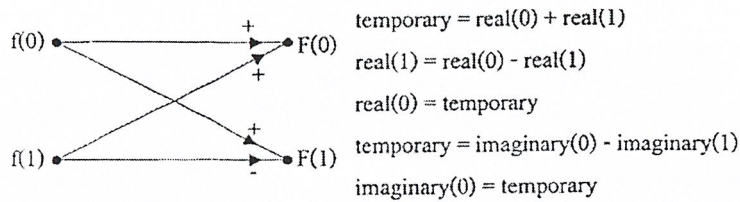
ในการคำนวณโดยใช้คอมพิวเตอร์จะต้องทำการแปลงฟังก์ชันเอ็กโปเนนท์เชิงลให้เป็นฟังก์ชัน

ตรีโกณมิติจากสูตร

$$e^{\pm ia} = \cos a \pm i \sin a$$

เนื่องจากโปรแกรมที่ใช้เขียนไม่รู้จักฟังก์ชันเอ็กโปเนนท์เชิงล ในการคำนวณจะยกตัวอย่างมาแค่บิตเตอร์ฟลายเดียว และจะต้องมีการเก็บค่าไว้ชั่วคราว ดังรูปที่ 2.42

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.42 หลักการคำนวณ โดยใช้บัตเตอร์ฟลายเดียว

จากรูปที่ 2.42 จะได้ว่า $F(0) = f(0) + f(1)$

$$F(0) = f(0) - f(1)$$

โดยข้อมูลที่ได้จะมีทั้งค่าจริงและค่าจินตภาพ

ตัวอย่างการคำนวณหาฟาสต์ฟูเรียร์ทรานฟอร์ม

ตัวอย่างที่ 1 เป็นการหาค่า 16 จุด ฟาสต์ฟูเรียร์ทรานฟอร์ม โดยมีการสุ่มข้อมูล 16 จุด และมีค่าสัญญาณดังนี้

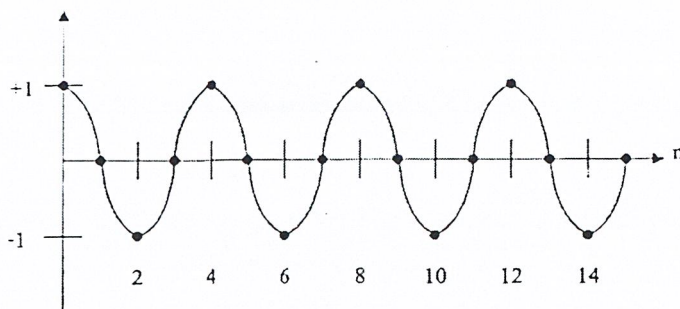
$$x(n) = \cos[2\pi(4n/16)]$$

จะเปรียบเทียบจำนวนบัตเตอร์ฟลายแบบดิสครีตฟูเรียร์ทรานฟอร์ม และฟาสต์ฟูเรียร์ทรานฟอร์ม ที่ใช้ในการคำนวณ

Length of Transform (N)	DFT Operations (N^2)	FFT Operation $N\log_2(N)$
8	64	24
16	256	64
32	1024	160
64	4096	384
128	16384	896

ตารางที่ 2.3 แสดงการเปรียบเทียบจำนวนการคำนวณระหว่างดิสครีตฟูเรียร์ทรานฟอร์ม และฟาสต์ฟูเรียร์ทรานฟอร์ม

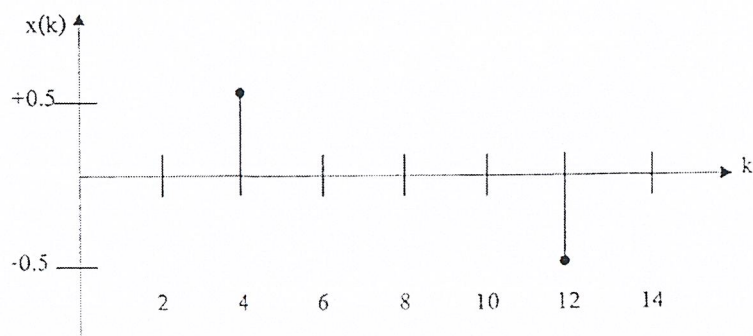
จากตัวอย่างมีค่าแอมพลิจูด “1” และสัญญาณเป็นค่าจริง ส่วนค่าจินตภาพเป็น “0” การสุ่มข้อมูลคู่ได้จากรูปที่ 2.43 ซึ่งมีจำนวนข้อมูล 16 จุด ตั้งแต่ $x_0 - x_{15}$



รูปที่ 2.43 อินพุต 16 จุด ฟาสต์ฟูเรียร์ทรานฟอร์ม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โดยค่าเอาต์พุตที่ได้จะโชว์ในรูปที่ 2.44 ซึ่งมีค่าอยู่ที่ $k = 4$ โดยมีขนาดแอมพลิจูด 0.5 และที่ $k = 12$ มีขนาดแอมพลิจูด -0.5



รูปที่ 2.44 เอาต์พุตของ 16 จุด ฟาสต์ฟูเรียร์ทรานฟอร์ม

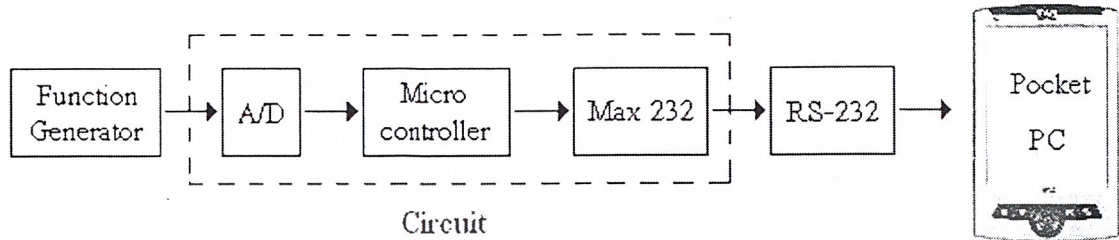


เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 3

การออกแบบและโครงสร้าง

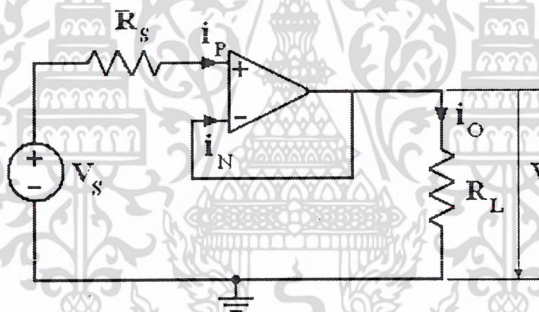
3.1 โครงสร้างโดยรวมของโครงการ



รูปที่ 3.1 แสดงโครงสร้างโดยรวมของโครงการ

3.2 วงจร (Circuit)

3.2.1 วงจรตามแรงดัน (Voltage Follower)



รูปที่ 3.2 วงจรตามแรงดัน

จากทฤษฎีของ Op-amp (Operation Amplifier) จะได้ว่ากระแสอินพุตของ op-amp มีค่าน้อยมาก

ซึ่งถือได้ว่า $i_p = i_n = 0$ ดังนั้นจะไม่เกิดแรงดันตกคร่อมที่ R_s

ดังนั้น $V_p = V_s$

และ $V_p = V_o$

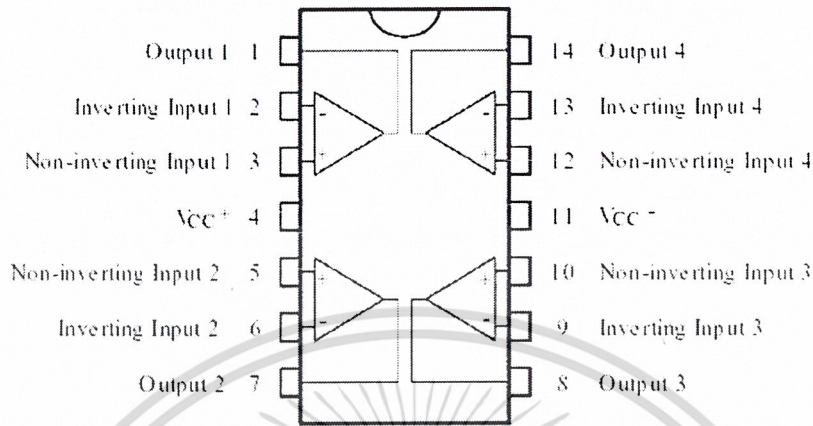
เพราะฉะนั้น $V_o = V_s$

ซึ่งแสดงผลให้แรงดันเอาต์พุต เท่ากับ แรงดันอินพุต หรือกล่าวอีกนัยหนึ่งว่า “แรงดันเอาต์พุตเป็นไปตามแรงดันอินพุต” ซึ่งเป็นที่มาของชื่อ “วงจรตามแรงดัน” (Voltage Follower)

วงจรตามแรงดันใช้สำหรับอินเทอร์เฟซระหว่างวงจร เพราะวงจรมีจะกั้นระหว่างอินพุตกับเอาต์พุต จึงอาจเรียกได้ว่าเป็นวงจรกันชน (Buffer Amplifier)

โดยเมื่อใช้วงจรตามแรงดันนี้แล้ว จะไม่ดึงกระแสจากวงจรแรกแต่สามารถจ่ายกระแสให้วงจรที่ 2 ได้
อย่างเพียงพอ

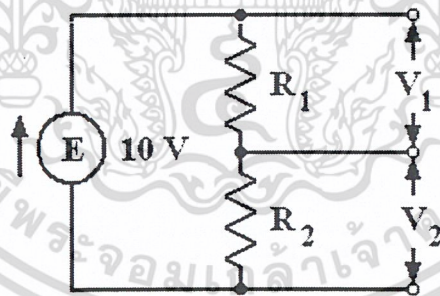
ซึ่งในโปรเจกต์นี้ได้ใช้ Buffer Amplifier เป็น IC เบอร์ LM324N ของบริษัท STMicroelectronics



รูปที่ 3.3 แสดงรูป IC เบอร์ LM324N

3.2.2 วงจรการแบ่งแรงดันที่ไม่มีโหลด

วงจรการแบ่งแรงดันที่ไม่มีโหลด (Unloaded Voltage Divider) เป็นวงจรแบบอนุกรม เราสามารถนำ
วงจรชนิดนี้ไปแบ่งแรงดันได้หลายค่า เพื่อนำไปจ่ายให้แก่โหลดที่ต้องการ ในระดับแรงดันต่างๆที่แต่ต่างกัน
ออกไป ซึ่งทั้งนี้แรงดันที่ถูกแบ่งทั้งหมดจะได้มาจากแหล่งกำเนิดแรงดันเพียงตัวเดียวเท่านั้น



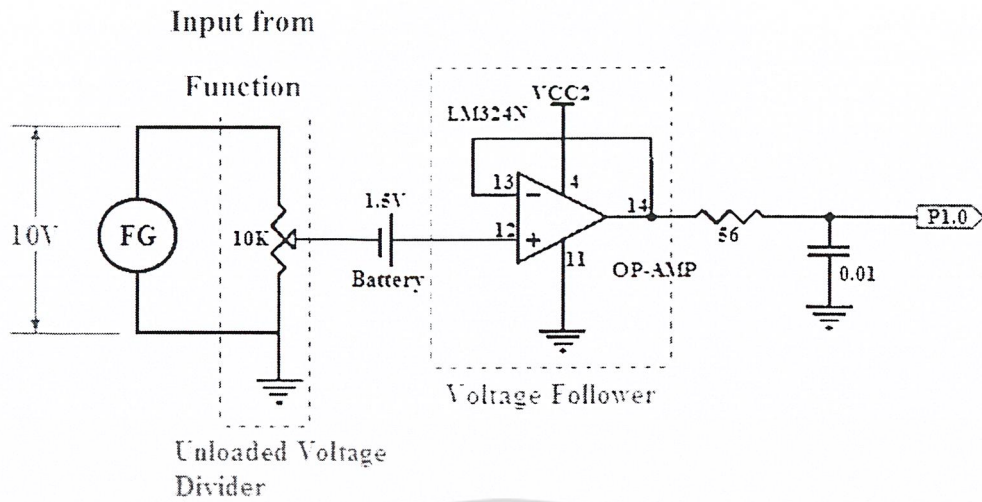
รูปที่ 3.4 วงจรการแบ่งแรงดันที่ไม่มีโหลด

จากรูปจะพิจารณาเห็นได้ว่า วงจรจะประกอบด้วยแหล่งกำเนิดแรงดัน E ซึ่งจ่ายแรงดันให้แก่ความ
ต้านทาน R_1 และ R_2 โดยความต้านทาน R_1 และ R_2 จะทำหน้าที่แบ่งแรงดันออกเป็นสองส่วนและทำให้ได้
แรงดันเอาต์พุตสองค่าคือ V_1 และ V_2 ซึ่งแรงดันเอาต์พุตทั้ง V_1 และ V_2 นี้มาจากแหล่งกำเนิดแรงดันตัว
เดียวกันคือ E และสามารถที่จะคำนวณหาค่าของแรงดันเอาต์พุต V_1 และ V_2 ได้ดังต่อไปนี้

$$\text{แรงดันเอาต์พุต} \quad V_1 = E \frac{R_1}{R_1 + R_2}$$

$$\text{แรงดันเอาต์พุต} \quad V_2 = E \frac{R_2}{R_1 + R_2}$$

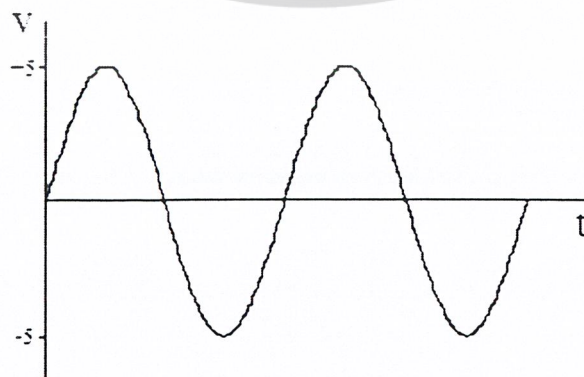
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.5 การเชื่อมต่อวงจร Unloaded Voltage Divider กับวงจร Voltage Follower เพื่อส่งสัญญาณเข้าไมโครคอนโทรลเลอร์

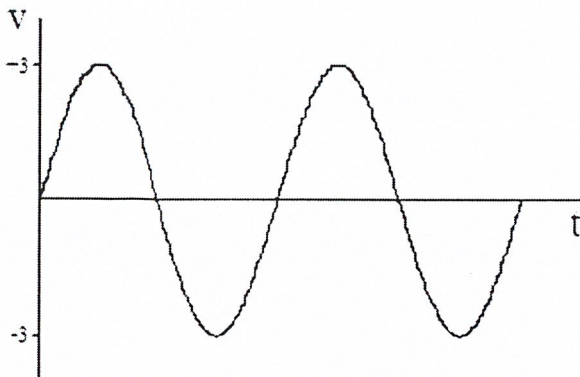
จากรูปเราจะทำการ Input สัญญาณที่สร้างจาก Function Generator ซึ่งมีระดับแรงดันที่ $\pm 5V$ ให้กับวงจร Unloaded Voltage Divider เพื่อทำการลดแรงดันจกระดับ $\pm 5V$ มาเป็น $\pm 1.5V$ โดยตัวต้านทานที่ใช้ นั้นได้เลือกใช้ตัวต้านทานชนิดปรับค่าได้ (Variable Resistor) เนื่องจากเราสามารถปรับค่าของแรงต้านทาน เพื่อให้ระดับแรงดันออกมาเท่ากับ $\pm 1.5V$ ในทุกกรณีที่ค่า output ที่ออกจาก Unloaded Voltage Divider มีค่าไม่เท่ากับ $\pm 1.5V$ อันเนื่องมาจากสาเหตุต่างๆ เช่น แรงดัน Drop ภายในวงจร เป็นต้น โดยที่เราไม่จำเป็นต้องแก้ไขวงจร

หลังจากนั้นแรงดันที่ได้รับลดระดับสัญญาณแล้วจะถูกส่งผ่านแหล่งจ่ายแรงดันกระแสตรง $1.5V$ เพื่อทำการปรับแรงดันไฟฟ้าให้กลายเป็น $0-3V$ เนื่องจาก Analog to Digital Converter (ADC) ที่ใช้นั้นสามารถทำงานได้ที่ระดับแรงดัน $0-3V$ เท่านั้น เมื่อปรับระดับเป็น $0-3V$ แล้วสัญญาณก็จะถูกส่งให้กับวงจร Voltage Follower ซึ่งวงจรดังกล่าวจะทำหน้าที่แปลงแรงดันให้กลายเป็นกระแสตามกฎของโอห์ม ($V = IR$) เพื่อจ่ายกระแสให้กับไมโครคอนโทรลเลอร์ นอกจากนี้ยังทำหน้าที่กันกระแสที่มากเกินไปก่อนเข้าไมโครคอนโทรลเลอร์เพื่อป้องกันความเสียหายที่จะเกิดขึ้นกับไมโครคอนโทรลเลอร์

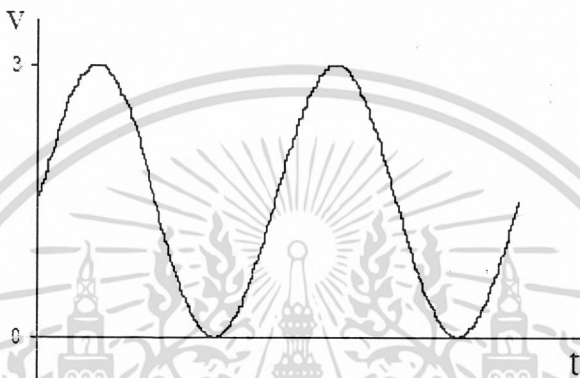


รูปที่ 3.6 ลักษณะกราฟของ v-t ที่ออกจาก Function Generator

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.7 ลักษณะกราฟของ v-t เมื่อผ่าน Voltage Follower



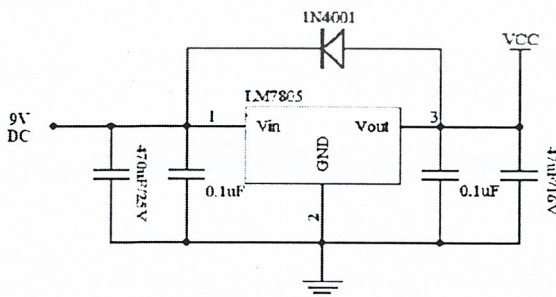
รูปที่ 3.8 ลักษณะกราฟของ v-t เมื่อผ่าน แหล่งจ่ายแรงดันกระแสตรง +1.5

3.2.3 วงจรแหล่งจ่ายแรงดัน

วงจรแหล่งจ่ายแรงดันทำหน้าที่แปลงจาก Battery ที่มีระดับแรงดันไฟฟ้า 9V ให้กลายเป็น 5V เพื่อจ่ายให้กับส่วนต่างๆของวงจรดังนี้ Max232, LM324, Microcontroller เบอร์ T89C51AC2 เนื่องจาก IC ทุกเบอร์ที่กล่าวมาข้างต้นนั้นจะทำงานได้ก็ต่อเมื่อจ่ายแรงดันให้ 5V

IC ที่ใช้ในวงจรแหล่งจ่ายแรงดัน คือ LM7805 ซึ่งมีคุณลักษณะดังนี้

- กระแสเอาต์พุตสูงถึง 1 A
- แรงดันเอาต์พุต 5 V
- ป้องกันการลัดวงจร



รูปที่ 3.9 วงจรแหล่งจ่ายแรงดัน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

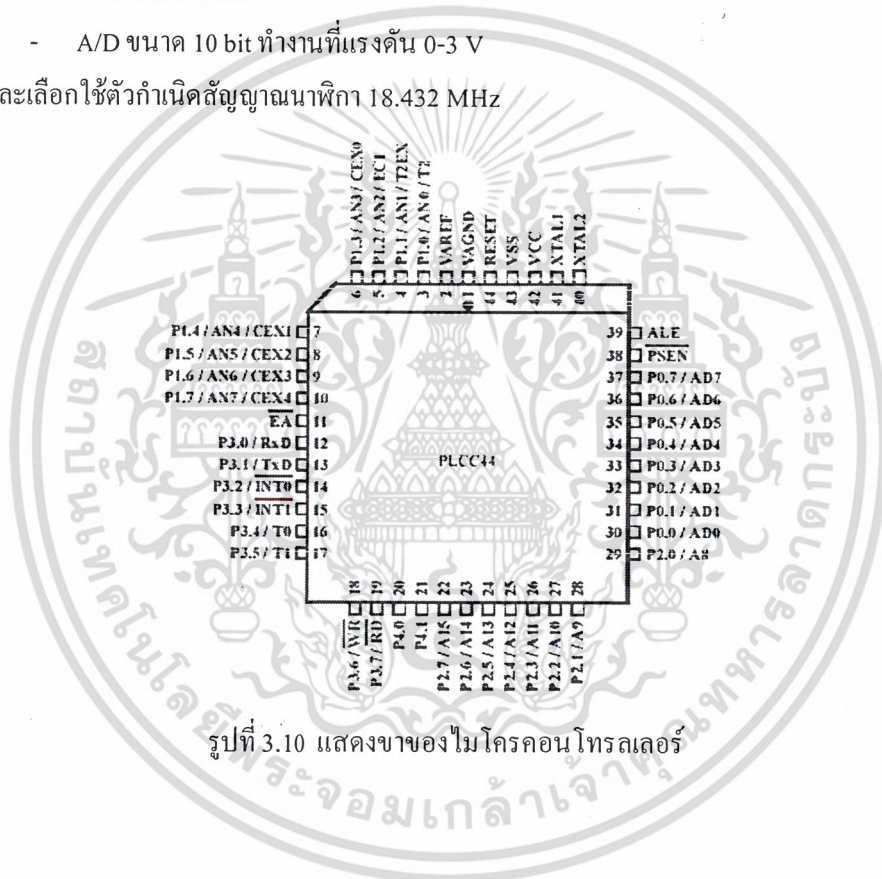
3.2.4 ไมโครคอนโทรลเลอร์

ไมโครคอนโทรลเลอร์ถูกนำมาใช้ในการควบคุมการรับและส่งค่า ที่ได้จาก Function Generator ไปยัง Pocket PC

ไมโครคอนโทรลเลอร์ที่ใช้เป็นเบอร์ T89C51AC2 โดยมีคุณสมบัติสำคัญดังนี้

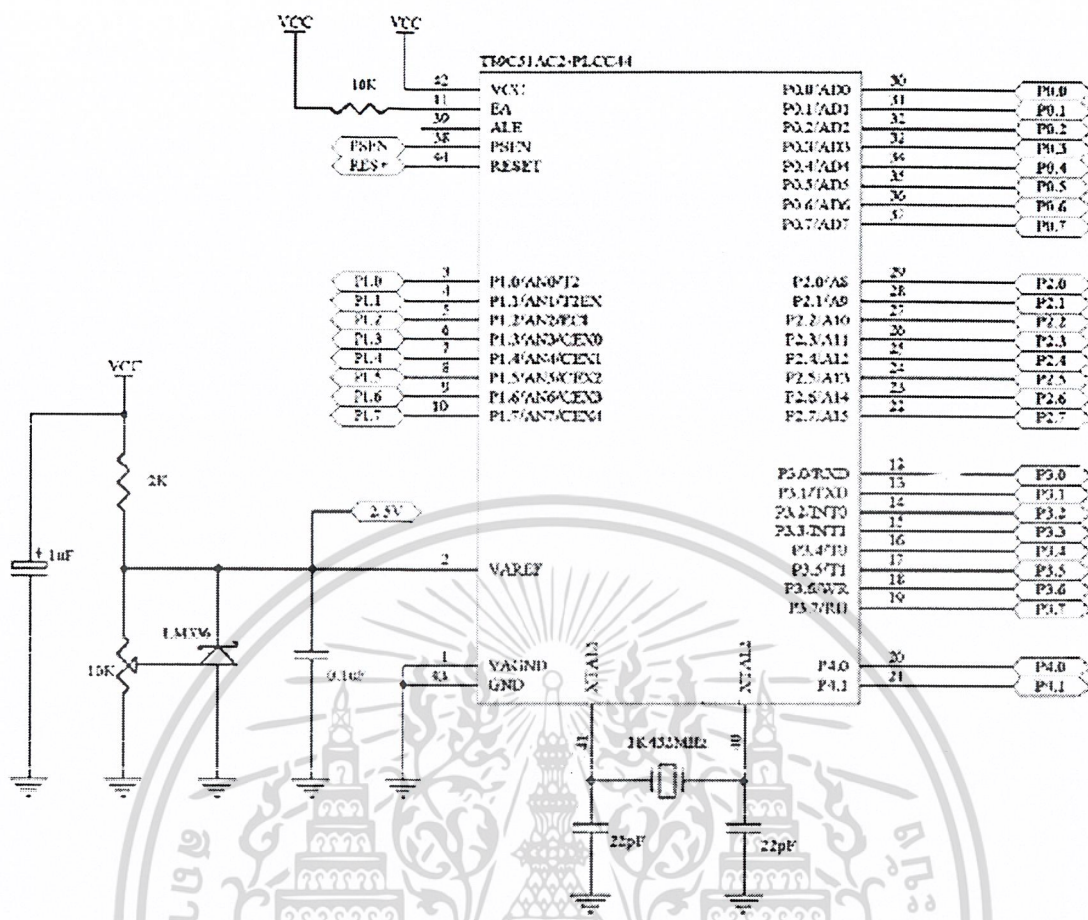
- สามารถดาวน์โหลดโปรแกรมบนตัวไมโครคอนโทรลเลอร์ผ่าน พอร์ต อนุกรมได้โดยไม่ต้องอาศัยเครื่อง Burn Program
- หน่วยความจำภายในประเภท RAM 256 byte
- หน่วยความจำ EEPROM 2 kbyte
- UART แบบ full duplex
- CPU ขนาด 8 bit
- A/D ขนาด 10 bit ทำงานที่แรงดัน 0-3 V

และเลือกใช้ตัวกำเนิดสัญญาณนาฬิกา 18.432 MHz



รูปที่ 3.10 แสดงขาของไมโครคอนโทรลเลอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.11 แสดงวงจรของไมโครคอนโทรลเลอร์

การต่อขาของไมโครคอนโทรลเลอร์เพื่อใช้งาน

- ขา Vcc (ขา 42) สำหรับต่อแหล่งจ่ายแรงดันไฟตรงขนาด 5 Volt
- ขา Vss/GND (ขา 20) สำหรับเปรียบเทียบแรงดันของไมโครคอนโทรลเลอร์
- ขา EA (ขา 11) เป็นขาสำหรับใช้เลือกให้ MCS-51 ทำงานจากโปรแกรมที่อยู่ภายในหรือภายนอกชิป โดยหากขานี้มีสถานะเป็น 0 หมายถึงให้ใช้โปรแกรมจากหน่วยความจำที่เก็บโปรแกรมจากภายนอก หากขานี้มีสถานะเป็น 1 หมายถึงบังคับให้ MCS-51 ใช้โปรแกรมจากหน่วยความจำสำหรับเก็บโปรแกรมภายในชิป และสำหรับ MCS-51 ที่มีหน่วยความจำสำหรับเก็บโปรแกรมภายในชิป สามารถเลือกให้ทำงานได้ทั้งจากโปรแกรมที่เก็บไว้ในหน่วยความจำภายนอกชิป ด้วยการต่อขา EA กับไฟเลี้ยงหรือ กราวด์ตามลำดับ ส่วนใน MCS-51 ที่ไม่มีหน่วยความจำสำหรับจัดเก็บโปรแกรมภายในชิป ให้ต่อขานี้ลงกราวด์เสมอ
- VAREF (ขา 2) เป็นขาสำหรับจ่ายแรงดันให้กับ ADC
- VAGND (ขา 1) เป็นขาที่ไว้สำหรับเปรียบเทียบแรงดันของ ADC

