

ระบบติดตามยานพาหนะด้วย GPS แสดงผลผ่าน GPRS

VEHICLE TRACKING SYSTEM BY GPS DISPLAYED BY GPRS



T119251



เลขหมู่.....
เลขทะเบียน.....
วัน,เดือน,ปี.....

119251

6 ส.ค. 2554

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

ปริญญานิพนธ์เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต

สาขาวิชาวิศวกรรมโทรคมนาคม

คณะวิศวกรรมศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2553

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ระบบติดตามยานพาหนะด้วย GPS แสดงผลผ่าน GPRS
VEHICLE TRACKING SYSTEM BY GPS DISPLAYED BY GPRS



โครงการนี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต
สาขาวิชาวิศวกรรมโทรคมนาคม
คณะวิศวกรรมศาสตร์
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
ปีการศึกษา 2553

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับนักเรียนเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปเผยแพร่เป็นการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มี
ผ่านการตรวจเรียบร้อยแล้ว (ลงชื่อ).....ผู้ตรวจ
ผ่านการตรวจรูปเล่มแล้ว (ลงชื่อ).....ผู้ตรวจ

ปริญญานิพนธ์ปีการศึกษา 2553

สาขาวิชาวิศวกรรมโทรคมนาคม

คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เรื่อง ระบบติดตามยานพาหนะด้วย GPS แสดงผลผ่าน GPRS

VEHICLE TRACKING SYSTEM BY GPS DISPLAYED BY GPRS

ผู้จัดทำ

1. นางสาว สุภาวดี ทิพย์กระโทก 50011746

2. นางสาวสุรางกุล กนิษฐกุล 50011768

.....
(ดร. สิริภพ ตู้ประกาย)

.....
(รศ.ดร. กอบชัย เดชหาญ)



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ระบบติดตามยานพาหนะด้วย GPS แสดงผลผ่าน GPRS

VEHICLE TRACKING SYSTEM BY GPS DISPLAYED BY GPRS

โดย	นางสาว สุภาวดี	ทิพย์กระโทก	50011746
	นางสาว สุรางกุล	กนิษฐกุล	50011768

อาจารย์ที่ปรึกษา ดร.สิรภพ ตู๊ประกาย

อาจารย์ที่ปรึกษาร่วม รศ.ดร.กอบชัย เดชหาญ

บทคัดย่อ

ในปัจจุบันได้มีการนำเครื่อง GPS มาประยุกต์ใช้งานอย่างแพร่หลาย โดยนำเครื่อง GPS มาใช้เป็นระบบนำร่อง สำหรับติดตั้งในรถยนต์ เพื่อให้สามารถทราบค่าตัวเลขพิกัดตำแหน่ง ทิศทางการเดินทาง และสามารถกำหนดระยะเวลาในการเดินทาง โดยจะสามารถแสดงข้อมูลผ่านระบบ GPRS และเพื่อเป็นประโยชน์ในความก้าวหน้าทางเทคโนโลยีอาจนำไปสู่การพัฒนาต่อไป

ABSTRACT

In present time, people use GPS to apply for using GPS become tracking system installed in vehicle system. Users can know the vehicle position and trip time. So GPS will display data by GPRS system. They are used for the advance technology, we should develop this project for users' comfortable.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กิตติกรรมประกาศ

ความดีทั้งหมด ขอมอบให้แก่ บิดา มารดา และผู้มีพระคุณทุกท่าน ที่คอยเป็นกำลังใจและเป็นทุกสิ่งทุกอย่างในชีวิตของข้าพเจ้า

โครงการ project เล่มนี้จะสำเร็จลงไม่ได้ถ้าหากไม่ได้คำแนะนำและคำปรึกษาที่เป็นประโยชน์ ขอขอบคุณ ดร.สิริภพ ตู้ประกาย เป็นอย่างมากที่ให้คำแนะนำ คำปรึกษาและให้ความช่วยเหลือในการทำ project ขอขอบคุณหนังสือของคุณพิชญ จุลศิริ เรื่อง ครบเครื่องเรื่อง GPS ที่เป็นส่วนช่วยอย่างมากในการทำโครงการ ขอขอบคุณอาจารย์ทุกท่านที่ได้ค่อยๆอบรมสั่งสอนและให้คำปรึกษา

ขอบคุณห้องสมุดที่เป็นแหล่งศึกษาหาความรู้ และเพื่อนๆทุกคนที่คอยให้ความช่วย และให้กำลังใจตลอดมาจนทำให้โครงการสำเร็จลุล่วงไปด้วยดี คณะผู้จัดทำขอขอบคุณทุกท่านมา ณ โอกาสนี้ด้วย



นางสาวสุภาวดี

นางสาวสุรางกุล

ทิพย์กระโทก

กนิษฐกุล

คณะผู้จัดทำ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ

	หน้า
กิตติกรรมประกาศ	I
บทกัดย่อ	II
สารบัญ	III
สารบัญรูปภาพ	V
สารบัญตาราง	VII

บทที่ 1 บทนำ

1.1	ความเป็นมาและความสำคัญของปัญหา	1
1.2	วัตถุประสงค์	3
1.3	ขอบเขตของการดำเนินการของโครงการ	3

