สำนักหอสมุดกลาง_พระจอมเกล้าลาดกระบัง

การพัฒนาการ์ดออสซิลโลสโคปด้วยเอฟพีจีเอ

OSCILLOSCOPE CARD DEVELOPMENT BY FPGA

T046232

โดย

นายจีระพงษ์ พุทธันบุตร

นายณัฐวุฒิ ยอดจิตใจ

นายสิทธิพงษ์ พรหมแก้ว

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต

สาขาวิชาอิเล็กทรอนิกส์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2544

เลขหมู่.....
เลขทะเบียน 46232
วัน, เดือน, ปี 21 มี.ค. 2546

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

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การพัฒนาการ์ดออสซิลโลสโคปด้วยเอฟพีจีเอ

OSCILLOSCOPE CARD DEVELOPMENT BY FPGA

โดย

นายจีระพงษ์ พุทธันบุตร 42515720

นายณัฐวุฒิ ยอดจิตใจ 42515726

นายสิทธิพงษ์ พรหมแก้ว 42515749

อาจารย์ที่ปรึกษา

รศ.ดร.มนัส สังวรศิลป์

ปริญญานิพนธ์สำหรับปริญญาวิศวกรรมศาสตรบัณฑิต

สาขาวิชาอิเล็กทรอนิกส์

คณะวิศวกรรมศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2544

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การพัฒนาการ์ดออสซิลโลสโคปด้วยเอฟพีจีเอ

OSILLOSCOPE CARD DEVELOPMENT BY FPGA

1.นายจีระพงษ์ พุทธันบุตร รหัส 42515720

2.นายณัฐวุฒิ ยอดจิตรใจ รหัส 42515726

3.นายสิทธิพงษ์ พรหมแก้ว รหัส 42515749

โครงงานนี้ได้รับการตรวจสอบแล้ว พร้อมที่จะทำการตรวจสอบแล้ว

.....

(รศ.ดร.มนัส สังวรศิลป์)

อาจารย์ที่ปรึกษา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ปริญญานิพนธ์ ปีการศึกษา 2544

ภาควิชาอิเล็กทรอนิกส์

คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เรื่อง การพัฒนาการ์ดออสซิลโลสโคปด้วยเอฟพีจีเอ

ผู้จัดทำ

1. นายจีระพงษ์ พุทธันบุตร รหัส 42515720
2. นายณัฐวุฒิ ยอดจิตรใจ รหัส 42515726
3. นายสิทธิพงษ์ พรหมแก้ว รหัส 42515749

..... อาจารย์ที่ปรึกษา

(รศ.ดร.มนัส สังวรศิลป์)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การพัฒนาการ์ดออสซิลโลสโคปด้วยเอฟพีจีเอ

ผู้จัดทำ
นายจีระพงษ์ พุทธันบุตร 42515720
นายณัฐวุฒิ ยอดจิตใจ 42515726
นายสิทธิพงษ์ พรหมแก้ว 42515749
อาจารย์ที่ปรึกษา
รศ.ดร.มนัส สังวรศิลป์
ปีการศึกษา 2544

บทคัดย่อ

การพัฒนาการ์ดออสซิลโลสโคปด้วยเอฟพีจีเอ (FPGA:Field Programmable Gate Array) ที่นำเสนอในโครงงานนี้ เป็นการพัฒนานำเอาคอมพิวเตอร์มาประยุกต์ใช้งานวัด และบันทึกรูปสัญญาณต่างๆ ซึ่งในการออกแบบวงจรในส่วนที่เป็นดิจิตอลเดิมทีจะใช้ไอซีสำเร็จรูปทั่วไปมาประกอบเป็นวงจร ซึ่งทำให้วงจรมีขนาดใหญ่และสิ้นเปลืองเนื้อที่ใช้งาน ฉะนั้นจึงได้มีการนำเอฟพีจีเอมาใช้แทนวงจรรวมดิจิตอลสำเร็จรูปดังกล่าว เพื่อให้วงจรมีขนาดเล็กลงและใช้งานได้สะดวกยิ่งขึ้น โดยเริ่มต้นนั้นการนำสัญญาณอนาลอกเข้าสู่คอมพิวเตอร์ เราจะใช้ตัวกลางในการแปลงสัญญาณอนาลอกให้เป็นสัญญาณดิจิตอลก่อน จากนั้นจึงนำสัญญาณดิจิตอลที่ได้ไปทำการประมวลผล แล้วจึงแสดงออกทางจอภาพคอมพิวเตอร์ต่อไป ประโยชน์ที่คาดว่าจะได้รับจากการนำคอมพิวเตอร์มาประยุกต์ใช้งานในลักษณะนี้ก็คือการประหยัดต้นทุนในการซื้อสตอเรจออสซิลโลสโคปซึ่งมีราคาแพงได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Osilloscope Card Development By FPGA

Student

Jeerapong Puthanbut 42515720

Nathawuth Yodchitchai 42515726

Sithipong Promkaew 42515749

Advisor

Assoc. Prof. Dr. Manas Sangworasil

Year 2001

ABSTRACT

Osilloscope card development by the FPGA(Field Programmable Gate Array) ,presented in this project uses the computer to measure and record the signal. In a digital circuit from the past we designed and developed by using the TTL (Transister Transister Logic) integrated circuits so it will be large circuit and consume area for using. So we use the FPGA to instead the TTL integrated circuits for decrease a size of circuit ,and convenient for using. At the beginnig the analog signal is converted by using analog to digital converter and then the digital signal is processed by using a computer and displayed by a monitor. The benefit from this project is saving a cost to spending for the expensive Storage Oscilloscope.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ

	หน้า
บทคัดย่อ	I
ABSTRACT	II
สารบัญ	III
สารบัญรูป	VI
สารบัญตาราง	IX
บทที่1 บทนำ	1
บทที่ 2 ดาต้าแอคควิทชั่น และ คอนเวอร์ชั่น	2
2.1 ทฤษฎีการแซมปลิง	4
2.2 การแปลงสัญญาณอนาลอกเป็นสัญญาณดิจิตอล	7
บทที่ 3 สัญญาณต่างๆบนสล๊อตไอซา(ISA)ของคอมพิวเตอร์	11
3.1 รายละเอียดเกี่ยวกับสัญญาณต่างๆ	11
3.2 บัสของแหล่งจ่ายไฟของระบบ	19
บทที่ 4 เอฟพีจีเอ	21
4.1 บทนำ	21
4.2 เทคโนโลยีของ FPGA	28
4.3 ทำไมการออกแบบถึงทำได้ง่ายและสะดวกรวดเร็ว	30
4.4 การออกแบบโดยใช้ภาษาอธิบายพฤติกรรมของฮาร์ดแวร์ (HDL)	31
4.5 การสังเคราะห์วงจร	32
4.6 การใช้งานโปรแกรม Max Plus II เบื้องต้น	33
4.7 สรุปขั้นตอนการใช้ MAX+PLUS IIวาดวงจร	45
บทที่ 5 หลักการออกแบบวงจร และโปรแกรมควบคุมการทำงาน	47
5.1 วงจรลดทอนสัญญาณ	48
5.2 วงจรเลือกโหมดการวัดและวงจรบัฟเฟอร์	49
5.3 วงจรขยายสัญญาณแนวตั้ง	50
5.4 วงจรยกระดับสัญญาณ	52

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หน้า

5.5 วงจรแปลงสัญญาณอนาลอกให้เป็นสัญญาณดิจิตอล 53
5.6 หลักการทำงานวงจรดิจิตอล 54
5.7 วงจรถอดรหัสตำแหน่งพอร์ท 56
5.8 วงจรควบคุมส่วนต่างๆของระบบ 58
5.9 วงจรควบคุมข้อมูลแบบ 2 ทิศทาง 59
5.10 วงจรกำเนิดสัญญาณนาฬิกาหลายความถี่ 60
5.11 วงจรนับตำแหน่งหน่วยความจำ 62
5.12 วงจรหน่วยความจำ 63
5.13 การเขียนโปรแกรม 64
5.14 การออกแบบโปรแกรม 65
บทที่ 6 การทดสอบการทำงาน 68
6.1 วัตถุประสงค์ 68
6.2 การทดลองวงจรลดทอน 68
6.3 การทดลองวงจรบัฟเฟอร์และวงจรเลือกความถี่ 69
6.4 การทดลองวงจรขยาย 69
6.5 การทดลองวงจรยกระดับสัญญาณ 70
6.6 การทดลองวงจรแปลงสัญญาณอนาลอกเป็นสัญญาณดิจิตอล 70
6.7 การทดลองวงจรกำเนิดสัญญาณนาฬิกาหลายความถี่ 71
6.7 ทดลองวัดสัญญาณ โดยใช้โปรแกรมคอมพิวเตอร์ออสซิลโลสโคป 72
บทที่ 7 สรุปผลการทดลอง 92
7.1 บทสรุปและวิจารณ์ 92
7.2 ปัญหาที่พบในโครงงาน 92
7.3 การแก้ปัญหา 92
7.4 ประโยชน์ที่ได้รับจากโครงงาน 93
7.5 แนวทางพัฒนา 93
7.6 คุณสมบัติของวงจร 93

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หน้า

เอกสารอ้างอิง 94

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูป

	หน้า
รูปที่ 2.1 ระบบควบคุมที่มีการประมวลผลข้อมูลทางดิจิตอล	2
รูปที่ 2.2 วิธีการพื้นฐานของADC	2
รูปที่ 2.3 ทรานสเฟอร์ฟังก์ชันของคอมพาราเตอร์	2
รูปที่ 2.4 แสดงการผิดผลาดในการวัดใน Aperture Time	4
รูปที่ 2.5 แสดง Spectrum ของสัญญาณอนาลอกที่จะถูกสุ่ม	5
รูปที่ 2.6 หลังจากการสุ่มเกิด Frequency Folding	6
รูปที่ 2.7 การเกิด Alias Frequency จากการสุ่มด้วยความถี่ต่ำกว่า 2 เท่า ของสัญญาณอนาลอก	6
รูปที่ 2.8 แสดงการต่อวงจร Parallel Comparator A/D Converter	8
รูปที่ 2.9 วงจรเปลี่ยนสัญญาณเอทูดีแบบสโลปเดี่ยว	9
รูปที่ 3.1 แผนผังขาต่อใช้งานของไอซาบัส	13
รูปที่ 4.1 แสดงแผนผังการแบ่งกลุ่มของวงจรรวม ASIC	22
รูปที่ 4.2 แสดงวงจรพื้นฐานของอุปกรณ์พีแอลดีซึ่งอยู่ในรูปผลคูณร่วมบวก	24
รูปที่ 4.3 แสดงลักษณะของพรอมเมื่อเปรียบเทียบเป็นวงจรในรูปผลคูณร่วมบวก	25
รูปที่ 4.4 แสดงโครงสร้างภายในของพีเอแอล	26
รูปที่ 4.5 แสดงวงจรพื้นฐานภายในของพีแอลเอ	27
รูปที่ 4.6ก โครงสร้างภายในของ FPGA ตระกูล MAX7000S	29
รูปที่ 4.6ข โครงสร้างภายในของ FPGA ตระกูล FLEK10K	30
รูปที่ 4.7 การโปรแกรมลงในชิพ	31
รูปที่ 4.8 โปรแกรม Max+Plus II 9.5 BASELINE	33
รูปที่ 4.9 การตั้งชื่อโปรเจค	34
รูปที่ 4.10 การเลือกประเภทไฟล์	34

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หน้า

รูปที่ 4.11 วงจรตัวอย่าง 35
รูปที่ 4.12 การกำหนดสัญลักษณ์ 36
รูปที่ 4.13 การบันทึกข้อมูล 37
รูปที่ 4.14 การกำหนดอุปกรณ์ 38
รูปที่ 4.15 รูปการคอมไพล์ 38
รูปที่ 4.16 การกำหนดโหนด 39
รูปที่ 4.17 การจำลองการทำงาน 40
รูปที่ 4.18 การบันทึกรูปกราฟ 41
รูปที่ 4.19 การจำลองการทำงานอีกครั้ง 42
รูปที่ 4.20 การจำลองการหน่วงเวลา 43
รูปที่ 4.21 การกำหนดขาใช้งาน 44
รูปที่ 4.22 การโปรแกรม 45
รูปที่ 5.1 บล๊อกไดอะแกรมแสดงการทำงานของคอมพิวเตอร์ออสซิลโลสโคปการ์ด 47
รูปที่ 5.2 วงจรลดทอนสัญญาณ 48
รูปที่ 5.3 วงจรเลือกโหมดการวัดและวงจรบัฟเฟอร์ 49
รูปที่ 5.4 วงจรขยายสัญญาณ 50
รูปที่ 5.5 วงจรยกระดับสัญญาณ 52
รูปที่ 5.6 วงจรแปลงสัญญาณอนาลอกให้เป็นสัญญาณดิจิตอล 53
รูปที่ 5.7 บล๊อกไดอะแกรมแสดงการทำงานของวงจรส่วนดิจิตอลในเอฟพีจีเอ 54
รูปที่ 5.8 วงจรถอดรหัสตำแหน่งพอร์ท 56
รูปที่ 5.9 วงจรควบคุมส่วนต่างๆของระบบ 58
รูปที่ 5.10 วงจรควบคุมข้อมูลแบบ 2 ทิศทาง 59
รูปที่ 5.11 วงจรกำเนิดสัญญาณนาฬิกาหลายความถี่ 60
รูปที่ 5.12 วงจรนับตำแหน่งหน่วยความจำ 62
รูปที่ 5.13 วงจรหน่วยความจำ 63
รูปที่ 5.14 แสดงโปรแกรมคอมพิวเตอร์ ออสซิลโลสโคป 64
รูปที่ 5.15 แสดงผังงานการเขียนโปรแกรมควบคุม 67

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หน้า

รูปที่ 6.1 ผลการวัดสัญญาณซายน์ความถี่ 10 Hz 72
รูปที่ 6.2 ผลการวัดสัญญาณซายน์ความถี่ 20 Hz 73
รูปที่ 6.3 ผลการวัดสัญญาณซายน์ความถี่ 50 Hz 73
รูปที่ 6.4 ผลการวัดสัญญาณซายน์ความถี่ 100 Hz 74
รูปที่ 6.5 ผลการวัดสัญญาณซายน์ความถี่ 1 kHz 74
รูปที่ 6.6 ผลการวัดสัญญาณซายน์ความถี่ 10 kHz 75
รูปที่ 6.7 ผลการวัดสัญญาณซายน์ความถี่ 100 kHz 75
รูปที่ 6.8 ผลการวัดสัญญาณซายน์ความถี่ 1 MHz 76
รูปที่ 6.9 ผลการวัดสัญญาณสี่เหลี่ยมความถี่ 100 Hz 76
รูปที่ 6.10 ผลการวัดสัญญาณสี่เหลี่ยมความถี่ 1 kHz 77
รูปที่ 6.11 ผลการวัดสัญญาณสี่เหลี่ยมความถี่ 10 kHz 77
รูปที่ 6.12 ผลการวัดสัญญาณสี่เหลี่ยมความถี่ 100 kHz 78
รูปที่ 6.13 ผลการวัดสัญญาณสามเหลี่ยม ความถี่ 100 Hz 78
รูปที่ 6.14 ผลการวัดสัญญาณสามเหลี่ยม ความถี่ 1 kHz 79
รูปที่ 6.15 ผลการวัดสัญญาณสามเหลี่ยม ความถี่ 10 kHz 79
รูปที่ 6.16 ผลการวัดสัญญาณสามเหลี่ยม ความถี่ 100 kHz 80
รูปที่ 6.17 ผลการวัดสัญญาณแรมป์ ความถี่ 100 Hz 80
รูปที่ 6.18 ผลการวัดสัญญาณแรมป์ ความถี่ 1 kHz 81
รูปที่ 6.19 ผลการวัดสัญญาณแรมป์ ความถี่ 10 kHz 81
รูปที่ 6.20 ผลการวัดสเปคตรัมของสัญญาณซายน์ความถี่ 10 Hz 82
รูปที่ 6.21 ผลการวัดสเปคตรัมของสัญญาณซายน์ความถี่ 20 Hz 83
รูปที่ 6.22 ผลการวัดสเปคตรัมของสัญญาณซายน์ความถี่ 50 Hz 83
รูปที่ 6.23 ผลการวัดสเปคตรัมของสัญญาณซายน์ความถี่ 100 Hz 84
รูปที่ 6.24 ผลการวัดสเปคตรัมของสัญญาณซายน์ความถี่ 1 kHz 84
รูปที่ 6.25 ผลการวัดสเปคตรัมของสัญญาณซายน์ความถี่ 10 kHz 85
รูปที่ 6.26 ผลการวัดสเปคตรัมของสัญญาณซายน์ความถี่ 100 kHz 85
รูปที่ 6.27 ผลการวัดสเปคตรัมของสัญญาณซายน์ความถี่ 1 MHz 86

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หน้า

รูปที่ 6.28 ผลการวัดสเปคตรัมของสัญญาณสี่เหลี่ยมความถี่ 100 Hz 86
รูปที่ 6.29 ผลการวัดสเปคตรัมของสัญญาณสี่เหลี่ยมความถี่ 1 kHz 87
รูปที่ 6.30 ผลการวัดสเปคตรัมของสัญญาณสี่เหลี่ยมความถี่ 10 kHz 87
รูปที่ 6.31 ผลการวัดสเปคตรัมของสัญญาณสี่เหลี่ยมความถี่ 100 kHz 88
รูปที่ 6.32 ผลการวัดสเปคตรัมของสัญญาณสามเหลี่ยมความถี่ 100 Hz 88
รูปที่ 6.33 ผลการวัดสเปคตรัมของสัญญาณสามเหลี่ยมความถี่ 1 kHz 89
รูปที่ 6.34ผลการวัดสเปคตรัมของสัญญาณสามเหลี่ยมความถี่ 10 kHz 89
รูปที่ 6.35 ผลการวัดสเปคตรัมของสัญญาณสามเหลี่ยมความถี่ 100 kHz 90
รูปที่ 6.36 ผลการวัดสเปคตรัมของสัญญาณสามเหลี่ยมความถี่ 100 Hz 90
รูปที่ 6.37 ผลการวัดสเปคตรัมของสัญญาณสามเหลี่ยมความถี่ 1 kHz 91
รูปที่ 6.38 ผลการวัดสเปคตรัมของสัญญาณสามเหลี่ยมความถี่ 10 kHz 91

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญตาราง

หน้า

ตารางที่ 2.1 ความสัมพันธ์ระหว่างแรงดันอินพุทที่เป็นอนาลอกกับเอาท์พุทที่เป็นดิจิตอล 7

ตารางที่ 5.1 อัตราขยายของวงจรขยายสัญญาณ 51

ตารางที่ 5.2 แสดงความหมายของพอร์ทต่างๆ 57

ตารางที่ 5.3 ความถี่ที่ได้จากวงจรกำเนิดสัญญาณนาฬิกาหลายความถี่ 61

ตารางที่ 6.1 แสดงการทดสอบผลของวงจรกำเนิดสัญญาณนาฬิกาหลายความถี่ 71

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่1
บทนำ

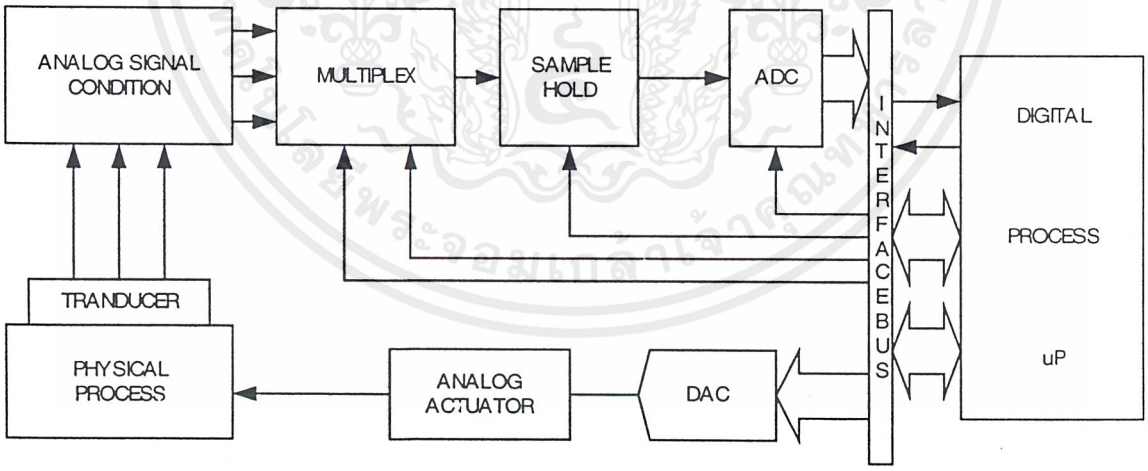
ปัจจุบันเครื่องออสซิลโลสโคป ได้แพร่หลายในวงการอิเล็กทรอนิกส์เป็นอย่างมาก ไม่ว่าจะเป็นการวิเคราะห์ งานซ่อม หรือการเก็บข้อมูลบางอย่างไว้อ้างอิง จะยังมีอื่นๆอีกมากมายในปัจจุบันเครื่องออสซิลโลสโคป รุ่นใหม่ที่เก็บสัญญาณ หรือบันทึกข้อมูลพร้อมยังมีฟังก์ชันพิเศษต่างๆมากมาย ซึ่งก็จะมีราคาค่อนข้างสูงมากและในโรงงานอุตสาหกรรมได้นำเอาคอมพิวเตอร์เข้ามาควบคุมเครื่องจักรเพื่อลดการผิดผลาด ลดเวลาที่สูญเสียต่างๆ และยังสามารถเพิ่มประสิทธิภาพการทำงานได้สูงขึ้น จากข้อดีของออสซิลโลสโคป และคอมพิวเตอร์นั้นจึงเป็นจุดเริ่มต้นของโครงงานนี้ที่จะสร้างการ์ดใช้งานร่วมกับคอมพิวเตอร์ให้เป็นออสซิลโลสโคป แสดงผลหน้าจอที่คอมพิวเตอร์โดยได้นำเอาเอฟพีจีเอมาประยุกต์ใช้งานในการลดขนาดของวงจรในส่วนที่เป็นดิจิตอลให้มีขนาดเล็กลง

ฉะนั้นจึงได้มีการสร้างเครื่องมือชนิดหนึ่งขึ้นมา เรียกว่า PC BASE STORAGE OSCILLOSCOPE เพื่อแก้ปัญหาในจุดนี้ และมีข้อดีคือใช้งานร่วมกับคอมพิวเตอร์ทั่วไปที่มีใช้กันอย่างแพร่หลาย เพียงแต่นำการ์ดของฮาร์ดแวร์เสียบที่สล๊อตของเครื่องคอมพิวเตอร์ และมีซอฟแวร์เป็นตัวควบคุมให้เครื่องคอมพิวเตอร์ทำหน้าที่เป็นดิจิตอลสตอเรจออสซิลโลสโคป โดยแสดงผลที่ได้ออกทางจอภาพ เครื่องพิมพ์ พร้อมบันทึกข้อมูลลงในคอมพิวเตอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 2
ดาต้าแอคควิทชั่น และ คอนเวอร์ชั่น
(Data Acquisition and Convertion)

รูปร่างสัญญาณทางไฟฟ้าที่เราพบเห็นและคุ้นเคยในชีวิตประจำวันจะอยู่ในรูปแบบของสัญญาณที่มีความต่อเนื่องกันหรือเรียกว่าสัญญาณอนาลอก(Analog Signal) ซึ่งแต่เดิมการนำเอาสัญญาณไฟฟ้าดังกล่าวมาประมวลผลจะกระทำในแบบอนาลอก แต่เมื่อเทคนิคการประมวลผลสัญญาณทางดิจิตอลได้รับการพัฒนาขึ้นมาเนื่องจากพบว่าในรูปแบบดิจิตอลการประมวลผล,เก็บ,สื่อสาร และแสดงกระทำได้ง่ายและมีประสิทธิภาพมากกว่า ดังนั้นการเปลี่ยนรูปแบบของสัญญาณ(Conversion)จึงได้มีความจำเป็นขึ้น จากสัญญาณอนาลอกที่มีอยู่ตามธรรมชาติถูกเปลี่ยนมาเป็นสัญญาณดิจิตอล โดย Analog to Digital Converter(ADC) และประมวลผลโดยตัวประมวลผลทางดิจิตอล (Digital Processor) เช่น คอมพิวเตอร์ จากนั้นจะถูกนำมาแสดง หรือ ถูกเปลี่ยนกลับมาในรูปแบบอนาลอก และใช้งานได้ง่ายกว่าโดยใช้ Digital to Analog Converter (DAC) รูปที่ 2.1 แสดงระบบควบคุมโดยการประมวลผลข้อมูลในระบบดิจิตอล

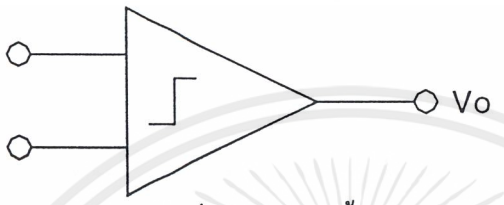


รูปที่ 2.1 ระบบควบคุมที่มีการประมวลผลข้อมูลทางดิจิตอล

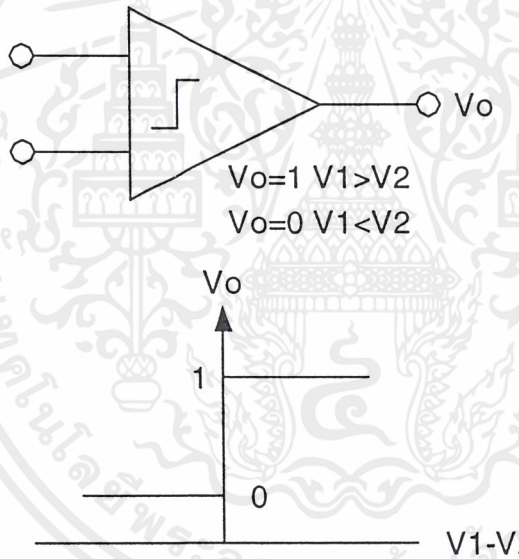
วิธีการแปลงสัญญาณอนาลอกเป็นดิจิตอลแบบง่ายๆแสดงดังรูป 2.2 โดยใช้หลักการของวงจรคอมพาราเตอร์ แรงดันอินพุทไม่ทราบค่า V_x จะต่อเข้ากับอินพุทขาหนึ่งของอนาลอกคอมพาราเตอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

และแรงดันอินพุทที่ขนาดแปรตามเวลา V_R ต่อเข้ากับอีกอินพุทหนึ่งของคอมพาราเตอร์ ลักษณะของทรานเฟอร์ฟังชั่นของคอมพาราเตอร์ ดังแสดงในรูป 2.3 ถ้าแรงดันอินพุท V_1 มากกว่า V_2แล้วแรงดันเอาท์พุทจะเป็น " 1 " ถ้าอินพุท V_1 น้อยกว่า V_2 และเอาท์พุทจะเป็นศูนย์



รูปที่ 2.2 วิธีการพื้นฐานของADC



รูปที่ 2.3 ทรานสเฟอร์ฟังก์ชันของคอมพาราเตอร์

วงจร Analog to Digital Converter (ADC)ที่ใช้งานทั่วไปมีหลายชนิด เช่น Counter type ADC, Tracking ADC , Integrating ADC, Successive Approximation ADC หรือ Parallel (Flash) ADC เป็นต้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.1 ทฤษฎีการแซมปลิง Sampling

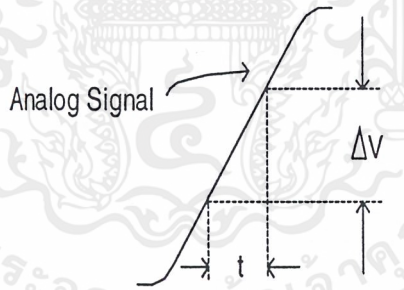
ในการแปลงสัญญาณอนาลอกเป็นดิจิตอลนั้นจะต้องใช้เวลาช่วงหนึ่งในการจัดการซึ่งช่วงเวลาดังกล่าวขึ้นอยู่กับตัวแปลงเช่น

1. ความละเอียดของการเปลี่ยนสัญญาณ
2. เทคนิคการแปลงสัญญาณ
3. ความเร็วในการทำงานของอุปกรณ์ร่วมอื่นๆ

ความเร็วของการแปลงสัญญาณนี้จำเป็นต่อการใช้งานเฉพาะอย่างและความแม่นยำที่ต้องการ

Aperture Time : คือช่วงเวลาในการแปลงสัญญาณ ซึ่งคำว่า Aperture time โดยทั่วไปหมายถึงช่วงเวลาที่เกิดความไม่แน่นอนในการวัดและผลก็คือความผิดพลาด (Error) ต่อค่าที่วัดได้

ในรูปที่ 2.4 สัญญาณอนาลอก V(t) มีอัตราการเปลี่ยน dV/dt ช่วง Aperture Time t_a ดังนั้นช่วงเวลาการเปลี่ยนแปลงสัญญาณอนาลอกจะเท่ากับ



รูปที่ 2.4 แสดงการผิดผลาดในการวัดใน Aperture Time

ดังนั้นหากเวลาที่ ADC ใช้ในการเปลี่ยนสัญญาณในเวลา t_s นี้ รหัสของสัญญาณดิจิตอลที่ได้อาจตรงกับสัญญาณอนาลอกค่าใดค่าหนึ่งในช่วงนี้ และส่วนที่เหลือคือความผิดพลาดที่เกิดขึ้นวึ่งแน่นอนในบ้างครั้งเป็นไปได้ที่รหัสของสัญญาณดิจิตอลจะตรงกับค่าของสัญญาณอนาลอกที่ถูกต้อง

Sample Hold and Aperture error

วงจร Sampling Hold จะทำการสุ่มสัญญาณอินพุท และนำสัญญาณนั้นมาเก็บ (Hold) ในช่วงเวลาหนึ่ง ซึ่งส่วนใหญ่จะใช้การประจุแรงดันนั้นในตัวเก็บประจุที่รั่วไหลต่ำ ดังนั้นในเมื่อสัญญาณอินพุทสามารถคงอยู่ได้นานพอ ทำให้ ADC ไม่จำเป็นต้องมีเวลาในการแปลง (Convertion Time) อย่างรวด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

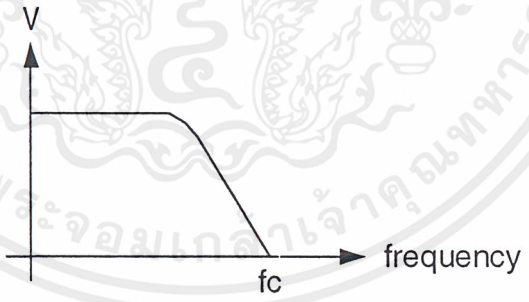
เร็วนัก Aperture Time ของ Sampling Hold คือเวลาตั้งแต่เริ่มสุ่มสัญญาณจนตัวเก็บประจุมีค่าแรงดันจนถึงค่าที่สุ่มวึ่งสำหรับ Sampling Hold แล้ว Aperture Time ขึ้นอยู่กับ Bandwidth และ Switching Time ของอุปกรณ์แอคทีฟที่ใช้ในวงจร วึ่งหาและสร้างได้ง่ายและถูกกว่าการสร้าง ADC ความเร็วสูง

มีปัญหาว่าอัตราการสุ่มสัญญาณนั้นควรจะมีค่าเท่าใดที่จะไม่ทำให้ข้อมูลสูญเสียไปเมื่อสัญญาณนั้นถูกเปลี่ยนกับมาเช่นเดิม อันนี้ขึ้นอยู่กับความถี่ของสัญญาณอนาลอก และทฤษฎีการสุ่มที่กล่าวไว้ว่า

"ถ้าสัญญาณต่อเนื่องซึ่งมีความถี่และฮาร์โมนิกส์ ไม่เกิน f_c แล้วสัญญาณดังกล่าวจะสามารถเปลี่ยนกลับมาได้เหมือนเดิมโดยไม่สูญเสียรายละเอียดหรือเพี้ยนไปถ้าอัตราการสุ่มไม่น้อยกว่า $2f_c$ ต่อวินาที"

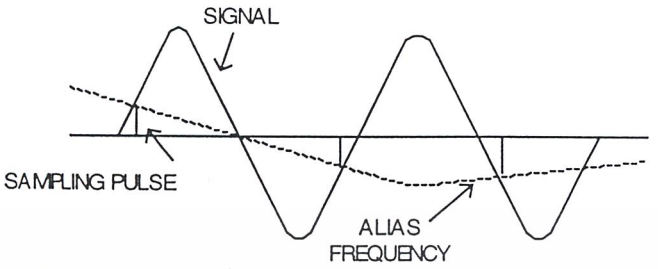
Frequency Folding and Aliasing

จากทฤษฎีการสุ่มสามารถอธิบายลักษณะรูปสเปกตรัม (Spectrum) ของสัญญาณในรูปที่ 2.5 แสดงให้เห็น สเปกตรัมของสัญญาณที่ถูกสุ่มซึ่งแบนด์วิดธ์ไม่เกิน f_c ในขณะที่สัญญานนี้จะถูกสุ่มด้วยความถี่ f_s ขบวณการมอดูเลชั่น (Modulation) จะทำให้แถบสเปกตรัมของสัญญาณสุ่มขยายออกจาก f_s เป็น $2f_s$, $3f_s$,... ได้ดังรูปที่ 2.6



รูปที่ 2.5 แสดง Spectrum ของสัญญาณอนาลอกที่จะถูกสุ่ม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



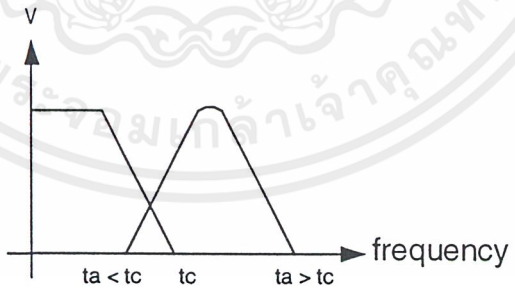
รูปที่ 2.6 หลังจากการสุ่มเกิด Frequency Folding

ถ้าความถี่ของสัญญาณสุ่ม f_s ไม่สูงพอหลังจากการสุ่มสัญญาณสเปคตรัมบางส่วนของ f_s จะมาซ้อนทับกับสเปกตรัมของสัญญาณซึ่งเรียกว่า Frequency Folding หากเป็นเช่นนี้ก็จะทำให้เกิดความเพี้ยนแก่สัญญาณอนาลอกจากการซ้อนทับกันของสเปกตรัม เมื่อสัญญาณถูกเปลี่ยนกลับให้อยู่ในรูปเดิม และถ้าเลื่อนความถี่ของการสุ่มสูงขึ้นจนโอกาสการซ้อนทับของสเปกตรัมหมดไป $(f_s - f_c) = f_c$ จะทำให้การเปลี่ยนกลับของสัญญาณหลังจากถูกสุ่มก็ยังคงเหมือนเดิมได้

จากที่กล่าวมาแสดงการสนับสนุนทฤษฎีการสุ่มที่ว่าให้ $f_s < 2f_c$ นั้นคือการกำจัดการซ้อนทับของสเปกตรัมได้ 2 วิธีคือ

1.ใช้อัตราการสุ่มที่สูงพอ

2. การทำการกรองความถี่ของสัญญาณอนาลอก ก่อนการสุ่มเพื่อให้ Bandwidth ไม่เกินไปกว่า $f_s/2$



รูปที่ 2.7 การเกิด Alias Frequency จากการสุ่มด้วยความถี่ต่ำกว่า 2 เท่า ของสัญญาณอนาลอก

ในทางปฏิบัติแล้วจะยังคงเกิด Frequency Folding ได้เสมอจากส่วนฮาร์โมนิกส์ของสัญญาณรวมทั้งสเปกตรัมของสัญญาณรบกวนที่ยังคงอยู่แม้ว่าจะทำการกรองความถี่มาก่อนหน้านี้แล้วก็ตาม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สำหรับการกำจัดการซ้อนทับของสเปกตัมนี้วิธีที่ได้ผลคือพยายามให้การสุ่มสัญญาณเป็นไปอย่างรวดเร็วมากที่สุด

ผลของการใช้อัตราสุ่มที่ไม่เหมาะสมอีกอย่างหนึ่งเกิดขึ้น ดังรูปที่ 2.7 เรียกว่า Alias Frequency ซึ่งเกิดกับสัญญาณที่เปลี่ยนกลับมาเช่นเดิมหลังจากถูกสุ่มแล้ว

2.2 การแปลงสัญญาณอนาลอกเป็นสัญญาณดิจิตอล (Analog to Digital Converter)

วงจร เอทูดีแบบใช้วงจรเปรียบเทียบหรือแบบ"แฟลช"

(Parallel Comparator Simultaneous "Flash" A/D Converter)

