สำนักหอสมุดกลาง พระจอมเกล้าลาดกระบัง

เครื่องต้นแบบสำหรับแสดงผลพิกัดทางภูมิศาสตร์

The Prototype of Monitoring Geography Position

T055470

โดย

นาย จักรพงษ์ โรจนแสง

นางสาว ชนิดา ทิพย์วงศ์

นาย อรรถพร ริยาพันธ์

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต

สาขาวิชาอิเล็กทรอนิกส์

คณะวิศวกรรมศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2546

เลขหมู่
เลขทะเบียน 55470
วัน,เดือน,ปี 9 พ.ค. 2548

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เครื่องต้นแบบสำหรับแสดงผลพิกัดทางภูมิศาสตร์

The Prototype of Monitoring Geography Position

โดย

นาย จักรพงษ์ โรจนแสง รหัสประจำตัว 43010053

นางสาว ชนิดา ทิพย์วงศ์ รหัสประจำตัว 43010083

นาย อรรถพร ริยาพันธ์ รหัสประจำตัว 43010531

อาจารย์ที่ปรึกษา

ศ.ดร.วัลลภ สุระกำพลธร

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต

สาขาวิชาอิเล็กทรอนิกส์

คณะวิศวกรรมศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2546

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ปริญญานิพนธ์ปีการศึกษา 2546

ภาควิชาอิเล็กทรอนิกส์

คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เรื่อง เครื่องต้นแบบสำหรับแสดงผลพิกัดทางภูมิศาสตร์

จัดทำโดย

1. นาย จักรพงษ์ โรจนแสง
2. นางสาว ชนิดา ทิพย์วงศ์
3. นาย อรรถพร ริยาพันธ์

โครงงานนี้ได้รับการตรวจสอบแล้ว พร้อมที่จะทำการสอบได้

.....

(ศ.ดร. วัลลภ สุระกำพลธร)

อาจารย์ที่ปรึกษา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

แบบฟอร์มรับรองความพร้อมในการสอบ

ปริญญานิพนธ์เรื่อง เครื่องต้นแบบสำหรับแสดงผลพิกัดทางภูมิศาสตร์

The Prototype of Monitoring Geography Position

จัดทำโดย

1. นาย จักรพงษ์ โรจนแสง
2. นางสาว ชนิดา ทิพย์วงศ์
3. นาย อรรถพร ริยาพันธ์

โครงงานนี้ได้รับการตรวจสอบแล้ว พร้อมที่จะทำการสอบได้

.....

(ศ.ดร.วัลลภ สุระกำพลธร)

อาจารย์ที่ปรึกษา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทคัดย่อ

ปริญญานิพนธ์ฉบับนี้เป็นการศึกษาหลักการเบื้องต้นของระบบ GPS (Global Positioning System) จากทฤษฎีดังกล่าว จึงได้นำเครื่องรับสัญญาณ GPS มาประยุกต์ใช้กับ FPGA (Field Programmable Gate Array) เพื่อให้ FPGA ทำหน้าที่ประมวลผลสัญญาณที่มาจากเครื่องรับสัญญาณ GPS ให้มาแสดงค่า Latitude, Longitude และ เวลา บนจอ LCD พร้อมทั้งสามารถบอกว่าค่าที่แสดงบนจอ LCD เป็นสถานที่ใด ให้ออกมาเป็นเสียงพูดได้

ABSTRACT

This thesis to study the basic principle of GPS (Global Positioning System). From this , we use GPS module to apply with FPGA (Field Programmable Gate Array) for processing signal. Latitude , longitude and time will be finally shown on the LCD and you will also hear the name of place by the speaker.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กิตติกรรมประกาศ

ปริญญานิพนธ์ฉบับนี้สำเร็จลุล่วงได้ด้วยดี ผู้เขียนขอบกราบขอบพระคุณ ศาสตราจารย์ ดร.วัลลภ สุระกำพลธร อาจารย์ที่ปรึกษาเป็นอย่างสูง ที่ได้ให้ความคิดริเริ่มสร้างสรรค์ และคำปรึกษาในการทำปริญญานิพนธ์ฉบับนี้

ขอขอบพระคุณ พี่อดุลย์ ขันติชนะกุล พี่เฉลิมภันท์ ฟองสมุทร และพี่มนูญ สันถวะคุปต์ที่ได้ให้คำปรึกษา วิธีการแก้ไขปัญหาต่าง ๆ ที่เกิดขึ้นในการทำปริญญานิพนธ์ รวมทั้งทีมงานของห้องปฏิบัติการ Mixed Signal Processing ทุกท่านที่ได้เอื้อเฟื้ออุปกรณ์เครื่องมือและสถานที่ในการทำงาน

ขอกราบขอบพระคุณ คุณพ่อ คุณแม่ ที่คอยห่วงใยและให้การสนับสนุนในการศึกษา รวมทั้งขอขอบคุณเพื่อน ๆ พี่ ๆ ที่คอยให้คำปรึกษา กำลังใจ และความช่วยเหลือในด้านต่าง ๆ แก่ผู้เขียนโดยตลอด

จักรพงษ์ โรจนแสง
ชนิดา ทิพย์วงศ์
อรรถพร ริยาพันธ์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ

หน้า

บทคัดย่อ..... I

กิตติกรรมประกาศ..... II

สารบัญ..... III

สารบัญรูปภาพ..... IV

บทที่ 1 บทนำ..... 1

1.1 วัตถุประสงค์ของปริญญานิพนธ์..... 2

1.2 รายละเอียดของปริญญานิพนธ์..... 2

บทที่ 2 ทฤษฎีเบื้องต้นของ GPS..... 4

2.1 Global Positioning Systems : GPS..... 4

2.2 องค์ประกอบของระบบดาวเทียม GPS..... 4

2.3 การนำ GPS มาประยุกต์ใช้งานในด้านต่างๆ..... 7

2.4 หลักการพื้นฐานของ GPS..... 8

บทที่ 3 NMEA 0183 Protocal..... 10

3.1 ลักษณะสัญญาณทางไฟฟ้าของ NMEA 0183 ที่ใช้ในการติดต่อสื่อสาร..... 10

3.2 ลักษณะโดยทั่วไปของรูปแบบข้อมูลของ NMEA 0183..... 11

3.3 ลักษณะของประโยค และรูปแบบต่าง ๆ..... 12

บทที่ 4 การออกแบบและการสร้างวงจร..... 16

4.1 การออกแบบวงจรรวม..... 16

4.2 วงจรกู้สัญญาณนาฬิกา..... 17

4.3 วงจรส่วนกรองข้อมูล..... 18

4.4 วงจรส่วนแสดงผล (LCD INTERFACE)..... 22

4.5 วงจรส่วนแสดงผลออกมาเป็นเสียงพูด (Voice Recorder IC Interface)..... 26

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ (ต่อ)

บทที่ 5 การทดลอง....28
5.1 การทดสอบโดยการจำลองการทำงาน (Simulation)....28
5.2 การทำ implementation บน FPGA รุ่น spartan2 เบอร์ xc2s50-6tq144....31
5.3 การทดสอบการทำงานจริงของวงจร....33

บทที่ 6 สรุปและวิจารณ์ผลการทดลอง....36
6.1 ปัญหาที่พบในการทดลอง....36
6.2 แนวทางในการพัฒนาต่อไป....37

บรรณานุกรม....38
ภาคผนวก
ภาคผนวก ก ภาคแสดงผล (LCD Interface)
ภาคผนวก ข โปรแกรมภาษา VHDL
ภาคผนวก ค Data sheets

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูป

หน้า

รูปที่ 2.1 แสดงส่วนประกอบของระบบดาวเทียม GPS....4
รูปที่ 2.2 แสดงตำแหน่งและการโคจรของดาวเทียม GPS รอบโลก....5
รูปที่ 2.3 แสดงสถานีควบคุมระบบดาวเทียม GPS 5 แห่ง....6
รูปที่ 2.4 เครื่องรับสัญญาณ GPS (GPS Receivers)....7
รูปที่ 2.5 แสดงการใช้งานทางบก ทางน้ำ และทางอากาศ....7
รูปที่ 2.6 แสดงการหาพิกัดตำแหน่ง....8
รูปที่ 3.1 แสดงตัวอย่างสัญญาณที่ส่งมาจากเครื่องรับสัญญาณ GPS....10
รูปที่ 4.1 แสดงการทำงานของอุปกรณ์ที่ออกแบบไว้....16
รูปที่ 4.2 หลักการของวงจรกู้สัญญาณนาฬิกา....17
รูปที่ 4.3 วงจรกู้สัญญาณนาฬิกา....18
รูปที่ 4.4 block diagram แสดงการรับและส่งข้อมูลระหว่างหน่วยอื่นๆ....19
รูปที่ 4.5 block diagram แสดงขั้นตอนการทำงาน....19
รูปที่ 4.6 แสดงโครงสร้างของการเก็บข้อมูลภายใน RAM....20
รูปที่ 4.7 แสดงการทำงานของกระบวนการจัดการข้อมูล....21
รูปที่ 4.8 แสดง ค่าต่างใน RAM2 หลังจากที่เสร็จสิ้นกระบวนการ....22
รูปที่ 4.9 สัญญาณควบคุม LCD Module ผ่านทาง 3 Bit Serial....23
รูปที่ 4.10 บล็อกไดอะแกรมภาค LCD Graphic Interface....24
รูปที่ 4.11 แสดงการนำข้อมูลมาแสดงผลทางหน้าจอ LCD....26
รูปที่ 4.12 แสดงขั้นตอนการทำงาน ในส่วนแสดงเสียง....26
รูปที่ 4.13 แสดงขั้นตอนการเปรียบเทียบค่าพิกัดเพื่อแสดงเสียงพูด....27
รูปที่ 5.1 แบบจำลองการ detect หา $GPRMC....29
รูปที่ 5.2 แบบจำลองการทำงานการระบุ address และส่งข้อมูลไปยัง RAM....29
รูปที่ 5.3 แบบจำลองการทำงานภาคเชื่อมต่อวงจรแสดงผล....30
รูปที่ 5.4 แบบจำลองการทำงานของส่วนควบคุม IC ISD 2590....30
รูปที่ 5.5 โครงสร้างการเชื่อมต่อภายในระหว่าง CLB....32
รูปที่ 5.6 แสดงผลเปรียบเทียบระหว่างสัญญาณที่เข้าเฟสดีเทคเตอร์กับสัญญาณ Clock recover...33
รูปที่ 5.7 แสดงสัญญาณเปรียบเทียบระหว่าง Data กับ Clock recovery....34
รูปที่ 5.8 แสดงตัวอย่างการแสดงผลทาง LCD Display ขณะที่เครื่องรับสัญญาณ GPS อยู่
ณ ตำแหน่ง (ก) อับสัญญาณ (ข) ตึก B (ค) ตึก A....35

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 1
บทนำ

ตั้งแต่อดีต มนุษย์มีการหาสิ่งเพื่อช่วยในการบอกทิศทางสำหรับการเดินทางไปยังที่ต่างๆ โดยมีวิวัฒนาการมาตั้งแต่ วิธีสังเกตจากดวงดาว การใช้เข็มทิศ จนมาถึงในยุคอิเล็กทรอนิกส์ มนุษย์ก็ได้คิดค้นเครื่องมือที่มีความทันสมัยและแม่นยำมากขึ้น เริ่มจากการใช้คลื่นวิทยุในระบบ LORAN ซึ่งมักใช้ในการเดินเรือ และอีกระบบต่อมา เป็นระบบที่ใช้ดาวเทียม "TRANSIT SYSTEM" หรือ "SATNAV" โดยทั้งสองระบบนี้ ปัจจุบัน ได้ถูกยกเลิกการใช้งานแล้ว เนื่องจากมีข้อบกพร่องในการบอกตำแหน่ง จนในที่สุด ก็ได้มาถึงการใช้ Global Positioning System (GPS) ซึ่งเป็นระบบที่ทำการระบุตำแหน่งพิกัด โดยใช้สัญญาณจากดาวเทียมที่ทำหน้าที่ส่งสัญญาณจีพีเอสโดยเฉพาะ การที่จะรับข้อมูลจากดาวเทียมได้จะต้องผ่านเครื่องรับสัญญาณ GPS ผ่านระบบการเชื่อมต่อ (Interface) เข้ากับคอมพิวเตอร์ หรือ ไมโครคอนโทรลเลอร์ เพื่อให้หน่วยประมวลผลเหล่านี้ทำการประมวลผลข้อมูลที่ได้รับมา เป็นข้อมูลพิกัด ละติจูด ลองจิจูด โดยระบบนี้สามารถลดข้อจำกัดของระบบเดิมได้ดี มีความแม่นยำสูง อีกทั้งจากการพัฒนาทางด้านอุปกรณ์คอมพิวเตอร์ทำให้สามารถผลิตเครื่องรับ GPS ที่มีขนาดลดลง และมีราคาถูกลงกว่าเครื่องรับระบบ TRANSIT เดิมเป็นอันมาก

GPS ได้ถูกนำมาใช้งานในหลายสาขาวิชาที่เกี่ยวข้องกับงานสำรวจ อาทิเช่น ภูมิศาสตร์ วิศวกรรมศาสตร์ สิ่งแวดล้อม การนำ GPS มาใช้ในการกำหนดขอบเขตและจุดที่แน่นอนของป่าสงวน และอุทยาน การใช้ GPS ในการสำรวจภูมิประเทศเพื่อทำแผนที่เส้นชั้นความสูง (Contour) หรือแม้แต่งานทำถนน เป็นต้น จากการใช้งานที่หลากหลายนี้เอง ผู้ผลิตจึงต้องพยายามหาเทคโนโลยีที่เหมาะสมมาทำการออกแบบและผลิตเครื่องรับGPS ที่มีขนาดเล็กหรือพยายามย่อให้ลงไปอยู่ในวงจรรวมหรือชิปเดียว เพื่อให้สะดวกต่อผู้ใช้งาน อีกทั้งเครื่องรับ GPS มักจะถูกเชื่อมต่อกับคอมพิวเตอร์เพื่อใช้ร่วมกับโปรแกรมสำเร็จรูป ดังนั้น เทคโนโลยีของการผลิตอุปกรณ์เพื่อใช้กับระบบ GPS จึงมักออกมาในรูปแบบของชิป โดยใช้การออกแบบวงจรดิจิตอลมาสร้างและพัฒนาให้ชิปมีการทำงานที่มีประสิทธิภาพมากขึ้น

ในปัจจุบัน ได้มีอุปกรณ์ที่รับและแสดงผลพิกัดโดยใช้ระบบ GPS ออกมาสู่ท้องตลาดเป็นจำนวนมาก อีกทั้งถูกออกแบบให้มีขนาดเล็ก พกพาได้สะดวก ซึ่งทำให้ผู้ใช้ สามารถทราบพิกัดและข้อมูลอื่นที่ต้องการผ่านทางหน้าจอของอุปกรณ์ได้โดยง่าย แต่อุปกรณ์ดังกล่าวมักมีราคาค่อนข้างแพงเนื่องจากถูกผลิตขึ้นในต่างประเทศ หรือสำหรับในเครื่องรับบางประเภท จะไม่มีหน้าจอแสดงผลในตัวเอง แต่จะใช้ซอฟแวร์มาประมวลผลแล้วแสดงที่หน้าจอคอมพิวเตอร์แทน ซึ่งจะไม่สะดวกในการใช้งาน เนื่องจากยากต่อการพกพา ดังนั้นในโครงงานนี้จึงเลือกใช้เครื่องรับ GPS ที่ไม่มีหน้าจอแสดงผลในตัว แต่จะใช้จอ LCD มาต่อเพิ่มเข้าไปเพื่อใช้ในการแสดงผล เฉพาะข้อมูลที่จำเป็น โดยจะใช้ภาษา VHDL มาออกแบบฮาร์ดแวร์แล้วโปรแกรมลงในชิป FPGA (Field

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Programmable Gate Array) เพื่อทำหน้าที่เป็นหน่วยประมวลผล เนื่องจากมีข้อดีที่ สามารถแก้ไขการออกแบบและพัฒนาวงจรได้ง่าย สามารถนำไปทำเป็นชิปเดียวที่มีขนาดเล็ก ช่วยป้องกันการถูกลอกเลียนแบบวงจร อีกทั้งยังมีประสิทธิภาพและความน่าเชื่อถือสูงอีกด้วย โดยสร้างเป็นวงจรต้นแบบเพื่อนำไปพัฒนาต่อในการผลิตเครื่องรับและแสดงผลพิกัดจาก GPS ที่มีความสามารถและประสิทธิภาพที่สูงขึ้นต่อไป

1.1 วัตถุประสงค์ในการทำปริญญานิพนธ์

1. ศึกษาการทำงานของระบบ GPS และเครื่องรับสัญญาณ GPS

2. สร้างอุปกรณ์ที่สามารถรับสัญญาณจากเครื่องรับสัญญาณ GPS ได้โดยสมารถแสดงพิกัดตำแหน่งที่เครื่องนี้ได้

