

เมาส์สามมิติ

Mouse 3D



ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต

ภาควิชาอิเล็กทรอนิกส์

คณะวิศวกรรมศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2546

เลขหมู่.....

เลขทะเบียน.....55782

วัน,เดือน,ปี25 ๗.๕. 2548

b. คณบดี.....
r. นำไปใช้.....

เม้าส์สามมิติ

Mouse 3D



ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต
ภาควิชาอิเล็กทรอนิกส์
คณะวิศวกรรมศาสตร์
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
ปีการศึกษา 2546

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เม้าท์สามมิติ

MOUSE 3D

ผู้จัดทำ

1. นายศิริชัย เรืองเดช รหัส 44015218
2. นายสุรเดช สनงาม รหัส 44015223



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ปริญญานิพนธ์ ปีการศึกษา 2546

ภาควิชา อิเล็กทรอนิกส์

คณะ วิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เรื่อง MOUSE 3D

ผู้จัดทำ

3. นายศิริชัย เรืองเดช รหัส 44015218

4. นายสุรเดช สอนงาม รหัส 44015223



(เทอดศักดิ์ ถิวหาทอง)

อาจารย์ที่ปรึกษา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เม้าส์สามมิติ

ศิริชัย เรืองเดช

สุรเดช สนงาม

อ. เทอดศักดิ์ ลีวาททอง (อาจารย์ที่ปรึกษา)

ปีการศึกษา 2546

บทคัดย่อ

ปริญญานิพนธ์ฉบับนี้นำเสนอการหาดำแหน่งของตัวส่งคลื่นอัลตราโซนิกในระนาบพิกัด 3 มิติ โดยอาศัยผลต่างของเวลาที่ตัวส่งเดินทางมายังตัวรับทั้ง 5 ตัว ซึ่งเราทราบตำแหน่งพิกัดของตัวรับและโดยอาศัยหลักการเดินทางของคลื่นอัลตราโซนิก ที่ส่งมาจากตัวส่งมายังตัวรับแต่ละตัว ซึ่งผลต่างของเวลาที่คลื่นอัลตราโซนิกเดินทางมาถึงตัวรับคลื่นแต่ละตัวจะใช้ FPGA เป็นตัววัดและคำนวณค่าเวลาและส่งต่อเข้าสู่ทางพอร์ตขนานของคอมพิวเตอร์เพื่อคำนวณหาพิกัดของตัวส่งโดยแสดงออกมาลักษณะพิกัด 3 มิติ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Mouse 3D

Sirichai Ruangdet

Suradet Sonngam

Thurdsak Leauhatong Advisor

2003

ABSTRACT

The thesis presents a systems for accurately estimating a position of an ultrasonic transmitter in 3-D space. The position is estimated by using differences in the time of flights (DTOFs) with a fired ultrasonic wave uses to travel from the transmitter to each of receivers. Each of receivers is fixed at a known position. First, DTOFs are measured by using a field program gate array (FPGA), and then are sent to a personal computer (PC) via a parallel port. Finally the coordinate of the transmitter is calculated and presented.

สารบัญ

บทคัดย่อ

Abstract

สารบัญ

สารบัญภาพ

บทที่1 บทนำ

บทที่2 ทฤษฎีที่ใช้ในการออกแบบ

2.1 ทฤษฎีเกี่ยวกับคลื่นอัลตราโซนิก

2.2 อัลตราโซนิกทรานสดิวเซอร์

2.3 ความรู้เบื้องต้นของพอร์ทขนาน

2.4 Field Programmable Gate Array (FPGA)

2.5 OpenGL

บทที่3 การออกแบบและการทำงานของเม้าท์สามมิติ

3.1 แนวทางการออกแบบเม้าท์สามมิติ

3.2 การทำงานของเม้าท์สามมิติ

3.3 วงจรภาคประมวลผล

3.4 หลักการทำงานของส่วนFPGA

3.5 การเขียนโปรแกรมด้วย Visual Basic

บทที่4 การทดลองและผลการทดลอง

4.1 สัญญาณ output ที่ตัวรับอัลตราโซนิกที่มุม 0 องศา ที่ระยะต่างๆ

(ขณะยังไม่ผ่านวงจรเปรียบเทียบแรงดัน)

4.2 สัญญาณ output ที่ตัวรับอัลตราโซนิกที่มุม 0 องศา ที่ระยะต่างๆ

(ขณะผ่านวงจรเปรียบเทียบแรงดัน)

4.3 สัญญาณ output ที่ตัวรับอัลตราโซนิกที่มุม 25 องศา ที่ระยะต่างๆ

(ขณะยังไม่ผ่านวงจรเปรียบเทียบแรงดัน)

4.4 สัญญาณ output ที่ตัวรับอัลตราโซนิกที่มุม 25 องศา ที่ระยะต่างๆ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1

5

5

12

16

23

29

31

31

33

35

40

44

45

47

50

53

56

(ขณะผ่านวงจรเปรียบเทียบแรงดัน)

4.5 แสดงผลการ Simulate ของ FPGA	59
4.6 ทดสอบการเชื่อมต่อระหว่าง FPGA กับ Computer	62

บทที่ 5 สรุปผลการทดลอง	65
-------------------------------	----

กิตติกรรมประกาศ

เอกสารอ้างอิง



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญภาพ

รูปที่1	แนวคิดในการออกแบบ	1
รูปที่1.1	แสดงการวางตำแหน่งของตัวส่งและตัวรับ	2
รูปที่1.2	ภาพแสดงการทำงานของส่วน FPGA	3
รูปที่1.3	ภาพแสดงการทำงานของส่วนประมวลผลและแสดงผล	4
รูปที่2.1	แสดงถึงช่วงความถี่ต่างๆที่ถูกนำไปใช้งาน	5
รูปที่2.2	แสดงลักษณะการเกิดคลื่นตามยาว	6
รูปที่2.3	แสดงลักษณะการเกิดคลื่นตามขวาง	7
รูปที่2.4	แสดงการบีบอัดของคลื่นเสียง 2 คลื่น	8
รูปที่2.5	แสดงลักษณะการเกิดคลื่นนิ่ง	8
รูปที่2.6	แสดงลักษณะของคลื่นเสียงที่ผ่านช่องแคบๆ จะเสมือนเป็นแหล่งกำเนิดตัวใหม่	9
รูปที่2.7	แสดงการหักเหของคลื่นเสียงในตัวกลางที่มีความหนาแน่นต่างกัน	9
รูปที่2.8	แสดงเมื่อคลื่นวงกลมกระทบกับฉากที่เป็นเส้นตรงคลื่นที่สะท้อนกลับมาจะเป็นคลื่นวงกลมเช่นเดียวกัน	10
รูปที่2.9	ผลของเปียโซอิเล็กทริก	12
รูปที่2.10	วงจรเสมือนของเปียโซอิเล็กทริก	12
รูปที่2.11	แสดงไดอะแกรมเวลาของการส่งข้อมูลไปยังเครื่องพิมพ์	16
รูปที่2.12	แสดงสัญญาณทั้งหมดที่อยู่บนพอร์ตขนาน	18
รูปที่2.13	วงจรภายในของพอร์ต Data	19
รูปที่2.14	วงจรภายในของพอร์ต Control	21
รูปที่2.15	แสดงวงจรภายในของพอร์ตแสดงสถานะ	22
รูปที่2.16	โครงสร้างภายใน FPGA ตระกูล MAX7000S	24
รูปที่2.17	โครงสร้างภายในของ FPGA ตระกูล FLEX10K	24
รูปที่ 3.1	แสดงภาคส่งคลื่น Ultrasonic	33
รูปที่ 3.2	แสดงภาครับคลื่น Ultrasonic	34
รูปที่ 3.3	แสดงการวางตำแหน่งของตัวส่งและตัวรับ	35
รูปที่ 3.4	แสดงภาพวงจรการทำงานภายใน Block Control	40
รูปที่ 3.5	แสดงภาพวงจรการทำงานภายใน Block Counter	41
รูปที่ 3.6	แสดงภาพวงจรในส่วนของ FPGA	42

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 3.7 แสดงภาพวงจรทั้งหมดในส่วนของ FPGA	43
รูปที่ 3.8 แสดงภาพ flow chart	44
รูปที่ 4.1 รูปสัญญาณที่สร้างโดย MCS-51	45
รูปที่ 4.2 รูปสัญญาณที่ output ภาคส่ง	46
รูปที่ 4.3 แสดง Timing Diagram ของ Block Control ที่ได้จากการ Simulate	59
รูปที่ 4.4 แสดง Timing Diagram ของ Block Counter ที่ได้จากการ Simulate	60
รูปที่ 4.5 แสดง Timing Diagram ของวงจรทั้งหมดของ FPGA	61
รูปที่ 4.6 แสดงผลการทดสอบโปรแกรมที่เชื่อมต่อกับFPGA	64
รูปที่ 5.1 รูปสัญญาณที่วัดได้จากภาคส่ง รูป (A) สัญญาณที่ออกจาก MCS-51 รูป (B) สัญญาณที่ output ของภาคส่ง	65
รูปที่ 5.2 รูปสัญญาณที่วัดได้จากภาครับ รูป (A) และ (B) สัญญาณขณะที่ยังไม่ผ่านวงจร เปรียบเทียบแรงดัน รูป (C) สัญญาณที่ผ่านวงจรเปรียบเทียบแรงดันแล้ว	66
รูปที่ 5.3 รูปกราฟแสดงค่าที่คำนวณผิดพลาดไปถ้าเวลาที่รับ ได้ผิดพลาด	67

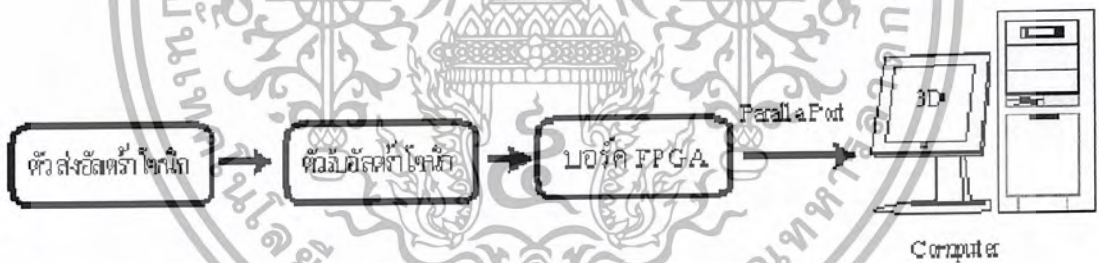


บทที่ 1

บทนำ

ปริญญานิพนธ์ฉบับนี้เสนอการออกแบบระบบ เพื่อใช้บอกตำแหน่งของตัวชี้ตำแหน่ง (X,Y,Z) บนจอคอมพิวเตอร์ โดยอาศัยหลักการทำงานของคลื่น Ultrasonic เริ่มต้นด้วยการส่งสัญญาณจากตัวส่ง(ตัวชี้ตำแหน่ง) มาที่ตัวรับทั้ง 5 ตัว ซึ่งจัดวางอยู่ในรูปสี่เหลี่ยมจำนวน 4 ตัวและอีก 1 ตัวอยู่ตรงกลาง ซึ่งตัวรับนี้ทั้งหมดนี้จะรับสัญญาณของตัวส่งได้ในเวลาที่ต่างกัน เราจึงนำผลต่างของเวลาที่รับได้ส่งเข้าไปที่ FPGA ไปเปรียบเทียบกับสัญญาณนาฬิกา เพื่อนับผลต่างของเวลาในรูปสัญญาณนาฬิกา

จำนวนผลต่างของสัญญาณนาฬิกา ที่ได้ส่งต่อไปคำนวณระยะทางที่คอมพิวเตอร์ผ่านทางพอร์ตขนาน โดยค่าของสัญญาณนาฬิกาที่ต่างกันจะนำไปคำนวณระยะทางในพิกัด X,Y,Z โดยใช้โปรแกรม Visual Basic และแสดงผลออกเป็นภาพสามมิติที่ต้องการ ต่อไป



รูปที่ 1 แนวคิดในการออกแบบ

จากที่ได้กล่าวถึงการทำงานของ Mouse3D ไปพอสมควรแล้ว ในส่วนนี้จะกล่าวถึงการทำงานโดยละเอียดในแต่ละส่วน โดยจะกล่าวไปถึงส่วนประกอบของโครงการให้ทราบเสียก่อนว่าประกอบไปด้วยอะไรบ้างแล้วจะกล่าวถึงหลักการการทำงานต่อไป

1.1 ส่วนประกอบของโครงการ

ในโครงการนี้จะประกอบไปด้วย 3 ส่วนหลักๆคือ

1.1.1 ชุดรับ-ส่งสัญญาณอัลตราโซนิก

ซึ่งจะแบ่งออกไปอีกคือ ตัวส่ง และ ตัวรับ สัญญาณอัลตราโซนิก



รูปที่ 1.1 แสดงการวางตำแหน่งของตัวส่งและตัวรับ

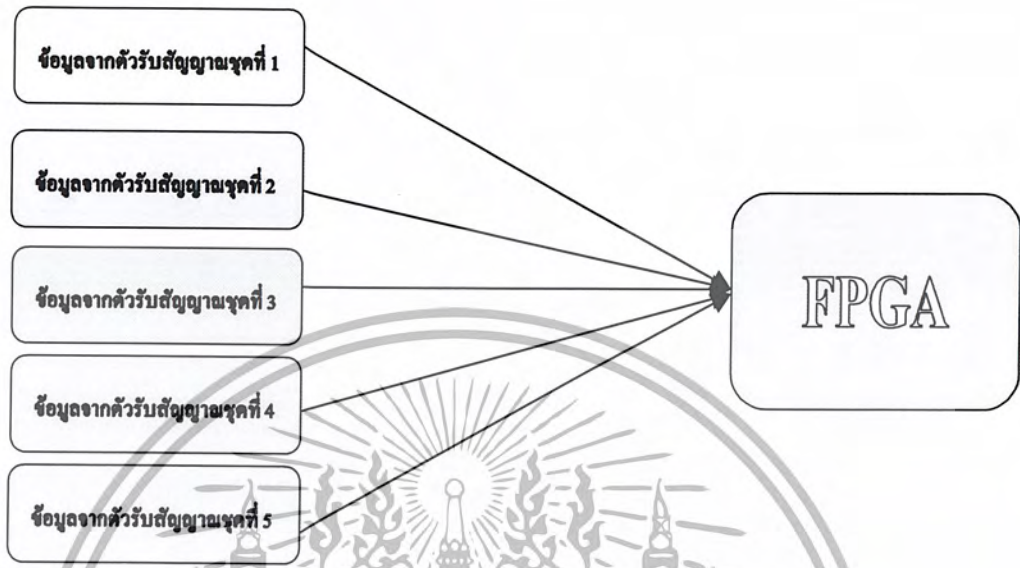
ตัวส่งสัญญาณอัลตราโซนิก

ส่วนนี้ทำหน้าที่ส่งคลื่นสัญญาณอัลตราโซนิก(ความถี่40kHz) ออกไป เพื่อเป็นตัวกำหนดพิกัดของตัวส่งสัญญาณว่าอยู่ที่ตำแหน่งใด โดยวงจรส่วนนี้มีหลักการทำงานซึ่งจะแสดงให้เห็นหัวข้อต่อไป

ตัวรับสัญญาณอัลตราโซนิก

วงจรส่วนนี้ทำหน้าที่รับสัญญาณอัลตราโซนิกที่ตัวส่งๆออกมา โดยตัวรับจะแบ่งออกเป็น 5 จุดวางตัวเป็นรูปสี่เหลี่ยมจำนวน 4 ตัวและตรงกลางอีก 1 ตัว ตัวรับแต่ละตัวจะได้รับสัญญาณที่เวลาแตกต่างกันเราจึงเอาหลักการนี้ไปเปรียบเทียบกับสัญญาณนาฬิกาที่ชูดับสัญญาณต่อไป เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1.1.2 บอร์ดFPGA

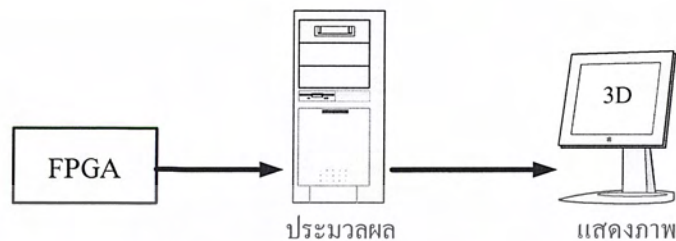


รูปที่ 1.2 ภาพแสดงการทำงานของส่วน FPGA

บอร์ดนี้เป็นบอร์ดทดลองเกี่ยวกับการSimulate พวกลอจิกที่เกี่ยวข้องกับอุปกรณ์ทางอิเล็กทรอนิกส์ โดยได้นำบอร์ดตัวนี้เข้ามาเพื่อทำหน้าที่เป็นวงจร Counter โดยนำผลต่างของเวลาที่ภาครับสัญญาณอัลตราโซนิคมาเปรียบเทียบกับสัญญาณนาฬิกาบนบอร์ดFPGA จะได้ผลข้อมูลที่เป็นจำนวนพัลส์ของตำแหน่งทั้ง 5 ตัว ของตัวรับเพื่อส่งไปประมวลผลผ่านทางพอร์ตขนานของตัวบอร์ดต่อไป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1.1.3 ภาคนำจำนวนและประมวลผลโดยเครื่องคอมพิวเตอร์



รูปที่ 1.3 ภาพแสดงการทำงานของส่วนประมวลและแสดงผล

ภาคนีทำงาน โดยการนำค่าCounterที่นับได้ส่งเข้าทางพอร์ตขนานของบอร์ดFPGA เพื่อเข้ามาคำนวณพิกัดของตัวส่งสัญญาณ ส่วนนี้ประกอบด้วย โปรแกรมVisual Basic เป็นส่วนแสดงการทำงานของโปรแกรมนี้น และนำค่าพิกัดที่คำนวณได้ ไปขึ้นเป็นภาคสามมิติ

บทที่ 2

ทฤษฎีที่ใช้ในการออกแบบ

2.1 ทฤษฎีเกี่ยวกับคลื่นอัลตราโซนิก

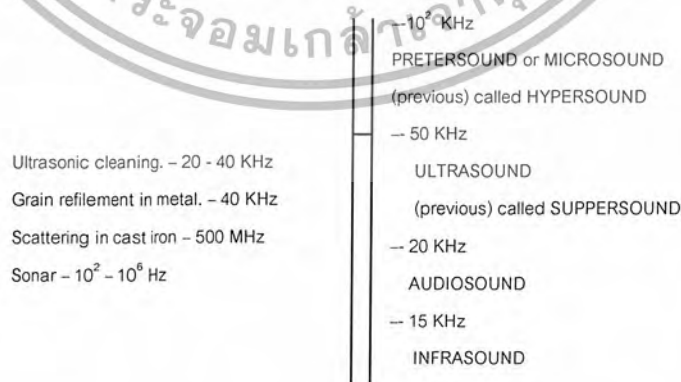
เครื่องซีดำแห่ง 3 มิตินี้ อาศัยหลักการวัดระยะทางด้วยคลื่นอัลตราโซนิก แล้วนำค่าระยะทางที่วัดได้มาคำนวณหาพิกัดตำแหน่ง โดยทฤษฎีของคลื่นอัลตราโซนิกจะกล่าวโดยสังเขปดังนี้

2.1.1 คุณสมบัติและธรรมชาติของคลื่นอัลตราโซนิก

หลักการของคลื่นเสียง คือ พลังของพลังงานจะถูกส่งออกมาจากตัวส่งไปซึ่งถ้าเรานำตัวรับชนิดเดียวกันกับตัวส่งซึ่งมีความถี่ตรงกันแล้วก็จะสามารถรับสัญญาณได้ ซึ่งคลื่นอัลตราโซนิกจะเป็นคลื่นที่มีทิศทางและจะพุ่งออกจากตัวส่งซึ่งจะมีขอบเขตประมาณ 30 องศาจากตัวส่ง เวลาที่ใช้ในการเดินทางในอากาศที่จะคำนวณระยะทางได้เพราะฉะนั้นการศึกษาถึงธรรมชาติของคลื่น จะสามารถทำให้เข้าใจในการนำไปประยุกต์ใช้ได้อย่างถูกต้อง

2.1.2 คลื่นอัลตราโซนิก

คลื่นอัลตราโซนิก คือ คลื่นเสียงที่มีความถี่สูงเกินที่มนุษย์จะได้ยิน โดยทั่วไปแล้วมนุษย์จะได้ยินคลื่นความถี่ประมาณ 20 Hz ถึง 20 kHz ดังนั้นแล้วคลื่นอัลตราโซนิกจึงหมายถึงคลื่นเสียงที่มีความถี่สูงกว่า 20kHz จนถึง 104 GHz โดยคลื่นที่มีความถี่สูงกว่า 104 GHz และแอมพลิจูดสูงๆ จะเรียกว่า “ไฮเปอร์ซาวนด์”



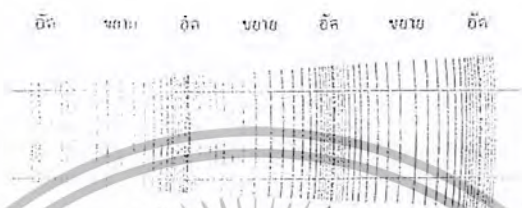
รูปที่ 2.1 แสดงถึงช่วงความถี่ต่างๆที่ถูกนำไปใช้งาน

2.1.3 ชนิดของคลื่นอัลตราโซนิก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

คลื่นอัลตราโซนิกที่เดินทางผ่านตัวกลาง ต่างๆ มีหลายชนิดด้วยกันซึ่งแต่ละชนิดแตกต่างกันตามการเคลื่อนของอนุภาคในตัวกลางนั้น

2.1.3.1 คลื่นตามยาว(Longitudinal Wave) คือคลื่นอนุภาคตัวกลางที่มีการเคลื่อนที่ไปในทิศทาง การเคลื่อนที่ของคลื่น



รูปที่ 2.2 แสดงลักษณะการเกิดคลื่นตามยาว

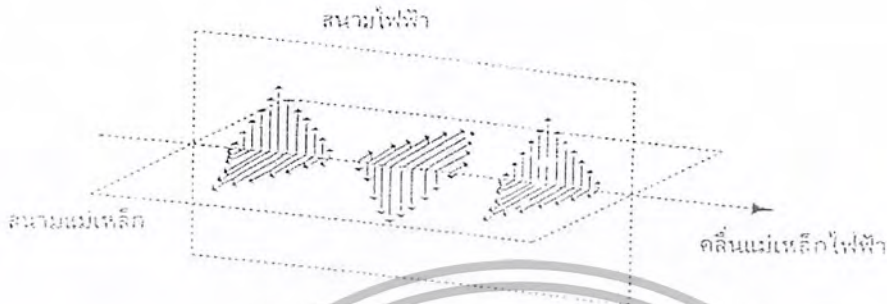
จากรูป 2.2 จะเห็นว่ามีส่วนที่เคลื่อนที่อัด (Compression) ซึ่งก็คือคลื่นช่วงของอนุภาคตัวกลางมีความชันสูงและคลื่นขยาย(Rarefaction) คือที่อนุภาคของตัวกลางมีความชันต่ำและเมื่อนำค่าของความดันที่เปลี่ยนแปลงตามระยะทางมาเขียนกราฟจะได้รูปไซน์ โดยยอดคลื่นจะตรงกับส่วนอัดและท้องคลื่นจะตรงกับส่วนขยาย ระยะทางระหว่างส่วนอัดถึงส่วนอัด หรือส่วนขยายถึงส่วนขยายคือ 1 ความยาวคลื่น และมีคาบเวลาเป็น (T) ซึ่งเท่ากับ $1/f$ โดยจุดที่เป็นแกนนั้นยาว 1 บรรยากาศ

2.1.3.2 คลื่นตามขวาง(Transverse Wave) คือที่ทุกๆจุดบนคลื่น มีการเคลื่อนที่ไปในทิศทางที่ตั้งฉากกับทิศทางเคลื่อนที่ คลื่นชนิดนี้จะเดินทางผ่านตัวกลางที่เป็นของแข็ง คลื่นชนิดนี้ไม่สามารถเดินทางผ่านตัวกลางที่เป็นของแข็งและก๊าซได้

คลื่นตามขวางที่ลักษณะเหมือนกับการเกิดขั้วลบและขั้วบวก ซึ่งเป็นเหตุผลที่ว่า การเปลี่ยนตำแหน่งของอนุภาคเป็นไปในทิศทางเดียว เช่น ในระนาบที่ตั้งฉากกับทิศทางของคลื่น ที่เคลื่อนที่ไปจากต้นกำเนิดของคลื่นตามขวาง เป็นพื้นที่หน้าเรียบของระนาบที่เกิดจากการเปลี่ยนแปลงของอนุภาคอันเนื่องมาจากการแกว่ง และความหนาของตัวกลาง จะไม่เปลี่ยนแปลงโดยการเคลื่อนที่ของคลื่นแบบตามขวาง ความเร็วของคลื่นชนิดนี้จะน้อยกว่าความเร็วของคลื่นชนิดตามยาว ในขณะที่

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เดินทางผ่านตัวกลางชนิดเดียวกัน ดังนั้นที่ความถี่เดียวกันความยาวของคลื่นตามขวางจะน้อยกว่า คลื่นตามยาวเสมอ



รูปที่ 2.3 แสดงลักษณะการเกิดคลื่นตามขวาง

2.1.3.3 คลื่นผิวหน้า (Surface Wave or Rayleigh) คือ คลื่นชนิดหนึ่งซึ่งคล้ายกับคลื่นตามขวางจะต่างกันตรงที่ว่า การเปลี่ยนตำแหน่งของอนุภาค ไม่เป็นไปในทิศทางที่ตั้งฉากกับทิศทางที่เคลื่อนที่เพียงอย่างเดียว แต่มีการเปลี่ยนแปลงในทิศทางเดียวกับทิศทางเคลื่อนที่ด้วย จึงทำให้คลื่นเคลื่อนที่ไปตามระนาบในแนวนอน ด้วยเหตุนี้คลื่นจึงเดินทางผ่านไปเฉพาะบนผิวของตัวกลางเท่านั้น

2.1.4 คุณสมบัติที่สำคัญของคลื่นอัลตราโซนิค

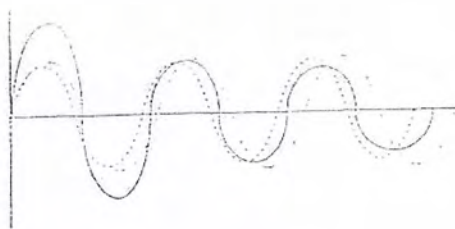
คุณสมบัติโดยทั่วไปของคลื่นเสียงจะแสดงได้ 4 แบบคือ

2.1.4.1 การแทรกสอดของเสียง (Interference)

การแทรกสอดของเสียงเกิดจากการรวมกันของคลื่น 2 คลื่นขึ้นไป ขณะเมื่อพบกันในตัวกลางเดียวกัน ซึ่งทำให้เกิดผลได้หลายลักษณะคือ