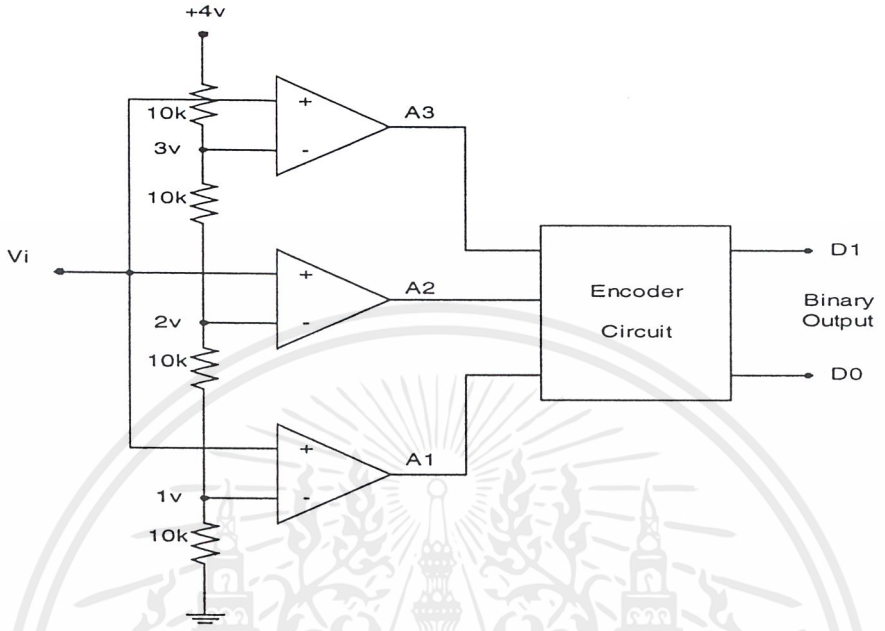
สำหรับการแปลงสัญญาณที่ต้องการความเร็วสูงมากๆ เช่น การแปลงภาพโทรทัศน์,เรดาห์ จำเป็นต้องใช้ ADC แบบพิเศษที่เรียกว่า Flash (Parallel)n ADC ซึ่งแสดงดังรูป Blockdiagram ดังรูปที่ 2.8

หลักการทำงานก็คือ จะใช้คอมพาราเตอร์เปรียบเทียบสัญญาณอนาลอกอินพุทกับสัญญาณอ้างอิงแบ่งแรงดันให้ตรงกับรหัสดิจิตอลซึ่งจะเห็นว่าอุปกรณ์ทางด้านความเร็วจะถูกจำกัดเพียง Propagation time ของคอมพาราเตอร์เท่านั้นแต่อุปสรรคที่สำคัญต่อการพัฒนาวงจรบนชิพไอซี คือ วงจรนี้ต้องการคอมพาราเตอร์ จำนวน 2^n-1 ตัว ซึ่งเป็นจำนวนที่มากพอสมควร

ตารางที่ 2.1 ความสัมพันธ์ระหว่างแรงดันอินพุทที่เป็นอนาลอกกับเอาท์พุทที่เป็นดิจิตอล

แรงดันอินพุท V_{in} (โวลต์)	เอาท์ของวงจรเปรียบเทียบ			เอาท์พุทเลขฐานสอง	
	A1	A2	A3	D1	D2
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

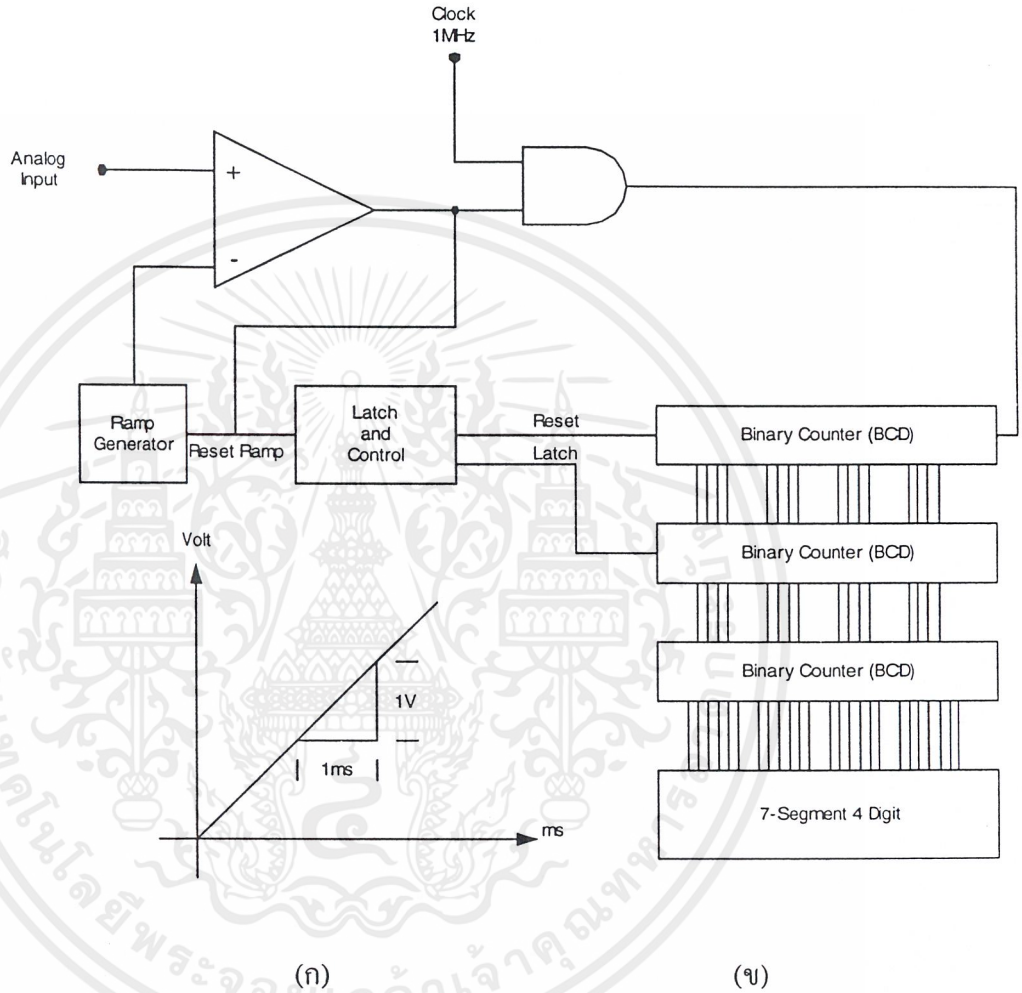
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.8 แสดงการต่อวงจร Parallel Comparator A/D Converter

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

วงจรเอทูดีแบบสโลปเดียวหรือแบบแรมป์(Single Ramp หรือ Single Slope A/D Converter)



รูปที่ 2.9 วงจรเปลี่ยนสัญญาณเอทูดีแบบสโลปเดี่ยว
(ก) ความชันของสัญญาณแรมป์ (ข) แสดงบล็อกไดอะแกรม

สัญญาณแรมป์เป็นบวกมากขึ้นเรื่อยๆจนมากกว่าระดับแรงดันอินพุท เอาท์พุทของวงจรเปรียบเทียบก็ตกลงมาเป็นระดับ "LOW" ปิดแอนด์เกตไม่มีสัญญาณไปให้กับวงจรนับ

วงจรนับจะหยุดนับและเก็บค่าไว้ในแลตช์ (Latch) จากนั้นจะทำการรีเซตวงจรนับ และวงจรกำเนิดสัญญาณแรมป์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สมมติให้สัญญาณนาฬิกามีความถี่ 1 MHz , วงจรนับ BCD 4 หลัก, แรงดันอินพุท V_{in} 2 โวลท์, สัญญาณแรมป์มีความชัน 1 V/ms ดังแสดงในรูป 2.9

จากจุดเริ่มต้นจนถึงค่าแรงดันสูงสุด(2โวลท์) สัญญาณแรมป์ใช้เวลา 2ms หลังจากนี้สัญญาณนาฬิกาที่ส่งไปนั้นจะถูกปิด

ในช่วง 2ms นี้มีการส่งพัลส์ไปให้วงจรนับทำการนับ 200 ลูก เอาท์พุทของวงจรเปรียบเทียบที่มีระดับ "HIGH" ซึ่งเป็นการสั่งให้วงจรแลตช์ส่งค่าที่นับได้ไปยังภาคแสดงผลและเติมจุดทศนิยมที่ตำแหน่งที่เหมาะสมของตัวแสดงผลให้เป็นค่า 2,000 ที่แรงดันอินพุท 2 โวลท์

วงจรแบบนี้เป็นหลักการเบื้องต้นของดิจิตอลโวลท์มิเตอร์ ซึ่งถ้าใช้วงจรนับเลขฐาน สองแทนแบบ BCD เอาท์พุทก็จะอ่านได้เป็นค่าเลขฐานสองโดยตรง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 3
สัญญาณต่างๆบนสล๊อตไอซา(ISA)ของคอมพิวเตอร์

ภายในคอมพิวเตอร์ได้มีการออกแบบให้สามารถที่จะเพิ่มเติมวงจรอินเตอร์เฟสเข้าไปในภายหลังได้ โดยผ่านทางสล๊อตที่อยู่บนเมนบอร์ด(Main Board) ซึ่งตัวSlot ISAนี้จะมีขาใช้งานทั้งหมด 62 ขาแบ่งออกเป็น2ข้าง ข้างละ 31 ขา ส่วนการเรียกขาของสล๊อตไอซานี้จะขึ้นอยู่กับว่าขานั้นอยู่ข้างใด (ซ้ายหรือขวา) ของสล๊อต โดยขาที่อยู่ด้านซ้ายของสล๊อต จะเรียกโดยใช้ตัวอักษร B นำหน้าเลขตำแหน่งขา เช่น B16 ก็คือขาทางด้านซ้ายของสล๊อต ขาที่ 16 (นับจากทางด้านท้ายของเครื่อง) ส่วนขาที่อยู่ทางด้านขวาของสล๊อตจะเรียกโดยใช้อักษร A นำหน้าเลขตำแหน่งของขา เช่น ขา A24 ก็คือขาทางด้านขวาของสล๊อตขาที่ 24 (นับจากทางด้านท้ายของเครื่อง)

แต่ละขาของสล๊อตเหล่านี้จะเชื่อมต่อกับเส้นสัญญาณต่างๆบนเมนบอร์ด ทำให้การสร้างวงจรอินเตอร์เฟส กับคอมพิวเตอร์ สามารถทำได้โดยสะดวก ซึ่งเส้นสัญญาณที่เชื่อมต่อกับขาของสล๊อตเหล่านี้จะประกอบไปด้วย เส้นสัญญาณของบัสแอดเดรส(Address Bus), บัสข้อมูล(Data Bus), ตัวควบคุมสำหรับการเขียน/อ่านข้อมูลจากหน่วยความจำ หรือพอร์ต I/O, เส้นสัญญาณสำหรับการขออินเตอร์รัป, เส้นสัญญาณสำหรับการขอ DMA , สัญญาณฐานเวลา(Timing Signal) ต่างๆที่ใช้ในระบบ, เส้นสัญญาณแสดงการรีเฟรชหน่วยความจำ และสัญญาณสำหรับการตรวจสอบความผิดพลาด(I/O Check)

3.1 รายละเอียดเกี่ยวกับสัญญาณต่างๆ

OSC(Oscillator : ขา B30) : ขานี้เป็นเอาท์พุตที่เชื่อมต่อกับสัญญาณคล๊อกที่มีค่าความถี่สูงสุดบนบอร์ดคือ 14.31818 MHz ซึ่งมีคาบเวลาประมาณ 70 ns และมี Duty Cycle 6 ช่วงเวลาใน 1 คาบที่สัญญาณคล๊อกมีลอจิกเป็น "1" หารด้วยคาบเวลาทั้งหมด) ประมาณ 50% สัญญาณคล๊อกอื่นๆของระบบ เช่น คล๊อกที่ป้อนให้กับ 8088 หรือชิพซัพพอร์ทต่างๆนั้นจะถูกสร้างขึ้น โดยการหารสัญญาณนี้ อย่างไรก็ตามสิ่งหนึ่งที่จะต้องคำนึงถึงในการใช้งานสัญญาณ OSC ก็คือ สัญญาณนี้จะไม่ Synchronize กับสัญญาณอื่นๆบนบัสของระบบ ดังนั้นจึงไม่ควรที่จะนำสัญญาณจากขา OSC นี้ไปใช้เป็นสัญญาณคล๊อกสำหรับวงจรภายนอกอื่นๆที่ทำงานร่วมกับระบบ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

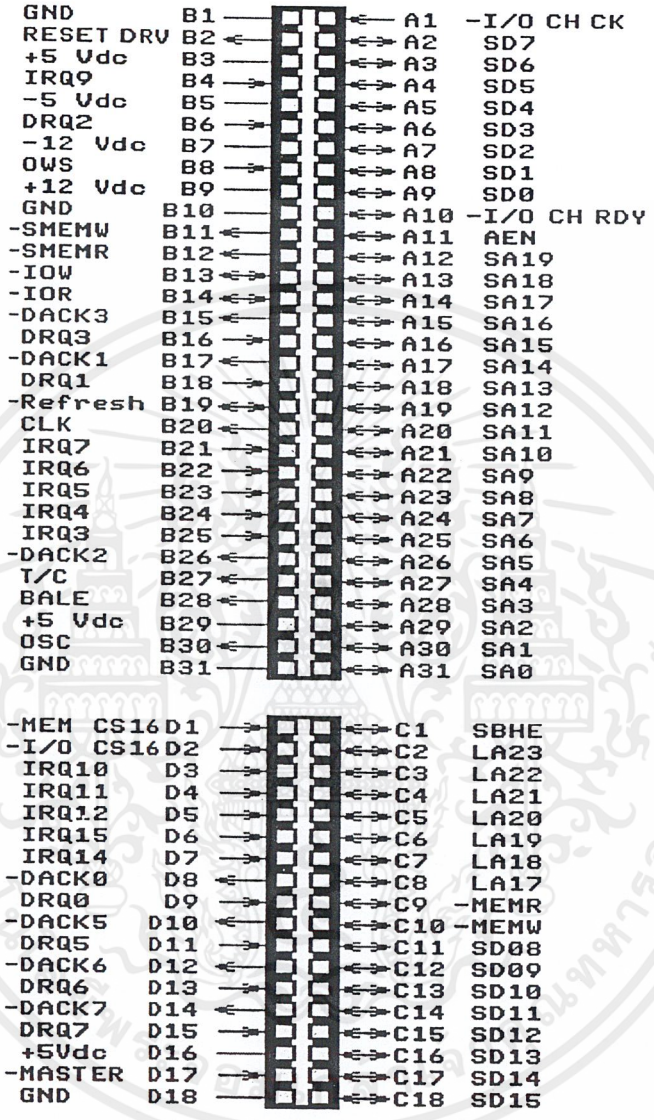
CLK(Clock : ขา B20) : ขาสัญญาณนี้เป็นเอาท์พุท ซึ่งต่อกับสัญญาณคล๊อกที่ถูกสร้างขึ้นโดยการหารสัญญาณ OSC ด้วย 3 ทำให้ได้ความถี่ประมาณ 4.77 MHz (14.31818 MHz/3) หรือ มีช่วงเวลาใน 1 คาบ(ช่วงเวลาของคล๊อก 1 ลูก)เท่ากับ 210 ns (1/4.77MHz) สำหรับค่า Duty Cycle ของสัญญาณนี้จะมีค่าประมาณ 1/3 คือ ใน 1 คาบจะมีช่วงเวลาที่เป็นลอจิก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

RESET DRV (ขา B2) : ขาสัญญาณนี้เป็นเอาท์พุท ซึ่งจะแอคทีฟ(ลอจิก "1") ในช่วงที่เราเริ่มจ่ายไฟให้กับระบบ และจะยังคงแอคทีฟไปจนกว่าระบบต่างๆ ภายในคอมพิวเตอร์ จะพร้อมที่จะทำงานได้จากนั้นสัญญาณนี้ก็จะเปลี่ยนกลับเป็นลอจิก "0" นอกจากนี้ในระหว่างการทำงานของคอมพิวเตอร์ถ้าระดับแรงดันของแหล่งจ่ายไฟลดลง สัญญาณนี้ก็จะถูกทำให้แอคทีฟเช่นกัน โดยทั่วไปแล้วสัญญาณนี้จะถูกนำไปใช้ในการรีเซตวงจรอินเตอร์เฟส หรืออุปกรณ์ I/O ต่างๆในช่วงที่เริ่มจ่ายไฟให้กับระบบซึ่งจะเป็นทำให้วงจร หรืออุปกรณ์เหล่านี้ถูกปรับให้อยู่ในสภาวะที่แน่นอนก่อนที่จะเริ่มต้นการทำงานในระบบ (สภาวะนี้เป็นสภาวะที่เราทราบ และต้องการให้วงจรทำงานในขณะที่ระบบถูกรีเซต)

A0-A19 (Address Bus : ขา A31-A12) : ขาสัญญาณทั้ง 20 ขานี้จะเป็นเอาท์พุท ซึ่งใช้สำหรับกำหนดแอดเดรสของหน่วยความจำ หรืออุปกรณ์ I/O ที่ 8088 ต้องการติดต่อด้วย โดยที่สัญญาณ A0 จะมีนัยสำคัญต่ำสุด(Least Significant Bit) และ A19 นี้จะมีนัยสำคัญสูงสุด (Most Significant Bit) สำหรับค่าแอดเดรสบนบัสแอดเดรส A0-A19 นี้จะถูกกำหนดโดย 8088 ในระหว่างขบวนการอ่าน/เขียนข้อมูลลงในหน่วยความจำหรืออุปกรณ์ I/O แต่ในช่วงของขบวนการ DMA นั้น DMA Controller จะเป็นผู้กำหนดค่าแอดเดรสบนบัสแอดเดรสเอง (ในระหว่างนี้ 8088 จะถูกตัดออกจากระบบ)

จะเห็นได้ว่าจำนวนเส้นแอดเดรสนี้จะมีอยู่ 20 เส้น ซึ่งสามารถที่จะอ้างแอดเดรสของหน่วยความจำได้ถึง 1 Mbyte แต่อย่างไรก็ตามจะมีแอดเดรสบางแอดเดรสที่ถูกใช้งานโดย คอมพิวเตอร์อยู่ก่อนแล้ว คือแอดเดรสของหน่วยความจำ RAM บนเมนบอร์ดที่ถูกใช้โดยระบบ จำนวน 64 Kbyte (สำหรับ คอมพิวเตอร์ จะเป็นจำนวน 256 Kbyte) และแอดเดรสสำหรับหน่วยความจำ ROM อีก 48 Kbyte ซึ่งถูกจัดในช่วงแอดเดรสบนสุดใน 1 Mbyte คือ 0FC00H จนถึง 0FFFFFH (สำหรับคอมพิวเตอร์จะเป็น 64 Kbyte)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.1 แผนผังขาต่อใช้งานของไอซาบัส

สำหรับการอ้างแอดเดรสของพอร์ท I/O นั้น จะใช้เส้นแอดเดรสเพียง 16 เส้น คือ A0-A15 ซึ่งจะทำให้อ้างแอดเดรสของพอร์ทได้ 64 K พอร์ท โดยผ่านทางชุดคำสั่ง IN และ OUT ส่วนเส้นแอดเดรสที่เหลือ คือ A16-A19 นั้นจะไม่ถูกใช้งาน อย่างไรก็ตามในคอมพิวเตอร์จะใช้เส้นแอดเดรสในการอ้างแอดเดรสของพอร์เพียง 10 เส้น คือ จาก A0-A9 และค่าแอดเดรสที่ใช้งานจะต้องอยู่ในช่วง 0200H จนถึง 03FFH เท่านั้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

D0-D7(Data Bus : ขา A9-A2) : ขาสัญญาณนี้จะเป็นแบบ Bi-Directional ซึ่งต่อกับบัสข้อมูลของระบบ เพื่อทำหน้าที่ในการส่งผ่านข้อมูลระหว่างพอร์ท I/O กับ คอมพิวเตอร์โดยบิท D0 จะมีนัยสำคัญต่ำสุดและบิท D7 จะมีนัยสำคัญสูงสุด

สำหรับบัสไซเคิลของการเขียนข้อมูลที่สร้างโดย 8088 นั้นข้อมูลจะถูกส่างออกมาบนบัสข้อมูลก่อนที่สัญญาณ IOW (ในกรณีที่ต้องการส่งข้อมูลให้กับพอร์ท) หรือ MEMW (ในกรณีที่ต้องการส่งข้อมูลให้กับหน่วยความจำ) จะเปลี่ยนจากลอจิก 0 เป็นลอจิก 1 (ขอบขาขึ้น) ซึ่งโดยทั่วไปขอบขาขึ้นของสัญญาณ IOW หรือ MEMW นี้ จะถูกใช้เพื่อสั่งให้พอร์ท I/O หรือหน่วยความจำที่มีแอดเดรสตรงกับค่าแอดเดรสบนบัสแอดเดรสนั้นรับข้อมูลไปเก็บไว้

สำหรับในบัสไซเคิลการอ่านข้อมูลที่สร้างขึ้นโดย 8088 นั้นพอร์ท I/O หรือหน่วยความจำที่ถูกอ้างถึงจะต้องส่งข้อมูลออกมาบนบัสข้อมูล ก่อนที่ IOR (ในกรณีที่ต้องการอ่านข้อมูลจากพอร์ท) หรือ MEMR (ในกรณีที่ต้องการอ่านข้อมูลจากหน่วยความจำ) จะเปลี่ยนจากลอจิก 0 เป็นลอจิก 1

ALE(Address Latch Enable : ขา B28) : ขาสัญญาณนี้เป็นขาเอาท์พุทที่ 8088 Bus Controller สร้างขึ้นเพื่อใช้สำหรับแสดงการเริ่มต้นของบัสไซเคิล และแสดงให้อุปกรณ์ภายนอกทราบว่าแอดเดรสที่ 8088 ต้องการจะติดต่อด้วยนั้นถูกส่งออกมาบนบัสแอดเดรสแล้ว โดยที่สัญญาณ ALE นี้จะเปลี่ยนจากลอจิก 1 เป็น 0 เมื่อค่าแอดเดรสที่ถูกต้องถูกส่งออกมาบนบัสข้อมูลเรียนร้อยแล้ว ดังนั้นขอบขาลงของสัญญาณ ALE นี้จะถูกใช้ในการแลทช์ค่าแอดเดรสจากบัสแอดเดรส/ข้อมูล Address/Data Bus : AD0-AD7 ของ 8088 ทำให้สามารถแยกค่าแอดเดรส (A0-A19) และข้อมูล(A0-A7) ออกจากกันได้ อย่างไรก็ตามสัญญาณ ALE จะแอคทีฟเฉพาะในบัสไซเคิลที่สร้างโดย 8088 เท่านั้น โดยจะไม่แอคทีฟในระหว่างขบวนการ DMA

I/O CHCK (I/O Channel Check : ขา A1) : ขาสัญญาณนี้เป็นอินพุทที่ใช้ในการแสดงความผิดพลาดเกี่ยวกับพาริตี้ ที่เกิดขึ้นในการทำงานของวงจรอินเตอร์เฟส หรืออุปกรณ์ I/O เมื่อขานี้ได้รับลอจิก 0 จะทำให้ 8088 ถูกอินเตอร์รัพท์แบบ Non-Maskable (NMI) อย่างไรก็ตามเราสามารถที่จะกำหนดให้วงจรภายในของคอมพิวเตอร์ทำการขออินเตอร์รัพท์ (เมื่อได้รับสัญญาณ I/O CHCK) หรือไม่ก็ได้ โดยการกำหนดลอจิกของบิทข้อมูลของพอร์ที่ควบคุมการขออินเตอร์รัพท์แบบ NMI คือ บิท D7 ของพอร์ท 00A0H ในกรณีที่บิท D7 ของพอร์ท 00A0H ถูกเซ็ทเป็น 1 ก็จะให้วงจรภายนอกขออินเตอร์รัพท์แบบ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

NMI ได้ (Enable) แต่ถ้าบิท D7 ของพอร์ท 00A0H ถูกเซ็ทเป็น 0 ก็จะทำการดิสเอเบิล(Disable) การขออินเตอร์รัพท์แบบ NMI ดังนี้

Enable : ใช้คำสั่ง OUT ส่งข้อมูล 80H ไปยังพอร์ท 00A0H

Disable : ใช้คำสั่ง OUT ส่งข้อมูล 00H ไปยังพอร์ท 00A0H

และเนื่องจากที่มีอุปกรณ์อื่นที่สามารถขออินเตอร์รัพท์แบบ NMI ได้อีก ดังนั้นซอฟท์แวร์ที่ใช้งานจะต้องสามารถตรวจสอบว่าการขออินเตอร์รัพท์นั้นเกิดขึ้นจากแหล่งใดได้ด้วย

I/O CHRDY (I/O Channel Ready : ขา A10) : ขาสัญญาณนี้เป็นอินพุทที่ใช้เพิ่มช่วงเวลาในบัสไซเคิลในกรณีที่อุปกรณ์ I/O หรือหน่วยความจำที่เกี่ยวข้องกับขบวนการในบัสไซเคิลที่เกิดขึ้นนั้น ไม่สามารถทำงานทันตามช่วงเวลาปกติของบัสไซเคิลนั้นๆได้ (ช่วงเวลาของบัสไซเคิลที่เกี่ยวกับหน่วยความจำใช้ช่วงเวลาเท่ากับช่วงเวลาคล๊อก 5 ลูก หรือ 1.05μS)

เมื่ออุปกรณ์ I/O หรือหน่ายความจำต้องการที่จะเพิ่มช่วงเวลาในบัสไซเคิลให้นานขึ้นอีกนั้น จะสามารถทำได้โดยการป้อนลอจิก 0 ให้กับขา I/O CHRDY ในช่วงที่ I/O หรือหน่วยวามจำที่ถูกกำหนดนั้น ได้รับสัญญาณจากการดีโค้ดแอดเดรส และสัญญาณ MEMR, MEMW, IOR หรือ IOW แอคทีฟ

IRQ2-IRQ7(Interrupt Request 2-7 : ขา B4 และ B25-B21) : ขาสัญญาณทั้ง 6 นี้เป็นขาอินพุทที่ใช้สำหรับการขออินเตอร์รัพท์จาก 8088 โดยสัญญาณเหล่านี้จะถูกต่อเข้ากับ 8259A บนเมนบอร์ดโดยตรง โปรแกรมในส่วน BIOS ของคอมพิวเตอร์ จะทำให้การโปรแกรม 8259A ให้ IRQ2 มีลำดับความสำคัญสูงสุด(Highest Priority) และ IRQ7 มีลำดับความสำคัญที่ต่ำสุด ในกรณีที่มีการขออินเตอร์รัพท์เกิดขึ้น คือระดับลอจิกที่ขา IRQ ขาใดขาหนึ่งถูกเปลี่ยนจากลอจิก 0 เป็นลอจิก 1 8259A ก็จะทำการส่งสัญญาณ INT ให้กับ 8088 เพื่อทำการขออินเตอร์รัพท์

สิ่งสำคัญในการขออินเตอร์รัพท์ผ่านทาง IRQ2-IRQ7 นี้ก็คืออุปกรณ์ที่จะทำการขออินเตอร์รัพท์โดยผ่านทาง IRQ ขาใดก็จะต้องรักษาระดับสัญญาณที่ขา IRQ นั้น ให้แอคทีฟอยู่จนกว่าจะได้รับสัญญาณ INTA (Interrupt Acknowledge) จาก 8088 เสียก่อน ถ้าไม่เช่นนั้นการขออินเตอร์รัพท์จะถูกยกเลิก และอินเตอร์รัพท์ Level 7 (IRQ7) ก็จะถูกสร้างขึ้นโดยอัตโนมัติไม่ว่าการขออินเตอร์รัพท์ที่ถูกยกเลิกนั้นจะเป็นการจออินเตอร์รัพท์ใน Level หรือ ขาใด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

แต่อย่างไรก็ตามสัญญาณ INTA นี้จะไม่ถูกต่อออกมาที่ขาของสล๊อตด้วยดังนั้นโปรแกรมที่มีการตอบสนองต่อการขออินเตอร์(Interrupt Service Routine) จะต้องทำการรีเซ็ตสัญญาณ IRQ เองโดยใช้คำสั่ง OUT ไปยังพอร์ท I/O ที่เกี่ยวข้อง

IOR(I/O READ : ขา B14) : ขาสัญญาณนี้เป็นขาเอาท์พุทแอคทีฟที่ลอจิก 0 ที่สร้างขึ้นโดย 8288 Bus Controller เพื่อใช้ในการแสดงว่าบัสไซเคิลที่เกิดขึ้นนี้เป็นบัสไซเคิลที่อ่านข้อมูลจากพอร์ท I/O เพื่อให้พอร์ท I/O ที่มีแอดเดรสที่ตรงกับที่แอดเดรสบนบัสแอดเดรสนั้นส่งข้อมูลออกมาบนบัสข้อมูลโดยข้อมูลจะต้องถูกส่งออกมาบนบัสข้อมูลก่อนขอบขาขึ้นของสัญญาณ IOR 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 ผ่านทางแชนเนลที่ 1 (DRQ1) เป็นต้น

IOW(I/O WRITE : ขา B13) : ขาสัญญาณนี้เป็นสัญญาณเอาท์พุทแอคทีฟที่ลอจิก "0" ซึ่งถูกสร้างขึ้นโดย 8288 Bus Controller เพื่อใช้แสดงว่าบัสไซเคิลที่เกิดขึ้นนี้เป็นบัสไซเคิลของการเขียนข้อมูลลงบนพอร์ท I/O ที่มีแอดเดรสตรงกับแอดเดรสบนบัสแอดเดรสนั้น รับข้อมูลที่อยู่บนบัสข้อมูลไปเก็บไว้อย่างไรก็ตามเนื่องมาจากในช่วงเวลาที่สัญญาณ IOW นี้แอคทีฟ (ลอจิก "0") นั้นข้อมูลบนบัสแอดเดรสอาจจะยังไม่สมบูรณ์ ดังนั้นในการออกแบบจึงควรใช้ขอบขาขึ้นของสัญญาณ IOW แทนขอบขาลงในการทำให้พอร์ท I/O ที่เกี่ยวข้องรับข้อมูลไปเก็บไว้เพื่อให้ข้อมูลบนบัสข้อมูลเสร็จเสียก่อน สำหรับในขบวนการ DMA นั้น DMA Controller จะทำการสร้างสัญญาณ IOW เอง โดยให้ค่าแอดเดรสที่อยู่บนบัสแอดเดรสจะเป็นค่าแอดเดรสของหน่วยความจำที่พอร์ท I/O ที่จขอ DMA ต้องการจะอ่านข้อมูล

MEMW (Memory Write : ขา B11) : ขานี้เป็นขาเอาท์พุทแอคทีฟที่ลอจิก "0" ซึ่ง 8288 Bus Controller สร้างขึ้นในระหว่างบัสไซเคิลในการเขียนข้อมูลงในหน่วยความจำของ 8088 สัญญาณ MEMW นี้จะถูกส่งออกมาเพื่อให้หน่วยความจำที่แอดเดรสตรงกับค่าแอดเดรสบนบัสแอดเดรสนั้น ทำ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สำนักหอสมุดกลาง พระจอมเกล้าลาดกระบัง

การรับข้อมูลที่อยู่บนบัสข้อมูลไปเก็บไว้ โดยทั่วไปหน่วยความจำจะรับข้อมูลในช่วงขอบขาขึ้นของสัญญาณ MEMW

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

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

สำหรับในระหว่างขบวนการ DMA นั้น DMA Controller จะควบคุมบัสต่างๆของระบบแทน 8088 และสัญญาณ MEMR จะถูกใช้ในบัสไซเคิลของการอ่านข้อมูลจากหน่วยความจำ (ข้อมูลถูกส่งจากหน่วยความจำไปให้กับอุปกรณ์ I/O)

DRQ1-DRQ3 (DMA Request 1-3 : ขา B18, B6 และ ขา B16) : ขาสัญญาณทั้งสามนี้เป็นสัญญาณอินพุทแอคทีฟที่ลอจิก "1" ซึ่งอุปกรณ์ภายนอกจะใช้ในการขอ DMA จากระบบ โดยการป้อนระดับสัญญาณลอจิก "1" ให้กับขา DRQ ขาใดขาหนึ่ง (ขา DRQ ทั้งสามนี้จะต่อเข้ากับ DRQ1-DRQ3 ของ 8237A-5)

เมื่อ 8237A-5 ได้รับสัญญาณนี้แล้วก็จะตรวจสอบว่ามีการของ 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 ของคอมพิวเตอร์ จะโปรแกรม 8237A-5 ให้ DRQ1 มีลำดับความสำคัญสูงสุดและ DRQ3 มีลำดับความสำคัญต่ำสุด ดังนั้นถ้ามีการขอ DMA ของอุปกรณ์ภายนอกผ่านทางแชนแนลที่ 1 ก่อน จากนั้นเมื่อเสร็จขบวนการ DMA ของแชนแนลที่ 1 แล้วจึงทำการขอ DMA ให้กับแชนแนลที่ 2

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

46232

อย่างไรก็ตาม 8237A-5 ยังมีแชนแนลสำหรับการขอ DMA อยู่อีก 1 แชนแนล คือแชนแนลที่ 0 (DRQ0) ซึ่งในความเป็นจริงแล้วแชนแนลนี้จะมีลำดับความสำคัญที่สูงกว่าแชนแนลที่ 1 แต่ไม่ถูกต่อออกมายังขาของสล๊อต เนื่องจากคอมพิวเตอร์จะใช้แชนแนลที่ 0 นี้ในการรีเฟรชหน่วยความจำที่เป็นไดนามิค RAM

ในการขอ DMA นั้นสัญญาณ DRQ นี้จะต้องแอคทีฟอยู่ในช่วงระยะเวลาหนึ่งเท่านั้น ถ้าสัญญาณนี้แอคทีฟอยู่นานเกินไป จะทำให้เกิดขบวนการขอ DMA ขึ้นมากกว่า 1 ขบวนการได้สำหรับวงจรที่ขอ DMA โดยทั่วไปจะใช้สัญญาณตอบรับการขอ DMA หรือสัญญาณ DACK ของแชนแนลที่ของ DMA นั้น ในการรีเซ็ทสัญญาณ DRQ เช่นอุปกรณ์ภายนอกที่ขอ DMA ผ่านทางแชนแนลที่ 1 (DRQ1) ก็จะคอยครวจสอบการตอบรับการขอ DMA จากสัญญาณ DACK ของแชนแนลที่ 1 (DACK1) เมื่อได้รับสัญญาณจาก DACK1 แล้วก็จะรีเซ็ทสัญญาณ DRQ1 (เปลี่ยนจากลอจิก "1" เป็น "0")

DACK0-DACK3(DMA Acknowledge 0-3 : ขา B19, B17, B26 และ B15) : สัญญาณทั้ง 4 นี้เป็นเอาท์พุทแอคทีฟที่ลอจิก "0" ซึ่ง 8237A-5 สร้างขึ้นเพื่อแสดงให้วงจรภายนอกที่ขอ DMA ทราบว่าการขอ DMA นั้นได้รับการตอบสนองแล้ว และ 8237A-5 จะเข้าสู่ขบวนการ DMA เพื่อให้การส่งผ่านข้อมูลระว่างอุปกรณ์ I/O ที่ขอ DMA กับหน่วยความจำเกิดขึ้นได้โดยตรง (คือไม่ต้องผ่าน 8088) โดยสัญญาณ DACK นี้จะแอคทีฟในแชนแนลใดก็ขึ้นอยู่กับว่าขบวนการ DMA ที่จะเกิดขึ้นนั้น เป็นการตอบสนองต่อการขอ DMA ในแชนแนลที่ 2 (DRQ2) สัญญาณ DACK2 ก็จะแอคทีฟ เป็นต้น

ดังที่ได้กล่าวแล้วว่าสัญญาณ DRQ0 นั้น จะไม่ถูกต่อออกมายังขาของสล๊อต ดังนั้นวงจรอินเตอร์เฟสจึงไม่สามารถจะขอ DMA ผ่านทางแชนแนล 0 ได้ แต่สัญญาณ DACK0 จะถูกต่อออกมายังสล๊อตด้วย (ขา B19) ทั้งนี้ก็เพื่อที่จะแสดงให้วงจรอินเตอร์เฟสต่างๆทราบว่าขบวนการ DMA เป็น Dynamic RAM ซึ่งวงจรอินเตอร์เฟสที่ใช้หน่วยความจำประเภทนี้สามารถจะนำไปใช้ในการรีเฟรช Dynamic RAM ที่อยู่ในวงจรได้

โดยที่การรีเฟรชหน่วยความจำนั้นจะต้องเกิดขึ้นในทุกๆ 15.12 μs หรือทุกๆ 72 คล๊อกดังนั้นสัญญาณ DACK0 นี้ก็จะแอคทีฟในทุกๆ 15.12 μs ด้วย

AEN (Address Enable : ขา A11) : สัญญาณนี้เป็นเอาท์พุทที่ใช้ในการแสดงว่าบัสไซเคิลที่เกิดขึ้นในช่วงเวลาที่สัญญาณ AEN แอคทีฟ (ลอจิก "1") นั้น เป็นบัสไซเคิลของขบวนการ DMA