1.2 รายละเอียดของปริญญานิพนธ์

ปริญญานิพนธ์ฉบับนี้ ได้แบ่งเนื้อหาออกเป็น 6 บท และภาคผนวกอีก 1 ภาค โดยในแต่ละบทมีรายละเอียดดังต่อไปนี้

บทที่ 1 เป็นบทนำ ที่จะกล่าวนำ ถึงความเป็นมา วัตถุประสงค์ ของปริญญานิพนธ์ฉบับนี้ พร้อมทั้งรายละเอียดของปริญญานิพนธ์ในแต่ละบท

บทที่ 2 กล่าวถึงทฤษฎีเบื้องต้นของระบบ GPS ว่ามีหลักการทำงานอย่างไร และการประยุกต์ใช้งานเครื่องรับสัญญาณ GPS

บทที่ 3 กล่าวถึง รูปแบบ และลักษณะของโปรโตคอลมาตรฐานที่ใช้ในระบบ GPS

บทที่ 4 กล่าวถึงหลักการออกแบบวงจรที่จะใช้ติดต่อกับเครื่องรับสัญญาณ GPS โดยใช้ FPGA เป็นอุปกรณ์ที่ควบคุมการติดต่อ

บทที่ 5 กล่าวถึงการทดลอง ผลการทดลอง และสรุปผลการทดลอง ของวงจรที่ออกแบบไว้ในบทที่ 4

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 6 เป็นบทส่งท้าย ซึ่งเป็นการสรุปผลงานที่ได้ออกแบบไว้ในปริญญานิพนธ์ฉบับนี้ พร้อมทั้งได้นำเสนอแนวทางในการพัฒนาต่อไป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 2
ทฤษฎีเบื้องต้นของ GPS

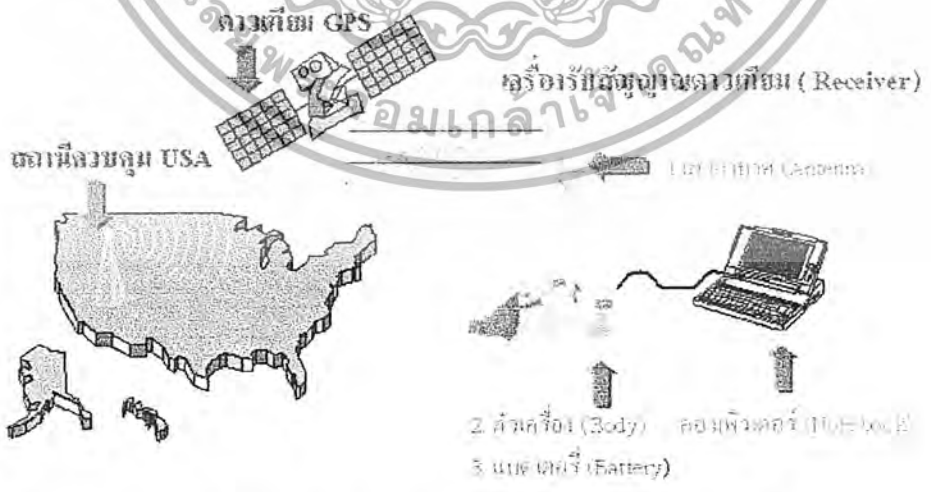
2.1 Global Positioning Systems : GPS

ระบบการกำหนดตำแหน่งบนพื้นพิภพ หรือ จีพีเอส (Global Positioning System : GPS) เป็นระบบที่มีการคิดค้นและพัฒนาขึ้นโดยกระทรวงกลาโหมของประเทศสหรัฐอเมริกา ซึ่งแรกเริ่มมีจุดประสงค์เพื่อใช้ในด้านการทหาร แต่ในปัจจุบันได้มีการอนุญาตให้มีส่วนที่พลเรือนสามารถนำไปใช้งานได้ ระบบนี้จะหน้าที่บอกตำแหน่งบนพื้นโลกโดยการอ้างอิงจากดาวเทียม 24 ดวงที่โคจรอยู่รอบโลก ในระดับความสูงที่พ้นจากคลื่นวิทยุรบกวนของโลก ซึ่งวิธีการนี้จะสามารถให้ความถูกต้องและแม่นยำเพียงพอที่จะใช้ชี้บอกตำแหน่งได้ทุกแห่งบนโลกตลอดเวลา 24 ชั่วโมง

2.2 องค์ประกอบของระบบดาวเทียม GPS

องค์ประกอบหลักของระบบจีพีเอส ประกอบด้วย 3 ส่วน คือ

1. ส่วนอวกาศ
2. ส่วนควบคุม
3. ส่วนผู้ใช้งาน



รูปที่ 2.1 แสดงส่วนประกอบของระบบดาวเทียม GPS

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.2.1 ส่วนอวกาศ (Space Segment)

ในระบบดาวเทียม GPS จะประกอบด้วยดาวเทียมทั้งหมด 24 ดวง โดยดาวเทียมจำนวน 21 ดวง จะใช้ในการบอกค่าพิกัด ส่วนที่เหลือ 3 ดวง จะสำรองเอาไว้ ดาวเทียมทั้ง 24 ดวงนี้จะมีวงโคจรอยู่ที่รัศมีวงโคจรสูงจากพื้นโลก 20,162.61 กิโลเมตร (12,600 ไมล์) โดยแบ่งจำนวนดาวเทียมเป็น 6 ระนาบ ระนาบละ 4 ดวง แต่ละระนาบจะเอียงทำมุม 55 องศา กับเส้นศูนย์สูตร (Equator) ในลักษณะสานกันคล้ายลูกตะกร้อ ดาวเทียมแต่ละดวงจะใช้เวลาในการโคจรรอบโลก 11 ชั่วโมง 58 นาที นั่น คือ คาบของการโคจรประมาณ 12 ชั่วโมง/รอบ โดยดาวเทียมจะส่งสัญญาณด้วยความถี่ 2 ย่านความถี่ คือ 1,575.42 MHz (L1) ที่ใช้งานทั่วไป และ ความถี่ 1,227.60 MHz (L2) ที่ใช้โดยทหาร

รูปที่ 2.2 แสดงตำแหน่งและการโคจรของดาวเทียม GPS รอบโลก

2.2.2 สถานีควบคุม (Control Station Segment)

ส่วนควบคุมประกอบด้วยสถานีที่ทำหน้าที่ควบคุมแล้วสังเกตการทำงานของดาวเทียม จำนวน 5 สถานี (Monitor Station) กระจายอยู่ทั่วโลก ได้แก่สถานี Diego Garcia, Ascension Island, Kwajalein, Hawaii และสถานี Colorado Springs โดยสถานีควบคุมหลักจะอยู่ที่ Falcon Air Force Base ใน Colorado Springs รัฐ Colorado สหรัฐอเมริกา สถานีควบคุมต่าง ๆ เหล่านี้มีหน้าที่

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

คอยติดต่อสื่อสาร (Tracking) กับดาวเทียม ทำการคำนวณผล (Computation) เพื่อบอกตำแหน่งของดาวเทียมแต่ละดวง และส่งข้อมูลที่ได้ไปยังดาวเทียมอยู่ตลอดเวลา ทำให้ข้อมูลที่ได้เป็นข้อมูลที่ทันสมัยอยู่เสมอ

รูปที่ 2.3 แสดงสถานีควบคุมระบบดาวเทียม GPS 5 แห่ง

2.2.3 ส่วนผู้ใช้ (Use Segment)

ผู้ใช้ประกอบด้วย 2 ส่วนใหญ่ ๆ คือ ส่วนที่เกี่ยวข้องกับทางทหาร (Military) และส่วนที่เกี่ยวกับพลเรือน(Civilian) การใช้งาน มี 2 โหมด โดยโหมดแรกคือโหมดละเอียด (Precise Positioning Service : PPS) ซึ่งผู้ใช้จะต้องได้รับการอนุญาตจากกระทรวงกลาโหมสหรัฐอเมริกาเท่านั้น และโหมดที่สองคือโหมดมาตรฐาน (Standard Positioning Service : SPS) สำหรับผู้ใช้ทั่วไปในการนำไปใช้งาน โดยจะมีความเที่ยงตรงต่ำกว่าแบบแรก

ในส่วนของผู้ใช้ การใช้งานทั้งหมดจะต้องใช้ผ่านอุปกรณ์ที่เรียกว่า ตัวรับสัญญาณจีพีเอส (GPS Receivers) ทำหน้าที่รับสัญญาณจากดาวเทียมจีพีเอส แล้วถอดรหัสสัญญาณเพื่อนำไปใช้ในการประมวลผลหาตำแหน่งต่อไป ผู้ใช้จึงมีหน้าที่พัฒนาเครื่องรับสัญญาณ (Receiver)ให้ทันสมัยและสะดวกแก่การใช้งาน สามารถที่จะใช้ได้ทุกแห่งในโลก และให้ค่าที่มีความถูกต้องสูง

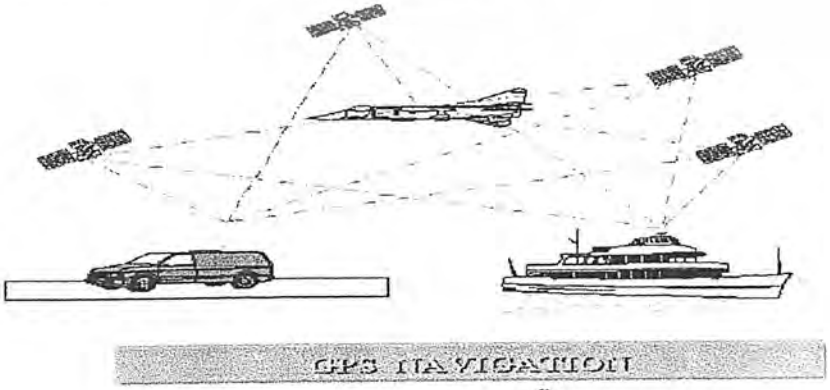
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 2.4 เครื่องรับสัญญาณ GPS

2.3 การนำ GPS มาประยุกต์ใช้งานในด้านต่างๆ

ด้วยความสามารถของ GPS ทำให้เราสามารถนำข้อมูลตำแหน่งมาใช้ประโยชน์ได้มากมาย โดยส่วนใหญ่ ได้แก่

1. Mobile Telecommunications
2. การคมนาคมในอวกาศ(Space navigation)
3. การเชื่อมโยงกับระบบการสื่อสาร(Position and Telecommunication)
4. การหาตำแหน่งยานพาหนะที่เคลื่อนที่
5. การสร้างแผนที่ (Mapping)
6. การวางแผนในการสำรวจเบื้องต้น (Survey)
7. สิ่งแวดล้อม (Environment) เป็นต้น



รูปที่ 2.5 แสดงการใช้งานทางบก ทางน้ำ และทางอากาศ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.4 หลักการพื้นฐานของ GPS

สามารถแบ่งออกได้เป็น 5 ขั้นตอน คือ

ขั้นที่ 1 การวัดพิกัด

การวัดตำแหน่งบนพื้นโลกเพื่อให้ได้ตำแหน่งพิกัดใน 3 มิติ (three dimensions : x, y ,z) และเวลา (GPS time) จะต้องใช้ดาวเทียมอย่างน้อย 4 ดวง โดยเมื่อรู้ระยะห่างของพิกัดที่ต้องการกับดาวเทียมแต่ละดวง ก็จะสามารถหาจุดตัดของทรงกลม จนได้พิกัดในตำแหน่งนั้นๆได้

รูปที่ 2.6 แสดงการหาระยะจากดาวเทียม

ขั้นที่ 2 การวัดระยะจากดาวเทียม

จากการที่ GPS ต้องรู้ระยะทางจากเครื่องรับถึงดาวเทียมในการหาพิกัด ซึ่งวิธีการหาระยะทำได้โดยการที่ระบบ GPS ทำงานเพื่อหาว่าสัญญาณวิทยุที่ส่งมาจากดาวเทียมจนถึงเครื่องรับใช้เวลาเดินทางนานเท่าใด แล้วนำเวลาที่หาได้มาคำนวณระยะทาง โดยใช้สมการคือ อัตราความเร็วคูณด้วยเวลา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ขั้นที่ 3 การหาเวลา

ในดาวเทียมจะใช้นาฬิกาอะตอมเพื่อให้เกิดความคลาดเคลื่อนในการคำนวณน้อยที่สุด เนื่องจากเวลาที่ผิดพลาดไปไม่กี่วินาที ก็สามารถทำให้คำนวณหาระยะทางผิดไปเป็นระยะทางหลายกิโลเมตรได้

ขั้นที่ 4 หาตำแหน่งของดาวเทียมที่อยู่ในอวกาศ

เพื่อใช้ประกอบการคำนวณหาพิกัดที่ต้องการได้ เพราะว่าในการหาค่าพิกัดจำเป็นที่จะต้องรู้ระยะห่างของตัวเครื่องรับกับดาวเทียม

ขั้นที่ 5 การหาความช้าของสัญญาณในการเดินทางผ่านชั้นบรรยากาศ

เมื่อแสงหรือสัญญาณวิทยุเดินทางผ่านตัวกลางที่มีความหนาแน่น เช่น ชั้นที่มีอนุภาคประจุไฟฟ้าที่หนา จะทำให้คลื่นวิทยุเดินทางได้ช้าลง ซึ่งจะทำให้ผลการคำนวณระยะที่ได้ไม่ถูกต้อง เราจึงต้องรู้ค่าที่อาจทำให้เกิดความช้านี้เพื่อทำการปรับผลของการคำนวณต่อไป

อย่างไรก็ตาม ยังมีสาเหตุของความคลาดเคลื่อนอื่นๆ ที่แก้ไขได้ยาก ทำให้การบอกตำแหน่งด้วยเครื่อง GPS ยังคงมีความไม่แม่นยำในบางครั้ง แต่จากที่กล่าวมาทั้งหมดนี้ ทำให้เราทราบว่า GPS มีหลักการอย่างไรบ้างในการกำหนดตำแหน่งของเครื่องรับ ผลที่ได้นั้น จะแสดงออกมาในรูปละติจูด ลองติจูด ความสูง เวลา และอื่นๆ ดังที่จะกล่าวในบทต่อไป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 3

NMEA 0183 Protocal

NMEA 0183 คือ โปรโตคอลมาตรฐานที่ใช้ในการสื่อสารในระบบ GPS ถูกพัฒนาขึ้นโดย National Marine Electronic Association (NMEA) ซึ่งเป็นหน่วยงานทหารเรือของสหรัฐอเมริกา โปรโตคอลตัวนี้จะกำหนดลักษณะของ สัญญาณทางไฟฟ้าที่ใช้ในการติดต่อสื่อสาร การรับ-ส่งข้อมูล เวลาและลักษณะจำเพาะของรูปแบบประโยคที่ใช้ในการสื่อสาร

3.1 ลักษณะสัญญาณทางไฟฟ้าของ NMEA 0183 ที่ใช้ในการติดต่อสื่อสาร

อุปกรณ์ที่จะทำหน้าที่ติดต่อสื่อสารที่ใช้โปรโตคอลชนิดนี้จะแบ่งเป็น 2 ประเภท คือ Listeners กับ Talkers โดย Listeners คือ อุปกรณ์รับข้อมูล หรือ Software และ Talkers คือ อุปกรณ์ส่งข้อมูล ระบบของโปรโตคอล NMEA 0183 สามารถที่จะมีหลาย Listeners ที่ใช้ติดต่อกับ Talker เพียงผู้เดียว โดยจะมีตัวแปรต่าง ๆ ดังนี้

รูปที่ 3.1 แสดงตัวอย่างสัญญาณที่ส่งมาจากเครื่องรับสัญญาณ GPS

Baud rate	:	4800 bps
Number of data bits	:	8 (bit 7 is 0)
Stop bits	:	1
Parity	:	None

เพื่อที่จะให้เป็นมาตรฐานเดียวกันในเรื่อง ระดับสัญญาณ รูปแบบ และลักษณะสัญญาณ โปรโตคอล NMEA 0183 จึงกำหนดให้มีการติดต่อผ่านพอร์ตอนุกรม RS-232

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.2 ลักษณะโดยทั่วไปของรูปแบบข้อมูลของ NMEA 0183

โดยทั่วไปรูปแบบของการส่งข้อมูลจะเป็นรหัส ASCII เริ่มต้นประโยคจะขึ้นต้นด้วย $ และจะปิดท้ายประโยคด้วย <CR><LF> จำนวนตัวอักษรแต่ละประโยคจะไม่เกิน 80 ตัวอักษร รูปแบบของประโยคโดยทั่วไปจะเป็น

$<header>,<data items>*cc<CRLF>.

