

สำนักหอสมุดกลาง พระจอมเกล้าลาดกระบัง

เครื่องประคัมความดี เสีย ไปเป็นขอมลิตี



9/พ.  
9พ/21ค  
2535

เลขหมู่.....  
เลขทะเบียน.....  
วันเดือนปี.....

นาย พงศ์สิริ อ่อนศรี  
นาย เทียม นิยมลาภเนื่อง

617654637

โครงการนี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรวิทยาศาสตรบัณฑิต  
ภาควิชาที่ศึกษาระบบ  
คณะวิทยาศาสตร์  
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง  
ปีการศึกษา 2535

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

**PITCH-TO-MIDI-CONVERTER**



**A Special project, Submitted in Partial Fulfillment of the  
Requirement, for the Degree of Bachelor of Science  
Department of Applied Physics  
Faculty of Science  
King Mongkut's Institute of Technology Ladkrabang**

**1992**

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หัวข้อโครงการพิเศษ

เครื่องแปลระดับความถี่เสียง ไปเป็น

ข้อมูลดิจิทัล

โดย

นาย ทงศ์ศิริ

อ่อนศรี

นาย เพ็ญภูมิ

นิยมลาภเนื่อง

ภาควิชา

ฟิสิกส์ประยุกต์

อาจารย์ที่ปรึกษา

อ. จิตี

ทญแก้ว

อ. วิชาญ

กนกนทา

ดร. เสน่ห์

เอกะวิภาต

ภาควิชาฟิสิกส์ประยุกต์ คณะวิทยาศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง  
อนุมัติให้โครงการพิเศษฉบับนี้ เป็นส่วนหนึ่งของการศึกษา ตามหลักสูตรวิทยาศาสตรบัณฑิต



หัวหน้าภาควิชาฟิสิกส์ประยุกต์

( ดร. เสน่ห์ เอกะวิภาต )

คณะกรรมการโครงการพิเศษ



ประธานกรรมการ

( อ. ตงทงศ์ สรงประภา )



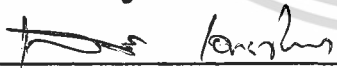
กรรมการ

( อ. จิตี ทญแก้ว )



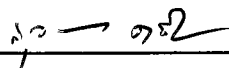
กรรมการ

( อ. วิชาญ กนกนทา )



กรรมการ

( ดร. เสน่ห์ เอกะวิภาต )



กรรมการ

( ผศ. สุวรรณ คุณาราช )

อธิบดีของภาควิชาฟิสิกส์ประยุกต์

คณะวิทยาศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หัวข้อโครงการพิเศษ	เครื่องแปลงระดับความถี่เสียง ไปเป็นข้อมูลมิตี	
นักศึกษา	นาย พงศ์สิริ	อ่อนศรี
อาจารย์ที่ปรึกษา	นาย เพ็ญภูมิ	เนียมลาภเนื่อง
	อ. จิตติ	ทนต์แก้ว
	อ. วิชาญ	กนกภททา
	ดร. เสน่ห์	เอกะวิภาต
ภาควิชา	ฟิสิกส์ประยุกต์	
ปีการศึกษา	2535	

## บทคัดย่อ

โครงการพิเศษนี้ เป็นการพัฒนาเครื่องแปลงระดับความถี่เสียง ไปเป็นข้อมูลมิตี ซึ่งเป็นอุปกรณ์ที่ไร้สำหรับหาความถี่ของรูปคลื่นสัญญาณเสียงที่อยู่ในย่านที่มนุษย์รับฟังได้ แล้วทำการเปลี่ยนให้เป็นข้อมูลตัวโน้ตทางมิตี ภาคอิเล็กทรอนิกส์ที่ออกแบบมานั้น สำหรับใช้กับไมโครโพรเซสเซอร์คอนโทรลเลอร์ ส่วนภาคเอาต์พุตจะให้ข้อมูลออกมา 2 รูปแบบ คือ ข้อมูลตัวอักษรซึ่งแสดงบนจอแสดงผล LCD แบบ 1 บรรทัด 16 ตัวอักษร และข้อมูลทางมิตีที่ส่งไปยังพอร์ต MIDI OUT เพื่อเชื่อมต่อเข้ากับเครื่องดนตรีอื่นๆ ในส่วนของ การควบคุมการทำงานนั้นใช้ Multi Processing Unit เบอร์ Z80180 และใช้โปรแกรมควบคุมระบบที่เขียนด้วยภาษาแอสเซมบลีของ Z80

๗

<b>Special Project Title</b>	Pitch-to-MIDI convertor
<b>Name</b>	Mr. Phongsiri Onsri Mr. Puapoom Niamlarpnuang
<b>Special Project Advisor</b>	Mr. Jiti Nukeaw Mr. Wichan Kogkonta Dr. Sanay Akavipat
<b>Department</b>	Applied Physics
<b>Academic year</b>	1991

### Abstract

This project concerns with the development of Pitch-to-MIDI Converter which is a device for determining the frequency of an audio waveform and converting to the corresponding MIDI note data. The input is designed for condensor microphone. There are 2 types of data for the output: text data is displayed on a 1-line 16-characters LCD monitor and MIDI data which is sent to MIDI OUT port for interfacing the other instrument. The control system uses Z80180 Multi Processing Unit and monitor program was used in Z80 assembly language.

## พิธีกรรมประกาศ

โครงการพิเศษนี้สามารถประสบความสำเร็จได้ด้วยความช่วยเหลือ และความอนุเคราะห์จากบุคคลต่างๆดังนี้

คุณพ่อคุณแม่	และ	
คุณ ทิวาลัย จุฑาเกตุ	ผู้คอยให้กำลังใจ	และให้ความอุปการะในทุกๆด้านจนสำเร็จการศึกษา
อ. จิต ทนุแก้ว	สำหรับโอกาสที่ให้เราได้ศึกษา	และต้นคว้าในด้านที่เรารัก ขอขอบคุณสำหรับคำแนะนำต่างๆ และอิสรภาพในการคิดและทำโครงการพิเศษนี้จนสำเร็จ
ดร. เสน่ห์ เอกะวิภาต	สำหรับคำปรึกษา	และความสะดวกต่างๆในด้านเงินทุนสำหรับโครงการพิเศษนี้
อ. ธนพงศ์ ช่างประภา	และ	
อ. ธนุภิต จารุณาวิน	สำหรับคำแนะนำทางด้านวิชาการ	
คุณ นมร รักษ์จรูญพร	และ	
คุณ ชนาท ภุทธิกรวิทย์พงศ์	และ	
คุณ สุรเมธ สีจจอิสริยวุฒิ	ผู้ให้คำปรึกษาในการทำโครงการพิเศษนี้	
น.ศ. ศรีไสภา นิมิวจิตร	และ	
น.ศ. ชุติ ๓ ถยาง	และ	
น้อง กฤษยา สีระวงศ์	และ	
น้อง ภทริษา ชูภาพทนะ	สำหรับความช่วยเหลือในการทำเอกสารประกอบโครงการพิเศษฉบับนี้	
คุณ เขมัญญ์ นุฎะริองรอด	สำหรับความช่วยเหลือในการประกอบกล่องบรรจุโครงการพิเศษนี้	และสำหรับกำลังใจที่ให้ตลอดมา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

คุณ ประสิทธิ์ วรจักราวชิร และ  
คุณ อุตัย เกียรติวิกรัย สำหรับโปรแกรม KMITL ซึ่งช่วยเราในการพัฒนา  
โครงการพิเศษนี้ได้อย่างมาก  
เพื่อขอบคุณ ที่คอยให้กำลังใจ และความช่วยเหลือในทุกๆอย่าง  
สมาคมศิษย์นารายณ์ ผู้ก่อตั้งและกำหนดมาตรฐานต่างๆของมิตี  
คณะกรรมการทรูทอล์ก ที่ช่วยตรวจรายงานฉบับนี้  
ภาควิชาฟิสิกส์ประยุกต์ ให้ยื่นอุปกรณ์และ เครื่องมือต่างๆ



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## สารบัญ

	หน้า
บทคัดย่อภาษาไทย.....	ก
บทคัดย่อภาษาอังกฤษ.....	ข
กิตติกรรมประกาศ.....	ค
สารบัญรูป.....	ข
สารบัญตาราง.....	ญ
บทที่ 1 บทนำ.....	1
บทที่ 2 หลักการสื่อสารข้อมูล ในระบวมิติ.....	5
2.1 ความหมายของมิติ.....	5
2.2 รูปแบบของการสื่อสาร ในระบวมิติ.....	6
2.3 ฮาร์ดแวร์ของระบวมิติ.....	7
2.4 รูปแบบของข้อมูล.....	8
2.5 ประเภทของข้อความ.....	9
2.5.1 ข้อความควบคุมของสัญญาณเสียง.....	9
2.5.2 ข้อความควบคุมระบบ.....	10
2.6 ประเภทของข้อมูล.....	11
2.6.1 ไบท์สถานะ.....	11
2.6.2 ไบท์ข้อมูล.....	12
2.7 โหมดควบคุมของสัญญาณ.....	12
2.8 ข้อความควบคุมเสียงของช่องสัญญาณ.....	14
2.8.1 การเล่นโน้ต.....	15
2.8.2 การหยุดโน้ต.....	16
2.8.3 การกำหนดแรงกดของคีย์แบบประสานเสียง.....	17
2.8.4 การกำหนดแรงกดรวม.....	18
2.8.5 การเปลี่ยนแปลงการควบคุม.....	19

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

	หน้า
2.8.6	การเปลี่ยนแปลงโปรแกรมเสียง.....20
2.8.7	การเปลี่ยนแปลงระดับความถี่เสียงอย่างต่อเนื่อง.....20
2.9	ข้อความควบคุมโหนดการทำงานของช่องสัญญาณ.....22
2.9.1	การควบคุมโหนดการทำงานของคีย์บอร์ด.....22
2.9.2	การหยุดเล่นโน้ตทุกตัว.....23
2.9.3	การเลือกโหนดการทำงาน.....24
2.10	ข้อความทั่วไปของระบบ.....24
2.10.1	การชี้ตำแหน่งของเพลง.....24
2.10.2	การเลือกเพลง.....25
2.10.3	ข้อความพิเศษของระบบ.....25
2.11	ข้อความเกี่ยวกับเวลาจริงของระบบ.....25
2.11.1	การรีเซ็ตระบบ.....25
2.11.2	สัญญาณนาฬิกา.....25
2.11.3	การกลับไปจุดเริ่มต้น.....26
2.11.4	การหยุด.....26
2.11.5	การเริ่มต้นต่อ.....26
2.11.6	การตรวจสอบสถานะการทำงาน.....26
บทที่ 3	แนวความคิดและหลักการดำเนินงาน.....28
3.1	ระดับความถี่เสียงและการได้ยิน.....28
3.2	การวิเคราะห์ทางฟูรีเยร์และองค์ประกอบทางฟูรีเยร์.....32
3.3	การวัดคาบเวลาของสัญญาณด้วยไมโครคอมพิวเตอร์.....38
3.4	การจัดการสัญญาณ.....40
บทที่ 4	ระบบควบคุมและประมวลผล.....42
4.1	Z80180 MPU.....43
4.1.1	รายละเอียดและการศึกษาของ Z80180.....45
4.1.2	ชาติ MULTIPLEX.....48

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

	หน้า
4.1.3 INTERNAL I/O REGISTER ภายใน Z80180.....	49
4.1.4 เกี่ยวกับ TIMING.....	51
4.1.5 HALT และ LOW POWER MODE.....	51
4.2 MEMORY MANAGEMENT UNIT.....	52
4.2.1 REGISTER CONTROL.....	54
4.3 INTERRUPT.....	56
4.3.1 REGISTER FLAG INTERRUPT.....	56
4.3.2 INTERRUPT VECTOR LOW REGISTER.....	56
4.3.3 INT/TRAP CONTROL REGISTER.....	57
4.4 ASYNCHRONOUS SERIAL COMMUNICATION INTERFACE.....	59
4.4.1 STAT 0,1.....	60
4.4.2 CNTLA 0,1.....	62
4.5 PROGRAMABLE RELOAD TIMER.....	66
4.5.1 TIMER DATA REGISTER.....	66
4.5.2 TIMER RELOAD REGISTER.....	67
4.6 คำสั่งเพิ่มเติมที่ใช้.....	69
<b>บทที่ 5 ระบบแปลงระดับความถี่เสียง ไปเป็นข้อมูลดิจิทัล</b> .....	<b>70</b>
5.1 วงจรตรวจับระดับความถี่เสียง.....	70
5.2 การปรับปรุง CP-180 ให้ทำงานในระบบดิจิทัล.....	73
5.2.1 การแก้ไขวงจรกำเนิดสัญญาณนาฬิกา.....	73
5.2.2 การแก้ไขส่วนเอาต์พุตของ ASCI.....	75
5.2.3 วงจรดีบอร์ค และจอแสดงผล LCD.....	76
5.3 โปรแกรมควบคุมการทำงานของระบบ.....	77
5.3.1 งานของระบบควบคุมและประมวลผล.....	78
5.3.2 โปรแกรมควบคุมการทำงานหลัก.....	79
5.3.3 โปรแกรมบริการอินพุต/เอาท์พุต.....	80

	หน้า
<b>บทที่ 6</b> <b>การใช้งานและข้อเสนอแนะ.....</b>	<b>91</b>
6.1 <b>การใช้งานเครื่องแปลงระดับความถี่เสียงไปเป็นข้อมูลดิจิทัล.....</b>	<b>91</b>
6.1.1 <b>โหมดการทำงาน (Perform mode).....</b>	<b>91</b>
6.1.2 <b>โหมดแก้ไข (Edit mode).....</b>	<b>93</b>
6.2 <b>คุณสมบัติของเครื่องแปลงระดับความถี่เสียงไปเป็นข้อมูลดิจิทัล.....</b>	<b>95</b>
6.3 <b>ข้อเสนอแนะ.....</b>	<b>95</b>
<b>ภาคผนวก ก.    วงจร CP-180.....</b>	<b>97</b>
<b>ภาคผนวก ข.    ASSEMBLY PROGRAM.....</b>	<b>100</b>
<b>ภาคผนวก ค.    DATA SHEET.....</b>	<b>126</b>
<b>ภาคผนวก ง.    ภาพถ่ายประกอบ.....</b>	<b>139</b>
<b>เอกสารอ้างอิง.....</b>	<b>143</b>
<b>ประวัติผู้เขียน.....</b>	<b>144</b>

## สารบัญ

		หน้า
รูปที่ 1.1	ส่วนประกอบหลักของ โครงงาน.....	2
รูปที่ 2.1	การเชื่อมต่อแบบพื้นฐานของมิตระหว่างคีย์บอร์ด 2 ตัว.....	5
รูปที่ 2.2	รูปแบบของข้อมูลมีมาตรฐานที่ความเร็ว 31.25 kbps.....	6
รูปที่ 2.3	ฮาร์ดแวร์มาตรฐานของมิตที่ใช้ในปัจจุบัน.....	7
รูปที่ 2.4	รูปแบบของข้อมูลในระบบมิต และตัวอย่าง.....	9
รูปที่ 2.5	การทำงานของชิปไมโครระบบมิต ใน โหนดต่างๆ.....	14
รูปที่ 2.6	แสดงข้อความการส่ง โหนด.....	15
รูปที่ 2.7	แสดงข้อความการหยุด โหนด.....	16
รูปที่ 2.8	แสดงข้อความกำหนดแรงกดคีย์แบบประสานเสียง.....	17
รูปที่ 2.9	แสดงข้อความกำหนดแรงกดรวม.....	18
รูปที่ 2.10	แสดงข้อความเปลี่ยนแปลงการควบคุม.....	19
รูปที่ 2.11	แสดงข้อความเปลี่ยนแปลง โปรแกรมเสียง.....	20
รูปที่ 2.12	แสดงข้อความเปลี่ยนแปลงระดับความถี่เสียงอย่างต่อเนื่อง.....	21
รูปที่ 2.13	แสดง โครงสร้างการทำงานของชิปไมโครระบบมิต.....	22
รูปที่ 3.1	แสดงกลไกการได้ยินของหูมนุษย์.....	29
รูปที่ 3.2	ความถี่ของระดับความถี่เสียงของ เบส โน.....	31
รูปที่ 3.3	คลื่นเชิงซ้อน และองค์ประกอบทางฮาร์โมนิก.....	33
รูปที่ 3.4	คลื่นสี่เหลี่ยมจัตุรัส และองค์ประกอบทางฮาร์โมนิกของมัน.....	34
รูปที่ 3.5	แสดงการสังเคราะห์ทางฟูรีเยร์ของคลื่นสี่เหลี่ยมจัตุรัส.....	35
รูปที่ 3.6	การสังเคราะห์ทางฟูรีเยร์ของคลื่นเชิงซ้อน 2 รูป.....	36
รูปที่ 3.7	รูปคลื่นและองค์ประกอบทางฮาร์โมนิกของ โหนด C.....	37
รูปที่ 3.8	รูปคลื่นและองค์ประกอบทางฮาร์โมนิกของ โหนด Bb.....	38
รูปที่ 3.9	แสดงแผนผังการใช้ไมโครคอมพิวเตอร์ในการหาคาบเวลา.....	39
รูปที่ 3.10	แสดงแผนผังการจัดรูปสัญญาณสำหรับการอินเทอร์เฟซ.....	40

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

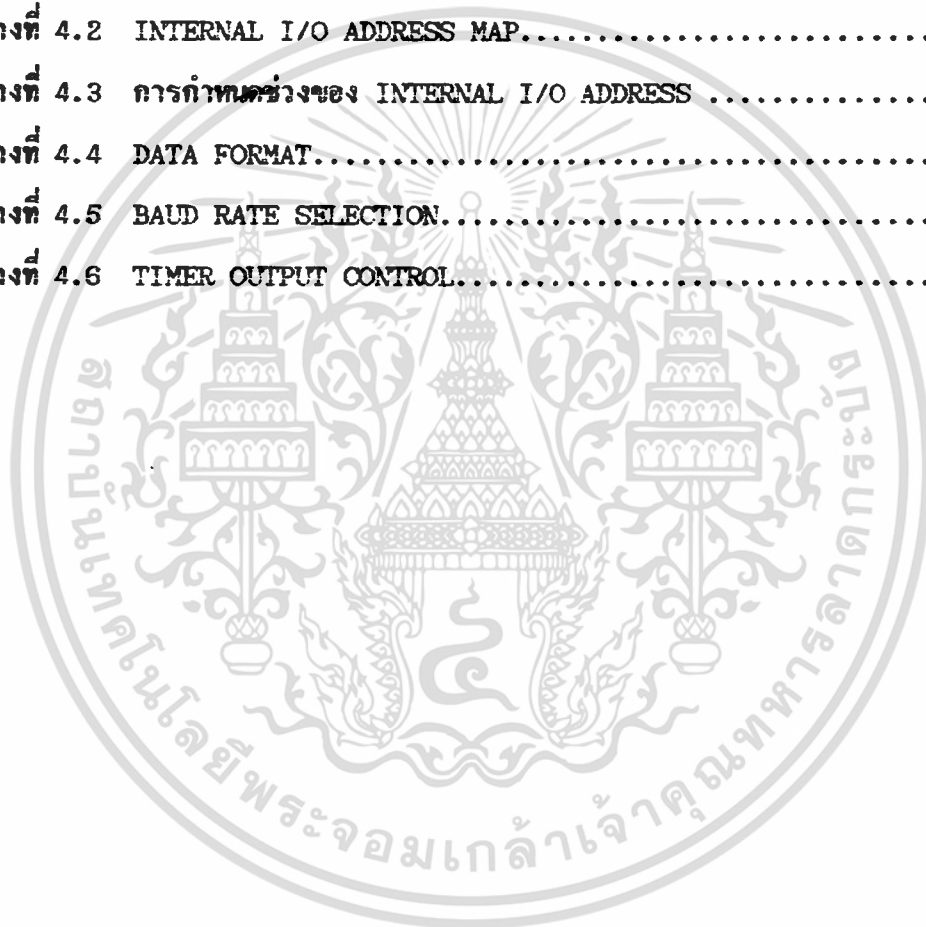
	หน้า
รูปที่ 4.1 สถาปัตยกรรมภายในของ Z80180.....	44
รูปที่ 4.2 I/O CONTROL REGISTER.....	50
รูปที่ 4.3 เปรียบเทียบ T STATE ของ Z80 กับ Z80180.....	51
รูปที่ 4.4 การโปรแกรม รีจิสเตอร์ CBAR.....	53
รูปที่ 4.5 MMU COMMON/BANK AREA REGISTER.....	55
รูปที่ 4.6 การแบ่งพื้นที่หน่วยความจำที่ใช้ในโครงการ.....	55
รูปที่ 4.7 VECTOR TABLE.....	57
รูปที่ 4.8 INT/TRAP CONTROL REGISTER.....	57
รูปที่ 4.9 ASCII BLOCK DIAGRAM.....	59
รูปที่ 4.10 ASCII STATUS REGISTER.....	60
รูปที่ 4.11 ASCII CONTROL REGISTER A.....	62
รูปที่ 4.12 ASCII CONTROL REGISTER B.....	64
รูปที่ 4.13 TIMER CONTROL REGISTER.....	67
รูปที่ 5.1 วงจรตรวจระดับความถี่เสียง.....	70
รูปที่ 5.2 รูปสัญญาณ ณ จุดต่างๆในวงจร Pitch detector.....	72
รูปที่ 5.3 ส่วนของวงจรถ่ายสัญญาณมาที่คาบเมมอรัค CP-180.....	74
รูปที่ 5.4 แสดงการแก้ไขส่วนเอาต์พุตของ ASCII เมมอรัค CP-180.....	75
รูปที่ 5.5 แสดงการเชื่อมต่อวงจรมีเมมอรัคและแสดงผล LCD กับพอร์ท 8255.....	77
รูปที่ 5.6 แสดงการทำงานของ โปรแกรมหลัก.....	82
รูปที่ 5.7 แสดงการทำงานของ โปรแกรมหลัก (ต่อ).....	83
รูปที่ 5.8 แสดงการทำงานของ โปรแกรมหลัก (ต่อ).....	84
รูปที่ 5.9 โปรแกรมให้บริการอินเตอร์รัคท์ของ INT0.....	85
รูปที่ 5.10 โปรแกรมให้บริการอินเตอร์รัคท์ของ PRTO.....	86
รูปที่ 5.11 โปรแกรมให้บริการอินเตอร์รัคท์ของ INT1 (MOD_KEY).....	87
รูปที่ 5.12 โปรแกรมให้บริการอินเตอร์รัคท์ของ INT1 (FUNC_KEY).....	88
รูปที่ 5.13 โปรแกรมให้บริการอินเตอร์รัคท์ของ INT1 (INC_KEY).....	89

## หน้า

รูปที่ 5.14 โปรแกรมให้บริการอินเทอร์เน็ตของ INT1 (DEC_KEY).....	90
รูปที่ 6.1 แสดงลักษณะภายนอกของเครื่องแปลงระดับความถี่เสียง ไปเป็นข้อมูลmidi...91	
รูปที่ 6.2 แสดงการเชื่อมต่อกับอินรีไซเคอร์ในสายmidi.....	92
รูปประกอบที่ 1 เครื่องแปลงระดับความถี่เสียง ไปเป็นข้อมูลmidi.....	140
รูปประกอบที่ 2 แสดงคอนเนคเตอร์ MIDI OUT ทางด้านหลังของเครื่อง.....	140
รูปประกอบที่ 3 ไมโครโฟนแบบ HEADSET ที่ใช้กับเครื่องแปลงระดับ- -ความถี่เสียง ไปเป็นข้อมูลmidi.....	141
รูปประกอบที่ 4 สายเชื่อมต่อข้อมูลทางmidi (MIDI cable).....	141
รูปประกอบที่ 5 แสดงส่วนประกอบภายในของเครื่องแปลงระดับเสียง- -ไปเป็นข้อมูลทางmidi.....	142
รูปประกอบที่ 6 แสดงการใช้งานร่วมกับ Synthesizer KAWAI รุ่น K1 II.....	142

**สารบัญตาราง**

	<b>หน้า</b>
ตารางที่ 3.1 แสดงการเปรียบเทียบอัตราส่วนของความถี่ในระบบต่างๆ.....	32
ตารางที่ 3.2 ความสัมพันธ์ทางแอมพลิจูดของฮาร์โมนิกของคลื่นมาตรฐาน.....	35
ตารางที่ 4.1 แสดงสถานะการทำงานของ CPU.....	47
ตารางที่ 4.2 INTERNAL I/O ADDRESS MAP.....	49
ตารางที่ 4.3 การกำหนดช่วงของ INTERNAL I/O ADDRESS .....	50
ตารางที่ 4.4 DATA FORMAT.....	64
ตารางที่ 4.5 BAUD RATE SELECTION.....	66
ตารางที่ 4.6 TIMER OUTPUT CONTROL.....	68



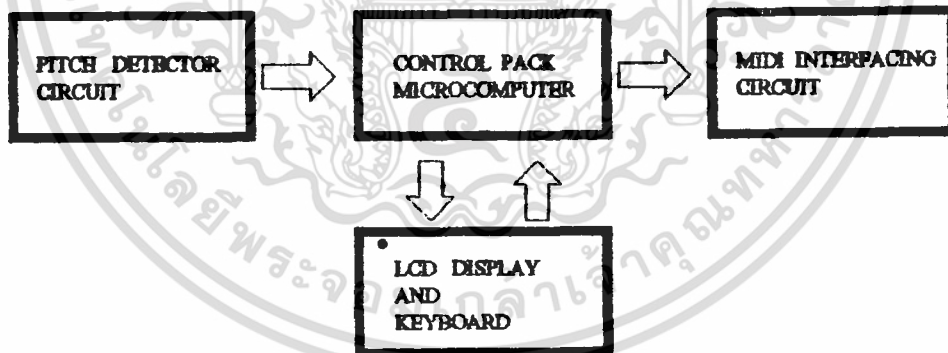
## บทที่ 1

### บทนำ

ดนตรีไฟฟ้าหรือ ELECTRONIC MUSIC นั้นเป็นศิลปะอีกแขนงหนึ่ง ซึ่งเป็นจุดเชื่อมประสานระหว่างศาสตร์แห่งศิลป์กับเทคโนโลยีทางด้านอิเล็กทรอนิกส์ และไมโครคอมพิวเตอร์ นับเป็นช่วงเวลาสามสิบปีที่มีการพัฒนาศาสตร์ทางด้านนี้อย่างจริงจัง จนกระทั่ง ในปัจจุบัน เราสามารถนำมาใช้ในการสร้างสรรค์งานดนตรีในแนวทางต่างๆ และนับวันยิ่งจะมีบทบาทเพิ่มมากขึ้น MIDI (Musical Instrument Digital Interface) เป็นส่วนหนึ่งของวิชาการทางดนตรีไฟฟ้านี้ อันมีวัตถุประสงค์เพื่อให้เป็นมาตรฐาน ในการสื่อสารระหว่างเครื่องดนตรีไฟฟ้าทั่วโลก มีค้ำช่วยให้เราสามารถลดขั้นตอน ค่าใช้จ่ายและเวลาในการสร้างงานดนตรีลง อีกทั้งยังช่วยเพิ่มคุณภาพของเสียงดนตรี ซึ่งเป็นผลมาจากการเชื่อมโยงเสียงจาก เครื่องดนตรีไฟฟ้ามาชนิดเข้าด้วยกัน ซินธิไซเซอร์ (synthesizer) เป็นเครื่องดนตรีไฟฟ้าประเภทคีย์บอร์ด ซึ่งมีความสามารถในการสังเคราะห์เสียงต่างๆ ได้มากมาย และเป็นเครื่องดนตรีไฟฟ้าที่มีความสัมพันธ์กันมากที่สุด ซินธิไซเซอร์ ทุกวันนี้ในปัจจุบัน จะติดตั้งระบบมิดีเอาไว้เพื่อให้สามารถควบคุมหรือสื่อสารกับ ซินธิไซเซอร์ ตัวอื่นๆหรือแม้แต่กับเครื่อง ไมโครคอมพิวเตอร์ที่สามารถสื่อสารในระบบมิดีได้

ในช่วงหนึ่งของการพัฒนาทางมิดี ได้มีความพยายามที่จะนำเสียงจากเครื่องดนตรีจริง เช่น กีตาร์ ทรัมเป็ต หรือแม้แต่เสียงร้องของมนุษย์มาใช้ในการควบคุมซินธิไซเซอร์ ให้เล่นตามท่วงทำนองที่ เกิดจากเสียงของ เครื่องดนตรีเหล่านั้น เพราะสามารถทำให้กับดนตรี หรือ ใครก็ตามที่มีจินตนาการทางด้านดนตรี เล่นหรือควบคุมให้ซินธิไซเซอร์ สร้างสรรค์ทำนองหรือบทเพลงได้อย่างอิสระอันเนื่องมาจาก ซินธิไซเซอร์ เป็นเครื่องดนตรีซึ่งมีความสามารถในการสร้างเสียงต่างๆ ไม่เพียงแต่เฉพาะเสียงที่มีอยู่จริงในธรรมชาติเท่านั้น แม้แต่เสียงที่เกิดจากจินตนาการก็สามารถสร้างขึ้นมาได้หากมีความรู้เกี่ยวกับโครงสร้างของเสียงเพียงพอ จากคุณสมบัติพิเศษข้อนี้เองทำให้เราสามารถสร้างเสียงใดๆก็ได้จากการเล่นเครื่องดนตรีที่เราสามารถเล่นได้เพียงเครื่องเดียว โดยอาศัยความสามารถในการสื่อสารกับ ซินธิไซเซอร์ ของมิดีนั่นเอง มาถึงจุดนี้เราจำเป็นต้อง

ต้องอาศัยอุปกรณ์ที่มีความสามารถในการเปลี่ยนท่วงทำนองทางดนตรี ซึ่งอยู่ในรูปของระดับความถี่เสียงของ โน้ตทางดนตรีให้อยู่ในรูปของข้อมูลทางมิติ เครื่องแปลงระดับความถี่เสียงไปเป็นรหัสมิติ หรือ Pitch-To-MIDI Converter จะเป็นอุปกรณ์ที่ตอบสนองความต้องการดังกล่าว โดยอาศัยหลักการที่ว่า "ระดับความถี่ของคลื่นที่มีความยาวที่แน่นอน (periodic wave) ใดๆ จะหาได้จากความถี่มูลฐานของคลื่นนั้น" จากแนวความคิดดังกล่าวทำให้เราสามารถออกแบบ Pitch-To-MIDI Converter ได้โดยอาศัยความรู้ทางด้านฮาร์ดแวร์และดิจิทัลอิเล็กทรอนิกส์ ประกอบกับความสามารถในการประมวลผลและความคมของไมโครคอมพิวเตอร์ ดังจะเห็นได้จากแผนผังแสดงส่วนประกอบทางวงจรของ Pitch-To-MIDI Converter อย่างคร่าวๆดังในรูปที่ 1.1



รูปที่ 1.1 ส่วนประกอบหลักของโครงการ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 1.1 แสดงถึงส่วนประกอบหลักทั้ง 4 ส่วนของ Pitch-To-MIDI Convertor อันประกอบด้วย

1. วงจรตรวจสอบระดับความถี่ (Pitch detector circuit) เป็นส่วนรับสัญญาณเสียงที่เข้ามาแล้วทำการสร้างสัญญาณอินเทอร์พรีทที่มีความสัมพันธ์กับคาบเวลาของสัญญาณนั้น เพื่อส่งไปกระตุ้นให้วงจรมicrocontroller ในภาคต่อไปทำการประมวลผล
2. แผงควบคุมไมโครคอมพิวเตอร์ (Control pack microcomputer) จะทำการรับสัญญาณอินเทอร์พรีทจากส่วนแรก มาทำการประมวลผลหาระดับความถี่ของสัญญาณอินพุต จากนั้นจะทำการเปรียบเทียบว่าระดับความถี่นั้นตรงกับรหัสมีดีอะไร ข้อมูลทางมีดีซึ่งแทนโน้ตทางดนตรีของระดับความถี่นั้น พร้อมทั้งรหัสควบคุมอื่นๆจะถูกส่งต่อไปยังวงจรมีดีอินเตอร์เฟซ
3. วงจรมีดีอินเตอร์เฟซ (MIDI interfacing circuit) ทำหน้าที่ในการส่งข้อมูลทางมีดีซึ่งอยู่ในรูปของ CURRENT LOOP ออกไปสู่เครื่องดนตรีไฟฟ้าในระบวมมีดีภายนอก
4. วงจรแสดงผล LCD และ คีย์บอร์ด จอ LCD นั้นจะทำการแสดงผลข้อมูลควโน้ตที่ตรวจที่ตรวจวัดได้ หรือข้อมูลอื่นๆที่จำเป็นในการติดต่อกับผู้ใช้ ส่วนคีย์บอร์ดนั้น ใช้สำหรับการแก้ไขเปลี่ยนแปลงข้อมูลบางอย่างที่ใช้ในการทำงาน

ทั้งหมดที่กล่าวมาแล้วข้างต้น เป็นเพียงแนวความคิดหลักที่ใช้ในการทำโครงการนี้ สำหรับรายละเอียดๆของแต่ละส่วนจะกล่าวไว้ในบทต่อไป

### 1.1 วัตถุประสงค์ในการจัดทำโครงการพิเศษ

1. เพื่อศึกษาการควบคุมเครื่องดนตรีไฟฟ้า ในระบวมิติด้วยเสียงดนตรี
2. เพื่อศึกษาการประยุกต์ใช้งาน ไมโครคอมพิวเตอร์ทางด้านดนตรี

### 1.2 ขั้นตอนการดำเนินงาน

1. ศึกษาหาวิธีการแปลงระดับความถี่เสียง ให้อยู่ในรูปสัญญาณควบคุม
2. ศึกษาการทำงานของระบบควบคุมและประมวลผลด้วย ไมโครคอมพิวเตอร์
3. ศึกษามาตรฐานของมิดี
4. ออกแบบสร้างวงจรส่วนต่างๆ และ โปรแกรมควบคุมการทำงาน
5. ทดสอบและปรับปรุง

### 1.3 ประโยชน์ที่ได้รับจากการทำโครงการพิเศษนี้

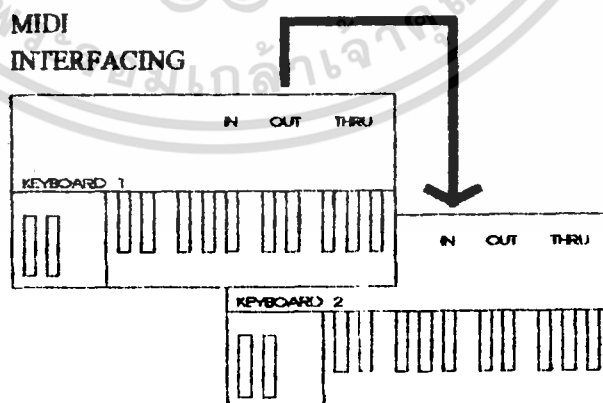
1. สามารถนำไปใช้ในงานแสดงดนตรีได้โดยตรง
2. สามารถนำไปใช้ร่วมกับซอฟต์แวร์ทางดนตรี เพื่อช่วยในการประพันธ์เพลงได้
3. สามารถช่วยให้ผู้ที่ขาดทักษะการเล่นดนตรี เล่นหรือสร้างงานดนตรีจากเครื่องดนตรีไฟฟ้าได้

## บทที่ 2

### หลักการสื่อสารข้อมูลในระนาบมิติ

#### 2.1 ความหมายของมิติ

มิติหรือ MIDI ย่อมาจากคำว่า Musical Instrument Digital Interface เป็นระบบมาตรฐานของการสื่อสารข้อมูลระหว่างเครื่องดนตรีไฟฟ้าที่มีไมโครโปรเซสเซอร์ วัตถุประสงค์หลักของระบบมิติก็คือ การทำให้เครื่องดนตรีไฟฟ้าที่ต่างรุ่นหรือต่างผู้ผลิตสามารถทำงานร่วมกันได้ เพื่อก่อให้เกิดระบบการสร้างดนตรีที่มีประสิทธิภาพดีเป็นระบบที่พัฒนามาจากแนวความคิดที่จะเชื่อมโยงการทำงานของเครื่องดนตรีไฟฟ้าประเภท Synthesizer ของนาย Dave Smith แห่งบริษัท Sequential circuit Incorporation ซึ่งภายหลังได้รับการปรับปรุงจนเป็นมาตรฐานภายใต้ชื่อ MIDI 1.0 โดยสมาคมมิคินานาชาติ(International MIDI Association หรือ IMA) ข้อกำหนดต่างๆของ MIDI 1.0 ได้รับการยอมรับจากบริษัทผู้ผลิตเครื่องดนตรีไฟฟ้าทั่วโลก และได้ถูกนำไปติดตั้งให้เครื่องดนตรีไฟฟ้าและอุปกรณ์ทางดนตรีต่างๆมากมาย จนทำให้เครื่องดนตรีไฟฟ้าเหล่านั้น สามารถติดต่อสื่อสารกันได้ผ่านทางมิติ แม้ว่ามันจะถูกสร้างจากโรงงานผู้ผลิตต่างกันก็ตาม ดังนั้นมิติก็เปรียบเสมือนภาษาสากลของเครื่องดนตรีไฟฟ้านั่นเอง



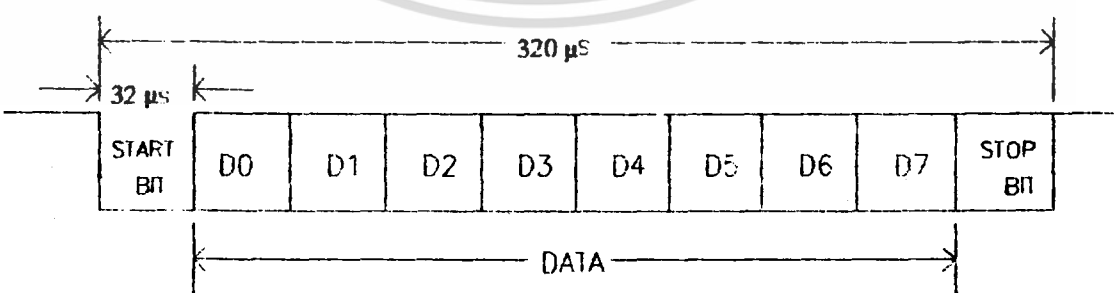
รูปที่ 2.1 การเชื่อมต่อแบบต้นฐานของมิติระหว่างคีย์บอร์ด 2 ตัว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ข้อมูลมีค่าน้อยอยู่ในรูปของข้อความทางดิจิทัล ที่มีความหมายไม่เชิงควบคุม ซึ่งไมโครโปรเซสเซอร์ของเครื่องดนตรีไฟฟ้าที่ระบบมีค่านำเข้าสามารถตีความได้ ภายใต้ข้อความเหล่านี้จะบรรจุคำสั่งและข้อมูลที่สามารถทำให้เครื่องดนตรีไฟฟ้าเครื่องหนึ่งปฏิบัติงานตามข้อความที่ได้รับจากเครื่องดนตรีไฟฟ้าอีกเครื่องหนึ่งได้ ตัวอย่างเช่นจากรูปที่ 2.1 เราให้คีย์บอร์ดตัวที่ 1 เป็นคีย์บอร์ดหลักทำหน้าที่ควบคุมคีย์บอร์ดตัวที่ 2 ซึ่งเป็นตัวถูกควบคุมผ่านทางมีคี่ ดังนั้นหากเราเล่นโน้ตตัวใดบนคีย์บอร์ดตัวที่ 1 บนคีย์บอร์ดตัวที่ 2 ก็จะเปลี่ยนเสียงโน้ตตัวนั้นด้วยเช่นกัน นั่นหมายความว่า มีข้อความทางมีคี่ถูกส่งจากคีย์บอร์ด A ไปยังคีย์บอร์ด B ให้กระทำการดังกล่าว

2.2 รูปแบบของการสื่อสารในระบบมีคี่

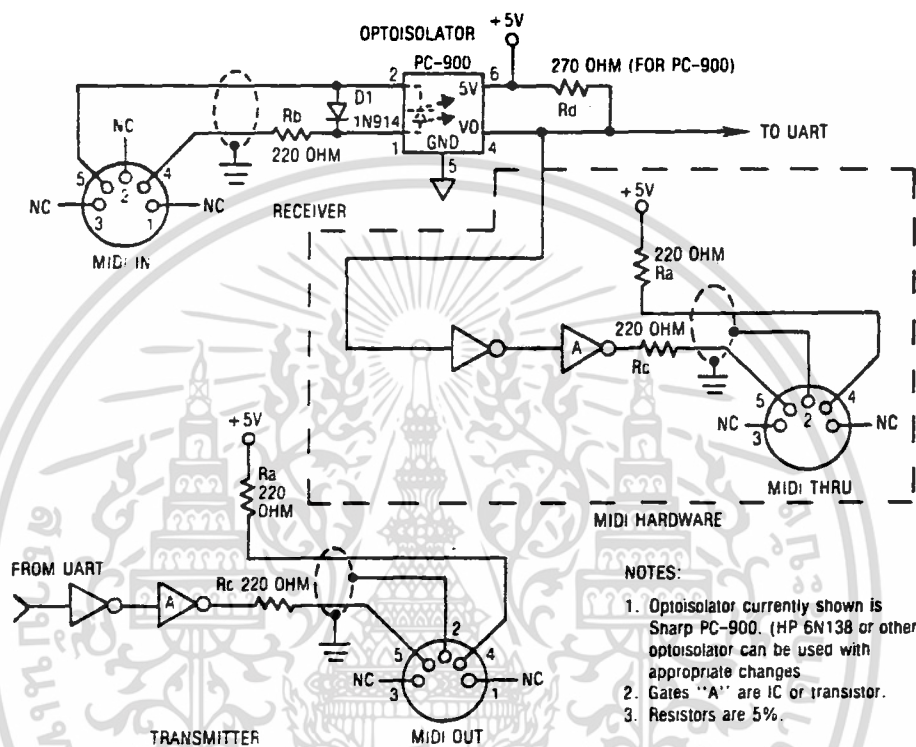
มีคี่ใช้การสื่อสารแบบอะซิงโครนัสคล้ายกับมาตรฐาน RS-232 หนึ่งคำของมีคี่จะมีขนาด 10 บิต ประกอบด้วย บิตเริ่มต้น (start bit) 1 บิต , บิตข้อมูล 8 บิต และตามด้วยบิตหยุด (stop bit) อีก 1 บิต ดังแสดงไว้ในรูปที่ 2.2 แต่ละบิตของคำทางมีคี่จะมีคาบเวลา 32 ไมโครวินาที โดยที่บิตเริ่มต้นจะมีลอจิกเป็น '0' ตามด้วยบิตข้อมูลซึ่งจะเรียงจากบิตที่มีนัยสำคัญต่ำสุด (LSB) ไปจนถึงบิตที่มีนัยสำคัญสูงสุด (MSB) จากนั้นก็จะ เป็นบิตหยุด ซึ่งมีลอจิกเป็น '1' รวมแล้ว 1 คำทางมีคี่จะมีคาบเวลาทั้งหมด 320 ไมโครวินาที หรือคิดเป็นอัตราเร็วในการรับส่งข้อมูลได้ 31.25 กิโลบิตต่อวินาที (kbps) นั่นเอง



รูปที่ 2.2 รูปแบบของข้อมูลมีคี่มาตรฐานที่มีความเร็ว 31.25 kbps

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## 2.3 ฮาร์ดแวร์ของระบบนี้



รูปที่ 2.3 ฮาร์ดแวร์มาตรฐานของมิดีที่ใช้ในปัจจุบัน

วงจรในรูปที่ 2.3 จะเห็นว่ามีการเสถียรขนาด 5 มิลลิแอมป์ไหลวนอยู่ในวงจร โดยกำหนดว่า ถ้าข้อมูลมีสถานะลอจิกเป็น '0' จะมีการเสถียรไหลในวงจร แต่ถ้ามีลอจิกเป็น '1' ก็จะไม่มีการเสถียรไหลในวงจร เราเรียกการทำงานในลักษณะนี้ว่า "สตูปกระแส" หรือ current loop ในภาคส่งนั้นจะรับข้อมูลจากส่วนที่เรียกว่า UART ( Universal Asynchronous Receiver-Transmitter ) ซึ่งเป็นข้อมูลแบบอนุกรมที่แปลงมาจากข้อมูลแบบขนานที่ไมโครโปรเซสเซอร์ของเครื่องดนตรีไฟฟ้าส่งออกมา โดยมีเกนแบบอินเวอร์เตอร์ 2 ตัวทำหน้าที่เป็นตัวขับสัญญาณ (line driver) ส่งออกไปทางคอนเนกเตอร์ที่เรียกว่า "MIDI OUT" ส่วนทางภาครับนั้นจะรับข้อมูลเข้ามาทาง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

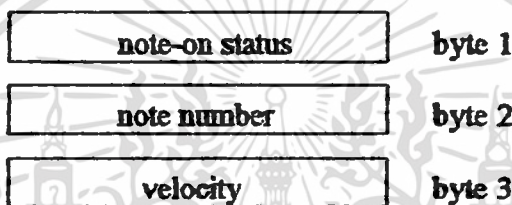
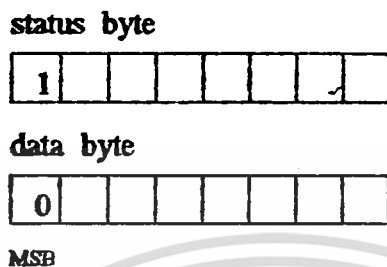
คอนเนกเตอร์ที่เรียกว่า "MIDI IN" และจะถูกขับไปถึงผ่านทางออปโตไอโซเลเตอร์เข้าสู่ UART เพื่อแปลงข้อมูลส่งให้ไมโครโปรเซสเซอร์อีกที ซึ่งตัวออปโตไอโซเลเตอร์นี้ควรเป็นแบบความเร็วสูง และควรมีเวลาในการสวิตช์ (rise and fall time) น้อยกว่า 2 ไมโครวินาที และจากรูปจะเห็นว่ามียังมีวงจรถูกใส่ส่วนหนึ่ง ซึ่งเชื่อมต่อกับขาเอาต์พุตของออปโตไอโซเลเตอร์ มีลักษณะเหมือนกับส่วนของภาคส่ง แต่ต่อเข้ากับคอนเนกเตอร์ที่มีชื่อว่า "MIDI THRU" ส่วนนี้จะทำหน้าที่ส่งผ่านข้อมูลที่เข้ามาทางคอนเนกเตอร์ "MIDI IN" ออกไปสู่ภายนอก