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สำหรับเมนบอร์ดของคอมพิวเตอร์นั้น จะใช้สัญญาณนี้ในการดิสเอเบิล (Disable) 8288 Bus Controller และจะใช้ดิสเอเบิลพอร์ท I/O ต่างๆ ที่ไม่เกี่ยวข้องกับขบวนการ DMA ที่เกิดขึ้นนี้ ที่จำเป็นต้องทำเช่นนี้ก็เพราะในระหว่างขบวนการ DMA นั้น 8237A-5 จะส่งแอดเดรสของหน่วยความจำออกมาบนบัสแอดเดรส และจะทำให้สัญญาณ IOR หรือ IOW แอคทีฟด้วยดังนี้ถ้าไม่ทำการดิสเอเบิลพอร์ท 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 ที่ทำการส่งผ่านข้อมูลเป็นบล๊อก เนื่องจากสัญญาณนี้จะแอคทีฟโดยไม่แสดงว่าเป็นของแชนแนลใด ดังนั้นจึงต้องทำการนำสัญญาณผ่านเกท Inverter แล้วนำไป OR กับสัญญาณ DACK เพื่อให้สามารถทราบได้ว่า สัญญาณ T/C ที่เกิดขึ้นนั้นเป็นสัญญาณของแชนแนลใด สำหรับในแชนแนลที่ 0 นั้น สัญญาณ T/C จะแอคทีฟในช่วงเวลาที่คงคือ ทุกๆ 990.804 millisec ซึ่งก็คือช่วงเวลาในการรีเฟรชหน่วยความจำขนาด 64 Kbyte นั่นเอง

3.2 บัสของแหล่งจ่ายไฟของระบบ

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

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

-5VDC (ขา B5) : ขาทั้งนี้ต่อกับแหล่งจ่ายไฟ DC -5V ของระบบ โดยจะมีค่าความเที่ยงตรง (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

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

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

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 4
เอฟพีจีเอ
(FPGA : Field Programable Gate Arrays)

4.1 บทนำ

ความก้าวหน้าของเทคโนโลยีอิเล็กทรอนิกส์ในปัจจุบัน ทำให้เกิดการพัฒนาความสามารถของอุปกรณ์อิเล็กทรอนิกส์ต่างๆมากมาย ซึ่งทำให้อุปกรณ์อิเล็กทรอนิกส์ มีขนาดที่เล็กลง ใช้งานง่ายขึ้น และราคาถูกลง ทำให้เกิดการลดค่าใช้จ่ายและใช้พลังงานน้อยลง ในขณะเดียวกันก็ได้มีการเพิ่มประสิทธิภาพและระดับความเชื่อถือได้ของวงจรรวมที่สูงขึ้น เห็นได้ชัดจากเทคโนโลยีไมโครเซสเซอร์และหน่วยความจำในปัจจุบัน ในการพัฒนาวงจรอิเล็กทรอนิกส์เราจะพิจารณาออกเป็น 2 ส่วน

ส่วนที่ 1 เป็นการพัฒนาทางด้านวงจรอนาลอก ซึ่งการพัฒนาในส่วนนี้เมื่อเราออกแบบวงจรออกมาได้แล้ว จำเป็นจะต้องนำไป Fabrication ซึ่งจะต้องทราบเทคโนโลยีที่จะใช้ในการสร้าง มีค่าใช้จ่ายสูงและใช้ระยะเวลานาน

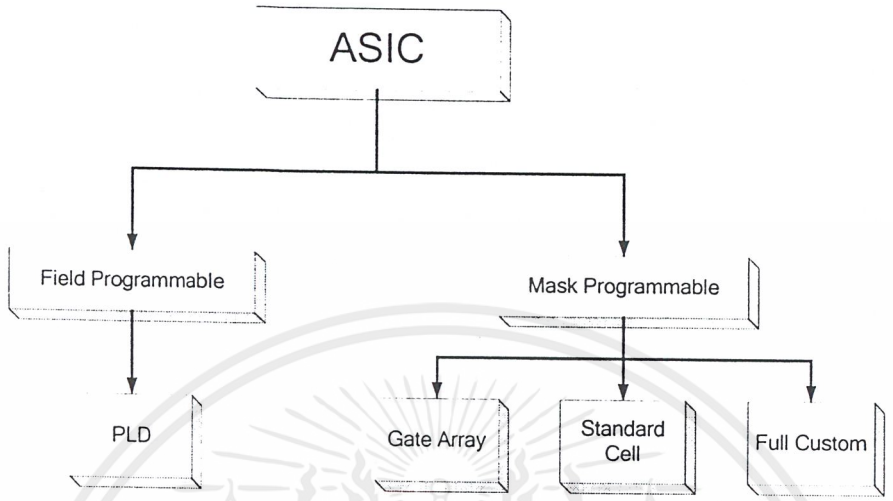
ส่วนที่ 2 เป็นการพัฒนาทางด้านวงจรดิจิตอล ในปัจจุบันนี้เทคโนโลยีการผลิตอุปกรณ์อิเล็กทรอนิกส์ ได้มีความก้าวหน้ามากขึ้นซึ้งเมื่อเราออกแบบวงจรทางดิจิตอลแล้วเราสามารถที่จะ Implement ลงบนซิฟไอซีได้เลยซึ้งจะกล่าวต่อไป

ในการพัฒนาวงจรดิจิตอลแบบเก่า ซึ่งใช้ดิจิตอลไอซีมาต่อวงจร จะเห็นว่ามีความยุ่งยากมากทั้งขนาดและการทดสอบความถูกต้องในการทำงาน ทำให้เกิดช่องว่างระหว่างไอซีมาตรฐาน และวงจรรวมมากยิ่งขึ้น ในการพัฒนาวงจรรวมทางดิจิตอลได้เพิ่มความหนาแน่นและจำนวน ฟังก์ชั่นลอจิก (Function Logic) ที่เหมาะสม นักออกแบบอุปกรณ์ทางด้านดิจิตอลได้พิจารณาถึงการผลิตให้มีปริมาณมากๆ และในการผลิตวงจรรวมเฉพาะงาน (ASIC: Application Specific Integrated Circuit) จะแบ่งตามการสร้างออกเป็น 2 กลุ่ม คือ

1. Mask Programmable
2. Field Programmable

ดังแสดงในรูปที่ 4.1

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.1 แสดงแผนผังการแบ่งกลุ่มของวงจรรวม ASIC

1. Mask Programmable

การใช้งานวงจรรวมเฉพาะงาน ASIC ในเชิงพาณิชย์ จำเป็นต้องใช้วงจรรวมเฉพาะงาน ASIC แบบ Mask programmable เนื่องจากต้นทุนต่อเนื่องต่อหนึ่งตัวจะต่ำกว่าวงจรแบบ Field programmable ASIC.ในกรณีที่ปริมาณการผลิตสูงนับพันนับหมื่นตัวขึ้นไป ตัวอย่างเช่น วงจร CPLD ตัวหนึ่งอาจสูงถึงหนึ่งพันบาท ในขณะที่ถ้าผลิตวงจรรวมที่มีคุณสมบัติเหมือนกันทุกประการโดยใช้ Mask programmable แล้ว ราคาตัวหนึ่งจะลดลงเหลือเพียงไม่ถึงหนึ่งร้อยบาท การใช้วงจรรวมแบบ Mask programmable จึงมีบทบาทสำคัญในการผลิตสินค้าอิเล็กทรานนิกส์ในเชิงพาณิชย์ในปัจจุบัน

วงจรรวมเฉพาะงานประเภทนี้ หลังจากผู้ใช้ออกแบบวงจรและตรวจสอบการทำงานจนเป็นที่น่าพอใจแล้ว ต้องส่งให้ผู้ผลิตทำการเจือสาร ไม่สามารถโปรแกรมได้ด้วยตนเองเหมือนกับวงจรรวมเฉพาะงานแบบ Field programmable ช่วงเวลาการผลิตออกใช้งานจึงใช้เวลานับเดือนและมีค่าใช้จ่ายเบื้องต้นในการเจือสารสูง วงจรรวมเฉพาะงานแบบ Mask Programmable ASIC ในปัจจุบันได้แก่ เกตอาเรย์ , เซลล์มาตรฐานและฟูลัสตัม (full custom)

2. Field Programmable

อุปกรณ์วงจรรวมเฉพาะงาน ASIC แบบ field programmable มีอยู่มากมายหลายชนิด แต่มีลักษณะการสร้างหรือกำหนดการทำงานของวงจรที่เหมือนกัน กล่าวคือ ผู้ใช้งานสามารถออกแบบและสร้างวงจรที่ต้องการใช้ลงในตัวอุปกรณ์ได้เองโดยไม่ต้องไปโรงงานเพื่อผลิต โดยเฉพาะอย่างยิ่งใน

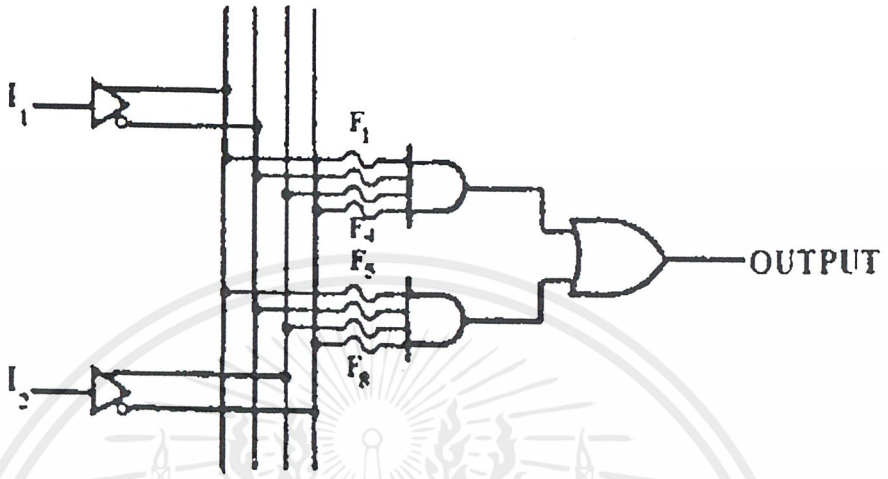
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ปัจจุบันนี้มีเครื่องมือที่ใช้ช่วยในการออกแบบ และสร้างวงจรร่วมกับไมโครคอมพิวเตอร์ที่มีความสามารถสูงในการพัฒนาตั้งแต่ขั้นการออกแบบ การจำลองการทำงาน จนถึงจัดสร้างวงจรลงในอุปกรณ์ รวมทั้งอุปกรณ์ Field Programmable เหล่านี้สามารถหาซื้อได้ง่ายทำให้การสร้างวงจรอิเล็กทรอนิกส์จนถึงระบบไมโครโพรเซสเซอร์หันมาใช้อุปกรณ์จำพวกนี้ เป็นอุปกรณ์ประกอบในวงจรแทนอุปกรณ์ย่อยๆ แยกชิ้น (Discrete component)

2.1 พีแอลดี (PLD: Programmable Logic Device)

ภายในอุปกรณ์พีแอลดีถูกเตรียมเป็นวงจรพื้นฐาน ทางด้านลอจิกต่อกันอยู่เป็นกลุ่มมีทั้งวงจรคอมไบเนชั่น (Combination) และซีเควลเชียล (Sequential) ซึ่งมีส่วนประกอบเป็นวงจรภายในเทคโนโลยีของวงจรที่ใช้สร้างพีแอลดีมีทั้ง ทีทีแอล (TTL) อีซีแอล (ECL) และ ซีมอส (CMOS) ตามความเหมาะสมของแต่ละระบบ อุปกรณ์พีแอลดีทุกชนิดมีหลักการพื้นฐานของวงจรภายในที่เหมือนกัน โดยมีวงจรคอมไบเนชั่นที่ให้ผลเป็นผลคูณร่วมบวก (Sum of product) ประกอบไปด้วยชุดของแอนด์เกตที่ต่อร่วมกับออเกต การโปรแกรมคือการเลือกว่าจะให้มีการต่ออินพุตภายในของแอนด์เกตกับสัญญาณอินพุตใดบ้าง ซึ่งมีทั้งจากภายนอกและสัญญาณป้อนกลับจากเอาต์พุตภายในเอง การติดต่ออินพุตของออเกตกับเอาต์พุตของ แอนด์เกต ตัวต่างๆ วิธีการเลือกหรือการโปรแกรมทางกายภาพ อินพุตต่างๆของอุปกรณ์ทุกตัวจะถูกต่อผ่านฟิวส์เข้ากับแหล่งสัญญาณ ซึ่งถ้าไม่ต้องการใช้สัญญาณใดจะตัดฟิวส์ทำให้สามารถโปรแกรมได้ครั้งเดียว อุปกรณ์พีแอลดีบางชนิดใช้มอสทรานซิสเตอร์แทนฟิวส์ทำให้สามารถโปรแกรมโดยใช้กระแสไฟฟ้า และสามารถลบและโปรแกรมใหม่เข้าไปได้อีก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

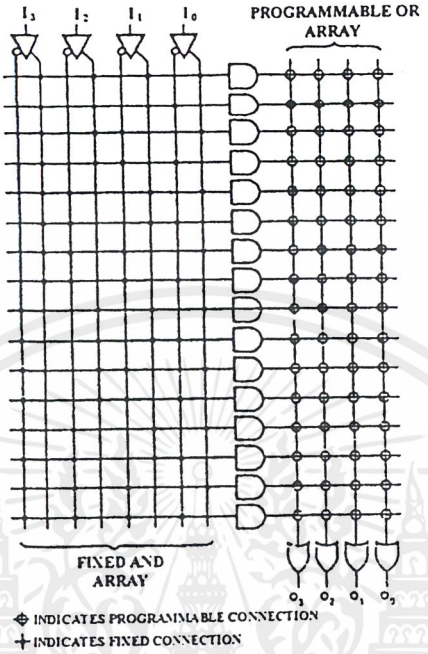


รูปที่ 4.2 แสดงวงจรพื้นฐานของอุปกรณ์พีแอลดีซึ่งอยู่ในรูปผลคูณร่วมบวก

2.2 พรอม (PROM: Programmable Read Only Memory)

พรอมคือหน่วยความจำรอม (ROM) โปรแกรมได้ ซึ่งนับว่าเป็นอุปกรณ์พีแอลดี ชนิดหนึ่งซึ่งวงจรภายในของพรอมเสมือนกับประกอบไปด้วย แถวลำดับของแอนด์และออเกต (And/Or Array) ผลเอาต์พุตที่ขาดาต้าเอาต์พุตสามารถแสดงในสมการของฟังก์ชั่นผลคูณร่วมบวก (Sum of product) ของสัญญาณอินพุตที่ขาแอดเดรส รูปที่ 4.3 แสดงถึงลักษณะการต่อเป็นแถวลำดับของแอนด์เกตและออเกตของพรอมขนาด 16 x 4 บิต วงจรทางด้านซ้ายบนสุดเป็น แอนด์เกตที่ให้ผลเป็นผลคูณ (Product) ของกรณีอินพุตเป็น 0000 แอนด์เกตที่อยู่ถัดลงมาเป็นผลคูณของกรณีที่อินพุตเป็น 0001, 0010, ... จนถึงตัวล่างสุดคือผลคูณในกรณีที่อินพุตเป็น 1111 ที่เอาต์พุตแต่ละบิตของหน่วยความจำ สามารถเลือกได้ว่าจะให้เป็น 1 ในกรณีที่อินพุตจากแอดเดรส เป็นอย่างไรบ้างเหมือนกันเป็นจากนำเอาต์พุตจากผลคูณที่ต้องการให้เอาต์พุตแต่ละบิตเป็น 1 ไปออกันจึงเปรียบเหมือนกับว่าในพรอมมีจำนวนแอนด์เกตเท่ากับจำนวนตำแหน่งความจำ และมีออเกตจำนวนเท่ากับจำนวนบิตของสัญญาณข้อมูลออก (Data output) อินพุตออเกตทุกตัวสามารถต่อเข้ากับแอนด์เกตตัวใดก็ได้ทุกตัว ซึ่งอาจเรียกได้ว่าเป็นพีแอลดีแบบ fixed AND / programmable OR

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

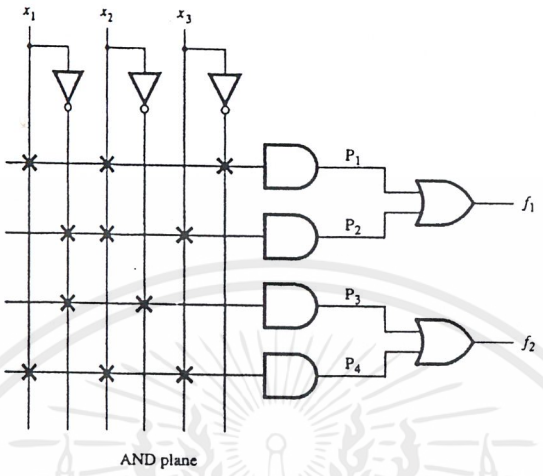


รูปที่ 4.3 แสดงลักษณะของพรอมเมื่อเปรียบเทียบเป็นวงจรในรูปผลคูณร่วมบวก

2.3 พีเอแอล (PAL: Programmable Array Logic)

ในช่วงกลางปี ค.ศ. 1970 บริษัทเอ็มเอ็มไอ (MMI : Monolithic Memory) ในประเทศสหรัฐอเมริกา ได้พัฒนาอุปกรณ์พีเอแอล เป็นพีเอแอลชนิดใหม่โดยใช้เทคโนโลยีแบบแอลเอสไอ (LSI : Large Scale Integration) สามารถโปรแกรมเลือกวงจรภายใน โดยใช้ฟิวส์ที่เชื่อมต่ออยู่ระหว่างสัญญาณอินพุตภายนอก และการป้อนกลับจากภายในกับแอนด์เกตที่ต่อเป็นฟังก์ชั่นผลคูณ (Product) อยู่ในตัววงจรรวม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

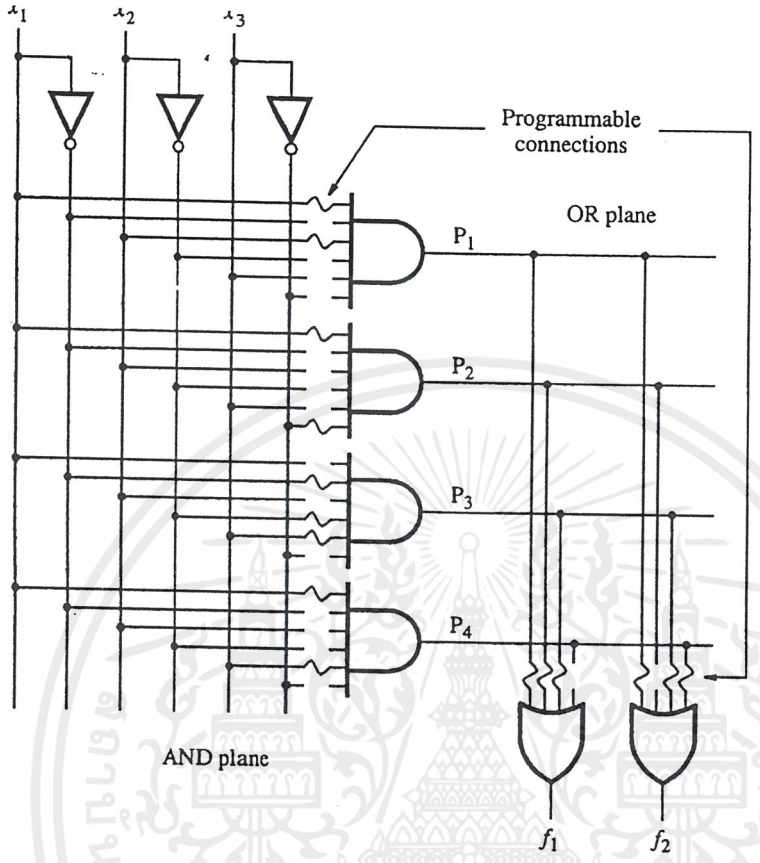


รูปที่ 4.4 แสดงโครงสร้างภายในของพีเอแอล

2.4 พีแอลเอ (PLA: Programmable Logic Array)

อุปกรณ์ที่สามารถโปรแกรมได้แบบพีแอลเอเกิดขึ้นเมื่อปี ค.ศ. 1975 โดยบริษัท ซิกเนทริกส์ (Signetics) สหรัฐอเมริกา ซึ่งเป็นบริษัทผู้ผลิตวงจรรวมรายใหญ่รายหนึ่ง ผลิตและนำเสนออุปกรณ์โดยใช้ชื่อว่า เอฟพีแอลเอ (FPLA : Field Programmable Logic Array) สามารถโปรแกรมการต่อลอจิกทั้งทางด้านแอนด์เกตและออเกตได้ และยังเลือกเอาต์พุตเป็น active high หรือ active low โดยต่อผ่านเอ็กคูลสีบออเกต ให้ทำหน้าที่เป็นนอนอินเวอร์เตอร์หรือเป็น อินเวอร์เตอร์แล้วแต่ภายในของพีแอลเอ ต่อมาปี ค.ศ. 1979 บริษัทซิกเนทริกส์ ได้สร้างเอฟพีแอลเอใหม่ที่มีเรจิสเตอร์ต่ออยู่ภายในวงจร เพิ่มขึ้น รวมทั้งสามารถเลือกสัญญาณอินพุตที่มาจากการป้อนกลับจากเรจิสเตอร์ได้ด้วย ทำให้สามารถใช้อุปกรณ์พีแอลเอใหม่นี้สร้างวงจร State machine ได้ อุปกรณ์ใหม่ที่มีเรจิสเตอร์ อยู่ด้วยนี้ถูกตั้งชื่อใหม่เป็น เอฟพีแอลเอส (FPLS: Field Programmable Logic Sepucncer) มีทั้งที่เป็นทีทีแอลและซีมอส

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.5 แสดงวงจรพื้นฐานภายในของพีแอลเอ

2.5 FPGA (Field Programmable Gate Array)

FPGA เป็นชิพในลักษณะของ Programmable Device ที่เราสามารถโปรแกรมตัวมันให้สามารถเป็นวงจรดิจิตอลใดๆก็ได้ โครงสร้างข้างในประกอบด้วย อาร์เรย์ของลอจิกเกทต่างๆมากมาย ซึ่งในปัจจุบันความจุเกทภายในตัวชิพ FPGA ได้เพิ่มขึ้นในระดับไม่กี่พันตัว จนถึงล้านตัว ดังนั้นจึงรองรับวงจรดิจิตอลที่มีความสลับซับซ้อนเป็นอย่างมาก และรวมถึงความเร็วในการทำงานที่ได้เปรียบอย่างมากมายเมื่อเปรียบเทียบกับไมโครโปรเซสเซอร์ นอกจากนี้ในด้านการออกแบบพัฒนาและทดสอบก็ทำได้ง่าย เนื่องจากเราสามารถโปรแกรมได้เองและสามารถโปรแกรมหลายๆครั้ง จึงไม่มีความเสี่ยงใดๆทั้งสิ้น และในปัจจุบันการออกแบบที่เป็น FPGA Base IC Design นั้นก็เป็นที่นิยมทำกันและมีแนวโน้มที่จะนำมาใช้งานมากขึ้นเรื่อยๆ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.2 เทคโนโลยีของ FPGA

เนื่องจากเป็นลักษณะของชิพที่สามารถโปรแกรมได้นั่นก็คือ สามารถกำหนดจุดเชื่อมต่อต่างๆ ภายในได้ เพื่อประกอบเป็นลักษณะของวงจรตามที่เราต้องการได้ ซึ่งเราสามารถแบ่งลักษณะของจุดเชื่อมต่อต่างๆได้ดังนี้

1. Physical Changing

1.1 Fused สามารถโปรแกรมได้เพียงครั้งเดียว หลังจากโปรแกรมจุดเชื่อมต่อขาดจากกัน

1.2 Anty Fuse สามารถโปรแกรมได้เพียงครั้งเดียว หลังจากโปรแกรม จุดเชื่อมต่อจะเชื่อมถึงกัน

2. Memory Base

2.1 EEPROM – Base FPGA

มักเรียก FPGA ประเภทนี้ว่า CPLD จะใช้เทคโนโลยีเหมือนกับ EEPROM ในการโปรแกรม ซึ่งจะทำให้มีความจุของเกตต่ำ โดยทั่วๆไปจะน้อยกว่า 20,000 เกต แต่ข้อดีของ EEPROM-Base FPGA คือสามารถเก็บข้อมูลที่โปรแกรมลงไปได้โดยไม่จำเป็นต้องมีไฟเลี้ยง และในการโปรแกรม จะใช้ทรานซิสเตอร์ 1 ตัวต่อ 1 บิท สามารถโปรแกรมได้ประมาณ 10,000 ครั้ง มักจะมีการจัดสถาปัตยกรรมในรูปแบบอาร์เรย์ ใช้ AND- OR Plane ในการทำลอจิกฟังก์ชัน

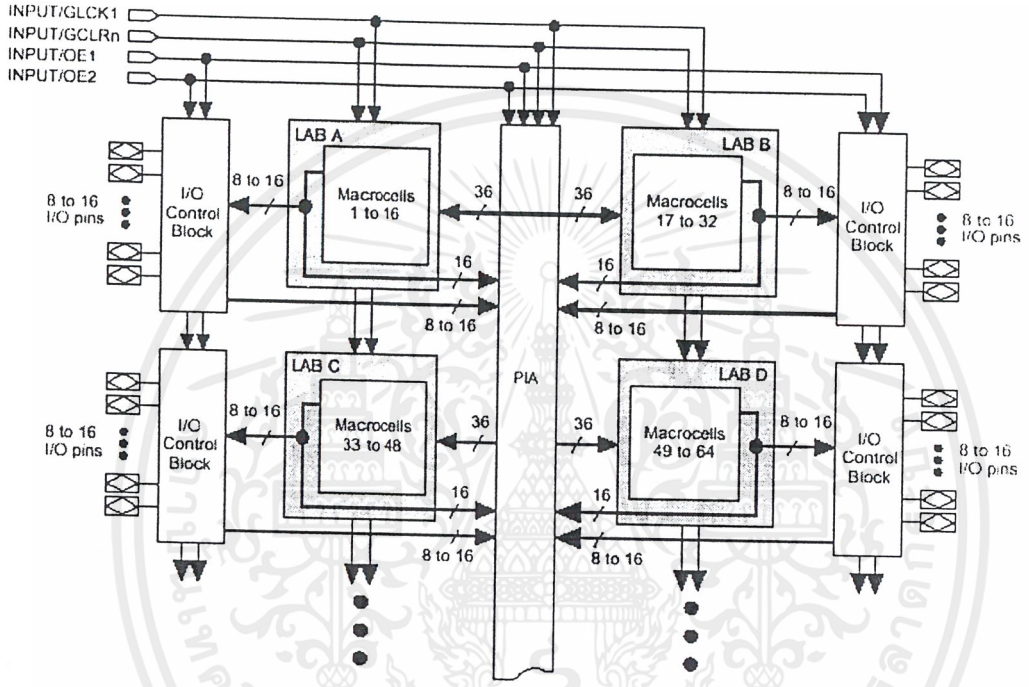
2.2 SRAM - Base FPGA

จะใช้เทคโนโลยีเหมือน SRAM ในการโปรแกรม ซึ่งจะสามารถทำให้โปรแกรมซ้ำได้ไม่จำกัดจำนวนครั้ง มีความจุของเกตปานกลางถึงสูงมาก (ประมาณ 10,000 – 1,000,000 เกต) จะใช้ Look-Up Table ในการทำลอจิกฟังก์ชัน (Logic Function) และจะมีการจัดทรัพยากรภายในโครงสร้างแบบอาร์เรย์ ข้อดีของ SRAM - Base FPGA คือจะใช้เวลาในการโปรแกรมน้อย (ในระดับ mS) การโปรแกรมจะทำได้ง่ายเทียบเท่ากับการเขียน SRAM ทั่วไป และไม่จำกัดจำนวนครั้งในกระบวนการผลิตจะทำได้ง่าย และเหมาะสมสำหรับการออกแบบวงจรที่มีความสลับซับซ้อน ข้อเสียก็คือไม่สามารถเก็บโปรแกรมในภาวะที่ไม่มีไฟเลี้ยงได้ มักจะใช้ FPGA ชนิดนี้ควบคู่กับ ROM เพื่อเก็บโปรแกรมและจะโหลดโปรแกรมเข้าในตัวชิพเมื่อเริ่มต้นใช้งาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

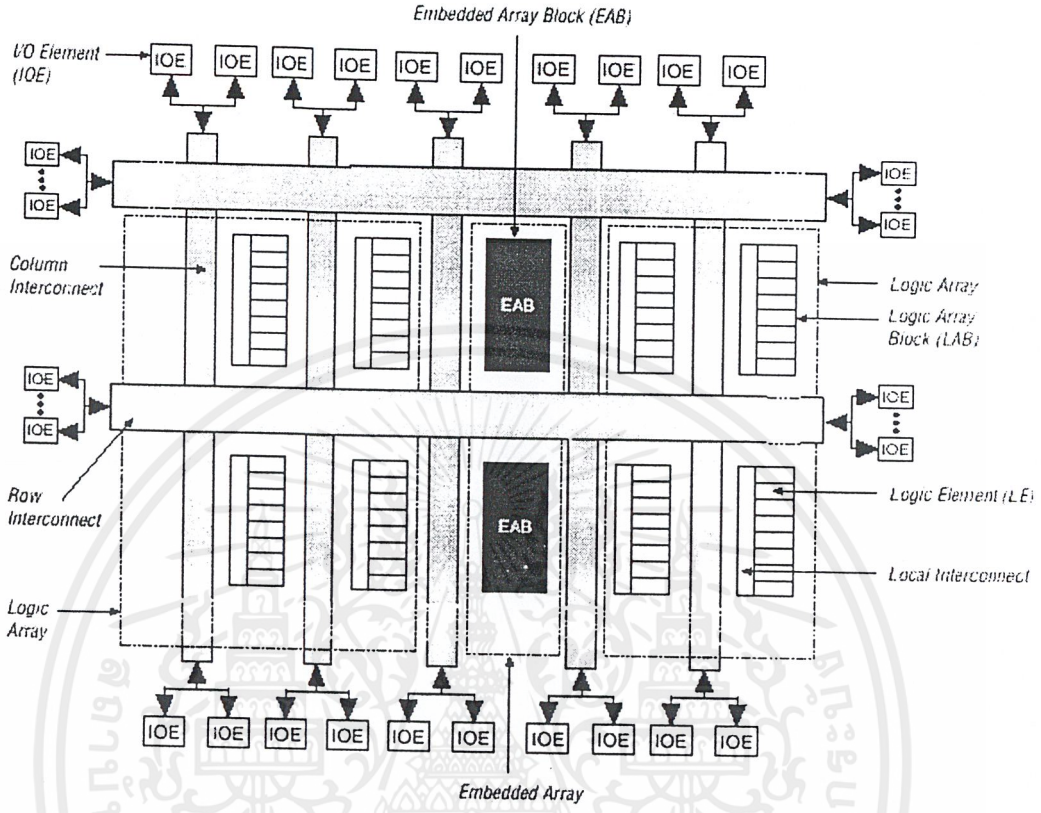
โครงสร้างภายใน

ลักษณะโครงสร้างภายใน เป็นอาร์เรย์ลอจิกบ๊อกซ์ที่สามารถทำการโปรแกรมได้ดังแสดงในรูปที่ 4.6



รูปที่ 4.6กโครงสร้างภายในของ FPGA ตระกูล MAX7000S

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.6ข โครงสร้างภายในของ FPGA ตระกูล FLEK10K

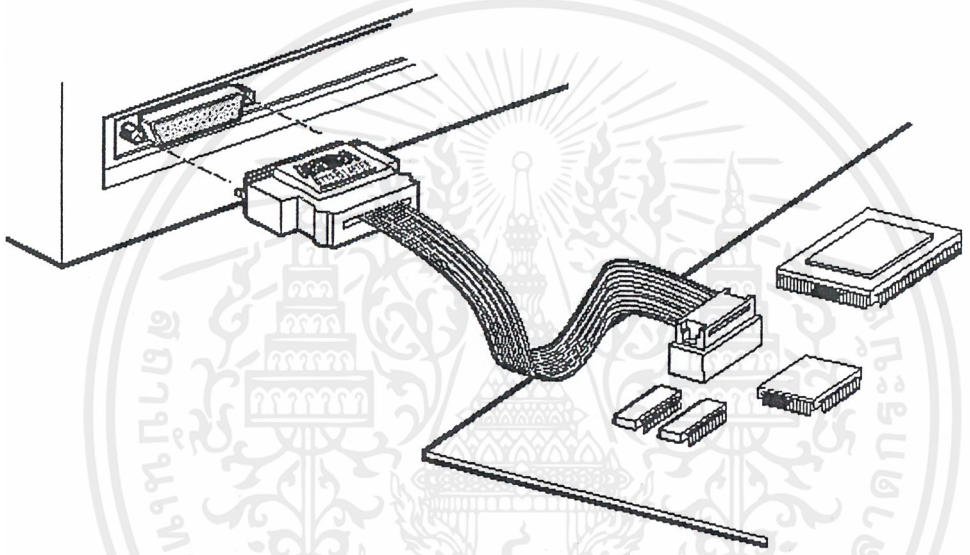
4.3 ทำไมการออกแบบถึงทำได้ง่ายและสะดวกรวดเร็ว

1. ในการออกแบบเราไม่จำเป็นต้องรู้ถึงโครงสร้างภายในของตัวชิพ เพียงแต่รู้ขั้นตอนการออกแบบลอจิกก็พอ ไม่เหมือนไมโครโปรเซสเซอร์ที่เราจำเป็นต้องรู้โครงสร้างภายในรวมถึงการศึกษาการเขียนภาษา Assembly ซึ่งแต่ละตัวก็ไม่เหมือนกันด้วย

2. การใช้ภาษาในการอธิบายการทำงานของวงจร ที่เรียกว่า **HDL (Hardware Description Language)** จะช่วยได้มากสำหรับการออกแบบ เนื่องจากเป็นวิธีการที่มีความยืดหยุ่นสูง ทำได้เร็ว และไม่จำเป็นต้องรู้ลักษณะของวงจรที่จะออกแบบว่าต่อกันอย่างไร เพียงแต่กำหนดลักษณะการทำงานให้มัน และตัวซอฟท์แวร์จะทำ Synthesis and Optimize ให้เราเอง นอกจากนี้ภาษาทีใช้ ยังเป็นมาตราฐานเดียวกัน สามารถใช้ได้กับชิพททุกตัวและทุกบริษัท

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3. การโปรแกรมสามารถทำได้เองและใช้เวลาไม่นาน โดยเพียงแค่ส่งข้อมูลผ่านสายดาวโหลดทางพอร์ตของคอมพิวเตอร์ก็สามารถโปรแกรมตัวชิพได้ขณะที่เอยู่ในระบบ โดยไม่จำเป็นต้องถอดมาโปรแกรมข้างนอก ดังรูปที่ 4.7 และที่สำคัญสามารถโปรแกรมได้หลายครั้ง จึงทำให้ง่ายในการแก้ไขและพัฒนาโดยที่ไม่ต้องเสียค่าใช้จ่ายเพิ่มแต่อย่างใด



รูปที่ 4.7 การโปรแกรมลงในชิพ

4.4 การออกแบบโดยใช้ภาษาอธิบายพฤติกรรมของฮาร์ดแวร์ (HDL)

ในการออกแบบวงจรดิจิตอลนั้นสามรถทำได้โดยการวาดวงจร (Schematic) หรือใช้ภาษาอธิบายพฤติกรรม (Hardware Description Language) ของฮาร์ดแวร์ ในกรณีของการออกแบบวงจรด้วย ASIC เราจะต้องเขียนวงจรด้วย Schematic แล้วนำวงจรนั้นไป Simulate หากผลออกมาเป็นที่พอใจก็จะต้อง Layout เป็นชั้นสาร และในการออกแบบ ASIC จำเป็นจะต้องรู้ว่าจะใช้เทคโนโลยีอะไรเช่นที่ NECTEC ใช้อยู่ จะใช้เทคโนโลยีของ ALCATEL 0.5 um. เมื่อได้ชั้น Layout เสร็จสมบูรณ์ จึงจะส่งไป Fabrication เป็นชิพไอซี แล้วจึงจะนำมาใช้งานได้ แต่ในการออกแบบวงจรด้วย FPGA โดยการใช้ Schematic หรือใช้ภาษาอธิบายการทำงานของวงจรจะทำได้สะดวกกว่า เพราะการทำวิธีนี้ ผู้ออกแบบไม่ต้องคำนึงถึงเทคโนโลยีที่จะใช้และที่สำคัญ การออกแบบโดยวิธีนี้สามารถแก้ไขโมเดล (Model)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หรือเปลี่ยนแปลงเทคโนโลยีได้สะดวกกว่า เพราะไม่ต้องวาดวงจรใหม่ นั่นคือการออกแบบโดยใช้ภาษาอธิบายฮาร์ดแวร์ จะทำให้โมเดลที่ได้ไม่ขึ้นกับเทคโนโลยี สำหรับภาษาที่ใช้สำหรับอธิบายพฤติกรรมของฮาร์ดแวร์ที่ใช้กันก็มี VHDL, AHDL, Verilogสำหรับในการทดลองนี้จะให้นักศึกษาใช้การวาด **Schematic** ส่วน VHDL นั้นจะมีใน Lab ต่อไป