-การบีตส์ (Beats) ของคลื่นเสียงเป็นปรากฏการณ์ที่เกิดจากการรวมคลื่นที่มีความถี่ต่างกันหรือต่างเฟสกัน เคลื่อนที่ไปในตัวกลางเดียวกันแล้วรวมเป็นคลื่นใหม่ ซึ่งทำให้แอมพลิจูดเปลี่ยนแปลงดังแสดงในรูปที่ 2.4

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

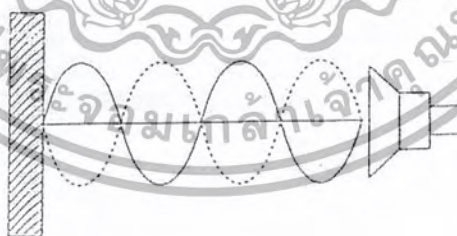


รูปที่ 2.4 แสดงการบีสต์ของคลื่นเสียง 2 คลื่น

จากรูปที่ 2.4 เป็นการแสดงบีสต์ของคลื่น 2 คลื่น ซึ่งคลื่นที่มีแอมพลิจูดเปลี่ยนไปซึ่งประโยชน์ของการบีสต์ของคลื่นนั้น จะนำไปใช้ในการเปรียบเทียบความถี่ของคลื่นให้แสดงผลออกมาในลักษณะของแอมพลิจูดที่แตกต่างกันนี้

-การเกิดคลื่นนิ่ง(Standing Wave) เกิดจากการแทรกสอดของคลื่นซูดที่มีแอมพลิจูดเท่ากันและความถี่เท่ากัน แต่มีทิศทางการเคลื่อนที่ตรงกันข้ามหรือเฟสตรงกันข้าม ซึ่งจะทำให้เกิดคลื่นนิ่งดังรูป เสียงจะมีความเข้มข้นสูงสุดที่ตำแหน่งต้นทาง. และเบาที่สุดที่ตำแหน่งปลายทาง. ซึ่งระยะระหว่างขั้วทั้ง 2 เท่ากับ

$$d = n\lambda/2; (n=1,2,3\dots)$$

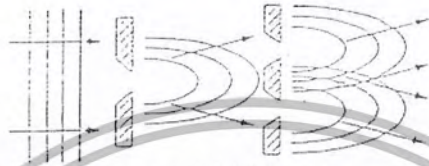


รูปที่ 2.5 แสดงลักษณะการเกิดคลื่นนิ่ง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.1.4.2 การเลี้ยวเบนของคลื่นเสียง (Diffraction)

คือ คลื่นเสียงจะเลี้ยวเบนอ้อมสิ่งกีดขวางที่มีลักษณะเป็นมุมหรือช่องแคบ ซึ่งปรากฏการณ์เช่นนี้พบในชีวิตประจำวันอยู่ตลอดเวลาเช่น ในกรณีที่เราได้ยินแตรรถที่อยู่คนละมุมถนนของตึกหรือการได้ยินเสียงลอดผ่านช่องเล็กๆ จากอีกห้องหนึ่ง ดังแสดงในรูป 2.6



รูปที่ 2.6 แสดงลักษณะของคลื่นเสียงที่ผ่านช่องแคบๆ จะเสมือนเป็นแหล่งกำเนิดตัวใหม่

2.1.4.3 การหักเหของคลื่น (Refraction)

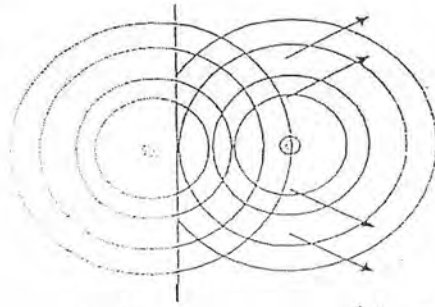
คลื่นเสียงเมื่อเดินทางผ่านตัวกลางที่มีความหนาแน่นต่างกันจะเกิดการหักเหของคลื่น ซึ่งทำให้ความเร็วของคลื่นเสียงเปลี่ยนไป โดยที่ความถี่ยังคงที่อยู่

รูปที่ 2.7 แสดงการหักเหของคลื่นเสียงในตัวกลางที่มีความหนาแน่นต่างกัน

2.1.4.4 การสะท้อนของคลื่นเสียง (Reflection)

คลื่นเสียงสามารถสะท้อนได้เมื่อตกกระทบตัวกลางโดยที่มุมสะท้อนและจะทำให้เกิดเสียงก้อง ซึ่งเสียงก้องที่สะท้อนกลับมาในเวลาที่มากกว่า 50 msec จะทำให้เราได้ยินเสียงนี้เป็นครั้งที่ 2

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.8 แสดงเมื่อคลื่นวงกลมกระทบกับฉากที่เป็นเส้นตรง
คลื่นที่สะท้อนกลับมาจะเป็นคลื่นวงกลมเช่นเดียวกัน

2.1.4.5 การเกิดคลื่นอัลตราโซนิก

อัลตราโซนิกเป็นคลื่นที่เกิดจากการเปลี่ยนแปลงพลังงานรูปอื่นให้เป็นพลังงานกล โดยการสั่นไปมา หรือ เกิดจากการเปลี่ยนแปลงพลังงานไฟฟ้ากับพลังงานกลให้เกิดคลื่นย่านอัลตราโซนิก กระจายออกไปในอากาศ ดังนั้นจึงถือได้ว่าคลื่นที่เกิดขึ้นเป็นคลื่นกล (Mechanical Wave) อัลตราโซนิกสามารถสร้างได้โดยตัวทรานสดิวเซอร์ ซึ่งเป็นอุปกรณ์ที่ใช้เปลี่ยนพลังงานไฟฟ้าเป็นพลังงานกล หลักการสร้างมีหลายวิธีดังนี้

- แบบเพียโซอิเล็กทริก (Piezo-electric Transducer) ซึ่งแปลงไปมาระหว่างพลังงานไฟฟ้ากับพลังงานกล โดยมีความถี่เรโซแนนท์คงที่ค่าหนึ่ง
- แบบแมกเนโตสตริกทีฟ (Magnetostrictive Transducer) ซึ่งแปลงไปมาระหว่างพลังงานไฟฟ้าในขดลวดกับตำแหน่งความยาวของแกนเหล็กที่สวมขดลวดนั้น
- แบบอิเล็กโตรสตริกทีฟ (Electrostrictive Transducer) ซึ่งแปลงไปมาระหว่างพลังงานไฟฟ้ากับพลังงานกล

*** สำหรับเพียโซอิเล็กทริกเป็นแบบที่นิยมใช้เพราะมีราคาถูกและหาซื้อง่าย

2.1.4.6 ความถี่และความยาวคลื่น (Frequency and Wave Length)

ความถี่ คือ จำนวนของการออสซิลเลทที่สมบูรณ์จากแหล่งกำเนิดคลื่นภายในหนึ่งวินาที คลื่นที่ถูกส่งจากแหล่งกำเนิดจะเดินทางด้วยความถี่เดียวกัน

ความยาวคลื่น คือระยะทางที่คลื่นเดินทางระหว่างการสั่นที่สมบูรณ์ หรือการเดินทางครบ 1 รอบ (Cycle) สามารถกล่าวได้ว่า ความยาวคลื่นเป็นระยะทางระหว่างการอัดอย่างต่อเนื่อง (Success Compression) หรือการเบาบางของอากาศ (Rare Fraction) การอัดคือการที่บริเวณนั้นมีความหนา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

แน่นของโมเลกุล และแรงดันมากกว่าบริเวณรอบๆ ส่วนการเบาบางเป็นบริเวณเฉพาะที่เกิดการลดความหนาแน่นของ โมเลกุลและแรงดันสัมพันธ์กับแรงดันของอากาศปกติ

ความถี่และความยาวคลื่นมีความสัมพันธ์กันตามสมการข้างล่าง

$$C=f\lambda$$

C : ความเร็วของการเดินทาง

f : ความเร็ว

λ : ความยาวคลื่น

2.1.4.7 ความเร็วของคลื่นอัลตราโซนิก

ความเร็วที่ยอมรับของคลื่นในอากาศที่อุณหภูมิปกติความสัมพันธ์ที่ใช้จะเป็นดังสมการ

$$V = 331.45 + 0.607T \text{ (m/sec)}$$

V : ความเร็วของคลื่นในตัวกลางอากาศ

T : อุณหภูมิของคลื่นในอากาศ(องศาเซลเซียส)

2.1.4.8 ปริมาณพลังงานของคลื่นอัลตราโซนิก

ปริมาณพลังงานของคลื่นอัลตราโซนิกจะถูกวัดในรูปความเข้มของคลื่นอัลตราโซนิกจะมีหน่วยเป็น วัตต์ต่อตารางเซนติเมตร เป็นการไหลของพลังงานผ่านพื้นที่ 1 ตารางเซนติเมตร ซึ่งตั้งฉากกับทิศทางการเดินทางของคลื่นใน 1 วินาที

2.1.4.9 การลดทอนของคลื่นอัลตราโซนิก

เมื่อคลื่นเดินทางผ่านตัวกลาง ลำคลื่น (Beam) ของคลื่นอัลตราโซนิกจะสูญเสียความเข้มซึ่งเกิดจากการลู่ออกของลำคลื่นอัลตราโซนิก หรือเกิดจากการกระจายพลังงานของคลื่นออกจากลำคลื่นเนื่องจากความไม่ต่อเนื่องของตัวกลาง และอาจเกิดจากการดูดซับพลังงานส่วนหนึ่งของคลื่นโดยตัวกลางที่คลื่นเคลื่อนที่ผ่าน พลังงานที่ดูดซับนี้จะเปลี่ยนเป็นพลังงานความร้อน การดูดซับพลังงานความร้อนนี้ขึ้นอยู่กับลักษณะของวัตถุ ความยืดหยุ่นและความหนาแน่น รวมทั้งความถี่ของคลื่นที่ใช้ยิ่งความถี่สูงพลังงานยิ่งถูกดูดซับมาก

2.2 อัลตราโซนิคทรานสดิวเซอร์

2.2.1 ปรากฎการณ์เพียโซอิเล็กทริก

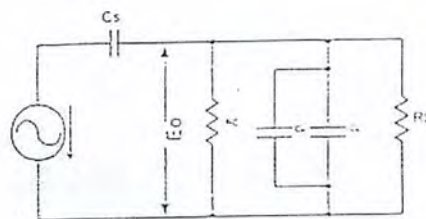
เพียโซอิเล็กทริกเป็นปรากฏการณ์ธรรมชาติอย่างหนึ่ง ซึ่งทำให้พลังงานสามารถเปลี่ยนแปลงจากรูปหนึ่งไปเป็นอีกรูปหนึ่งได้ กล่าวคือถ้าป้อนแรงกลให้แก่ Solek Crystalline Dielectric ดังในรูปที่ 2.9 ก็จะทำให้เกิดความเค้น (Stress) ภายในคริสตอลและทำให้ผลึกของคริสตอลผิดรูปไปเช่นพวกควอทซ์(Quartz) ผลก็คือประจุเปลี่ยนไป การผิดรูปร่างของแลททิซเป็นผลให้ความสัมพันธ์ระหว่างการแทนที่ (Displacement) ของประจุบวกและลบในแลททิซเปลี่ยนไป การแทนที่ของประจุภายในจะเท่ากับประจุภายนอกของขั้วที่ตรงข้ามของคริสตอล เรียกว่าผลของเพียโซอิเล็กทริก

การวัดประจุทำได้โดยการต่ออิเล็กโทรด (electrod) เข้าที่ผิวนอก แล้ววัดความต่างศักย์ระหว่างขั้วทั้งสอง ขนาด(Magnetude) และการมีขั้ว (Polarity) ของประจุบนผิวที่ถูกเหนี่ยวนำ(induce) เป็นสัดส่วนโดยตรงกับขนาดและทิศทางของแรงที่มากระทำ



รูปที่ 2.9 ผลของเพียโซอิเล็กทริก

วงจรเสมือนของเพียโซอิเล็กทริกทรานสดิวเซอร์



รูปที่ 2.10 วงจรเสมือนของเพียโซอิเล็กทริก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

C_c : คาปาซิแตนซ์ของชิ้นส่วนเปียโซอิเล็กทริก

R_c : Leagage Resistance ของเปียโซอิเล็กทริก ปกติจะมีค่าสูงประมาณ $10^8 - 10^{10}$

ความต้านทานระหว่างเทอร์มินอล โดยทั่วไปจะเป็นความต้านทานทางโหลด R_L (ในช่วง $10^6 - 10^7$)

C_L : คาปาซิแตนซ์ ของ Subsequent Stage(Load) ร่วมกับ Connection Cable

ที่ความถี่ปานกลางและความถี่สูง โวลต์เตจ E_0 ที่คร่อมโหลดหาได้โดย C_c และ C_L ค่าเหล่านี้ มาได้โดยการใช้การแบ่ง โวลต์เตจ โดยโวลต์ที่เตจ E_0 ได้จากสมการ

$$E_0 = E(C_c / (C_c + C_L))$$

โดยขึ้นกับความถี่ ถ้าเอาที่พหุโวลต์เตจมีค่ามากสามารถทำให้ลดลงได้โดยการเพิ่มค่า C_L เช่นการนำค่า C_c ต่อขนานเข้าไปที่ความถี่ต่ำ โวลต์เตจ E_0 คร่อมโหลดหาได้โดยค่ารีแอคแตนซ์ของ C_c และอิมพีแดนซ์ของ C_L และ R_L ที่ต่อขนานกัน โวลต์เตจ E_0 ขึ้นกับความถี่และจะลดลงถ้าความถี่ลดลง การตอบสนองต่อความถี่สามารถปรับปรุงได้โดยการใช้ทรานสดิวเซอร์ที่มีค่าคาปาซิแตนซ์สูงหรือโดยการเพิ่มความต้านทานโหลด RL

2.2.2 วัสดุเปียโซอิเล็กทริก

วัสดุเปียโซอิเล็กทริกที่ใช้กันอย่างกว้างขวาง เช่น Quartz , Tourmaline sulphate, Barium Titanate และ Ziconmate (TZI) โดยทั่วไปแล้วพวกควอตซ์ และคริสตอลที่เป็นเปียโซอิเล็กทริกธรรมชาติมักจะมีข้อของมันเป็นเองตามธรรมชาติ แต่พวกวัสดุเปียโซอิเล็กทริกที่สังเคราะห์ขึ้นมาเช่น แบเรียม ไททานเนท เซรามิก จะต้องนำมาทำการอบคริสตอลภายในแรงดัน และวางวัสดุที่ได้นี้ในสนามไฟฟ้าที่มีแรงดันของไฟฟ้ากระแสตรงมาก หลังจากชิ้นส่วนนี้ถูกนำไปในสนามไฟฟ้าแล้ว คริสตอลนี้จะมีขั้วตามแนวของทิศทางของสนาม และประจุติดตัวตามคุณสมบัติของเปียโซอิเล็กทริก สำหรับชิ้นส่วนที่ทำจากวัสดุสังเคราะห์นี้ ไม่มีข้อจำกัดทางขนาดโดยโครงสร้างของคริสตอลและยังสามารถทำให้มีรูปร่างและขนาดต่างๆ และทิศทางของขั้วก็จะถูกสร้างขึ้นระหว่างขั้นตอนการผลิต

2.2.3 ชนิดของเปียโซอิเล็กทริกทรานสดิวเซอร์

เปียโซอิเล็กทริกทรานสดิวเซอร์สามารถแบ่งได้เป็น 2 ชนิดคือ

1. แบบ Generation-Action Transducer ใช้เป็นตัวรับโดยแรงดันไฟฟ้าที่เกิดขึ้นจะหาได้จากแรงดันและความถี่ที่มากระทำต่อวัสดุเปียโซอิเล็กทริก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2. แบบ *Motor-Action Transducer* ใช้เป็นตัวส่ง โดยการเปลี่ยนแปลงของรูปร่างที่ทำให้เกิดคลื่นอัลตราโซนิก โดยจะขึ้นอยู่กับขนาดความสูงและความถี่ของแรงดันไฟฟ้าที่ให้ในทั้งสองกรณีค่าแรงดันไฟฟ้าที่เกิดขึ้นกับขนาดของวัสดุ

2.2.4 การทำงานของทรานสดิวเซอร์ตัวส่งตัวรับ

เมื่อเซรามิกได้รับสัญญาณแรงดันมาครบวงจรจะทำให้ชิ้นสารเซรามิกโค้งงอทำให้เกิดการอัดอากาศโดยรอบเกิดเป็นคลื่นขึ้นมา ดังนั้นถ้าป้อนสัญญาณเป็นห้วงๆ (Electrically Pulse) จากการออสซิลเลทก็จะให้ชิ้นสารโค้งงอมากขึ้นหรือทิศทางใดตามขนาดและทิศทางการเปลี่ยนแปลงขนาดของสัญญาณไฟฟ้าจากการออสซิลเลทนั้นออกไป โดยทั่วไปกำลังเอาท์พุทที่ออกมาจะตกประมาณ 10% ของกำลังไฟฟ้าที่ป้อนให้แต่เอาท์พุทจะสูงที่ค่านี้ โดยประมาณก็ต่อเมื่อความถี่ของสัญญาณออสซิลเลเตอร์ที่ป้อนเข้าชิ้นสารเซรามิก ตรงกับความถี่เรโซแนนท์ที่เป็นความถี่ทางกลตามธรรมชาติของชิ้นสารเซรามิกนั้นๆ ส่วนที่ความถี่อื่นๆ กำลังเอาท์พุทจะลดลงกว่านี้ ส่วนการทำงานของทรานสดิวเซอร์ตัวรับมักมีการทำงานตรงข้ามกับตัวส่ง คือเมื่อมีคลื่นเสียงที่มีความถี่ตรงกับความถี่เรโซแนนท์ของชิ้นสารเซรามิกเข้ามา จะทำให้ชิ้นสารโค้งตัวไปมาและเกิดสัญญาณแรงดันไฟฟ้าที่มีขนาดเล็กลงขึ้นคร่อมขั้วทั้งสองของตัวมัน

คุณสมบัติทั่วไปของอัลตราโซนิกทรานสดิวเซอร์แบบเปียโซอิเล็กทริกก็คือ มีความต้านทานไฟฟ้าตรงสูงมากอาจถึง 100 เมกกะโอห์ม เรียกว่าถ้าเอาโวลต์มิเตอร์ธรรมดา ตั้งสเกลวัดค่าความต้านทานสูงๆ เข็มจะไม่กระดิกเลยแต่ขณะที่มันทำงานความต้านทานไฟฟ้าสลับจะลดลง

2.2.5 ข้อควรระวังในการใช้งานตัวรับ-ตัวส่งทรานสดิวเซอร์

1. ไม่ควรให้ตัวต้านทานทรานสดิวเซอร์ได้รับการกระแทกหรือตกจากที่สูง เพื่อป้องกันโครงสร้างภายในไม่ให้เสียหาย
2. ทรานสดิวเซอร์ที่มีขายกันโดยทั่วไปจะทนแรงดันตกคร่อมตัวมันได้ไม่เกิน $10 V_{rms}$ ดังนั้นขนาดของสัญญาณที่จะป้อนให้กับตัวทรานสดิวเซอร์ก็ควรอยู่ในขีดจำกัดนี้
3. ความถี่เรโซแนนท์ (คือความถี่ที่ตัวมันสามารถทำงานอย่างมีประสิทธิภาพที่สุด) ของทรานสดิวเซอร์ 40 KHz (Bandwidth) ที่มีขายโดยทั่วไปจะผิดพลาดไม่เกิน ± 1 KHz และมีแถบความถี่ประมาณ 4.5 KHz (Bandwidth) สำหรับตัวส่ง และมีแถบความถี่ประมาณ 5.0 KHz (Bandwidth) สำหรับตัวรับ จะเห็นได้ว่าแถบความถี่ของตัวรับจะกว้างกว่าตัวส่งเล็กน้อย เพื่อให้แน่ใจว่าตัวรับจะสามารถรับความถี่ทั้งหมดที่ออกจากตัวส่งได้
4. อุณหภูมิที่ใช้งานของตัวทรานสดิวเซอร์ควรอยู่ในช่วง -20 ถึง 60 องศาเซลเซียส

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5. ตัวส่งและตัวรับจะมีทิศทางคล้ายกันมาก กล่าวคือที่ตำแหน่งเบนจากแนวแกนของตัวส่งไปประมาณ 30 องศา ความแรงของคลื่นเสียงที่ถูกส่งออกไปจะลดลงจากแนวแกนประมาณ 10 dB ทำนองเดียวกัน ถ้าคลื่นเสียงพุ่งเข้ามาในแนวที่เบี่ยงเบนไป จากแกนของตัวรับประมาณ 30 องศาความไวหรือขนาดของแรงดันที่ออกมาจะลด จึงควรจะทำให้ตัวรับและตัวส่งอยู่ในแนวที่พุ่งตรงมากที่สุดอย่างไรก็ตามในกรณีที่อยู่ในห้องจะเกิดการเบี่ยงเบนกันได้มาก เพราะคลื่นเสียงอัลตราโซนิก จะสะท้อนกับกำแพงและวัสดุที่อยู่ในห้อง ทำให้คลื่นเสียงเข้าไปหาตัวรับได้หลายทาง

อัลตราโซนิกนี้สามารถทำให้เป็นลำแคบได้โดยที่ใช้เลนส์ที่เรียกว่า Planacone acelenes วางข้างหน้าของทรานสดิวเซอร์ แต่การทำให้เป็นลำแคบ(Beam) นี้ จะทำให้ near field สั่นลงและ far field กระจายกว้างขึ้น การทำให้เป็นลำคลื่นเหมาะสำหรับการใช้ทรานสดิวเซอร์ที่สร้างคลื่นสูงที่เหมาะสมสำหรับที่ใช้ในงานแพทย์(Ultrasound) ที่มีความถี่ตั้งแต่ 2-5 MHz เพื่อใช้ตรวจเนื้อเยื่อที่มีการส่งเป็นลำแคบ

6. ในกรณีที่ใช้งานตัวรับจะต้องมีความต้านทานต่อขานกับตัวรับที่ทำหน้าที่เป็นโหลดตามปกติแล้วตัวต้านทานนี้ควรมีค่าอยู่ระหว่าง $10\text{ K}\Omega$ ถึง $100\text{ K}\Omega$ จากการทดลองพบว่าถ้าเปลี่ยนจาก $100\text{ K}\Omega$ มาเป็น $10\text{ K}\Omega$ ความไวจะลดลงประมาณ 10-12 dB แต่ความถี่จะกว้างขึ้น ถ้าความต้านทานต่ำลงไปอีก ความถี่เรโซแนนท์ (ความถี่กลาง) จะลดลงไปจากที่ระบุไว้ ถ้าการใช้งานมีสัญญาณรบกวนมากควรใช้โหลดที่มีค่าความต้านทานสูงสักหน่อย เพื่อให้ตัวส่งมีความไวสูงและมีความถี่แคบ

7. ตามปกติแล้วสามารถนำตัวส่งและตัวรับมาใช้งานแทนกันได้ ในการใช้งานส่วนใหญ่ตัวส่งและตัวรับรุ่นใด ก็สามารถใช้แทนกันได้ในงานส่วนใหญ่ ขอเพียงแต่ให้มีความถี่เรโซแนนท์เดียวกันเท่านั้นเอง อย่างไรก็ตามในบางกรณีอาจต้องเปลี่ยนแปลงค่าความต้านทานสมมูลย์ทางไฟฟ้าทางด้านไฟฟ้าสลับเพื่อให้เกิดลักษณะผลตอบสนองทางความถี่สอดคล้องกับของเดิม

8. ประโยชน์การใช้งานคลื่นอัลตราโซนิก คลื่นอัลตราโซนิกเป็นคลื่นที่มีทิศทาง ทำให้สามารถเล็งคลื่นไปตามเป้าหมายที่ต้องการได้เจาะจง ยิ่งคลื่นที่มีความถี่สูงขึ้นความยาวคลื่นจะยิ่งสั้นลง ถ้าความยาวคลื่นยาวกว่าช่องเปิด(ที่ทำให้เสียงออกมา) ของตัวที่ทำให้เกิดเสียงความถี่นี้ เช่น คลื่นความยาว 300Hz ในอากาศ จะมีความยาวคลื่นถึงประมาณ 1 เมตรกว่าๆ ซึ่งจะยาวกว่าช่องเปิดที่ทำให้เสียงออกมาจากตัวกำเนิดเสียงโดยทั่วไปมากมาย คลื่นจะหักเหที่ขอบด้านนอกของตัวกำเนิดเสียงที่ทำให้เกิดการกระจายทิศทางของคลื่น แต่ถ้าความถี่สูงขึ้นมาอยู่ในย่านอัลตราโซนิก อย่างเช่น 40 KHz จะมีความยาวคลื่นในอากาศเพียง 8 มิลลิเมตร เท่านั้น ซึ่งเล็กกว่าตัวที่ทำให้เกิดคลื่นความถี่นี้มากๆ คลื่นเสียงจะไม่มีกรเลี้ยวเบนที่ขอบ จึงพุ่งออกมาเป็นลักษณะลำแคบๆหรือที่เราเรียกว่า “มีทิศทาง” นั่นเองการมีทิศทางของคลื่นเสียงอัลตราโซนิกทำให้เราสามารถนำเอาไปใช้งานได้หลาย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

อย่าง เช่น คลื่นวัดความหนาแน่นของวัตถุ โดยสังเกตระยะที่คลื่นสะท้อนกลับมา เครื่องวัดความลึก ทำแผนที่ใต้ท้องทะเล ส่วนการใช้งานทางการแพทย์ อาจใช้ความถี่ในช่วง 1MHz ถึง 10MHz ขณะที่ความถี่เป็น GHz ก็มีใช้กันหลายๆการใช้งานที่ตัวกลางที่คลื่นเสียงเดินทางผ่าน ไม่ใช่อากาศ

2.3 ความรู้เบื้องต้นของพอร์ตขนาน

พอร์ตขนาน(Parallel port) เนื่องด้วยการถ่ายถอดข้อมูลของพอร์ตนี้เป็นแบบขนาน ทำให้พอร์ตขนานมีอัตราการถ่ายถอดข้อมูลสูงกว่า การถ่ายถอดข้อมูลแบบอนุกรมประมาณ 8-10 เท่า และการประมวลผลข้อมูลส่วนใหญ่จะมีขนาด 8 บิต ดังนั้นพอร์ตขนานจึงสามารถรองรับการถ่ายถอดข้อมูล 8 บิตได้โดยไม่ต้องต่อส่วนเพิ่มเติมใดๆ