สำหรับคอนเนกเตอร์นี้ มีข้อกำหนดให้ใช้เป็นแบบ DIN 5 ขา โดยที่ขา 1 และขา 3 จะไม่ถูกใช้งาน คอนเนกเตอร์ที่ติดตั้งบนเครื่องดนตรีไฟฟ้าจะเป็นแบบตัวเมีย เมื่อต้องการเชื่อมต่อจะใช้สายชนิดที่ควรเป็นสายชนิดคู่ที่ทำการตีเกลียวกัน โดยต่อส่วนที่รีดค์เข้ากับขา 2 ของคอนเนกเตอร์ตัวผู้ที่ปลายทั้งสองของสาย สายชนิดนี้ไม่ควรยาวเกิน 50 ฟุตหรือ 15 เมตร

ข้อมูลเหล่านี้ เป็นข้อกำหนดทางฮาร์ดแวร์ของระบบนี้เพื่อให้เป็นมาตรฐานเดียวกันทั่วโลก ในหัวข้อต่อไปจะกล่าวถึงข้อกำหนดและรายละเอียดต่างๆ ที่เกี่ยวกับรูปแบบทางภาษาของมิดีที่เรากำลังจะดูกัน

#### 2.4 รูปแบบของข้อมูล ( Data Format )

สำหรับการสื่อสารด้วยข้อความทางมิดีนั้น ในแต่ละข้อความจะประกอบไปด้วย ไบต์สถานะ (status byte) อันเป็นส่วนคำสั่งทางมิดี และตามด้วยไบต์ข้อมูล (data byte) ซึ่งใช้เป็นตารางมีเตอร์ของคำสั่งนั้นอีก 1 หรือ 2 ไบต์ ยกเว้นในกรณีของข้อความที่เกี่ยวกับเวลาจริง (Real-time message) และข้อความพิเศษ (Exclusive message) ซึ่งจะกล่าวถึงในภายหลัง ในรูปที่ 2.4 จะแสดงตัวอย่างของรูปแบบดังกล่าว โดยไบต์ 1 จะเป็นคำสั่งบอกให้เล่นโน้ต ส่วนไบต์ 2 จะบอกว่าเป็นโน้ตตัวใด และสำหรับไบต์ 3 จะบอกถึงความดังของโน้ตตัวนั้น ซึ่งประมวลผลจากความเร็วในการกดคีย์



รูปที่ 2.4 รูปแบบของข้อมูลในระบวมิติ และตัวอย่าง

### 2.5 ประเภทของข้อความ ( Message Types )

โดยทั่วไปแล้วข้อความทางมิติจะถูกแบ่งออกเป็น 2 ประเภทใหญ่ๆ ตามลักษณะหน้าที่ของมัน อันได้แก่

#### 2.5.1 ข้อความควบคุมช่องสัญญาณ ( Channel message )

เครื่องดนตรีในระบวมิตินั้น สามารถติดต่อกับสื่อสารกับเครื่องดนตรีไฟฟ้าเครื่องอื่นๆ ได้หลายเครื่องพร้อมกัน ข้อความทางมิติจึงจะมีส่วนที่ไว้บอกว่าจะทำการติดต่อกับอุปกรณ์ตัวใด มิฉะนั้นจะใช้ 4 บิตล่างของไบต์สถานะเป็นตัวกำหนดช่องสัญญาณของอุปกรณ์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ที่ต้องการจะคิดต่อด้วย เพราะฉะนั้นจึงสามารถกำหนดได้ 16 ช่องสัญญาณ ข้อความนี้ จะได้รับการตอบสนองเฉพาะ ไบท์สถานะที่มีช่องสัญญาณตรงกันกับช่องสัญญาณที่ด้านรับเท่านั้น ข้อความเกี่ยวกับช่องสัญญาณมี 2 ประเภทคือ

1. ข้อความควบคุมเสียง (Voice message) ใช้ควบคุมเสียงของเครื่อง คนตรีไฟฟ้า ข้อความเกี่ยวกับเสียงจะถูกส่งผ่านทางช่องสัญญาณเสียง (Voice channel)
2. ข้อความควบคุมโหมดการทำงาน (Mode message) ใช้กำหนดลักษณะ การตอบสนองของเครื่องคนตรี ต่อข้อความเกี่ยวกับเสียง หรือ เป็นการ กำหนดโหมดการทำงานของระบบนั่นเอง โดยจะส่งผ่านทางช่องสัญญาณ พื้นฐาน (Basic Channel)

#### 2.5.2 ข้อความควบคุมระบบ ( System message )

ข้อความที่ใช้ควบคุมระบบจะ ไม่มีการกำหนดหมายเลขช่องสัญญาณ ดังนั้น ไม่ว่าด้านรับจะเป็นช่องสัญญาณอะไรก็จะตอบสนองข้อความควบคุมระบบเสมอ ข้อความ ควบคุมระบบเมื่ออยู่ด้วยกัน 3 แบบคือ

1. ข้อความทั่วไป (Common message) เป็นข้อความที่ใช้สำหรับอุปกรณ์ ทุกตัวในระบบ
2. ข้อความเกี่ยวกับเวลาจริง (Real-Time message) จะมีเฉพาะไบท์ สถานะเท่านั้น ไม่มีไบท์ข้อมูล ข้อความนี้จะถูกส่งออกไปเมื่อไรก็ได้
3. ข้อความพิเศษ (Exclusive message) มีหลายไบท์ข้อมูลและจะจบ ด้วยไบท์ที่เรียกว่า End of Exclusive (EOX) หรือ ไบท์สถานะอื่นๆ เป็นข้อความที่ใช้เฉพาะกับอุปกรณ์ของผู้ผลิตนั้นๆที่กำหนดไว้ในรหัสประจำ เครื่อง (Identification Codes , ID codes) ของแต่ละบริษัท ผู้ผลิตเท่านั้น ถ้าอุปกรณ์ทางด้านรับ มีรหัสประจำเครื่องที่ไม่ตรงกับ ข้อความพิเศษที่ได้ มันจะไม่สนใจไบท์ข้อมูลที่ตามมา ผู้ผลิตแต่ละ รายสามารถกำหนดรูปแบบความหมายข้อมูลของตนเองได้ โดยส่งเป็น

ข้อความพิเศษที่รับประจำเครื่องของคุณ ( ซึ่งเป็นรหัสเดียวกันกับทางด้านรับ )

## 2.6 ประเภทของข้อมูล ( Data Types )

ในทางมีได้กำหนดประเภทของข้อมูลไว้ 2 แบบด้วยกันดังนี้คือ

### 2.6.1 ไบต์สถานะ ( Status Bytes )

เป็นรหัสคำสั่งของมีคี่ขนาด 8 บิต โดยที่บิตท้ายสำคัญสูงสุด (MSB) จะถูกเซตเป็น 1 ไบต์สถานะจะเป็นตัวบอกถึงประเภทของข้อความ และบอกให้รู้ว่าไบต์ข้อมูลที่จะตามมาจะเอา ไปใช้เพื่ออะไร ไบต์สถานะจะมีผลกับสถานะการทำงานต่างๆของระบบดังนี้

**สถานะทำงาน (Running Status)** เป็นสถานะสำหรับข้อความที่เกี่ยวกับเรื่องเสียงและ โหมดเท่านั้น กล่าวคือเมื่อด้านรับได้รับ ไบต์สถานะและประมวลผลแล้ว ด้านรับจะยังคงอยู่ในสถานะเดิมจนกว่าจะ ได้รับไบต์สถานะ ใหม่ที่ไม่เหมือนเดิม เพราะฉะนั้น ถ้ามีการส่ง ไบต์สถานะ เดิมเข้ามาอีกครั้งระบบจะ ไม่สนใจ ไบต์ดังกล่าวจนกว่าจะได้รับข้อมูลที่ถูกต้องเสียก่อน ภายใต้อสถานะทำงานข้อความที่สมบูรณ์จะต้องประกอบไปด้วยไบต์ข้อมูลที่ถูกต้องตามข้อกำหนด

สถานะทำงานจะหยุดเมื่อมีไบต์สถานะใหม่เข้ามา ยกเว้นกรณีที่ข้อความเกี่ยวกับ เวลาจริงเข้ามา มันจะทำการจัดจังหวะสถานะที่กำลังทำงานไว้เพียงชั่วคราวเท่านั้น

**สถานะที่ไม่สามารถทำงานได้ (Unimplement Status)** ทางด้านรับจะ ไม่สนใจ ไบต์สถานะที่มัน ไม่สามารถทำงานได้ และ ไม่สนใจ ไบต์ข้อมูลที่ตามมาจาก

**สถานะที่ไม่มีนิยามความหมาย (Undefined Status)** จะต้องไม่มีการใช้ไบต์สถานะที่ไม่มีนิยามความหมาย ซึ่งอาจเกิดจากในขณะเปิดเครื่องหรือปิดเครื่อง ถ้าหากมี

ไบต์สถานะดังกล่าวถูกรับได้ ระบบจะ ไม่รับรู้และ ไม่สนใจ ไบต์ข้อมูลที่ตามมาด้วย

### 2.6.2 ไบต์ข้อมูล ( Data Bytes )

เป็นส่วนของพารามิเตอร์ที่รับประกอบกับคำสั่ง ซึ่งจะถูส่งตาม ไบต์สถานะ โดยมีความยาวประมาณ 1-2 ไบต์ ( ยกเว้นข้อความที่เกี่ยวกับเวลาจริง ) ไบต์ข้อมูลมี 8 บิต โดยบิตที่มีนัยสำคัญสูงสุดจะถูกรีเซตเป็น 0 จำนวนและขอบเขตของ ไบต์ข้อมูล จะเป็นไปตามที่ไบต์สถานะกำหนด ในแต่ละ ไบต์สถานะจะส่งต้องส่ง ไบต์ข้อมูลจำนวนที่ถูก ต้องตามไปด้วยเสมอ การตอบสนองต่อข้อความจะมีได้ก็ต่อเมื่อด้านรับได้รับ ไบต์ข้อมูลตาม ต้องการ และทางด้านรับจะ ไม่สนใจ ไบต์ข้อมูลที่ตามหลัง ไบต์สถานะที่มันไม่รู้จัก

### 2.7 โหมดควบคุมช่องสัญญาณ ( Channel Modes )

เครื่องดนตรีประเภทซินทีไซเซอร์ (Synthesizers) ภายในจะประกอบด้วยองค์ประกอบพื้นฐานในการให้กำเนิดเสียงเรียกว่า voice โดย voice นี้มีหน้าที่ ในการทำการระบวนการสร้างเสียงจากข้อมูลการเล่นโน้ต (note on) หรือ การหยุดโน้ต (note off) ที่ส่งมาจากคีย์บอร์ด ไปให้กับตัวมัน จนสามารถเล่นโน้ตดนตรีได้อย่างถูกต้อง ด้วยจังหวะที่แม่นยำ

ในการควบคุมการทำงานของมิดี้ผ่านทางช่องสัญญาณมิดี้ทั้ง 16 ช่อง และการ กำหนดหน้าที่ให้กับ voice แต่ละตัวของซินทีไซเซอร์นั้น สามารถทำได้โดยใช้ข้อความ ควบคุมโหมดการทำงาน (Mode message) ทางมิดี้ อันได้แก่ Omni(On/Off), Poly และ Mono โดยที่ Omni นั้นใช้เลือกที่จะให้ซินทีไซเซอร์สนใจข้อมูลจากทุกช่องสัญญาณ หรือสนใจเพียงเฉพาะจากช่องที่กำหนด ส่วน Poly และ Mono จะเป็นการกำหนด จำนวนของ voice ที่ทำงานในแต่ละช่องสัญญาณซึ่งขึ้นอยู่กับการตั้ง Omni ด้วย สำหรับ Poly และ Mono นั้นเป็นโหมดตรงข้ามซึ่งกันและกันอยู่ หมายความว่า จะกำหนดให้เป็น

Poly หรือ Mono ได้เพียงอย่างเดียวเท่านั้น เมื่อเรารวมข้อความควบคุมโหมดเหล่านี้เข้าด้วยกัน เราจะได้เป็นโหมดการทำงาน 4 โหมดด้วยกัน คือ

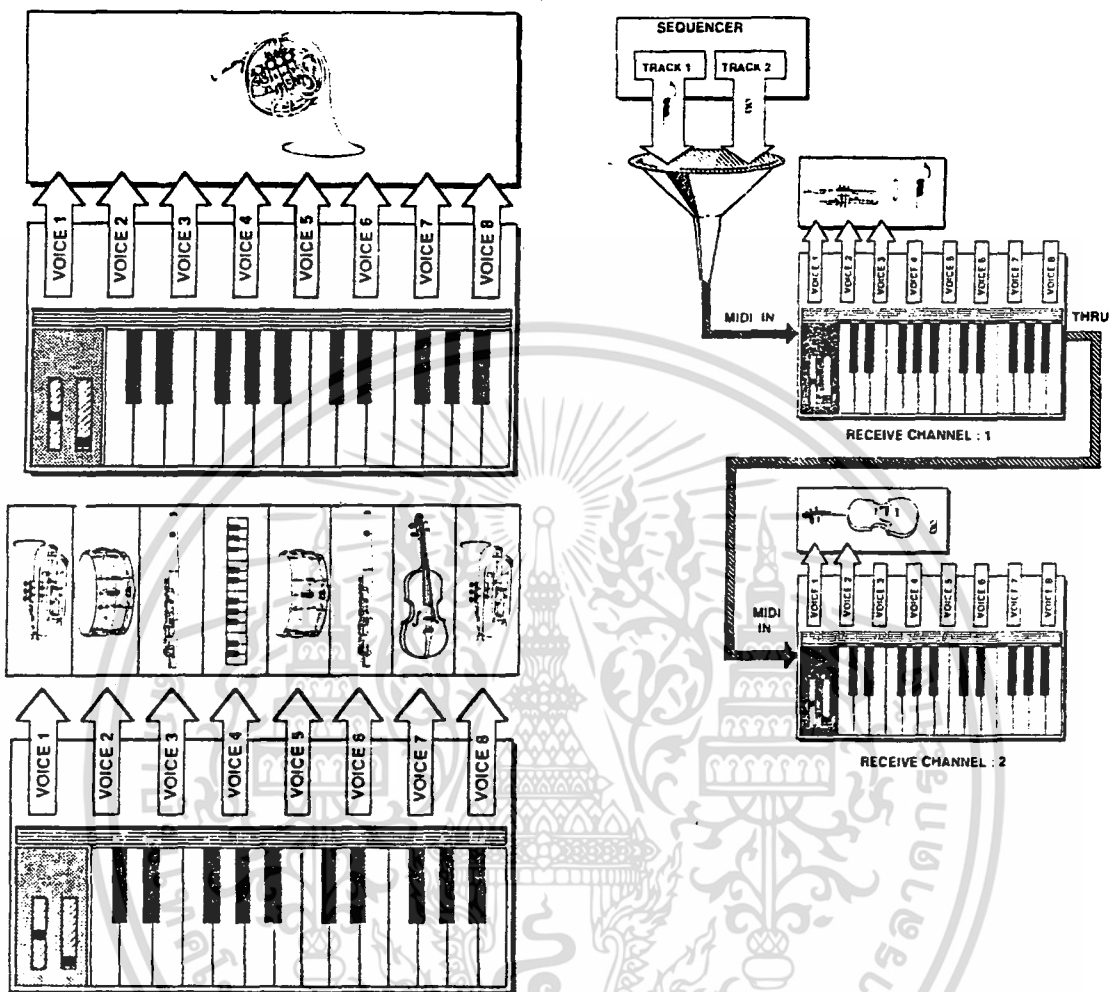
โหมด 1. Omni-on/Poly หรือ Omni mode ข้อความ Omni-on นั้นจะทำให้ขั้วที่ไซเซอร์ตอบสนองข่าวสารที่เข้ามาทางช่องสัญญาณทางมิติทั้ง 16 ช่อง ในขณะที่ Poly จะทำให้ขั้วที่ไซเซอร์ทำงานในแบบ Polyphonic ซึ่งกำหนดให้ voice แต่ละตัวสร้างเสียงประสานกันได้ตามข้อมูลที่ได้รับ

โหมด 2. Omni-on/Mono เป็นหนึ่งในจำนวนสองโหมดที่รู้จักในชื่อของ Mono mode ในโหมดนี้จะกำหนดให้ voice เพียงตัวเดียว ตอบสนองข่าวสารทั้งหมดที่เข้ามาทางช่องสัญญาณใดๆก็ตาม โหมดนี้ใช้สำหรับขั้วที่ไซเซอร์หรือเครื่องดนตรีที่เป็นระบบ Monophonic คือสามารถเล่นได้เพียงทีละ โน้ตเท่านั้น

โหมด 3. Omni-off/Poly หรือ Poly mode ขั้วที่ไซเซอร์ที่ทำงานในโหมดนี้จะตอบสนองเพียงข่าวสารที่เข้ามาทางช่องสัญญาณที่ถูกกำหนดไว้แล้วเท่านั้น ข้อความควบคุมเสียงจะถูกกำหนดให้เป็นแบบ Polyphonic โหมดนี้ใช้สำหรับควบคุมเครื่องดนตรีหลายๆ เครื่อง ให้เล่นท่วงทำนองทางดนตรีที่แตกต่างกันในเวลาเดียวกัน ซึ่งมักจะเป็นการควบคุมจากซีควเอนเซอร์ (sequencer) หรือเครื่องบันทึกข้อมูลทางดนตรี

โหมด 4. Omni-off/mono ในโหมดนี้แต่ละ voice ของขั้วที่ไซเซอร์จะถูกโปรแกรมให้ตอบสนองข่าวสารจากช่องสัญญาณที่แตกต่างกัน ซึ่งใช้สำหรับควบคุมเครื่องดนตรีที่เป็นระบบ multi-timbral คือสามารถสร้างเสียงเครื่องดนตรีต่างชนิดได้พร้อมๆกัน

เครื่องดนตรีระนาดคือเครื่องหนึ่ง จะสามารถทำการรับหรือส่งข้อมูลได้ทีละ โหมดเท่านั้น และโดยทั่วไปมักจะส่งและรับในโหมดเดียวกัน ในการรับข้อความควบคุม โหมดนั้น เครื่องดนตรีที่เป็นตัวรับจะรับได้ทางช่องสัญญาณที่อนุญาตเท่านั้น และโดยทั่วไปจะกำหนดให้เป็นช่องแรก หรือช่อง 1 นั้นเอง



รูปที่ 2.5 การทำงานของซินธิไซเซอร์ระบบมิดี้ ในโหมดต่างๆที่นิยมใช้กัน

2.8 ข้อความควบคุมเสียงของช่องสัญญาณ ( Channels Voice Message )

ข้อความต่างๆเหล่านี้ จะประกอบไปด้วยข้อมูลข่าวสารที่ใช้ในการควบคุม Voice ของซินธิไซเซอร์ ให้ทำงานตามระบบควบคุมต่างๆ ที่ติดต่อกับผู้เล่น เพื่อให้ซินธิไซเซอร์นั้น สามารถแปลงเสียงดนตรีได้ตามลักษณะการเล่นของผู้เล่น ซึ่งจะทำให้เสียงดนตรีที่ได้มีลักษณะที่สมจริง ระบบควบคุมดังกล่าวก็ได้แก่ เมโนทีมเปอร์ด , วงล้อควบคุมการขึ้นของเสียง ( Modulation Wheel ) และระบบควบคุมด้วยลมหายใจ ( Breath controller ) เป็นต้น ระบบต่างๆเหล่านี้ จะมีวงจรในการตรวจวัดปริมาณ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การเปลี่ยนแปลงทางกายภาพ เพื่อเปลี่ยนไปเป็นข้อมูลในการควบคุม ซึ่งใช้ประกอบเป็น  
ข้อมูลสำหรับแต่ละช่องสัญญาณ ตามแต่จะกำหนด



### 2.8.1 การเล่นโน้ต ( Note on )

ข้อความนี้จะถูกส่งออกไป เมื่อมีการกดคีย์บอร์ดของซินธิไซเซอร์ ประกอบ  
ไปด้วยข้อมูล 3 ไบต์ได้แก่

- ไบต์ที่ 1 เป็นรหัสคำสั่งและหมายเลขช่องสัญญาณของ โน้ตที่ถูกเล่น
- ไบต์ที่ 2 เป็นหมายเลขของ โน้ตหรือรหัสคีย์ของ โน้ตที่ถูกเล่น มีค่าตั้งแต่  
0 ถึง 127 โดย 0 จะเป็นคีย์โน้ตตัวที่ต่ำสุด 127 จะเป็นคีย์  
โน้ตตัวที่สูงสุด และ 60 จะเป็นคีย์โน้ตคอกกลาง (middle c)
- ไบต์ที่ 3 เป็นความเร็วในการกดคีย์ มีค่าตั้งแต่ 0 ถึง 127 โดย  
0 หมายถึงไม่มีการกดคีย์บอร์ด และ 127 จะหมายถึงการกด  
ด้วยความเร็วสูงสุด สำหรับคีย์บอร์ดที่ไม่มีการตรวจจับความเร็ว  
จะถูกกำหนดให้เป็น 64 ความเร็วในการกดคีย์ ซึ่งสัมพันธ์กับการ  
เปลี่ยนแปลงในการเล่น จะเป็นตัวกำหนดระดับความดังโดยรวม  
ของเสียง

1	0	0	1	channel
---	---	---	---	---------

0	note number
---	-------------

0	velocity
---	----------

←----- 7 ----->

### รูปที่ 2.6 แสดงข้อความการเล่นโน้ต ( Note on )

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ข้อความเหล่านี้สามารถถูกกำหนดให้เป็น polyphonic ได้ในแต่ละช่อง แม้ว่าจะมีข้อมูลโน้ตหลายตัว ซึ่งถูกส่งแบบอนุกรมทิศทางกันมา แต่ด้วยความเร็วที่สูงมาก เราจะได้ยินเสียงของโน้ตเหล่านั้นเกิดขึ้นพร้อมๆกัน

### 2.8.2 การหยุดโน้ต ( Note off )

เป็นข้อความที่ถูกส่งออกไป เมื่อเราทำการยกนิ้วขึ้นจากคีย์บอร์ดที่ถูกกดอยู่ ประกอบด้วยข้อมูล 3 ไบต์ได้แก่

ไบต์ที่ 1 เป็นรหัสคำสั่งและหมายเลขช่องสัญญาณของโน้ตที่จะหยุดเล่น

ไบต์ที่ 2 เป็นหมายเลขหรือรหัสมีคี่ของ โน้ตตัวที่จะหยุดเล่น

ไบต์ที่ 3 เป็นความเร็วในการปล่อยคีย์ มีลักษณะเดียวกับความเร็วในการกดคีย์ ข้อมูลนี้จะบอกได้ว่าเราปล่อยคีย์เร็วเพียงใด ซึ่งจะ ทำให้ช่วงเวลาของสภาวะการปล่อยโน้ต ถูกควบคุมได้จากรูปแบบการเล่น

1	0	0	0	channel
---	---	---	---	---------

0	note number
---	-------------

0	release velocity
---	------------------

←————— 7 —————→

รูปที่ 2.7 แสดงข้อความการหยุดโน้ต ( Note off )

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

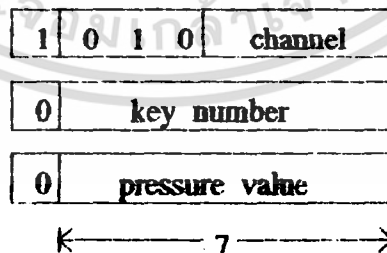
2.8.3 การกำหนดแรงกดของคีย์แบบประสานเสียง ( Polyphonic Key Pressure )

นอกจากการตอบสนองจากความเร็วในการกดคีย์ มิติยังได้กำหนดให้มีการตรวจสอบแรงกดของคีย์บอร์ดอีกด้วย เมื่อคีย์บอร์ดถูกกด แรงกดของคีย์ในแต่ละคีย์ที่ถูกกด จะทำให้ได้ข้อมูล 3 ไบต์ เป็นข้อมูลเฉพาะในแต่ละคีย์ อันประกอบด้วย

ไบต์ที่ 1 เป็นหมายเลขช่องสัญญาณที่ข้อมูลแรงกดคีย์นั้นจะถูกส่ง

ไบต์ที่ 2 เป็นหมายเลขของโน้ตตัวที่ทำให้เกิดแรงกดคีย์นั้น ซึ่งมีค่าอยู่ระหว่าง 0-127 โดยที่ 0 เป็นโน้ตตัวที่ต่ำที่สุด และ 127 เป็นโน้ตตัวที่สูงที่สุด ส่วน 60 จะเป็นโน้ตโดกลาง (middle c)

ไบต์ที่ 3 เป็นค่าแรงกดของคีย์ มีค่าอยู่ระหว่าง 0-127 การตรวจสอบแรงกดแบบนี้ เป็นแบบที่ต้องติดตั้งตัวตรวจวัดแรงกดกับทุกๆคีย์ของแผงคีย์บอร์ด จึงเป็นวิธีที่ไร้นั่นสูง ดังนั้นคีย์ไฮเซอร์ส่วนใหญ่จะไม่ใช่วิธีนี้แต่จะใช้วิธีตรวจสอบแรงกดโดยรวมของคีย์แทน ซึ่งเป็นหัวข้อที่เราจะกล่าวต่อไป



รูปที่ 2.8 แสดงข้อความกำหนดแรงกดคีย์แบบประสานเสียง ( Polyphonic Key Pressure )

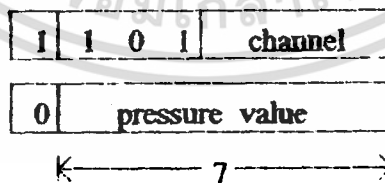
**2.8.4 การกำหนดแรงกดรวม ( Overall Pressure หรือ Channel After-touch )**

การตรวจสอบแรงกดด้วยวิธีนี้ เป็นการกำหนดแรงกดคีย์ของช่องสัญญาณ กล่าวคือ ใช้การเฉลี่ยค่าแรงกดคีย์ของคีย์ทั้งหมดที่ถูกกด ให้เป็นข้อมูลแรงกดคีย์ของช่องสัญญาณที่กำหนด ข้อความนี้ประกอบด้วยข้อมูล 2 ไบต์ ได้แก่

ไบต์ที่ 1 เป็นรหัสคำสั่งและหมายเลขของช่องสัญญาณที่ข้อมูลแรงกดคีย์นั้นจะ ถูกส่ง

ไบต์ที่ 2 เป็นค่าแรงกดคีย์ของช่องสัญญาณ ซึ่งมีค่าระหว่าง 0-127

การกำหนดแรงกดของคีย์ทั้งสองแบบนี้ มีกรู๊ปกันในเรื่องของ After-Touch อันเป็นที่หนึ่งในสองของการตรวจสอบการสัมผัส (Touch sensitivity) ของคีย์บอร์ด ซึ่งประกอบด้วย การตรวจสอบความเร็ว (Velocity sensitivity) และการตรวจสอบแรงกด (Pressure sensitivity) ระยะเวลาความดังกล่าวนั้นเป็นส่วนสำคัญอย่างยิ่ง ในการที่จะทำให้เครื่องดนตรีไฟฟ้า เล่นเสียงที่ให้อารมณ์และความรู้สึกเช่นเดียวกับเครื่องดนตรีจริง



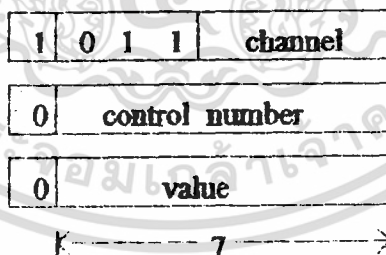
**รูปที่ 2.9 แสดงข้อความกำหนดแรงกดรวม ( Overall Pressure หรือ Channel After-touch )**

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 2.8.5 การเปลี่ยนแปลงการควบคุม ( Control Change )

เป็นการส่งข้อมูลสำหรับตัวควบคุม (controller) ต่างๆในระบบ ข้อมูลเหล่านี้อาจใช้ในการควบคุมการเร่งหรือใช้สำหรับแก้ไขเปลี่ยนแปลงข้อมูลใดๆ ซึ่งขึ้นอยู่กับหน้าที่ของตัวควบคุมเหล่านั้น มีข้อกำหนดให้ตัวควบคุมได้ถึง 128 ตัว (0-127) ข้อความนี้ประกอบด้วยข้อมูล 3 ไบท์ ได้แก่

- ไบท์ที่ 1 เป็นรหัสคำสั่งและหมายเลขช่องสัญญาณที่ข้อมูลควบคุมจะถูกส่ง
- ไบท์ที่ 2 เป็นหมายเลขของตัวควบคุม (controller number) มีค่าตั้งแต่ 0 ถึง 127 (รายละเอียดอ่านได้จากภาคผนวกท้ายเล่ม)
- ไบท์ที่ 3 เป็นค่าข้อมูลของตัวควบคุมนั้นๆ ตัวควบคุมหมายเลข 122-127 นั้นจะส่งวนไว้ให้กับ Control Mode Message ซึ่งจะกล่าวถึงในภายหลัง



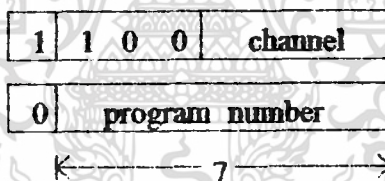
รูปที่ 2.10 แสดงข้อความเปลี่ยนแปลงการควบคุม ( Control Change )

### 2.8.6 การเปลี่ยนแปลงโปรแกรมเสียง ( Program Change )

ในระบมิตินี้ เราสามารถควบคุมการเปลี่ยนแปลง โปรแกรมของเสียง จากเสียงหนึ่ง ไปสู่อีกเสียงหนึ่งได้ โดยทำการส่งข้อความขอเปลี่ยนแปลง โปรแกรมไปยังเครื่องดนตรีตัวที่ต้องการ ข้อความการเปลี่ยนแปลง โปรแกรม จะประกอบด้วยข้อมูล 2 ไบท์ ได้แก่

ไบท์ที่ 1 เป็นรหัสคำสั่งและหมายเลขช่องสัญญาณของอุปกรณ์ตัวที่ต้องการให้มีการเปลี่ยนแปลง โปรแกรม

ไบท์ที่ 2 เป็นหมายเลขของโปรแกรมที่ถูกเลือก มีค่าตั้งแต่ 0 ถึง 127



รูปที่ 2.11 แสดงข้อความเปลี่ยนแปลงโปรแกรมเสียง ( Program Change )

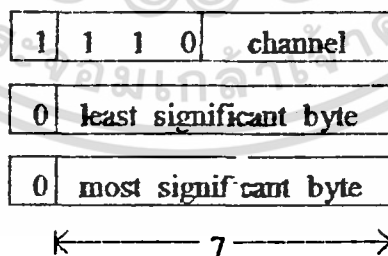
### 2.8.7 การเปลี่ยนแปลงระดับความถี่เสียงอย่างต่อเนื่อง ( Pitch Bend )

เพื่อที่จะเลียนแบบลักษณะการเปลี่ยนแปลงเสียง ของตัวโน้ตบนเครื่องดนตรีจริงบางชนิด เช่น การเล่นโน้ตโดยวิธีการคันสายของกีตาร์, การใช้คันชักของทรมโบ เป็นต้น เสียงของตัวโน้ตดนตรีที่เกิดขึ้นจากการเล่นด้วยวิธีการเหล่านี้ จะมีการเปลี่ยนแปลงอย่างต่อเนื่องจากตัวโน้ตหนึ่ง ไปสู่อีกตัวโน้ตหนึ่ง มีได้กำหนดข้อความที่ใช้ตอบสนอง การกำเนิดเสียงในลักษณะดังกล่าวไว้ ซึ่งประกอบด้วยข้อมูล 3 ไบท์ โดยที่

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- ไบท์ที่ 1 เป็นรหัสคำสั่งและหมายเลขช่องสัญญาณของข้อมูลควบคุมที่จะถูกส่ง
- ไบท์ที่ 2 เป็นค่าการเปลี่ยนแปลงระดับความถี่เสียงอย่างต่อเนื่อง (Pitch และ bend Value) โดยไบท์ที่ 2 เป็นไบท์ที่นิยมสำคัญต่ำ
- ไบท์ที่ 3 ส่วนไบท์ที่ 3 นั้นเป็นไบท์ที่นิยมสำคัญสูง และโดยทั่วไปแล้ว มักจะใช้เฉพาะ ไบท์ที่นิยมสำคัญสูง จึงมีค่าการเปลี่ยนแปลงได้จาก 0-127 โดยที่ค่า 64 จะหมายถึง ไม่มีการเปลี่ยนแปลง ดังนั้นค่าที่มากกว่าหรือน้อยกว่า 64 จึงมีผลทำให้โน้ตตัวที่เล่นอยู่มีเสียงสูงขึ้นหรือต่ำลงตามลำดับ ส่วนไบท์ที่ 2 นั้นจะถูกส่งด้วยค่า 0

ตามที่กล่าวมาเป็นการทำงานในแบบความละเอียดต่ำ (Low resolution) สำหรับการทำงานในแบบความละเอียดสูง (High resolution) นั้นจะใช้ไบท์ที่ 2 ด้วย แต่ไม่ค่อยได้รับความนิยมมากนัก โดยทั่วไปแล้ว pitch bend value นี้จะถูกควบคุมจากวงล้อควบคุม (Wheel) หรือจอยสติค (Joystick)



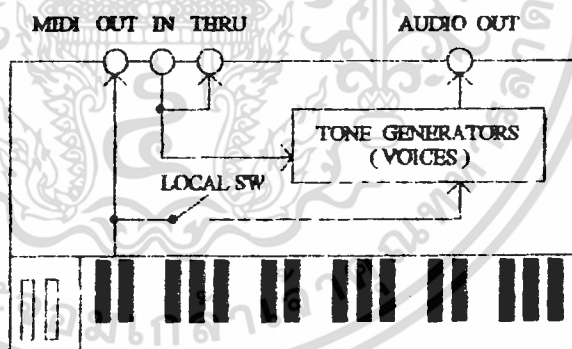
รูปที่ 2.12 แสดงข้อความเปลี่ยนแปลงระดับความถี่เสียงอย่างต่อเนื่อง  
( Pitch Bend )

**2.9 ข้อความควบคุมโหมดการทำงานของช่องสัญญาณ ( Channel Mode Massage )**

ข้อความเหล่านี้มีผลกับโหมดการทำงานของช่องสัญญาณ และถือเป็นการตี  
พิเศษของการเปลี่ยนแปลงการควบคุม (Control Change) ซึ่งได้กล่าวไปแล้วในหัวข้อ

2.8.5 ดังนั้น ไบท์สถานะของข้อความเหล่านี้ ซึ่งเป็นส่วนของรหัสคำสั่งและช่องสัญญาณ  
จะเหมือนกัน และไบท์ที่ตามมาจะต้องเป็นหมายเลขของโหมดการทำงานนั้นๆ ซึ่งความ  
จริงก็คือหมายเลขของตัวเอง

**2.9.1 การควบคุมโหมดการทำงานของคีย์บอร์ด ( Local/Remote Keyboard Control )**



**รูปที่ 2.13 แสดงโครงสร้างการทำงานของซินธิไซเซอร์ระบบคีย์**

เป็นข้อความที่เกี่ยวกับโหมดการทำงาน เพื่อเลือกว่าจะให้ซินธิไซเซอร์  
ทำงานในโหมด Local หรือ Remote จากรูป 2.13 ถ้าเป็นการควบคุมในโหมด Local

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ข้อมูลควบคุมเสียงจากคีย์บอร์ดจะถูกส่งไปยังส่วนให้กำเนิดเสียง และพร้อมกันนั้น ข้อมูลเดียวกันนี้จะถูกส่งออกไปทาง MIDI OUT ด้วย ถ้ามีการเชื่อมต่อทางมีตรีระหว่างเครื่องดนตรีดังในรูป 2.1 เมื่อเล่นคีย์บอร์ดตัวที่ 1 ก็จะมีเสียงออกมาจากคีย์บอร์ดทั้งตัวที่ 1 และ ตัวที่ 2 แต่ถ้าเป็นการควบคุมแบบ Remote แล้ว ข้อมูลจากคีย์บอร์ดจะถูกส่งไปให้แค่เฉพาะกับ MIDI OUT เท่านั้น ดังนั้น เมื่อมีการเล่นคีย์บอร์ดตัวที่ 1 จะมีแค่เสียงที่ออกมาจากคีย์บอร์ดตัวที่ 2 เพียงตัวเดียว

โดยทั่วไป เราจะควบคุมการทำงานนี้ได้โดยผ่านทางสวิทช์ Local on/off ซึ่งมักเป็นสวิทช์ทางซอฟต์แวร์ Local on นั้นจะหมายถึง การทำงานในโหมด Local ส่วน Local off จะเป็นการทำงานในโหมด Remote

การควบคุมโหมดการทำงานของคีย์บอร์ด จะใช้ข้อมูล 3 ไบต์ ประกอบด้วย ไบต์ที่ 1 เป็นรหัสคำสั่งและหมายเลขช่องสัญญาณ ที่ข้อมูลการควบคุมจะถูกส่ง  
 ไบต์ที่ 2 เป็นหมายเลขตัวควบคุม ไบต์นี้คือ 122  
 ไบต์ที่ 3 ในโหมด Remote ไบต์นี้จะเป็น 0 แต่สำหรับใน Local ไบต์นี้จะเป็น 127

### 2.9.2 การหยุดเล่นโน้ตทุกตัว ( All Note off )

เป็นการหยุดเล่นโน้ตทุกตัวที่เล่นอยู่ในขณะนั้น ประกอบด้วยข้อมูล 3 ไบต์  
 ได้แก่

ไบต์ที่ 1 เป็นรหัสคำสั่งและหมายเลขช่องสัญญาณที่ข้อมูลการควบคุมจะถูกส่ง  
 ไบต์ที่ 2 เป็นหมายเลขตัวควบคุม ไบต์นี้คือ 123  
 ไบต์ที่ 3 เป็นไบต์ที่ไม่มีความหมายอะไร เป็นการใส่ข้อมูล 3 ไบต์เท่านั้น เนื่องจากข้อความเกี่ยวกับการควบคุมมีขนาด 3 ไบต์

### 2.9.3 การเลือกโหมดทำงาน ( Omni/Poly/Mono Select )

เราได้อธิบายถึงโหมดการทำงานเหล่านี้แล้ว ในหัวข้อ 2.7 ในการส่งข้อความเปลี่ยนแปลง โหมดนั้น จะเป็นการหยุดการเล่นโน้ตทุกตัว (All Note off) ลงด้วย ข้อความที่ประกอบด้วย

ไบท์ที่ 1 เป็นรหัสคำสั่งและหมายเลขของสัญญาณที่ข้อมูลการควบคุมจะถูกส่ง

ไบท์ที่ 2 เป็นหมายเลขของตัวควบคุมในแต่ละ โหมด มีดังนี้

Omni Off = 124

Omni On = 125

Mono On = 126

Poly On = 127

ไบท์ที่ 3 สำหรับตัวควบคุมหมายเลข 123, 124, 125 และ 127 ไบท์นี้จะ

เป็น 0 แต่ในกรณีของตัวควบคุมหมายเลข 126 (Mono) ไบท์นี้

จะเป็นการกำหนดหมายเลขของสัญญาณที่จะให้เป็น Mono

### 2.10 ข้อความทั่วไปของระบบ ( System Common Message )

เนื่องจากข้อมูลนี้เป็นข้อมูลของระบบ จึงไม่มีการกำหนดหมายเลขของสัญญาณ นั่นคือทุกช่องสัญญาณจะต้องตอบสนองต่อข้อมูลทั่วไปของระบบ

#### 2.10.1 การชี้ตำแหน่งของเพลง ( Song Position Pointer )

บอกให้รู้ถึงตำแหน่งของเพลงที่กำลังเล่นอยู่ การบอกตำแหน่งจะบอกว่า เพลงได้ดำเนินไปกี่บีต (MIDI beat) บอกได้สูงสุด 16,384 บีต โดย 1 บีตมีค่าเท่ากับโน้ตซิมล์ 2 ชั้น 1 ตัว

### 2.10.2 การเลือกเพลง ( Song Select )

คล้ายๆกับการเลือกโปรแกรมเสียง แต่แทนที่จะเป็นโปรแกรมเสียงจะเป็นโปรแกรมรูปแบบ (pattern) ส่วนๆของเพลงแทน (ความยาวไม่กั้ห้อง) โดยมากจะใช้ในกล่องอิเล็กทรอนิกส์ แต่ละเพลงจะมีหมายเลขประจำตัว เช่น เพลงที่ 1 เป็น จังหวะ ร็อคและเพลงที่ 2 เป็นจังหวะวอลท์ เป็นต้น

### 2.10.3 ข้อความพิเศษของระบบ ( System Exclusive )

เป็นข้อความที่ใ้บอกว่ ข้อมูลที่จะทำการส่งต่อไปเป็นข้อมูลเฉพาะของแต่ละบริษัท ซึ่งแต่ละบริษัทจะมีหมายเลขประจำตัวของตนเอง

### 2.11 ข้อความเกี่ยวกับเวลาจริงของระบบ ( System Real-time Message )

เป็นข้อความที่ทำให้เครื่องดนตรีต่างๆที่ค้่อยู่ด้วยกัน เล่นอยู่ในจังหวะเดียวกัน (Synchronized) เช่น ในการพ่วงซินทีไซเซอร์กับ กลองอิเล็กทรอนิกส์เข้าด้วยกันจำเป็นต้องทำงานไปพร้อมๆกัน เพื่อให้เกิดบทเพลงที่สอดคล้องประสานคล้้องจองกัน

#### 2.11.1 การรีเซทระบบ ( System Reset )

เป็นข้อความที่บอกให้อุปกรณ์ระบบอิเล็กทรอนิกส์ทำการรีเซทตัวเอง (กลับไปอยู่ในสภานปรกหลังจากการเปิดไฟเข้าเครื่อง)

#### 2.11.2 สัญญาณนาฬิกา ( Timing Clock )

ในการกำหนดจังหวะจะมีการส่งสัญญาณนาฬิกาออกไปด้วยอัตรา 24 ลูกค้อนโน้ตตัวค่าหนึ่งตัว (ส่งออกไปอย่างต่อเนื่อง) เพื่อให้เป็นจังหวะอ้างอิงในระบบ ครวรีใช้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สัญญาณนาฬิกาเดียวกัน (มีอุปกรณ์การส่งสัญญาณนาฬิกาเพียงตัวเดียว)

### 2.11.3 การกลับไปจุดเริ่มต้น ( Start From First Messure )

เป็นข้อความที่บอกให้คนที่ไซเซอร์ และกล่องอิเล็กทรอนิกส์กลับไปยังจุดเริ่มต้นของเพลง แล้วเริ่มเล่นเพลง ในทันทีที่ได้รับสัญญาณนาฬิกา นั่นคือจะบอกให้แต่ละอุปกรณ์เริ่มเล่นพร้อมๆกัน

### 2.11.4 การหยุด ( Stop )

ข้อความนี้ จะบอกให้หยุดการกระทำทุกอย่างที่มีผลมาจากการกำหนดจังหวะ (timing-sensitive)

### 2.11.5 การเริ่มเล่นต่อ ( Continue Start )

เป็นการบอกให้อุปกรณ์ทำการเล่นต่อหลังตำแหน่งเพลงที่ถูกหยุดครั้งสุดท้าย

### 2.11.6 การตรวจสอบสถานะการทำงาน ( Active Sensing )

เป็นข้อความที่ใช้แก้ปัญหาตัวส่งตัวรับถูกตัดขาดออกจากกัน (เช่นสายหลุด) ซึ่งจะก่อให้เกิดความผิดพลาด เช่นส่งข้อความให้เล่นโน้ตออกไปแล้วเกิดสายหลุด เมื่อทางด้านส่งส่งข้อความให้เลิกเล่นโน้ตตามไป ก็จะไปไม่ถึงด้านรับ ทำให้เกิดเสียงค้างที่เครื่องดนตรีด้านรับ ปัญหานี้จะเกิดความเสียหายมากในการแสดงสด

Active sensing นั้นจะทำงานโดยส่งข้อความที่เรียกว่า active sensing ออกไปเรื่อยๆ เมื่อด้านรับได้รับ active sensing แล้ว ถ้าภายในระยะเวลาสั้นๆที่กำหนด ด้านรับยังไม่ได้รับข้อความนี้เข้ามาอีก มันจะหยุดทำการสร้าง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## เสียงในทันที

ทั้งหมดที่ได้กล่าวมาแล้วนั้น เป็นข้อกำหนดและความหมายต่างๆในทางมิดีโดยสังเขป รายละเอียดต่างๆเหล่านี้ จะทำให้เราเข้าใจถึงระบบการทำงานของมัน และสามารถนำมาประยุกต์ใช้ในงานต่างๆ ได้เช่น

- สามารถควบคุมให้เครื่องดนตรีไฟฟ้าหลายๆเครื่อง เล่นไปพร้อมๆกันได้ เพื่อเพิ่มขีดความสามารถในการสร้างเสียง
- สามารถทำให้เครื่องดนตรีไฟฟ้าต่างชนิด ต่างรุ่น หรือต่างบริษัทผู้ผลิต สามารถทำงานร่วมกันได้
- สามารถใช้ในการประพันธ์เพลงหรือบันทึกเสียง โดยการบันทึกข้อมูลของตัวโน้ตในเพลงเป็นรหัสมิดี ซึ่งช่วยลัดขั้นตอนและเวลาองได้มาก
- สามารถเชื่อมต่อเครื่องดนตรีไฟฟ้าเข้ากับเครื่องคอมพิวเตอร์ได้ เพื่อเพิ่มขีดความสามารถในการควบคุม และการแก้ไขปรับปรุงเสียง
- สามารถนำไปประยุกต์ใช้กับระบบอื่นๆ เช่น การควบคุมระบบไฟถนนเวที การแสดง, การแสดงผลของปริมาณสารเคมีในน้ำปัสสาวะ และในอีกจุดหนึ่งได้มีการส่งข้อมูลมิดีผ่านระบบการสื่อสารอย่าง โมเด็ม (Modem) หรือ ดาวเทียม (satellite) เพื่อการบันทึกเสียงทางไกลเกิดขึ้นแล้ว