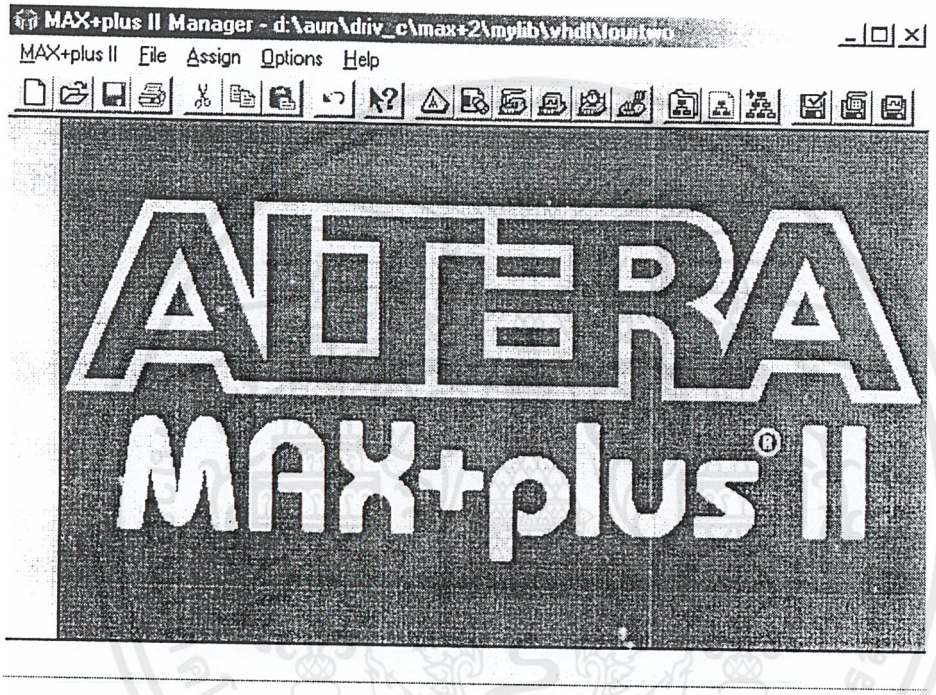
4.5 การสังเคราะห์วงจร (Logic Synthesis)

ในขั้นตอนนี้จะใช้ซอร์พแวร์ในการสังเคราะห์วงจร (Synthesis Tools) ทำการสังเคราะห์พฤติกรรมของวงจรที่ได้จากการออกแบบด้วย Schematic หรือ VHDL ซึ่งต้องทำการตรวจสอบด้วยว่าซอร์พแวร์นั้นสนับสนุนเทคโนโลยี FPGA (FPGA Library) ที่ต้องการหรือไม่ ตัวอย่างเช่น FPGA ของบริษัท XILINX และบริษัท ALTERA จะมีซอร์พแวร์หลายตัวที่สามารถใช้ได้ เช่น **Max+Plus II** ในขั้นตอนนี้ซอร์พแวร์สังเคราะห์วงจรจะทำการแปลงโค้ด VHDL และทำการ Optimize เพื่อให้ได้วงจรตามเทคโนโลยีที่เลือกใช้ ในการสังเคราะห์วงจรนั้นวงจรระดับเกต (Gate Level) จะไม่เหมาะสมกับโครงสร้างทีมีอยู่ในอุปกรณ์ FPGA ดังนั้นในการ Optimize ซอร์พแวร์สังเคราะห์วงจร จะต้องทำการ Optimize ให้ได้เป็นวงจรที่ประกอบด้วยกลุ่มของลอจิกที่เหมาะสมกับอุปกรณ์ FPGA นั้นๆ จึงทำให้ผลที่ได้มีประสิทธิภาพและในขั้นตอนการสังเคราะห์วงจรนี้ ผู้ออกแบบสามารถกำหนดข้อบังคับสำหรับโมเดลแต่ละตัวได้ เช่น ข้อบังคับในเรื่องเวลา (Timing Constraints) หรือข้อบังคับในเรื่องของพื้นที่ (Area) หรือกำหนดชนิดและตำแหน่งของ I/O ซึ่งข้อบังคับเหล่านี้จะถูกนำไปใช้ในขั้นตอน Optimize เพื่อให้วงจรที่ได้เป็นไปตามที่กำหนด ส่วนสำคัญในการ Optimize คือการเทียบ (Mapping) โมเดลให้เข้ากับเทคโนโลยีที่ใช้เพื่อให้ได้วงจรที่เหมาะสมกับโครงสร้างและสถาปัตยกรรมภายในอุปกรณ์ FPGA เมื่อทำการสังเคราะห์วงจรเสร็จแล้ว ซอร์พแวร์ การสังเคราะห์วงจรก็จะมีการรายงานผลว่าโมเดลที่ออกแบบไปนั้นเป็นอย่างไร เช่นมีค่าความหน่วง (Delay) เท่าไหร่ ใช้ทรัพยากรต่างๆใน FPGA อะไรบ้าง เมื่อมาถึงขั้นตอนนี้ ผู้ออกแบบก็จะทราบว่าโมเดลเป็นไปตามข้อบังคับหรือไม่ ถ้าไม่ก็สังเคราะห์ใหม่จนกว่าจะเป็นไปตามที่กำหนด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.6 การใช้งานโปรแกรม Max Plus II เบื้องต้น

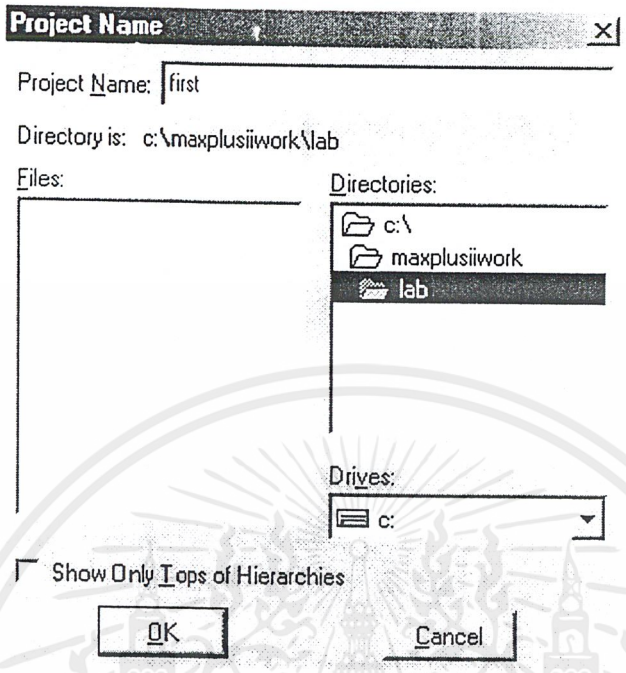
1. Run โปรแกรม Max+Plus II 9.5 BASELINE ขึ้นมา



รูปที่ 4.8 โปรแกรม Max+Plus II 9.5 BASELINE

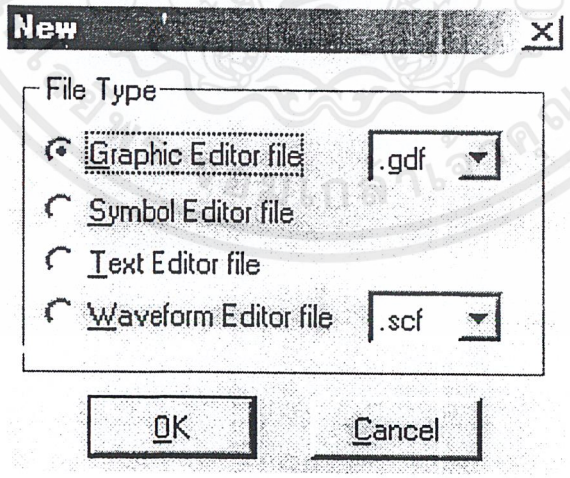
จากนั้นทำการตั้งชื่อของโปรเจ็ค ที่เราจะสร้างขึ้นมา ชื่อของไฟล์ทุกไฟล์ที่อยู่ในโปรเจ็คเดียวกันจะมีชื่อที่เหมือนกันแตกต่างกันที่นามสกุล การตั้งชื่อโปรเจ็คเริ่มจาก **File / Project / Name** เลือก Folder ที่จะเก็บโปรเจ็คของเราและใส่ชื่อของโปรเจ็คเป็น **First**

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.9 การตั้งชื่อโปรเจค

2.เลือกประเภทของไฟล์ที่จะสร้าง **File / New / Graphic Editor File**จะใช้งานขึ้นมาวางไว้ที่ Graphic Form



รูปที่ 4.10 การเลือกประเภทไฟล์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โดยที่

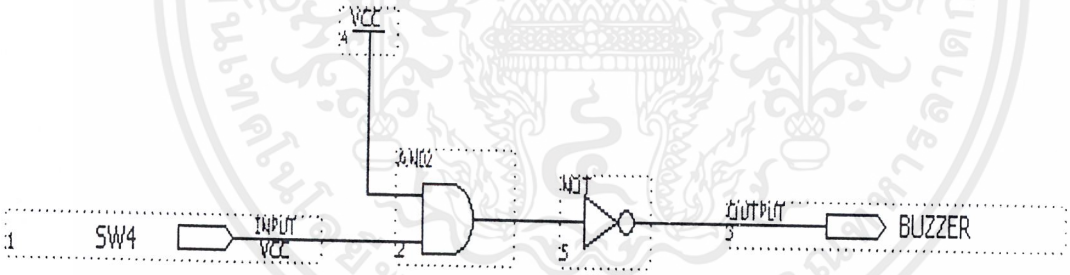
Graphic Editor File: เป็นไฟล์กราฟฟิกส์ที่เราสามารถนำอุปกรณ์ต่างๆใน Library มาวางต่อกันได้เลย

Symbol Editor File: เป็นไฟล์ที่ใช้เก็บสัญลักษณ์เพื่อสื่อให้รู้ว่าโมเดลที่เราได้สร้างขึ้นมีอินพุท, เอาท์พุทเป็นอย่างไร

Text Editor File: เป็น Text file ใช้สำหรับเขียน Source code เพื่ออธิบายพฤติกรรมของวงจรหรือโมเดลต่างๆที่เราจะสร้างขึ้นหรือเพื่อไว้สำหรับเก็บข้อความทั่วๆไป

Waveform Editor File: เป็นไฟล์ไว้สำหรับการกำหนดรูปแบบของสัญญาณอินพุทเพื่อใช้ในการ Simulate และไว้สำหรับให้ Max Plus II แสดงผลของเอาท์พุทที่ได้จากการ Simulate

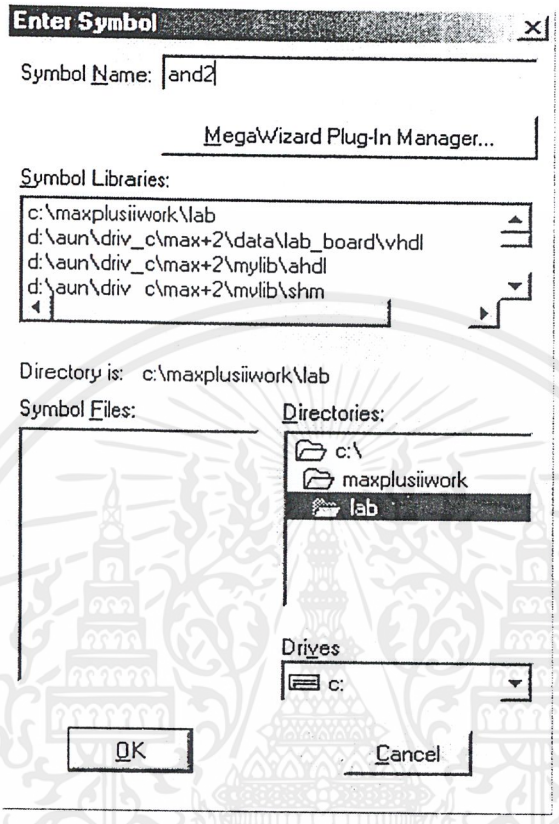
3.ในการอธิบายการใช้งานโปรแกรม Max Plus II นี้เราจะใช้วงจรดังรูปที่ 4.11 ประกอบคำอธิบาย



รูปที่ 4.11 วงจรตัวอย่าง

จากรูปที่ 4.11 AND GATE, NOT GATE, INPUT, OUTPUT, VCC จะเป็นอุปกรณ์ที่มาพร้อมกับโปรแกรม Max Plus II ในการที่จะเอาอุปกรณ์ต่างๆ ใน Library ของ Max Plus II มาใช้ให้ทำการ Double Click ที่ Graphic Form จะมี Dialog Enter Symbol ปรากฏขึ้นมา ให้เลือกอุปกรณ์ที่เราต้องการจะใช้งานขึ้นมาวางไว้ที่ Graphic Form

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.12 การกำหนดสัญลักษณ์

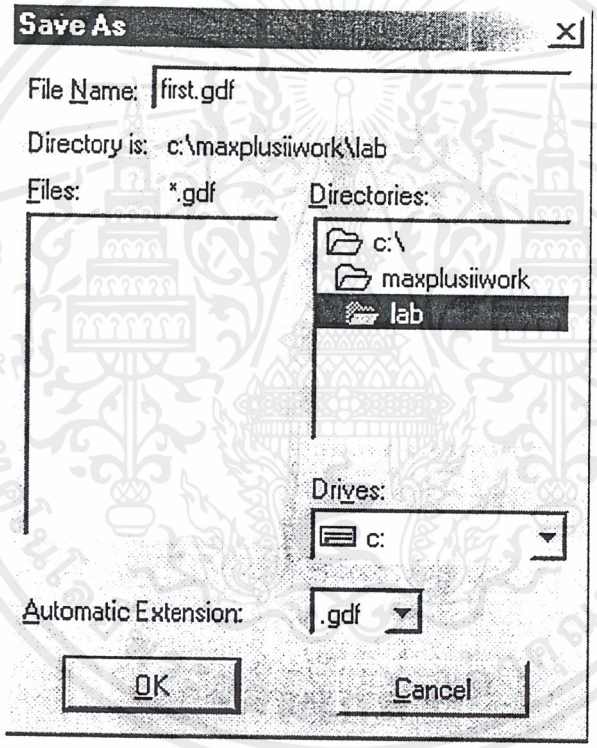
โดยที่

- AND 2 input จะใช้ชื่อ Symbol Name ว่า **and2**
- NOT 1 input จะใช้ชื่อ Symbol Name ว่า **not**
- Input จะใช้ชื่อ Symbol Name ว่า **input**
- Output จะชื่อ Symbol Name ว่า **output**
- VCC จะใช้ชื่อ Symbol Name ว่า **vcc**

อุปกรณ์ทั้งหมดจะอยู่ใน Symbol Library ชื่อ **...\maxplusii\max2lib\primg** เมื่อทำการวางอุปกรณ์จนครบทุกตัวแล้ว ให้ทำการเชื่อมต่ออุปกรณ์ต่างๆเข้าด้วยกันโดยเลื่อน pointer ของ mouse ไป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

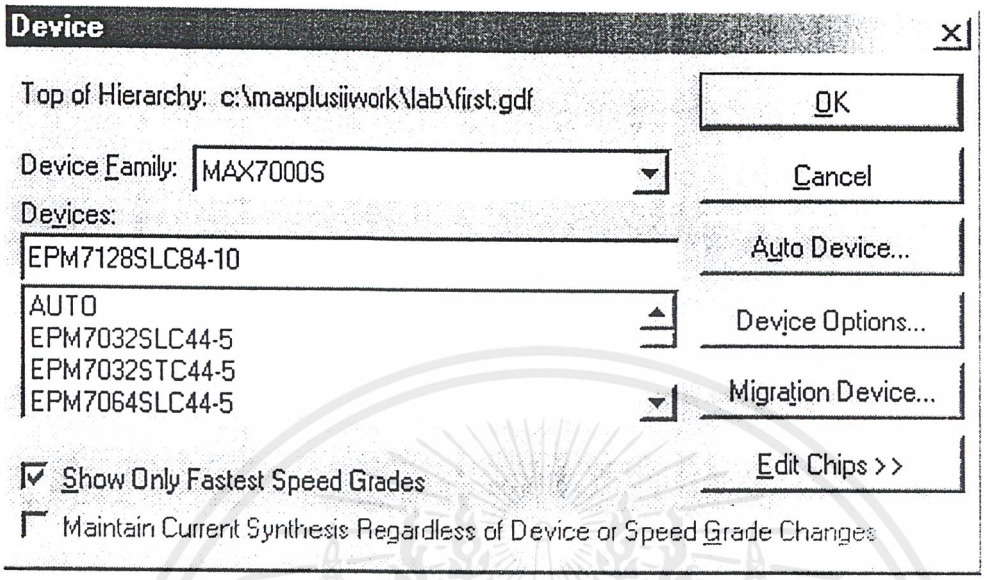
ที่ขาต่างๆของอุปกรณ์ pointer ของ mouse จะเปลียนเป็นเครื่องหมาย + ให้ทำการ Click ขวาค้างไว้แล้วลากเส้นไปต่อกับ input หรือ output ของอุปกรณ์ตัวอื่น เมื่อเชื่อมต่ออุปกรณ์ต่างๆจนครบทั้งหมดแล้วให้ทำการเปลียน PIN_NAME ของอุปกรณ์ input กับ output โดยการไป **Double Click** ที่ PIN_NAME ของ input แล้วเปลี่ยนชื่อเป็น **SW1** สำหรับ PIN_NAME ของ output เปลี่ยนเป็น **BUZZER** เมื่อวาดรูปวงจรเสร็จเรียบร้อยแล้วให้ทำการบันทึกวงจรที่เราสร้างขึ้นมา **File / Save as** ให้ใส่ชื่อไฟล์ปีน **first.gdf** แล้วก็ OK



รูปที่ 4.13 การบันทึกข้อมูล

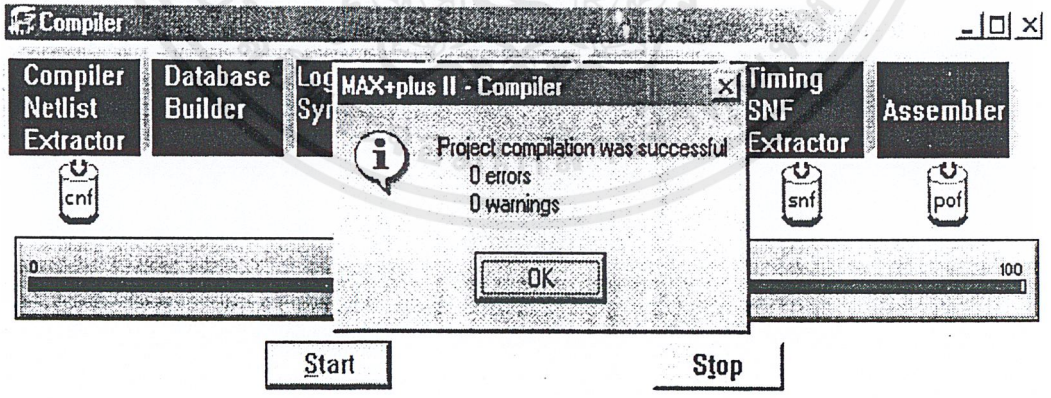
4. ทำการระบุเบอร์ของชิพที่เราจะใช้ **Assign / Device** จะมีไดอะล็อก Device ปรากฏขึ้นมา ให้ทำการเลือก Device Family เป็น **MAX700S** และเลือก Device เป็น **EPM7128SLC84-10**

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.14 การกำหนดอุปกรณ์

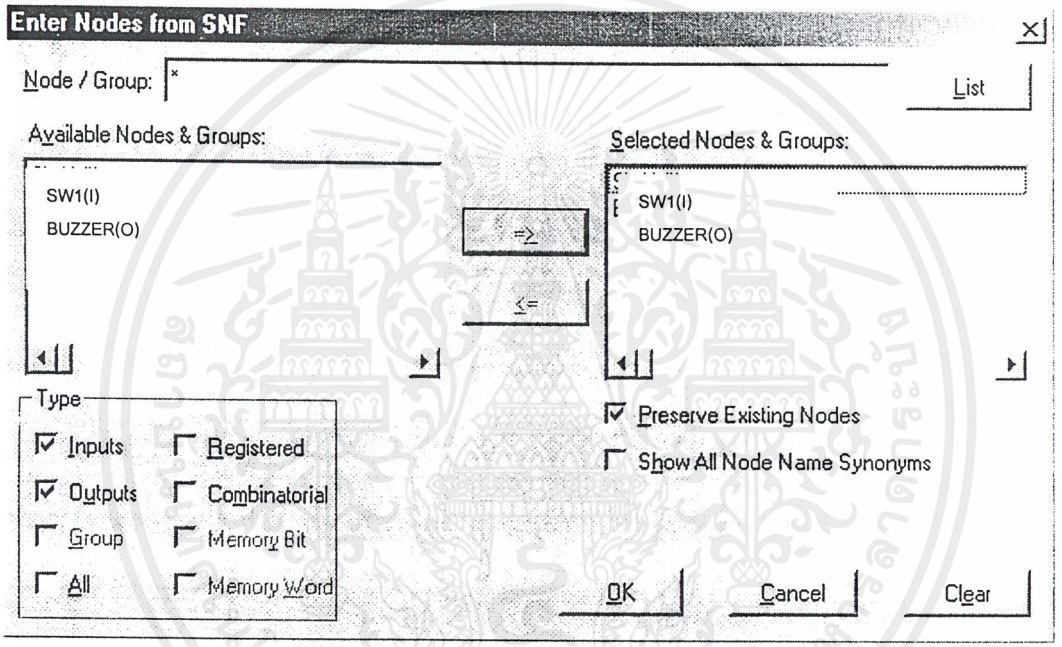
4. ทำการคอมไพล์วงจรที่เราได้สร้างขึ้นมา **Max+Plus II / Compiler / Start** เมื่อคอมไพล์เสร็จจะมีหน้าต่างรายงานผลการคอมไพล์ error กับ warring หากมีความผิดพลาดเกิดขึ้นจะมีข้อความสีแดงบอกว่า error และจะบอกด้วยว่า error เพราะอะไร



รูปที่ 4.15 รูปการคอมไพล์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

6. ทำการ Simulate โดยจะต้องสร้างไฟล์ Waveform ขึ้นมาก่อน **File / New / Waveform Editer file** (จะได้หน้าต่างดังรูปที่ 4.17) ก่อนที่จะทำการ Simulate จะต้องทำการกำหนดอินพุทให้แก่วงจรก่อน โดยโหลด Noad ต่างๆ เข้ามา **Noad / Enter Noad from SNF จะมีไดอะล๊อก** Enter Nodes from SNF ปรากฏขึ้นมาให้คลิกที่ **List** จะมี Noad ต่างๆ ที่อยู่ในวงจรปรากฏขึ้นมาให้เราเลือก Noad ที่เป็น input กับ output ดังรูปที่ 4.16 โดยการกดปุ่มที่มี **เครื่องหมายลูกศรชี้ไปทางขวา**



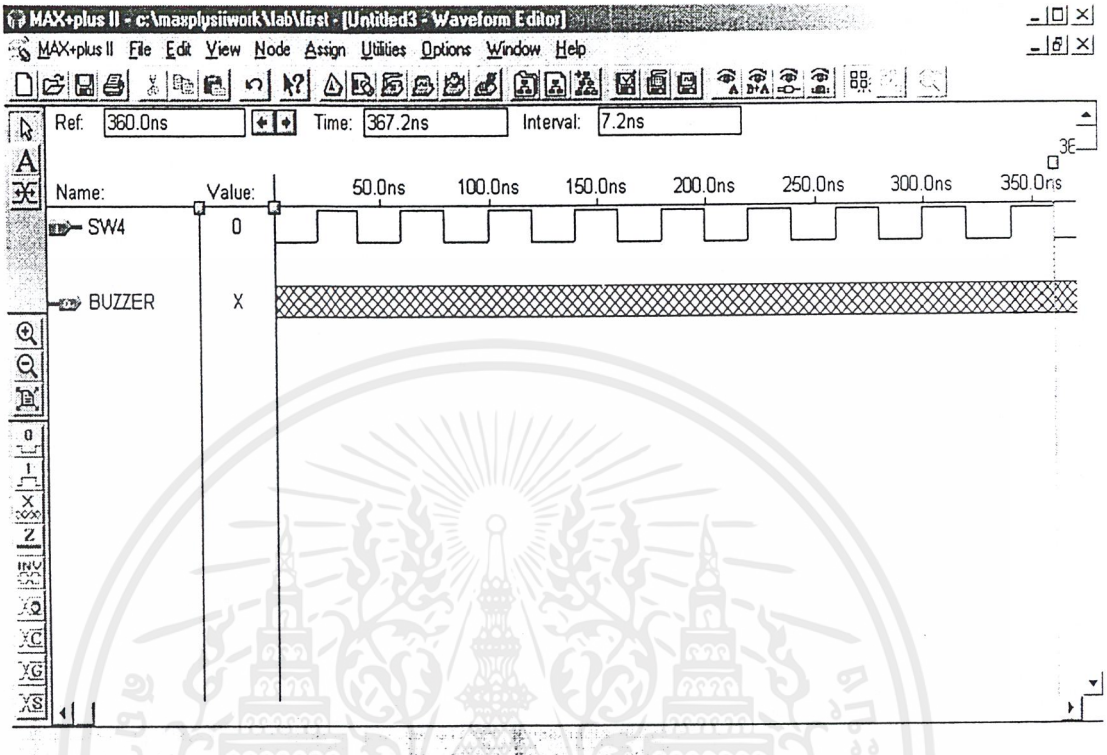
รูปที่ 4.16 การกำหนดโหนด

- กำหนดเวลาสิ้นสุดการ Simulate (End Time) ให้กับตัวโปรแกรม **File / End Time** ให้ค่า End Time เท่ากับ **1.0 us** ซึ่งจะเป็นการบอกว่าให้โปรแกรมทำการ Simulate ตั้งแต่ 0.0 us - 1.0 us

- กำหนดขนาดของกริด **Option / Gride Size** กำหนดให้กริดมีขนาดเท่ากับ **10.00ns**

- ทำการกำหนดรูปแบบสัญญาณให้กับ Node Input โดยที่ คลิกที่ **SW1** แถบดำปรากฏขึ้นมา หลังจากนั้นทำการกำหนดรูปแบบสัญญาณให้มีลักษณะเป็นพัลซ์ **Edit / Over write / Clock ในช่อง Multiply By ให้ใส่ 2** แล้วคลิก OK ซึ่งจะเป็นการกำหนดให้สัญญาณในช่วง Logic 1 และ Logic 0 มีค่าเวลาเป็น 2 เท่าของกริด

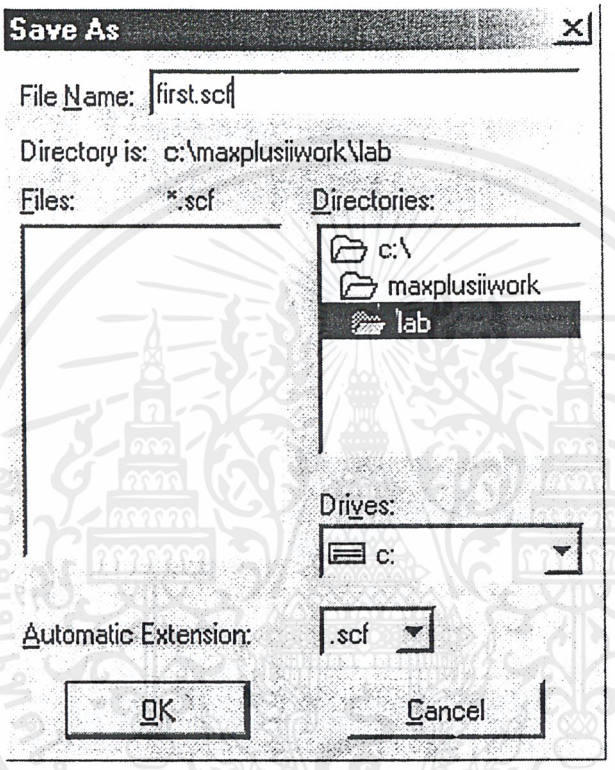
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.17 การจำลองการทำงาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

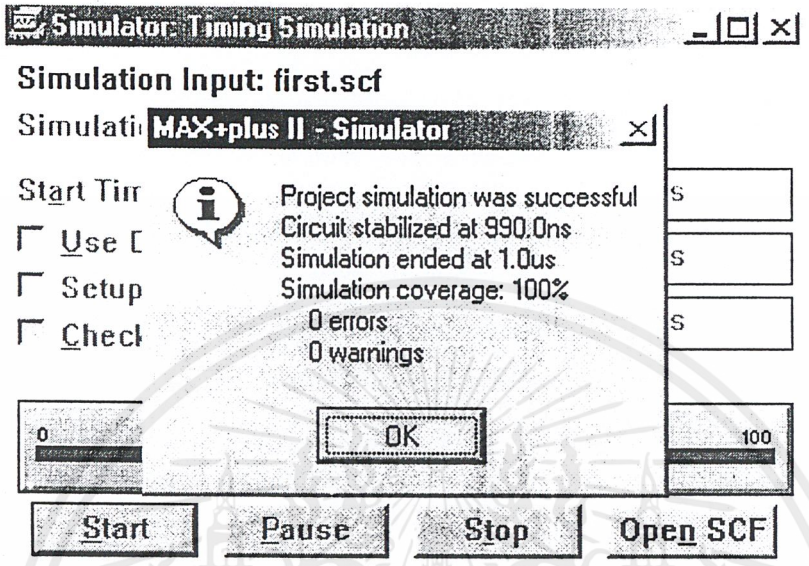
- บันทึกไฟล์ Waveform ที่ได้สร้างขึ้น **File / Save as** กำหนดให้ชื่อไฟล์ที่จะบันทึกเป็น **First.scf**



รูปที่ 4.18 การบันทึกรูปกราฟ

- ทำการจำลองการทำงานของวงจร Max+**Plus II / Simulation** เมื่อ Simulate เสร็จจะมีไดอะล็อกขึ้นมารายงานผลการ Simulate ว่ามี error หรือ warring หรือไม่ และเราสามารถดูผลการ Simulate ได้ที่ Form ของ Waveform Editor พร้อมทั้ง**วาด Waveform ต่างๆไว้**

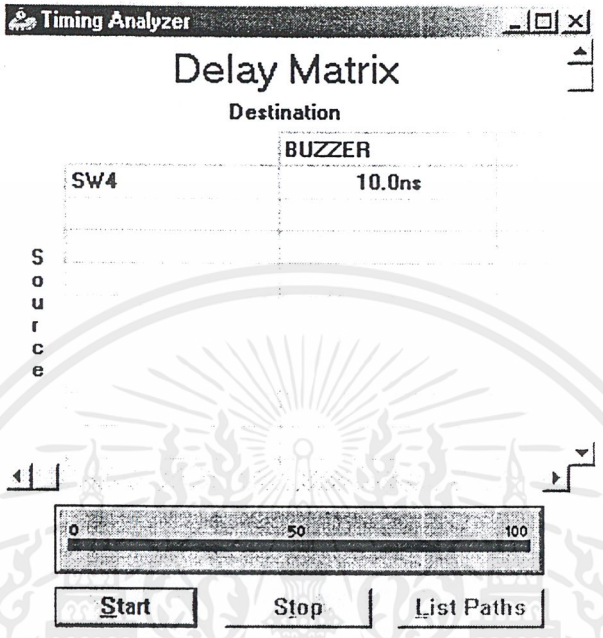
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.19 การจำลองการทำงานอีกครั้ง

7. วิเคราะห์ Timing Analyzer จะใช้สำหรับหาค่า Delay Time ระหว่าง Noad ต่างๆ **Max+Plus II / Timing Analysis** จะมี ไดอะล็อก Timing Analyzer ปรากฏขึ้นมาให้เลือกชนิดของการวิเคราะห์ ให้ทำการเลือกวิเคราะห์ค่าเวลาหน่วง **Analysis / Delay Time**

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

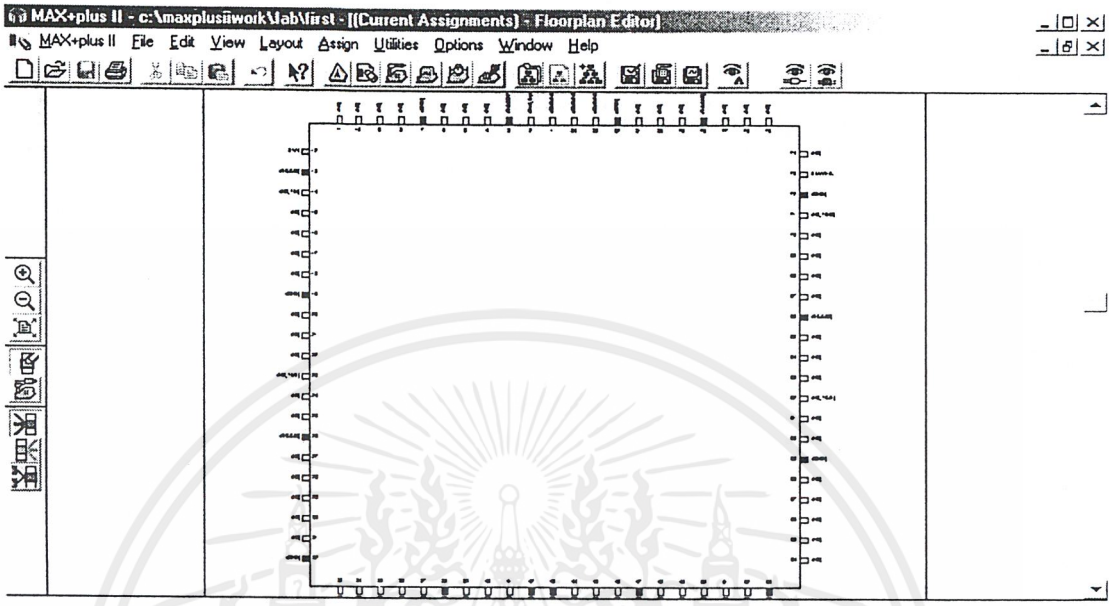


รูปที่ 4.20 การจำลองการหน่วงเวลา

8.การเปลี่ยนแปลงตำแหน่งขาของชิพ

เราสามารถเปลี่ยนแปลงตำแหน่งขาของชิพโดยคลิกที่ **Max+Plus / Floorplan Editor** จะปรากฏฟอร์มของ Floorplan Editor ขึ้นมาเลือกที่เมนู **Assign / Back-Annotate Project** คลิกที่ **Chip,Pin & Device / OK** สังเกตุว่าเมื่อผ่านขั้นตอนนี้เราจะเห็นตำแหน่งขาที่โปรแกรม Max+Plus II ได้เลือกให้เรา ที่นี้เราสามารถที่จะเปลี่ยนแปลงตำแหน่งขาต่างๆเหล่านี้ได้เพื่อให้เหมาะสมกับบอร์ดทดลองของเราโดยคลิกที่ **Layout / Current Assignment Foorplan** ให้ทำการเปลี่ยนตำแหน่งขาของ **sw1 เป็นขาที่ 49** และตำแหน่งของขาของ **Buzzer เป็นขาที่ 10** โดยการคลิกขวาที่ขา sw1 ค้างไว้แล้วลากมาที่ขา 49 การเปลี่ยนแปลงตำแหน่งของขา Buzzer ก็ให้ทำเช่นเดียวกัน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

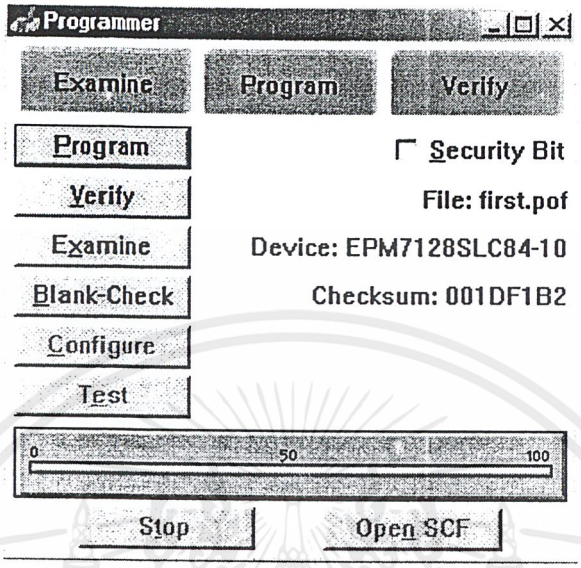


รูปที่ 4.21 การกำหนดขาใช้งาน

เมื่อทำการเปลี่ยนแปลงตำแหน่งขาของชิพเรียบร้อยแล้วให้ save และ คอมไพล์ใหม่อีกครั้ง Project / Save Compile & Simulate

9. การโปรแกรมวงจรลงในชิพ Max+**Plus II / Programmer / Program** ไฟล์ข้อมูลที่จะโปรแกรมลงในชิพหากเลือก Device ของ FPGA เป็นอุปกรณ์ประเภท EEPROM – Base FPGA หรือที่เรียกว่า CPLD (เบอร์ IC จะขึ้นต้นด้วย EPM) ไฟล์ที่จะโปรแกรมจะมีนามสกุล *.pof

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.22 การโปรแกรม

10.ทดสอบการทำงานจริงของวงจรที่เราได้โปรแกรมลงในชิพ แล้วอธิบายการทำงาน

4.7 สรุปขั้นตอนการใช้ MAX+PLUS IIวาดวงจร

1. Run โปรแกรม **Max+Plus II 9.5 BASELINE** ขึ้นมา
2. ตั้งชื่อโปรเจ็คเริ่มจาก **File / Project / Name** เลือก Folder ที่จะเก็บโปรเจ็คของเราและใส่ชื่อของโปรเจ็คตามต้องการ
3. สร้างไฟล์วงจร **File / New / Graphic Editor File** แล้ววาดวงจรที่ต้องการ แล้ว save