ประโยคพื้นฐานจะมีอยู่ 3 ชนิด คือ

1. Talker sentence
2. Proprietary sentence
3. Query sentence

3.2.1 Talker sentence

ประโยคชนิดนี้จะเริ่มต้นประโยคด้วย $ แล้วตามด้วยอักษรอีกสองตัว (tt) ซึ่งเป็นตัวบอกว่าเป็น Talker sentence อักษรอีกสามตัวถัดมาจะเป็นตัวบอกว่าข้อมูลนี้คือข้อมูลอะไร (sss) ตัวอักษรที่เหลือจะตามด้วยกลุ่มของตัวเลข (d1,d2,d3,...,dn) โดยแต่ละกลุ่มจะถูกแบ่งด้วยคอมมา (,) ถ้าเลขกลุ่มใดไม่มีข้อมูลส่งมาตัวคอมมาก็ยังคงต้องมีอยู่ ในตอนท้ายประโยคจะปิดประโยคด้วย <CR><LF> รูปแบบของประโยคทั่วไป คือ

$ttsss,d1,d2,d3,...,dn,<CR><LF>

3.2.2 Proprietary sentence

โปรโตคอล NMEA 0183 ผู้ใช้สามารถที่จะกำหนดรูปแบบของประโยคเองได้แต่ประโยคจะต้องมีลักษณะดังนี้ คือ จะเริ่มต้นประโยคด้วย $P ตามด้วยอักษรสามตัวซึ่งเป็นตัวกำหนด ID ของผู้ใช้ ตัวอักษรหลังจากนี้จะเป็นไปตามที่ผู้ใช้จะกำหนด ซึ่งต้องกำหนดให้มีลักษณะตามรูปแบบประโยคมาตรฐาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.2.3 Query sentence

ประโยค Query sentence จะเป็นประโยคที่อุปกรณ์ listeners จะร้องขอข้อมูลจากอุปกรณ์ talkers ประโยคจะขึ้นต้นด้วย $ แล้วตามด้วยอักษรสอง(tt) ตัวซึ่งเป็นตัวบ่งบอกว่าเป็นประโยค talkers ชนิดไหนที่อุปกรณ์ listener ต้องการ ตัวอักษรอีกสองตัวถัดมา (ll) จะเป็นตัว ID ของอุปกรณ์ talkers ตัวอักษรตัวที่ห้าจะต้องเป็นตัวอักษร Q เสมอ เพื่อเป็นตัวบ่งบอกว่าเป็น Query sentence กลุ่มของอักษรถัดไปอีกสามตัว (sss) จะเป็นตัวกำหนดว่าประโยคนี้เป็นประโยคอะไรที่ listeners ต้องการ รูปแบบโดยทั่วไปของประโยคเป็นดังนี้

$ttllQ,SSS,<CR><LF>

3.3 ลักษณะของประโยค และรูปแบบต่าง ๆ

ลักษณะของประโยคจะเป็นตัวบ่งบอกว่าข้อมูลที่บรรจุอยู่ในตัวมันจะเป็นข้อมูลอะไรบ้าง และมีรูปแบบอย่างไร โดยประโยคมาตรฐาน NMEA 0183 version 2.01 มีประโยคหลักดังนี้

GGA : GPS Fix Data
: แสดงเวลา ตำแหน่งและข้อมูลดงที่ต่าง ๆ สำหรับเครื่องรับ GPS

GSA : GPS DOP and Active Satellites
: แสดงข้อมูลรูปแบบของข้อมูลที่รับได้

GSV : GPS Satellites in View
: จะแสดงหมายเลข PRN ของดาวเทียม

RMC : Recommended Minimum Specific GPS Data
: จะแสดงข้อมูลพื้นฐานที่จำเป็น เช่น เวลา วันที่ ตำแหน่ง และความเร็ว

ลักษณะของประโยคข้างต้นจะเป็นดังนี้

ประโยค GGA

$GPGGA,hhmmss.ss,llll.ll,a,yyyyy.yy,a,x,xx,x.x,x.x,M,x.x,M,x.x,xxxx*hh

คำแปลของประโยคนี้ GGA = Global Positioning System Fix Data

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ประโยค GGA จะประกอบด้วยข้อมูลแบ่งตามเครื่องหมายลูกน้ำดังนี้

Field 1 hhmmss.ss = UTC of Position

Field 2 lll.ll = Latitude

Field 3 a = N or S

Field 4 yyyyy.yy = Longitude

Field 5 a = E or W

Field 6 x = GPS quality indicator (0=invalid; 1=GPS fix; 2=Diff. GPS fix)

Field 7 xx = Number of satellites in use [not those in view]

Field 8 x.x = Horizontal dilution of position

Field 9 x.x = Antenna altitude above/below mean sea level (geoid)

Field 10 M = Meters (Antenna height unit)

Field 11 x.x = Geoidal separation (Diff. between WGS-84 earth ellipsoid and mean sea l
level. -=geoid is below WGS-84 ellipsoid)

Field 12 M = Meters (Units of geoidal separation)

Field 13 x.x = Age in seconds since last update from diff. reference station

Field 14 xxxx = Diff. reference station ID#

Field 15 = Checksum

ตัวอย่างประโยค GGA

$GPGGA,222435,3339.7334,N,11751.7598,W,2,06,1.33,27.0,M,-34.4,M,7,0000*41

ประโยค GSA

$GPGSA,A,3,19,28,14,18,27,22,31,39,,,,,1.7,1.0,1.3*35

คำแปลของประโยคนี้ GSA = GPS receiver operating mode, SVs used for navigation, and DOP values.

ประโยค GSA จะประกอบด้วยข้อมูลแบ่งตามเครื่องหมายลูกน้ำดังนี้

Field 1 a = Mode:

M=Manual, forced to operate in 2D or 3D

A=Automatic, 3D/2D

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Field 2 x = Mode:

1=Fix not available

2=2D

3=3D

Field 3-14 xx, xx , ... = IDs of SVs used in position fix (null for unused fields)

Field 15 x.x = PDOP

Field 16 x.x = HDOP

Field 17 x.x = VDOP

ประโยค GSV

$GPGSV,4,1,13,02,02,213,,03,-3,000,,11,00,121,,14,13,172,05*67

คำแปลของประโยคนี้ GSV = Number of SVs in view, PRN numbers, elevation, azimuth & SNR values.

ประโยค GSV จะประกอบด้วยข้อมูลแบ่งตามเครื่องหมายลูกน้ำดังนี้

Field 1 x = Total number of messages of this type in this cycle

Field 2 x = Message number

Field 3 xx = Total number of SVs in view

Field 4 xx = SV PRN number

Field 5 xx = Elevation in degrees, 90 maximum

Field 6 xxx = Azimuth, degrees from true north, 000 to 359

Field 7 xx = SNR, 00-99 dB (null when not tracking)

Field 8-11 xx , xx , xxx , xx = Information about second SV, same as field 4-7

Field 12-15 xx , xx , xxx , xx = Information about third SV, same as field 4-7

Field 16-19 xx , xx , xxx , xx = Information about fourth SV, same as field 4-7

ประโยค RMC

$GPRMC,hhmmss.ss,A,llll.ll,a,yyyyy.yy,a,x.x,x.x,ddmmyy,x.x,a*hh

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

คำแปลของประโยคนี้ RMC = Recommended Minimum Specific GPS/TRANSIT Data

ประโยค RMC จะประกอบด้วยข้อมูลแบ่งตามเครื่องหมายลูกน้ำดังนี้

Field 1 hhmmss.ss = UTC of position fix

Field 2 a = Data status (V=navigation receiver warning)

Field 3 lll.ll = Latitude of fix

Field 4 a = N or S

Field 5 yyyyy.yy = Longitude of fix

Field 6 a = E or W

Field 7 x.x = Speed over ground in knots

Field 8 x. x = Track made good in degrees True

Field 9 xxxxxx = UT date

Field 10 x.x = Magnetic variation degrees (Easterly var. subtracts from true course)

Field 11 a = E or W

Field 12 *h = Checksum

ตัวอย่างประโยค RMC

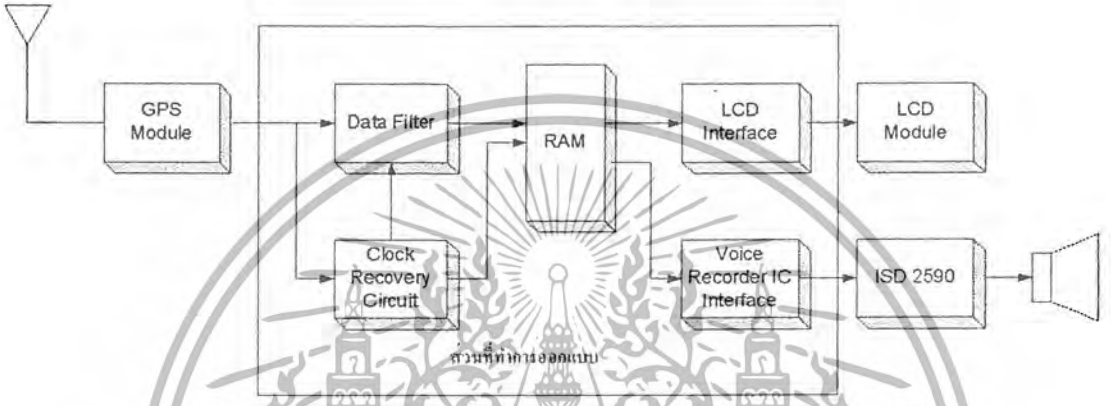
$GPRMC,185203,A,3339.7332,N,11751.7598,W,0.000,121.7,160496,13.8,E*55

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 4
การออกแบบและการสร้างวงจร

ในบทนี้จะได้กล่าวถึงการออกแบบ และสร้างวงจรที่ทำงานตามที่ต้องการ

4.1 การออกแบบวงจรรวม



รูปที่ 4.1 แสดงการทำงานของอุปกรณ์ที่ออกแบบไว้

จากรูปที่ 4.1 จะแสดงโครงสร้างของการทำงานในส่วนที่จะติดต่อกับ LCD โดยจะประกอบไปด้วย วงจรเครื่องรับสัญญาณ GPS (GPS Module) , วงจรกู้สัญญาณนาฬิกา (Clock recover circuit), วงจรส่วนกรองข้อมูล (Data filter) , วงจร LCD Interface , วงจรควบคุม IC เสียง , จอ LCD และ ISD 2590

การทำงานของวงจรดังกล่าวจะทำงานโดย ส่วนกรองข้อมูลจะรับสัญญาณข้อมูลที่มาจากเครื่องรับสัญญาณ GPS เพื่อทำการคัดเลือกข้อมูลที่จำเป็นเท่านั้น ในที่นี้จะทำการเลือกข้อมูลที่มาจากโปรโตคอล NMEA0183 โดยเลือกใช้ข้อมูล RMC ซึ่งข้อมูลนี้จะมีข้อมูล ละติจูด , ลองจิจูด , วันที่ และ เวลา แล้วนำไปเก็บไว้ใน RAM หลังจากนั้นจะทำการดึงข้อมูลที่เก็บไว้ไปแสดงผลที่ LCD Module โดยวงจรส่วน LCD INTERFACE

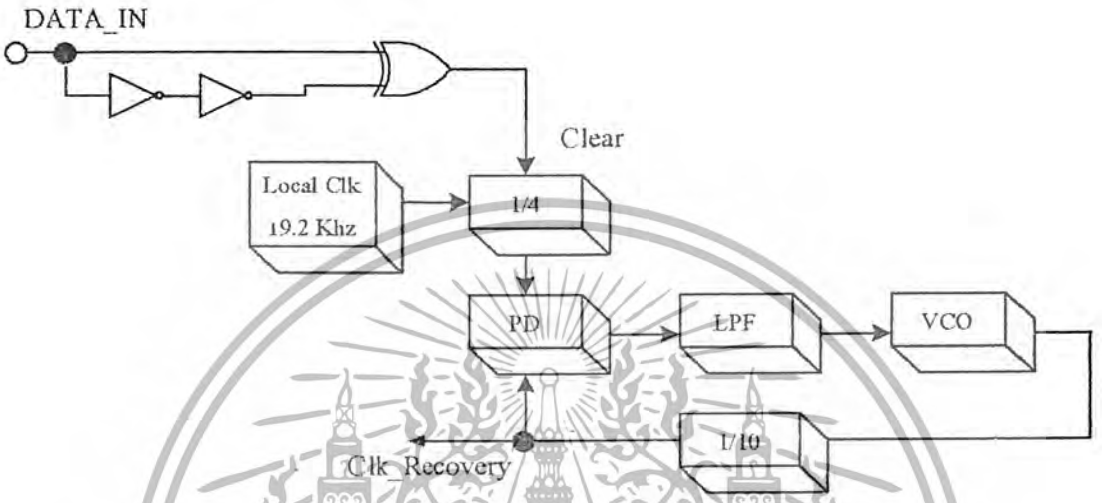
การทำงานในส่วนควบคุม IC ISD 2590 (Voice recorder IC Interface) นั้นจะทำการดึงข้อมูลจาก RAM โดยข้อมูลที่นำมานี้จะเป็นข้อมูล ละติจูด และ ลองจิจูด นำมาเปรียบเทียบกับข้อมูลที่อยู่ในหน่วยความจำ ถ้ามีข้อมูลที่ตรงกัน จะส่ง Address ไปให้ IC เพื่อให้พูดออกมาเป็นเสียง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สำนักหอสมุดกลาง พระจอมเกล้าลาดกระบัง

4.2 วงจรกู้สัญญาณนาฬิกา

จะทำการสร้างสัญญาณนาฬิกาที่ซิงโครไนซ์กับข้อมูลที่รับมาจากเครื่องรับสัญญาณ GPS ที่มีความถี่ 4800 บิตต่อวินาที โดยใช้หลักการของเฟสล็อคลูป ดังรูป



รูปที่ 4.2 หลักการของวงจรกู้สัญญาณนาฬิกา

จากรูปจะประกอบด้วยอุปกรณ์ สร้างความถี่ 19.2 KHz , วงจรหาร 4 , วงจรเฟสล็อคลูป (PLL) , วงจรหาร 10 , วงจรหาการเปลี่ยนแปลงขอบของสัญญาณ โดยวงจรสร้างความถี่ 19.2 KHz จะเป็นวงจรหารความถี่จากออสซิลเลเตอร์ที่มีอยู่ใน FPGA ในที่นี้จะใช้ความถี่ของคริสตอลที่ 24 MHz

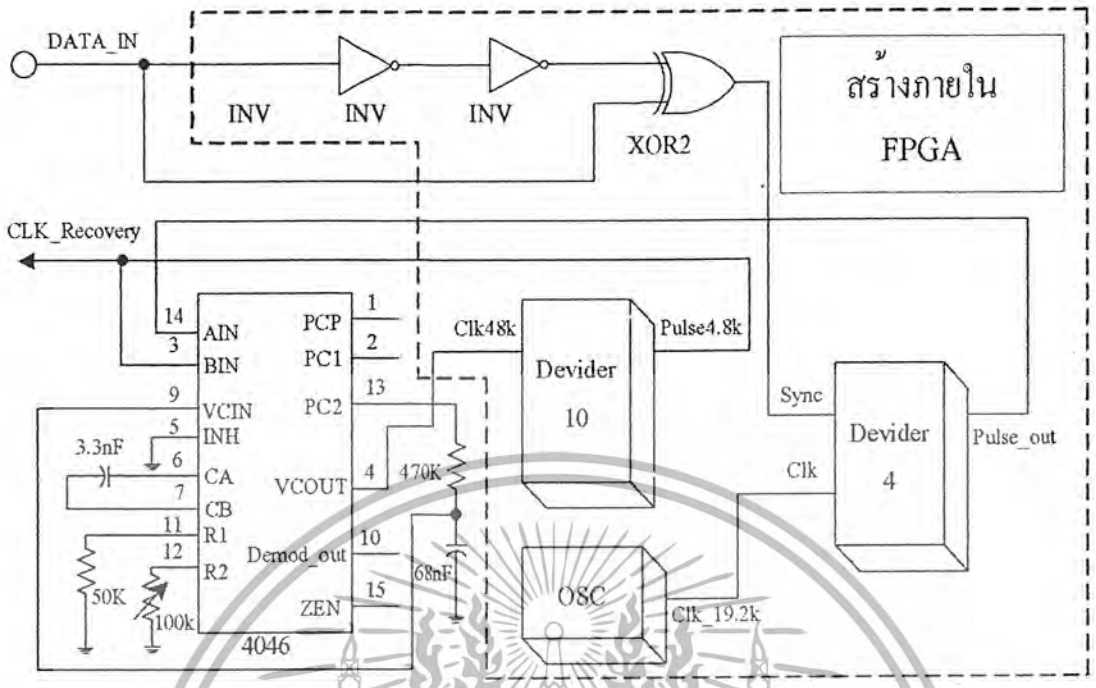
การทำงานของวงจรนี้จะทำงานโดย สัญญาณ DATA_IN จะนำมาผ่านเอ็กซ์คลูซีฟออร์ที่ขาข้างหนึ่ง ส่วนขาอีกข้างจะนำมาผ่าน NOT GATE เพื่อหน่วงเวลา ดังนั้นที่ขาเอาท์พุตของเอ็กซ์คลูซีฟออร์ จะเป็นสัญญาณพัลส์แคบ ๆ สัญญาณนี้จะเป็นตัวไปรีเซ็ต วงจรหาร 4 ทำให้สัญญาณเอาท์พุตของวงจรหาร 4 ซิงโครไนซ์ กับสัญญาณ DATA_IN สัญญาณที่ได้ออกจากวงจรหาร 4 จะเป็นสัญญาณนาฬิกาที่มีความถี่ 4800 Hz สัญญาณนาฬิกาตัวนี้จะไปผ่านเฟสดีเทคเตอร์ของ PLL โดยที่ PLL จะทำการล็อคให้ได้สัญญาณนาฬิกาที่ 4800 Hz