สำหรับในโครงการพิเศษนี้ เป็นเพียงการนำบางส่วนของมันมาประยุกต์ใช้ในระบบการควบคุมเครื่องดนตรีไฟฟ้าด้วยเสียงของคน หรือเสียงจากเครื่องดนตรีจริงอื่นๆ ซึ่งจะขอกล่าวรายละเอียดในบทต่อไป

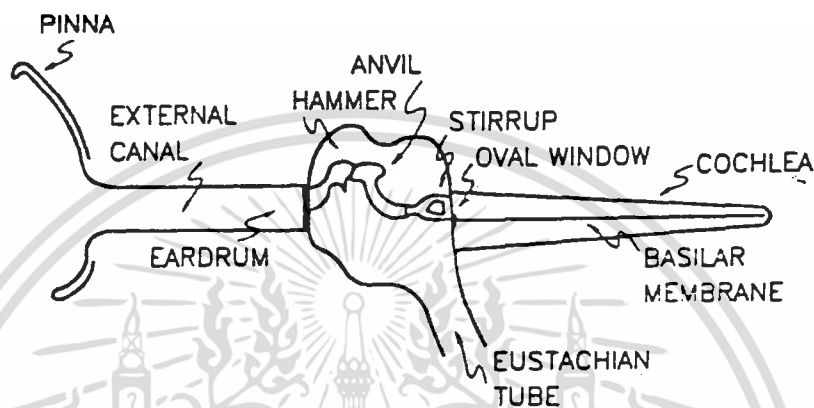
### บทที่ 3

#### แนวความคิดและหลักการทำงาน

จากเนื้อหาในบทนำนั้น เราได้กล่าวถึงแนวความคิดเบื้องต้น ของเครื่องแปลงระดับความถี่เสียงไปเป็นข้อมูลทางมิดี หรือ Pitch-to-MIDI Converter ไปบ้างแล้ว สำหรับในบทนี้จะกล่าวถึงรายละเอียดต่างๆ รวมทั้งทฤษฎีและหลักการที่ใช้สนับสนุนแนวความคิดในการออกแบบ Pitch-to-MIDI converter ของเรา และเพื่อความคล่องตัวในการนำเสนอ ในบางครั้งเราจะขอใช้อักษรย่อ PTMC แทนความหมายของเครื่องแปลงระดับความถี่เสียงไปเป็นข้อมูลทางมิดี ตลอดเนื้อหาของโครงการนี้

#### 3.1 ระดับความถี่เสียงและการได้ยิน

เสียงที่ถูกรับโดยหูและถูกประมวลผลโดยระบบการฟัง ซึ่งเชื่อมต่อจากหูชั้นนอกจนถึงเยื่อหุ้มสมอง หูจะกระทำตัวเสมือนทรานสดิวเซอร์ (Transducer) ซึ่งแปลงการเคลื่อนที่เชิงกลของรูปแบบการสั่นไปเป็นการกระตุ้นทางประสาท รูปที่ 3.1 จะแสดงส่วนสำคัญของระบบการทำงานนี้เมื่อเสียงเดินทางมาถึงหู มันจะเดินทางลงไปตามช่องภายนอก (External canal) ไปสู่แก้วหู ซึ่งจะส่งการสั่นนั้นไปสู่กระดูก 3 ชิ้นที่เรียงต่อกัน อันประกอบด้วย กระดูกรูปฆ้อง (Hammer) กระดูกรูปทั่ง (Anvil) และกระดูกรูปโกลน กระดูกรูปโกลนจะส่งการเคลื่อนไหวนั้นไปสู่เนื้อเยื่อที่เรียกว่า oval window ซึ่งเป็นเหตุให้คลื่นซึ่งกำลังเดินทางแพร่ไปในของไหลที่อยู่ในส่วนที่เรียกว่า Cochlea ให้ไปกระตุ้นคนที่อยู่บนเนื้อเยื่อที่เรียกว่า Basilar membrane เส้นประสาทที่เชื่อมต่อกับคนเหล่านี้ ไปสู่สมอง เมื่อมีการสั่นแบบคลื่นรูปไซน์ที่มีความถี่ซึ่งอยู่ในช่วงที่สามารถรับฟังได้เข้ามาถึงหูชั้นใน มันจะทำการกระตุ้นปลายประสาทที่อยู่บน Basilar membrane เป็นสัดส่วนกับความถี่ของเสียงนั้น ความรู้สึกถึงความดังถูกกำหนดจากปริมาณพลังงานซึ่งรับได้โดยหู



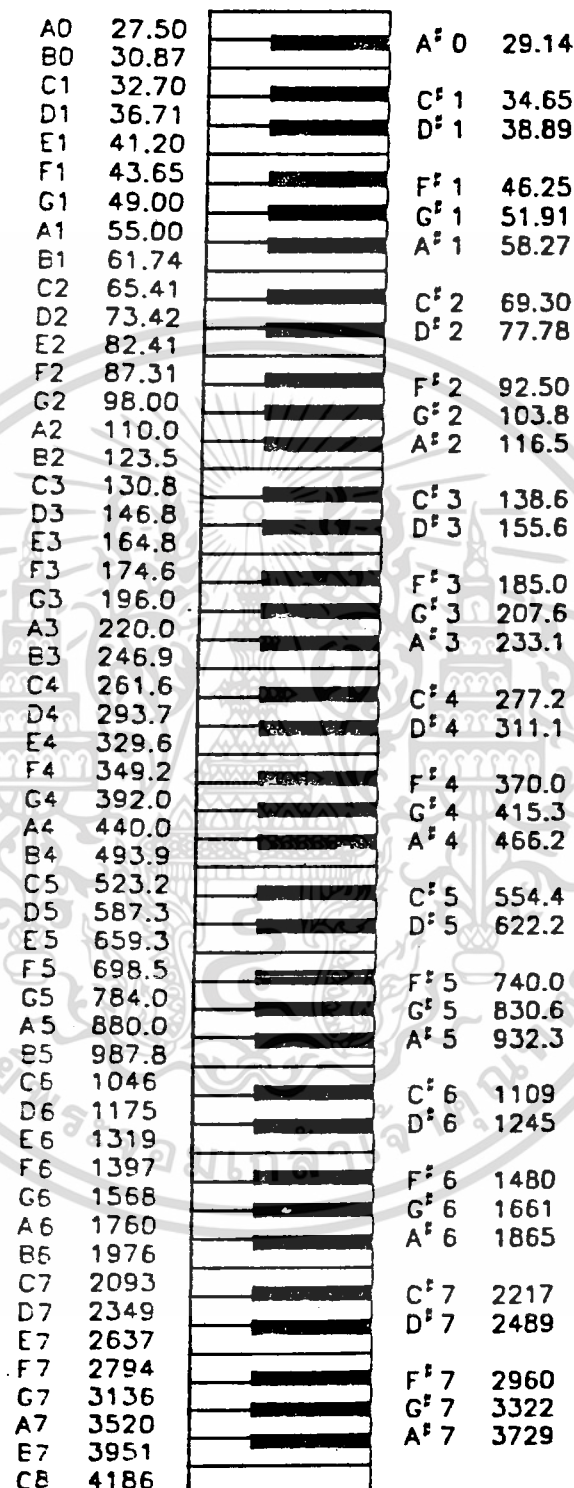
รูปที่ 3.1 แสดงกลไกการได้ยินของมนุษย์

ระดับความถี่ของเสียงหรือพิทช์ (Pitch) เป็นพารามิเตอร์ที่ตอบสนองตามความถี่ มันเป็นความรู้สึกหลัก ซึ่งเป็นผลมาจากการกระตุ้นปลายประสาทบน Basilar membrane

ขอบเขตโดยประมาณของความถี่ที่มนุษย์สามารถรับฟังได้อยู่ระหว่าง 20 - 20,000 เฮิรตซ์ ความถี่ตั้งแต่ 20 จนถึง 4000 เฮิรตซ์ ประกอบด้วยย่านที่ไวต่อการได้ยิน และไวต่อการเปลี่ยนแปลงความถี่ที่สุด ย่านความถี่นี้จะครอบคลุมสองในสามส่วนของ Basilar membrane ขณะที่ความถี่สูงส่วนที่เหลือจะติดต่อกับเพียงหนึ่งในสาม ภายในย่านที่ความไวสูงสุดนั้น การแยกแยะความถี่สำหรับคลื่นเสียงเป็นสัดส่วนที่เอวองที่ความถี่ต่ำ และจะยิ่งยากขึ้น ถ้าช่วงเวลาที่ได้รับสั้นลง

ความสัมพันธ์ระหว่างความสามารถในการรับรู้ความถี่จะเป็นแบบไม่เชิงเส้น การกระตุ้นบน basilar membrane จะปรากฏเป็นจุดที่มีลักษณะเกือบจะเป็นสัดส่วนกับลอการิทึมของความถี่ สำหรับความถี่ที่สูงกว่า เมื่อความถี่เพิ่มขึ้น 2 เท่า

ระยะทางระหว่างจุดที่ถูกกระตุกบน Baislar membrane จะเปลี่ยนไปเท่ากับระยะทางคงที่โดยประมาณค่าหนึ่ง ดังนั้นจึงสามารถเปรียบเทียบเสียงได้จากทฤษฎีของคาบทางดนตรีซึ่งแบ่งแยกมันออกจากกัน นั่นก็คือใช้อัตราส่วนของความถี่ของมันเอง ตัวอย่างเช่น ช่วงของระดับความถี่เสียงที่ห่างกัน 1 ออกเตฟ จะมีอัตราส่วนของความถี่เป็น 2:1 เมื่อเราได้ยินมันในลักษณะความต่อเนื่องทางความถี่ มันก็จะให้ความรู้สึกที่เหมือนกัน ในรูปที่ 3.2 ได้แสดงความถี่ของลิ้นนิ้วแต่ละตัวของเปียโน ตีเปียโนนั้นถูกกำกับด้วยชื่อของระดับความถี่เสียงต่างๆ ซึ่งจะถูกใช้อยู่ในโครงงานนี้เช่นกัน จากที่กล่าวมาแล้วทั้งหมด เราที่พอจะสรุปได้คือว่า ระดับความถี่เสียงหรือพิทช์ นั่นก็คือ ความถี่ของโน้ตทางดนตรี นั่นเอง จากรูปจะเห็นได้ว่า ความถี่ของโน้ต A3 (220 เฮิรต) นั้นเป็นครึ่งหนึ่งของความถี่ของโน้ตตัวที่อยู่สูงกว่ามัน 1 ออกเตฟซึ่งก็คือ A4 (440 เฮิรต) พอดีและความถี่ของ A4 ก็เป็นครึ่งหนึ่งของความถี่ของ A5 (880 เฮิรต) เช่นกัน จะสังเกตเห็นว่าผลต่างของความถี่ระหว่างโน้ต 2 ตัวใดๆที่อยู่ติดกันจะไม่เท่ากัน นั่นหมายความว่าเราไม่สามารถหาความถี่ของโน้ตตัวหนึ่งจากความถี่ของโน้ตตัวที่อยู่ก่อนหน้านั้น ด้วยการบวกค่าคงที่ใดๆได้ เนื่องจากค่าในหนึ่งออกเตฟความถี่ของโน้ตแต่ละตัวจะเป็นอัตราส่วนกับความถี่ของโน้ตตัวแรกในบันไดเสียง (Unison) จะเห็นได้จากตารางที่ 3.1 ซึ่งแสดงอัตราส่วนของความถี่ระหว่างโน้ตตัวแรกของบันไดเสียงกับโน้ตตัวอื่นๆในระนาบต่างๆ แต่ที่ได้รับความนิยมนที่สุด และเป็นระบบที่เราใช้อ้างอิงในโครงงานนี้ก็คือ Equal temperament ในทางดนตรีนั้น บันไดเสียงโครมาติก (Chromatic scale) จะประกอบด้วยตัวโน้ตทั้งหมด 12 ตัว Temperament นั่นก็คือ ความถี่ที่ถูกต้องของโน้ตทั้ง 12 ตัวนั่นเอง ในตารางที่ 3.1 คอลัมน์แรกสุดจะเป็นชื่อของตัวโน้ตซึ่งบอกความสัมพันธ์ของมันกับโน้ตตัวแรก เช่น Maj 3rd นั้นหมายถึงว่ามันเป็นขั้นคู่เสียงอันดับที่ 3 ทางเมเจอร์ของโน้ตตัวแรกในบันไดเสียง คอลัมน์ถัดมาเป็นอัตราส่วน (ratio) ของความถี่ของโน้ตแต่ละตัวกับความถี่ของโน้ตตัวแรก สำหรับคอลัมน์ที่สามเป็นการแบ่งโน้ตทั้ง 12 ตัว ออกเป็นส่วนที่เล็กที่สุด 1200 ส่วนเท่าๆกัน เรียกว่าเซนต์ (cent) จะเห็นได้ว่าแต่ละตัวจะห่างกัน 100 เซนต์ และนี่คือเหตุผลที่เราเรียกมันว่า Equal temperament



รูปที่ 3.2 ความถี่ของระดับความถี่เสียงของเปียโนในระนาบ Equal Temperament

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

	EQUAL TEMPERAMENT		PYTHAGOREAN		JUST		MEAN-TONE	
	Ratio	Cents	Ratio	Cents	Ratio	Cents	Ratio	Cents
Unison	1:1	0	1:1	0	1:1	0	1:1	0
Aug unis					1:1.055	92	1:1.045	76
Mi 2nd	1:1.059	100	1:1.053	90	1:1.067	112	1:1.070	117
Maj 2nd	1:1.122	200	1:1.125	204	1:1.125	204	1:1.118	193
Mi 3rd	1:1.189	300	1:1.185	294	1:1.200	316	1:1.196	310
Maj 3rd	1:1.260	400	1:1.265	408	1:1.250	386	1:1.250	386
P 4th	1:1.335	500	1:1.333	498	1:1.333	498	1:1.337	503
Tritone	1:1.414	600						
Aug 4th			1:1.404	588	1:1.406	590	1:1.398	580
Dim 5th			1:1.424	612	1:1.422	610		
P 5th	1:1.498	700	1:1.500	702	1:1.500	702	1:1.496	697
Mi 6th	1:1.587	800	1:1.580	792	1:1.600	814	1:1.600	814
Maj 6th	1:1.682	900	1:1.687	906	1:1.667	884	1:1.672	890
Aug 6th					1:1.778	996	1:1.747	966
Mi 7th	1:1.782	1000	1:1.778	996	1:1.800	1018	1:1.789	1007
Maj 7th	1:1.888	1100	1:1.898	1109	1:1.875	1088	1:1.869	1083
Octave	1:2	1200	1:2	1200	1:2	1200	1:2	1200

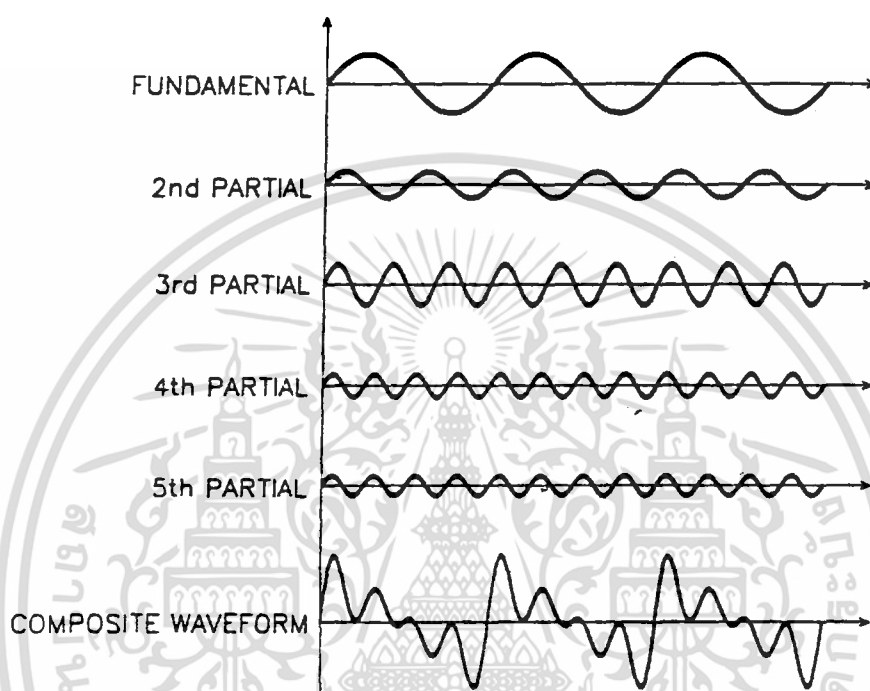
ตารางที่ 3.1 แสดงการเปรียบเทียบอัตราส่วนของความถี่ในระบบต่างๆ

### 3.2 การวิเคราะห์ทางฟูรีเยร์และองค์ประกอบทางฟูรีเยร์

ในความเป็นจริงแล้ว เสียงที่เราได้ยินอยู่ในธรรมชาตินั้น ส่วนใหญ่เป็นคลื่นเชิงซ้อน (complex wave) ซึ่งประกอบไปด้วยคลื่นหลายคลื่นรวมกันอยู่ คลื่นขายนี่มีค่าความถี่เป็นค่าเริ่มต้นของคลื่นขายนี่ทั้งหมด ที่มีความสัมพันธ์เชิงความถี่กันอยู่ เรียกว่า คลื่นมูลฐาน (Fundamental) ส่วนคลื่นขายนี่อื่นๆนั้นเรียกว่า ฮาร์โมนิก (Harmonic) ฮาร์โมนิกเหล่านี้จะมีค่าความถี่เป็นจำนวนเท่าทวีคูณกับความถี่มูลฐาน (Fundamental frequency) โดยใช้เลขจำนวนเต็มสำหรับบอกว่าฮาร์โมนิกนั้นมีค่าความถี่เป็นกี่เท่าของความถี่มูลฐานยกตัวอย่าง เช่น คลื่นมูลฐานมีความถี่ 100 เฮิร์ต ฮาร์โมนิกที่สองจะมีความถี่เป็น 200 เฮิร์ต ฮาร์โมนิกที่สามก็จะมีค่าความถี่เป็น 300 เฮิร์ต เป็นเช่นนี้เรื่อยไปตามลำดับ คลื่นเชิงซ้อนแต่ละคลื่นจะมีจำนวนองค์ประกอบทางฮาร์โมนิกไม่เท่ากัน อันดับขององค์ประกอบทางฮาร์โมนิกที่ประกอบขึ้นมาเป็นคลื่นเชิงซ้อนรูปกึ่งขนาดของแอมพลิจูด และเฟสของมันจะเป็นตัวกำหนดรูปร่าง (waveform) ของคลื่นเชิง-

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

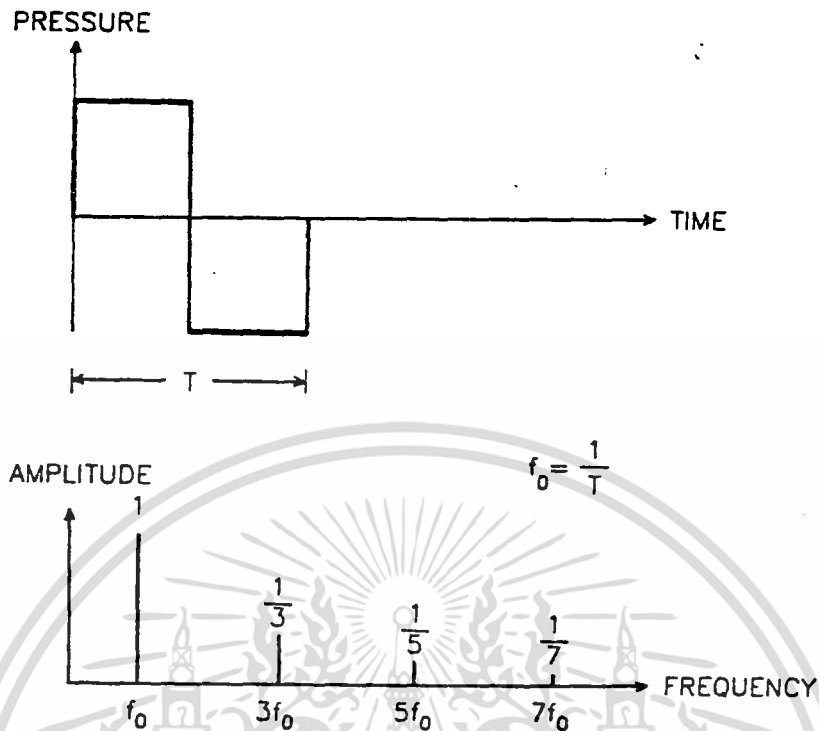
-ชั้นรูปที่ 3.3 จะแสดงตัวอย่างของความซับซ้อนดังกล่าว



รูปที่ 3.3 คลื่นเชิงซ้อน และองค์ประกอบทางฮาร์โมนิกทั้งห้าที่ประกอบกันเป็นมัน

เพื่อความเข้าใจ เราจะขอยกตัวอย่างเพื่อแสดงให้เห็นถึงความซับซ้อนดังกล่าวให้ชัดเจนยิ่งขึ้น จากรูปที่ 3.4 แสดงผลการวิเคราะห์ทางฟูรีเยร์ของคลื่นรูปสี่เหลี่ยมจัตุรัส (square wave) เมื่อ  $N$  เป็นอันดับที่ของฮาร์โมนิก ( $N=1$  หมายถึงฮาร์โมนิกที่ 1 หรือ คลื่นมูลฐาน) และ  $A$  เป็นแอมพลิจูดของคลื่นนั้น จะเห็นได้ว่า คลื่นสี่เหลี่ยมจัตุรัสจะประกอบขึ้นจากฮาร์โมนิกที่มีอันดับเป็นเลขคี่เท่านั้น และแอมพลิจูดของฮาร์โมนิกเหล่านั้นจะมีความเท่ากับเศษหนึ่งส่วน  $N$  เท่าของแอมพลิจูดของคลื่นมูลฐานนั้น ในรูปที่ 3.4 ล่างนั้นคือองค์ประกอบทางฮาร์โมนิก (Spectrum) หรือองค์ประกอบทางฟูรีเยร์ ของคลื่นสี่เหลี่ยมจัตุรัสนั่นเอง

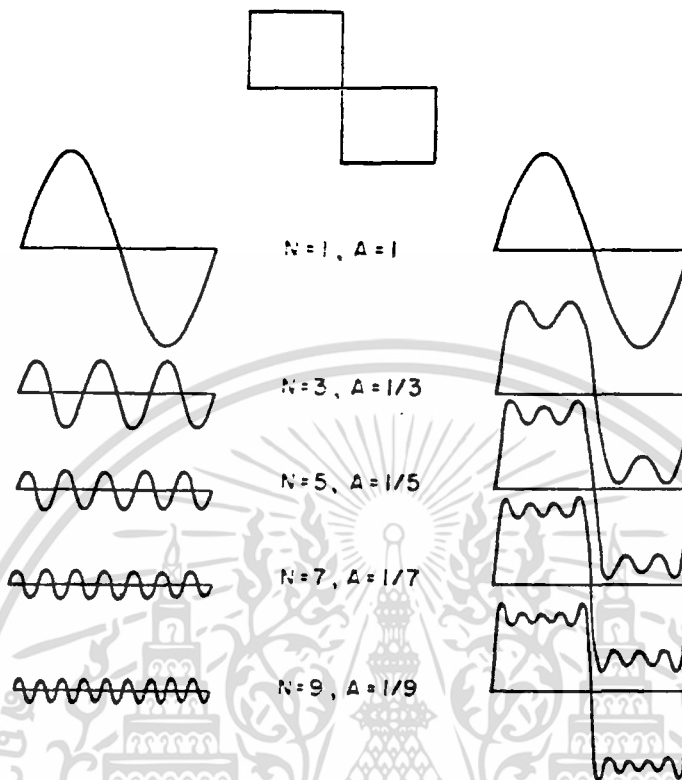
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.4 คลื่นสี่เหลี่ยมจัตุรัส และองค์ประกอบทางฮาร์โมนิกของมัน

ในทางตรงกันข้าม เราสามารถสังเคราะห์คลื่นสี่เหลี่ยมจัตุรัสขึ้นได้จากคลื่นไซน์ที่มีความถี่หนึ่ง โดยการเพิ่มองค์ประกอบทางฮาร์โมนิกที่มีอันดับเลขคี่ ซึ่งมีขนาดแอมพลิจูดและเฟสของแต่ละฮาร์โมนิกเหมาะสมตามเงื่อนไขข้างต้นเข้าไป ดังแสดงไว้ในรูปที่ 3.5 จะเห็นได้ว่า ยิ่งจำนวนขององค์ประกอบทางฮาร์โมนิกยิ่งเพิ่มขึ้น คลื่นเชิงซ้อนที่ได้ก็จะยิ่งใกล้เคียงกับคลื่นสี่เหลี่ยมจัตุรัสมากขึ้น

อันที่จริง ในเสียงจริงๆตามธรรมชาติ เราอาจจะไม่พบคลื่นจัตุรัสที่มีลักษณะสมบูรณ์ดังในรูป คลื่นรูปสี่เหลี่ยมที่ใช้นั้นเป็นส่วนหนึ่งในจำนวนคลื่นเชิงซ้อนมาตรฐาน (Standard complex wave) ที่ใช้อธิบายการวิเคราะห์และสังเคราะห์ของคลื่นเชิงซ้อน คลื่นเชิงซ้อนมาตรฐานเหล่านี้ประกอบด้วย คลื่นไซน์ (sine), คลื่นสามเหลี่ยม (Triangular), คลื่นสี่เหลี่ยมจัตุรัส (square), คลื่นฟันเลื่อย (Sawtooth หรือ Ramp) และพัลส์ต่อเนื่อง (Pulse train) ความสัมพันธ์ทางแอมพลิจูดของฮาร์โมนิกต่างๆของคลื่นเหล่านี้ได้แสดงไว้ในตารางที่ 3.2 แล้ว



รูปที่ 3.5 แสดงการสังเคราะห์ทางฟูรีเยร์ของคลื่นสี่เหลี่ยมจัตุรัส

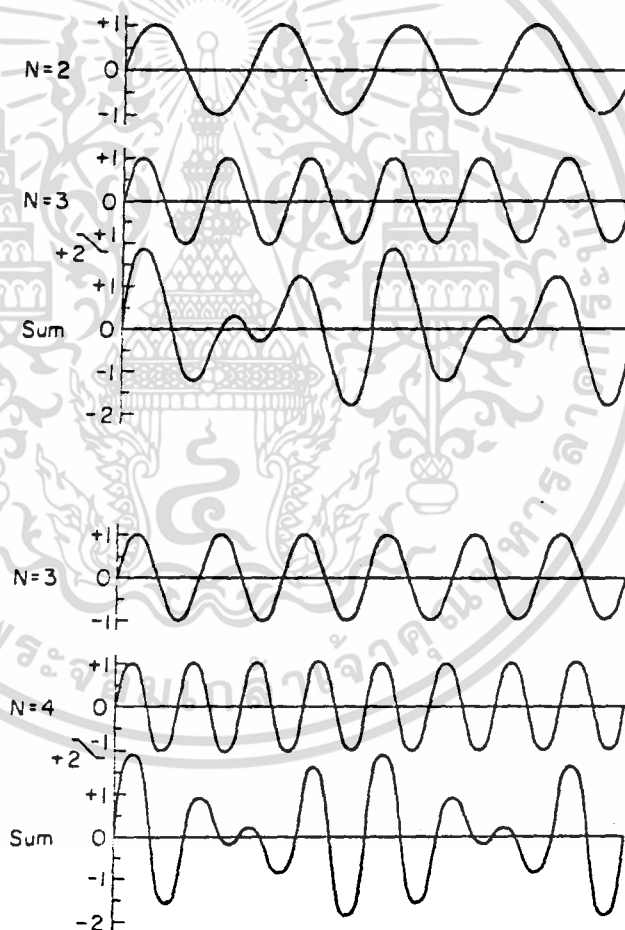
ตารางที่ 3.2 ความสัมพันธ์ทางแอมพลิจูดของฮาร์โมนิกของคลื่นมาตรฐาน

Wave	Harmonic amplitudes
	$N = 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, \dots$
Sine	1, 0, 0, 0, 0, 0, 0, 0, 0, 0, $\dots, N = 1$ only
Triangle	1, 0, $\frac{1}{6}$ , 0, $\frac{1}{25}$ , 0, $\frac{1}{49}$ , 0, $\frac{1}{81}$ , 0, $\dots, \frac{1}{N^2}$ for odd $N$
Square	1, 0, $\frac{1}{3}$ , 0, $\frac{1}{5}$ , 0, $\frac{1}{7}$ , 0, $\frac{1}{9}$ , 0, $\dots, \frac{1}{N}$ for odd $N$
Sawtooth (ramp)	1, $\frac{1}{2}$ , $\frac{1}{3}$ , $\frac{1}{4}$ , $\frac{1}{5}$ , $\frac{1}{6}$ , $\frac{1}{7}$ , $\frac{1}{8}$ , $\frac{1}{9}$ , $\frac{1}{10}$ , $\dots, \frac{1}{N}$ for all $N$
Pulse train	1, 1, 1, 1, 1, 1, 1, 1, 1, 1, $\dots$ , equal for all $N$

จากที่กล่าวมาแล้วข้างต้น เราพอสรุปได้ว่า คลื่นที่ซับซ้อนที่แน่นอนใดๆ นั้น สามารถสังเคราะห์ขึ้นได้จากความถี่ฐานค่าหนึ่ง และฮาร์โมนิกที่มีแอมพลิจูดและเฟสที่เหมาะสม โดยที่ความถี่ของคลื่นเรียงซ้อนที่เกิดขึ้นนั้นก็คือ ความถี่มูลฐานของมันนั่นเอง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ลองพิจารณารูปที่ 3.6 บน จะเห็นคลื่นเชิงซ้อนซึ่งเกิดจากการรวมกันของฮาร์โมนิกอันดับที่ 2 และ 3 ส่วนรูปร่าง จะเป็นคลื่นเชิงซ้อนที่เกิดจากการรวมกันของฮาร์โมนิกอันดับที่ 3 และ 4 ของความถี่มูลฐานเดียวกับรูปบน เราจะเห็นว่ารูปร่างของคลื่นเชิงซ้อนที่เกิดในทั้งสองกรณีจะมีลักษณะที่แตกต่างกัน แต่คาบของคลื่นเชิงซ้อนทั้งสองรูปนั้นจะเท่ากัน และเท่ากับคาบของคลื่นมูลฐานของมันทั้งสองด้วย นี่แสดงให้เห็นว่า คลื่นเชิงซ้อนใดๆ จะมีความถี่เท่ากับความถี่มูลฐานของมัน ถึงแม้ว่าจะไม่มีคลื่นมูลฐานนั้นในโครงสร้างทางฮาร์โมนิกของมันก็ตาม

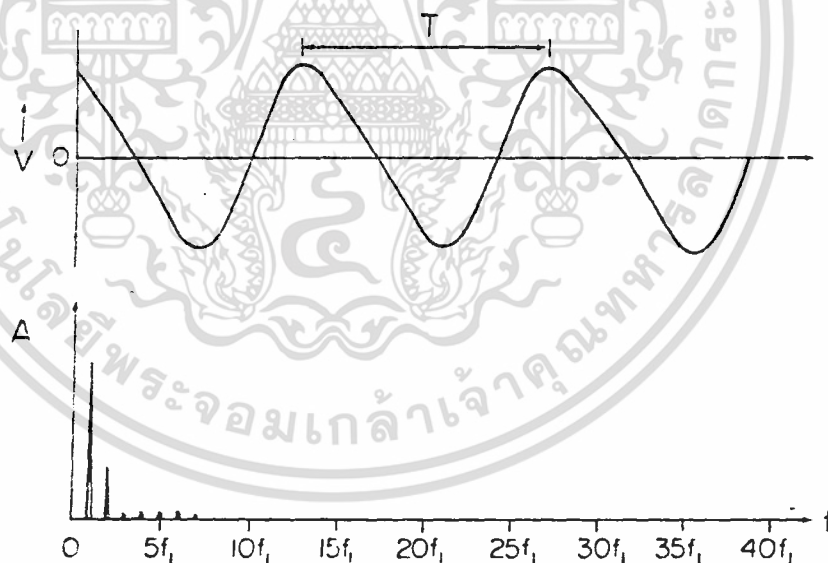


รูปที่ 3.6 การสังเคราะห์ทางฟูรีเยร์ของคลื่นเชิงซ้อน 2 รูป ซึ่งได้จากการรวมกันขององค์ประกอบทางฮาร์โมนิกที่มีความถี่มูลฐานเดียวกัน และแอมพลิจูดเท่ากัน แต่ต่างอันดับกัน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

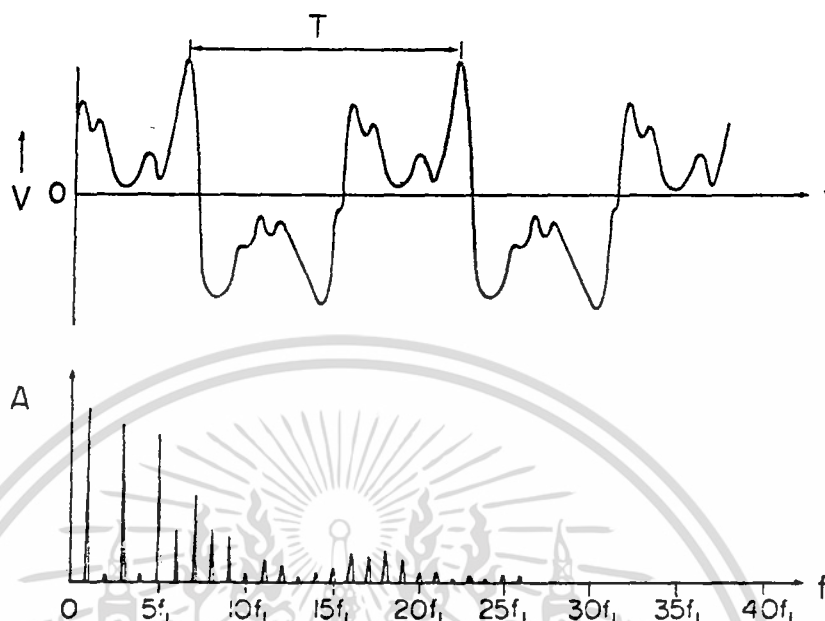
เราสามารถวิเคราะห์ทางองค์ประกอบทางฮาร์โมนิกของคลื่นเชิงซ้อนใด ๆ ได้ด้วยการวิเคราะห์ทางฟูเรียร์ ซึ่งในปัจจุบันสามารถกระทำได้โดยใช้การคำนวณเครื่องคอมพิวเตอร์ จากการศึกษาการทางองค์ประกอบทางฮาร์โมนิกของคลื่นเชิงซ้อนด้วยการวิเคราะห์ทางฟูเรียร์ เราสามารถสรุปได้ว่าระดับความถี่ของเสียงที่เกิดจากคลื่นที่มีคาบเวลาสั้นนั้นฮาร์โมนิก ๆ สามารถหาได้จากความถี่มูลฐานของตัวมันเอง ตัวอย่างขององค์ประกอบทางฮาร์โมนิกของเสียงจากเครื่องดนตรีที่ได้จากการวิเคราะห์ฟูเรียร์ ได้แสดงไว้ในรูป 3.7 และ รูป 3.8

จากข้อสรุปดังกล่าว ทำให้เราได้แนวความคิดในการวิเคราะห์หาระดับความถี่เสียงจากเสียงดนตรี โดยการหาความถี่มูลฐานของเสียงนั้น ซึ่งรายละเอียดจะขงกล่าวในหัวข้อถัดไป



รูปที่ 3.7 ราคี่มูลและองค์ประกอบทางฮาร์โมนิกของ โน้ต C ซึ่งมีความถี่ 523.25 เฮิร์ต ของเสียง alto recorder.

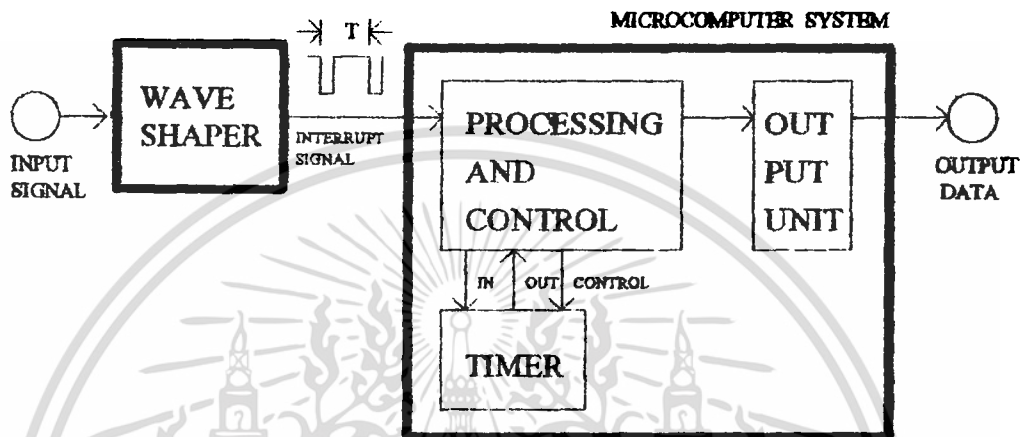
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.8 รูปคลื่นและองค์ประกอบทางฮาร์โมนิกของ โน้ต Bb ซึ่งมีความถี่ 493.88 เฮิรต ของเสียง violin.

### 3.3 การวัดคาบเวลาของสัญญาณด้วยไมโครคอมพิวเตอร์

การวัดคาบเวลาของสัญญาณนั้น สามารถทำให้เราทราบถึงความถี่ของสัญญาณได้ เพราะความถี่เป็นส่วนกลับของคาบเวลา ( $F = 1/T$ ) แต่เหตุผลสำคัญที่ทำให้เราใช้วิธีการหาคาบเวลาของสัญญาณก็เป็นเพราะว่า การใช้ไมโครคอมพิวเตอร์ในการหาคาบเวลาของสัญญาณทำได้ง่ายกว่าการหาความถี่ แม้มันจะแสดงหลักการทำงานของการใช้ไมโครคอมพิวเตอร์ ในการหาคาบเวลาได้แสดงไว้ในรูปที่ 3.9



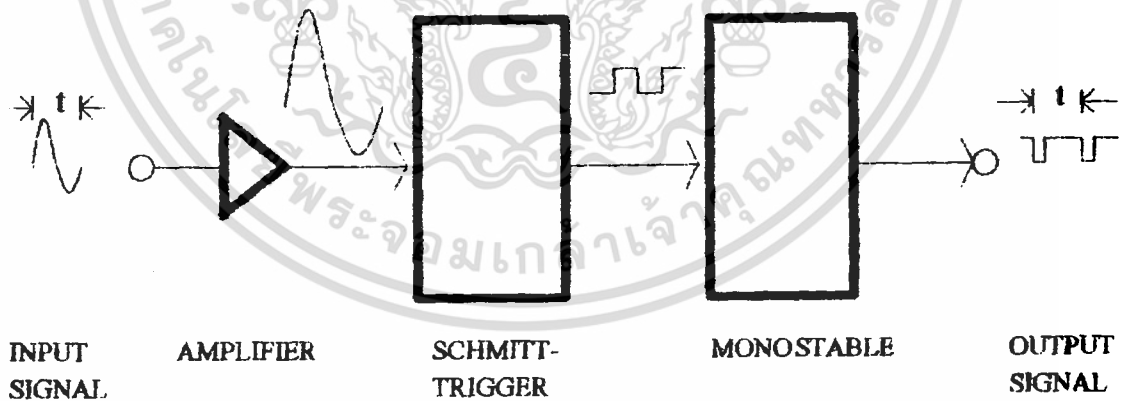
รูปที่ 3.9 แสดงแผนผังการใช้ไมโครคอมพิวเตอร์ในการหาคาบเวลา

เมื่อมีสัญญาณอินพุตที่เป็นสัญญาณเชิงมีคาบเวลา เข้ามาที่ส่วนจกรูปสัญญาณ (WAVE SHAPER) ส่วนนี้จะทำการเปลี่ยนรูปของสัญญาณอินพุตใดๆให้อยู่ในรูปของสัญญาณอินเทอร์รัท โดยที่ช่วงเวลาระหว่างสัญญาณอินเทอร์รัทแต่ละลูกนั้น จะมีค่าเท่ากับคาบเวลาของสัญญาณนั้น สัญญาณอินเทอร์รัทนี้ จะทำให้ส่วนควบคุมและประมวลผล ส่งสัญญาณไปควบคุมให้ส่วนจับเวลาเริ่มต้นและสิ้นสุดการจับเวลาตามจังหวะที่มีสัญญาณอินเทอร์รัทเข้ามา ข้อมูลที่ส่วนจับเวลาบันทึกไว้ นั่นคือ จุดเริ่มต้นและจุดสิ้นสุดของการจับเวลา ซึ่งข้อมูลทั้งสองนี้จะถูกส่งกลับไปประมวลผลโดยวิธีการทางคณิตศาสตร์ ที่ส่วนควบคุมและประมวลผล ข้อมูลที่ได้จะเป็นข้อมูลซึ่งแสดงคาบเวลาของสัญญาณ เราสามารถนำข้อมูลนี้ไปแสดงผลหรือนำไปใช้ในวัตถุประสงค์อื่นๆได้ สำหรับในการนี้ของเรานั้นจะนำข้อมูลนี้มาหาข้อมูลที่มีความถี่และความถี่เสียงที่สอดคล้องกับคาบเวลาของข้อมูลนั้น เพื่อใช้ในการควบคุมทางมิติอีกที

เท่าที่กล่าวมาแล้วนี้ เป็นเพียงหลักการคร่าวๆในการนำไมโครคอมพิวเตอร์ มาประยุกต์ใช้งานในการหาคาบเวลาของสัญญาณเท่านั้น ใ้การใช้งานจริง อาจมี รายละเอียดต่างๆ แตกต่างกันไป ซึ่งขึ้นกับระบบไมโครโปรเซสเซอร์ที่ใ้ใช้ว่ามีสมรรถภาพ เพียงใด

3.4 การจั้รรูปของสัญญาณ

อันเนื่องมาจากสัญญาณอินพุทของเราั้น จะอยู่ในรูปของคลื่นเชิงซ้อนต่าง ๆ ซึ่งมีรูปร่างที่ไม่แน่นอน แต่สัญญาณอินพุทที่ระบบไมโครคอมพิวเตอร์ต้องการสำหรับการ อินเทอร์รั้ทกัน จะต้องมึลักษณะเป็นสัญญาณทึ่ลัส ดังนั้นเราจึงจำเป็นต้องทำการจั้รรูปของ สัญญาณของเราใ้มีลักษณะที่เหมารวมกับการทำงานของแต่ละส่วน ม่เพียงแสดงการจั้ร รูปของสัญญาณได้มแสดงไว้ใ้รูป 3.10



รูปที่ 3.10 แสดงแผนผังการจั้รรูปสัญญาณสำหรับการอินเทอร์รั้ทไมโครคอมพิวเตอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานใ้การศึกษาเท่านั้น ไม่อนุญาตใ้ให้นำไปใ้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิใ้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใ้

เมื่อมีสัญญาอันตกเข้ามา สัญญาดังกล่าวจะถูกขยายด้วยวงจรมายสัญญาเพื่อส่งไปให้วงจรมัททริกเกอร์ สัญญาที่มีเหลี่ยมยังไม่เหมาะกับการใช้เป็นสัญญาอินเทอร์เน็ต เพราะช่วงเวลาของสถานะที่ใช้ในการแอคทีฟ สำหรับการอินเทอร์เน็ตอาจยาวนานเกินไป อาจมีผลทำให้เกิดการตรวจจับอินเทอร์เน็ตที่ช้าจนได้ เราจึงจำเป็นต้องทำการลดช่วงเวลาดังกล่าวลง โดยอาศัยหลักที่ว่า ความกว้างของสัญญาอินเทอร์เน็ตจะต้องไม่มากไปกว่าเวลาที่ใช้ในการตอบสนองโปรแกรมบริการอินเทอร์เน็ต และไม่ทำให้คาบเวลารวมของสัญญาเปลี่ยนแปลง เอาท์พุทของวงจรมัททริกเกอร์จึงต้องถูกส่งให้กับวงจรมอนิเตอร์ เพื่อทำให้เกิดเป็นหน่วยพัลส์ ซึ่งมีช่วงเวลาระหว่างพัลส์แต่ละลูก เท่ากับ คาบเวลาของสัญญานั้น หมายความว่า พัลส์ของสัญญาอินเทอร์เน็ตจะทำให้เกิดพัลส์ออกจากเอาท์พุทของวงจรมอนิเตอร์สองลูก โดยที่พัลส์ลูกแรกจะใช้ในการเริ่มทำการจับเวลา ส่วนพัลส์ลูกที่สองจะใช้ในการหยุดการจับเวลานั้นเอง

เมื่อเรารวบรวมทฤษฎีและแนวความคิดต่างๆ ในบทนี้เข้าด้วยกัน จะทำให้เราสามารถออกแบบวงจรมอนิเตอร์ส่วนต่างๆ ซึ่งตอบสนองแนวความคิดนั้น ๆ เพื่อสร้างเป็นระบบที่สามารถทำการแปลงสัญญาที่มีระดับความถี่เสียง ให้ได้เป็นข้อมูลมิตติ์ที่จะตอบสนอง ระดับความถี่เสียงดังกล่าว เพื่อใช้ในการควบคุมเครื่องดนตรีไฟฟ้าในระบบมิตติ์ให้เล่นตามเสียงเหล่านั้น อันจะก่อให้เกิดประโยชน์ต่าง ๆ ดังที่เพยกล่าวมาแล้วในบทนี้

