สำนักหอสมุดกลาง พระจอมเกล้าลาดกระบัง

สตอเรจออสซิลโลสโคป

STORAGE OSCILLOSCOPE

T 0 4 2 7 4 2

โดย

นายพุทธมนต์ ภิญโญดม

นายมารุต ศิริวังโส

เลขหมู่.....
เลขทะเบียน 42742
วัน, เดือน, ปี 7 มิ.ย. 2545

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

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรวิศวกรรมศาสตร์บัณฑิต

สาขาวิชาอิเล็กทรอนิกส์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2543

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สตอเรจออสซิลโลสโคป

STORAGE OSCILLOSCOPE

โดย

นายพุทธมนต์ ภิญโญดม 40010531

นายมารุต ศิริวังโส 40010600

อาจารย์ที่ปรึกษา

รศ.ดร. มนัส สังวรศิลป์

ปริญญานิพนธ์สำหรับปริญญาวิศวกรรมศาสตรบัณฑิต

สาขาวิชาอิเล็กทรอนิกส์

คณะวิศวกรรมศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2543

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ปริญญานิพนธ์ปีการศึกษา 2543

ภาควิชา อิเล็กทรอนิกส์

คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เรื่อง สตอเรจออสซิลโลสโคป

ผู้จัดทำ

1. นายพุทธมนต์ ภิญโญดม 40010531
2. นายมารุต ศิริวังโส 40010600

....อาจารย์ที่ปรึกษา

(รศ.ดร.มนัส สังวรศิลป์)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สตอเรจ ออสซิลโลสโคป

STORAGE OSCILLOSCOPE

นายพุทธมนต์ ภิญโญคม 40010531

นายมารุต ศิริวังโส 40010600

โครงงานได้รับการตรวจสอบแล้ว พร้อมที่จะทำการสอบได้

....

(รศ. ดร.มนัส สังวรศิลป์)

อาจารย์ที่ปรึกษา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สตอเรจออสซิลโลสโคป

นายพุทธมนต์ ภิญโญคม
นายมารุต ศิริวังโส
รศ.ดร.มนัส สังวรศิลป์ อาจารย์ที่ปรึกษา
ปีการศึกษา 2543

บทคัดย่อ

สตอเรจออสซิลโลสโคปที่นำเสนอในโครงงานนี้ เป็นการนำคอมพิวเตอร์มาประยุกต์ใช้งานประกอบด้วยฮาร์ดแวร์ที่ทำงานในลักษณะของดิจิตอลออสซิลโลสโคป โดยทำการการสุ่มและแปลงสัญญาณที่วัดได้จากสัญญาณอนาลอกเป็นสัญญาณดิจิตอล นำสัญญาณดิจิตอลที่ได้ไปทำการประมวลผลและแสดงผลออกทางจอภาพ สามารถบันทึกรูปสัญญาณต่าง ๆ มีการอินเตอร์เฟสผ่าน Isa Bus และใช้ซอฟต์แวร์ภาษาDelphi ในส่วนการควบคุมจากผู้ใช้และการแสดงผล ประโยชน์ที่ได้จากการนำคอมพิวเตอร์มาประยุกต์ใช้ในงานลักษณะนี้คือการประหยัดต้นทุนในการซื้อสตอเรจออสซิลโลสโคปซึ่งมีราคาแพงได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

STORAGE OSCILLOSCOPE

Puttamon Pinyodom

Marut Siriwangso

Assoc.Prof.Dr.Manas Sangwornsin Advisor

Education Year 2000

ABSTRACT

Storage Oscilloscope, presented in this project, apply to use by the computer consist of hardware and software. Hardware, that work like digital oscilloscope for sampling and converting signal that measure in analog to digital signal. Afterward, the digital signal is processed and displayed to a monitor.

Storage Oscilloscope in this project interfaces with ISA bus. In the part of controlling and displaying use Delphi software. The benefit from this project is saving a cost, spend for the expensive Storage Oscilloscope.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ

	หน้า
บทคัดย่อ	I
ABSTRACT	II
สารบัญ	III
สารบัญรูป	VI
สารบัญตาราง	VIII
บทที่ 1 บทนำ	1
บทที่ 2 ทฤษฎีการรับและการแปลงข้อมูล (Data Acquisiton and Conversion Theory)	2
2.1 ทฤษฎีการสุ่ม(Sampling Theory)	3
2.2 Aperture time error	4
2.3 Sample and Hold	4
2.4 Frequency folding and Aliasing	6
2.5 วงจรแปลงสัญญาณอนาลอกเป็นสัญญาณดิจิตอล (Analog to Digital Converter)	7
บทที่ 3 หลักการของดิจิตอลสตอเรจออสซิลโลสโคป	9
3.1 การทำงานของดิจิตอลสตอเรจออสซิลโลสโคป (DSO)	9
3.2 ขอบเขตการวัด	10
3.3 การสุ่มตัวอย่างของดิจิตอลสตอเรจออสซิลโลสโคป	16
3.3.1 การสุ่มตัวอย่างแบบเวลาจริง (Real Time or One-shot)	16
3.3.2 การสุ่มตัวอย่างแบบเวลาเสมือน (Equivalent Time)	17
3.4 ขั้นตอนการสุ่มตัวอย่าง	19
3.5 การเก็บข้อมูลของดิจิตอลสตอเรจออสซิลโลสโคป	20
3.6 ปัญหาพื้นฐานของดิจิตอลสตอเรจออสซิลโลสโคป	20
3.7 ความละเอียดถูกต้องแม่นยำ	20
3.8 สัญญาณชั่วขณะ (Transient capture)	22
3.9 แบนวิดธ์และอัตราการสุ่มตัวอย่าง (Bandwidth and Sample rate)	24
3.10 ประโยชน์ของหน่วยความจำที่ยาวในดิจิตอลสตอเรจออสซิลโลสโคป	25
3.10.1 การเก็บรายละเอียด	26
3.10.2 การจับสัญญาณกลิทช์ถาวร	27

เอกสารนี้เป็นเอกสารที่ระบบได้จัดทำเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ในการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หน้า

3.10.3 การแก้ปัญหาเรื่องความถี่เวลา 27

3.10.4 ความเชื่อถือได้ในการจับสัญญาณที่ไม่สามารถทำนายได้ 27

3.10.5 ไม่มีการสูญเสียเวลาระหว่างการรับข้อมูล 27

บทที่ 4 สล็อต ISA ภายใน IBM/PC 29

4.1 รายละเอียดของสัญญาณที่ขาต่าง ๆ ของสล็อต ISA 31

4.2 บัสแหล่งจ่ายไฟของระบบ 38

บทที่ 5 หลักการออกแบบวงจรและโปรแกรมควบคุมการทำงาน 39

5.1 โครงสร้างของสตอเรจออสซิลโลสโคป 39

5.2 ส่วนปรับแต่งขนาดและยกระดับสัญญาณอินพุท 39

5.2.1 วงจรลดทอนสัญญาณ (Attenuator) 40

5.2.2 วงจรบัฟเฟอร์ (Buffer) 40

5.2.3 วงจรขยายสัญญาณแนวตั้ง (Vertical Amplifier) 40

5.2.4 วงจรยกระดับสัญญาณ (DC Adjust) 40

5.3 วงจรแปลงสัญญาณอนาลอกเป็นสัญญาณดิจิตอล (A/D Converter) 45

5.4 ส่วนกำเนิดสัญญาณนาฬิกาสำหรับการสุ่ม (Clock Generator) 45

5.5 วงจรถอดรหัสตำแหน่งพอร์ตและวงจรเชื่อมต่อพอร์ต 48

5.6 วงจรนับตำแหน่งหน่วยความจำ 48

5.7 วงจรหน่วยความจำ 48

5.8 โปรแกรมควบคุมการทำงาน 53

5.9 หลักการเขียนโปรแกรมควบคุมการทำงาน 53

5.10 การเขียนโปรแกรม Interface กับ ISA Bus 54

5.11 การควบคุมและการใช้งานการ์ด 55

บทที่6 การทดลอง 57

6.1ผลการทดลองวัดสัญญาณอินพุทที่เป็นรูปซายน์ที่ความถี่ต่าง ๆ 58

6.2ผลการทดลองวัดสัญญาณอินพุทที่เป็นรูปสี่เหลี่ยมที่ความถี่ต่าง ๆ 60

6.3ผลการทดลองวัดสัญญาณอินพุทที่เป็นสามเหลี่ยมที่ความถี่ต่าง ๆ 62

บทที่7 สรุปผลการทำโครงงาน 65

7.1 ปัญหาที่พบในการทำโครงงาน 65

7.2 แนวทางการแก้ไขปัญหา 66

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หน้า

7.3 ประโยชน์ที่ได้รับจากการทำโครงงานนี้ 66
ภาคผนวก
กิตติกรรมประกาศ
บรรณานุกรม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูป

หน้า

รูปที่ 2.1 ระบบควบคุมที่มีการประมวลข้อมูลทางดิจิตอล 2

รูปที่ 2.2 วิธีการพื้นฐานของ ADC 3

รูปที่ 2.3 ทรานเฟอร์ฟังก์ชันของคอมพาราเตอร์ 3

รูปที่ 2.4 แสดงความผิดพลาดจากการวัดใน Aperture Time 4

รูปที่ 2.5 แสดงการสุ่มสัญญาณ 5

รูปที่ 2.6 แสดง Spectrum ของสัญญาณอนาลอกที่ถูกสุ่ม 6

รูปที่ 2.7 หลังจากการสุ่มเกิด Frequency folding 6

รูปที่ 2.8 การเกิด Alias Frequency folding จากการสุ่มด้วยความถี่ต่ำกว่า 2 เท่าของสัญญาณอนาลอก 7

รูปที่2.9 การต่อวงจร Parallel Comparator A/D Converter 8

รูปที่ 3.1 แสดงบล็อกไดอะแกรมของดิจิตอลสตอเรจออสซิลโลสโคป (DSO) 10

รูปที่ 3.2 แสดงลักษณะของการเกิดเอเลียสซิ่ง 12

รูปที่ 3.3 ทฤษฎีการสอดแทรก 13

รูปที่ 3.4 แสดงการจัดช่วงขอบขาขึ้นซึ่งค่าคลาดเคลื่อนที่เกิดขึ้นอยู่กับตำแหน่งในการสุ่มตัวอย่าง 15

รูปที่ 3.5 เทคนิคการสุ่มตัวอย่างแบบต่าง ๆ ของดิจิตอลสตอเรจออสซิลโลสโคป 17

รูปที่ 3.6 แสดงให้เห็นถึงผลที่ออกมาจากการสุ่มตัวอย่างสัญญาณอินพุทแบบเป็นลำดับ 18

รูปที่ 3.7 การควอนไตซิ่งโดยใช้วงจรแปลงสัญญาณอนาลอกเป็นสัญญาณดิจิตอลคอนเวอร์เตอร์เปลี่ยนแรงดันอนาลอกเป็นเลขฐานสองที่ระดับต่างๆ 19

รูปที่ 3.8 บล็อคไดอะแกรมของ RIS ดิจิไตเซอร์ 23

รูปที่ 3.9 รูปคลื่นออสซิลโลสโคปที่มีหน่วยความจำ 2.5 M 25

รูปที่ 3.10 รูปคลื่นออสซิลโลสโคปที่มีหน่วยความจำ 50 k 26

รูปที่ 3.11 รูปสัญญาณเบิร์สต่อเนื่อง 28

รูปที่ 3.12 แสดงสัญญาณที่มีช่วงเวลาระหว่างสัญญาณเบิร์สที่ยาวกว่า 28

รูปที่ 4.1 แสดงการอินเตอร์เฟสบนสล็อต ISA 8 บิต 29

รูปที่ 4.2 แสดงการอินเตอร์เฟสบนสล็อต ISA 16 บิต 30

รูปที่ 4.3 ขาสัญญาณต่าง ๆ ของ ISA bus 31

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หน้า

รูปที่ 5.1 แสดงบล็อกไดอะแกรมของดิจิตอลสตอเรจออสซิลโลสโคป	39
รูปที่ 5.2 วงจรลดทอนสัญญาณ	41
รูปที่ 5.3 วงจรบัฟเฟอร์	42
รูปที่ 5.4 วงจรขยายสัญญาณ	43
รูปที่ 5.5 วงจรยกระดับสัญญาณกระแสตรง	44
รูปที่ 5.6 วงจรแปลงสัญญาณอนาลอกเป็นดิจิตอล	46
รูปที่ 5.7 วงจรกำเนิดสัญญาณนาฬิกาหลายความถี่	47
รูปที่ 5.8 วงจรถอดรหัสตำแหน่งพอร์ต	49
รูปที่ 5.9 วงจรเชื่อมต่อพอร์ต ISA กับวงจรต่างๆ	50
รูปที่ 5.10 วงจรนับตำแหน่งหน่วยความจำ	51
รูปที่ 5.11 วงจรหน่วยความจำ	52
รูปที่ 5.12 แสดงผังงานของโปรแกรมควบคุม	53
รูปที่ 5.13 แสดงผังงานของการควบคุมฮาร์ดแวร์ เมื่อทำการวัดสัญญาณ	56
รูปที่ 6.1 หน้าจอควบคุม Storage Oscilloscope	57
รูปที่ 6.2 สัญญาณรูปซายน์ที่ความถี่ 500 Hz ขนาด 0.5 Vp(1Vpp)	58
รูปที่ 6.3 สัญญาณรูปซายน์ที่ความถี่ 20 kHz ขนาด 0.5 Vp(1Vpp)	59
รูปที่ 6.4 สัญญาณรูปซายน์ที่ความถี่ 100 kHz ขนาด 0.5 Vp(1Vpp)	59
รูปที่ 6.5 สัญญาณรูปซายน์ที่ความถี่ 500 kHz ขนาด 0.5 Vp(1Vpp)	60
รูปที่ 6.6 สัญญาณรูปสี่เหลี่ยมที่ความถี่ 4 kHz ขนาด 0.5 Vp(1Vpp)	61
รูปที่ 6.7 สัญญาณรูปสี่เหลี่ยมที่ความถี่ 20 kHz ขนาด 0.5 Vp(1Vpp)	61
รูปที่ 6.8 สัญญาณรูปสี่เหลี่ยมที่ความถี่ 100 kHz ขนาด 0.5 Vp(1Vpp)	62
รูปที่ 6.9 สัญญาณรูปสามเหลี่ยมที่ความถี่ 4 kHz ขนาด 0.5 Vp(1Vpp)	63
รูปที่ 6.10 สัญญาณรูปสามเหลี่ยมที่ความถี่ 20 kHz ขนาด 0.5 Vp(1Vpp)	63
รูปที่ 6.11 สัญญาณรูปสามเหลี่ยมที่ความถี่ 100 kHz ขนาด 0.5 Vp(1Vpp)	64

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญตาราง

	หน้า
ตารางที่ 2.1 ตารางความสัมพันธ์ระหว่างแรงดันอินพุทที่เป็นอนาลอกกับเอาท์พุทที่เป็นดิจิตอล	8
ตารางที่ 5.1 อัตราขยายของวงจรขยายสัญญาณ	40
ตารางที่ 5.2 ความถี่ของวงจรกำเนิดสัญญาณ	45
ตารางที่ 6.1 แสดงผลการทดลองเมื่ออินพุทเป็นรูปซายน์ที่ความถี่ต่าง ๆ	58
ตารางที่ 6.2 แสดงผลการทดลองเมื่ออินพุทเป็นรูปสี่เหลี่ยมที่ความถี่ต่าง ๆ	60
ตารางที่ 6.3 แสดงผลการทดลองเมื่ออินพุทเป็นรูปสามเหลี่ยมที่ความถี่ต่าง ๆ	62

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

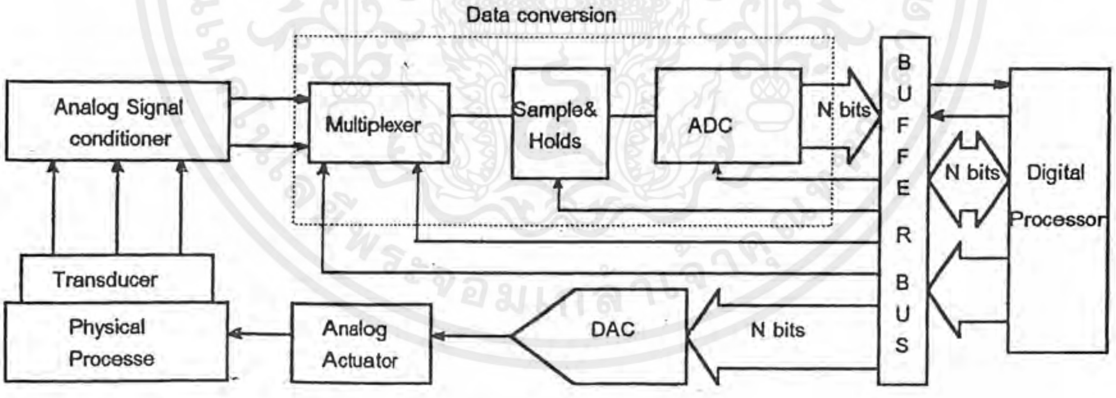
บทที่ 1
บทนำ

ออสซิลโลสโคป(Oscilloscope)เป็นเครื่องมือวัดที่สำคัญอย่างมาก ไม่ว่าจะเป็นการวิเคราะห์การซ่อมแซมวงจรอิเล็กทรอนิกส์ หรือการเก็บข้อมูลบางอย่างไว้อ้างอิง และอีกมากมาย ในปัจจุบันเครื่องออสซิลโลสโคปรุ่นใหม่ที่เก็บสัญญาณหรือบันทึกข้อมูล พร้อมกับยังมีฟังก์ชันพิเศษต่าง ๆ ในโรงงานอุตสาหกรรมได้นำเอาคอมพิวเตอร์เข้ามาควบคุมเครื่องจักรเพื่อลดการผิดพลาด ลดเวลาการสูญเสียต่าง ๆ และยังสามารถเพิ่มประสิทธิภาพการทำงานได้สูงขึ้น จากข้อดีของออสซิลโลสโคปและคอมพิวเตอร์นี้ จึงเป็นที่เริ่มของโครงงานนี้ที่จะสร้างการ์ดใช้งานร่วมกับคอมพิวเตอร์ให้เป็นออสซิลโลสโคปแสดงผลหน้าจอที่คอมพิวเตอร์ โดยได้ตัดบางฟังก์ชันออกไปเพราะบางฟังก์ชันไม่ได้ถูกใช้งาน นำไปบันทึกสัญญาณและวิเคราะห์ เช่น นำไปบันทึกสัญญาณของเครื่องตรวจสอบไอซี โดยการตรวจสอบสัญญาณบางสัญญาณว่าเครื่องตรวจสอบไอซีสามารถตรวจสอบสัญญาณ โดยที่ยังอยู่ในคุณสมบัติที่ต้องการ เพื่อที่จะได้ผลผลิตที่มีคุณภาพหรือป้องกันของเสียหรือนำไปใช้ SPC (Statistic Process Control) ทำให้เกิดประโยชน์สูงสุดในการใช้คอมพิวเตอร์ ส่วนเครื่องออสซิลโลสโคปก็นำไปใช้งานอย่างอื่น ซึ่งจะเป็นการลดค่าใช้จ่ายในการที่จะซื้อออสซิลโลสโคปหลายตัวมาใช้งาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 2
ทฤษฎีการรับและการแปลงข้อมูล

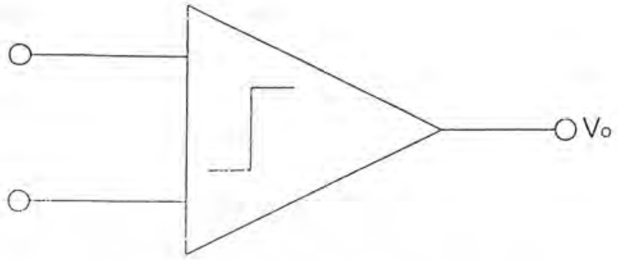
รูปร่างสัญญาณทางไฟฟ้าที่เราพบเห็นในชีวิตประจำวันจะอยู่ในรูปแบบของสัญญาณที่ต่อเนื่องหรือเรียกว่าสัญญาณอนาลอก ซึ่งแต่เดิมการนำเอาสัญญาณไฟฟ้าดังกล่าวมาประมวลผล (Process) ใช้ในรูปแบบที่มีประโยชน์จะกระทำในแบบอนาลอก แต่เมื่อเทคนิคการประมวลผลสัญญาณทางดิจิตอลได้รับการพัฒนาขึ้นมา พบว่าในรูปแบบสัญญาณดิจิตอลการประมวลผลกระทำได้ง่ายและมีประสิทธิภาพมากกว่า การเปลี่ยนรูปแบบของสัญญาณ (Conversion) จึงได้มีความจำเป็นขึ้น จากสัญญาณอนาลอกที่มีอยู่ตามธรรมชาติถูกเปลี่ยนมาเป็นสัญญาณดิจิตอล โดยการแปลงสัญญาณอนาลอกเป็นสัญญาณดิจิตอล(Analog to Digital Convertor :A/D) และประมวลผลโดยตัวประมวลผลทางดิจิตอล (Digital Processor) เช่น คอมพิวเตอร์ จากนั้นจะถูกนำมาแสดงหรือ ถูกเปลี่ยนกลับมาอยู่ในแบบอนาลอก และใช้งานได้ง่ายกว่าโดยการแปลงสัญญาณดิจิตอลเป็นสัญญาณอนาลอก(Digital to Analog Converter : D/A)



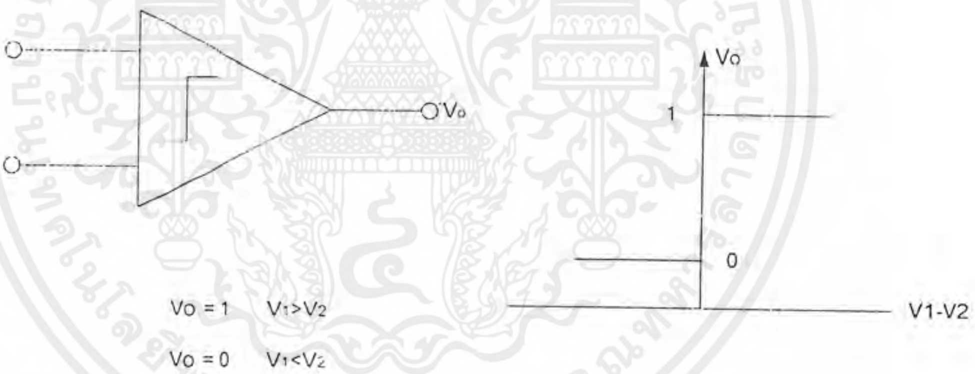
รูปที่ 2.1 แสดงระบบควบคุมที่มีการประมวลข้อมูลแบบดิจิตอล

วิธีการแปลงสัญญาณอนาลอกเป็นดิจิตอลแบบง่าย ๆ แสดงในรูป 2.2 โดยใช้หลักของวงจรคอมพาราเตอร์ แรงดันอินพุทที่ไม่ทราบค่า V_X จะต่อเข้ากับขาอินพุทขาหนึ่งของอนาลอกคอมพาราเตอร์ และแรงดันอ้างอิงที่ขนาดแปรตามเวลา V_R ต่อเข้ากับอีกอินพุทหนึ่งของคอมพาเรเตอร์ ลักษณะของทรานเฟอร์ฟังก์ชันของคอมพาราเตอร์ แสดงในรูป 2.3 ถ้าแรงดันอินพุท V_1 มากกว่า V_2 แล้วแรงดันเอาท์พุทจะเป็น “1” ถ้าอินพุท V_1 น้อยกว่า V_2 แล้วเอาท์พุทจะเป็นศูนย์

เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์ให้บริการใช้งานเพื่อการศึกษาเท่านั้นไม่ให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.2 แสดงวิธีการพื้นฐานของ ADC



รูปที่ 2.3 แสดงทรานเฟอร์ฟังก์ชันของคอมพาเรเตอร์

2.1 ทฤษฎีการสุ่ม (Sampling theory)

ในการแปลงสัญญาณอนาลอกเป็นสัญญาณดิจิตอล เบื้องต้นต้องมีการสุ่ม (Sampling) ก่อน ซึ่งความถี่ในการสุ่มโดยไม่ทำให้สัญญาณสูญเสียข้อมูลสำคัญไปนั้น ทฤษฎีการสุ่มของ แชนนอน (Shannon) กล่าวไว้ว่า

ถ้าหากเรามีสัญญาณอนาลอกที่ค่าการแปลงฟูริเยอร์หรือสเปกตรัมกำลังของมันมีแถบความถี่ปฏิบัติงาน(Band width) เท่ากับ f_0 แล้ว เราสามารถทำการสุ่มโดยสัญญาณที่ได้ไม่สูญเสียเนื้อ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หาสำคัญก็ต่อเมื่อความถี่ในการสุ่มตัวอย่าง f_s มีค่ามากกว่าหรือเท่ากับสองเท่าของความถี่f_0หรือ

$$f_s \geq 2f_0 \quad (2.1)$$

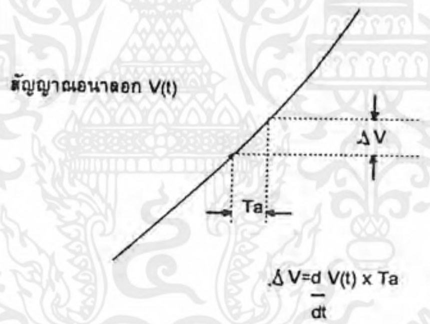
โดยทั่วไปเราอาจสุ่มด้วยค่าความถี่ $f_{SN} = 2f_0$ พอดี เรียกค่าความถี่นี้ว่า ความถี่ไนควิสต์ (Nyquist frequency) และคาบเวลา $T_n = 1 / (2f_0)$ นี้เรียกว่า ช่วงเวลาสุ่มตัวอย่างไนควิสต์ (Nyquist interval)

2.2 Aperture time Error

Aperture time คือช่วงเวลาในการแปลงสัญญาณ ซึ่งคำว่า Aperture time โดยทั่วไป หมายถึง ช่วงเวลาที่เกิดความไม่แน่นอนในการวัดและผลก็คือเกิดความผิดพลาด (Error) ต่อค่าที่วัดได้

ในรูปที่ 2.4 สัญญาณอนาลอก V(t) มีอัตราการเปลี่ยน dV/dt ช่วง Aperture Time t_a ดังนั้น ช่วงการเปลี่ยนแปลงสัญญูยณอนาลอกจะเท่ากับ ΔV โดย

$$\Delta V = t_a \, dV(t)/dt \quad (2.2)$$



รูปที่ 2.4 แสดงการผิดพลาดจากการวัดในAperture Time

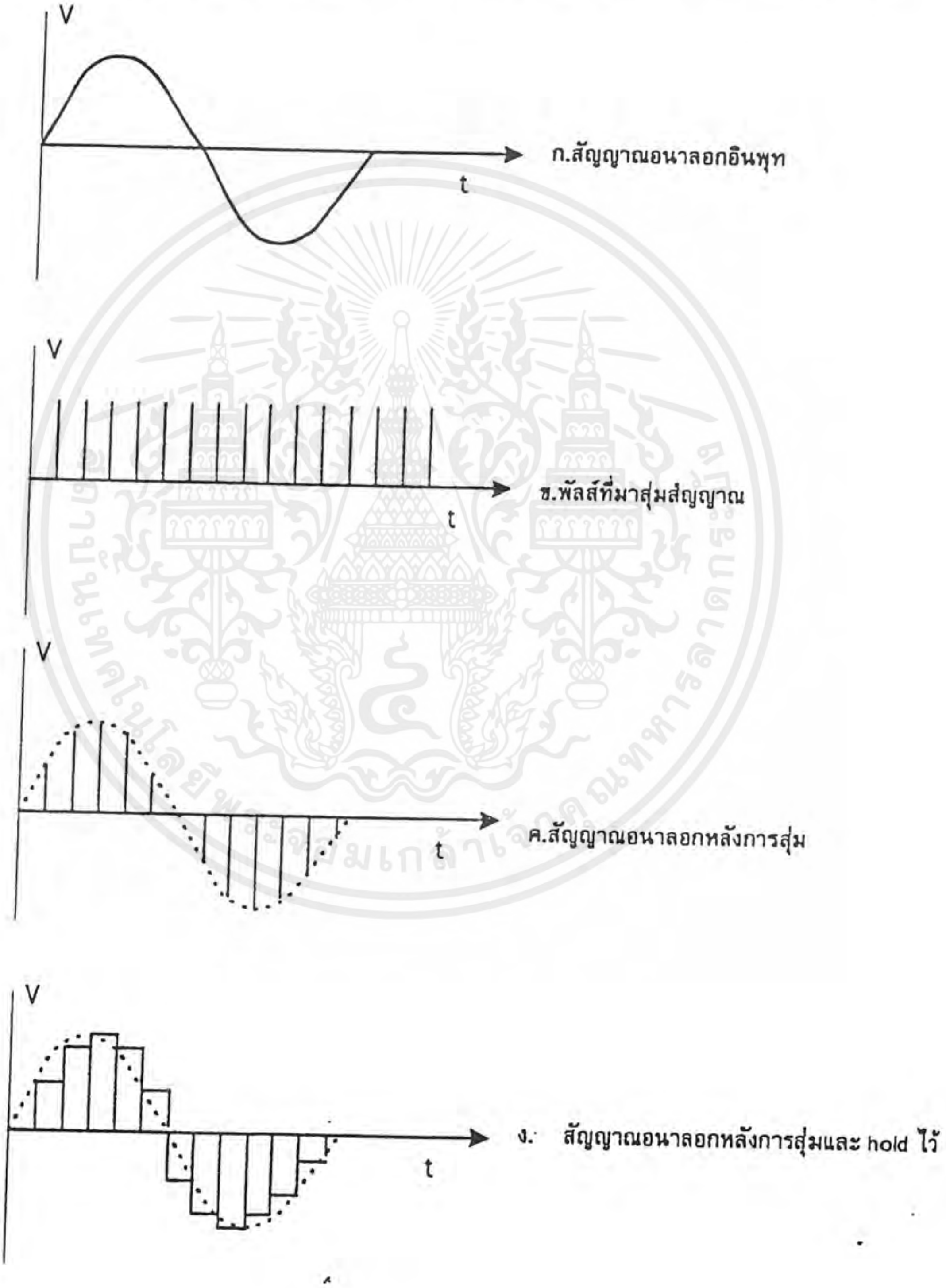
ดังนั้นหากเวลาที่ A/D ใช้ในการเปลี่ยนสัญญาณในเวลา t_a นี้รหัสของสัญญาณดิจิตอลที่ได้อาจจะตรงกับขนาดของสัญญาณอนาลอกค่าใดค่าหนึ่งในช่วงนี้ และส่วนอื่น ๆ ที่เหลือคือความผิดพลาดที่เกิดขึ้น ซึ่งแน่นอนในบางครั้งเป็นไปได้ที่รหัสของสัญญาณดิจิตอลจะตรงกับค่าของสัญญาณอนาลอกที่ถูกต้อง

2.3 Sample and Hold

วงจร Sample and Hold จะทำการสุ่มสัญญาณอินพุท และนำสัญญาณนั้นมาเก็บ (Hold) ไว้ในเวลาหนึ่ง ซึ่งส่วนใหญ่จะใช้การประจุแรงดันนั้นในการเก็บประจุที่รั่วไหลต่ำดังนั้น ในเมื่อ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

แรงดันอินพุทสามารถคงอยู่ได้นานพอ ทำให้ A/D ไม่จำเป็นต้องมีเวลาในการแปลง (Conversion Time) อย่างรวดเร็วนัก Aperture Time ของ Sample and Hold คือเวลาตั้งแต่เริ่มสุ่มสัญญาณจนตัวเก็บประจุมีค่าแรงดันจนถึงค่าที่สุ่ม ในการสุ่มสัญญาณอนาลอกจะถูกสุ่มเป็นระยะ ๆ คงที่ ดังรูปที่2.5

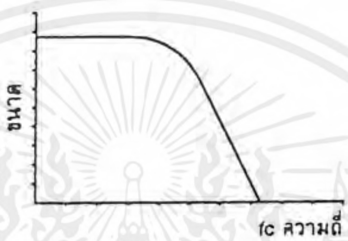


รูปที่ 2.5 แสดงการสุ่มสัญญาณ

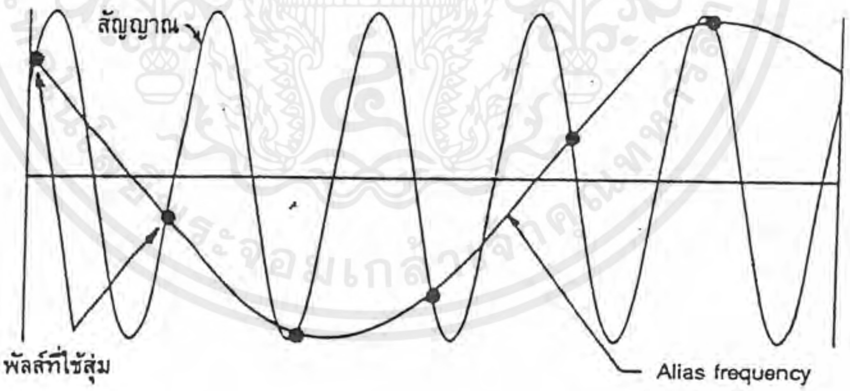
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.4 Frequency folding and Aliasing

จากทฤษฎีการสุ่มสามารถอธิบายลักษณะรูปสเปคตรัม (Spectrum) ของสัญญาณในรูปที่ 2.5 แสดงให้เห็นสเปคตรัมของสัญญาณที่ถูกสุ่มซึ่งแบนด์วิดธ์ไม่เกินกว่า f_c ในขณะที่สัญญาณนี้จะถูกสุ่มด้วยความถี่ f_s ขบวนการมอดูเลชั่น (Modulation) จะทำให้แถบสเปคตรัมของสัญญาณสุ่มขยายกว้างออกจาก f_s เป็น $2f_s$, $3f_s$,...