ในส่วนของวงจร PLL ทำการออกแบบให้ VCO ผลิตความถี่ออกมาเป็น 10 เท่า ของความถี่อินพุต ซึ่งในที่นี้คือ 48 KHz ดังนั้นจึงต้องทำการหารความถี่ตัวนี้ลงก่อน เพื่อที่จะได้นำสัญญาณนี้ป้อนกลับไปเปรียบเทียบกับความถี่อินพุต ซึ่งสัญญาณส่วนนี้จะเป็นความถี่ที่ให้แก่วงจรในภาคอื่น ๆ ต่อไป

จากหลักการข้างต้นสามารถที่ออกแบบวงจรได้ดังนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

55470



รูปที่ 4.3 วงจรกู้สัญญาณนาฬิกา

4.3 หน่วยกรองข้อมูล (Data filter)

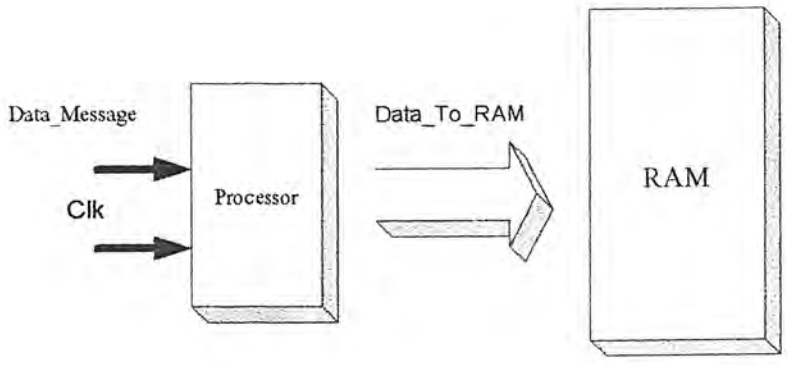
ในส่วนนี้จะขอแบ่งการทำงานออกเป็น 2 ส่วน ด้วยกันคือ

1. ส่วนที่ทำหน้าที่ในการ Detect ข้อมูลจากเครื่องรับ GPS
2. ส่วนที่ทำหน้าที่ในการ filter ข้อมูลที่ต้องการ

4.3.1 ส่วนที่ทำหน้าที่ในการ Detect ข้อมูลจากเครื่องรับ GPS

ส่วนนี้จะทำหน้าที่ในการพิจารณาข้อมูลแบบอนุกรมที่ถูกส่งมาจากเครื่องรับ GPS (GPS 9540) เพื่อนำข้อมูลส่วนที่ต้องการ ไปเก็บใน RAM1 เพื่อให้ส่วนอื่นนำไปใช้งานต่อไป

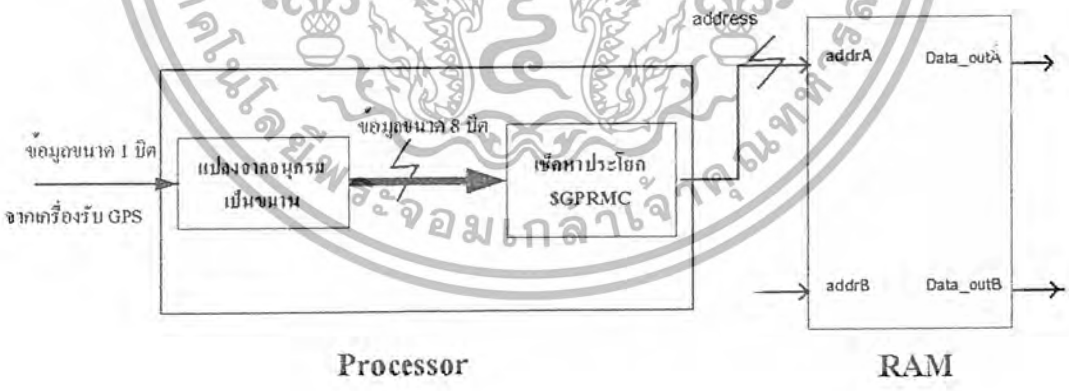
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.4 block diagram แสดงการรับและส่งข้อมูลระหว่างหน่วยอื่นๆ

การทำงานของวงจร

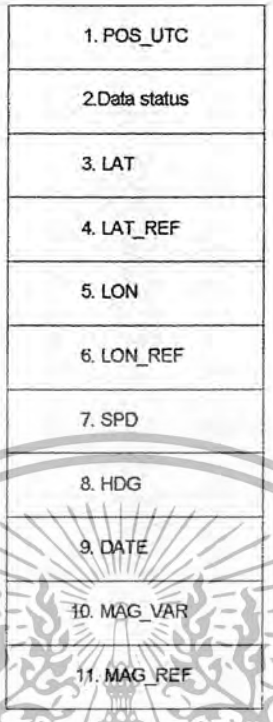
เนื่องจากข้อมูลที่ต้องการนั้น มีเพียงแค่ส่วนของประโยค RMC เท่านั้น ดังนั้นเราจะทำการค้นหา และพิจารณาเฉพาะส่วนของประโยค RMC โดยขั้นแรกจะทำการแปลงข้อมูลอนุกรมที่ได้มาจากเครื่องรับ GPS ให้เป็นข้อมูลแบบขนาน ที่มีขนาด 8 bit ซึ่งจะได้เป็นรหัส ASCII จากนั้นจึงค่อยพิจารณาค่า ASCII ที่ได้ โดยที่วงจรจะเริ่มเช็คหา $ ซึ่งเป็นตัวบอกจุดเริ่มต้นของประโยค จากนั้นจึงเช็คหา G, P, R, M และ C ต่อไปตามลำดับ ซึ่งก็จะทำให้ทราบได้ว่าเป็นประโยคที่ต้องการหรือไม่ เมื่อได้แล้วจึงจะทำการตัดส่วนของข้อมูลที่เหลือ ไปเก็บไว้ใน RAM I



รูปที่ 4.5 block diagram แสดงขั้นตอนการทำงาน

RAM สร้างขึ้นมาจากส่วนของ BLOCK RAM ใน FPGA โดยจะใช้แบบ dual port นั่นคือ สามารถนำข้อมูลเข้าไปเก็บ และอ่านข้อมูลที่เก็บอยู่ ออกมาได้ผ่านทางคนละ port กันแยกกันโดยอิสระ โดยการที่จะเอาข้อมูลเข้าไปใส่หรือดึงข้อมูลจาก RAM นั้น จะต้องทำการส่ง address เพื่อระบุตำแหน่งของข้อมูล แล้วจึงจะส่งหรือรับข้อมูลตามต้องการ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.6 แสดงโครงสร้างของการเก็บข้อมูลภายใน RAM

4.3.2 ส่วนที่ทำหน้าที่ในการกรองข้อมูลที่ต้องการ

หน้าที่ของการทำงานในส่วนนี้คือ นำข้อมูลที่ได้จากส่วนแรกมาทำการตัดข้อมูลในส่วนที่ไม่ได้ใช้งานออก

ข้อมูลที่อยู่ใน RAM1 หลังจากผ่านกระบวนการแรกแล้ว

,hhmmss.ss,A,lll.ll,a,yyyyy.yy,a,x.x,x.x,ddmmyy,x.x,a*hh

จะต้องนำข้อมูลนี้ไปตัดตัวอักษรComma (,) ออกและนำข้อมูลในส่วนที่ใช้งานออกมาโดย ข้อมูลที่ใช้งานคือ ค่า TIME, STATUS, LAT, LAT_REF, LON, LON_REF แล้วนำข้อมูลที่ต้องการใส่ไว้ใน RAM2 เพื่อสำรองเอาไว้ให้ในส่วนของ การแสดงผลทาง LCD และทางเสียง (ISD) ดึงข้อมูลที่ในแต่ละส่วนต้องการออกไปใช้งาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

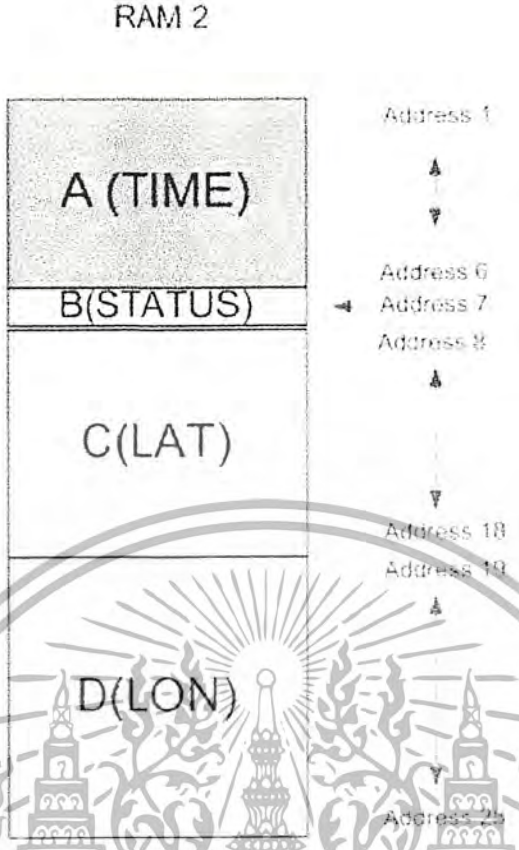
การทำงานของวงจร

เริ่มจากดึงข้อมูลจาก RAM1 ออกมา แล้วทำการตรวจสอบเครื่องหมายจุลภาคซึ่งจะถูกส่งมาก่อนข้อมูลตัวถัดไป โดยเมื่อเจอเครื่องหมายจุลภาคแต่ละตัว ก็จะต้องทำการนับไปด้วยเพื่อให้ได้ค่า field number อันจะบอกถึงข้อมูลที่ตามมา ว่าเป็นค่าของอะไร เช่น ถ้าเป็นเครื่องหมายจุลภาคตัวที่ 3 ข้อมูลที่ตามมาจะเป็นค่าของละติจูด เป็นต้น

จากนั้นข้อมูลหลังจากเครื่องหมายจุลภาคแต่ละตัว จะถูกส่งไปเก็บยัง RAM2 ในตำแหน่งที่กำหนดไว้แน่นอนในการเก็บข้อมูลแต่ละส่วน เพื่อให้ง่ายต่อการเรียกข้อมูลไปใช้งาน ดังรูป ทำเช่นนี้ไปจนกว่าจะได้ข้อมูลครบทุก field number เมื่อครบแล้วก็จะให้ไปเริ่มทำการหาประโยค $GPRMC ใหม่ไปเรื่อยๆ

รูปที่ 4.7 แสดงการทำงานของกระบวนการจัดการข้อมูล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



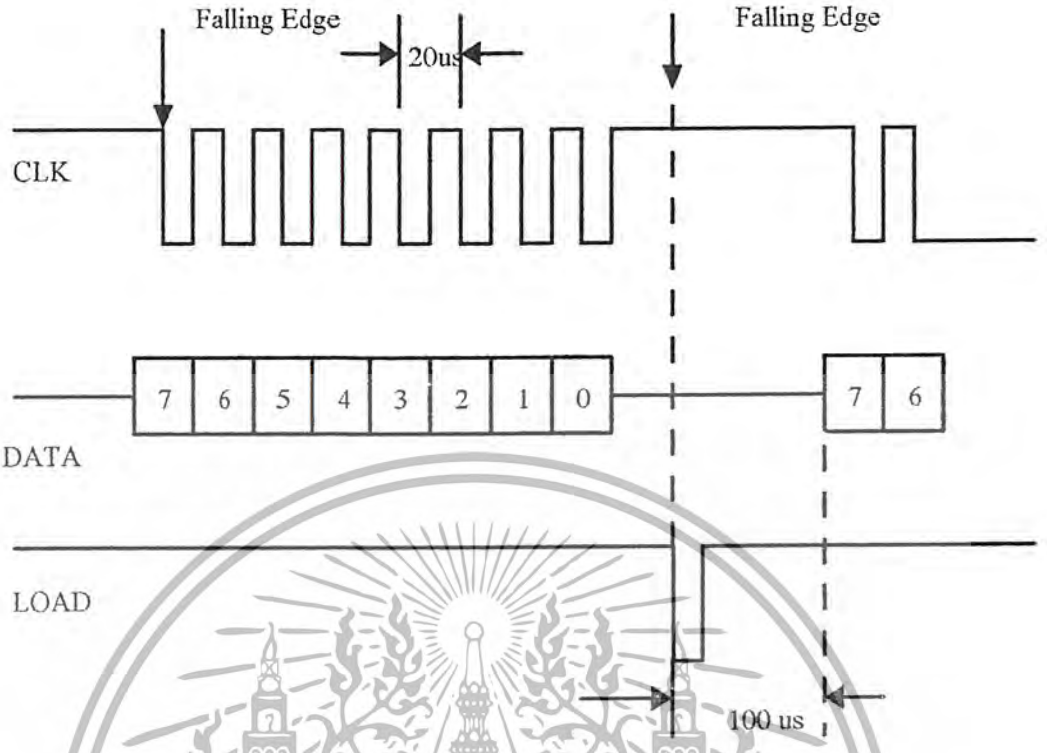
รูปที่ 4.8 แสดง ค่าต่างใน RAM2 หลังจากที่เสร็จสิ้นกระบวนการ

4.4 ส่วนแสดงผลทางหน้าจอ LCD (LCD INTERFACE)

4.4.1 สัญญาณควบคุมการทำงานของ LCD Graphic Display

การควบคุมการทำงาน LCD Module จะควบคุมผ่าน 3 Bit Serial ซึ่งการทำงานจะเป็นการส่งข้อมูลแบบ Synchronous กล่าวคือ สัญญาณจะขึ้นอยู่กับ Clock ขอบขาลงโดยไม่ขึ้นกับระยะเวลาของสัญญาณ(เหมือนRS232) ซึ่งจะเหมาะกับงานที่ค่อนข้างซับซ้อน และต้องดูแล INPUT/OUTPUT ที่ไม่แน่นอนจำนวนมาก การควบคุมแบบนี้จะใช้ขาสัญญาณ 3 เส้นคือ DIN (Data In), CLK (Clock) และ LD (Load) โดยรูปแบบการส่งข้อมูลจะกระทำครั้งละ 8 บิต (1 Byte) ทั้งนี้รูปแบบของข้อมูลจะเป็นเหมือนกับ RS232 ทุกประการ การควบคุมแบบนี้จะไม่มีการตอบรับสัญญาณการรับส่งข้อมูล เพราะฉะนั้นจะต้องหน่วงเวลาตาม Busy Time ที่กำหนดเสมอ หลังจากที่ส่งคำสั่งไปแล้วรายละเอียดของสัญญาณแสดงได้ดังรูปที่ 5.11

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



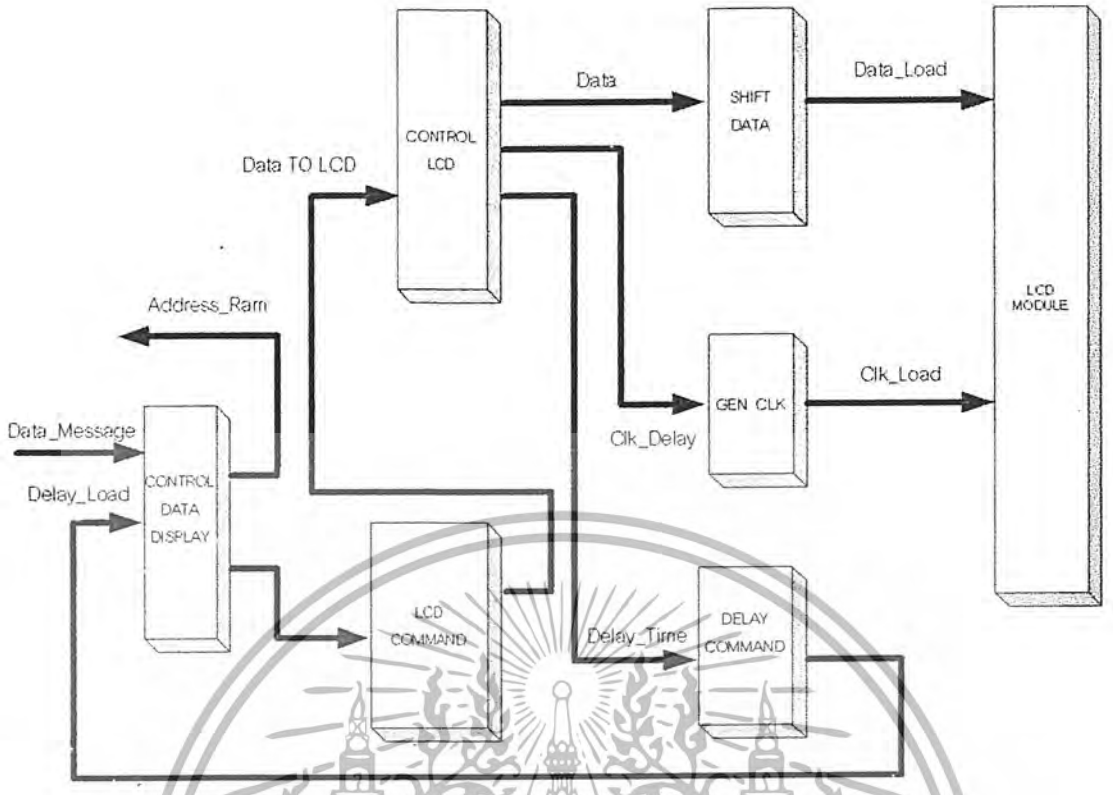
รูปที่ 4.9 สัญญาณควบคุม LCD Graphic Display ผ่านทาง 3 Bit Serial