## บทที่ 4

### ระบบควบคุมและประมวลผล

สำหรับในโครงการพิเศษนี้ เรามีความจำเป็นจะต้องใช้ไมโครโปรเซสเซอร์ในการควบคุมและประมวลผล ทั้งนี้เนื่องจากเป็นงานที่ต้องการความรวดเร็วในการตอบสนองค่อนข้างสูง อีกทั้งยังมีงานที่จะต้องทำพร้อมๆกันหลายงานในเวลาเดียว การนำระบบไมโครคอมพิวเตอร์มาใช้มันจะช่วยลดความยุ่งยากซับซ้อนในการออกแบบระบบฮาร์ดแวร์ เพราะเราสามารถออกแบบระบบซอฟต์แวร์ให้ทำงานต่างๆเหล่านั้นมันได้ ทำให้เราสามารถลดขนาดของวงจรของเราลงได้อย่างมาก และเพื่อความสะดวกเราได้เลือกใช้แผงวงจรไมโครคอมพิวเตอร์สำเร็จรูปหรือที่เรียกกันว่า CONTROL PACK เหตุผลที่เลือกใช้ CONTROL PACK ก็เพราะมันเป็นแผงวงจรไมโครคอมพิวเตอร์ที่ออกแบบมาเพื่อใช้กับงานควบคุมโดยเฉพาะ ส่วนประกอบโดยทั่วไปของมันจะมีเพียงส่วนประมวลผลกลางหรือ CPU หน่วยความจำถาวรหรือ ROM ซึ่งทำหน้าที่ในการเก็บโปรแกรมควบคุมระบบ หน่วยความจำชั่วคราวหรือ RAM สำหรับเก็บค่าพารามิเตอร์ต่างๆที่ต้องการให้แก้ไขเปลี่ยนแปลงได้ และส่วนของ PORT สำหรับเชื่อมต่อกับอุปกรณ์อินพุตและเอาต์พุตภายนอก จะเห็นได้ว่าเราเพียงแต่ออกแบบส่วนอุปกรณ์อินพุต เอาท์พุตที่ใช้สำหรับงานของเราเพื่อเชื่อมต่อเข้ากับมัน ประกอบกับการเขียนโปรแกรมควบคุมการทำงานอีกเล็กน้อย เราก็จะได้ระบบไมโครคอมพิวเตอร์ที่สามารถทำงานเฉพาะด้านให้กับเราตามต้องการ

CP-180 คือ CONTROL PACK ของบริษัท อีซีที จำกัด เป็นแผงวงจรไมโครคอมพิวเตอร์ที่เราได้นำมาใช้ในโครงการนี้ คุณสมบัติต่างๆของ CP-180 พอจะสรุปได้ดังนี้

- ใช้ MPU เบอร์ Z80180 ทำงานที่ความถี่ X'TAL 12.288 MHz
- หน่วยความจำถาวรเป็น ROM 32K และ RAM 8K
- มี SERIAL PORT ที่สามารถโปรแกรมได้ 2 ช่อง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดลอกเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- มี 8255 2 ตัวเป็น I/O PORT ขนาด 48 บิต
- มี PRINTER PORT 1 PORT
- มี BUS สำหรับ Z80 และ EXTENDED BUS สำหรับ Z80180
- มี REGULATOR 5V เบอร์ 7805 เบอร์

ที่กล่าวมาแล้วข้างต้นเป็นเพียงคุณลักษณะภายนอกของ CP-180 เท่านั้น แต่เหตุผลสำคัญที่เราเลือกใช้งานเบอร์นี้เนื่องจากคุณสมบัติพิเศษของ MPU เบอร์ Z80180 ที่เอื้ออำนวยเป็นอย่างยิ่งกับการทำงานที่ต้องอาศัยการสื่อสารข้อมูลแบบอนุกรม และการอินเทอร์เฟซจำนวนมาก ดังจะอธิบายรายละเอียดไว้ในหัวข้อต่อไป

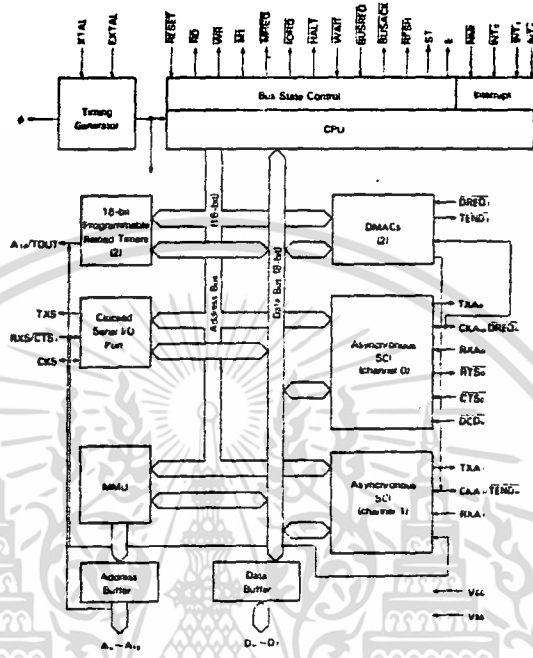
#### 4.1 Z80180 MPU (Multi Processing Unit)

Z80180 เป็น MPU ขนาด 8 BIT จากบริษัท ZILOG ที่ถูกออกแบบมาเพื่อลดต้นทุนการสร้างระบบและสามารถทำงานโดยกินกำลังไฟต่ำ ในขณะที่ให้ประสิทธิภาพการทำงานสูง และที่สำคัญคือสามารถให้ซอฟต์แวร์ร่วมกับ Z80 ซึ่งเป็น CPU ที่ได้รับความนิยมอย่างสูงได้ หรืออาจกล่าวได้ว่า Z80180 เป็นการพัฒนา Z80 ให้มีประสิทธิภาพสูงขึ้น เพราะนอกจากอุปกรณ์ อินพุต/เอาต์พุต ที่เพิ่มขึ้นบนชิปไอซีแล้ว สถาปัตยกรรมหลักยังคงเหมือนของ Z80 ทุกประการ

ส่วนที่ทำให้ประสิทธิภาพการทำงานสูงขึ้นได้แก่ สามารถทำงานที่ความถี่ของสัญญาณนาฬิกาที่สูงขึ้น ลดเวลาในการปฏิบัติการคำสั่งให้สั้นลง เพิ่มชุดคำสั่งและเพิ่มหน่วยจัดการระบบหน่วยความจำหรือ MMU (Memory Management Unit) ลงบนชิป ทำให้สามารถอ้างหน่วยความจำได้ถึง 1 เมกะไบต์

คุณสมบัติสำคัญในการที่ทำให้ Z80180 สามารถลดต้นทุนในการสร้างระบบนั้นอยู่ที่ การเพิ่มอุปกรณ์อินพุต/เอาต์พุต ที่สำคัญๆ ลงไว้บนชิปเดียวกัน อันได้แก่ DMA , UART และ TIMER ซึ่งสามารถทำได้อย่างอิสระ นอกจากนี้ยังมีระบบควบคุมการรีเฟรชหน่วยความจำแบบไดนามิก ตัวกำเนิดสัญญาณ wait state วงจรกำเนิดสัญญาณนาฬิกา และที่สำคัญก็คือ

ตัวควบคุมการอินเทอร์เฟซที่มีประสิทธิภาพ โครงสร้างภายในต่างๆของ Z80180 ได้แสดงไว้ดังรูปที่ 4.1



รูปที่ 4.1 สถาปัตยกรรมภายในของ Z80180

เนื่องจากพัฒนาขึ้นจากเทคโนโลยีทางด้าน CMOS ทำให้ Z80180 ทำงานได้โดยใช้กำลังไฟฟ้าในขณะทำงานปกติ แต่ยังสามารถโปรแกรมให้ทำงานในโหมดที่ลดการใช้กระแสลงได้อีกถึง 2 โหมด นั่นคือ โหมด SLEEP และโหมด SYSTEM STOP ซึ่งจะทำให้อุปกรณ์ภายในบางส่วนเข้าสู่สภาวะหยุดทำงาน เป็นผลทำให้ไอซีใช้กระแสลดลงจากทั้งหมดที่กล่าวมาลงข้างต้น เราสามารถสรุปลักษณะพิเศษของ MPU Z80180 ได้ดังนี้

ลักษณะพิเศษของ MPU Z80180

- ทำงานที่ความถี่ได้สูงถึง 10 MHz
- มี MMU ในตัว เพื่อสนับสนุนการขยายเนื้อที่หน่วยความจำ
- มี DMA 2 ช่อง
- มีตัวกำเนิดสัญญาณ wait state ในตัว
- มี UART 2 ช่อง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- มี TIMER ขนาด 16 บิต 2 ช่อง
- มีตัวควบคุมการอินเทอร์พรัทในตัว
- มีวงจรกำเนิดสัญญาณนาฬิกาในตัว
- ใช้รหัสคำสั่งร่วมกับ CPU เบอร์ Z80 ได้
- มีรหัคำสั่งเพิ่มเติม

จากที่กล่าวมาแล้วข้างต้น คงทำให้เราพอจะทราบถึงขีดความสามารถของ MPU ตัวนี้ได้เป็นอย่างดี สำหรับในหัวข้อต่อไปเราจะกล่าวถึง รายละเอียดต่างๆของ Z80180 เฉพาะส่วนที่เกี่ยวข้องกับโครงงานพิเศษนี้เท่านั้น สำหรับส่วนอื่นที่มิได้กล่าวไว้ในบทนี้ จะรวมอยู่ในภาคผนวกท้ายเล่ม

#### 4.1.1 รายละเอียดและการจัดขาของ Z80180

Z80180 มีลักษณะเป็นตัวถัง 68 ขา แบบ PLCC แต่ละขามีหน้าที่แตกต่างกันไป ซึ่งมีรายละเอียดดังต่อไปนี้

AO-A19	ADDRESS BUS ระหว่าง RESET จะเป็น HIGH IMPEDANCE
BUSAK	BUS ACKNOWLEDGE เป็นขา OUTPUT ACTIVE LOW ทำงานก็ต่อเมื่อ Z80180 ตอบสนองต่อการขอใช้ BUS ของ BUSRQ และจะทำให้ BUS ข้อมูล BUS ADDRESS และสัญญาณ CONTROL บางเส้นเป็น HIGH IMPEDANCE
BUSRQ	BUS REQUEST เป็นขา INPUT ACTIVE LOW ซึ่งจะมีความสำคัญกว่า NMI โดยจะมีการตรวจสอบสัญญาณนี้ทุกๆการสิ้นสุดของ MACHINE CYCLE
CKA0, CKA1	ASYNCHRONOUS CLOCK 0 และ 1 เป็นขาสัญญาณ CLOCK แบบ 2 ทิศทาง คือจะใช้เป็นขา INPUT หรือ OUTPUT ก็ได้
CKS	SERIAL CLOCK เป็นขา CLOCK 2 ทิศทางของ CSI/O

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

CLOCK	เป็นขา OUTPUT โดยจะเป็นครึ่งหนึ่งของ X'TAL. หรือ CLOCK OUT เช่น X'TAL 12 MHz Z80180 จะ RUN ที่ 6 MHz
CTS0-CTS1	CLEAR TO SEND 0 และ 1 เป็นขา INPUT ACTIVE LOW ใช้ในการควบคุม MODEM
DO-D7	DATA BUS เป็นแบบ 2 ทิศทาง
DCDO	DATA CARRIER DETECT 0 เป็นขา INPUT ACTIVE LOW ใช้ควบคุมในการติดต่อกับ MODEM ของ ASCII CHANNEL 0
DRBQ0-DRBQ1	DMA REQUEST 0 และ 1 เป็นขา INPUT ACTIVE LOW ใช้ในการขอ DMA และขานี้จะ โปรแกรมได้ว่าให้ตรวจสอบสถานะที่ขอบหรือระดับได้
E	ENABLE CLOCK เป็นขา OUTPUT ACTIVE HIGH ซึ่งใช้บังคับการทำงานกับอุปกรณ์ภายนอก ระหว่างการทำงานเกี่ยวกับ BUS และใช้เชื่อมต่อกับอุปกรณ์ในตระกูล 68XX 80XX
HALT	เป็นขา OUTPUT ACTIVE LOW จะทำงานเมื่อทำคำสั่ง HALT หรือ SLP
INT0	MASKABLE INTERRUPT 0 เป็น INPUT ACTIVE LOW สถานะที่ขานี้จะถูกตรวจทุกการสิ้นสุดของคำสั่ง
INT1, INT2	เช่นเดียวกับ INT0 แต่มีระดับความสำคัญรองลงมาตามลำดับ
IORQ	เป็นขา OUTPUT เพื่อบอกว่ากำลังติดต่อกับ I/O
M1	MACHINE CYCLE 1 เป็นขา OUTPUT ACTIVE LOW จะทำงานเมื่อ FETCH OP-CODE
NMI	NON MASKABLE INTERRUPT เป็นขา INPUT ACTIVE LOW ขานี้จะคอยรับการอินเทอร์รัพท์ INTERRUPT เสมอ โดยไม่สามารถหยุดด้วย SOFTWARE
RD	เป็นขาที่ใช้ทำการอ่านข้อมูลจาก MEMORY หรือ I/O

- RFSH** เป็นขาที่ให ADDRESS LOW (A0-A7) ไป REFRESH DYNAMIC RAM
- RTSO** REQUEST TO SEND เป็นขา OUTPUT ACTIVE LOW ขานี้ใช้โปรแกรมสัญญาณควบคุมโมเด็มของ ASCI CHANNEL 0
- RXA0,RXA1** RECIEVE DATA 0 และ 1 เป็นขารับสัญญาณจาก SERIAL PORT ของ ASCI
- RXS** CLOCK SERIAL RECIEVE DATA เป็นขารับสัญญาณ SERIAL ของ CSIO
- ST** STATUS เป็นขา OUTPUT ACTIVE HIGH ใช้แสดงสถานะการทำงานของ CPU โดยร่วมกับ M1 และ HALT ดังตารางที่ 4.1

ST	HALT	M1	Operation
0	1	0	CPU operation (1st op-code fetch)
1	1	0	CPU operation (2nd op-code and 3rd op-code fetch)
1	1	0	CPU operation (MC except for op-code fetch)
0	X	1	DMA operation
0	0	0	HALT mode
1	0	1	SLEEP mode (and SYSTEM STOP mode)

ตารางที่ 4.1 แสดงสถานะการทำงานของ CPU

- TEND0-TEND1** TRANSFER END 0 และ 1 เป็นขา OUTPUT ACTIVE LOW ใช้แสดงถึงว่าทำ DMA สิ้นสุดลงแล้ว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

TOUT	TIMER OUT ใช้กำเนิดพัลส์จาก PRT CHANNEL 1
TXA0, TXA1	TRANSMIT DATA 0 และ 1 เป็นขาส่งข้อมูล SERIAL ของ ASCII
TXS	CLOCK SERIAL TRANSMIT DATA เป็นขาส่งข้อมูล SERIAL ของ CSIO
WAIT	ขา INPUT ACTIVE LOW จะถูกตรวจที่ขอบขาของ CLOCK ลุคที่ 2 ของทุกๆ MACHINE CYCLE เพื่อเป็นการรอให้อุปกรณ์ภายนอกทำงานให้ทันกับ CPU
WR	ใช้สำหรับการส่งข้อมูลไปยัง I/O หรือ MEMORY
X'TAL	เป็นขาที่ใช้ต่อกับ X'TAL

#### 4.1.2 ขาที่ MULTIPLEX

A18/TOUT	ระหว่าง RESET จะเป็น A18 แต่ถ้ามีการเลือก SET BIT TOC1 หรือ TOC0 ใน TIMER CONTROL REGISTER (TCR) ก็จะทำหน้าที่เป็น TOUT
CKA0/DREQ	ระหว่าง RESET จะเป็น CKA0 แต่ถ้า DM1 หรือ SM1 ใน DMA MODE REGISTER (DMODE) ถูก SET เป็น 1 จะเป็น ขา DREQ
CKA1/TEND0	ระหว่าง RESET จะเป็น CKA1 แต่ถ้า BIT CKA1D ใน ASCII ถูก SET จะเป็นขา TEND0
RXS/CTS1	ระหว่าง RESET ขานี้จะเป็น RXS แต่ถ้า BIT CTS1E ใน ASCII ถูก SET จะเป็นขา CTS1

### 4.1.3 INTERNAL I/O REGISTER ภายใน Z80180 ภายในโครงการ

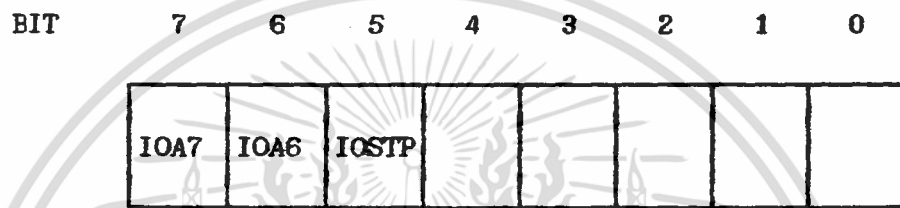
#### มีด้วยกัน 64 I/O ADDRESS ดังตารางที่ 4.2

	Register	Mnemonic	Address	
			Binary	Hexadecimal
ASCII	ASCII Control Register A Ch 0	CNTLA0	XX000000	00H
	ASCII Control Register A Ch 1	CNTLA1	XX000001	01H
	ASCII Control Register B Ch 0	CNTLB0	XX000010	02H
	ASCII Control Register B Ch 1	CNTLB1	XX000011	03H
	ASCII Status Register Ch 0	STAT0	XX000100	04H
	ASCII Status Register Ch 1	STAT1	XX000101	05H
	ASCII Transmit Data Register Ch 0	TDRO	XX000110	06H
	ASCII Transmit Data Register Ch 1	TDRI	XX000111	07H
	ASCII Receive Data Register Ch 0	RDRO	XX001000	08H
ASCII Receive Data Register Ch 1	RDRI	XX001001	09H	
CSI/O	CSI/O Control Register	CNTR	XX001010	0AH
	CSI/O Transmit/Receive Data Register	TRDR	XX001011	0BH
Timer	Timer Data Register Ch 0L	TMDR0L	XX001100	0CH
	Timer Data Register Ch 0H	TMDR0H	XX001101	0DH
	Reload Register Ch 0L	RLDR0L	XX001110	0EH
	Reload Register Ch 0H	RLDR0H	XX001111	0FH
	Timer Control Register	TCR	XX010000	10H
	Reserved		XX010001	11H
			}	}
			XX010011	13H
	Timer Data Register Ch 1L	TMDR1L	XX010100	14H
	Timer Data Register Ch 1H	TMDR1H	XX010101	15H
Reload Register Ch 1L	RLDR1L	XX010110	16H	
Reload Register Ch 1H	RLDR1H	XX010111	17H	
Others	Free Running Counter	FRG	XX011000	18H
	Reserved		XX011001	19H
			}	}
		XX011111	1FH	
DMA	DMA Source Address Register Ch 0L	SAR0L	XX100000	20H
	DMA Source Address Register Ch 0H	SAR0H	XX100001	21H
	DMA Source Address Register Ch 0B	SAR0B	XX100010	22H
	DMA Destination Address Register Ch 0L	DAR0L	XX100011	23H
	DMA Destination Address Register Ch 0H	DAR0H	XX100100	24H
	DMA Destination Address Register Ch 0B	DAR0B	XX100101	25H
	DMA Byte Count Register Ch 0L	BCR0L	XX100110	26H
	DMA Byte Count Register Ch 0H	BCR0H	XX100111	27H
	DMA Memory Address Register Ch 1L	MAR1L	XX101000	28H
	DMA Memory Address Register Ch 1H	MAR1H	XX101001	29H
	DMA Memory Address Register Ch 1B	MAR1B	XX101010	2AH
	DMA I/O Address Register Ch 1L	MAR1L	XX101011	2BH
	DMA I/O Address Register Ch 1H	MAR1H	XX101100	2CH
	Reserved		XX101101	2DH
	DMA Byte Count Register Ch 1L	BCR1L	XX101110	2EH
	DMA Byte Count Register Ch 1H	BCR1H	XX101111	2FH
DMA Status Register	CSSTAT	XX110000	30H	
DMA Mode Register	DMODE	XX110001	31H	
DMA/WAIT Control Register	DCNTL	XX110010	32H	
INT	IL Register (Interrupt Vector Low Register)	IL	XX110011	33H
	INT/TRAP Control Register	ITC	XX110100	34H
	Reserved		XX110101	35H
Refresh	Refresh Control Register	RCP	XX110110	36H
	Reserved		XX110111	37H
MMU	MMU Common Base Register	CSB	XX111000	38H
	MMU Bank Base Register	NSB	XX111001	39H
	MMU Common/Bank Area Register	CSBAR	XX111010	3AH
I/O	Reserved		XX111011	3BH
			}	}
			XX111101	3DH
	Operation Mode Control Register	OMCR	XX111110	3EH
	I/O Control Register	ICR	XX111111	3FH

ตารางที่ 4.2 INTERNAL I/O ADDRESS MAP

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จาก MAP I/O ภายในจะเห็นว่าโปรแกรม Z80 เท่าที่เรามียู่อาจจะมีการส่ง PORT ซ้ำ กับ I/O ภายในทำให้โปรแกรมเดิมทำงานไม่ได้ สามารถแก้ไขได้โดยการโปรแกรมย้าย MAP I/O ภายในโดยการ CONTROL BIT ใน REGISTER I/O ICR ADDRESS 3FH ซึ่งสามารถย้ายไปที่ใดก็ได้ภายใน 256 ตำแหน่ง ดังรูปที่ 4.2



รูปที่ 4.2 I/O CONTROL REGISTER

และการโปรแกรมเป็นดังตารางที่ 4.3

IOA7	IOA6	ช่วง ADDRESS I/O
0	0	0000 - 003FH
0	1	0040 - 007FH
1	0	0080 - 00BFH
1	1	00C0 - 00FFH

ตารางที่ 4.3 การกำหนดช่วงของ INTERNAL I/O ADDRESS

เช่น ต้องการย้าย I/O ภายในไป I/O ADDRESS 80H เป็นต้นไป จะโปรแกรมได้เป็น

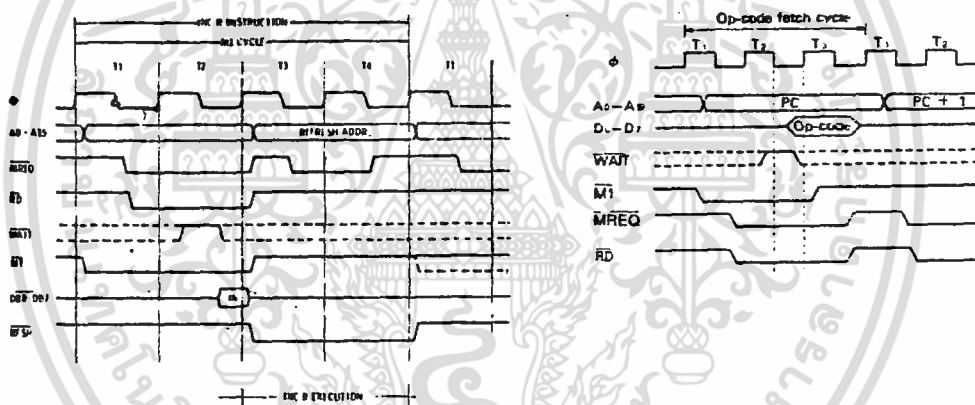
```
LD A,80H
OUTD (3FH),A
```

ส่วน I/O STOP MODE BIT 5 เป็น 1 จะทำให้ I/O ภายในหยุดทำงานเมื่อ RESET BIT นี้จะเป็น 0

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

#### 4.1.4 เกี่ยวกับ TIMING

ให้ดูรายละเอียดที่ภาคผนวกดังกล่าวสรุปได้ว่า Z80180 ใช้เวลาในการทำคำสั่งใน 1 MACHINE CYCLE น้อยกว่า Z80 อยู่ 1 T STATE คือ ใช้เวลาใน 1 MACHINE CYCLE เพียง 3 T STATE ในขณะที่ Z80 ใช้ 4 T STATE จะเห็นได้ว่า ในขณะที่ Z80180 RUN ความถี่เดียวกันกับ Z80 CPU Z80180 ก็ยังให้ความเร็วกว่า Z80 ถึงอีก 25% แต่ในขณะเดียวกัน Z80180 ยังสามารถต่อ CLOCK สูงกว่า Z80 ได้มากกว่า 1 เท่า จึงทำให้ความเร็วในการทำงานของ Z80180 ดีอย่างมาก ดูรูปที่ 4.3 เปรียบเทียบ T STATE ของ Z80 กับ Z80180



รูปที่ 4.3 เปรียบเทียบ T STATE ของ Z80 กับ Z80180

#### 4.1.5 HALT AND LOW POWER MODE

มีด้วยกัน 4 MODE คือ

HALT MODE

โดยการตั้งค่าสั่ง 76H จะทำให้ CPU หยุดทำงานคำสั่ง แต่การทำงานต่างๆของ CPU ยังดำเนินไป การออกจากคำสั่ง HALT โดย RESET หรือ INTERRUPT

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

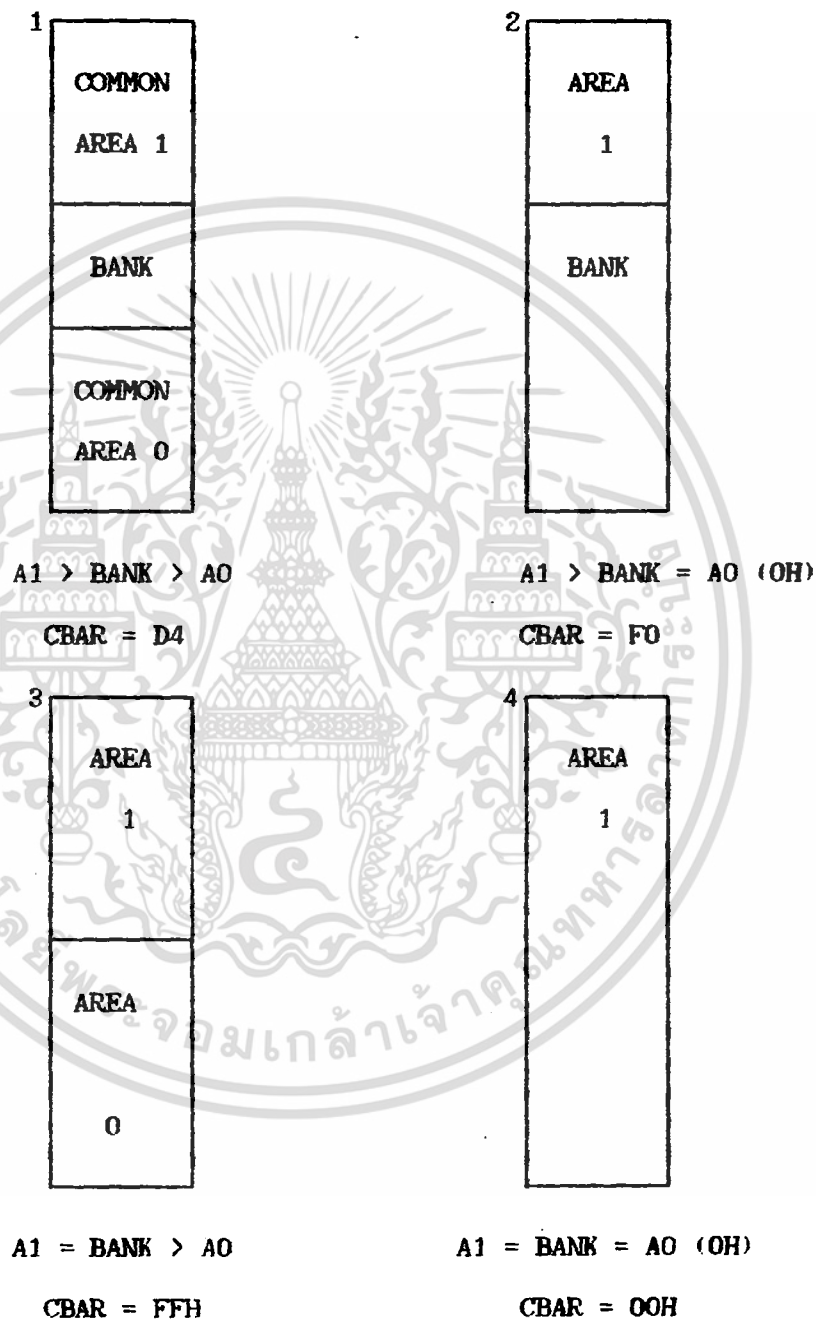
SLEEP MODE	โดยการนำคำสั่ง SLP ซึ่ง CPU จะหยุด CLOCK ภายในทำให้ ADDRESS เป็น HIGH , DATA BUS เป็น TRISTATE , DRAM REFRESH , INTERNAL DMAC หยุดทำงานออกจาก SLEEP MODE โดยการกด RESET หรือ INTERRUPT
IOSTOP MODE	ใช้หยุดการทำงานของ CHIP ภายในคือ ASCII, CSI/I/O และ PRT โดยการ SET BIT ใน I/O CONTROL REGISTER (ICR I/O ADDRESS 3FH) เป็น 1 และจะให้ทำงานต่อก็กด RESET หรือ โปรแกรมให้ BIT ใน ICR เป็น 0
SYSTEM STOP MODE	เป็นการรวมกันของ IOSTOP กับ SLEEP MODE โดยการ SET BIT ใน ICR แล้วตามด้วยคำสั่ง SLP จะทำให้ I/O ภายในหยุดทำงาน และ CPU หยุดทำงานเพื่อเป็นการประหยัดพลังงาน ซึ่งใน MODE นี้ CPU จะกินกระแสเพียง 7.5 mA ในขณะที่ปกติจะกินกระแสประมาณ 35 mA เมื่อออกจาก SYSTEM STOP MODE ก็โดยการ RESET หรือ INTERRUPT จากภายนอก

#### 4.2 MEMORY MANAGEMENT UNIT (MMU)

ใช้เป็นตัวขยาย MEMORY จาก 64 K (LOGICAL) เป็น 1 MBYTE (PHYSICAL) โดยการแบ่ง 64 K BYTE LOGICAL (คือ ADDRESS ปกติที่ใช้เช่นเดียวกับ Z80) เป็น 3 ส่วน ในการใช้งานด้วยกัน คือ COMMON AREA 0, BANK AREA และ COMMON AREA 1 โดยการกำหนดโปรแกรมจัด MAP LOGICAL ใน REGISTOR I/O CBAR (ADDRESS 3AH) ซึ่งใน REGISTOR นี้จะถูกแบ่งเป็น 2 บิตเป็นคือ 4 BIT สูง และ 4 BIT ต่ำ โดย 4 BIT สูงใช้โปรแกรมทั้งส่วนของ COMMON AREA 1 และ 4 BIT ต่ำใช้โปรแกรมส่วน BANK AREA ดังนั้นการโปรแกรม REGISTOR CBAR นี้ก็จะจัด MAP

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดลอกเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ได้เป็น  $2^2$  คือ 4 รูปแบบดังรูปที่ 4.4



รูปที่ 4.4 การโปรแกรม รีจิสเตอร์ CBAR

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากค่าที่โปรแกรมใน CBAR จึงทำให้โปรแกรม COMMON AREA ทั้ง 2 และ BANK ได้ตั้งแต่ 4 K BYTE ขึ้นไป เช่น ไทน์เบิ้ลสูงของ CBAR = 0FH ก็คือ AREA 1 มีค่าตั้งแต่ F000-FFFFH (คือ 4 K อย่างต่ำนั่นเอง) และจุดที่นำฮาร์ดแวร์จากการจัด MAP ทั้ง 4 รูปแบบนั้นก็คือ COMMON 0 และ BANK สามารถมีตำแหน่งที่ซ้อนกันได้ (ตำแหน่งเดียวกัน) และ COMMON AREA 1 กับ BANK ก็สามารถโปรแกรมให้อยู่ที่ใดก็ได้โดยอิสระตั้งแต่ 4 K BYTE ขึ้นไปของส่วน PHYSICAL ADDRESS (1 MBYTE โดยใช้ร่วมกับ REGISTER อีก 2 ตัว) แต่ส่วน COMMON AREA 0 แล้วจะเป็น BASED หรือ MONITOR ของระบบนั่นเอง

จากที่กล่าวมาเรายังไม่พูดถึงการขยาย MEMORY ออกไปมากกว่า 64 K เพราะว่าการที่จะขยาย MEMORY เกินกว่า 64 K นั้น จะต้องอิงส่วนของ LOGICAL ด้วย เนื่องด้วยคำสั่งของ Z80 ไม่สามารถอ้าง MEMORY เกินนี้ได้ ดังนั้นการอ้างถึง MEMORY ทั้งหมดจึงยังเป็นส่วนของ LOGICAL แต่ข้อมูลที่ถูกระทำจริงจะเป็นส่วนของ PHYSICAL เช่น ในคำสั่งอาจเป็นดังนี้

LD A, (8000H)

ซึ่งดูจากคำสั่งนี่จะเป็นการทำกับตำแหน่ง 8000H (สมมติในส่วนของ BANK AREA) แต่เรา SET PHYSICAL AREA ไว้ที่ 10000H นั่นก็หมายความว่าการทำงานคำสั่งข้างบนนี้ข้อมูลจะถูกกระทำที่ ADDRESS 10000H นั่นเอง

#### 4.2.1 REGISTER CONTROL

CBAR : COMMON/BANK AREA REGISTER ใช้กำหนดที่ตั้งของ LOGICAL  
ที่เป็น COMMON AREA 0 , BANK AREA และ COMMON AREA 1

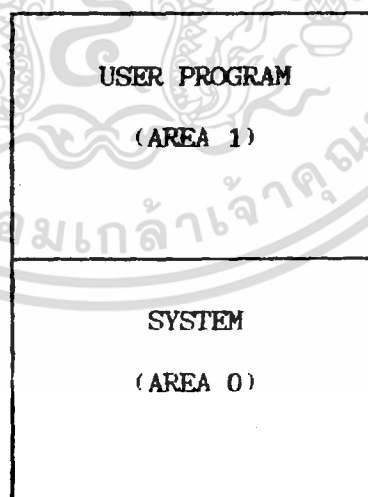
BIT            7        6        5        4        3        2        1        0

CA 3	CA 2	CA 1	CA 0	BA 3	BA 2	BA 1	BA 0
------	------	------	------	------	------	------	------

รูปที่ 4.5 MMU COMMON/BANK AREA REGISTER

- CA 3 - CA 0    เป็นตัวกำหนด ADDRESS เริ่มต้นของ COMMON AREA 1
- BA 3 - BA 0    เป็นตัวกำหนดจุด ADDRESS สุดท้ายของพื้นที่ BANK AREA ที่ต่อจากจุดเริ่มต้นของ COMMON AREA 1
- CBR :            COMMON BASE REGISTER เป็น REGISTER I/O 8 BIT เพื่อใช้กำหนด PHYSICAL COMMON AREA 1
- BBR :            BANK BASE REGISTER ใช้กำหนด PHYSICAL BANK AREA

สำหรับการใช้งานจริงเราได้แบ่ง MAP เป็นลักษณะดังรูปที่ 4.6



CBAR = 80H

รูปที่ 4.6 การแบ่งพื้นที่หน่วยความจำที่ใช้ในโครงการ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โดยกำหนดให้ AREA 1 เป็นส่วน USER PROGRAM ส่วน SYSTEM เป็นของ COMMON AREA 0 ดังนั้นเมื่อเราให้ AREA 1 เริ่มที่ 8000H ก็จะให้ RAM ได้เต็มที่ถึง 32 K เต็ม ส่วน SYSTEM ก็คือ ของ AREA 0 ซึ่งเป็นส่วน MONITOR แต่ในส่วนนี้เราได้กำหนดไว้ถึง 32 K คือจาก 7FFF ลงไปจนถึง 0000H ซึ่งใน SYSTEM เราอาจจะใช้ RAM ไว้ใน ADDRESS ช่วงนี้เพื่อเป็นเนื้อที่ของ STACK ก็จะทำให้เราย้ายเนื้อที่การใช้งานได้เต็ม

#### 4.3 INTERRUPT

มีด้วยกัน 12 INTERRUPT แบ่งเป็น 4 INTERRUPT ภายนอก และ 8 INTERRUPT ภายใน โดยลำดับความสำคัญจากมากไปหาน้อย ดังนี้ TRAP (ภายใน) , (ภายนอก) NMI , INTO , INT1 , INT2 , (ภายใน) TIMER 0 , TIMER 1 , DMA CHANNEL 0 , DMA CHANNEL 1 , CLOCK SERIAL , ASCI CHANNEL 0 และ ASCI CHANNEL 1

##### 4.3.1 REGISTER FLAG INTERRUPT

INTERRUPT VECTOR LOW (IL) , INTERRUPT VECTOR HIGH (I) , INTERRUPT TRAP CONTROL (ITC) และ FLAG IEF1 , IEF2 โดยที่ FLAG IEF1 จะใช้ในการ ENABLE INTERRUPT ภายในทั้งหมดยกเว้น TRAP

##### 4.3.2 INTERRUPT VECTOR LOW REGISTER

ใช้เป็น VECTOR TABLE BYTE ต่ำของ INTERRUPT ภายนอก INT1, INT2 และ INTERRUPT ภายในทั้งหมดยกเว้น "TRAP" โดย 3 BIT สูงของ IL สามารถโปรแกรมได้ แต่ 5 BIT จะถูก FIX ดังรูปที่ 4.7

Interrupt Source	Priority	IL			Fixed Code				
		b7	b6	b5	b4	b3	b2	b1	b0
INT <sub>1</sub>	Highest                   Lowest	*	*	*	0	0	0	0	0
INT <sub>2</sub>		*	*	*	0	0	0	1	0
PRT channel 0		*	*	*	0	0	1	0	0
PRT channel 1		*	*	*	0	0	1	1	0
DMA channel 0		*	*	*	0	1	0	0	0
DMA channel 1		*	*	*	0	1	0	1	0
CS/O		*	*	*	0	1	1	0	0
ASCI channel 0		*	*	*	0	1	1	1	0
ASCI channel 1		*	*	*	1	0	0	0	0

\* Programmable

รูปที่ 4.7 VECTOR TABLE

ตั้งมีการ INTERRUPT ส่วนใหญ่จะเป็น MODE 2 คือ นำค่าใน I และ IL หรือจากอุปกรณ์ที่ขอ INTERRUPT ในกรณี INTO มาประกอบกันเป็น ADDRESS ที่จะเก็บ ข้อมูลที่จะกระโดดไปเช่น I = 10H และ IL = 40H และใน ADDRESS 1040H มีข้อมูล 00H , 60H ตามลำดับ เมื่อเกิด INTERRUPT ขึ้น ก็จะกระโดดไปทำโปรแกรมที่ตำแหน่ง 6000H นั้นเอง

4.3.3 INT/TRAP CONTROL REGISTER

BIT 7 6 5 4 3 2 1 0

TRAP	UFO				ITE2	ITE1	ITE0
------	-----	--	--	--	------	------	------

รูปที่ 4.8 INT/TRAP CONTROL REGISTER

ITE 2,1,0 : INTERRUPT ENABLE 2,1,0 ใช้ ENABLE และ DISBLE INTERRUPT ภายนอก ถ้าเป็น 0 จะ DISBLE แต่ถ้า BIT นี้จะไม่ใช่ทำให้เกิด INTERRUPT ขึ้นทันทีจนกว่าจะทำคำสั่ง EI ตั้งขึ้น INTO

จะต่างกับ Z80 ตรงที่ส่วนนี้ แต่เมื่อเกิด RESET ITE0 จะถูก SET เป็น 1 โดยอัตโนมัติเพื่อใช้ร่วมกับ EI หรือ DI อย่างเดียว เช่น Z80 แต่ ITE1 และ ITE2 จะเป็น 0

TRAP : จะเป็น 1 เมื่อทำคำสั่งที่ไม่มีใน Z80180 TRAP สามารถ RESET ภายได้โปรแกรมควบคุมได้ แต่ไม่สามารถเขียน 1 เข้าไปได้ระหว่าง RESET จะถูก CLEAR

UFO : UNDEFINED FETCH OBJECT เมื่อ TRAP เกิดขึ้น UFO จะให้ค่าของตำแหน่งที่คิดในคำสั่งนั้นไว้ใน STACK เนื่องจาก TRAP อาจเกิดขึ้นจาก OPCODE 2 หรือ 3 BYTE UFO จะปรับค่า PC ให้ คือ ถ้าเป็นคำสั่ง OPCODE 2 BYTE UFO จะเป็น 0 และจะทำให้ PC ของคำสั่งถัดไปจากคำสั่งที่ไม่ใช่ของ Z80180 จะถูกลดลง 1 แต่ถ้า UFO = 1 คำสั่งที่คิดจะมี OPCODE 3 BYTE และ PC จะถูกลดลง 2 ตำแหน่ง และค่า PC นี้จะถูกเก็บไว้ใน STACK เช่น

2000 ED 99

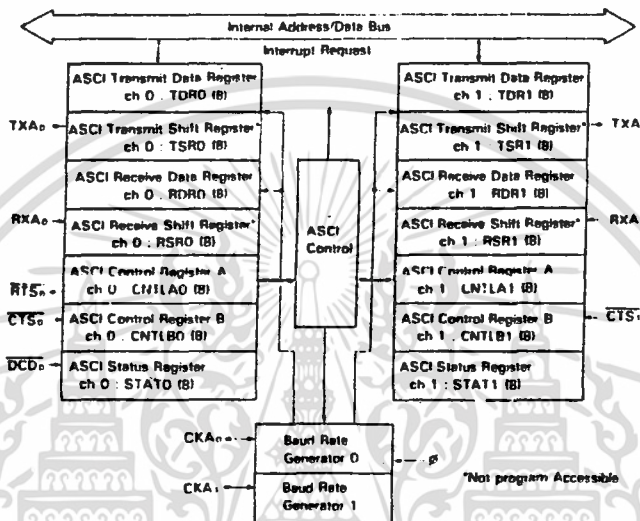
2002 <--PC ที่คำสั่งถัดไป

เมื่อ CPU RUN มาพบข้อมูลที่ตำแหน่งที่ 2000 ก็จะเกิด INTERRUPT TRAP ขึ้น และรู้ด้วยว่าเป็นคำสั่ง 2 BYTE และ PC ที่คำสั่งถัดไปคือ ADDRESS 2002 แต่ FLAG UFO จะถูกทำให้เป็น 0 เพื่อปรับค่า PC นั้นด้วยการลดลง 1 เช่น ADDRESS 2001 ซึ่งก็คือตำแหน่งข้อมูลที่คิดนั่นเอง

TRAP INTERRUPT เป็นเหมือน NMI คือ ไม่สามารถหยุดได้เมื่อเกิดการกระทำคำสั่งผิด ซึ่งซึ่งเป็นตัวช่วยให้เกิดความน่าเชื่อถือทางด้าน SOFTWARE และอาจใช้เพิ่มคำสั่งได้อีกด้วย BIT TRAP ใน ITC จะถูก SET เป็น 1 และ UFO จะ SET หรือไม่ SET ขึ้นอยู่กับว่าเป็นคำสั่ง 2 หรือ 3 BYTE และ FLAG UFO นี้ก็จะไปปรับ PC ให้ถูกต้อง และเก็บไว้ใน STACK แล้วกระโดดไป RUN ที่ ADDRESS 0000H

4.4 ASYNCHRONOUS SERIAL COMMUNICATION INTERFACE (ASCI)

มีด้วยกัน 2 CHANNEL โดยมี BLOCK DIAGRAM ดังรูปที่ 4.9



รูปที่ 4.9 ASCI BLOCK DIGRAM

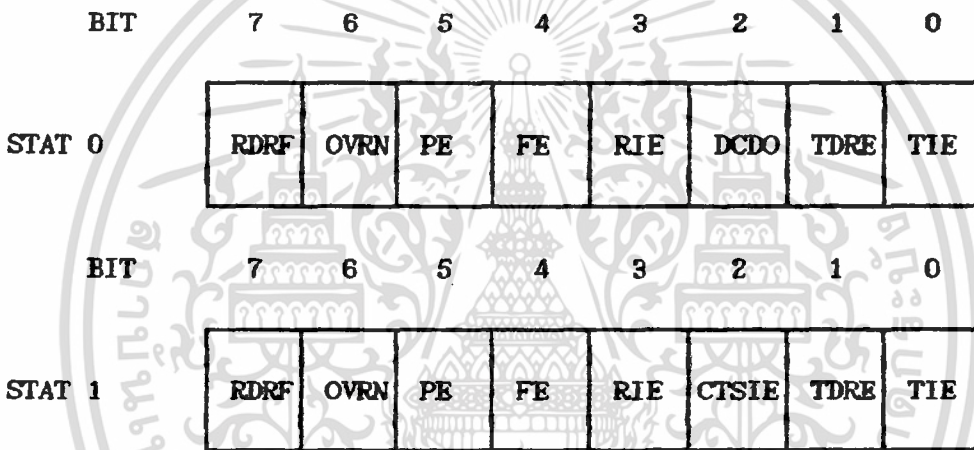
- TSR 0,1 เป็นการ SHIFT REGISTOR ที่รับข้อมูลจาก TRANSMIT DATA REGISTOR (TDR) แล้วนำข้อมูลนั้น SHIFT ออกที่ขา TXA
- TDR 0,1 เป็น REGISTOR ที่ใช้ส่ง DATA ออกไปที่ขา TXA โดยการนำข้อมูลใน TDR ส่งไปที่ TSR เมื่อ TSR ว่างลง และสามารถที่จะเขียนข้อมูลเข้าไปที่ TDR ได้อีกในขณะที่ TSR กำลัง SHIFT ข้อมูลออกไปที่ขา TXA
- RSR 0,1 เป็น REGISTOR ที่รับข้อมูลจาก RXA PTN เมื่อรับเต็ม RUFFER แล้ว ก็จะ SHIFT ไปที่ RDR ถ้า RSR ไม่ว่างเมื่อมีการรับข้อมูล BYTE ต่อไปเข้าไปอีกจะเกิดการชนกันขึ้น จะทำให้เกิดการผิดพลาดและผลของการผิดพลาดก็จะแสดงที่ REGISTOR สถานะ REGISTOR นี้ไม่สามารถโปรแกรมได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

RDR 0,1 คือ REGISTER ที่ใช้เก็บข้อมูลที่รับเข้ามาจาก RXA PIN และในขณะที่ RDR กำลังบรรจุข้อมูลที่รับเข้ามาจาก RSR ข้อมูล BYTE ถัดไปสามารถเข้ามาต่อได้