รูปที่ 2.6 แสดง Spectrum ของสัญญาณอนาลอกที่ถูกสุ่ม



รูปที่ 2.7 หลังจากการสุ่มเกิด Frequency folding

ถ้าความถี่ของสัญญาณสุ่ม fs ไม่สูงพอหลังจากการสุ่มสัญญาณสเปคตรัมบางส่วนของ fs จะมาซ้อนทับกับสเปคตรัมของสัญญาณ ซึ่งเรียกว่า Frequency folding หากเป็นเช่นนี้ก็จะทำให้เกิดความเพี้ยนแก่สัญญาณอนาลอกจากการซ้อนทับกันของสเปคตรัม เมื่อสัญญาณถูกเปลี่ยนกลับ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

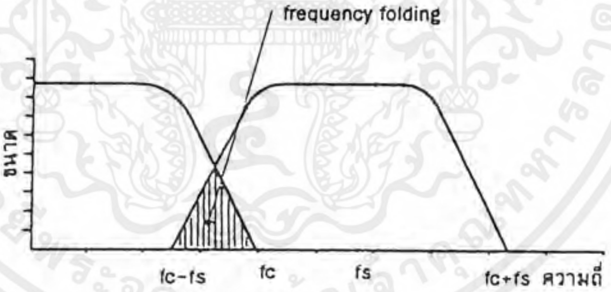
ให้อยู่ในรูปเดิม และถ้าเลื่อนความถี่ของการสุ่มสูงขึ้นจนโอกาสการซ้อนทับกันของสเปคตรัมหมดไป $(f_s - f_c) = f_c$ จะทำให้การเปลี่ยนกลับของสัญญาณหลังจากถูกสุ่มก็ยังคงเหมือนเดิมได้

จากการกล่าวมาแสดงการสนับสนุนทฤษฎีการสุ่มที่ว่าให้ $f_s < 2f_c$ นั่นคือการกำจัดการซ้อนกันของสเปตรัม ได้สองวิธีคือ

1. ใช้อัตราการสุ่มที่สูงพอ
2. การทำการกรองความถี่ของสัญญาณอนาลอก ก่อนการสุ่มเพื่อให้ Bandwidth ไม่เกินไปกว่า $f_s/2$

ในทางปฏิบัติแล้วจะยังคงเกิด Frequency folding ได้เสมอจากส่วนฮาร์โมนิคส์ของสัญญาณ รวมทั้งสเปคตรัมของสัญญาณรบกวนที่ยังคงอยู่แม้ว่าจะทำการกรองความถี่ก่อนหน้ามาแล้วก็ตามการกำจัดการซ้อนทับของสเปคตรัมนี้วิธีที่ได้ผลคือ พยายามให้การสุ่มสัญญาณเป็นไปอย่างรวดเร็วมากที่สุด

ผลของการใช้อัตราการสุ่มที่ไม่เหมาะสมอีกประการหนึ่งเกิดขึ้น ดังรูปที่ 2.7 เรียกว่า Alias Frequency ซึ่งเกิดกับสัญญาณที่เปลี่ยรกลับมาเช่นเดิมหลังจากถูกสุ่มแล้ว

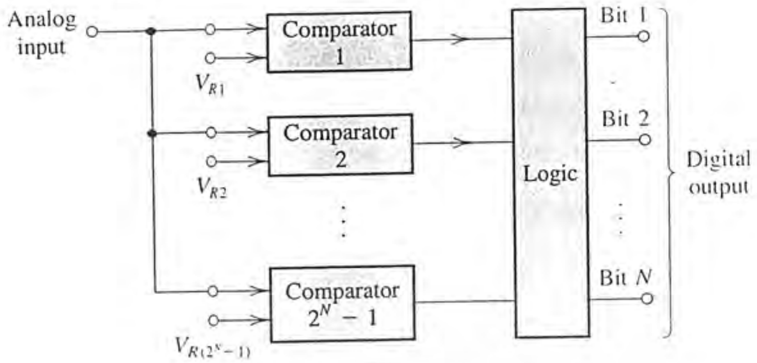


รูปที่ 2.8 การเกิด Alias Frequency จากการสุ่มด้วยความถี่ต่ำกว่า 2 เท่าของของสัญญาณอนาลอก

2.5 การแปลงสัญญาณอนาลอกเป็นสัญญาณดิจิตอล(Analog to Digital Convertor)

ในการแปลงสัญญาณอนาลอกเป็นสัญญาณดิจิตอล มีหลายวิธีการขึ้นกับลักษณะเฉพาะและความเหมาะสมในการใช้งาน โดยโครงงานนี้ใช้วงจรแปลงสัญญาณอนาลอกเป็นสัญญาณดิจิตอลแบบ Parallel (Flash) A/D Converter ซึ่งเป็นตัวแปลงสัญญาณที่ทำงานได้เร็วมาก ภายในตัวมันจะใช้การวัดระดับแรงดันและการเปรียบเทียบแรงดัน ซึ่งจะแสดงบล็อกไดอะแกรม ดังรูปที่ 2.9

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.9 แสดงการต่อวงจร Parallel Comparator A/D Converter

หลักการทำงานก็คือ จะใช้คอมพาราเตอร์ทำการเปรียบเทียบสัญญาณอนาลอกอินพุทกับแรงดันอ้างอิงที่แบ่งแรงดันให้สอดคล้องกับรหัสดิจิตอล โดยใช้ตัวต้านทาน และแปลงเอาท์พุทจากคอมพาราเตอร์ให้ตรงกับรหัสดิจิตอล ซึ่งจะเห็นว่าอุปกรณ์ทางด้านความเร็วจะถูกจำกัดเพียง Propagation time ของคอมพาราเตอร์เท่านั้น แต่อุปสรรคที่สำคัญต่อการพัฒนาวงจรบนชิพไอซี คือ วงจรนี้ต้องการคอมพาราเตอร์ จำนวน $2^n - 1$ ตัว ซึ่งเป็นจำนวนที่มากพอสมควร

แรงดันอินพุท	**เอาท์พุทของวงจรเปรียบเทียบ**			**เอาท์พุทเลขฐานสอง**	
V_{in} (โวลท์)	**A1**	**A2**	**A3**	**D1**	**D0**
0 – 1	0	0	0	0	0
1 – 2	1	0	0	0	1
2 – 3	1	0	1	1	0
3 - 4	1	1	1	1	1

ตารางที่ 2.1 ความสัมพันธ์ระหว่างแรงดันอินพุทเป็นอนาลอกกับเอาท์พุทที่เป็นดิจิตอล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 3
หลักการของดิจิตอลสตอเรจออสซิลโลสโคป

ดิจิตอลสตอเรจออสซิลโลสโคป (Digital Storage Oscilloscope:DSO) ได้รับการออกแบบให้มีความสามารถในการตอบสนองความต้องการของผู้ใช้เพิ่มมากขึ้น เช่น การเก็บข้อมูลของสัญญาณที่วัดได้ เพื่อการนำมาวิเคราะห์ในภายหลัง การวิเคราะห์รูปคลื่นอย่างละเอียดโดยการซูมรูปคลื่นส่วนที่ต้องการการจับสัญญาณที่เกิดขึ้นเพียงชั่วขณะหนึ่ง การทำปรีทริกเกอร์ (Pre-Trigger) และการตรวจจับสัญญาณที่มีการเปลี่ยนแปลงความถี่อย่างรวดเร็วหรือช้าเกินไป

จุดสำคัญของสโคปแบบนี้อยู่ที่วิธีการในการเก็บข้อมูลและวิธีการดิจิตอลที่นำมาใช้ โดยปกติแล้วข้อมูลที่ถูกนำมาเก็บไว้นั้นสามารถนำไปบันทึกไว้ในเครื่องคอมพิวเตอร์ได้และยังสามารถต่อพ่วงพล็อตเตอร์หรือพรินเตอร์ได้ด้วย

3.1 การทำงานของดิจิตอลสตอเรจออสซิลโลสโคป

สตอเรจออสซิลโลสโคปนั้นมีทั้งแบบอนาลอกและแบบดิจิตอล โดยอนาลอกสตอเรจออสซิลโลสโคป จะมีวิธีการเก็บบันทึกรูปแบบคลื่นแบบอนาลอก นั่นคือ การใช้หลอดสตอเรจ (Storage CRT) ซึ่งมีความสามารถทำให้สัญญาณที่เกิดขึ้นเพียงชั่วขณะแสดงผลบนจอได้เป็นเวลา 1 ชั่วโมงหรือมากกว่า

ส่วนดิจิตอลสตอเรจออสซิลโลสโคป จะมีวิธีการเก็บบันทึกรูปคลื่นด้วยระบบดิจิตอล ข้อมูลจะถูกบันทึกไว้ในหน่วยความจำแบบดิจิตอล ซึ่งจะเห็นเมื่อเทียบกับแบบดิจิตอลแล้ว แบบอนาลอกจะเก็บรูปคลื่นด้วยอุปกรณ์ แต่แบบดิจิตอลจะเก็บรูปคลื่นด้วยวงจร

ดิจิตอลสตอเรจออสซิลโลสโคป จะมีกระบวนการในการทำงานอยู่ 3 ช่วงใหญ่ ๆ คือ

1 ช่วงของการสุ่มตัวอย่างและการแปลงสัญญาณเป็นดิจิตอล

2 ช่วงของการเก็บข้อมูล

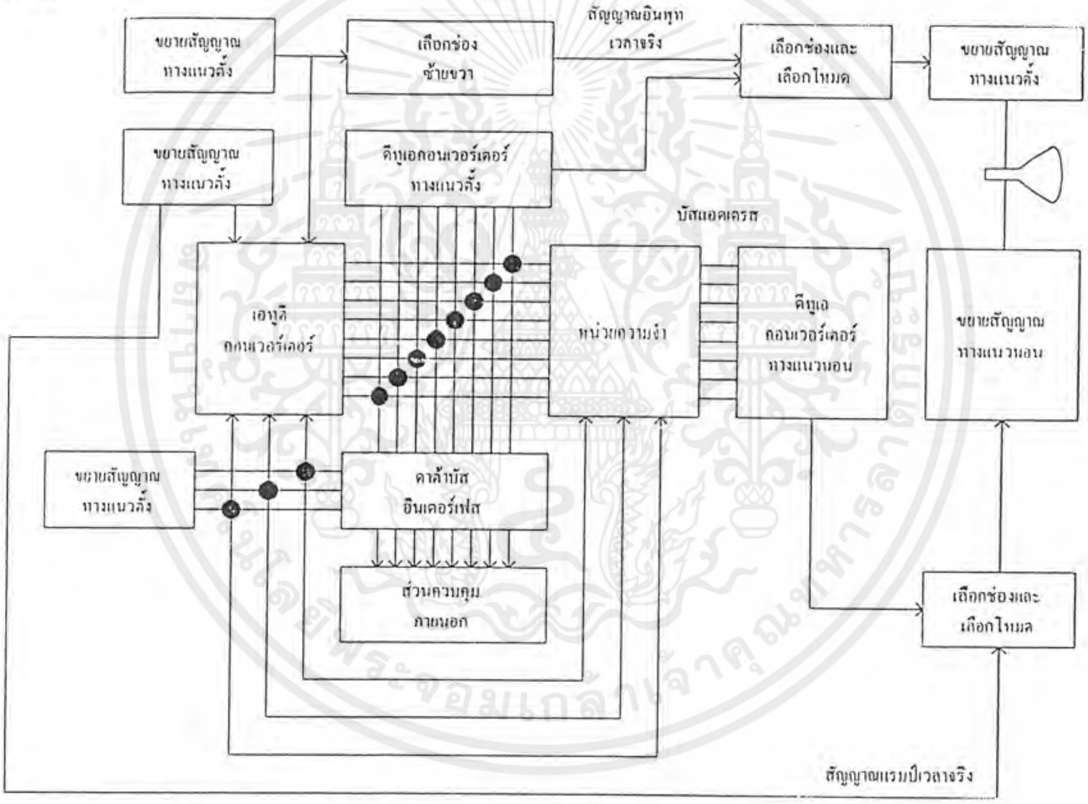
3 ช่วงของการแสดงผลค่าต่าง ๆ

เริ่มต้นด้วยการสุ่มตัวอย่างสัญญาณที่เข้ามาเพื่อให้ได้จำนวนของจุดบนสัญญาณเหล่านี้ จากนั้นจะเป็นขั้นตอนการนำค่าที่ได้จากการสุ่มตัวอย่างมาแปลงเป็นสัญญาณดิจิตอลด้วยวงจรแปลงสัญญาณอนาลอกเป็นดิจิตอล (Analog to Digital Converter) ซึ่งหากเป็นวงจรที่ใช้สัญญาณดิจิตอล 8 บิต ก็จะได้ระดับของสัญญาณที่แปลงออกมา 2^8 หรือ 256 ระดับ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ขั้นตอนที่กล่าวมาจะถูกควบคุมความเร็วในการทำงานด้วยฐานเวลา (time base) เดียวกัน ซึ่งเป็นฐานเวลาแบบดิจิตอลที่มีความเที่ยงตรงสูงมาก และจะใช้เป็นเวลาอ้างอิงในการทำงานของวงจร ซึ่งโดยปกติแล้วค่าฐานเวลาจะมีหน่วยเป็น MS/s หรือจำนวนจุดต่อวินาที ค่าที่แปลงเป็นสัญญาณดิจิตอลแล้วจะถูกเก็บไว้ในหน่วยความจำ

สำหรับการแสดงผลของดิจิตอลสตอเรจออสซิลโลสโคป จะอาศัยการทำงานแบบอนาลอก ดังนั้นในช่วงของการแสดงผลจึงต้องแปลงสัญญาณดิจิตอลไปเป็นสัญญาณอนาลอกเสียก่อน เพื่อให้ได้สัญญาณควบคุมลำอิเล็กตรอนทั้งแนวตั้งและแนวนอน รายละเอียดแสดงดังรูปที่ 3.1



รูปที่ 3.1 แสดงบล็อคไดอะแกรมของดิจิตอลสตอเรจออสซิลโลสโคป

3.2 ขอบเขตการวัด

เมื่อดิจิตอลสตอเรจออสซิลโลสโคปถูกสร้างขึ้นมาด้วยขบวนการทางดิจิตอล ดังนั้นขีดความสามารถของมันจึงถูกจำกัดด้วยวิธีการทางดิจิตอล ในอนาลอกออสซิลโลสโคปหรือดิจิตอลออสซิลโลสโคปทั่วไป ผู้ผลิตกำหนดขอบเขตการวัดเป็นค่าความถี่สูงสุดที่เครื่องสามารถกำหนดได้ เช่น 20 MHz, 60MHz, 100MHz, 150MHz เป็นต้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สำหรับดิจิตอลสตอเรจออสซิลโลสโคปนั้น ขอบเขตการวัดนอกจากสามารถบอกเป็นค่าความถี่สูงสุดที่เครื่องสามารถทำการวัดได้แล้ว ยังต้องบอกค่าอัตราการสุ่มตัวอย่างข้อมูลควบคู่กันไปด้วย โดยปกติแล้ว จะมีค่าเป็นล้านครั้งต่อวินาที (MS/s) เช่น 20 (MS/s) 20 MHz, 40MS/s 100MHz, 100MS/s 100MHz, 200 MS/s 70 MHz เป็นต้น

อัตราการสุ่มตัวอย่างข้อมูลจะชี้ให้เห็นถึงความละเอียดและคุณภาพของภาพที่ปรากฏบนจอ ซึ่งประกอบขึ้นมาจากจุดที่ได้จากการสุ่มตัวอย่างบนรูปคลื่นที่วัด ถ้าจำนวนจุดที่ได้มีจำนวนมากภาพที่ได้ก็จะมีความสมบูรณ์มาก การสูญเสียจากการสุ่มตัวอย่างไปเพียงเล็กน้อยในบางรูปคลื่น ก็อาจเป็นการสูญเสียรายละเอียดของรูปคลื่นนั้นไปได้ ถ้าอัตราการสุ่มตัวอย่างมีค่ามาก นั่นคือ ดิจิตอลสตอเรจออสซิลโลสโคปจะมีความละเอียดของรูปคลื่นสูงมากด้วย ส่วนความถี่ที่บอกมานั้นก็ยังเป็นขอบเขตการวัดความถี่สูงสุดของดิจิตอลสตอเรจออสซิลโลสโคป ในลักษณะเดียวกับอนาลอกสโคป

ถ้าต้องการจะวัดความละเอียดของรูปคลื่นที่ปรากฏบนจอภาพกสามารถทำได้ด้วยการคำนวณตามสูตรที่ว่า

$$\text{จำนวนจุดต่อ1รูปคลื่น} = \frac{\textit{จำนวนการสุ่มตัวอย่างของดิจิตอลสตอเรสออสซิลโลสโคป}}{\textit{ค่าความถี่ที่ทำการวัด}}$$

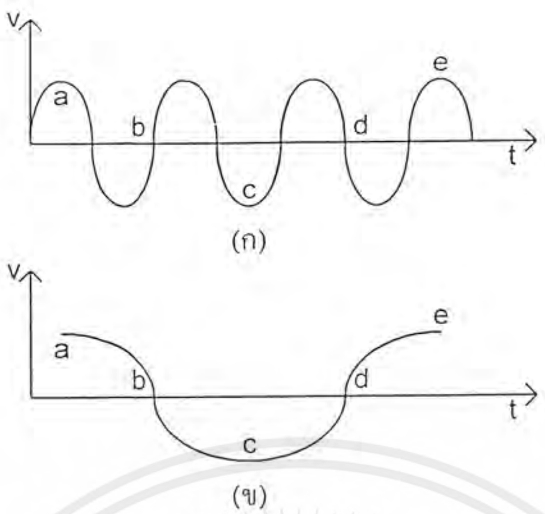
ตัวอย่างเช่น ดิจิตอลสตอเรจออสซิลโลสโคปมีอัตราการสุ่มตัวอย่าง 50 MS/s ใช้วัดสัญญาณที่มีความถี่ 500kHz ดังนั้นจะได้ว่า

จำนวนจุดต่อหนึ่งรูปคลื่น = 50 MS/s / 500 kHz = 100 จุด

ภาพที่แสดงผลบนจอภาพนั้นในหนึ่งรูปคลื่นจะเกิดการเรียงกันของจุดจำนวน 100 จุด แต่ถ้าหากความถี่ที่ใช้วัดเปลี่ยนเป็น 10 MHz ที่อัตราการสุ่มตัวอย่างเดิม ความละเอียดของภาพในหนึ่งรูปคลื่นจะมีเพียง 5 จุด ต่อรูปคลื่นเท่านั้น ซึ่งจากทั้ง 2 กรณี ทำให้สรุปได้ว่าเมื่อนำมาวัดค่าสัญญาณที่มีความถี่สูง ๆ แล้ว จะให้ภาพที่มีความละเอียดลดลง

ปัญหาอย่างหนึ่งซึ่งเกิดขึ้นกับดิจิตอลสตอเรจออสซิลโลสโคป คือ การที่สัญญาณที่จะทำการวัดมีความถี่สูงมาก ๆ เมื่อมีการสุ่มตัวอย่างสัญญาณดังกล่าว แล้วนำมาสร้างให้ปรากฏบนจออีกครั้ง ผลที่ได้บนจอจะเป็นภาพที่ต่างไปจากสัญญาณที่วัด โดยดูได้จากตัวอย่างในรูปที่ 3.2 การใช้งานควรพยายามหลีกเลี่ยงอย่าให้เกิดสัญญาณเอเลียสขึ้น เพราะว่าสัญญาณเอเลียสเป็นรูปคลื่นที่มีความถี่ต่างไปจากรูปคลื่นจริงอย่างมาก ซึ่งไม่สามารถนำไปวิเคราะห์ได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.2 แสดงลักษณะของการเกิดเอเลียสซิ่ง
(ก) สัญญาณอินพุทเมื่อมีการสุ่มตัวอย่างต่ำเกินไป
(ข) เมื่อแสดงผลเกิดเป็นสัญญาณเอเลียส

การทำให้รูปคลื่นที่ปรากฏบนจอมีความสมบูรณ์ถูกต้องนั้น จำนวนจุดที่ได้จากการสุ่มตัวอย่างต้องมีค่ามากกว่า 2 เท่าของความถี่ที่วัดนั้นใน 1 รูปคลื่น และเมื่อทำการตรวจวัดขอบขาขึ้นและขอบขาลงของพัลส์จะต้องได้จำนวนพัลส์ที่สุ่มตัวอย่างมากกว่า 10 ครั้งต่อ 1 รูปคลื่น นอกจากการสุ่มตัวอย่างที่ต้องการความเร็วในการทำงานพอสมควรแล้วยังต้องพิจารณาว่าวงจรแปลงสัญญาณอนาลอกเป็นดิจิตอลนั้น สามารถทำงานด้วยความเร็วขนาดไหน

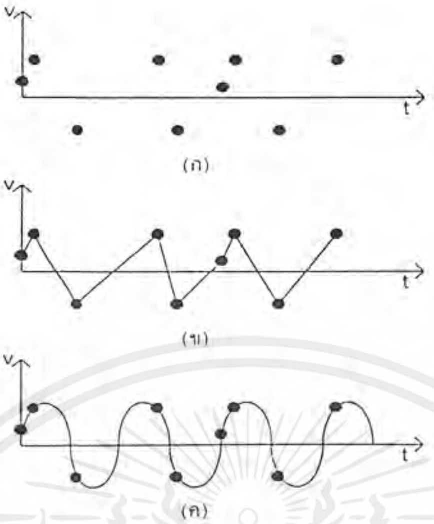
สิ่งที่จะทำให้รู้ถึงขอบเขตการวัดของดิจิตอลสตอเรจออสซิลโลสโคป อีกอย่างหนึ่งก็คือค่าความถี่สูงสุดของสัญญาณที่ดิจิตอลสตอเรจออสซิลโลสโคปสามารถทำการเก็บข้อมูลได้ ซึ่งเราเรียกว่าค่า USB (Usefus Storage Bandwidth) ซึ่งค่าจะขึ้นอยู่กับความสามารถในการสุ่มตัวอย่างของดิจิตอลสตอเรจออสซิลโลสโคปด้วย สามารถหาได้จากสูตร

USB = อัตราการสุ่มตัวอย่างสูงสุด (MS/s) / 25

ตัวเลข 25 เป็นจำนวนจุดที่ใช้ในการสร้างภาพบนจอใน 1 รูปคลื่น ซึ่งเป็นค่าที่ใช้โดยทั่วไป

เนื่องจากดิจิตอลสตอเรจออสซิลโลสโคป ไม่สามารถแสดงรายละเอียดของสัญญาณที่มีความถี่สูงกว่าขีดจำกัดได้มากพอ ดังนั้นจึงได้มีการคิดค้นวิธีการเพิ่มรายละเอียดของสัญญาณโดยการสร้างสัญญาณสุ่มตัวอย่างสอดแทรกลงไปในการสุ่มตัวอย่างแบบปกติ ผลที่ได้คือ จะมีจุดเสริมเข้าไปในรูปคลื่นเดิม ภาพที่ได้จะมีความต่อเนื่องสม่ำเสมอมากขึ้นและเรียกวิธีการนี้ว่า "ทฤษฎีการสอดแทรก" (Interpolation Methods) ซึ่งแสดงตัวอย่างการสอดแทรกสัญญาณในรูปที่ 3.3

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.3 ทฤษฎีการสอดแทรก
(ก) การแสดงผลแบบจุด
(ข) การสอดแทรกด้วยวิธีเวกเตอร์
(ค) การสอดแทรกแบบซายน์

การแสดงให้ให้เห็นถึงการสร้างภาพให้ปรากฏบนจอโดยการเรียงจุดที่ได้จากการสุ่มตัวอย่างโดยปกติจะได้ประมาณ 25 จุดต่อรูปคลื่น ส่วนการลากเส้นเชื่อมต่อระหว่างจุดที่ได้มาทำให้ได้ภาพที่เป็นเส้นตรงที่เชื่อมต่อกัน 10 เส้นต่อรูปคลื่น ซึ่งวิธีการนี้เรียกว่า การสอดแทรกแบบลิเนียร์ (Linear Interpolation) หรือ การแสดงผลแบบเวกเตอร์ (Vector Display) ดังนั้น เมื่อเป็นการแสดงผลของรูปคลื่นซายน์ ค่า USB จะเป็น

USB = อัตราการสุ่มตัวอย่างสูงสุด (MS/s) / 10

สำหรับการสอดแทรกโดยการนำเอาฟังก์ชันซายน์มาคำนวณลากเส้นเชื่อมต่อ ระหว่างจุด (Sine Interpolation) ซึ่งวิธีการนี้จะใช้จุดเพียง 2.5 จุดต่อรูปคลื่นเท่านั้น ในการแสดงผลดังนั้นค่า USB จะเป็น

USB = อัตราการสุ่มตัวอย่างสูงสุด (MS/s) / 2.5

ซึ่งทำให้ค่า USB มีค่ามากกว่า แสดงถึงขอบเขตของดิจิตอลสตอเรจออสซิลโลสโคป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ถูกขยายมากไปด้วย

ดิจิตอลสตอเรจออสซิลโลสโคปกับรูปคลื่นพัลส์

ส่วนที่ความสำคัญมากต่อรูปคลื่นพัลส์ก็คือ ช่วงเวลาขอบขาขึ้นและขอบขาลงของพัลส์ ดังนั้นเครื่องมือที่จะใช้วัดรูปคลื่นเหล่านี้จะต้องสามารถวัดช่วงขอบขาขึ้นและขอบขาลงได้โดยมีความผิดเพี้ยนน้อยที่สุด ในอนาลอกออสซิลโลสโคปสามารถคำนวณหาช่วงเวลาขอบขาขึ้น (t_r : rise time) ของพัลส์ได้ดังนี้

$$t_r \text{ (ns)} = 0.35 \text{ / แบนวิดธ์ (MHz)}$$

จากสูตรดังกล่าวหากอนาลอกสออสซิลโลสโคปมีย่านการวัด 100 MHz จะได้ช่วงเวลาขอบขาขึ้นของพัลส์เท่ากับ 3.5 ns นี่คือวิธีการของอนาลอกออสซิลโลสโคป ส่วนในดิจิตอล
ออสซิลโลสโคปช่วงเวลาขอบขาขึ้นที่วัดได้จะมีการเปลี่ยนแปลงอยู่ระหว่าง 0.8 ถึง 1.6 เท่าขอบช่วงเวลาการสุ่มตัวอย่างหากคิดช่วงเวลาขอบขาขึ้นเป็น 10 ถึง 90 เปอร์เซ็นต์ของแอมปลิจูดขอบพัลส์จะได้

$$t_r = 0.8*2 \text{ (ช่วงเวลาการสุ่มตัวอย่าง)}$$

หรือ

$$t_r = 1.6 \text{ / (อัตราการสุ่มตัวอย่าง)}$$

เพื่อให้การคำนวณมีการคลาดเคลื่อนน้อยลง สูตรการหาช่วงเวลาขอบขาขึ้นที่นำไปใช้งานจริง (Ut_r = Use Rise Time) จึงใช้ค่า 1.6 ตลอด ดังนี้

$$Ut_r = 1.6 \text{ / (อัตครการสุ่มตัวอย่าง๗}$$

หรือ

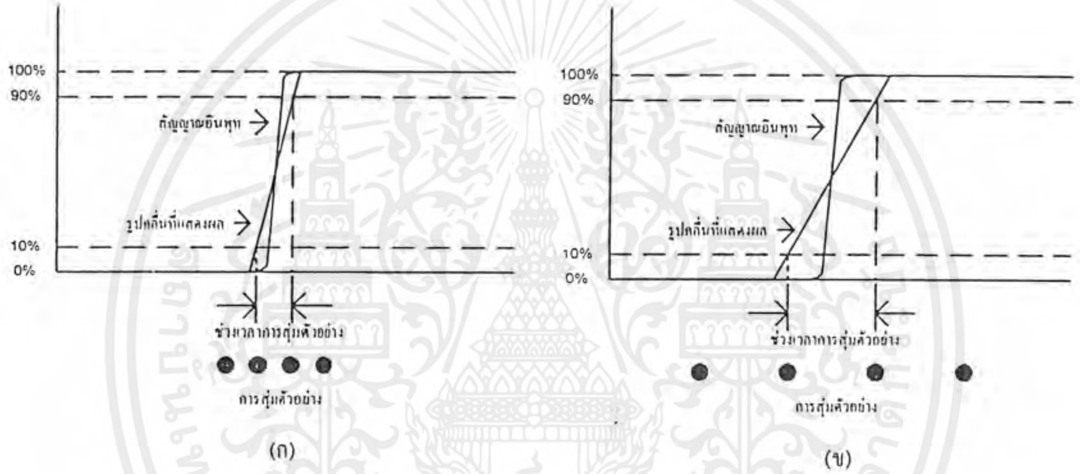
$$Ut_r = \text{(ช่วงเวลาการสุ่มตัวอย่างต่ำสุด)}*1.6$$

สำหรับดิจิตอลสตอเรจออสซิลโลสโคปแล้ว ในการหาช่วงขอบขาขึ้นที่สามารถจะวัดได้จะมีกาคำนวณเช่นเดียวกับดิจิตอลออสซิลโลสโคป เช่นถ้า มีอัตราการสุ่มตัวอย่างเป็น 100MS/s ก็จะมีการสุ่มตัวอย่างเป็น 0.01 μs และ t_rDSO = 0.01 μs* 1.6 = 16 ns นั่นคือ ดิจิตอลออสซิลโลสโคปเครื่องนี้จะสามารถแสดงผลช่วงขอบขาขึ้นในการวัดได้ภายในเวลา 16 ns เมื่อคิดช่วงขอบขาขึ้นเป็น 10 ถึง 90 เปอร์เซ็นต์ของแอมปลิจูด

จะเห็นได้ว่าความสามารถในการแสดงผลรูปคลื่นพัลส์นั้นจะขึ้นอยู่กับช่วงเวลาในการสุ่มตัวอย่างของ ดิจิตอลสตอเรจออสซิลโลสโคป และจะเป็นการแสดงให้เห็นถึงช่วงเวลาขอบ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ขาขึ้นที่อยู่ระหว่างช่วงการสุ่มตัวอย่าง โดยรูปคลื่นที่แสดงผลจะอยู่ในช่วง 0.8 * (ช่วงเวลาการสุ่มตัวอย่าง) รูปคลื่นที่แสดงผลออกมามากจะมีความชันของขอบขาขึ้นต่างจากรูปคลื่นจริงอยู่บ้างและเมื่อดูรูปสัญญาณอินพุทเดียวกัน แต่มีช่วงเวลาการสุ่มตัวอย่างต่างกัน จะแสดงผลช่วงขอบขาขึ้นจะอยู่ในช่วง 1.6 * (ช่วงเวลาการสุ่มตัวอย่าง) ซึ่งถ้าเปรียบแล้ว ความชันของขอบขาขึ้น จะแตกต่างจากสัญญาณอินพุทจริงค่อนข้างมาก



รูปที่ 3.4 แสดงการจัดช่วงขอบขาขึ้น ซึ่งค่าคลาดเคลื่อนที่เกิดขึ้นอยู่กับตำแหน่งในการสุ่ม

(ก) ช่วงขอบขาขึ้นอยู่ระหว่าง 0.85

(ข) ช่วงขอบขาขึ้นอยู่ระหว่าง 0.85

จากที่ทราบในตอนต้นว่า การจัดช่วงขอบขาขึ้นและขอบขาลงของพัลส์จะต้องทำการสุ่มตัวอย่างมากกว่า 10 ครั้ง เราสามารถหาช่วงเวลาที่ต้องใช้ในการวัดขอบขาขึ้นได้ โดยหารอัตราการสุ่มตัวอย่างด้วย 10 แล้วกลับเศษเป็นส่วนดังนี้

Resolution = 10 / (อัตราการสุ่มตัวอย่าง)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ถ้าเป็นดิจิตอลสตอเรจออสซิลสโคป ที่มีอัตราการสุ่มตัวอย่าง 100MS/s จะมีการสุ่มตัวอย่างทุก ๆ 10ns (1/100 MS/s) และจะใช้การทำงานกับขอบขาขึ้นเป็นเวลา 100 ns ถ้าพัลส์ที่เข้ามามีความถี่สูงมาก ดิจิตอลสตอเรจออสซิลสโคปที่ใช้วัดจะต้องมีความเร็วในการสุ่มตัวอย่างสูงขึ้นด้วย เช่น 500 MS/s ซึ่งจะมีการสุ่มตัวอย่างทุก ๆ 2 ns และสามารถวัดช่วงขอบขาขึ้นที่อยู่ระหว่างช่วงการสุ่มตัวอย่างโดยใช้เวลา 1.17 ns

จะเห็นว่าดิจิตอลสตอเรจออสซิลสโคปตอบสนองการวัดรูปคลื่นพัลส์ได้ไม่ค่อยจะดีเท่าอนาลอกออสซิลโลสโคป และเราไม่สามารถนำค่า Ut_r เป็นค่าซึ่งแสดงถึงขอบเขตจำกัดของดิจิตอลสตอเรจออสซิลสโคปได้

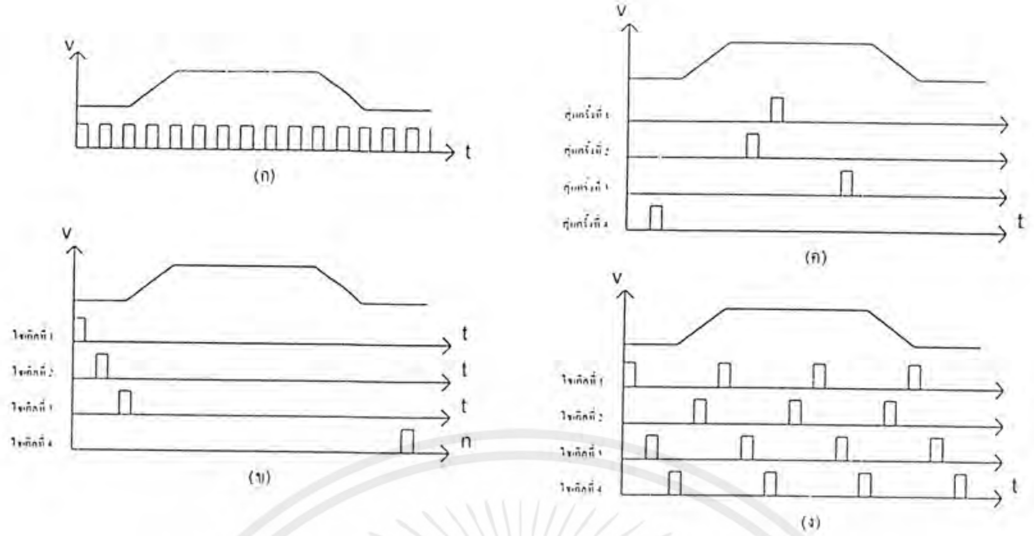
3.3 การสุ่มตัวอย่างของดิจิตอลสตอเรจออสซิลสโคป

ดิจิตอลสตอเรจออสซิลโลสโคป ใช้การสุ่มตัวอย่างสองแบบ คือ

3.3.1 การสุ่มตัวอย่างแบบเวลาจริง (Real Time or One-shot) จะเกิดการสุ่มตัวอย่างขึ้นทันทีที่มีการจับสัญญาณอินพุทได้ โดยผลที่ได้จะมาจากรูปคลื่นจริงและจะมีความเหมือนกับรูปคลื่นจริงมาก ส่วนมากจะใช้กับสัญญาณที่เกิดขึ้นเพียงชั่วขณะหรือใช้ในการสุ่มตัวอย่างบางส่วนของสัญญาณที่ต้องการวิเคราะห์เป็นพิเศษ ดังนั้นความเร็ซของการสุ่มตัวอย่างแบบเวลาจริงนี้จึงต้องสูงพอที่จะทำงานในกรณีดังกล่าวได้ การสุ่มตัวอย่างแบบนี้จะทำงานแบบผ่านครั้งเดียวคือ จะมีการสุ่มตัวอย่างจุดเดียวในแต่ละครั้งของการทำงาน จึงสามารถนำไปตรวจจับสัญญาณที่เกิดขึ้นเพียงชั่วขณะเช่น แรงไฟกระชาก, สัญญาณควบคุมเครื่องกล, สัญญาณในลักษณะทางชีววิทยา ฯลฯ ซึ่งจะปรากฏให้สามารถวัดได้ในระยะเวลาที่สั้นมาก การจะใช้วิธีสุ่มตัวอย่างจึงต้องใช้แบบที่เมื่อเริ่มสุ่มและสิ้นสุดการสุ่มต้องได้รูปสัญญาณที่เกิดขึ้น รูป 3.5 (ก) จะเห็นการสุ่มตัวอย่างแบบเวลาจริงที่ใช้ในการวัดรูปคลื่นพัลส์หนึ่งลูก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สำนักหอสมุดกลาง พระจอมเกล้าลาดกระบัง