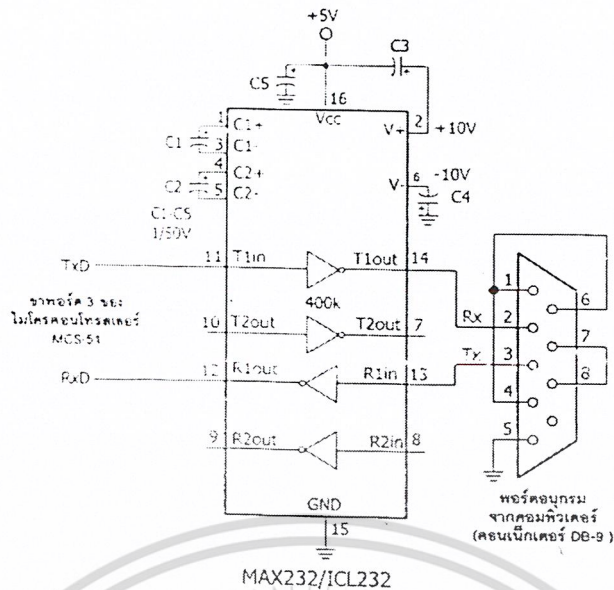
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- ขา RST (ขา 44) ใช้สำหรับการรีเซตวงจรทุกอย่างภายในชิป เพื่อเริ่มต้นการทำงานใหม่การรีเซตใช้เมื่อเริ่มจ่ายพลังงานหรือเมื่อโปรแกรมเกิดการทำงานผิดพลาด เมื่อต้องการรีเซตชิป MCS-51 ขานี้ต้องมีสถานะ 1 อย่างน้อย 2 Machine cycle ระหว่าง ที่ออสซิลเลเตอร์ยังทำงานอยู่ โดยต้องต่อตัวต้านทานค่า 8.2 กิโลโอห์มเพื่อทำหน้าที่พูลดาวน์ (รักษาระดับแรงดันไฟฟ้าให้มีสถานะเป็นกราวด์) และเพื่อให้ตัวชิปรีเซตเองเมื่อเริ่มจ่ายพลังงานให้ต่อตัวเก็บประจุขนาด 10 ไมโครฟารัด คร่อมระหว่างขา RST กับ Vcc
- ขา PSEN (ขา 38) ใช้ส่งสัญญาณสโตรบ เพื่ออ่านคำสั่งจากโปรแกรมที่เก็บไว้ในหน่วยความจำภายนอกชิป (Program strobe enable) เมื่อชิปทำงานด้วยโปรแกรมจากภายนอกชิปนี้จะส่งสัญญาณสโตรบ 2 ครั้งในแต่ละแมชชีน-ไซเคิล แต่ในช่วงการเขียนหรืออ่านข้อมูลกับหน่วยความจำภายนอกหรือเมื่อใช้โปรแกรมจากหน่วยความจำสำหรับเก็บโปรแกรมภายในชิป จะไม่มีสัญญาณออกมาจากขา
- ขา XTAL1 (ขา 41) ใช้ต่อคริสตัลภายนอก โดยเป็นอินพุตเข้าสู่วงจรรอสซิลเลเตอร์
- ขา XTAL2 (ขา 40) ใช้ต่อคริสตัลภายนอก โดยเป็นเอาต์พุตออกจากวงจรรอสซิลเลเตอร์
- Port P1.0 (ขา 3) ทำหน้าที่รับสัญญาณอินพุตจาก Function Generator
- Port P3.0 (ขา 12) ทำหน้าที่ส่งสัญญาณเอาต์พุตไปยัง Pocket PC
- Port P3.1 (ขา 13) ทำหน้าที่รับสัญญาณอินพุตจาก Pocket PC

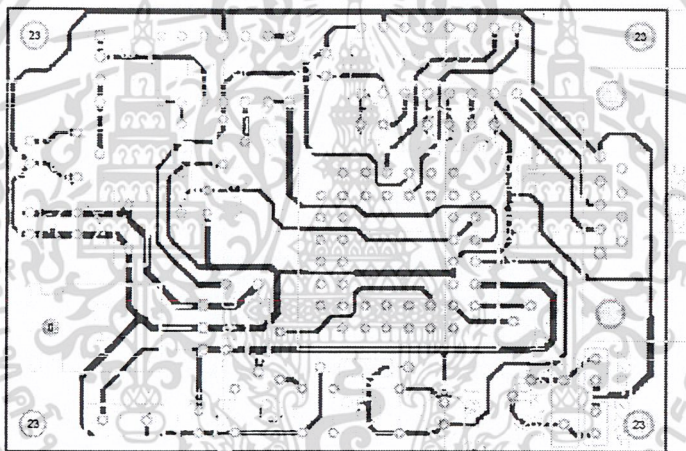
ไมโครคอนโทรลเลอร์นั้น มีหลักการทำงานดังนี้ คือจะทำการรับสัญญาณอนาล็อกจาก function generator ที่มีการปรับความดันให้เป็น 0-3 V แล้ว ผ่าน Port 1-0 จากนั้น A/D Converter ที่อยู่ในไมโครคอนโทรลเลอร์ จะทำการแปลงสัญญาณอนาล็อกให้กลายเป็นสัญญาณดิจิทัลแล้ว ไมโครคอนโทรลเลอร์ก็จะส่งสัญญาณที่ประมวลผลได้นี้ ออกไปยัง IC MAX232 โดยผ่านทาง Port 3.1 แล้วจึงส่งต่อไปยัง Pocket PC นอกจากนี้เรายังสามารถส่งข้อมูลจาก Pocket PC ให้กับไมโครคอนโทรลเลอร์ โดยผ่านทาง Port 3.1

3.2.5 การเชื่อมต่อกับพอร์ตอนุกรมของคอมพิวเตอร์

การใช้งานวงจรพอร์ตอนุกรมของไมโครคอนโทรลเลอร์ MCS-51 มักนิยมใช้ในการติดต่อเพื่อแลกเปลี่ยนข้อมูลกับคอมพิวเตอร์ผ่านทางอนุกรมในมาตรฐาน RS-232 เป็นส่วนใหญ่ แต่เนื่องจากระดับสัญญาณของพอร์ตอนุกรม RS-232 มีระดับ ± 3 ถึง ± 12 V ในขณะที่ระดับสัญญาณของไมโครคอนโทรลเลอร์ MCS-51 อยู่ในระดับ TTL ดังนั้นจึงไม่สามารถเชื่อมต่อพอร์ตอนุกรมของไมโครคอนโทรลเลอร์ MCS-51 เข้ากับพอร์ตอนุกรมของคอมพิวเตอร์ได้โดยตรง จึงต้องอาศัยการเชื่อมต่อผ่านไอซีที่ทำหน้าที่แปลงสัญญาณ ซึ่งต้องทำการแปลงข้อมูลส่งของไมโครคอนโทรลเลอร์ MCS-51 จากระดับ TTL ไปเป็นระดับ RS-232 และทำการแปลงข้อมูลรับของคอมพิวเตอร์จากระดับ RS-232 เป็นระดับ TTL ไอซีที่เป็นเบอร์ MAX232 ของ MAXIM ซึ่งแสดงไอซีและการเชื่อมต่อ ไว้ดังรูป

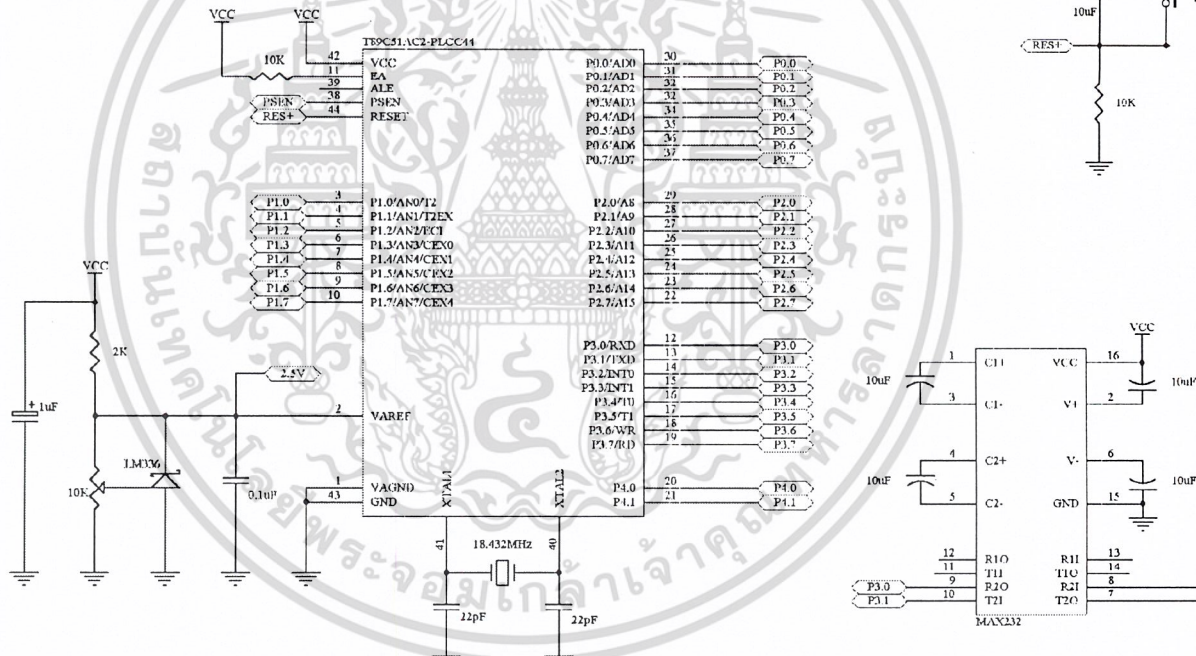
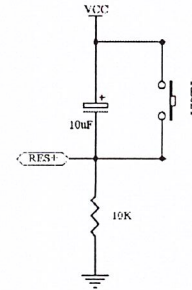
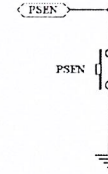
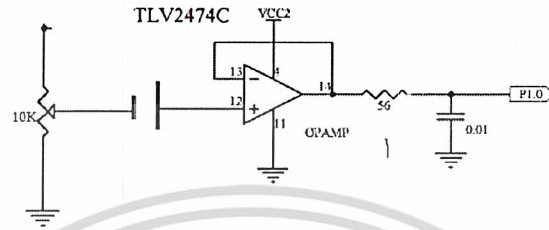
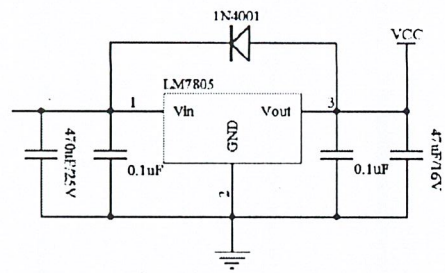


รูปที่ 3.12 วงจรเชื่อมต่อ MAX232 เข้ากับพอร์ตอนุกรมของคอมพิวเตอร์ และ MCS-51



รูปที่ 3.13 แสดงลายวงจรที่ออกแบบด้วยโปรแกรม Portel99

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
 ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.14 แสดงส่วนประกอบของวงจรทั้งหมด

3.3 Pocket PC

PDA (Personal Digital Assistant) เป็นอุปกรณ์คอมพิวเตอร์พกพาขนาดเล็กที่ช่วยอำนวยความสะดวกในการจดบันทึก, เก็บข้อมูล, เตือนเวลานัดหมาย หรือ จัดการงานต่างๆ ได้อย่างสะดวกและรวดเร็ว รวมไปถึงความสามารถของการเพิ่มเติมแอปพลิเคชันเพื่อให้ใช้งานด้านอื่นๆ ได้เหมาะสมกับความต้องการยิ่งขึ้น เช่น ดูเวลารอบโลก, อัตราแลกเปลี่ยนเงินตรา, คู่มือหนังสือพิมพ์ออนไลน์, บันทึกรายรับรายจ่าย แม้แต่ในเรื่องของมัลติมีเดียและเอนเตอร์เทน เช่น คู่มือ, ฟังเพลง, หรือ เล่นเกมส์ ก็สามารถรวมเข้าไปอยู่ในเจ้าอุปกรณ์เล็กๆ นี้ได้เช่นกัน PDA นั้นยังแยกออกมาได้อีกหลายประเภท ตามลักษณะของการใช้งานและระบบปฏิบัติการที่ถูกติดตั้งอยู่ในเครื่อง PDA นั้นๆ ซึ่งหลักๆ ที่เรารู้จักกันก็จะมี PDA ที่ใช้ระบบปฏิบัติการ Palm OS หรือที่เรียกว่า Palm และ PDA ที่ใช้ระบบปฏิบัติการ Microsoft Windows Mobile หรือที่เรียกกันว่า Pocket PC นั่นเอง

Pocket PC เป็นอุปกรณ์ที่ทำหน้าที่คำนวณและแสดงผลโดยรับสัญญาณดิจิทัลที่เป็น time domain จากไมโครคอนโทรลเลอร์เข้ามาผ่านทาง Serial Port (RS-232) แล้วทำการแปลงจาก time domain ให้กลายเป็น frequency domain โดยอาศัยคำนวณแบบ Fast Fourier Transform (FFT) จากนั้นนำค่าที่คำนวณได้ไปแสดงผลบนหน้าจอ Pocket PC ในรูปแบบของ Graph Spectrum

3.3.1 ข้อเปรียบเทียบระหว่าง Palm และ Pocket PC

- **Palm** เป็น PDA ที่ใช้ระบบปฏิบัติการ Palm Operation System Platform

ข้อดี

1. ใช้พื้นที่ในการเก็บข้อมูลไม่มาก เนื่องจากโปรแกรมมีขนาดเล็ก
2. มีการทำงานที่รวดเร็ว ไม่ต้องเสียเวลาประมวลผลนาน เนื่องจากโปรแกรมมีความซับซ้อนน้อย
3. มีอัตราการสิ้นเปลืองพลังงานต่ำ
4. ระบบปฏิบัติการมีความเสถียรสูง

ข้อเสีย

1. การทำงานด้าน Multimedia และสียังไม่เทียบ Pocket PC ไม่ได้

- **Pocket PC** เป็น PDA ที่ใช้ระบบปฏิบัติการ Microsoft Windows Mobile™

ข้อดี

1. มีรูปแบบการใช้งานที่คล้ายคลึงกับระบบปฏิบัติการ Microsoft Window บนเครื่อง PC
2. รองรับการใช้งานด้าน Multimedia อย่างครบถ้วน หน้าจอมีความละเอียดของสีสูง และสมจริง

ข้อเสีย

1. โปรแกรมมีขนาดใหญ่ จึงต้องใช้หน่วยความจำมากกว่า
2. มีการประมวลผลที่ช้าเนื่องจากโปรแกรมมีความซับซ้อน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

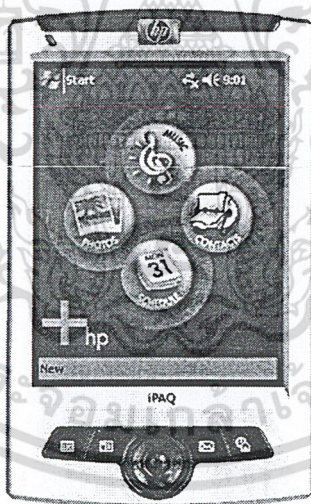
3. มีอัตราการสิ้นเปลืองพลังงานค่อนข้างสูง
4. ระบบปฏิบัติการยังไม่ค่อยเสถียร จึงพบปัญหา Hang บ่อยครั้ง

3.3.2 สาเหตุที่เลือกใช้ Pocket PC

เนื่องจาก Pocket PC ใช้ระบบปฏิบัติการ Microsoft Windows Mobile™ ซึ่งมีลักษณะการใช้งานที่คล้ายคลึงกับ ระบบปฏิบัติการ Microsoft Windows บน PC และมีแนวโน้มว่าผู้คนในประเทศไทยจะหันมาใช้งานเพิ่มมากขึ้น นอกจากนี้ยังสามารถหาหนังสือสอนวิธีใช้โปรแกรมที่จะนำมาพัฒนา software บน Pocket PC ได้ง่ายกว่า Palm ตัวอย่างโปรแกรมที่ใช้พัฒนา software บน Pocket PC เช่น Visual C++, Visual C#, Visual Basic เป็นต้น

เครื่อง Pocket PC ที่เลือกใช้ใน โครงการงานนี้คือ รุ่น HP iPAQ rz 1710 ซึ่งมีคุณสมบัติที่สำคัญดังนี้

- CPU 203 MHz processor
- 32 MB SDRAM, 32 MB Flash ROM memory
- SD/MMC memory card slot
- ระบบปฏิบัติการ Microsoft Windows Mobile™
- รองรับ IrDA, USB และ Serial Port
- Sync Connector รองรับ RS-232 เต็มรูปแบบ



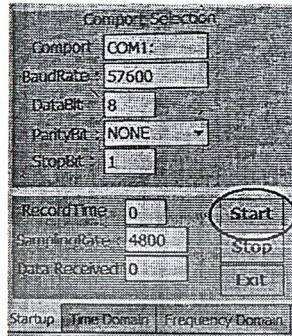
รูปที่ 3.15 Pocket PC รุ่น iPAQ 1710

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.4 ขั้นตอนการทำงาน

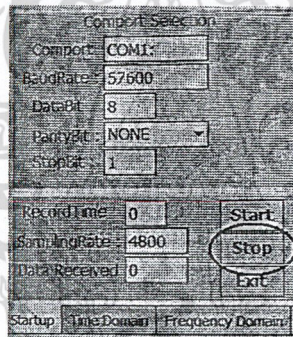
ขั้นตอนการทำงานมีดังนี้ คือ

1. รับสัญญาณอินพุตจากเครื่องกำเนิดสัญญาณหรือฟังก์ชันเจนเนอเรเตอร์ โดยการกดปุ่ม start



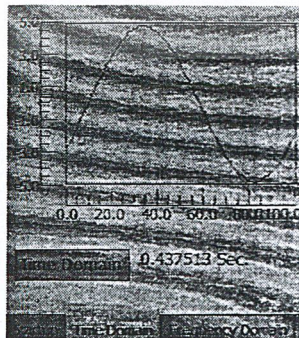
รูปที่ 3.16 แสดงการรับสัญญาณอินพุต

2. หยุดการรับสัญญาณอินพุตด้วยการกดปุ่ม stop



รูปที่ 3.17 แสดงการหยุดรับสัญญาณอินพุต

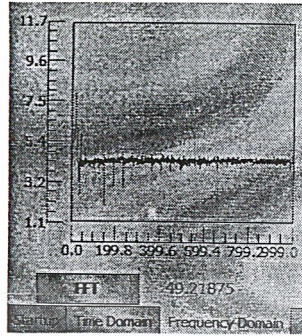
3. กดปุ่ม Time Domain เพื่อแสดงกราฟของสัญญาณอินพุต (Time Domain)



รูปที่ 3.18 แสดงกราฟ Time Domain

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4. กดปุ่ม Frequency Domain เพื่อทำการคำนวณหาค่าความถี่สูงสุดและแสดงกราฟสเปกตรัม



รูปที่ 3.19 แสดงกราฟ Spectrum



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 4

การทดลองและผลการทดลอง

การทดลองทั้งหมดจะแบ่งออกเป็น 2 ส่วนคือ การวัด Sampling Rate ของวงจร และการทดลองการวัดสัญญาณบนโดเมนของความถี่ โดยจะแสดงผลบนหน้าจอ ฟ็ोकเก็ท พีซี

การทดลองที่ 1 การวัด Sampling Rate ของวงจร

เป็นการวัด Sampling Rate ของวงจร เพื่อที่จะนำ ค่า Sampling Rate ที่ได้ไป ออกแบบ Software อุปกรณ์ที่ใช้ในการทดลอง

1. วงจร
2. Function Generator
3. Oscilloscope

วิธีการทดลอง

1. เชื่อมต่อวงจรกับ Function Generator
2. ทำการ Generate สัญญาณ Sine Wave ที่มีระดับแรงดัน $\pm 5\text{ V}$ ให้กับวงจร
3. นำ Oscilloscope ไปวัดที่ ขาที่ 3 ของ RS-232
4. ทำการอ่าน และบันทึกเวลาของสัญญาณใน 1 คาบ

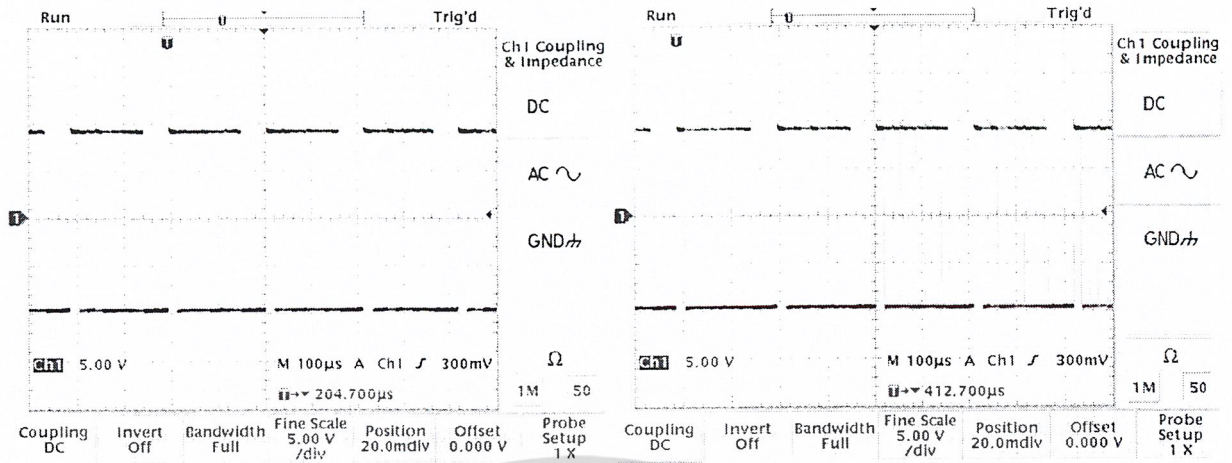
ผลการทดลอง

การทดลองครั้งที่	เวลาที่ใช้ต่อ 1 คาบ	อัตรา Sampling rate
1	208 μs	4807.6920
2	210 μs	4761.9047
3	208 μs	4807.6920
4	208 μs	4807.6920
5	208 μs	4807.6920
6	208 μs	4807.6920
7	208 μs	4807.6920
8	210 μs	4761.9047
9	208 μs	4807.6920
10	208 μs	4807.6920

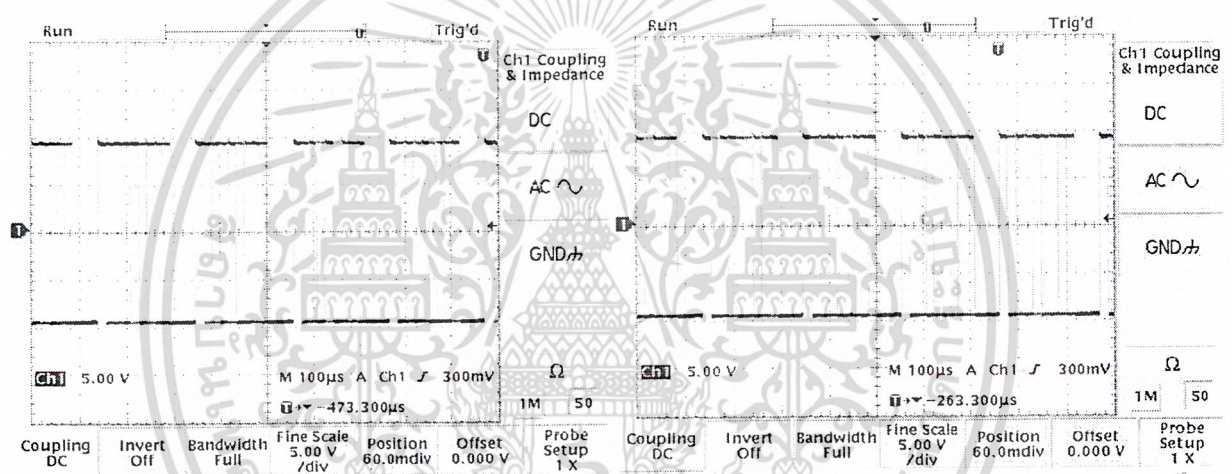
ตารางที่ 4.1 ตารางผลการทดลองการวัด sampling rate ของวงจร

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โดยแสดงระยะเวลาที่ใช้ต่อสัญญาณ 1 คาบ ดังรูปที่ 4.1 และ 4.2



รูปที่ 4.1 แสดงระยะเวลาที่ใช้ต่อสัญญาณ 1 คาบ เท่ากับ 208 μs



รูปที่ 4.2 แสดงระยะเวลาที่ใช้ต่อสัญญาณ 1 คาบ เท่ากับ 210 μs

วิเคราะห์ผลการทดลอง

อัตรา sampling rate ที่นับได้จากเครื่องออสซิโลสโคป นั้นมีค่าเท่ากับจำนวนไบต์ที่นับได้ โดยใช้ เวลา 1 วินาที ซึ่งมีวิธีคำนวณดังนี้

$$SR = \frac{1}{T}$$

เมื่อ SR = sampling rate มีหน่วยเป็นไบต์ต่อวินาที

T = เวลาที่ใช้ต่อ 1 คาบ

จากผลการทดลอง เราจะนำค่า sampling rate ในแต่ละครั้งที่ได้จากการทดลองนั้นมาทำการเฉลี่ยเพื่อ หาค่า sampling rate ที่จะนำไปใช้ในการออกแบบโปรแกรม

โดย
$$SR_{avg} = \frac{\sum(SR)}{n}$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เมื่อ (SR)_{avg} = อัตรา sampling rate เฉลี่ย มีหน่วยเป็นไบต์ต่อวินาที

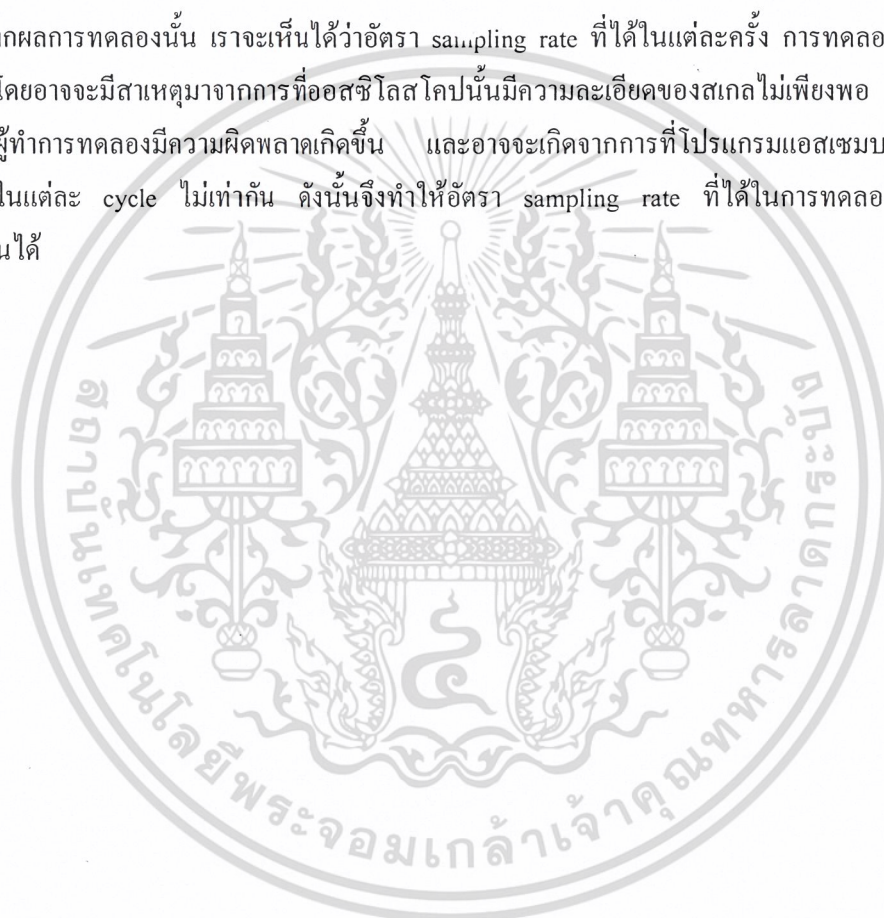
SR = อัตรา sampling rate ในแต่ละครั้งในการทดลอง มีหน่วยเป็นไบต์ต่อวินาที

n = จำนวนครั้งที่ใช้ในการทดลอง

ซึ่งทำการคำนวณออกมาแล้วจะได้ค่า sampling rate เฉลี่ย = 4798.53 ดังนั้นมีค่าประมาณ 4800 ไบต์ต่อวินาที โดยเราจะนำค่า sampling rate = 4800 ไบต์ต่อวินาที นั้นไปทำการออกแบบโปรแกรมต่อไป

วิจารณ์ผลการทดลอง

จากผลการทดลองนั้น เราจะเห็นได้ว่าอัตรา sampling rate ที่ได้ในแต่ละครั้ง การทดลองนั้นมีค่าไม่เท่ากัน โดยอาจจะมีสาเหตุมาจากการที่ออสซิลโลสโคปนั้นมีความละเอียดของสเกลไม่เพียงพอ ทำให้การสังเกตของผู้ทำการทดลองมีความผิดพลาดเกิดขึ้น และอาจจะเกิดจากการที่โปรแกรมแอสเซมบลีใช้เวลาในการทำงานในแต่ละ cycle ไม่เท่ากัน ดังนั้นจึงทำให้อัตรา sampling rate ที่ได้ในการทดลองอาจจะมีค่าคลาดเคลื่อนได้



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การทดลองที่ 2 ทดสอบค่าเปอร์เซ็นต์ error ของค่าความถี่ที่ Pocket PC วิเคราะห์ได้

เป็นการทดสอบค่าเปอร์เซ็นต์ error ของค่าความถี่ที่ Pocket PC วิเคราะห์ได้ โดยอาศัยการเปรียบเทียบกับความถี่ที่ทำการ Generate เข้าไป