2.3.1 ลักษณะทางกายภาพของพอร์ตขนาน



รูป 2.11 แสดงไคอะแกรมเวลาของการส่งข้อมูลไปยังเครื่องพิมพ์

เพื่อให้เข้าใจหลักการการทำงานของพอร์ตขนาน ก่อนอื่นต้องมาทำความเข้าใจก่อนว่า ปกตินั้น การสั่งพิมพ์งานจากคอมพิวเตอร์ไปยังพอร์ตขนานนั้นมีรูปแบบการทำงานภายในอย่างไร ในรูปที่ 2.11 แสดงไคอะแกรมเวลาของติดต่อระหว่างพอร์ตขนานกับเครื่องพิมพ์ ซึ่งจะเห็นได้ว่ามีสัญญาณที่ใช้งานจริงๆ มีไม่มาก เริ่มจากสัญญาณพอร์ต Data ถูกส่งออกไปยังเครื่องพิมพ์ พร้อมทั้งส่งสัญญาณ Strobe ออกไปด้วยเพื่อให้เครื่องพิมพ์รับรู้ว่าการส่งข้อมูลใหม่มาที่ขา Data แล้วจากนั้นคอมพิวเตอร์จะต้องรอการตอบกลับจากเครื่องพิมพ์ นั่นคือเครื่องพิมพ์จะสร้างสัญญาณ Busy หรือ เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เพื่อบอกว่าเครื่องพิมพ์ยังไม่พร้อมที่จะรับข้อมูลใหม่ จนกระทั่งเมื่อเครื่องพิมพ์พร้อม เครื่องพิมพ์จะสร้างสัญญาณ ACK ส่งไปยังคอมพิวเตอร์เพื่อแจ้งว่า พร้อมที่จะรับข้อมูลใหม่แล้ว

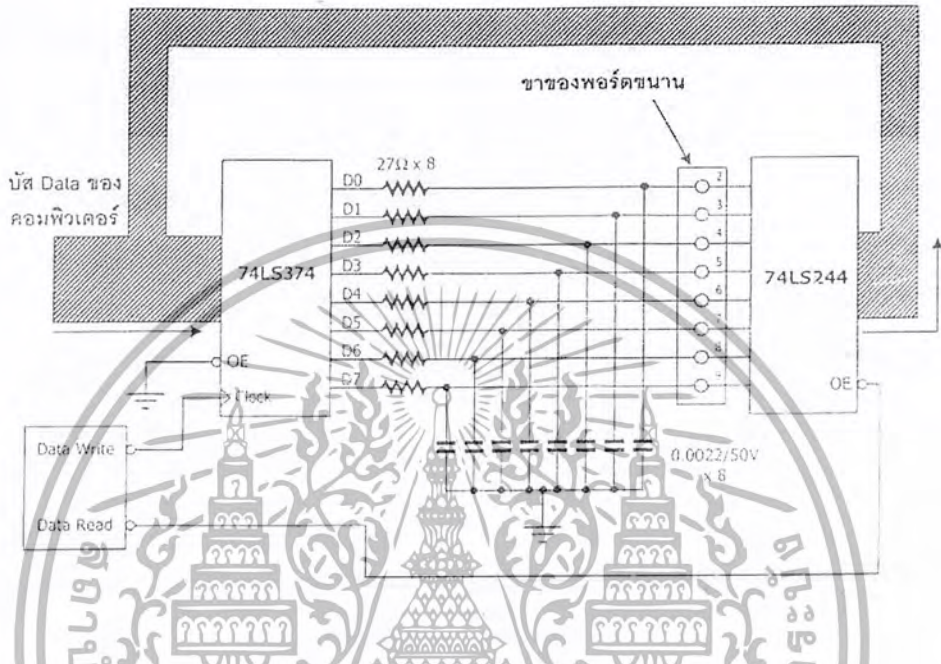
สัญญาณข้อมูลขนาด 8 บิต สัญญาณ Strobe และสัญญาณ ACK (Acknowledge) เป็นสัญญาณที่สำคัญในการส่งข้อมูลจากคอมพิวเตอร์ไปยังเครื่องพิมพ์ นอกจากสัญญาณทั้งสามแล้ว ส่วนใหญ่การติดต่อกับเครื่องพิมพ์ยังต้องมีสัญญาณอื่นๆร่วมด้วย เนื่องจากเครื่องพิมพ์ต้องทำหน้าที่ถึง 3 อย่างด้วยกันคือ รับข้อมูลจากคอมพิวเตอร์ , พิมพ์ข้อมูลที่รับเข้ามา และตอบสนองต่อการใช้งานของผู้ใช้ เช่น การเปลี่ยนฟอนต์ เป็นต้น บางครั้งอาจเกิดเหตุการณ์ที่ไม่ปกติ เช่น บัฟเฟอร์สำหรับรับข้อมูลเต็ม(เนื่องจากเครื่องพิมพ์เป็นอุปกรณ์ที่ทำงานทางกลย่อมทำงานได้ช้ากว่าการส่งข้อมูลของคอมพิวเตอร์) เครื่องพิมพ์จะต้องแจ้งไปยังคอมพิวเตอร์ว่าให้หยุดส่งข้อมูลชั่วคราว เนื่องจากไม่สามารถรับข้อมูลมากกว่านี้ได้แล้ว สัญญาณที่ส่งจากเครื่องพิมพ์ไปยังคอมพิวเตอร์คือสัญญาณ Busy และเมื่อเครื่องพิมพ์เกิดข้อผิดพลาด เช่นกระดาษติด เครื่องพิมพ์จะต้องแจ้งไปยังคอมพิวเตอร์เช่นกัน โดยสัญญาณที่แจ้งไปยังคอมพิวเตอร์เรียกว่าสัญญาณ Error นอกจากนี้เมื่อคอมพิวเตอร์ต้องการรีเซ็ตเครื่องพิมพ์คอมพิวเตอร์จะต้องส่งสัญญาณ Reset ไปยังเครื่องพิมพ์เพื่อรีเซ็ตเครื่องพิมพ์ด้วย สามารถสรุปงานสัญญาณที่จำเป็นสำหรับการติดต่อดังตารางที่ 2.1 จะเห็นได้ว่าพอร์ตขนานของคอมพิวเตอร์ยังแยกย่อยออกเป็นอีก 3 พอร์ต ได้แก่ พอร์ตเอาต์พุตที่ทำหน้าที่ส่งข้อมูลจากคอมพิวเตอร์ไปยังเครื่องพิมพ์ พอร์ตเอาต์พุตสำหรับสัญญาณ Strobe และ Reset พอร์ตอินพุตสำหรับการอ่านค่าสัญญาณ Acknowledge, Busy และสัญญาณ Error จากเครื่องพิมพ์

สัญญาณ	หน้าที่การทำงาน	ทิศทาง
ข้อมูล 8 บิต	ข้อมูลที่ส่งจากคอมพิวเตอร์ไปยังเครื่องพิมพ์	คอมพิวเตอร์
Strobe	แจ้งเครื่องพิมพ์ถึงข้อมูลที่ส่งมาใหม่	คอมพิวเตอร์
Acknowledge	เครื่องพิมพ์แจ้งมายังคอมพิวเตอร์ว่าได้รับข้อมูลแล้ว	เครื่องพิมพ์
Busy	แจ้งสถานะว่าเครื่องพิมพ์ไม่ว่างที่จะรับข้อมูลใหม่	เครื่องพิมพ์
Error	แจ้งสถานะว่าเครื่องพิมพ์เกิดข้อผิดพลาด	เครื่องพิมพ์
Reset	รีเซ็ตเครื่องพิมพ์	คอมพิวเตอร์

ตารางที่ 2.1 สัญญาณสำคัญของพอร์ตขนานที่ใช้ติดต่อกับเครื่องพิมพ์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่นิยมนำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.3.2 พอร์ตดาต้า (Data Port)



รูปที่ 2.13 วงจรภายในของพอร์ต Data

จากรูป 2.13 แสดงให้เห็นว่าพอร์ต Data ประกอบไปด้วยบัพเฟอร์ 1 ตัว และไอซีแลตซ์อีก 1 ตัว เมื่อคอมพิวเตอร์ต้องการส่งข้อมูลไปยังเครื่องพิมพ์ คอมพิวเตอร์จะเขียนข้อมูลไปยังไอซีแลตซ์ 1 ทั้ง 8 บิตเอาต์พุตของไอซีแลตซ์ 1 คือ D0-D7 ซึ่งเอาต์พุตนี้จะไปปรากฏอยู่ที่พอร์ตขนานในตำแหน่งขา 2 ถึงขา 9 และที่ขาเอาต์พุตนี้สัญญาณ Data จะส่งกลับไปเป็นอินพุตของบัพเฟอร์ 1 ด้วย ทำให้คอมพิวเตอร์สามารถอ่านค่าสถานะปัจจุบันที่เกิดขึ้นกับพอร์ต Data ได้

เมื่อคอมพิวเตอร์ส่งข้อมูล ข้อมูลจะถูกส่งมาจากบัสข้อมูลของคอมพิวเตอร์ผ่านไปที่บัพเฟอร์ไอซี 74LS374 ซึ่งเป็นไอซีแลตซ์ข้อมูล และเมื่อต้องการให้ข้อมูลปรากฏที่เอาต์พุต คอมพิวเตอร์จะส่งสัญญาณ Data write ออกไปที่ขา CLK ของ 74LS374 เอาต์พุตจาก 74LS374 จะถูกกรองด้วยวงจร RC ซึ่งประกอบด้วยตัวต้านทานค่า 27Ω และตัวเก็บประจุ $0.0022\mu\text{F}$ เพื่อให้ช่วงเวลาที่เปลี่ยนจากลอจิก "0" เป็นลอจิก "1" หรือจากลอจิก "1" เป็นลอจิก "0" เป็นไปอย่างช้าๆ เนื่องจากการเปลี่ยนแปลงแรงดันที่รวดเร็วทำให้เกิดสัญญาณรบกวนเหนี่ยวนำข้ามไปยังข้อมูลบิตอื่นๆ ได้ทำให้ข้อมูลที่ส่ง

ออกไปมีข้อผิดพลาด จากค่าตัวต้านทานและตัวเก็บประจุในวงจรทำให้เกิดการหน่วงเวลาไปประมาณ 60 นาโนวินาที จากวงจรในรูป 2.13 ทำให้เอาต์พุตของพอร์ต Data มีคุณสมบัติดังนี้

- กระแสซิงค์สูงสุด 24 mA
- กระแสซอร์สสูงสุด 2.6 mA
- ระดับแรงดันของลอจิก “1” ต่ำสุดเท่ากับ 2.4 V
- ระดับแรงดันสูงสุดสำหรับลอจิก “0” เท่ากับ 0.5 V

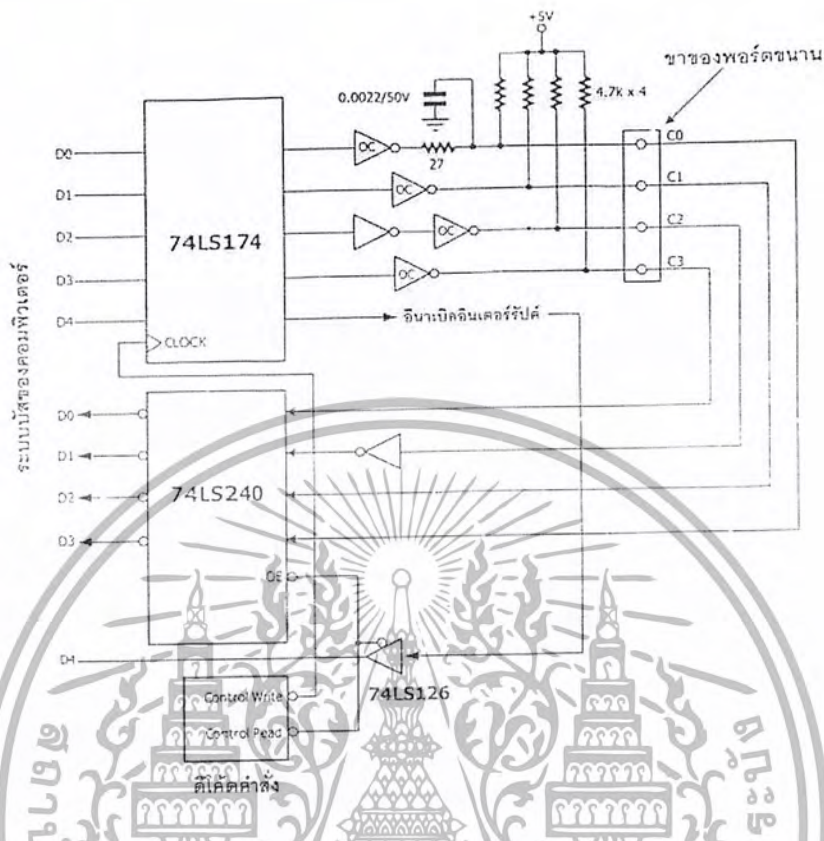
สำหรับบัฟเฟอร์สำหรับการอ่านข้อมูลกลับได้แก่เบอร์ 74LS244 ซึ่งเมื่อต้องการอ่านค่าคอมพิวเตอร์จะส่งสัญญาณ Dataread ออกมาเพื่ออินพุตไอซี 74LS244 สำหรับพอร์ตขนานแบบมาตรฐาน (Standard Parallel Port : SPP) พอร์ต Data จะต้องใช้เพื่อการส่งค่าออกเอาต์พุตเท่านั้น แต่สำหรับพอร์ตขนานที่มีการสื่อสารสองทิศทาง (Bidirectional Parallel Port) สามารถอ่านค่าจากพอร์ต Data ได้ด้วย แต่ก่อนที่จะอ่านค่าต้องจำไว้เสมอว่าจะต้องป้อนค่าเอาต์พุตให้มีค่าลอจิก “1” ทั้งหมดก่อน

2.3.3 พอร์ต Control

พอร์ต Control ใช้สำหรับคอมพิวเตอร์ควบคุมเครื่องพิมพ์จากรูปที่ 2.12 จะเห็นว่าพอร์ต Control ประกอบไปด้วยบิตเอาต์พุต 4 บิตที่ต่อออกไปยังเครื่องพิมพ์ ส่วนบิตอินพุตอินเทอร์รัปต์ไม่ได้ถูกต่อออกไปรูปที่ 2.14 แสดงบล็อกไออะแกรมของพอร์ต Control เอาต์พุตของพอร์ต Control มีอินเวอร์เตอร์แบบคอลลีเกเตอร์เปิดต่อรวมอยู่ โดยเอาต์พุตเหล่านี้จะถูกพูลอัพไว้ด้วยตัวต้านทานค่า 4.7 K Ω สำหรับบิต C2 จะผ่านอินเวอร์เตอร์ถึงสองตัวทำให้ที่เอาต์พุตของบิต C2 ไม่มีการกลับสถานะลอจิก

สถานะของพอร์ต Control สามารถอ่านกลับได้โดยการใช้บัฟเฟอร์เบอร์ 74LS240 ซึ่งเอาต์พุตของ 74LS240 มีอินเวอร์เตอร์อยู่ภายใน ทำให้ค่าที่อ่านได้ตรงกับค่าที่ส่งออกไป การควบคุมการอ่านและเขียนข้อมูลกับพอร์ต Control คอมพิวเตอร์จะส่งข้อมูลมาที่ขา Control Write และ Control Read

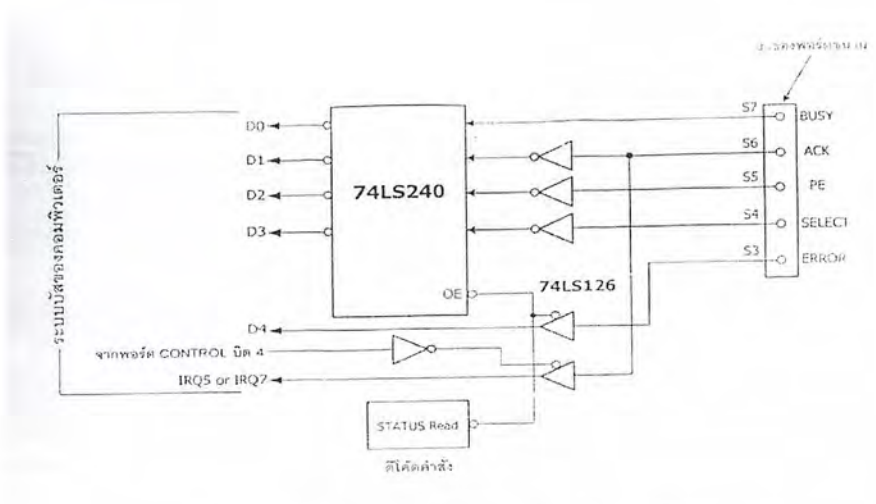
เนื่องจากเอาต์พุตของพอร์ต Control เป็นแบบคอลลีเกเตอร์เปิด ดังนั้นผู้ใช้งานสามารถใช้พอร์ตนี้ในการอ่านค่าสัญญาณอินพุตจากภายนอกได้ โดยก่อนที่จะอ่านค่าจะต้องทำให้ขาพอร์ตที่ต้องการอ่านค่ามีลอจิก “1” เสียก่อน



รูปที่ 2.14 วงจรภายในของพอร์ต Control

2.3.4 พอร์ตแสดงสถานะหรือพอร์ต Status

พอร์ต Status เป็นพอร์ตที่คอมพิวเตอร์ใช้สำหรับการอ่านค่าสถานะจากเครื่องพิมพ์ รูปที่ 2.15 แสดงรายละเอียดภายในของพอร์ต Status จะสังเกตเห็นว่ามีขาสัญญาณอยู่ทั้งหมด 5 สัญญาณด้วยกันและจะเรียกชื่อเป็น S3,S4,S5,S6 และ S7 ซึ่งตัวเลขนั้นหมายถึงตำแหน่งบิตของขาเหล่านี้ภายในรีจิสเตอร์ Status นั้นเอง สำหรับ S7 จะมีชื่อแตกต่างจากบิตอื่นๆที่เมื่อสัญญาณจากภายนอกส่งเข้ามาแล้วจะไม่ผ่านอินเวอร์เตอร์ ในขณะที่ขาอื่นๆ ผ่านอินเวอร์เตอร์ทั้งหมด ดังนั้นเมื่อข้อมูลผ่านจากขาอินพุตไปยัง 74LS240 ซึ่งเอาท์พุทมีการกลับสถานะทำให้บิต S7 เป็นบิตเดียวที่มีการกลับสถานะ นอกเหนือนี้ในการใช้งานถ้าต้องการให้มีการสร้างสัญญาณอินเตอร์รัปต์จากขอบขาขึ้นของขา S6 สามารถกำหนดค่าได้จากพอร์ต Control บิต 4



รูปที่ 2.15 แสดงวงจรภาพในของพอร์ตแสดงสถานะ

2.3.5 การนำพอร์ตขนานไปใช้งาน

สำหรับพอร์ตขนานแบบมาตรฐานผู้ใช้งานสามารถนำพอร์ตอินพุต 5 บิต(พอร์ต Status) พอร์ตเอาต์พุต 4 บิต(พอร์ต Control)และพอร์ตเอาต์พุตอีก 8 บิต(พอร์ต Data)ไปใช้งานได้โดยตรง โดยที่ 4บิตของพอร์ตเอาต์พุตหรือพอร์ต Control นั้นสามารถดัดแปลงให้ใช้งานเป็นพอร์ตอินพุต ขนาด 4 บิตได้ด้วยดังนั้นผู้ใช้งานจึงสามารถนำสัญญาณจากพอร์ตขนานที่มีมากถึง 17 เส้นไปใช้งานในการควบคุม โดยใช้ระดับสัญญาณ TTL

2.3.6 การเขียนโปรแกรมติดต่อกับพอร์ตขนาน

พอร์ตขนานของคอมพิวเตอร์จะมีลักษณะเช่นเดียวกับอุปกรณ์อินพุตเอาต์พุตตัวอื่นๆ คือ เมื่อต้องการติดต่อจะต้องกำหนดแอดเดรสที่ต้องการติดต่อด้วย ตารางที่2.2 แสดงแอดเดรสของพอร์ตขนาน โดยแบ่งออกเป็น 3 ตำแหน่งคือ แอดเดรสของรีจิสเตอร์ Data , รีจิสเตอร์ Status และ รีจิสเตอร์ Control โดยแอดเดรสนี้จะมีอยู่ทั้งหมด 3 ชุดสำหรับพอร์ตขนาน 3 ชุดคือ LPT1,LPT2 และ LPT3

ชื่อพอร์ต	LPT1:		LPT2:		LPT3:	
	ฐานสิบ	ฐานสิบหก	ฐานสิบ	ฐานสิบหก	ฐานสิบ	ฐานสิบหก
DATA	888	378H	956	3BCH	632	278H
STATUS	889	379H	957	3BDH	633	279H
CONTROL	890	37AH	958	3BEH	634	27AH

ตารางที่2.2 แสดงแอดเดรสของพอร์ตขนาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.4 FPGA(Field Programmable Gate Array)

โครงสร้างภายในของ FPGA ประกอบไปด้วยอะเรย์ของลอจิกเกตต่างๆมากมาย ซึ่งในปัจจุบันความจุเกตภายในตัวชิพ FPGA ได้เพิ่มขึ้นจากระดับไม่กี่พันตัวจนระดับล้านตัว ซึ่งสามารถรองรับวงจรดิจิทัลที่มีความสลับซับซ้อนได้เป็นอย่างดี นอกจากนี้ในด้านการออกแบบพัฒนาและทดสอบก็ทำได้ง่าย ซึ่งในปัจจุบันการออกแบบวงจรโดยใช้ FPGA กำลังเป็นที่นิยมและมีแนวโน้มที่จะนำมาใช้งานมากขึ้นเรื่อยๆ

2.4.1 FPGAแบ่งได้ตามเทคโนโลยีที่ใช้ในการโปรแกรมซึ่งมีอยู่ 2 แบบคือ

การโปรแกรมโดยการทำให้เกิดการเปลี่ยนแปลงทางกายภาพของตัวชิพ และการโปรแกรมโดยการใช้หน่วยความจำ

1. การโปรแกรมโดยการทำให้เกิดการเปลี่ยนแปลงทางกายภาพ
 - 1.1 *Fuse* เป็นวิธีการ โปรแกรมที่สามารถทำได้เพียงครั้งเดียว ซึ่งหลังจากที่โปรแกรมแล้วจุดเชื่อมต่อจะขาดจากกัน
 - 1.2 *Anti Fuse* เป็นวิธีการ โปรแกรมที่คล้ายกับแบบ *Fuse* แต่ต่างกันที่หลังจากทำการโปรแกรมแล้ว จะเชื่อมต่อจะเชื่อมถึงกัน

2. การโปรแกรมโดยใช้หน่วยความจำ

2.1 *EEPROM Based FPGA*

FPGA ที่ใช้การโปรแกรมแบบนี้มันเรียกว่า CPLD ซึ่งเทคโนโลยีที่ใช้จะเหมือนกับ EEPROM ทำให้มีความจุของเกตต่ำ โดยทั่วไปจะน้อยกว่า 20,000 เกต แต่ข้อดีของ EEPROM Based FPGA คือสามารถเก็บข้อมูลที่โปรแกรมลงไปได้โดยไม่จำเป็นต้องมีไฟเลี้ยง และในการโปรแกรมจะใช้ทรานซิสเตอร์ 1 ตัวต่อ 1 บิต ซึ่งการโปรแกรมสามารถทำได้ประมาณ 10,000 ครั้ง

2.2 *SRAM Based FPGA*

FPGA แบบนี้จะใช้เทคโนโลยีในการโปรแกรมเหมือนกับ SRAM (Static RAM) ทำให้สามารถโปรแกรมซ้ำได้โดยไม่จำกัดจำนวนครั้ง นอกจากนี้ยังมีความจุของเกตในระดับปานกลางถึงสูงมาก (ประมาณ 10,000 – 1,000,000 เกต) ซึ่งข้อดีของ SRAM Based FPGA คือใช้เวลาในการโปรแกรมน้อย (ระดับ ns) การโปรแกรมทำได้ง่ายเทียบได้กับการเขียน SRAM ทั่วไป และเหมาะสำหรับการออกแบบวงจรที่มีความสลับซับซ้อนส่วนข้อเสียคือไม่สามารถเก็บโปรแกรมในภาวะที่ไม่มีไฟเลี้ยงได้ ดังนั้น FPGA ชนิดนี้จึงมักใช้ควบคู่กับ ROM เพื่อเก็บโปรแกรมและทำการโหลดโปรแกรมลงในตัวชิพในขณะที่เริ่มต้นใช้งาน

2.4.2 โครงสร้างภายในของ FPGA

ลักษณะโครงสร้างภายในของ FPGA จะเป็นระเบียบของบล็อกลอจิกสามารถทำการโปรแกรมได้ดังรูปที่ 2.16 และ 2.17



รูปที่ 2.16 โครงสร้างภายใน FPGA ตระกูล MAX7000S

รูปที่ 2.17 โครงสร้างภายใน ของ FPGA ตระกูล FIEX10K

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่นับญาติให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.4.3 รายละเอียดของขั้นตอนในการออกแบบสามารถอธิบายได้ดังนี้

1. การสังเคราะห์วงจร (Logic Synthesis)

ในขั้นตอนนี้จะใช้ซอฟต์แวร์ในการสังเคราะห์วงจร (Synthesis Tools) ทำการสังเคราะห์พฤติกรรมของวงจรที่ได้จากการออกแบบด้วย Schematic หรือ VHDL ซึ่งต้องทำการตรวจสอบด้วยว่าซอฟต์แวร์นั้นสนับสนุนเทคโนโลยี FPGA (FPGA Library) ที่ต้องการหรือไม่ ตัวอย่างเช่น FPGA ของบริษัท XILINX และบริษัท ALTIUM จะมีซอฟต์แวร์สังเคราะห์วงจรจะทำการแปลงโค้ด VHDL และทำการ Optimize เพื่อให้ได้วงจรตามเทคโนโลยี ที่เลือกใช้ ในการสังเคราะห์วงจรนี้วงจรระดับเกต (Gate Level) จะไม่เหมาะสมกับโครงสร้างที่มีอยู่ในอุปกรณ์ FPGA นั้นๆ จึงทำให้ผลที่ได้มีประสิทธิภาพและในขั้นตอนการสังเคราะห์วงจรนี้ ผู้ออกแบบสามารถกำหนดข้อบังคับสำหรับโมเดลแต่ละตัวได้ เช่น ข้อบังคับในเรื่องเวลา (Timing Constraints) หรือข้อบังคับในเรื่องของพื้นที่ (Area) หรือกำหนดชนิดและตำแหน่งของ I/O ซึ่งข้อบังคับเหล่านี้จะถูกนำไปใช้ในขั้นตอน Optimize เพื่อให้วงจรที่ได้เป็นไปตามที่กำหนด ส่วนสำคัญในการ Optimize คือการเทียบ (Mapping) โมเดลให้เข้ากับเทคโนโลยีที่ใช้เพื่อให้ได้วงจรที่เหมาะสมกับโครงสร้างและสถาปัตยกรรมภายในอุปกรณ์ FPGA เมื่อทำการสังเคราะห์วงจรเสร็จแล้ว ซอฟต์แวร์การสังเคราะห์วงจรก็จะมีรายงานผลว่าโมเดลที่ออกแบบไปนั้นเป็นอย่างไร เช่นมีค่าความหน่วง(Delay) เท่าใดใช้ทรัพยากรต่างๆใน FPGA อะไรบ้าง เมื่อมาถึงขั้นตอนนี้ ผู้ออกแบบก็จะทราบว่าโมเดลเป็นไปตามข้อบังคับหรือไม่ ถ้าไม่ก็สังเคราะห์ใหม่จนกว่าจะเป็นไปตามที่กำหนด