รูปที่ 3.5 เทคนิคการสุ่มตัวอย่างแบบต่าง ๆ ของดิจิตอลสตอเรจออสซิลโลสโคป

(ก) การสุ่มแบบเวลาจริง

(ข) การสุ่มแบบเวลาเสมือนชนิดเป็นลำดับ

(ค) การสุ่มแบบเวลาเสมือนชนิดไม่เป็นลำดับโดยใช้จุดเดียวในการทำงานหนึ่งครั้ง

(ง) การสุ่มแบบเวลาเสมือนชนิดไม่เป็นลำดับโดยใช้หลายเดียวในการทำงานหนึ่งครั้ง

3.3.2 การสุ่มตัวอย่างแบบเวลาเสมือน (Equivalent Time) การสุ่มตัวอย่างแบบนี้โดยมากจะใช้กับสัญญาณที่เกิดขึ้นซ้ำ ๆ กันหรือใช้ตรวจจับบางส่วนของสัญญาณที่เกิดขึ้นซ้ำ ๆ กัน การสุ่มตัวอย่างแบบเวลาเสมือนสามารถแบ่งออกเป็น 2 ชนิดย่อย ๆ อีก คือ การสุ่มตัวอย่างเป็นลำดับ (Sequential Sampling) และการสุ่มชนิดไม่เป็นลำดับ (Random Sampling)

ในการสุ่มตัวอย่างชนิดลำดับจะเป็นดังรูปที่ 3.5 (ข) ซึ่งการสุ่มตัวอย่างชนิดนี้ในการทำงานหนึ่งครั้งหรือหนึ่งไซเคิลจะต้องได้จุดบนรูปคลื่น 1 จุด แล้วนำไปเก็บไว้ในหน่วยความจำ จนกระทั่งหน่วยความจำเต็ม เมื่อนำไปแสดงผลก็จะเป็นไปตามรูปที่ 3.6 ส่วนการสุ่มตัวอย่างแบบเวลาเสมือนชนิดไม่เป็นลำดับนั้นจะเป็นดังรูป 3.5 (ค) ซึ่งการสุ่มตัวอย่างแบบนี้จะใช้เมื่อมีการทำปริทริกเกอร์ (pre-trigger) เพื่อดูรูปที่อยู่หลังจุดทริก ซึ่งการสุ่มตัวอย่างแบบเป็นลำดับจะไม่สามารถทำได้

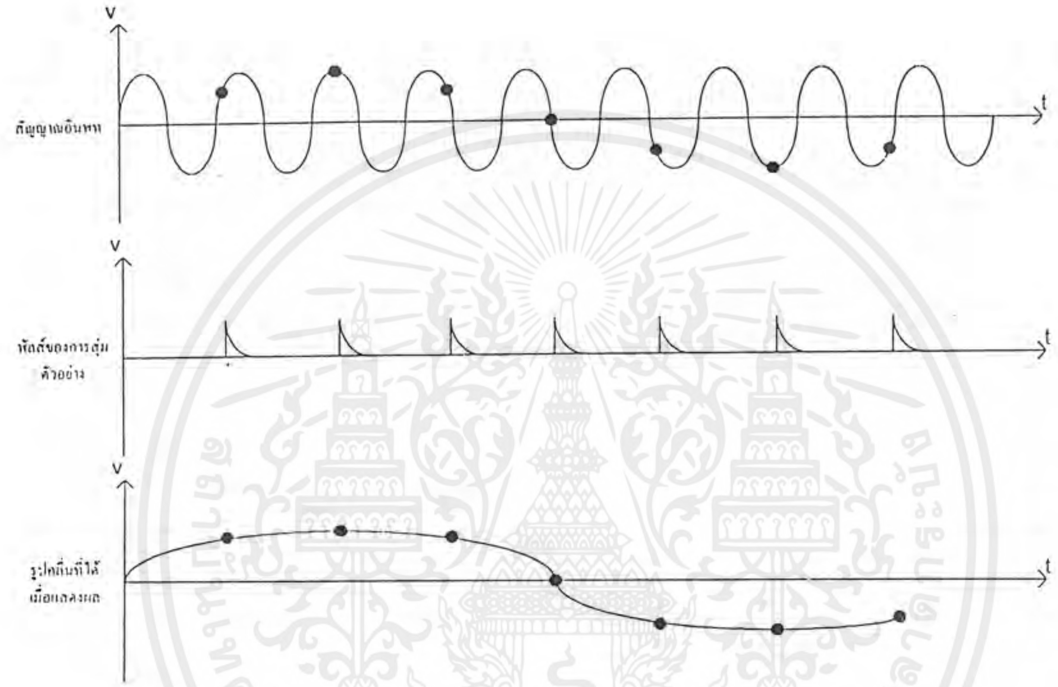
ในการทำงานหนึ่งไซเคิลจะได้จุดบนรูปคลื่น 1 จุดเช่นกัน แต่จะเป็นจุดบนส่วนใดของรูปคลื่นก็ได้ โดยจะทำไปเรื่อย ๆ จนหน่วยความจำเต็ม

ในรูปที่ 3.5 (ง) จะเป็นการสุ่มตัวอย่างแบบไม่เป็นลำดับเช่นกัน แต่จะต่างกันตรงที่ในการทำงานหนึ่งไซเคิลนั้นจะได้หลายจุดบนรูปคลื่น และในการทำงานครั้งต่อไปก็จะได้จุดหลายจุดแต่จะเป็นจุดที่ไม่ซ้ำจุดเดิม (จุดใดก็ได้) การสุ่มตัวอย่างในรูปนี้จะนำไปใช้ประโยชน์

เอกสารนี้เป็นเอกสารที่สงวนสิทธิ์สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

42742

เมื่อมีการวิเคราะห์รูปคลื่น โดยการนำข้อมูลที่ได้จากการสุ่มตัวอย่างแบบนี้มาเปรียบเทียบกับรูปคลื่นอื่น ๆ โดยเฉพาะอย่างยิ่งการวิเคราะห์รูปคลื่นในส่วนที่เป็นการทริกในลักษณะต่าง ๆ กัน



รูปที่ 3.6 แสดงให้เห็นถึงผลที่ออกมาจากการสุ่มตัวอย่างสัญญาณอินพุทแบบเป็นลำดับ

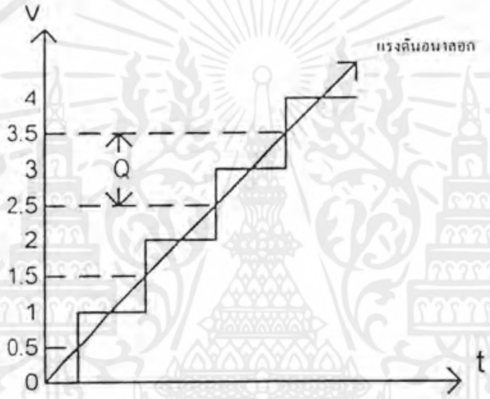
สำหรับดิจิตอลสตอเรจออสซิลโลสโคป ที่มีการใช้การแสดงผลแบบอนาลอก ซึ่งระบุความสามารถในการวัดไว้ เช่น 100 MHz ความหมายของ 100 MHz นี้ก็คือ ความสามารถในการวัดสัญญาณความถี่ 100 MHz แล้วดิจิตอลสตอเรจออสซิลโลสโคปจะแสดงผลในแบบอนาลอกได้ทันที 100 MHz เฉพาะสัญญาณที่เกิดขึ้นซ้ำ ๆ กัน แต่หากเป็นสัญญาณชั่วขณะเกิดขึ้นแล้วใช้การสุ่มตัวอย่างแบบเวลาจริงแล้ว ดิจิตอลสตอเรจออสซิลโลสโคปนี้จะสามารถวัดสัญญาณชั่วขณะได้เพียง 10 MHz เท่านั้น

เมื่อคิดเป็นอัตราส่วนระหว่างการทำงานกับสัญญาณปกติและการทำงานกับสัญญาณชั่วขณะของดิจิตอลสตอเรจออสซิลโลสโคป ก็จะได้เป็น 10:1 และสามารถเพิ่มเป็น 5:1 หรือ 2.5:1 ได้ ซึ่งหมายถึง ดิจิตอลสตอเรจออสซิลโลสโคปจะต้องมีการใช้วิธีการสอดแทรกอย่างมาก สาเหตุที่เกิดกรณีดังกล่าวขึ้นมาจากการที่ความสามารถในการสุ่มตัวอย่างถูกกำหนดด้วยสัญญาณนาฬิกาภายใน แต่สัญญาณที่เข้าวัดจะเป็นการทำงานแสดงผลแบบอนาลอก ซึ่งขอบเขตของความถี่จะกว้างมาก นี่คือการนำข้อดีของอนาลอกมาใช้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.4 ขั้นตอนการสุ่มตัวอย่าง

เมื่อเราได้ทำการสุ่มตัวอย่างไปแล้ว ค่าที่ได้จะไม่ทำการแปลงเป็นสัญญาณดิจิตอลในทันที โดยทั้งนี้เนื่องมาจากค่าที่มาจากการสุ่มตัวอย่างนั้นเป็นค่าที่มีระดับแตกต่างกันอย่างมากมาย ในทางปฏิบัติเราไม่สามารถแปลงเป็นสัญญาณดิจิตอลได้มากขนาดนั้น จึงจำเป็นต้องแบ่งค่าที่ได้มาจากการสุ่มตัวอย่างออกเป็นกลุ่ม ๆ เราเรียกว่า ควอนไตซิ่ง (Quantizing) ซึ่งการควอนไตซิ่งนี้จะเป็นการนำค่าจากการสุ่มตัวอย่างที่มีค่าใกล้เคียงกันมากอยู่ในกลุ่มเดียวกัน ต่อจากนั้นจะนำค่าตัวเลขมาใช้แทนกลุ่มสัญญาณดังกล่าว



รูปที่ 3.7 การควอนไตซิ่ง โดยใช้วงจรแปลงสัญญาณอนาลอกเป็นดิจิตอล คอนเวอร์เตอร์เปลี่ยนวงจรอนาลอกเป็นเลขฐานสองที่ระดับต่าง ๆ

การแบ่งกลุ่มสัญญาณจะมีลักษณะเป็นขั้นบันได โดยให้มีการเปลี่ยนแปลงของค่าที่ได้จากการสุ่มตัวอย่างอยู่ระหว่าง 0.4 โวลท์ และจัดแบ่งกลุ่มได้ 4 กลุ่ม คือ ช่วงระดับ 0.1, 1-2, 2-3 และ 3-4 โวลท์ โดยให้สัญญาณที่อยู่ในระดับ 0.1 มีค่า 0.5 แทนกลุ่มนี้ และ 1.5 แทนกลุ่ม 1-2, ระดับ 2.5 แทนกลุ่ม 2-3 และระดับ 3.5 แทนกลุ่ม 3-4 ค่า 0.5, 1.5, 2.5 และ 3.5 จะนำไปใช้ใน A/D อีกครั้งเพื่อทำเป็นสัญญาณดิจิตอล

เนื่องจากวิธีการควอนไตซิ่งมีโอกาสที่จะเกิดค่าความคลาดเคลื่อนได้ อันเนื่องมาจากสัญญาณรบกวนที่เข้ามาในระบบทำให้ค่าที่ได้ในการจัดกลุ่มผิดไปจากความเป็นจริง และเนื่องจากการควอนไตซิ่งนั้นประกอบจากวงจรดิจิตอลอิเล็กทรอนิกส์ ดังนั้นจึงเกิดความไม่แน่นอนของผลที่ได้โดยจะแสดงค่าบวก ลบ ของบิตสุดท้าย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เมื่อผ่าน A/D แล้ว นอกจากบิตสุดท้ายจะเป็นค่าของความไม่แน่นอนแล้ว บิตอื่น ๆ ก็จะเป็นค่าผิดพลาดที่เกิดจากสัญญาณรบกวนได้แต่สัญญาณดิจิตอลที่ได้ส่วนใหญ่ก็ยังมีความเที่ยงตรงอยู่บ้าง

3.5 การเก็บข้อมูลของดิจิตอลสตอเรจออสซิลโลสโคป

ความสามารถของดิจิตอลสตอเรจออสซิลโลสโคป ที่สำคัญมากอย่างหนึ่งก็คือ ความสามารถในการเก็บข้อมูล รองรับข้อมูลได้มากมายเช่นเดียวกับคอมพิวเตอร์ หน่วยความจำที่ใช้มี 2 แบบ คือ RAM (Random Access Memory) และ ROM (Read Only Memory)

สำหรับ RAM สามารถเก็บข้อมูลที่เป็นปัจจุบันได้อยู่ตลอดเวลา ในขณะที่ ROM นั้นจะเป็นหน่วยความจำแบบถาวร ไม่สามารถบันทึกข้อมูลเพิ่มเติมได้ ส่วนมากจะใช้ในการแสดงคุณสมบัติอุปกรณ์หรือวงจรต่าง ๆ รวมทั้งการแก้ไขปัญหาในลักษณะต่าง ๆ กัน

การเก็บข้อมูลนั้นยังสามารถบันทึกลงบนแผ่นดิสเก็ตต์, เทปแม่เหล็ก หรือแม้กระทั้งการพ่วงเข้ากับคอมพิวเตอร์เพื่อถ่ายโอนข้อมูลได้ด้วย

สำหรับการเก็บข้อมูลแบบ RAM นั้น ปกติจะมีทั้งแบบ CMOS, NMOS หรือแม้แต่ ECL (Emitter coupled bipolar Logic) ซึ่งแบบต่าง ๆ เหล่านี้นั้นเป็นหน่วยความจำ RAM ซึ่งอยู่ภายในดิจิตอลสตอเรจออสซิลโลสโคป และมีขนาดหน่วยความจำที่จำกัด สำหรับหน่วยความจำขนาดใหญ่และมีความยาวข้อมูลมาก ๆ จะต้องมีการแก้ไขปรับปรุงให้ทันสมัยอยู่ตลอดเวลาด้วยและขนาดหน่วยความจำที่ดิจิตอลสตอเรจออสซิลโลสโคปใช้ จะอยู่ในช่วง 4k ถึง 32k นอกจากนี้ในปัจจุบันยังมีหน่วยความจำแบบแผ่นการ์ด (RAM card) อีกด้วย

3.6 ปัญหาพื้นฐานของดิจิตอลสตอเรจออสซิลโลสโคป

ปัญหาที่มักจะเกิดขึ้นบ่อย ๆ กับอนาลอกออสซิลโลสโคปและดิจิตอลออสซิลโลสโคป คือ การสั่นไหวของภาพของแนวนอน (Horizontal Jitter) อาการดังกล่าวจะทำให้ภาพที่ได้จากการวัดเกิดอาการพร่ามัว เนื่องจากมีการสั่นของภาพ และจะเกิดขึ้นกับรูปคลื่นซายน์ไปจนถึงรูปคลื่นพัลส์ด้วย เราจึงคำนวณให้มีค่าเป็น $\pm 1/2$ ของช่วงเวลาระหว่างการสุ่มตัวอย่าง

3.7 ความละเอียดถูกต้องแม่นยำ

ความละเอียดถูกต้องแม่นยำเป็นสิ่งที่สำคัญอย่างมาก ๆ สำหรับเครื่องวัดทุกชนิด ดิจิตอลสตอเรจออสซิลโลสโคปก็เช่นเดียวกัน เมื่อพูดถึงความละเอียดทางด้านแนวตั้งย่อมจะหมายถึง ความสามารถในการแยกแยะระหว่างสัญญาณซึ่งอยู่ติดกัน

โดยปกติแล้วความละเอียดทางด้านแนวตั้งของดิจิตอลสตอเรจออสซิลโลสโคปจะถูกกำหนดด้วยจำนวนเลขฐานสองที่เป็นเอาต์พุทของ A/D ตัวอย่างเช่น A/D ให้เอาท์พุทออกมา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

8 บิตก็จะมีความละเอียดได้ถึง 256 (2^8) ตำแหน่ง หรือคิดเป็น 0.391 % (1/256) เพื่อให้ชัดเจนยิ่งขึ้นนอกจากต้องรู้จำนวนบิตของ A/D แล้วจะต้องรู้อีกว่าถ้าเราตั้งค่าโวลท์ต่อช่วงไว้ที่ค่าหนึ่งแล้วจะสามารถอ่านค่าความละเอียดออกมาเป็นเท่าไร ตัวอย่างเช่น ถ้าเราตั้งค่าโวลท์ต่อช่องไว้ที่ 50mV/div เมื่ออ่านค่าเต็มหน้าจอซึ่งมี 8 ช่อง ก็จะอ่านค่าได้ทั้ง 400 mV และถ้า ADC ใช้ 8 บิต เราก็จะได้ความละเอียดของแนวตั้งทั้งสิ้น 256 ตำแหน่ง ซึ่งแต่ละช่วงระหว่างตำแหน่งจะมีค่าเป็น

$$400*10^{-3} \ / \ 256 = 1.5625 \text{ mV}$$

นั่นคือ 1 ตำแหน่งจะแทนเป็นระดับแรงดันได้ 1.5625 mV ในทำนองเดียวกันหาก A/D ให้เอาท์พุทออกมา 4 บิต ความละเอียดทางแนวตั้งก็จะได้เป็น 16 ตำแหน่งและตั้งค่าโวลท์ต่อช่วงไว้ที่ค่าเดิม ดังนั้นสำหรับ 1 ตำแหน่งก็จะมีค่า 25 mV (0.4/16)

ดังนั้นจำนวนบิตของ A/D และมาตราส่วนทางแนวตั้งที่ตั้งค่าไว้จะเป็นตัวกำหนดความสามารถในการแยกแยะรายละเอียดต่าง ๆ ของรูปคลื่น เมื่อรายละเอียดต่าง ๆ ของการแสดงผลมีความสมบูรณ์นั้นย่อมหมายความไปถึงว่าจะไม่มีเครื่องมือวัดที่จะมีความถูกต้องแม่นยำกว่า

สำหรับความละเอียดทางด้านแนวนอนนั้นเป็นค่าที่เกี่ยวพันกับคาบเวลาที่ถูกเก็บไว้ในหน่วยความจำแบบดิจิตอล หากสัญญาณถูกเก็บไว้ในข้อมูล 1024 เวิร์ด ก็จะได้ค่าความละเอียดของด้านแนวนอนเป็น 1/1024 หรือคิดเป็น 0.098 %

เปรียบเทียบระหว่างอนาลอกและดิจิตอล ถ้าเป็นอนาลอกสโคปความละเอียดของภาพที่ได้จะขึ้นอยู่กับพื้นที่ผิวด้านหน้าจอ CRT และขนาดรูปร่างของลำอิเล็กตรอนที่ใช้ในการเขียน รวมไปถึงความสามารถในการทำงานของวงจรขยายสัญญาณทางแนวตั้งและแนวนอนซึ่งกลายเป็นปัจจัยอันหนึ่งซึ่งกลายเป็นขีดจำกัดอย่างมากสำหรับอนาลอกออสซิลโลสโคปด้วย

แต่สำหรับดิจิตอลออสซิลโลสโคปความละเอียดทางแนวตั้งจะขึ้นอยู่กับจำนวนบิตที่ A/D ใช้ และในเรื่องของความถูกต้องแม่นยำ อนาลอกจะคล้ายกับดิจิตอลในบางกรณี ซึ่งโดยปกติค่าความคลาดเคลื่อนทางแนวตั้งจะไม่เกิน ±2-4 % และทางแนวนอนจะไม่เกิน ±1-3 %

อย่างไรก็ตามดิจิตอลออสซิลโลสโคปมีข้อได้เปรียบเรื่องความถูกต้องแม่นยำเพิ่มขึ้นอีกเล็กน้อย ด้วยลักษณะของภาพที่เรียกว่า frozen ที่ปรากฏบนจอ CRT และโดยการใช้ตัวชี้ค่าที่เราเรียกกันว่า cursors ซึ่งมีการชี้ค่าได้แม่นยำมากทีเดียว สิ่งที่เสริมเข้ามานี้ทำให้ดิจิตอลออสซิลโลสโคปสามารถลบจุดด้อยด้านความแม่นยำบางเรื่องลงไป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สำหรับเรื่องของฐานเวลานั้นดิจิตอลสตอเรจออสซิลโลสโคป จะไม่ใช้วงจรกำเนิดฐานเวลาแบบลิเนียร์สวีพ (linear sweep) แต่จะใช้วงจรกำเนิดฐานเวลาแบบคริสตอลแทน เนื่องจากฐานเวลาที่ได้จากแบบคริสตอลจะมีค่าความฉลาดเคลื่อนน้อยกว่า ± 0.01 %

เมื่อมาถึงจุดนี้เราจะเห็นว่าความแม่นยำด้านแนวนอนจะเกี่ยวพันกับสัญญาณนาฬิกา, ขนาดของหน่วยความจำและตัวชี้ค่า จะเห็นว่าความแม่นยำด้านแนวนอนจะต้องอาศัยองค์ประกอบมาก ดังนั้นเพื่อให้เกิดค่าความแม่นยำทางแนวนอนสูงสุดสำหรับดิจิตอลสตอเรจออสซิลโลสโคป เราจะต้องทำการวัดสัญญาณนั้น ๆ ซ้ำ ๆ กัน และต้องลดทอนค่าคลาดเคลื่อนที่เกิดเกิดจากการเลือกใช้ตัวชี้ค่าไม่เหมาะสมลงด้วย

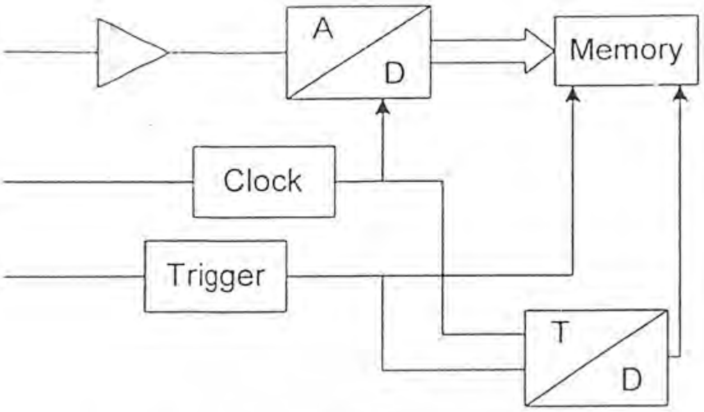
3.8 สัญญาณชั่วขณะ (Transient Capture)

ออสซิลโลสโคปแบบอนาลอกส่วนใหญ่ไม่สามารถแสดงค่าชั่วขณะที่เกิดขึ้นได้ ในทางตรงข้ามออสซิลโลสโคปแบบดิจิตอลถูกออกแบบมาเพื่อแสดงค่าชั่วขณะ โดยมีโครงสร้างของดิจิไตเซอร์ 3 แบบ สำหรับทรานเชี้ยนดิจิไตเซอร์ และ RIS (Random Interleaved Sampling) สามารถตรวจจับสัญญาณชั่วขณะและบันทึกข้อมูลรูปคลื่นปรีทริกเกอร์ (Pre-trigger) ได้ ในขณะที่แซมปลิ้งดิจิไตเซอร์ไม่สามารถทำงานได้ ดิจิไตเซอร์ทั้ง 3 ชนิดสามารถบันทึกสัญญาณที่เกิดขึ้นซ้ำ ๆ หลายครั้งได้

ในทรานเชี้ยนดิจิไตเซอร์ มีอุปกรณ์แปลงสัญญาณอนาลอกเป็นสัญญาณดิจิตอล และหน่วยความจำรูปคลื่น อุปกรณ์เปลี่ยนสัญญาณอนาลอกเป็นดิจิตอลจะเปลี่ยนสัญญาณต่อเนื่องให้อยู่ในรูปตัวเลข และส่งสัญญาณที่สุ่มได้ไปยังหน่วยความจำแบบ circular addressing หลังจากใช้หน่วยความจำตำแหน่งสุดท้าย ระบบจะเขียนข้อมูลทับอันเดิม โดยเริ่มเขียนทับที่หน่วยความจำเริ่มต้น หลังจากทริกเกอร์ถูกผลิตขึ้น หน่วยความจำจะยังคงทำงานต่อไปโดยเป็นค่าที่ถูกเลือกใช้ของตัวอย่างโพสทริกเกอร์ หลังจากนั้นอุปกรณ์เปลี่ยนสัญญาณอนาลอกเป็นสัญญาณดิจิตอลจะหยุดป้อนข้อมูลสู่หน่วยความจำ ถ้าผู้ใช้เลือกใช้ข้อมูลปรีทริกเกอร์ 100 % แล้วอุปกรณ์เปลี่ยนสัญญาณอนาลอกเป็นสัญญาณดิจิตอลจะหยุดส่งข้อมูลทันทีที่ทริกเกอร์มาถึง ถ้าผู้ใช้เลือกใช้โพส

ทริกเกอร์ 100 % ระบบจะเติมข้อมูลแอดเดรสของหน่วยความจำมากกว่า 1 ครั้งแล้วหยุด หน่วยความจำจะมีข้อมูลสรุปคลื่นที่เกิดขึ้นหลังทริกเกอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.8 บล็อคไดอะแกรมของ RIS ดิจิไตเซอร์

RIS ดิจิไตเซอร์ ประกอบด้วยทรานเชี้ยนดิจิไตเซอร์ที่เพิ่มโหมดการสอดแทรกสำหรับทริกเกอร์ แต่ละอัน RIS ดิจิไตเซอร์จะบันทึกชุดของจุดตัวอย่างรูปคลื่น จุดตัวอย่างในดิจิไตเซอร์มาจากการได้มาของการเพิ่มทริกเกอร์ เพื่อสร้างรายละเอียดของรูปร่างสัญญาณต้นกำเนิด เนื่องจากดิจิไตเซอร์ไม่มีทางรู้ได้ว่ามีสัญญาณทริกเกอร์แล้ว สัญญาณนาฬิกาและจุดทริกเกอร์จึงไม่สอดคล้องกัน ดังนั้นเวลาระหว่างทริกเกอร์และทุก ๆ สัญญาณนาฬิกาเปลี่ยนแปลงอย่างสุ่มจากรูปคลื่นที่ได้มา โครงสร้างของ RIS ใช้ TDC ในการวัดความสัมพันธ์และความแม่นยำของการได้มาของรูปคลื่น TDC มีการแก้ปัญหาเรื่องเวลามากกว่าการเว้นช่วงของเวลาของตัวอย่าง ดังนั้น RIS จะช่วยแสดงผลให้เห็นรายละเอียดที่ทรานเชี้ยนดิจิไตเซอร์อย่างเดียวพลาดไป RIS สามารถเลือกการบันทึกปรีทริกเกอร์เช่นเดียวกับทรานเชี้ยนดิจิไตเซอร์

ดิจิไตเซอร์การสุ่มตัวอย่าง (Sampling Digitizer) ที่มีประสิทธิภาพประกอบด้วยส่วนหัวของการสุ่มตัวอย่าง, อุปกรณ์แปลงสัญญาณอนาลอกเป็นดิจิตอล, หน่วยความจำรูปคลื่นและวงจรสัญญาณเวลา ส่วนหัวของการสุ่มตัวอย่างจะเก็บค่าศักดาไฟฟ้าและค่าคงที่ไว้ ขณะที่อุปกรณ์แปลงสัญญาณอนาลอกเป็นดิจิตอลทำการแปลงเป็นตัวเลข ดิจิไตเซอร์การสุ่มตัวอย่างจะได้ 1 ตัวอย่างต่อทริกเกอร์ สำหรับแต่ละทริกเกอร์ตามลำดับ วงจรสัญญาณเวลาจะหน่วงเวลาจากทริกเกอร์ถึงจุด ตัวอย่าง เช่น อัตราการสุ่ม 1GS/S จุดตัวอย่างแรกจะอยู่ที่จุดทริกเกอร์ จุดที่ 2 หน่วงไป 1 นาโนวินาที จุดที่ 3 หน่วงไป 2 นาโนวินาที เนื่องจากจุดตัวอย่างถูกหน่วงจากจุดทริกเกอร์ ดังนั้นดิจิไตเซอร์การสุ่มตัวอย่างไม่สามารถบันทึกข้อมูลที่มีก่อนทริกเกอร์ได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.9 แบนวิดธ์และอัตราการสุ่มตัวอย่าง (Bandwidth and Sample rate)

แบนวิดธ์เป็นส่วนสำคัญที่ต้องระบุ สำหรับดิจิไตเซอร์เช่นเดียวกับออสซิลโลสโคป แบบอนาลอก การขยายข้อมูลที่ป้อนเข้าของดิจิไตเซอร์และการกรองเป็นตัวตัดสินกำหนดแบนวิดธ์ ขอบสัญญาณพัลส์ที่เร็วและจุดยอดของรูปคลื่นที่ฉับพลัน ประกอบด้วยส่วนประกอบสัญญาณความถี่สูงเพื่อที่จะบันทึกยอดและขอบของสัญญาณได้ถูกต้องแม่นยำ ดิจิไตเซอร์ต้องมีแบนวิดธ์ที่เพียงพอเพื่อจะผ่านสัญญาณความถี่สูงที่มีการลดทอนต่ำ

เพื่อแสดงการขยายจุดยอดสัญญาณอย่างถูกต้องแม่นยำ แบนวิดธ์ของดิจิไตเซอร์ควรมีความจุมากกว่าแบนวิดธ์ของสัญญาณ ดังนั้นสิ่งแรกที่ต้องพิจารณาแบนวิดธ์ของสัญญาณ โดยการประมวลค่าเวลาที่เพิ่มขึ้นเร็วกว่า สมมติผลตอบสนองต่อระบบมีโพลเดียวแบนวิดธ์ของสัญญาณจะเป็นดังนี้

แบนวิดธ์ของสัญญาณ 0.35/(10% - 90% rise time)

แบนวิดธ์ของดิจิไตเซอร์แสดงความถี่ซึ่งถูกลดทอนลง 3 เดซิเบล การลดทอนนี้เกิดขึ้นอย่างค่อยเป็นค่อยไป เริ่มจากความถี่ต่ำกว่า ดังนั้นต้องเลือกดิจิไตเซอร์มีแบนวิดธ์สูงกว่าของสัญญาณผลกระทบต่ออัตราการสุ่มตัวอย่างของแบนวิดธ์ที่ใช้

อัตราการสุ่มตัวอย่างของดิจิไตเซอร์สามารถลดแบนวิดธ์ที่ใช้ เพื่อให้แน่ใจว่าการสุ่มตัวอย่างที่ใช้เพียงพอ รับมา 4 ตัวต่อรอบกับการแทรก sin x / x หรือ 10 ตัวอย่างต่อรอบกับการแทรกเส้นตรง ถ้าเป็นสัญญาณทรานเชี้ยนพิจารณารายละเอียดอัตราการสุ่มตัวอย่าง ซิงเกิลช๊อต (Single Shot) ถ้าเกิดซ้ำ ๆ หลายครั้งสามารถใช้อัตราการสุ่มที่เท่า ๆ กันอย่างรวดเร็ว

ดิจิไตเซอร์ในอุดมคติที่ไม่มีสัญญาณรบกวนและให้สัญญาณจำกัด หลักการไนควิสต์ (Nyquist Theorem) ต้องได้อย่างน้อย 2 ตัวอย่างต่อแต่ละลูกคลื่น อาจกล่าวอีกอย่างได้ว่าความถี่ข้อมูลป้อนเข้าที่สุดห้ามเกินครึ่งหนึ่งของอัตราการสุ่มตัวอย่าง ขั้นตอนการสอดแทรก sin x / x สามารถจำลองสัญญาณข้อมูลป้อนเข้าในดิจิไตเซอร์อย่างแม่นยำถูกต้องเหมาะสม ขั้นตอน sin x / x เหมาะกับส่วนโค้งระหว่างจุดตัวอย่างเพื่อแสดงรูปคลื่นที่เรียบ มีข้อเสียการสอดแทรก sin x / x สามารถขยายสัญญาณรบกวน เนื่องจากสัญญาณรบกวนมีอยู่ในสัญญาณจริง และในดิจิไตเซอร์ ดังนั้น sin x / x ควรจะใช้อย่างระมัดระวังโดยเฉพาะการสุ่มตัวอย่างเพียง 2 ตัวอย่างต่อรูปคลื่น

ขั้นตอน sin x / x จะสร้างโอเวอร์ชู้ทและพรีชู้ทบนขอบที่ไม่เป็นที่ต้องการด้วย ดังนั้นจึงต้องการตัวอย่างข้อมูลอย่างน้อย 2 อันบนขอบของสัญญาณ ผู้ใช้สามารถตรวจสอบจุดข้อมูลดิบที่ได้มาในสโคปโดยใช้แสดง sin x / x

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

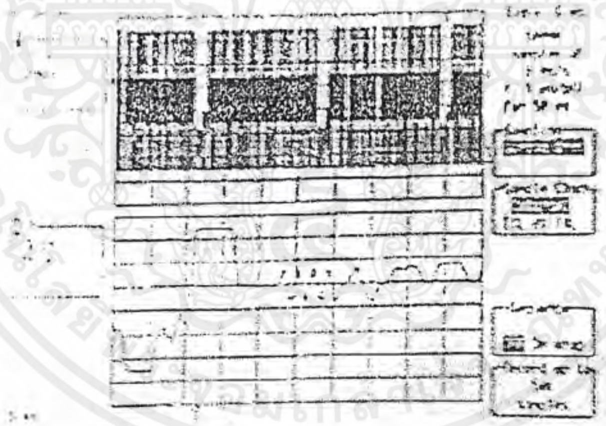
สำหรับการแทนรูปคลื่นที่แม่นยำกว่า ดิจิไตเซอร์ควรบันทึกอย่างน้อย 4 จุด ตัวอย่างต่อลูกคลื่นของส่วนประกอบของคลื่นซายน์ที่มีความถี่สูงสุด จุดตัวอย่างที่เพิ่มเติมเพิ่มอัตราส่วนของสัญญาณต่อสัญญาณรบกวน สำหรับการสอดแทรก sin x / x

การสอดแทรกเส้นตรงสามารถแสดงรูปคลื่นที่ถูกต้องแม่นยำโดยปราศจากการขยายสัญญาณรบกวน เพื่อผลที่ดีจะต้องการอย่างน้อย 10 ตัวอย่างต่อลูกคลื่น