4.4.1 STAT 0,1

แต่ละ CHANNEL จะมี REGISTER ใช้สำหรับตรวจสอบการสื่อสารเกี่ยวกับการผิดพลาด และสถานะสัญญาณ CONTROL MODEM การ ENABLE และ DISBLE ASCI ดังรูปที่ 4.10



รูปที่ 4.10 ASCI STATUS REGISTER

**RDRF:** RECIEVE DATA REGISTER FILL จะถูก SET เป็น 1 เมื่อข้อมูลที่รับเข้ามาถูกส่งเข้ามาที่ RDR เรียบร้อยแล้ว (ครบ BYTE) แต่ถ้การรับเกิด ERROR ขึ้น RDRF ก็จะถูก SET ต่าง และข้อมูลที่ผิดพลาดนั้นก็จะถูกส่งมาที่ RDR และคงอยู่ ดังนั้นจะต้องทำการ CLEAR FLAG ERROR RDRF จะถูก CLEAR เป็น 0 เมื่ออ่าน RDR , DCDO เป็น HIGH สำหรับ CHANNEL 0 , IOSTOP และการ RESET

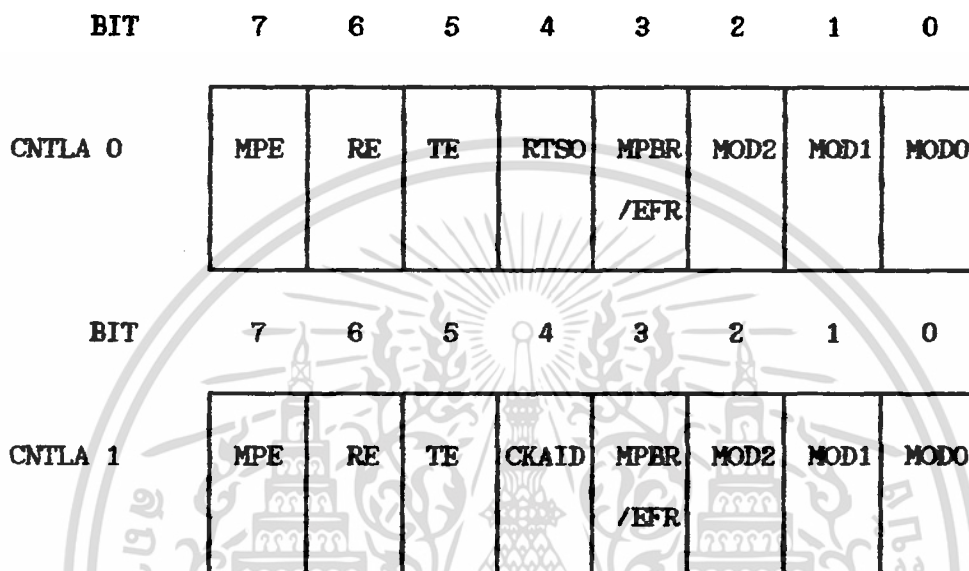
**OVRN:** OVERUN ERROR จะเป็น 1 เมื่อ RDR เต็มและ RSR เต็ม แต่ยังไม่มีการรับข้อมูลเข้ามาอีกจะถูก CLEAR ได้ EFR BIT ใน CNTLA เป็น 0 , DCDO เป็น HIGH IOSTOP และ RESET

- PE: PARITY ERROR เป็น 1 เมื่อข้อมูลที่ได้รับเข้ามา PARITY ผิดและ CLEAR ได้เช่นเดียวกับ OVERUN
- FE: FRAMING ERROR เมื่อข้อมูลที่ได้รับเข้ามาถูกไปจากที่กำหนด BIT FE จะถูก SET เป็น 1 และการ CLEAR เช่นเดียวกับ OVERUN
- RIE: RECIEVE INTERRUPT ENABLE เมื่อเป็น 1 จะอนุญาตให้ ASCII ทำการขอ INTERRUPT ได้ เมื่อ RDRF , OVRN , PE หรือ FE ถูก SET เป็น 1 ด้วย เมื่ออ่าน ASCII ก็จะทำให้สัญญาณ INTERRUPT สำหรับ ASCII CHANNEL 0 INTERRUPT สามารถเกิดขึ้นได้โดยการเปลี่ยนแปลงที่ขารับสัญญาณ INPUT ภายนอกที่ขา DCDO จาก LOW เป็น HIGH และ RIE จะถูก CLEAR เป็น 0 ระหว่าง RESET
- DCDO: DATA CARRIER DETECT BIT นี้จะถูก SET เป็น 1 เมื่อขา INPUT DCDO เป็น HIGH และจะถูก CLEAR เป็น 0 จากการอ่าน STAT 0 ครั้งแรก จากนั้นขา INPUT DCDO จะถูกเปลี่ยนจาก HIGH เป็น LOW และระหว่าง RESET เมื่อ DCDO เป็น 1 ส่วนของภาครับ จะไม่ทำงาน
- CTSIE: CHANNEL 1 CTS ENABLE ที่ CHANNEL 1 มีขา INPUT CTS1 ภายนอก ซึ่ง MULTIPLEX กับ RXS เมื่อ SET BIT นี้เป็น 1 จะถูกเลือกเป็น CTS1
- TDRE: TRANSMIT DATA REGISTER EMPTY เป็นตัวบอกว่าข้อมูลพร้อมแล้วที่จะส่งได้หรือไม่ ถ้าเป็น 1 คือ พร้อมที่จะส่งข้อมูลแล้วให้เขียนข้อมูลเข้าไปที่ TDR ได้ และเมื่อมีการเขียนข้อมูลเข้าไปที่ TDR ก็จะทำให้ TDRE เป็น 0 และข้อมูลใน TDR ก็จะทำการส่งให้ TSR จน TDR ว่างลง TDRE ก็จะกลับเป็น 1 อีกครั้ง
- TIE: TRANSMIT INTERRUPT ENABLE เมื่อเป็น 1 จะอนุญาตให้ ASCII ใช้การส่งแบบ INTERRUPT ได้ โดยที่ TDRE ต้องเป็น 1 ด้วย TIE จะถูก CLEAR เป็น 0 ระหว่าง RESET

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## 4.4.2 CNTLA 0,1

เป็น REGISTER กำหนดการทำงานประกอบด้วย



รูปที่ 4.11 ASCII CONTROL REGISTER A

- MPE :** MULTIPROCESSOR MODE ENABLE ให้อำนาจในการสื่อสารแบบไมโครโปรเซสเซอร์ร่วมจากเมื่อมีการเลือก MODE การสื่อสารแล้ว (MP = 1 ใน CNTLB) ในการสื่อสารแบบ FORMAT ของการสื่อสารจะมี BIT พิเศษเพิ่มเข้ามาเรียกว่า MPE BIT ซึ่ง BIT นี้จะถูกใช้ตรวจสอบหรือใช้งาน เมื่อ ENABLE MPE ให้เป็น 1 และถ้า MPB = 1 เมื่อนั้นภาครับของ MULTIPROCESSOR จะทำงานคือ RDRF และ ERROR FLAG จะทำงาน แต่ถ้า MPB = 0 ASCII จะไม่สนใจข้อมูล BYTE ใดๆถ้า MPE = 0 จะไม่สามารถทำการสื่อสารแบบไมโครโปรเซสเซอร์ร่วมได้ แม้จะ SET MP เป็น 1 แล้วก็ตาม
- RE:** RECIEVER ENABLE ถ้าเป็น 1 จะ ENABLE การรับของ ASCII แต่ถ้าเป็น 0 จะ DISABLE การรับ แต่ RDRF และ ERROR FLAG จะ

### ไม่ถูก RESET ตาม

- TE:** TRANSMIT ENABLE เป็น 1 จะ ENABLE การส่ง ถ้าเป็น 0 จะ DISABLE แต่ TDRE FLAG จะไม่ถูก RESET ตาม
- RTSO:** REQUEST TO SEND CHANNEL 0 เป็น BIT ที่ให้ผลเช่นเดียวกับขา OUTPUT RTS0 คือ ถ้า BIT นี้เป็น 1 ขา OUTPUT RTS0 ก็จะเป็น 1 ถ้า BIT นี้เป็น 0 ขา OUTPUT ก็เป็น 0 RTS0 BIT นี้ จะถูก SET เป็น 1 ระหว่าง RESET
- CKA1D:** CKA1 CLOCK DISABLE ซึ่งขา CKA1 จะ MULTIPLEX กับ TEND0 เมื่อ BIT นี้เป็น 1 จะเลือกเป็นขา TEND0 แต่ถ้าเป็น 0 ก็จะเป็น ขา CLOCK ของ ASCII CHANNEL 1 BIT นี้จะเป็น 0 ระหว่าง RESET
- MPBR/EFR:** MULTIPROCESSOR BIT RECIEVE / ERROR FLAG เมื่อ BIT นี้ ถูกอ่านจะรีเซ็ต MPB BIT ในกรณีไมโครโปรเซสเซอร์ที่จะทำการ ติดต่อกันแล้ว และจะ DISABLE ไมโครโปรเซสเซอร์ตัวอื่นๆโดยการ ส่ง MPB BIT ให้เป็น 0 เมื่ออ่านก็จะรู้ว่า MPB เป็น 0 จริง แต่ถ้า เขียน 0 ให้ BIT นี้จะเป็นการ RESET ERROR FLAG ในการรับ
- MOD 2,1,0:** ASCII DATA FORMAT MODE 2,1,0 โดยสรุปได้ดังตารางที่ 4.4

MOD 2	MOD 1	MOD 0	DATA FORMAT
0	0	0	START + 7 BIT DATA + 1 STOP
0	0	1	START + 7 BIT DATA + 2 STOP
0	1	0	START + 7 BIT DATA + PARITY + 1 STOP
0	1	1	START + 7 BIT DATA + PARITY + 2 STOP
1	0	0	START + 8 BIT DATA + 1 STOP
1	0	1	START + 8 BIT DATA + 2 STOP
1	1	0	START + 8 BIT DATA + PARITY + 1 STOP
1	1	1	START + 8 BIT DATA + PARITY + 2 STOP

ตารางที่ 4.4 DATA FORMAT

ASCII CONTROL REGISTER B 0,1 (CNTLB 0,1)

BIT 7 6 5 4 3 2 1 0

MPBT	MP	CTS/ PS	FEO	DR	SS2	SS1	SS0

รูปที่ 4.12 ASCII CONTROL REGISTER B

MPBT: MULTIPROCESSOR BIT TRANSMIT ใช้ส่ง MPB BIT โดยถ้า  
MPBT = 1 เมื่อนั้น MPB BIT = 1 และ MPBT = 0 MPB ก็ = 0  
ด้วย ระหว่าง RESET ไม่สามารถกำหนดได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- MP: MULTIPROCESSOR MODE ถ้าเป็น 1 จะเป็นการ SET การติดต่อแบบไมโครโปรเซสเซอร์ร่วม โดยใช้ FORMAT ของ MOD 2 กับ MOD 0 โดยยกเว้น MOD 1 ดังนี้  
START BIT + 7 หรือ 8 DATA BIT + MPB BIT + 1 หรือ 2 STOP BIT ระหว่าง RESET MP จะเป็น 0
- CTS/PS: CLEAR TO SEND / PRESCALE เมื่ออ่าน BIT นี้จะให้เห็นสถานะของขา INPUT CTS ภายนอก ถ้าขา CTS เป็น HIGH ภาคส่งของ ASCII จะไม่ทำงาน แต่ถ้าเขียนเข้าไปที่ BIT นี้จะเป็นการกำหนด BAUD RATE BIT นี้เป็น 0 ระหว่าง RESET
- PEO: PARITY EVEN ODD BIT นี้จะไม่มีผลต่อการ ENABLE หรือ DISBLE ของ PARITY (MOD 1 ใน CNTLA) แต่จะใช้เลือกกว่ามีการ ENABLE PARITY ใน MOD 1 จะให้ PARITY PEO = 0 คือ 0 แต่ถ้า = 1 คือ 1
- DR: DIVIDE RATIO ใช้กำหนด BAUD RATE BIT นี้จะเป็น 0 ระหว่าง RESET
- SS 2,1,0: SOURCE / SPEED SELECT 2,1,0 ใช้กำหนด CLOCK ว่าจะเป็นภายในหรือภายนอก (โดยภายนอก คือ ขา CLOCK CKA) และเป็นตัวกำหนด BAUD RATE ด้วย ระหว่าง RESET ทั้ง 3 BIT นี้จะเป็น 1 คือ เป็นการให้ CLOCK จากภายนอกนั่นเองซึ่งจากที่กล่าวมาในการกำหนด BAUD RATE จึงมีด้วยกันหลายตัวสามารถสรุปเป็นตารางที่ 4.5

Prescaler		Sampling Rate		Baud Rate				General Divide Ratio	Baud Rate (Example) (BPS)			CKA	
PS	Divide Ratio	DR	Rate	SS2	SS1	SS0	Divide Ratio		φ = 6.144 MHz	φ = 4.608 MHz	φ = 3.072 MHz	I/O	Clock Frequency
0	φ + 10	0	16	0	0	0	+1	φ + 160	38400		19200		φ + 10
				0	0	1	2	320	19200		9600		70
				0	1	0	4	640	9600		4800		40
				0	1	1	8	1280	4800		2400		80
				1	0	0	16	2560	2400		1200		160
				1	0	1	32	5120	1200		600		320
	1	1	0	64	10240	600		300		640			
	1	1	1	-	fc + 16	-	-	-	-	-	fc		
	0	1	64	0	0	0	+1	φ + 640	9600		4800		φ + 10
	0			0	1	2	1280	4800		2400		70	
	0			1	0	4	2560	2400		1200		40	
	0			1	1	8	5120	1200		600		80	
1	0			0	16	10240	600		300		160		
1	0			1	32	20480	300		150		320		
1	1	0	64	40960	150		75		640				
1	1	1	-	fc + 64	-	-	-	-	-	fc			
1	φ + 30	0	16	0	0	0	+1	φ + 480		9600		φ + 30	
				0	0	1	2	960		4800		80	
				0	1	0	4	1920		2400		120	
				0	1	1	8	3840		1200		240	
				1	0	0	16	7680		600		480	
				1	0	1	32	15360		300		960	
	1	1	0	64	30720		150		1920				
	1	1	1	-	fc + 16	-	-	-	-	fc			
	0	1	64	0	0	0	+1	φ + 1920		2400		φ + 30	
	0			0	1	2	3840		1200		80		
	0			1	0	4	7680		600		120		
	0			1	1	8	15360		300		240		
1	0			0	16	30720		150		480			
1	0			1	32	61440		75		960			
1	1	0	64	122880		37.5		1920					
1	1	1	-	fc + 64	-	-	-	-	fc				

ตารางที่ 4.5 BAUD RATE SELECTION

4.5 PROGRAMABLE RELOAD TIMER (PRT)

มีด้วยกัน 2 CHANNEL เป็น 16 BIT PROGRAMABLE RELOAD TIMER และสำหรับ CHANNEL 1 มีขา OUTPUT สามารถให้สัญญาณได้ถึง 2 CHANNEL ประกอบด้วย

4.5.1 TIMER DATA REGISTER (TMDR CH0, CH1)

เป็น REGISTER 16 BIT ใช้กำหนด TIMER โดย ADDRESS I/O สูงเท่ากับค่า TIMER ค่าสูง ระหว่าง RESET TMDR0 และ TMDR1 จะเป็น OFFFFH โดย TMDR จะนับลง 1 ครั้งทุกๆ 20 CLOCK SYSTEM เมื่อ TMDR นับลงเป็น 0 ค่าใน RELOAD จะถูก LOAD มาให้ TMDR โดยอัตโนมัติ การอ่านค่าใน TMDR อ่านได้เลยโดยไม่ต้องหยุด PRT แต่ถ้าเป็นการเขียนต้องหยุด PRT ก่อน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

#### 4.5.2 TIMER RELOAD REGISTER (RLDR CH0, CH1)

ใช้ LOAD ค่าที่อยู่ใน RLDR ไปให้ TMDR เมื่อ TMDR ลดลงเป็น 0

#### TIMER CONTROL REGISTER (TCR)

เป็น REGISTER ควบคุมสถานะ และ CONTROL ดังรูปที่ 4.13



รูปที่ 4.13 TIMER CONTROL REGISTER

- TIF 1:** TIMER INTERRUPT FLAG 1 เมื่อ TMDR 1 ลดลงเป็น 0 TIF จะถูก SET เป็น 1 และถ้า TIE 1 = 1 ก็จะทำให้เกิด INTERRUPT ขึ้นได้ TIF จะถูก CLEAR เป็น 0 ก็คือเมื่อทำการอ่านค่าใน TCR ก็อ่านค่าใน BYTE HIGH หรือ LOW ของ TMDR 1 ระหว่าง RESET TIF = 0
- TIF 0:** TIMER INTERRUPT FLAG หลักการเช่นเดียวกับ TIF 1
- TIE 1,0:** TIMER INTERRUPT ENABLE 0,1 เมื่อ SET เป็น 1 จะอนุญาตให้ INTERRUPT ได้ ระหว่าง RESET 2 BIT นี้จะเป็น 0
- TOC 1,0:** TIMER OUTPUT CONTROL 2 BIT นี้ใช้ควบคุม OUTPUT ของ PRT 1 โดยถ้าทั้ง 2 BIT นี้เป็น 0 จะเป็นการใช้งาน A18 นอกนั้นก็จะเป็นการกำหนดให้ TOUT เป็น HIGH, LOW หรือ TOGGLE ดังตารางที่ 4.6 ระหว่าง RESET 2 BIT นี้จะเป็น 0

TOC 1	TOC 0	OUTPUT
0	0	ADDRESS A18
0	1	TOGGLE
1	0	0
1	1	1

ตารางที่ 4.6 TIMER OUTPUT CONTROL

TDE 1,0 : TIMER DOWN COUNT ENABLE เมื่อ SET เป็น 1 ก็คือให้เริ่มทำการนับ TMDR ได้ แต่ถ้าเป็น 0 การนับจะหยุดทำงาน ระหว่าง RESET BIT ทั้งคู่จะเป็น 0

การคำนวณเวลาเราทราบแล้วว่า TIMER จะนับลงทุกๆ 20 CLOCK ดังนั้น ถ้า X'TAL ที่ใช้งานคือ 10 MHz ความถี่ที่ RUN บน BOARD = 5 MHz พาค่า TIMER คือ

$$T = 1/F = 0.200 \text{ } \mu\text{S} \text{ ต่อ } 1 \text{ CLOCK}$$

$$20 \text{ CLOCK} = 0.200 \times 20 = 4 \text{ } \mu\text{S}$$

นั่นก็คือ TIMER นับลง 1 ครั้ง ทุกๆ 4  $\mu\text{S}$  ที่ X'TAL 10 MHz เช่นนี้ TMDR มีค่า = 1 และ SET FLAG INTERRUPT ไว้ ก็จะทำให้ PRT เกิดการ INTERRUPT ทุกๆ 8  $\mu\text{S}$  เพราะการนับลงจะนับจากตัวเองลงก่อนคือ 1 แล้วก็ 0 จึงเท่ากับ 2 ครั้งนั่นเอง และถ้าให้กำเนิดสัญญาณสแควที่ TOUT ก็จะทำให้เกิดการ TOGGLE กันทุกๆจำนวนที่ให้นับ เช่น จากตัวอย่างข้างบนก็จะเห็น HIGH 8  $\mu\text{S}$  และ LOW 8  $\mu\text{S}$  ดังนั้น 1 ออกสัญญาณจะประมาณ 16  $\mu\text{S}$  หรือความถี่จะต่ำลงเท่าหนึ่งของค่าเวลาที่คิดจากจำนวนครั้งในการนับค่าก็ได้

#### 4.6 คำสั่งเพิ่มเติมที่ใช้

SLP	เมื่อใช้คำสั่งนี้ CPU จะหยุดทำงานบางอย่างทำให้ใช้กำลังงานต่ำ
MLT	MULTIPLY ใช้สำหรับคูณเลข 8 BIT 2 จำนวน โดยผลลัพธ์จะเป็น 16 BIT โดย REGISTOR ที่ใช้ในการคูณอาจจะเป็น BC, DE , HL หรือ SP โดยผลลัพธ์จะได้ที่ REGISTOR คู่กัน
OUTO (m),g	OUTPUT, IMMEDIATE I/O OUT ค่าจาก REGISTOR ใดๆ ไปยัง PORT 8 BIT (A0-A7) REGISTOR ที่มี A, BC , DE , HL

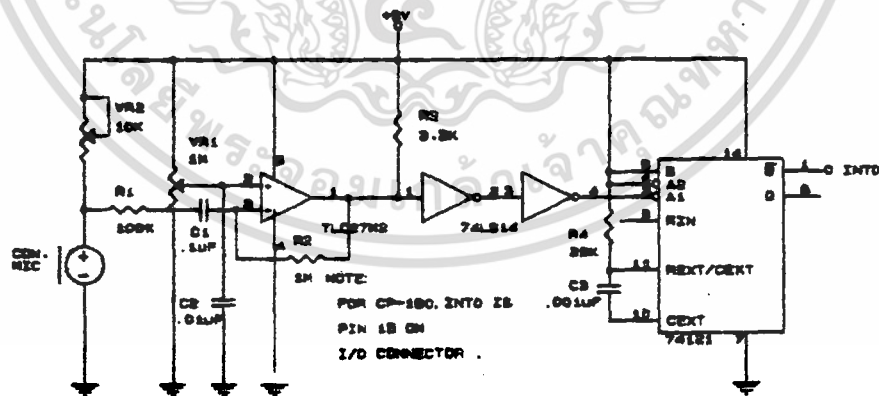
ทั้งหมดที่กล่าวมาข้างต้น เป็นรายละเอียดบางส่วน ของ Z80180 ที่จำเป็น ในการทำโครงงานนี้ อันเป็นพื้นฐานสำคัญในการทำความเข้าใจเนื้อหาในบทอื่นๆ ต่อไป และเพื่อความคล่องตัวในการนำเสนอ ในบทต่อไป เราจะขอเรียก MPU เบอร์ Z80180 สั้นๆ ว่า Z180 MPU

## บทที่ 5

### ระบบแปลงระดับความถี่เสียง ไปเป็นข้อมูลดิจิทัล

ในบทนี้จะเป็นการนำเอาความรู้ต่างๆ ในบทก่อนหน้านี้มาใช้ เพื่อออกแบบสร้าง ระบบแปลงระดับความถี่เสียง ไปเป็นข้อมูลดิจิทัล โดยเนื้อหาในบทนี้จะถูกแบ่งออกเป็น 2 ส่วน ประกอบด้วย ส่วนของวงจรต่างๆที่ใช้ในระบบหรือส่วนของฮาร์ดแวร์ กับส่วนของโปรแกรมที่ใช้ในการควบคุมการทำงานของระบบ ซึ่งเป็นส่วนที่ใช้สำหรับควบคุมไมโครคอมพิวเตอร์ เพื่อการประมวลผลและควบคุมการเชื่อมต่อกับ วงจรอินพุต/เอาต์พุต ต่าง ๆ ให้ทำงานสอดคล้องประสานกันตามแนวความคิดที่เราได้กล่าวไว้ในบทที่ 3

#### 5.1 วงจรตรวจจับระดับความถี่เสียง ( Pitch detector )

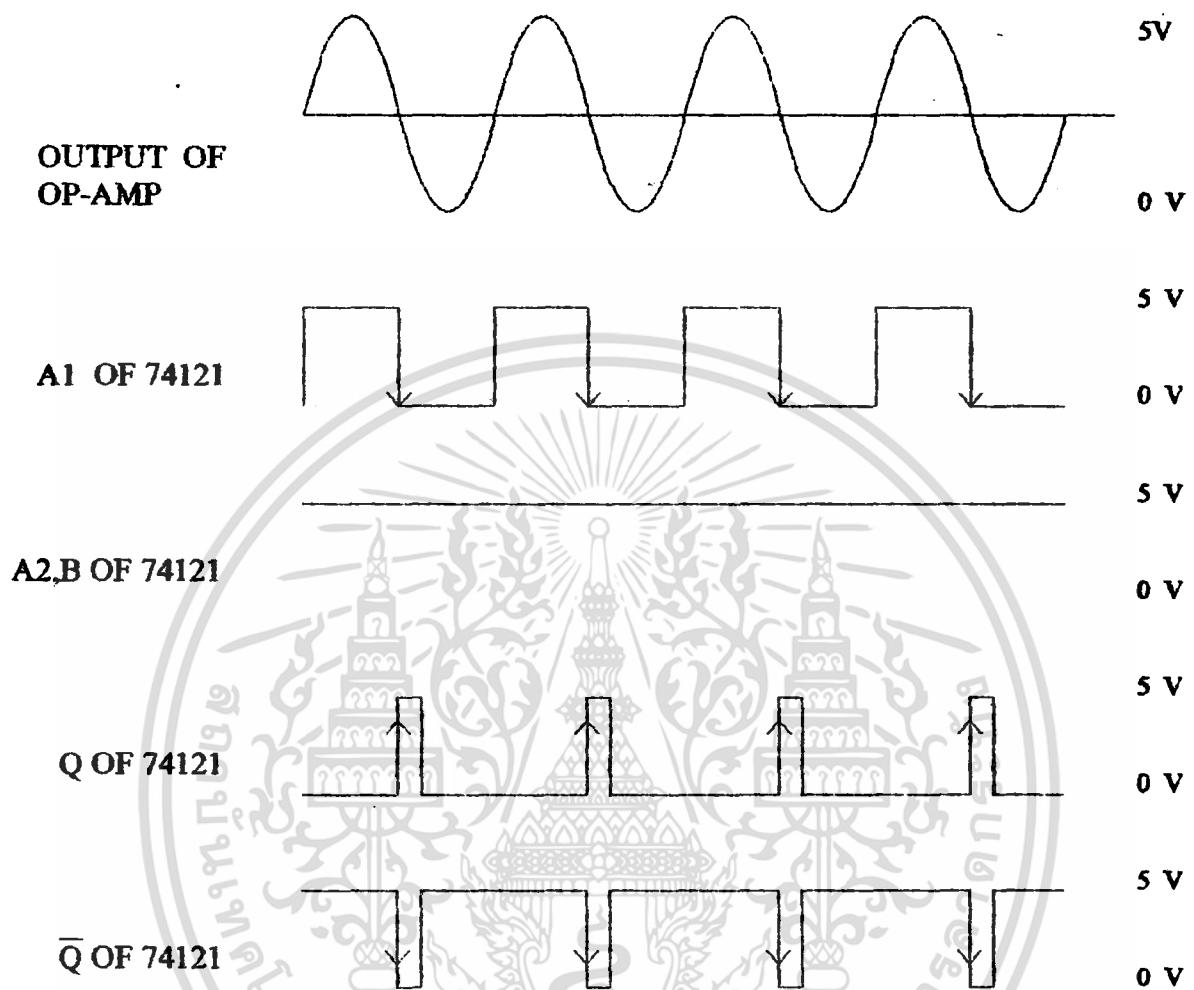


รูปที่ 5.1 วงจรตรวจจับระดับความถี่เสียง ( Pitch detector )

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

วงจรมีเป็นส่วนที่ค่อนข้างซับซ้อนการทำงานของส่วนที่เรียกว่า Wave Shaper ซึ่งได้กล่าวไว้ในหัวข้อที่ 3.3 แต่ในที่นี้เราจะเรียกมันว่า Pitch detector เพื่อให้สอดคล้องกับหน้าที่การทำงานของมันในระบอบของเรา รูปที่ 5.1 ได้แสดงวงจรตรวจจับระดับความถี่เสียงที่เราออกแบบขึ้นเมื่อใช้ในโครงการพิเศษนี้

วงจรตรวจจับระดับความถี่เสียงนี้เป็นภาคอิมิตเตอร์ของระบอบ เมื่อมีสัญญาณเสียงเข้ามาทางไมโครโฟนชนิดคอนเดนเซอร์ ซึ่งทำหน้าที่เปลี่ยนคลื่นเสียงให้เป็นสัญญาณไฟฟ้าและจะถูกขับไปถึงโดย C1 เข้าทางขาอินเวอร์ตของออปแอมป์ ชนิดมอสมอสเบอร์ TLC27MC ซึ่งประกอบเป็นวงจรขยายแบบกลับเฟสที่มีอัตราขยายแรงดันประมาณ 10 เท่า โดยกำหนดได้จากค่า R1 และ R2 สำหรับ VR1 จะทำหน้าที่สร้างแรงดันอ้างอิงให้กับขาอินเวอร์ตของ TLC27MC C2 นั้นจะทำหน้าที่บายพาสสัญญาณรบกวนที่ไม่ต้องการลงกราวด์ไป สัญญาณที่ถูกขยายแล้วจะถูก PULL-UP ด้วย R3 เพื่อส่งเข้าขาอินพุตของไมโครคอนโทรลเลอร์ 2 ตัวซึ่งต่ออนุกรมกันอยู่ ไมโครคอนโทรลเลอร์จะทำการปรับรูปคลื่นสัญญาณให้อยู่ในรูปของคลื่นสี่เหลี่ยมโดยที่คาบเวลาของคลื่นสัญญาณยังคงเดิม สำหรับไมโครคอนโทรลเลอร์นั้น เราเลือกใช้เบอร์ 74LS14 ซึ่งภายในประกอบด้วยไมโครคอนโทรลเลอร์ถึง 4 ตัว สัญญาณสี่เหลี่ยมจากเอาต์พุตของไมโครคอนโทรลเลอร์นั้น จะถูกป้อนให้กับขา A1 ของไอซี โมโนสเตเบิลเบอร์ 74121 วงจรโมโนสเตเบิลจะทำการเปลี่ยนแปลงคลื่นสี่เหลี่ยมให้อยู่ในรูปของขบวนพัลส์สลับ โดยมีระยะเวลาห่างของเวลาระหว่างพัลส์แต่ละลูกเท่ากับคาบเวลาของสัญญาณที่ขา A1 และเท่ากับคาบเวลาของสัญญาณที่อินพุตของออปแอมป์ด้วย และสำหรับความกว้างของพัลส์ สามารถกำหนดได้จากค่าของ R4 และ R3 เอาต์พุตจากขา Q ของ 74121 ซึ่งเป็นขบวนพัลส์สลับนั้น จะถูกส่งไปอินเทอร์รัพท์การทำงานของ Z180 MPU เพื่อให้ Z180 MPU ทำการจับเวลาระหว่างพัลส์แต่ละลูกที่เข้ามาอินเทอร์รัพท์ ทำให้เราสามารถทราบคาบเวลาของสัญญาณเสียงที่เราทำการตรวจจับได้



รูปที่ 5.2 รูปสัญญาณ อ. จุดต่างๆ ในวงจร Pitch detector

TLC27M2 เป็นไอซีอินเวอร์ตที่มีคุณสมบัติพิเศษคือ สามารถทำงานที่แรงดันไฟเลี้ยง 5 โวลต์ ซึ่งเป็นระดับแรงดันไฟเลี้ยงของไอซีชนิด TTL สำหรับ DATA SHEET ของ TLC27M2 และ ไอซีตัวอื่นๆ ที่อยู่ในวงจรสามารถดูได้จากภาคผนวกท้ายเล่ม

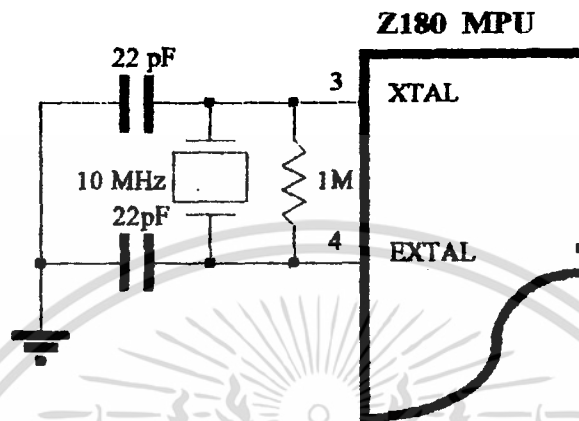
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดลอกเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## 5.2 การปรับปรุง CP-180 ให้ทำงานในระบบมิติ

การนำ CP-180 มาใช้ในโครงการพิเศษนี้ จะต้องทำการแก้ไขดัดแปลงฮาร์ดแวร์บางส่วนเสียก่อน เพื่อให้เหมาะสมกับการทำงานในระบบมิติ มีส่วนประกอบทางฮาร์ดแวร์อยู่ 2 ส่วน ที่จะต้องถูกแก้ไขก่อน ส่วนแรกคือ วงจรกำเนิดสัญญาณเวลาของระบบ อีกส่วนหนึ่งคือ ส่วนของเอาต์พุตของ ASCII (Asynchronous Serial Interface) ซึ่งเป็นตัวควบคุมการสื่อสารข้อมูลแบบอนุกรมของ Z180 MPU

### 5.2.1 การแก้ไขวงจรกำเนิดสัญญาณนาฬิกา

การสื่อสารข้อมูลในระบบมิตินั้น ถูกกำหนดให้ทำการรับส่งข้อมูลด้วยอัตราเร็ว 31,250 บิตต่อวินาที (BPS) หรือมีความกว้างของช่องข้อมูลเท่ากับ 32 ไมโครวินาที CP-180 นั้นใช้คริสตอลที่มีความถี่ 12.288 เมกกะเฮิร์ต เป็นตัวกำเนิดความถี่สัญญาณนาฬิกาให้กับ Z180 MPU ความถี่ 12.288 เมกกะเฮิร์ตนี้ จะถูกวางจรรยาในของ Z180 MPU ทำการหารด้วย 2 เพื่อให้ได้เป็นสัญญาณนาฬิกาของระบบ ซึ่งมีค่าประมาณ 6.144 เมกกะเฮิร์ต จากตาราง 4.5 ในบทที่ 4 หากเราดูในช่องของอัตราเร็วในการรับส่งข้อมูล (Baud rate) อนุกรมแบบอะซิงโครนัส จะเห็นว่า ในคอลัมน์แรก (0 = 6.144 MHz) ไม่มีอัตราการส่งในแถวใดที่ตรงตามข้อกำหนดของมิติเลย วิธีที่ง่ายที่สุดที่จะสามารถปรับปรุงให้การสื่อสารข้อมูลอนุกรมของ CP-180 สามารถทำงานที่อัตราการรับส่ง 31,250 บิตต่อวินาทีได้ ก็คือการเปลี่ยนความถี่ของสัญญาณนาฬิกาในระบบเสียใหม่ ทั้งนี้เพราะ Z180 MPU นั้นใช้สัญญาณนาฬิกาของระบบมาทำการหารด้วยค่าต่างๆ เพื่อให้ได้เป็นอัตราการรับส่งข้อมูลที่ต้องการ เราเพียงแต่ทำการเปลี่ยนคริสตอลในวงจรกำเนิดสัญญาณให้เป็นค่าที่เหมาะสม ประกอบกับการควบคุมการหารความถี่บนตัว Z180 MPU เราก็จะสามารถทำให้ CP-180 ทำงานกับระบบมิติได้



รูปที่ 5.3 ส่วนของวงจรกำเนิดสัญญาณนาฬิกาบนบอร์ด CP-180

ถ้าเราใช้คริสตัลที่มีความถี่ประมาณ 10 เมกะเฮิร์ต เราจะได้ความถี่ของสัญญาณนาฬิกาที่มีค่าประมาณ 5 เมกะเฮิร์ต เราจะต้องทำการตั้งค่าให้กับรีจิสเตอร์ควบคุมของ ASCII เสียใหม่ โดยการกำหนดค่าให้แฟลท (Prescale) มีค่าเป็น 0 เพื่อเลือกหารความถี่ของสัญญาณนาฬิกาด้วย 10 ก่อน จากนั้นกำหนดค่าให้ แฟลท DR (Divide Ratio) มีค่าเป็น 0 เช่นกัน เพื่อหารความถี่สัญญาณนาฬิกาด้วย 16 อีกครั้ง จากนั้นเราจะทำการกำหนดค่าให้แฟลท SS2, SS1 และ SS0 เป็น 0 ด้วย เพื่อให้ อัตราการหารเป็น 1 อัตราการหารรวมทั้งหมดจะเป็น 160 ซึ่งเราสามารถทำการกำหนดได้ในขั้นตอนของการโปรแกรม

ความถี่สัญญาณนาฬิกาของระบบมีค่า	5	เมกะเฮิร์ต
อัตราการหารรวมของ ASCII เป็น	160	
จะได้อัตราส่งมีค่า	=	(5 MHz) / 160
	=	31.25 Kbaud

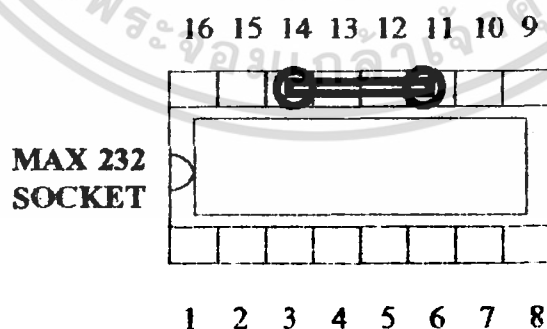
การแก้ไขทางฮาร์ดแวร์ทำได้โดยการนำคริสตัลที่มีความถี่ 10 เมกะเฮิร์ต มาเปลี่ยนแทนคริสตัลตัวเดิม ในตำแหน่งซึ่งแสดงไว้ในรูป 5.3

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 5.2.2 การแก้ไขส่วนเอาต์พุตของ ASCII

CP-180 นั้นได้ถูกออกแบบให้สื่อสารแบบอนุกรมโมเด็มมาตรฐาน RS-232 เอาต์พุตของ ASCII ที่เราใช้คือ ขา TXA1 ของ Z180 MPU โดยจะถูกเชื่อมต่อเข้ากับขา T1I ของ MAX-232 ซึ่งจะทำหน้าที่ในการแปลงระดับแรงดันให้ได้ 12 โวลต์เสียก่อน ขา T1O ของ MAX-232 จะถูกเชื่อมเข้ากับคอนเนคเตอร์ 4 ขาที่มีชื่อว่า Serial1 เพื่อให้ข้อมูลที่ทำการแปลงระดับแรงดันแล้วถูกนำไปใช้จากจุดนี้ แต่ในการสื่อสารข้อมูลทางนี้ เราใช้วิธีการที่เรียกว่า "ลูปของกระแส" ดังนั้นเราไม่จำเป็นต้องทำการแปลงระดับแรงดัน แต่ต้องการวงจรขับกระแส (Driver) เพื่อขับกระแสขนาด 5 มิลลิแอมป์ ในการสื่อสารข้อมูลแทน สำหรับส่วนของวงจรขับกระแสนี้ เราจะทำการเพิ่มเติมเข้าไป

ในโครงงานนี้ เราได้ทำการถอดไอซี MAX-232 ออก แล้วทำการเชื่อมสายบนหมอกเทศของมัน โดยเชื่อมขา 11 เข้ากับขา 14 เพื่อส่งข้อมูลจากขา TXA1 ตรงไปยังคอนเนคเตอร์ Serial1 เลย จากคอนเนคเตอร์ Serial1 นี้เราจะทำการเชื่อมต่อเข้ากับวงจรขับกระแสอีกที



รูปที่ 5.4 แสดงการแก้ไขส่วนเอาต์พุตของ ASCII บนบอร์ด CP-180

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

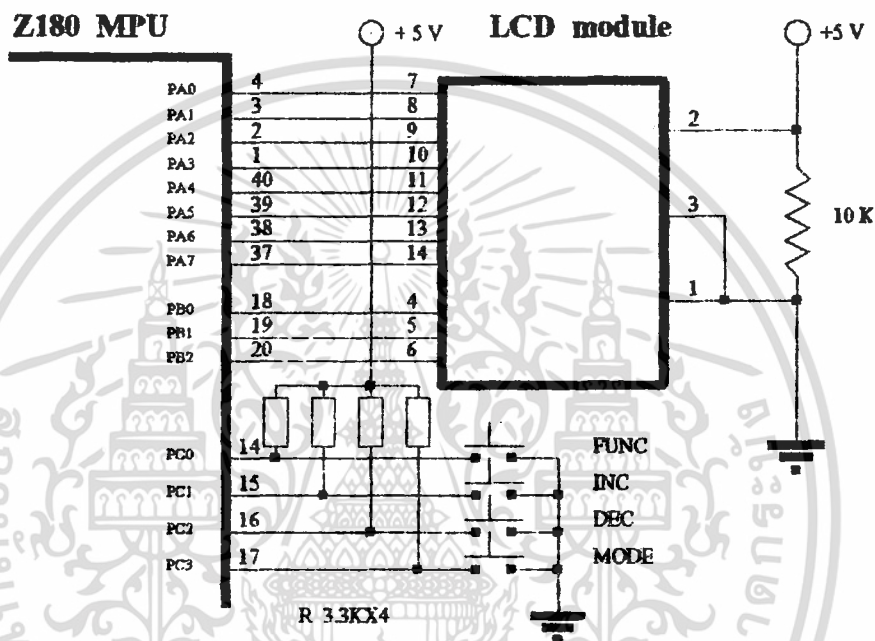
### 5.2.3 วงจรคีย์บอร์ด และ จอมแสดงผล LCD

โดยทั่วไปแล้ว ระบบที่ใช้ไมโครคอมพิวเตอร์ในการควบคุม จะต้องมีส่วนที่ใช้สำหรับติดต่อกับผู้ใช้ อันประกอบด้วยคีย์บอร์ดและจอมแสดงผล สำหรับระบบแปลงระดับความถี่เสียงไปเป็นข้อมูลดิจิทัลของเรา ได้ออกแบบให้ใช้จอ LCD ในการแสดงผลของระดับความถี่เสียงที่ตรวจจับได้ และใช้ในการแสดงค่าของพารามิเตอร์ในฟังก์ชันต่างๆที่สามารถแก้ไขได้ โดยผ่านทางคีย์บอร์ดทั้ง 4 คีย์ ซึ่งมีหน้าที่แตกต่างกันไป ดังนี้

1. MODE KEY (MODE) ใช้สำหรับเลือกจะให้ทำงานในโหมด PERFORM คือ การแปลงระดับความถี่เสียงไปเป็นข้อมูลดิจิทัล หรือเข้าสู่โหมด EDIT เพื่อแก้ไขค่าพารามิเตอร์
2. FUNCTION KEY (FUNC) ใช้สำหรับ กำหนดหน้าที่การทำงานให้กับคีย์ INC และ คีย์ DECว่าจะให้แก้ไขค่าพารามิเตอร์ตัวใด คีย์ FUNC นี้ จะทำงานในลักษณะวนรอบ และ จะทำงานหลังจากมีการกดคีย์ MODE แล้ว เท่านั้น ส่วนค่าพารามิเตอร์ที่สามารถแก้ไขได้ ประกอบด้วย
  - TRANSPOSE ใช้กำหนดจำนวนครั้งเสียงของตัวโน้ตที่ต้องการเลื่อนไปในการเล่น
  - CHANNEL ใช้กำหนดหมายเลขช่องในการส่ง
  - VELOCITY ใช้กำหนดระดับความดัง ( อ่านรายละเอียดตอนที่ 2 )
  - PITCH BEND ใช้สำหรับกำหนดว่า จะให้ส่งข้อมูล PITCH BEND หรือไม่
3. INCREMENT KEY (INC) ใช้สำหรับเพิ่มค่าของข้อมูลขึ้นทีละหนึ่ง
4. DECREMENT KEY (DEC) ใช้สำหรับลดค่าของข้อมูลลงทีละหนึ่ง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สำหรับวงจรมีบอร์ดและวงจรมแสดงผล LCD รวมทั้งวงจรมีขั้วกระแสนั้น เราได้ทำการประกอบลงไว้บนแผ่นวงจรมีบอร์ดเดียวกันโดยใช้วิธี wire wrap วงจรของส่วนที่มีบอร์ดและจอแสดงผล ซึ่งจะต้องทำการเชื่อมต่อกับพอร์ต 8255 ของ CP-180 ได้แสดงไว้ในรูปที่ 5.5



รูปที่ 5.5 แสดงการเชื่อมต่อวงจรมีบอร์ดและแสดงผล LCD ที่พอร์ต 8255 ของ CP-180

### 5.3 โปรแกรมควบคุมการทำงานของระบบ

โปรแกรมที่ใช้ควบคุมการทำงานของ เครื่องแปลงระดับความถี่เสียง ไปเป็น ข้อมูลดิจิทัลในโครงงานพิเศษนี้ จะแบ่งได้เป็น 2 ส่วนใหญ่ๆ คือ

1. ส่วนของโปรแกรมควบคุมการทำงานหลัก
2. ส่วนของโปรแกรมบริการอินเทอร์พรีต

ก่อนที่เราจะกล่าวถึง โปรแกรมทั้งสอง เราจะขออธิบายถึงงานต่างๆ ที่ระบบควบคุมและประมวลผล หรือ Z180 MPU จะต้องกระทำทั้งหมด เพื่อให้สามารถเข้าใจส่วน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ของโปรแกรมได้ง่ายขึ้น งานต่างๆเหล่านี้จะเกี่ยวข้องกับระบบฮาร์ดแวร์โดยตรง ดังนั้น เราจึงต้องทำการอ้างอิงถึงรายละเอียดต่างๆในบทที่ 4 ด้วย

### 5.3.1 งานของระบบควบคุมและประมวลผล

ถ้ากล่าวโดยสรุปแล้ว งานของส่วนควบคุมและประมวลผลก็คือ การตรวจสอบว่า สัญญาณที่เข้ามาที่อินพุทของระบบมีระดับความถี่เสียงตรงกันไหมทางคนตรีตัวใด แล้วทำการส่งข้อมูลที่มีระดับความถี่เสียงของ โน้ตตัวนั้น ออกไปควบคุมเครื่องดนตรีไฟฟ้าในระบบมิดีภายนอก ในระบบการทำงานจริงนั้น เราได้ออกแบบให้หน่วยอุปกรณ์ภายในของ Z180 MPU มาใช้งานให้มากที่สุด เพื่อที่จะลดความยุ่งยากในการออกแบบฮาร์ดแวร์ ดังนั้น งานต่างๆของระบบจะถูกแยกย่อย ไปกระทำโดยอุปกรณ์ภายในแต่ละตัว ตามความเหมาะสม และอุปกรณ์ทั้งหมดสามารถจะทำงานสอดคล้องประสานกันได้ด้วยการอินเทอร์รัพท์ ซึ่ง Z180 MPU นั้นได้ถูกออกแบบมาให้เป็นระบบที่สามารถตอบสนองการอินเทอร์รัพท์ได้อย่างมีประสิทธิภาพ