Compile

4. ระบุเบอร์ของชิพ **Assign / Device** ให้ทำการเลือก Device Family เป็น **MAX700S** และเลือก Device เป็น **EPM7128SLC84-10**
5. คอมไพล์วงจรที่เราได้สร้างขึ้นมา **Max+Plus II / Compiler / Start**

Simulate

6. Simulate โดยจะต้องสร้างไฟล์ Waveform ขึ้นมาก่อน **File / New / Waveform Editer file**
7. โหลด Noad ต่างๆ เข้ามา **Noad / Enter Noad from SNF**

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

8. กำหนด **File / End Time** เท่ากับ **1.0 us**
9. กำหนด **Option / Gride Size** มีขนาดเท่ากับ **10.00ns**
10. กำหนดรูปแบบสัญญาณให้กับ **Node Input**
11. บันทึกไฟล์ Waveform ที่ได้สร้างขึ้น **File / Save as**
12. การจำลองการทำงานของวงจร Max+**Plus II / Simulation**

กำหนดขา

13. เปลี่ยนแปลงตำแหน่งขาของชิพ **Max+Plus / Floorplan Editor**
14. เลือกที่เมนู **Assign / Back-Annotate Project** คลิกที่ **Chip,Pin & Device / OK**
15. คลิกที่ **Layout / Current Assignment Foorplan**
16. ทำการ **Drag and Drop** ขาต่างๆ ให้ตรงกับความต้องการ

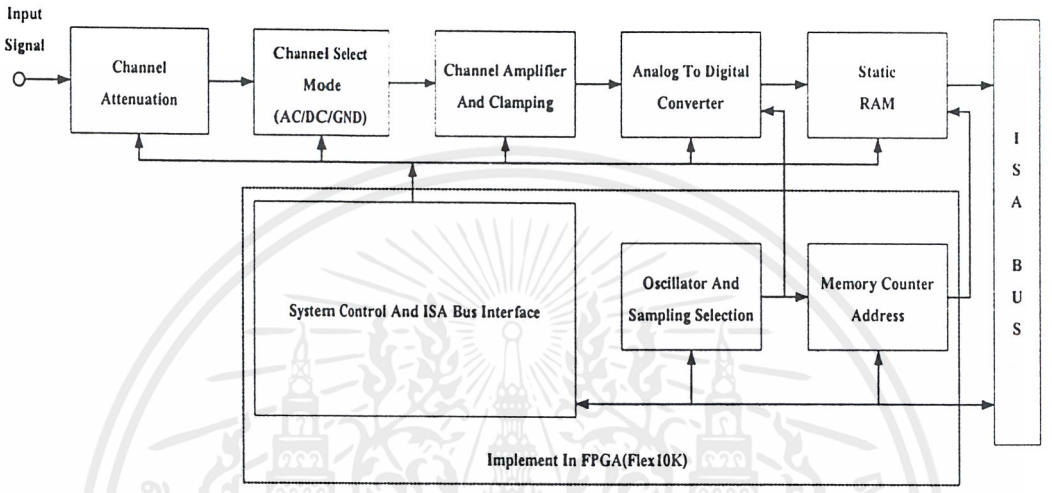
โปรแกรม

17. โปรแกรมวงจรลงในชิพ Max+**Plus II / Programmer / Program** ทดสอบการทำงานจริงของวงจร

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 5

หลักการออกแบบวงจร และโปรแกรมควบคุมการทำงาน



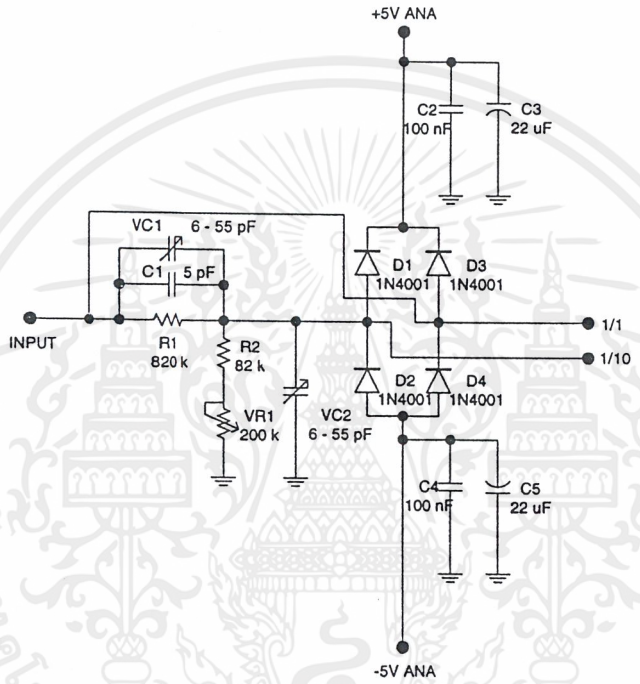
รูปที่ 5.1 บล๊อกไดอะแกรมแสดงการทำงานของคอมพิวเตอร์ออสซิลโลสโคปการ์ด

จากบล๊อกไดอะแกรม สัญญาณอินพุทที่เข้ามาของแต่ละช่อง จะถูกส่งเข้าสู่วงจรลดทอน (Attenuator) เพื่อลดทอนแรงดันสูงๆให้มีค่าขนาดสัญญาณลดลง วงจรเลือกโหมดการทำงาน(Channel Select Mode) จะทำการเลือกว่าจะทำการวัดในโหมด AC, DC และ Ground สัญญาณที่ผ่านการเลือกโหมดจะถูกส่งเข้าไปในวงจร ขยายขนาดสัญญาณ(Channel Amplifier) เพื่อขยายขนาดสัญญาณให้เหมาะสมที่จะทำการแซมปลิง(Sampling)ได้ นอกจากนี้ภายในส่วนนี้ยังมีวงจรยกระดับสัญญาณเพื่อใช้ยกระดับสัญญาณที่ต่ำๆ ให้อยู่ในระดับที่เหมาะสม และการแสดงผลที่ดีขึ้นจากนั้นสัญญาณจะถูกส่งไปยังวงจรแปลงสัญญาณอนาลอกเป็นสัญญาณดิจิตอล(Analog To Digital Converter) ข้อมูลที่ได้จากการแปลงสัญญาณ(Convert) จะถูกส่งเข้าไปยังหน่วยความจำของการ์ด ส่วนบล๊อกไดอะแกรมส่วนล่างนั้นการใช้งานจริงจะถูกสังเคราะห์เก็บไว้ในชิปเอฟพีจีเอซึ่งส่วนการผลิตสัญญาณหลายความถี่ (Oscillator And Sampling Selection) จะทำหน้าที่ผลิตความถี่ และเลือกความถี่ที่จะใช้ในการแซมปลิง (Sampling) มาเป็นคล๊อกเพื่อจ่ายให้กับวงจรนับแอดเดรสของหน่วยความจำ(Memory Counter Address) ในส่วนของวงจรควบคุมการทำงานของระบบ และการติดต่อกับไอซาบัส(System Control

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

And ISA Bus Interface) จะทำหน้าที่ควบคุมการทำงานของส่วนต่างๆของระบบ และเชื่อมต่อผ่านพอร์ทไอซาบัสของคอมพิวเตอร์

5.1 วงจรลดทอนสัญญาณ (Attenuation)

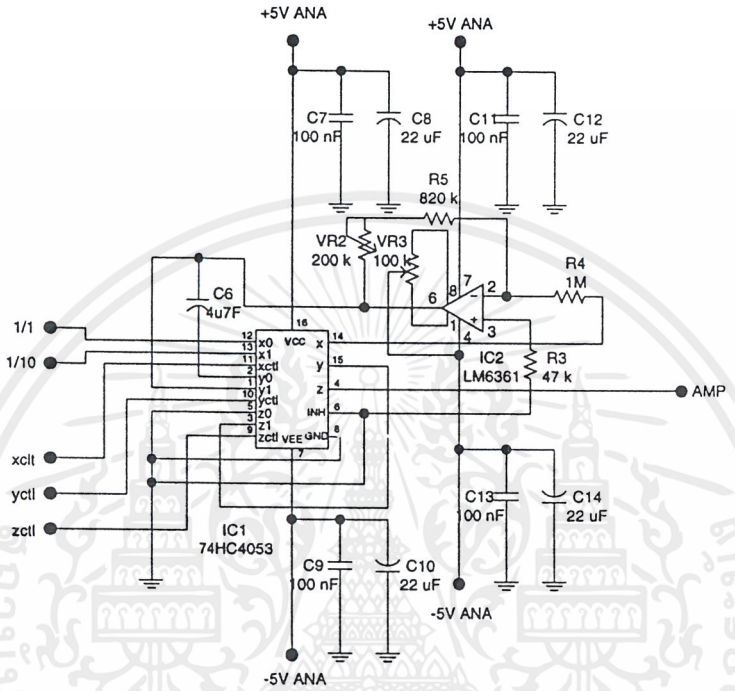


รูปที่ 5.2 วงจรลดทอนสัญญาณ

เนื่องจากสัญญาณที่เข้ามามีขนาดที่แตกต่างกัน ดังนั้นจะต้องปรับระดับสัญญาณให้เหมาะสมก่อนที่จะส่งไปยังภาคอื่นๆต่อไป ซึ่งการลดทอนสัญญาณจะใช้หลักการแบ่งแรงดัน (Voltage Divider) และการลดทอนสัญญาณจะต้องเป็นอิสระกับความถี่ ซึ่งจะต้องใช้วงจรลดทอนแบบที่มีการชดเชยความถี่ (Compensate Attenuator) ซึ่งจะเป็นวงจรอาซีโวลท์เตจดีไวเดอร์ (RC Voltage Divider) ในวงจรมีการลดทอนทั้งหมด 2 ระดับ คือ 1/1 และ 1/10 ไดโอด D1-D4 มีเอาไว้เพื่อป้องกันการป้อนแรงดันเกินกว่าที่กำหนด ซึ่งถ้าไม่มีวงจรในส่วนนี้วงจรในภาคต่อไปอาจเกิดความเสียหายได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5.2 วงจรเลือกโหมดการวัดและวงจรบัฟเฟอร์ (Selector Mode and Buffer)

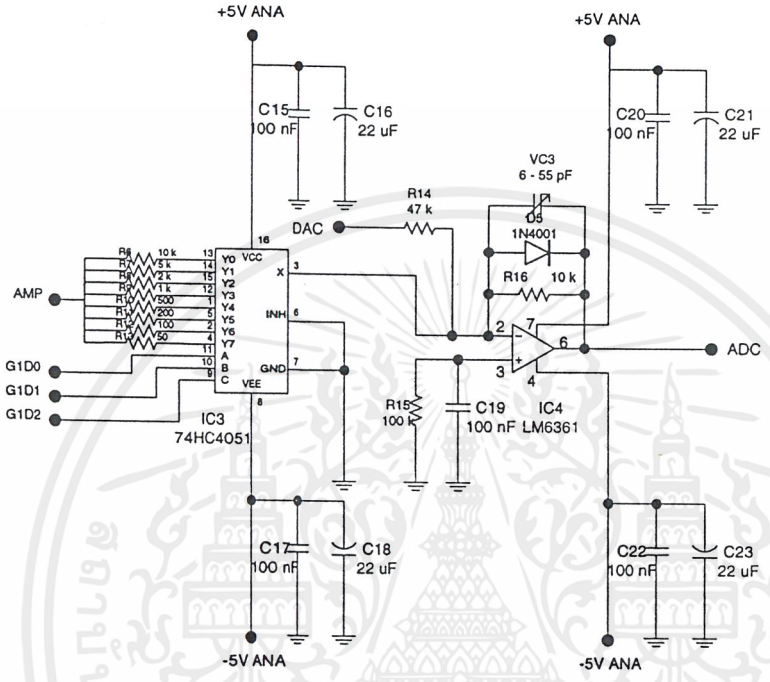


รูปที่ 5.3 วงจรเลือกโหมดการวัดและวงจรบัฟเฟอร์

สัญญาณที่ผ่านการลดทอนแบบชดเชยความถี่แล้ว จะถูกเลือกด้วยอิเล็กทรอนิกส์สวิตช์และนำเอาสัญญาณที่ได้ผ่านวงจรบัฟเฟอร์ เพื่อเป็นการส่งผ่านสัญญาณจากภาควงจรลดทอน ซึ่งมีเอาท์พุทอิมพีแดนซ์สูงไปยังภาควงจรขยาย โดยใช้ออปแอมป์เป็นบัฟเฟอร์ต่อแบบ Inverting และมี VR_3 ใช้ในการปรับออฟเซทให้แก่ออปแอมป์ จากนั้นสัญญาณจะถูกแยกโดย C_6 ออกเป็น 2 ส่วนคือ สัญญาณ AC และสัญญาณ DC สัญญาณทั้ง 2 ถูกป้อนเข้าสู่อิเล็กทรอนิกส์สวิตช์อีกครั้ง เพื่อทำการเลือก โหมด ในการวัด โดย โหมด ในการวัดมีอยู่ 3 โหมดด้วยกันคือ AC โหมด DC โหมด และ Ground โดยการทำงานของอิเล็กทรอนิกส์สวิตช์จะถูกควบคุมด้วยสัญญาณ Xclt , Yclt และ Zclt ซึ่งถูกส่งมาจาก FPGA

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5.3 วงจรขยายสัญญาณแนวตั้ง (Vertical Amplifier)



รูปที่ 5.4 วงจรขยายสัญญาณ

ในการวัดสัญญาณที่มีขนาดเล็กมีความจำเป็นอย่างยิ่งที่ต้องมีการขยายสัญญาณให้มีขนาดที่เหมาะสมก่อนป้อนเข้าสู่ A/D Converter และวงจรขยายตรงนั้นต้องสามารถเลือกอัตราการขยายได้เพื่อในมีขาดที่เหมาะสมกับสัญญาณที่นำมาขยายด้วย จากวงจรในรูปที่ 5.4 วงจรขยายที่เราใช้จะเป็นวงขยายกลับเฟส (Inverting Amp) โดยสามารถเลือกระดับการขยายได้ 8 ระดับ ดังตารางที่ 5.1

การเลือกอัตราการขยายของวงจรจะใช้ไอซีเบอร์ 74HC4051 ซึ่งทำหน้าที่เป็น วงจรมัลติเพล็กเซอร์โดยการทำงานของวงจรมัลติเพล็กเซอร์จะถูกควบคุมด้วยคอมพิวเตอร์ ไดโอด D5 ที่ต่ออยู่ในวงจรมีหน้าที่ในการป้องกันกระแสมิให้ย้อนกลับเข้าไปในออปแอมป์ ซึ่งอาจสร้างความเสียหายให้กับออปแอมป์ได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

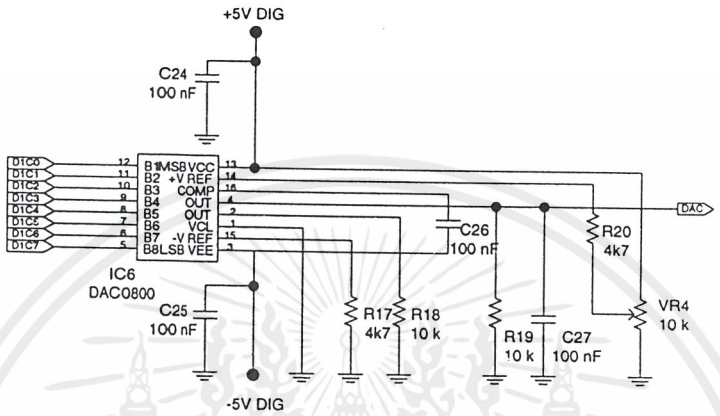
ตารางที่ 5.1 อัตราขยายของวงจรขยายสัญญาณ

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

สิ่งที่เราต้องคำนึงอย่างมากในการออกแบบวงจรขยายก็คือเราต้องทำให้วงจรมีอัตราขยายที่คงที่ตลอดทุกๆย่านความถี่ของสัญญาณอินพุทและมีแบนวิดธ์ที่กว้างที่สุด แต่จากการทดลองเราพบว่าอัตราขยายเพิ่มขึ้นเมื่อความถี่สูงขึ้นและจะค่อยๆลดลง ความไม่คงที่ของวงจรขยายเกิดขึ้นได้จาก 2 สาเหตุนั่นคือ ผลของ Zero ทำให้อัตราขยายเพิ่มขึ้นและผลของ Pole ทำให้อัตราขยายลดลง วิธีหนึ่งที่สามารถแก้ปัญหาดังกล่าวได้ก็คือการต่อ VC_3 เข้าไป เพราะที่ความถี่สูง VC_3 จะทำให้ Feed Back Impedance (Zf) มีค่าลดลง ผลก็คืออัตราขยายมีค่าลดลงแต่การทำในลักษณะดังกล่าวนี้จะทำให้แบนวิดธ์ลดลงเล็กน้อย ซึ่งอยู่ในขอบเขตที่สามารถยอมรับได้ จากนั้นสัญญาณก็จะถูกส่งต่อไปยัง A/D Converter ต่อไป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5.4 วงจรยกระดับสัญญาณ (DC Adjust)

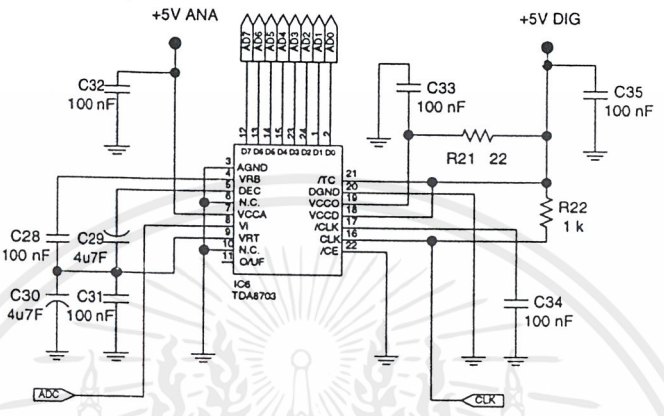


รูปที่ 5.5 วงจรยกระดับสัญญาณ

จากเงื่อนไขในการทำงานของ A/D Converter ที่ว่าระดับแรงดันอินพุทของ A/D Converter จะต้องมีค่าอยู่ระหว่าง 1.55V – 3.26V ดังนั้นเราต้องสร้างวงจรขึ้นมาเพื่อยกระดับสัญญาณให้สัญญาณสวิงอยู่ในช่วงที่ A/D Converter ทำงานได้ โดยวงจรที่เราสร้างจะใช้ไอซีเบอร์ DAC0808ต่อวงจรดังรูปที่ 5.5 ซึ่งระดับของสัญญาณDC ที่เกิดทางเอาท์พุทจะถูกควบคุมด้วยข้อมูลที่มาจากคอมพิวเตอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5.5 วงจรแปลงสัญญาณอนาลอกให้เป็นสัญญาณดิจิตอล (Analog to Digital Converter)

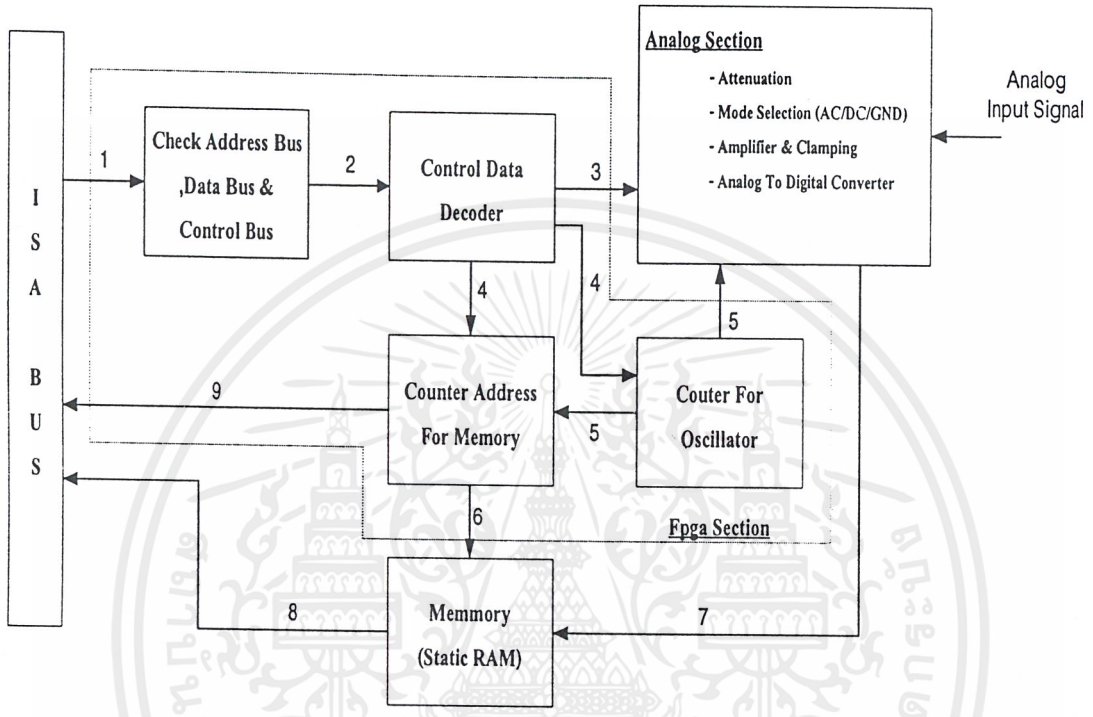


รูปที่ 5.6 วงจรแปลงสัญญาณอนาลอกให้เป็นสัญญาณดิจิตอล

วงจรแปลงสัญญาณอนาลอกให้เป็นสัญญาณดิจิตอล จะใช้ไอซีเบอร์ TDA8703 ซึ่งเป็นแบบแฟลชมีความละเอียด 8 บิตและมีอัตราการสุ่มสูงสุดที่ 40 MHz ในการแปลงสัญญาณอนาลอกให้เป็นสัญญาณดิจิตอลเราจำเป็นอย่างยิ่งที่จะต้องเลือกสุ่มให้เหมาะสมกับสัญญาณอินพุททั้งนี้ก็เพื่อเวลาที่เราแปลงกลับสัญญาณเราจะได้รายละเอียดของสัญญาณที่ชัดเจนและถูกต้องมากที่สุดโดยสัญญาณสุ่มจะได้มาจาก FPGA และถูกเลือกด้วยคอมพิวเตอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5.6 หลักการทำงานวงจรดิจิตอล



รูปที่ 5.7 บล๊อกไดอะแกรมแสดงการทำงานของวงจรส่วนดิจิตอลในเอฟพีจีเอ

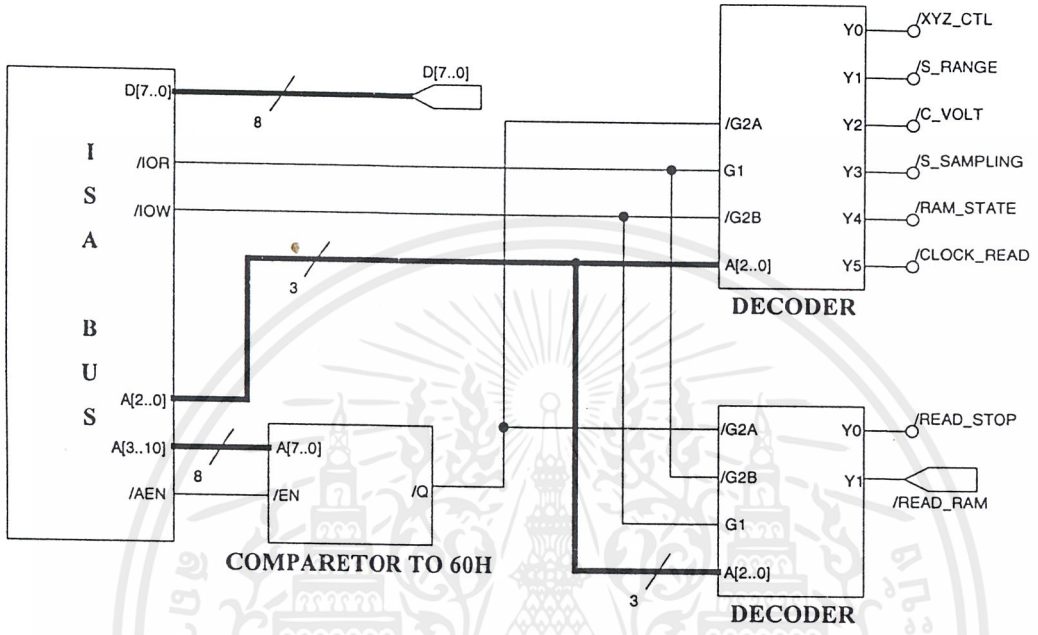
การทำงานของวงจรเริ่มจาก สัญญาณข้อมูลบัส ,สัญญาณแอดเดรสบัส และสัญญาณควบคุม (สัญญาณหมายเลข 1)ส่งออกมาจากไอซาบัสผ่านในส่วนวงจรถอดรหัสเพื่อตรวจสอบว่าคอมพิวเตอร์มีการเรียกใช้งานให้การ์ดทำงานจริงหรือไม่ ถ้าไม่ถูกต้องกระบวนการต่างๆของวงจรในการ์ดจะไม่มีการทำงานใดๆเลยแต่ถ้าถูกต้องแล้ววงจรต่างๆในการ์ดก็จะทำงานต่อไปเริ่มที่สัญญาณที่ 2 นั่นคือสัญญาณข้อมูลการควบคุม ซึ่งสัญญาณนี้เป็นสัญญาณข้อมูลที่ส่งออกมาจากคอมพิวเตอร์เพื่อให้การทำงานในส่วนนี้ตรวจสอบว่าคอมพิวเตอร์ต้องการสั่งงานให้การ์ดทำงานอะไรบ้าง การทำงานถัดมานั้นสัญญาณส่วนที่ 3 (สัญญาณการควบคุมวงจรอนาลอก) จะควบคุมวงจรอนาลอกให้ทำการสั่งงานควบคุมการทำงานของวงจรในส่วนอนาลอกให้ทำงานตามที่กำหนดเอาไว้ สัญญาณส่วนที่ 4 (สัญญาณควบคุมวงจรนับ) นั้นเป็นสัญญาณควบคุมวงจรดิจิตอลซึ่งเป็นวงจรนับ 2 วงจรเมื่อสังเกตที่บล๊อกไดอะแกรมจะพบว่าการส่งสัญญาณที่ 4 นั้นจะส่งให้กับ 2 บล๊อกนั่นคือส่วนหนึ่งเป็นส่วนที่ทำการนับแอดเดรสของ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หน่วยความจำที่ใช้งานในวงจร และอีกส่วนหนึ่งนั้นจะเป็นส่วนที่ทำการนับเพื่อสร้างสัญญาณความถี่ต่างๆ(สัญญาณที่ 5 สัญญาณการเซมปลิง) เพื่อควบคุมการทำงานของวงจรแปลงสัญญาณอนาลอกเป็นสัญญาณดิจิตอล และช่วยในการนับแอดเดรสของหน่วยความจำ เมื่อแอดเดรสของหน่วยความจำมีการเปลี่ยนแปลง(สัญญาณที่ 6 สัญญาณแอดเดรสหน่วยความจำของการ์ด) ก็จะทำให้หน่วยความจำเปลี่ยนแปลงตำแหน่งการบันทึกข้อมูล ซึ่งการทำงานจะทำงานไปพร้อมๆกับการส่งข้อมูลจากวงจรแปลงสัญญาณอนาลอกเป็นสัญญาณดิจิตอล (สัญญาณที่ 7 สัญญาณข้อมูลดิจิตอลที่ได้จากการแปลง) เมื่อคอมพิวเตอร์มีการต้องการข้อมูลจากการแปลงก็จะสั่งงานให้การ์ดส่งข้อมูลของจำนวนหน่วยความจำที่ใช้(สัญญาณที่ 9)มาอ่านก่อน จากนั้นข้อมูลจากหน่วยความจำ(สัญญาณที่ 8) ก็จะถูกส่งให้คอมพิวเตอร์แสดงผลสัญญาณที่ได้ออกทางหน้าจอคอมพิวเตอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5.7 วงจรถอดรหัสตำแหน่งพอร์ท



รูปที่ 5.8 วงจรถอดรหัสตำแหน่งพอร์ท

การใช้งานการ์ดนี้เราจะเรียกใช้ผ่านทางคอมพิวเตอร์โดยเรียกผ่านจากพอร์ท 300H – 305Hวงจรถอดรหัสพอร์ทจะทำตรวจสอบว่ามีการเรียกใช้พอร์ท 300H – 305H ในช่วงนี้หรือไม่ ถ้ามีก็จะทำการส่งการทำงานให้แก่ภาคอื่นๆต่อไป ซึ่งการเรียกใช้งานพอร์ทต่างๆนั้นมีความหมายดังตารางที่ 5.2

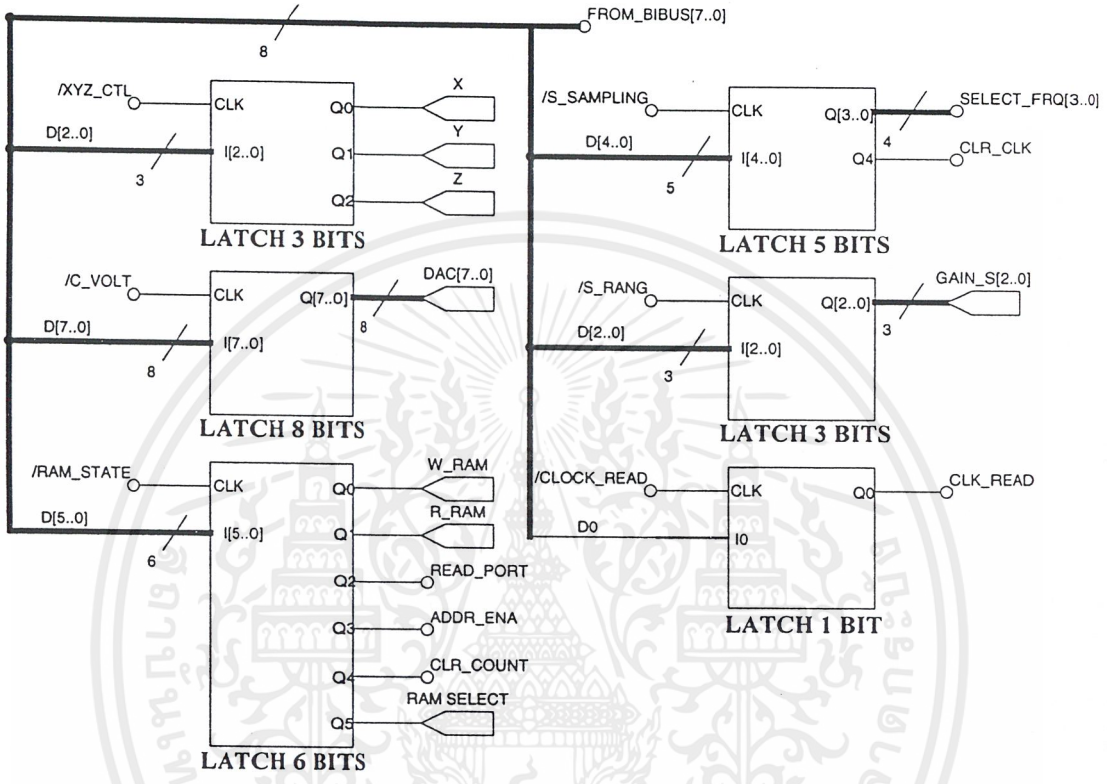
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 5.2 แสดงความหมายของพอร์ทต่างๆ

หมายเลขพอร์ท	สถานะของ /IOR	สถานะของ /IOW	ความหมาย
300H	1	0	เลือกอัตราการลดทอน และโหมด
301H	1	0	เลือกจำนวนเท่าของอัตราการขยาย
302H	1	0	เลือกระดับการยกสัญญาณ
303H	1	0	เลือกความถี่ที่ใช้ในการแซมปลิง
304H	1	0	กำหนดการทำงานของหน่วยความจำ
305H	1	0	กำหนดคล๊อกของการอ่านข้อมูล
300H	0	1	อ่านจำนวนแอดเดรสหน่วยความจำ
301H	0	1	อ่านข้อมูลจากหน่วยความจำ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5.8 วงจรควบคุมส่วนต่างๆของระบบ

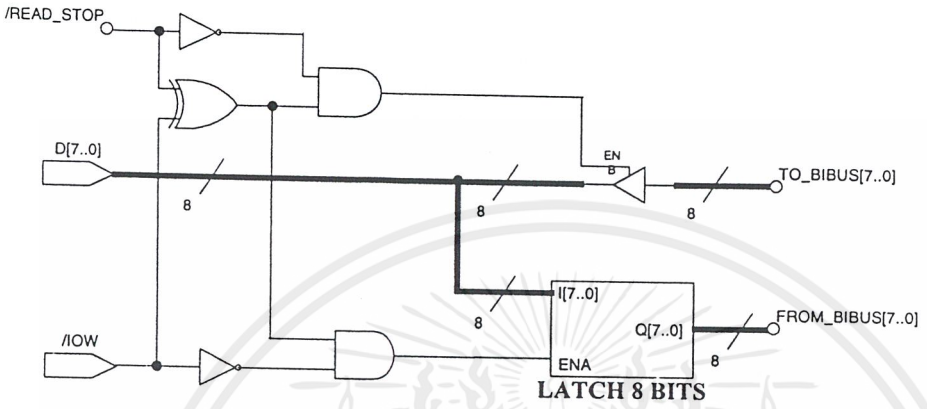


รูปที่ 5.9 วงจรควบคุมส่วนต่างๆของระบบ

ที่วงจรถอดรหัสพอร์ทจะส่งข้อมูลการเลือกการทำงานให้แก่วงจรควบคุมส่วนต่างๆของระบบ เพื่อทำการควบคุมการทำงานทั้งส่วนที่เป็นดิจิตอล และอนาลอก ซึ่งส่วนที่เป็นแลช์ทจะเป็นตัวแสดงสถานะของระบบ และทำให้ระบบทราบว่าขณะนี้เวลานี้ควรทำงานอย่างไร ฉะนั้นการเปลี่ยนแปลงการทำงานของระบบนั้นจะเปลี่ยนแปลงเมื่อมีสัญญาณพัลซ์จากวงจรถอดรหัสพอร์ท

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5.9 วงจรควบคุมข้อมูลแบบ 2 ทิศทาง

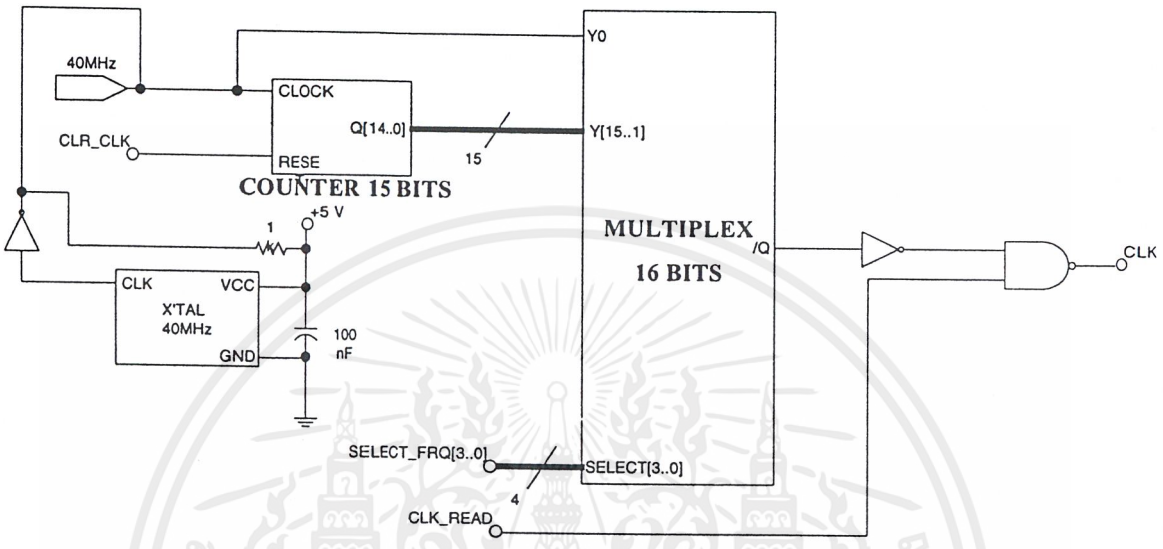


รูปที่ 5.10 วงจรควบคุมข้อมูลแบบ 2 ทิศทาง

วงจรควบคุมข้อมูลแบบ 2 ทิศทางนี้จะทำการเชื่อมต่อกับบัสข้อมูล(Data Bus)ของคอมพิวเตอร์ และเป็นตัวควบคุมทิศทางการส่งข้อมูลของคอมพิวเตอร์เพิ่มไม่ให้ชนกับข้อมูลภายในและภายนอกของคอมพิวเตอร์ ซึ่งการควบคุมนั้นจะอยู่ภายใต้การควบคุมของการเขียน และการอ่านบัสข้อมูลของคอมพิวเตอร์ และวงจรถอดรหัสพอร์ทของการ์ด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5.10 วงจรกำเนิดสัญญาณนาฬิกาหลายความถี่