3.10 ประโยชน์ของหน่วยความจำที่ยาวในดิจิตอลสตอเรจออสซิลโลสโคป

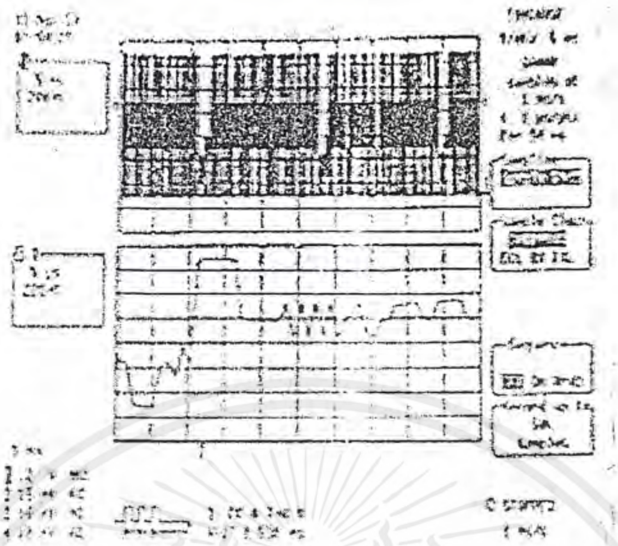
การเพิ่มความยาวของหน่วยความจำในดิจิตอลสตอเรจออสซิลโลสโคป ทำให้เกิดข้อได้เปรียบหลายประการ คือ

1. ไม่สูญเสียข้อมูลของรูปคลื่น เนื่องจากอัตราการสุ่มลูกคลื่นที่มีประสิทธิภาพสูง
2. จับสัญญาณกลิทช์ได้โดยไม่ทำให้รูปคลื่นผิดรูปหรือบิดเบือนไป
3. ความละเอียดเรื่องความถี่และเวลาที่ดีกว่า
4. การจับที่เชื่อถือได้ของเหตุการณ์ซึ่งไม่สามารถทำนายได้ในเวลานั้น
5. ไม่เสียเวลาระหว่างเหตุการณ์แต่ละเหตุการณ์ที่ได้รับ



รูปที่ 3.10 แสดงรูปคลื่นออสซิลโลสโคปที่มีหน่วยความจำ 2.5M

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.11 แสดงรูปคลื่นดิจิตอลออสซิลโลสโคปที่มีหน่วยความจำ 50 k

3.10.1 การเก็บรายละเอียด

รูปที่ 3.10 และ 3.11 แสดงรูปคลื่นเดียวกันโดยออสซิลโลสโคปที่ต่างกัน มีหน่วยความจำ 2.5 M และ50 k ตามลำดับ การตัดสินใจที่ดีกว่าของออสซิลโลสโคปที่มีหน่วยความจำมากกว่าสามารถเห็นได้จากการเปรียบเทียบส่วนขยายของรูปคลื่นในรูปที่ 3.10 กับการขยายในรูปที่ 3.11 ของสโคปที่มีหน่วยความจำน้อยกว่า สโคปที่มีหน่วยความจำมากกว่าจะแสดงรูปคลื่นที่ไม่บิดเบือน โดยตัวอย่างที่ชัดเจน

จากตัวอย่างแสดงให้เห็นผลของความยาวที่บันทึกบนอัตราการสุ่มตัวอย่าง สโคปทั้งสองแสดง 50 msec ของข้อมูล ดังนั้น 50 k สโคปจะมีความเร็ว 1 MS/s และ 2.5 M สโคปจะมีความเร็ว 50MS/s

ดังนั้นอัตราการสุ่มตัวอย่างเป็นหน้าที่โดยตรงของความยาวของหน่วยความจำ (จะเป็นจริงถึงขีดจำกัดของอัตราการสุ่มตัวอย่างสูงสุดของสโคป) ดังนั้นหน่วยความจำที่ยาวกว่าจะรักษาแบนวิดธ์ไว้มากกว่าการตั้งเวลาต่อช่วงโดยปราศจากอะไรที่ครึ่ง ๆ กลาง ๆ กับอัตราการสุ่มตัวอย่างที่ต่ำกว่า ถ้าสโคป 2 อัน มีความสามารถอัตราการสุ่มตัวเหมือนกัน สำหรับรูปคลื่นสั้น ๆ ดิจิตอลสตอเรจออสซิลโลสโคปที่มีหน่วยความจำยาวกว่าสามารถเก็บข้อมูลรูปคลื่นได้ยาวกว่าและให้แบนวิดธ์มากกว่าสำหรับสัญญาณที่ยาวกว่า

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.10.2 การจับสัญญาณกลิทช์ถาวร

ออสซิลโลสโคปแสดงข้อมูลโดยวิธีที่ต่างกัน บางอันแสดงเฉพาะส่วนเล็ก ๆ ของหน่วยความจำหรือเลื่อนตำแหน่งบนจอเพื่อแสดงข้อมูลที่เหลือ การแสดงจุดที่วัดทั้งหมดบนจอโดยวิธีที่รูปคลื่นที่สามารถแสดงร่วมกัน นี่เป็นการใช้วิธีทำให้รวมกันและรับรองกลิทช์การแสดงเป็นส่วนน้อย 1/8000000 ของรูปคลื่นที่แสดงจะถูกจับอย่างแม่นยำสม่ำเสมอ และแสดงผลเปรียบเทียบกับออสซิลโลสโคปซึ่งเชื่อใจได้ในการค้นพบจุดยอดเพื่อจับกลิทช์ ข้อมูลการค้นพบจุดยอดอาจจะมีประโยชน์ในการพิจารณา แต่มันจะให้ผลที่ไม่แน่นอน เพราะตลอดเวลาข้อมูลจะมีการเบ้เอียง

ดิจิตอลสตอเรจออสซิลโลสโคปบางอันจะมีส่วนสำหรับการค้นหาจุดยอดซึ่งอุปกรณ์แปลงสัญญาณอนาลอกเป็นดิจิตอล จะมีอัตราการสุ่มตัวอย่างเฉพาะจุดสูงสุดและต่ำสุดของค่าสัญญาณและจะถูกนำไปเก็บไว้ในหน่วยความจำ เราไม่สามารถรู้เวลาที่จะเกิดจุดยอด

3.10.3 การแก้ปัญหาเรื่องความถี่และเวลา

เปรียบเทียบออสซิลโลสโคปในในรูป 3.10 และ 3.11 ออสซิลโลสโคปอันแรกจะมีจุดตามแนวขวางมากกว่า 50 ครั้ง นั่นคือมีแฟคเตอร์ 50:1 การแก้ปัญหาทางแกน x ที่ดีจะปรับปรุงความแม่นยำถูกต้องของการวัดความสัมพันธ์ที่เวลาขณะใด ๆ และผลนี้ก็จะช่วยปรับปรุงการแสดงทางด้านความถี่ เพราะว่าจำนวนของจุดที่แสดงตามแบบ FFT จะเท่ากับจำนวนจุดที่บันทึกต้นแบบ

3.10.4 ความเชื่อถือได้ในการจับสัญญาณที่ไม่สามารถทำนายได้

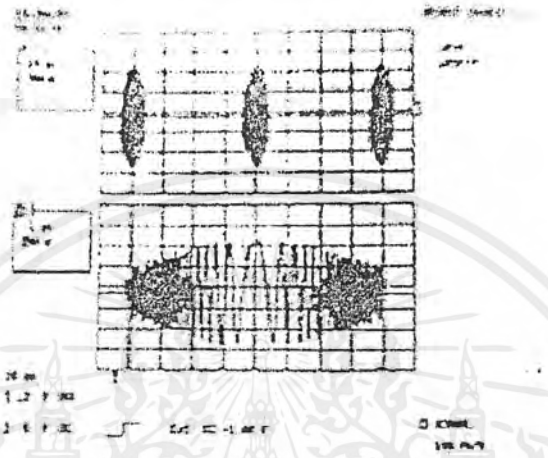
เหตุการณ์ที่เกิดขึ้นบางเหตุการณ์ไม่สามารถทำนายได้ซึ่งเป็นการยากที่จะทริกเกอร์ได้วิธีที่ง่ายคือใช้ออสซิลโลสโคปที่มีหน่วยความจำยาว ขบวนพัลส์จะถูกจับและขยาย มีการทำความเข้าใจกับความผิดพลาดที่เกิดขึ้น และได้ใช้สมาร์ททริกเกอร์กับเหตุการณ์เฉพาะที่เกิดขึ้นนี้

3.10.5 ไม่มีการสูญเสียเวลาระหว่างการรับข้อมูล

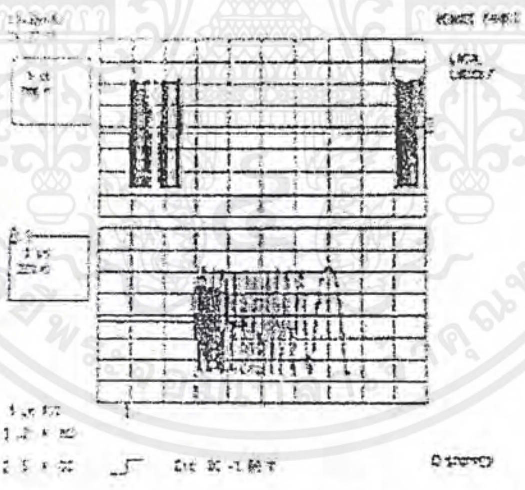
ช่วงเวลาระหว่างการได้มาจะมีช่วงเวลาที่แน่นอน ระหว่างช่วงเวลานี้สโคปมีขบวนการที่แสดงเป็นปกติ เวลานี้ทำให้เกิดปัญหาเมื่อได้รับเหตุการณ์ต่อเนื่องเข้ามา จากตัวอย่างเป็นสัญญาณเบิร์สต่อเนื่อง เพื่อความแม่นยำในการแสดงผลที่ต้องการอัตราการสุ่มตัวอย่างสัญญาณสูงสัมพันธ์กับช่วงระยะเวลา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 3.13 แสดงสัญญาณที่มีช่วงเวลาระหว่างสัญญาณเบิร์สที่ยาวกว่า ซึ่งจะใช้วิธีการแบ่งหน่วยความจำออกเป็นส่วนย่อย ๆ ทำให้สามารถลดเวลาเหลือน้อยกว่า 100 ไมโครวินาที ดังนั้นสัญญาณเบิร์สที่แสดงถูกเก็บเป็น 50 ส่วน ส่วนละ 1 กิโลไบต์



รูปที่ 3.12 รูปสัญญาณเบิร์สต่อเนื่อง



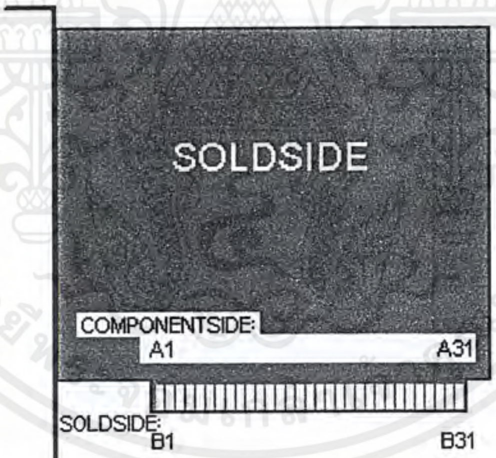
รูปที่ 3.13 แสดงสัญญาณที่มีช่วงเวลาระหว่างสัญญาณเบิร์สที่ยาวกว่า

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 4
สล็อต ISA ภายใน IBM/PC

สล็อต ISA ภายใน IBM/PC ได้มีการออกแบบให้สามารถที่จะเพิ่มเติมวงจรอินเตอร์เฟส เข้าไปในภายหลังได้ โดยผ่านทางสล็อตที่อยู่บนเมนบอร์ด (Main Board) แบ่งเป็น 2ลักษณะ ตามจำนวนบิตในการรับ-ส่ง ข้อมูล คือ

1) **แบบ 8 บิต** แต่ละสล็อตจะมีจำนวนขาทั้งสิ้น 62 ขา แบ่งออกเป็นสองข้าง ๆ ละ 31 ขา ส่วนการเรียกขาของสล็อตเหล่านี้ ขาทางด้านซ้ายจะเรียกโดยใช้อักษร B นำหน้าเลขตำแหน่งของขา เช่น B16 ก็คือขาทางด้านซ้ายของสล็อต ขาที่ 16 (นับจากทางด้านซ้ายของเครื่อง) ส่วนขาที่อยู่ทางด้านขวาของสล็อตจะเรียกโดยใช้อักษร A นำหน้าเลขตำแหน่งของขา เช่น ขา A24 ก็คือขาทางด้านขวาของสล็อตขาที่ 24 (นับจากทางด้านท้ายของเครื่อง) โดยโครงงานนี้ใช้การอินเตอร์เฟสแบบนี้ ดังรูปที่ 4.1



(ก)

(ข)

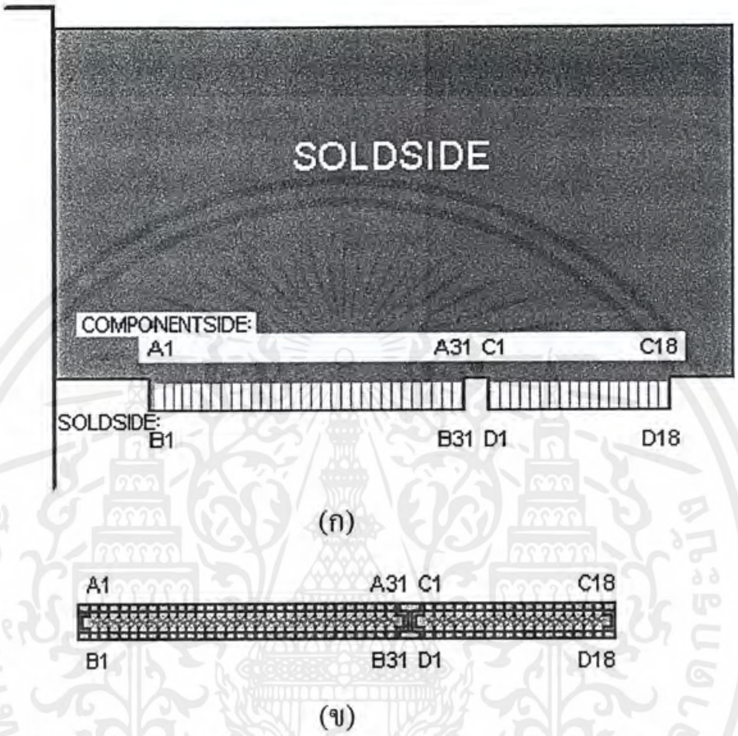
รูปที่ 4.1 แสดงการอินเตอร์เฟสบนสล็อต ISA แบบ 8 บิต

(ก) การ์ดอินเตอร์เฟสแบบ 8 บิต

(ข) สล็อต ISA แบบ 8 บิต

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2) **แบบ 16 บิต** ต่อมาเมื่อพัฒนาเป็นเครื่องเอทีทำให้มีขีดความสามารถเพิ่มขึ้น เช่นมี Data bus 16 bits จึงจำเป็นต้องมีสล็อตเสริมขึ้นมาอีก 36 ขา โดยขาที่อยู่ที่ทางด้านซ้ายจะใช้อักษร D นำหน้าเลขตำแหน่งของขา ส่วนขาที่อยู่ทางด้านขวาของสล็อตจะใช้อักษรC ดังรูปที่ 4.2



รูปที่ 4.2 แสดงการอินเตอร์เฟสทางสล็อต ISA แบบ 16 บิต

(ก) การ์ดอินเตอร์เฟสแบบ 16 บิต

(ข) สล็อต ISA แบบ16 บิต

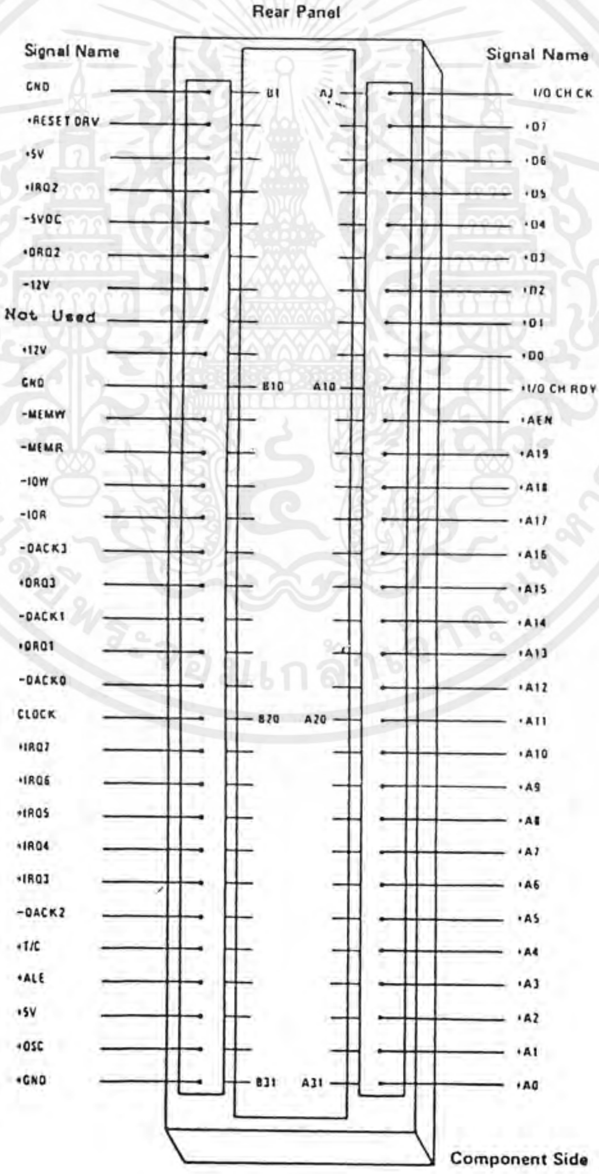
แต่ละขาของสล็อตเหล่านี้จะเชื่อมต่อกับเส้นสัญญาณต่าง ๆ บนเมนบอร์ด ทำให้การสร้างวงจรอินเตอร์เฟสกับ IBM/PC สามารถทำได้โดยสะดวก ซึ่งเส้นสัญญาณที่เชื่อมต่อกับขาของสล็อตเหล่านี้จะประกอบไปด้วย

- เส้นสัญญาณของบัสแอดเดรส (Address Bus) และบัสข้อมูล (Data Bus)
- บัสควบคุมสำหรับการเขียน/อ่านข้อมูลจากหน่วยความจำ หรือพอร์ต I/O
- เส้นสัญญาณสำหรับการขออินเตอร์เฟส
- เส้นสัญญาณสำหรับการขอ DMA
- สัญญาณฐานเวลา (Timing Signal) ต่าง ๆ ที่ใช้ในระบบ
- เส้นสัญญาณแสดงการรีเฟรชหน่วยความจำ
- เส้นสัญญาณสำหรับการตรวจสอบความผิดพลาด (I/O CHCK)
- ขาไฟเลี้ยง+5V,-5V,+12V,-12V

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.1 รายละเอียดของสัญญาณที่ขาต่าง ๆ ของสล็อต ISA

OSC (Oscillator ; ขา B30) : ขานี้เป็นเอาท์พุทที่เชื่อมต่อสัญญาณคล็อคที่มีค่าความถี่สูงสุดบนเมนบอร์ด คือ 14.31818 MHz ซึ่งมีคาบเวลาประมาณ 70 ns และมี Duty Cycle 6 ช่วง (เวลาใน 1 คาบที่สัญญาณคล็อกมีลอจิกเป็น 1 หารด้วยคาบเวลาทั้งหมด) ประมาณ 50% สัญญาณคล็อกอื่น ๆ ของระบบ เช่น คล็อกที่ป้อนให้กับ 8088 หรือ ชิพซัพพอร์ทต่าง ๆ นั้นจะถูกสร้างขึ้นโดยการหารสัญญาณนี้ อย่างไรก็ตามสิ่งหนึ่งที่จะต้องคำนึงถึงในการใช้งานสัญญาณ OSC ก็คือ สัญญาณนี้จะไม่ Synchronize กับสัญญาณอื่น ๆ บนบัสของระบบ ดังนั้นจึงไม่ควรที่จะนำสัญญาณจากขา OSC นี้ไปใช้เป็นสัญญาณคล็อกสำหรับวงจรภายนอกอื่น ๆ ที่ทำงานร่วมกับระบบ



รูปที่ 4.3 ขาสัญญาณต่าง ๆ ของ ISA bus

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

CLK (Clock ; ขา B20) : ขาสัญญาณนี้เป็นเอาท์พุท ซึ่งต่อกับสัญญาณคล็อกที่ถูกสร้างขึ้นโดยการหารสัญญาณ OSC ด้วย 3 ทำให้ได้ความถี่ประมาณ 4.77 MHz (14.31818 MHz/3) หรือ มีช่วงเวลาใน 1 คาบ (ช่วงเวลาของคล็อก 1 ลูก) เท่ากับ 210 ns (1/4.77 MHz) สำหรับค่า Duty Cycle) ของสัญญาณนี้จะมีค่าประมาณ 1/3 คือ ใน 1 คาบจะมีช่วงเวลาที่เป็นลอจิก 1 เท่ากับ 1/3 ของคาบเวลาทั้งหมด หรือประมาณ 70 ns และช่วงเวลาที่เป็นลอจิก 0 เท่ากับ 2/3 ของคาบทั้งหมด หรือประมาณ 140 ns สัญญาณนี้เป็นสัญญาณที่ถูกใช้เป็นคล็อกของระบบ

RESET DRV (ขา Bs) : ขาสัญญาณนี้เป็นเอาท์พุท ซึ่งจะแอคทีฟ (ลอจิก 1) ในช่วงที่เราเริ่มจ่ายไฟให้กับระบบ และจะยังคงแอคทีฟไปจนกว่าระบบต่าง ๆ ภายใน IBM/PC จะพร้อมที่จะทำงานได้จากนั้นสัญญาณนี้ก็จะเปลี่ยนกลับเป็นลอจิก 0 นอกจากนี้ในระหว่างการทำงานของ IMM/PC ถ้าระดับแรงดันของแหล่งจ่ายไฟลดลง สัญญาณนี้ก็จะถูกทำให้แอคทีฟเช่นกัน โดยทั่วไปแล้วสัญญาณนี้จะถูกนำไปใช้ในการรีเซตวงจรอินเตอร์เฟสหรืออุปกรณ์ I/O ต่าง ๆ ในช่วงที่เริ่มจ่ายไฟให้กับระบบ วึ่งจะเป็นการทำให้วงจรหรืออุปกรณ์เหล่านั้นถูกปรับให้อยู่ในสภาวะที่แน่นอนก่อนที่จะเริ่มต้นการทำงานในระบบ (สภาวะนี้เป็นสภาวะที่เราทราบ และต้องการให้วงจรทำงานในขณะที่ระบบถูกรีเซต)

A0-A19 (Address Bus ; ขา A31-A12) : ขาสัญญาณทั้ง 20 ขานี้เป็นเอาท์พุท ซึ่งใช้สำหรับกำหนดแอดเดรสของหน่วยความจำหรืออุปกรณ์ I/O ที่ 8088 ต้องการติดต่อด้วย โดยที่สัญญาณ A0 จะมีนัยสำคัญต่ำสุด (Least Significant Bit) และ A19 นี้จะมีนัยสำคัญสูงสุด (Most Significant Bit) สำหรับค่าแอดเดรสบนบัสแอดเดรส A0-A19 นี้ จะถูกกำหนดโดย 8088 ในระหว่างขบวนการอ่าน/เขียนข้อมูลลงในหน่วยความจำหรืออุปกรณ์ I/O แต่ในช่วงของขบวนการ DMA นั้น DMA-Controler จะเป็นผู้กำหนดค่าแอดเดรสบนบัสแอดเดรสเอง (ในระหว่างนี้ 8088 จะถูกตัดออกจากระบบ)

จะเห็นได้ว่าจำนวนเส้นแอดเดรสนี้จะมีอยู่ 20 เส้น ซึ่งสามารถที่จะอ้างแอดเดรสของหน่วยความจำได้ถึง 1 Mbyte แต่อย่างไรก็ตามจะมีแอดเดรสบางแอดเดรสที่ถูกใช้โดยระบบจำนวน 64 Kbyte (สำหรับ IBM PC/XT จะเป็นจำนวน 256 Kbyte) และแอดเดรสสำหรับหน่วยความจำ ROM อีก 48 Kbyte ซึ่งถูกจัดในช่วงแอดเดรสบนสุดใน 1 Mbyte คือ 0FC00H จนถึง 0FC00H จนถึง 0FFFFFH (สำหรับ IBM PC/XT จะเป็น 64 Kbyte)

สำหรับการอ้างแอดเดรสของพอร์ท I/O นั้น จะใช้เส้นแอดเดรสเพียง 16 เส้น คือ A0-A15 ซึ่งจะทำให้อ้างแอดเดรสของพอร์ทได้ 64 K พอร์ท โดยผ่านทางชุดคำสั่ง IN และ OUT ส่วนเส้นแอดเดรสที่เหลือ คือ A16-A19 นั้นจะไม่ถูกใช้งาน อย่างไรก็ตามใน IBM/PC จะใช้เส้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

แอดเครสในการอ้างแอดเครสของพอร์ทเพียง 10 เส้น คือ จาก A0-A9 และค่าแอดเครสที่ใช้งานจะต้องอยู่ในช่วง 0200H จนถึง 03FFH เท่านั้น

D0-D7 (Data Bus ; ขา A9-A2) : ขาสัญญาณนี้จะเป็นแบบ Bi-Direcional ซึ่งต่อกับบัสข้อมูลของระบบ เพื่อทำหน้าที่ในการส่งผ่านข้อมูลระหว่างพอร์ท I/O กับ IBM/PC โดยบิท D0 จะมีนัยสำคัญต่ำสุดและบิท D7 จะมีนัยสำคัญสูงสุด

สำหรับบัสไซเคิลของการเขียนข้อมูลที่สร้างโดย 8088 นั้น ข้อมูลจะถูกส่งออกมาบนบัสข้อมูล ก่อนที่สัญญาณ IOW (ในกรณีที่ต้องการส่งข้อมูลให้กับพอร์ท) หรือ MEMW (ในกรณีที่ต้องการส่งข้อมูลให้กับหน่วยความจำ) จะเปลี่ยนจากลอจิก 0 เป็น 1 (ขอบขาขึ้น) ซึ่งโดยทั่วไปขอบขาขึ้นของสัญญาณ IOW หรือ MEMW นี้ จะถูกใช้เพื่อสั่งให้พอร์ท I/O หรือหน่วยความจำที่มีแอดเครสตรงกับค่าแอดเครสบนบัสแอดเครสนั้นรับข้อมูลไปเก็บไว้

สำหรับในบัสไซเคิลการอ่านข้อมูลที่สร้างขึ้นโดย 8088 นั้น พอร์ท I/O หรือหน่วยความจำที่ถูกอ้างถึงจะต้องส่งข้อมูลออกมาบนบัสข้อมูล ก่อนที่ IOR (ในกรณีที่ต้องการอ่านข้อมูลจากพอร์ท) หรือ MEMR (ในกรณีที่ต้องการอ่านข้อมูลจากหน่วยความเร็ว) จะเปลี่ยนจากลอจิก 0 เป็นลอจิก 1 (ขอบขาขึ้น)

ALE (Address Latch Enable ; ขา B28) : ขาสัญญาณนี้เป็นขาเอาท์พุทที่ 8088 Bus Controller สร้างขึ้นเพื่อใช้สำหรับแสดงการเริ่มต้นของบัสไซเคิล และแสดงให้อุปกรณ์ภายนอกทราบว่าแอดเครสที่ 8088 ต้องการจะติดต่อด้วยนั้นถูกส่งออกมาบนบัสแอดเครสข้อมูลแล้ว โดยที่สัญญาณ ALE นี้จะเปลี่ยนจากลอจิก 1 เป็น 0 เมื่อค่าแอดเครสที่ถูกต้องถูกส่งออกมาบนบัสข้อมูเรียบร้อยแล้ว ดังนั้นขอบขาลงของสัญญาณ ALE นี้จะถูกใช้ในการแลทช์ค่าแอดเครสจากบัสแอดเครส/ข้อมูล Address/Data Bus ; AD0-AD7) ของ 8088 เท่านั้น โดยจะไม่แอคทีฟในระหว่างขบวนการ DMA

I/OCHCK (I/O Channel Chack ; ขา A1) : ขาสัญญาณนี้เป็นอินพุทที่ใช้ในการแสดงความผิดพลาดเกี่ยวกับพาริตี้ ที่เกิดขึ้นในการทำงานของวงจรอินเตอร์เฟสหรืออุปกรณ์ I/O เมื่อขานี้ได้รับลอจิก 0 จะทำให้ 8088 ถูกอินเตอร์รัพท์แบบ Non-Maskable (NMI) อย่างไรก็ตามเราสามารถที่จะกำหนดให้วงจรภายในของ IBM/PC ทำการขออินเตอร์รัพท์ (เมื่อได้รับสัญญาณ I/O CHCK) หรือไม่ก็ได้ โดยการกำหนดลอจิกของบิทข้อมูลของพอร์ที่ควบคุมการขออินเตอร์รัพท์แบบ NMI คือ บิท D7 ของพอร์ท 00A0H ในกรณีที่บิท D7 ของพอร์ท 00A0H ถูกเซ็ทเป็น 1 ก็จะให้วงจรภายนอกขออินเตอร์รัพท์แบบ NMI ได้ (Enable) แต่ถ้าบิท D7 ของพอร์ท 00A0H ถูกเซ็ทเป็น 0 ก็จะการดิสเอเบิล (Disable) การขออินเตอร์รัพท์แบบ NMI ดังนี้

Enable : ใช้คำสั่ง OUT ส่งข้อมูล 80H ไปยังพอร์ท 00A0H

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Disable : ใช้คำสั่ง OUT ส่งข้อมูล 00H ไปยังพอร์ท 00A0H

และเนื่องจากที่มีอุปกรณ์อื่นที่สามารถขออินเตอร์รัพท์แบบ NMI ได้อีก ดังนั้นซอฟแวร์ที่ใช้งานจะต้องสามารถตรวจสอบว่าการขออินเตอร์รัพท์นั้นเกิดขึ้นจากแหล่งใดได้ด้วย

I/O CHRDY (I/O Channel Ready ; ขา A10) : ขาสัญญาณนี้เป็นอินพุทที่ใช้เพิ่มช่วงเวลาในบัสไซเคิลกรณีที่อุปกรณ์ I/O หรือหน่วยความจำที่เกี่ยวข้องกับขบวนการในบัสไซเคิลที่เกิดขึ้นนั้น ไม่สามารถทำงานทันตามช่วงเวลาปกติของบัสไซเคิลนั้น ๆ ได้ (ช่วงเวลาของบัสไซเคิลที่เกี่ยวกับหน่วยความจำใช้ช่วงเวลาเท่ากับช่วงเวลาคล็อก 4 ลูก หรือ 840 ns ในขณะที่บัสไซเคิลที่เกี่ยวกับ I/O จะใช้ช่วงเวลาเท่ากับช่วงเวลาของคล็อก 5 ลูก หรือ 1.05μS)

เมื่ออุปกรณ์ I/O หรือหน่วยความจำต้องการที่จะเพิ่มช่วงเวลาในบัสไซเคิลให้นานขึ้นอีกนั้น จะสามารถทำได้โดยการป้อนลอจิก 0 ให้กับขา I/O หรือหน่วยความจำที่ถูกกำหนดนั้น ได้รับสัญญาณจากการดีโค้ดแอดเดรส และสัญญาณ MEMR, MEMW, IOR หรือ IOW แอคทีฟ

IRQ2-IRQ7 (Interrupt Request 2-7 ; ขา B4 และ B25-B21) : ขาสัญญาณทั้ง 6 นี้เป็นขาอินพุทที่ใช้สำหรับการขออินเตอร์รัพจาก 8088 โดยสัญญาณเหล่านี้จะต่อเข้ากับ 8259A บนเมนบอร์ดโดยตรง โปรแกรมในส่วน BIOS ของ IBM/PC จะทำให้การโปรแกรม 8259A ให้ IRQ2 มีลำดับความสำคัญสูงสุด (Highest Priority) และ IRQ7 มีลำดับความสำคัญที่ต่ำสุด ในกรณีที่มีการขออินเตอร์รัพท์เกิดขึ้น คือระดับลอจิกที่ขา IRQ ขาใดขาหนึ่งถูกเปลี่ยนจากลอจิก 0 เป็นลอจิก 1 (ขอบขาขึ้น) 8259A ก็จะทำการส่งสัญญาณ INT ให้กับ 8088 เพื่อทำการขออินเตอร์รัพท์

สิ่งสำคัญในการขออินเตอร์รัพท์ผ่านทาง IRQ2-IRQ7 นี้ ก็คืออุปกรณ์ที่จะทำการขออินเตอร์รัพท์โดยผ่านทาง IRQ ขาใดก็จะต้องรักษาระดับสัญญาณที่ขา IRQ นั้น ให้แอคทีฟ (ลอจิก 1) อยู่จนกว่าจะได้รับสัญญาณ INTA (Interrupt Acknowledge) จาก 8088 เสียก่อน ถ้าไม่เช่นนั้นการขออินเตอร์รัพท์จะถูกยกเลิก และอินเตอร์รัพท์ Level 7 (IRQ7) ก็จะถูกสร้างขึ้นโดยอัตโนมัติไม่ว่าการขออินเตอร์รัพท์ที่ถูกยกเลิกนั้นจะเป็นการขออินเตอร์รัพท์ใน Level หรือขาใด

แต่อย่างไรก็ตามสัญญาณ INTA นี้จะไม่ถูกต่อออกมาที่ขาของสล็อตด้วย ดังนั้นโปรแกรมที่ทำการตอบสนองต่อการขออินเตอร์รัพท์ (Interrupt Service Routine) จะต้องทำการรีเซ็ทสัญญาณ IRQ เอง โดยใช้คำสั่ง OUT ไปยังพอร์ท I/O ที่เกี่ยวข้อง

IOR (I/O Read ; ขา B14) : ขาสัญญาณนี้เป็นขาเอาท์พุทแอคทีฟที่ลอจิก 0 ที่สร้างขึ้นโดย 8288 Bus Controller เพื่อใช้ในการแสดงว่าบัสไซเคิลที่เกิดขึ้นนี้เป็นบัสไซเคิลที่อ่านข้อมูลจากพอร์ท I/O เพื่อให้พอร์ท I/O ที่มีแอดเดรสที่ตรงกับที่แอดเดรสบนบัสแอดเดรสนั้นส่งข้อมูลออกมาบนข้อมูลโดยข้อมูลจะต้องถูกส่งออกมาบนบัสข้อมูลก่อนขอบขาขึ้นของสัญญาณ IOR