2. การแบ่งวงจร (Partitioning)

ขั้นตอนนี้เป็นการแบ่งวงจรที่ได้จากการสังเคราะห์ เป็นส่วนย่อยๆ สำหรับลงใน CLBs, IOBs หรือองค์ประกอบอื่นๆ ภายในอุปกรณ์ FPGA สำหรับเกณฑ์ที่ใช้ในการแบ่ง คือให้แต่ละส่วนที่จะแยกออกจากกันมีจำนวนสัญญาณที่เชื่อมต่อระหว่างกันน้อยที่สุดเท่าที่จะทำได้ เพื่อลดความหนาแน่นในตอนที่ทำการเชื่อมต่อสัญญาณ (Routing) ในขั้นตอนนี้จะใช้ซอฟต์แวร์ทำ โดยซอฟต์แวร์จะเทียบส่วนประกอบของวงจรเช่น เกต(gate), ฟลิป-ฟลอป (flip-flop) ลงในทรัพยากรต่างๆ ที่มีอยู่ในอุปกรณ์ FPGA

หลังจากทำขั้นตอนนี้เสร็จแล้วผู้ออกแบบสามารถที่จะทราบว่าวงจรใช้จำนวนทรัพยากรภายในอุปกรณ์ FPGA ไปเท่าไร ส่วนข้อมูลทางเวลานั้นผู้ออกแบบจะทราบเฉพาะความหน่วงลอจิก (Logic Delay) ส่วนซอฟต์แวร์จะรวมเอาซอฟต์แวร์ย่อยอื่นๆ อีก เพื่อให้การทำ PPR (Partitioning Placement & Routing) เป็นไปอย่างต่อเนื่อง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่นิยมนำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3. การวางอุปกรณ์ (Placement)

ขั้นตอนนี้เป็นการเลือกทำเลที่ตั้งของแต่ละส่วนของวงจรที่ผ่านการแบ่งวงจร (Partitioning) มาแล้วว่าจะอยู่ ณ ตำแหน่งไหนในอุปกรณ์ FPGA เพื่อให้ได้ผลลัพธ์ที่ดีที่สุด เช่น วงจรส่วนไหน ควรอยู่ใกล้กันเพื่อจะได้ค้นหาเส้นทางได้ (route) ง่ายหรือช่วยลดความหน่วง จะเห็นได้ว่าตำแหน่งภายในอุปกรณ์ FPGA นั้นมีความสำคัญเพราะถ้าจัดวางวงจรลงในตำแหน่งที่ไม่เหมาะสมแล้วจะทำให้ความหน่วงเพิ่มขึ้นหรือ Router ทำการค้นหาเส้นทางสัญญาณได้ไม่หมด

การวางอุปกรณ์ที่ดีควรวางส่วนต่างๆ ให้อยู่ใกล้กันโดยเฉพาะส่วนที่มีการเชื่อมต่อสัญญาณด้วยกันนอกจากนั้นการกำหนดตำแหน่งขา I/O (I/O pin) ตามตำแหน่งขา I/O ของ FPGA บนแผ่น PCB ก็จะมีผลโดยตรงเลยคือซอฟต์แวร์จะวาง I/O ลงในตำแหน่งที่ผู้ออกแบบกำหนด ซึ่งบางครั้งตำแหน่งที่กำหนดไปไม่เหมาะสม ดังนั้นการกำหนดขา I/O ควรกำหนดตำแหน่งให้เหมาะสม หรือไม่ก็ให้ซอฟต์แวร์จัดการเอง

4. การเชื่อมต่อสัญญาณ (Routing)

ในขั้นตอนนี้เป็นการเชื่อมต่อสัญญาณระหว่างองค์ประกอบต่างๆภายในอุปกรณ์ FPGA ขั้นตอนนี้จะทำได้เนื่องจากการวางอุปกรณ์ ในกรณีที่ทำการวางอุปกรณ์ไว้ไม่ดีซอฟต์แวร์ก็จะทำการเชื่อมต่อสัญญาณได้ไม่หมด (เนื่องจากจำนวนทรัพยากรสำหรับเชื่อมต่อสัญญาณนั้นมีอยู่จำกัด) หรือเกิดความหน่วงเกินค่าที่กำหนดในข้อบังคับ

ผู้ออกแบบสามารถทำขั้นตอนนี้ได้โดยใช้ซอฟต์แวร์หรือผู้ออกแบบจะทำการเชื่อมต่อสัญญาณด้วยตนเองก็ได้ แต่ทางที่ดีควรใช้ซอฟต์แวร์ทำดีกว่า นอกจากนั้นการกำหนดข้อบังคับทางเวลา จะช่วยให้ผลที่ได้จากการเชื่อมต่อสัญญาณดีขึ้นได้

5. ความหน่วงด้านเวลา (Delay)

ในการทำ FPGA นั้นความหน่วงที่เกิดขึ้นเป็นความหน่วงที่เกิดจากการวางตำแหน่ง (layout) ของอุปกรณ์ ซึ่งผู้ออกแบบไม่สามารถเข้าไปแก้ไขได้ แต่สามารถทำให้มีความหน่วงน้อยที่สุดได้ สำหรับความหน่วงที่เกิดขึ้นนั้นแยกได้เป็นสองประเภทคือ

5.1.1 ความหน่วงทางลอจิก (Logic delay) เป็นความหน่วงภายในองค์ประกอบของอุปกรณ์ FPGA เอง

5.1.2 ความหน่วงที่เกิดจากการเชื่อมต่อสัญญาณ (Routing delay) เป็นความหน่วงที่เกิดจากการเชื่อมต่อสัญญาณระหว่างองค์ประกอบภายในอุปกรณ์ FPGA

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โดยปกติแล้วค่าความหน่วงลอจิกไม่ควรเกิน 50% ของค่าความหน่วงที่ยอมรับได้ เพราะความหน่วงที่เกิดจากการเชื่อมต่อสัญญาณมันจะมีค่ามากกว่าค่าความหน่วงลอจิก ดังนั้นในการวางอุปกรณ์ และเชื่อมต่อสัญญาณ ผู้ออกแบบควรกำหนดข้อบังคับทางเวลาเพื่อให้ซอฟต์แวร์ได้ทำงานอย่างมีประสิทธิภาพเพิ่มขึ้น และเพื่อให้ได้ผลลัพธ์ที่ดีขึ้น

ค่าความหน่วงที่ได้หลังจากการวางอุปกรณ์ และเชื่อมต่อสัญญาณแล้วจะมีค่าความหน่วงที่ค่อนข้างแน่นอน ซึ่งผู้ออกแบบสามารถทราบได้ว่าโมเดลที่ออกแบบนั้น เป็นไปตามข้อกำหนดหรือไม่

6. การจำลองการทำงานของวงจร (Simulation)

ในขั้นตอนนี้เป็นขั้นตอนที่สำคัญอีกขั้นตอนหนึ่ง เพราะเป็นขั้นตอนที่สำคัญอีกขั้นตอนหนึ่ง เพราะเป็นขั้นตอนที่ผู้ออกแบบตรวจสอบฟังก์ชันการทำงานของโมเดลว่าถูกต้องหรือไม่ ข้อผิดพลาดตรงไหนเพื่อจะได้ทำการแก้ไขให้ถูกต้อง ในขั้นตอนนี้จะมีซอฟต์แวร์ที่ใช้สำหรับทำการจำลองการทำงานของวงจรที่ใช้อยู่ เช่น Model Sim ของบริษัท Model Technology หรือ Max Plus II ของบริษัท Altera ในการจำลองการทำงานของวงจรทำทุกครั้งหลังจากที่มีการทำแต่ละขั้น ตอนหลักเสร็จแล้ว เพื่อจะได้ทราบว่าข้อผิดพลาดของโมเดลเกิดขึ้นตอนไหน จะได้แก้ไขข้อผิดพลาดตรงขั้นตอนนี้ๆ ได้เลย ไม่ต้องมาคอยตรวจหาขั้นตอนที่ทำให้เกิดข้อผิดพลาดตรงขั้นตอนนี้ๆ นั่นคือการทำการจำลองการทำงานของวงจร ต้องทำทั้งหลังการเขียนโค้ด, การสังเคราะห์วงจร และการทำ PPR การจำลองการทำงานของวงจรหลังจากที่เขียนโค้ดเสร็จแล้วนั้น ผู้ออกแบบสามารถทราบได้แค่โมเดลทำงานถูกต้องหรือไม่เท่านั้น (functional test) ยังไม่สามารถตรวจสอบการทำงานในเชิงเวลาได้ถูกต้อง ในการจำลองการทำงานของวงจรหลังจากที่สังเคราะห์เป็นวงจรแล้ว เพื่อตรวจสอบว่าฟังก์ชันการทำงานยังคงถูกต้องหรือไม่ และค่าความหน่วงที่เกิดขึ้นเป็นไปตามข้อบังคับหรือไม่ มีข้อผิดพลาดเกิดขึ้นหรือถ้ามีจะแก้ไขให้ถูกต้อง

ในการจำลองการทำงานของวงจรหลังจากที่ทำการวางอุปกรณ์การเชื่อมต่อสัญญาณ(Post layout simulation) แล้วก็มีความสำคัญเช่นกันเพราะผลที่ได้จากการจำลองการทำงานของวงจรในตอนนี้จะเป็นผลลัพธ์ของโมเดลเลย ซึ่งผู้ออกแบบนอกจากจะตรวจสอบฟังก์ชันการทำงานแล้วยังต้องตรวจสอบคุณสมบัติอื่นๆ เช่น ความหน่วงที่ได้จากการทำ PPR ในรูปแบบค่าความหน่วงมาตรฐาน (Standard Delay Format : SDF) ว่าตรงตามที่กำหนดหรือไม่ หรือตรวจสอบว่า วงจรรวมสามารถใช้งานที่ความถี่สูงสุดเท่าไรนั่นเอง ในการจำลองการทำงานของ วงจรควรใช้ซอฟต์แวร์ตัวเดียวกันตลอดเพื่อจะได้เปรียบเทียบผลที่ได้จากขั้นตอนต่างๆ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

7. การโปรแกรมอุปกรณ์ FPGA (Configuration)

หลังจากที่โมเดลผ่านขั้นตอนต่างๆ จนกระทั่งผ่านการทำ PPR (Partitioning, Placement & Routing) แล้วนั้น ถึงตอนนี้ก็สามารถที่จะดาวน์โหลด (Download) ลงในอุปกรณ์ FPGA ได้แล้ว ใน การดาวน์โหลดนี้ก่อนอื่นต้องแปลงแบบวงจรรวมที่ได้เป็นข้อมูลวงจร (Configuration data) ซึ่งอยู่ในรูปของบิตสตรีม (Bit stream) ก่อนแล้วจึงดาวน์โหลดลงไปเพื่อให้อุปกรณ์ FPGA มีฟังก์ชันการทำงานตามโมเดลที่ผู้ออกแบบต้องการ ซึ่งในขั้นตอนนี้จะใช้วิธีที่แตกต่างกันออกไปสำหรับอุปกรณ์ FPGA ของแต่ละบริษัทผู้ผลิตคือ

ในกรณีที่เป็นอุปกรณ์ FPGA ชนิดที่ต้องโปรแกรมโดยวิธี SRAM นั้นในการใช้งานผู้ออกแบบจะต้องเก็บข้อมูลวงจรไว้ในหน่วยความจำประเภท EPROM หรือ Serial PROM ด้วยเพื่อจะใช้งานสะดวกขึ้น คือในการใช้งานโมเดลครั้งต่อไปไม่ต้องดาวน์โหลดข้อมูลวงจรจากเครื่องคอมพิวเตอร์อีก เพราะมีข้อมูลวงจรเก็บอยู่ในหน่วยความจำอยู่แล้ว แต่กรณีที่อุปกรณ์ FPGA เป็นชนิดที่โปรแกรมโดยวิธี EPROM หรือ Anti fuse ก็ไม่จำเป็นต้องมีหน่วยความจำสำหรับเก็บข้อมูลวงจร เพราะว่าอุปกรณ์ FPGA ชนิดนี้เมื่อดาวน์โหลดข้อมูลวงจรลงไป ข้อมูลที่ดาวน์โหลดลงไปก็ยังคงอยู่ในอุปกรณ์ FPGA และครั้งต่อไปก็ใช้งานโมเดลที่ออกแบบไว้ได้เลย

2.5 OpenGL

OpenGL เป็นซอร์สแวร์ไลบรารีที่ใช้ติดต่อกับฮาร์ดแวร์แสดงผล โดยมีคำสั่งพื้นฐานประมาณ 120 คำสั่งที่สามารถใช้กำหนดคุณลักษณะและควบคุมการทำงานของแอปพลิเคชัน 3 มิติ OpenGL ถูกออกแบบมาโดยไม่ยึดติดกับระบบ สามารถทำงานได้บนทุกแพลตฟอร์ม (Portability) เพื่อที่จะบรรลุเป้าหมายนี้ OpenGL จะไม่มีคำสั่งที่จัดการกับระบบเลย อีกทั้งยังไม่มีคำสั่งเพื่อรับอินพุตจากผู้ใช้อีกด้วย หน้าที่ทั้งสองอย่างนี้จึงตกอยู่กับผู้เขียนโปรแกรม อย่างไรก็ตามยังมีซอร์สแวร์เพิ่มเติมที่ช่วยจัดการงานทั้งสองนี้หากพัฒนาโปรแกรมบนระบบปฏิบัติการแบบ Windows (GLUT : OpenGL Utility Toolkit) นอกจากนี้ OpenGL ยังไม่มีคำสั่งระดับสูงที่จะใช้วาดวัตถุ 3 มิติที่ซับซ้อนอย่างเช่น รถยนต์ อวัยวะ หรือ โมเลกุล สิ่งที่ OpenGL เตรียมไว้ให้สำหรับสร้างรูปจำลอง 3 มิติคือรูปทรงเรขาคณิตอย่างง่าย ได้แก่ จุด เส้น และรูปหลายเหลี่ยม ซึ่งผู้ใช้งานจะต้องนำรูปทรงเหล่านี้มาประกอบกันเพื่อให้เกิดรูปทรง 3 มิติที่ซับซ้อน

ความแตกต่างระหว่าง OpenGL กับ DirectX คือ DirectX เป็นชุดพัฒนาซอฟต์แวร์เกมของไมโครซอฟท์ซึ่งสนับสนุนการทำงานหลายด้านทั้งภาพกราฟิก 2 มิติ 3 มิติ เสียง อุปกรณ์อินพุต และการเล่นพร้อมกันหลายคน DirectX มีการเปลี่ยนแปลงอย่างมากทุกครั้งที่มีการออกเวอร์ชันใหม่ แต่ยังคงรองรับการทำงานของคำสั่งในเวอร์ชันเก่า อย่างไรก็ตามยังคงต้องทำงานบนระบบปฏิบัติการของไมโครซอฟท์และเครื่อง Xbox เท่านั้น ตัวไลบรารีของ OpenGL ต่างจาก DirectX เนื่องจากมันสนับสนุนการทำงานเฉพาะกราฟิก 2 มิติ และ 3 มิติ ผู้ที่จะเขียนเกมต้องใช้ไลบรารีตัวอื่นช่วยเพื่อให้เกมสมบูรณ์ แต่ชุดคำสั่งของ OpenGL สามารถทำงานได้บนทุกแพลตฟอร์ม การทำงานค่อนข้างเสถียร

คำสั่งและชนิดของข้อมูลของ OpenGL

`gl_color 3 f v`

OpenGL	Command	Number of Input	Input Data type	Vector
--------	---------	-----------------	-----------------	--------

การใช้งานคำสั่ง OpenGL จะต้อง include ไฟล์ header ชื่อ gl.h คำสั่ง OpenGL จะขึ้นต้นด้วย gl เช่น `glClearColor()` คำสั่งที่จะขึ้นต้นด้วย GL_ เช่น `GL_COLOR_BUFFER_BIT` สำหรับตัวลงท้ายหรือ suffix ของบางคำสั่งจะประกอบด้วยตัวเลขและตัวอักษร เช่น `glColor3f` ซึ่งเลข 3 บ่งบอกถึงจำนวนตัวแปรอินพุตคือสี RGB ส่วน f หมายถึงอินพุตจะต้องเป็นข้อมูลชนิดเลขทศนิยม(float)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

GLUT : OpenGL Utility Toolkit

GLUT เขียนขึ้น โดย Mark Kilgard เป็นเครื่องมือช่วยติดต่อระหว่างแอปพลิเคชันของ OpenGL กับระบบปฏิบัติการแบบ Windows หรือเรียกว่าเป็น API (Application Programming Interface) ทำให้การเขียนโปรแกรมโดยใช้ OpenGL มีความซับซ้อนน้อยลง เหมาะสำหรับการพัฒนาโปรแกรมขนาดเล็กถึงขนาดกลาง คำสั่งของ GLUT จะขึ้นต้นด้วยคำว่า glut



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 3

การออกแบบและการทำงานของเม้าท์สามมิติ

3.1 แนวทางการออกแบบเม้าท์สามมิติ

3.1.1 จุดประสงค์ก่อนออกแบบ

1. วงจรภาคส่งและรับคลื่นอัลตราโซนิกสามารถทำงานได้
2. ควบคุมให้สามารถวัดระยะทางระหว่างตัวรับ ตัวส่งได้
3. นำผลต่างของเวลาที่รับได้ส่งเข้าไปที่ FPGA ไปเปรียบเทียบกับสัญญาณนาฬิกา เพื่อที่จะนำความต่างของสัญญาณนาฬิกาไปเขียนเป็น โปรแกรมแสดงผล 3 มิติ
4. เขียนโปรแกรมในส่วนแสดงผล 3 มิติโดยใช้ Visual Basic

3.1.2 ขั้นตอนการออกแบบ

เพื่อใช้บอกตำแหน่งของตัวชี้ตำแหน่ง บนจอคอมพิวเตอร์ โดยมีหลักการทำงานประมาณได้ดังนี้ เริ่มด้วย ตัวชี้ตำแหน่งส่งสัญญาณไปที่ตัวรับ โดยตัวรับจะมีอยู่ 5 ตำแหน่ง คือ P0,P1,P2,P3 และ P4 ตามลำดับ ที่ตำแหน่ง $(X_0, Y_0, Z_0), (X_1, Y_1, Z_1), (X_2, Y_2, Z_2), (X_3, Y_3, Z_3)$ และ (X_4, Y_4, Z_4) ตามลำดับ การหาพิกัด (X, Y, Z) ทำได้โดยการวัดระยะทางระหว่างตัวชี้ไปยังจุดอ้างอิงทั้ง 5 คือ $R+R_0, R+R_1, R+R_2, R+R_3$ และ $R+R_4$ โดยใช้ FPGA เป็นตัวควบคุม จากนั้นก็จะส่งข้อมูลของระยะทางทั้ง 5 ไปยังคอมพิวเตอร์ โดยออกทางพอร์ตขนานของบอร์ด FPGA เพื่อให้คอมพิวเตอร์นำค่าที่ได้ไปคำนวณหาพิกัด (X, Y, Z) และนำไปแสดงผลเป็นภาพสามมิติ

ภาคส่งคลื่น

เนื่องจากตัวส่งสัญญาณ Ultrasonic จะมีความถี่ใช้งานที่ 40 KHz วงจรภาคส่งจึงเป็นวงจรผลิตความถี่ 40 KHz ป้อนให้กับตัวส่ง แต่เนื่องจากต้องเป็นตัวป้อนสัญญาณเพื่อเป็นจุดอ้างอิงให้กับ FPGA ด้วย ซึ่งสัญญาณที่จะป้อนให้กับ FPGA ต้องเป็นสัญญาณพัลส์ที่มีความถี่ต่ำ ดังนั้น วงจรภาคส่งนี้จะทำหน้าที่ผลิตความถี่ 40 KHz และผลิตสัญญาณพัลส์แล้วนำมาผสมกัน ป้อนให้กับตัวส่งอัลตราโซนิกและสัญญาณพัลส์ที่ผลิตขึ้นมานี้ก็จะถูกส่งให้กับ FPGA ด้วยพร้อมๆกัน

ภาครับคลื่น

วงจรในส่วนของภาครับคลื่น จะต้องสามารถรับสัญญาณที่ส่งมาจากตัวส่ง ซึ่งเป็นสัญญาณที่ถูกผสมกันระหว่างความถี่สูง 40 KHz กับสัญญาณพัลส์ความถี่ต่ำประมาณ 100 Hz โดยจะสร้างวงจรภาครับขึ้นมา 5 ชุด เมื่อวงจรภาครับตัวใดตัวหนึ่งได้รับสัญญาณเป็นตัวแรก เคาน์เตอร์ตัวอื่นจะเริ่มนับและหยุดเมื่อวงจรภาครับชุดนั้นได้รับสัญญาณครบทั้ง 5 จุด หรือ รอจนถึงเวลาที่ได้คำนวณไว้ว่าถ้าเวลาผ่านไปเท่านี้ภาครับยังรับสัญญาณได้ไม่ครบให้ตัวรับเริ่มรับค่าใหม่ ค่าที่ได้จะนำไปคำนวณหาตำแหน่งของตัวชี้ต่อไป

ภาคประมวลผล

วงจรในส่วนของ FPGA ที่ทำหน้าที่รับสัญญาณจากวงจรภาครับ เปรียบเทียบและนับระยะเวลาที่เกิดขึ้นระหว่างตัวส่งกับตัวรับแต่ละจุด ค่าเวลาที่ได้จะส่งให้กับคอมพิวเตอร์คำนวณผ่านทางพอร์ตสื่อสารแบบขนาน จากนั้น โปรแกรมก็จะทำการอ่านค่าเพื่อนำไปคำนวณหาพิกัดระนาบ X Y Z พร้อมทั้งแสดงออกเป็นภาพ 3 มิติโดยใช้โปรแกรมที่เขียน โดย Visual Basic

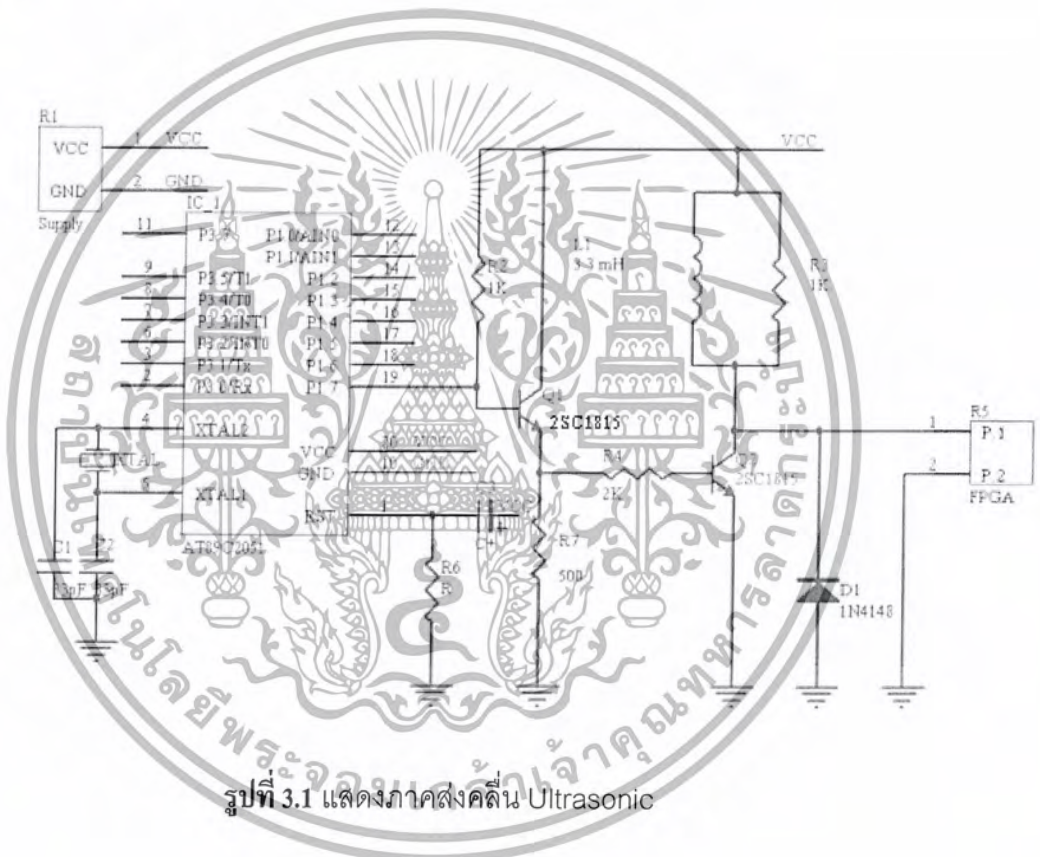


เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.2 การทำงานของเม้าส์ 3 มิติ

วงจรภาคส่งคลื่น

ภาคส่งสัญญาณจะทำหน้าที่ในการให้กำเนิดสัญญาณ ซึ่งคลื่นที่เราใช้ในที่นี้จะเป็นคลื่นอัลตราโซนิก โดยตัวส่งสัญญาณที่เราใช้จะสามารถทำงานได้ดีที่สุดที่ความถี่ 40 kHz ดังนั้นเราจึงสร้างสัญญาณนาฬิกาที่ความถี่ 40 KHz และสัญญาณความถี่ต่ำประมาณ 100 Hz ซึ่งที่นี่ได้ใช้ MCS-51 เป็นแหล่งกำเนิดสัญญาณ



รูปที่ 3.1 แสดงภาคส่งคลื่น Ultrasonic

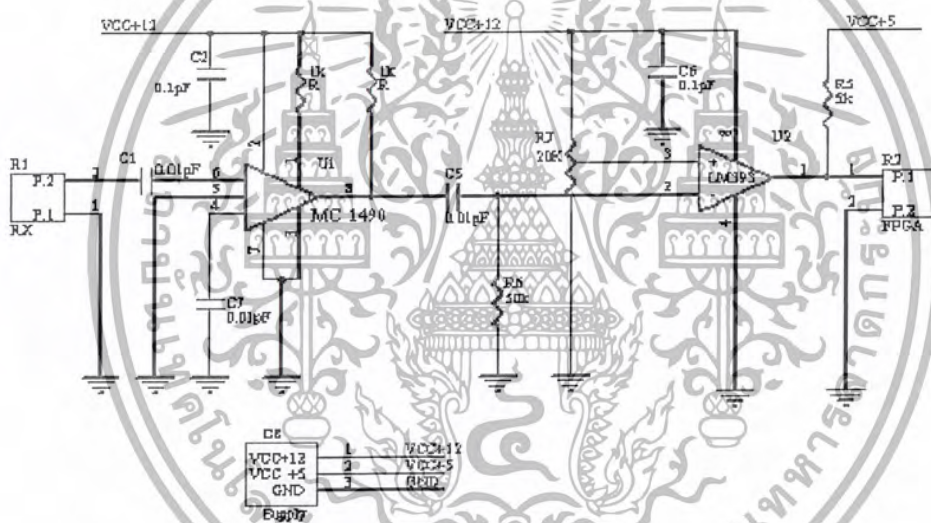
การทำงานของวงจรเราสามารถแบ่งได้เป็น 2 ส่วน โดยส่วนแรกคือ ส่วนกำเนิดสัญญาณในที่นี้ได้ใช้ MCS-51 เป็นตัวกำเนิดสัญญาณ 40 KHz และ 100 Hz ให้ output ออกที่ขา P1.7 สัญญาณมีลักษณะเป็นพัลส์รูปสี่เหลี่ยมความถี่ภายนอกประมาณ 100 Hz และความถี่ภายใน 40 KHz ส่วนที่ 2 คือ การนำสัญญาณมาผ่านวงจรบัฟเฟอร์ Q1 ซึ่ง Q1 จะทำงานอยู่สองสถานะคือ สภาวะคัตออฟ กับ สภาวะอิ่มตัว ดังนั้นสัญญาณที่เอาท์พุทจะเปลี่ยนแปลงตามไฟเลี้ยง โดยสามารถควบคุมคุณสมบัติการทำงานของ Q1 ได้ด้วยการกำหนดค่า R2 จากนั้นนำสัญญาณที่ได้มาขยายกำลังให้มากขึ้น โดยในที่นี้ได้ใช้คุณสมบัติของ L ($V_L = L di/dt$) ซึ่งจะทำให้ได้แรงดันเอาท์พุทที่สูงขึ้นเป็นประมาณ 22 -