รูปที่ 5.11 วงจรกำเนิดสัญญาณนาฬิกาหลายความถี่

วงจรเริ่มทำงานจากการได้รับคล็อกพัลซ์จากวงจรกำเนิดสัญญาณของคลิสตัลขนาด 40 MHz ทำให้วงจรนับขนาด 15 บิททำงาน ส่วนเอาท์พุทจะถูกเลือกผ่านทาง วงจรมัลติเพลกซ์ขนาด16 บิท ซึ่งวงจรกำเนิดสัญญาณนาฬิกาหลายความถี่ก็จะเปรียบเหมือนทำงานเป็นวงจรหารความถี่ 40MHz ขนาด 16 บิทนั่นเอง ซึ่งสามารถเลือกความถี่ที่ต้องการได้จากวงจรมัลติเพลกขนาด 16 บิทและผลลัพธ์ของการหารความถี่นั้นเป็นไปดังตารางที่ 5.3

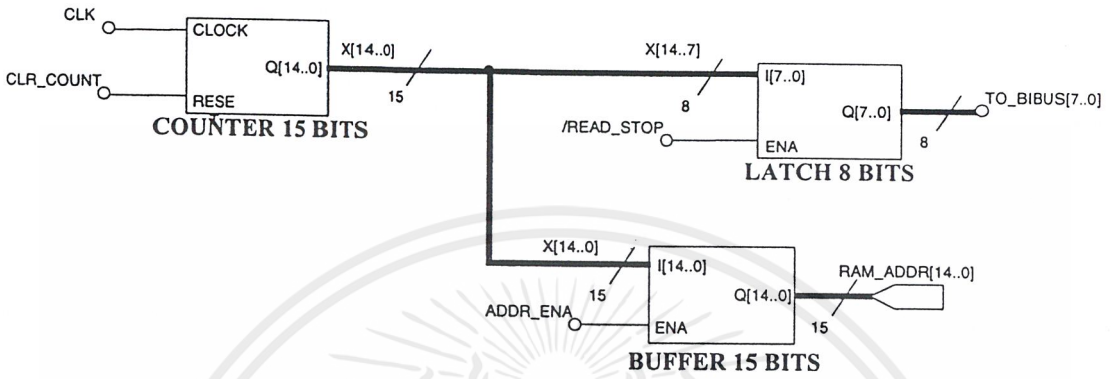
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 5.3 ความถี่ที่ได้จากวงจรกำเนิดสัญญาณนาฬิกาหลายความถี่

Select3	Select2	Select1	Select0	ความถี่เอาท์พุท(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

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5.11 วงจรนับตำแหน่งหน่วยความจำ

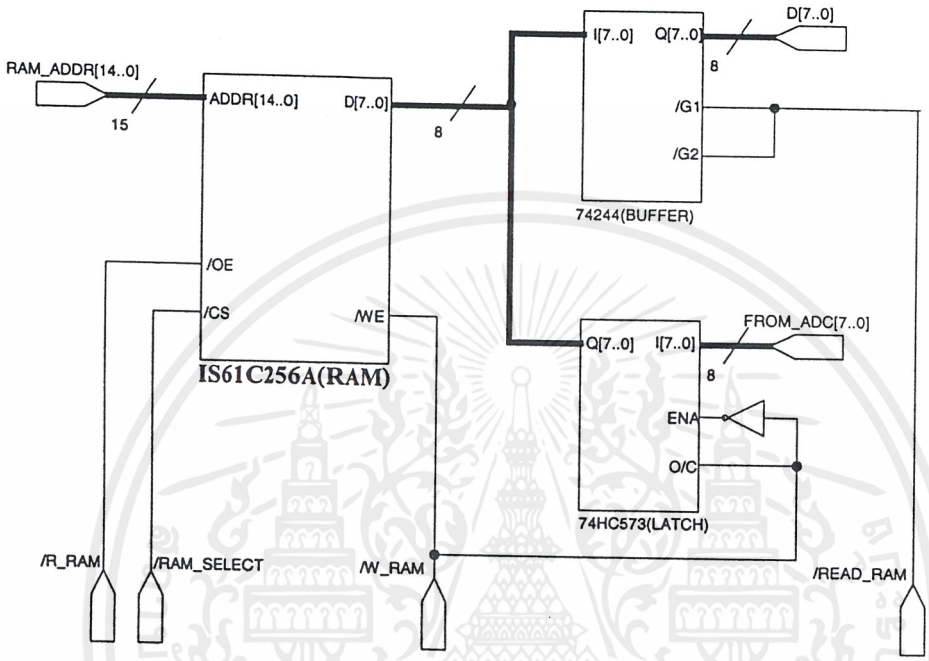


รูปที่ 5.12 วงจรนับตำแหน่งหน่วยความจำ

วงจรนับตำแหน่งหน่วยความจำนี้จะเป็นวงจรนับขนาด 15 บิทโดยคล็อกของการนับนั้นจะได้จากสัญญาณแซมปลิงที่เกิดจากวงจรกำเนิดสัญญาณหลายความถี่ เอาท์พุทของวงจรจะมี 2 ส่วน คือ ส่วนของตำแหน่งแอดเดรส(เอาท์พุทของบัฟเฟอร์ขนาด 8 บิท) และส่วนของการอ่านตำแหน่งหน่วยความจำ(เอาท์พุทของแลท์ชขนาด 8 บิท)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5.12 วงจรหน่วยความจำ



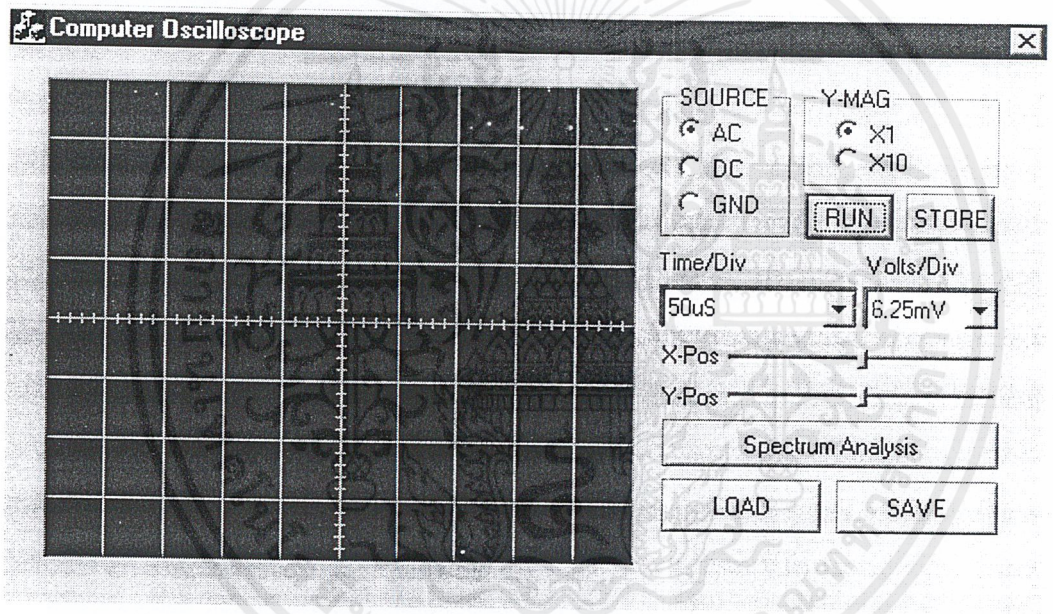
รูปที่ 5.13 วงจรหน่วยความจำ

วงจรหน่วยความจำจะอยู่ภายนอกเอฟพีจีเอ ข้อมูลในหน่วยความจำได้จากการแปลงสัญญาณของวงจรแปลง อนาลอกเป็นดิจิตอล โดยผ่านวงจรแลท์ชก่อนที่ข้อมูลจะถูกเก็บในหน่วยความจำ และเมื่อต้องการอ่านข้อมูลจากหน่วยความจำก็จะให้หน่วยความจำนั้นส่งข้อมูลผ่านวงจรบัฟเฟอร์ออกมาที่ไอซาบัส จุดสังเกตคือวงจรบัฟเฟอร์ และวงจรแลท์ชจะทำงานไม่พร้อมกัน โดยจะสลับกันปิดเปิดการส่งข้อมูลตามการเขียนและอ่านหน่วยความจำ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5.13 การเขียนโปรแกรม

คอมไพเลอร์(Compiler) ที่ใช้ในการเขียนโปรแกรมควบคุมการทำงานนั้นใช้ไมโครซอฟท์ วิชวลซีพลัสพลัส(Microsoft Virtual C++) เนื่องจากกลุ่มผู้พัฒนาได้อาศัยความรู้พื้นฐานมาจาก ภาษาซีพลัสพลัส(C++) และข้อสำคัญอีกประการหนึ่งภาษาซีพลัสพลัสนั้นมีการทำงานที่เป็นโครงสร้างทำงานได้อย่างรวดเร็ว อีกทั้งด้วยเทคโนโลยีออปเจคโอเรี่ยนเทดโปรแกรมมิ่ง(OOP : Object Oriented Programming) ช่วยให้การพัฒนาในอนาคตมีความสะดวกมากยิ่งขึ้น



รูปที่ 5.14 แสดงโปรแกรมคอมพิวเตอร์ ออสซิลโลสโคป

จากโปรแกรมการใช้งานเริ่มต้นเราจะต้องกำหนดค่าคุณสมบัติต่างๆให้กับตัวการ์ดโดยสั่งงานผ่านทางโปรแกรม อันดับแรกของการกำหนดค่านั้นเราจะเลือกที่ฟังก์ชันซอร์ซ(SOURCE) ให้ทำงานในโหมดใดโหมดหนึ่งถ้าเลือกในโหมดเอซี(AC) วงจรจะทำการวัดสัญญาณไฟสลับโดยจะไม่สนใจระดับของไฟตรงแต่อย่างใด หากเลือกในโหมดดีซี(DC) วงจรจะทำการตรวจวัดคล้ายกับการวัดสัญญาณไฟสลับเพียงแต่จะต่างกันที่ในโหมดดีซีนี้จะมีการตรวจสอบระดับของสัญญาณไฟตรงด้วย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โหมดกราวด์(GND : GROUND) จะเป็นการวัดระดับของสัญญาณกราวด์ที่ได้จากการแปลงสัญญาณอนาลอกเป็นสัญญาณดิจิตอล

ลำดับต่อไปจะเป็นการกำหนดอัตราการลดทอนของวงจรกำหนดได้จากฟังก์ชันแอทเทนนูเอท (ATTENUATE) ซึ่งสามารถเลือกอัตราส่วนการลดทอนได้ 2 แบบคือ 1/1(ไม่มีการลดทอน) และ1/10 (ลดทอนด้วยอัตราส่วน 10 เท่า) ผู้ใช้สามารถสั่งงานให้วงจรเริ่มทำงานได้โดยคลิกที่ปุ่มอินพุทซอร์ซ (INPUT SOURCE)

ส่วนการปรับอัตราการแซมปลิงสามารถกำหนดได้จากฟังก์ชันแซมปลิงฟรีเควนซี่(Sampling Frequency)ซึ่งผู้ใช้สามารถเลือกความถี่แซมปลิงได้จากรายการที่ได้กำหนดเอาไว้ ฟังก์ชันเกนแอดจัส (Gain Adjust) จะเป็นการกำหนดว่าต้องการขยายสัญญาณที่ป้อนเข้ามายังวงจรให้มีเป็นเท่าใดตามอัตราส่วนจำนวนเท่าที่กำหนดเอาไว้ในรายการ

ฟังก์ชันแคมปลิงคอนโทรล(Clamping Control) จะเป็นการกำหนดระดับของสัญญาณที่จะวัดโดยสามารถปรับระดับได้โดยการเลื่อนที่สไลด์บาร์(Slide Bar) ผู้ใช้สามารถสั่งงานให้วงจรเริ่มทำงานได้โดยคลิกที่ปุ่มแคมปิง แอน แซมปลิง(CLAMPING AND SAMPLING)

เมื่อเราต้องการที่จะแสดงผลข้อมูลที่อ่านจากหน่วยความจำให้ทำการคลิกที่ปุ่มโหลดทูพล๊อต (LOAD TO PLOT) ข้อมูลจะแสดงผลออกมาทางจอภาพโดยรูปสัญญาณที่ได้จะแสดงในส่วนด้านซ้ายมือ และข้อมูลทางดิจิตอลในหน่วยความจำจะถูกแสดงที่ฝั่งขวามือซึ่งจะแสดงออกมาเป็นค่าต่างๆในแอดเดรสของหน่วยความจำเรียงกันไปตามลำดับ และหากเมื่อใดที่เราต้องการดูการเปลี่ยนแปลงของสัญญาณอีกครั้งให้ทำการคลิกที่ปุ่มโหลดทูพล๊อตซึ่งที่หน้าจอจะมีการแสดงค่าของสัญญาณที่เปลี่ยนแปลงในขณะนั้น หากต้องการลบข้อมูลที่แสดงผลก็สามารถทำได้คลิกที่ปุ่มเคลียร์(CLEAR) ข้อมูลต่างๆที่แสดงผลก็จะถูกลบออกในทันที

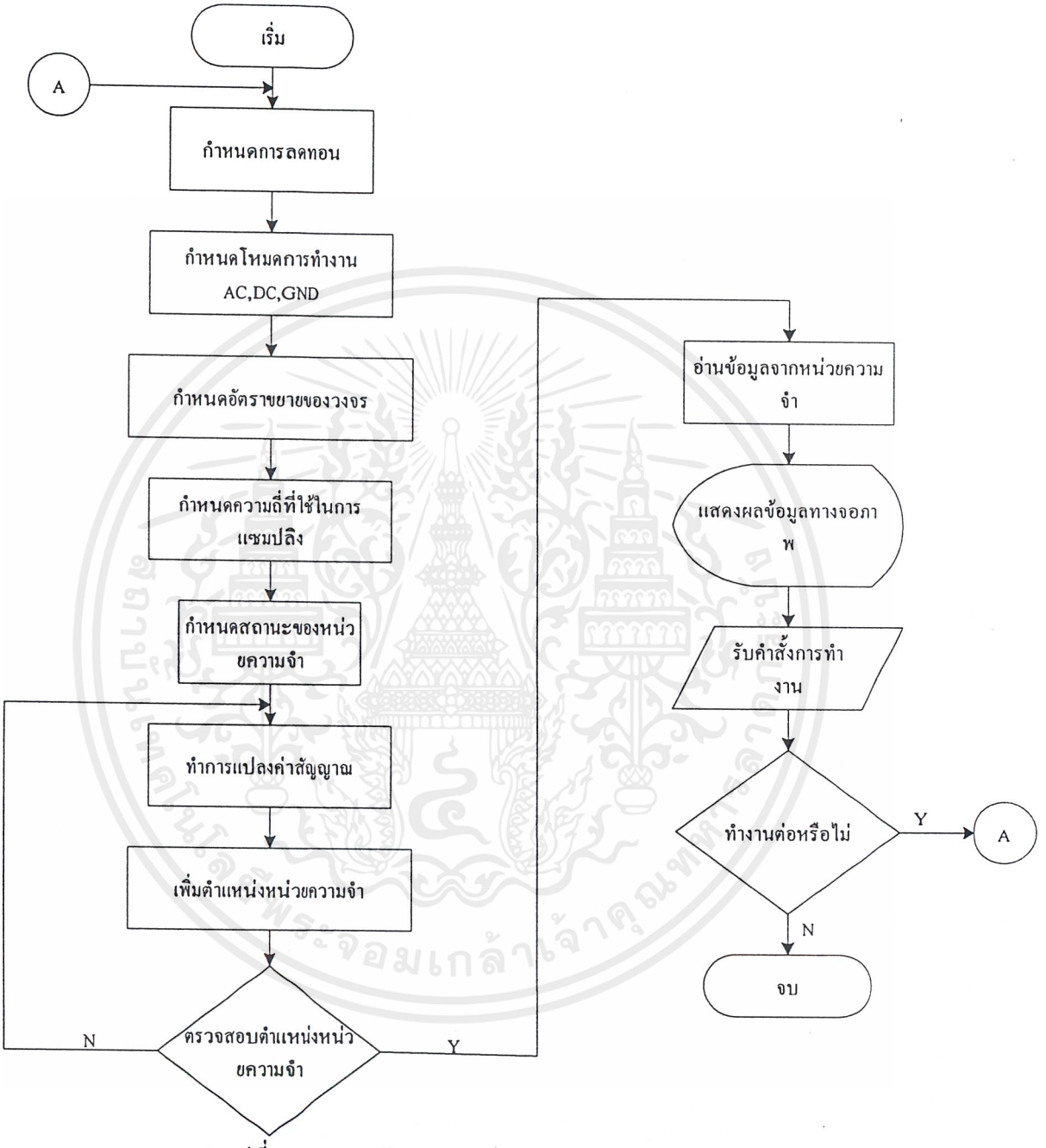
5.14 การออกแบบโปรแกรม

เริ่มต้นของการออกแบบนั้นเราจะให้โปรแกรมสั่งงานออกมาที่พอร์ทต่างๆตามที่กล่าวไว้ในขั้นต้น เพื่อกำหนดค่าเริ่มต้นของการทำงานให้แก่ระบบ (พิจารณาจากรูปที่ 5.15 ที่ผังงานนั้นในช่วงเริ่มต้นเราจะกำหนดค่าเริ่มต้นให้แก่ระบบ) จากนั้นเมื่อทุกอย่างพร้อมแล้วลำดับต่อไปเราจะสั่งงานเปิดหน่วยความจำให้รับข้อมูลจากการแปลงค่าเก็บมาไว้ในหน่วยความจำพร้อมๆกับทำการเลื่อนตำแหน่งหน่วยความจำเพื่อรับข้อมูลต่อไป ซึ่งจะมีการตรวจสอบไปพร้อมๆกันว่ามีการเก็บข้อมูลไว้เป็นจำนวน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เท่าไร และครบตามจำนวนแล้วหรือไม่ เมื่อครบตามจำนวนที่กำหนดไว้แล้วก็จะให้คอมพิวเตอร์อ่านข้อมูลจากหน่วยความจำพร้อมทั้งแสดงผลที่ได้ออกมาที่หน้าจอคอมพิวเตอร์ จากนั้นก็ตรวจสอบว่าผู้ใช้ต้องการใช้งานโปรแกรมต่อไปอีกหรือไม่หากไม่ก็ให้ออกจากโปรแกรมแต่ถ้าต้องการทำงานต่อระบบการทำงานก็จะกลับมาเริ่มทำงานตั้นแต่การทำงานที่จุดเริ่มต้นจนถึงจุดสุดท้ายของการทำงาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 5.15 แสดงผังงานการเขียนโปรแกรมควบคุม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 6
การทดสอบการทำงาน

6.1 วัตถุประสงค์

1. เพื่อเป็นการทดสอบการทำงานของวงจรในส่วนต่างๆ
2. เพื่อเป็นการปรับปรุงและแก้ไขข้อบกพร่องของวงจรต่างๆ
3. สร้างฮาร์ดแวร์ และ ซอฟแวร์ สำหรับใช้ในการประยุกต์คอมพิวเตอร์ให้สามารถทำงานเป็นสตอเรจออสซิลโลสโคปได้

6.2 การทดลองวงจรลดทอน

1. ต่อวงจรตามรูปที่ 5.2
2. จ่ายไฟให้กับวงจรวงจรลดทอน
3. ป้อนแรงดันอินพุทมีค่ามากกว่า 8.6 V_{p-p} แล้วทำการวัดที่เอาท์พุท 1/1สังเกตผลการทดลอง
4. ป้อนแรงดันอินพุทมีค่ามากกว่า 86 V_{p-p} แล้วทำการวัดที่เอาท์พุท 1/10สังเกตผลการทดลอง
5. ป้อนแรงดันอินพุทมีค่าเท่ากับ 10 V_{DC} ทำการปรับ VR_1 จนทำให้สัญญาณที่เอาท์พุท 1/10มีค่าเท่ากับ 1V
6. ป้อนสัญญาณอินพุทมีค่าเท่ากับ 10_{VP-P} ความถี่ DC – 10 MHz ทำการปรับ VC_1 และ VC_2 จนแรงดันเอาท์พุท 1/10 มีค่าเท่ากับ $1V_{p-p}$ ในทุกๆค่าความถี่

สรุปผลการทดลอง

จากการทดลองที่3 เราสรุปได้ว่าถ้าเราป้อนแรงดันอินพุทเกิน 8.6 V_{p-p} , $86V_{p-p}$สัญญาณที่วัดได้ที่เอาท์พุท 1/1และ 1/10 จะถูกคลิบไม่สามารถใช้งานได้ ดังนั้นเฉพาะวงจรลดทอนจะรับอินพุทได้ไม่เกิน$86V_{p-p}$ แต่เนื่องจากข้อจำกัดของ A/D Converter ที่ต้องการอินพุทอยู่ระหว่าง 1.55V – 3.26V ดังนั้นเอินพุทสวิงสูงสุดมีค่าไม่เกิน17.1 V_{p-p} แต่จากการทดลองเราพบว่าอินพุทมีค่าสูงสุดเท่ากับ $12V_{p-p}$

จากการทดลองที่ 5 เป็นการทดลองเพื่อปรับจูนวงจรให้มีอัตราการลดทอนเท่ากับ 1/10 ส่วนในการทดลองที่ 6 เป็นการทดลองเพื่อทำการปรับจูนให้อัตราการลดทอนมีค่าคงที่เท่ากับ 1/10

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

6.3 การทดลองวงจรบัฟเฟอร์และวงจรเลือกความถี่

1. ต่อวงจรตามรูปที่ 5.3

2. จ่ายไฟให้กับวงจรบัฟเฟอร์และวงจรเลือกความถี่

3.ที่ อินพุท 1/1และ1/10 ป้อนแรงดัน1V_{p-p} ความถี่ 1KHz จากนั้นทดสอบการทำงานของอิเล็กทรอนิกส์สวิตช์ซึ่งทำหน้าที่เป็นสวิตช์เลือกโหมดการทำงานของวงจร

4. ปรับความต้านทาน VR_2 ให้วงจรบัฟเฟอร์มีอัตราขยายเท่ากับ –1 พร้อมตรวจสอบการทำงานของวงจรบัฟเฟอร์

สรุปผลการทดลอง

จากวงจรเลือกความถี่ Xclt คือส่วนควบคุมการลดทอนว่าจะเป็น 1/1 หรือ 1/10 โดยถ้า Xclt เป็น "0" เป็นการเลือก 1/1 แต่ถ้าเป็น "1" จะเป็นการเลือก 1/10

Yclt คือส่วนควบคุมการเลือกสัญญาณการวัดว่าจะเป็น AC หรือ DC ถ้า Yclt เป็น "0" จะเป็นการเลือกสัญญาณ AC แต่ถ้าเป็น "1" จะเป็นการเลือกสัญญาณ DC

Zclt คือส่วนควบคุมการเลือกสัญญาณการวัดว่าจะเป็นสัญญาณหรือว่าจะเป็นกราวด์ โดยถ้า Zclt มีค่าเป็น"0" จะเป็นการเลือกกราวด์แต่ถ้า Zclt มีค่าเป็น"1" จะเป็นการเลือกสัญญาณ

อินพุทอิมพีแดนซ์วงจรบัฟเฟอร์มีค่าเท่ากับ R4 เท่ากับ 1 M AV เท่ากับ –1 และมีเอาท์พุทอิมพีแดนซ์ต่ำมากๆ

6.4 การทดลองวงจรขยาย

1. ต่อวงจรตามรูปที่ 5.4

2. จ่ายไฟให้กับวงจรขยาย

3. ป้อนความถี่ของสัญญาณที่ DC – 10 MHz ทำการปรับจูน VC_3 เพื่อทำให้อัตราขยายคงที่ทุกๆค่าความถี่

4. ตรวจสอบการทำงานให้เป็นไปตามตารางการทดลองที่ 5.1

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สรุปผลการทดลอง

จากการทดลองเราพบว่าเมื่อสัญญาณมีความถี่สูงอัตราการขยายสัญญาณจะมีค่ามาขึ้นเราสามารถควบคุมอัตราขยายให้คงที่ได้โดยการปรับจูน VC_3 แต่การทำแบบนี้จะทำให้แบนด์วิดธ์ของวงจรขยายมีค่าลดลงเล็กน้อย จากการทดสอบเราพบว่าวงจรของเราจะยังคงทำงานได้ดีที่ความถี่ไม่เกิน 3.5 MHz

6.5 การทดลองวงจรยกระดับสัญญาณ

1. ต่อวงจรตามรูปที่ 5.5
2. จ่ายไฟให้กับวงจรยกระดับสัญญาณ
3. ตรวจสอบการทำงานว่าทำงานได้อย่างถูกต้อง

สรุปผลการทดลอง

จากการทดลองเราพบว่าแรงดันสูงที่สุดที่ขา 4 ของไอซี DAC0800 จะขึ้นอยู่กับแรงดันอ้างอิง (V_{REF}) และเราสามารถปรับแรงดันที่ขา 4 ของไอซี DAC0800 ได้ด้วยการเปลี่ยนแปลงข้อมูลที่ขา 5 ถึงขา 12 ซึ่งเป็นข้อมูลขนาด 8 บิต แรงดันที่ได้จาก DAC0800 จะใช้ในการยกระดับสัญญาณที่เราจะทำการวัดให้อยู่ในช่วงที่สามารถแปลงเป็นสัญญาณดิจิตอลได้

6.6 การทดลองวงจรแปลงสัญญาณอนาลอกเป็นสัญญาณดิจิตอล

1. ต่อวงจรตามรูปที่ 5.6
2. จ่ายไฟให้กับวงจรแปลงสัญญาณอนาลอกเป็นสัญญาณดิจิตอล
3. ป้อนสัญญาณนาฬิกาความถี่ 1 KHz ให้แก่ขา 16 ของ IC 12
4. เพิ่มค่าแรงดันดีซีขึ้นเรื่อยๆจนค่าเอาท์พุทD0–D7มีค่าเป็นFF_Hจะได้แรงดันอินพุทมีค่าเท่ากับ 3.25 V
5. ลดค่าแรงดันดีซีลงเรื่อยๆจนค่าเอาท์พุท D0 – D7 มีค่าเป็น 00_H จะได้แรงดันอินพุทมีค่าเท่ากับ 1.55 V

สรุปผลการทดลอง

จากการทดลองเป็นการหาค่าแรงดันต่ำสุดและสูงสุดที่ A/D Converter สามารถแปลงค่าออกมาเป็นข้อมูลดิจิตอลได้ ซึ่งถ้าแรงดันอินพุทมีค่ามากกว่า 3.25 V ระดับเอาท์พุทที่ได้จะม่ค่าเป็น FF_H และถ้าแรงดันอินพุทมีค่าต่ำกว่า 1.55 V ระดับเอาท์พุทที่ได้จะม่ค่าเป็น 00_H

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

6.7 การทดลองวงจรกำเนิดสัญญาณนาฬิกาหลายความถี่

1. ที่โปรแกรมคอมพิวเตอร์ออสซิลโลสโคป เลือกรายการความถี่การแซมปลิงตามตารางที่ 6.1 แล้วคลิก ที่ปุ่ม CLAMPING AND SAMPLING

2. วัดสัญญาณที่ขา 22 ของชิปเอฟพีจีเอตรวจดูความถี่แล้วบันทึกผลที่ได้ลงในตารางที่ 6.1

3. คำนวนหาเปอร์เซนต์ความผิดพลาด

ตารางที่ 6.1 แสดงการทดสอบผลของวงจรกำเนิดสัญญาณนาฬิกาหลายความถี่

ความถี่ในรายการ	ความถี่ในการวัดได้ (Hz)	เปอร์เซนต์ความผิดพลาด
40M	40M	0%
20M	20M	0%
10M	10M	0%
5M	5M	0%
2.5M	2.564M	0.0256%
1.25M	1.315M	0.052%
625k	645k	0.032%
312.5k	322k	0.0304%
156.25k	156.25k	0%
78.125k	76.923k	0.015%
39.062k	38.462k	0.015%
19.53k	19.231k	0.015%
9.76k	10k	0.0246%
4.88k	5k	0.0246%
2.44k	2.5k	0.024%
1.22k	1.25k	0.024%

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สรุปผลการทดลอง

จากตารางพบว่าค่าสัญญาณที่ได้จากการวัดมีความใกล้เคียงกับค่าที่กำหนดไว้ในรายการซึ่งเป็นค่าที่รับได้แล้วสามารถนำไปใช้งานจริง

ทดลองวัดสัญญาณโดยใช้โปรแกรมคอมพิวเตอร์ออสซิลโลสโคป

การทดลองจะแบ่งออกเป็น 2 ส่วน คือ

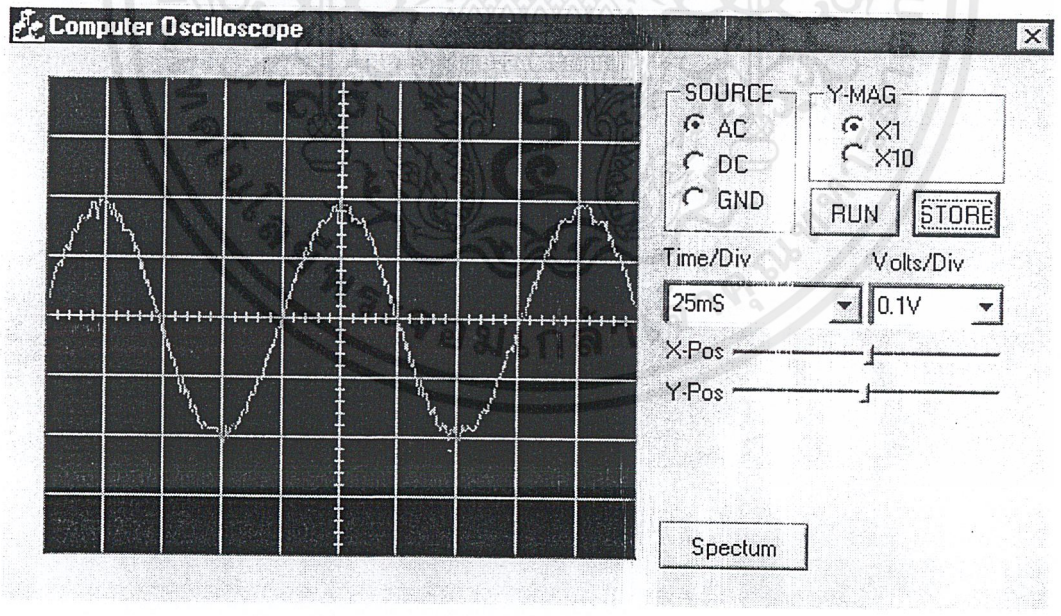
- การทดลองวัดสัญญาณที่ความถี่ต่างๆ
- การทดลองวัดสัญญาณสเปคตรัมของสัญญาณที่ความถี่ต่างๆ

6.8.1 การทดลองวัดสัญญาณที่ความถี่ต่าง ๆ

ลำดับขั้นการทดลอง

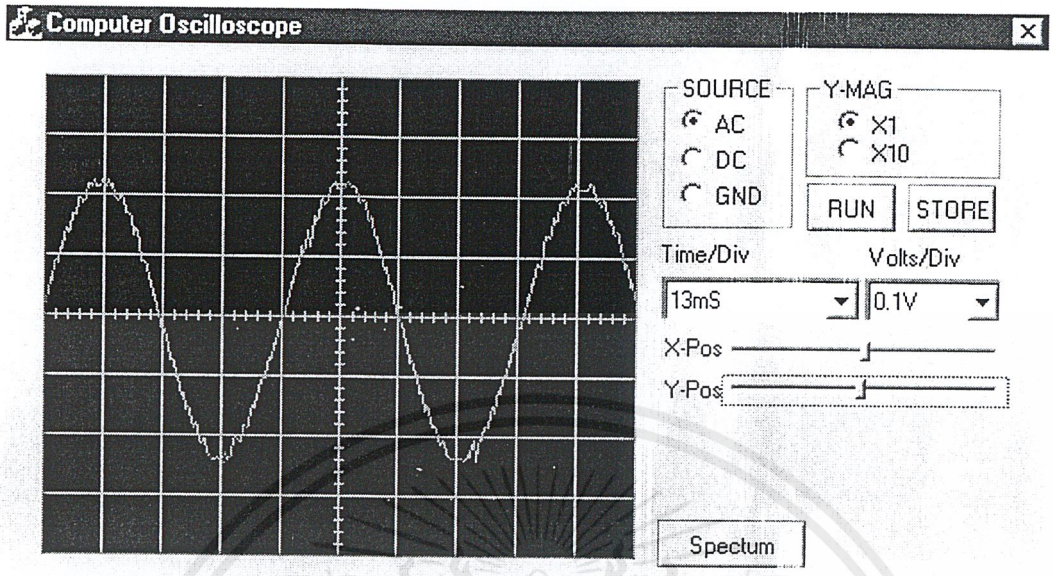
1. ป้อนสัญญาณซายน์จากเครื่องกำเนิดสัญญาณให้มีขนาดเท่ากับ 0.5 V_{P-P}
2. ทำการปรับจูนความถี่ตั้งแต่ความถี่ต่ำจนถึงความถี่สูง
3. ทำการบันทึกผลการทดลอง
5. เปลี่ยนสัญญาณจากสัญญาณซายน์ไปเป็นสัญญาณ สี่เหลี่ยม,สามเหลี่ยม และ สัญญาณ แรมป์ตามลำดับ
6. ทำการทดลองตามลำดับขั้นการทดลองที่ 1,2 และ3ตามลำดับ