เมื่อมีสัญญาณความถี่เสียงเข้ามาที่วงจร Pitch detector มันจะทำการส่งขบวนพัลส์ลบ ที่มีระยะห่างระหว่างพัลส์ลบแต่ละลูก เท่ากับคาบ เวลาของสัญญาณที่อินพุทของตัวมัน พัลส์ลบเหล่านี้จะทำหน้าที่ในการขออินเทอร์รัพท์การทำงานผ่านทางขา TINTO ของ Z180 MPU ระบบจะทำการตอบสนองด้วยการตรวจสอบว่าพัลส์ลบเป็นพัลส์ลบแรกหรือเปล่า ถ้าเป็นพัลส์ลบแรกระบบจะควบคุมให้ PRT 1 (Programmable Reload Timer) เริ่มทำการนับถอยหลัง จากนั้นระบบก็จะกลับไปทำงานเดิมที่ทำค้างอยู่ก่อนหน้าที่จะมีการอินเทอร์รัพท์ แต่ถ้าสัญญาณขออินเทอร์รัพท์นั้น ไม่ใช่พัลส์ลบแรก ระบบจะตอบสนองด้วยการอ่านค่าเวลาในขณะนั้นจาก PRT 1 เพื่อส่งไปทำการคำนวณหาระยะเวลาระหว่างพัลส์ทั้งสองลูกที่เข้ามาติดกันนั้น โดยการลบค่าเวลาปัจจุบันออกจากค่าตั้งต้น คาบเวลาที่ได้ออกนำไปเปรียบเทียบกับตารางข้อมูลของคาบเวลาของ ระดับความถี่เสียงกับรหัสข้อมูลทางมิดี ซึ่งได้ถูกบันทึกไว้ล่วงหน้าแล้ว ในหน่วยความจำของระบบประมวลผล หากคาบเวลาที่คำนวณได้ตรงกับค่าของโน้ตตัวใด รหัสข้อมูลมิดีของโน้ตตัวนั้นจะถูกส่ง ไปให้กับ ASCII 0 (Asynchronous serial communication

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Interface ) เพื่อแปลงข้อมูลให้สามารถส่งในมาตรฐานมิตีได้ และถูกส่งออกไปทางขา TXAO นอกจากนี้ ข้อมูลที่ได้จากการเปรียบเทียบจะถูกแปลงให้อยู่ในรูปของสัญญาณซึ่งมีระดับความถี่เสียงของโน้ตแต่ละตัว เพื่อส่งไปแสดงผลบนจอ LCD อีกที

ระบบการแปลงระดับความถี่เสียง ไปเป็นข้อมูลมิตีของเรานั้น ได้ถูกออกแบบให้สามารถแก้ไขค่าของพารามิเตอร์บางตัวได้ โดยการเข้าสู่โหมดการแก้ไข ซึ่งถูกควบคุมโดยคีย์ MODE ผ่านทางอินเทอร์พอร์ท INT1 เมื่อมีการกดคีย์ MODE ระบบจะตอบสนองโดยการออกจากโหมดการทำงานปกติ เพื่อเข้าสู่โหมดการแก้ไข และจะกลับคืนสู่โหมดทำงานปกติได้ก็ต่อเมื่อมีการกดคีย์ MODE ซ้ำอีกครั้ง

จากที่กล่าวมานั้น เป็นงานเพียง 1 รอบการทำงาน โดยถ้านับจากการตรวจจับพัลส์สัญญาณแรกของรอบการทำงานแรก จนถึงการตรวจจับพัลส์สัญญาณแรกของรอบการทำงานถัดมา จะใช้เวลาประมาณ 10 มิลลิวินาที ซึ่งเวลานี้จะถูกควบคุมโดย PRT 0 ผ่านทางอินเทอร์พอร์ทของ PRT 0 หมายความว่า PRT 0 จะส่งสัญญาณอินเทอร์พอร์ท ไปบอกระบบให้เริ่มต้นรอบการทำงานใหม่ทุกๆ 10 มิลลิวินาที

เมื่อทราบระบบการทำงานของ การแปลงระดับความถี่เสียง ไปเป็นข้อมูลมิตีแล้ว ในหัวข้อต่อไป เราจะกล่าวถึงส่วนของโปรแกรมควบคุม ซึ่งเขียนขึ้นเพื่อควบคุมการทำงานในส่วนต่างๆ ของระบบ

### 5.3.2 โปรแกรมควบคุมการทำงานหลัก ( Main program )

ในการออกแบบระบบการทำงานของ เครื่องแปลงระดับความถี่เสียง ไปเป็นข้อมูลมิตี เราได้ทำการกำหนดให้งานหลักของระบบคือ การนำค่าคาบเวลาที่อ่านได้ไปเปลี่ยนให้เป็นข้อมูลของโน้ตทางมิตี เพื่อส่งออกไปควบคุมเครื่องดนตรีในระบบมิตีภายนอก และรวมทั้งการแสดงผลทางจอ LCD ด้วย เมื่ออยู่ในโหมดการทำงานปกติ (Perform) ระบบของเราจะทำงานตามโปรแกรมหลักที่วางไว้คือ นำค่า PITCH ซึ่งอยู่ในรูปของคาบเวลา ไปทำการเปรียบเทียบกับข้อมูลของคาบเวลาของระดับความถี่เสียงแต่ละตัวที่บันทึกไว้แล้วล่วงหน้า โดยที่ข้อมูลแต่ละตัวจะมีรหัสมิตีแต่ละตัวกำกับอยู่ เมื่อพบข้อมูลที่ค่าใกล้เคียงกับค่า PITCH ที่ได้จากการคำนวณ ระบบจะทำการส่งรหัสมิตีของข้อมูลตัวนั้น

พร้อมกับข้อมูล PITCH BEND ซึ่งหาได้จากผลต่างระหว่างค่า PITCH ที่อ่านได้กับข้อมูล ตัวที่ถูกเลือก และเนื่องจากเครื่องแปลงระดับความถี่เสียงของเรามีฟังก์ชันในการเลื่อน ค่าตัวโน้ตเพื่อประโยชน์ในการเปลี่ยนคีย์ของดนตรี ในโปรแกรมหลักจึงจะมีการบวกหรือลบค่าของตัวโน้ตด้วยจำนวนของครึ่งเสียงที่ต้องการให้เลื่อนไป ซึ่งสามารถกำหนดได้ในโหมดของการแก้ไข (Edit) โดยรหัสสมิติที่ได้มา จะถูกตรวจสอบก่อนส่งออกไปว่า ระดับความถี่เสียงอยู่ในช่วงที่สามารถเล่นได้หรือไม่ ซึ่งถ้าหากไม่อยู่ในช่วงดังกล่าว ระบบจะไม่ทำการแสดงผลของตัวโน้ตตัวนั้น ขั้นตอนการทำงานของโปรแกรมเราได้แสดงไว้เป็นไฟล์ชาร์ทในรูปที่ 5.6 ถึง 5.8 สำหรับโปรแกรมที่ใช้งานจริงเราได้รวบรวมไว้ในภาคผนวกท้ายเล่ม

### 5.3.3 โปรแกรมบริการอินเทอร์รัท ( Service Interrupt Routine )

สำหรับโครงงานนี้ โปรแกรมบริการอินเทอร์รัท เป็นส่วนที่มีความสำคัญอย่างมาก เพราะระบบมีงานมากมายที่ต้องทำไปพร้อมๆกัน การอินเทอร์รัทจึงเป็นทางออกที่ดีในการออกแบบระบบ เพื่อให้ทุกส่วนสามารถทำงานได้อย่างอิสระแยกจากกัน ทำให้เวลารวมที่ใช้ในการทำงานลดลง โปรแกรมบริการอินเทอร์รัทของเรา มีหน้าที่ในการตอบสนองการขออินเทอร์รัทจากฮาร์ดแวร์ ทั้งภายใน และ ภายนอกของส่วนควบคุมและประมวลผล ซึ่งมีระดับความสำคัญแตกต่างกันไป นั่นหมายความว่าถ้ามีการอินเทอร์รัทเกิดขึ้นพร้อมกัน ระบบจะทำการตอบสนองกับอุปกรณ์ที่มีความสำคัญสูงกว่า ก่อน โดยระดับความสำคัญนี้จะถูกกำหนดโดย Z180 MPU

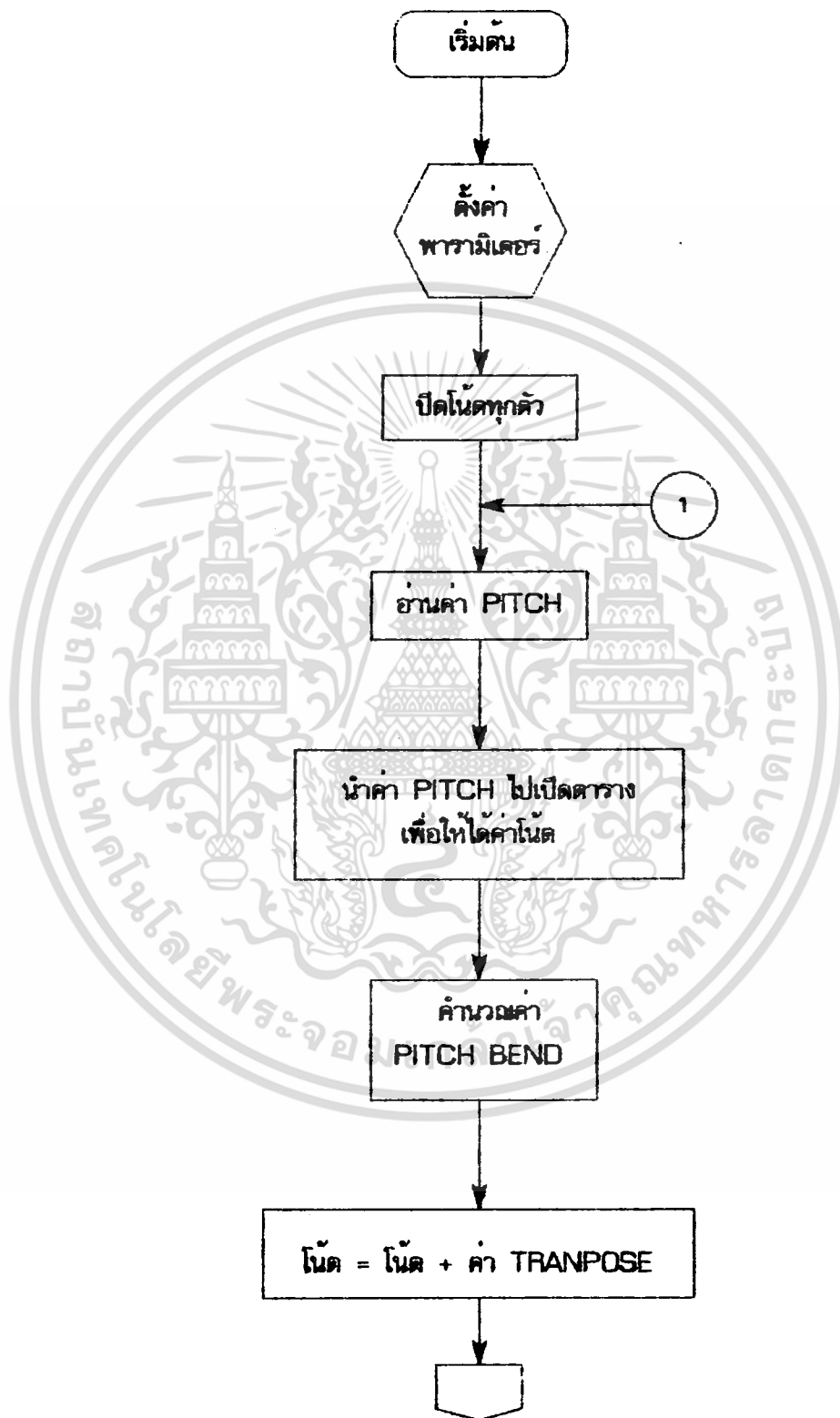
เราสามารถแบ่ง โปรแกรมบริการอินเทอร์รัททางฮาร์ดแวร์ของระบบ ได้เป็น 3 ส่วน คือ

1. โปรแกรมบริการอินเทอร์รัทของ INTO เป็นโปรแกรมที่ตอบสนองการขออินเทอร์รัทจากวงจร PITCH DETECTOR ซึ่งต่อเข้ากับขา INTO ของ Z180 MPU โปรแกรมนี้จะทำการควบคุมการจับเวลาของระยะห่างระหว่างสัญญาณอินเทอร์รัท 2 ลูก ที่เข้ามาติดกัน ที่ขา INTO โดยควบคุมการเริ่มต้นและการหยุดพักของ PRT 1 พร้อมทั้งส่งข้อมูลที่อ่านได้จาก PRT 1 ไปเก็บไว้ที่พเพอร์เพื่อรอการเรียกใช้จากโปรแกรมหลัก

2. โปรแกรมบริการอินเทอร์พรีตของ INT1 เป็นส่วนที่ตอบสนองการกดคีย์ MODE เพื่อเข้าหรือออกจากโหมดการแก้ไข เนื่องจากคีย์นี้ผูกต่อเข้ากับ INT1 ของ Z180 MPU ด้วย โปรแกรมส่วนนี้จะทำการจัดการการทำงานของคีย์ต่างๆที่ใช้ในการแก้ไขพารามิเตอร์ต่างๆของระบบ โดยถ้าระบบตรวจพบอินเทอร์พรีต INT1 ครั้งแรก จะทำการเข้าสู่โหมดการแก้ไข และจะกลับสู่โหมดการทำงานปกติเมื่อตรวจพบอินเทอร์พรีตนี้อีกครั้ง

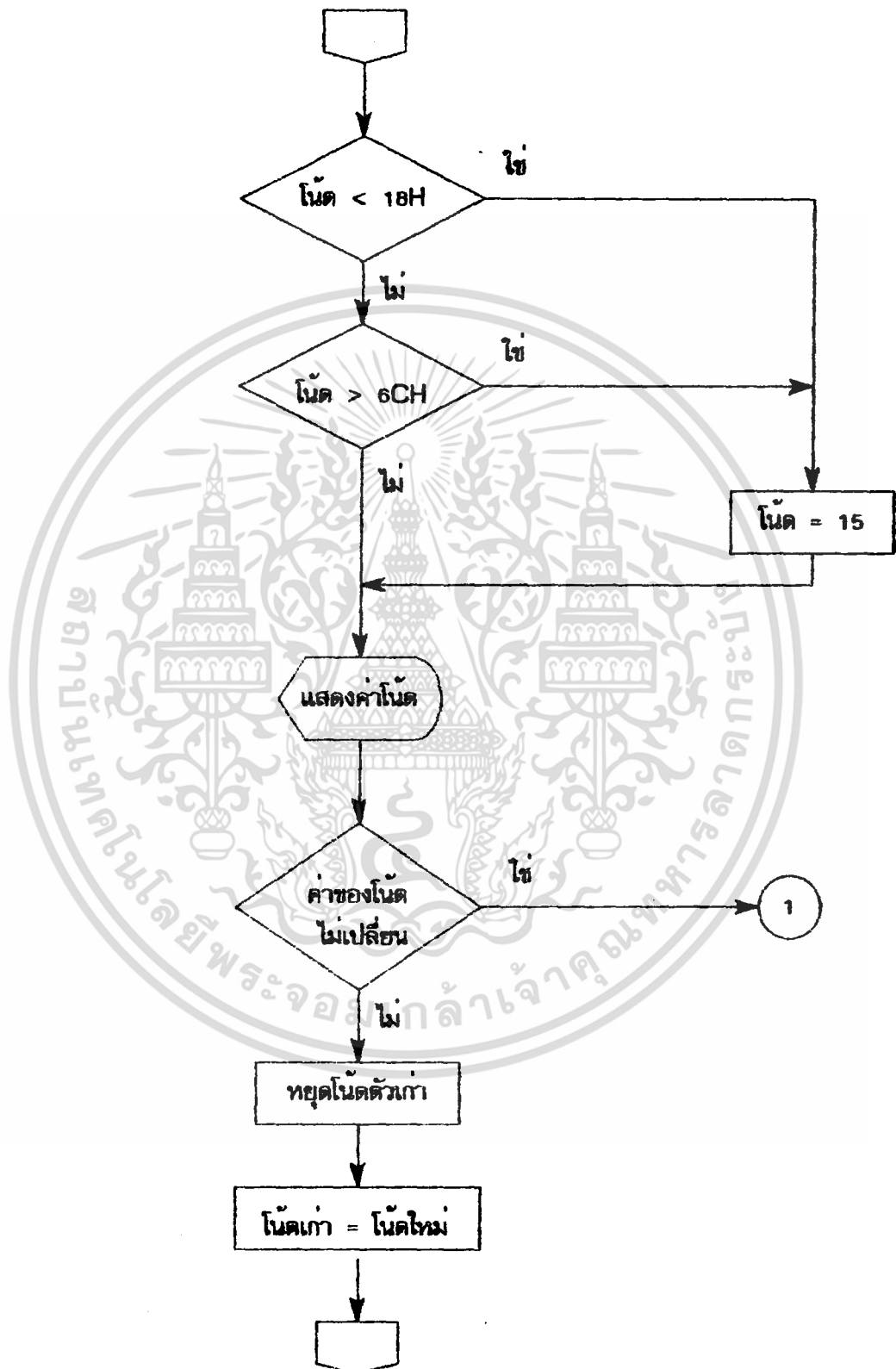
3. โปรแกรมบริการอินเทอร์พรีตของ PRT 0 เป็นส่วนที่ตอบสนองการอิน-  
-เทอร์พรีตของ PRT 0 ซึ่งทำหน้าที่ควบคุมเวลาใน 1 รอบการทำงานของกรมแปลง  
ระดับความถี่เสียง ไปเป็นข้อมูลดิจิทัล เมื่อ PRT 0 ทำการนับเวลาออกถึงจนครบ 10  
มิลลิวินาทีแล้ว มันจะส่งสัญญาณอินเทอร์พรีต ซึ่งเป็นการอินเทอร์พรีตภายในไปยัง CPU ของ  
Z180 MPU ระบบจะตอบสนองโดยการรีเซ็ตระบบใหม่ เพื่อรอการทำงานในรอบต่อไป  
ในระบบนี้ที่มัน ได้กำหนดให้มีการตรวจสอบการผิดพลาดด้วยการส่งข้อความ  
ที่เรียกว่า Active Sense ออกไปจากตัวควบคุม ซึ่งจะส่งหนึ่งครั้งทุกๆ ประมาณ 300  
มิลลิวินาที เราจึงกำหนดให้โปรแกรมบริการอินเทอร์พรีต PRT0 เป็นตัวควบคุมเวลาใน  
การส่ง Active Sense ด้วย

อันที่จริงในระบบเรายังใช้อุปกรณ์อื่นๆในการทำงานร่วมอยู่ด้วย อันได้แก่  
PRT 1 และ ASCII 1 เพียงแต่เราไม่ได้ใช้มันในการทำงานแบบอินเทอร์พรีต ดังนั้นจึง  
ไม่มีโปรแกรมบริการทางอินเทอร์พรีต สำหรับขั้นตอนการทำงานของ โปรแกรมบริการ  
อินเทอร์พรีตทั้งสาม เราได้แสดงไว้ในรูปที่ 5.9 ถึง 5.14



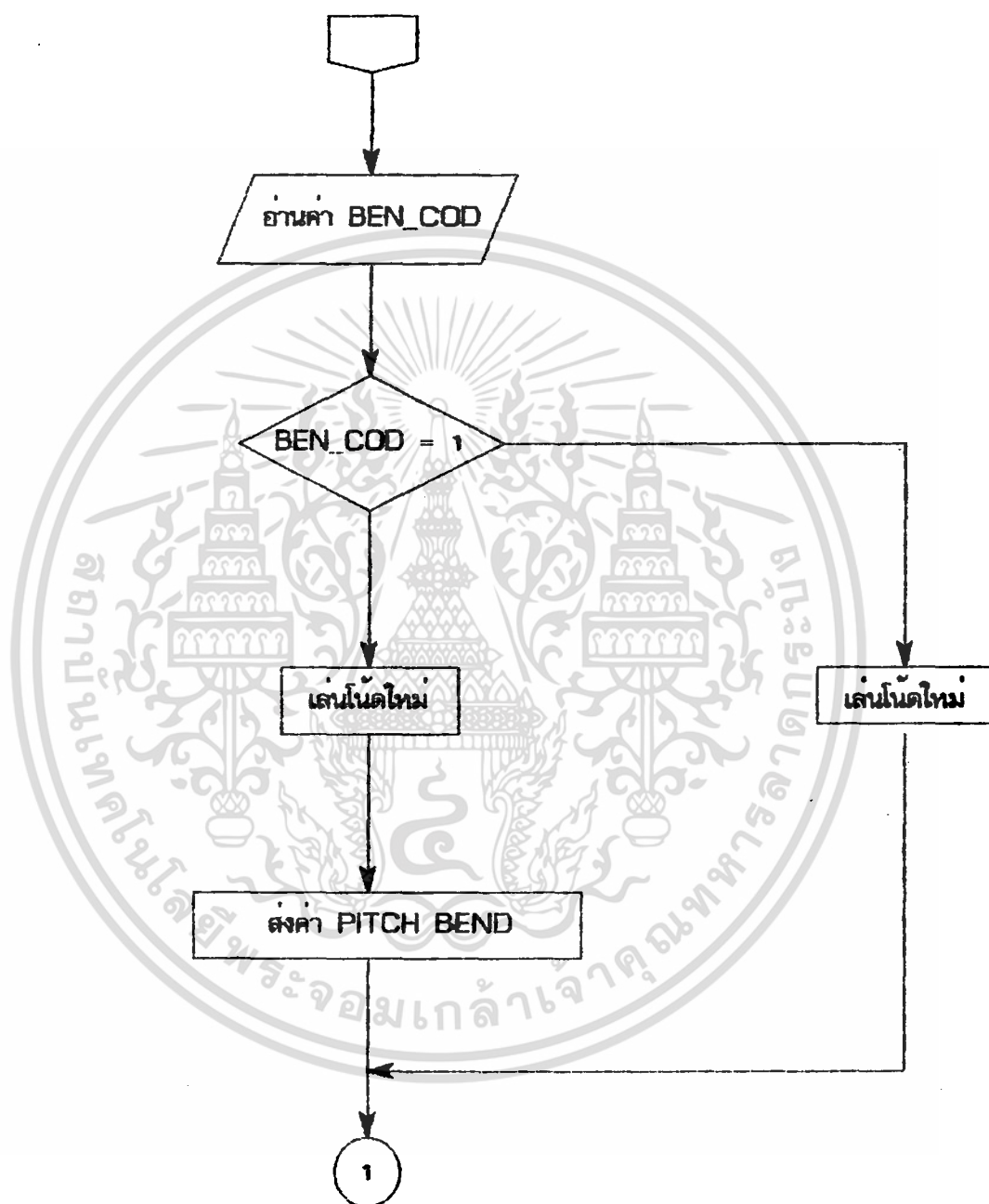
รูปที่ 5.6 แสดงการทำงานของโปรแกรมหลัก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



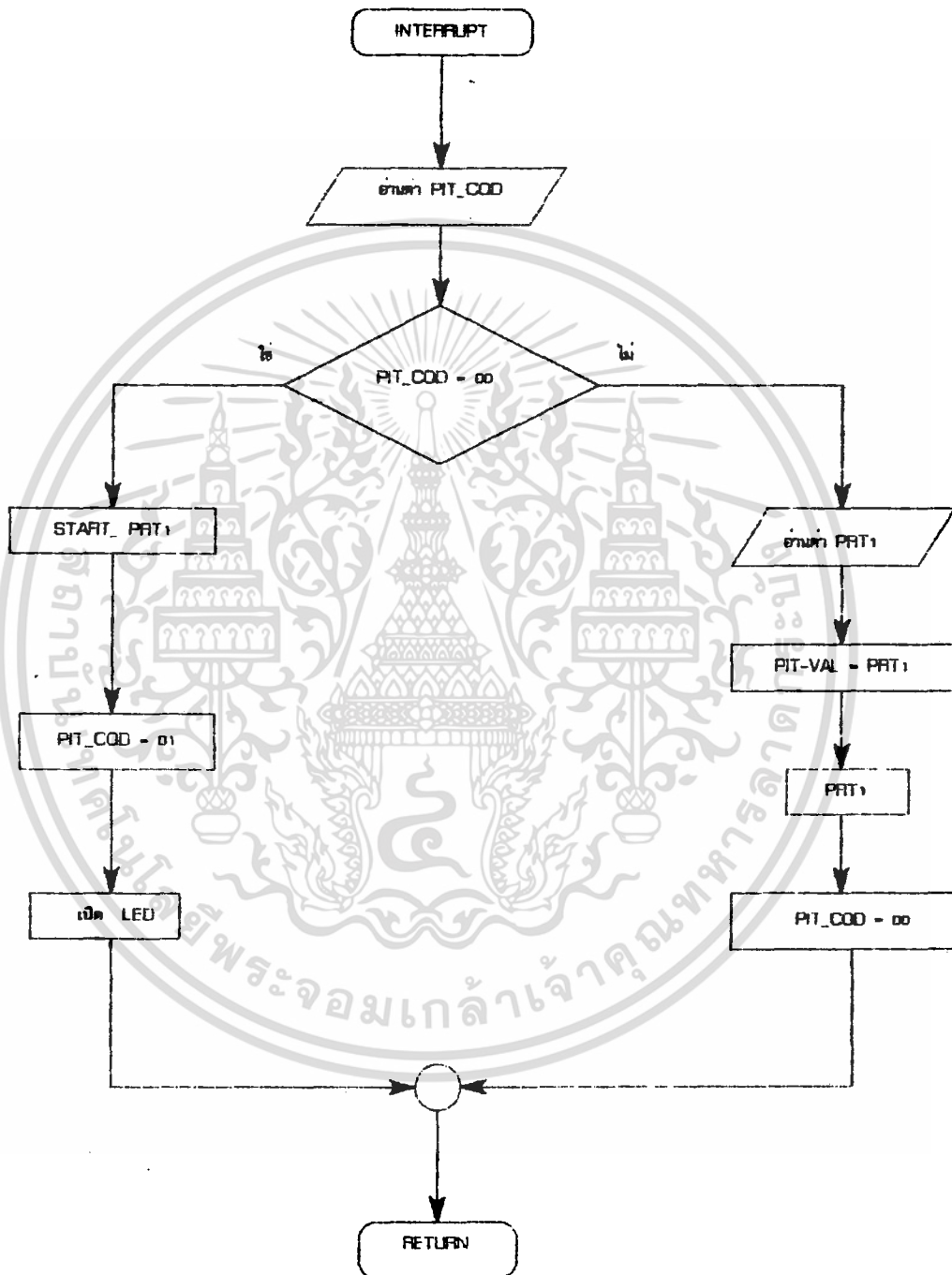
รูปที่ 5.7 แสดงการทำงานของโปรแกรมหาค่า (ต่อ)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



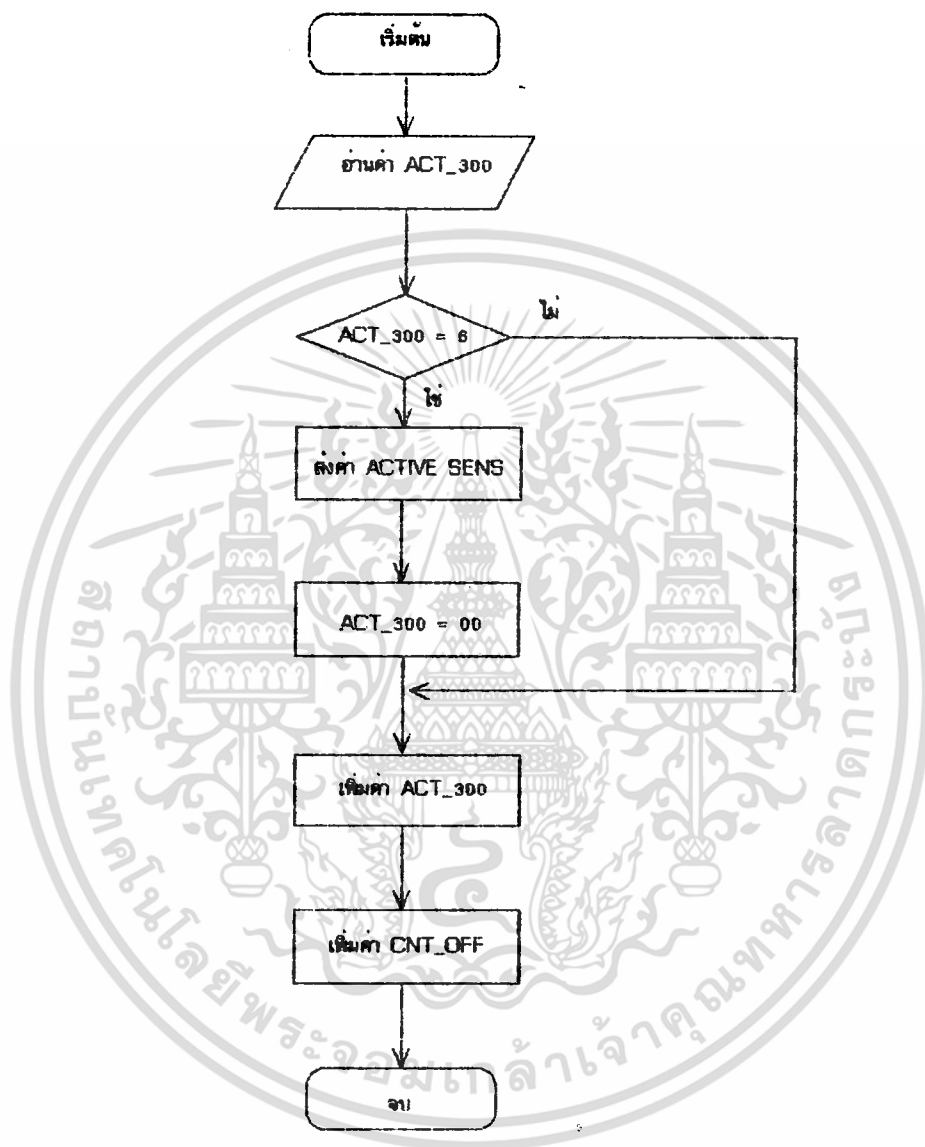
รูปที่ 5.8 แสดงการทำงานของโปรแกรมหลัก (ต่อ)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



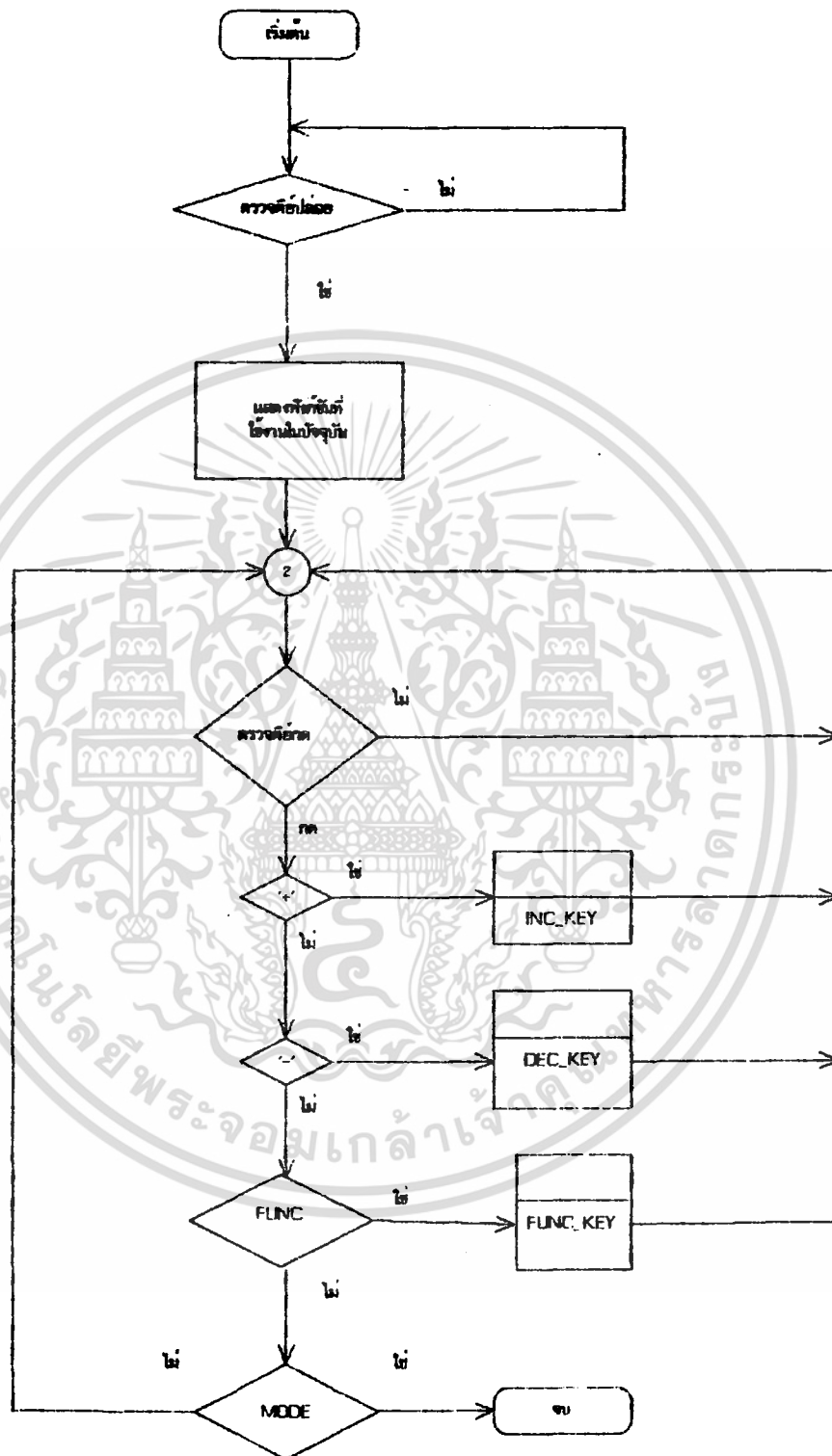
รูปที่ 5.9 โปรแกรมให้บริการอินเทอร์รัพท์ของ INTO

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



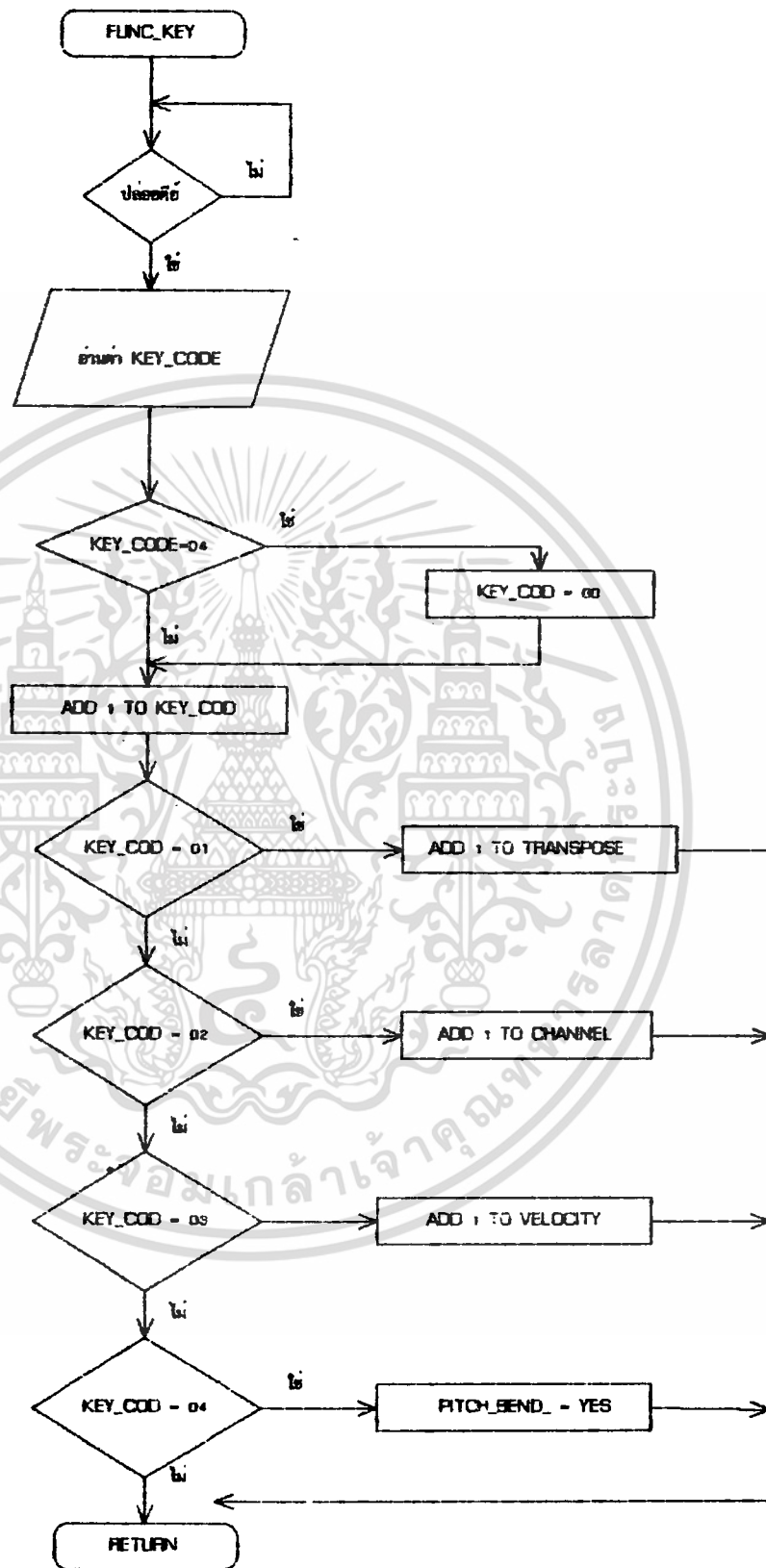
**รูปที่ 5.10 โปรแกรมให้บริการอินเทอร์ล็อกของ PRTO**

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



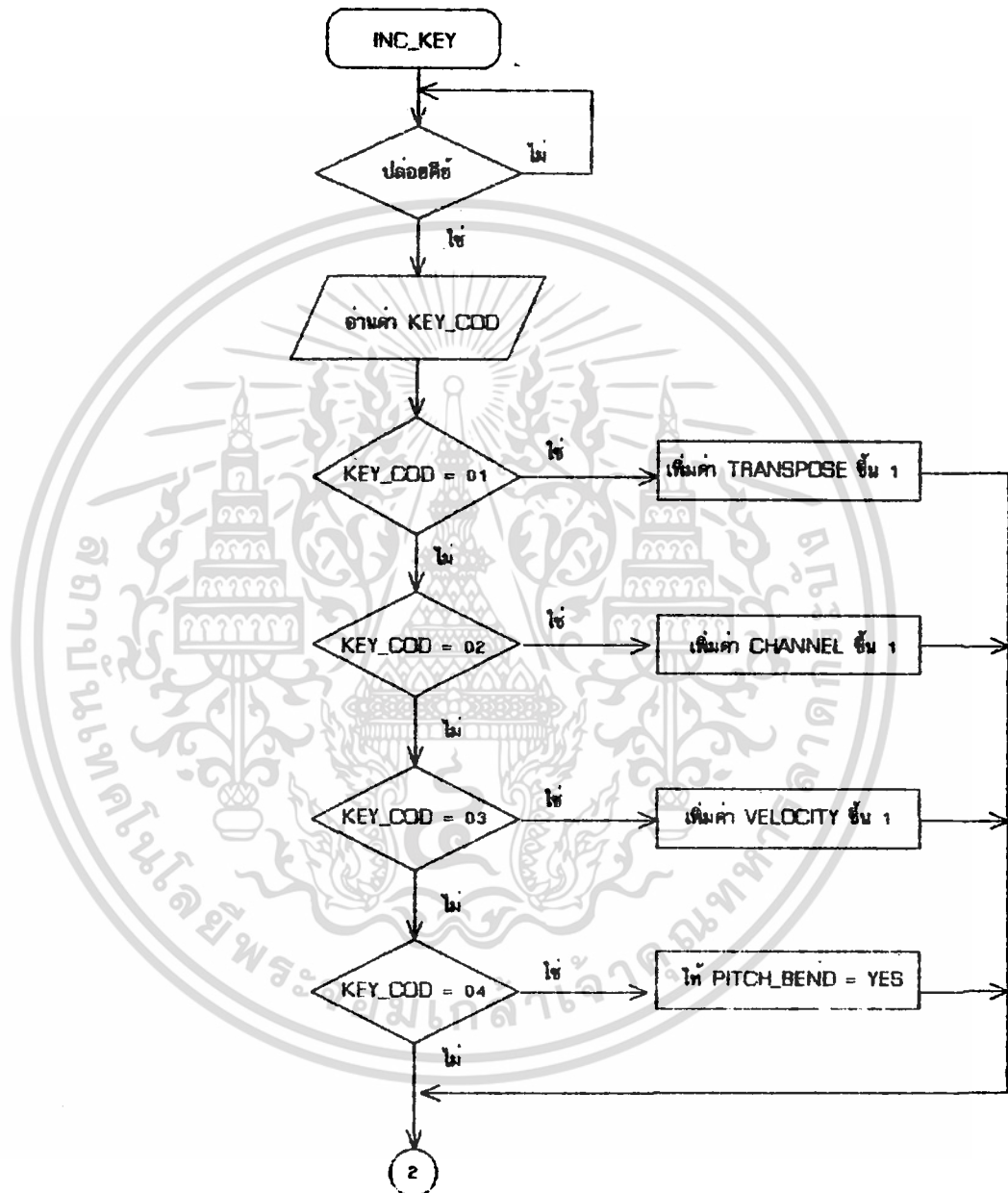
รูปที่ 5.11 โปรแกรมให้บริการอินเทอร์พรีตของ INT1 (MOD\_KEY)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



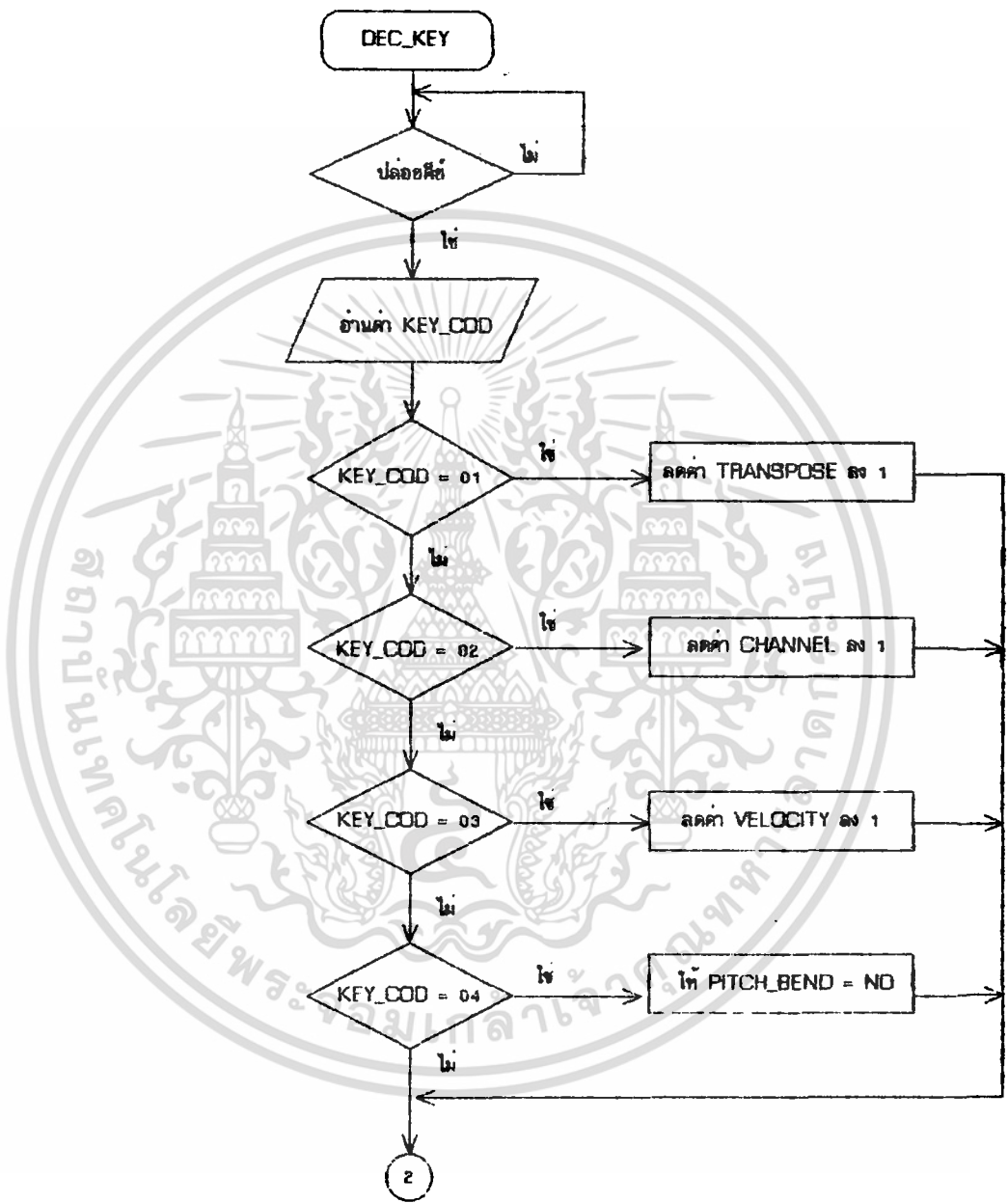
รูปที่ 5.12 โปรแกรมให้บริการอินเทอร์เน็ตของ INT1 (FUNC\_KEY)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 5.13 โปรแกรมให้บริการอินเทอร์คัทของ INT1 (INC\_KEY)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 5.14 โปรแกรมให้บริการอินเทอร์เน็ต INT1 (DEC\_KEY)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## บทที่ 6