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

24 V ที่ R3 มีค่าเท่ากับ $2.2\text{ K}\Omega$ ส่วน R3 และ R4 ทำหน้าควบคุมการทำงานและคุณสมบัติของวงจรในส่วนนี้ ส่วนไดโอดมีหน้าป้องกันการเสียหายแก่ทรานซิสเตอร์ในช่วงพัลส์ลบ

วงจรถ่ายคลื่น

หน้าที่ของวงจรถ่ายคลื่นคือการแปลงที่รับมาในรูปของสัญญาณที่ถูกผสมกันระหว่างสัญญาณความถี่สูง 40kHz กับสัญญาณความถี่ต่ำ 100Hz ให้เป็นพัลส์รูปสี่เหลี่ยม แล้วหลังจากนั้นก็จะทำการส่งสัญญาณไปในส่วนของ FPGA ต่อไป



รูปที่ 3.2 แสดงภาครับคลื่น Ultrasonic

การทำงานในส่วนนี้สามารถแบ่งออกได้เป็น 2 ส่วนด้วยกันคือ ส่วนแรกได้แก่ภาครับสัญญาณเข้ามาแล้วทำการขยายสัญญาณ โดยในที่นี้ใช้ MC 1490 เป็นตัวขยายสัญญาณ input ที่รับเข้ามาซึ่งจะได้เกณฑ์ประมาณ 150 เท่า ส่วนที่สองคือ นำสัญญาณที่ได้จากส่วนแรกมาทำการเปรียบเทียบแรงดัน ในที่นี้ได้ใช้ LM 393 เป็นตัวเปรียบเทียบโดยสามารถควบคุมได้จาก R7 ซึ่งเป็นตัวต้านทานชนิดปรับค่าได้ซึ่งจะทำให้ได้พัลส์รูปสี่เหลี่ยม แล้วจึงนำผลที่ได้ส่งต่อไปในส่วนของ FPGA ต่อไป

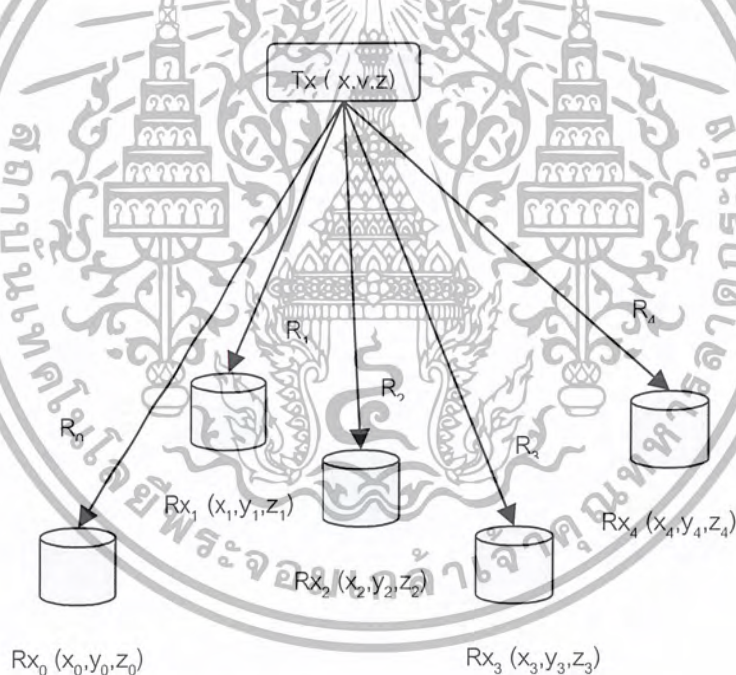
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.3 วงจรภาคประมวลผล

วงจรในส่วนของภาคประมวลผลจะประกอบไปด้วยส่วน FPGA , Visual Basic ซึ่งในส่วน
ของ FPGA จะเขียนด้วยโปรแกรม MAXplus2 โดยสร้างวงจรตัวนับ (Counter) 5 ตัวและอื่นๆ แล้ว
ทำการเชื่อมต่อกับคอมพิวเตอร์โดยผ่านทางพอร์ทขนาน ซึ่งค่าที่ได้จะส่งไปคำนวณในโปรแกรม
Visual Basic และแสดงเป็นภาพสามมิติ

3.3.1 การคำนวณหาพิกัดตำแหน่ง

การคำนวณหาพิกัดตำแหน่ง อาศัยหลักการทางคณิตศาสตร์ โดยมีวิธีการคำนวณดังนี้



รูปที่ 3.3 แสดงการวางตำแหน่งของตัวส่งและตัวรับ

สมมติว่าตัวส่งอยู่ที่ตำแหน่ง (x, y, z) และเราทำการวางตำแหน่งของตัวรับทั้ง 5 ตัว ตามจุด
ดังรูป เมื่อส่งสัญญาณ Ultrasonic จากตัวส่งไปยังตัวรับทั้ง 5 ตัว ระยะเวลาที่คลื่นเดินทางระหว่างตัว
ส่งกับตัวรับจะถูกส่งไปที่บอร์ด FPGA เพื่อนับค่าเวลาที่ต่างกัน และส่งไปที่คอมพิวเตอร์ ผ่านทาง
พอร์ทขนาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สัญญาณจากภาคส่งคลื่นจะมีลักษณะเป็น Saque Wave ที่มีความถี่ 40KHz เป็นช่วงๆมีลักษณะคล้ายๆพัลส์ ความถี่ 100 Hz ส่งไปที่ภาครับ โดยภาครับจะแปลงสัญญาณที่ได้เป็นพัลส์ที่มีขนาด 5 V เพื่อไปใช้งานต่อไป

เมื่อตัวรับตัวแรกได้รับสัญญาณดังกล่าวแล้วจะทำให้เคาน์เตอร์เริ่มทำงาน โดยรับค่าไปเรื่อยๆจนครบทุกตัว แล้วนำค่าที่ได้ไปคำนวณหาค่า R_0, R_1, R_2, R_3, R_4 ออกมาแล้วนำค่า R_0, R_1, R_2, R_3, R_4 ไปคำนวณหาค่าพิกัด x,y,z ต่อไป

การคำนวณหาพิกัด

1. การคำนวณเพื่อให้ได้ค่า R_0, R_1, R_2, R_3, R_4
เมื่อเราได้ค่าที่เคาน์เตอร์แต่ละตัวนับได้ (n) เราสามารถหาเวลาที่ใช้ได้ (t) จากสูตรต่อไปนี้

$$T = N/f$$

โดย T คือ เวลาที่แตกต่างกันระหว่างตัวรับตัวแรกกับตัวรับตัวนั้น

N คือ จำนวนที่เคาน์เตอร์นับได้

f คือ ความถี่ที่เคาน์เตอร์ใช้นับ

และเราสามารถหาค่า R_0, R_1, R_2, R_3, R_4 ได้จากสูตรต่อไปนี้

$$\text{จาก } R = VT$$

$$\text{จะได้ } R_0 = \frac{VN_0}{f}$$

$$R_1 = \frac{VN_1}{f}$$

$$R_2 = \frac{VN_2}{f}$$

$$R_3 = \frac{VN_3}{f}$$

$$R_4 = \frac{VN_4}{f}$$

โดย V คือ ความเร็วของคลื่นอัลตราโซนิก (ประมาณ 350 m/s)

R_0, R_1, R_2, R_3, R_4 คือ รัศมีที่แตกต่างกันระหว่างตัวรับตัวนั้นกับตัวรับที่อยู่ใกล้ตัวส่งมากที่สุด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หลังจากนั้นเราจะนำค่า R_0, R_1, R_2, R_3, R_4 ไปคำนวณหาค่า X, Y, Z ต่อไป

2. การคำนวณหาค่า X, Y, Z

จากรูปที่ 3.3 เราสามารถคำนวณได้ดังนี้

$$(X - X_0)^2 + (Y - Y_0)^2 + Z^2 = (R + R_0)^2 \dots\dots\dots(1)$$

$$(X - X_1)^2 + (Y - Y_1)^2 + Z^2 = (R + R_1)^2 \dots\dots\dots(2)$$

$$(X - X_2)^2 + (Y - Y_2)^2 + Z^2 = (R + R_2)^2 \dots\dots\dots(3)$$

$$(X - X_3)^2 + (Y - Y_3)^2 + Z^2 = (R + R_3)^2 \dots\dots\dots(4)$$

$$(X - X_4)^2 + (Y - Y_4)^2 + Z^2 = (R + R_4)^2 \dots\dots\dots(5)$$

จัดรูปสมการด้านบน จะได้

$$X^2 - 2X_0X + X_0^2 + Y^2 - 2Y_0Y + Y_0^2 + Z^2 = R^2 + 2R_0R + R_0^2$$

$$X^2 - 2X_1X + X_1^2 + Y^2 - 2Y_1Y + Y_1^2 + Z^2 = R^2 + 2R_1R + R_1^2$$

$$X^2 - 2X_2X + X_2^2 + Y^2 - 2Y_2Y + Y_2^2 + Z^2 = R^2 + 2R_2R + R_2^2$$

$$X^2 - 2X_3X + X_3^2 + Y^2 - 2Y_3Y + Y_3^2 + Z^2 = R^2 + 2R_3R + R_3^2$$

$$X^2 - 2X_4X + X_4^2 + Y^2 - 2Y_4Y + Y_4^2 + Z^2 = R^2 + 2R_4R + R_4^2$$

$$X^2 + Y^2 + Z^2 - R^2 = 2X_0X + 2Y_0Y + 2R_0R - X_0^2 - Y_0^2 + R_0^2 \dots\dots\dots(6)$$

$$X^2 + Y^2 + Z^2 - R^2 = 2X_1X + 2Y_1Y + 2R_1R - X_1^2 - Y_1^2 + R_1^2 \dots\dots\dots(7)$$

$$X^2 + Y^2 + Z^2 - R^2 = 2X_2X + 2Y_2Y + 2R_2R - X_2^2 - Y_2^2 + R_2^2 \dots\dots\dots(8)$$

$$X^2 + Y^2 + Z^2 - R^2 = 2X_3X + 2Y_3Y + 2R_3R - X_3^2 - Y_3^2 + R_3^2 \dots\dots\dots(9)$$

$$X^2 + Y^2 + Z^2 - R^2 = 2X_4X + 2Y_4Y + 2R_4R - X_4^2 - Y_4^2 + R_4^2 \dots\dots\dots(10)$$

นำสมการที่ (6) = (7) ; (6) = (8) ; (6) = (9) ; (6) = (10)

แล้วแทน $R = VT$

$$2(X_0 - X_1)X + 2(Y_0 - Y_1)Y + 2(T_0 - T_1)RV + V^2(T_0^2 - T_1^2) = X_0^2 - X_1^2 + Y_0^2 - Y_1^2$$

$$2(X_0 - X_2)X + 2(Y_0 - Y_2)Y + 2(T_0 - T_2)RV + V^2(T_0^2 - T_2^2) = X_0^2 - X_2^2 + Y_0^2 - Y_2^2$$

$$2(X_0 - X_3)X + 2(Y_0 - Y_3)Y + 2(T_0 - T_3)RV + V^2(T_0^2 - T_3^2) = X_0^2 - X_3^2 + Y_0^2 - Y_3^2$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

$$2(X_0 - X_4)X + 2(Y_0 - Y_4)Y + 2(T_0 - T_4)RV + V^2(T_0^2 - T_4^2) = X_0^2 - X_4^2 + Y_0^2 - Y_4^2$$

กำหนดให้

$$A_1 = 2(X_0 - X_1) \quad A_2 = 2(X_0 - X_2) \quad A_3 = 2(X_0 - X_3) \quad A_4 = 2(X_0 - X_4)$$

$$B_1 = 2(Y_0 - Y_1) \quad B_2 = 2(Y_0 - Y_2) \quad B_3 = 2(Y_0 - Y_3) \quad B_4 = 2(Y_0 - Y_4)$$

$$C_1 = 2(T_0 - T_1) \quad C_2 = 2(T_0 - T_2) \quad C_3 = 2(T_0 - T_3) \quad C_4 = 2(T_0 - T_4)$$

$$D_1 = (T_0^2 - T_1^2) \quad D_2 = (T_0^2 - T_2^2) \quad D_3 = (T_0^2 - T_3^2) \quad D_4 = (T_0^2 - T_4^2)$$

$$E_1 = X_0^2 - X_1^2 + Y_0^2 - Y_1^2 \quad E_2 = X_0^2 - X_2^2 + Y_0^2 - Y_2^2$$

$$E_3 = X_0^2 - X_3^2 + Y_0^2 - Y_3^2 \quad E_4 = X_0^2 - X_4^2 + Y_0^2 - Y_4^2$$

จะได้

$$A_1X + B_1Y + C_1RV + VD_1 = E_1$$

$$A_2X + B_2Y + C_2RV + VD_2 = E_2$$

$$A_3X + B_3Y + C_3RV + VD_3 = E_3$$

$$A_4X + B_4Y + C_4RV + VD_4 = E_4$$

$$\begin{array}{l} A_1 \\ A_2 \\ A_3 \\ A_4 \end{array} \begin{array}{|c|c|c|c|} \hline B_1 & C_1 & D_1 & \\ \hline B_2 & C_2 & D_2 & \\ \hline B_3 & C_3 & D_3 & \\ \hline B_4 & C_4 & D_4 & \\ \hline \end{array} \begin{array}{|c|} \hline X \\ \hline Y \\ \hline RV \\ \hline V^2 \\ \hline \end{array} \begin{array}{|c|} \hline E_1 \\ \hline E_2 \\ \hline E_3 \\ \hline E_4 \\ \hline \end{array}$$

กำหนดให้

$$U = \begin{array}{|c|c|c|c|} \hline A_1 & B_1 & C_1 & D_1 \\ \hline A_2 & B_2 & C_2 & D_2 \\ \hline A_3 & B_3 & C_3 & D_3 \\ \hline A_4 & B_4 & C_4 & D_4 \\ \hline \end{array}$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จะได้

$$X = \begin{array}{c} \left| \begin{array}{cccc} E_1 & B_1 & C_1 & D_1 \\ E_2 & B_2 & C_2 & D_2 \\ E_3 & B_3 & C_3 & D_3 \\ E_4 & B_4 & C_4 & D_4 \end{array} \right| \\ \hline U \end{array}$$

$$Y = \begin{array}{c} \left| \begin{array}{cccc} A_1 & E_1 & C_1 & D_1 \\ A_2 & E_2 & C_2 & D_2 \\ A_3 & E_3 & C_3 & D_3 \\ A_4 & E_4 & C_4 & D_4 \end{array} \right| \\ \hline U \end{array}$$

$$RV = \begin{array}{c} \left| \begin{array}{cccc} A_1 & B_1 & E_1 & D_1 \\ A_2 & B_2 & E_2 & D_2 \\ A_3 & B_3 & E_3 & D_3 \\ A_4 & B_4 & E_4 & D_4 \end{array} \right| \\ \hline U \end{array}$$

$$V^2 = \begin{array}{c} \left| \begin{array}{cccc} A_1 & B_1 & C_1 & E_1 \\ A_2 & B_2 & C_2 & E_2 \\ A_3 & B_3 & C_3 & E_3 \\ A_4 & B_4 & C_4 & E_4 \end{array} \right| \\ \hline U \end{array}$$

$$Z = \sqrt{R^2 - (X - X_0)^2 - (Y - Y_0)^2}$$

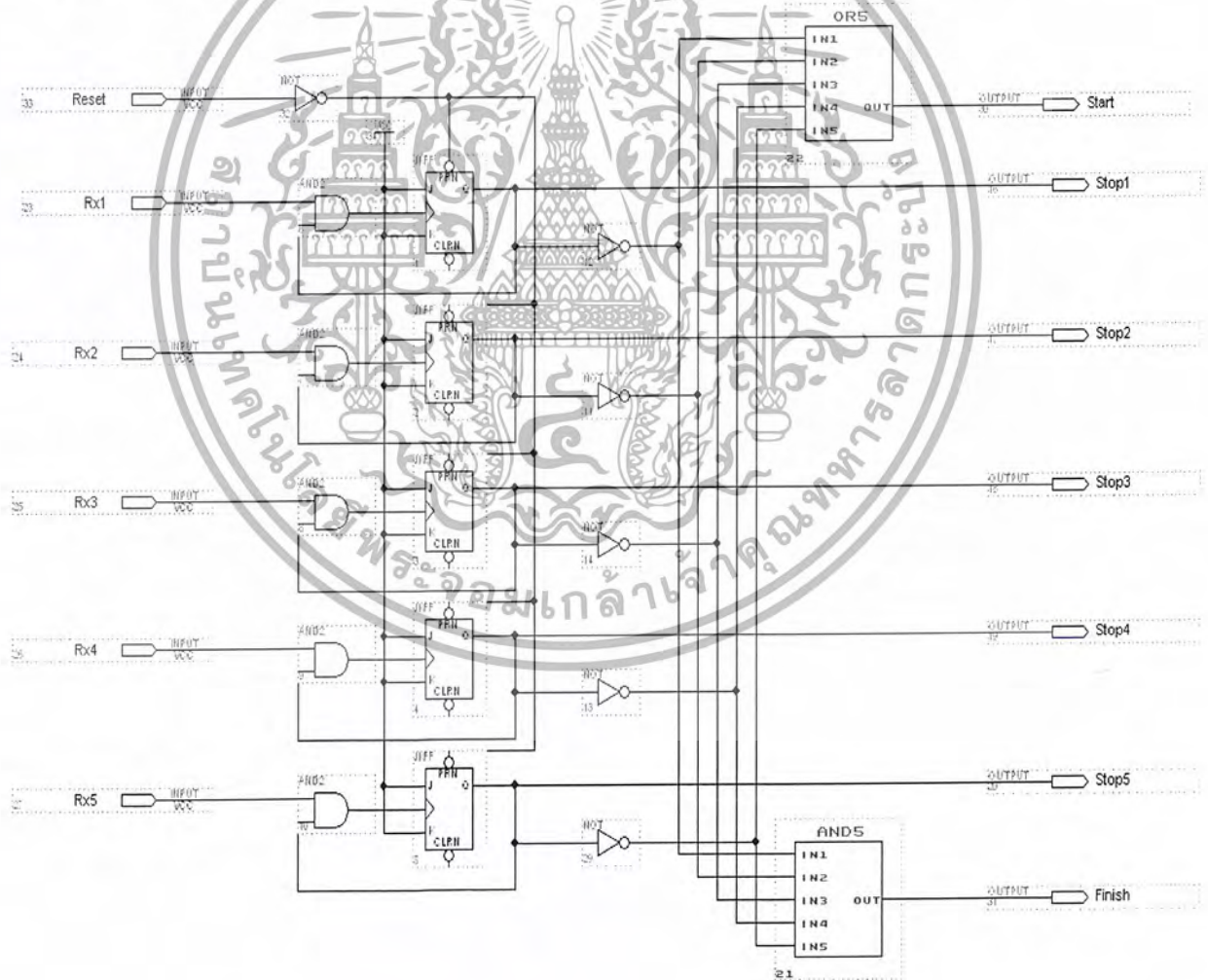
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.4 หลักการทำงานของส่วนFPGA

ในส่วนของ FPGA นี้จะมีอยู่ 2 ส่วนที่สำคัญคือส่วนของภาค Control และ ภาคCounter แล้วนำทั้งสองส่วนนี้มาประกอบเข้าด้วยกัน โดยหลักการแล้วเราต้องการให้ส่วนของ FPGA นี้ทำหน้าที่นับผลต่างและแสดงผลต่างที่นับได้ในแต่ละภาครับที่รับคลื่นอัลตราโซนิกได้ที่เวลาต่างกัน ก็เพื่อนำส่วนของผลต่างทางเวลานี้ไปเป็นข้อมูลในการคำนวณหาพิคัด X , Y , Z ต่อไปในส่วนของการคำนวณ

ภาค Control

ภาคนี้จะทำหน้าที่ในการควบคุมการเริ่มนับและการหยุดนับของ ภาคCounter ในแต่ละส่วนของภาครับ ดังนั้นภาคนี้จึงต้องทำหน้าที่ผลิตสัญญาณการเริ่มต้นการนับ และสัญญาณการหยุดนับ โดยจะต้องสัมพันธ์กับสัญญาณ Rx ในแต่ละตัวแต่ละภาครับที่จะรับคลื่นได้จากภาครับส่ง

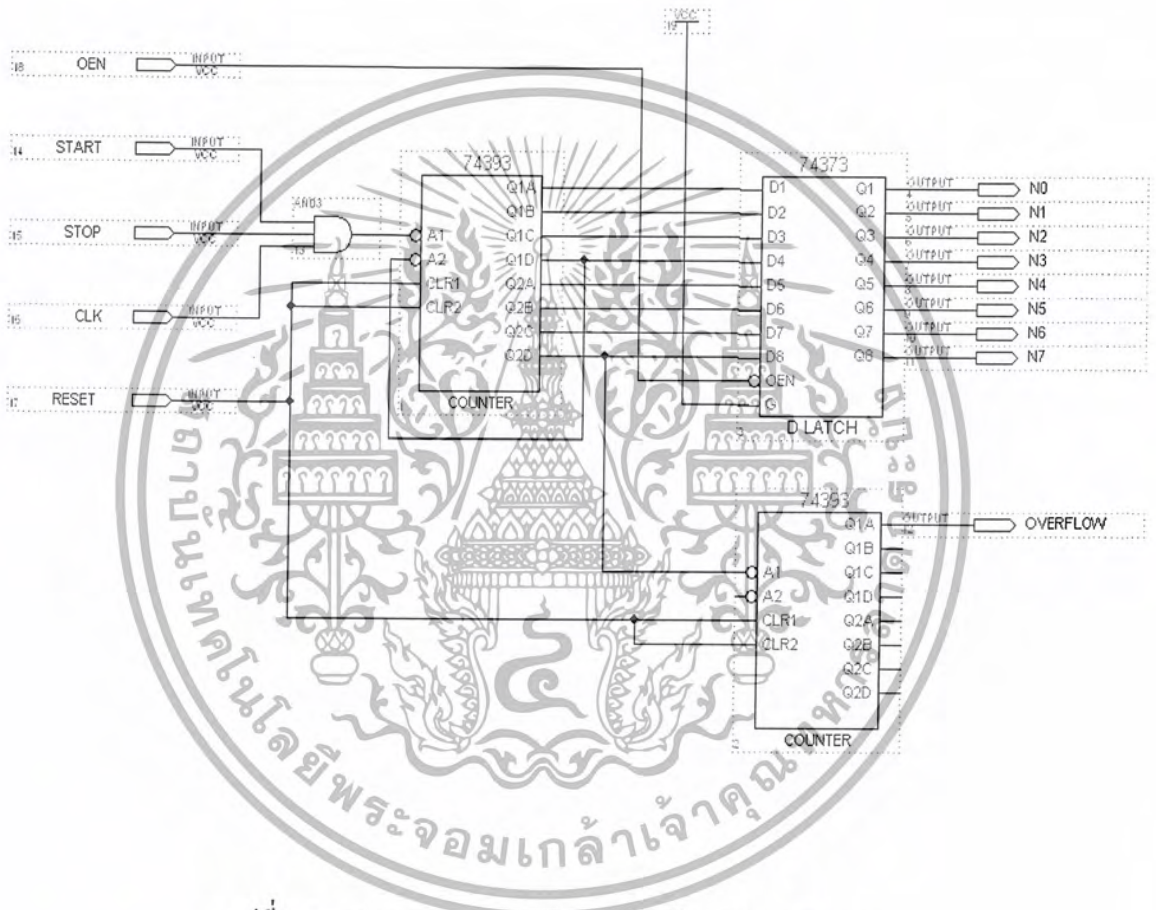


รูปที่ 3.4 แสดงภาพวงจรการทำงานภายใน Block Control

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาค Counter

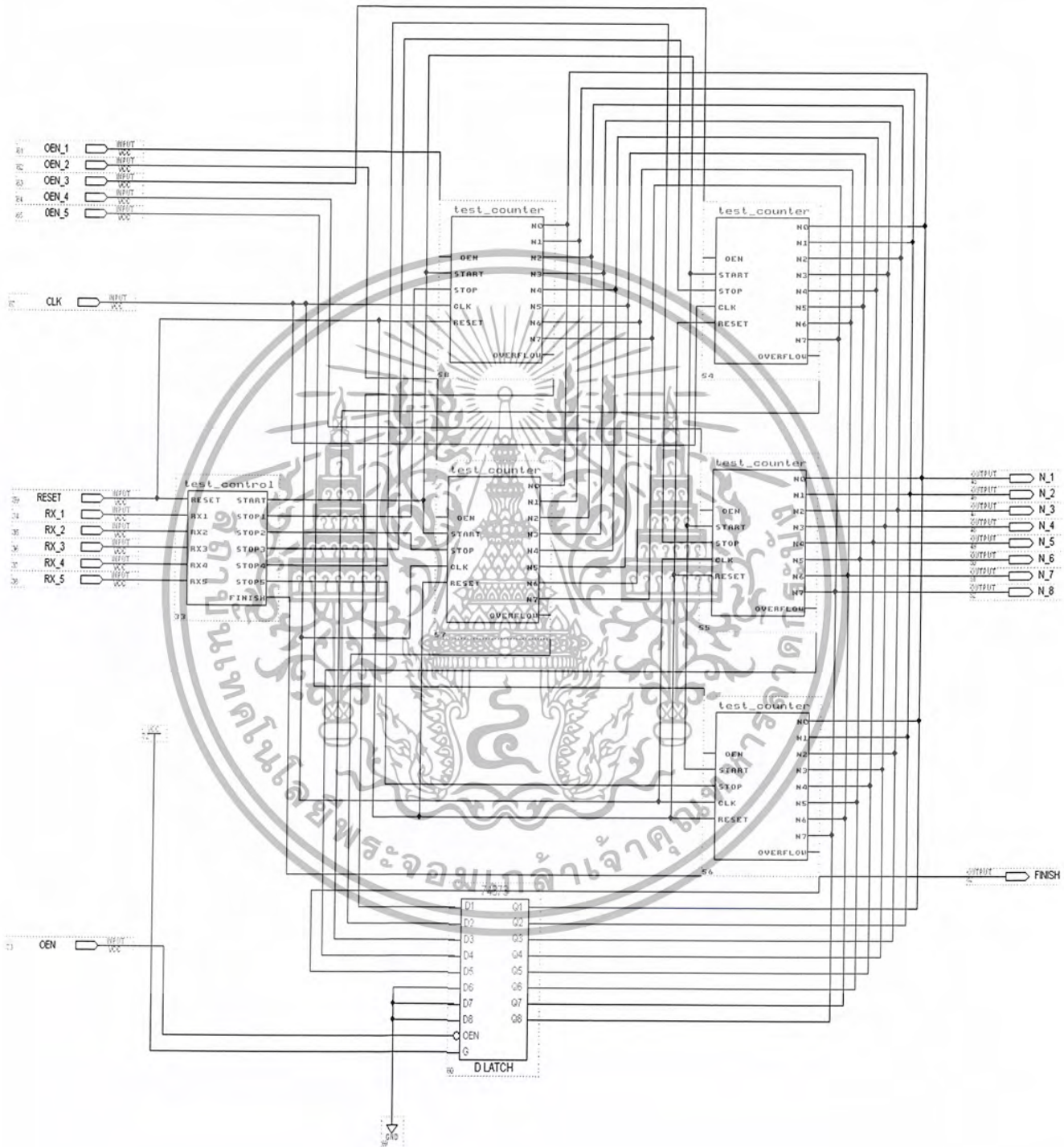
ภาคนี้จะทำหน้าที่หลักคือ นับผลต่างทางเวลาของสัญญาณ Rx ในแต่ละตัวในรูปของสัญญาณนาฬิกา ภาคนี้จะประกอบด้วย IC สองชนิดด้วยกันคือ 74393 คือ IC Counter ชนิดนับฐาน 16 8บิต และ IC 74373 คือ IC ดีแลตซ์ ดังนั้นภาคนี้จะนับได้ 0-255 ถ้าเกินนี้ก็จะส่งสัญญาณ Overflow ออกมาเสมือนว่าภาครับคลื่นภาคนั้นไม่สามารถที่จะรับคลื่นที่ส่งมาจากภาคส่งได้