4.4.2 ภาค LCD Graphic Interface

4.4.2.1 บล็อกไดอะแกรมภาค LCD Interface

วงจรเชื่อมต่อ LCD Interface จะทำหน้าที่นำข้อมูลที่ได้จากภาค DATA FILTER มาทำการแสดงผลโดยมีวงจรชิฟข้อมูล(Shifter Data) ทำหน้าที่รับข้อมูลเข้ามาครั้งละ 8 บิตและทำการชิฟข้อมูลที่ละบิต เพื่อป้อนให้กับตัวแสดงผล โดยมีวงจรกำเนิดสัญญาณนาฬิกา(Gen Clock) ทำหน้าที่กำเนิดสัญญาณนาฬิกาครั้งละ 8 ลูกเพื่อควบคุมการทำงานโดยจะทำการชิฟข้อมูลแต่ละครั้ง 8 ลูก โดยมีวงจรหน่วงเวลา (LCD Delay) ทำหน้าที่ในการหน่วงเวลาการทำงานในแต่ละคำสั่งในการนำข้อมูลไปแสดงผลซึ่งค่าในการหน่วงเวลาก็ขึ้นอยู่กับคำสั่งในแต่ละคำสั่งของ LCD Display ซึ่งคำสั่งต่างๆจะอยู่ใน วงจร LCD Command โดยมีวงจรควบคุมแอลซีดี (Control LCD) เป็นตัวกำหนดควบคุมการทำงานในการนำคำสั่งต่างๆไปแสดงผล ส่วน วงจรควบคุมการแสดงผลข้อมูล (Control Data Display) ทำหน้าที่ในการควบคุมการนำข้อมูลจากภาคประมวล มาแสดงผลดังแสดงดังรูป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.10 บล็อกไดอะแกรมภาค LCD Graphic Interface

4.4.2.2 วงจร LCD Graphic Interface

ทำหน้าที่เป็นวงจรเชื่อมต่อการส่งข้อมูลกับอุปกรณ์แสดงผล(LCD Graphic Display) ซึ่งการแสดงผลข่าวสารข้อมูลโดยการนำข้อมูลจากภาคจัดเรียงข้อมูลซึ่งมีขนาด 8 บิต มาแสดงผลโดยมีวงจรการทำงานดังแสดงดังในรูปที่ 5.13

4.4.2.3 วงจร Control Data Display

ทำหน้าที่ควบคุมการทำงานในการนำข้อมูลจาก ROM COMMAND เพื่อกำหนดคำสั่งในการแสดงผลของ LCD Graphic Display และจัดลำดับการส่งข้อมูล โดยการรับสัญญาณจากวงจร Delay Command ซึ่งเป็นการหน่วงเวลาข้อมูลแสดงผลและค่า BUSY TIME ของตัวแสดงผลด้วย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.4.2.4 วงจร Control LCD

ทำหน้าที่ควบคุมการนำข้อมูลไปแสดงผลโดยควบคุมการทำงานผ่านวงจรชิฟข้อมูล (LCD Shifter) และวงจรกำเนิดสัญญาณนาฬิกา(GEN Clk) ตลอดจนสัญญาณหน่วงเวลา (Delay Command) ในการนำข้อมูลไปแสดงผล

4.4.2.5 วงจรชิฟข้อมูล(Data Shifter)และวงจรกำเนิดสัญญาณ นาฬิกา(GEN Clk)

โดยวงจรทำหน้าที่รับข้อมูลเข้ามาขนาด 8 บิตและทำการชิฟข้อมูลครั้งละ 1 บิตโดยวงจรชิฟข้อมูล โดยมีวงจรกำเนิดสัญญาณนาฬิกาชุดละ 8ลูกในการให้จังหวะการชิฟข้อมูลขนาด 8บิต ข้อมูลไปยังตัวแสดงผล

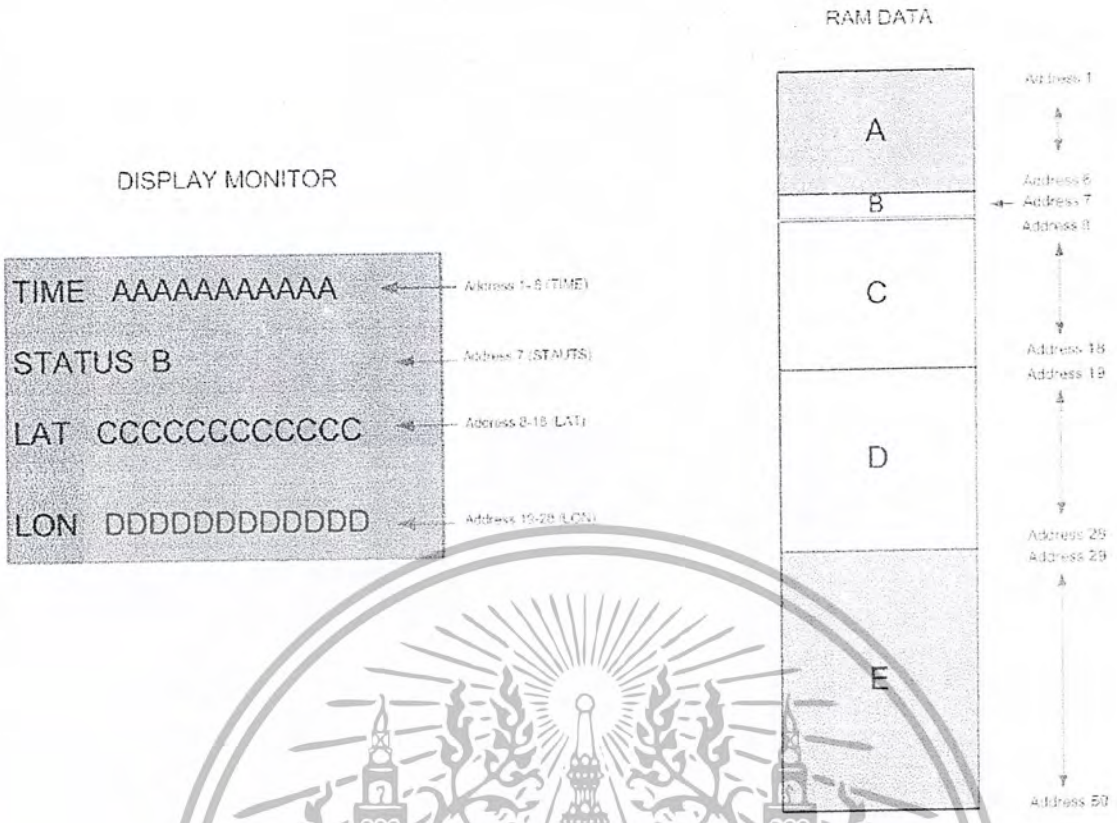
4.4.2.6 วงจร LCD Command

ทำหน้าที่เก็บคำสั่งในการทำงานของตัวแสดงผล(LCD Graphic Display) เช่นคำสั่งการเคลียร์จอภาพ การกำหนดบรรทัด(Row and Cloum) การกำหนดจำนวนการแสดงผลตัวอักษร การกำหนดให้ Cursor On หรือ OFF ตลอดจนการ Set Graphic ในการแสดงภาพแสดงบนจอภาพ เป็นต้น

4.4.2.7 วงจร Delay Command

วงจร Delay Command ทำหน้าที่หน่วงเวลาการแสดงผลข้อมูลตามค่า Busy Time ของ LCD Graphic Display

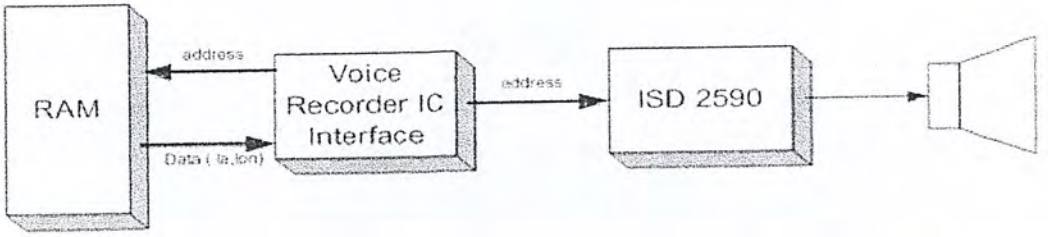
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.11 แสดงการนำข้อมูลมาแสดงผลทางหน้าจอ LCD

4.5 ส่วนแสดงผลออกมาเป็นเสียงพูด (Voice Recorder IC Interface)

เมื่อเราได้ข้อมูลพิกัดในตำแหน่งที่ยืนอยู่นั้น มาเก็บไว้ใน RAM เรียบร้อยแล้ว เราก็จะดึงข้อมูลเฉพาะส่วนของ latitude และ longitude มาจาก RAM เพื่อนำมาเปรียบเทียบกับค่าพิกัดของสถานที่ที่มีอยู่ในฐานข้อมูล เพื่อให้สามารถระบุได้ว่า พิกัดเรายืนอยู่นั้นตรงกับสถานที่ใด ก่อนที่จะส่งคำสั่งไปควบคุมให้ไอซี isd2590 แสดงผลออกมาเป็นเสียงพูดตามตำแหน่งสถานที่นั้นๆ เมื่อต้องการให้แสดงผลออกมาเป็นเสียงพูด ก็ทำได้โดยการกดปุ่มพูด เพื่อให้เครื่อง บอกชื่อสถานที่นั้นๆออกมา



รูปที่ 4.12 แสดงขั้นตอนการทำงานในส่วนแสดงเสียง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ยกตัวอย่างเช่น ตึก A มีพิกัดอยู่ในช่วง latitude ที่ 1343.6278N ถึง 1343.6320N และ longitude ที่ 10046.5984E ถึง 10046.5685E จะเห็นได้ว่า เลขนัยสำคัญอยู่ที่ 62 กับ 63 สำหรับ latitude และ 56 กับ 59 สำหรับ longitude ดังนั้น ในการดึงตัวเลขออกมาจาก RAM เพื่อนำมาเปรียบเทียบกับค่าพิกัดของตึกนี้ จึงไม่จำเป็นที่จะต้องดึงค่าทั้งหมดของ latitude และ longitude แต่ดึงมาเฉพาะตำแหน่งของเลขนัยสำคัญเท่านั้น โดยป้อน address ที่ตรงกับตำแหน่งที่เก็บค่าดังกล่าวแล้วนำค่าที่ได้ มาเก็บไว้ในตัวแปรที่กำหนดขึ้น โดยในที่นี้ จะกำหนดให้ LAT เป็นตัวแปรที่เก็บค่าของ latitude จาก RAM และตัวแปร LON เก็บค่า longitude จากนั้น จะนำค่านี้ไปเปรียบเทียบกับค่า latitude และ longitude ของตึกนี้และตึกอื่นๆเพื่อแสดงผลออกเป็นเสียง สำหรับส่วนการแสดงเสียงนั้น เราจะใช้ไอซีบันทึกเสียง ISD 2590 เพื่อเก็บเสียงที่ใช้ในการบอกสถานที่ที่เราต้องการ เราแค่ส่งค่า address ที่เก็บคำพูดที่เราต้องการเท่านั้น

ขั้นตอนต่างๆในการเปรียบเทียบค่าในกรณีที่ต้องการแสดงผลเพียงแค่ 2 อาคาร แสดงได้ดังนี้

รูปที่ 4.13 แสดงขั้นตอนการเปรียบเทียบค่าพิกัดเพื่อแสดงเสียงพูด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 5
การทดลอง

ในบทนี้จะเป็นการกล่าวถึงส่วนของการทดสอบการทำงานของวงจรที่ได้ออกแบบมาว่าสามารถทำงานได้จริงตามที่ออกแบบไว้หรือไม่ ในการทดสอบจะแบ่งออกเป็นสองส่วนคือ ส่วนที่หนึ่งจะเป็นส่วนของการจำลองการทำงาน(Simulation)ของวงจรก่อนที่จะโปรแกรมลงชิฟเอฟพีจีเอเพื่อดูผลตอบสนองของวงจรต่ออินพุทต่างๆที่ได้กำหนดให้ ส่วนที่สองจะเป็นส่วนของการทดสอบการทำงานของจริงวงจรทั้งหมดที่ได้ออกแบบมาเพื่อให้ทำงานเป็นตัวถอดรหัสในระบบ POCSAG สำหรับวิทยุติดตามตัว โดยใช้อุปกรณ์เอฟพีจีเอ(FPGA)

5.1 ทดสอบโดยการจำลองการทำงาน (Simulation)

หลังจากได้ออกแบบวงจรโดยการอธิบายลักษณะพฤติกรรมของวงจร(Hardware Description Language) เป็นที่เรียบร้อยแล้ว ผู้ออกแบบจำเป็นจะต้องทำการจำลองการทำงานเพื่อดูผลการตอบสนองของวงจรต่างๆ ต่ออินพุทที่กำหนดให้ซึ่งหากผลลัพธ์ที่ได้ไม่ถูกต้องจะได้ทำการแก้ไขต่อไป ในส่วนของการออกแบบ โดยอธิบายลักษณะพฤติกรรมของวงจรจะทำการออกแบบโดยใช้โปรแกรมในการออกแบบชื่อโปรแกรม Xilinx ISE 4.2I ซึ่งเป็นตัว Synthesis Tool และดาวน์โหลดลงในชิฟเอฟพีจีเอโดยใช้ตระกูล Virtex-E เบอร์ XCV1600E BG560 -6C ของบริษัท Xilinx

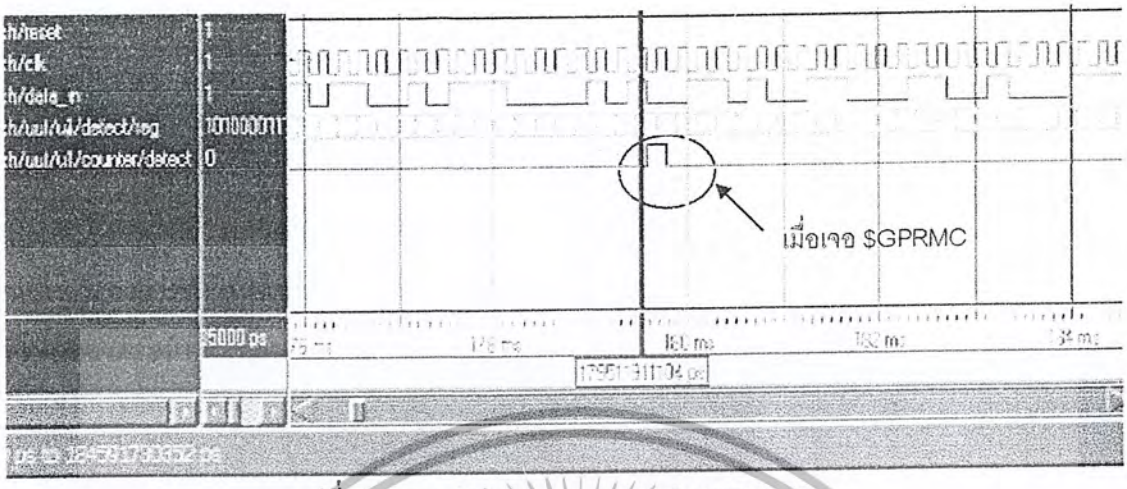
ในส่วนของการแสดงผลการทดสอบการทำงานของวงจรที่ได้ออกแบบด้วยภาษาวีเอชดีแอลที่สำคัญดังต่อไปนี้

5.1.1 ภาครับและกรองข้อมูล

จะต้องทำการกรองเอาเฉพาะข้อมูลในส่วนของ $GPRMC และนำข้อมูลที่ได้ไปเก็บใน RAM โดยการทำงาน จะแบ่งออกเป็น 2 ส่วนใหญ่ คือ ส่วนแรก จะทำการค้นหาประโยค $GPRMC เมื่อเจอแล้ว จะต้องส่งสัญญาณไปสั่งให้ส่วนที่สองทำงาน โดยจะให้ส่วนที่สองเริ่มเก็บข้อมูลนั้นลง RAM นั่นคือ จะต้องระบุ address และส่งข้อมูลที่ต้องการจะเก็บ ออกไปยัง RAM พร้อมๆกัน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ผลของการทำงานทั้งสองส่วน แสดงได้ดังนี้