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ประมาณ 30 ns เพื่อให้มั่นใจได้ว่า 8088 สามารถรับข้อมูลได้ถูกต้อง สำหรับขบวนการ DMA 8237A-5 DMA Controller จะทำการสร้างสัญญาณ IOR เอง โดยที่ค่าแอดเดรสที่อยู่บนบัสแอดเดรสจะเป็นค่าแอดเดรสของหน่วยความจำ (แทนที่จะเป็นแอดเดรสของพอร์ท I/O) ที่พอร์ท I/O ที่ขอ DMA ต้องการจะนำข้อมูลไปเก็บ การที่พอร์ทใดจะส่งข้อมูลออกมาบนบัสข้อมูลนั้น จะต้องอาศัยสัญญาณ DACK จาก DMA Controller เป็นตัวกำหนด เช่นกรณีที่สัญญาณ DACK1 แอคทีฟก็จะแสดงว่าพอร์ท I/O ที่จะต้องส่งข้อมูลออกมา ก็คือพอร์ท I/O ที่ขอ DMA ผ่านทางแชนแนลที่ 1 (DRQ1) เป็นต้น

IOW (I/O Write ; ขา B13) : ขาสัญญาณนี้เป็นเอาท์พุทแอคทีฟที่ลอจิก "0" ซึ่งถูกสร้างขึ้นโดย 8288 Bus Controller เพื่อใช้แสดงว่าบัสไซเคิลที่เกิดขึ้นนี้เป็นบัสไซเคิลของการเขียนข้อมูลลงบนพอร์ท I/O ที่มีแอดเดรสตรงกับแอดเดรสบนบัสแอดเดรสนั้น รับข้อมูลที่อยู่บนบัสข้อมูลไปเก็บไว้อย่างไรก็ตามเนื่องจากในช่วงเวลาที่สัญญาณ IOW นี้แอคทีฟ (ลกจิก "0") นั้น ข้อมูลบนบัสแอดเดรสอาจจะยังไม่สมบูรณ์ ดังนั้นในการออกแบบจึงควรใช้ขอบขาขึ้นของสัญญาณ IOW แทนขอบขาลงในการทำให้พอร์ท I/O ที่เกี่ยวข้องรับข้อมูลไปเก็บไว้ เพื่อให้ข้อมูลบนบัสข้อมูลเสร็จเสียก่อน สำหรับในขบวนการ DMA นั้น DMA-Controller จะทำการสร้างสัญญาณ IOW เอง โดยให้ค่าแอดเดรสที่อยู่บนบัสแอดเดรสจะเป็นค่าแอดเดรสของหน่วยความจำที่พอร์ท I/O ที่ขอ DMA ต้องการจะอ่านข้อมูล

MEMW (Memory Write ; ขา B11) : ขานี้เป็นเอาท์พุทแอคทีฟที่ลอจิก "0" ซึ่ง 8288 Bus Controller สร้างขึ้นในระหว่างบัสไซเคิลในการเขียนข้อมูลลงในหน่วยความจำของ 8088 สัญญาณ MEMW นี้จะถูกส่งออกมาเพื่อให้หน่วยความจำที่แอดเดรสตรงกับค่าแอดเดรสบนบัสแอดเดรสนั้น ทำการรับข้อมูลที่อยู่บนบัสข้อมูลไปเก็บไว้ โดยทั่วไปหน่วยความจำจะรับรู้ข้อมูลในช่วงขอบขาขึ้นของสัญญาณ MEMW

สำหรับในระหว่างขบวนการ DMA นั้น 8237A-5 DMA-Controller จะทำการควบคุมบัสต่าง ๆ ของระบบแทน 8088 และสัญญาณ MEMW จะถูกใช้ในบัสไซเคิลของการเขียนข้อมูลลงในหน่วยความจำ (ข้อมูลถูกส่งจากอุปกรณ์ I/O ไปให้กับหน่วยความจำ)

MEMR (Memory Read ; ขา B12) : ขานี้เป็นขาเอาท์พุทจาก 8288 ซึ่งสัญญาณนี้จะแอคทีฟ (ลอจิก "0") ในระหว่างบัสไซเคิลของการอ่านข้อมูลจากหน่วยความจำของ 8088 เพื่อให้หน่วยความจำที่มีแอดเดรสกับค่าแอดเดรสบนบัสแอดเดรสนั้น ทำการส่งข้อมูลออกมาบนบัสข้อมูลโดยหน่วยความจำนั้นจะต้องส่งข้อมูลออกมาในช่วงเวลา 30 nanosec ก่อนที่สัญญาณ MEMW จะกลับเป็นลอจิก "1" ทั้งนี้ก็เพื่อให้ 8088 ได้รับข้อมูลที่ถูกต้อง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สำหรับในระหว่างขบวนการ DMA นั้น DMA-Controller จะควบคุมบัสต่าง ๆ ของระบบแทน 8088 และสัญญาณ MEMR จะถูกใช้ในบัสไซเคิลของการอ่านข้อมูลจากหน่วยความจำ (ข้อมูลถูกส่งจากหน่วยความจำไปให้กับอุปกรณ์ I/O)

DBQ 1-DRQ3 (DMA Request 1-3 ; ขา B18, B6 และ ขา B16) : ขาสัญญาณทั้งสามนี้เป็นสัญญาณอินพุทแอคทีฟที่ลอจิก "1" ซึ่งอุปกรณ์ภายนอกจะใช้ในการขอ DMA จากระบบ โดยการป้อนระดับสัญญาณลอจิก "1" ให้กับขา DRQ ขาใดขาหนึ่ง (ขา DRQ ทั้งสามนี้จะต่อเข้ากับ DRQ 1-DRQ3 ของ 8237A-5)

เมื่อ 8237A-5 ได้รับสัญญาณนี้แล้วก็จะตรวจสอบว่ามีการขอ DMA ในแชนแนลที่มีลำดับความสำคัญ (Priority) สูงกว่าหรือไม่ ถ้าไม่มีก็จะทำการขอ DMA จาก 8088 และตอบรับการขอ DMA จากอุปกรณ์ภายนอก (สัญญาณ DACK ของแชนแนลที่ขอ DMA จะแอคทีฟ) แต่ถ้ามี 8237A-5 ก็จะทำการขอ DMA ให้กับแชนแนลที่มีลำดับความสำคัญสูงกว่าแล้วจึงทำการขอ DMA ให้กับแชนแนลที่มีลำดับความสำคัญต่ำกว่า ภายใน ROM BIOS ของ IBM/PC จะโปรแกรม 8237A-5 ให้ DRQ1 มีลำดับความสำคัญสูงสุดและ DRQ3 มีลำดับความสำคัญต่ำสุด ดังนั้นถ้ามีการขอ DMA ของอุปกรณ์ภายนอกผ่านทางแชนแนลที่ 1 (DRQ1) และแชนแนลที่ 2 (DRQ2) 8237A-5 ก็จะทำการขอ DMA ให้กับแชนแนลที่ 1 ก่อน จากนั้นเมื่อเสร็จขบวนการ DMA ของแชนแนลที่ 1 แล้ว จึงทำการขอ DMA ให้กับแชนแนลที่ 1 ก่อน จากนั้นเมื่อเสร็จขบวนการ DMA ของแชนแนลที่ 1 แล้ว จึงทำการขอ DMA ให้กับแชนแนลที่ 2

อย่างไรก็ตาม 8237A-5 ยังมีแชนแนลสำหรับการขอ DMA อยู่อีก 1 แชนแนล คือแชนแนลที่ 0 (DRQ0) ซึ่งในความเป็นจริงแล้วแชนแนลนี้จะมีลำดับความสำคัญที่สูงกว่าแชนแนลที่ 1 แต่จะไม่ถูกต่อออกมายังขาของสล็อต เนื่องจาก IBM/PC จะใช้แชนแนลที่ 0 นี้ ในการรีเฟรชหน่วยความจำที่เป็น dynamic RAM

ในการขอ DMA นั้นสัญญาณ DRQ นี้ จะต้องแอคทีฟอยู่ในช่วงระยะเวลาหนึ่งเท่านั้น ถ้าสัญญาณนี้แอคทีฟอยู่นานเกินไป จะทำให้เกิดขบวนการขอ DMA ขึ้นมากกว่า 1 ขบวนการได้ สำหรับวงจรที่ขอ DMA โดยทั่วไปจะใช้สัญญาณตอบรับการขอ DMA หรือสัญญาณ DACK ของแชนแนลที่ขอ DMA นั้น ในการรีเซ็ทสัญญาณ DRQ เช่นอุปกรณ์ภายนอกที่ขอ DMA ผ่านทางแชนแนลที่ 1 (DRQ1) ก็จะคอยตรวจสอบการตอบรับการขอ DMA จากสัญญาณ DACK ของแชนแนลที่ 1 (DACK) เมื่อได้รับสัญญาณจาก DACK1 แล้ว ก็จะรีเซ็ทสัญญาณ DRQ1 (เปลี่ยนจากลอจิก "1" เป็น "0")

DACK0-DACK3 (DMA Acknowledge 0-3 ; ขา B19, B17, B26 และ B15) : สัญญาณทั้ง 4 นี้เป็นเอาท์พุทแอคทีฟที่ลอจิก "0" ซึ่ง 8237A-5 สร้างขึ้นเพื่อแสดงให้วงจรภาย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

นอกที่ขอ DMA ทราบว่าการขอ DMA นั้นได้รับการตอบสนองแล้ว และ 8237A-5 จะเข้าสู่ขบวนการ DMA เพื่อให้การส่งผ่านข้อมูลระหว่างอุปกรณ์ I/O ที่ขอ DMA กับหน่วยความจำเกิดขึ้นได้โดยตรง (คือ ไม่ต้องผ่าน 8088) โดยสัญญาณ DACK นี้จะแอคทีฟในแชนแนลใดก็ขึ้นอยู่กับว่าขบวนการ DMA ที่จะเกิดขึ้นนั้น เป็นการตอบสนองต่อการขอ DMA ในแชนแนลใด เช่น ถ้าขบวนการขอ DMA ที่เกิดขึ้นนั้นเป็นการตอบสนองต่อการขอ DMA ในแชนแนลที่ 2 (DRQ2) สัญญาณ DACK2 ก็จะแอคทีฟ เป็นต้น

ดังที่ได้กล่าวแล้วว่าสัญญาณ DRQ0 นั้น จะไม่ถูกต่อออกมายังขาของสล็อต ดังนั้นวงจรอินเตอร์เฟสจึงไม่สามารถจะขอ DMA ผ่านทางแชนแนล 0 ได้ แต่สัญญาณ DACK0 จะถูกต่อออกมายังสล็อตด้วย (ขา B19) ทั้งนี้ก็เพื่อที่จะแสดงให้วงจรอินเตอร์เฟสต่าง ๆ ทราบว่าขบวนการ DMA ที่เกิดขึ้นในช่วงเวลาที่ DACK0 แอคทีฟนั้น เป็นขบวนการที่ใช้สำหรับการรีเฟรชหน่วยความจำที่เป็น Dynamic RAM ซึ่งวงจรอินเตอร์เฟสที่ใช้หน่วยความจำประเภทนี้สามารถจะนำไปใช้ในการรีเฟรช Dynamic RAM ที่อยู่ในวงจรได้ โดยที่การรีเฟรชหน่วยความจำนั้นจะต้องเกิดขึ้นในทุก ๆ 15.12 μsec ด้วย

AEN (Address Enable ; ขา A11) : สัญญาณนี้เป็นเอาท์พุทที่ใช้ในการแสดงว่าบัสไซเคิลที่เกิดขึ้นในช่วงเวลาที่สัญญาณ AEN แอคทีฟ (ลอจิก “1”) นั้น เป็นบัสไซเคิลของขบวนการ DMA

สำหรับเมนบอร์ดของ IBM/PC นั้น จะใช้สัญญาณนี้ในการดิสเอเบิล (Disable) 8288 Bus Controller และจะใช้ดิสเอเบิลพอร์ท I/O ต่างๆ ที่ไม่เกี่ยวข้องกับขบวนการ DMA ที่เกิดขึ้นนี้ ที่จำเป็นต้องทำเช่นนี้ก็เพราะในระหว่างขบวนการ DMA นั้น 8237A-5 จะส่งแอดเดรสของหน่วยความจำออกมาบนบัสแอดเดรส และจะทำให้สัญญาณ IOR หรือ IOW แอคทีฟด้วยดังนั้นถ้าไม่ทำการดิสเอเบิลพอร์ท I/O ที่ไม่เกี่ยวข้องไว้ ก็อาจจะทำให้พอร์ท I/O ที่มีแอดเดรสตรงกับค่าแอดเดรสบนบัสแอดเดรส (ซึ่งเป็นแอดเดรสของหน่วยความจำ) นั้นทำการอ่านหรือส่งข้อมูลออกมาบนบัสข้อมูลทำให้เกิดความผิดพลาดขึ้นได้

T/C (Terminal Count ; ขา B27) : สัญญาณนี้จะถูกสร้างขึ้นจากการนำเอาสัญญาณเอาท์พุทที่ขา EOP ของ 8237A-5 มากลับลอจิก (โดยใช้เกท Inverter) ทำให้สัญญาณ T/C นี้แอคทีฟที่ลอจิก “1”

สำหรับสัญญาณนี้จะแอคทีฟเมื่อจำนวนไบท์ในการส่งผ่านข้อมูลของขบวนการ DMA ในแชนแนลใดแชนแนลหนึ่ง ครบตามจำนวนที่กำหนดไว้ โดยทั่วไปแล้วสัญญาณที่จะถูกใช้ในการสิ้นสุดขบวนการ DMA ที่ทำการส่งผ่านข้อมูลเป็นบล็อก เนื่องจากสัญญาณนี้จะแอคทีฟโดยไม่แสดงว่าเป็นของแชนเนลใด ดังนั้นจึงต้องทำการนำสัญญาณ T/C นี้ผ่านเกท Inverter แล้วนำไป OR

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กับสัญญาณ DACK เพื่อให้สามารถทราบได้ว่า สัญญาณ T/C ที่เกิดขึ้นนั้นเป็นสัญญาณของแชนแนลใด สำหรับใน แชนแนลที่ 0 นั้น สัญญาณ T/C จะแอคทีฟในช่วงเวลาที่คงที่คือ ทุก ๆ 990.804 millisec ซึ่งก็คือช่วงเวลาในการรีเฟรชหน่วยความจำขนาด 64 Kbyte นั่นเอง

4.2 บัสแหล่งจ่ายไฟของระบบ

+5Vdc (ขา B3 และ B29) : ขาทั้งสองนี้ต่อกับแหล่งจ่ายไฟ DC + 5V ของระบบ โดยจะมีค่าความเที่ยงตรง (Regulate) ±5% คือ อยู่ในช่วง +4.75 ถึง +5.25 Vdc

+12Vdc (ขา B9) : ขานี้ต่อกับแหล่งจ่ายไฟ DC + 12V ของระบบ โดยจะมีค่าความเที่ยงตรง (Regular) ±5% คืออยู่ในช่วง +11.4 ถึง +12.6 Vdc

-5Vdc (ขา B5) : ขานี้ต่อกับแหล่งจ่ายไฟ DC –5V ของระบบ โดยมีความเที่ยงตรง (Regulate) ±10% คืออยู่ในช่วง -5.5 ถึง -4.5 Vdc

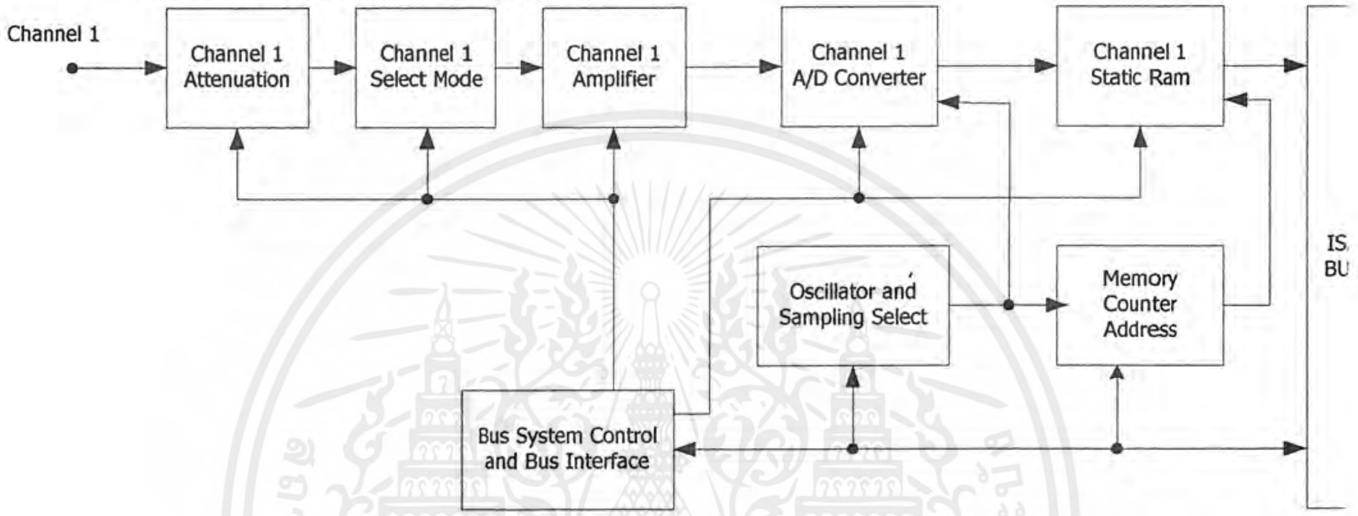
-12Vdc (ขา B7) : ขานี้ต่อกับแหล่งจ่ายไฟ DC –12V ของระบบ โดยจะมีค่าความเที่ยงตรง (Regulate) ±10% คืออยู่ในช่วง -13.2 ถึง -10.8 Vdc

GND (ขา B1, B10 และ B31) : ขาทั้งสามนี้จะต่อเข้ากับกราวด์ (Ground) ของระบบ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 5
หลักการออกแบบวงจรและโปรแกรมควบคุมการทำงาน

5.1 โครงสร้างของสตอเรจออสซิลโลสโคป



รูปที่ 5.1 แสดงบล็อคไดอะแกรมของดิจิตอลออสซิลโลสโคป

จากบล็อกไดอะแกรม สัญญาณอินพุทที่เข้ามาของแต่ละช่อง จะถูกส่งเข้าสู่ วงจร Atenuator เพื่อลดทอนแรงดันสูง ๆ ให้มีค่าขนาดสัญญาณลดลง วงจร Select Mode จะทำการเลือกว่าจะทำการวัดใน Mode AC, DC หรือ Ground สัญญาณที่ผ่านการ Select Mode จะถูกส่งเข้าไปให้วงจร Amplifier เพื่อขยายสัญญาณให้มีขนาดเพียงพอที่จะทำการ Sampling ได้ หลังจากผ่านวงจร Amplifier แล้ว ก็จะถูกส่งเข้าสู่วงจร A/D Converter ข้อมูลที่ได้จากการ Converter จะถูกส่งเข้าไปเก็บไว้ในหน่วยความจำ

ส่วนบล็อคไดอะแกรม Oscillator and sampling จะทำหน้าที่ผลิตความถี่ และเลือกความถี่ที่จะใช้ในการ sampling ส่วน Counter Address Memory จะนำความถี่ที่ใช้ในการ sacpling มาเป็น Counter ให้แก่แอดเดรสของ SRAM ส่วน Bus System & Bus Interface จะทำหน้าที่ควบคุมส่วนต่าง ๆ และเชื่อมต่อกับคอมพิวเตอร์ทาง ISA bus

5.2 ส่วนปรับแต่งขนาดและยกระดับสัญญาณอินพุท

เนื่องจากในส่วนแปลงสัญญาณอนาลอกเป็นสัญญาณดิจิตอลใช้วงจรรวมเบอร์ TDA 8703 มีลักษณะเฉพาะคือสามารถแปลงสัญญาณอนาลอกได้เฉพาะในช่วง 1.55 ถึง 3.26 โวลต์เท่านั้น จึงจำเป็นต้องมีวงจรในส่วนปรับแต่งขนาดและยกสัญญาณอินพุทขึ้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5.2.1 วงจรลดทอนสัญญาณ (Attenuator)

เนื่องจากสัญญาณที่เข้ามามีขนาดที่แตกต่างกัน ดังนั้นจะต้องปรับระดับสัญญาณให้เหมาะสมก่อนที่จะส่งไปยังภาคอื่น ๆ ต่อ ซึ่งการลดทอนสัญญาณจะใช้หลักการแบ่งแรงดัน (Voltage Divider) และการลดทอนสัญญาณจะต้องเป็นอิสระกับความถี่ ซึ่งจะต้องใช้วงจรลดทอนแบบชดเชยความถี่ (Compensated Attenuator) ซึ่งจะเป็นวงจร อาซีโวลท์เตจดีไวเดอร์ (RC Voltage Divider) ในวงจรมีการลดทอนทั้งหมด 2 ระดับ คือ 1/1 และ 1/10 ดังรูปที่ 5.2

5.2.2 วงจรบัฟเฟอร์ (Buffer)

สัญญาณที่ผ่านการลดทอนแบบชดเชยความถี่แล้ว จะถูกเลือกด้วยอิเล็กทรอนิกส์สวิตช์ และนำสัญญาณที่ได้ผ่านวงจรบัฟเฟอร์ เพื่อเป็นการส่งผ่านสัญญาณจากภาควงจรลดทอนซึ่งมีเอาท์พุทอินพิแดนซ์สูงไปยังภาควงจรขยาย โดยใช้ออปแอมป์ เป็นบัฟเฟอร์แบบ Inerting และมี VR_3 ใช้ในการปรับออฟเซทให้แก่ออปแอมป์ ดังรูปที่ 5.3

5.2..3 วงจรขยายสัญญาณแนวตั้ง (Vertical Amplifier)

โดยใช้วงจรขยายแบบกลับเฟส (Inverting Amplifier) ในการเลือกอัตราขยายจะใช้ไอซีเบอร์ 74HC4051 ซึ่งเป็นมัลติเพลกซ์ ส่วนไดโอด ใช้ในการจำกัดสัญญาณช่วงบวกไม่ให้ถูกขยายออกออปแอมป์ได้ดังรูปที่ 5.4

ในการขยายสัญญาณจะมีการขยายทั้งหมด 8 ระดับ ดังตาราง

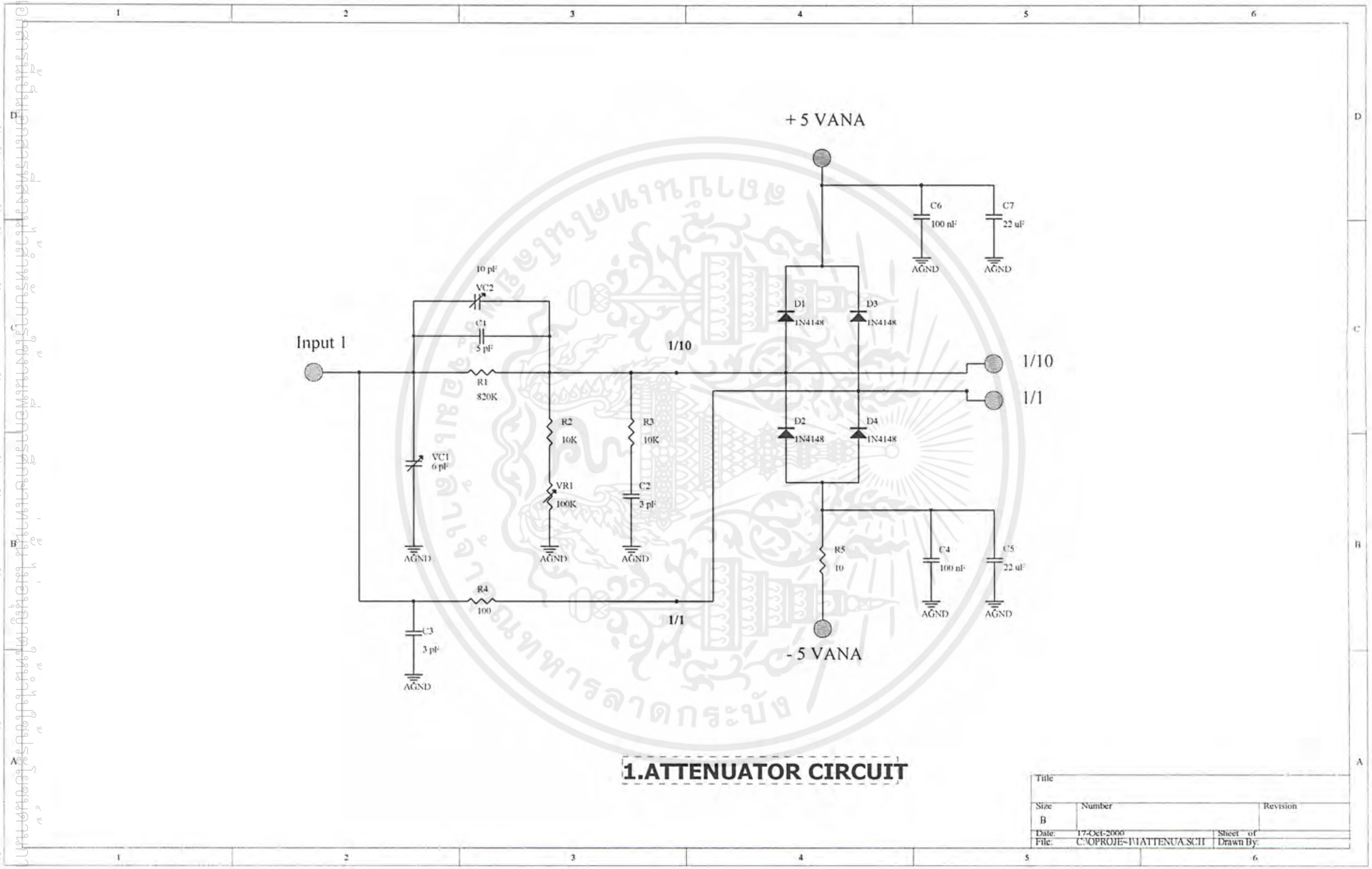
C	B	A	อัตราการขยาย (เท่า)
0	0	0	1
0	0	1	2
0	1	0	5
0	1	1	10
1	0	0	20
1	0	1	50
1	1	0	100
1	1	1	200

ตาราง 5.1 อัตราขยายของวงจรขยายสัญญาณ

5.2.4 วงจรยกระดับสัญญาณ (DC Adjust)

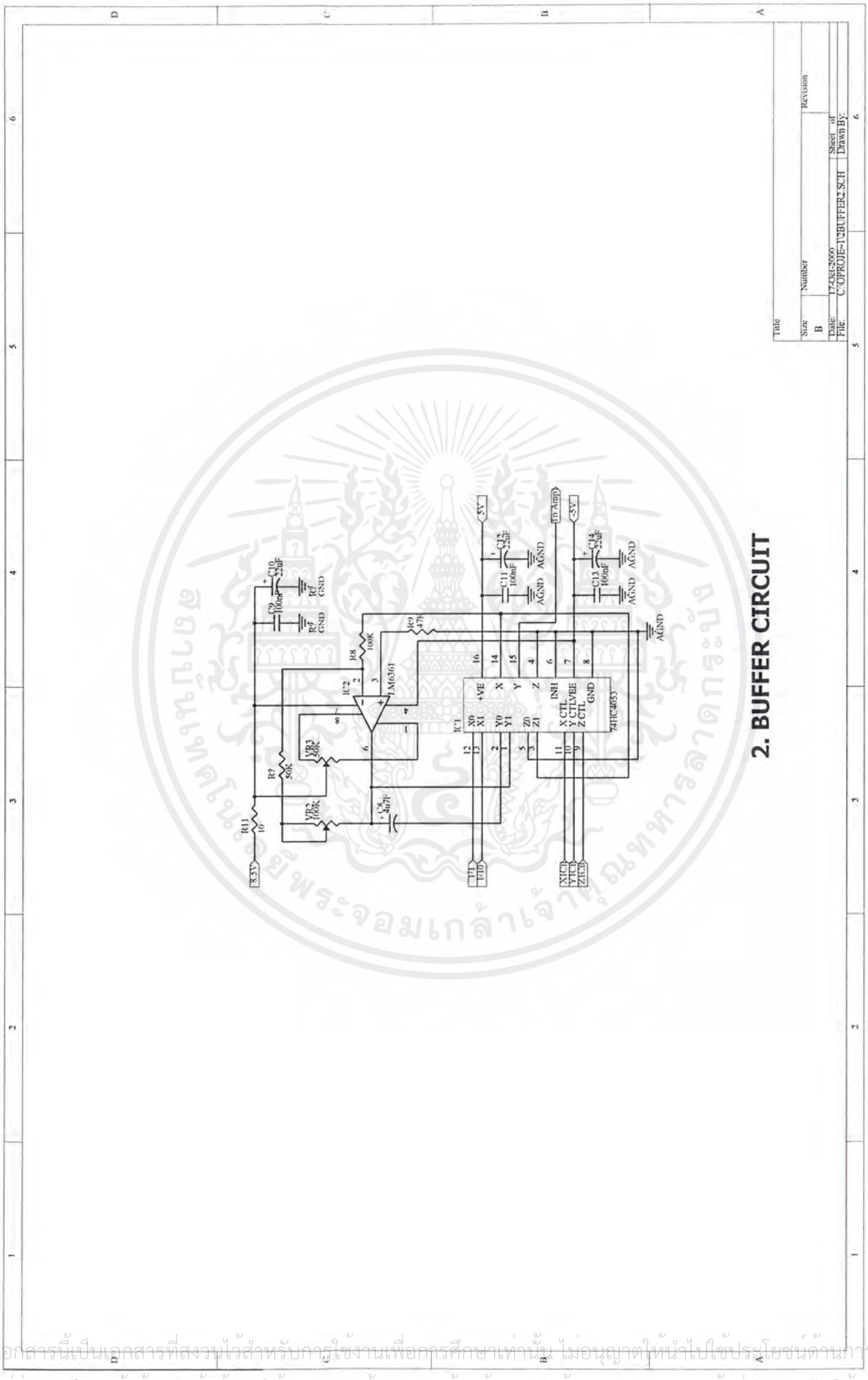
จะใช้ในการยกระดับสัญญาณให้อยู่ในช่วงที่สามารถนำสัญญาณอนาลอกที่ได้ไปแปลงเป็นสัญญาณดิจิตอลได้ ดังรูปที่ 5.5

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



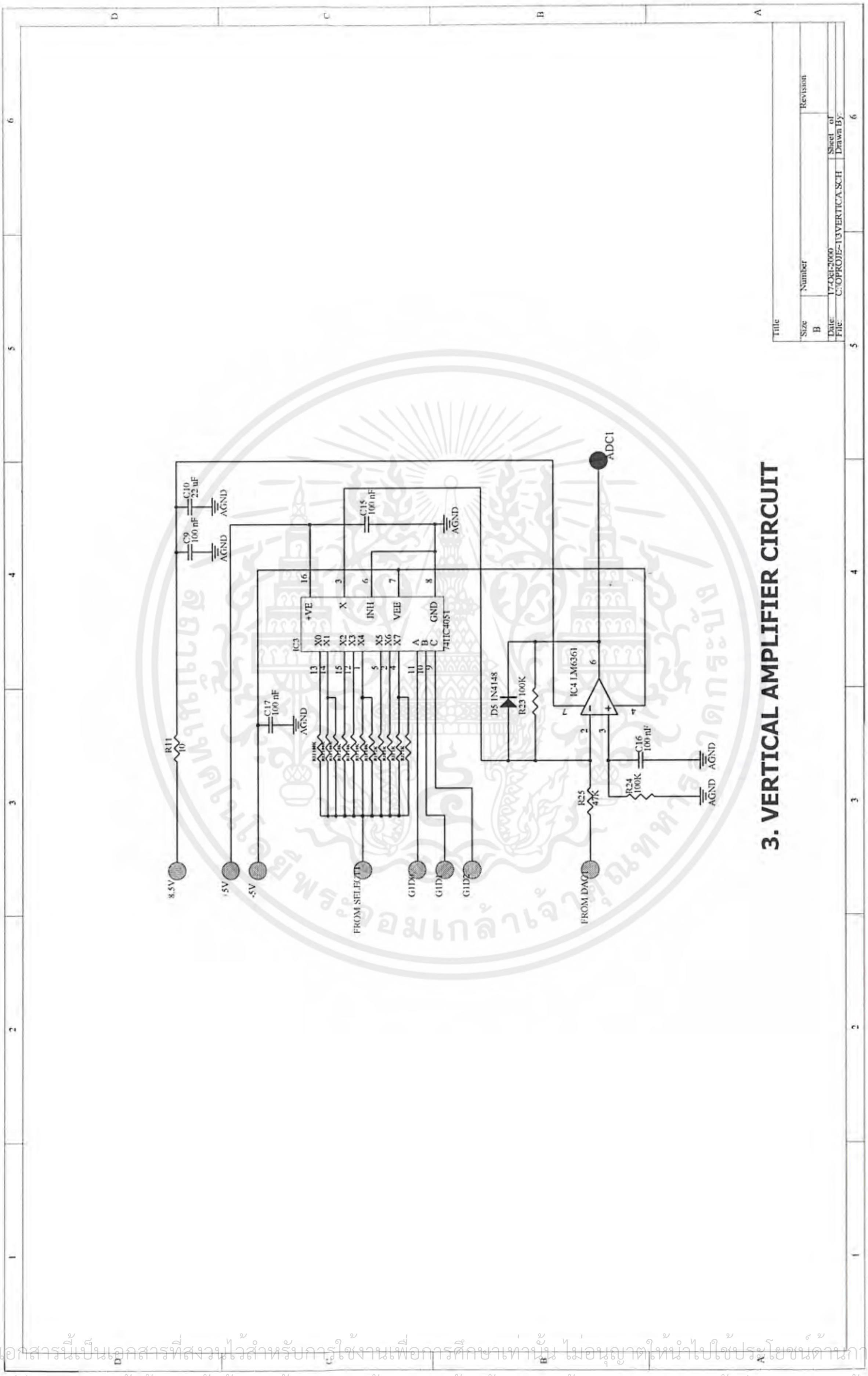
รูปที่ 5.2 แสดงวงจรลดทอนสัญญาณ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