รูปที่ 3.5 แสดงภาพวงจรการทำงานภายใน Block Counter

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

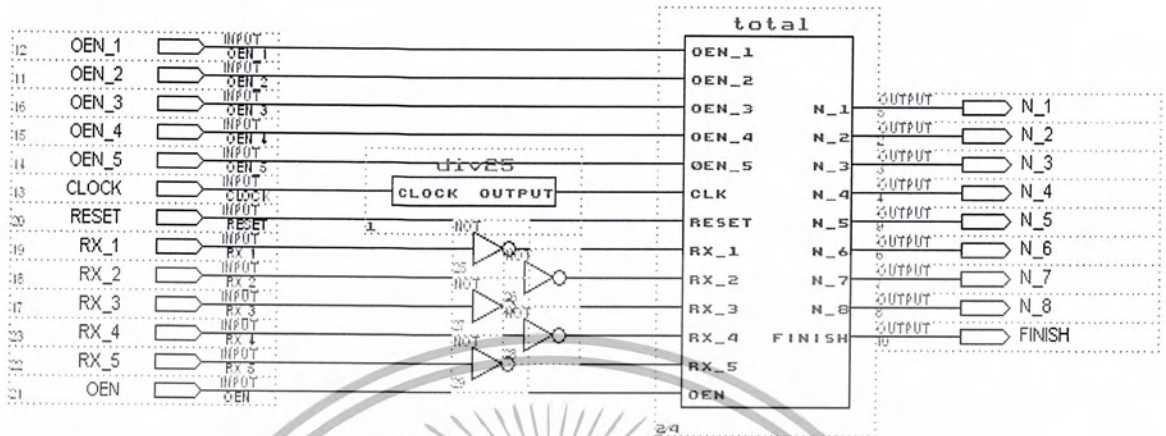
วงจร FPGA



รูปที่ 3.6 แสดงภาพวงจรในส่วนของ FPGA

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

วงจรรวม FPGA

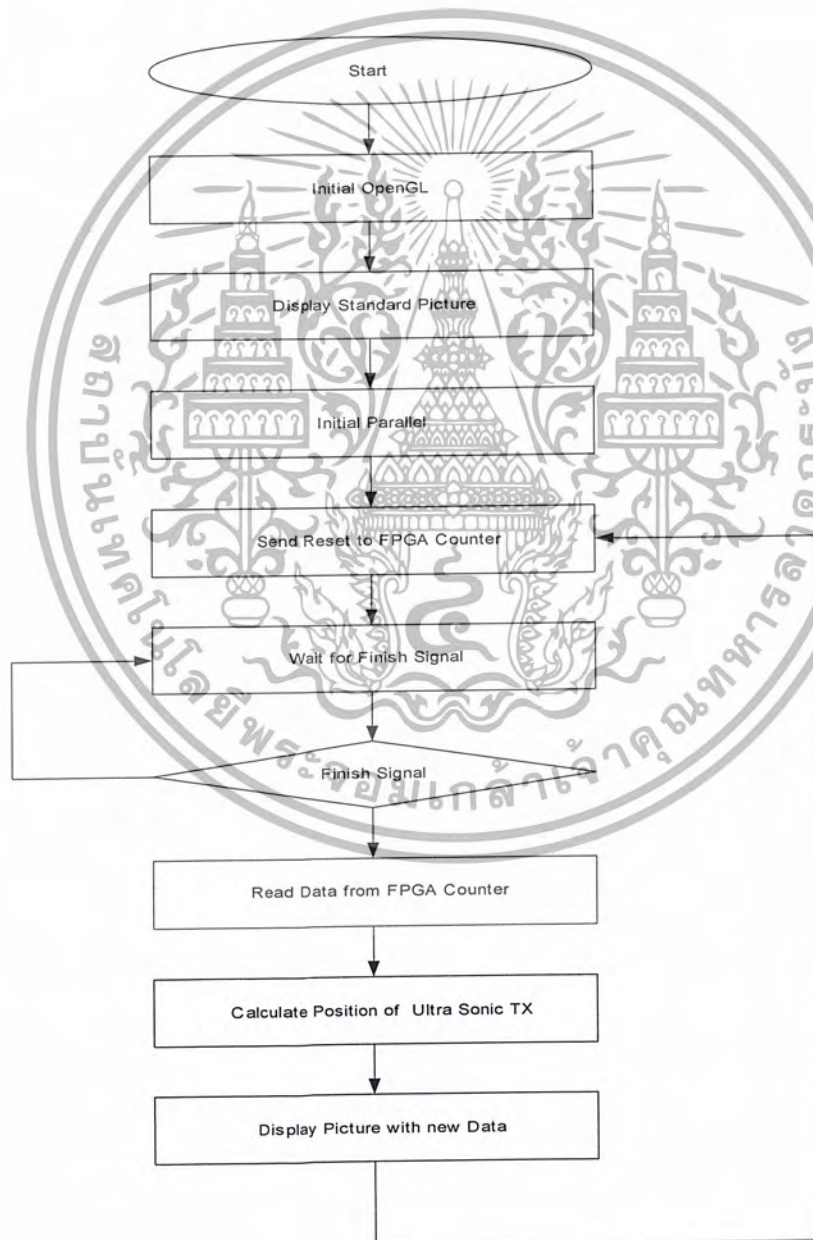


รูปที่ 3.7 แสดงภาพวงจรทั้งหมดในส่วนของ FPGA

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.5 การเขียนโปรแกรมด้วย Visual Basic

เนื่องจาก Visual Basic นั้นเป็นโปรแกรมที่ไม่สนับสนุนการทำงานในลักษณะการเชื่อมต่ออุปกรณ์ภายนอกโดยผ่านทางพอร์ตขนาน ดังนั้นเราต้องอาศัยไฟล์ Inpout32.DLL ซึ่งเป็นชุดฟังก์ชันของ API ที่ช่วยในการเชื่อมต่อผ่านทางพอร์ตขนาน ซึ่งทำให้ลดความยุ่งยากในการเขียนโปรแกรม ซึ่งไฟล์นี้สามารถหาโหลดได้จากทางเว็บไซต์อินเทอร์เน็ตทั่วไปที่มีให้ดาวน์โหลดฟรี ส่วนการทำภาพ 3 มิตินั้นเราอาศัยชุดฟังก์ชัน OpenGL ช่วยในการสร้างภาพ 3 มิติ ส่วนหลักการออกแบบโปรแกรมได้แสดงตามไฟล์ชาตด้านล่างนี้



รูปที่ 3.8 แสดงภาพ flow chart

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

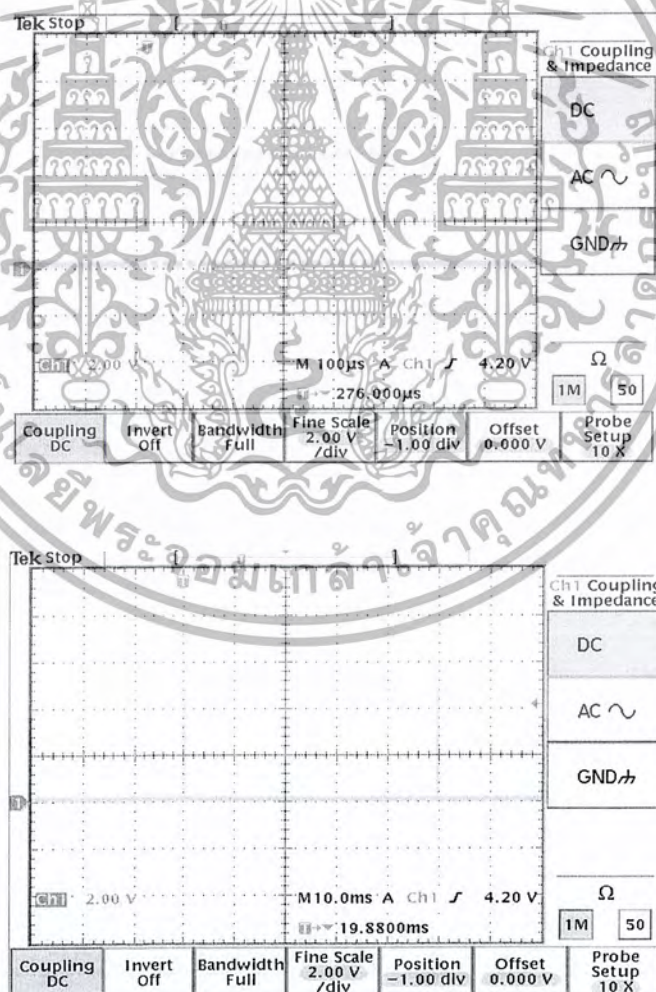
บทที่ 4

การทดลองและผลการทดลอง

ในบทที่ 4 เป็นการทดลอง และแสดงผลการทดลองในรูปแบบต่าง ๆ โดยแยกออกมาเป็นส่วนๆในการทดลองและบันทึกผล ซึ่งสามารถแยกออกเป็นข้อๆดังนี้

1. ทดสอบการส่งสัญญาณของภาครับ – ส่ง อัลตราโซนิกที่ระยะต่างๆ โดยที่ตำแหน่งหัวรับวางไว้ คัดกัน
2. ทดสอบการทำงานในส่วนของ FPGA ในภาคต่างๆ
3. ทดสอบการทำงานระหว่าง FPGA กับ Computer

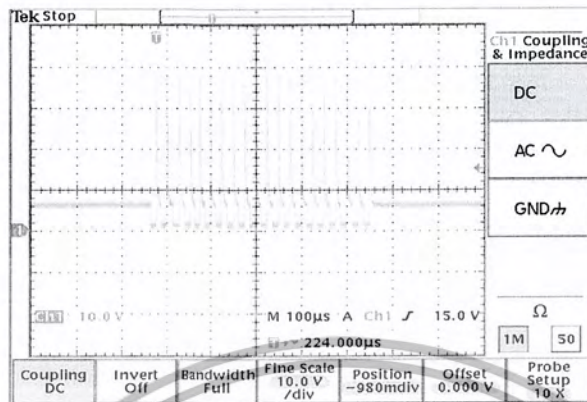
โดยเริ่มต้นได้ทำการวัดสัญญาณที่ output ที่ MCS-51 ได้รูปสัญญาณดังนี้ (ในภาคส่ง)



รูปที่ 4.1 รูปสัญญาณที่สร้างโดย MCS-51

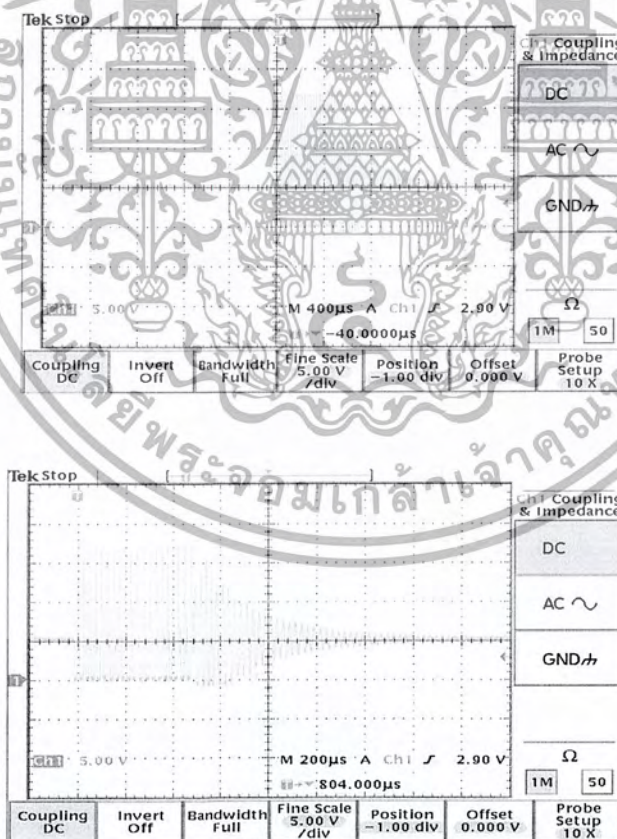
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ทำการวัดสัญญาณหลังจากผ่านวงจรระดับแรงดัน (ในภาคส่ง) ได้ดังรูป



รูปที่ 4.2 รูปสัญญาณที่ผ่านวงจรระดับแรงดัน

ทำการวัดสัญญาณ output ของภาคส่งคลื่นอัลตราโซนิก ได้ดังรูป



รูปที่ 4.3 รูปสัญญาณที่ output ภาคส่ง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

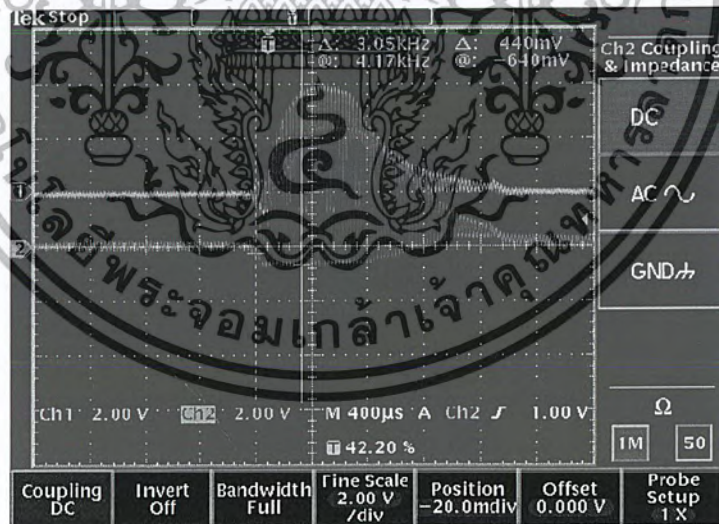
ทดสอบการส่งสัญญาณของภาครับ – ส่ง อัลตราโซนิคที่ระยะต่างๆ

ในการทดลองนี้เราต้องที่จะแสดงให้เห็นถึงลักษณะของสัญญาณที่รับได้จากหัวรับคลื่นอัลตราโซนิค โดยที่เราได้วางไว้ติดกัน 2 หัวรับและทำการทดสอบที่ระยะต่างๆ โดยทำการบันทึกสัญญาณทั้งก่อนเข้าวงจรเปรียบเทียบกับแรงดันและหลังผ่านวงจรเปรียบเทียบกับแรงดัน อีกทั้งทำการทดสอบปรับมุมรับให้ต่างกัน (25 องศา) เพื่อแสดงให้เห็นว่าไม่มีความต่างของเวลาที่หัวรับแต่ละหัวจะรับได้

ผลการทดลอง

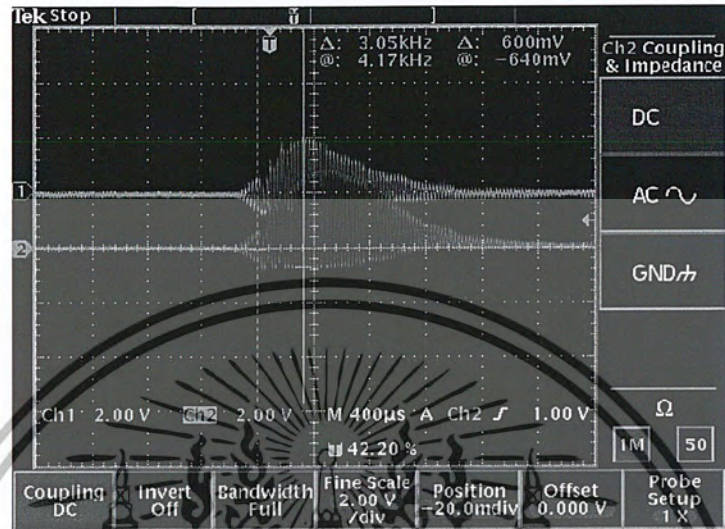
4.1 สัญญาณ output ที่ตัวรับอัลตราโซนิคที่มุม 0 องศา (ขณะยังไม่ผ่านวงจรเปรียบเทียบกับแรงดัน)

4.1.1 ที่ระยะ 50 เซนติเมตร



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.1.2 ที่ระยะ 100 เซนติเมตร



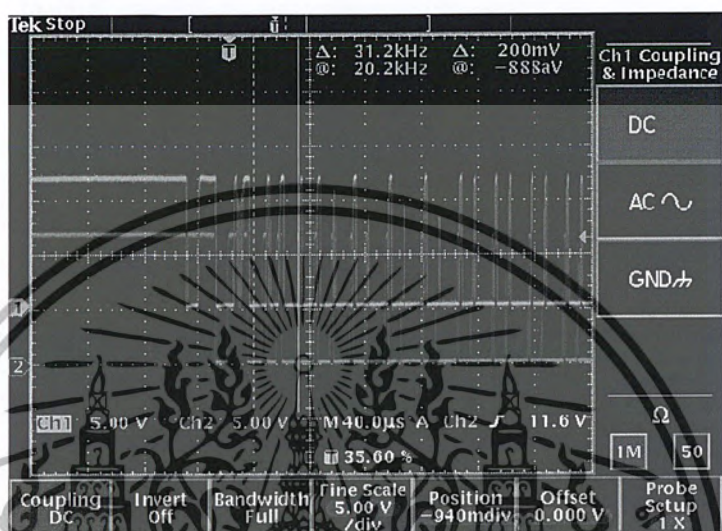
4.1.3 ที่ระยะ 150 เซนติเมตร



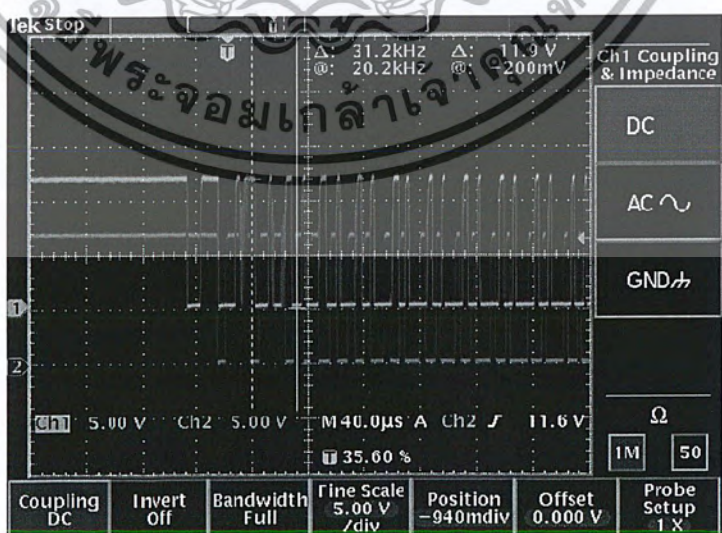
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.2 สัญญาณ output ที่ตัวรับอัลตราโซนิกที่มุม 0 องศา (ขณะผ่านวงจรเปรียบเทียบแรงดัน)

4.2.1 ที่ระยะ 50 เซนติเมตร

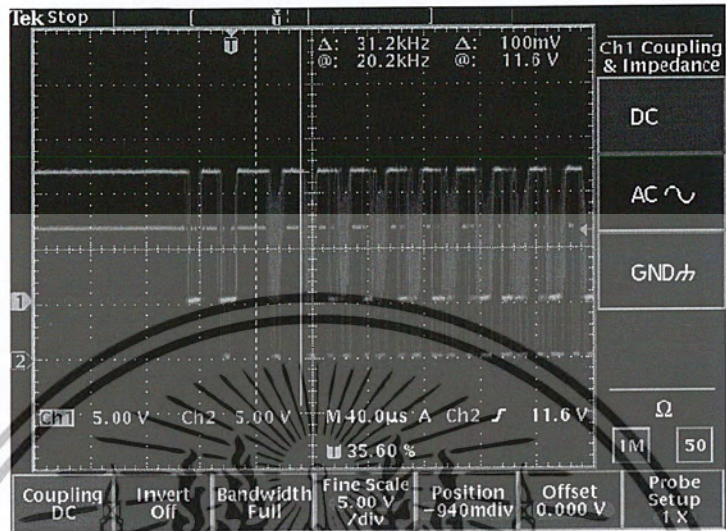


4.2.2 ที่ระยะ 100 เซนติเมตร



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

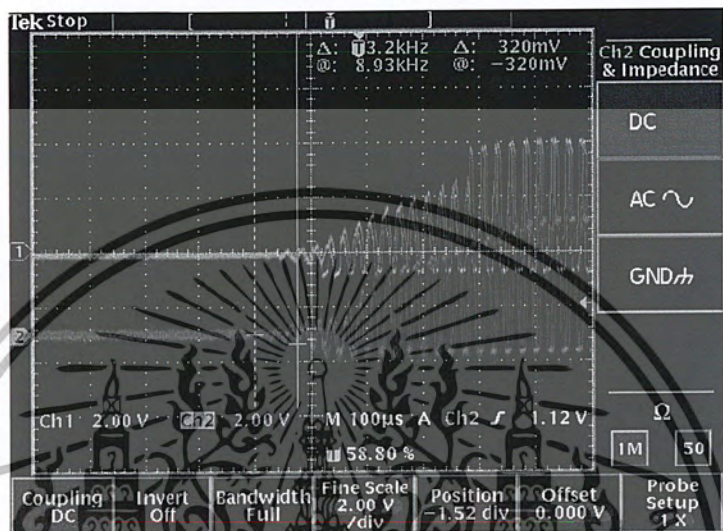
4.2.3 ที่ระยะ 150 เซนติเมตร



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.3 สัญญาณ output ที่ตัวรับอัลตราโซนิกที่มุม 25 องศา (ขณะยังไม่ผ่านวงจรเปรียบเทียบแรงดัน)

4.3.1 ระยะ 50 เซนติเมตร

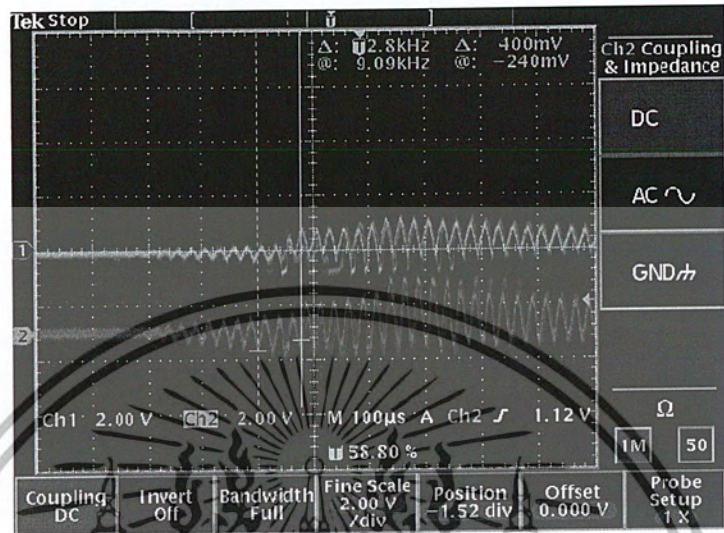


4.3.2 ที่ระยะ 100 เซนติเมตร



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

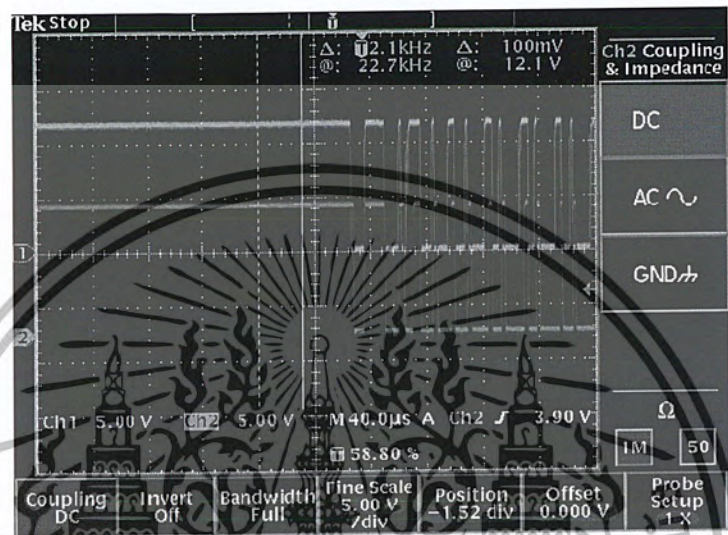
4.3.3 ที่ระยะ 150 เซนติเมตร



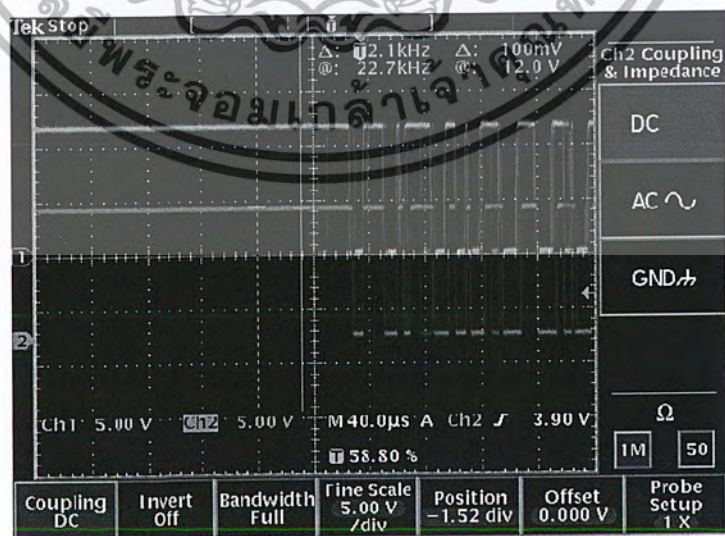
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.4 สัญญาณ output ที่ตัวรับอัลตราโซนิกที่มุม 25 องศา (ขณะผ่านวงจรเปรียบเทียบแรงดัน)