อุปกรณ์ที่ใช้ในการทดลอง

1. วงจร
2. Function Generator
3. Pocket PC

วิธีการทดลอง

1. ทำการเชื่อมต่อ Pocket PC เข้ากับวงจร และเชื่อมต่อวงจรเข้ากับ Function Generator
2. ทำการ Generate สัญญาณ Sine Wave ที่มีระดับแรงดัน ± 5 V ให้กับวงจรโดยกำหนดค่าความถี่ตามที่ต้องการทดสอบ
3. ทำการอ่าน และบันทึกค่าความถี่ที่ Pocket PC วิเคราะห์ได้

ผลการทดลอง

ความถี่สัญญาณอินพุต (เฮิร์ต)	จำนวนไบต์	ความถี่สัญญาณเอาต์พุต (เฮิร์ต)	ค่าเปอร์เซ็นต์error
20	2435	21.0937	5.4687
	4325	19.9218	0.3906
	8354	19.9218	0.3906
30	2446	30.4687	1.5625
	4452	30.4687	1.5625
	8254	30.4687	1.5625
40	2345	42.1875	5.4687
	4352	39.8437	0.3906
	8534	39.8437	0.3906
50	2634	50.3906	0.7812
	4275	49.8046	0.3906
	8628	49.8046	0.3906
100	2354	100.7812	1.0937
	4721	99.6093	0.3906
	8936	99.6093	0.3906
150	2344	150	0
	4325	150	0
	8542	150	0

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

250	2344	250.7812	0.3124
	4235	250.7812	0.3124
	8525	250.7812	0.3124
500	2435	499.2187	0.1562
	4525	500.0936	0.0187
	8403	500.0936	0.0187
750	2654	750	0
	4523	750	0
	8542	750	0
1000	3827	1000.7810	0.0781
	5031	999.6094	0.0391
	9325	999.6094	0.0391

ตารางที่ 4.2 ตารางผลการทดลองการทดสอบค่าเปอร์เซ็นต์ error ของค่าความถี่ที่ Pocket PC วิเคราะห์ได้

วิเคราะห์ผลการทดลอง

จากผลการทดลองค่าเปอร์เซ็นต์ error ที่คำนวณได้นั้นเกิดจากการนำเอาค่าความถี่ของสัญญาณอินพุตที่ส่งออกจากเครื่องกำเนิดสัญญาณ ไปเปรียบเทียบกับค่าความถี่ของสัญญาณเอาต์พุตที่แสดงบนเครื่องพีคเก็ต พีซี ซึ่งมีวิธีการคำนวณดังนี้

$$\text{ค่าเปอร์เซ็นต์ error} = \left| \left(\frac{OP - IP}{IP} \right) * 100 \right|$$

เมื่อ IP = ค่าความถี่ของสัญญาณอินพุต มีหน่วยเป็นเฮิรตซ์
 SR = ค่าความถี่ของสัญญาณเอาต์พุต มีหน่วยเป็นเฮิรตซ์

วิจารณ์ผลการทดลอง

จากผลการทดลองจะเห็นว่าค่าเปอร์เซ็นต์ error ของการทดลองที่ใช้เวลาในการรับสัญญาณอินพุตมากกว่าจะมีค่าเปอร์เซ็นต์ error น้อยกว่า เนื่องจากมีการนำสัญญาณอินพุตที่มีจำนวนไบต์มากกว่าไปทำการคำนวณฟูเรียร์อย่างรวดเร็ว ซึ่งถ้ายังมีจำนวนไบต์ของสัญญาณอินพุตมากก็จะเป็นผลที่ทำให้มีการใช้จำนวนจุดของการแปลงฟูเรียร์อย่างรวดเร็วมาก ดังนั้นจึงเป็นผลให้มีค่าเปอร์เซ็นต์ error น้อย

แต่ยังมีสาเหตุที่ทำให้เกิดค่าเปอร์เซ็นต์ error นั้นก็คือ sampling rate ที่คำนวณได้จากการทดลองที่ 1 sampling rate ที่นำไปใช้งานนั้นเป็นการนำเอา sampling rate ของทุกๆการทดลองในการทดลองที่ 1 ไปทำการเฉลี่ยซึ่งอาจจะเป็นผลให้เกิดค่าเปอร์เซ็นต์ error

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 5

สรุปผลการทดลอง

5.1 สรุปผลการทดลอง

จากการออกแบบและพัฒนา เครื่องมือวัดโดยใช้เครื่องฟ็อกเก็ต พีซี ได้ถูกออกแบบขึ้นโดยมีหลักการคือ รับสัญญาณอินพุตจากเครื่องกำเนิดสัญญาณหรือเครื่องฟังก์ชันเจเนอเรเตอร์ โดยสัญญาณอินพุตจะถูกส่งเข้าไปในส่วนของวงจรประมวลผล เพื่อทำการแปลงสัญญาณอนาลอกให้กลายเป็นสัญญาณดิจิทัล จากนั้นสัญญาณที่ถูกแปลงเป็นดิจิทัลนั้นจะถูกส่งเข้าเครื่องฟ็อกเก็ต พีซี ผ่านทางพอร์ตอนุกรม (อาร์เอต-232) เพื่อทำการแปลงสัญญาณที่อยู่ในรูปแอมพลิจูดเชิงเวลาให้อยู่ในรูปของแอมพลิจูดเชิงความถี่ โดยใช้วิธีคำนวณฟูริเยร์อย่างรวดเร็ว จากนั้นผลที่คำนวณได้จะถูกนำมาแสดงผลบนหน้าจอของเครื่องฟ็อกเก็ต พีซี ในรูปของกราฟสเปกตรัม พร้อมทั้งบอกค่าความถี่สูงสุด ซึ่งเครื่องวิเคราะห์การสั่นสะเทือนนี้ สามารถวัดความถี่ได้ตั้งแต่ 20 ถึง 1000 เฮิร์ต ซึ่งถือว่าครอบคลุมเกือบทุกย่านความถี่ของการสั่นสะเทือนของเครื่องจักรกลทั่วไป โดยที่ข้อแตกต่างระหว่างเครื่องวิเคราะห์การสั่นสะเทือนของโครงการนี้กับเครื่องวิเคราะห์การสั่นสะเทือนที่มีขายอยู่ทั่วไป เป็นดังตารางที่ 5.1

	เครื่องวิเคราะห์การสั่นสะเทือน ของโครงการนี้	เครื่องวิเคราะห์การสั่นสะเทือน ที่มีอยู่ทั่วไป
ขนาด	มีขนาดเล็กกระทัดรัดสามารถพกพาได้	มีขนาดใหญ่ไม่สามารถพกพาได้
ราคา	มีราคาถูก	มีราคาค่อนข้างแพง
ความหลากหลายในการใช้งาน	สามารถนำไปประยุกต์ใช้เป็นเครื่องมือวัดค่าอื่นๆได้	ไม่สามารถนำไปประยุกต์ใช้เป็นเครื่องมือวัดค่าอื่นๆได้

ตารางที่ 5.1 ข้อแตกต่างระหว่างเครื่องวิเคราะห์การสั่นสะเทือนของโครงการนี้กับที่มีอยู่ทั่วไป

5.2 ปัญหา

1. การทำงานของโปรแกรมบน Pocket PC ไม่สามารถทำงานได้เร็วเท่าที่ควร เนื่องจากโปรแกรมที่ใช้ในการทำงานบน Pocket PC นั้นมีข้อจำกัดในการรับและส่งข้อมูล

2. สัญญาณแอมพลิจูดสามารถใช้ได้ในช่วงความกว้างจำกัด โดยไม่สามารถปรับค่าจากโปรแกรมได้เนื่องจากมีข้อจำกัดทางด้านฮาร์ดแวร์

3. มีการเกิดสัญญาณรบกวน (noise) ในวงจรประมวลผล

4. สัญญาณรูปคลื่นที่ถูกส่งออกจากเครื่องกำเนิดสัญญาณนั้น มีค่า error เล็กน้อย เนื่องจากเป็น

เอกสารที่ได้ออกสารซึ่งมอบไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5.3 ข้อเสนอแนะ

1. ควรใช้ไอซี A/D ที่มีประสิทธิภาพสูงขึ้น
2. ความพัฒนาโปรแกรมให้สามารถใช้งานได้เร็วขึ้น
3. ความยาวของสายส่งสัญญาณ ไม่ควรยาวเกินไป
4. ควรพัฒนาให้สามารถใช้งานได้ในช่วงความถี่มากขึ้น
5. ควรพัฒนาโปรแกรมให้สามารถใช้งานได้หลากหลายมากขึ้น



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 6

Application การใช้งาน

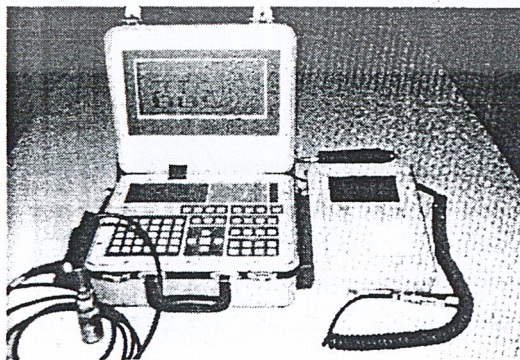
6.1 การสั่นสะเทือน

การสั่นสะเทือน คือ ปรากฏการณ์ของการเคลื่อนที่กลับไปกลับมาของวัตถุภายใต้แรงกระทำ ซึ่งอาจเป็นแรงภายในหรือภายนอกก็ได้ โดยทั่วไปแล้วการสั่นสะเทือนเป็นสิ่งที่ไม่ต้องการ แต่มักหลีกเลี่ยงไม่ได้ อดีที่สุดก็คือการพยายามจำกัดขนาดของการสั่นสะเทือนให้อยู่ในขอบเขตที่ยอมรับได้ สำหรับเครื่องจักรแล้วสาเหตุของการสั่นสะเทือนมีอยู่หลายสาเหตุ การไม่สมดุลของเครื่องจักรที่หมุนนั้นก็จะเป็นส่วนสาเหตุใหญ่สาเหตุหนึ่งที่ทำให้เกิดการสั่นสะเทือน การสั่นสะเทือนนี้อาจเกิดได้จากการที่ชิ้นส่วนเครื่องจักรเคลื่อนที่ตามแนวเส้นตรงด้วยความเร่ง ชิ้นส่วนเครื่องจักรสองชิ้นที่สั่นหรือถูกกันก็เป็นอีกสาเหตุหนึ่งของการสั่นสะเทือน ซึ่งการสั่นสะเทือนนี้เกิดจากความไม่เรียบของผิวหน้าของชิ้นส่วนสองชิ้นที่สัมผัสกัน การหลุดหลวมของชิ้นส่วนของเครื่องจักรที่เข้าคู่กันจะก่อให้เกิดการกระแทก ซึ่งก่อให้เกิดการสั่นสะเทือนเช่นกัน การสั่นสะเทือนเหล่านี้ของเครื่องจักรก็มีผลต่อสมรรถนะและสภาพการใช้งานของเครื่องจักร ดังนั้นในการทำให้เครื่องจักรทำงานที่มีประสิทธิภาพสูงสุดนั้นจึงจำเป็นต้องมีความเข้าใจถึงพฤติกรรมของการสั่นสะเทือนเพื่อจะจะสามารถจำกัดขนาดการสั่นสะเทือนให้มีค่าน้อยที่สุดเท่าที่จะทำได้

6.2 เครื่องวิเคราะห์การสั่นสะเทือน (Vibration Analyzer)

เครื่องวิเคราะห์การสั่นสะเทือน (Vibration Analyzer) เป็นเครื่องมือที่สามารถวิเคราะห์การสั่นสะเทือนได้อย่างมีประสิทธิภาพโดยเครื่องชนิดนี้จะมีหลักการทำงาน คือ จะรับสัญญาณการสั่นสะเทือนที่อยู่ในรูปโดเมนของเวลาจาก Sensor ซึ่งโดยมากจะเป็นหัววัด Accelerometer แล้วเครื่องจะทำการแปลงสัญญาณที่อยู่ในรูปโดเมนของเวลานั้นให้กลายเป็นโดเมนของความถี่ โดยอาศัยการคำนวณแบบ Fast Fourier Transform (FFT) จากนั้นก็นำค่าที่ได้มาแสดงเป็นกราฟสเปกตรัม

เครื่องวิเคราะห์การสั่นสะเทือนชนิดนี้สามารถเรียกได้หลายอย่าง เช่น Vibration Analyzer, Spectrum Analyzer, FFT Analyzer เป็นต้น



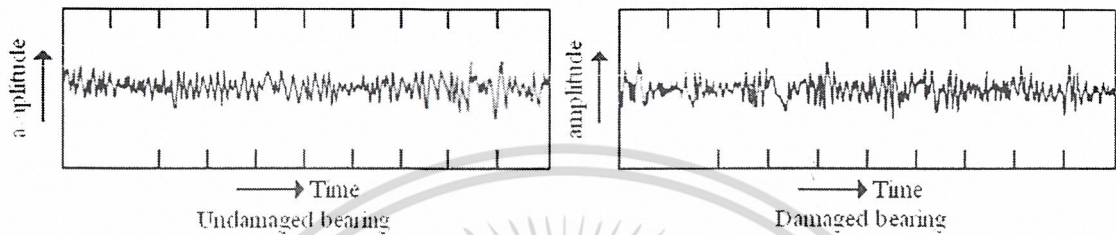
รูปที่ 6.1 Vibration Analyzer

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

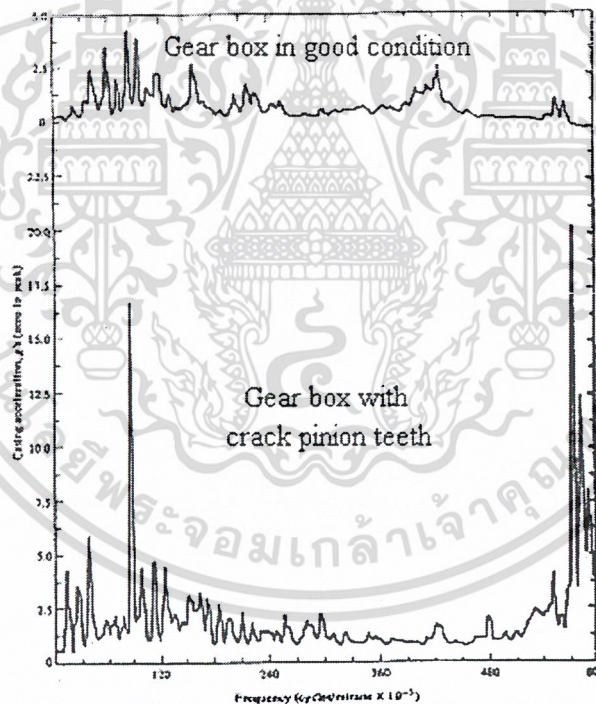
6.3 การวิเคราะห์แบบสเปกตรัม

การวิเคราะห์แบบสเปกตรัม คือ การวิเคราะห์สัญญาณบน โดเมนของความถี่ ซึ่งการวิเคราะห์สัญญาณนี้หากนำมาใช้กับเครื่องจักร จะให้ข้อมูลการสันสะเทือนได้มากกว่าข้อมูลจากการวิเคราะห์สัญญาณ โดเมนของเวลา

เมื่อเรานำแถบของสัญญาณของโดเมนของความถี่ หรือสเปกตรัมมาทำการวิเคราะห์ แนวโน้มของสเปกตรัมจะทำให้เรารู้ถึงสาเหตุที่ทำให้เกิดการสันสะเทือนของเครื่องจักร ดังตัวอย่างต่อไปนี้

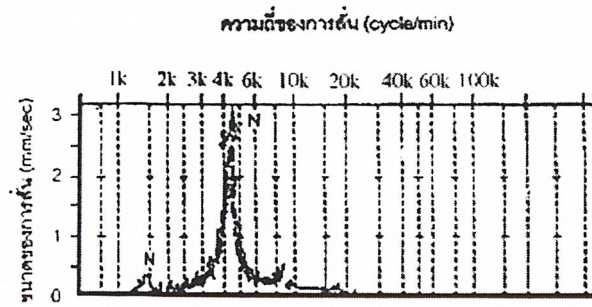


รูปที่ 6.2 สัญญาณที่อยู่ในรูปโดเมนของเวลา

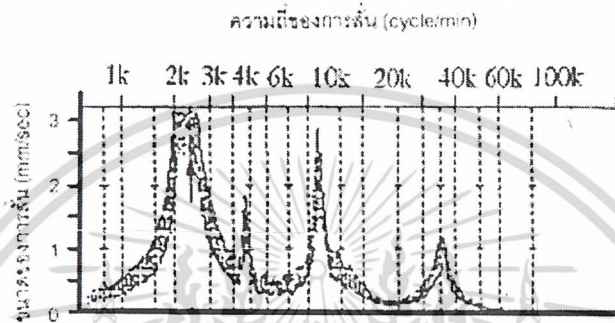


รูปที่ 6.3 สัญญาณที่อยู่ในรูปโดเมนของความถี่

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 6.4 แสดงแถบความถี่ของเครื่องจักรที่มีความเสียหายของชิ้นงานที่เชื่อมศูนย์



รูปที่ 6.5 แสดงแถบความถี่ของเครื่องจักรที่มีความเสียหายของชิ้นงานที่ไม่สมดุล

จากรูปที่ 6.2 จะเห็นได้ว่ากราฟที่อยู่ในรูปของโดเมนเวลานั้น เราจะนำมาวิเคราะห์หาสาเหตุของการสั่นสะเทือนได้ยาก เนื่องจากลักษณะของกราฟค่อนข้างใกล้เคียงกัน ไม่มีจุดที่เด่นชัด แต่เมื่อเราดูจากรูปที่ 6.3 ซึ่งเป็นกราฟสเปกตรัมนั้นเราจะเห็นได้ว่าความแตกต่างของ Gear box ที่ไม่มีความเสียหายกับ Gear box ที่มีการแตกหักของ Pinion Teeth นั้นถูกเห็นได้อย่างชัดเจน โดยลักษณะของกราฟสเปกตรัมของ Gear box ที่มีการแตกหักของ Pinion Teeth นั้นจะมีลักษณะเฉพาะตัวดังที่ได้แสดงไว้ในรูป หรือกล่าวอีกนัยหนึ่งว่า ถ้าลักษณะของกราฟเป็นดังรูปที่ 6.3 ก็แสดงว่าเกิดความเสียหายที่ Pinion Teeth ของ Gear box

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



ภาคผนวก ก.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

MC78XX/LM78XX/MC78XXA

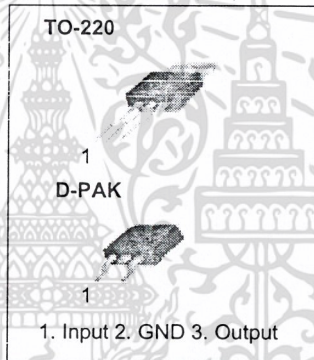
3-Terminal 1A Positive Voltage Regulator

Features

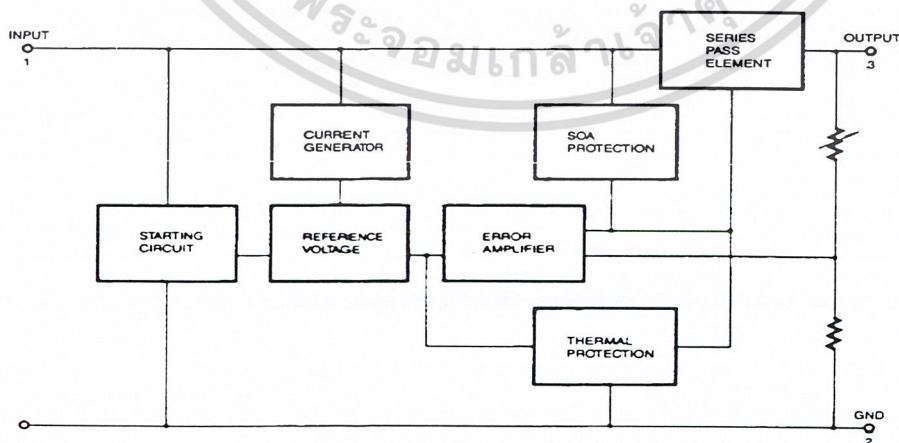
- Output Current up to 1A
- Output Voltages of 5, 6, 8, 9, 10, 12, 15, 18, 24V
- Thermal Overload Protection
- Short Circuit Protection
- Output Transistor Safe Operating Area Protection

Description

The MC78XX/LM78XX/MC78XXA series of three terminal positive regulators are available in the TO-220/D-PAK package and with several fixed output voltages, making them useful in a wide range of applications. Each type employs internal current limiting, thermal shut down and safe operating area protection, making it essentially indestructible. If adequate heat sinking is provided, they can deliver over 1A output current. Although designed primarily as fixed voltage regulators, these devices can be used with external components to obtain adjustable voltages and currents.



Internal Block Diagram



Rev. 1.0.1

Absolute Maximum Ratings

Parameter	Symbol	Value	Unit
Input Voltage (for $V_O = 5V$ to $18V$) (for $V_O = 24V$)	V_I	35	V
	V_{I1}	40	V
Thermal Resistance Junction-Cases (TO-220)	$R_{\theta JC}$	5	$^{\circ}C/W$
Thermal Resistance Junction-Air (TO-220)	$R_{\theta JA}$	65	$^{\circ}C/W$
Operating Temperature Range	T_{OPR}	0 ~ +125	$^{\circ}C$
Storage Temperature Range	T_{STG}	-65 ~ +150	$^{\circ}C$

Electrical Characteristics (MC7805/LM7805)

(Refer to test circuit, $0^{\circ}C < T_J < 125^{\circ}C$, $I_O = 500mA$, $V_I = 10V$, $C_I = 0.33\mu F$, $C_O = 0.1\mu F$, unless otherwise specified)

Parameter	Symbol	Conditions	MC7805/LM7805			Unit	
			Min.	Typ.	Max.		
Output Voltage	V_O	$T_J = +25^{\circ}C$	4.8	5.0	5.2	V	
		$5.0mA \leq I_O \leq 1.0A$, $P_O \leq 15W$ $V_I = 7V$ to $20V$	4.75	5.0	5.25		
Line Regulation (Note1)	Regline	$T_J = +25^{\circ}C$	$V_O = 7V$ to $25V$	-	4.0	100	mV
			$V_I = 8V$ to $12V$	-	1.6	50	
Load Regulation (Note1)	Regload	$T_J = +25^{\circ}C$	$I_O = 5.0mA$ to $1.5A$	-	9	100	mV
			$I_O = 250mA$ to $750mA$	-	4	50	
Quiescent Current	I_Q	$T_J = +25^{\circ}C$	-	5.0	8.0	mA	
Quiescent Current Change	ΔI_Q	$I_O = 5mA$ to $1.0A$	-	0.03	0.5	mA	
		$V_I = 7V$ to $25V$	-	0.3	1.3		
Output Voltage Drift	$\Delta V_O / \Delta T$	$I_O = 5mA$	-	-0.8	-	mV/ $^{\circ}C$	
Output Noise Voltage	V_N	$f = 10Hz$ to $100KHz$, $T_A = +25^{\circ}C$	-	42	-	$\mu V/V_O$	
Ripple Rejection	RR	$f = 120Hz$ $V_O = 8V$ to $18V$	62	73	-	dB	
Dropout Voltage	V_{Drop}	$I_O = 1A$, $T_J = +25^{\circ}C$	-	2	-	V	
Output Resistance	r_O	$f = 1KHz$	-	15	-	$m\Omega$	
Short Circuit Current	ISC	$V_I = 35V$, $T_A = +25^{\circ}C$	-	230	-	mA	
Peak Current	IPK	$T_J = +25^{\circ}C$	-	2.2	-	A	

Note:

1. Load and line regulation are specified at constant junction temperature. Changes in V_O due to heating effects must be taken into account separately. Pulse testing with low duty is used.

Typical Performance Characteristics

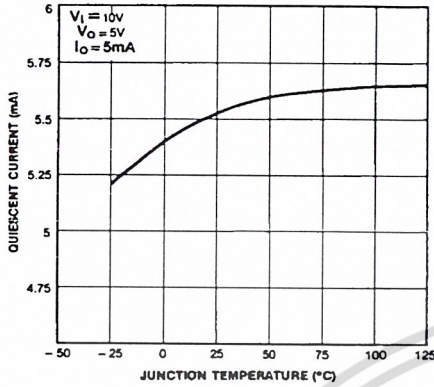


Figure 1. Quiescent Current

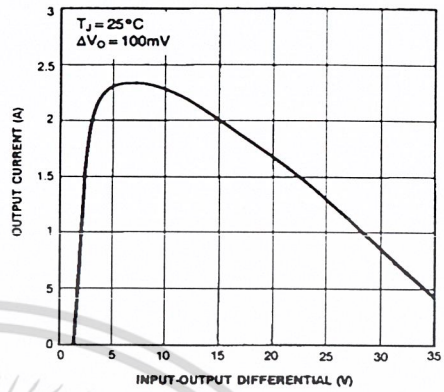


Figure 2. Peak Output Current

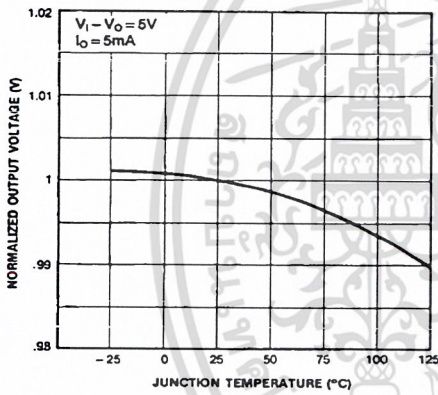


Figure 3. Output Voltage

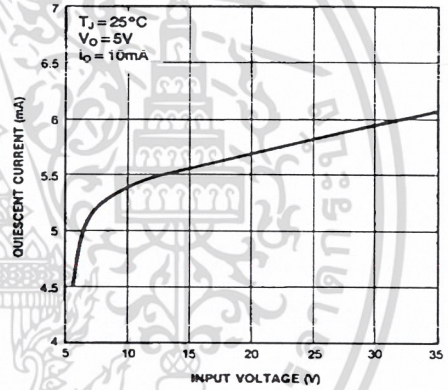


Figure 4. Quiescent Current

Typical Applications

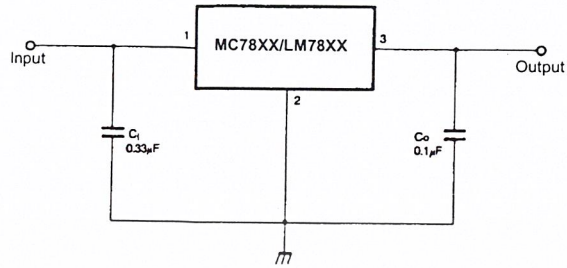


Figure 5. DC Parameters

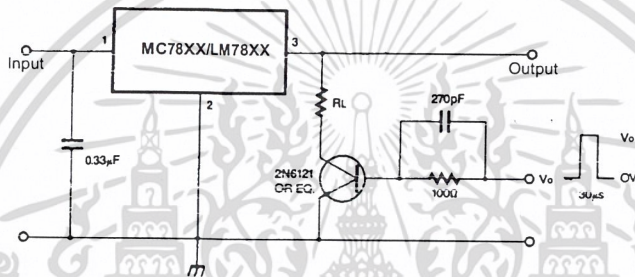


Figure 6. Load Regulation

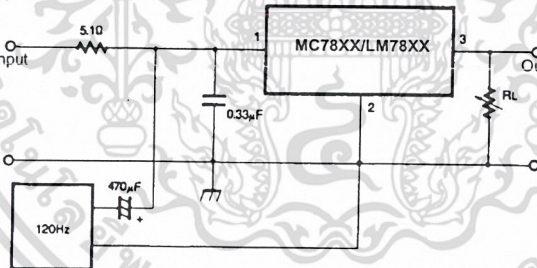


Figure 7. Ripple Rejection

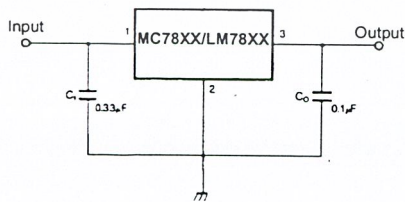


Figure 8. Fixed Output Regulator

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

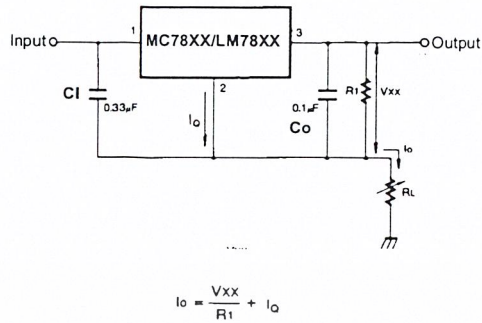
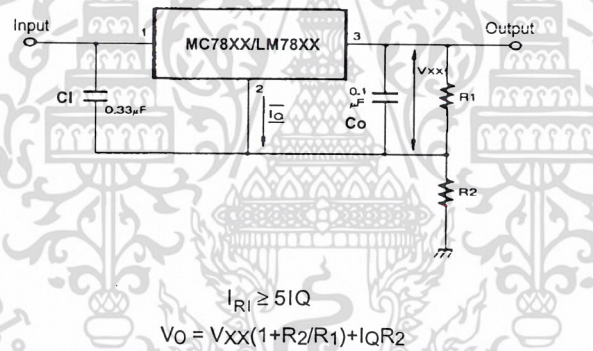