### การใช้งานและข้อเสนอแนะ

เครื่องแปลงระดับความถี่เสียง ไปเป็นข้อมูลทางมิติหรือ Pitch-to-MIDI converter ซึ่งสร้างขึ้นจากแนวความคิดทางฟิสิกส์ ผสมกับเทคโนโลยีทางอิเล็กทรอนิกส์ และไมโครคอมพิวเตอร์ได้บรรลุเป้าหมายในขั้นต้น นั่นคือ การพิสูจน์ให้เห็นถึงกระบวนการแปลงรูปสัญญาณอีกวิธีหนึ่ง อันเป็นวิธีที่เราได้พัฒนาขึ้นมาใช้สำหรับงานด้านดนตรี ซึ่งต้องการการตอบสนองที่ค่อนข้างจะทันทีทันใด

#### 6.1 การใช้งานเครื่องแปลงระดับความถี่เสียง ไปเป็นข้อมูลมิติ

เครื่องแปลงระดับความถี่เสียง ไปเป็นข้อมูลมิตินี้ สามารถทำงานได้ 2 โหมด

##### 6.1.1 โหมดการทำงาน (Perform mode)

เมื่อเราเริ่มต้นจ่ายไฟให้กับเครื่อง และหลังจากที่เครื่องได้ทำการเซตอัพ (set up) พารามิเตอร์ต่างๆของตัวมันเองเสร็จเรียบร้อยแล้ว มันจะเข้าสู่การทำงานในโหมดนี้โดยอัตโนมัติ งานหลักของการทำงานในโหมดนี้ คือ การรับสัญญาณเสียงที่เข้ามาทางไมโครโฟนคอนเดนเซอร์ แล้วทำการเปลี่ยนแปลงสัญญาณนั้นให้อยู่ในรูปของข้อมูลทางมิติ แล้วส่งออกทางพอร์ต MIDI เพื่อใช้ในการควบคุมการเล่นของเครื่องดนตรีไฟฟ้าภายนอก พร้อมทั้งหน้าจอแสดงผล LCD จะแสดงตัวอักษรซึ่งบอกถึงระดับความถี่เสียงทางดนตรีของสัญญาณนั้น

ลักษณะภายนอกของ Pitch-to-MIDI Converter ของเราได้แสดงไว้ในรูปที่ 6.1 จะเห็นว่าบนหน้าปัดจะประกอบด้วย จอแสดงผล LCD แบบเรืองแสง อยู่ทางด้านซ้ายมือ ถัดมาจะเป็น LED 2 ดวงโดยที่ดวงล่างจะเป็นสีแดง มีอักษรกำกับว่า 'ON' ใช้เพื่อแสดงสถานะทำงานของระบบทั้งหมด ส่วน LED ดวงบนจะเป็นสีเหลือง มีอักษรกำกับว่า 'RUN' LED ดวงนี้จะติดก็ต่อเมื่อมีสัญญาณเสียงเข้ามาที่ไมโครโฟน

และ สุดท้ายจะเป็นกลุ่มของคีย์บอร์ดจำนวน 4 คีย์ โดยเรียงตามลำดับจากซ้ายไปขวาดังนี้ คือ MODE, FUNC, DEC และ INC สำหรับความหมายและหน้าที่ของคีย์เหล่านี้ได้กล่าวไว้แล้วในหัวข้อที่ 5

**รูปที่ 6.1 แสดงลักษณะภายนอกของ เครื่องแปลงระดับความถี่เสียงไปเป็นข้อมูลมิตี**

ในโหมดนี้ จะสามารถทำการแปลงระดับความถี่เสียงได้ตั้งแต่ตัวโน้ต C1 (32.70 เฮิร์ต) ไปจนถึง โน้ตตัว C6 (1046 เฮิร์ต) ข้อมูลทางมิตีที่ส่งออกมาจะประกอบด้วย ข้อมูลของตัวโน้ต และข้อมูล pitch bend ของตัวโน้ตนั้น เราสามารถสรุปขั้นตอนการใช้งานโหมดการทำงานนี้ได้ดังนี้ คือ

1. คอ ไมโคร โฟนคอนเตนเซอร์ เข้ากับช่องที่เขียนว่า MIC ทางด้านข้างของตัวเครื่อง
2. เปิดสวิตช์ POWER ของเครื่อง LED สีแดง (ON) จะติด โดยที่ทางด้านซ้ายของจอ LCD จะปรากฏคำว่า PERFORM ซึ่งแสดงว่าขณะนี้กำลังอยู่ในโหมดการทำงาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3. ถ้ามีสัญญาณเสียงเข้ามาที่คอนเซนเซอร์ไมโครโฟน LED สีเหลือง (RUN) จะกระพริบ ทางด้านขวาของจอ LCD จะแสดงตัวโน้ตตัวที่ตรวจจับได้ในขณะนั้น
4. ทำการเชื่อมต่อ MIDI OUT ของเครื่อง เข้ากับ MIDI IN ของเครื่องดนตรีไฟฟ้าตัวอื่นๆ จะมีการเล่นโน้ตตัวที่มีระดับความถี่เสียงตรงกับระดับความถี่ของสัญญาณเสียงที่ตรวจจับได้บนเครื่องดนตรีนั้นๆ



รูปที่ 6.2 แสดงการเชื่อมต่อเข้ากับมิไรเซอร์ในระบบนี้

#### 6.1.2 โหมดแก้ไข (Edit mode).

โหมดนี้จะเป็นการใช้คีย์บอร์ดทั้ง 4 คีย์ เพื่อเข้าไปแก้ไขค่าพารามิเตอร์บางตัวที่เกี่ยวข้องกับข้อมูลนี้ เมื่อ Pitch-to-MIDI Converter ทำงานในโหมดนี้ มันจะหยุดทำการแปลงข้อมูลเอาไว้ชั่วคราว เราสามารถควบคุมการเข้าออกโหมดนี้ได้โดยการกดคีย์ MODE ในโหมดแก้ไขนี้เราสามารถแก้ไขค่าของพารามิเตอร์ได้ 4 ตัว อัน

เอกสารนี้เป็นเอกสารลิขสิทธิ์ของ บริษัท ออริจินัล เทคโนโลยี จำกัด การนำเอกสารนี้ไปเผยแพร่โดยไม่ได้รับอนุญาตถือว่าผิดกฎหมาย การนำเอกสารนี้ไปใช้โดยไม่ได้รับอนุญาตถือว่าผิดกฎหมาย

ประกอบด้วยค่า TRANSPOSE, CHANNEL, VELOCITY และ PITCH BEND ON-OFF ซึ่งเราได้กล่าวไว้แล้วในหัวข้อที่ 5 เราจะสรุปการใช้งานโหมดนี้เป็นขั้นตอนต่างๆดังนี้ คือ

1. เมื่อจะเข้าสู่โหมดแก้ไข ให้กดคีย์ MODE 1 ครั้ง ระบบจะออกจากโหมดการทำงานเข้าสู่โหมดแก้ไข บนจอ LCD จะแสดงคำว่า EDIT ขึ้นมาแทนคำว่า PERFORM
2. กดคีย์ FUNC 1 ครั้ง เพื่อเข้าไปแก้ไขค่า TRANSPOSE, 2 ครั้ง เพื่อเข้าไปแก้ไขค่า CHANNEL, 3 ครั้ง เพื่อเข้าไปแก้ไขค่า VELOCITY และ 4 ครั้ง เพื่อเข้าไปแก้ไขค่า PITCH BEND ON-OFF ตามลำดับ ถ้ากดครั้งที่ 5 มันจะทำการวนไปเริ่มที่ TRANSPOSE ใหม่ และถ้ากดคีย์ FUNC ต่อไปอีก มันก็จะเปลี่ยน FUNCTION ในการแก้ไขหมุนวนไปตามลำดับ โดยจะแสดงชื่อของ FUNCTION ไว้หลังคำว่า EDIT
3. เมื่อต้องการลดหรือเพิ่มค่าของพารามิเตอร์ที่กล่าวไว้ในขั้นที่ 2 ก็ให้ใช้คีย์ INC หรือคีย์ DEC
4. ถ้าต้องการที่จะแก้ไขค่าพารามิเตอร์ตัวใดอีกก็ให้กดคีย์ FUNC เพื่อเลื่อน FUNCTION หมุนวนไปหาพารามิเตอร์ที่ต้องการ ซึ่งจะถูกแสดงไว้บนจอ LCD
5. เมื่อทำการแก้ไขค่าเสร็จเรียบร้อยแล้ว เราสามารถกลับเข้าสู่โหมดการทำงานได้ด้วยการกดคีย์ MODE อีกครั้ง

ที่กล่าวมาแล้ว เป็นขั้นตอนการใช้งาน Pitch-to-MIDI Converter ในทั้งสองโหมดโดยสังเขป ซึ่งจะเห็นได้ว่า การใช้งานไม่ยุ่งยากซับซ้อน สามารถเรียนรู้ได้โดยง่าย

## 6.2 คุณสมบัติของ เครื่องแปลงระดับความถี่เสียง ไปเป็นข้อมูลmidi

1. ควบคุมการทำงานด้วย Z80180 ซึ่งทำงานที่ความถี่ 10 MHz
2. อินพุตเป็น Phone Connector สำหรับไมโครโฟนชนิดคอนเดนเซอร์
3. มี MIDI OUT สำหรับเชื่อมต่อกับซินธิไซเซอร์หรือเครื่องดนตรีในระบบ มิดี 1 ช่อง
4. ใช้กับ Power Supply ขนาด 10 V 1 Amp ผ่านทาง DC Connector บนตัวเครื่อง
5. แสดงผลข้อความด้วยจอแสดงผล LCD แบบ 1 บรรทัด 16 ตัวอักษร
6. มี LED สำหรับแสดงสภาวะการทำงาน และสภาวะการแปลงข้อมูล
7. สามารถแก้ไขค่าพารามิเตอร์ทางมิดี ได้ 4 ตัว คือ Transpose , Channel , Velocity และ Pitch Bend on-off ผ่านทางคีย์ทั้ง 4 คีย์ บนตัวเครื่อง
8. มี Battery Back-up ข้อมูลในตัว

## 6.3 ข้อเสนอแนะ

Pitch-to-MIDI Converter ของเรานั้นยังถือว่าเป็นระบบพื้นฐาน ซึ่งสร้างขึ้นเพื่อทดสอบแนวความคิดในการแปลงข้อมูลทางความถี่ ไปเป็นข้อมูลในการควบคุมซินธิไซเซอร์ จึงยังมีอีกหลายส่วนที่สามารถพัฒนาให้ดีขึ้นได้ โดยอาศัยพื้นฐานจากปัญหาต่างๆที่เกิดขึ้นในระบบของเรา และบางสิ่งบางอย่างที่เราได้ลดความซับซ้อนลงในการขั้นตอนการออกแบบ เราพอสรุปเป็นข้อเสนอแนะ ได้ดังนี้

1. ทำให้มีขนาดเล็กลง สามารถพกติดตัวนักดนตรีได้ในขณะใช้งานแสดงบนเวที โดยการออกแบบลายวงจรใหม่ เนื่องจากเราใช้บอร์ดสำเร็จรูปจึงทำให้ระบบของเรามีขนาดใหญ่

เอกสารนี้เป็นเอกสารที่ 2. พัฒนาศักยภาพในการตรวจจับความถี่ของเสียงไป และการแสดงการคำนวณค่าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ผลบ่งชี้ขอให้ความสวยงาม

3. ออกแบบให้สามารถใช้กับไมโครโพรเซสเซอร์อื่น ๆ และเพิ่มส่วนของ  
เชื่อมต่อโดยตรงจากเอาต์พุตของเครื่องดนตรีไฟฟ้าอื่น

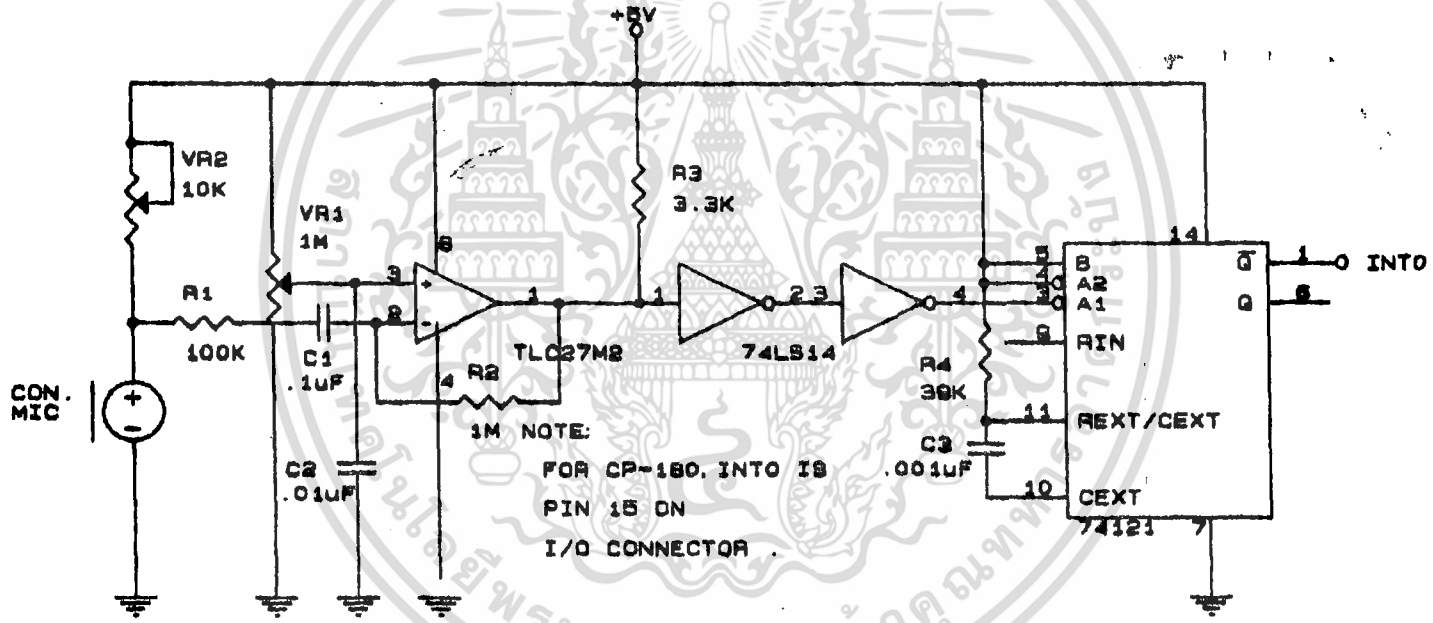


เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้





Size Document Number		REV
A		



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้





```

LD    (PIT_COD),A
JR    INTODONE
STOP: LD    A,11H
      OUTO (TCR),A
      INO  C,(TMDR1L)
      INO  B,(TMDR1H)
      LD  (PIT_VAL),BC
      LD  HL,TIMER1
      OUTO (TMDR1L),L
      OUTO (RLDR1L),L
      OUTO (TMDR1H),H
      OUTO (RLDR1H),H
      LD  A,00H
      LD  (PIT_COD),A
      LD  (CNT_OFF),A
      LD  A,00H
      OUTO (ITC),A
      JP  INTODONE
START: LD  A,13H
      OUTO (TCR),A
      LD  A,01H
      LD  (PIT_COD),A
      LD  BC,PORTC1 ;SIGNAL LED ON
      LD  A,80H
      OUT (C),A
INTODONE: POP HL
        POP BC
        POP AF
        EI
        RETI

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้











```

CHEK_6CH: LD  A,6CH
           CP  D
           JR  NC,SAVE_NTE
           LD  D,00H

```

```

;
;  SAVE NOTE
;

```

```

SAVE_NTE: LD  A,D
           LD  (NEW_NTE),A
           LD  B,A
           CALL WRT_NTE

```

```

;
;  COMPARE NEW NOTE & OLD NOTE
;

```

```

CP_NTE:   LD  A,(OLD_NTE)
           LD  B,A
           LD  A,(NEW_NTE)
           CP  B
           JR  Z,MAIN

```

```

NEW_OK:   LD  IX,OLD_NTE
           CALL STOP_NTE
           LD  A,(NEW_NTE)
           LD  (OLD_NTE),A
           LD  IX,NEW_NTE
           CALL PLAY_NTE
           JP  MAIN

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

;
; LCD DISPLAY SEGMENT
;

```

```

;
; WRITE 1 CHAR
;

```

```
WRT_1CH0: LD A,00H
```

```
CALL GOTO
```

```
JP WRT
```

```
WRT_1CH1: LD A,01H
```

```
CALL GOTO
```

```
JP WRT
```

```
WRT_1CH2: LD A,02H
```

```
CALL GOTO
```

```
JP WRT
```

```
WRT_1CH3: LD A,03H
```

```
CALL GOTO
```

```
JP WRT
```

```
WRT_1CH4: LD A,04H
```

```
CALL GOTO
```

```
JP WRT
```

```
WRT_1CH5: LD A,05H
```

```
CALL GOTO
```

```
JP WRT
```

```
WRT_1CH6: LD A,06H
```

```
CALL GOTO
```

```
JP WRT
```

```
WRT_1CH7: LD A,07H
```

```
CALL GOTO
```

```
JP WRT
```

```
WRT_1CH8: LD A,08H
```

```
CALL GOTO
```

```
JP WRT
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

WRT_1CH9: LD A,41H
           CALL GOTO
           JP WRT
WRT_1CHA: LD A,42H
           CALL GOTO
           JP WRT
WRT_1CHB: LD A,43H
           CALL GOTO
           JP WRT
WRT_1CHC: LD A,44H
           CALL GOTO
           JP WRT
WRT_1CHD: LD A,45H
           CALL GOTO
           JP WRT
WRT_1CHE: LD A,46H
           CALL GOTO
           JP WRT
WRT_1CHF: LD A,47H
           CALL GOTO
           JP WRT
;
; WRITE TEXT 3 CHAR
;
WRT_OCT: LD A,44H ; " 000 "
           CALL GOTO
           LD B,03H
           JP WRL
;
; INITIAL LCD DISPLAY
;
INITLCD: PUSH AF
          PUSH BC

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

LD BC,PORTB1
LD A,0000000B
OUT (C),A
LD BC,PORTA1
LD A,0011100B
OUT (C),A
CALL EPLUSE
CALL DLAYLCD
LD BC,PORTA1
LD A,0000110B
OUT (C),A
CALL EPLUSE
LD BC,PORTA1
LD A,0000011B
OUT (C),A
CALL EPLUSE
LD BC,PORTA1
LD A,0000001B
OUT (C),A
CALL EPLUSE
CALL DLAYLCD
POP BC
POP AF
RET

```

```

;
; WRITE LINE 16 CHAR
;

```

```

WRT_TXT: LD A,01H
CALL WRLINE
LD A,02H
CALL WRLINE
RET

```

```

WRLINE: CP 1

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

        JP    Z,WRL1
        CP    2
        JP    Z,WRL2
        RET
WRL1:   LD    A,00H                ;00 - 07
        CALL GOTO
        JP    WRLM
WRL2:   LD    A,40H                ;40 - 47
        CALL GOTO
WRLM:   LD    B,8                  ;8 char
WRL:    LD    D,(HL)
        PUSH BC
        CALL WRBYTE
        POP  BC
        INC  HL
        DJNZ WRL
        RET
;
;
;
;
WRT:    PUSH BC
        CALL WRBYTE
        POP  BC
        RET
;
;
;
EPLUSE: LD    BC,PORTB1
        IN   A,(C)
        SET  2,A                  ;Enable bit 2 = 1
        OUT (C),A
        LD   B,00H
EP1:    DEC  B

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

NOP
NOP
JP    NZ,EP1
LD    BC,PORTB1
RES   2,A           ;Enable bit 2 = 0
OUT   (C),A
RET

```

```

;
;
;

```

GOTO POSITION

```

GOTO:  PUSH BC
        LD  BC,PORTA1
        SET 7,A           ;Set DD RAM
        OUT (C),A
        LD  BC,PORTB1
        XOR A           ;Set RS = 0, R/W = 0
        OUT (C),A
        CALL EPLUSE     ;Enable signal pluse
        POP BC
        RET

```

```

;
;
;

```

WRITE DATA

```

WRBYTE: LD  BC,PORTB1
         LD  A,00000001B ;Data write
         OUT (C),A
         LD  BC,PORTA1
         LD  A,D         ;Data byte
         OUT (C),A
         CALL EPLUSE     ;Enable signal pluse
         RET

```







```

POP      AF          ;LOAD
CVERT:   AND      OFH
         ADD      A,90H
         DAA
         ADC      A,40H
         DAA
         LD       E,A          ;STORE
         RET

```



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

;
; SUBROUTINE SEGMENT
;

```

```

;
; PLAY MIDI NOTE
;

```

```

PLAY_NTE: CALL CHEK_TDR
          LD  A,90
          OUT (TDR1),A
          CALL CHEK_TDR
          LD  A,(IX+0)
          OUTO (TDR1),A
          CALL CHEK_TDR
          LD  A,40H
          OUTO (TDR1),A
          RET

```

```

;
; STOP MIDI NOTE
;

```

```

STOP_NTE: CALL CHEK_TDR
          LD  A,80
          OUTO (TDR1),A
          CALL CHEK_TDR
          LD  A,(IX+0)
          OUTO (TDR1),A
          CALL CHEK_TDR
          LD  A,00H
          OUTO (TDR1),A
          RET

```

```

;
; CHECK Tx DATA EMPTY
;

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

CHEK_TDR:  INO  A,(STAT1)
            BIT  1,A
            JR   NZ,CHEK_TDR
            RET

```

```

;
;  DELAY LOOP  10 ms
;

```

```

DELAY:     PUSH HL
           PUSH DE
           LD   B,(IX+0)
           LD   DE,-1
DLYO:     LD   HL,047EH
DLY1:     ADD  HL,DE
           JR   C,DLY1
           DJNZ DLYO
           POP  DE
           POP  HL
           RET

```

```

;
;  ALL NOTE OFF
;

```

```

ALL_OFF:  CALL CHEK_TDR      ;ALL NOTE OFF
           LD   A,OBOH
           OUTO (TDR1),A
           CALL CHEK_TDR
           LD   A,7BH
           OUTO (TDR1),A
           CALL CHEK_TDR
           LD   A,OOH
           OUTO (TDR1),A
           RET

```

```

;
;   I N T E R R U P T       S E G M E N T
;

```

```

;
;   I N T E R R U P T   T I M E R   C H A N N E L   0
;

```

```

PRT_0:   DI
          PUSH AF
          PUSH BC
TIF_0:   INO  A,(TCR)
          BIT  6,A
          JR   Z,TIF_0
          INO  A,(TMDROL)
          LD   A,(ACT_300)
          CP   06H
          JR   NZ,INC_ACT
          CALL CHK_TDR
          LD   A,0FEH
          OUTO (TDR1),A
          LD   A,00H
INC_ACT: INC  A
          LD   (ACT_300),A

```

```

;
;   N O   S O U N D   C O U N T E R   1 0 0   m s
;

```

```

          LD   A,(CNT_OFF)
          INC  A
          LD   (CNT_OFF),A

```

```

;
;   E N A B L E   I N T E R R U P T   0
;

```

```

          LD   A,01H
          OUTO (ITC),A

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

POP BC  
POP AF  
EI  
RETI



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้





```

.DB 00H, 10H, 08H, 09H, 0AH, 04H, 00H, 00H ;02
.DB 00H, 00H, 10H, 0BH, 04H, 00H, 00H, 00H ;03
.DB 00H, 00H, 04H, 1BH, 00H, 00H, 00H, 00H ;04
.DB 0FH, 09H, 0FH, 08H, 08H, 18H, 18H, 00H ;05
.DB 0FH, 09H, 0FH, 09H, 0BH, 1BH, 18H, 00H ;06
.DB 1FH, 04H, 1BH, 1BH, 1FH, 11H, 1FH, 00H ;07
NOTE_TB: .DB 3CH, 3EH, 40H, 41H, 43H, 45H, 47H, 48H
         .DB 47H, 45H, 43H, 41H, 40H, 3EH
TIME_TB: .DB 04H, 04H, 04H, 04H, 04H, 04H, 04H, 04H
         .DB 04H, 04H, 04H, 04H, 04H, 04H
.ORG 8000H
;
; DATA BUFFER IN RAM
;
OLD_NTE: .RS 1
NEW_NTE: .RS 1
CNT_NTE: .RS 1
DIF_NEW: .RS 1
DIF_OLD: .RS 1
PIT_COD: .RS 1
ACT_300: .RS 1
CNT_OFF: .RS 1
PIT_REC: .RS 2
PIT_VAL: .RS 2
OLD_PIT: .RS 2
NEW_PIT: .RS 2
DSP_BUF: .RS 16
.END

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## Z80180 Z180 MPU

### FEATURES:

- Operating Frequency to 12.5 MHz
- On-Chip MMU Supports Extended Address Space
- Two DMA Channels
- On-Chip Wait State Generators
- Two UART Channels
- Two 16-Bit Timer Channels
- On-Chip Interrupt Controller
- On-Chip Clock Oscillator/Generator
- Clocked Serial I/O Port
- Code Compatible with Zilog Z80 CPU
- Extended Instructions
- 8 MHz Version Supports 6.144 MHz CPU Clock Operation

### GENERAL DESCRIPTION:

Based on a microcoded execution unit and an advanced CMOS manufacturing technology, the Z80180 is an 8-bit MPU which provides the benefits of reduced system costs and low power operation while offering higher performance and maintaining compatibility with a large base of industry standard software written around the Zilog Z80 CPU.

Higher performance is obtained by virtue of higher operating frequencies, reduced instruction execution times, an enhanced instruction set, and an on-chip memory management unit (MMU) with the capability of addressing up to 1 Mbyte of memory.

Reduced system costs are obtained by incorporating several key system functions on-chip with the CPU. These key functions include I/O devices such as DMA, UART, and timer channels. Also included on-chip are several "glue"

functions such as dynamic RAM refresh control, wait state generators, clock oscillator, and interrupt controller.

Not only does the Z80180 consume a low amount of power during normal operation, but it also provides two operating modes that are designed to drastically reduce the power consumption even further. The SLEEP mode reduces power by placing the CPU into a "stopped" state, thereby consuming less current, while the on-chip I/O device is still operating. The SYSTEM STOP mode places both the CPU and the on-chip peripherals into a "stopped" mode, thereby reducing power consumption even further.

When combined with other CMOS VLSI devices and memories, the Z80180 provides an excellent solution to system applications requiring high performance, and low power operation.

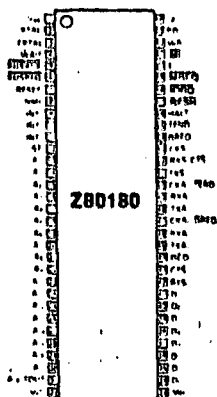


Figure 1. 64 Pin DIP

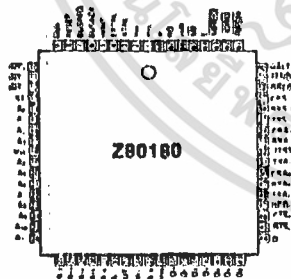


Figure 2. 60 Pin PLCC

## Z80180 ELECTRICAL CHARACTERISTICS

### ABSOLUTE MAXIMUM RATINGS

Item	Symbol	Value	Unit
Supply Voltage	$V_{CC}$	-0.3 - +7.0	V
Input Voltage	$V_{in}$	-0.3 - $V_{CC} + 0.3$	V
Operating Temperature	Standard	0 - 70	°C
	Extended	-40 - 85	°C
Storage Temperature	$T_{stg}$	-55 - +150	°C

[NOTE] Permanent LSI damage may occur if maximum ratings are exceeded. Normal operation should be under recommended operating conditions. If these conditions are exceeded, it could affect reliability of LSI.

### DC CHARACTERISTICS

( $V_{CC} = 5V \pm 10\%$ ,  $V_{SS} = 0V$ , over specified temperature range, unless otherwise noted.)

Symbol	Item	Condition	min	typ	max	Unit
$V_{OH1}$	Input "H" Voltage RESET, EXTERNAL NMI		$V_{CC} - 0.8$	-	$V_{CC} + 0.3$	V
$V_{OH2}$	Input "H" Voltage Except RESET, EXTERNAL NMI		2.0	-	$V_{CC} + 0.3$	V
$V_{OL1}$	Input "L" Voltage RESET, EXTERNAL NMI		-0.3	-	0.8	V
$V_{OL2}$	Input "L" Voltage Except RESET, EXTERNAL NMI		-0.3	-	0.8	V
$V_{OH}$	Output "H" Voltage All outputs	$I_{OH} = -200\mu A$	2.4	-	-	V
		$I_{OH} = -20\mu A$	$V_{CC} - 1.2$	-	-	V
$V_{OL}$	Output "L" Voltage All Outputs	$I_{OL} = 22 mA$	-	-	0.45	V
			-	-	-	V
$I_{IH}$	Input Leakage Current All Inputs Except XTAL, EXTERNAL	$V_{in} = 0.5 - V_{CC} - 0.5$	-	-	1.0	$\mu A$
$I_{IL}$	Input Leakage Current All Inputs Except XTAL, EXTERNAL	$V_{in} = 0.5 - V_{CC} - 0.5$	-	-	1.0	$\mu A$
$P_{CC}$	Power Dissipation* (Normal Operation)	$f = 6 MHz$	-	15	40	mW
		$f = 8 MHz$	-	20	50	
		$f = 10 MHz$	-	25	60	
	Power Dissipation* (SYSTEM STOP mode)	$f = 6 MHz$	-	3.6	12.5	
		$f = 8 MHz$	-	5	15.0	
		$f = 10 MHz$	-	6.3	17.5	
$C_D$	Pin Capacitance	$V_{in} = 0V, f = 1 MHz$ $T_B = 25^\circ C$	-	-	12	pF

\*  $V_{OHmin} = V_{CC} - 1.0V$ ,  $V_{OLmax} = 0.8V$  (all output terminals are in no load)  
 $V_{CC} = 5.00V$

**Z80180 AC CHARACTERISTICS**

 (V<sub>CC</sub>=5V ± 10%, V<sub>SS</sub>=0V, over specified temperature range, unless otherwise noted.)

No	Sym	Parameter	Z8018006		Z8018008		Z8018010		Z8018012 [3,4]		Unit	Note
			Min	Max	Min	Max	Min	Max	Min	Max		
1	t <sub>CC</sub>	Clock Cycle time	162	2000	125	2000	100	2000	80	2000	ns	[1]
2	ICLW	Clock Pulse width (high)	65		55		40		35		ns	[1]
3	ICLW	Clock Pulse width (low)	65		55		40		35		ns	[1]
4	t <sub>CF</sub>	Clock Fall time		15		15		10		5	ns	[1]
5	t <sub>CR</sub>	Clock Rise time		15		15		10		5	ns	[1]
6	IAD	Address valid from Clock Rise		90		80		70		50	ns	
7	IAS	Address valid to /MREQ, /ORQ Fall	30		20		10		10		ns	
8	IMED1	Clock Fall to /MREQ Fall delay		60		50		50		45	ns	
9	IRDD1	Clock Fall to /RD Fall (/OC=1)		60		50		50		45	ns	
		Clock Rise to /RD Fall (/OC=0)		65		60		55		50	ns	
10	IM1D1	Clock Rise to /M1 Fall delay		80		70		60		50	ns	
11	IAH	Address Hold time (/MREQ, /ORQ, /RD, /WR)	35		20		10		10		ns	
12	IMED2	Clock Fall to /MREQ Rise Delay		60		50		50		45	ns	
13	IR1D2	Clock Fall to /RD Rise delay		60		50		50		45	ns	
14	IM1D2	Clock Rise to /M1 Rise delay		60		70		60		50	ns	
15	IDHS	Data Hold Setup Time	40		30		25		20		ns	
16	IDRH	Data Read Hold time	0		0		0		0		ns	
17	ISTD1	Clock Edge to ST Fall		90		70		60		50	ns	
18	ISTD2	Clock Edge to ST Rise		90		70		60		50	ns	
19	IWS	WAIT setup time to Clock Fall	40		40		30		20		ns	
20	IWH	WAIT Hold time from Clock Fall	40		40		30		20		ns	
21	IWD2	Clock Rise to Data Hold Delay		95		70		60		60	ns	
22	IWRD1	Clock Rise to /WR Fall delay		65		60		50		30	ns	
23	IWDD	Clock Fall to Write Data Delay		90		80		60		50	ns	
24	IWDS	Write Data Setup time to /WR	40		20		15		10		ns	
25	IWRD2	Clock Fall to /WR Rise		80		60		50		45	ns	
26	IWRP	/WR Pulse Width (Memory Write Cycles)	170		130		110		85		ns	
26a		/WR Pulse Width (/O Write Cycles)	332		225		210		185		ns	
27	IWDH	Write Data Hold time from /WR Rise	40		15		10		10		ns	
28	I1D1	Clock Fall to /IORQ Fall delay (/OC=1)		60		50		50		45	ns	
		Clock Rise to /IORQ Fall delay (/OC=0)		65		60		55		50	ns	
29	I1D2	Clock Fall /IORQ Rise Delay		60		50		50		50	ns	
30	I1D3	/M1 Fall to /IORQ Fall delay	340		250		200		160		ns	
31	IINTS	/INT Setup Time to Clock Fall	40		40		30		20		ns	
32	IINTH	/INT Hold Time from Clock Fall	40		40		30		20		ns	
33	I1NMIW	/NMI Pulse width	120		100		80		60		ns	
34	IBRS	/BUSREQ Setup Time to Clock Fall	40		40		30		20		ns	
35	IBRH	/BUSREQ Hold Time from Clock Fall	40		40		30		20		ns	
36	IBAD1	Clock Rise to /BUSACK Fall delay		95		70		60		50	ns	
37	IBAD2	Clock Fall to /BUSACK Rise delay		95		70		60		50	ns	
38	IRZD	Clock Rise to Bus Floating Delay Time		125		90		80		60	ns	
39	IMEWH	/MREQ Pulse Width (High)	110		90		70		60		ns	
40	IMEWL	/MREQ Pulse Width (Low)	125		100		80		60		ns	

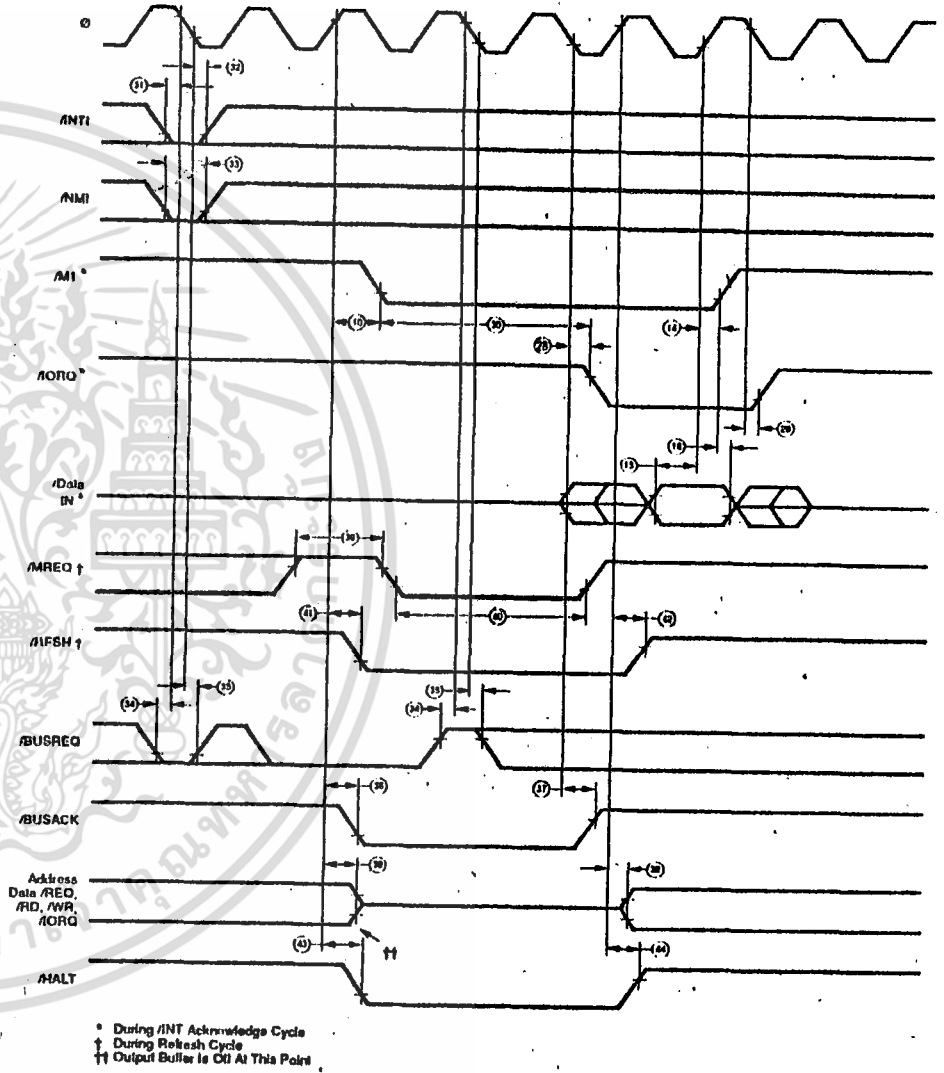
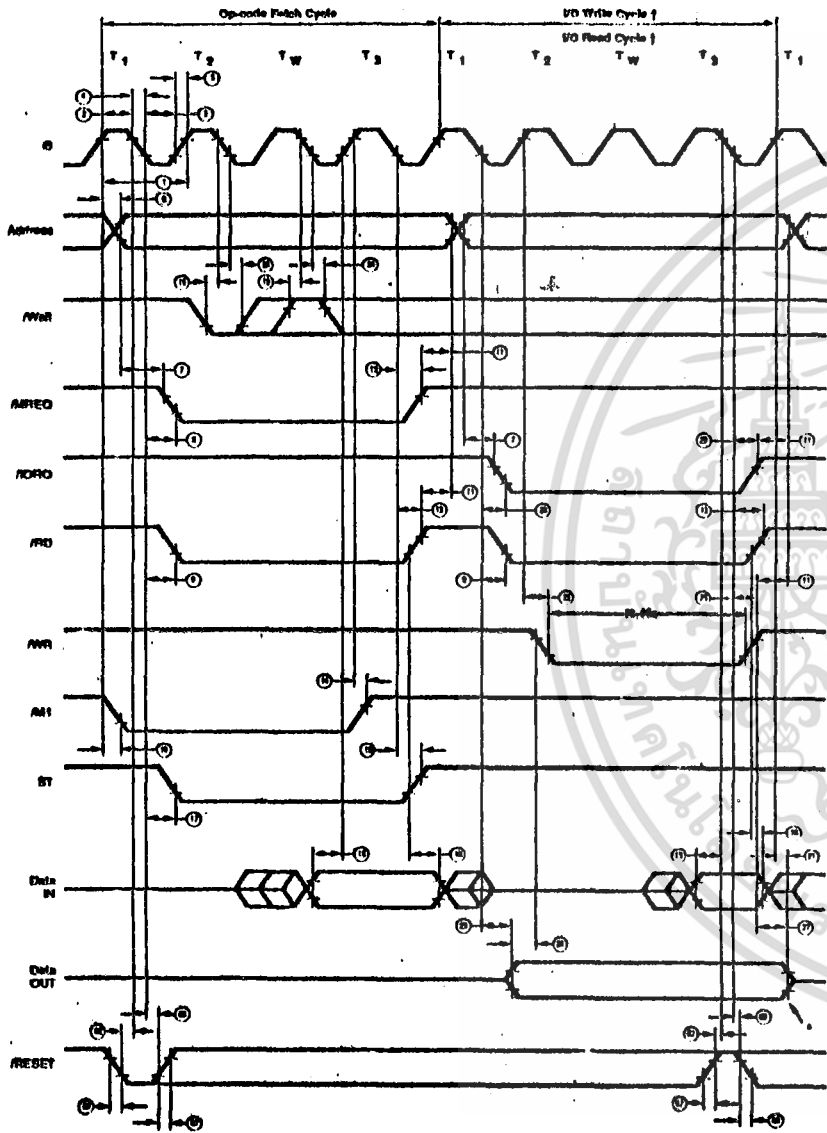
**Z80180 AC CHARACTERISTICS (Continued)**

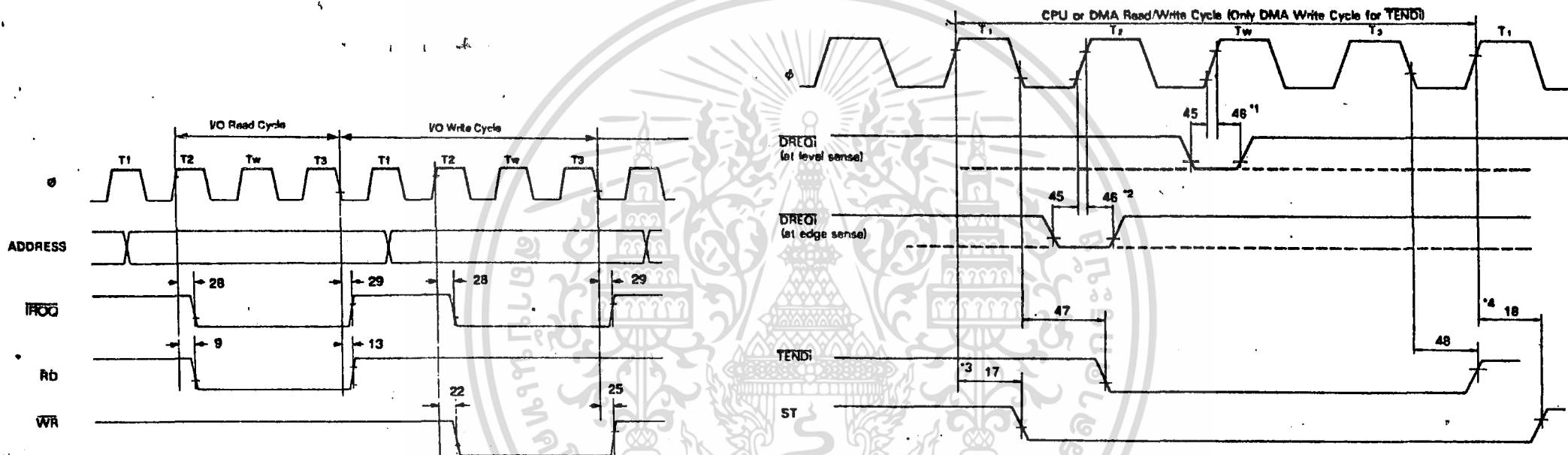
 (V<sub>CC</sub>=5V ± 10%, V<sub>SS</sub>=0V, over specified temperature range, unless otherwise noted.)

No	Sym	Parameter	Z8018006		Z8018008		Z8018010		Z8018012 [3,4]		Unit	Note
			Min	Max	Min	Max	Min	Max	Min	Max		
41	IRFD1	Clock Rise to /RFSH Fall Delay		90		80		60		40	ns	
42	IRFD2	Clock Rise to /RFSH Rise Delay		90		80		60		40	ns	
43	IHAD1	Clock Rise to /HALT Fall Delay		90		80		50		30	ns	
44	IHAD2	Clock Rise to /HALT Rise Delay		90		80		50		30	ns	
45	IDROS	/DREQ Setup Time to Clock Rise	40		40		30		20		ns	
46	IDROH	/DREQ Hold Time from Clock Rise	40		40		30		20		ns	
47	ITLD1	Clock Fall to /TENDI Fall Delay		70		60		50		50	ns	
48	ITLD2	Clock Fall to /TENDI Rise Delay		70		60		50		50	ns	
49	ITD1	Clock Rise to E Rise Delay		95		70		60		40	ns	
50	ITD2	Clock Edge to E Fall Delay		95		70		60		40	ns	
51	PWH	E Pulse Width (High)	75		65		55		45		ns	
52	PWL	E Pulse Width (Low)	180		130		110		90		ns	
53	IER	Enable Rise Time		20		20		20		10	ns	
54	IF1	Enable Fall Time		20		20		20		10	ns	
55	I1OD	Clock Fall to Timer Output Delay		300		200		150		120	ns	
56	IS1D1	CS/O 1x Data Delay Time (Internal Clock Operation)		200		200		150		120	ns	
57	ISTDE	CS/O 1x Data Delay Time (External Clock Operation)		7.5t <sub>CC</sub> +300		7.5t <sub>CC</sub> +200		7.5t <sub>CC</sub> +150		7.5t <sub>CC</sub> +120	ns	
58	ISRS1	CS/O Rx Data Setup Time (Internal Clock Operation)	1		1		1		1		t <sub>CC</sub>	
59	ISRH1	CS/O Rx Data Hold Time (Internal Clock Operation)	1		1		1		1		t <sub>CC</sub>	
60	ISRS2	CS/O Rx Data Setup Time (External Clock Operation)	1		1		1		1		t <sub>CC</sub>	
61	ISRH2	CS/O Rx Data Hold Time (External Clock Operation)	1		1		1		1		t <sub>CC</sub>	
62	IRES	/RESET Setup time to Clock Fall	120		100		80		60		ns	
63	IRFH	/RESET Hold time from Clock Fall	80		70		60		45		ns	
64	IOSC	Oscillator Stabilization Time		20		20		20		20	ns	
65	ITEX	External Clock Rise Time (EXTAL)		25		25		15		10	ns	
66	ITEX1	External Clock Fall Time (EXTAL)		25		25		15		10	ns	
67	ITR	/RESET Rise Time		50		50		50		50	ms	[2]
68	ITRF	/RESET Fall Time		50		50		50		50	ms	[2]
69	ITR	Input Rise Time (Except EXTAL, /RESET)		100		100		100		80	ns	[2]
70	ITF	Input Fall Time (Except EXTAL, /RESET)		100		100		100		80	ns	[2]

**Notes:**

- (1) t<sub>CC</sub>=ICLW+ICLW+t<sub>CF</sub>+t<sub>CR</sub>
- (2) This parameter has to be modified if other specification(s) can not be met.
- (3) For Z8018012, V<sub>CC</sub> range is limited to ± 5%.
- (4) Numbers are preliminary and subject to change without notice.