รูปที่ 5.1 แบบจำลองการ detect หา $GPRMC

รูปที่ 5.2 แบบจำลองการทำงานการระบุ address และส่งข้อมูลไปยัง RAM

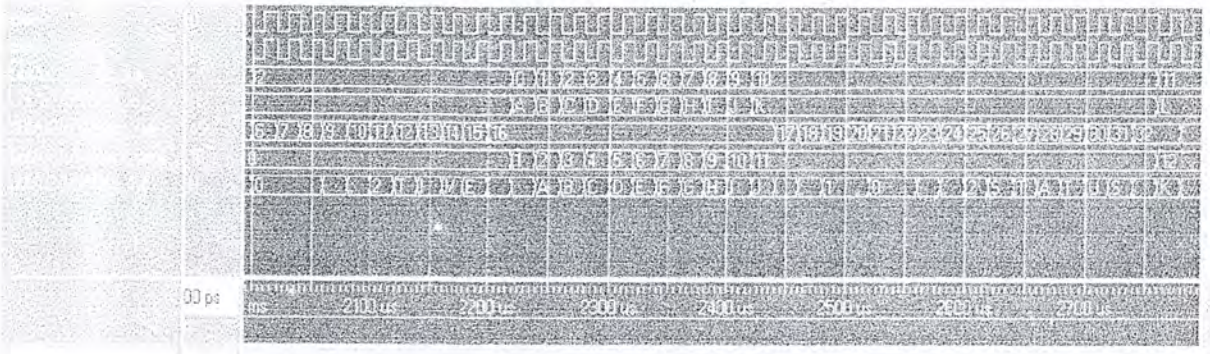
จากรูป จะเห็นได้ว่า การทำงานทั้งสองส่วนถูกต้องตามที่ต้องการ

5.1.2 ภาคเชื่อมต่อวงจรแสดงผล (LCD Display Interface)

เป็นภาคแสดงพิกัดโดยใช้ อุปกรณ์แอลซีดี เป็นตัวแสดงผลข้อมูลที่รับเข้ามาโดยจะแสดงผลเป็นภาษาอังกฤษ มีขนาด128×64 DOT 16×4 บรรทัด ซึ่งลักษณะการส่งข้อมูลจะทำการส่งข้อมูลแบบ 3 Bit Serial

ซึ่งในการแสดงผลนั้นจะแสดงค่าข้อมูลทั้งหมด 4 ค่าด้วยกันคือ ค่าเวลา (Time), ค่าสถานการทำงานของเครื่อง (Status), ค่าละติจูด (Latgitude), ค่าลองติจูด (Longitude)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 5.3 แบบจำลองการทำงานภาคเชื่อมต่อวงจรแสดงผล

การทดลองในส่วนนี้ทำการทดลองโดยทำการป้อนค่าข้อมูลสมมติเข้าไปให้ส่วนการทำงานภาคเชื่อมต่อวงจรแสดงผล จากรูปการจำลองการทำงานในส่วนนี้สามารถทำงานได้อย่างถูกต้อง

5.1.3 ภาคควบคุมการแสดงผลผ่าน IC ISD 2590

มีการทำงานหลักคือ จะต้องทำการเปรียบเทียบข้อมูลที่เก็บมาได้ใน RAM ว่าอยู่ในช่วงตรงกับพิกัดอาคารใด เมื่อทราบแล้ว จะต้องระบุ address ไปยัง ISD 2590 เพื่อให้แสดงเสียงพูดที่ตรงกันกับอาคารนั้นๆ โดยในการจำลองการทำงานนี้ จะสมมุติค่าต่างๆดังนี้

ตึก ก มีค่า latitude ที่ 1343.62 - 1343.63 และ longitude ที่ 10046.50 - 10046.54

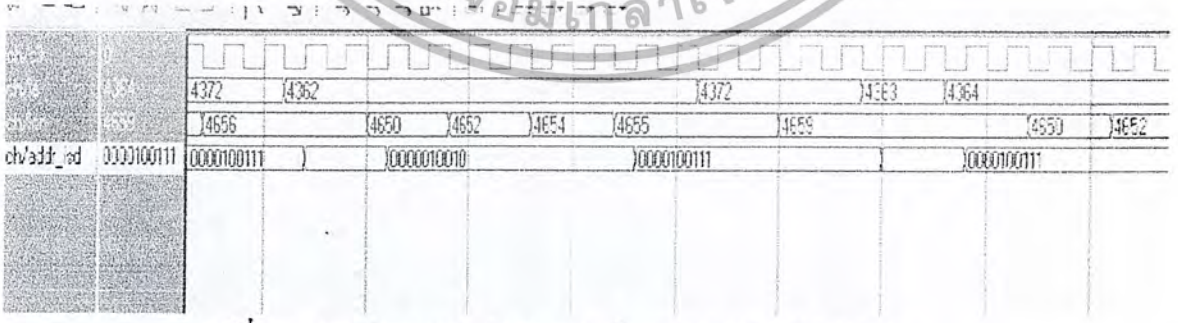
ตึก ข มีค่า latitude ที่ 1343.62 - 1343.63 และ longitude ที่ 10046.56 - 10046.59

และตำแหน่ง address ที่เก็บเสียงสถานที่ต่างๆ ใน ISD 2590 มีดังนี้

"ตึก ก" ตรงกับ address "0000100111"

"ตึก ข" ตรงกับ address "0000010010"

"ไม่มีข้อมูล" ตรงกับ address "0000100111"



รูปที่ 5.4 แบบจำลองการทำงานของส่วนควบคุม IC ISD 2590

จะเห็นได้ว่า วงจรสามารถส่งค่า address ที่ถูกต้องตามตำแหน่งอาคารนั้นๆ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5.2 การทำ Implementation บน FPGA รุ่น Spartan2 เบอร์ XC2S50-6TQ144

จากการออกแบบวงจรตัวถอดรหัสในระบบทั้งหมดเมื่อนำไปทำการ Implement ลงบน FPGA เบอร์ XC2S50-6TQ144 ซึ่งเป็นชิพขนาด 144ขา โดยจำนวนองค์ประกอบกลุ่มต่างๆ ที่มีอยู่ในชิพและจำนวนที่ถูกใช้งานไปแสดงดังตารางที่ 6.1 ส่วนตารางที่ 6.2 แสดงถึงค่าของเวลาที่หน่วงที่จะเกิดขึ้นในกรณีต่างๆที่จะต้องคำนึงถึงเมื่อจะนำแบบงานนี้ไปใช้งาน

ตารางที่ 5.1 จำนวนองค์ประกอบภายใน XC2S50-6TQ144 ที่ถูกใช้ในการทำ Implementaion

ลำดับ	องค์ประกอบของชิฟ(Element)	จำนวนที่ใช้	จำนวนที่มี	เปอร์เซ็นต์
1	Number of External GCLKIOBs	2	4	50
2	Number of External IOBs	41	92	44
3	Number of LOCed External IOBs	41	41	100
4	Number of BLOCKRAMs	1	8	12
5	Number of SLICEs	766	768	99
6	Number of GCLKs	2	4	50

ตารางที่ 5.2 ค่าเวลาหน่วงที่เกิดขึ้นภายใน FPGA

Delay Summary Report	Delay
Average Connection Delay for this Design is	1.799 ns
Maximum Pin Delay is	5.385 ns
Average Connection Delay on the 10 Worst Nets is	4.702 ns

จากตารางที่ 5.1 จะพบว่ามีการใช้งานของ CLB ไปทั้งหมด 766 Slice คิดเป็น 99% ของปริมาณทั้งหมดที่มีอยู่ภายในชิพเบอร์ XC2S50-6TQ144

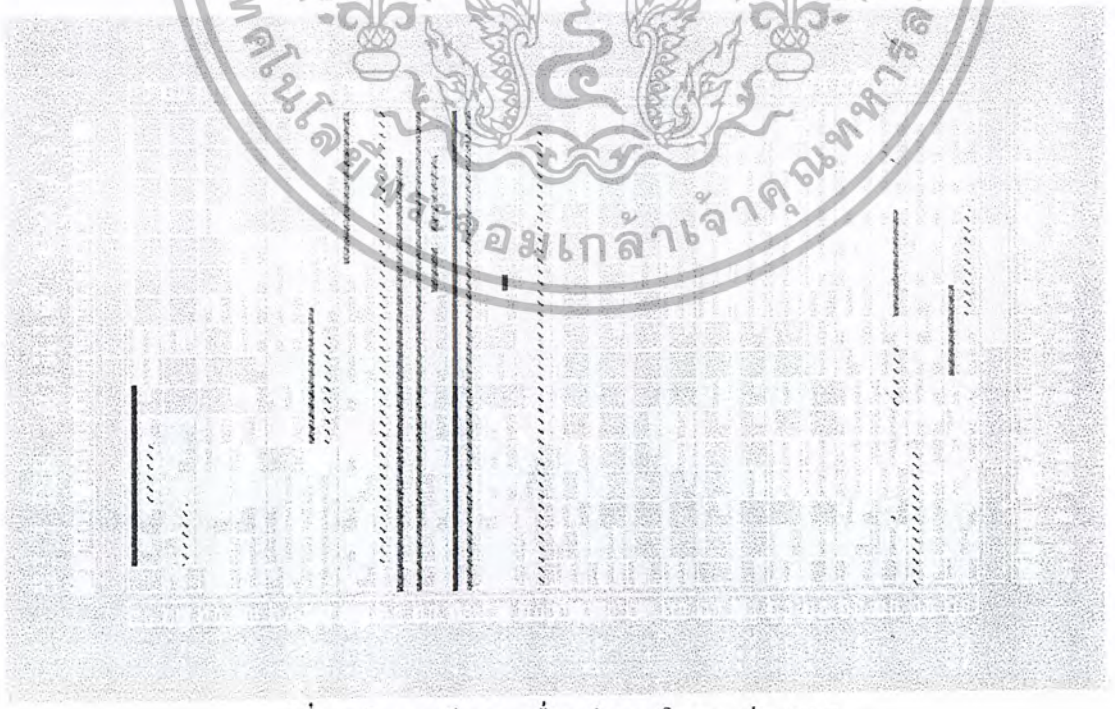
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากตารางที่ 5.2 เป็นข้อมูลที่เกี่ยวกับค่าของเวลาหน่วงที่ได้จากการทำ Implement บนชิพที่ได้กำหนดในตอนต้นซึ่งพบว่าจะมีค่าหน่วงเวลาของเส้นทางการเชื่อมต่อข้อมูล ในเส้นทางที่ยาวที่สุดภายในชิพประมาณ 4.702 ns และค่าเวลาหน่วงของขา I/O ประมาณ 5.385 ns

ส่วนตารางที่ 5.3 แสดงค่าของเวลาหน่วงสูงสุดของระบบที่เกิดขึ้นเป็น 10.024 ns ซึ่งเป็นการตอบสนองเพื่อให้เกิดการส่งผ่านของข้อมูลภายในชิพไปเป็นสัญญาณเอาท์พุตที่จะต้องใช้เวลาที่มากที่สุดภายหลังจากการเปลี่ยนแปลงของสัญญาณนาฬิกาของระบบ แต่อย่างไรก็ตามจากค่าของคาบเวลาที่ใช้ในการทำงานน้อยที่สุดเป็น 14.926 ns ทำให้ระบบสามารถทำงานที่ความถี่ได้สูงถึง 66.997 MHz และในรูปที่ 6.4 แสดงให้เห็นถึงโครงสร้างการเชื่อมต่อภายในระหว่าง CLB ต่างๆ

ตารางที่ 5.3 ค่าเวลาหน่วงและความเร็วสูงสุดที่ระบบสามารถทำงานได้

Timing Constraint	
Minimum Period	: 14.926 ns (Maximum Frequency : 66.997 MHz)
Minimum input arrival time before clock	: 10.024 ns
Maximum output required time after clock	: 13.682 ns
Maximum combination path delay	: 9.395 ns



รูปที่ 5.5 โครงสร้างการเชื่อมต่อภายในระหว่าง CLB

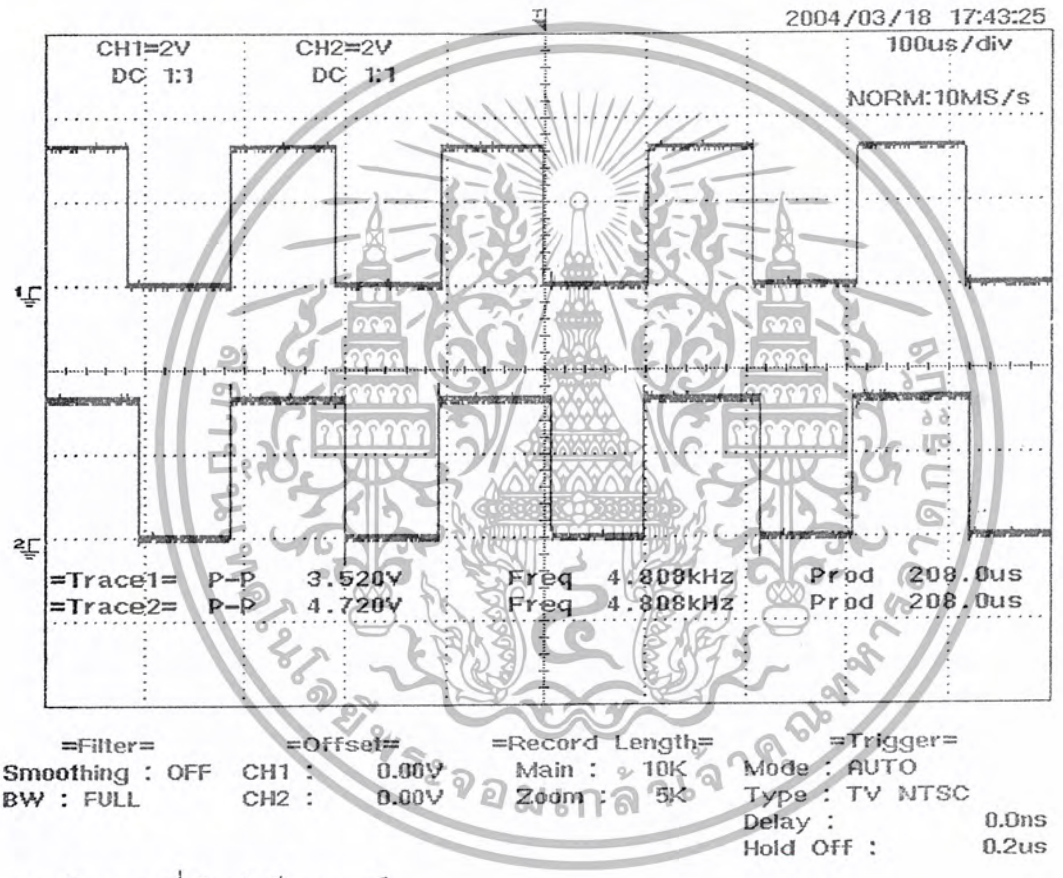
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5.3 การทดสอบการทำงานจริงของวงจร

จากบทที่ 4 ที่ได้ทำการออกแบบวงจร แล้วในบทนี้จะทำการทดสอบวงจรที่ได้ออกแบบมา โดยจะแบ่งเป็น 2 การทดลอง

5.3.1 การทดลองส่วน Clock recovery circuit

จากวงจรเฟสล็อคลูปในรูปที่ ทำการวัดสัญญาณระหว่าง สัญญาณที่เข้าเฟสดีเทคเตอร์ กับ สัญญาณ Clock recovery สามารถแสดงผลได้ดังนี้



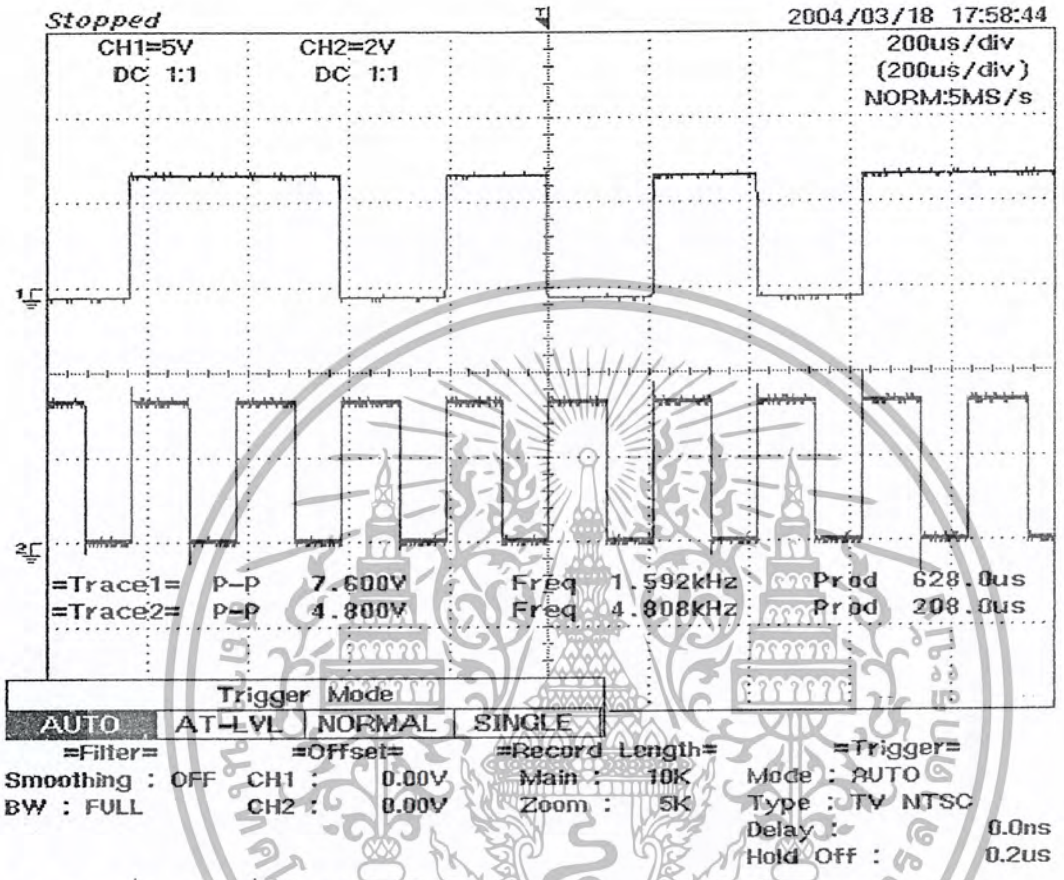
ch1: สัญญาณที่เข้าเฟสดีเทคเตอร์

ch2: สัญญาณ Clock recovery

รูปที่ 5.6 แสดงผลเปรียบเทียบระหว่างสัญญาณที่เข้าเฟสดีเทคเตอร์กับสัญญาณ Clock recover