Figure 9. Constant Current Regulator

Notes:

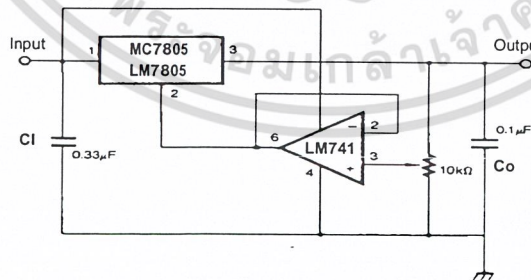
- (1) To specify an output voltage, substitute voltage value for "XX." A common ground is required between the input and the Output voltage. The input voltage must remain typically 2.0V above the output voltage even during the low point on the input ripple voltage.
- (2) C1 is required if regulator is located an appreciable distance from power Supply filter.
- (3) Co improves stability and transient response.



$$I_{R1} \geq 5I_Q$$

$$V_O = V_{XX}(1+R_2/R_1)+I_Q R_2$$

Figure 10. Circuit for Increasing Output Voltage



$$I_{R1} \geq 5I_Q$$

$$V_O = V_{XX}(1+R_2/R_1)+I_Q R_2$$

Figure 11. Adjustable Output Regulator (7 to 30V)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

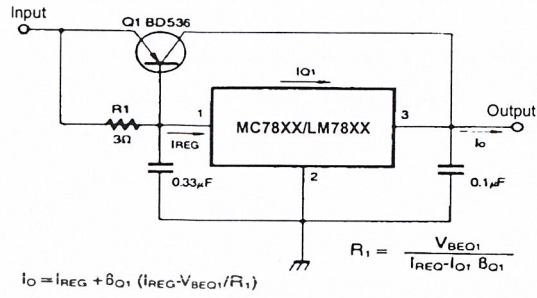


Figure 12. High Current Voltage Regulator

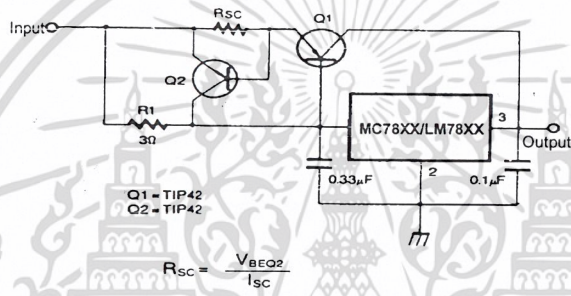


Figure 13. High Output Current with Short Circuit Protection

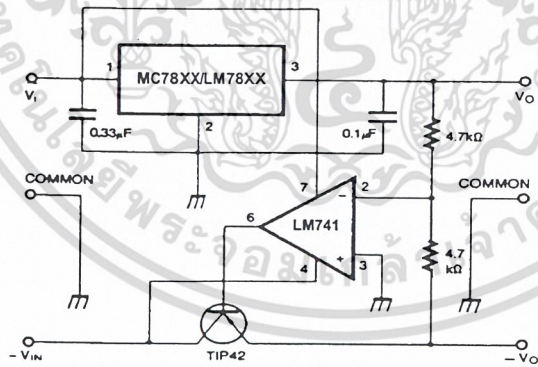


Figure 14. Tracking Voltage Regulator

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

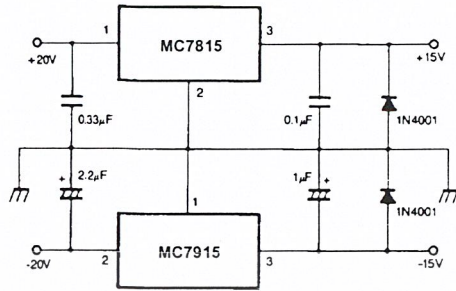


Figure 15. Split Power Supply (±15V-1A)

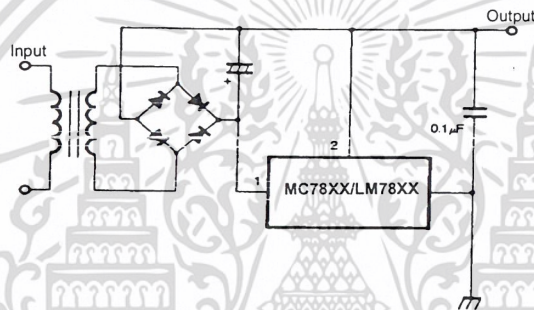


Figure 16. Negative Output Voltage Circuit

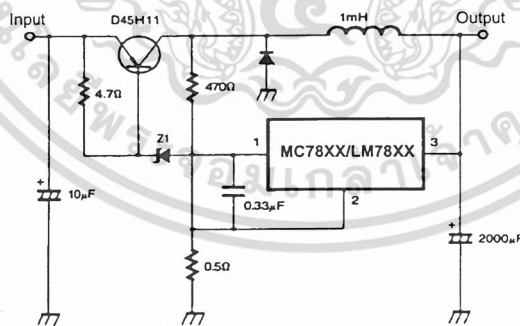
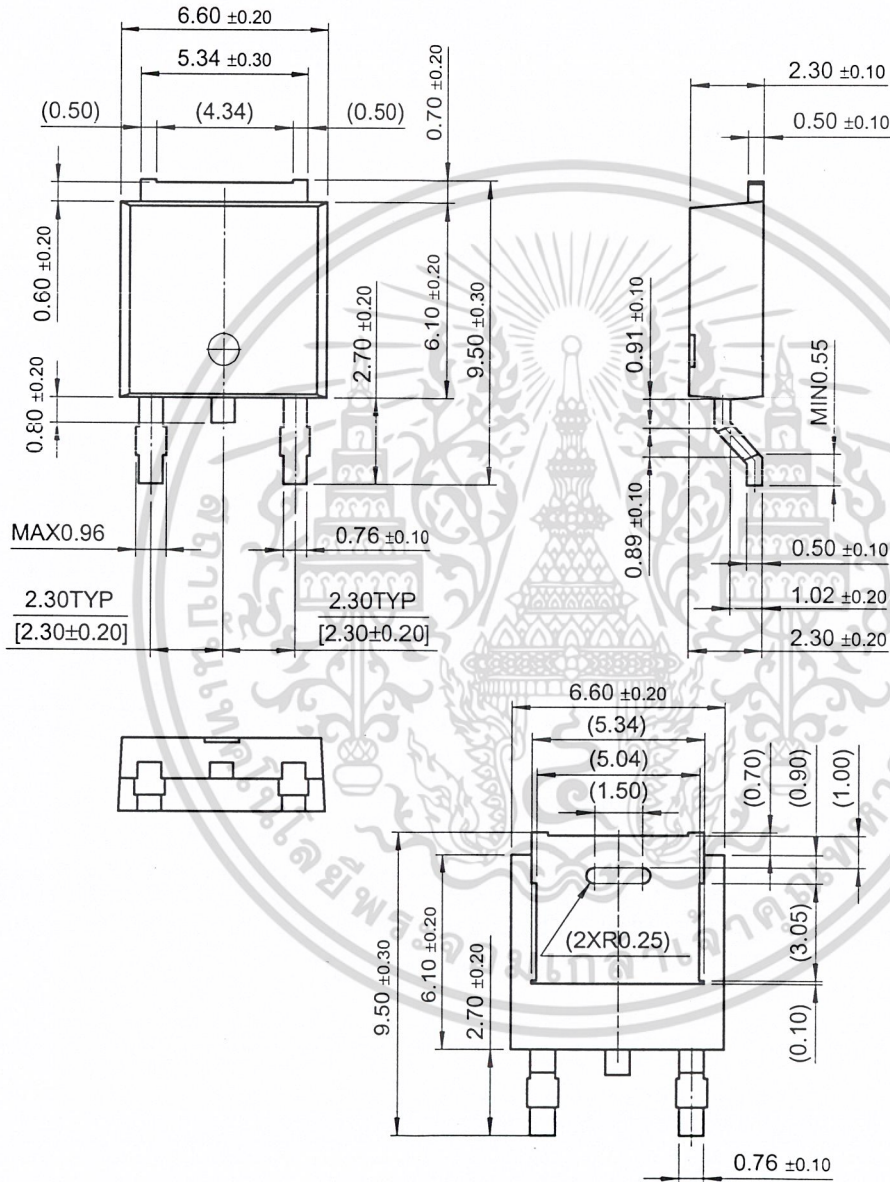


Figure 17. Switching Regulator

Mechanical Dimensions (Continued)

Package

D-PAK



Ordering Information

Product Number	Output Voltage Tolerance	Package	Operating Temperature
LM7805CT	±4%	TO-220	0 ~ + 125°C

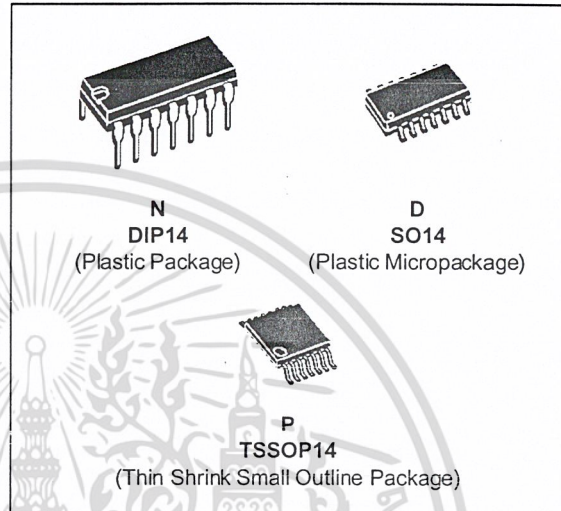
Product Number	Output Voltage Tolerance	Package	Operating Temperature
MC7805CT	±4%	TO-220	0 ~ + 125°C
MC7806CT			
MC7808CT			
MC7809CT			
MC7810CT			
MC7812CT			
MC7815CT			
MC7818CT			
MC7824CT			
MC7805CDT			
MC7806CDT			
MC7808CDT			
MC7809CDT			
MC7810CDT			
MC7812CDT	TO-220		
MC7805ACT			
MC7806ACT			
MC7808ACT			
MC7809ACT			
MC7810ACT			
MC7812ACT			
MC7815ACT			
MC7818ACT			
MC7824ACT			



LM124 LM224 - LM324

LOW POWER QUAD OPERATIONAL AMPLIFIERS

- WIDE GAIN BANDWIDTH : 1.3MHz
- INPUT COMMON-MODE VOLTAGE RANGE INCLUDES GROUND
- LARGE VOLTAGE GAIN : 100dB
- VERY LOW SUPPLY CURRENT/AMPLI : 375 μ A
- LOW INPUT BIAS CURRENT : 20nA
- LOW INPUT OFFSET VOLTAGE : 5mV max.
(for more accurate applications, use the equivalent parts LM124A-LM224A-LM324A which feature 3mV max)
- LOW INPUT OFFSET CURRENT : 2nA
- WIDE POWER SUPPLY RANGE :
SINGLE SUPPLY : +3V TO +30V
DUAL SUPPLIES : $\pm 1.5V$ TO $\pm 15V$



DESCRIPTION

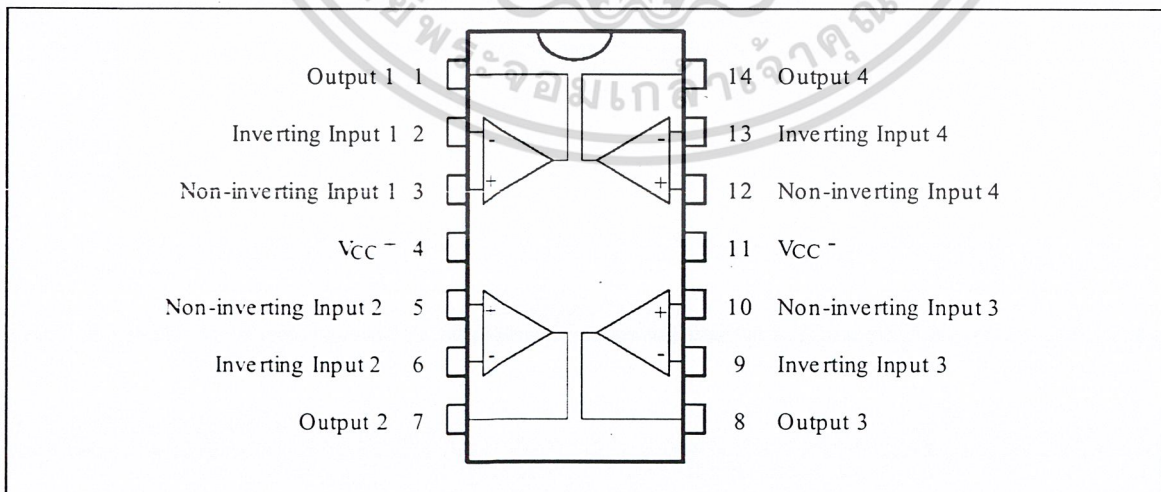
These circuits consist of four independent, high gain, internally frequency compensated operational amplifiers. They operate from a single power supply over a wide range of voltages. Operation from split power supplies is also possible and the low power supply current drain is independent of the magnitude of the power supply voltage.

ORDER CODES

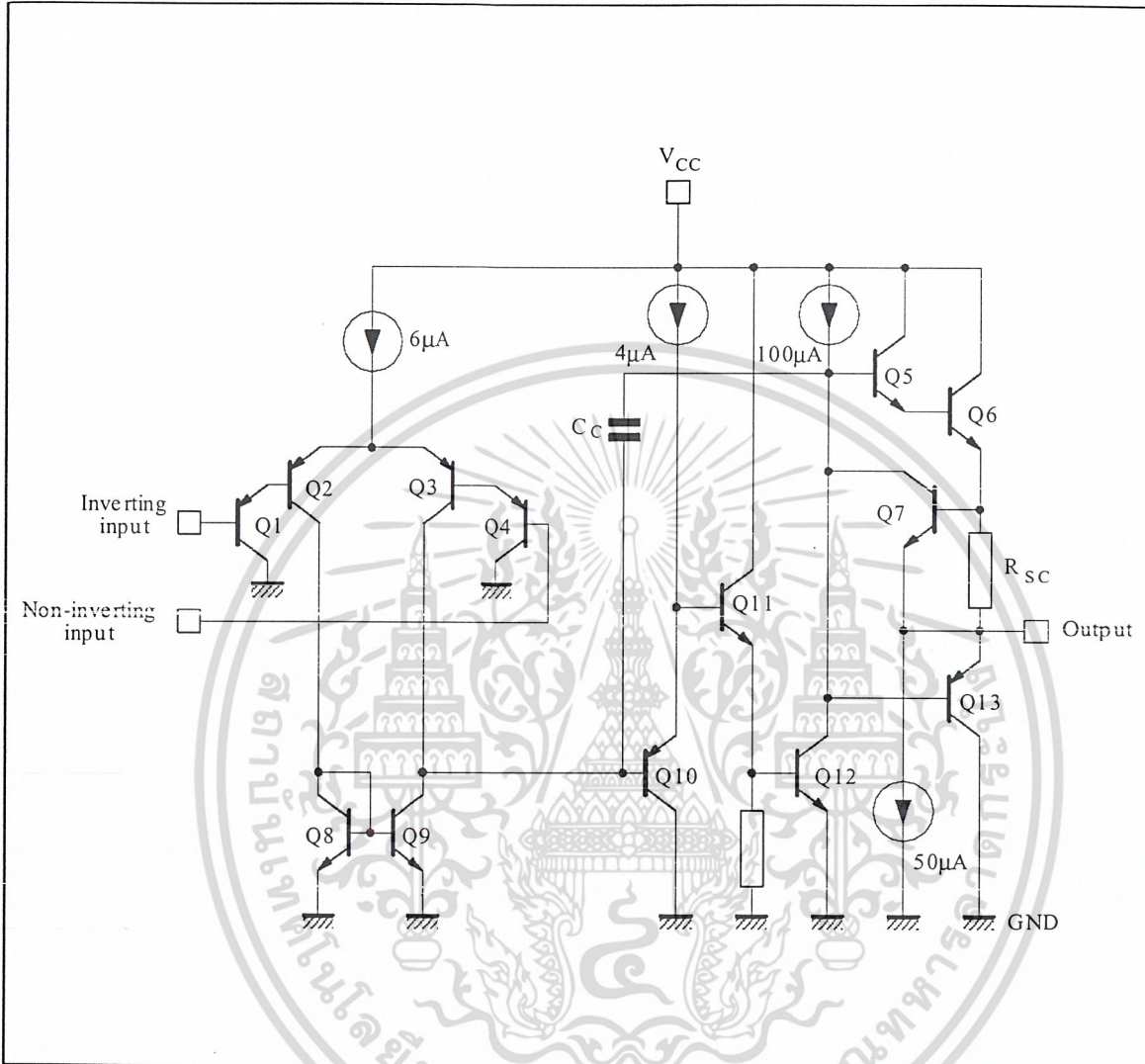
Part Number	Temperature Range	Package		
		N	D	P
LM124	-55°C, +125°C	•	•	•
LM224	-40°C, +105°C	•	•	•
LM324	0°C, +70°C	•	•	•

Example : LM224N

PIN CONNECTIONS (top view)



SCHEMATIC DIAGRAM (1/4 LM124)



ABSOLUTE MAXIMUM RATINGS

Symbol	Parameter	LM124	LM224	LM324	Unit
V_{cc}	Supply Voltage		±16 or 32		V
V_i	Input Voltage		-0.3 to +32		V
V_{id}	Differential Input Voltage - (*)	+32	+32	+32	V
P_{tot}	Power Dissipation N Suffix D Suffix	500 -	500 400	500 400	mW mW
-	Output Short-circuit Duration - (note 1)	Infinite			
I_{in}	Input Current - (note 6)	50	50	50	mA
T_{oper}	Operating Free Air Temperature Range	-55 to +125	-40 to +105	0 to +70	°C
T_{stg}	Storage Temperature Range	-65 to +150	-65 to +150	-65 to +150	°C

ELECTRICAL CHARACTERISTICS

$V_{CC}^+ = +5V$, $V_{CC}^- = \text{Ground}$, $V_O = 1.4V$, $T_{amb} = +25^\circ\text{C}$ (unless otherwise specified)

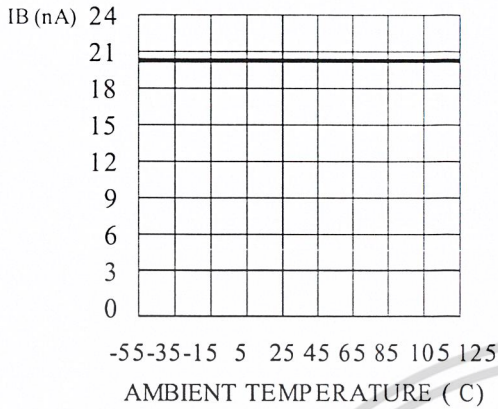
Symbol	Parameter	LM124 - LM224 - LM324			Unit
		Min.	Typ.	Max.	
V_{io}	Input Offset Voltage (note 3) $T_{amb} = +25^\circ\text{C}$ LM324 $T_{min.} \leq T_{amb} \leq T_{max.}$ LM324		2	5 7 7 9	mV
i_{io}	Input Offset Current $T_{amb} = +25^\circ\text{C}$ $T_{min.} \leq T_{amb} \leq T_{max.}$		2	30 100	nA
i_{ib}	Input Bias Current (note 2) $T_{amb} = +25^\circ\text{C}$ $T_{min.} \leq T_{amb} \leq T_{max.}$		20	150 300	nA
A_{vd}	Large Signal Voltage Gain ($V_{CC}^+ = +15V$, $R_L = 2k\Omega$, $V_O = 1.4V$ to $11.4V$) $T_{amb} = +25^\circ\text{C}$ $T_{min.} \leq T_{amb} \leq T_{max.}$	50 25	100		V/mV
SVR	Supply Voltage Rejection Ratio ($R_S \leq 10k\Omega$) ($V_{CC}^+ = 5V$ to $30V$) $T_{amb} = +25^\circ\text{C}$ $T_{min.} \leq T_{amb} \leq T_{max.}$	65 65	110		dB
I_{CC}	Supply Current, all Amp, no load $T_{amb} = +25^\circ\text{C}$ $T_{min.} \leq T_{amb} \leq T_{max.}$		$V_{CC} = +5V$ 0.7 $V_{CC} = +30V$ 1.5 $V_{CC} = +5V$ 0.8 $V_{CC} = +30V$ 1.5	1.2 3 1.2 3	mA
V_{icm}	Input Common Mode Voltage Range ($V_{CC} = +30V$) - (note 4) $T_{amb} = +25^\circ\text{C}$ $T_{min.} \leq T_{amb} \leq T_{max.}$	0 0		$V_{CC} - 1.5$ $V_{CC} - 2$	V
CMR	Common-mode Rejection Ratio ($R_S \leq 10k\Omega$) $T_{amb} = +25^\circ\text{C}$ $T_{min.} \leq T_{amb} \leq T_{max.}$	70 60	80		dB
I_{source}	Output Current Source ($V_{id} = +1V$) $V_{CC} = +15V$, $V_O = +2V$	20	40	70	mA
I_{sink}	Output Sink Current ($V_{id} = -1V$) $V_{CC} = +15V$, $V_O = +2V$ $V_{CC} = +15V$, $V_O = +0.2V$	10 12	20 50		mA μA

ELECTRICAL CHARACTERISTICS (continued)

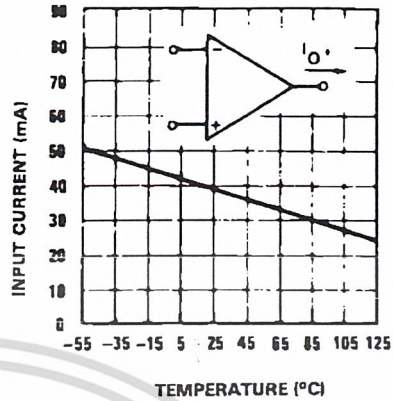
Symbol	Parameter	LM124 - LM224 - LM324			Unit
		Min.	Typ.	Max.	
V _{OH}	High Level Output Voltage (V _{CC} = +30V) T _{amb} = +25°C T _{min.} ≤ T _{amb} ≤ T _{max.} R _L = 2kΩ	26	27		V
	T _{amb} = +25°C T _{min.} ≤ T _{amb} ≤ T _{max.} (V _{CC} = +5V, R _L = 2kΩ)	26	28		
V _{OL}	Low Level Output Voltage (R _L = 10kΩ) T _{amb} = +25°C T _{min.} ≤ T _{amb} ≤ T _{max.}		5	20	mV
				20	
SR	Slew Rate V _{CC} = 15V, V _I = 0.5 to 3V, R _L = 2kΩ, C _L = 100pF, unity gain)		0.4		V/μs
GBP	Gain Bandwidth Product V _{CC} = 30V, f = 100kHz, V _{in} = 10mV R _L = 2kΩ, C _L = 100pF		1.3		MHz
THD	Total Harmonic Distortion f = 1kHz, A _V = 20dB, R _L = 2kΩ, V _O = 2V _{pp} C _L = 100pF, V _{CC} = 30V		0.015		%
e _n	Equivalent Input Noise Voltage f = 1kHz, R _s = 100Ω, V _{CC} = 30V		40		$\frac{nV}{\sqrt{Hz}}$
DV _{io}	Input Offset Voltage Drift		7	30	μV/°C
DI _{io}	Input Offset Current Drift		10	200	pA/°C
V _{O1} /V _{O2}	Channel Separation (note 5) 1kHz ≤ f ≤ 20kHz		120		dB

- Notes :**
- Short-circuits from the output to V_{CC} can cause excessive heating if V_{CC} > 15V. The maximum output current is approximately 40mA independent of the magnitude of V_{CC}. Destructive dissipation can result from simultaneous short-circuit on all amplifiers.
 - The direction of the input current is out of the IC. This current is essentially constant, independent of the state of the output so no loading change exists on the input lines.
 - V_O = 1.4V, R_s = 0Ω, 5V < V_{CC} < 30V, 0 < V_{ic} < V_{CC} - 1.5V
 - The input common-mode voltage of either input signal voltage should not be allowed to go negative by more than 0.3V. The upper end of the common-mode voltage range is V_{CC} - 1.5V, but either or both inputs can go to +32V without damage.
 - Due to the proximity of external components insure that coupling is not originating via stray capacitance between these external parts. This typically can be detected as this type of capacitance increases at higher frequencies.
 - This input current only exists when the voltage at any of the input leads is driven negative. It is due to the collector-base junction of the input PNP transistor becoming forward biased and thereby acting as input diodes clamps. In addition to this diode action, there is also NPN parasitic action on the IC chip. This transistor action can cause the output voltages of the Op-amps to go to the V_{CC} voltage level (or to ground for a large overdrive) for the time duration than an input is driven negative.
This is not destructive and normal output will set up again for input voltage higher than -0.3V.