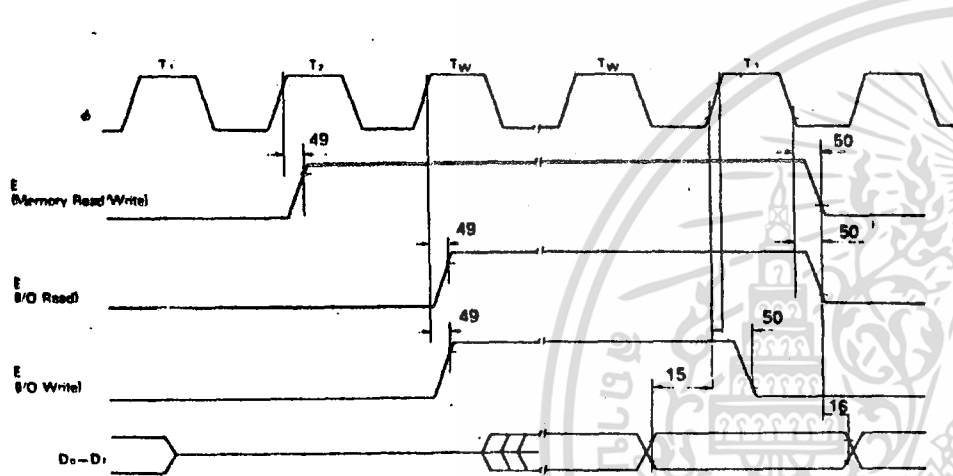


**CPU Timing (IFC=0)**

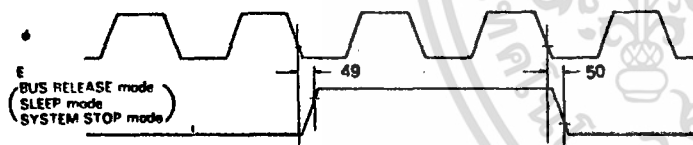
{ I/O Read Cycle  
 I/O Write Cycle }

**DMA Control Signals**

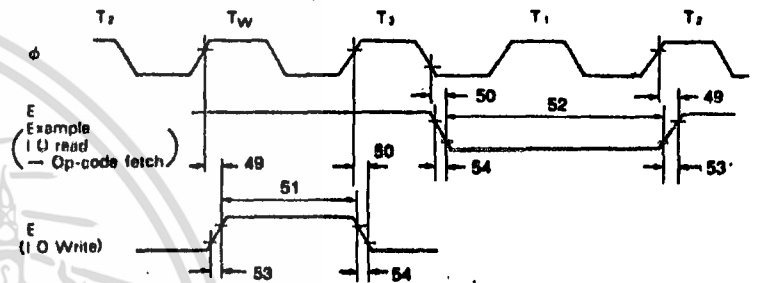
- \*1  $t_{PROS}$  and  $t_{DROH}$  are specified for the rising edge of clock followed by  $T_3$ .
- \*2  $t_{PROS}$  and  $t_{DROH}$  are specified for the rising edge of clock.
- \*3 DMA cycle starts.
- \*4 CPU cycle starts.



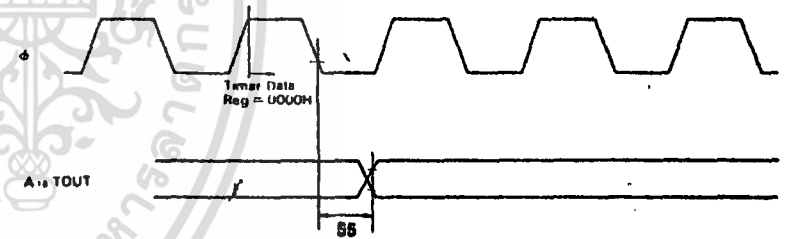
E Clock Timing (Memory Read/Write Cycle / I/O Read/Write Cycle)



E Clock Timing (BUS RELEASE Mode / SLEEP Mode / SYSTEM STOP Mode)



E Clock Timing (Minimum timing example of P<sub>WEL</sub> and P<sub>WEH</sub>)



Timer Output Timing

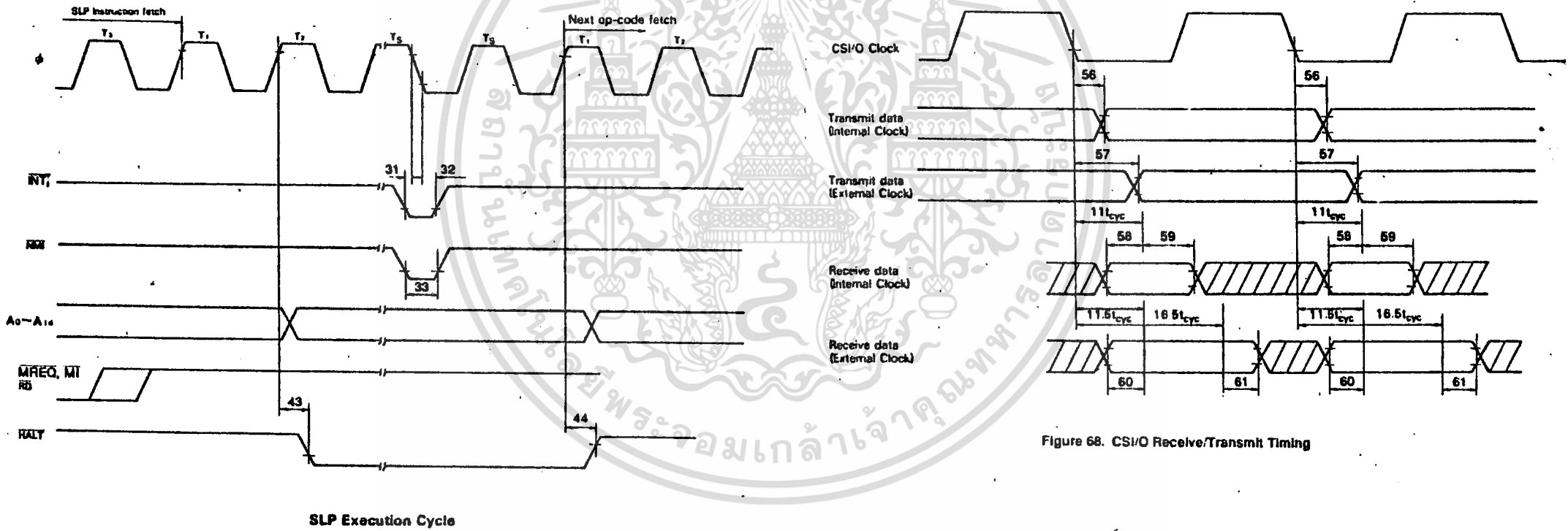


Figure 68. CSIO Receive/Transmit Timing

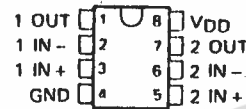
**TLC27M2, TLC27M2A, TLC27M2B, TLC27M7**  
**LinCMOS™ PRECISION DUAL OPERATIONAL AMPLIFIERS**

**TLC27M2, TLC27M2A, TLC27M2B, TLC27M7**  
**LinCMOS™ PRECISION DUAL OPERATIONAL AMPLIFIERS**

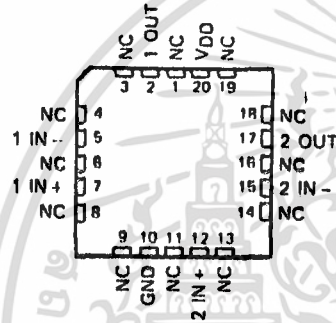
D3140, OCTOBER 1987 - REVISED FEBRUARY 1989

- Trimmed Offset Voltage:  
**TLC27M7** . . . 500  $\mu$ V Max at 25 °C,  
 $V_{DD} = 5$  V
- Input Offset Voltage Drift . . . Typically  
**0.1  $\mu$ V/Month**, including the First 30 Days
- Wide Range of Supply Voltages over  
 Specified Temperature Range:  
 -55 °C to 125 °C . . . 4 V to 16 V  
 -40 °C to 85 °C . . . 4 V to 16 V  
 0 °C to 70 °C . . . 3 V to 16 V
- Single-Supply Operation
- Common-Mode Input Voltage Range  
 Extends Below the Negative Rail (C-Suffix,  
 I-Suffix types)
- Low Noise . . . Typically 32 nV/ $\sqrt{\text{Hz}}$   
 at  $f = 1$  kHz
- Low Power . . . Typically 2.1 mW at 25 °C,  
 $V_{DD} = 5$  V
- Output Voltage Range Includes Negative  
 Rail
- High Input Impedance . . . 10<sup>12</sup>  $\Omega$  Typical
- ESD-Protection Circuitry
- Small-Outline Package Option Also Available  
 In Tape and Reel
- Designed-In Latchup Immunity

**D, JG, OR P PACKAGE**  
 (TOP VIEW)



**FK PACKAGE**  
 (TOP VIEW)



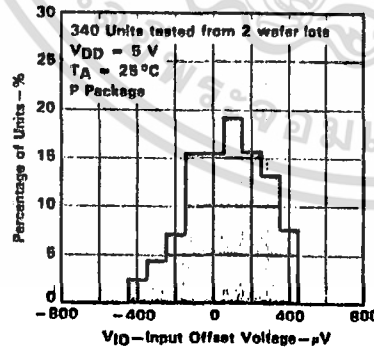
NC—No internal connection

**AVAILABLE OPTIONS**

T <sub>A</sub>	V <sub>IO</sub> max at 25 °C	PACKAGE			
		SMALL OUTLINE (D)	CHIP CARRIER (FK)	CERAMIC DIP (JG)	PLASTIC DIP (P)
0 °C to 70 °C	500 $\mu$ V	TLC27M7CD	—	TLC27M7CJG	TLC27M7CP
	2 mV	TLC27M2BCD	—	TLC27M2BCJG	TLC27M2BCP
	5 mV	TLC27M2ACD	—	TLC27M2ACJG	TLC27M2ACP
-40 °C to 85 °C	10 mV	TLC27M2CD	—	TLC27M2CJG	TLC27M2CP
	500 $\mu$ V	TLC27M7ID	—	TLC27M7IJG	TLC27M7IP
	2 mV	TLC27M2BID	—	TLC27M2BIJG	TLC27M2BIP
-55 °C to 125 °C	5 mV	TLC27M2AID	—	TLC27M2AIJG	TLC27M2AIP
	10 mV	TLC27M2ID	—	TLC27M2IJG	TLC27M2IP
	500 $\mu$ V	—	TLC27M7MFK	TLC27M7MJG	—
10 mV	—	TLC27M2MFK	TLC27M2MJG	—	

The D package is available in tape and reel. Add R suffix to the device type (e.g., TLC27M7CDR).

**DISTRIBUTION OF TLC27M7  
 INPUT OFFSET VOLTAGE**



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

Supply voltage, $V_{DD}$ (see Note 1) . . . . .	18 V
Differential input voltage (see Note 2) . . . . .	$\pm V_{DD}$
Input voltage range, $V_I$ (any input) . . . . .	-0.3 V to $V_{DD}$
Input current, $I_I$ . . . . .	$\pm 5$ mA
Output current, $I_O$ (each output) . . . . .	$\pm 30$ mA
Total current into $V_{DD}$ terminal . . . . .	45 mA
Total current out of ground terminal . . . . .	45 mA
Duration of short-circuit current at (or below) 25 °C (see Note 3) . . . . .	unlimited
Continuous total dissipation . . . . .	See Dissipation Rating Table
Operating free-air temperature, $T_A$ : M-suffix . . . . .	-55 °C to 125 °C
I-suffix . . . . .	-40 °C to 85 °C
C-suffix . . . . .	0 °C to 70 °C
Storage temperature range . . . . .	-65 °C to 150 °C
Case temperature for 60 seconds: FK package . . . . .	280 °C
Lead temperature 1.6 mm (1/16 inch) from case for 10 seconds: D and P package . . . . .	280 °C
Lead temperature 1.6 mm (1/16 inch) from case for 60 seconds: JG package . . . . .	300 °C

- NOTES: 1. All voltage values, except differential voltages, are with respect to network ground.  
 2. Differential voltages are at the noninverting input with respect to the inverting input.  
 3. The output may be shorted to either supply. Temperature and/or supply voltages must be limited to ensure that the maximum dissipation rating is not exceeded (see application section).

**DISSIPATION RATING TABLE**

PACKAGE	$T_A \leq 25$ °C	DERATING FACTOR	$T_A = 70$ °C	$T_A = 85$ °C	$T_A = 125$ °C
	POWER RATING	ABOVE $T_A = 25$ °C	POWER RATING	POWER RATING	POWER RATING
D	725 mW	6.8 mW/°C	464 mW	377 mW	
FK	1375 mW	11 mW/°C	880 mW	715 mW	275 mW
JG (M-suffix)	1050 mW	8.4 mW/°C	672 mW	546 mW	210 mW
JG (C-, I-suffix)	825 mW	6.6 mW/°C	528 mW	429 mW	
P	1000 mW	8.0 mW/°C	640 mW	520 mW	

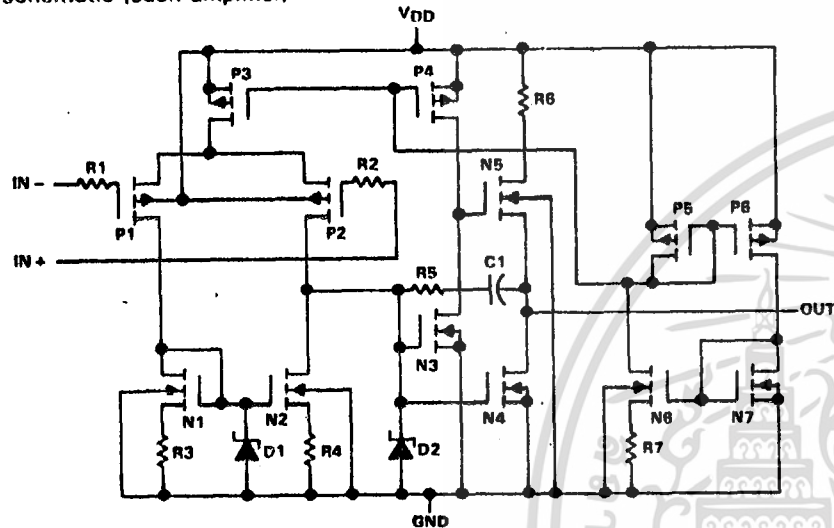
**recommended operating conditions**

		M-SUFFIX			I-SUFFIX			C-SUFFIX			UNIT
		MIN	NOM	MAX	MIN	NOM	MAX	MIN	NOM	MAX	
Supply voltage, $V_{DD}$		4		18	4		18	3		18	V
Common-mode input voltage, $V_{IC}$	$V_{DD} = 5$ V	0		3.5	-0.2		3.5	-0.2		3.5	V
	$V_{DD} = 10$ V	0		8.5	-0.2		8.5	-0.2		8.5	V
Operating free-air temperature, $T_A$		-55		125	-40		85	0		70	°C

TLC27M2, TLC27M2A, TLC27M2B, TLC27M7  
LinCMOS™ PRECISION DUAL OPERATIONAL AMPLIFIERS

TLC27M2, TLC27M2A, TLC27M2B, TLC27M7  
LinCMOS™ PRECISION DUAL OPERATIONAL AMPLIFIERS

Equivalent schematic (each amplifier)



description

The TLC27M2 and TLC27M7 dual operational amplifiers combine a wide range of input offset voltage grades with low offset voltage drift, high input impedance, low noise, and speeds comparable to that of general-purpose bipolar devices. These devices use Texas Instruments silicon-gate LinCMOS™ technology, which provides offset voltage stability far exceeding the stability available with conventional metal-gate processes.

The extremely high input impedance and low bias currents make these cost-effective devices ideal for applications which have previously been reserved for general-purpose bipolar products, but with only a fraction of the power consumption. Four offset voltage grades are available (C-suffix and I-suffix types), ranging from the low-cost TLC27M2 (10 mV) to the high-precision TLC27M7 (500  $\mu$ V). These advantages, in combination with good common-mode rejection and supply voltage rejection, make these devices a good choice for new state-of-the-art designs as well as for upgrading existing designs.

In general, many features associated with bipolar technology are available on LinCMOS™ operational amplifiers, without the power penalties of bipolar technology. General applications such as transducer interfacing, analog calculations, amplifier blocks, active filters, and signal buffering are easily designed with the TLC27M2 and TLC27M7. The devices also exhibit low voltage single-supply operation and low power consumption, making them ideally suited for remote and inaccessible battery-powered applications. The common-mode input voltage range includes the negative rail.

A wide range of packaging options is available, including small-outline and chip carrier versions for high-density system applications.

The device inputs and outputs are designed to withstand  $-100$ -mA surge currents without sustaining latchup.

The TLC27M2 and TLC27M7 incorporate internal ESD-protection circuits that prevent functional failures at voltages up to 2000 V as tested under MIL-STD-883C, Method 3015.2; however, care should be exercised in handling these devices as exposure to ESD may result in the degradation of the device parametric performance.

M-suffix devices are characterized for operation over the full military temperature range of  $-55^{\circ}\text{C}$  to  $125^{\circ}\text{C}$ . I-suffix devices are characterized for operation from  $-40^{\circ}\text{C}$  to  $85^{\circ}\text{C}$ , and C-suffix devices are characterized for operation from  $0^{\circ}\text{C}$  to  $70^{\circ}\text{C}$ .

**TLC27M2M, TLC27M7M**  
**LinCMOS™ PRECISION DUAL OPERATIONAL AMPLIFIERS**

electrical characteristics at specified free-air temperature,  $V_{DD} = 5\text{ V}$  (unless otherwise noted)

PARAMETER		TEST CONDITIONS		$T_A$ †	MIN	TYP	MAX	UNIT
$V_{IO}$	Input offset voltage	TLC27M2M	$V_O = 1.4\text{ V}$ , $R_S = 50\ \Omega$ , $V_{IC} = 0\text{ V}$ , $R_L = 100\text{ k}\Omega$	25°C		1.1	10	mV
				Full range			12	
$V_{IO}$	Input offset voltage	TLC27M7M	$V_O = 1.4\text{ V}$ , $R_S = 80\ \Omega$ , $V_{IC} = 0\text{ V}$ , $R_L = 100\text{ k}\Omega$	25°C		185	500	$\mu\text{V}$
				Full range			3750	
$\alpha_{VIO}$	Average temperature coefficient of input offset voltage			25°C to 125°C		1.7		$\mu\text{V}/^\circ\text{C}$
$I_{IO}$	Input offset current (see Note 4)	$V_O = 2.5\text{ V}$ , $V_{IC} = 2.5\text{ V}$		25°C		0.1		pA
				125°C		1.4	15	nA
$I_{IB}$	Input bias current (see Note 4)	$V_O = 2.5\text{ V}$ , $V_{IC} = 2.5\text{ V}$		25°C		0.6		pA
				125°C		9	85	nA
$V_{ICR}$	Common-mode input voltage range (see Note 5)			25°C	0	-0.3		V
					$t_0$	$t_0$		
$V_{ICR}$	Common-mode input voltage range (see Note 5)			Full range	0			V
					$t_0$	$t_0$		
$V_{OH}$	High-level output voltage	$V_{ID} = 100\text{ mV}$ , $R_L = 100\text{ k}\Omega$		25°C	3.2	3.9		V
				-55°C	3	3.9		
				125°C	3	4		
$V_{OL}$	Low-level output voltage	$V_{ID} = -100\text{ mV}$ , $I_{OL} = 0$		25°C		0	50	mV
				-55°C		0	50	
				125°C		0	50	
$A_{VD}$	Large-signal differential voltage amplification	$V_O = 0.25\text{ V to } 2\text{ V}$ , $R_L = 100\text{ k}\Omega$		25°C	25	170		V/mV
				-55°C	15	290		
				125°C	15	120		
CMRR	Common-mode rejection ratio	$V_{IC} = V_{ICR\text{ min}}$		25°C	66	91		dB
				-55°C	60	89		
				125°C	60	91		
$k_{SVR}$	Supply-voltage rejection ratio ( $\Delta V_{DD}/\Delta V_{IO}$ )	$V_{DD} = 5\text{ V to } 10\text{ V}$ , $V_O = 1.4\text{ V}$		25°C	70	93		dB
				-55°C	80	91		
				125°C	60	94		
$I_{DD}$	Supply current (two amplifiers)	$V_O = 2.5\text{ V}$ , No load	$V_{IC} = 2.5\text{ V}$	25°C		210	560	$\mu\text{A}$
				-55°C		340	880	
				125°C		140	360	

† Full range is -55°C to 125°C.

NOTES: 4. The typical values of input bias current and input offset current below 5 pA were determined mathematically.  
5. This range also applies to each input individually.

**TLC27M2M, TLC27M7M**  
**LinCMOS™ PRECISION DUAL OPERATIONAL AMPLIFIERS**

electrical characteristics at specified free-air temperature,  $V_{DD} = 10\text{ V}$  (unless otherwise noted)

PARAMETER		TEST CONDITIONS		$T_A$ †	MIN	TYP	MAX	UNIT
$V_{IO}$	Input offset voltage	TLC27M2M	$V_O = 1.4\text{ V}$ , $R_S = 50\ \Omega$ , $V_{IC} = 0\text{ V}$ , $R_L = 100\text{ k}\Omega$	25°C		1.1	10	mV
				Full range			12	
$V_{IO}$	Input offset voltage	TLC27M7M	$V_O = 1.4\text{ V}$ , $R_S = 80\ \Omega$ , $V_{IC} = 0\text{ V}$ , $R_L = 100\text{ k}\Omega$	25°C		180	800	$\mu\text{V}$
				Full range			4300	
$\alpha_{VIO}$	Average temperature coefficient of input offset voltage			25°C to 125°C		2.1		$\mu\text{V}/^\circ\text{C}$
$I_{IO}$	Input offset current (see Note 4)	$V_O = 5\text{ V}$ , $V_{IC} = 5\text{ V}$		25°C		0.1		pA
				125°C		1.8	16	nA
$I_{IB}$	Input bias current (see Note 4)	$V_O = 5\text{ V}$ , $V_{IC} = 5\text{ V}$		25°C		0.7		pA
				125°C		10	35	nA
$V_{ICR}$	Common-mode input voltage range (see Note 5)			25°C	0	-0.3		V
					$t_0$	$t_0$		
$V_{ICR}$	Common-mode input voltage range (see Note 5)			Full range	0			V
					$t_0$	$t_0$		
$V_{OH}$	High-level output voltage	$V_{ID} = 100\text{ mV}$ , $R_L = 100\text{ k}\Omega$		25°C	8	8.7		V
				-55°C	7.8	8.8		
				125°C	7.8	8.8		
$V_{OL}$	Low-level output voltage	$V_{ID} = -100\text{ mV}$ , $I_{OL} = 0$		25°C		0	60	mV
				-55°C		0	60	
				125°C		0	60	
$A_{VD}$	Large-signal differential voltage amplification	$V_O = 1\text{ V to } 6\text{ V}$ , $R_L = 100\text{ k}\Omega$		25°C	25	276		V/mV
				-55°C	16	420		
				125°C	16	180		
CMRR	Common-mode rejection ratio	$V_{IC} = V_{ICR\text{ min}}$		25°C	88	84		dB
				-55°C	80	83		
				125°C	80	83		
$k_{SVR}$	Supply-voltage rejection ratio ( $\Delta V_{DD}/\Delta V_{IO}$ )	$V_{DD} = 5\text{ V to } 10\text{ V}$ , $V_O = 1.4\text{ V}$		25°C	70	93		dB
				-55°C	80	91		
				125°C	60	94		
$I_{DD}$	Supply current (two amplifiers)	$V_O = 5\text{ V}$ , No load	$V_{IC} = 5\text{ V}$	25°C		286	600	$\mu\text{A}$
				-55°C		480	1000	
				125°C		180	480	

† Full range is -55°C to 125°C.

NOTES: 4. The typical values of input bias current and input offset current below 5 pA were determined mathematically.  
5. This range also applies to each input individually.

**TLC27M2M, TLC27M7M**  
**LinCMOS™ PRECISION DUAL OPERATIONAL AMPLIFIERS**

operating characteristics,  $V_{DD} = 5\text{ V}$

PARAMETER	TEST CONDITIONS		$T_A$	MIN	TYP	MAX	UNIT
SR Slew rate at unity gain	$R_L = 100\text{ k}\Omega$ , $C_L = 20\text{ pF}$ , See Figure 1	$V_{ipp} = 1\text{ V}$	25°C		0.43		V/ $\mu$ s
			-55°C		0.54		
			125°C		0.29		
		$V_{ipp} = 2.5\text{ V}$	25°C		0.40		
			-55°C		0.49		
			125°C		0.28		
$V_n$ Equivalent input noise voltage	$f = 1\text{ kHz}$ , See Figure 2	$R_S = 100\ \Omega$	25°C		32		nV/ $\sqrt{\text{Hz}}$
BOM Maximum output swing bandwidth	$V_O = V_{OH}$ , $R_L = 100\text{ k}\Omega$ , See Figure 1	$C_L = 20\text{ pF}$ , See Figure 1	25°C		55		kHz
			-55°C		80		
			125°C		40		
B <sub>1</sub> Unity-gain bandwidth	$V_I = 10\text{ mV}$ , See Figure 3	$C_L = 20\text{ pF}$	25°C		525		kHz
			-55°C		850		
			125°C		330		
$\phi_m$ Phase margin	$V_I = 10\text{ mV}$ , $C_L = 20\text{ pF}$	$f = B_1$ , See Figure 3	25°C		40°		
			-55°C		44°		
			125°C		36°		

operating characteristics,  $V_{DD} = 10\text{ V}$

PARAMETER	TEST CONDITIONS		$T_A$	MIN	TYP	MAX	UNIT
SR Slew rate at unity gain	$R_L = 100\text{ k}\Omega$ , $C_L = 20\text{ pF}$ , See Figure 1	$V_{ipp} = 1\text{ V}$	25°C		0.62		V/ $\mu$ s
			-55°C		0.81		
			125°C		0.38		
		$V_{ipp} = 5.5\text{ V}$	25°C		0.66		
			-55°C		0.73		
			125°C		0.35		
$V_n$ Equivalent input noise voltage	$f = 1\text{ kHz}$ , See Figure 2	$R_S = 100\ \Omega$	25°C		32		nV/ $\sqrt{\text{Hz}}$
BOM Maximum output swing bandwidth	$V_O = V_{OH}$ , $R_L = 100\text{ k}\Omega$ , See Figure 1	$C_L = 20\text{ pF}$ , See Figure 1	25°C		35		kHz
			-55°C		50		
			125°C		20		
B <sub>1</sub> Unity-gain bandwidth	$V_I = 10\text{ mV}$ , See Figure 3	$C_L = 20\text{ pF}$	25°C		635		kHz
			-55°C		960		
			125°C		440		
$\phi_m$ Phase margin	$V_I = 10\text{ mV}$ , $C_L = 20\text{ pF}$	$f = B_1$ , See Figure 3	25°C		43°		
			-55°C		47°		
			125°C		39°		

5414/7414 Hex Schmitt-Trigger Inverter

	Schottky TTL					High-Speed TTL					Low-Power Schottky TTL					Standard TTL					Low-Power TTL				
	Device Type		Package			Device Type		Package			Device Type		Package			Device Type		Package			Device Type		Package		
			C	P	M	CF			C	P	M	CF			C	P	M	CF			C	P	M	CF	
T.I.																									
FAIRCHILD																									
MOTOROLA																									
N.S.C.																									
PHILIPS																									
SIGNETICS																									
SIEMENS																									
FUJITSU																									
HITACHI																									
mitsubishi																									
NEC																									
TOSHIBA																									

**Electrical Characteristics SN54LS14/SN74LS14**  
absolute maximum ratings over operating free-air temperature range

Supply voltage, V <sub>CC</sub>	7V	Operating free-air temperature range	SN54LS	-55°C to 125°C
Input voltage	7V		SN74LS	0°C to 70°C
		Storage temperature range		-55°C to 150°C

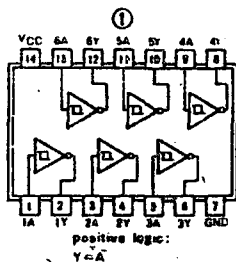
recommended operating conditions

	SN54LS14			SN74LS14			UNIT
	MIN	TYP	MAX	MIN	TYP	MAX	
Supply voltage, V <sub>CC</sub>	4.5	5	5.5	4.75	5	5.25	V
High-level output current, I <sub>OH</sub>			-400			-400	mA
Low-level output current, I <sub>OL</sub>			4			4	mA
Operating free-air temperature, T <sub>A</sub>	-55		125	0		70	°C

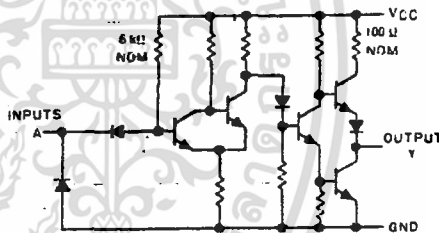
electrical characteristics over recommended operating free-air temperature range

PARAMETER	TEST CONDITIONS †	MIN	TYP ‡	MAX	UNIT	
V <sub>T+</sub>	Positive-going threshold voltage	V <sub>CC</sub> =5V	1.4	1.6	1.9	V
V <sub>T-</sub>	Negative-going threshold voltage	V <sub>CC</sub> =5V	0.5	0.8	1	V
	Hysteresis (V <sub>T+</sub> - V <sub>T-</sub> )	V <sub>CC</sub> =5V	0.4	0.8	V	
V <sub>I</sub>	Input clamp voltage	V <sub>CC</sub> =MIN, I <sub>I</sub> =-18 mA		-1.5	V	
V <sub>OH</sub>	High-level output voltage	V <sub>CC</sub> =MIN, V <sub>I</sub> =V <sub>T-</sub> , min, I <sub>OH</sub> =MAX	2.7	3.4	V	
V <sub>OL</sub>	Low-level output voltage	V <sub>CC</sub> =MIN, V <sub>I</sub> =V <sub>T+</sub> , max, I <sub>OL</sub> =4mA	0.25	0.4	V	
I <sub>T+</sub>	Input current at positive-going threshold	V <sub>CC</sub> =5V, V <sub>I</sub> =V <sub>T+</sub>	-0.14		mA	
I <sub>T-</sub>	Input current at negative-going threshold	V <sub>CC</sub> =5V, V <sub>I</sub> =V <sub>T-</sub>	-0.18		mA	
I <sub>I-</sub>	Input current at maximum input voltage	V <sub>CC</sub> =MAX, V <sub>I</sub> =7V		0.1	mA	
I <sub>IH</sub>	High-level input current	V <sub>CC</sub> =MAX, V <sub>I</sub> =2.7V		20	μA	
I <sub>IL</sub>	Low-level input current	V <sub>CC</sub> =MAX, V <sub>I</sub> =0.4V		-0.4	mA	
I <sub>CS</sub>	Short-circuit output current †	V <sub>CC</sub> =MAX	-20	-100	mA	
I <sub>CC</sub>	Supply current	Total, output high	V <sub>CC</sub> =MAX		6.6	16
		Total, output low	V <sub>CC</sub> =MAX		12	21
		Average per gate	V <sub>CC</sub> =5V, 50% duty cycle	1.72		mA
t <sub>PLH</sub>	Propagation delay time, low-to-high-level output	V <sub>CC</sub> =5V, T <sub>A</sub> =25°C	15	22	ns	
t <sub>PHL</sub>	Propagation delay time, high-to-low-level output	C <sub>L</sub> =15PF, R <sub>L</sub> =2kΩ	15	22	ns	

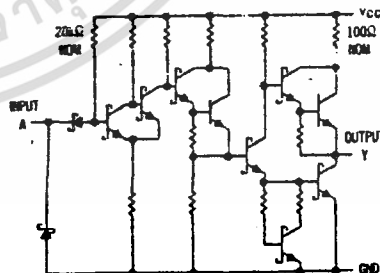
Pin Assignment (Top View)



Schematics (each gate)



'14 CIRCUIT



'LS14 CIRCUIT

Resistor values shown are nominal and in ohms.

† For conditions shown as MIN or MAX, use the appropriate value specified under recommended operating conditions.  
‡ All typical values are at V<sub>CC</sub>=5V, T<sub>A</sub>=25°C.  
§ Not more than one output should be shorted at a time.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่นอนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

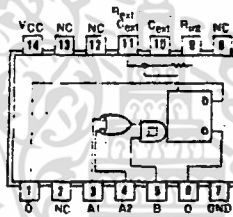
54121/74121 Monostable Multivibrator with Schmitt-Trigger Input

	Schottky TTL				High-Speed TTL				Low-Power Schottky TTL				Standard TTL				Low-Power TTL				
	Device Type	Package			Device Type	Package			Device Type	Package			Device Type	Package			Device Type	Package			
		C	P	M	CF		C	P	M	CF		C	P	M	CF		C	P	M	CF	
T.I.																					
FAIRCHILD																					
MOTOROLA																					
N.S.C.																					
PHILIPS																					
SIGNETICS																					
SIEMENS																					
FUJITSU																					
HITACHI																					
MITSUBISHI																					
NEC																					
TOSHIBA																					

Electrical Characteristics SN54S121 SN74S121

absolute maximum ratings over operating free-air temperature range			
Supply voltage VCC	7V	Derating for air temperature range	50% 50°C to 125°C
Input voltage	5.5V	Derating for air temperature range	50% 0°C to 70°C
Storage temperature range			
-65°C to 150°C			
recommended operating conditions			
	SN54S121	SN74S121	UNIT
	MIN	NOM	MAX
Supply voltage VCC	4.5	5	5.5
High-level output current IOH			40
Low-level output current IOL			16
Rate of rise fall of input pulse dV/dt	1		1
Input pulse width tPW	50		20
External timing capacitor Cext	1.0		1.0
External timing resistor Rext	0	100	100
Output rate	10		10
Operating free-air temperature TA	55	0	70

Pin Assignment (Top view)

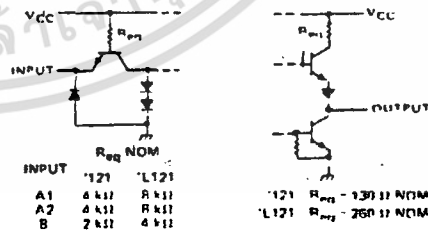


NC-No internal connection  
 \*I21-Rint = 2 kΩ NOM  
 L121-Rint = 4 kΩ NOM

Function Table

INPUTS		OUTPUTS	
A1	A2	Q	Q̄
L	X	H	L
X	L	H	H
X	X	L	L
H	H	X	L
H	L	H	L
L	H	H	L
L	X	L	L
X	L	L	L

Schematics of inputs and outputs



EQUIVALENT OF EACH INPUT. TYPICAL OF BOTH OUTPUTS

- NOTES: A. H=high level (steady state), L=low level (steady state), ↑=transition from low to high level, ↓=transition from high to low level, JL=one high-level pulse, J̄L=one low-level pulse, X=irrelevant (any input, including transitions).
- B. To use the internal timing resistor connect Rint to VCC.
- C. An external timing capacitor may be connected between Cext (positive) and Rext/Cext.
- D. For accurate repeatable pulse widths, connect an external resistor between Rext/Cext and VCC with Rint open-circuited.
- E. To obtain variable pulse widths, connect external values resistance between Rint or Rext/Cext and VCC.

electrical characteristics over recommended operating free-air temperature range							
PARAMETER	TEST CONDITIONS †	MIN	TYP ‡	MAX	UNIT		
V <sub>T+</sub>	Positive-going threshold voltage	VCC=MIN	A input	1.4	2	V	
V <sub>T-</sub>	Negative-going threshold voltage	VCC=MIN	A input	0.8	1.4	V	
V <sub>T+</sub>	Positive-going threshold voltage	VCC=MIN	B input		1.35	2	V
V <sub>T-</sub>	Negative-going threshold voltage	VCC=MIN	B input	0.8	1.35	V	
V <sub>I</sub>	Input clamp voltage	VCC=MIN, I <sub>I</sub> =-12mA			-1.5	V	
V <sub>OH</sub>	High-level output voltage	VCC=MIN, I <sub>OH</sub> =MAX		2.4	3.4	V	
V <sub>OL</sub>	Low-level output voltage	VCC=MIN, I <sub>OL</sub> =MAX		0.2	0.4	V	
I <sub>I</sub>	Input current at maximum input voltage	VCC=MAX, V <sub>I</sub> =5.5V			1	mA	
I <sub>IH</sub>	High-level input current	VCC=MAX, V <sub>I</sub> =2.4V	A1 or A2		40	μA	
I <sub>IH</sub>	High-level input current	VCC=MAX, V <sub>I</sub> =2.4V	B		80	μA	
I <sub>IL</sub>	Low-level input current	VCC=MAX, V <sub>I</sub> =0.4V	A1 or A2		-1.6	mA	
I <sub>IL</sub>	Low-level input current	VCC=MAX, V <sub>I</sub> =0.4V	B		-3.2	mA	
I <sub>OS</sub>	Short-circuit output current †	VCC=MAX	54 Family	-20	-55	mA	
I <sub>OS</sub>	Short-circuit output current †	VCC=MAX	74 Family	-18	-55	mA	
I <sub>CC</sub>	Supply current	VCC=MAX	Quiescent	13	25	mA	
I <sub>CC</sub>	Supply current	VCC=MAX	Triggered	23	40	mA	
t <sub>PLH</sub>	Propagation delay time, low-to-high-level Q output from either A input	VCC=5V, TA=25°C, CL=15pF, RL=400Ω		45	70	ns	
t <sub>PLH</sub>	Propagation delay time, low-to-high-level Q output from B input	VCC=5V, TA=25°C, CL=15pF, RL=400Ω		35	55	ns	
t <sub>PHL</sub>	Propagation delay time, high-to-low-level Q output from either A input	VCC=5V, TA=25°C, CL=15pF, RL=400Ω		50	80	ns	
t <sub>PHL</sub>	Propagation delay time, high-to-low-level Q output from B input	VCC=5V, TA=25°C, CL=15pF, RL=400Ω		40	65	ns	
t <sub>w(out)</sub>	Pulse width obtained using internal timing resistor	VCC=5V, TA=25°C, CL=15pF, RL=400Ω		70	110	150	ns
t <sub>w(out)</sub>	Pulse width obtained with zero timing capacitance	VCC=5V, TA=25°C, CL=0, RL=400Ω		30	50	ns	
t <sub>w(out)</sub>	Pulse width obtained using external timing resistor	CT=100pF, RT=10kΩ		600	800	800	ns
t <sub>w(out)</sub>	Pulse width obtained using external timing resistor	CT=1μF, RT=10kΩ		6	7	8	ms

† For conditions shown as MIN or MAX, use the appropriate value specified under recommended operating conditions.

‡ All typical values are at VCC=5V, TA=25°C

• Not more than one output should be shorted at a time.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่นอนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
 ไม่สามารถใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

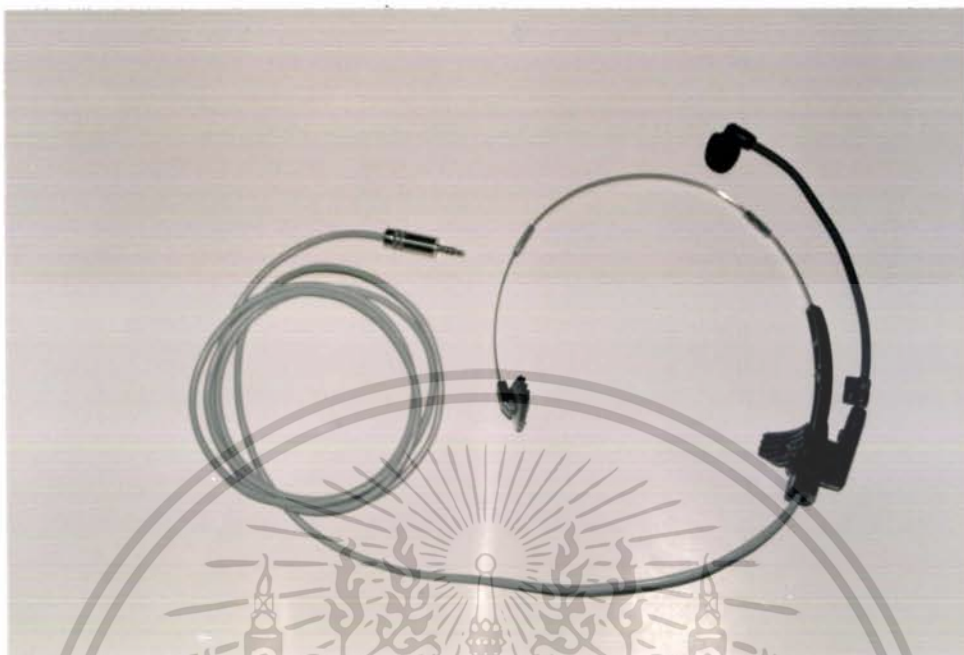


รูปประกอบที่ 1 เครื่องแปลงระดับความถี่เสียง ไปเป็นข้อมูลmidi



รูปประกอบที่ 2 แสดงคอนเนคเตอร์ MIDI OUT ทางด้านหลังของเครื่อง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่นิยญาติให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

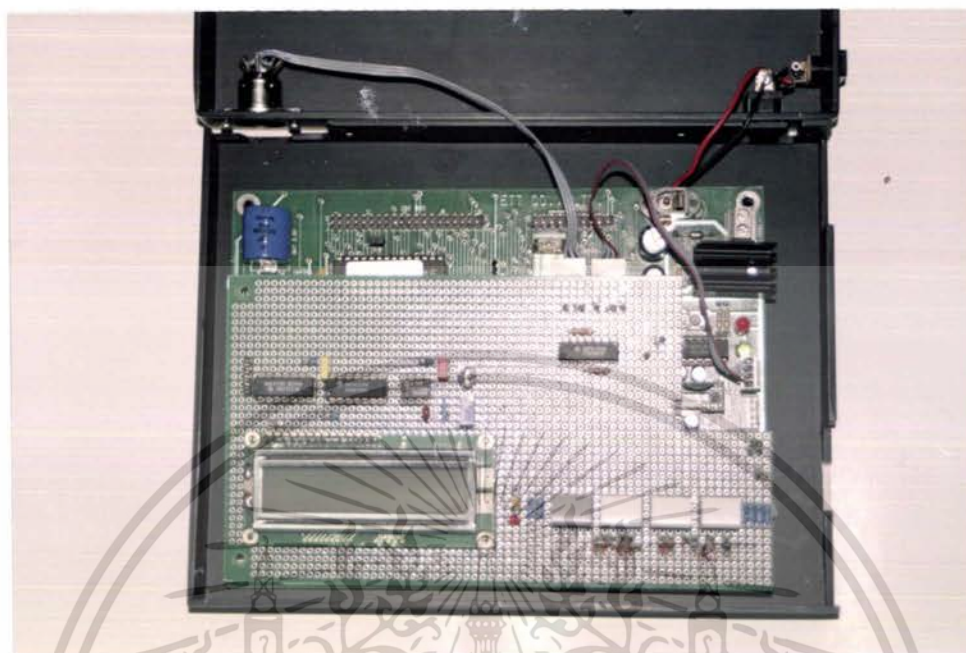


รูปประกอบที่ 3 ไมโครโฟนแบบ HEADSET ที่ใช้กับเครื่องแปลงระดับความถี่เสียง ไปเป็นข้อมูลดิจิทัล



รูปประกอบที่ 4 สายเชื่อมต่อข้อมูลทางมิดี (MIDI cable)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปประกอบที่ 5 แสดงส่วนประกอบภายในของเครื่องแปลงระดับความถี่เสียง-  
ไปเป็นข้อมูลทางนิต



รูปประกอบที่ 6 แสดงการใช้งานร่วมกับ Synthesizer KAWAI รุ่น K1 II  
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่นิยญาติให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1. Craig Anderton , MIDI for Musicians. (Howard Jones ed.)  
Amsco publications, New York, 1986.
2. Dominic Milano , MIND OVER MIDI , GPI Publications,  
California, 1987.
3. ทีมงาน ETT , Z80180 MPU USER MANUAL , พิมพ์ครั้งที่ 1 , ETT ,  
พ.ศ. 2533
4. ทีมงาน ETT , Dot Matrix LCD Module , พิมพ์ครั้งที่ 1 , ETT ,  
พ.ศ. 2533
5. William Barden, Jr. , The Z-80 Microcomputer Handbook ,  
Howard W.Sams & Co., Inc., Indianapolis , 1978.
6. Paul Horowitz , Winfield Hill , THE ART OF ELECTRONICS  
Second Edition , Cambridge University Press , NEW YORK ,  
1989.
7. Richard E.Berg , David G.Stork , The Physics of Sound ,  
Prentice-Hall International Editions , New Jersey , 1982.
8. Charles Dodge , Thomas A.Jerse , COMPUTER MUSIC SYNTHESIS  
, COMPOSITION, AND PERFORMANCE , Schirmer Books A  
Division of Macmillan, Inc. , NEW YORK , 1985.
9. Texas Instruments Incorporated , Linear Circuit Data Book  
1989 Volume. 1 , Texas Instruments Incorporated , DALLAS  
TEXAS , 1989.
10. คู่มือเทียบเบอร์ไอซี TTL , พิมพ์ครั้งที่ 1 , บริษัท ซีเอ็ดยูเคชั่น จำกัด ,  
พ.ศ. 2521

นาย พงศ์สิริ อ่อนศรี เกิดวันที่ 10 กรกฎาคม พ.ศ.2512 สำเร็จการศึกษา  
ระดับมัธยมศึกษาตอนต้นและระดับมัธยมศึกษาตอนปลายจากโรงเรียนสามเสนวิทยาลัย แล้ว  
เข้าศึกษาต่อในระดับปริญญาตรีสาขาฟิสิกส์ประยุกต์ คณะวิทยาศาสตร์ สถาบันเทคโนโลยี  
พระจอมเกล้าเจ้าคุณทหารลาดกระบัง และสำเร็จการศึกษาในปีการศึกษา 2534

นาย เพ็ญภูมิ เนียมลาภเนื่อง เกิดวันที่ 28 กันยายน พ.ศ.2510 สำเร็จ  
การศึกษาระดับมัธยมศึกษาตอนต้น และระดับมัธยมศึกษาตอนปลายจากโรงเรียนสวนกุหลาบ  
วิทยาลัย แล้วเข้าศึกษาต่อในระดับปริญญาตรีสาขาฟิสิกส์ประยุกต์ คณะวิทยาศาสตร์  
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง และสำเร็จการศึกษาใน  
ปีการศึกษา 2534



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้