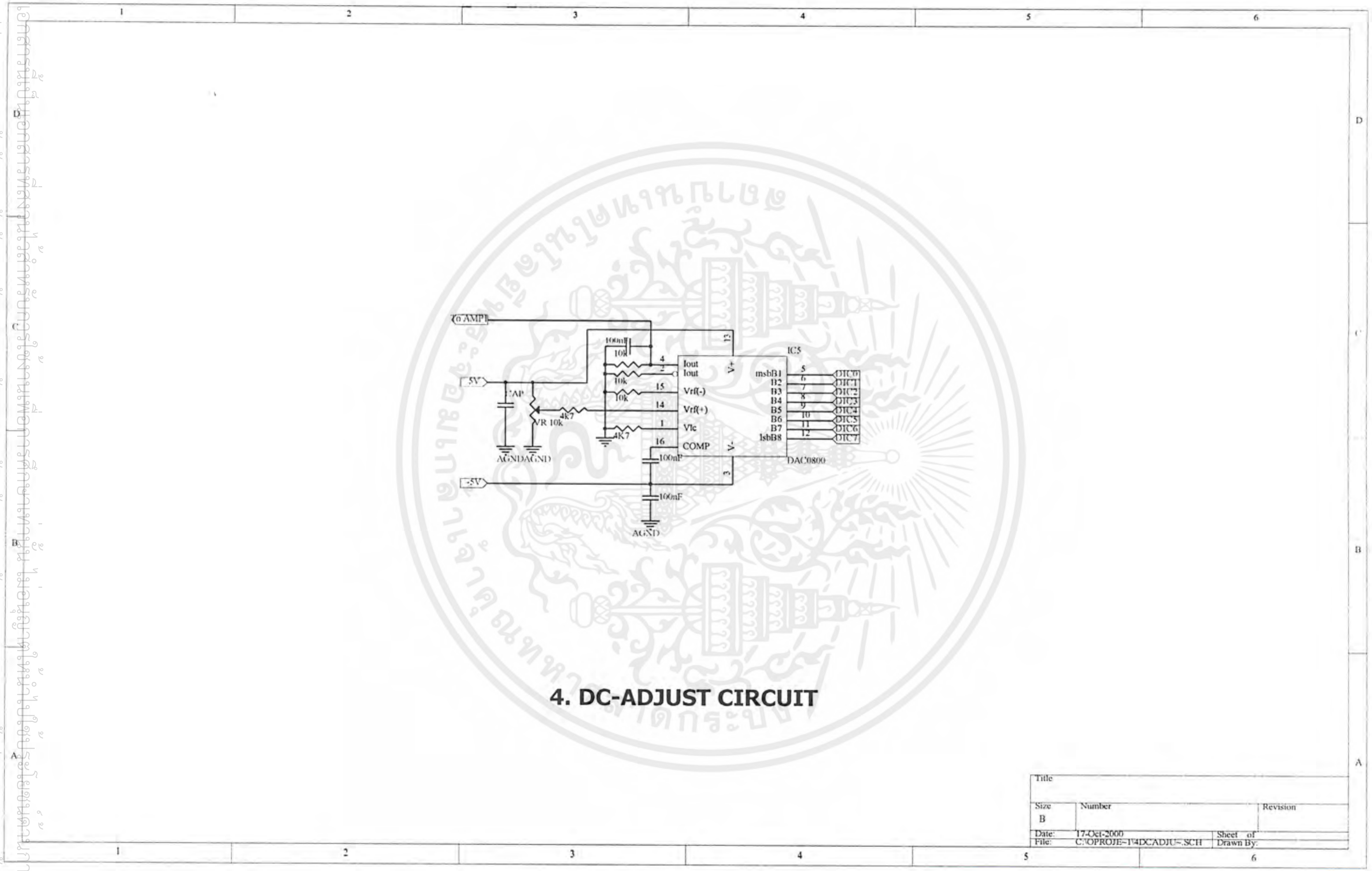
รูปที่ 5.3 แสดงวงจรบัฟเฟอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 5.4 แสดงวงจรขยายสัญญาณแนวตั้ง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 5.5 แสดงวงจรยกระดับสัญญาณ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5.3 วงจรแปลงสัญญาณอนาลอกเป็นสัญญาณดิจิตอล (Analog to Digital Converter)

วงจรแปลงสัญญาณอนาลอกเป็นดิจิตอล จะใช้ไอซีเบอร์ TDA 8703 ซึ่งเป็นแบบแฟลชมีความละเอียด 8 บิต และมีอัตราการสุ่มสูงสุดที่ 40 MHz โดยมีระดับการแปลง 1.55ถึง 3.26โวลต์ (รายละเอียดและคุณสมบัติต่าง ๆ ต่างสามารถดูจากภาคผนวก ข.) ดังรูปที่5.6

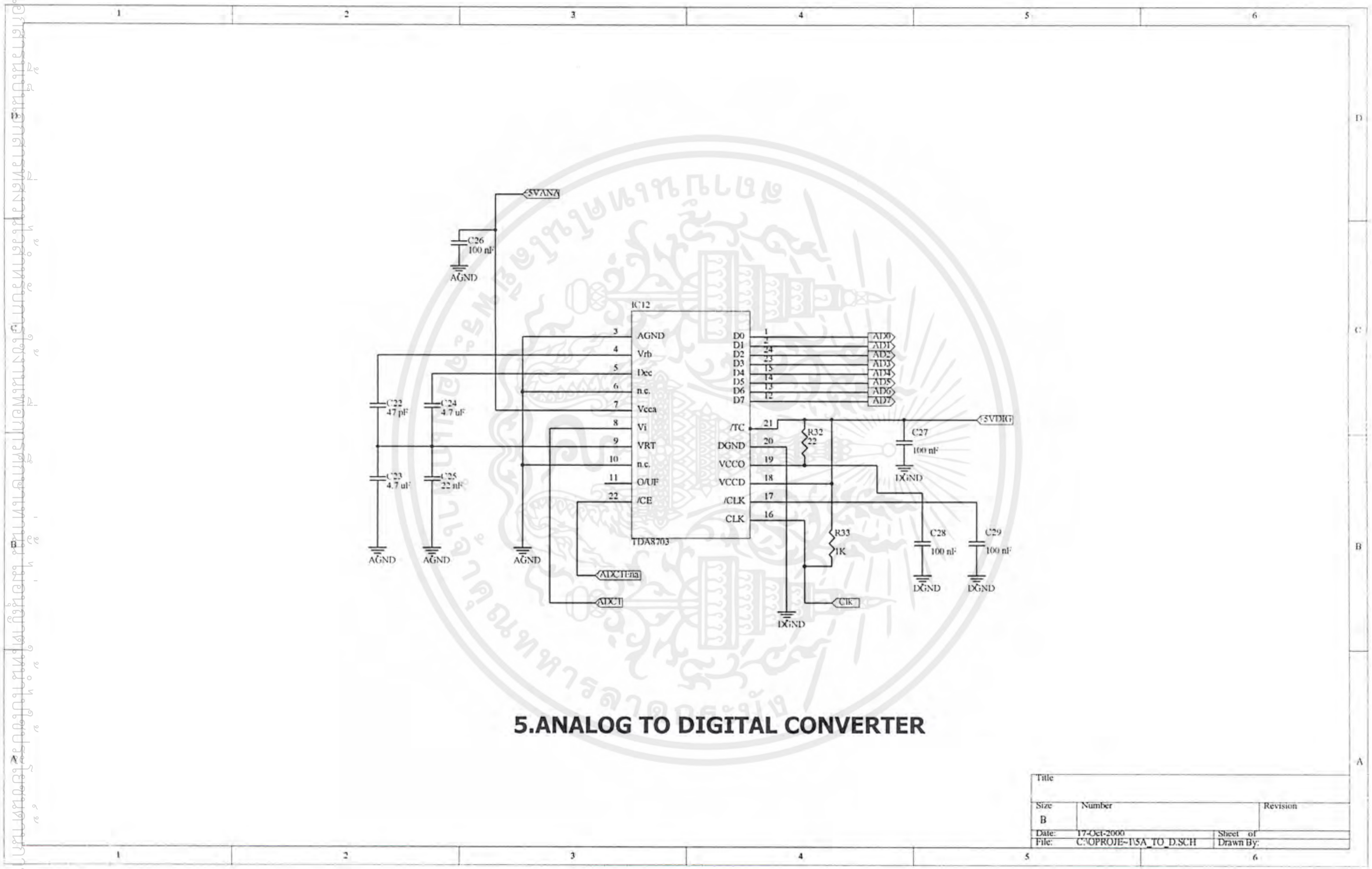
5.4 ส่วนกำเนิดสัญญาณนาฬิกาสำหรับการสุ่ม (Clock Generator)

วงจรนี้ทำหน้าที่ในการสร้างสัญญาณนาฬิกาสำหรับใช้ในการทำงานของวงจรทั้งหมดใน วงจรนี้จะประกอบด้วย ออสซิลเลเตอร์ความถี่ 40 MHz วงจรนับใช้ไอซีเบอร์ 74HC161 4 ตัว ใช้เป็นวงจรหารความถี่เพื่อให้เกิดความถี่ต่าง ๆ ซึ่งในวงจรนี้จะได้ความถี่ 16 ความถี่ด้วยกัน ดังตารางที่5.2และใช้ไอซีเบอร์ 74HC150 ในการเลือกความถี่ไปใช้งานดังรูปที่ 5.7

D	C	B	A	ความถี่ (Hz)
0	0	0	0	40 M
0	0	0	1	20 M
0	0	1	0	10 M
0	0	1	1	5 M
0	1	0	0	2.5 M
0	1	0	1	1.25 M
0	1	1	0	625 k
0	1	1	1	312.5 k
1	0	0	0	156.25 k
1	0	0	1	78.125 k
1	0	1	0	39.062 k
1	0	1	1	19.53 k
1	1	0	0	9.76 k
1	1	0	1	4.88 k
1	1	1	0	2.44 k
1	1	1	1	1.22 k

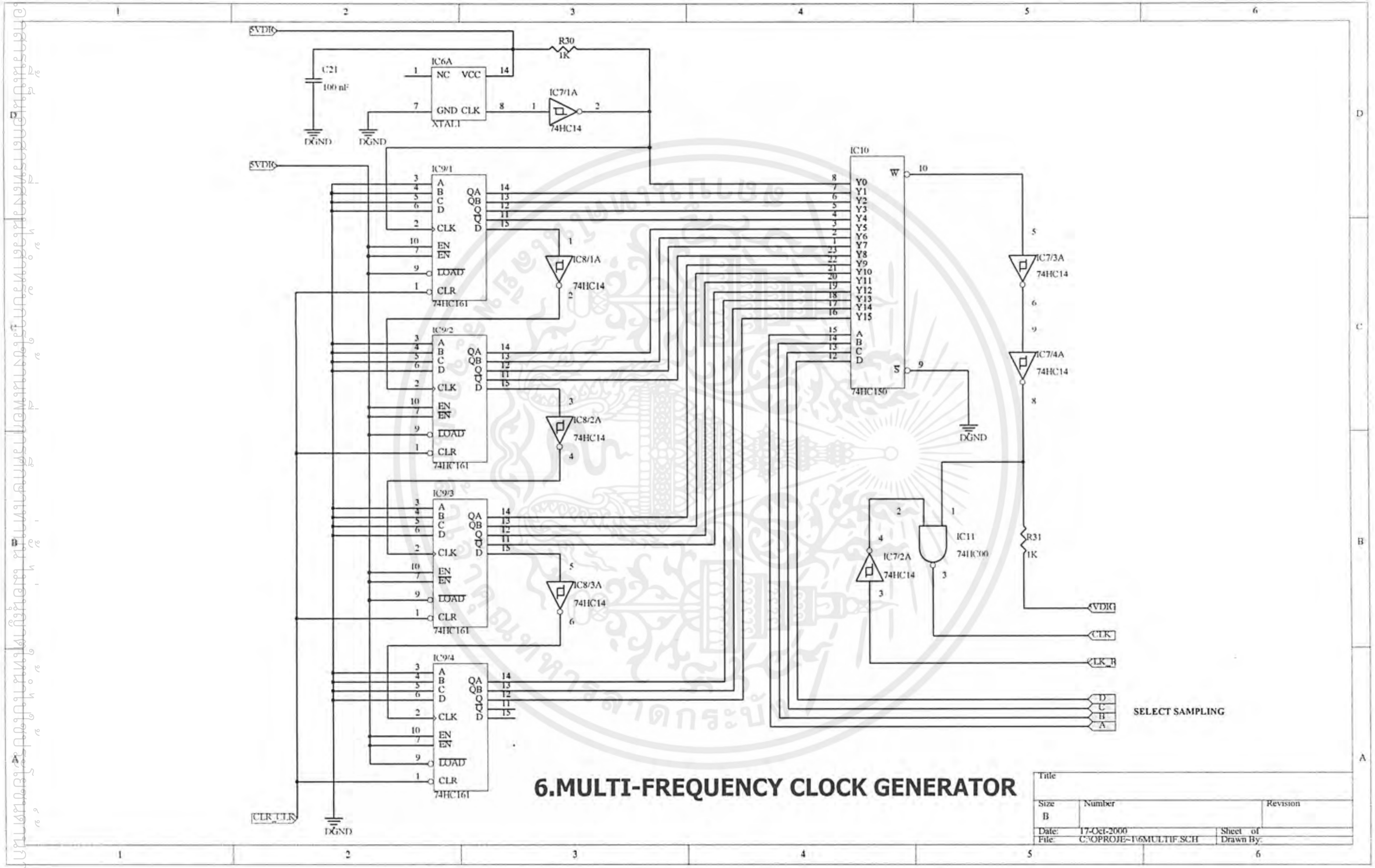
ตารางที่5.2 ความถี่ของวงจรกำเนิดสัญญาณ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 5.6 แสดงวงจรแปลงสัญญาณอนาลอกเป็นสัญญาณดิจิตอล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 5.7 แสดงวงจรกำเนิดสัญญาณนาฬิกาหลายความถี่

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5.5 วงจรถอดรหัสตำแหน่งพอร์ต และวงจรเชื่อมต่อพอร์ต

เป็นวงจรเปรียบเทียบตำแหน่งพอร์ตที่สล็อต ISA กับตำแหน่งพอร์ตที่กำหนดไว้ ดังรูปที่ 5.8 และ 5.9

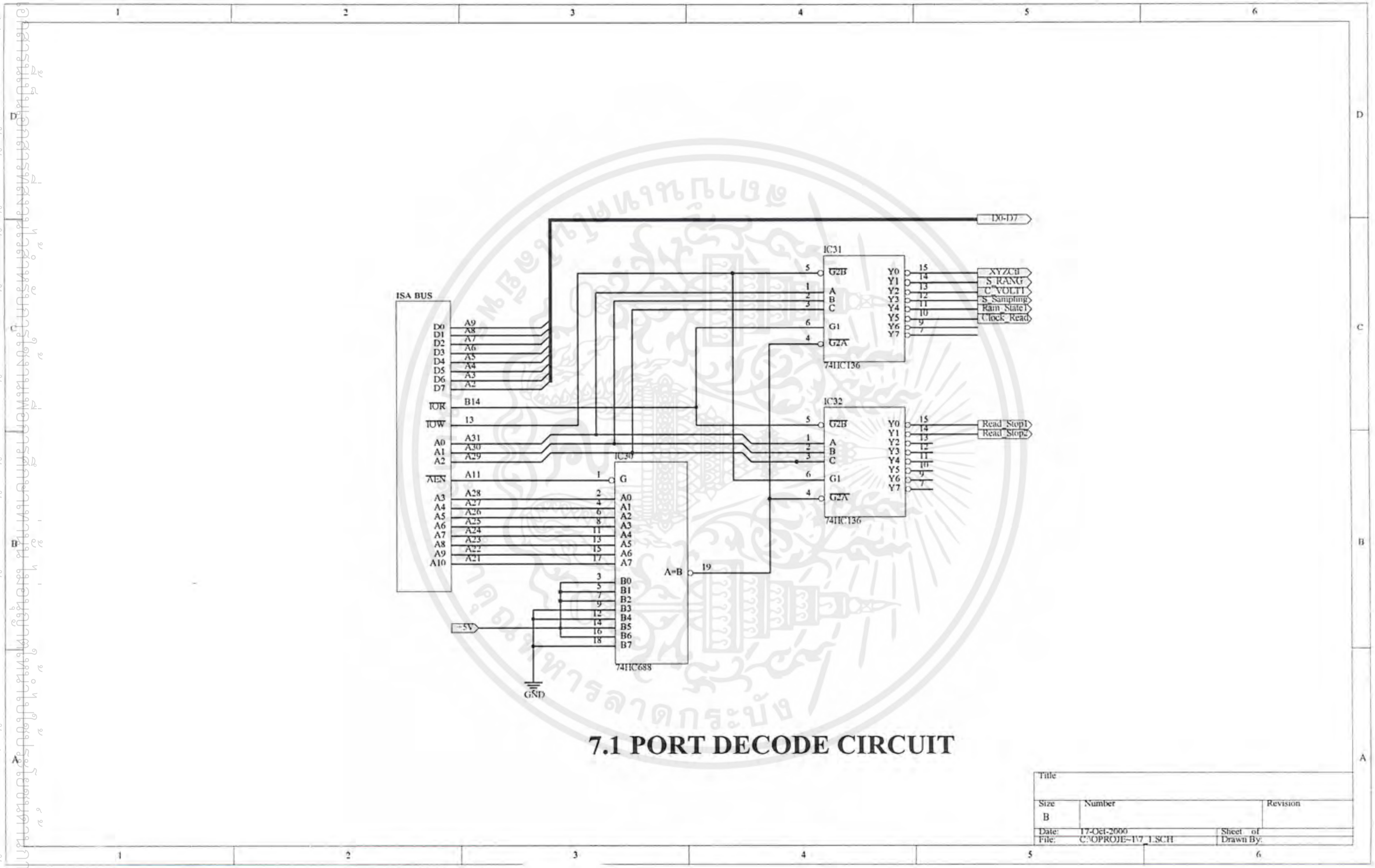
5.6 วงจรนับตำแหน่งหน่วยความจำ

เป็นวงจรนับตำแหน่งหน่วยความ โดยที่เราสามารถเลือกได้ว่าจะใช้หน่วยความจำเท่าไหร่ในการเก็บข้อมูล ในการนับจะใช้ไอซีเบอร์ 74HC161 เป็นตัวนับ ซึ่งจะนับได้สูงสุด 128 kbyte ดังรูปที่ 5.10

5.7 วงจรหน่วยความจำ

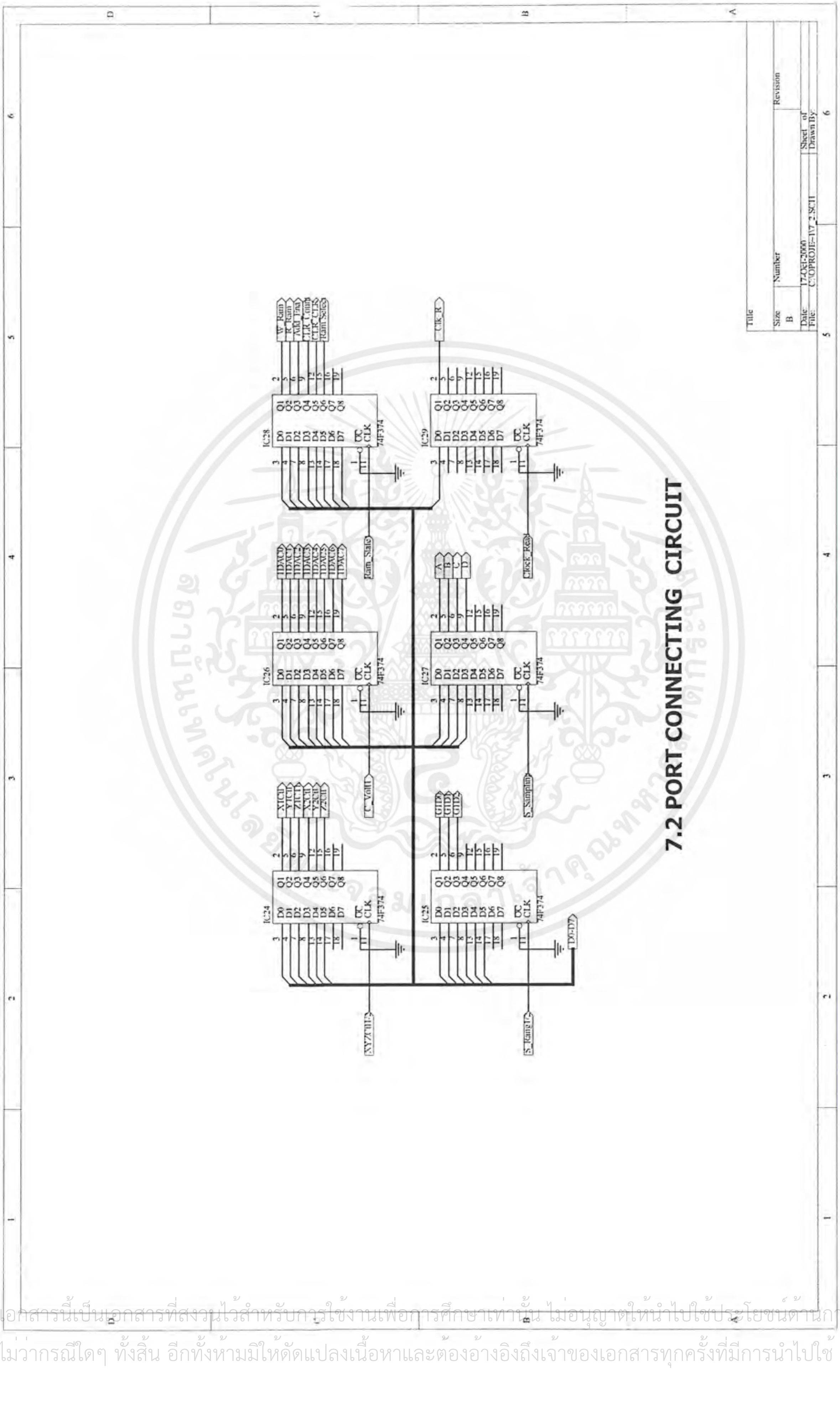
วงจรหน่วยความจำจะใช้ในการเก็บข้อมูลที่ได้จากการแปลงสัญญาณอนาลอกเป็นดิจิตอลแล้วนำมาเก็บไว้ในหน่วยความจำ ซึ่งหน่วยความจำในวงจรนี้ใช้ไอซีเบอร์ 681000C ซึ่งเป็น SDRAM ขนาด 128 kbyte แต่จะใช้เพียงแค่ 64 kbyte เท่านั้น ดังรูปที่ 5.11

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



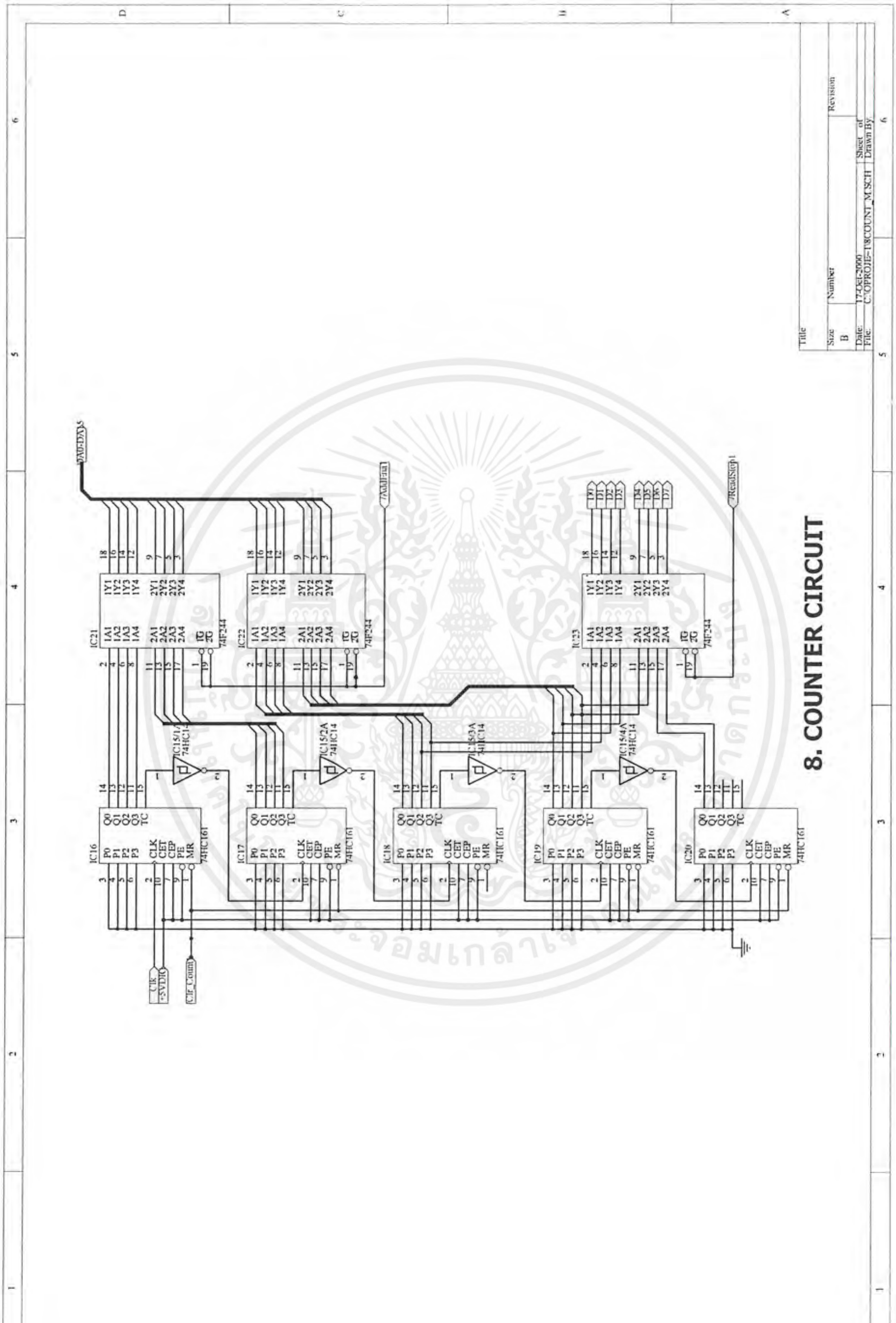
รูปที่ 5.8 แสดงวงจรถอดรหัสตำแหน่งพอร์ต

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



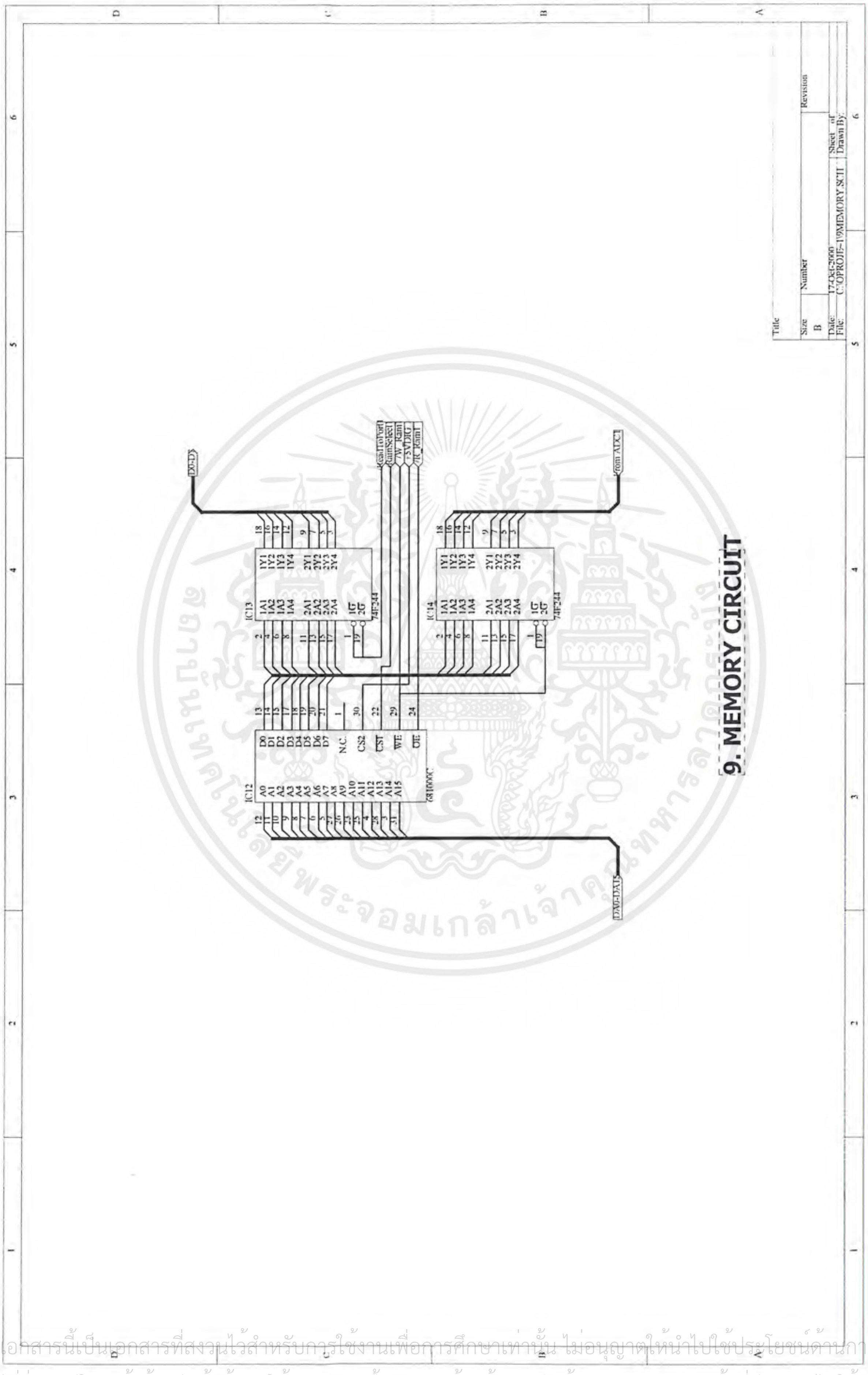
รูปที่ 5.9 แสดงวงจรเชื่อมต่อพอร์ต ISA กับวงจรต่าง ๆ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 5.10 แสดงวงจรนับตำแหน่งหน่วยความจำ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 5.11 แสดงวงจรหน่วยความจำ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

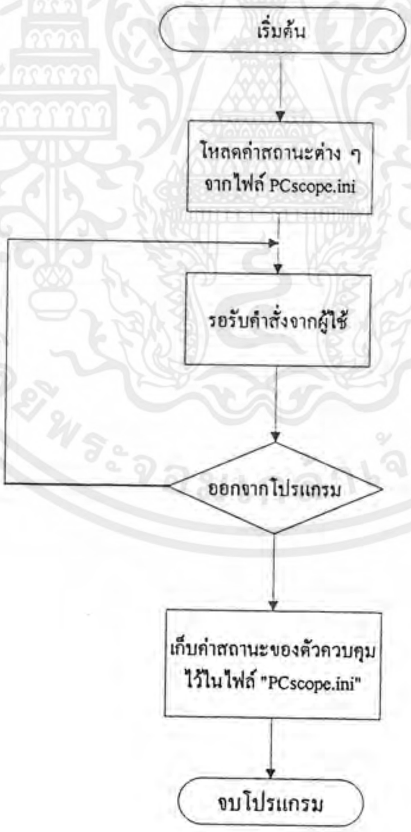
5.8 โปรแกรมควบคุมการทำงาน

คอมไพเลอร์ที่ใช้ในการเขียนโปรแกรมควบคุมการทำงานนั้น ใช้ Deiphi 5.0 ซึ่งมีพื้นฐานเดียวกับภาษา Pascal เป็นโปรแกรมที่มีความยืดหยุ่นทางโครงสร้าง และใช้เวลาในการคอมไพน้อยกว่าโปรแกรมอื่น ๆ

โปรแกรมควบคุมการทำงานของการ์ด เราสามารถควบคุมการทำงานได้จากการเลือกคอนโทรลที่นหน้าจอหลัก หรือจะเลือกการควบคุมจากเมนูก็ได้ คอนโทรลต่าง ๆ จะมีหน้าที่ ๆ แตกต่างกัน ซึ่งแต่ละคอนโทรลจะมีชื่อบอกไว้ เมื่อเลื่อนเมาท์เข้าไปในคอนโทรล จะมีชื่อแสดงให้เห็นขึ้นมา และแถบข้างล่างสุดจะมีคำบอกหน้าที่ของคอนโทรลอันนั้น

5.9 หลักการเขียนโปรแกรมควบคุมการทำงาน

หลักการเขียนโปรแกรมควบคุมการทำงานของการ์ดสามารถสรุปเป็นแผนงานได้ดังนี้



รูปที่ 5.13 แสดงผังงานของโปรแกรมควบคุม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เมื่อเริ่มต้นโหลดโปรแกรม Computer Oscilloscope and Spectrum Analysis โปรแกรมจะทำการโหลดค่าซึ่งเป็นค่าสถานะเริ่มต้นของสโคป เช่น Vlolt/Div, Time/Div, Xpos, Ypos ฯลฯ ออกมาเก็บไว้ และแสดงผลออกมาที่ตัวควบคุมต่าง ๆ จากนั้นจะรอรับคำสั่งจากผู้ใช้งานและปฏิบัติตามคำสั่งตาม ๆ ซึ่งจะอธิบายในหัวข้อตัวไป และเมื่อผู้ใช้ทำการปิดโปรแกรม ตัวโปรแกรมจะทำการเก็บค่าสถานะต่าง ๆ ของสโคปเก็บไว้ในไฟล์ PCScope.ini เพื่อจะนำมาใช้เป็นค่าสถานะเริ่มต้น เมื่อทำการเปิดโปรกรมมาใช้งานในครั้งต่อไป

5.10 การเขียนโปรแกรม Interface กับ ISA-Bus

โครงงานนี้ใช้บัสอินเตอร์เฟสเป็นแบบ ISA-Bus การเขียนโปรแกรมเพื่อติดต่อกับ ISA-Bus โดยใช้ Delphi 5.0 สามารถทำได้หลายวิธี แต่ที่ใช้ในโครงงานนี้จะใช้วิธีการเขียนโปรแกรมแอสเซมบลี (Assembly) ใน Delphi เพื่อติดต่อกับระบบ ISA-Bus ซึ่งมีซอร์ดโค้ดและวิธีใช้งานดังนี้

โปรแกรมสำหรับส่งค่าข้อมูลออกพอร์ตซึ่งค่าข้อมูลเป็น 8 bit

```
procedure OutPort (Address, Data : Word) ; Assembly
begin
   asm
         push dx
         push ax
         mov ax,data
         mov dx,address
         out dx,ax
         pop ax
         pop dx
end ;
```

วิธีการใช้งาน เช่น ต้องการส่งข้อมูล 80H ซึ่งเป็นข้อมูล 8 bit ออกที่ พอร์ต 300H

```
Var
   AddressData : Word;
   DataSend : Byte;
Begin
   AddressData := $300;
   DataSend := $80;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
  OutPort (AddressData, DataSend);
End;
```

โปรแกรมสำหรับอ่านค่าข้อมูลเข้ามาจากพอร์ตซึ่งค่าข้อมูลเป็นแบบ 8 bit

```
function InPort(Address : Word):byte; Assembly
begin
  asm
    push dx
    mov dx,Address
    pop dx
  end;
  end;
```

วิธีใช้งาน เช่น ต้องการรับค่าข้อมูล 8 bit จากพอร์ต 300H มาเก็บไว้ที่ Data ซึ่ง Data เป็นตัวแปรชนิด byte