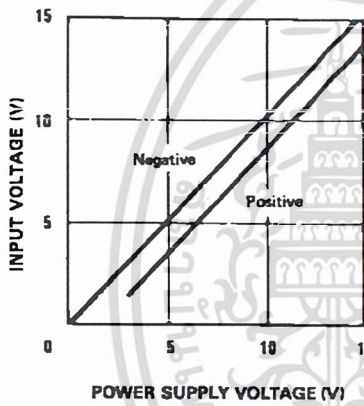
INPUT BIAS CURRENT
versus AMBIENT TEMPERATURE



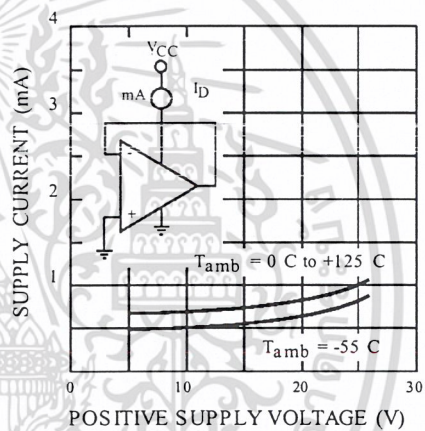
CURRENT LIMITING (Note 8)



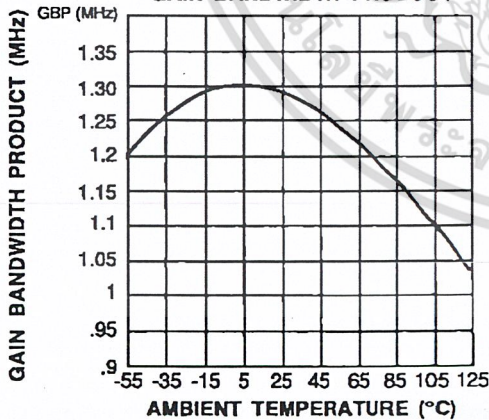
INPUT VOLTAGE RANGE



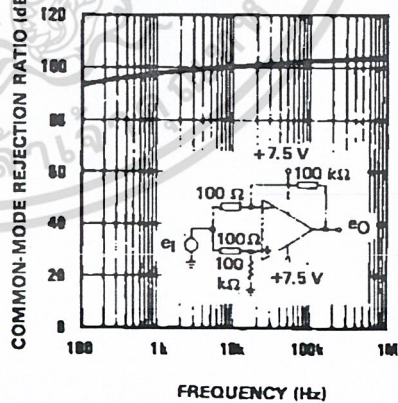
SUPPLY CURRENT



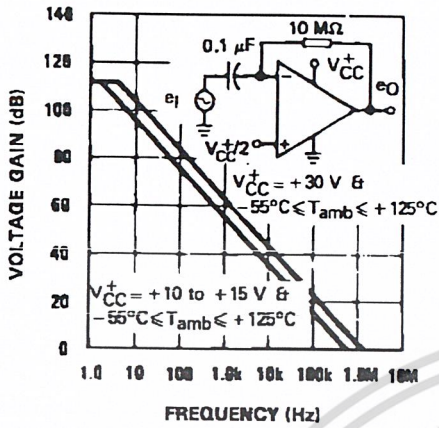
GAIN BANDWIDTH PRODUCT



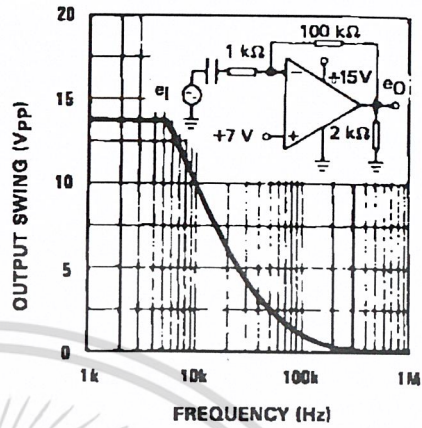
COMMON-MODE REJECTION RATIO



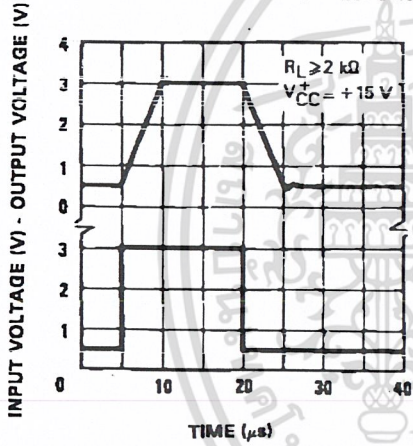
OPEN LOOP FREQUENCY RESPONSE



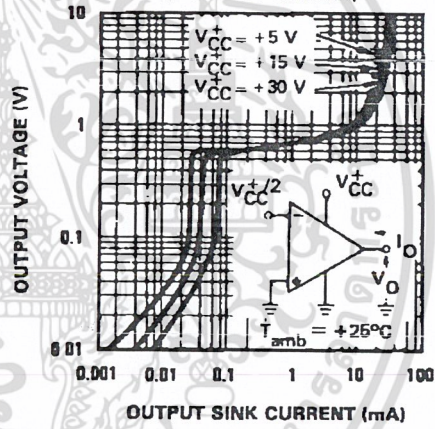
LARGE SIGNAL FREQUENCY RESPONSE



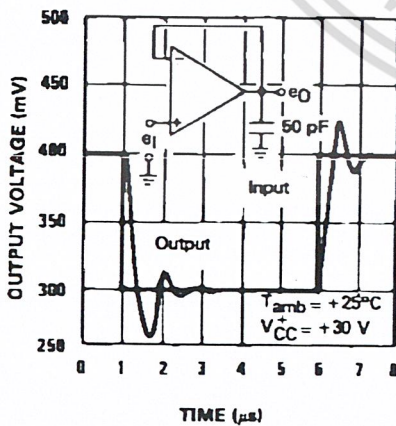
VOLTAGE FOLLOWER PULSE RESPONSE



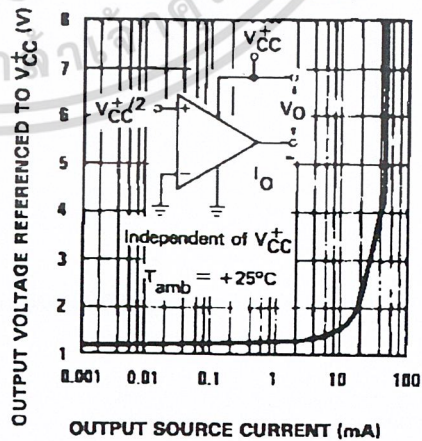
OUTPUT CHARACTERISTICS (CURRENT SINKING)

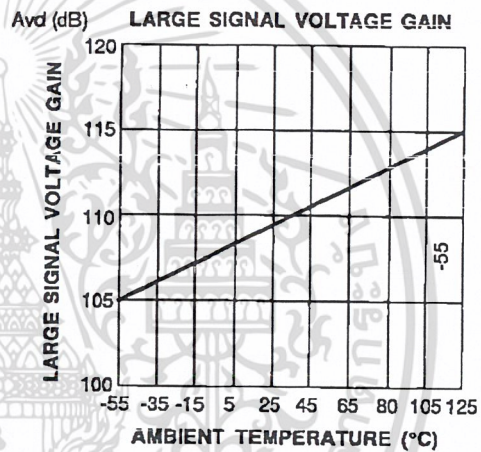
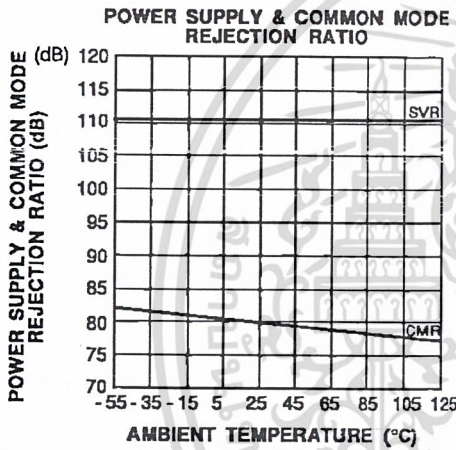
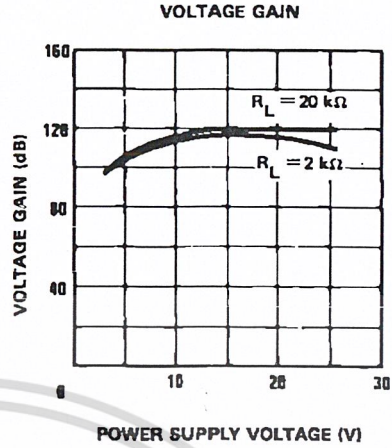
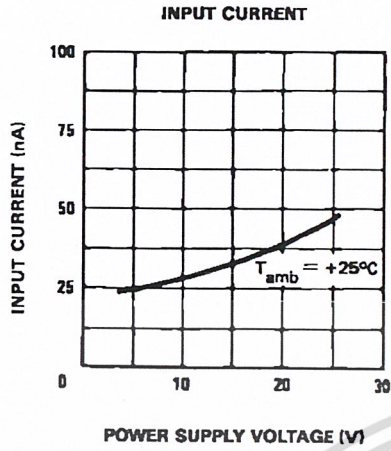


VOLTAGE FOLLOWER PULSE RESPONSE (SMALL SIGNAL)



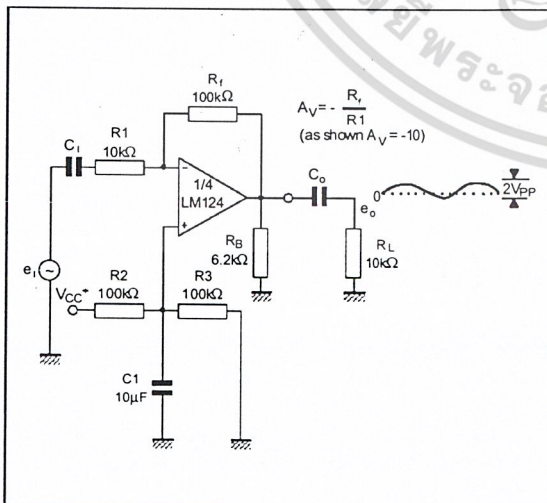
OUTPUT CHARACTERISTICS (CURRENT SOURCING)



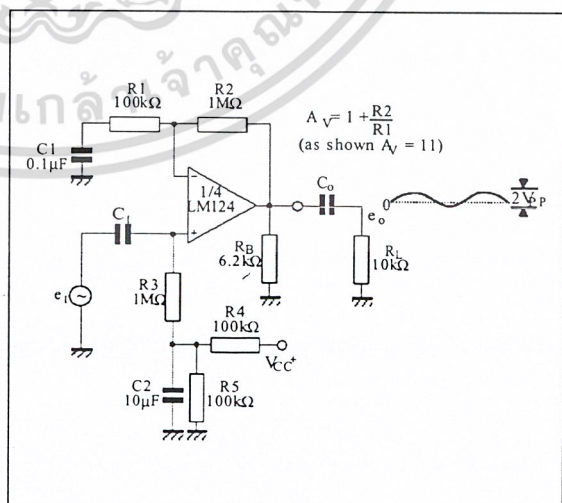


TYPICAL SINGLE - SUPPLY APPLICATIONS

AC COUPLED INVERTING AMPLIFIER



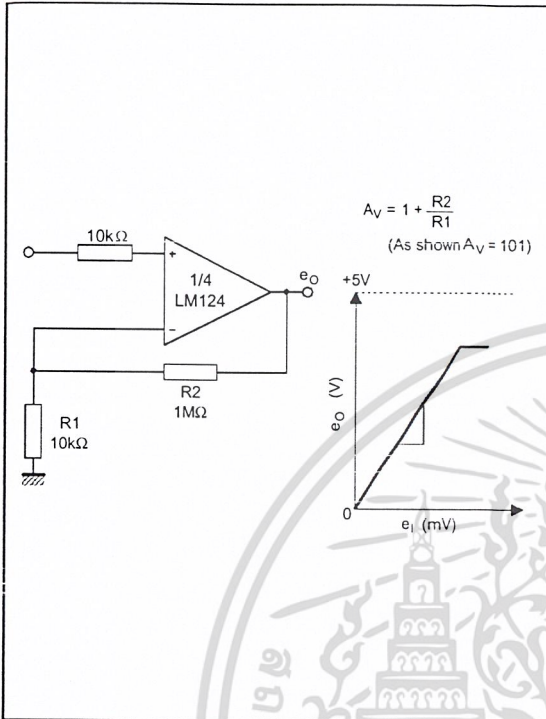
AC COUPLED NON-INVERTING AMPLIFIER



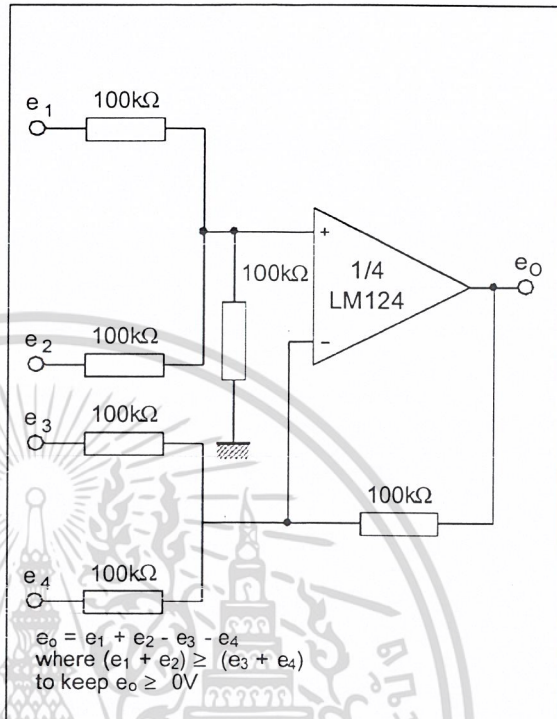
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

TYPICAL SINGLE - SUPPLY APPLICATIONS

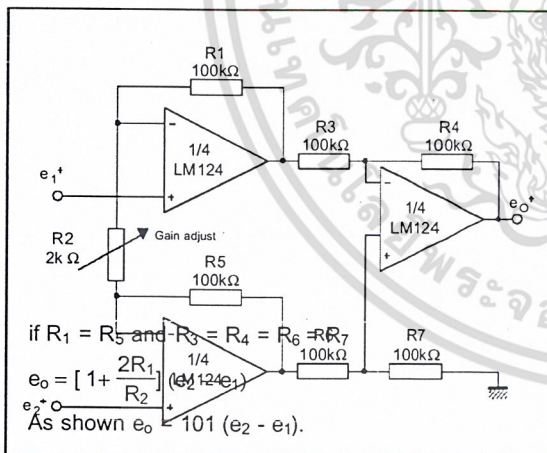
NON-INVERTING DC GAIN



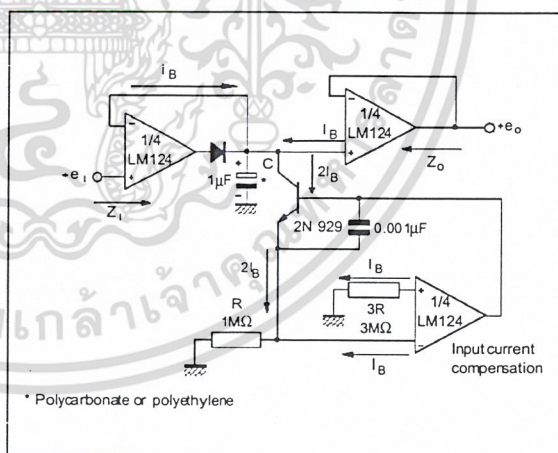
DC SUMMING AMPLIFIER



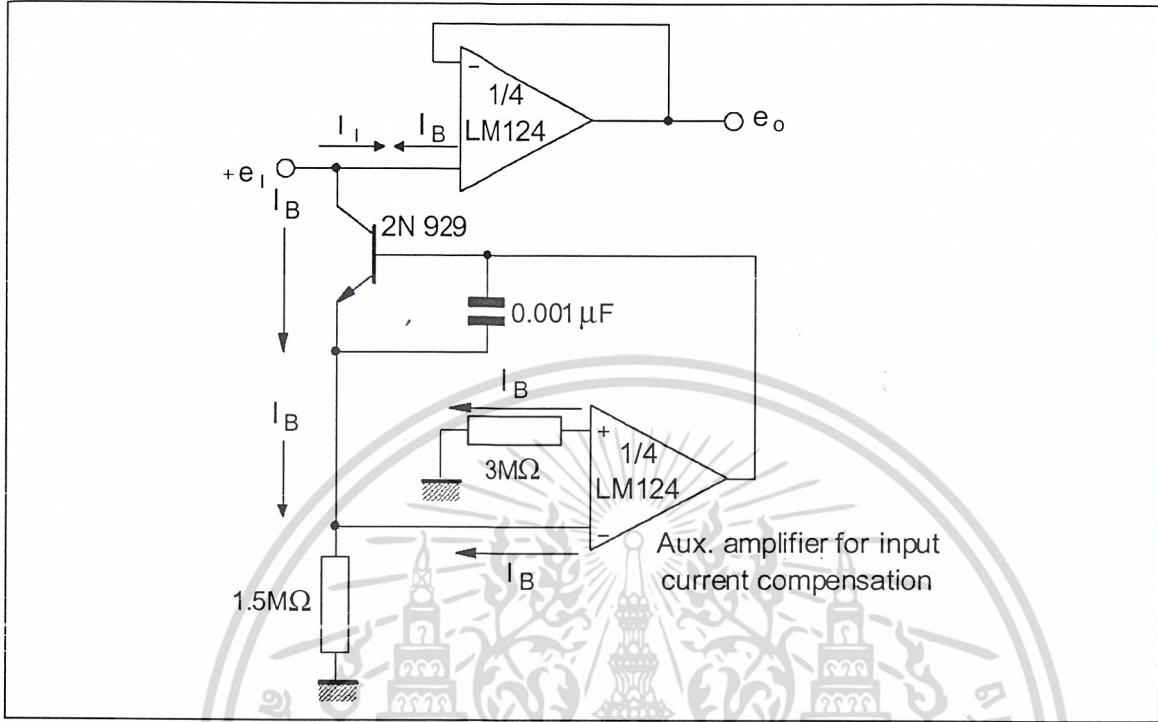
HIGH INPUT Z ADJUSTABLE GAIN DC INSTRUMENTATION AMPLIFIER



LOW DRIFT PEAK DETECTOR

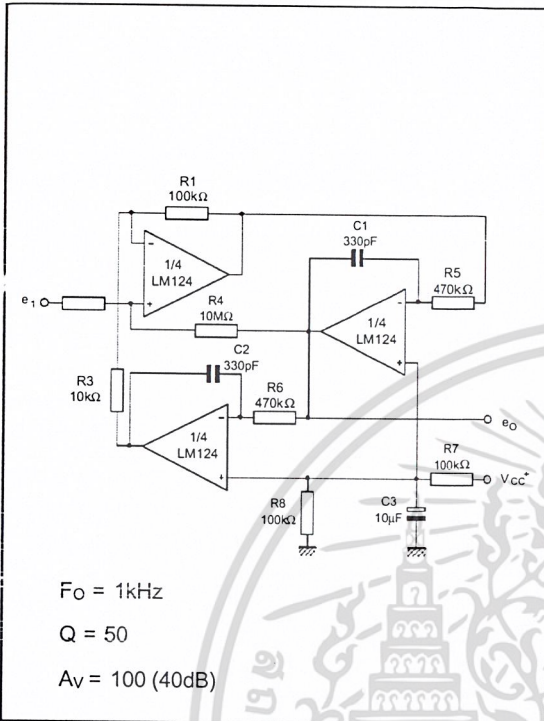


USING SYMMETRICAL AMPLIFIERS TO REDUCE INPUT CURRENT (GENERAL CONCEPT)

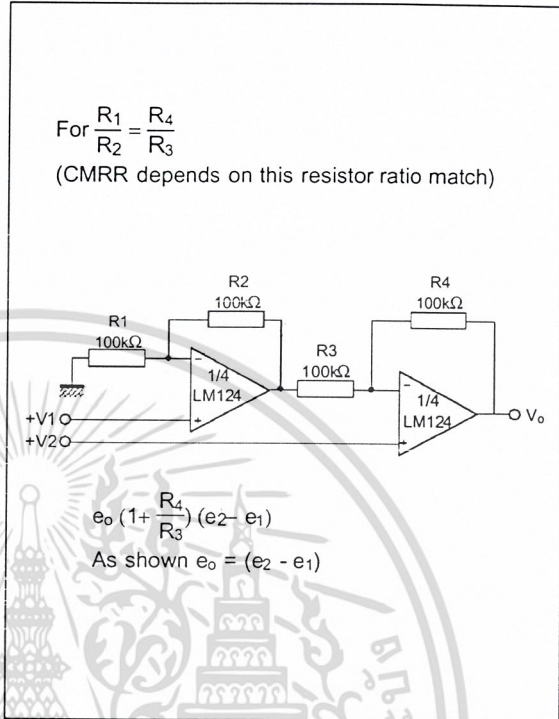


TYPICAL SINGLE - SUPPLY APPLICATIONS

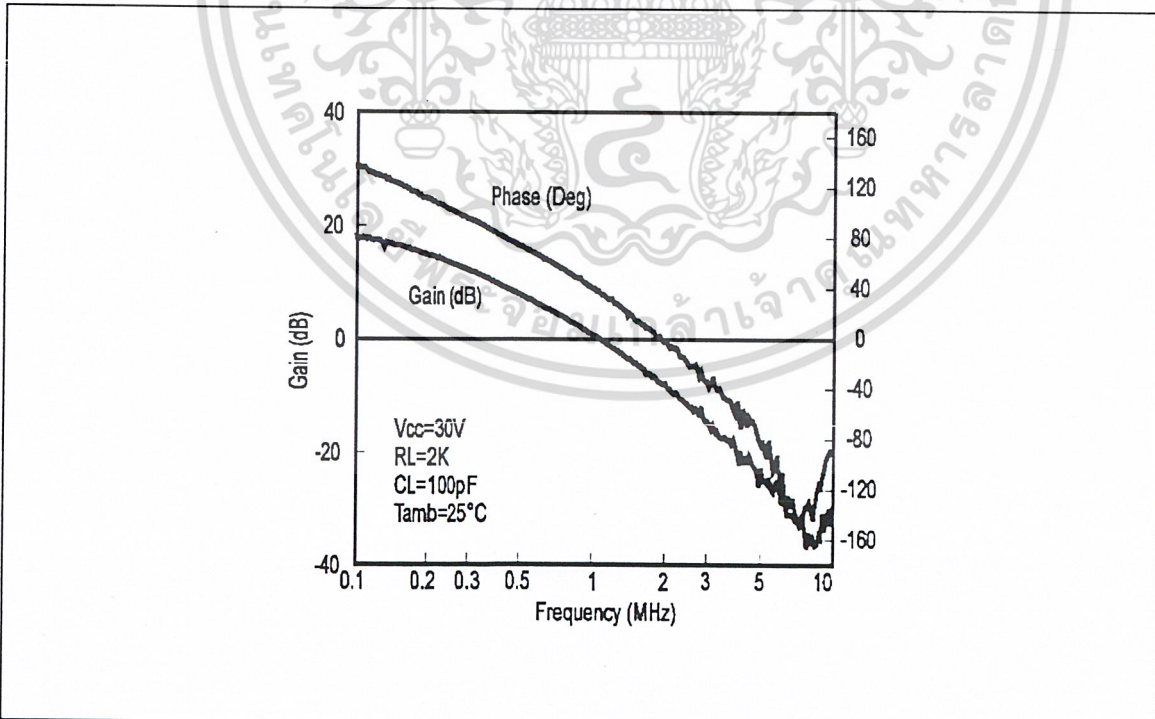
ACTIVER BANDPASS FILTER



HIGH INPUT Z, DC DIFFERENTIAL AMPLIFIER



VOLTAGE GAIN AND PHASE vs FREQUENCY



- LARGE VOLTAGE GAIN : 100dB
- VERY LOW SUPPLY CURRENT/AMPLI : 375µA
- LOW INPUT BIAS CURRENT : 20nA
- LOW INPUT OFFSET VOLTAGE : 2mV
- LOW INPUT OFFSET CURRENT : 2nA
- WIDE POWER SUPPLY RANGE : SINGLE SUPPLY : +3V to +30V DUAL SUPPLIES : ±1.5V to ±15V

Applies to : LM124-LM224-LM324

** Standard Linear Ics Macromodels, 1993.

** CONNECTIONS :

* 1 INVERTING INPUT

* 2 NON-INVERTING INPUT

* 3 OUTPUT

* 4 POSITIVE POWER SUPPLY

* 5 NEGATIVE POWER SUPPLY

DINR 15 18 MDTH 400E-12

VIP 4 18 2.000000E+00

FCP 4 5 VOFP 3.400000E+01

FCN 5 4 VOFN 3.400000E+01

FIBP 2 5 VOFN 2.000000E-03

FIBN 5 1 VOFP 2.000000E-03

* AMPLIFYING STAGE

FIP 5 19 VOFP 3.600000E+02

FIN 5 19 VOFN 3.600000E+02

RG1 19 5 3.652997E+06

RG2 19 4 3.652997E+06

CC 19 5 6.000000E-09

DOPM 19 22 MDTH 400E-12

DONM 21 19 MDTH 400E-12

HOPM 22 28 VOUT 7.500000E+03

VIPM 28 4 1.500000E+02

HONM 21 27 VOUT 7.500000E+03

VINM 5 27 1.500000E+02

EOUT 26 23 19 5 1

VOUT 23 5 0

ROUT 26 3 20

COUT 3 5 1.000000E-12

DOP 19 25 MDTH 400E-12

VOP 4 25 2.242230E+00

DON 24 19 MDTH 400E-12

VON 24 5 7.922301E-01

.ENDS

.SUBCKT LM124 1 3 2 4 5 (analog)

.MODEL MDTH D IS=1E-8 KF=3.104131E-15 CJO=10F

* INPUT STAGE

CIP 2 5 1.000000E-12

CIN 1 5 1.000000E-12

EIP 10 5 2 5 1

EIN 16 5 1 5 1

RIP 10 11 2.600000E+01

RIN 15 16 2.600000E+01

RIS 11 15 2.003862E+02

DIP 11 12 MDTH 400E-12

DIN 15 14 MDTH 400E-12

VOFP 12 13 DC 0

VOFN 13 14 DC 0

IPOL 13 5 1.000000E-05

CPS 11 15 3.783376E-09

DINN 17 13 MDTH 400E-12

VIN 17 5 0.000000E+00

ELECTRICAL CHARACTERISTICS

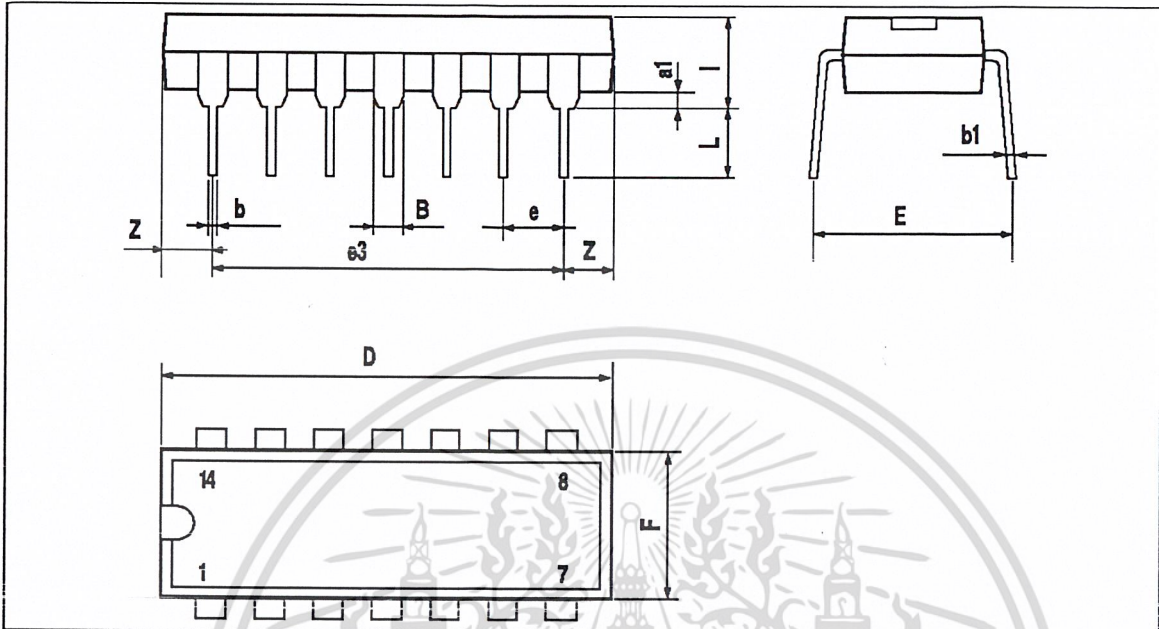
V_{CC}⁺ = +5V, V_{CC}⁻ = 0V, T_{amb} = 25°C (unless otherwise specified)

Symbol	Conditions	Value	Unit
V _{io}		0	mV
A _{vd}	R _L = 2kΩ	100	V/mV
I _{cc}	No load, per operator	350	µA
V _{icm}		-15 to +13.5	V
V _{OH}	R _L = 2kΩ (V _{CC} ⁺ = 15V)	+13.5	V
V _{OL}	R _L = 10kΩ	5	mV
I _{os}	V _O = +2V, V _{CC} = +15V	+40	mA
GBP	R _L = 2kΩ. C _L = 100pF	1.3	MHz
SR	R _L = 2kΩ. C _L = 100pF	0.4	V/µs



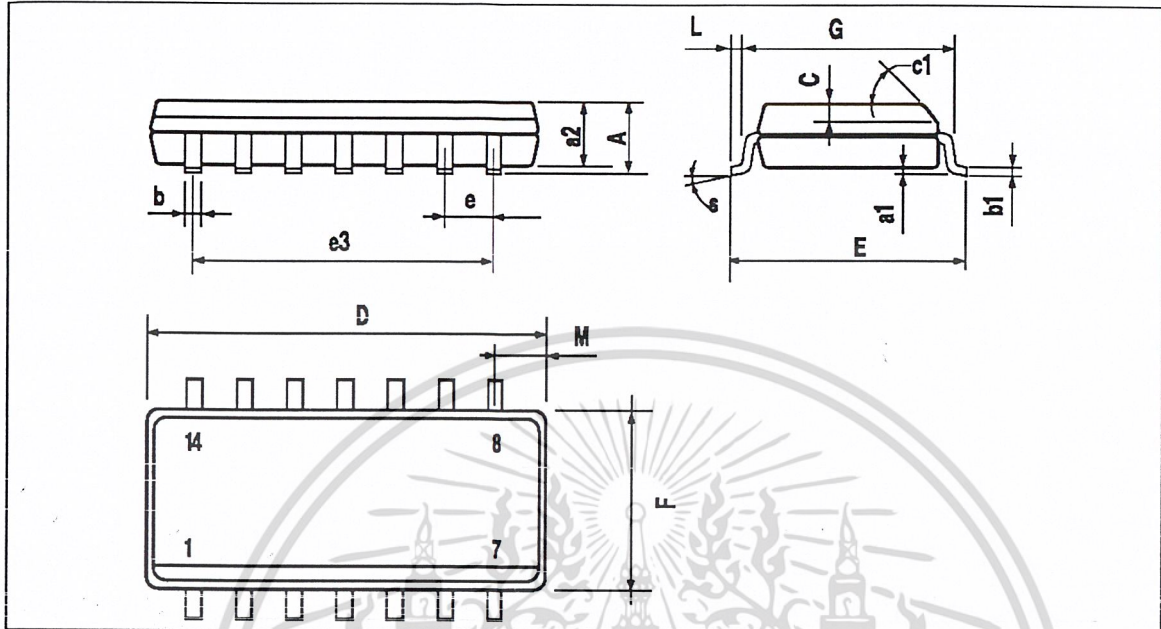
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

PACKAGE MECHANICAL DATA
14 PINS - PLASTIC DIP



Dimensions	Millimeters			Inches		
	Min.	Typ.	Max.	Min.	Typ.	Max.
a1	0.51			0.020		
B	1.39		1.65	0.055		0.065
b		0.5			0.020	
b1		0.25			0.010	
D			20			0.787
E		8.5			0.335	
e		2.54			0.100	
e3		15.24			0.600	
F			7.1			0.280
i			5.1			0.201
L		3.3			0.130	
Z	1.27		2.54	0.050		0.100

PACKAGE MECHANICAL DATA
14 PINS - PLASTIC MICROPACKAGE (SO)



PM-SO14.EPS

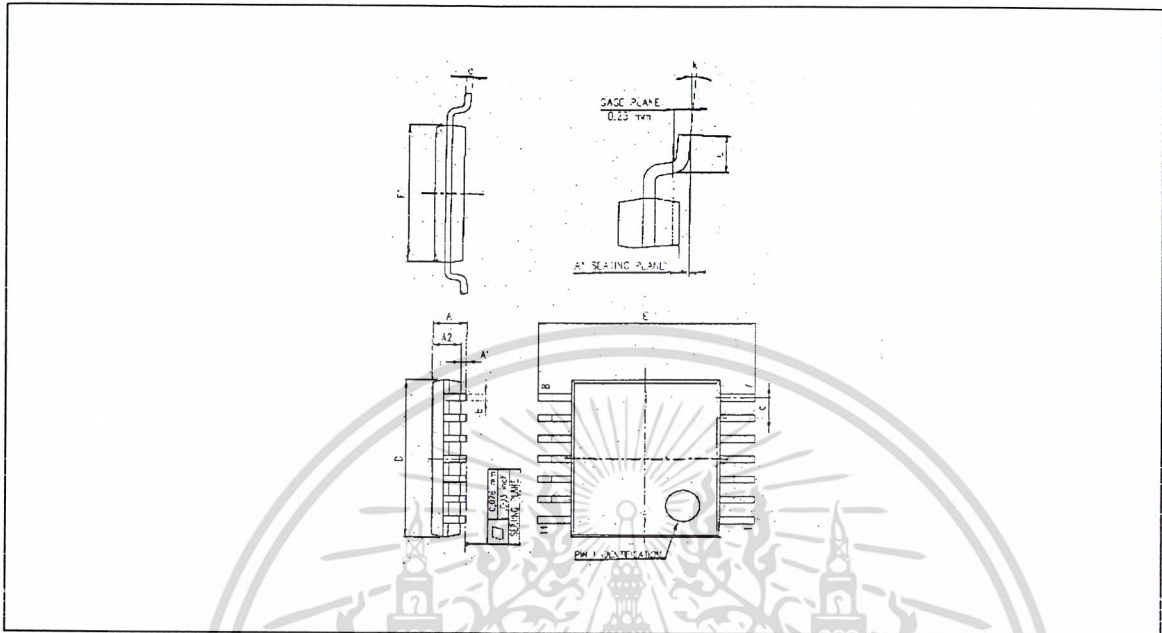
Dimensions	Millimeters			Inches		
	Min.	Typ.	Max.	Min.	Typ.	Max.
A			1.75			0.069
a1	0.1		0.2	0.004		0.008
a2			1.6			0.063
b	0.35		0.46	0.014		0.018
b1	0.19		0.25	0.007		0.010
C		0.5			0.020	
c1	45° (typ.)					
D	8.55		8.75	0.336		0.334
E	5.8		6.2	0.228		0.244
e		1.27			0.050	
e3		7.62			0.300	
F	3.8		4.0	0.150		0.157
G	4.6		5.3	0.181		0.208
L	0.5		1.27	0.020		0.050
M			0.68			0.027
S	8° (max.)					