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากรูปที่ 5.6 จะเห็นได้ว่าสัญญาณ Clock recovery ที่ได้ออกมาจากเฟสล็อคลูป นั้น ค่อนข้างที่จะ synchronize กับ สัญญาณที่เข้าที่เฟสดีเทคเตอร์ แต่ก็ยังไม่สามารถที่จะ synchronize กันได้พอดี เนื่องมาจากการออกแบบในส่วนวงจรกรองความถี่ต่ำผ่านยังไม่เหมาะสมเท่าที่ควร



ch1: Data ที่มาจากเครื่องรับสัญญาณ GPS

ch2: สัญญาณ Clock recovery

รูปที่ 5.7 แสดงสัญญาณเปรียบเทียบระหว่าง Data กับ Clock recovery

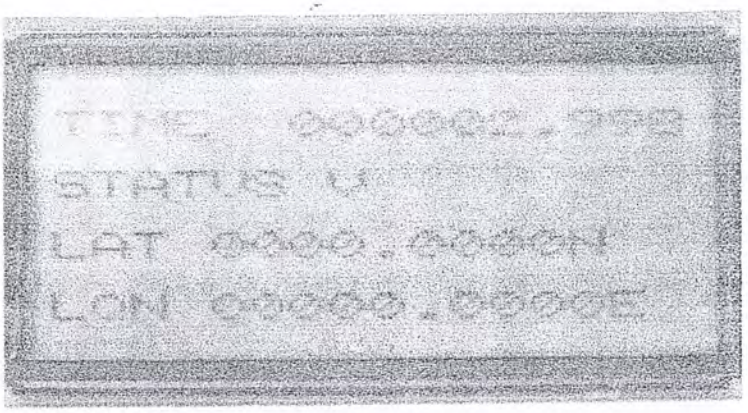
จากรูปที่ 5.7 จะเห็นได้ว่า สัญญาณ Clock recovery สามารถที่จะ synchronize สัญญาณ Data ที่มาจากเครื่องรับสัญญาณ GPS ได้ สังเกตจากการที่ขอบของข้อมูล และขอบของ clock ตรงกัน

5.3.2 ผลการทดลองภาคการแสดงผล

5.3.2.1 การแสดงผลทางจอ LCD

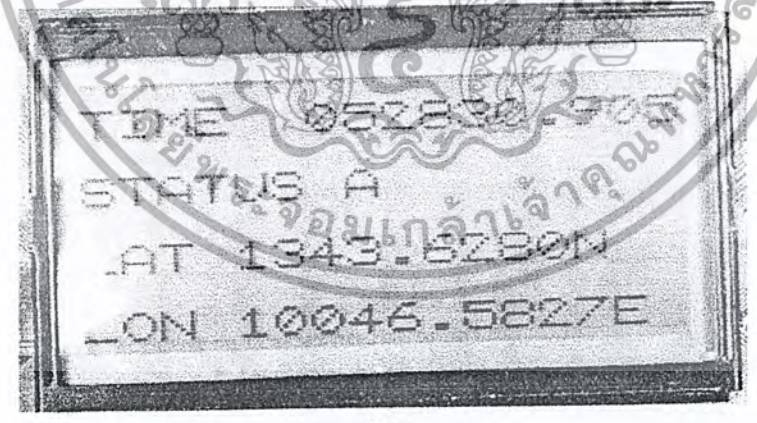
ผลการทดลอง สามารถแสดงข้อมูลเวลา สถานะของเครื่อง ละติจูด และลองจิจูด ได้ถูกต้องดังรูปที่ 5.8

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



(ก)

(ข)



(ค)

รูปที่ 5.8 แสดงตัวอย่างการแสดงผลทาง LCD Display ขณะที่เครื่องรับสัญญาณ GPS อยู่ ณ ตำแหน่ง (ก) อับสัญญาณ (ข) ตึก B (ค) ตึก A

5.3.2 การแสดงผลทางเสียง

การแสดงผลทางเสียง ได้ผลเป็นเสียงพูดออกมาถูกต้อง ตรงตามชื่อสถานที่ที่ยืนอยู่

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 6
สรุปและวิจารณ์ผลการทดลอง

ปริญญานิพนธ์ฉบับนี้ได้ทำการศึกษาการทำงานอย่างคร่าวๆของระบบ GPS รวมทั้งการนำเอาระบบนี้ ไปประยุกต์ใช้งานเพื่อให้เกิดประโยชน์ในชีวิตประจำวัน จากการศึกษาพบว่าระบบ GPS สามารถที่จะระบุตำแหน่งพิกัดได้ค่อนข้างแม่นยำ จึงได้ทำการทดลองสร้างเครื่องที่สามารถใช้ติดต่อกับเครื่องรับสัญญาณ GPS เพื่อให้แสดงผลค่าพิกัดที่จอ LCD โดยใช้ FPGA เป็นอุปกรณ์ควบคุมทั้งระบบ เนื่องจากข้อดีของ FPGA คือสามารถที่จะทำการโปรแกรมลงไปได้หลายครั้ง และสามารถออกแบบให้ตรงความต้องการได้ง่าย

6.1 ปัญหาที่พบในโครงงาน

จากบทที่ 4 ได้แบ่งวงจรออกเป็น 4 ส่วนคือ

1. Data filter
2. Clock recovery circuit
3. LCD interface
4. Voice recorder IC interface

ปัญหาที่พบมีดังนี้

1. Data filter จะทำการกรองข้อมูลที่ต้องการเพื่อส่งสัญญาณนี้ไปให้ส่วน LCD interface จากการทดลองพบว่า ข้อมูลที่ออกมาจากส่วนนี้ยังมีข้อผิดพลาดอยู่บ้าง เนื่องมาจากสัญญาณนาฬิกาที่ต้องจ่ายให้กับส่วนนี้ต้อง synchronize กับสัญญาณที่มาจากเครื่องรับสัญญาณ GPS นั้นยังไม่สมบูรณ์ จึงยังคงมีข้อมูลที่ผิดพลาดออกมาได้

2. Clock recovery circuit จะทำการกู้สัญญาณนาฬิกาที่มาจากเครื่องรับสัญญาณ GPS เพื่อเป็นสัญญาณนาฬิกาที่จ่ายให้กับทั้งวงจรที่ได้ออกแบบไว้ จากวงจรที่ได้ออกแบบไว้นั้นใช้ Phase lock loop เป็นอุปกรณ์ในการทำให้สัญญาณนาฬิกาที่จะจ่ายให้กับวงจรภาคอื่น ๆ นั้น synchronize กับข้อมูลที่มาจากเครื่องรับสัญญาณ GPS จากการทดลองพบว่าค่อนข้างคงที่อยู่ที่ 4.8 kHz แต่มีในบางช่วงที่วงจรส่วนนี้จ่ายความถี่ที่เกิน หรือ ต่ำกว่า 4.8 kHz ซึ่งเป็นสาเหตุที่ทำให้วงจรในภาคอื่น ๆ ทำงานผิดพลาดไป ส่วนสาเหตุที่ยังไม่สามารถที่จะทำให้ synchronize ได้อย่างสมบูรณ์นั้น ก็เนื่องมาจากการออกแบบวงจรกรองความถี่ต่ำผ่าน ยังออกแบบมาได้ไม่ดีนัก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3. LCD interface เนื่องจากตัว LCD ที่ใช้จะต้องทำการส่งข้อมูลที่อยู่ในรูปแบบอนุกรม วงจรในส่วนนี้จึงต้องจัดการส่งข้อมูลที่เหมาะสม วงจรในส่วนนี้จึงค่อนข้างใช้เวลาในการทำงานค่อนข้างนานกว่าทุกภาคในการแสดงผล เมื่อเทียบกับทุกภาค ซึ่งผลของเวลาที่นานนี้เองทำให้การแสดงผลบนจอ LCD ไม่สามารถทำงานอย่าง Real time ได้อย่างที่ต้องการ

4. Voice recorder IC interface โดย IC ที่ใช้คือ ISD 2590 ซึ่งมีการติดต่อกันแบบขนาน ทำให้วงจรส่วนนี้ไม่มีปัญหาเรื่องเวลาที่ใช้ แค่ส่งค่า Address ที่ตรงกับค่า Address ที่บันทึกเสียงไว้ก็สามารถออกเสียงได้

ผลโดยรวมของอุปกรณ์ที่สร้างขึ้นมานี้ ค่อนข้างให้ค่าที่สามารถที่จะออกไปใช้งานได้ แต่ยังมีข้อบกพร่องอยู่หลายประการ ซึ่งจะต้องมีการพัฒนาต่อไป

6.2 แนวทางในการพัฒนาต่อไป

เนื่องจากเครื่องที่สร้างขึ้นมานี้ ยังมีจุดที่น่าจะทำการปรับปรุงและแก้ไขต่อไป เพื่อทำให้เครื่องนี้ใช้งานได้ดี และมีประสิทธิภาพมากขึ้น ซึ่งมีดังนี้

1. LCD Display นั้นควรสามารถทำให้มีการแสดงผลแบบ Real time ได้ดีขึ้น โดยการสร้างวงจรที่สามารถรองรับการจ่ายสัญญาณ Clock ที่มีความถี่เพิ่มขึ้นได้ ซึ่ง LCD ตัวนี้สามารถรับสัญญาณ clock ได้ตั้งแต่ 0 kHz ถึง 50 k Hz

2. พัฒนาในส่วนวงจรกรองความถี่ให้เหมาะสมมากยิ่งขึ้น เนื่องจากเป็นส่วนที่มีความสำคัญต่อวงจร Clock recovery เพื่อให้สามารถที่จะล็อคความถี่ได้แม่นยำมากขึ้น

3. ในกรณีที่ต้องการบันทึกข้อมูลให้ได้จำนวนสถานที่มากกว่านี้ จะต้องเพิ่มหน่วยความจำให้มีขนาดใหญ่ขึ้น รวมทั้งต้องเลือกใช้ไอซีบันทึกเสียงที่สามารถบันทึกเสียงได้ยาวนานขึ้นตามไปด้วย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บรรณานุกรม

Elliott D.Kaplan , UNDERSTANDING GPS : PRINCIPLES AND APPLICATTION (Artech House ,1996)

Stefan Sjoholm and Lennart Lindh , VHDL for Designers (PRENTICE HALL , 1997)

Weng Fook Lee , VHDL : Coding and logic Synthesis with Synopsys (ACADEMIC PRESS)

http://www.rs.psu.ac.th/gps/gps.htm

http://www.tidi.nectec.or.th/download/GPSdesign.pdf

http://www.tidi.nectec.or.th/download/GPSdesignbyFPGA.pdf

http://project.cs.kku.ac.th/2545/seminar/extra/26/sheet.html#1

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก ก

ภาคแสดงผล(LCD Display)

ภาคแสดงผลซึ่งเป็น LCD รุ่นกราฟฟิกขนาด 128 x 64 Dots รุ่น TLCD-164 VERSION 1.0 THAI LCD MODULE ของบริษัท ศิลารีเสิร์ช จำกัด ที่มีส่วนไมโครคอนโทรล เลอร์อยู่ในชุดเดียวกัน ทำให้ใช้งานง่ายและสะดวก ต่อการควบคุมผ่านสัญญาณ RS232 หรือ 3 BIT Serial ซึ่งแสดงผลเป็นภาษาไทย (รหัส สมอ.) ที่มีการจัดบรรทัดอย่างถูกต้องได้ในตัวเอง ซึ่งมีคุณสมบัติการทำงานคือ

1. รับข้อมูลและคำสั่งควบคุมได้ 2 รูปแบบ คือแบบ RS-232 และแบบ 3 Bit Serial
2. แสดงอักษรไทย-อังกฤษได้16 อักษร 4 บรรทัดหรือในแบบกราฟฟิกขนาด128x64 Dots
3. มี VR ปรับความเข้มของ LCD
4. ใช้แหล่งจ่ายไฟ + 5VDC เท่านั้น
5. สามารถเลือกสัญญาณควบคุมได้ 3 แบบ คือ RS232 มาตรฐาน, RS232 แบบ LOGIC และ 3 BIT SERIAL
6. มีชุดคำสั่งในการควบคุม 8 คำสั่ง
7. สามารถแสดงข้อมูลในแบบกราฟฟิกได้ พร้อมทั้งมีคำสั่งสำหรับ On/Off Cursor และทำ Backspace ได้

ชุดคำสั่งการควบคุม

คำสั่งในการควบคุมจะมีทั้งหมด 8 คำสั่ง โดยมีรายละเอียดจะเป็นดังนี้

: CXX.... X <CR>

: คือรหัสนำของคำสั่ง (3AH)

C คือรหัสคำสั่งตั้งแต่ 1-8 (31H – 38H)

XX..... X คือข้อมูลติดตามของแต่ละคำสั่ง ซึ่งอาจจะมีหรือไม่มีก็ได้ รวมมีความยาวตามกำหนดในแต่ละคำสั่งด้วย

<CR> คือรหัสลงท้ายของคำสั่ง (ODH)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ ก.1 ชุดคำสั่งของ LCD Graphic Interface

คำสั่ง	รายละเอียด	BUSY TIME
:1RCC<CR>	Set Character Row & Colum R = 0-3 CC = 00 - 15	4 mS
:2XX .. X <CR>	Set Character (Max = 25 Char)	115 mS
:3PPYY <CR>	Set Graphic Page & Y PP = 00 –15 YY = 00 – 63	6mS
:4XX .. X <CR>	Set Graphic (Max = 25 Data)	3mS
:5 <CR>	Clear LCD	50 mS
:6 <CR>	Cursor ON	3 mS
:7 <CR>	Cursor OFF	6 mS
: 8 <CR>	Back Space	6 mS

รูปที่ ก.1 ตัวแสดงผล(LCD Display)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก ข

ไฟล์ข้อมูลภาษา VHDL ของ

CLOCK_4800_ALL.VHDL

```
library IEEE;
use IEEE.STD_LOGIC_1164.ALL;
use IEEE.STD_LOGIC_ARITH.ALL;
use IEEE.STD_LOGIC_UNSIGNED.ALL;

entity clk_4800 is
    Port (
---------- clk = 19.25 kHz + clk = 4800 Hz ----------
           reset_div_1250   : in std_logic;
           reset_div_10     : in std_logic;
           clk_in           : in std_logic;     -- from x'tal
           clk_out          : out std_logic;    -- to phase comparator2
---------- div_10 ----------
           from_VCO         : in std_logic ;    -- from vco
           to_Pll           : out std_logic ;   -- to phase comparator1
---------- edge detector ----------
           data             : in std_logic;
           to_not           : out std_logic;
           from_not         : in std_logic ;
----------
           clk_192          : out std_logic ;
           edge_out         : out std_logic );
end clk_4800;
architecture structure of clk_4800 is
        component div_1250
           Port ( reset     : in std_logic;
                  clk_in    : in std_logic;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                clk_out         : out std_logic);
    end component;

    component div_16
        Port ( reset            : in std_logic;
               clk_in           : in std_logic;
               clk_out          : out std_logic);
    end component;
    component div_10
        Port ( clk_in           : in std_logic;
               reset            : in std_logic;
               clk_out          : out std_logic);
    end component;
    component edge_1
        Port ( data_in          : in std_logic;
               from_not         : in std_logic;
               to_not           : out std_logic;
               edge_out         : out std_logic);
    end component;

signal  tmp             : std_logic;--_vector (1 downto 0);
signal  tmp_clk_192     : std_logic;
begin
    U0 : div_1250 port map
        ( clk_in        => clk_in ,
        clk_out         => tmp_clk_192,--tmp(0),
        reset           => reset_div_1250);

    U1 : div_16 port map
        ( clk_in        => tmp_clk_192,
        clk_out         => clk_out,
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        reset           => tmp );       --reset_div_16);

    U2 : div_10 port map
        ( clk_in        => from_VCO ,
          clk_out       => to_PII ,
          reset         => reset_div_10 );

    U3 : edge_1 port map
        ( data_in       => data,            -- from data
          edge_out      => tmp,             -- detec edge
          to_not        => to_not,          -- to not output
          from_not      => from_not);       -- from not input
          edge_out      <= tmp;
          clk_192       <= tmp_clk_192 ;
end structure;
```