ผลการทดลองวัดสัญญาณ sine

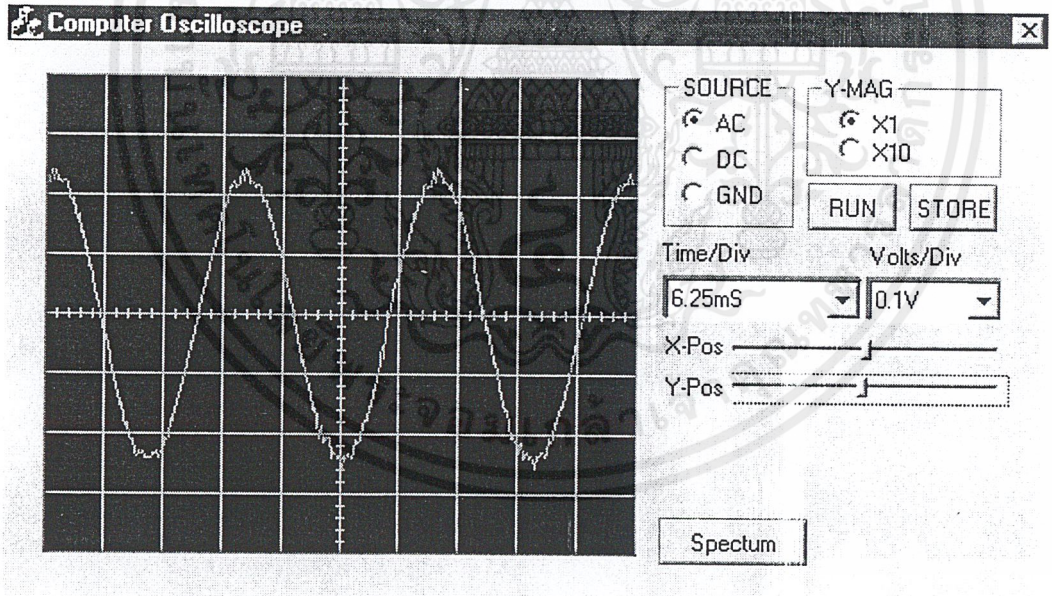


รูปที่ 6.1 ผลการวัดสัญญาณซายน์ความถี่ 10 Hz

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

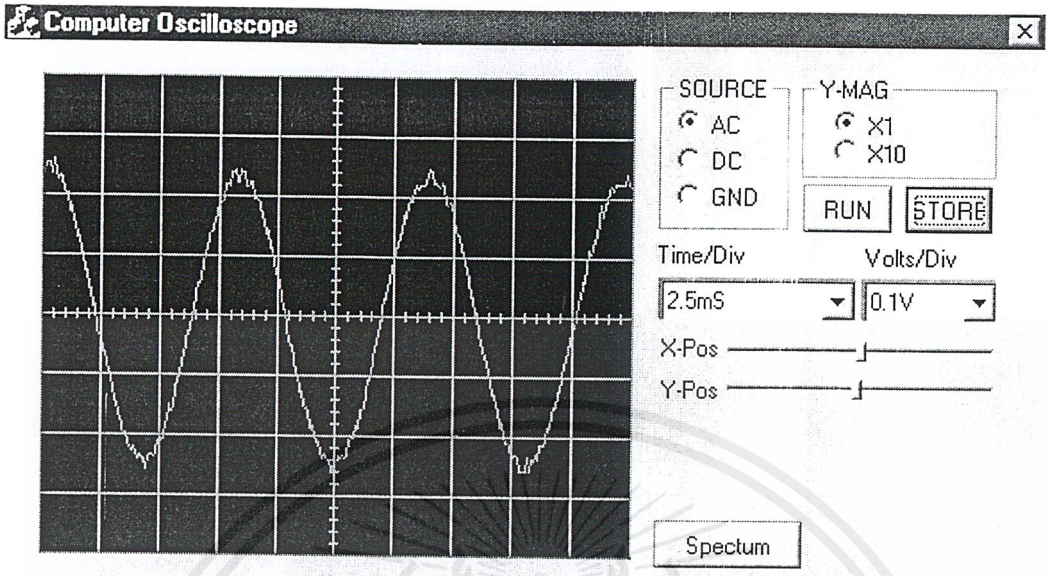


รูปที่ 6.2 ผลการวัดสัญญาณซายน์ความถี่ 20 Hz

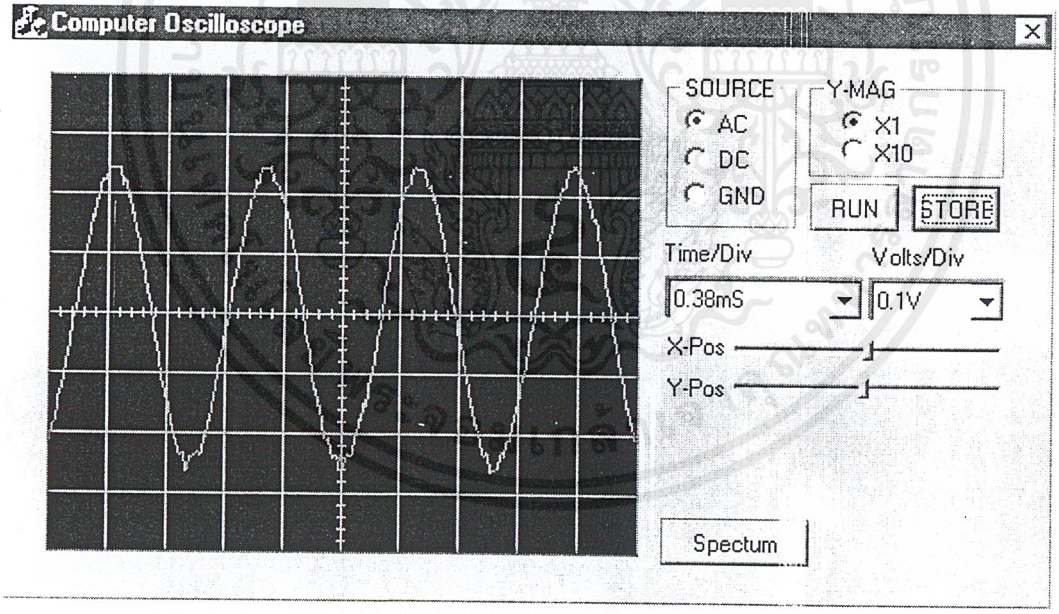


รูปที่ 6.3 ผลการวัดสัญญาณซายน์ความถี่ 50 Hz

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

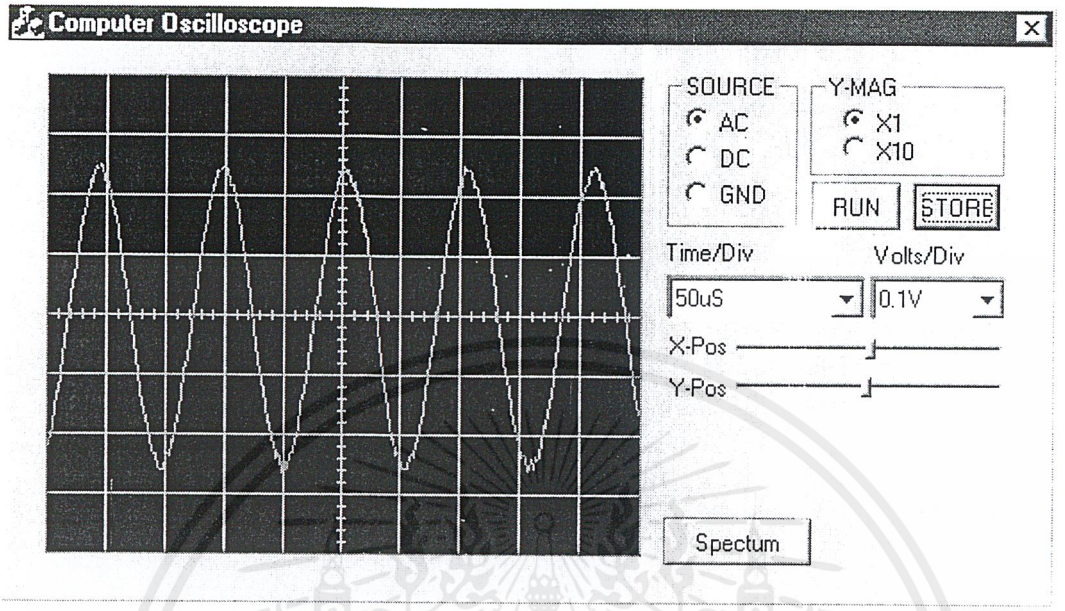


รูปที่ 6.4 ผลการวัดสัญญาณซายน์ความถี่ 100 Hz

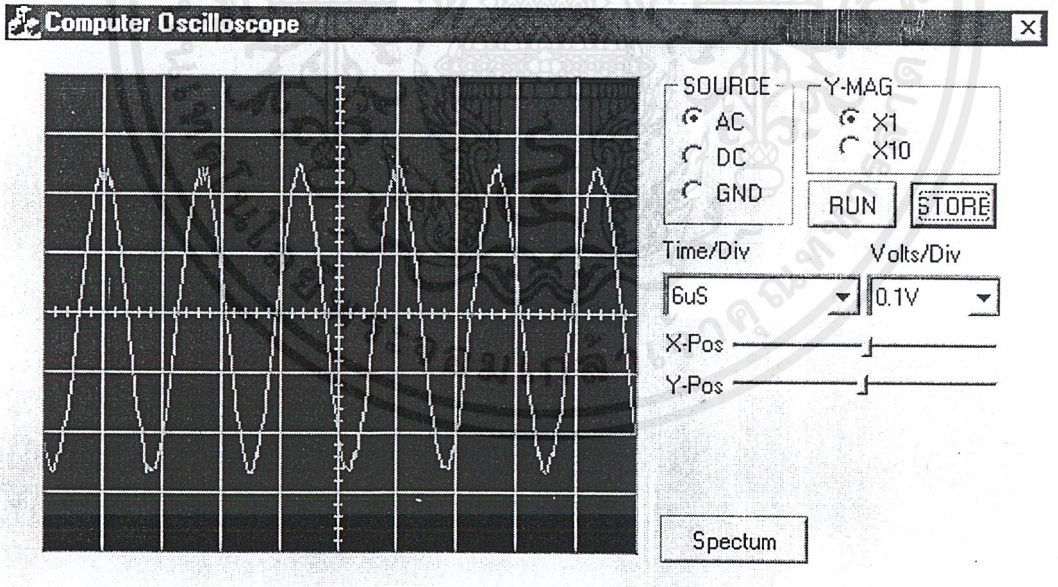


รูปที่ 6.5 ผลการวัดสัญญาณซายน์ความถี่ 1 kHz

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

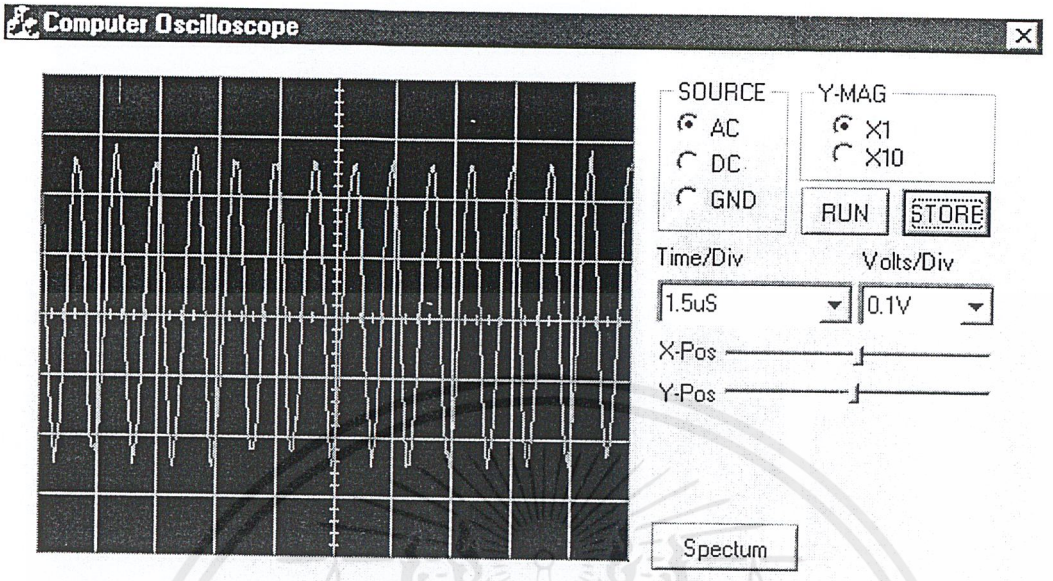


รูปที่ 6.6 ผลการวัดสัญญาณซายน์ความถี่ 10 kHz



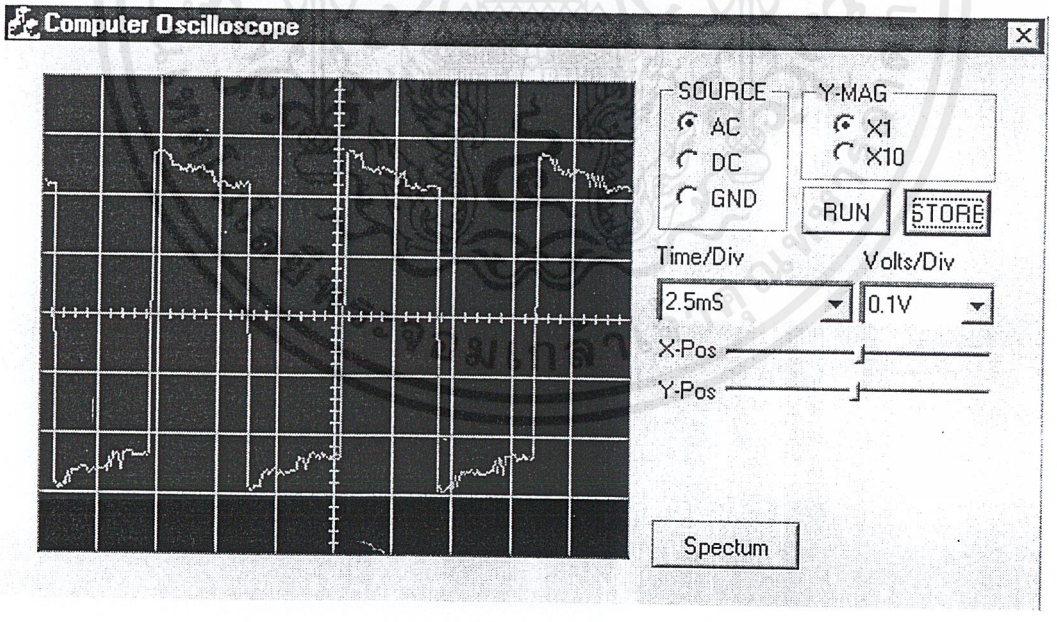
รูปที่ 6.7 ผลการวัดสัญญาณซายน์ความถี่ 100 kHz

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



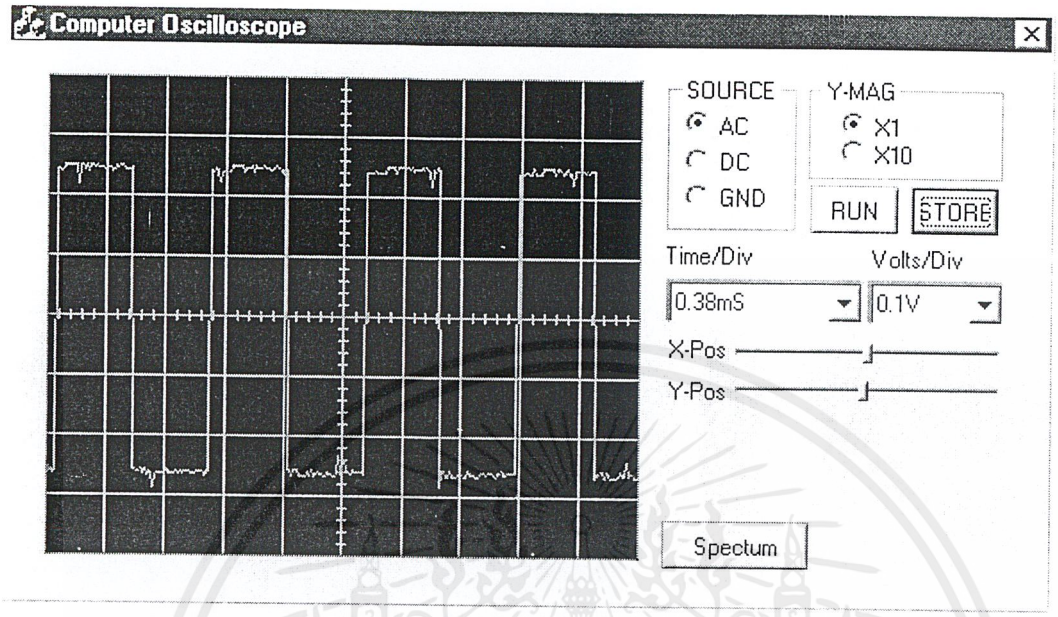
รูปที่ 6.8 ผลการวัดสัญญาณซายน์ความถี่ 1 MHz

ผลการทดลองวัดสัญญาณสี่เหลี่ยม

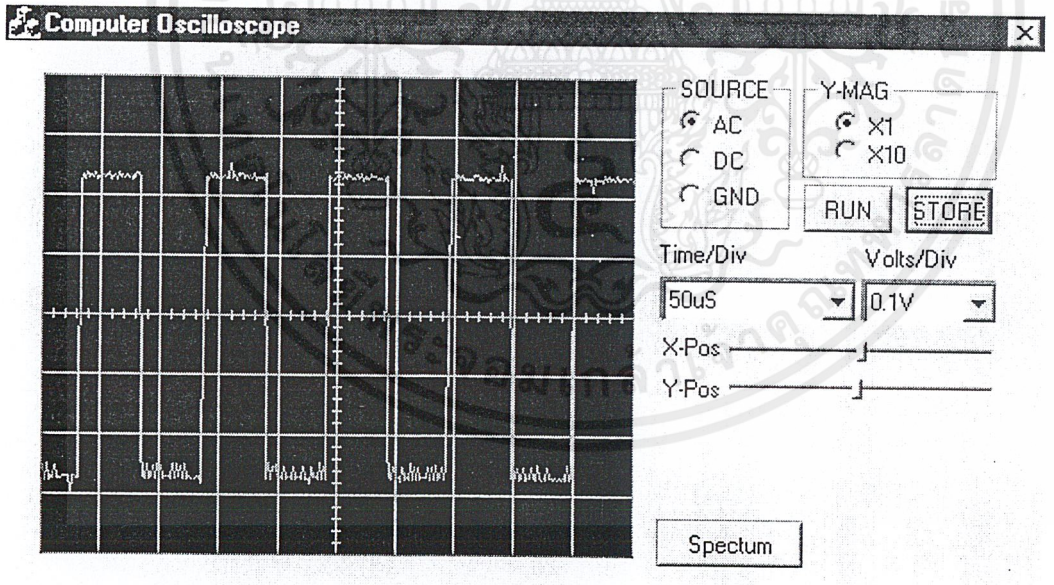


รูปที่ 6.9 ผลการวัดสัญญาณสี่เหลี่ยมความถี่ 100 Hz

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

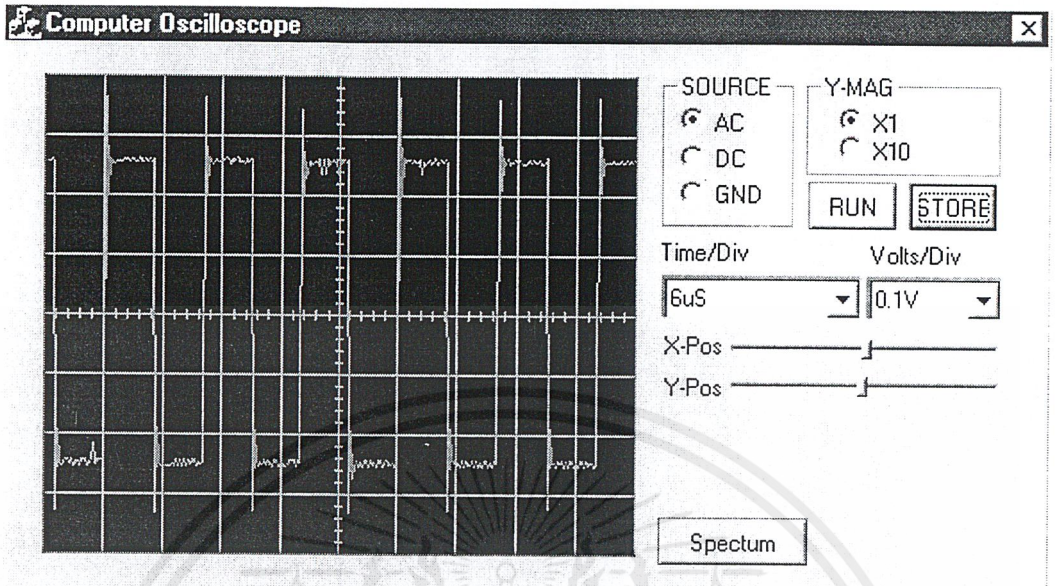


รูปที่ 6.10 ผลการวัดสัญญาณสี่เหลี่ยมความถี่ 1 kHz



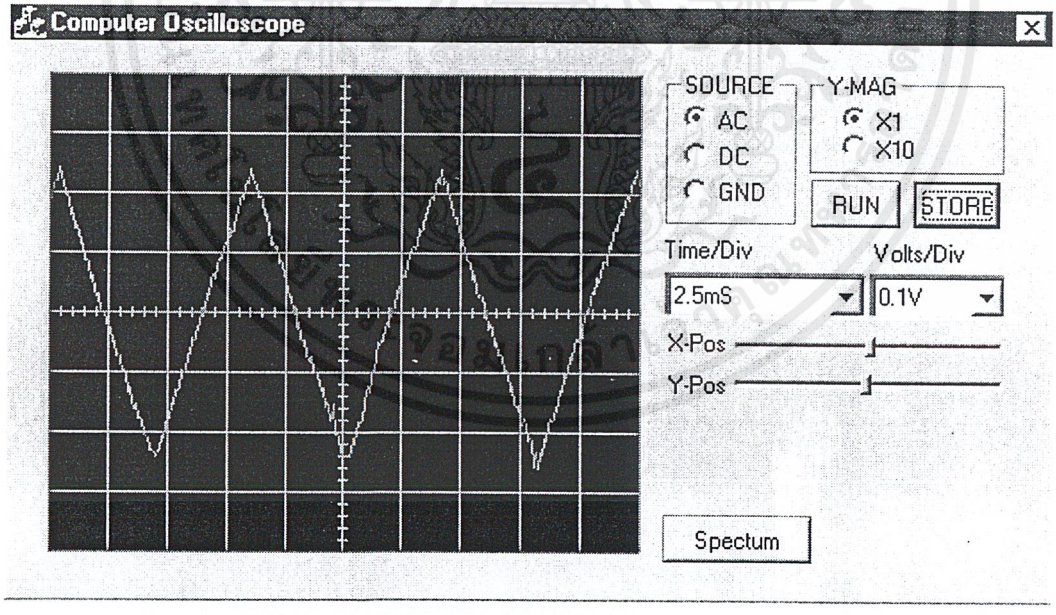
รูปที่ 6.11 ผลการวัดสัญญาณสี่เหลี่ยมความถี่ 10 kHz

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



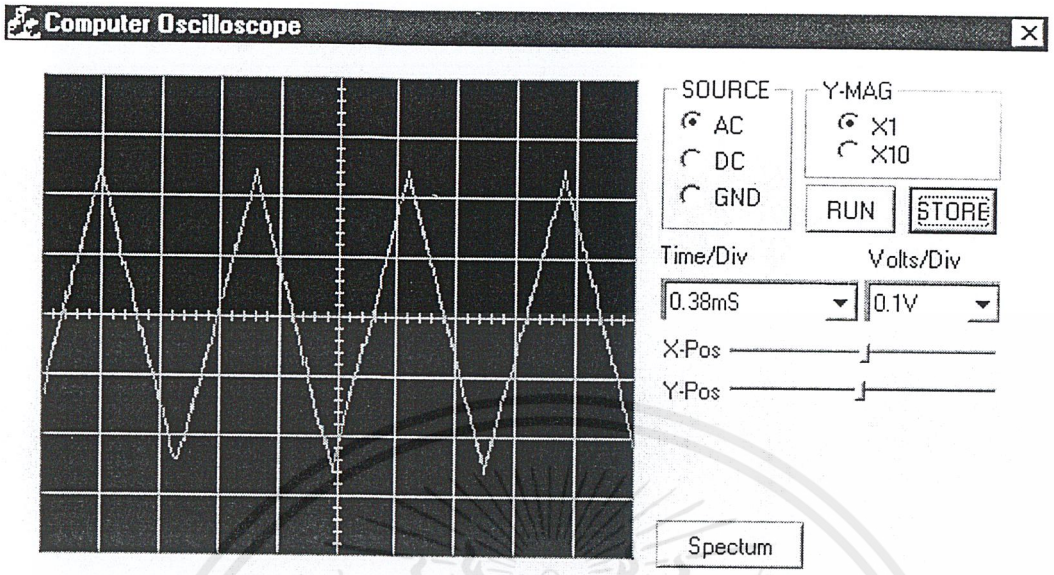
รูปที่ 6.12 ผลการวัดสัญญาณสี่เหลี่ยมความถี่ 100 kHz

ผลการทดลองวัดสัญญาณสามเหลี่ยม

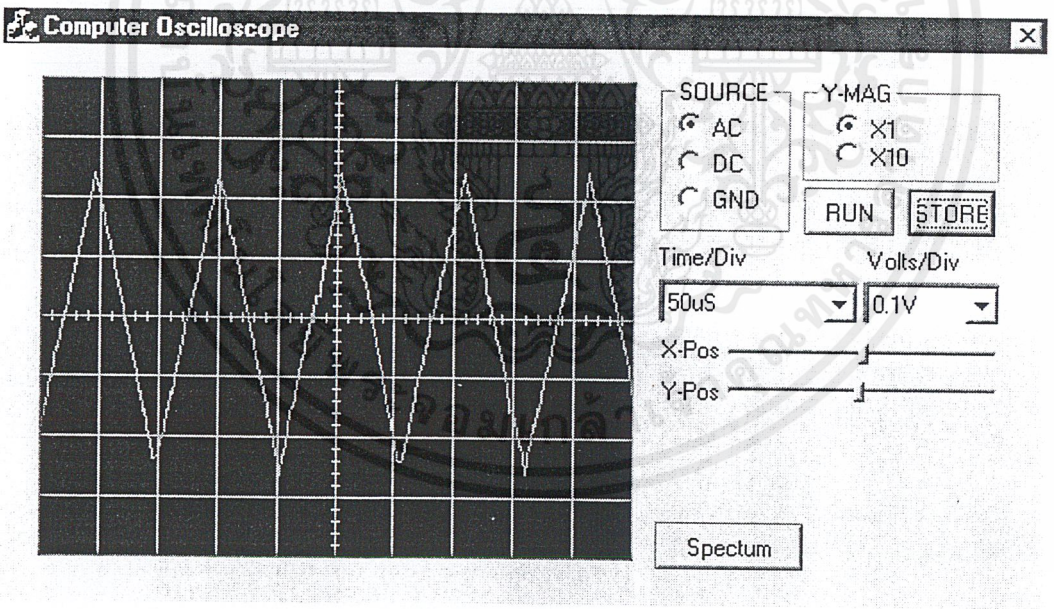


รูปที่ 6.13 ผลการวัดสัญญาณสามเหลี่ยม ความถี่ 100 Hz

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

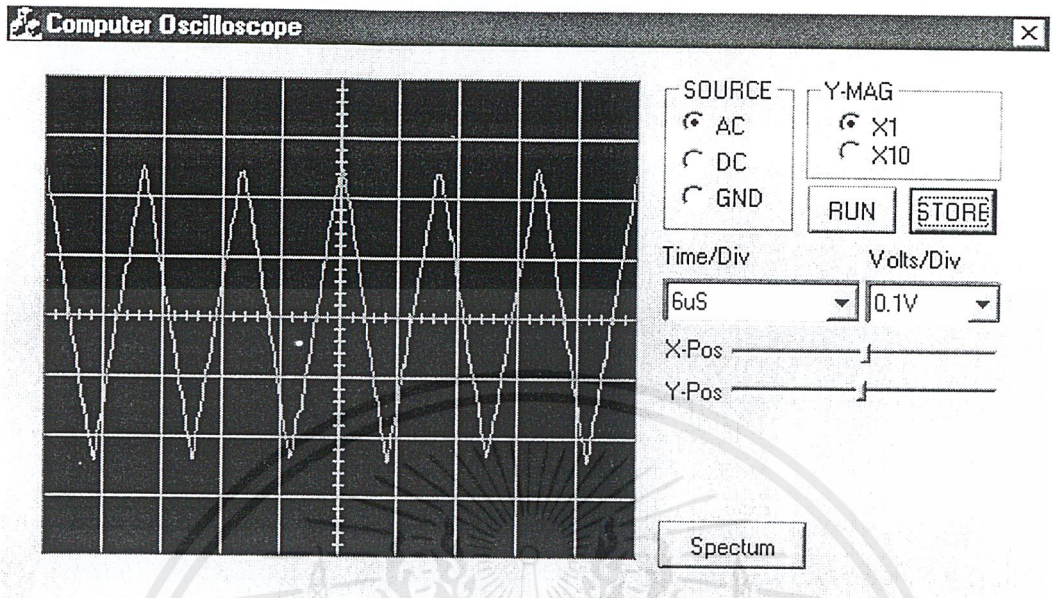


รูปที่ 6.14 ผลการวัดสัญญาณสามเหลี่ยม ความถี่ 1 kHz



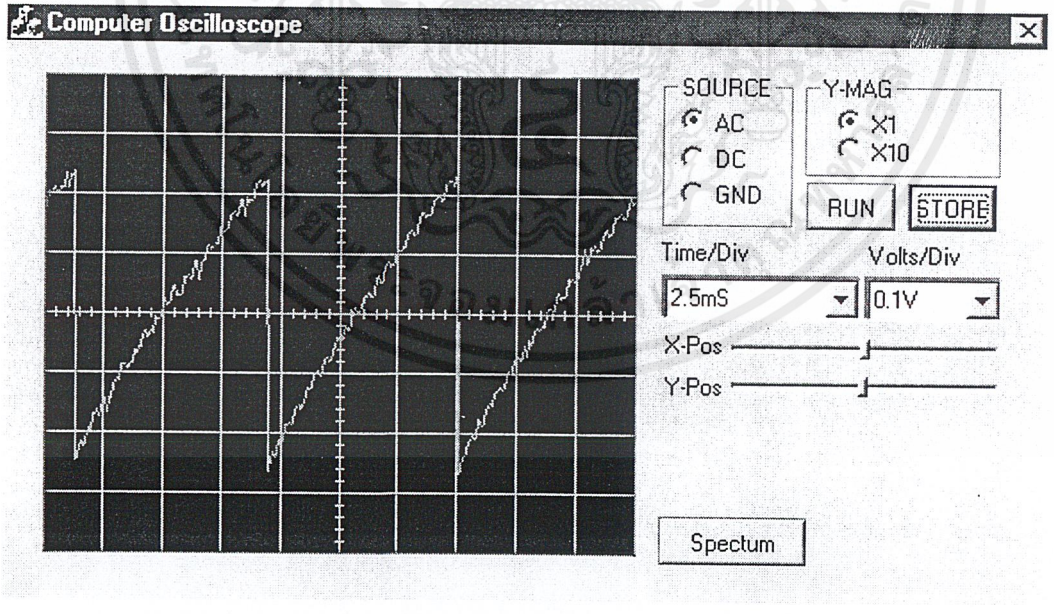
รูปที่ 6.15 ผลการวัดสัญญาณสามเหลี่ยม ความถี่ 10 kHz

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



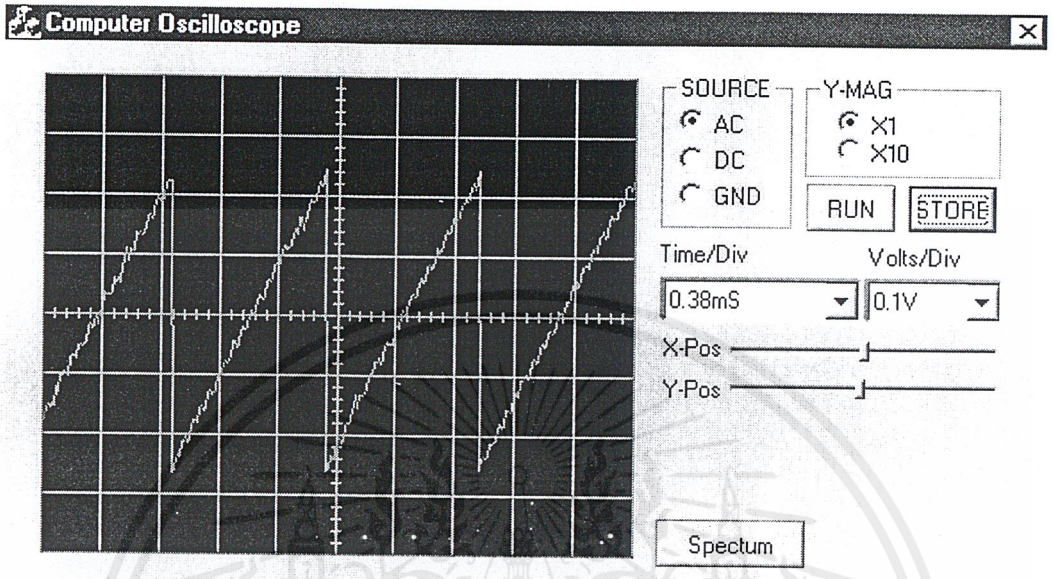
รูปที่ 6.16 ผลการวัดสัญญาณสามเหลี่ยม ความถี่ 100 kHz

ผลการทดลองวัดสัญญาณแรมป์

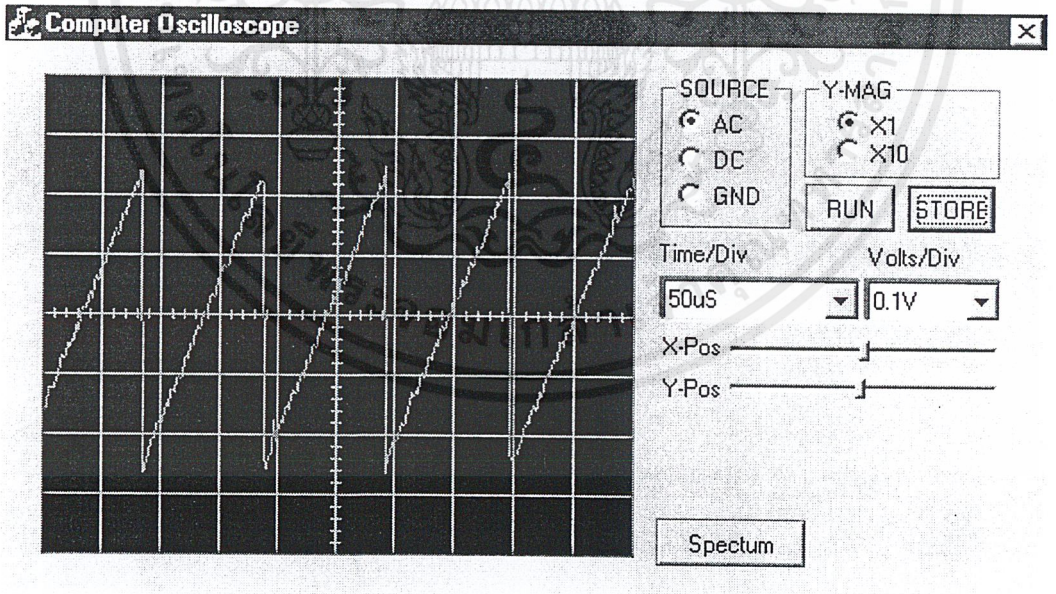


รูปที่ 6.17 ผลการวัดสัญญาณแรมป์ ความถี่ 100 Hz

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 6.18 ผลการวัดสัญญาณแรมป์ ความถี่ 1 kHz



รูปที่ 6.19 ผลการวัดสัญญาณแรมป์ ความถี่ 10 kHz

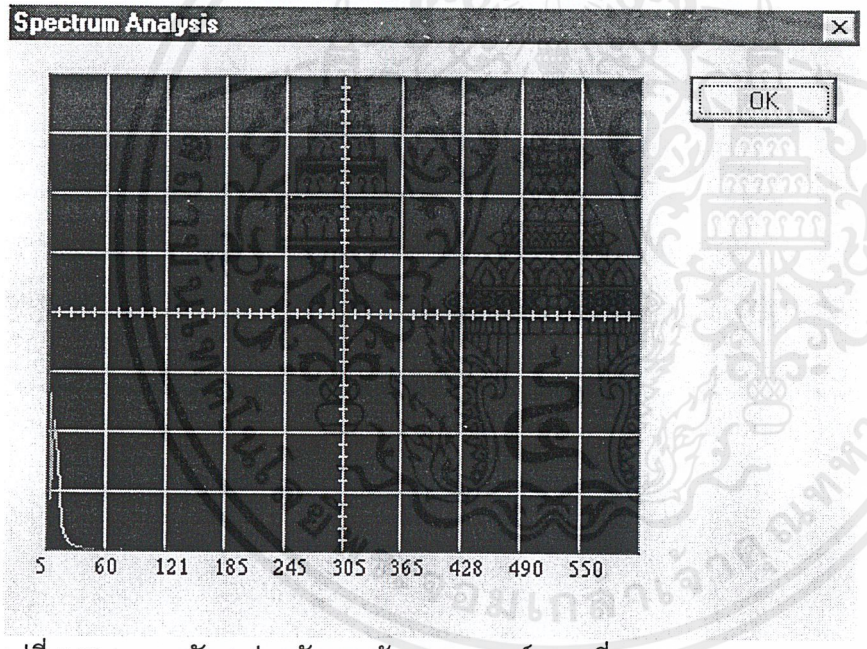
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

6.8.2 การทดลองวัดสัญญาณสเปคตรัมของสัญญาณที่ความถี่ต่าง ๆ

ลำดับขั้นการทดลอง

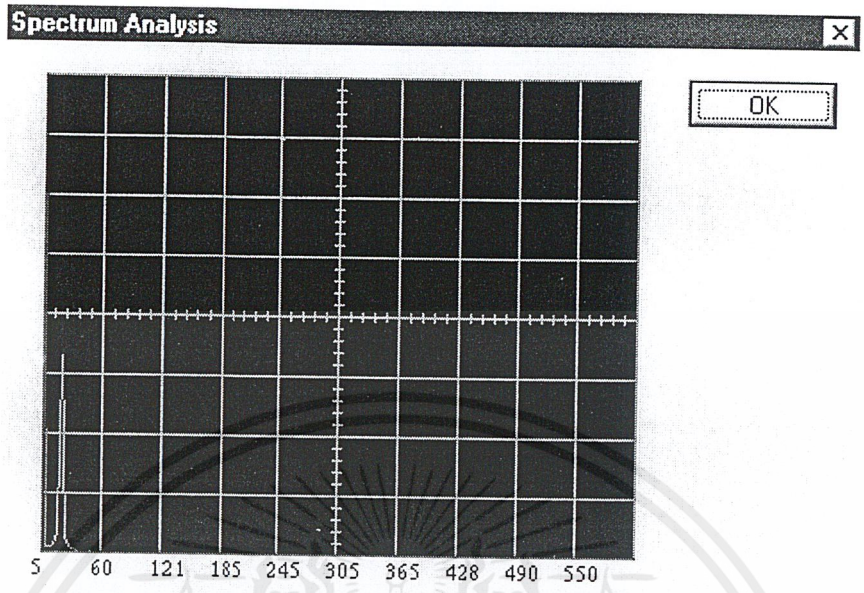
1. ป้อนสัญญาณซายน์จากเครื่องกำเนิดสัญญาณให้มีขนาดเท่ากับ 0.5 V_{P-P}
2. ทำการปรับจูนความถี่ตั้งแต่ความถี่ต่ำจนถึงความถี่สูง
3. ทำการบันทึกผลการทดลอง
5. เปลี่ยนสัญญาณจากสัญญาณซายน์ไปเป็นสัญญาณ สี่เหลี่ยม,สามเหลี่ยม และ สัญญาณ แรมป์ตามลำดับ
6. **ทำการทดลองตามลำดับขั้นการทดลองที่ 1,2 และ3ตามลำดับ**