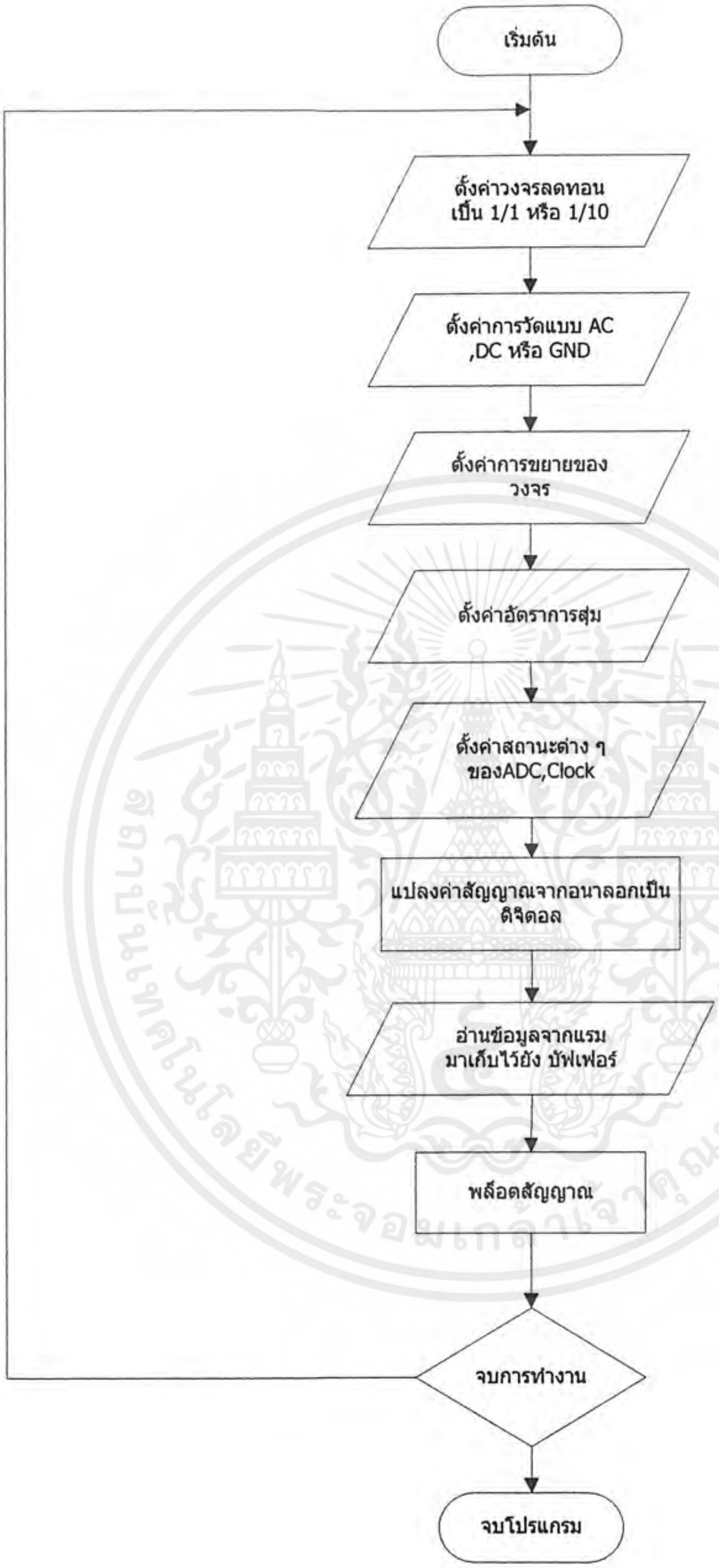
```
Var
   AddressData : Word;
   Data : Byte;
Begin
   AddressDAta : = $300;
   Data : = InPort(AddressData)
End;
```

5.11การควบคุมและการใช้งานการ์ด

ในส่วนของการควบคุมฮาร์แวร์, การเซ็ตค่าต่าง ๆ เช่น ค่าสถานะ, ค่าข้อมูลควบคุม ชนิดของการวัดตลอดจนการอ่านค่าข้อมูลเก็บไว้ในหน่วยความจำ การแสดงผลสัญญาณ และการวิเคราะห์ สเปกตรัมสามารถเขียนเป็นผังงานได้ดังรูปที่ 6.14

การแสดงผลสามารถแสดงผลได้ 2 แบบ คือ แสดงผลเป็นรูปสัญญาณที่วัดได้ และสามารถนำค่าที่วัดหรือวิเคราะห์ได้ แปลงออกมาเก็บไว้เป็น Fileภาพชนิดต่าง ๆ ตามที่เลือก ไปหรือพิมพ์สัญญาณที่วัดได้ออกยังเครื่องพิมพ์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 5.14 แสดงผังงานของการควบคุมฮาร์ดแวร์ เมื่อทำการวัดสัญญาณ

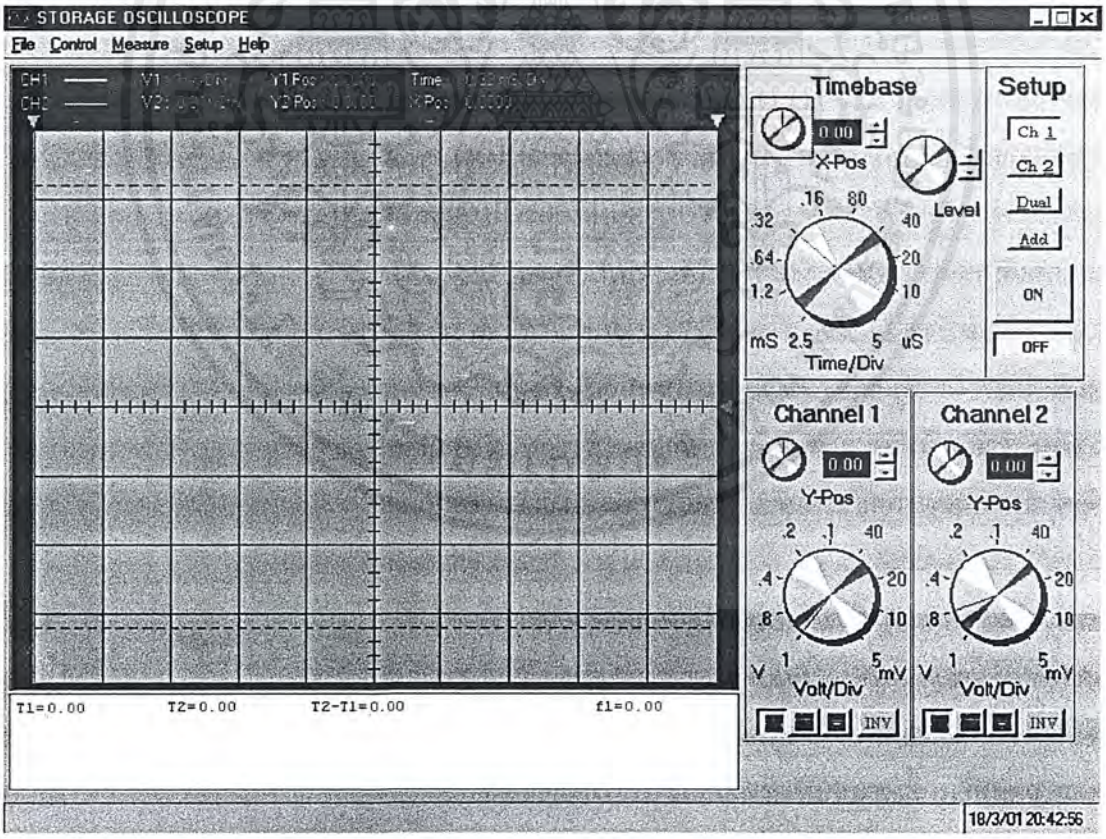
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 6
การทดลอง

6.1การทดลองวัดสัญญาณและการนำข้อมูลที่ได้มาวิเคราะห์ต่อ

เมื่อทำการออกแบบวงจรส่วนต่าง ๆ และเขียนโปรแกรมเป็นที่เรียบร้อยแล้ว ขั้นตอนต่อไปคือการทดสอบการทำงานของโปรแกรมการทำงานของเครื่องออสซิโลสโคปบนเครื่องคอมพิวเตอร์เครื่องนี้ โดยจะทำการทดลองวัดรูปเครื่องสัญญาณแบบต่าง ๆ ได้แก่ สัญญาณรูปซายน์ (Sine wave) สัญญาณรูปสามเหลี่ยม (Triangle wave) สัญญาณรูปสี่เหลี่ยม (Square wave) ที่ความถี่และขนาดต่าง ๆ กัน

โดยจะเริ่มโดยการทำการทำงานโปรแกรมที่ได้สร้างขึ้นมา ในที่นี้คือ M_Oscilloscope.exe โดยจะปรากฏดังนี้



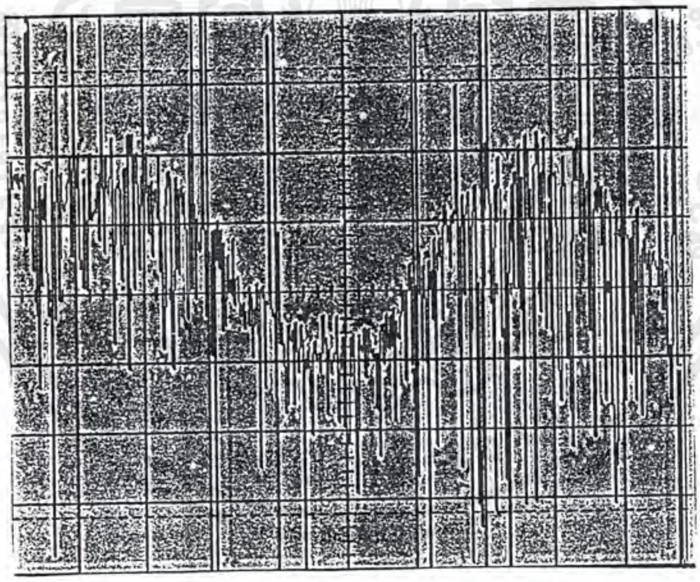
รูปที่ 6.1 หน้าจอควบคุม Storage Oscilloscope

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

6.1.1 ผลการทดลองวัดสัญญาณอินพุทที่เป็นรูปซายน์

ผลการทดลองรูปที่	ความถี่ (Hz)	ขนาดสัญญาณ (Vpp)
6.2	500	1
6.3	20K	1
6.4	100K	1
6.5	500K	1

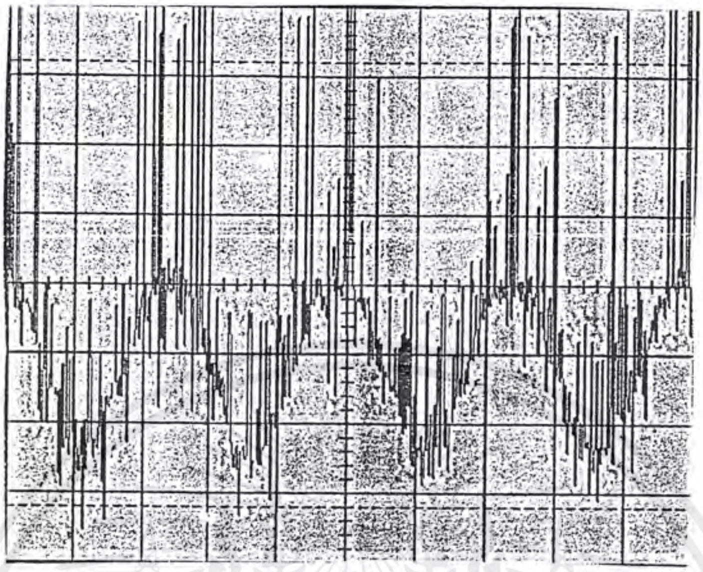
ตารางที่ 6.1 แสดงผลการทดลองสัญญาณอินพุทเป็นซายน์ที่ความถี่ต่าง ๆ



รูปที่6.2 สัญญาณรูปซายน์ที่ความถี่ 500 Hz ขนาด 0.5 Vp(1Vpp)

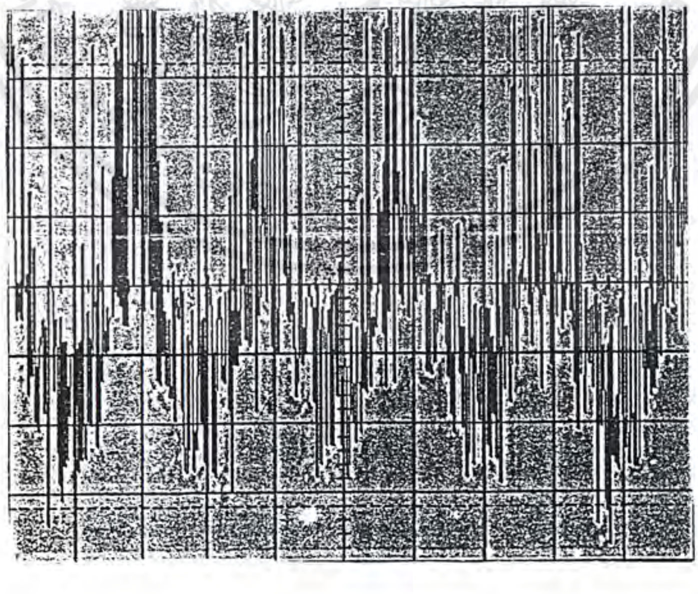
(0.32mS ,0.1V/DIV)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 6.3 สัญญาณรูปซายน์ที่ความถี่ 20 kHz ขนาด 0.5 Vp(1Vpp)

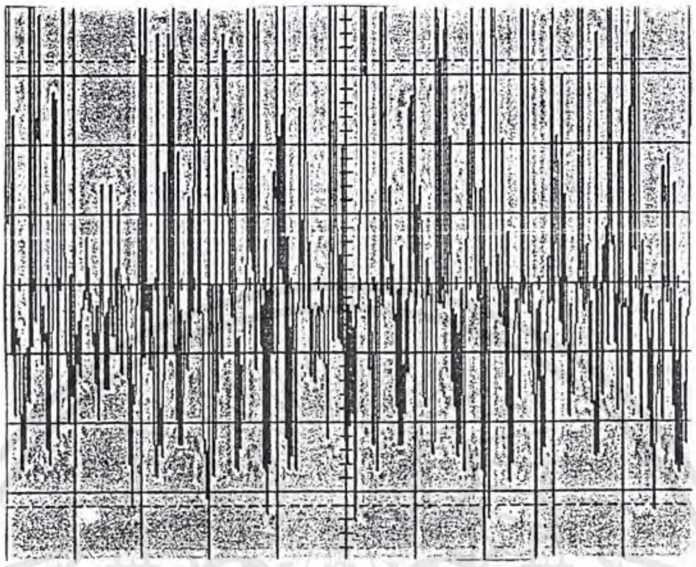
(20uS ,0.1V/DIV)



รูปที่ 6.4 สัญญาณรูปซายน์ที่ความถี่ 100 kHz ขนาด 0.5 Vp(1Vpp)

(5uS ,0.1V/DIV)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 6.5 สัญญาณรูปซายน์ที่ความถี่ 500 kHz ขนาด 0.5 Vp(1Vpp)

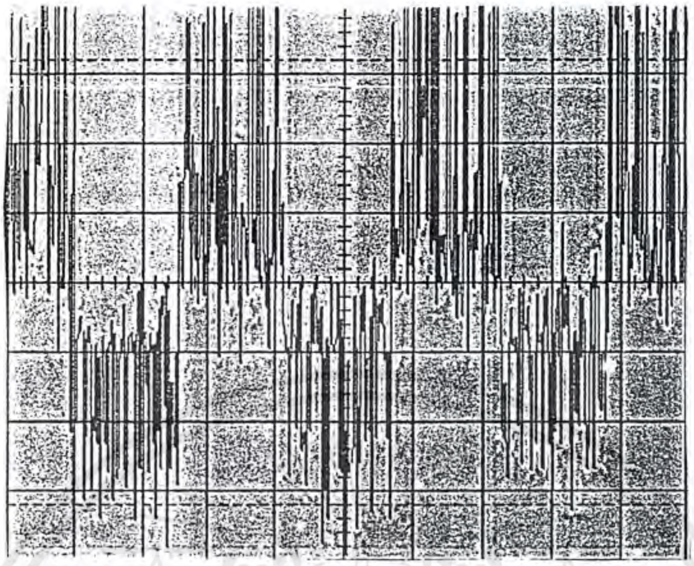
(5uS ,0.1V/DIV)

6.1.2 ผลการทดลองวัดสัญญาณอินพุทที่เป็นรูปสี่เหลี่ยม

ผลการทดลองรูปที่	ความถี่(HZ)	ขนาดสัญญาณ (Vpp)
6.6	4K	1
6.7	20K	1
6.8	100K	1

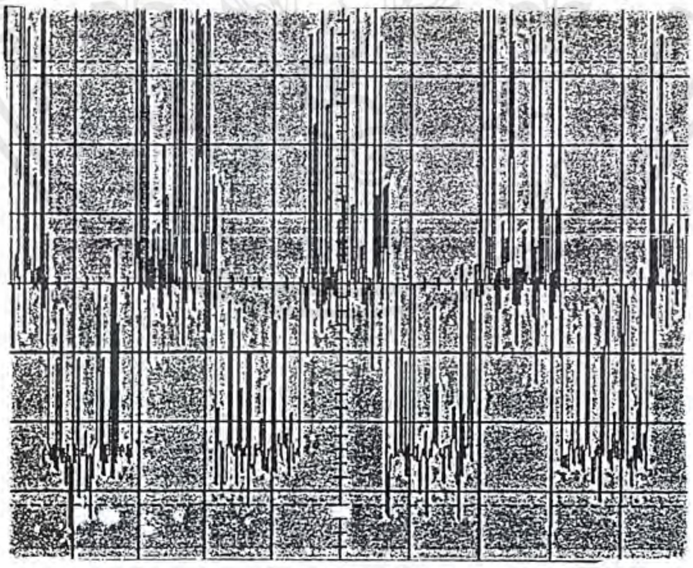
ตารางที่ 6.2 แสดงผลการทดลองสัญญาณอินพุทเป็นคลื่นสี่เหลี่ยมที่ความถี่ต่าง ๆ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 6.6 สัญญาณรูปสี่เหลี่ยมที่ความถี่ 4 kHz ขนาด 0.5 Vp(1Vpp)

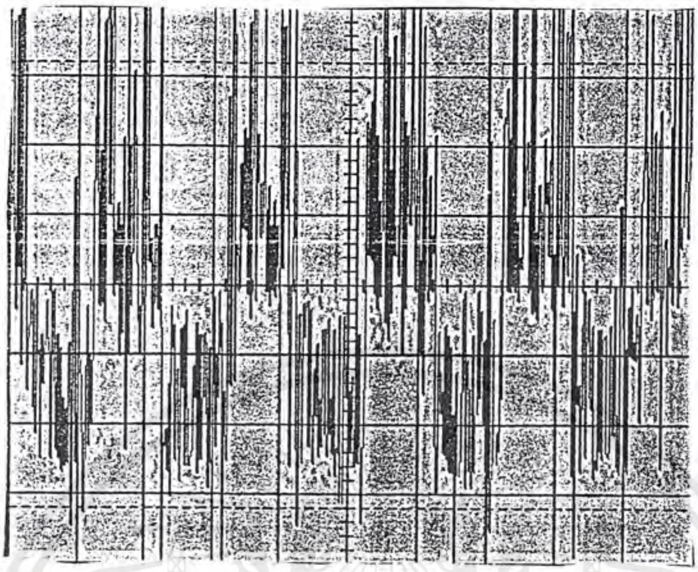
(80 uS ,0.1V/DIV)



รูปที่ 6.7 สัญญาณรูปสี่เหลี่ยมที่ความถี่ 20 kHz ขนาด 0.5 Vp(1Vpp)

(20uS ,0.1V/DIV)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 6.8 สัญญาณรูปสี่เหลี่ยมที่ความถี่ 100 kHz ขนาด 0.5 Vp(1Vpp)

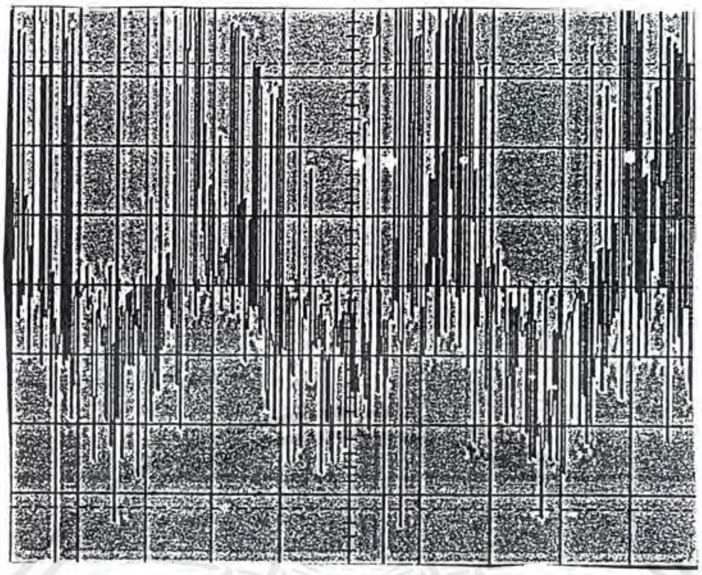
(5uS ,0.1V/DIV)

6.1.3 ผลการทดลองวัดสัญญาณอินพุทที่เป็นรูปสามเหลี่ยม

ผลการทดลองรูปที่	ความถี่ (Hz)	ขนาดสัญญาณ (Vp-p)
6.9	4K	1
6.10	20K	1
6.11	100K	1

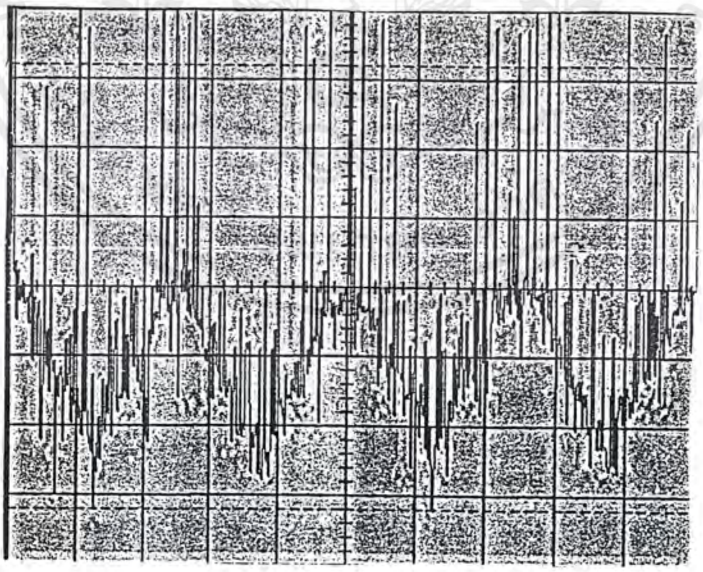
ตารางที่ 6.3 แสดงผลการทดลองสัญญาณอินพุทเป็นคลื่นสามเหลี่ยมที่ความถี่ต่าง ๆ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 6.9 สัญญาณรูปสามเหลี่ยมที่ความถี่ 4 kHz ขนาด 0.5 Vp(1Vpp)

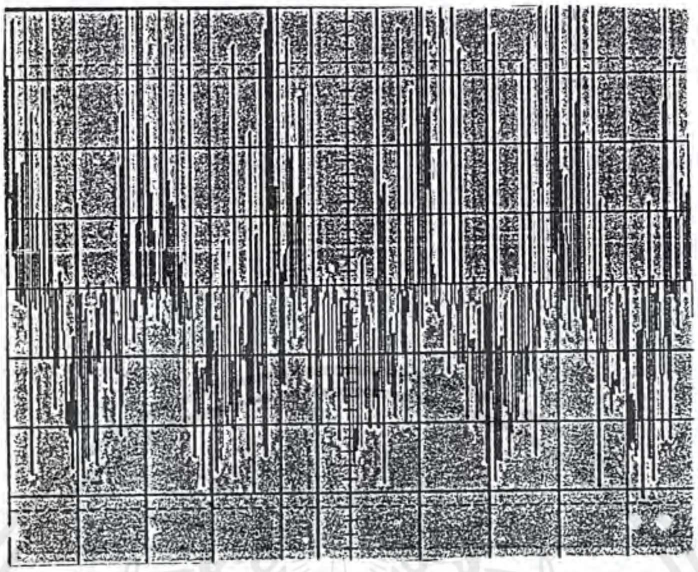
(80uS ,0.1V/DIV)



รูปที่ 6.10 สัญญาณรูปสามเหลี่ยมที่ความถี่ 20 kHz ขนาด 0.5 Vp(1Vpp)

(20uS ,0.1V/DIV)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 6.11 สัญญาณรูปสามเหลี่ยมที่ความถี่ 100 kHz ขนาด 0.5 Vp(1Vpp)

(5uS ,0.1V/DIV)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 7
สรุปผลการทำโครงงาน

จากผลการทดลองวงจรในส่วนต่าง ๆ ในบทที่ 6 ทั้งวงจรในส่วนของภาคอนาลอก และวงจรในส่วนของภาคดิจิตอล โดยจากการทดสอบแยกเป็นวงจร ผลที่ออกมาแสดงให้เห็นว่าแต่ละวงจรสามารถทำงานได้เป็นอย่างดี แต่เมื่อนำวงจรมารวมกันและทดสอบโดยการทำงานจริง ผลที่ออกมาไม่เป็นไปตามที่คาดเอาไว้เลย โดยจะเห็นได้ว่ากราฟของสัญญาณที่ใส่เข้าไปต่างกับผลลัพท์ที่ออกมาทางหน้าจอคอมพิวเตอร์อย่างมาก ซึ่งปัญหาเกิดจาก สัญญาณรบกวนของวงจรนั่นเอง

การปรับระดับทริกเกอร์ ในโครงงานนี้จะไม่ใช้ ฮาร์ด์แวร์ทริกเกอร์ (hardware trigger)แต่จะใช้เป็นซอฟท์แวร์ทริกเกอร์ (software trigger) แทน ข้อดีของ ซอฟท์แวร์ทริกเกอร์คือสัญญาณที่จะแสดงผลจะนิ่งกว่าแบบฮาร์ด์แวร์ทริกเกอร์ แต่ก็มีข้อเสียคือการแสดงผลจะช้าลงเพราะต้องแบ่งเวลาส่วนหนึ่งไปประมวลผลในส่วนของการรักษาระดับทริกเกอร์ไว้ ข้อมูลที่ได้มาจากการแปลงอนาลอกเป็นดิจิตอล จะถูกเก็บในหน่วยความจำ และตัวโปรแกรมจะอ่านข้อมูลขึ้นมาแสดงผล ในส่วนของการแปลงข้อมูลของวงจรอนาลอกเป็นดิจิตอล แปลงข้อมูลได้อย่างแม่นยำ ข้อมูลที่ผิดพลาดส่วนใหญ่จะเกิดจากสัญญาณรบกวนที่เกิดขึ้นมากกว่า

7.1 ปัญหาที่พบในการทำโครงงาน

1. สัญญาณที่ปรากฏบนคอมพิวเตอร์มีความผิดเพี้ยนไปจากสัญญาณเอาท์พุทเป็นอย่างมาก อันเป็นผลเนื่องจากสัญญาณรบกวน (noise) ที่เกิดขึ้นจากการที่ ในการต่อวงจรส่วนดิจิตอลนั้นได้ทำการใช้แผ่นปรินท์เอนกประสงค์ และซอกเกตขายาวในการพันสายแทนการทำการบัดกรี ซึ่งเป็นเหตุผลหนึ่งในการที่ทำให้เกิดสัญญาณรบกวน เนื่องมาจากการที่ใช้การพันนั้นจะมีโอกาสที่ทำให้เกิดออกไซด์ง่าย คล้ายกับว่าเกิดตัวเก็บประจุทุกจุดที่เกิดออกไซด์ขึ้น ซึ่งจะเป็นผลเสียต่อวงจรทำให้เกิดสัญญาณรบกวนขึ้นได้

และอีกเหตุผลหนึ่งที่ทำให้เกิดสัญญาณรบกวนขึ้นได้อันเนื่องมาจาก การจ่ายไฟเลี้ยงให้กับวงจรนั้นได้นำเอาไฟเลี้ยงของคอมพิวเตอร์มาใช้อาจเป็นเหตุผลที่ทำให้เกิดสัญญาณรบกวนขึ้นเพราะคอมพิวเตอร์นั้นมีตัว oscillator ที่มีความถี่สูงมากอาจทำให้เกิดสัญญาณรบกวนได้ง่าย

2. มีปัญหาในการวัดสัญญาณที่มีขนาดเล็กหรือมีความละเอียดมาก ดังจะเห็นในการทดลองในบทที่ 6 สาเหตุที่ยิ่งสัญญาณยิ่งเล็กยิ่งไม่ชัดเจนนี้เนื่องมาจาก

- ความละเอียดของ A/D ที่ใช้มีขนาด 8 บิต ดังนั้นจึงมีความละเอียดสูงสุดได้ 256 ค่า และเราออกแบบให้สามารถวัดขนาดใหญ่สุดได้ 20 V_{pp} ทำให้ A/D เปลี่ยนค่าไป 1 ค่า เมื่อแรงดันเปลี่ยนไป 0.078 V

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- สัญญาณรบกวนจากภายนอก หรือเกิดจากไอซี A/D เอง

3. เนื่องจากจากการทดลองเห็นได้ว่าการใช้วงจรที่ใช้การพันสายนั้นมีปัญหาเกิดขึ้นอย่างมาก จึงได้มีการออกแบบวงจรโดยใช้ protel และได้ทำการกัดปรินท์ แต่ก็มีปัญหาเกิดขึ้นเนื่องจากมีประสบการณ์ในการออกแบบวงจรน้อย และ วงจรที่ทำนี้มีรายละเอียดเยอะมากจึงใช้การ ออกแบบสองหน้า (two layer) โดยไม่ทำเป็น page to hold และการใช้การ autoroute แทนการลากสายด้วยตัวเองเพื่อความสะดวก จะทำให้มีปัญหามากขึ้น เพราะการ ไม่ทำเป็นแบบ page to hold และการทำ autoroute บางทีจะทำให้การลงอุปกรณ์นั้นเป็นไปได้ยากมากแทบจะไม่มีทางลงได้เลย

7.2 แนวทางการแก้ไขปัญหา

1. ใช้การบัดกรีแทนการพันสาย เพื่อที่จะลดการเกิดออกไซด์ พยายามวางอุปกรณ์ให้ใกล้กันมากที่สุด เพื่อที่จะลดสัญญาณรบกวนที่เกิดขึ้น และที่สำคัญควรจะทำ ภาคแหล่งจ่ายไฟให้แก่ไอซีแทนที่จะใช้แหล่งจ่ายไฟจากคอมพิวเตอร์ในโครงงานนี้

2. ถ้าจะใช้การกัดปรินท์กับวงจรในโครงงานนี้ควรจะกัดแบบ page to hold เพื่อความสะดวกในการลงอุปกรณ์ และถ้าเป็นไปได้ควรจะทำการลากสายด้วยตัวเอง แทนการใช้ autoroute

3. ควรแยกไฟเลี้ยงที่จะจ่ายให้กับวงจรอนาลอก และวงจรดิจิตอลแยกออกจากกัน

4. ในเครื่องคอมพิวเตอร์ที่มีการออสซิเลทของอุปกรณ์บางตัว ซึ่งจะมารบกวนสัญญาณอนาลอกภาคอินพุทของเราได้ดังนั้นที่จุดต่อสัญญาณอินพุทที่วงจรควรจะใช้สัญญาณที่มีการหุ้ม (Shield) เพื่อที่จะลดสัญญาณรบกวนให้น้อยที่สุด