4.4.1 ที่ระยะ 50 เซนติเมตร

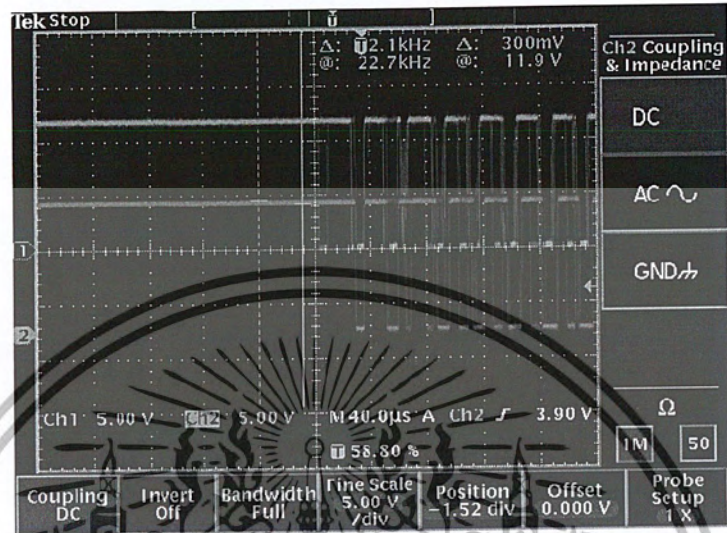


4.4.2 ที่ระยะ 100 เซนติเมตร



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.4.3 ที่ระยะ 150 เซนติเมตร



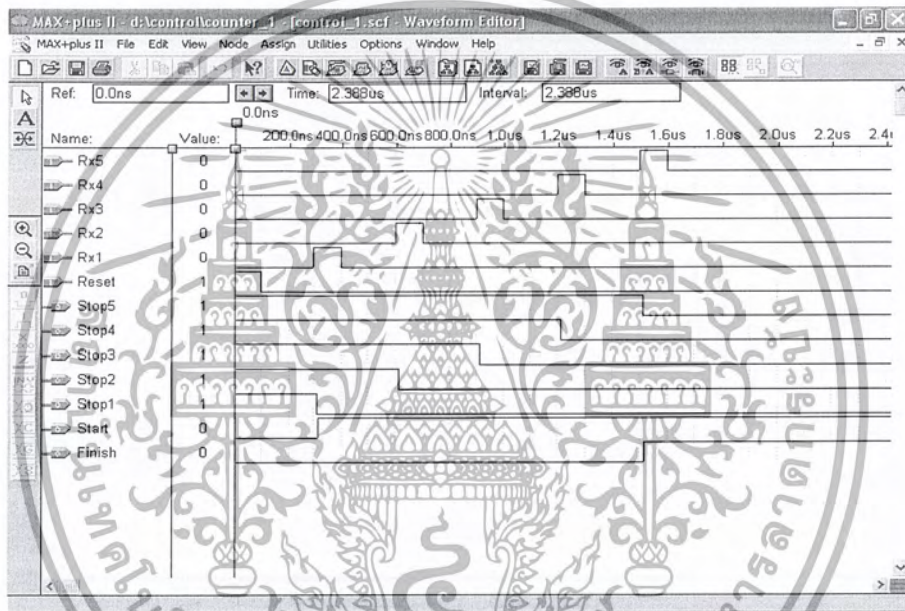
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.5 แสดงผลการ Simulate ของ FPGA

ในหัวข้อนี้ได้แสดงผลการ Simulate วงจรที่ได้ออกแบบไว้ในแต่ละส่วนการทำงาน ซึ่งส่วนหลักๆ ได้แก่ ส่วนของ Block Control และ Block Counter

4.5.1 ผลการ Simulate ส่วนของ Block Control

ในส่วนของ Block Control นี้เป็นส่วนที่เริ่มรับสัญญาณจากอัลตราโซนิกตัวรับ และไปควบคุมการทำงานในส่วน Block Counter ให้เริ่มนับและหยุดนับ

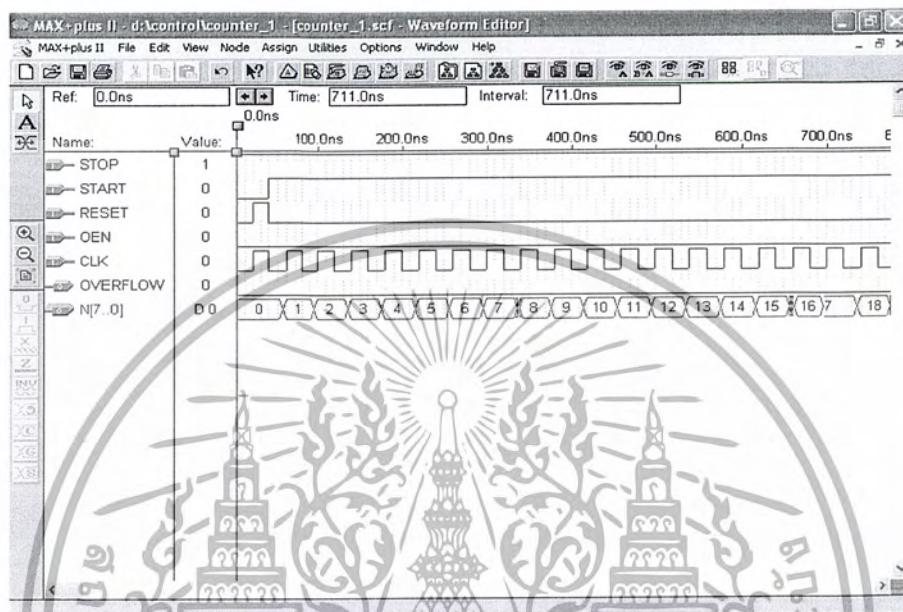


รูปที่ 4.4 แสดง Timing Diagram ของ Block Control ที่ได้จากการ Simulate

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.5.2 ผลการ Simulate ส่วนของ Block Counter

ในส่วนของ Block Counter นี้ได้ทำหน้าที่เป็นตัวจับเวลาของสัญญาณที่ส่งจากตัวส่งจนกระทั่งตัวรับรับสัญญาณได้

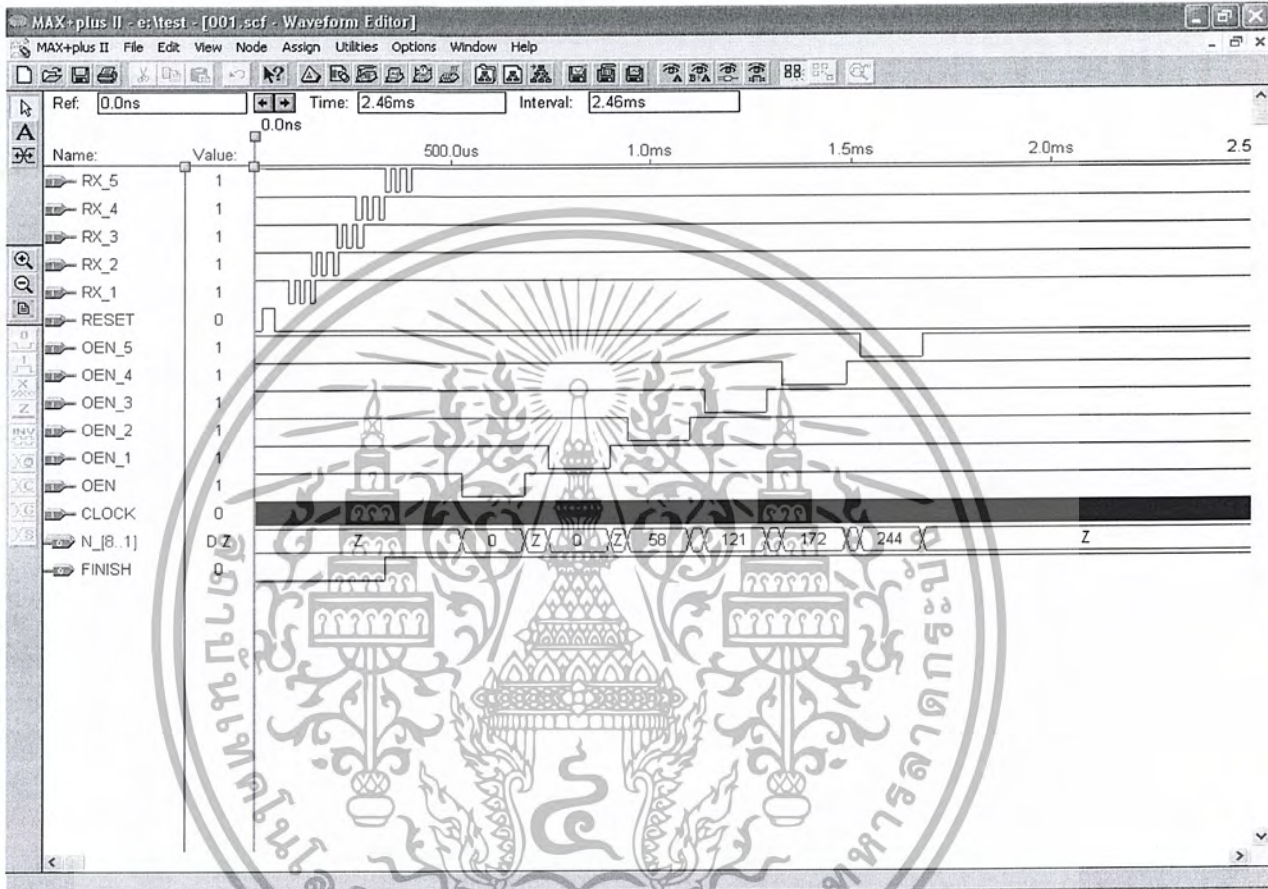


รูปที่ 4.5 แสดง Timing Diagram ของ Block Counter ที่ได้จากการ Simulate

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.5.3 ผลการ Simulate วงจรทั้งหมดของ FPGA

ในหัวข้อนี้ได้แสดงภาพวงจรการทำงานทุกๆ ส่วนของ FPGA ที่ได้นำมาประกอบกันอย่างสมบูรณ์และผลการ Simulate วงจรที่ได้ประกอบแล้วดังนี้

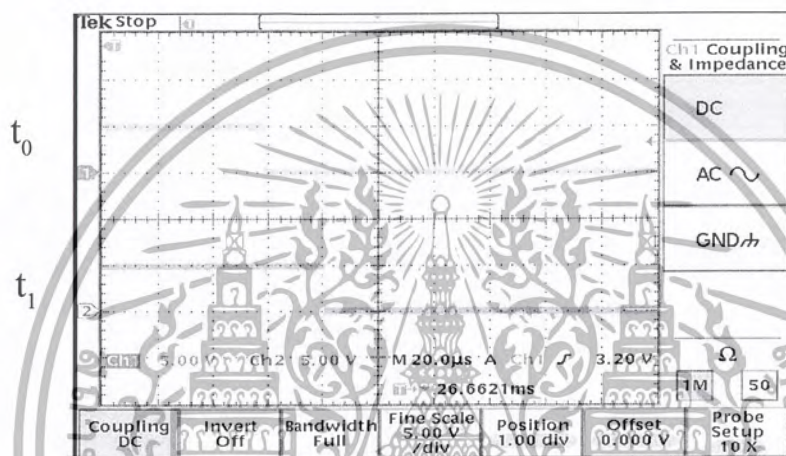


รูปที่ 4.6 แสดง Timing Diagram ของวงจรทั้งหมดของ FPGA

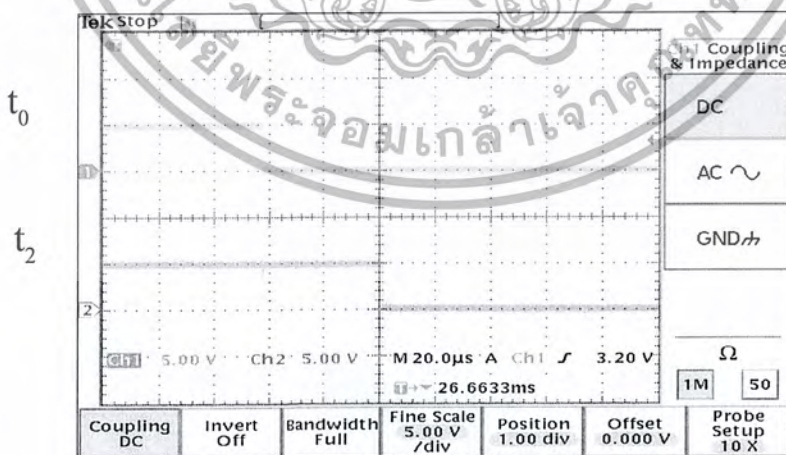
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ทดสอบการเชื่อมต่อระหว่าง FPGA กับ Computer

ในการทดสอบนี้เราทำการทดลองโดยทำการจำลองสัญญาณอินพุตที่ภาครับอัลตราโซนิกรับได้ โดยทำการเขียนโปรแกรมให้ MCS - 51 สร้างพัลส์เรียงแบบจำนวน 5 อินพุตตามจำนวนภาครับทั้งหมด โดยสัญญาณอินพุตแต่ละตัวโดย t_0 ห่างจาก $t_1 = 20 \text{ us}$ t_0 ห่างจาก $t_2 = 40 \text{ us}$ t_0 ห่างจาก $t_3 = 60 \text{ us}$ t_0 ห่างจาก $t_4 = 80 \text{ us}$ ดังรูป

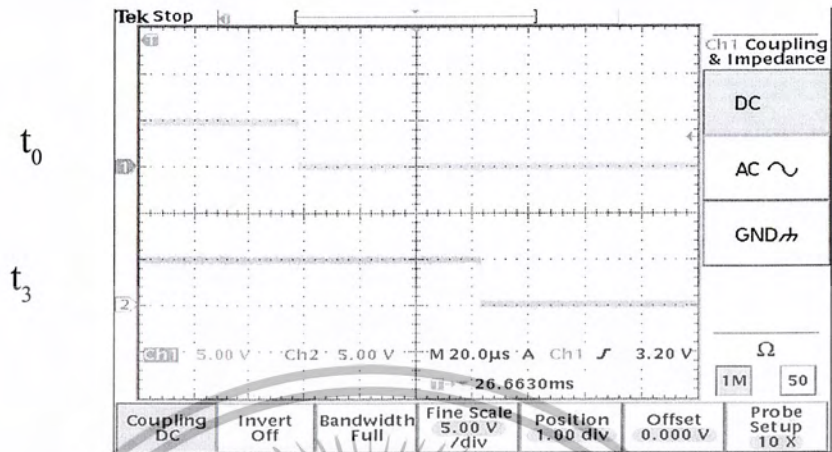


(A) t_0 ห่างจาก $t_1 = 20 \text{ us}$

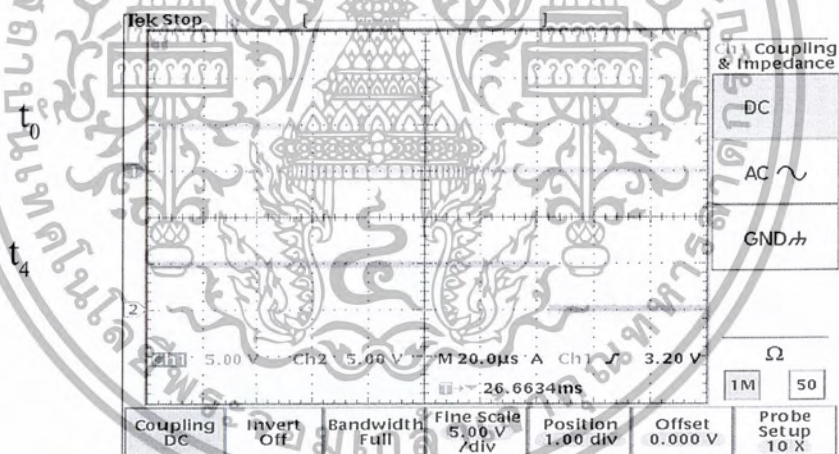


(B) t_0 ห่างจาก $t_2 = 40 \text{ us}$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



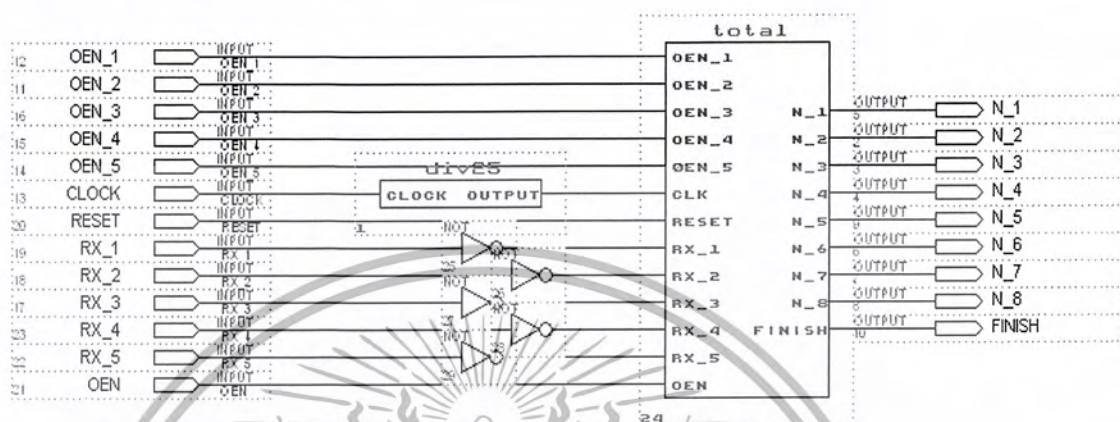
(C) t_0 ห่างจาก $t_3 = 60 \mu$ s



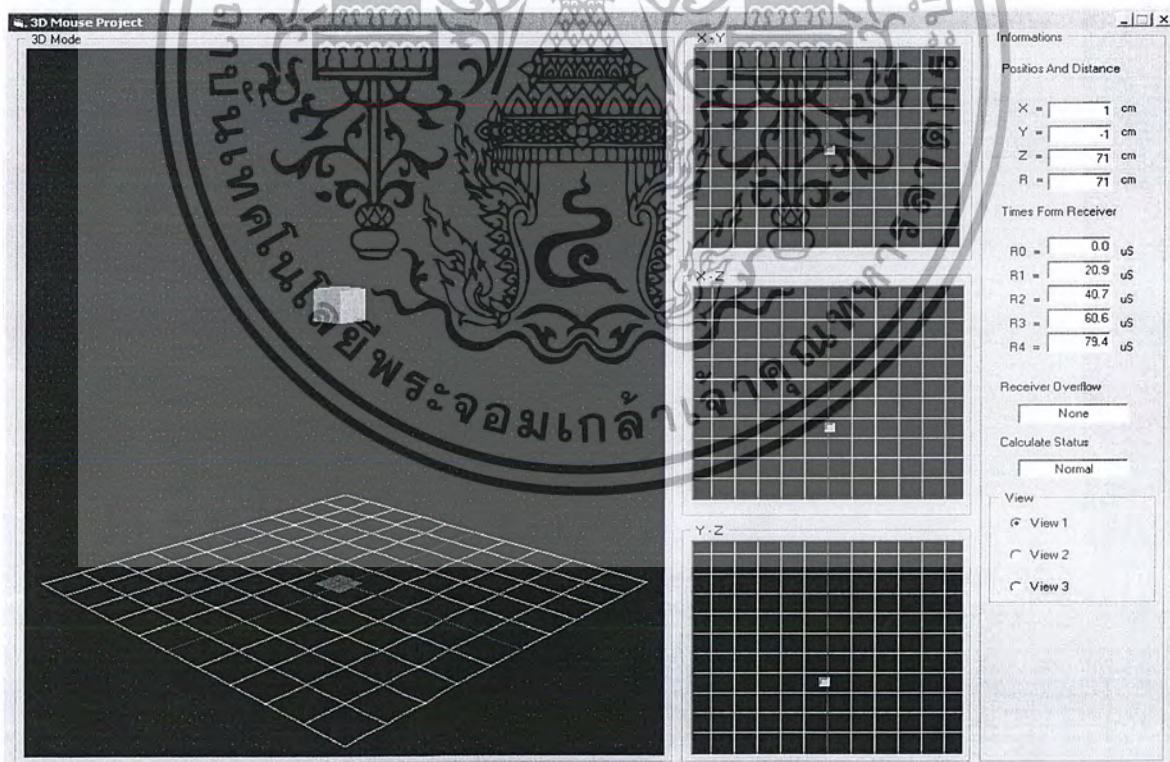
(D) t_0 ห่างจาก $t_4 = 80 \mu$ s

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โดยทำการป้อนเข้า FPGA ซึ่ง FPGA ก็จะทำการนับผลต่างทางเวลาของแต่ละตัวออกมาเป็นเลขฐาน 16 และเมื่อการทดสอบก็สามารถที่จะทำงานได้ โดยสามารถที่แสดงออกมาเป็นรูป 3 มิติโดยวงจร FPGA ที่ใช้นี้ได้แสดงดังนี้



รูปผลที่แสดงพิกัด 3 มิติเมื่อทดลอง ดังนี้



รูปที่ 4.7 แสดงผลการทดสอบ โปรแกรมที่เชื่อมต่อกับFPGA

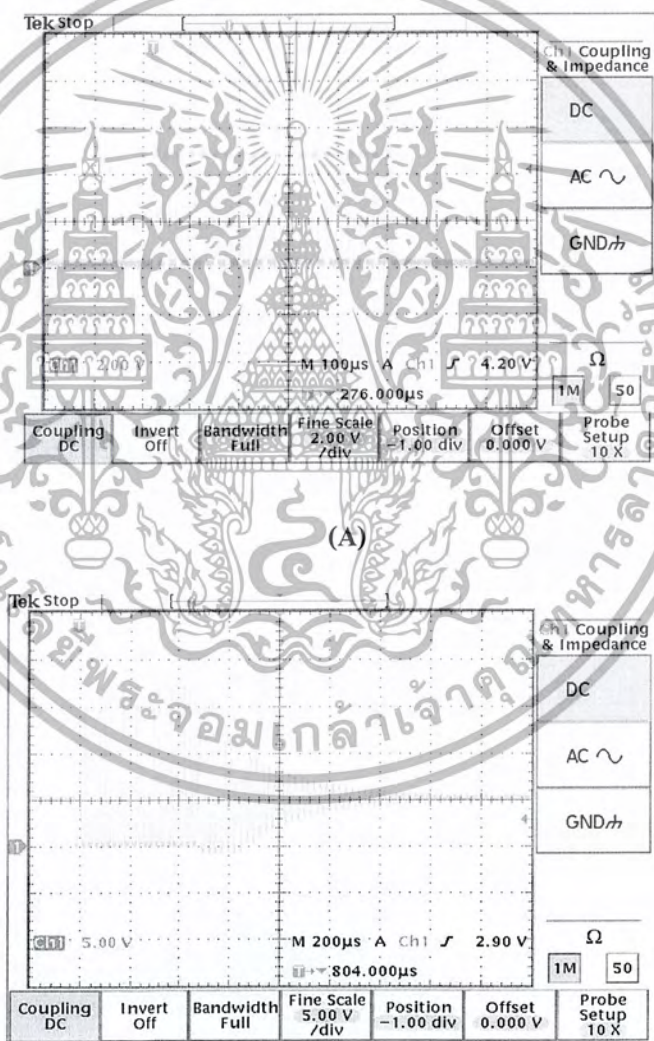
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 5

สรุปผลการทดลอง

การออกแบบเบบเมาส์ 3 มิติในโครงการนี้ สามารถที่จะทำงานได้ในระดับหนึ่งแต่ยังมีปัญหาในส่วนของภาครับ-ภาคส่ง ซึ่งเกิดขึ้นจากคุณสมบัติของหัวอัลตราโซนิก ซึ่งสามารถพิจารณาได้จากรูปคลื่นสัญญาณด้านล่าง

กรณีที่ภาคส่ง

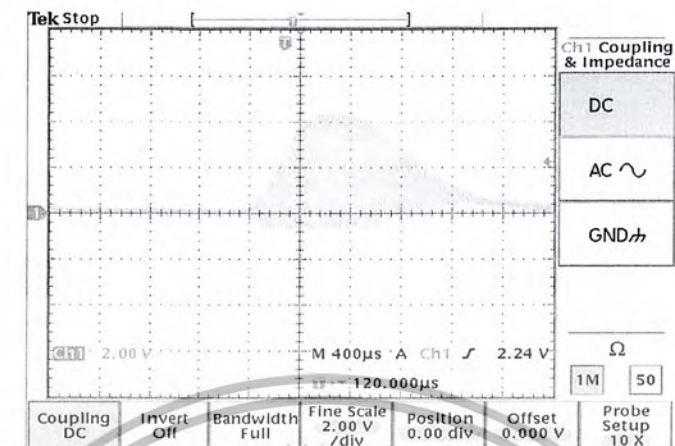


รูปที่ 5.1 รูปสัญญาณที่วัดได้จากภาคส่ง รูป (A) สัญญาณที่ออกจาก MCS-51

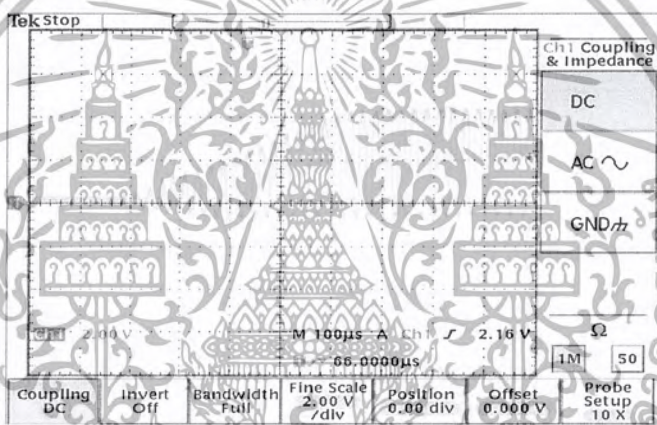
รูป (B) สัญญาณที่ output ของภาคส่ง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