ผลการวัดสเปคตรัมของสัญญาณซายน์

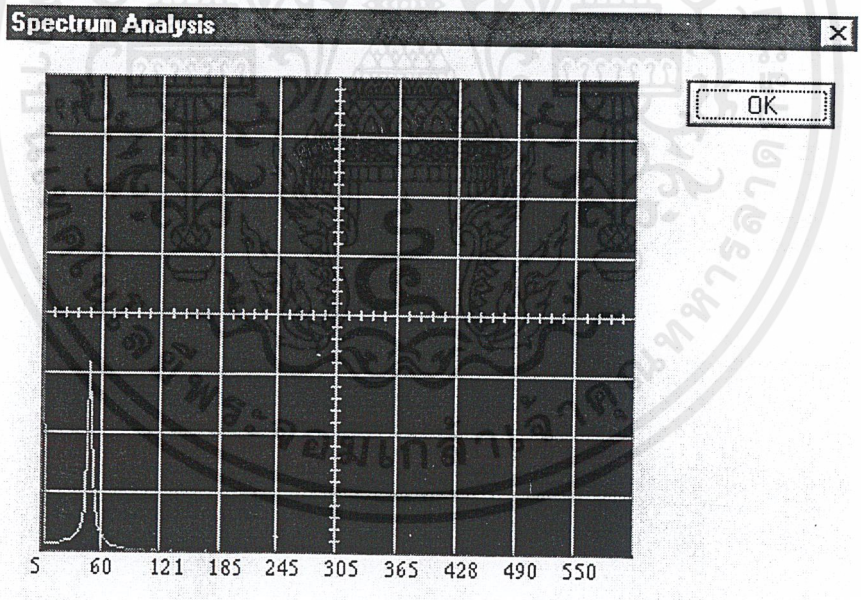


รูปที่ 6.20 **ผลการวัดสเปคตรัมของสัญญาณซายน์ความถี่** 10 Hz

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

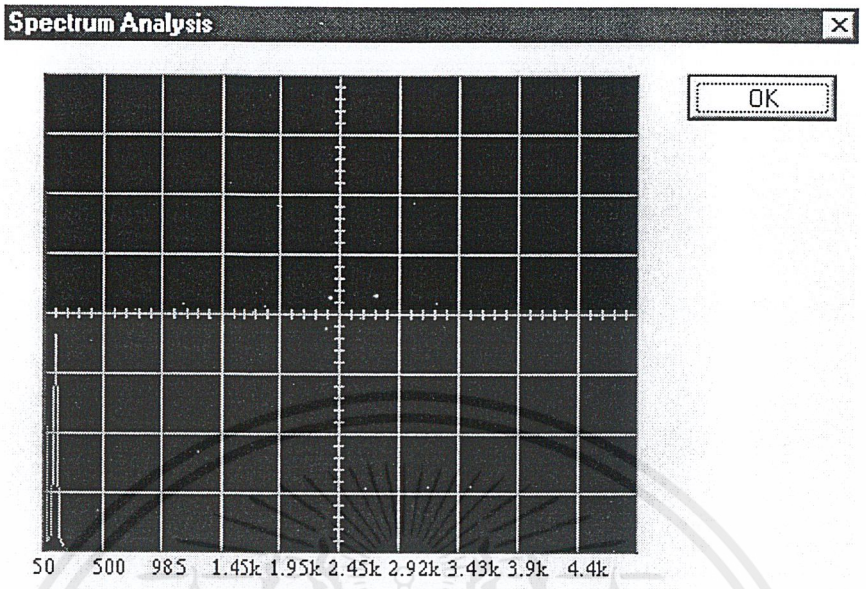


รูปที่ 6.21 ผลการวัดสเปคตรัมของสัญญาณซายน์ความถี่ 20 Hz

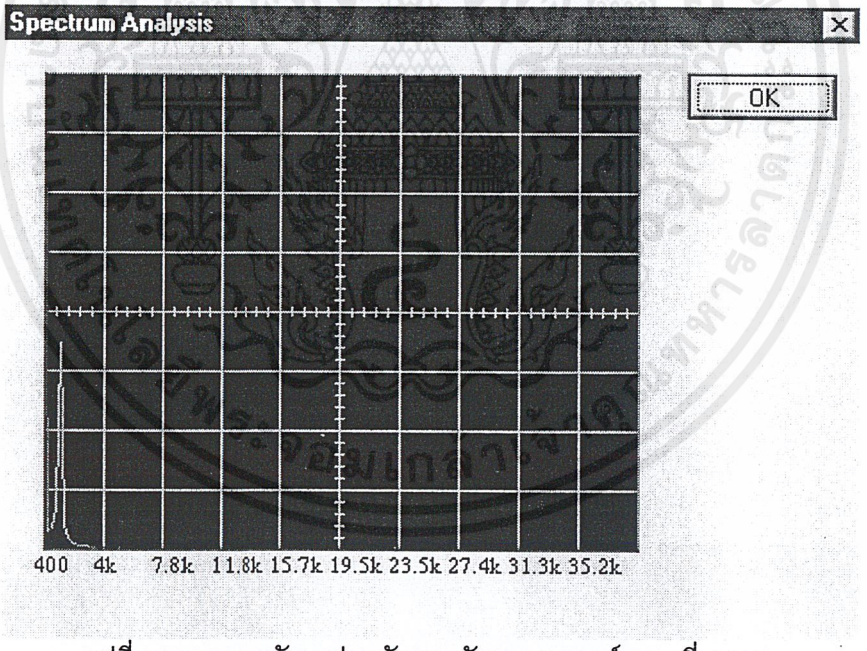


รูปที่ 6.22 ผลการวัดสเปคตรัมของสัญญาณซายน์ความถี่ 50 Hz

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

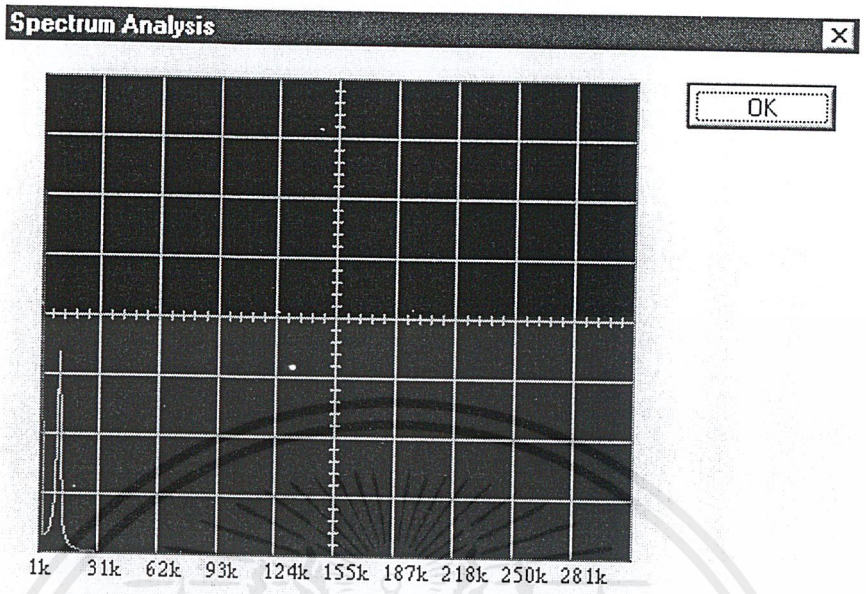


รูปที่ 6.23 ผลการวัดสเปคตรัมของสัญญาณซายน์ความถี่ 100 Hz

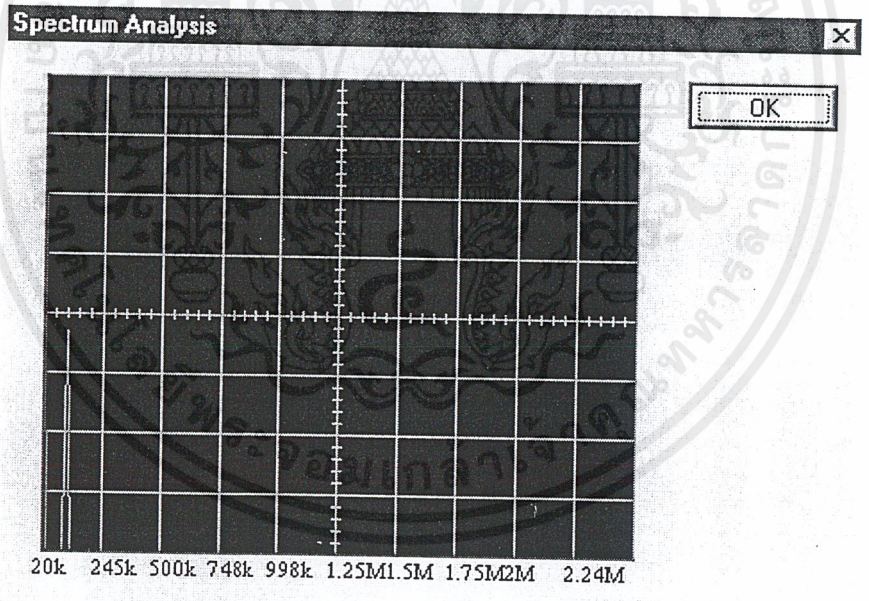


รูปที่ 6.24 ผลการวัดสเปคตรัมของสัญญาณซายน์ความถี่ 1 kHz

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

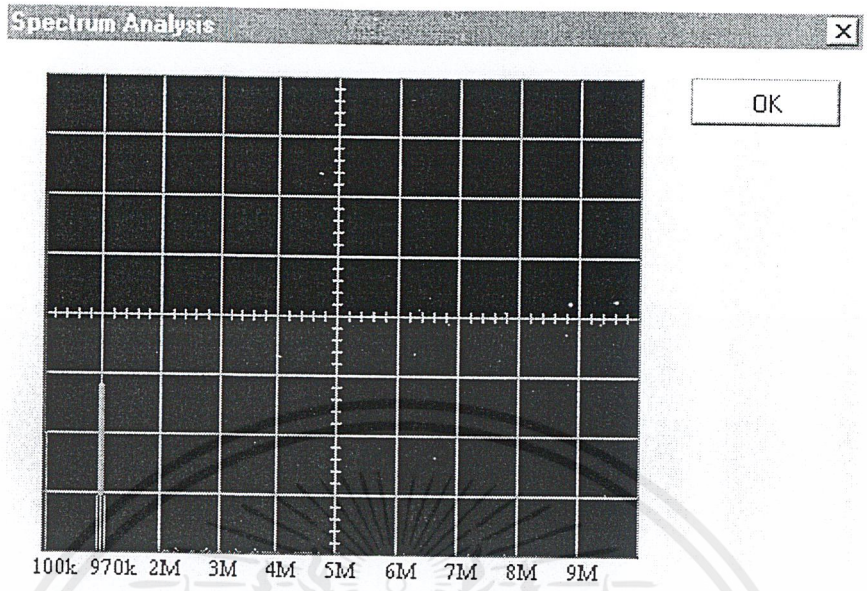


รูปที่ 6.25 ผลการวัดสเปคตรัมของสัญญาณซายน์ความถี่ 10 kHz



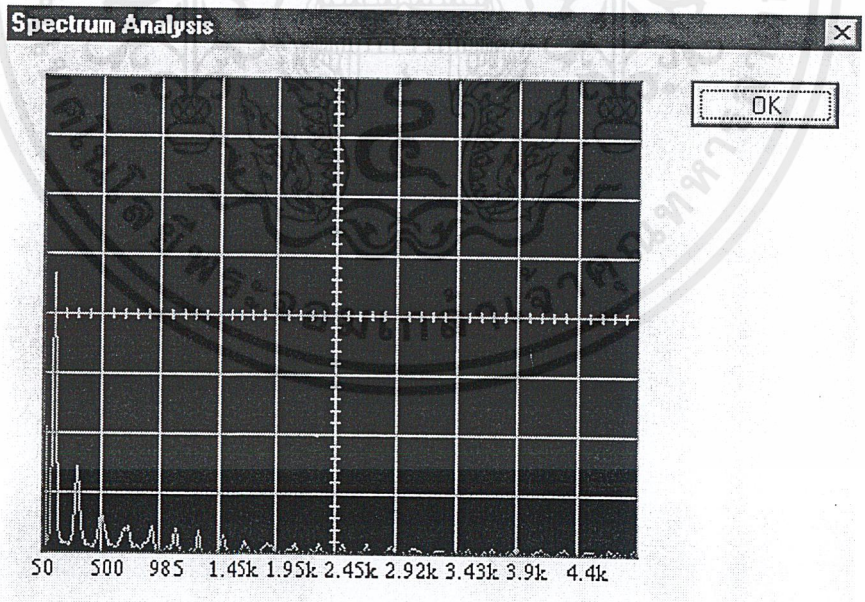
รูปที่ 6.26 ผลการวัดสเปคตรัมของสัญญาณซายน์ความถี่ 100 kHz

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



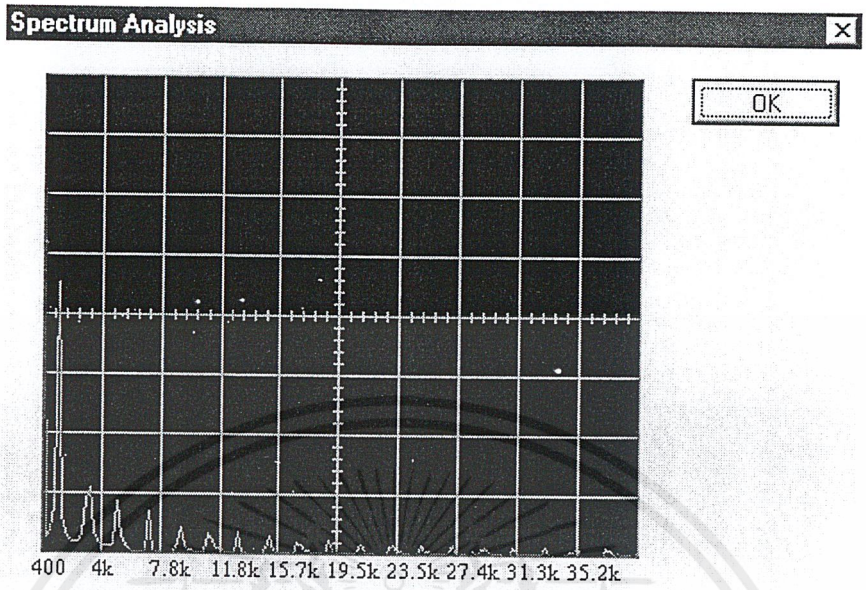
รูปที่ 6.27 ผลการวัดสเปคตรัมของสัญญาณซายน์ความถี่ 1 MHz

ผลการวัดสเปคตรัมของสัญญาณสี่เหลี่ยม

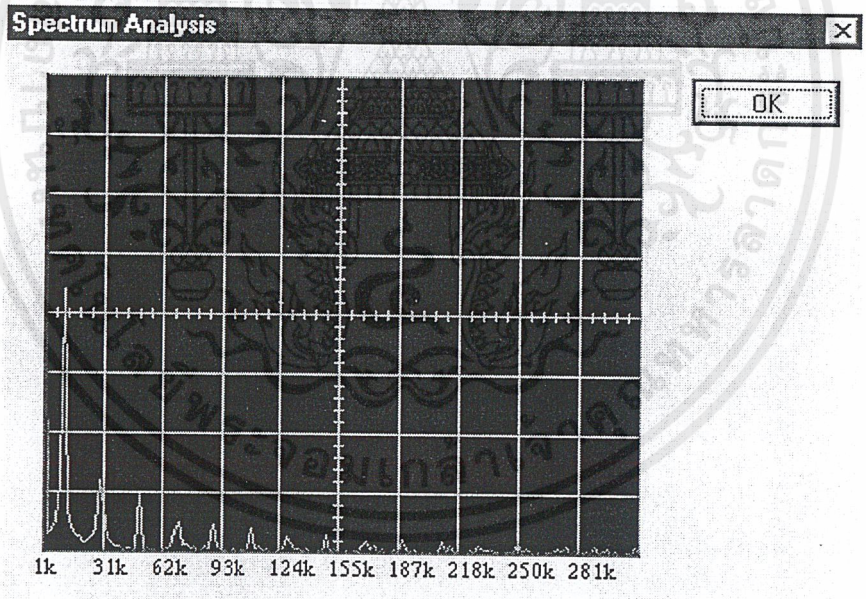


รูปที่ 6.28 ผลการวัดสเปคตรัมของสัญญาณสี่เหลี่ยมความถี่ 100 Hz

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

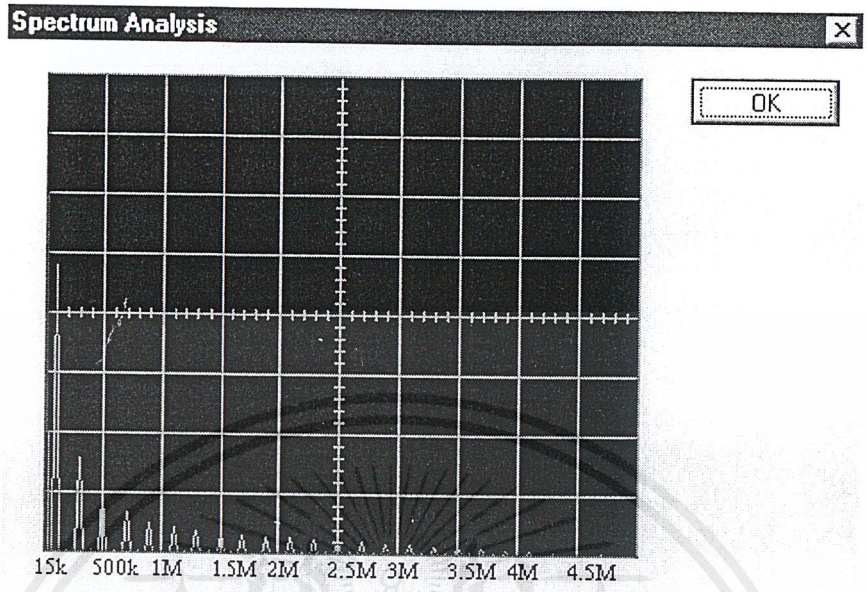


รูปที่ 6.29 ผลการวัดสเปคตรัมของสัญญาณสี่เหลี่ยมความถี่ 1 kHz



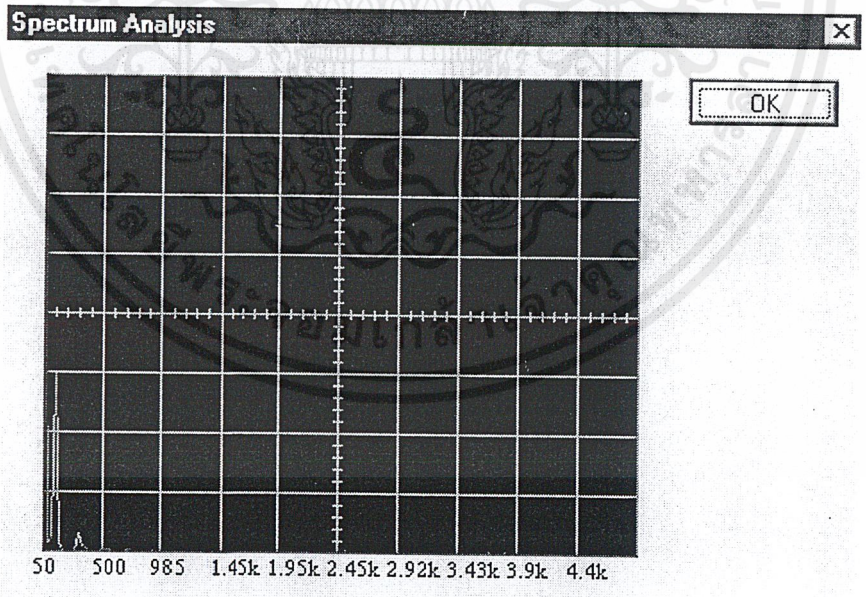
รูปที่ 6.30 ผลการวัดสเปคตรัมของสัญญาณสี่เหลี่ยมความถี่ 10 kHz

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



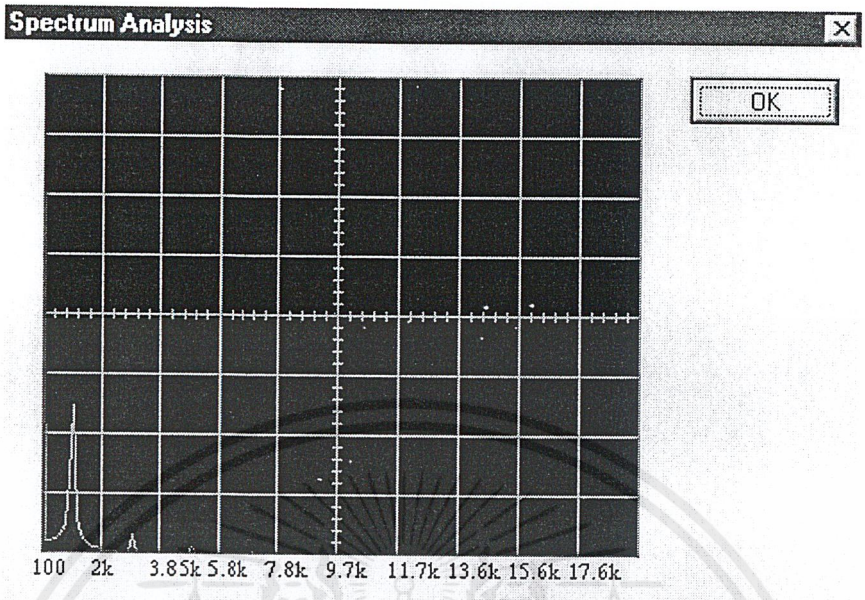
รูปที่ 6.31 ผลการวัดสเปคตรัมของสัญญาณสี่เหลี่ยมความถี่ 100 kHz

ผลการวัดสเปคตรัมของสัญญาณสามเหลี่ยม

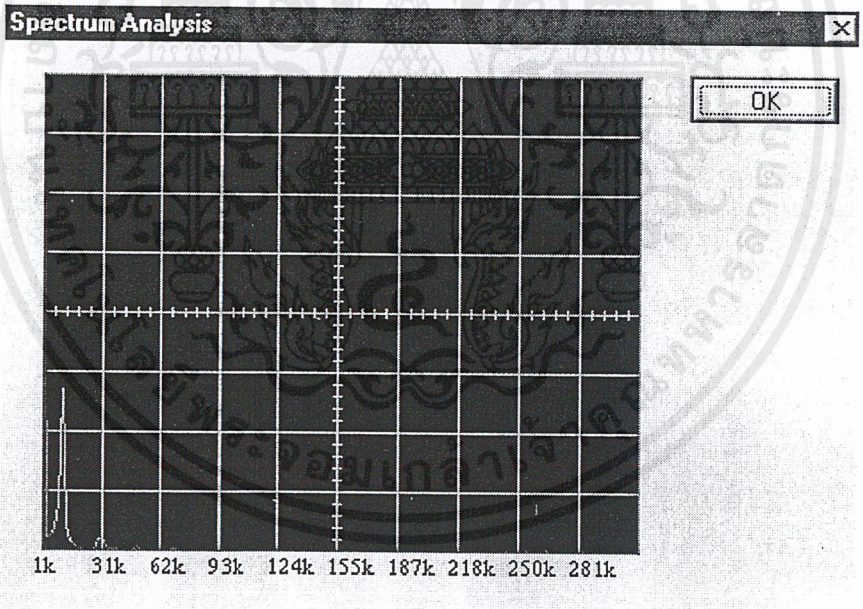


รูปที่ 6.32 ผลการวัดสเปคตรัมของสัญญาณสามเหลี่ยมความถี่ 100 Hz

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

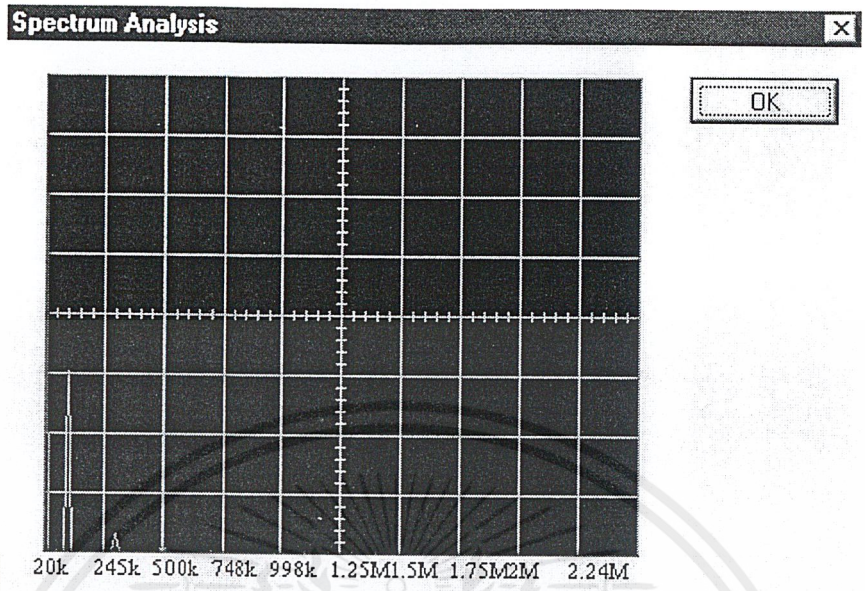


รูปที่ 6.33 ผลการวัดสเปคตรัมของสัญญาณสามเหลี่ยมความถี่ 1 kHz



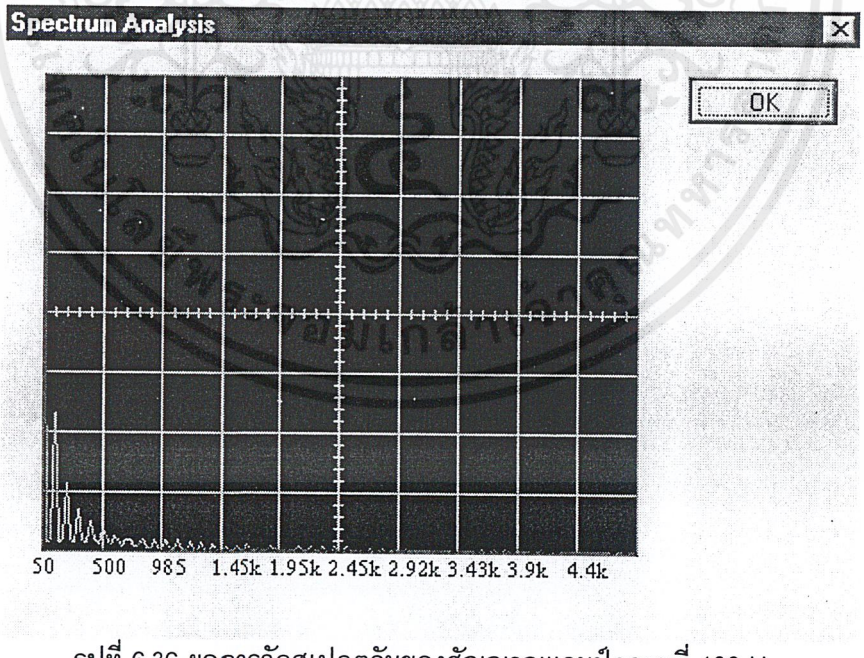
รูปที่ 6.34ผลการวัดสเปคตรัมของสัญญาณสามเหลี่ยมความถี่ 10 kHz

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



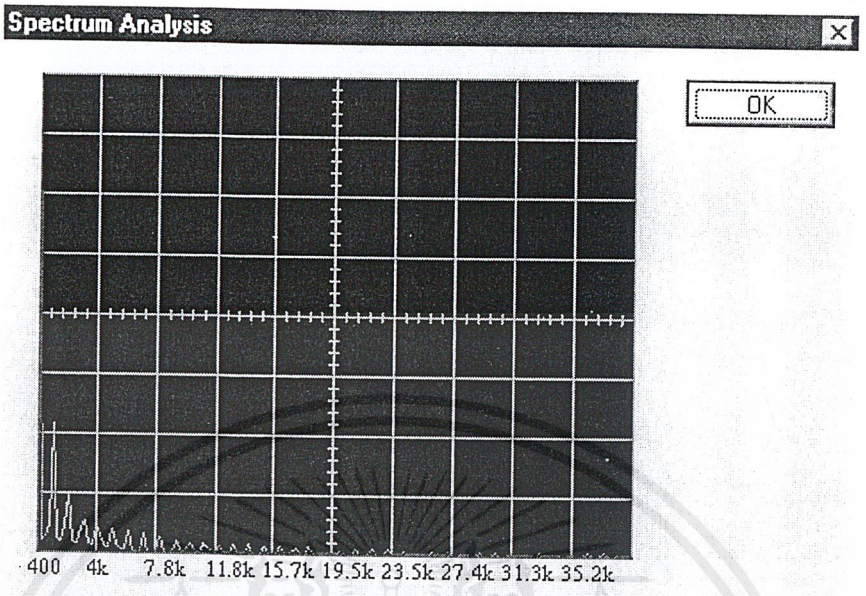
รูปที่ 6.35 ผลการวัดสเปคตรัมของสัญญาณสามเหลี่ยมความถี่ 100 kHz

ผลการวัดสเปคตรัมของสัญญาณแรมป์

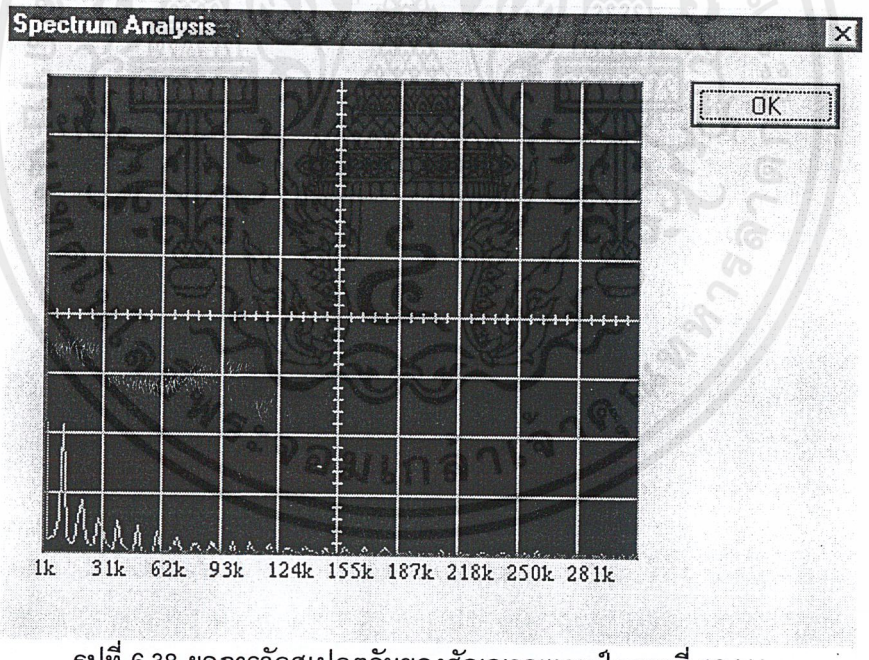


รูปที่ 6.36 ผลการวัดสเปคตรัมของสัญญาณแรมป์ความถี่ 100 Hz

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 6.37 ผลการวัดสเปคตรัมของสัญญาณแรมป์ความถี่ 1 kHz



รูปที่ 6.38 ผลการวัดสเปคตรัมของสัญญาณแรมป์ความถี่ 10 kHz

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 7
สรุปผลการทดลอง

7.1 บทสรุปและวิจารณ์

จากผลการทดลองวงจรในส่วนต่างๆทั้ง วงจรในส่วนที่เป็นอนาลอก และวงจรในส่วนที่เป็นดิจิตอลเราพบว่าวงจรของเราสามารถทำงานได้ค่อนข้างดี การปรับระดับทริกเกอร์ในโครงงานนี้จะไม่ใช้ฮาร์ดแวร์ทริกเกอร์ แต่จะใช้ซอฟแวร์ทริกเกอร์แทน ข้อดีของซอฟแวร์ทริกเกอร์ คือ สัญญาณที่แสดงจะนิ่งกว่า ฮาร์ดแวร์ทริกเกอร์ แต่ก็มีข้อเสียตรงที่มีการแสดงผลจะช้าลงเพราะต้องแบ่งเวลาส่วนหนึ่งไปประมวลผลในส่วนของการรักษาระดับทริกเกอร์ไว้ ข้อมูลที่ได้จากการแปลงอนาลอกเป็นดิจิตอล จะถูกเก็บลงในหน่วยความจำ และตัวโปรแกรมจะอ่านค่าข้อมูลขึ้นมาแสดงผล ในส่วนของการแปลงข้อมูลของวงจรอนาลอกเป็นดิจิตอล แปลงข้อมูลได้อย่างแม่นยำ ข้อมูลที่ผิดพลาดส่วนใหญ่จะเกิดจากสัญญาณรบกวนที่เกิดขึ้นในภาคของวงจรอนาลอก

7.2 ปัญหาที่พบในโครงงาน

1.พบว่ามีการเกิดสัญญาณรบกวนจากอุปกรณ์ภายนอก และอุปกรณ์ภายในวงจรทำให้สัญญาณที่วัดได้มีความผิดเพี้ยนไปจากรูปสัญญาณจริง

2.ต้องใช้เวลานานในการศึกษาการใช้งานอุปกรณ์เอฟพีจีเอ

3.เนื่องจากการวางสายเพื่อต่อวงจรมีความซับซ้อนมาก ฉะนั้นเวลาที่เกิดปัญหา และต้องมีการแก้ไขวงจร จะทำได้ค่อนข้างยาก

4.ต้องใช้เวลาในการเขียนโปรแกรมค่อนข้างนาน เพราะต้องศึกษาการเขียนโปรแกรมวิชวลซีพลัสพลัส

7.3 การแก้ปัญหา

1.วางอุปกรณ์ให้ชิดกันให้มากที่สุด และควรวางสายให้สั้นที่สุด นอกจากนี้ควรแยกแหล่งจ่ายไฟของวงจรทางอนาลอก และดิจิตอลออกจากกัน เพื่อลดสัญญาณรบกวน

2.ควรมีการศึกษาหาความรู้เพิ่มเติม อีกทั้งต้องรู้จักปรึกษาผู้ที่มีความรู้เพื่อชี้แนะแนวทางในการแก้ปัญหาให้รวดเร็วยิ่งขึ้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.ควรวางอุปกรณ์ให้เป็นระเบียบ และออกแบบลายวงจรให้เป็นแผงวงจร เพื่อป้องกันปัญหาจากการหลุดของการวางสาย

4.ต้องรู้จักปรึกษาปัญหากับผู้ที่มีความรู้เกี่ยวกับการเขียนโปรแกรมเพื่อลดเวลาในการแก้ปัญหา

7.4 ประโยชน์ที่ได้รับจากโครงงาน

1.ได้ดิจิตอลสตอเรจออสซิลโลสโคป ที่มีต้นทุนที่ต่ำ

2.ด้วยประโยชน์ของเอฟพีจีเอช่วยให้วงจรมีขนาดเล็ก สามารถใช้งานจริงได้สะดวก

3.ทำให้ได้รับประสบการณ์มากขึ้นเกี่ยวกับการออกแบบวงจร และการเขียนโปรแกรมควบคุมการทำงานอุปกรณ์นั้นๆ

4.เป็นแนวทางในการพัฒนาโครงงานอื่นๆ ให้มีประสิทธิภาพมากขึ้น

7.5 แนวทางพัฒนา

1.พัฒนาในส่วนของวงจรเพื่อให้วงจรทางด้านอินพุท สามารถรับสัญญาณอินพุทให้มีขนาดสูงขึ้น

2.พัฒนาวงจรให้ตอบสนองความถี่ที่สูงขึ้น

3.ควรมีการประยุกต์ใช้งานเกี่ยวการประมวลผลสัญญาณดิจิตอลเพื่อช่วยให้สัญญาณมีความสมบูรณ์มากขึ้น

4.พัฒนาให้วงจรสามารถรับสัญญาณได้หลายๆช่องสัญญาณมากขึ้น

7.6 คุณสมบัติของวงจร

1. วงจรมีอินพุทอิมพีแดนซ์ประมาณ 1 เมกกะโอห์ม
2. สามารถวัดสัญญาณที่มีขนาดสูงสุดได้ไม่เกิน 16 V_{p-p}
3. สามารถวัดสัญญาณที่มีขนาดต่ำสุดเท่ากับ 20 mV_{p-p}
4. สามารถวัดสัญญาณได้ที่ความถี่สูงสุดเท่ากับ 1 MHz
5. สามารถวัดและวิเคราะห์สเปกตรัมได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เอกสารอ้างอิง

ก. <u>เอกสารอ้างอิงที่เป็นวารสารภาษาอังกฤษ</u> จัดเรียงตามลำดับดังนี้

John Becker, "PC SCOPE INTERFACE" Everyday Electronics, October 1991, pp 620-625

ข. <u>เอกสารอ้างอิงที่เป็นวารสารภาษาไทย</u> จัดเรียงตามลำดับดังนี้

เปรมจิตร วิสุทธิศิริ, "พื้นฐานวงจรเอทูดี, ดีทูเอ ตอน 2 วงจรแปลงอนาลอกเป็นดิจิตอล",
เซมิคอนดักเตอร์ อิเล็กทรอนิกส์ ฉบับที่ 103

ธานินทร์ ถาวรศาสนวงศ์ "บทที่ 4 สัญญาณต่างๆบนสล๊อตของ IBM/PC",
การอินเตอร์เฟส IBM/PC, มีนาคม 2536

รศ.กฤษฎา วิศวธีรานนท์ ผศ.ประยูร เขี่ยววัฒนา, "หลักการทำงาน และเทคนิคการใช้งานออสซิลโลสโคป", ภาควิชาวิศวกรรมไฟฟ้า จุฬาลงกรณ์มหาวิทยาลัย, 158 หน้า 2521

ค. <u>เอกสารอ้างอิงที่เป็นปริญญานิพนธ์</u> จัดเรียงตามลำดับดังนี้

บุญอนันต์ เกียงเอียง อลงกรณ์ ศศิธรวรรณกุล, "คอมพิวเตอร์ออสซิลโลสโคป", ภาควิชาอิเล็กทรอนิกส์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง, 2542

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้