7.3 ประโยชน์ที่ได้รับจากการทำโครงงานนี้

1. ได้เครื่องดิจิตอลออสซิโลสโคป ที่มีราคาถูกโดยใต้นทุนน้อยกว่าที่ซื้อในตลาดหลายเท่าตัวโดยเครื่องที่เราสร้างขึ้นมานั้นก็มีคุณสมบัติใกล้เคียวกัน
2. สามารถนำเอาข้อมูลที่ได้จากการวัดไปทำการประมวลผลต่อได้ เช่นการทำ inverse หรือ การนำไปวิเคราะห์ สเปกตรัม
3. ฮาร์ดแวร์ที่สร้างขึ้นมาสามารถใช้ร่วมกับคอมพิวเตอร์ ที่เรามีใช้กันอยู่อย่างแพร่หลายทำให้สะดวกในการใช้งานเป็นอย่างยิ่ง อีกทั้งยังทำให้เครื่องคอมพิวเตอร์ใช้งานได้อย่างกว้างขวางขึ้นอีกด้วย
4. โปรแกรมที่สร้างขึ้นมานั้นทำการพัฒนาได้ง่าย สามารถนำไปพัฒนาได้อีกทำให้เราได้รัยความชำนาญในการเขียนโปรแกรมมากขึ้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก ก.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โปรแกรมที่ใช้ในส่วนควบคุมและแสดงผล

```
unit send2;


interface
uses
  Windows, Messages, SysUtils, Classes, Graphics, Controls, Forms, Dialogs,
  ComCtrls, ExtCtrls, Menus, StdCtrls, Buttons, ImgList, zPanel,Math,
  ExtDlgs,Inifiles,Mystd, zLed, zonzoff, Fourier,printers;


type
  TScope = class(TForm)
    MainMenu1: TMainMenu;
    File1: TMenuItem;
    Help1: TMenuItem;
    Panel1: TPanel;
    StatusBar1: TStatusBar;
    Bevel1: TBevel;
    Control1: TMenuItem;
    Setup1: TMenuItem;
    Measure1: TMenuItem;
    Label2: TLabel;
    Label1: TLabel;
    Label3: TLabel;
    Label4: TLabel;
    Label5: TLabel;
    Label6: TLabel;
    Label7: TLabel;
    Label8: TLabel;
    Panel2: TPanel;
    Label11: TLabel;
    Label14: TLabel;
    Label22: TLabel;
    Label25: TLabel;
    Label27: TLabel;
    ST1: TLabel;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
ST2: TLabel;
DT: TLabel;
V1Max: TLabel;
F1: TLabel;
Open1: TMenuItem;
N1: TMenuItem;
Save1: TMenuItem;
Saveas1: TMenuItem;
N2: TMenuItem;
PrintSetup1: TMenuItem;
Print1: TMenuItem;
N3: TMenuItem;
Export1: TMenuItem;
Import1: TMenuItem;
N4: TMenuItem;
Exit1: TMenuItem;
Shape1: TShape;
Shape2: TShape;
Image1: TImage;
Mark2: TPanel;
Mark1: TPanel;
Image2: TImage;
DateTime: TTimer;
P90: TLabel;
P10: TLabel;
Report1: TMenuItem;
N5: TMenuItem;
Time1: TMenuItem;
N6: TMenuItem;
Channel11: TMenuItem;
Channel21: TMenuItem;
N7: TMenuItem;
Trigger1: TMenuItem;
Channel1Only: TMenuItem;
Channel2Only1: TMenuItem;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
All1: TMenuItem;
N8: TMenuItem;
SetGrid1: TMenuItem;
SetScreen1: TMenuItem;
N9: TMenuItem;
SetFont1: TMenuItem;
ThaiFont1: TMenuItem;
EnglishFont1: TMenuItem;
Cont1: TMenuItem;
Index1: TMenuItem;
N10: TMenuItem;
About1: TMenuItem;
TimeBase: TPanel;
XPosT: TzKnob;
Ch1Panel: TPanel;
Y1Pos: TzKnob;
Ch1Volt: TzKnob;
Inv1: TSpeedButton;
Inv11: TSpeedButton;
AC1: TSpeedButton;
DC1: TSpeedButton;
GND1: TSpeedButton;
UpDown1: TUpDown;
Image3: TImage;
DXPos: TEdit;
UpDown3: TUpDown;
DPos1: TEdit;
Panel3: TPanel;
XY1: TSpeedButton;
Real: TSpeedButton;
Store: TSpeedButton;
XY11: TSpeedButton;
Ch1: TSpeedButton;
ChkRam: TTimer;
Label20: TLabel;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
Plot: TImage;
LV1: TLabel;
LV2: TLabel;
LY1: TLabel;
LY2: TLabel;
LT: TLabel;
LX: TLabel;
LevelV: TzKnob;
UpDown4: TUpDown;
AnalysisSpectrum1: TMenuItem;
SavePictureDialog1: TSavePictureDialog;
Discreat1: TMenuItem;
Continues1: TMenuItem;
OpenPictureDialog1: TOpenPictureDialog;
z3DLabel1: Tz3DLabel;
z3DLabel2: Tz3DLabel;
z3DLabel3: Tz3DLabel;
z3DLabel4: Tz3DLabel;
z3DLabel5: Tz3DLabel;
z3DLabel6: Tz3DLabel;
z3DLabel7: Tz3DLabel;
z3DLabel8: Tz3DLabel;
z3DLabel9: Tz3DLabel;
z3DLabel10: Tz3DLabel;
z3DLabel11: Tz3DLabel;
z3DLabel12: Tz3DLabel;
z3DLabel13: Tz3DLabel;
z3DLabel14: Tz3DLabel;
z3DLabel29: Tz3DLabel;
z3DLabel30: Tz3DLabel;
z3DLabel31: Tz3DLabel;
z3DLabel32: Tz3DLabel;
ColorDialog1: TColorDialog;
Channel12: TMenuItem;
Channel22: TMenuItem;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
SetGrid2: TMenuItem;
N1Volt1: TMenuItem;
N8Volt1: TMenuItem;
N04Volt1: TMenuItem;
N02Volt1: TMenuItem;
N01Volt1: TMenuItem;
N40mVolt1: TMenuItem;
N20mVolt1: TMenuItem;
N10mVolt1: TMenuItem;
N5mVolt1: TMenuItem;
N1Volt2: TMenuItem;
N08Volt1: TMenuItem;
N04Volt2: TMenuItem;
N02Volt2: TMenuItem;
N01Volt2: TMenuItem;
N40mVolt2: TMenuItem;
N20mVolt2: TMenuItem;
N10mVolt2: TMenuItem;
N5mVolt2: TMenuItem;
Time: TzKnob;
z3DLabel33: Tz3DLabel;
z3DLabel34: Tz3DLabel;
z3DLabel35: Tz3DLabel;
z3DLabel36: Tz3DLabel;
z3DLabel37: Tz3DLabel;
z3DLabel38: Tz3DLabel;
z3DLabel39: Tz3DLabel;
z3DLabel40: Tz3DLabel;
z3DLabel41: Tz3DLabel;
z3DLabel42: Tz3DLabel;
z3DLabel43: Tz3DLabel;
z3DLabel44: Tz3DLabel;
On1: TMenuItem;
SpectrumAnalysis: TPanel;
z3DLabel45: Tz3DLabel;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
SpectrumType: TRadioGroup;
Windows: TComboBox;
PrinterSetupDialog1: TPrinterSetupDialog;
PrintDialog1: TPrintDialog;
PHz: TLabel;
Center: TLabel;
S1: TLabel;
S2: TLabel;
S3: TLabel;
S4: TLabel;
ImageFile: TMenuItem;
WaveFile: TMenuItem;
OpenDialog1: TOpenDialog;

procedure FormCreate(Sender: TObject);
procedure Image2MouseMove(Sender: TObject; Shift: TShiftState; X,
  Y: Integer);
procedure Image2MouseDown(Sender: TObject; Button: TMouseButton;
  Shift: TShiftState; X, Y: Integer);
procedure Image2MouseUp(Sender: TObject; Button: TMouseButton;
  Shift: TShiftState; X, Y: Integer);
procedure DateTimeTimer(Sender: TObject);
procedure ThaiFont1Click(Sender: TObject);
procedure XY1Click(Sender: TObject);
procedure Inv1Click(Sender: TObject);
//procedure Inv2Click(Sender: TObject);
procedure UpDown1Changing(Sender: TObject; var AllowChange: Boolean);
procedure Y1PosChange(Sender: TObject);
procedure UpDown2Changing(Sender: TObject; var AllowChange: Boolean);
procedure Y2PosChange(Sender: TObject);
procedure ChkRamTimer(Sender: TObject);
procedure AC1Click(Sender: TObject);
procedure DC1Click(Sender: TObject);
procedure RealClick(Sender: TObject);
procedure StoreClick(Sender: TObject);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
SpectrumType: TRadioGroup;
Windows: TComboBox;
PrinterSetupDialog1: TPrinterSetupDialog;
PrintDialog1: TPrintDialog;
PHz: TLabel;
Center: TLabel;
S1: TLabel;
S2: TLabel;
S3: TLabel;
S4: TLabel;
ImageFile: TMenuItem;
WaveFile: TMenuItem;
OpenDialog1: TOpenDialog;

procedure FormCreate(Sender: TObject);
procedure Image2MouseMove(Sender: TObject; Shift: TShiftState; X,
  Y: Integer);
procedure Image2MouseDown(Sender: TObject; Button: TMouseButton;
  Shift: TShiftState; X, Y: Integer);
procedure Image2MouseUp(Sender: TObject; Button: TMouseButton;
  Shift: TShiftState; X, Y: Integer);
procedure DateTimeTimer(Sender: TObject);
procedure ThaiFont1Click(Sender: TObject);
procedure XY1Click(Sender: TObject);
procedure Inv1Click(Sender: TObject);
//procedure Inv2Click(Sender: TObject);
procedure UpDown1Changing(Sender: TObject; var AllowChange: Boolean);
procedure Y1PosChange(Sender: TObject);
procedure UpDown2Changing(Sender: TObject; var AllowChange: Boolean);
procedure Y2PosChange(Sender: TObject);
procedure ChkRamTimer(Sender: TObject);
procedure AC1Click(Sender: TObject);
procedure DC1Click(Sender: TObject);
procedure RealClick(Sender: TObject);
procedure StoreClick(Sender: TObject);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
procedure XPosTChange(Sender: TObject);
procedure UpDown3Changing(Sender: TObject; var AllowChange: Boolean);
procedure LevelVChange(Sender: TObject);
procedure UpDown4Changing(Sender: TObject; var AllowChange: Boolean);
procedure EnglishFont1Click(Sender: TObject);
procedure Export1Click(Sender: TObject);
procedure AnalysisSpectrum1Click(Sender: TObject);
procedure Cont1Click(Sender: TObject);
procedure Index1Click(Sender: TObject);
procedure Discreat1Click(Sender: TObject);
procedure Continues1Click(Sender: TObject);
procedure Edit2Change(Sender: TObject);
procedure Edit3Change(Sender: TObject);
procedure Exit1Click(Sender: TObject);
procedure Ch1VoltChange(Sender: TObject);
//procedure Ch2VoltChange(Sender: TObject);
procedure FormClose(Sender: TObject; var Action: TCloseAction);
procedure Channel12Click(Sender: TObject);
procedure Channel22Click(Sender: TObject);
procedure SetGrid2Click(Sender: TObject);
procedure TimeChange(Sender: TObject);
procedure On1Click(Sender: TObject);
//procedure WindowsChange(Sender: TObject);
//procedure SpectrumTypeClick(Sender: TObject);
procedure PrintSetup1Click(Sender: TObject);
procedure AC1MouseMove(Sender: TObject; Shift: TShiftState; X,
  Y: Integer);
procedure DC1MouseMove(Sender: TObject; Shift: TShiftState; X,
  Y: Integer);
procedure GND1MouseMove(Sender: TObject; Shift: TShiftState; X,
  Y: Integer);
procedure AC2Click(Sender: TObject);
procedure DC2Click(Sender: TObject);
procedure GND2Click(Sender: TObject);
procedure ImageFileClick(Sender: TObject);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    procedure WaveFileClick(Sender: TObject);
    procedure WSpecClick(Sender: TObject);

  private
    { Private declarations }
  public
    { Public declarations }
    procedure PaintMonitor;
    procedure PaintMark(ox1,ox2:Integer);
    procedure ProcessMark(o1,o2:Integer);
    procedure PaintPosY(Pos1,Pos2:Integer);
    procedure DisplaySignal;
    //procedure DisplaySpectrum;
    procedure DisplayGnd;
    procedure LoadInitial;
    procedure Channel1Chang;
    //procedure Channel2Chang;
    procedure TimebaseChang;
    procedure CallTime1(Position:Integer);
    procedure CallTime2(Position:Integer);
    procedure DisplayForPrintSignal;
    //procedure DisplayForPrintSpectrum;
    procedure CalVolt(XMark,Mark:Integer);
    //procedure PlotWave(WaveSize:LongInt);
    //procedure WaveSpectrum;
  end;
Const
    ScreenLeft=16;
    ScreenTop=45;
    GainPort=$339;
    AttPort=$338;
    SamplingPort=$33b;
    ClockReadPort=$33d;
    //ClamPort=$306;
    RamStatePort=$33c;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
  ReadToPort=$339;
  ReadStopPort=$338;

  R1K=$2;  //0000 0010
  R2K=$4;  //0000 0100
  R4K=$8;  //0000 1000
  R8K=$10;  //0001 0000
  R16K=$20; //0010 0000
  R32K=$40; //0100 0000
  R64K=$82; //1000 0010
  R128K=$1;//0000 0001
  //Rate=1;     //Gain=5Vi
  PixelGndInvAc=66;
  PixelGndInvDc=110;
  PixelGndNorAc=329;
  PixelGndNorDc=290;

var
 Scope: TScope;
 Old_ox1,Old_ox2,Origin1,Origin2,MarkSelect,cl,Old_Y1,Old_Y2,
 x1,x2:Integer;
 DataWave:Array[0..200000] of byte;
 DisplayBuff:array[0..500] of byte;
 Data:Array [0..131072] of byte;
 DataBuff:Array [0..1024] of byte;
 SamPlingRate,RamState,AttCtl,GainCtl,LevelTrig:byte;

DivRam,MonitorColor,Color1,Color2,GndNor,GndInv,DPos,YPos1,YPos2,XPos,RamE,RBu
ff,NTemp:integer;
 GndC1,Run:Boolean;
 MonitorName,TimeScale:String;
 TB,t1,t2,DT12,Frq,Rate:Extended;
 XMark1,XMark2,Temp:Integer;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
 WSize:Longint;
implementation

{$R *.DFM}
procedure OutPort(Address,D:word);Assembler;
begin
   asm
   push dx
   push ax
   mov  ax,d
   mov  dx,address
   out  dx,ax
   pop  ax
   pop  dx
   end;
end;


function InPort(Address:word):byte;assembler;
begin
   asm
   push dx
   push ax
   mov  dx,address
   in   al,dx
   mov  result,al
   pop  ax
   pop  dx
   end;
end;

{procedure TScope.WaveSpectrum;
var n,nfft:integer;
   y0,y1:Double;
begin
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
FFT1.ClearImag;
FFT1.SpectrumSize:=1024;
Plot.Picture.LoadFromFile('E:\DATA\P_Scope\SpecMon.bmp');
for n:=0 to 500 do FFT1.RealSpec[n]:=4*DisplayBuff[n];

FFT1.Transform;
 for nfft:=1 to 500 do
  begin
   case SpectrumType.ItemIndex of
    0 : begin
        y0 := FFT1.Magnitude[nfft-2];
        {y1 := FFT1.Magnitude[nfft+1];
       end;
    1 : begin
        y0 := 0.00002*FFT1.PowerSpec[nfft];
        y1 := 0.00002*FFT1.PowerSpec[nfft+1];
       end;
    2 : begin
        y0 := 20*FFT1.Phase[nfft];
        y1 := 20*FFT1.Phase[nfft+1];
       end;
    3 : begin
        y0 := FFT1.FourSerCosCoeff[nfft];
        y1 := FFT1.FourSerCosCoeff[nfft+1];
       end;
    4 : begin
        y0 := FFT1.FourSerSinCoeff[nfft];
        y1 := FFT1.FourSerSinCoeff[nfft+1];
       end;
    5 : begin
        y0 := 0.004*FFT1.RealSpec[nfft+1];
        y1 := 0.004*FFT1.RealSpec[nfft+2];
       end;
    6 : begin
        y0 := 0.004*FFT1.ImagSpec[nfft+1];
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
          y1 := 0.004*FFT1.ImagSpec[nfft+2];
         end;
      end;

      if Continues1.Checked then
       begin
        Plot.Canvas.Pen.Color:=Color1;
        //Plot Positive Frequency
        Plot.Canvas.MoveTo(nfft,Round(200-y0));
        Plot.Canvas.LineTo(nfft+1,Round(200-y1));
       end
       else
       begin
       //Plot Positive Frequency
       Plot.Canvas.Pixels[nfft,Round(200-y0)]:=Color1;
       end;
    end;
end;

procedure TScope.PlotWave(WaveSize:LongInt);
var Dt,i:Integer;
begin
  if WaveSize>2000 then
   begin
    Dt:=Round((WaveSize-2000)/500);
    Plot.Picture.LoadFromFile('E:\DATA\P_Scope\MONITOR.BMP');
    for i:=0 to 500 do DisplayBuff[i]:=DataWave[50+(i*dt)];
    for i:=1 to 499 do
     begin
      Plot.Canvas.MoveTo(i,72+DisplayBuff[i]);
      Plot.Canvas.Pen.Color:=Color1;
      Plot.Canvas.LineTo(i+1,72+DisplayBuff[i+1]);
     end;
  end
  else
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    begin
     MessageDlg('Can not plot this wave file ', mtInformation,
      [mbOk], 0);
    end;
end;}


procedure TScope.LoadInitial;
var IniFile:TIniFile;
    LevelTrig,TimeBase,VoltDiv,ColorS,YPos,XPos:integer;
    FontShow,ScreenShow,Mode:String;
    TrigOn,Invert:Boolean;

begin
    IniFile:=TIniFile.Create('PCScope.ini');
    // Channel 1 Load parameter from PCScope.ini
    with IniFile do
     begin
     VoltDiv:=ReadInteger('Channel 1','Volt/Div',0);
     ColorS:=ReadInteger('Channel 1','Color',0);
     YPos:=ReadInteger('Channel 1','YPos',0);
     Mode:=ReadString('Channel 1','Mode','');
     Invert:=ReadBool('Channel 1','Invert',False);
     end;

     Ch1Volt.Value:=VoltDiv;
     Channel1Chang;
     Color1:=ColorS;
     Shape1.Brush.Color:=Color1;
     Y1Pos.Value:=YPos;
     if Mode='AC' then AC1.Down:=True;
     if Mode='DC' then DC1.Down:=True;
     if Mode='GND' then GND1.Down:=True;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
if Invert then
  begin
  //Invert plot
  Inv1.Tag:=1;
  Inv1.Down:=True;
  //Inv11.Down:=True;
  end
  else
  begin
  //Noninvert plot
  Inv1.Tag:=0;
  Inv1.Down:=False;
 //Inv11.Down:=False;
  end;

// Channel 2 Load parameter from PCScope.ini
//with IniFile do
 //begin
 //VoltDiv:=ReadInteger('Channel 2','Volt/Div',0);
 //ColorS:=ReadInieger('Channel 2','Color',0);
 //YPos:=ReadInteger('Channel 2','YPos',0);
 //Mode:=ReadString('Channel 2','Mode','');
 //Invert:=ReadBool('Channel 2','Invert',False);
 //end;

 //Ch2Volt.Value:=VoltDiv;
 //Channel2Chang;
 //Color1:=ColorS;
 //Y2Pos.Value:=YPos;
 //if Mode='AC' then AC2.Down:=True;
 //if Mode='DC' then DC2.Down:=True;
 //if Mode='GND' then GND2.Down:=True;

 //if Invert then
  //begin
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
 //Invert plot
 //Inv2.Tag:=1;
 //Inv2.Down:=True;
 //Inv22.Down:=True;
 //end
 //else
 //begin
 //Noninvert plot
 //Inv2.Tag:=0;
 //Inv2.Down:=False;
 //Inv22.Down:=False;
 //end;

//Load Time Setting From INI File
with IniFile do
 begin
 TimeBase:=ReadInteger('TimeBase','Time/Div',0);
 LevelTrig:=ReadInteger('TimeBase','Level',0);
 XPos:=ReadInteger('TimeBase','XPos',0);
 end;

 Time.Value:=TimeBase;
 TimebaseChang;
 LevelV.Value:=LevelTrig;
 XPosT.Value:=XPos;

//Setup Parameter Load From INI File
with IniFile do
 begin
 MonitorColor:=ReadInteger('Setup','MonitorColor',0);
 MonitorName:=ReadString('Setup','MonitorFile','');
 FontShow:=ReadString('Setup','Font','');
 ScreenShow:=ReadString('Setup','Screen','');
 TrigOn:=ReadBool('Setup','TrigOn',False);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    P90.Font.Color:=MonitorColor;
    P10.Font.Color:=MonitorColor;
    if FontShow='English' then
     begin
      EnglishFont1.Click;
     end
     else
     begin
     ThaiFont1.Click;
     end;

    if ScreenShow='Discreat' then
     begin
     Discreat1.Click;
     end
     else
     begin
     Continues1.Click;
     end;

     On1.Checked:=TrigOn;
    end;

end;

procedure TScope.Channel1Chang;
begin
   case Ch1Volt.Value of
  1:begin
    LV1.Caption:='1 V/Div';
    GainCtl:=1;//1;
    AttCtl:=1;
    Rate:=3.9;
    if not Inv1.Down then
     DPos:=-466
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
   else
   DPos:=466;
  end;
 2:begin
   LV1.Caption:='0.8 V/Div';
   GainCtl:=2;//1
   AttCtl:=1;
   Rate:=2.1;
   if not Inv1.Down then
    DPos:=-145
    else
    DPos:=145;
  end;
 3:begin
   LV1.Caption:='0.4 V/Div';
   GainCtl:=3;//2
   AttCtl:=1;
   Rate:=2.2;
   if not Inv1.Down then
    DPos:=-127
    else
    DPos:=127;
  end;
 4:begin
   LV1.Caption:='0.2 V/Div';
   GainCtl:=0;
   AttCtl:=0;
   Rate:=1.8;
   if not Inv1.Down then
    DPos:=-100
    else
    DPos:=100;
  end;
 5:begin
   LV1.Caption:='0.1 V/Div';
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
  GainCtl:=1;
  AttCtl:=0;
  Rate:=1.8;
  if not Inv1.Down then
   DPos:=-100
   else
   DPos:=100;
 end;
6:begin
  LV1.Caption:='40 mV/Div';
  GainCtl:=2;
  AttCtl:=0;
  Rate:=1.9;
  if not Inv1.Down then
   DPos:=-115
   else
   DPos:=115;
 end;
7:begin
  LV1.Caption:='20 mV/Div';
  GainCtl:=3;
  AttCtl:=0;
  Rate:=1.9;
  if not Inv1.Down then
   DPos:=-110
   else
   DPos:=110;
 end;
8:begin
  LV1.Caption:='10 mV/Div';
  GainCtl:=4;
  AttCtl:=0;
  Rate:=2;
  if not Inv1.Down then
   DPos:=-125
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    else
    DPos:=125;
   end;
  9:begin
   LV1.Caption:='5 mV/Div';
   GainCtl:=5;
   AttCtl:=0;
   Rate:=2.4;
   if not Inv1.Down then
    DPos:=-180
    else
    DPos:=180;
   end;
 end;
end;

//procedure TScope.Channel2Chang;
//begin
//case Ch2Volt.Value of
 //1:begin
   //LV2.Caption:='1 V/Div';
  //end;
 //2:begin
   //LV2.Caption:='0.8 V/Div';
  //end;
 //3:begin
   //LV2.Caption:='0.4 V/Div';
  //end;/

 //4:begin
  // LV2.Caption:='0.2 V/Div';
  //end;
 //5:begin
  // LV2.Caption:='0.1 V/Div';
  //end;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
//6:begin
 // LV2.Caption:='40 mV/Div';
 //end;
//7:begin
 // LV2.Caption:='20 mV/Div';
//   end;
//8:begin
 // LV2.Caption:='10 mV/Div';
//   end;
// 9:begin
 //  LV2.Caption:='5 mV/Div';
 //end;
//end;
//end;

procedure TScope.TimebaseChang;
begin
 case Time.Value of
  1:begin
  //Set Spectrum Rang
    S1.Caption:='1k';
    S2.Caption:='2k';
    S3.Caption:='3k';
    S4.Caption:='4k';
  //Set Time Base
    LT.Caption:='2.5 mS/Div';
    RamE:=R128K;
    RBuff:=128*1024;
    DivRam:=256;
    SamPlingRate:=3;
    ChkRam.Interval:=400;
    TimeScale:=' mS';
    TB:=2.5;
//     ChkRam.Interval:=StrToInt(Edit5.Text);
//     Label9.Caption:=IntToStr(ChkRam.Interval);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
  end;
 2:begin
 //Set Spectrum Rang
  S1.Caption:='2k';
  S2.Caption:='4k';
  S3.Caption:='6k';
  S4.Caption:='8k';
 //Set Time Base
  LT.Caption:='1.2 mS/Div';
  RamE:=R128K;
  RBuff:=128*1024;
  DivRam:=256;
  SamPlingRate:=2;
  ChkRam.Interval:=400;
  TimeScale:=' mS';
  TB:=1.2;
//    ChkRam.Interval:=StrToInt(Edit5.Text);
//    Label9.Caption:=IntToStr(ChkRam.Interval);
  end;
 3:begin
 //Set Spectrum Rang
  S1.Caption:='4k';
  S2.Caption:='8k';
  S3.Caption:='12k';
  S4.Caption:='16k';
 //Set Time Base
  LT.Caption:='0.64 mS/Div';
  RamE:=R128K;
  RBuff:=128*1024;
  DivRam:=128;
  SamPlingRate:=2;
  ChkRam.Interval:=400;
  TimeScale:=' mS';
  TB:=0.64;
//    ChkRam.Interval:=StrToInt(Edit5.Text);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
//    Label9.Caption:=IntToStr(ChkRam.Interval);
     end;
   4:begin
    //Set Spectrum Rang
     S1.Caption:='8k';
     S2.Caption:='16k';
     S3.Caption:='24k';
     S4.Caption:='32k';
    //Set Time Base
     LT.Caption:='0.32 mS/Div';
     RamE:=R128K;
     RBuff:=128*1024;
     DivRam:=64;
     SamPlingRate:=2;
     ChkRam.Interval:=400;
     TimeScale:=' mS';
     TB:=0.32;
//    ChkRam.Interval:=StrToInt(Edit5.Text);
//    Label9.Caption:=IntToStr(ChkRam.Interval);
     end;
   5:begin
    //Set Spectrum Rang
     S1.Caption:='16k';
     S2.Caption:='32k';
     S3.Caption:='48k';
     S4.Caption:='64k';
    //Set Time Base
     LT.Caption:='0.16 mS/Div';
     RamE:=R16K;
     RBuff:=16*1024;
     DivRam:=32;
     SamPlingRate:=2;
     ChkRam.Interval:=300;
     TimeScale:=' mS';
     TB:=0.16;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
//     ChkRam.Interval:=StrToInt(Edit5.Text);
//     Label9.Caption:=IntToStr(ChkRam.Interval);
    end;
   6:begin
    //Set Spectrum Rang
     S1.Caption:='32k';
     S2.Caption:='64k';
     S3.Caption:='96k';
     S4.Caption:='128k';
    //Set Time Base
     LT.Caption:='80 uS/Div';
     RamE:=R8K;
     RBuff:=8*1024;
     DivRam:=1;
     SamPlingRate:=6;
     ChkRam.Interval:=300;
     TimeScale:=' uS';
     TB:=80;
//     ChkRam.Interval:=StrToInt(Edit5.Text);
//     Label9.Caption:=IntToStr(ChkRam.Interval);
    end;
   7:begin
    //Set Spectrum Rang
     S1.Caption:='64k';
     S2.Caption:='128k';
     S3.Caption:='192k';
     S4.Caption:='256k';
    //Set Time Base
     LT.Caption:='40 uS/Div';
     RamE:=R8K;
     RBuff:=8*1024;
     DivRam:=1;
     SamPlingRate:=5;
     ChkRam.Interval:=300;
     TimeScale:=' uS';
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    TB:=40;
//    ChkRam.Interval:=StrToInt(Edit5.Text);
//    Label9.Caption:=IntToStr(ChkRam.Interval);
   end;
  8:begin
  //Set Spectrum Rang
    S1.Caption:='128k';
    S2.Caption:='256k';
    S3.Caption:='384k';
    S4.Caption:='512k';
  //Set Time Base
    LT.Caption:='20 uS/Div';
    RamE:=R8K;
    RBuff:=8*1024;
    DivRam:=1;
    SamPlingRate:=4;
    ChkRam.Interval:=300;
    TimeScale:=' uS';
    TB:=20;
//    ChkRam.Interval:=StrToInt(Edit5.Text);
//    Label9.Caption:=IntToStr(ChkRam.Interval);
   end;
  9:begin
  //Set Spectrum Rang
    S1.Caption:='256k';
    S2.Caption:='512k';
    S3.Caption:='768k';
    S4.Caption:='1024k';
  //Set Time Base
    LT.Caption:='10 uS/Div';
    RamE:=R4K;
    RBuff:=4*1024;
    DivRam:=1;
    SamPlingRate:=3;
    ChkRam.Interval:=300;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    TimeScale:=' uS';
    TB:=10;
//    ChkRam.Interval:=StrToInt(Edit5.Text);
//    Label9.Caption:=IntToStr(ChkRam.Interval);
   end;
  10:begin
  //Set Spectrum Rang
    S1.Caption:='512k';
    S2.Caption:='1024k';
    S3.Caption:='1536k';
    S4.Caption:='2048k';
  //Set Time Base
   LT.Caption:='5 uS/Div';
    RamE:=R4K;
    RBuff:=4*1024;
    DivRam:=1;
    SamPlingRate:=2;
    ChkRam.Interval:=300;
    TimeScale:=' uS';
    TB:=5;
//    ChkRam.Interval:=StrToInt(Edit5.Text);
//    Label9.Caption:=IntToStr(ChkRam.Interval);
   end;
  end;

end;

procedure TScope.CallTime1(Position:Integer);
var DataTemp:String;
begin
  if (Position>=10) and (Position<=510) then
   begin
   t1:=TB*(Position-10)*2/100;
   DT12:=t2-t1;
   if DT12<>0 then Frq:=(1/DT12)*1000;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
ST1.Caption:=FloatToStr(t1)+TimeScale;
DT.Caption:=FloatToStrF(DT12,ffNumber,1,2)+TimeScale;
if TimeScale=' uS' then
 begin
 DataTemp:=Copy(FloatToStr(Frq),1,Pos('.',floatToStr(Frq))+3);
 F1.Caption:=DataTemp+'KHz';
 end;

if TimeScale=' mS' then
 begin
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
  if o1>500+ScreenLeft-6 then
   begin
   o1:=500;
   end
   else
   begin
   o1:=o1-10;
   end;
 end;

 if o2<ScreenLeft-7 then
 begin
 o2:=-1;
 end
 else
 begin
  if o2>500+ScreenLeft-6 then
  begin
  o2:=500;
  end
  else
  begin
  o2:=o2-10;
  end;
 end;
 PaintMark(o1,o2);
end;

procedure TScope.DisplaySignal;
var Exit,Smooth:Boolean;
  n,yo:Integer;
  DataTemp:byte;
begin
 //Rate:=StrToFloat(Edit5.Text);
 Plot.Picture.LoadFromFile('Monitor.bmp');
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
NTemp:=0;
Exit:=False;
NTemp:=10;

if On1.Checked then
 begin
  Repeat
  DataTemp:=Data[NTemp];
  NTemp:=NTemp+1;

  if NTemp<RBuff then
   begin
    if (DataTemp=LevelTrig) and (Data[NTEmp+1]>=LevelTrig) and (Data
[NTEmp+2]>=LevelTrig) and
      (Data[NTEmp+3]>=LevelTrig) and (Data[NTEmp+4]>=LevelTrig) then
      begin
       NTemp:=NTemp;
       Exit:=True;
      end;
   end
   else
   begin
   Exit:=True;
   end;
  Until Exit=True;
 end
 else
 begin
 NTemp:=200;
 end;

//NTemp:=StrToInt(Edit4.Text);

if (NTemp+(500*DivRam))<=RBuff then
 begin
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
n:=NTemp;
While n<=(NTemp+(500*DivRam)) do
 begin
   case Inv1.Tag of
    0:yo:=GndNor-Round(Rate*data[n])-(YPos1+DPos); //Normal Plot
    1:yo:=GndInv+Round(Rate*data[n])-(YPos1-DPos); //Invert plot
   end;

  if Continues1.Checked then
    begin
     Plot.Canvas.MoveTo((n-NTemp) div DivRam+XPos,yo);
     //DataBuff[n-NTemp]:=yo;
     Plot.Canvas.Pen.Color:=Color1;
     case Inv1.Tag of
      0:yo:=GndNor-Round(Rate*data[n+DivRam])-(YPos1+DPos); //Normal Plot
      1:yo:=GndInv+Round(Rate*data[n+DivRam])-(YPos1-DPos); //Invert plot
     end;
     Plot.Canvas.LineTo((n-NTemp) div DivRam+XPos,yo);
    end
    else
    begin
     Plot.Canvas.Pixels[(n-NTemp) div DivRam+XPos,yo]:=Color1;
    end;
 //DataBuff[n-NTemp]:=yo;
 n:=n+DivRam;
 end;


end;
Label20.Caption:=IntToStr(n);

end;

{procedure TScope.DisplayForPrintSpectrum;
var n,nfft:integer;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
  y0,y1:Double;
begin
 FFT1.ClearImag;
 FFT1.SpectrumSize:=1024;
 NTemp:=200;
  if (NTemp+(500*DivRam))<=RBuff then
   begin
    n:=NTemp;
    While n<=(NTemp+(500*DivRam)) do
     begin
      n:=n+DivRam;
      nfft:=(n-NTemp-DivRam) Div 2;
      FFT1.RealSpec[nfft]:=4*data[n+DivRam];
     end;
   end;
  FFT1.Transform;
  for nfft:=0 to 500 do
   begin
    case SpectrumType.ItemIndex of
     0 : begin
          y0 := FFT1.Magnitude[nfft];
          y1 := FFT1.Magnitude[nfft+1];
         end;
     1 : begin
          y0 := 0.00002*FFT1.PowerSpec[nfft];
          y1 := 0.00002*FFT1.PowerSpec[nfft+1];
         end;
     2 : begin
          y0 := 20*FFT1.Phase[nfft];
          y1 := 20*FFT1.Phase[nfft+1];
         end;
     3 : begin
          y0 := FFT1.FourSerCosCoeff[nfft];
          y1 := FFT1.FourSerCosCoeff[nfft+1];
         end;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    4 : begin
       y0 := FFT1.FourSerSinCoeff[nfft];
       y1 := FFT1.FourSerSinCoeff[nfft+1];
      end;
    5 : begin
       y0 := 0.004*FFT1.RealSpec[nfft+1];
       y1 := 0.004*FFT1.RealSpec[nfft+2];
      end;
    6 : begin
       y0 := 0.004*FFT1.ImagSpec[nfft+1];
       y1 := 0.004*FFT1.ImagSpec[nfft+2];
      end;
   end;

   if Continues1.Checked then
    begin
    { PrinForm.BWMon.Canvas.Pen.Color:=clBlack;
     //Plot Positive Frequency
     PrinForm.BWMon.Canvas.MoveTo(nfft,Round(200-y0));
     PrinForm.BWMon.Canvas.LineTo(nfft+1,Round(200-y1));
     }
    //end
    //else
    //begin
    //Plot Positive Frequency
    //PrinForm.BWMon.Canvas.Pixels[nfft,Round(200-y0)]:=clBlack;
   // end;
 // end;


//end;

procedure TScope.DisplayForPrintSignal;
var Exit,Smooth:Boolean;
   n,yo:Integer;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
  DataTemp:byte;
begin
 //PrinForm.BWMon.Picture.LoadFromFile('E:\DATA\P_Scope\MonBW.bmp');
 NTemp:=0;
 Exit:=False;
 NTemp:=10;

 if On1.Checked then
  begin
   Repeat
   DataTemp:=Data[NTemp];
   NTemp:=NTemp+1;

   if NTemp<RBuff then
    begin
     if (DataTemp=LevelTrig) and (Data[NTEmp+1]>=LevelTrig) and (Data
[NTEmp+2]>=LevelTrig) and
       (Data[NTEmp+3]>=LevelTrig) and (Data[NTEmp+4]>=LevelTrig) then
       begin
        NTemp:=NTemp;
        Exit:=True;
       end;
    end
    else
    begin
    Exit:=True;
    end;
  Until Exit=True;
 end
 else
 begin
 NTemp:=200;
 end;

//NTemp:=StrToInt(Edit4.Text);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
if (NTemp+(500*DivRam))<=RBuff then
 begin
  n:=NTemp;
  While n<=(NTemp+(500*DivRam)) do
   begin
     case Inv1.Tag of
      0:yo:=GndNor-Round(Rate*data[n])-(YPos1+DPos); //Normal Plot
      1:yo:=GndInv+Round(Rate*data[n])-(YPos1-DPos); //Invert plot
     end;
    if Continues1.Checked then
      begin
      ' //PrinForm.BWMon.Canvas.MoveTo((n-NTemp) div DivRam+XPos,yo);
       //PrinForm.BWMon.Canvas.Pen.Color:=clBlack;
       case Inv1.Tag of
        0:yo:=GndNor-Round(Rate*data[n+DivRam])-(YPos1+DPos); //Normal Plot
        1:yo:=GndInv+Round(Rate*data[n+DivRam])-(YPos1-DPos); //Invert plot
       end;
       //PrinForm.BWMon.Canvas.LineTo((n-NTemp) div DivRam+XPos,yo);
      end
      else
      begin
       //PrinForm.BWMon.Canvas.Pixels[(n-NTemp) div DivRam+XPos,yo]:=clBlack;
      end;
   n:=n+DivRam;
   end;
  end;

end;

{procedure TScope.DisplaySpectrum;
var n,nfft:integer;
   y0,y1:Double;
begin
  FFT1.ClearImag;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้