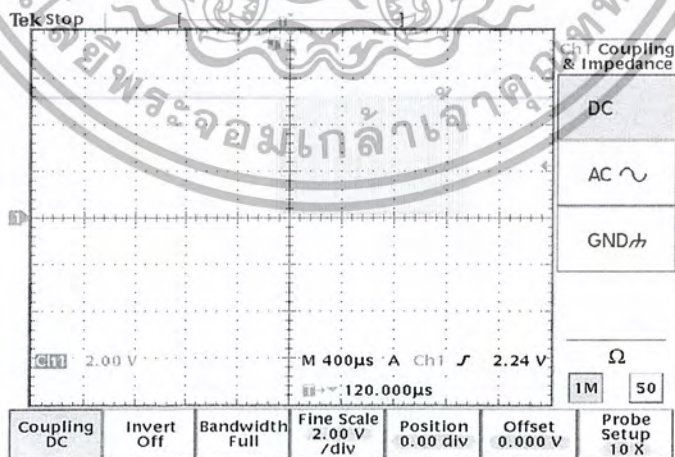
กรณีศึกษาที่ 5.2



(A)



(B)

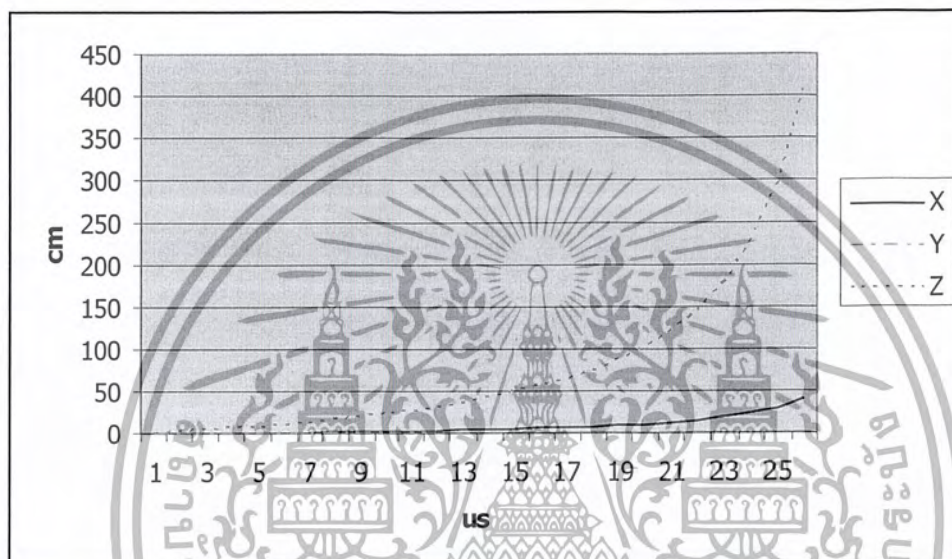


(C)

รูปที่ 5.2 รูปสัญญาณที่วัดได้จากภาครับ รูป (A) และ (B) สัญญาณขณะที่ยังไม่ผ่านวงจรเปรียบเทียบแรงดัน รูป (C) สัญญาณที่ผ่านวงจรเปรียบเทียบแรงดันแล้ว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากคุณสมบัติของหัวอัลตราโซนิกจากที่สามารถดูได้จากรูปด้านบนทำให้พบว่าสัญญาณมีลักษณะเป็นสโโลปทั้งขาขึ้นและขาลง จึงไม่สามารถที่จะกำหนดสัญญาณอ้างอิงได้ ซึ่งส่งผลกระทบต่อภาคถัดไป อันเป็นผลทำให้การคำนวณพิกัด X Y Z ผิดไป ซึ่งจะแสดงให้เห็นว่าถ้าหากเกิดการผิดของค่าเวลาที่รับได้จะส่งผลกระทบต่อพิกัดที่จะคำนวณ ในที่นี้ได้จำลองให้มีการรับค่าเวลาที่ผิดพลาดไป 1 – 25 us (สมมุติว่า T_0 ผิดพลาดไป) โดยแสดงเป็นกราฟได้ดังนี้



รูปที่ 5.3 รูปกราฟแสดงค่าที่คำนวณผิดพลาดไปถ้าเวลาที่รับได้ผิดพลาด

ซึ่งถ้าหากค่าเวลาที่รับได้ผิดพลาดไปทั้ง 5 ตัว คือ t_0, t_1, t_2, t_3, t_4 ก็จะทำให้ค่าที่จะคำนวณได้ผิดพลาดมากยิ่งขึ้น เนื่องจากถ้าผิดพลาดไป 1 รูปคลื่นสัญญาณก็หมายถึงค่าเวลาที่ผิดพลาดไปเท่ากับ 25 us

กิตติกรรมประกาศ

Project นี้จะไม่สามารถสำเร็จออกมาได้ หากไม่ได้รับความร่วมมือกันทำงานของเหล่าผู้ร่วมงานทุกคน ขอขอบคุณเหล่าเพื่อนพ้องที่คอยให้คำแนะนำ ขอขอบคุณอาจารย์เทอดศักดิ์ที่ให้คำแนะนำและปรึกษาในด้านการทำงานทั้งในโปรเจกและสิ่งทีควรทำในอนาคต และที่สำคัญที่สุดคือคุณพ่อ และ คุณแม่ ที่พวกเราเคารพรัก และทำให้มีพวกเราในวันนี้ ถ้ามีอะไรผิดพลาดประการใดทางผู้จัดทำต้องขออภัยมา ณ ที่นี้ด้วย



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก

โปรแกรมกำเนิดคลื่นอัลตราโซนิค

```

ORG          0000H
LOOP1:      MOV          R7,#14H
LOOP2:      SETB        P1.7
            ACALL       DELAY_40K
            CLR         P1.7
            NOP
            ACALL       DELAY_40K
            DJNZ        R7,LOOP2
LOOP3:      MOV          R0,#34H
            MOV          R1,#0FFH
            DJNZ        R1,$
            DJNZ        R0,LOOP3
            SJMP        LOOP1
DELAY_40K:  MOV          R0,#02H
            DJNZ        R0,$
            RET
            END

```

โปรแกรมประมวลผลและสร้างภาพ 3 มิติ

Option Explicit

' ประกาศฟังก์ชัน

Private Declare Function Inp32 Lib "Dllport.dll" (ByVal PortAddress As Integer) As Integer

Private Declare Sub Out32 Lib "Dllport.dll" (ByVal PortAddress As Integer, ByVal Value As Integer)

' ตัวแปรที่ใช้ติดต่อกับ hardware

Public DATA As Integer, Bit0 As Integer, Bit1 As Integer, Bit2 As Integer, Bit3 As Integer, Bit4 As Integer, Bit5 As Integer, Bit6 As Integer, Bit7 As Integer, Carry As Integer

Public Over_Flow As Integer

' ค่าคงที่

Const Clock = 1007000# '1936538.462

Const X0 = 0: Const X1 = 0.1: Const X2 = 0: Const X3 = -0.1: Const X4 = 0

Const Y0 = 0: Const Y1 = 0: Const Y2 = 0.1: Const Y3 = 0: Const Y4 = -0.1

' ตัวแปรภายนอกที่ใช้ในการคำนวณ matrix

Public T0 As Double, T1 As Double, T2 As Double, T3 As Double, T4 As Double

' ตัวแปรภายในที่ใช้ในการคำนวณ matrix

Public Det As Single, Det1 As Single, Det2 As Single, Det3 As Single, Det4 As Single

Public X As Single, Y As Single, Z As Single, V As Single, R As Single

Public RV As Single, V_Sq As Single

Public A1 As Single, A2 As Single, A3 As Single, A4 As Single

Public B1 As Single, B2 As Single, B3 As Single, B4 As Single

Public C1 As Single, C2 As Single, C3 As Single, C4 As Single

Public D1 As Single, D2 As Single, D3 As Single, D4 As Single

Public E1 As Single, E2 As Single, E3 As Single, E4 As Single

' Plot 2d

Private X_Plot As Single, Y_Plot As Single, Z_Plot As Single

' Status

Private Status_R0 As String, Status_R1 As String, Status_R2 As String, Status_R3 As String,

Status_R4 As String

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

'Open Gl declare
Private X_look As Single, Y_look As Single, Z_look As Single
Dim m_fieldOfView As Double
Dim m_NearPlane As Double
Dim m_FarPlane As Double
Dim m_AspectRatio As Double
Dim m_hGLRC&
Dim X_3d As Double, Y_3d As Double, Z_3d As Double
' ฟังก์ชันย่อย hardware Set_Port Input_Data Carry_Data Get_Time Plot_2D
'*****
Public Sub Set_Port()
    Out32 &H378, 127
    Out32 &H379, 120
    Out32 &H37A, 4
End Sub
Public Sub Input_Data()
    If ((Inp32(&H379)) And 16) = 16 Then
        Bit7 = 128
    Else
        Bit7 = 0
    End If
    If ((Inp32(&H379)) And 32) = 32 Then
        Bit6 = 64
    Else
        Bit6 = 0
    End If
    If ((Not (Inp32(&H379))) And 128) = 128 Then
        Bit5 = 32
    Else
        Bit5 = 0
    End If

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

End If
If ((Inp32(&H379)) And 64) = 64 Then
    Bit4 = 16
Else
    Bit4 = 0
End If
If (Not ((Inp32(&H37A))) And 8) = 8 Then
    Bit3 = 8
Else
    Bit3 = 0
End If
If ((Inp32(&H37A)) And 4) = 4 Then
    Bit2 = 4
Else
    Bit2 = 0
End If
If ((Inp32(&H379)) And 8) = 8 Then
    Bit1 = 2
Else
    Bit1 = 0
End If
If (Not ((Inp32(&H37A))) And 2) = 2 Then
    Bit0 = 1
Else
    Bit0 = 0
End If
DATA = Bit0 + Bit1 + Bit2 + Bit3 + Bit4 + Bit5 + Bit6 + Bit7
End Sub
Public Sub Carry_Data()
    If ((Not (Inp32(&H37A))) And 1) = 1 Then

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

    Carry = 1
Else
    Carry = 0
End If
End Sub
Private Sub Get_Time()
' Check overflow
    Over_Flow = 0
    Out32 &H378, 126
    Input_Data
    Over_Flow = DATA
' T0
    T0 = 0
    Out32 &H378, 125
    Input_Data
    T0 = DATA / Clock
' T1
    T1 = 0
    Out32 &H378, 123
    Input_Data
    T1 = DATA / Clock
' T2
    T2 = 0
    Out32 &H378, 119
    Input_Data
    T2 = DATA / Clock
' T3
    T3 = 0
    Out32 &H378, 63
    Input_Data

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

T3 = DATA / Clock

' T4

T4 = 0

Out32 &H378, 95

Input_Data

T4 = DATA / Clock

Out32 &H378, 127

End Sub

Public Sub Calculate_Det()

A1 = 2 * (X0 - X1); A2 = 2 * (X0 - X2); A3 = 2 * (X0 - X3); A4 = 2 * (X0 - X4)

B1 = 2 * (Y0 - Y1); B2 = 2 * (Y0 - Y2); B3 = 2 * (Y0 - Y3); B4 = 2 * (Y0 - Y4)

C1 = 2 * (T0 - T1); C2 = 2 * (T0 - T2); C3 = 2 * (T0 - T3); C4 = 2 * (T0 - T4)

D1 = ((T0 ^ 2) - (T1 ^ 2)); D2 = ((T0 ^ 2) - (T2 ^ 2)); D3 = ((T0 ^ 2) - (T3 ^ 2)); D4 = ((T0 ^ 2) - (T4 ^ 2))

E1 = ((X0 ^ 2) - (X1 ^ 2) + (Y0 ^ 2) - (Y1 ^ 2)); E2 = ((X0 ^ 2) - (X2 ^ 2) + (Y0 ^ 2) - (Y2 ^ 2));

E3 = ((X0 ^ 2) - (X3 ^ 2) + (Y0 ^ 2) - (Y3 ^ 2)); E4 = ((X0 ^ 2) - (X4 ^ 2) + (Y0 ^ 2) - (Y4 ^ 2))

'หาdet

Det1 = A1 * (B2 * C3 * D4 + C2 * D3 * B4 + D2 * B3 * C4 - B4 * C3 * D2 - C4 * D3 * B2 - D4 * B3 * C2)

Det2 = (-1) * A2 * (B1 * C3 * D4 + C1 * D3 * B4 + D1 * B3 * C4 - B4 * C3 * D1 - C4 * D3 * B1 - D4 * B3 * C1)

Det3 = A3 * (B1 * C2 * D4 + C1 * D2 * B4 + D1 * B2 * C4 - B4 * C2 * D1 - C4 * D2 * B1 - D4 * B2 * C1)

Det4 = (-1) * A4 * (B1 * C2 * D3 + C1 * D2 * B3 + D1 * B2 * C3 - B3 * C2 * D1 - C3 * D2 * B1 - D3 * B2 * C1)

Det = Det1 + Det2 + Det3 + Det4

If Det = 0 Then

Text11.Text = "Calculate Error"

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Else

 Text11.Text = "Normal"

End If

End Sub

Private Sub Calculate_Etc()

 'ห1x

 Det1 = E1 * (B2 * C3 * D4 + C2 * D3 * B4 + D2 * B3 * C4 - B4 * C3 * D2 - C4 * D3 * B2 - D4 * B3 * C2)

 Det2 = (-1) * E2 * (B1 * C3 * D4 + C1 * D3 * B4 + D1 * B3 * C4 - B4 * C3 * D1 - C4 * D3 * B1 - D4 * B3 * C1)

 Det3 = E3 * (B1 * C2 * D4 + C1 * D2 * B4 + D1 * B2 * C4 - B4 * C2 * D1 - C4 * D2 * B1 - D4 * B2 * C1)

 Det4 = (-1) * E4 * (B1 * C2 * D3 + C1 * D2 * B3 + D1 * B2 * C3 - B3 * C2 * D1 - C3 * D2 * B1 - D3 * B2 * C1)

 X = (Det1 + Det2 + Det3 + Det4) / Det

 'ห1y

 Det1 = A1 * (E2 * C3 * D4 + C2 * D3 * E4 + D2 * E3 * C4 - E4 * C3 * D2 - C4 * D3 * E2 - D4 * E3 * C2)

 Det2 = (-1) * A2 * (E1 * C3 * D4 + C1 * D3 * E4 + D1 * E3 * C4 - E4 * C3 * D1 - C4 * D3 * E1 - D4 * E3 * C1)

 Det3 = A3 * (E1 * C2 * D4 + C1 * D2 * E4 + D1 * E2 * C4 - E4 * C2 * D1 - C4 * D2 * E1 - D4 * E2 * C1)

 Det4 = (-1) * A4 * (E1 * C2 * D3 + C1 * D2 * E3 + D1 * E2 * C3 - E3 * C2 * D1 - C3 * D2 * E1 - D3 * E2 * C1)

 Y = (Det1 + Det2 + Det3 + Det4) / Det

 'ห1rv

 Det1 = A1 * (B2 * E3 * D4 + E2 * D3 * B4 + D2 * B3 * E4 - B4 * E3 * D2 - E4 * D3 * B2 - D4 * B3 * E2)

 Det2 = (-1) * A2 * (B1 * E3 * D4 + E1 * D3 * B4 + D1 * B3 * E4 - B4 * E3 * D1 - E4 * D3 * B1 - D4 * B3 * E1)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
Det3 = A3 * (B1 * E2 * D4 + E1 * D2 * B4 + D1 * B2 * E4 - B4 * E2 * D1 - E4 * D2 * B1 - D4
* B2 * E1)
```

```
Det4 = (-1) * A4 * (B1 * E2 * D3 + E1 * D2 * B3 + D1 * B2 * E3 - B3 * E2 * D1 - E3 * D2 * B1
- D3 * B2 * E1)
```

```
RV = (Det1 + Det2 + Det3 + Det4) / Det
```

```
V_Sq = 350 ^ 2
```

```
'ห1 v r z
```

```
V = (V_Sq) ^ 0.5
```

```
R = RV / V
```

```
Z = (R ^ 2) - ((X - X0) ^ 2) - ((Y - Y0) ^ 2)
```

```
If Z >= 0 Then
```

```
    Z = Z ^ 0.5
```

```
    Text11.Text = "Normal"
```

```
Else
```

```
    Text11.Text = "Calculate Error"
```

```
End If
```

```
End Sub
```

```
Private Sub Plot_2D()
```

```
    X_Plot = 30 * X * 100 ' *100=cm
```

```
    Y_Plot = 30 * Y * 100 ' *100=cm
```

```
    Z_Plot = 15 * Z * 100 ' *100=cm
```

```
    Picture4.Top = 1440 - Y_Plot
```

```
    Picture4.Left = 1640 + X_Plot
```

```
    Picture5.Top = 3120 - Z_Plot
```

```
    Picture5.Left = 1650 + X_Plot
```

```
    Picture6.Top = 3120 - Z_Plot
```

```
    Picture6.Left = 1650 + Y_Plot
```

```
End Sub
```

```
Private Sub Informations_2D()
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
Text1.Text = Format(T0 * 1000000, "##0.0")
```

```
Text2.Text = Format(T1 * 1000000, "##0.0")
```

```
Text3.Text = Format(T2 * 1000000, "##0.0")
```

```
Text4.Text = Format(T3 * 1000000, "##0.0")
```

```
Text5.Text = Format(T4 * 1000000, "##0.0")
```

```
If Text11.Text = "Normal" Then
```

```
Text6.Text = Format(X * 100, "##0")
```

```
Text7.Text = Format(Y * 100, "##0")
```

```
Text8.Text = Format(Z * 100, "##0")
```

```
Text9.Text = Format(R * 100, "##0")
```

```
End If
```

```
End Sub
```

```
Private Sub Show_Overflow()
```

```
If ((Over_Flow And 1) = 1) Then
```

```
    Status_R0 = "R0 "
```

```
Else
```

```
    Status_R0 = ""
```

```
End If
```

```
If ((Over_Flow And 2) = 2) Then
```

```
    Status_R1 = "R1 "
```

```
Else
```

```
    Status_R1 = ""
```

```
End If
```

```
If ((Over_Flow And 4) = 4) Then
```

```
    Status_R2 = "R2 "
```

```
Else
```

```
    Status_R2 = ""
```

```
End If
```

```
If ((Over_Flow And 8) = 8) Then
```

```
    Status_R3 = "R3 "
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

Else
    Status_R3 = ""
End If
If ((Over_Flow And 16) = 16) Then
    Status_R4 = "R4 "
Else
    Status_R4 = ""
End If

If ((Status_R0 <> "") Or Status_R1 <> "" Or Status_R2 <> "" Or Status_R3 <> "" Or Status_R4
<> "") Then
    Text10.Text = Status_R0 + Status_R1 + Status_R2 + Status_R3 + Status_R4
Else
    Text10.Text = "None"
End If
End Sub
Private Sub Initialize()
    'SET DEAFUE
    *****
    Dim pfd As PIXELFORMATDESCRIPTOR
    Dim R As Long
    pfd.nSize = Len(pfd)
    pfd.nVersion = 1
    pfd.dwFlags = PFD_SUPPORT_OPENGL Or PFD_DRAW_TO_WINDOW Or
PFD_DOUBLEBUFFER Or PFD_TYPE_RGBA
    pfd.iPixelFormat = PFD_TYPE_RGBA
    pfd.cColorBits = 24
    pfd.cDepthBits = 16
    pfd.iLayerType = PFD_MAIN_PLANE
    'check error

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

R = ChoosePixelFormat(animate1.hDC, pfd)
If R = 0 Then
    MsgBox "ChoosePixelFormat failed"
    Exit Sub
End If
R = SetPixelFormat(animate1.hDC, R, pfd)
'palette
m_hGLRC = wglCreateContext(animate1.hDC)
wglMakeCurrent animate1.hDC, m_hGLRC
glClearColor 0, 0, 0, 1
'depht
glClearDepth 1
glEnable GL_DEPTH_TEST
'color
glEnable glColorMaterial
glColorMaterial faceFront, GL_AMBIENT_AND_DIFFUSE
'light
glEnable glColorMaterial
glColorMaterial faceFront, GL_AMBIENT_AND_DIFFUSE
Dim pos!(0 To 3)
glEnable GL_LIGHTING
glEnable glColorLight0
pos(0) = 60: pos(1) = 60: pos(2) = 100: pos(3) = 1
glLightfv lLight0, lpmPosition, pos(0)
pos(0) = 0#: pos(1) = 0.5: pos(2) = 0.5: pos(3) = 1
glLightfv lLight0, lpmAmbient, pos(0)
'viewport
m_AspectRatio = 1
m_FarPlane = 200
m_NearPlane = 0.5

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

m_fieldOfView = 45
'set resize
Dim h As Integer
Dim w As Integer
h = Form1.animate1.ScaleHeight
w = Form1.animate1.ScaleWidth
If h = 0 Then h = 1
m_AspectRatio = w / h
'set viewport
glViewport 0, 0, w, h
glMatrixMode mmmProjection
glLoadIdentity
gluPerspective m_fieldOfView, m_AspectRatio, m_NearPlane, m_FarPlane
glMatrixMode mmmModelView
glLoadIdentity
*****
End Sub
Public Sub Display()
glClear clrColorBufferBit Or clrDepthBufferBit
glPushMatrix
*****
gluLookAt X_look, Y_look, Z_look, 0, 0, 0, 0, 0, 1
'สร้างตาราง 5=10 , coridinate 0,0,0=0,0,-15
glColor3f 0, 1, 0
Dim n As Integer
For n = -25 To 25 Step 5
If n = 0 Then
    glColor3f 1, 0, 0
Else
    glColor3f 0, 1, 0

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

End If
glBegin GL_LINES
    glVertex3d n, 25, -25
    glVertex3d n, -25, -25
glEnd
Next
For n = -25 To 25 Step 5
If n = 0 Then
    glColor3f 1, 0, 0
Else
    glColor3f 0, 1, 0
End If
glBegin GL_LINES
    glVertex3d 25, n, -25
    glVertex3d -25, n, -25
glEnd
Next
'transmister
glColor3d 0.5, 0.5, 1
glBegin GL_QUADS

glNormal3f 0#, 0#, -1#:
glVertex3f -2# + X_3d, -2# + Y_3d, 0# + Z_3d:
glVertex3f 2# + X_3d, -2# + Y_3d, 0# + Z_3d:
glVertex3f 2# + X_3d, 2# + Y_3d, 0# + Z_3d:
glVertex3f -2# + X_3d, 2# + Y_3d, 0# + Z_3d:

glNormal3f 0#, 0#, 1#:
glVertex3f -2# + X_3d, -2# + Y_3d, 4# + Z_3d:
glVertex3f 2# + X_3d, -2# + Y_3d, 4# + Z_3d:

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
glVertex3f 2# + X_3d, 2# + Y_3d, 4# + Z_3d:
glVertex3f -2# + X_3d, 2# + Y_3d, 4# + Z_3d:
```

```
glNormal3f 0#, 1#, 0#:
glVertex3f -2# + X_3d, 2# + Y_3d, 0# + Z_3d:
glVertex3f 2# + X_3d, 2# + Y_3d, 0# + Z_3d:
glVertex3f 2# + X_3d, 2# + Y_3d, 4# + Z_3d:
glVertex3f -2# + X_3d, 2# + Y_3d, 4# + Z_3d:
```

```
glNormal3f 0#, -1#, 0#:
glVertex3f -2# + X_3d, -2# + Y_3d, 0# + Z_3d:
glVertex3f 2# + X_3d, -2# + Y_3d, 0# + Z_3d:
glVertex3f 2# + X_3d, -2# + Y_3d, 4# + Z_3d:
glVertex3f -2# + X_3d, -2# + Y_3d, 4# + Z_3d:
```

```
glNormal3f 1#, 0#, 0#:
glVertex3f 2# + X_3d, -2# + Y_3d, 0# + Z_3d:
glVertex3f 2# + X_3d, 2# + Y_3d, 0# + Z_3d:
glVertex3f 2# + X_3d, 2# + Y_3d, 4# + Z_3d:
glVertex3f 2# + X_3d, -2# + Y_3d, 4# + Z_3d:
```

```
glNormal3f -1#, 0#, 0#:
glVertex3f -2# + X_3d, -2# + Y_3d, 0# + Z_3d:
glVertex3f -2# + X_3d, 2# + Y_3d, 0# + Z_3d:
glVertex3f -2# + X_3d, 2# + Y_3d, 4# + Z_3d:
glVertex3f -2# + X_3d, -2# + Y_3d, 4# + Z_3d:
```

```
glEnd
```

```
'shadow
```

```
glColor3d 0.3, 0.2, 0.4
```

```
glBegin GL_QUADS
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

glNormal3f 0#, 0#, 1#:
glVertex3f -2# + X_3d, -2# + Y_3d, -25:
glVertex3f 2# + X_3d, -2# + Y_3d, -25:
glVertex3f 2# + X_3d, 2# + Y_3d, -25:
glVertex3f -2# + X_3d, 2# + Y_3d, -25:

```

```
glEnd
```

```
!*****
```

```
****
```

```
glPopMatrix
```

```
glFinish
```

```
SwapBuffers animate1.hDC 'display to the screen
```

```
End Sub
```

```
Private Sub Form_Load()
```

```
Call Initialize
```

```
X_look = 75
```

```
Y_look = 75
```

```
Z_look = 25
```

```
Call Set_Port
```

```
Out32 &H378, 255
```

```
Out32 &H378, 127
```

```
End Sub
```

```
Private Sub Option1_Click()
```

```
X_look = 75
```

```
Y_look = 75
```

```
Z_look = 25
```

```
End Sub
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
Private Sub Option2_Click()
```

```
    X_look = 75
```

```
    Y_look = -75
```

```
    Z_look = 25
```

```
End Sub
```

```
Private Sub Option3_Click()
```

```
    Y_look = -0.001
```

```
    X_look = 0.0000000001
```

```
    Z_look = 90
```

```
End Sub
```

```
Private Sub Timer3_Timer()
```

```
    X_3d = Text6.Text
```

```
    Y_3d = Text7.Text
```

```
    Z_3d = Text8.Text
```

```
    X_3d = X_3d / 2
```

```
    Y_3d = Y_3d / 2
```

```
    Z_3d = (Z_3d - 50) / 2
```

```
    DoEvents
```

```
    Display
```

```
End Sub
```

```
Private Sub Form_Unload(Cancel As Integer)
```

```
    If m_hGLRC <> 0 Then
```

```
        wglMakeCurrent 0, 0
```

```
        wglDeleteContext m_hGLRC
```

```
    End If
```

```
End Sub
```

```
Private Sub Timer1_Timer()
```

```
    Call Set_Port
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

Call Carry_Data
If Carry = 1 Then
    Call Get_Time
    Call Show_Overflow
    Call Calculate_Det
    If Det <> 0 Then
        Call Calculate_Etc
    End If
*****
Call Informations_2D
If Text11.Text = "Normal" Then
    Call Plot_2D
End If
Timer1.Enabled = False
Timer2.Enabled = True
End If
End Sub
Private Sub Timer2_Timer()
    Out32 &H378, 255
    Out32 &H378, 127
    Timer2.Enabled = False
    Timer1.Enabled = True
End Sub

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้