ENABLE_STATE.VHDL

```
library IEEE;
use IEEE.STD_LOGIC_1164.ALL;
use IEEE.STD_LOGIC_ARITH.ALL;
use IEEE.STD_LOGIC_UNSIGNED.ALL;

entity ena_clk is
  generic ( N : integer := 500); -- divide by N
      Port ( clk_in         : in std_logic;
             reset          : in std_logic;
             clk_out        : out std_logic);
      end ena_clk;
architecture Behavioral of ena_clk is
signal count          : integer range 0 to N ;
signal tmp            : std_logic ;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
begin
        process (clk_in,reset)
        begin
                if reset = '1' then
                        count <= 0 ;
                        tmp <= '0' ;
                elsif clk_in'event and clk_in = '1' then
                                if count < N - 1 then
                                        count <= count + 1 ;
                                        tmp <= tmp;
                                else
                                        count <= 0;
                                        tmp <= not tmp ;
                                end if;
                        end if;
        end process;
        clk_out <= tmp;
end Behavioral;
```

ISD_CONTROLER.VHDL

```
library IEEE;
use IEEE.STD_LOGIC_1164.ALL;
use IEEE.STD_LOGIC_ARITH.ALL;
use IEEE.STD_LOGIC_UNSIGNED.ALL;
entity isd_controller is
   Port ( clk                   : in std_logic;
          din                   : in std_logic_vector(7 downto 0);
          addr_ram_out          : out std_logic_vector(7 downto 0);
          addr_isd_out          : out std_logic_vector(9 downto 0));
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
end isd_controller;

architecture Behavioral of isd_controller is
    component get_data
        Port    ( clk           : in std_logic;
                  addr_out      : out std_logic_vector(7 downto 0);
                  din           : in std_logic_vector(7 downto 0);
                  la            : out std_logic_vector(31 downto 0);
                  lon           : out std_logic_vector(31 downto 0)
                  );
    end component;

    component comparator
        Port    ( clk           : in std_logic;
                  la            : in std_logic_vector(31 downto 0);
                  lon           : in std_logic_vector(31 downto 0);
                  addr_isd      : out std_logic_vector(9 downto 0)
                  );
    end component;

signal  la              : std_logic_vector(31 downto 0);
signal  lon             : std_logic_vector(31 downto 0);

begin
    u1 : get_data  port map
        (clk            => clk,
        addr_out        => addr_ram_out,
        din             => din,
        la              => la,
        lon             => lon);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    u2 : comparator port map
            (clk            => clk,
            ia              => ia,
            ion             => ion,
            addr_isd        => addr_isd_out);
end Behavioral;
```

LCD_MODULE_INTERFACE.VHDL

```
use IEEE.STD_LOGIC_1164.ALL;
use IEEE.STD_LOGIC_ARITH.ALL;
use IEEE.STD_LOGIC_UNSIGNED.ALL;

entity lcd_module is
   Port ( start          :in std_logic;
          reset          :in std_logic;
          clock          :in std_logic;
          seladd_isd     :in std_logic_vector (7 downto 0);
          data_in        :in std_logic_vector (7 downto 0 );
          data_lcd       :out std_logic;
          load_lcd       :out std_logic;
          clock_lcd      :out std_logic;
          seladd_cut     :out std_logic_vector ( 6 downto 0);
          do_isd         :out std_logic_vector (7 downto 0));
end lcd_module;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
architecture Behavioral of lcd_module is

    component rom_command is
        port    (reset              : in std_logic;
                 clock              : in std_logic;
                 start              : in std_logic;
                 data_in            : in std_logic_vector (7 downto 0);
                 out_8clk_lcd       : out std_logic;
                 load_out_lcd       : out std_logic;
                 data_out_lcd       : out std_logic;
                 adr_sel            : out std_logic_vector (7 downto 0));
    end component;
    component cut_data is
        Port    (reset              : in std_logic;
                 clk                : in std_logic;
                 start              : in std_logic;
                 adr_rom            : in std_logic_vector ( 7 downto 0 );
                 adr_sound          : in std_logic_vector ( 7 downto 0 );
                 data_in            : in std_logic_vector ( 7 downto 0 );
                 do_rom             : out std_logic_vector( 7 downto 0 );
                 do_sound           : out std_logic_vector( 7 downto 0 );
                 start_rom          : out std_logic;
                 seladr_cut         : out std_logic_vector( 6 downto 0 ));
    end component;

signal buf_start_rom        : std_logic;
signal buf_adr_rom          : std_logic_vector( 7 downto 0 );
signal buf_do_rom           : std_logic_vector( 7 downto 0 );
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
begin

        u1: rom_command port map
                (reset              => reset,
                clock               => clock,
                start               => buf_start_rom,
                data_in             => buf_do_rom,
                out_8clk_lcd        => clock_lcd,
                load_out_lcd        => load_lcd,
                data_out_lcd        => data_lcd,
                adr_sel             => buf_adr_rom);

        u2: cut_data port map
                (reset              => reset,
                clk                 => clock,
                start               => start,
                adr_rom             => buf_adr_rom,
                adr_sound           => seladd_isd,
                data_in             => data_in,
                do_rom              => buf_do_rom,
                do_sound            => do_isd,
                start_rom           => buf_start_rom,
                seladr_cut          => seladd_cut);

end Behavioral;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

MIX_ALL(๑๑).VHDL

```
library IEEE;
use IEEE.STD_LOGIC_1164.ALL;
use IEEE.STD_LOGIC_ARITH.ALL;
use IEEE.STD_LOGIC_UNSIGNED.ALL;

entity mix_all is
    Port ( clk          : in std_logic;
           reset        : in std_logic;
           data_in      : in std_logic;
           wea          : in std_logic;
           addrb        : in std_logic_VECTOR(6 downto 0);
           ena_in       : in std_logic;
           data_Rx      : out std_logic_vector(7 downto 0);
           doa          : out std_logic_VECTOR(7 downto 0);
           dob          : out std_logic_VECTOR(7 downto 0));
end mix_all;

architecture Behavioral of mix_all is
signal  dia             : std_logic_VECTOR(7 downto 0);
signal  addra           : std_logic_VECTOR(6 downto 0);
signal  tmp_data_in     : std_logic ;
component mix
        Port    ( clk           : in std_logic;
                  reset         : in std_logic;
                  se_data_in    : in std_logic;
                  ena_in        : in std_logic;
                  data_Rx       : out std_logic_vector(7 downto 0);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                data_out        : out std_logic_vector(7 downto 0);
                addr            : out std_logic_vector (6 downto 0));
end component;

component ram1
        Port    (addra          : IN std_logic_VECTOR(6 downto 0);
                addrb           : IN std_logic_VECTOR(6 downto 0);
                clka            : IN std_logic;
                clkb            : IN std_logic;
                dia             : IN std_logic_VECTOR(7 downto 0);
                doa             : OUT std_logic_VECTOR(7 downto 0);
                dob             : OUT std_logic_VECTOR(7 downto 0);
                wea             : IN std_logic);
end component;
begin
        tmp_data_in             <= not data_in ;
      .. mix1 : mix port map
                (clk            => clk,
                reset           => reset,
                se_data_in      => tmp_data_in,
                 ena_in         => ena_in,
                data_Rx         => data_Rx,
                data_out        => dia,
                addr            => addra);
        ram_mix : ram1 port map (
                addra           => addra,
                addrb           => addrb,
                clka            => clk,
                clkb            => clk,
                dia             => dia,
                doa             => doa,
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
            dob         =>    dob,
            wea         => wea );
end Behavioral;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Leadtek®
We Make Dreams a Reality

LEADTEK GPS MODULE
TECHNICAL SPECIFICATION

GPS 9540

Leadtek®
We Make Dreams a Reality

Version 1.05
28 August 2002
All Rights Reserved

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Leadtek
We Make Dreams a Reality

2. Technical Specifications

2.1. Electrical Characteristics

2.1.1 General

Frequency	L1, 1575.42 MHz
C/A code	1.023 MHz chip rate
Channels	12

2.1.2 Accuracy

Position	15 meters, 2D RMS, SA off
	7 meters 2D RMS, WAAS corrected
	1-5 meters, DGPS corrected
Velocity	0.1 meters/second
Time	1 microsecond synchronized to GPS time

2.1.3 Datum

WGS-84

2.1.4 Acquisition Rate

Reacquisition	0.1 sec., average
Snap start	2 sec., average
Hot start	8 sec., average
Warm start	38 sec., average
Cold start	45 sec., average

2.1.5 Dynamic Conditions

Altitude	18,000 meters (60,000 feet) max.
Velocity	515 meters/second (1000 knots) max.
Acceleration	4g, max.
Jerk	20 meters/second^3, max.

2.1.6 Power

	GPS 9540 5V Version (TTL)	GPS 9540 3.3V Version (TTL)
Main Power	+5VDC ± 5%	3.3VDC ± 10%
Supply Current	170mA Typical	160mA Typical
Backup Power	+2.5V to 3.3V	+2.5V to 3.3V

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Backup Current	10μA Typical	10μA Typical

2.1.7 Serial Port

Electrical interface	Two full duplex serial communication, TTL interface
Protocol messages	SiRF binary and NMEA-0183, version 2.20 with a baud rate selection. SiRF binary-position, velocity, altitude, status, and control NMEA - GGA, GLL, GSA,GSV, RMC and VTG
DGPS protocol	RTCM SC-104

2.1.8 Time-1PPS Pulse

Level	TTL
Pulse duration	100ms
Time reference	At the pulse positive edge
Measurements	Aligned to GPS second, ±1 microsecond

2.2. Environmental Characteristics

Operating temperature range	-40 deg. C to +85 deg. C
Storage temperature range	-55 deg. C to +100 deg. C -45 deg. C to +80 deg with battery

2.3. Physical Characteristics

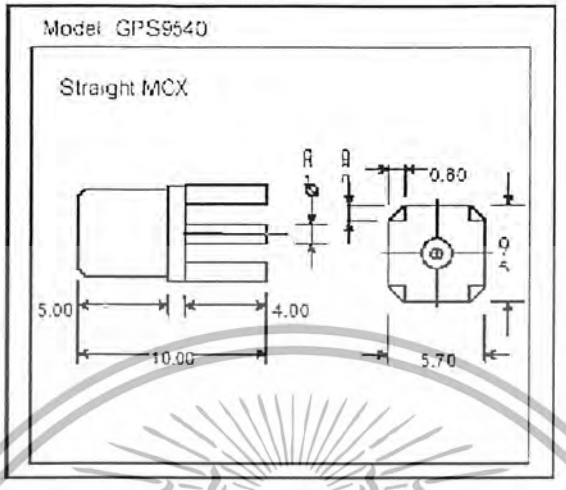
Length	2.8" (71.12 mm)
Width	1.6"(40.64mm)
Height	0.57"(14.4 mm)
Weight	25g
Antenna connector	MCX type
Interface connector	20-pin straight header, 2mm pitch board-to-board

2.4. Trickle Power Description

The GPS 9540 design supports the SiRF Trickle Power™ mode of operation. In this mode, the lowest average power dissipation is achieved by powering down the board (after a position is determined) in such a manner that when it is turned back on it can recomputed a position fix in the shortest amount of time. Standard Trickle Power operates in three states:

2.4.1 Tracking State

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



4. Pin Assignment of Connector

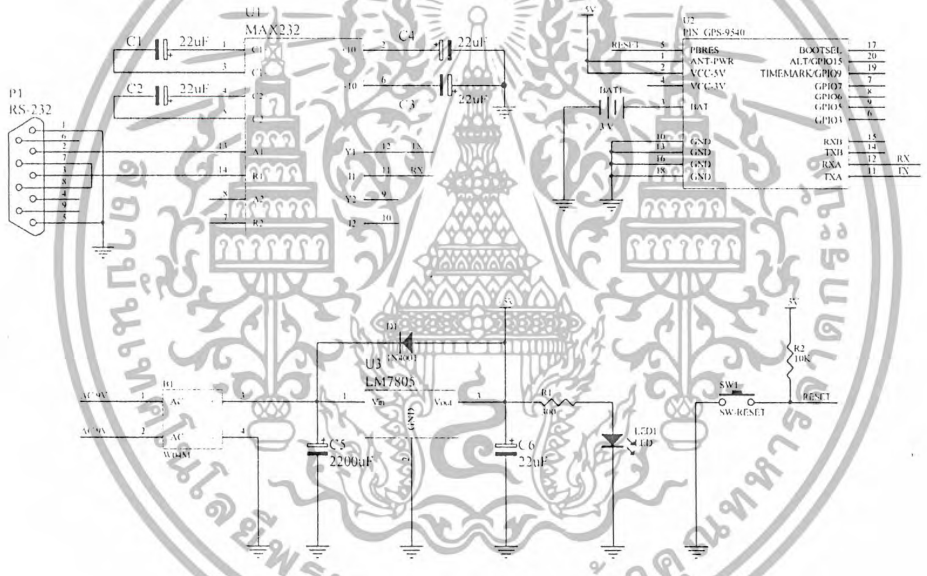
Table 1-1 Pin list of the 20-Pin Digital Interface Connector (J1)

Pin Number	*Name*	*Type*	*Description*
1	ANT_PWR	PWR	Antenna DC Voltage(**note1**)
2	VCC_5V	PWR	+5 DC Power Input(**note2**)
3	BAT	PWR	Backup Battery(**note3**)
4	VCC_3V	PWR	+3.3V DC Power Input(**note4**)
5	PBRES	I	Push Button Reset Input. Active Low
6	GPIO3	I/O	SW dependent functions (**note5**)
7	GPIO7	I/O	SW dependent functions (**note5**)
8	GPIO6	I/O	SW dependent functions (**note5**)
9	GPIO5	I/O	SW dependent functions (**note5**)
10	GND	PWR	Ground
11	TXA	O	Serial Data Output A
12	RXA	I	Serial Data Input A
13	GND	PWR	Ground
14	TXB	O	Serial Data Output B
15	RXB	I	Serial Data Input B
16	GND	PWR	Ground
17	BOOTSEL	I	Booting Mode Select
18	GND	PWR	Ground
19	TIMEMARK/GPIO9	I/O	1PPS Time Mark Output (**note5**)
20	ALT/GPIO15	I/O	Alternative output (**note6**)

Note: 1) If the module is build-in antenna power type, the pin is no used.
2) If the module is 3.3V type, the pin is no used.
3) Maximum voltage is 1.9V
4) If the module is 5V type, the pin in no used.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Connection GPS module



Title		
Size: B	Number:	Revision:
Date: 16-Jul-2003		Sheet of
File: D:\GPS\GPS.SCH		Drawn By:

3.1.5 Recommended Minimum Specific GPS Data (RMC). This message contains time, date, position, course, and speed data. The fields in this message will always contain data even when the receiver is not navigating. This allows user-initialized, stored, or default values to be displayed before a solution is obtained.

The contents of the RMC Message are described in Table III-6.

Table III-6. RMC Message: Recommended Minimum Specific GPS Data Message

Message ID:		RMC		
Rate:		Variable; defaults to 1 Hz		
Fields:		11		
Field No.:	Symbol:	Field Description:	Field Type:	Example:
	S_ _RMC	Start of sentence and address field		$GPRMC
1	POS_UTC	UTC of position (hours, minutes, seconds, decimal seconds)	hhmmss.ss	185203
2	POS_STAT	Position status (A = Data valid, V = Data invalid) (Note 1)	a	A
3	LAT	Latitude	llll.ll	3339.7332
4	LAT_REF	Latitude direction (N = north, S = south)	a	N
5	LON	Longitude	yyyyy.yy	11751.7598
6	LON_REF	Longitude direction (E = east, W = west)	a	W
7	SPD	Speed over ground (knots)	x.x	0.000
8	HDG	Heading/track made good (degrees True)	x.x	121.7
9	DATE	Date (dd/mm/yy)	xxxxxx	160496
10	MAG_VAR	Magnetic variation (degrees)	x.x	13.8
11	MAG_REF	Magnetic variation (E = east, W = west) (Note 2)	a	E
	CKSUM	Checksum	*hh	*55
	<CR><LF>	Sentence terminator		<CR><LF>

Note 1:

The position status flag will be set to "V" (data invalid) until the receiver is navigating. At that time, the flag is changed to "A" (data valid) and the information provided in the RMC message will reflect a navigation solution.

Note 2:

Easterly variation (E) subtracts from True course.

Westerly variation (W) adds to True course.

Sample Message:

```
$GPRMC,185203,A,3339.7332,N,11751.7598,W,0.000,121.7,160496,13.8,E*55
```


เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้