SO14.TBL



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

PACKAGE MECHANICAL DATA
14 PINS - THIN SHRINK SMALL OUTLINE PACKAGE



Dim.	Millimeters			Inches		
	Min.	Typ.	Max.	Min.	Typ.	Max.
A			1.20			0.05
A1	0.05		0.15	0.01		0.006
A2	0.80	1.00	1.05	0.031	0.039	0.041
b	0.19		0.30	0.007		0.15
c	0.09		0.20	0.003		0.012
D	4.90	5.00	5.10	0.192	0.196	0.20
E		6.40			0.252	
E1	4.30	4.40	4.50	0.169	0.173	0.177
e		0.65			0.025	
k	0°		8°	0°		8°
l	0.50	0.60	0.75	0.09	0.0236	0.030

Information furnished is believed to be accurate and reliable. However, STMicroelectronics assumes no responsibility for the consequences of use of such information nor for any infringement of patents or other rights of third parties which may result from its use. No license is granted by implication or otherwise under any patent or patent rights of STMicroelectronics. Specifications mentioned in this publication are subject to change without notice. This publication supersedes and replaces all information previously supplied. STMicroelectronics products are not authorized for use as critical components in life support devices or systems without express written approval of STMicroelectronics.

© The ST logo is a trademark of STMicroelectronics

© 1999 STMicroelectronics – Printed in Italy – All Rights Reserved

STMicroelectronics GROUP OF COMPANIES

Australia - Brazil - Canada - China - France - Germany - Italy - Japan - Korea - Malaysia - Malta - Mexico - Morocco
 The Netherlands - Singapore - Spain - Sweden - Switzerland - Taiwan - Thailand - United Kingdom - U.S.A.

© <http://www.st.com>



MAXIM

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

General Description

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

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

Applications

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

Features

Superior to Bipolar

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

Ordering Information

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

Ordering Information continued at end of data sheet.

*Contact factory for dice specifications.

Selection Table

Part Number	Power Supply (V)	No. of RS-232 Drivers/Rx	No. of Ext. Caps	Nominal Cap. Value (μF)	SHDN & Three-State	Rx Active in SHDN	Data Rate (kbps)	Features
MAX220	+5	2/2	4	4.7(10)	No	—	120	Ultra-low-power, industry-standard pinout
MAX222	+5	2/2	4	0.1	Yes	—	200	Low-power shutdown
MAX223 (MAX213)	+5	4/5	4	1.0 (0.1)	Yes	✓	120	MAX241 and receivers active in shutdown
MAX225	+5	5/5	0	—	Yes	✓	120	Available in SO
MAX230 (MAX200)	+5	5/0	4	1.0 (0.1)	Yes	—	120	5 drivers with shutdown
MAX231 (MAX201)	+5 and +7.5 to +13.2	2/2	2	1.0 (0.1)	No	—	120	Standard +5/+12V or battery supplies; same functions as MAX232
MAX232 (MAX202)	+5	2/2	4	1.0 (0.1)	No	—	120 (64)	Industry standard
MAX232A	+5	2/2	4	0.1	No	—	200	Higher slew rate, small caps
MAX233 (MAX203)	+5	2/2	0	—	No	—	120	No external caps
MAX233A	+5	2/2	0	—	No	—	200	No external caps, high slew rate
MAX234 (MAX204)	+5	4/0	4	1.0 (0.1)	No	—	120	Replaces 1488
MAX235 (MAX205)	+5	5/5	0	—	Yes	—	120	No external caps
MAX236 (MAX206)	+5	4/3	4	1.0 (0.1)	Yes	—	120	Shutdown, three state
MAX237 (MAX207)	+5	5/3	4	1.0 (0.1)	No	—	120	Complements IBM PC serial port
MAX238 (MAX208)	+5	4/4	4	1.0 (0.1)	No	—	120	Replaces 1488 and 1489
MAX239 (MAX209)	+5 and +7.5 to +13.2	3/5	2	1.0 (0.1)	No	—	120	Standard +5/+12V or battery supplies; single-package solution for IBM PC serial port
MAX240	+5	5/5	4	1.0	Yes	—	120	DIP or flatpack package
MAX241 (MAX211)	+5	4/5	4	1.0 (0.1)	Yes	—	120	Complete IBM PC serial port
MAX242	+5	2/2	4	0.1	Yes	✓	200	Separate shutdown and enable
MAX243	+5	2/2	4	0.1	No	—	200	Open-line detection simplifies cabling
MAX244	+5	8/10	4	1.0	No	—	120	High slew rate
MAX245	+5	8/10	0	—	Yes	✓	120	High slew rate, int. caps, two shutdown modes
MAX246	+5	8/10	0	—	Yes	✓	120	High slew rate, int. caps, three shutdown modes
MAX247	+5	8/9	0	—	Yes	✓	120	High slew rate, int. caps, nine operating modes
MAX248	+5	8/8	4	1.0	Yes	✓	120	High slew rate, selective half-chip enables
MAX249	+5	6/10	4	1.0	Yes	✓	120	Available in quad flatpack package

MAXIM

Maxim Integrated Products 1

For free samples & the latest literature: <http://www.maxim-ic.com>, or phone 1-800-998-8800.
For small orders, phone 1-800-835-8769.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

MAX220-MAX249

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

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

Supply Voltage (V _{CC})	-0.3V to +6V	20-Pin Plastic DIP (derate 8.00mW/°C above +70°C)	440mW
Input Voltages		16-Pin Narrow SO (derate 8.70mW/°C above +70°C)	696mW
T _{IN}	-0.3V to (V _{CC} - 0.3V)	16-Pin Wide SO (derate 9.52mW/°C above +70°C)	762mW
R _{IN} (Except MAX220)	±30V	18-Pin Wide SO (derate 9.52mW/°C above +70°C)	762mW
R _{IN} (MAX220)	±25V	20-Pin Wide SO (derate 10.00mW/°C above +70°C)	800mW
T _{OUT} (Except MAX220) (Note 1)	±15V	20-Pin SSOP (derate 8.00mW/°C above +70°C)	640mW
T _{OUT} (MAX220)	±13.2V	16-Pin CERDIP (derate 10.00mW/°C above +70°C)	800mW
Output Voltages		18-Pin CERDIP (derate 10.53mW/°C above +70°C)	842mW
T _{OUT}	±15V	Operating Temperature Ranges	
R _{OUT}	-0.3V to (V _{CC} + 0.3V)	MAX2__AC__, MAX2__C__	0°C to +70°C
Driver/Receiver Output Short Circuited to GND	Continuous	MAX2__AE__, MAX2__E__	-40°C to +85°C
Continuous Power Dissipation (T _A = +70°C)		MAX2__AM__, MAX2__M__	-55°C to +125°C
16-Pin Plastic DIP (derate 10.53mW/°C above +70°C)	842mW	Storage Temperature Range	-65°C to +160°C
18-Pin Plastic DIP (derate 11.11mW/°C above +70°C)	889mW	Lead Temperature (soldering, 10sec)	+300°C

Note 1: Input voltage measured with T_{OUT} in high-impedance state, SHDN or V_{CC} = 0V.

Note 2: For the MAX220, V₊ and V₋ can have a maximum magnitude of 7V, but their absolute difference cannot exceed 13V.

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

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

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

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

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

MAX220-MAX249

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

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

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

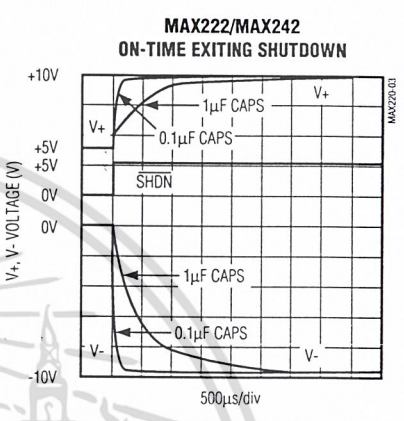
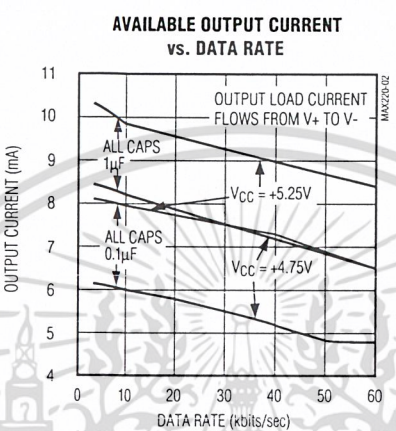
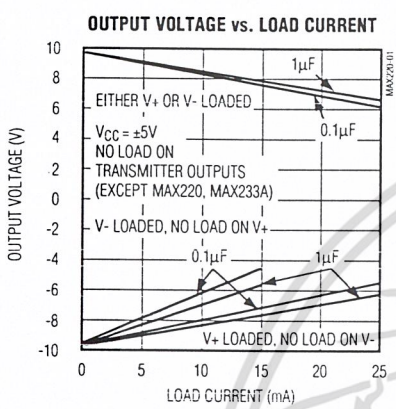
Note 3: MAX243 R_{2OUT} is guaranteed to be low when R_{2IN} is ≥ 0V or is floating.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

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

Typical Operating Characteristics

MAX220/MAX222/MAX232A/MAX233A/MAX242/MAX243



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
 ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

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

MAX220-MAX249

ABSOLUTE MAXIMUM RATINGS—MAX223/MAX230-MAX241

V _{CC}	-0.3V to +6V	20-Pin Wide SO (derate 10.00mW/°C above +70°C).....	800mW
V ₊	(V _{CC} - 0.3V) to +14V	24-Pin Wide SO (derate 11.76mW/°C above +70°C).....	941mW
V ₋	+0.3V to -14V	28-Pin Wide SO (derate 12.50mW/°C above +70°C).....	1W
Input Voltages		44-Pin Plastic FP (derate 11.11mW/°C above +70°C).....	889mW
T _{IN}	-0.3V to (V _{CC} + 0.3V)	14-Pin CERDIP (derate 9.09mW/°C above +70°C).....	727mW
R _{IN}	±30V	16-Pin CERDIP (derate 10.00mW/°C above +70°C).....	800mW
Output Voltages		20-Pin CERDIP (derate 11.11mW/°C above +70°C).....	889mW
T _{OUT}	(V ₊ + 0.3V) * (V ₋ - 0.3V)	24-Pin Narrow CERDIP	
R _{OUT}	-0.3V to (V _{CC} + 0.3V)	(derate 12.50mW/°C above +70°C).....	1W
Short-Circuit Duration, T _{OUT}	Continuous	24-Pin Sidebraze (derate 20.0mW/°C above +70°C).....	1.6W
Continuous Power Dissipation (T _A = +70°C)		28-Pin SSOP (derate 9.52mW/°C above +70°C).....	762mW
14-Pin Plastic DIP (derate 10.00mW/°C above +70°C).....		Operating Temperature Ranges	
16-Pin Plastic DIP (derate 10.53mW/°C above +70°C).....		MAX2___C.....	0°C to +70°C
20-Pin Plastic DIP (derate 11.11mW/°C above +70°C).....		MAX2___E.....	-40°C to +85°C
24-Pin Narrow Plastic DIP		MAX2___M.....	-55°C to +125°C
(derate 13.33mW/°C above +70°C).....		Storage Temperature Range.....	-65°C to +160°C
24-Pin Plastic DIP (derate 9.09mW/°C above +70°C).....		Lead Temperature (soldering, 10sec).....	+300°C
16-Pin Wide SO (derate 9.52mW/°C above +70°C).....			

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

ELECTRICAL CHARACTERISTICS—MAX223/MAX230-MAX241

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

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

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

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

ELECTRICAL CHARACTERISTICS—MAX223/MAX230—MAX241 (continued)

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

PARAMETER	CONDITIONS		MIN	TYP	MAX	UNITS
RS-232 Input Threshold Low	$T_A = +25^\circ C$, $V_{CC} = 5V$	Normal operation SHDN = 5V (MAX223) SHDN = 0V (MAX235/236/240/241)	0.8	1.2		V
		Shutdown (MAX223) SHDN = 0V, EN = 5V (R4IN, R5IN)	0.6	1.5		
RS-232 Input Threshold High	$T_A = +25^\circ C$, $V_{CC} = 5V$	Normal operation SHDN = 5V (MAX223) SHDN = 0V (MAX235/236/240/241)		1.7	2.4	V
		Shutdown (MAX223) SHDN = 0V, EN = 5V (R4IN, R5IN)		1.5	2.4	
RS-232 Input Hysteresis	$V_{CC} = 5V$, no hysteresis in shutdown		0.2	0.5	1.0	V
RS-232 Input Resistance	$T_A = +25^\circ C$, $V_{CC} = 5V$		3	5	7	k Ω
TTL/CMOS Output Voltage Low	$I_{OUT} = 1.6mA$ (MAX231/232/233, $I_{OUT} = 3.2mA$)				0.4	V
TTL/CMOS Output Voltage High	$I_{OUT} = -1mA$		3.5	$V_{CC} - 0.4$		V
TTL/CMOS Output Leakage Current	$0V \leq R_{OUT} \leq V_{CC}$; EN = 0V (MAX223); EN = V_{CC} (MAX235-241)			0.05	± 10	μA
Receiver Output Enable Time	Normal operation	MAX223		600		ns
		MAX235/236/239/240/241		400		
Receiver Output Disable Time	Normal operation	MAX223		900		ns
		MAX235/236/239/240/241		250		
Propagation Delay	RS-232 IN to TTL/CMOS OUT, $C_L = 150pF$	Normal operation		0.5	10	μs
		SHDN = 0V (MAX223)	t_{PHLS}	4	40	
			t_{PLHS}	6	40	
Transition Region Slew Rate	MAX223/MAX230/MAX234-241, $T_A = +25^\circ C$, $V_{CC} = 5V$, $R_L = 3k\Omega$ to $7k\Omega$, $C_L = 50pF$ to $2500pF$, measured from $+3V$ to $-3V$ or $-3V$ to $+3V$		3	5.1	30	V/ μs
	MAX231/MAX232/MAX233, $T_A = +25^\circ C$, $V_{CC} = 5V$, $R_L = 3k\Omega$ to $7k\Omega$, $C_L = 50pF$ to $2500pF$, measured from $+3V$ to $-3V$ or $-3V$ to $+3V$			4	30	
Transmitter Output Resistance	$V_{CC} = V+ = V- = 0V$, $V_{OUT} = \pm 2V$		300			Ω
Transmitter Output Short-Circuit Current				± 10		mA

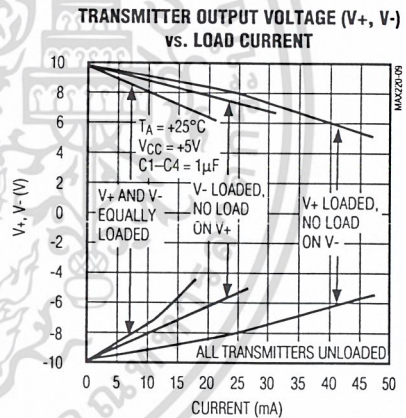
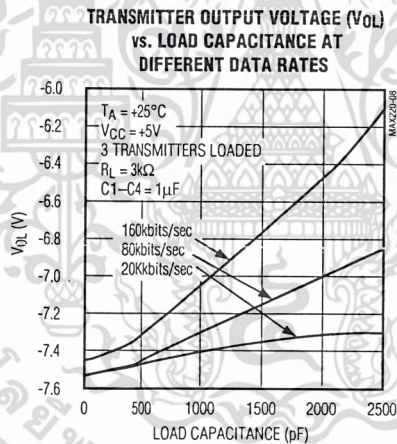
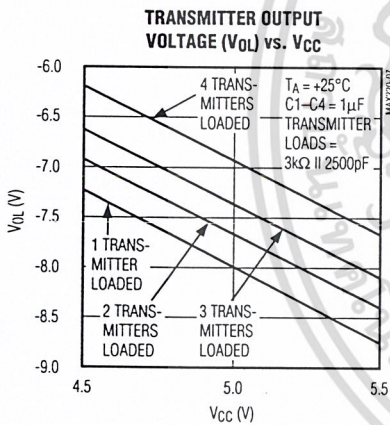
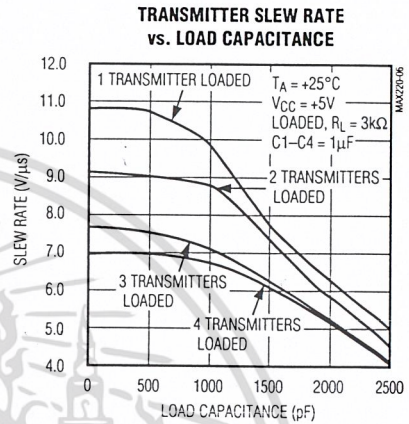
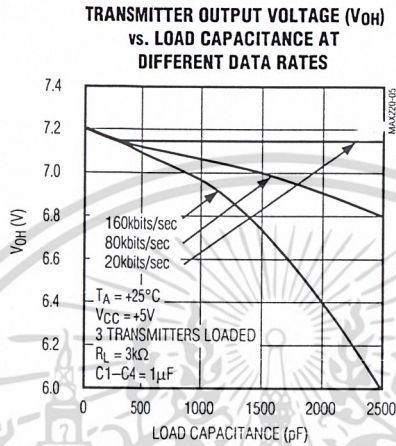
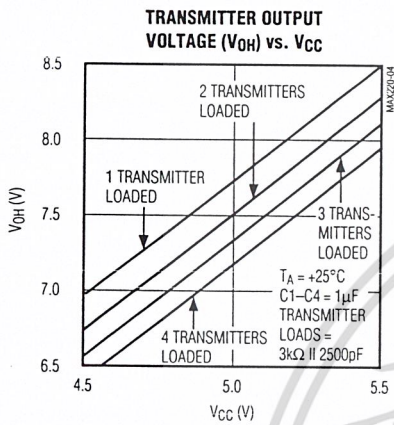
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

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

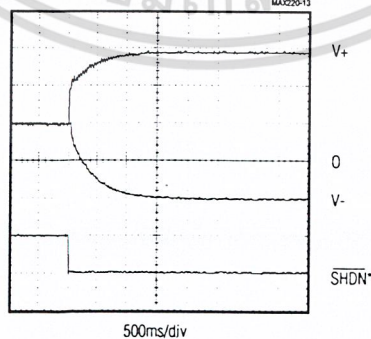
Typical Operating Characteristics

MAX223/MAX230-MAX241

MAX220-MAX249



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



*SHUTDOWN POLARITY IS REVERSED FOR NON MAX241 PARTS

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

Detailed Description

The MAX220–MAX249 contain four sections: dual charge-pump DC-DC voltage converters, RS-232 drivers, RS-232 receivers, and receiver and transmitter enable control inputs.

Dual Charge-Pump Voltage Converter

The MAX220–MAX249 have two internal charge-pumps that convert +5V to $\pm 10V$ (unloaded) for RS-232 driver operation. The first converter uses capacitor C1 to double the +5V input to +10V on C3 at the V+ output. The second converter uses capacitor C2 to invert +10V to -10V on C4 at the V- output.

A small amount of power may be drawn from the +10V (V+) and -10V (V-) outputs to power external circuitry (see the *Typical Operating Characteristics* section), except on the MAX225 and MAX245–MAX247, where these pins are not available. V+ and V- are not regulated, so the output voltage drops with increasing load current. Do not load V+ and V- to a point that violates the minimum $\pm 5V$ EIA/TIA-232E driver output voltage when sourcing current from V+ and V- to external circuitry.

When using the shutdown feature in the MAX222, MAX225, MAX230, MAX235, MAX236, MAX240, MAX241, and MAX245–MAX249, avoid using V+ and V- to power external circuitry. When these parts are shut down, V- falls to 0V, and V+ falls to +5V. For applications where a +10V external supply is applied to the V+ pin (instead of using the internal charge pump to generate +10V), the C1 capacitor must not be installed and the SHDN pin must be tied to VCC. This is because V+ is internally connected to VCC in shutdown mode.

RS-232 Drivers

The typical driver output voltage swing is $\pm 8V$ when loaded with a nominal $5k\Omega$ RS-232 receiver and $V_{CC} = +5V$. Output swing is guaranteed to meet the EIA/TIA-232E and V.28 specification, which calls for $\pm 5V$ minimum driver output levels under worst-case conditions. These include a minimum $3k\Omega$ load, $V_{CC} = +4.5V$, and maximum operating temperature. Unloaded driver output voltage ranges from (V+ -1.3V) to (V- +0.5V).

Input thresholds are both TTL and CMOS compatible. The inputs of unused drivers can be left unconnected since $400k\Omega$ input pull-up resistors to VCC are built in (except for the MAX220). The pull-up resistors force the outputs of unused drivers low because all drivers invert. The internal input pull-up resistors typically source $12\mu A$, except in shutdown mode where the pull-ups are disabled. Driver outputs turn off and enter a high-impedance state—where leakage current is typically microamperes (maximum $25\mu A$)—when in shutdown

mode, in three-state mode, or when device power is removed. Outputs can be driven to $\pm 15V$. The power-supply current typically drops to $8\mu A$ in shutdown mode. The MAX220 does not have pull-up resistors to force the outputs of the unused drivers low. Connect unused inputs to GND or VCC.

The MAX239 has a receiver three-state control line, and the MAX223, MAX225, MAX235, MAX236, MAX240, and MAX241 have both a receiver three-state control line and a low-power shutdown control. Table 2 shows the effects of the shutdown control and receiver three-state control on the receiver outputs.

The receiver TTL/CMOS outputs are in a high-impedance, three-state mode whenever the three-state enable line is high (for the MAX225/MAX235/MAX236/MAX239–MAX241), and are also high-impedance whenever the shutdown control line is high.

When in low-power shutdown mode, the driver outputs are turned off and their leakage current is less than $1\mu A$ with the driver output pulled to ground. The driver output leakage remains less than $1\mu A$, even if the transmitter output is backdriven between 0V and ($V_{CC} + 6V$). Below -0.5V, the transmitter is diode clamped to ground with $1k\Omega$ series impedance. The transmitter is also zener clamped to approximately $V_{CC} + 6V$, with a series impedance of $1k\Omega$.

The driver output slew rate is limited to less than $30V/\mu s$ as required by the EIA/TIA-232E and V.28 specifications. Typical slew rates are $24V/\mu s$ unloaded and $10V/\mu s$ loaded with 3Ω and $2500pF$.

RS-232 Receivers

EIA/TIA-232E and V.28 specifications define a voltage level greater than 3V as a logic 0, so all receivers invert. Input thresholds are set at 0.8V and 2.4V, so receivers respond to TTL level inputs as well as EIA/TIA-232E and V.28 levels.

The receiver inputs withstand an input overvoltage up to $\pm 25V$ and provide input terminating resistors with

Table 2. Three-State Control of Receivers

PART	SHDN	SHDN	EN	EN(R)	RECEIVERS
MAX223	—	Low High High	X Low High	—	High Impedance Active High Impedance
MAX225	—	—	—	Low High	High Impedance Active
MAX235 MAX236 MAX240	Low Low High	—	—	Low High X	High Impedance Active High Impedance

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

nominal 5k Ω values. The receivers implement Type 1 interpretation of the fault conditions of V.28 and EIA/TIA-232E.

The receiver input hysteresis is typically 0.5V with a guaranteed minimum of 0.2V. This produces clear output transitions with slow-moving input signals, even with moderate amounts of noise and ringing. The receiver propagation delay is typically 600ns and is independent of input swing direction.

Low-Power Receive Mode

The low-power receive-mode feature of the MAX223, MAX242, and MAX245-MAX249 puts the IC into shutdown mode but still allows it to receive information. This is important for applications where systems are periodically awakened to look for activity. Using low-power receive mode, the system can still receive a signal that will activate it on command and prepare it for communication at faster data rates. This operation conserves system power.

Negative Threshold—MAX243

The MAX243 is pin compatible with the MAX232A, differing only in that RS-232 cable fault protection is removed on one of the two receiver inputs. This means that control lines such as CTS and RTS can either be driven or left floating without interrupting communication. Different cables are not needed to interface with different pieces of equipment.

The input threshold of the receiver without cable fault protection is -0.8V rather than +1.4V. Its output goes positive only if the input is connected to a control line that is actively driven negative. If not driven, it defaults to the 0 or "OK to send" state. Normally, the MAX243's other receiver (+1.4V threshold) is used for the data line (TD or RD), while the negative threshold receiver is connected to the control line (DTR, DTS, CTS, RTS, etc.).

Other members of the RS-232 family implement the optional cable fault protection as specified by EIA/TIA-232E specifications. This means a receiver output goes high whenever its input is driven negative, left floating, or shorted to ground. The high output tells the serial communications IC to stop sending data. To avoid this, the control lines must either be driven or connected with jumpers to an appropriate positive voltage level.

Shutdown—MAX222-MAX242

On the MAX222, MAX235, MAX236, MAX240, and MAX241, all receivers are disabled during shutdown. On the MAX223 and MAX242, two receivers continue to operate in a reduced power mode when the chip is in shutdown. Under these conditions, the propagation delay increases to about 2.5 μ s for a high-to-low input transition. When in shutdown, the receiver acts as a CMOS inverter with no hysteresis. The MAX223 and MAX242 also have a receiver output enable input ($\overline{\text{EN}}$ for the MAX242 and EN for the MAX223) that allows receiver output control independent of SHDN (SHDN for MAX241). With all other devices, SHDN (SHDN for MAX241) also disables the receiver outputs.

The MAX225 provides five transmitters and five receivers, while the MAX245 provides ten receivers and eight transmitters. Both devices have separate receiver and transmitter-enable controls. The charge pumps turn off and the devices shut down when a logic high is applied to the ENT input. In this state, the supply current drops to less than 25 μ A and the receivers continue to operate in a low-power receive mode. Driver outputs enter a high-impedance state (three-state mode). On the MAX225, all five receivers are controlled by the $\overline{\text{ENR}}$ input. On the MAX245, eight of the receiver outputs are controlled by the $\overline{\text{ENR}}$ input, while the remaining two receivers (RA5 and RB5) are always active. RA1-RA4 and RB1-RB4 are put in a three-state mode when $\overline{\text{ENR}}$ is a logic high.

Receiver and Transmitter Enable Control Inputs

The MAX225 and MAX245-MAX249 feature transmitter and receiver enable controls.

The receivers have three modes of operation: full-speed receive (normal active), three-state (disabled), and low-power receive (enabled receivers continue to function at lower data rates). The receiver enable inputs control the full-speed receive and three-state modes. The transmitters have two modes of operation: full-speed transmit (normal active) and three-state (disabled). The transmitter enable inputs also control the shutdown mode. The device enters shutdown mode when all transmitters are disabled. Enabled receivers function in the low-power receive mode when in shutdown.

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

Tables 1a–1d define the control states. The MAX244 has no control pins and is not included in these tables.

The MAX246 has ten receivers and eight drivers with two control pins, each controlling one side of the device. A logic high at the A-side control input (\overline{ENA}) causes the four A-side receivers and drivers to go into a three-state mode. Similarly, the B-side control input (\overline{ENB}) causes the four B-side drivers and receivers to go into a three-state mode. As in the MAX245, one A-side and one B-side receiver (RA5 and RB5) remain active at all times. The entire device is put into shutdown mode when both the A and B sides are disabled ($\overline{ENA} = \overline{ENB} = +5V$).

The MAX247 provides nine receivers and eight drivers with four control pins. The \overline{ENRA} and \overline{ENRB} receiver enable inputs each control four receiver outputs. The \overline{ENTA} and \overline{ENTB} transmitter enable inputs each control four drivers. The ninth receiver (RB5) is always active. The device enters shutdown mode with a logic high on both \overline{ENTA} and \overline{ENTB} .

The MAX248 provides eight receivers and eight drivers with four control pins. The \overline{ENRA} and \overline{ENRB} receiver enable inputs each control four receiver outputs. The \overline{ENTA} and \overline{ENTB} transmitter enable inputs control four drivers each. This part does not have an always-active receiver. The device enters shutdown mode and transmitters go into a three-state mode with a logic high on both \overline{ENTA} and \overline{ENTB} .

The MAX249 provides ten receivers and six drivers with four control pins. The \overline{ENRA} and \overline{ENRB} receiver enable inputs each control five receiver outputs. The \overline{ENTA} and \overline{ENTB} transmitter enable inputs control three drivers each. There is no always-active receiver. The device enters shutdown mode and transmitters go into a three-state mode with a logic high on both \overline{ENTA} and \overline{ENTB} . In shutdown mode, active receivers operate in a low-power receive mode at data rates up to 20kbits/sec.

Applications Information

Figures 5 through 25 show pin configurations and typical operating circuits. In applications that are sensitive to power-supply noise, VCC should be decoupled to ground with a capacitor of the same value as C1 and C2 connected as close as possible to the device.

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

MAX220-MAX249

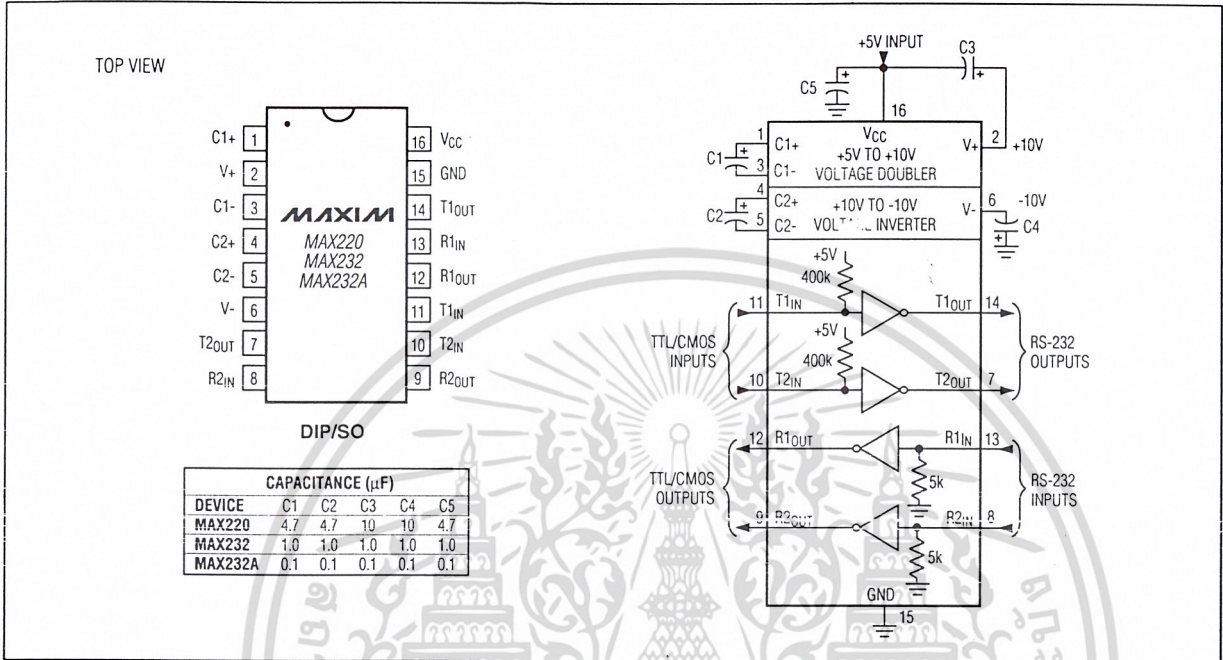


Figure 5. MAX220/MAX232/MAX232A Pin Configuration and Typical Operating Circuit

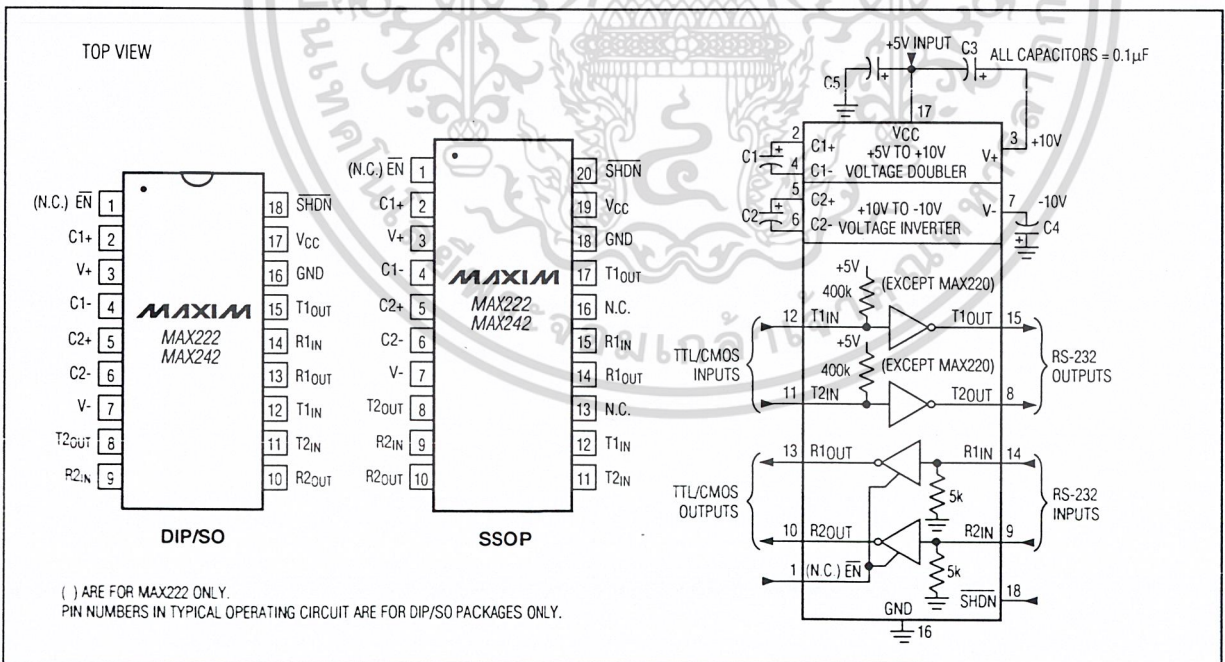


Figure 6. MAX222/MAX242 Pin Configurations and Typical Operating Circuit

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

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

Ordering Information (continued)

MAX220-MAX249

PART	TEMP. RANGE	PIN-PACKAGE
MAX222CPN	0°C to +70°C	18 Plastic DIP
MAX222CWN	0°C to +70°C	18 Wide SO
MAX222C/D	0°C to +70°C	Dice*
MAX222EPN	-40°C to +85°C	18 Plastic DIP
MAX222EWN	-40°C to +85°C	18 Wide SO
MAX222EJN	-40°C to +85°C	18 CERDIP
MAX222MJN	-55°C to +125°C	18 CERDIP
MAX223CAI	0°C to +70°C	28 SSOP
MAX223CWI	0°C to +70°C	28 Wide SO
MAX223C/D	0°C to +70°C	Dice*
MAX223EAI	-40°C to +85°C	28 SSOP
MAX223EWI	-40°C to +85°C	28 Wide SO
MAX225CWI	0°C to +70°C	28 Wide SO
MAX225EWI	-40°C to +85°C	28 Wide SO
MAX230CPP	0°C to +70°C	20 Plastic DIP
MAX230CWP	0°C to +70°C	20 Wide SO
MAX230C/D	0°C to +70°C	Dice*
MAX230EPP	-40°C to +85°C	20 Plastic DIP
MAX230EWP	-40°C to +85°C	20 Wide SO
MAX230EJP	-40°C to +85°C	20 CERDIP
MAX230MJP	-55°C to +125°C	20 CERDIP
MAX231CPD	0°C to +70°C	14 Plastic DIP
MAX231CWE	0°C to +70°C	16 Wide SO
MAX231CJD	0°C to +70°C	14 CERDIP
MAX231C/D	0°C to +70°C	Dice*
MAX231EPD	-40°C to +85°C	14 Plastic DIP
MAX231EWE	-40°C to +85°C	16 Wide SO
MAX231EJD	-40°C to +85°C	14 CERDIP
MAX231MJD	-55°C to +125°C	14 CERDIP
MAX232CPE	0°C to +70°C	16 Plastic DIP
MAX232CSE	0°C to +70°C	16 Narrow SO
MAX232CWE	0°C to +70°C	16 Wide SO
MAX232C/D	0°C to +70°C	Dice*
MAX232EPE	-40°C to +85°C	16 Plastic DIP
MAX232ESE	-40°C to +85°C	16 Narrow SO
MAX232EWE	-40°C to +85°C	16 Wide SO
MAX232EJE	-40°C to +85°C	16 CERDIP
MAX232MJE	-55°C to +125°C	16 CERDIP
MAX232MLP	-55°C to +125°C	20 LCC
MAX232ACPE	0°C to +70°C	16 Plastic DIP
MAX232ACSE	0°C to +70°C	16 Narrow SO
MAX232ACWE	0°C to +70°C	16 Wide SO

MAX232AC/D	0°C to +70°C	Dice*
MAX232AEPE	-40°C to +85°C	16 Plastic DIP
MAX232AESE	-40°C to +85°C	16 Narrow SO
MAX232AEWE	-40°C to +85°C	16 Wide SO
MAX232AEJE	-40°C to +85°C	16 CERDIP
MAX232AMJE	-55°C to +125°C	16 CERDIP
MAX232AML	-55°C to +125°C	20 LCC
MAX233CPP	0°C to +70°C	20 Plastic DIP
MAX233EPP	-40°C to +85°C	20 Plastic DIP
MAX233ACPP	0°C to +70°C	20 Plastic DIP
MAX233ACWP	0°C to +70°C	20 Wide SO
MAX233AEPP	-40°C to +85°C	20 Plastic DIP
MAX233AEWP	-40°C to +85°C	20 Wide SO
MAX234CPE	0°C to +70°C	16 Plastic DIP
MAX234CWE	0°C to +70°C	16 Wide SO
MAX234C/D	0°C to +70°C	Dice*
MAX234EPE	-40°C to +85°C	16 Plastic DIP
MAX234EWE	-40°C to +85°C	16 Wide SO
MAX234EJE	-40°C to +85°C	16 CERDIP
MAX234MJE	-55°C to +125°C	16 CERDIP
MAX235CPG	0°C to +70°C	24 Wide Plastic DIP
MAX235EPG	-40°C to +85°C	24 Wide Plastic DIP
MAX235EDG	-40°C to +85°C	24 Ceramic SB
MAX235MDG	-55°C to +125°C	24 Ceramic SB
MAX236CNG	0°C to +70°C	24 Narrow Plastic DIP
MAX236CWG	0°C to +70°C	24 Wide SO
MAX236C/D	0°C to +70°C	Dice*
MAX236ENG	-40°C to +85°C	24 Narrow Plastic DIP
MAX236EWG	-40°C to +85°C	24 Wide SO
MAX236ERG	-40°C to +85°C	24 Narrow CERDIP
MAX236MRG	-55°C to +125°C	24 Narrow CERDIP
MAX237CNG	0°C to +70°C	24 Narrow Plastic DIP
MAX237CWG	0°C to +70°C	24 Wide SO
MAX237C/D	0°C to +70°C	Dice*
MAX237ENG	-40°C to +85°C	24 Narrow Plastic DIP
MAX237EWG	-40°C to +85°C	24 Wide SO
MAX237ERG	-40°C to +85°C	24 Narrow CERDIP
MAX237MRG	-55°C to +125°C	24 Narrow CERDIP
MAX238CNG	0°C to +70°C	24 Narrow Plastic DIP
MAX238CWG	0°C to +70°C	24 Wide SO
MAX238C/D	0°C to +70°C	Dice*
MAX238ENG	-40°C to +85°C	24 Narrow Plastic DIP

* Contact factory for dice specifications.

8-bit MCU with 32K byte Flash, 10 bit A/D and EEPROM

1. Description

The T89C51AC2 is a high performance FLASH version of the 80C51 single chip 8-bit microcontrollers. It contains a 32Kbyte Flash memory block for program and data.

The 32K byte FLASH memory can be programmed either in parallel mode or in serial mode with the ISP capability or with software. The programming voltage is internally generated from the standard V_{CC} pin.

The T89C51AC2 retains all features of the 80C52 with 256 bytes of internal RAM, a 7-source 4-level interrupt controller and three timer/counters.

In addition, the T89C51AC2 has a 10 bit A/D converter, a 2Kbytes Boot Flash memory, 2 Kbyte EEPROM for data, a Programmable Counter Array, an XRAM of 1024 bytes, a Hardware Watchdog Timer and a more versatile serial channel that facilitates multiprocessor communication (EUART).

The fully static design of the T89C51AC2 allows to reduce system power consumption by bringing the clock frequency down to any value, even DC, without loss of data.

The T89C51AC2 has 2 software-selectable modes of reduced activity and an 8 bit clock prescaler for further reduction in power consumption. In the idle mode the CPU is frozen while the peripherals and the interrupt system are still operating. In the power-down mode the RAM is saved and all other functions are inoperative.

The added features of the T89C51AC2 make it more powerful for applications that need A/D conversion, pulse width modulation, high speed I/O and counting capabilities such as industrial control, consumer goods, alarms, motor control, ...

While remaining fully compatible with the 80C52 it offers a superset of this standard microcontroller. In X2 mode a maximum external clock rate of 20 MHz reaches a 300 ns cycle time.

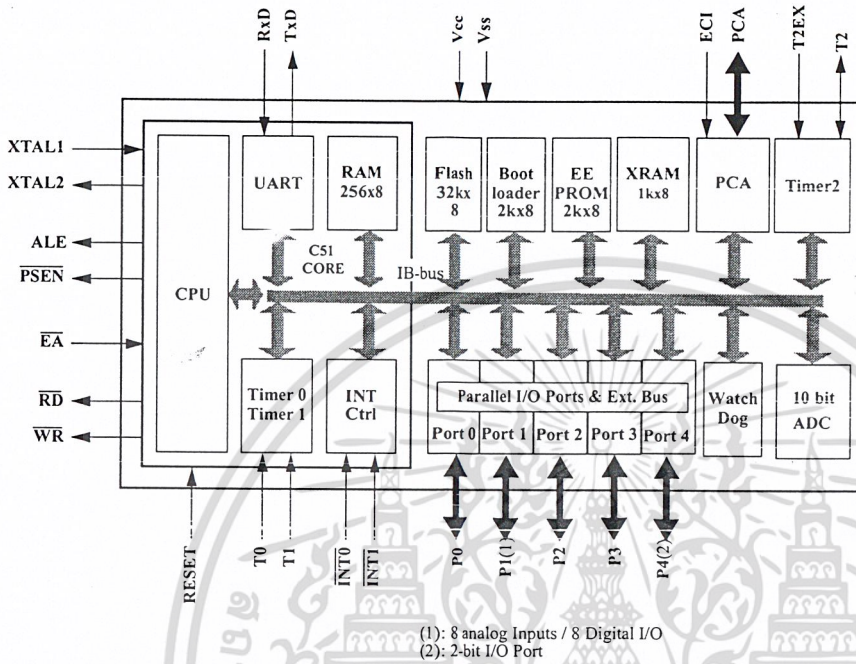
2. Features

- 80C52 core architecture
 - Double Data Pointer
 - 256 bytes of on-chip RAM
- 1Kbytes of on-chip XRAM
- 32 Kbytes of on-chip Flash memory
- 2 Kbytes of on-chip Flash for Bootloader
- 2 Kbytes of on-chip EEPROM
- 14 interrupt sources with 4 priority level
- Three 16-bit timer/counters
- Full duplex Enhanced UART
- High speed architecture
 - 40 MHz in standard mode
 - 20 MHz in X2 mode (6 clocks/machine cycle)
- Five ports: 32 + 2 digital I/O lines
- Programmable Counter Array with 5 16-bit channels :
 - High-speed output
 - Compare / Capture
- Pulse Width Modulator
- Watchdog Timer (Channel 4)
- Hardware watchdog timer (One-time enabled with Reset-out)
- A 10-bit resolution analog to digital converter (ADC) with 8 multiplexed inputs
 - 20 microsecond conversion time
 - Two conversion modes
- Asynchronous port reset
- Low EMI (inhibit ALE)
- Power control modes:
 - Idle mode
 - Power down mode
- Power supply: 4.5V to 5.5V
- Temperature range: Industrial (-40 to +85C)
- Packages: VQFP44 1,4 mm, PLCC44, CA-BGA64*
* on request

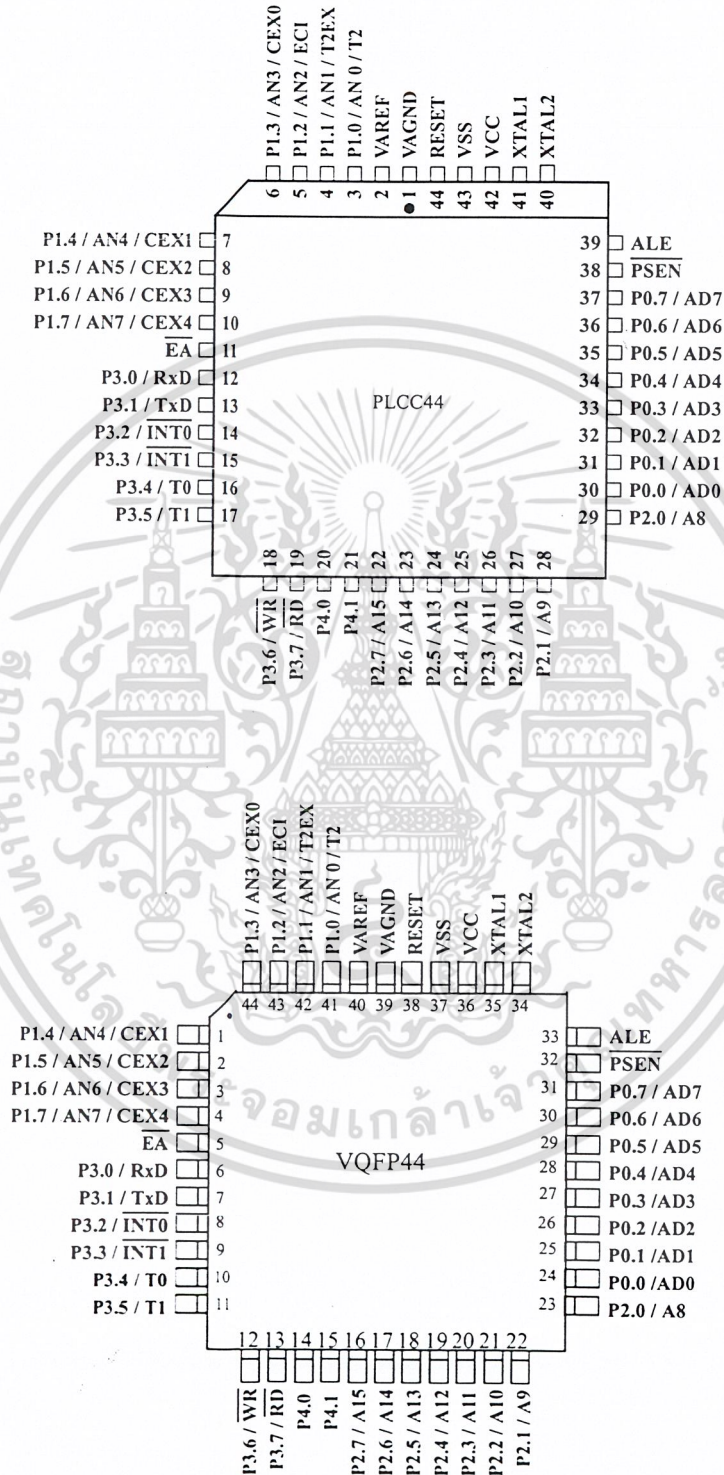
T89C51AC2



3. Block Diagram



4. Pin Configuration



Preliminary

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานหรือการศึกษเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Table 1. Pin Description

Pin Name	Pin Number		Type	Description
	PLCC44	VQFP44		
P0.0-P0.7	30-37	24-31	I/O	<p>Port 0:</p> <p>Port 0 is an 8-bit open drain bi-directional I/O port. Port 0 pins that have 1's written to them float, and in this state can be used as high-impedance inputs. Port 0 is also the multiplexed low-order address and data bus during accesses to external Program and Data Memory. In this application it uses strong internal pull-ups when emitting 1's. Port 0 also outputs the code bytes during program validation. External pull-ups are required during program verification. In the T89C51AC2 Port 0 can sink or source 5mA. It can drive CMOS inputs without external pull-ups.</p>
P1.0-P1.7	3-10	41-44, 1-4	I/O	<p>Port 1:</p> <p>Port 1 is an 8-bit bi-directional I/O port with internal pull-ups. Port 1 pins can be used for digital input/output or as analog inputs for the Analog Digital Converter (ADC). Port 1 pins that have 1's written to them are pulled high by the internal pull-up transistors and can be used as inputs in this state. As inputs, Port 1 pins that are being pulled low externally will be the source of current (IIL, on the datasheet) because of the internal pull-ups. Port 1 pins are assigned to be used as analog inputs via the ADCCF register.</p> <p>As a secondary digital function, port 1 contains the Timer 2 external trigger and clock input; the PCA external clock input and the PCA module I/O.</p> <p>Port 1 receives the low-order address byte during FLASH programming and program verification. In the T89C51AC2 Port 1 can sink or source 5mA. It can drive CMOS inputs without external pull-ups.</p>
	3	41	I/O	<p>P1.0 : Input / Output</p>
			I	<p>AN0 : Analog input channel 0</p>
			I	<p>T2 : External clock input for Timer/counter2</p>
	4	42	I/O	<p>P1.1 : Input / Output</p>
			I	<p>AN1 : Analog input channel 1</p>
			I	<p>T2EX : Trigger input for Timer/counter2</p>
	5	43	I/O	<p>P1.2 : Input / Output</p>
			I	<p>AN2 : Analog input channel 2</p>
			I	<p>ECI : PCA external clock input</p>
	6	44	I/O	<p>P1.3 : Input / Output</p>
			I	<p>AN3 : Analog input channel 3</p>
			O	<p>CEX0 : PCA module 0 Entry of input/PWM output</p>
	7	1	I/O	<p>P1.4 : Input / Output</p>
			I	<p>AN4 : Analog input channel 4</p>
			O	<p>CEX1 : PCA module 1 Entry of input/PWM output</p>
	8	2	I/O	<p>P1.5 : Input / Output</p>
			I	<p>AN5 : Analog input channel 5</p>
			O	<p>CEX2 : PCA module 2 Entry of input/PWM output</p>
	9	3	I/O	<p>P1.6 : Input / Output</p>
			I	<p>AN6 : Analog input channel 6</p>
			O	<p>CEX3 : PCA module 3 Entry of input/PWM output</p>
	10	4	I/O	<p>P1.7 : Input / Output</p>
			I	<p>AN7 : Analog input channel 7</p>
			O	<p>CEX4 : PCA module 4 Entry of input/PWM output</p>

T89C51AC2



Pin Name	Pin Number		Type	Description
	PLCC44	VQFP44		
RESET	44	38	I/O	Reset: A high level on this pin during two machine cycles while the oscillator is running resets the device. An internal pull-down resistor to VSS permits power-on reset using only an external capacitor to VCC.
ALE	39	33	O	ALE: An Address Latch Enable output for latching the low byte of the address during accesses to the external memory. The ALE is activated every 1/6 oscillator periods (1/3 in X2 mode) except during an external data memory access. When instructions are executed from an internal FLASH ($\overline{EA} = 1$), ALE generation can be disabled by the software.
\overline{PSEN}	38	32	O	PSEN: The Program Store Enable output is a control signal that enables the external program memory of the bus during external fetch operations. It is activated twice each machine cycle during fetches from the external program memory. (However, when executing outside of the external program memory two activations of PSEN are skipped during each access to the external Data memory). The PSEN is not activated during fetches from the internal data memory.
\overline{EA}	11	5	I	EA: When External Access is held at the high level, instructions are fetched from the internal FLASH when the program counter is less than 8000H. When held at the low level, T89C51AC2 fetches all instructions from the external program memory.
XTAL1	41	35	I	XTAL1: Input of the inverting oscillator amplifier and input of the internal clock generator circuits. To drive the device from an external clock source, XTAL1 should be driven, while XTAL2 is left unconnected. To operate above a frequency of 16 MHz, a duty cycle of 50% should be maintained.
XTAL2	40	34	O	XTAL2: Output from the inverting oscillator amplifier.
VAGND	1	39	I	Reference Ground for ADC
VCC	42	36	I	Supply voltage during normal, idle, and power-down operation.
VSS	43	37	I	Circuit ground potential.
VAREF	2	40	I	Reference Voltage for ADC



ภาคผนวก ข.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

ADCON EQU 0F3H
ADCF EQU 0F6H
ADCLK EQU 0F2H
ADDH EQU 0F5H
ADDL EQU 0F4H

```

```

;////////////////////////////////////

```

```

TH2 EQU 0CDH
TL2 EQU 0CCH
T2CON EQU 0C8H
T2MOD EQU 0C9H
RCAP2H EQU 0CBH
RCAP2L EQU 0CAH

```

```

CKCON EQU 08FH

```

```

ORG 0000H

```

```

LJMP MAIN

```

```

MAIN : MOV ADCF, #01H ; CH 0-7, '0000 0001'
MOV ADCON, #20H ; EN ADC, 00'1'0 0000 SETB ADEN
;////////////////////////////////////
MOV A, T2CON
ORL A, #00110000B
MOV T2CON, A

; MOV A, CKCON
; ANL A, #00000001B
; MOV CKCON, A ; 6 CLOCK
MOV T2MOD, #20H
MOV SCON, #50H
; MOV PCON, #80H
MOV TH2, #0FFH
MOV TL2, #0F6H
MOV RCAP2H, #0FFH
MOV RCAP2L, #0FBH ; EC = 57600 FB = 230400
; SETB TR1

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

MOV    A, T2CON
ORL    A, #00000100B
MOV    T2CON, A

SEQUEN : MOV    A, ADCON        ; SELEC ANALOG I/P
        ANL    A, #0F8H        ; 1111 1'000' CLR &or SELC CH 0
        MOV    ADCON, A

LOOP_ADC : MOV    A, ADCON        ; SELEC CHANNAL 0 & START ADC
        ORL    A, #08H
        MOV    ADCON, A

LOOP :   MOV    A, ADCON        ; JNB ADEOC, LOOP , IF xxx0 xxxx IS xxx1 x
xxx DO NEXT
        ANL    A, #10H        ; { SET BY HARDWARE }
        CJNE   A, #10H, LOOP
        MOV    A, ADCON        ; CLR ADEOC ( CLEARED BY SOFTWARE )
        ANL    A, #0EFH
        MOV    ADCON, A

        MOV    A, ADDH        ; GET ADC

        MOV    SBUF, A        ; SEND RS232

JNB    TI, $
        CLR    TI

        NOP
        NOP
        NOP
        NOP

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

NOP

NOP

NOP

NOP

NOP

NOP

LJMP LOOP_ADC

END



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
using System;
using System.Reflection;
using System.Drawing;
using System.Collections;
using System.ComponentModel;
using System.Windows.Forms;
using System.Data;
using System.IO;
using System.Text;
using System.Threading;
using OpenNETCF;
using OpenNETCF.IO.Serial;
```

```
namespace PocketReader
{
    /// <summary>
    /// Summary description for Form1.
    /// </summary>
    public class frmMain : System.Windows.Forms.Form
    {
        private System.Windows.Forms.TabControl tcMain;
        private System.Windows.Forms.TabPage tpStartup;
        private System.Windows.Forms.TabPage tpTimeDomain;
        private System.Windows.Forms.Panel pnlPort;
        private System.Windows.Forms.Label label1;
        private System.Windows.Forms.Label label2;
        private System.Windows.Forms.Label label3;
        private System.Windows.Forms.Label label4;
        private System.Windows.Forms.Label label5;
        private System.Windows.Forms.Label label6;
        private System.Windows.Forms.Panel panel1;
        private System.Windows.Forms.TextBox txtComPort;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
private System.Windows.Forms.TextBox txtBaudRate;
private System.Windows.Forms.TextBox txtDataBit;
private System.Windows.Forms.ComboBox cbParityBit;
private System.Windows.Forms.TextBox txtStopBit;
private System.Windows.Forms.Label label7;
private System.Windows.Forms.Label label8;
private System.Windows.Forms.Label label9;
private System.Windows.Forms.TextBox txtRecordTime;
private System.Windows.Forms.TextBox txtSamplingRate;
private System.Windows.Forms.TextBox txtDataReceive;
private System.Windows.Forms.Button btnStart;
private System.Windows.Forms.Button btnStop;
private Mooseworks.XYGraph GraphFreq;
private System.Windows.Forms.Button button1;
private Mooseworks.XYGraph GraphTime;
private System.Windows.Forms.Button btnCalFFT;
private System.Windows.Forms.TabPage tpFreqDomain;
private System.Windows.Forms.Timer tmRecord;

private TDataBuffer dt;
private int ReceiveCount;
public static Port port;
private System.Windows.Forms.Button button2;
private System.Windows.Forms.Button btnExit;
private System.Windows.Forms.MainMenu mainMenu1;
private System.Windows.Forms.MenuItem menuItem1;
private System.Windows.Forms.MenuItem menuItem2;
private System.Windows.Forms.MenuItem menuItem3;
private System.Windows.Forms.MenuItem menuItem4;
private System.Windows.Forms.Label lbFreq;

private DetailedPortSettings portSettings;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

public frmMain()
{
    //
    // Required for Windows Form Designer support
    //
    InitializeComponent();

    //
    // TODO: Add any constructor code after InitializeComponent call
    //

    // create the port settings
    portSettings = new HandshakeNone();

    // create a default port on COM2 with no handshaking
    port = new Port("COM1:", portSettings);

    // these are fairly inefficient settings
    // for a terminal we want to send/receive all data as it is queued
    // so 1 byte at a time is the best way to go
    port.RThreshold = 16; // get an event for every 16 bytes received
    port.InputLen = 16; // calling Input will read 16 bytes
    port.SThreshold = 1; // send 1 byte at a time

    port.DataReceived += new Port.CommEvent(port_DataReceived);

    cbParityBit.SelectedIndex = 0;

    tmRecord.Tick += new EventHandler(tmRecord_Tick);

    tcMain.SelectedIndexChanged += new
EventHandlder(tcMain_SelectedIndexChanged);

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

        ReceiveCount = 0;

        dt = new TDataBuffer(65536);
    }

    /// <summary>
    /// Clean up any resources being used.
    /// </summary>
    protected override void Dispose( bool disposing )
    {
        base.Dispose( disposing );
    }

    #region Windows Form Designer generated code
    /// <summary>
    /// Required method for Designer support - do not modify
    /// the contents of this method with the code editor.
    /// </summary>
    private void InitializeComponent()
    {
        this.tcMain = new System.Windows.Forms.TabControl();
        this.tpStartup = new System.Windows.Forms.TabPage();
        this.panel1 = new System.Windows.Forms.Panel();
        this.btnExit = new System.Windows.Forms.Button();
        this.btnStop = new System.Windows.Forms.Button();
        this.btnStart = new System.Windows.Forms.Button();
        this.txtDataReceive = new System.Windows.Forms.TextBox();
        this.txtSamplingRate = new System.Windows.Forms.TextBox();
        this.txtRecordTime = new System.Windows.Forms.TextBox();
        this.label9 = new System.Windows.Forms.Label();
        this.label8 = new System.Windows.Forms.Label();
        this.label7 = new System.Windows.Forms.Label();
        this.pnlPort = new System.Windows.Forms.Panel();
        this.txtStopBit = new System.Windows.Forms.TextBox();
        this.cbParityBit = new System.Windows.Forms.ComboBox();
        this.txtDataBit = new System.Windows.Forms.TextBox();
    }

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

this.txtBaudRate = new System.Windows.Forms.TextBox();
this.txtComPort = new System.Windows.Forms.TextBox();
this.label6 = new System.Windows.Forms.Label();
this.label5 = new System.Windows.Forms.Label();
this.label4 = new System.Windows.Forms.Label();
this.label3 = new System.Windows.Forms.Label();
this.label2 = new System.Windows.Forms.Label();
this.label1 = new System.Windows.Forms.Label();
this.tpTimeDomain = new System.Windows.Forms.TabPage();
this.button2 = new System.Windows.Forms.Button();
this.GraphTime = new Mooseworks.XYGraph();
this.button1 = new System.Windows.Forms.Button();
this.tpFreqDomain = new System.Windows.Forms.TabPage();
this.lbFreq = new System.Windows.Forms.Label();
this.btnCalFFT = new System.Windows.Forms.Button();
this.GraphFreq = new Mooseworks.XYGraph();
this.tmRecord = new System.Windows.Forms.Timer();
this.mainMenu1 = new System.Windows.Forms.MainMenu();
this.menuItem1 = new System.Windows.Forms.MenuItem();
this.menuItem2 = new System.Windows.Forms.MenuItem();
this.menuItem3 = new System.Windows.Forms.MenuItem();
this.menuItem4 = new System.Windows.Forms.MenuItem();
//
// tcMain
//
this.tcMain.Controls.Add(this.tpStartup);
this.tcMain.Controls.Add(this.tpTimeDomain);
this.tcMain.Controls.Add(this.tpFreqDomain);
this.tcMain.SelectedIndex = 0;
this.tcMain.Size = new System.Drawing.Size(240, 256);
//
// tpStartup
//

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

this.tpStartup.Controls.Add(this.panel1);
this.tpStartup.Controls.Add(this.pnlPort);
this.tpStartup.Location = new System.Drawing.Point(4, 4);
this.tpStartup.Size = new System.Drawing.Size(232, 230);
this.tpStartup.Text = "Startup";
//
// panel1
//
this.panel1.BackColor =
System.Drawing.Color.FromArgb(((System.Byte)(192)), ((System.Byte)(192)),
((System.Byte)(255)));
this.panel1.Controls.Add(this.btnExit);
this.panel1.Controls.Add(this.btnStop);
this.panel1.Controls.Add(this.btnStart);
this.panel1.Controls.Add(this.txtDataReceive);
this.panel1.Controls.Add(this.txtSamplingRate);
this.panel1.Controls.Add(this.txtRecordTime);
this.panel1.Controls.Add(this.label9);
this.panel1.Controls.Add(this.label8);
this.panel1.Controls.Add(this.label7);
this.panel1.Location = new System.Drawing.Point(8, 150);
this.panel1.Size = new System.Drawing.Size(216, 80);
//
// btnExit
//
this.btnExit.Location = new System.Drawing.Point(160, 53);
this.btnExit.Size = new System.Drawing.Size(48, 24);
this.btnExit.Text = "Exit";
this.btnExit.Click += new System.EventHandler(this.btnExit_Click);
//
// btnStop
//
this.btnStop.Location = new System.Drawing.Point(160, 28);

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

this.btnStop.Size = new System.Drawing.Size(48, 24);
this.btnStop.Text = "Stop";
this.btnStop.Click += new System.EventHandler(this.btnStop_Click);
//
// btnStart
//
this.btnStart.Location = new System.Drawing.Point(160, 3);
this.btnStart.Size = new System.Drawing.Size(48, 24);
this.btnStart.Text = "Start";
this.btnStart.Click += new System.EventHandler(this.btnStart_Click);
//
// txtDataReceive
//
this.txtDataReceive.Location = new System.Drawing.Point(88, 46);
this.txtDataReceive.Size = new System.Drawing.Size(48, 20);
this.txtDataReceive.Text = "0";
//
// txtSamplingRate
//
this.txtSamplingRate.Location = new System.Drawing.Point(88, 24);
this.txtSamplingRate.Size = new System.Drawing.Size(48, 20);
this.txtSamplingRate.Text = "4800";
//
// txtRecordTime
//
this.txtRecordTime.Location = new System.Drawing.Point(88, 3);
this.txtRecordTime.Size = new System.Drawing.Size(32, 20);
this.txtRecordTime.Text = "0";
//
// label9
//
this.label9.Location = new System.Drawing.Point(4, 49);
this.label9.Size = new System.Drawing.Size(85, 16);

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

this.label9.Text = "Data Received :";
//
// label8
//
this.label8.Location = new System.Drawing.Point(3, 27);
this.label8.Size = new System.Drawing.Size(85, 16);
this.label8.Text = "SamplingRate :";
//
// label7
//
this.label7.Location = new System.Drawing.Point(8, 4);
this.label7.Size = new System.Drawing.Size(72, 16);
this.label7.Text = "RecordTime :";
//
// pnlPort
//
this.pnlPort.BackColor =
System.Drawing.Color.FromArgb(((System.Byte)(192)), ((System.Byte)(192)),
((System.Byte)(255)));
this.pnlPort.Controls.Add(this.txtStopBit);
this.pnlPort.Controls.Add(this.cbParityBit);
this.pnlPort.Controls.Add(this.txtDataBit);
this.pnlPort.Controls.Add(this.txtBaudRate);
this.pnlPort.Controls.Add(this.txtComPort);
this.pnlPort.Controls.Add(this.label6);
this.pnlPort.Controls.Add(this.label5);
this.pnlPort.Controls.Add(this.label4);
this.pnlPort.Controls.Add(this.label3);
this.pnlPort.Controls.Add(this.label2);
this.pnlPort.Controls.Add(this.label1);
this.pnlPort.Location = new System.Drawing.Point(8, 8);
this.pnlPort.Size = new System.Drawing.Size(216, 136);
//

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

// txtStopBit
//
this.txtStopBit.Location = new System.Drawing.Point(72, 106);
this.txtStopBit.Size = new System.Drawing.Size(40, 20);
this.txtStopBit.Text = "1";
//
// cbParityBit
//
this.cbParityBit.Items.Add("NONE");
this.cbParityBit.Items.Add("ODD");
this.cbParityBit.Items.Add("EVEN");
this.cbParityBit.Location = new System.Drawing.Point(72, 85);
this.cbParityBit.Size = new System.Drawing.Size(80, 21);
//
// txtDataBit
//
this.txtDataBit.Location = new System.Drawing.Point(72, 63);
this.txtDataBit.Size = new System.Drawing.Size(40, 20);
this.txtDataBit.Text = "8";
//
// txtBaudRate
//
this.txtBaudRate.Location = new System.Drawing.Point(72, 41);
this.txtBaudRate.Size = new System.Drawing.Size(80, 20);
this.txtBaudRate.Text = "57600";
//
// txtComPort
//
this.txtComPort.Location = new System.Drawing.Point(72, 19);
this.txtComPort.Size = new System.Drawing.Size(80, 20);
this.txtComPort.Text = "COM1:";
//
// label6

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

//
this.label6.Location = new System.Drawing.Point(16, 108);
this.label6.Size = new System.Drawing.Size(56, 16);
this.label6.Text = "StopBit :";
//
// label5
//
this.label5.Location = new System.Drawing.Point(16, 88);
this.label5.Size = new System.Drawing.Size(56, 16);
this.label5.Text = "ParityBit :";
//
// label4
//
this.label4.Location = new System.Drawing.Point(16, 65);
this.label4.Size = new System.Drawing.Size(48, 16);
this.label4.Text = "DataBit :";
//
// label3
//
this.label3.Location = new System.Drawing.Point(8, 43);
this.label3.Size = new System.Drawing.Size(64, 16);
this.label3.Text = "BaudRate :";
//
// label2
//
this.label2.Location = new System.Drawing.Point(16, 21);
this.label2.Size = new System.Drawing.Size(56, 16);
this.label2.Text = "Comport :";
//
// label1
//
this.label1.ForeColor =
System.Drawing.Color.FromArgb(((System.Byte)(0)), ((System.Byte)(0)), ((System.Byte)(192)));

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

this.label1.Size = new System.Drawing.Size(216, 16);
this.label1.Text = "Comport Selection";
this.label1.TextAlign = System.Drawing.ContentAlignment.TopCenter;
//
// tpTimeDomain
//
this.tpTimeDomain.Controls.Add(this.button2);
this.tpTimeDomain.Controls.Add(this.GraphTime);
this.tpTimeDomain.Controls.Add(this.button1);
this.tpTimeDomain.Location = new System.Drawing.Point(4, 4);
this.tpTimeDomain.Size = new System.Drawing.Size(232, 230);
this.tpTimeDomain.Text = "Time Domain";
//
// button2
//
this.button2.Location = new System.Drawing.Point(27, 206);
this.button2.Size = new System.Drawing.Size(89, 24);
this.button2.Text = "Time Domain";
this.button2.Click += new System.EventHandler(this.button2_Click);
//
// GraphTime
//
this.GraphTime.AutoScale = Mooseworks.AutoScale.Both;
this.GraphTime.Caption = "";
this.GraphTime.Graphs = new Mooseworks.Graph[] {

        new Mooseworks.Graph(System.Drawing.Color.Red, true),

        new Mooseworks.Graph(System.Drawing.Color.RoyalBlue,
true),

        new Mooseworks.Graph(System.Drawing.Color.Purple, true),

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

new Mooseworks.Graph(System.Drawing.Color.Turquoise,
true),

new Mooseworks.Graph(System.Drawing.Color.Magenta,
true),

new Mooseworks.Graph(System.Drawing.Color.Aqua, true),

new Mooseworks.Graph(System.Drawing.Color.CadetBlue,
true),

new
Mooseworks.Graph(System.Drawing.Color.LightSlateGray, true));
this.GraphTime.MajorGrid.Color = System.Drawing.Color.Gray;
this.GraphTime.MajorGrid.ControlRef = null;
this.GraphTime.MajorGrid.Visible = false;
this.GraphTime.MinorGrid.Color = System.Drawing.Color.Gray;
this.GraphTime.MinorGrid.ControlRef = null;
this.GraphTime.MinorGrid.Visible = false;
this.GraphTime.PlotBackgroundColor = System.Drawing.Color.Linen;
this.GraphTime.Size = new System.Drawing.Size(232, 208);
this.GraphTime.XAxis = new Mooseworks.Axis(0, 100, 4, 3,
System.Drawing.Color.Red, System.Drawing.Color.Black, System.Drawing.Color.Black, new
System.Drawing.Font(" Arial", 8F, System.Drawing.FontStyle.Regular), 1, "",
Mooseworks.AxisPlacement.Bottom);

this.GraphTime.YAxis = new Mooseworks.Axis(-5, 5, 4, 3,
System.Drawing.Color.Red, System.Drawing.Color.Black, System.Drawing.Color.Black, new
System.Drawing.Font(" Arial", 8F, System.Drawing.FontStyle.Regular), 1, "",
Mooseworks.AxisPlacement.Left);

//
// button1
//

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

this.button1.Location = new System.Drawing.Point(168, 209);
this.button1.Size = new System.Drawing.Size(48, 24);
this.button1.Text = "load";
this.button1.Visible = false;
this.button1.Click += new System.EventHandler(this.button1_Click);
//
// tpFreqDomain
//
this.tpFreqDomain.Controls.Add(this.lbFreq);
this.tpFreqDomain.Controls.Add(this.btnCalFFT);
this.tpFreqDomain.Controls.Add(this.GraphFreq);
this.tpFreqDomain.Location = new System.Drawing.Point(4, 4);
this.tpFreqDomain.Size = new System.Drawing.Size(232, 230);
this.tpFreqDomain.Text = "Frequency Domain";
//
// lbFreq
//
this.lbFreq.Location = new System.Drawing.Point(123, 211);
this.lbFreq.Size = new System.Drawing.Size(72, 16);
this.lbFreq.Text = "0.00";
//
// btnCalFFT
//
this.btnCalFFT.Location = new System.Drawing.Point(24, 208);
this.btnCalFFT.Size = new System.Drawing.Size(80, 24);
this.btnCalFFT.Text = "FFT";
this.btnCalFFT.Click += new
System.EventHandler(this.btnCalFFT_Click);
//
// GraphFreq
//
this.GraphFreq.AutoScale = Mooseworks.AutoScale.Both;
this.GraphFreq.Caption = "";

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

this.GraphFreq.Graphs = new Mooseworks.Graph[] {

    new Mooseworks.Graph(System.Drawing.Color.Red, true),

    new Mooseworks.Graph(System.Drawing.Color.RoyalBlue,
true),

    new Mooseworks.Graph(System.Drawing.Color.Purple, true),

    new Mooseworks.Graph(System.Drawing.Color.Turquoise,
true),

    new Mooseworks.Graph(System.Drawing.Color.Magenta,
true),

    new Mooseworks.Graph(System.Drawing.Color.Aqua, true),

    new Mooseworks.Graph(System.Drawing.Color.CadetBlue,
true),

    new
Mooseworks.Graph(System.Drawing.Color.LightSlateGray, true)};
this.GraphFreq.MajorGrid.Color = System.Drawing.Color.Gray;
this.GraphFreq.MajorGrid.ControlRef = null;
this.GraphFreq.MajorGrid.Visible = false;
this.GraphFreq.MinorGrid.Color = System.Drawing.Color.Gray;
this.GraphFreq.MinorGrid.ControlRef = null;
this.GraphFreq.MinorGrid.Visible = false;
this.GraphFreq.PlotBackgroundColor = System.Drawing.Color.Linen;
this.GraphFreq.Size = new System.Drawing.Size(232, 208);
this.GraphFreq.XAxis = new Mooseworks.Axis(0, 100, 4, 3,
System.Drawing.Color.Red, System.Drawing.Color.Black, System.Drawing.Color.Black, new

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

System.Drawing.Font("Arial", 8F, System.Drawing.FontStyle.Regular), 1, "",
Mooseworks.AxisPlacement.Bottom);

        this.GraphFreq.YAxis = new Mooseworks.Axis(0, 10, 4, 3,
System.Drawing.Color.Red, System.Drawing.Color.Black, System.Drawing.Color.Black, new
System.Drawing.Font("Arial", 8F, System.Drawing.FontStyle.Regular), 1, "",
Mooseworks.AxisPlacement.Left);

//

// tmRecord

//

this.tmRecord.Interval = 1000;

//

// mainMenu1

//

this.mainMenu1.MenuItems.Add(this.menuItem1);

//

// menuItem1

//

this.menuItem1.MenuItems.Add(this.menuItem2);
this.menuItem1.MenuItems.Add(this.menuItem3);
this.menuItem1.MenuItems.Add(this.menuItem4);
this.menuItem1.Text = "File";

//

// menuItem2

//

this.menuItem2.Text = "";

//

// menuItem3

//

this.menuItem3.Text = "";

//

// menuItem4

//

this.menuItem4.Text = "Exit";

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

        this.menuItem4.Click += new
System.EventHandler(this.menuItem4_Click_1);

        this.FormBorderStyle =
System.Windows.Forms.FormBorderStyle.Sizable;

        //
        // frmMain
        //

        this.ClientSize = new System.Drawing.Size(240, 253);
        this.Controls.Add(this.tcMain);
        this.Menu = this.mainMenu1;
        this.Text = "PocketReader";

    }
#endregion

/// <summary>
/// The main entry point for the application.
/// </summary>
static void Main()
{

    Application.Run(new frmMain());

}

private double ConvertToVolt(int v)
{

    double ret = 0.0;

    int b = v;

    //-5 -> +5

    double step = (double)(10) / (double)(256);

    ret = (b * step) + (-5);

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้.

```

        return ret;
    }

    private void button1_Click(object sender, System.EventArgs e)
    {
        // load data
        Module[] moduleArray;

        moduleArray = Assembly.GetExecutingAssembly().GetModules();
        // In a simple project with only one module, the module at index
        // 0 will be the module containing this class.
        Module myModule = moduleArray[0];
        string fileName = myModule.FullyQualifiedName;
        string filePath = System.IO.Path.GetDirectoryName(fileName);
        dt.LoadFromFile(filePath+"\\DataFile.dat");

        int segment = 0;
        int segment_limit = 0;
        int len = dt.Length();
        if(len > 100)
        {
            segment_limit = len / 100;
            for(int s=0;s<len;s+=100)
            {
                if(segment < segment_limit)
                {
                    segment++;

                    double[,] data = new double[2,100];
                    for (int i=0; i<100; i++)
                    {
                        if( (i+s) < len)
                        {
                            data[0,i] = i;

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้


```

{
    datai[i].real = (float)dt.Buffer[i];
    datai[i].imaginary = 0;
}
MathEx.FFT fft = new MathEx.FFT();

```

```
Complex[] res = fft.Calculate(datai,false);
```

```
len = res.Length;
```

```

double[] fftgraph = new double[len];
fftgraph[0] = res[0].real * res[0].real ;
for(int k = 1; k < (len+1)/2; ++k)
    fftgraph[k] = res[k].real * res[k].real + res[len-k].real
* res[len-k].real ;
if (len % 2 == 0)
    fftgraph[len/2] = res[len/2].real * res[len/2].real ;
int fSize = (len+1)/2;
double[] oBuff = new double[fSize];
double fMinValue = 1e12F;
double fMaxValue = -1e12F;

```

```
int MaxPosition = 0;
```

```

for(int i=0;i<fSize;i++)
{
    oBuff[i] = fftgraph[i];
    double fTemp = fftgraph[i];
    if(i > 1)
    {
        if(fTemp >= fMaxValue)
        {

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

        fMaxValue = fTemp;
        MaxPosition = i;
    }
}
if(fTemp < fMinValue ) fMinValue = fTemp;
}

```

```

float Freq = 4800F; // Byte / Sec
float aSize = len;
float ratio = Freq/aSize;
lbFreq.Text = Convert.ToString(ratio*MaxPosition);

len = 1000;
double[,] data = new double[2,len];
for(int i=0; i<len; i++)
{
    if( (i) < len)
    {
        data[0,i] = i;
        data[1,i] = Math.Log10(fftgraph[i]);
    }
}
GraphFreq.XYValues = data;
GraphFreq.Operate();
GraphFreq.Refresh();
//Thread.Sleep(100);

```

```

}
catch(Exception ex)
{
    //...
}

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

    }

    private void tmRecord_Tick(object sender, EventArgs e)
    {
        // on timer

        tmRecord.Enabled = false;
        if(port.IsOpen)
            port.Close();
    }

    private void port_DataReceived()
    {
        // since RThreshold = 1, we get an event for every character
        byte[] inputData = new byte[16];
        // read the character
        inputData = port.Input;
        ReceiveCount += inputData.Length;
        txtDataReceive.Text = Convert.ToString(ReceiveCount);

        dt.Add(inputData,inputData.Length);

        // display as text
        //Encoding enc = Encoding.ASCII;
        //displayString = enc.GetString(inputData, 0, inputData.Length);
        //try
        //{
        //    this.Invoke(new EventHandler(SetText));
        //}
        //catch(Exception e)
    }

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

    //}
    // Console.WriteLine(e.Message);
    //}
}

private void btnStart_Click(object sender, System.EventArgs e)
{
    // start

    ReceiveCount = 0;

    if(port.IsOpen)
    port.Close();

    port.PortName = txtComPort.Text;
    //port.Settings.BaudRate =
(BaudRates)Convert.ToUInt16(txtBaudRate.Text);
    port.Settings.BaudRate = BaudRates.CBR_57600;
    port.Settings.ByteSize = Convert.ToByte(txtDataBit.Text);
    switch(cbParityBit.SelectedIndex)
    {
        case 0: port.Settings.Parity = Parity.none; break;
        case 1: port.Settings.Parity = Parity.odd; break;
        case 2: port.Settings.Parity = Parity.even; break;
    }

    port.Settings.StopBits = (StopBits)Convert.ToByte(txtStopBit.Text);

    if(port.Open())
    {
        int inv = Convert.ToInt16(txtRecordTime.Text);

        inv = inv * 1000;

        if(inv > 0)

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

    {
        tmRecord.Interval = intv;
        tmRecord.Enabled = true;
    }
}
}

```

```

private void btnStop_Click(object sender, System.EventArgs e)

```

```

{
    // stop
    tmRecord.Enabled = false;
    if(port.IsOpen)
        port.Close();
}

```

```

private void button2_Click(object sender, System.EventArgs e)

```

```

{
    //show
    int segment = 0;
    int segment_limit = 0;
    int len = dt.Length();
    if(len > 100)
    {
        segment_limit = len / 100;
        for(int s=0;s<len;s+=100)
        {
            if(segment < segment_limit)
            {
                segment++;
                double[,] data = new double[2,100];
                for (int i=0; i<100; i++)
                {
                    if( (i+s) < len)

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

        {
            data[0,i] = i;
            data[1,i] =
ConvertToVolt(dt.Buffer[i+s]);
        }
    }
    GraphTime.XYValues = data;
    GraphTime.Operate();
    GraphTime.Refresh();
    Thread.Sleep(100);
}
}
}
private void btnExit_Click(object sender, System.EventArgs e)
{
    //
    Application.Exit();
}

private void menuItem4_Click(object sender, System.EventArgs e)
{
    Application.Exit();
}

private void menuItem4_Click_1(object sender, System.EventArgs e)
{
    Application.Exit();
}

private void tcMain_SelectedIndexChanged(object sender, EventArgs e)
{

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

//
switch(tcMain.SelectedIndex)
{
    case 0: break;

    case 1:
    {
        double[,] data = new double[2,100];
        for (int i=0; i<100; i++)
        {
            data[0,i] = i;
            data[1,i] = 0;
        }
        GraphTime.XYValues = data;
        GraphTime.Operate();
        GraphTime.Refresh();
    }
    break;

    case 2:
    {
        double[,] data = new double[2,100];
        for (int i=0; i<100; i++)
        {
            data[0,i] = i;
            data[1,i] = 0;
        }

        GraphFreq.XYValues = data;
        GraphFreq.Operate();
        GraphFreq.Refresh();
    }
}

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
lbFreq.Text = "0.00";
```

```
}
```

```
break;
```

```
}
```

```
}
```

```
private void button3_Click(object sender, System.EventArgs e)
```

```
{
```

```
}
```

```
}
```

```
}
```



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บรรณานุกรม

- [1] Brigham, E.O., The Fast Fourier Transform, Englewood Cliffs, NJ; Prentice-Hall, Inc., 1974.
- [2] Jon Axelson, The Microcontroller Idea Book, LAKEVIEW RESEARCH
- [3] Andrew D.Dimarogonas, Sam Haddad, "VIBRATION FOR ENGINEERING", Prentice-Hall International Edition, Page 675-704
- [4] Richard J.Higgins, "DIGITAL SIGNAL PROCESSING IN VLSI", Prentice-Hall International Edition, Page 117-172
- [5] David J.Defatta, Joseph G.Lucas, William S.Hodgkiss, "DIGITAL SIGNAL PROCESSING : A System Design Approach", John Wiley, Page 237-327
- [6] รศ.สมยศ จุณณะปิยะ, "การประยุกต์ใช้งานไมโครคอนโทรลเลอร์ MCS-51", คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง, 2543
- [7] ผศ.โกศล โอฬารไพโรจน์, 2544 พิมพ์ครั้งที่ 1, ทฤษฎีวงจรไฟฟ้า 2, กรุงเทพฯ, สำนักพิมพ์ ส.ส.ท. (สมาคมส่งเสริมเทคโนโลยีไทย-ญี่ปุ่น)
- [8] ชัยวัฒน์ ลีมพรจิตรวิไล และ วรพจน์ กรแก้ววัฒนากุล, เรียนรู้และปฏิบัติการไมโครคอนโทรลเลอร์ MCS-51, บริษัท อินโนเวติฟ เอ็กเพอริเมนต์ จำกัด
- [9] ก่อเกียรติ บุญชูกุล, สมศักดิ์ ไชยะภินันท์, ชัยโรจน์ คุณพนิชกิจ, "การวิเคราะห์การสั้นสะเทือน", สมาคมส่งเสริมเทคโนโลยี (ไทย-ญี่ปุ่น), หน้า 1-101

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้