บทที่ 2 ทฤษฎีและหลักการ

2.1	GPS คืออะไร	4
2.2	องค์ประกอบที่สำคัญของ GPS	5
2.2.1	ส่วนอวกาศ (Space Segment)	5
2.2.2	สถานีควบคุม (Control Station Segment)	6
2.2.3	ส่วนผู้ใช้งาน (User Segment)	6
2.3	ส่วนประกอบของเครื่องรับสัญญาณดาวเทียม GPS	6
2.4	การทำงานของ GPS	7
2.4.1	การรับสัญญาณจากดาวเทียมเพื่อให้ได้ตำแหน่ง	7
2.4.2	การวัดระยะจากดาวเทียม	7
2.4.3	การได้เวลาที่ถูกต้อง	8
2.4.4	ต้องรู้ตำแหน่งของดาวเทียมก่อน	9
2.4.5	ความช้าของสัญญาณในการเดินทางผ่านชั้นบรรยากาศ	10
2.5	รหัสสุ่มเทียม (Pseudo Random Code)	12
2.6	การวัดด้วยวิธีอนุพันธ์ (Differential)	13
2.7	ประเภทเครื่องรับสัญญาณ GPS	14
2.8	มาตรฐาน NMEA และ โปรโตคอลที่ใช้ในการสื่อสารของจีพีเอส	16
2.8.1	มาตร NMEA (NMEA Standard)	16
2.8.2	การอินเตอร์เฟสทางไฟฟ้า	17
2.8.3	มาตรฐาน NMEA-0183	17

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ (ต่อ)

	หน้า
2.8.4 รูปแบบประโยค NMEA	17
2.9 จีพีอาร์เอส(GPRS)	21
2.10 เอสคิวแอล (SQL)	22
2.11 มายเอสคิวแอล (MySQL)	23
2.12 พีเอชพี (PHP)	24
2.13 จาواسคริปต์ (JavaScript)	26
บทที่ 3 หลักการออกแบบและการสร้าง	28
3.1 โครงสร้างการทำงานของระบบ	28
3.2 ส่วนของเครื่องรับสัญญาณ	29
3.2.1 เครื่องรับระบบพิกัดตำแหน่ง (GPS Receiver)	29
3.2.2 คุณสมบัติของ SIM508	30
3.3 ส่วนของไมโครคอนโทรลเลอร์	30
3.4 ส่วนของการส่งข้อมูลผ่าน GPRS module	35
3.5 SERVER หรือฐานข้อมูล	36
บทที่ 4 การทดลองและผลการทดลอง	38
4.1 การทดลองส่วน GPS Module	38
4.2 การทดลองเปรียบเทียบการรับสัญญาณดาวเทียมด้วยเครื่องรับจีพีเอส	39
4.3การทดลองการกรองข้อมูลด้วย ARM7	40
4.4การใช้คำสั่ง AT Commandควบคุมโมดูลจีพีอาร์เอส	41
4.5การสร้างเซิร์ฟเวอร์และฐานข้อมูล	42
4.6 ทดลองโดยนำอุปกรณ์ใส่บนรถยนต์แล้วเคลื่อนที่	43
บทที่ 5 สรุปผลและข้อเสนอแนะ	
5.1 สรุปผล	48
5.2 ข้อเสนอแนะ	48
บรรณานุกรม	49
ภาคผนวก	51

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูป

	หน้า
รูปที่1.1 Chronometer by John Harrison	1
รูปที่1.2 รูปดาวเทียมสปุตนิก1	2
รูปที่2.1 ดาวเทียมทรานซิท 1	4
รูปที่2.2 ตำแหน่งและการโคจรของดาวเทียม GPS รอบโลก	5
รูปที่2.3 การโคจรของดาวเทียม GPS รอบโลก	6
รูปที่2.4 ส่วนประกอบของระบบดาวเทียม GPS	7
รูปที่2.5 การทำงานของระบบ GPS	9
รูปที่3.1 บล็อกกรวมของระบบ	28
รูปที่3.2 โครงสร้างการทำงานของระบบ	29
รูปที่3.3 บล็อกการรับสัญญาณ GPS	29
รูปที่3.4 เครื่องรับ GPS และ GPRS รุ่น SIM508_EVB	30
รูปที่3.5 บล็อกการทำงานส่วนไมโครคอนโทรลเลอร์	30
รูปที่3.6 แผนผังระบบการคัดกรองข้อมูลจากสัญญาณจีพีเอส	31
รูปที่3.7 ลักษณะของบอร์ด CP-JR ARM7 USB-LPC2148	34
รูปที่3.8 บล็อกการส่งข้อมูลผ่าน GPRS ไปยัง SERVER	35
รูปที่3.9 GPRS module	35
รูปที่3.10 บล็อกการทำงานรวมของฐานข้อมูล	36
รูปที่3.11 บล็อกการทำงานของฐานข้อมูล	36
รูปที่4.1 รูปเครื่อง GPS Module	38
รูปที่4.2 ข้อมูลที่ส่งออกมาจากเครื่องรับ GPS เมื่อเชื่อมต่อกับคอมพิวเตอร์ผ่านทางพอร์ตอนุกรม	38
รูปที่4.3 แสดงผลเทียบระหว่างสัญญาณขาเข้า6Vกับสัญญาณขาออก 5V	41
รูปที่4.4 ข้อมูลที่ยังไม่ได้ทำการกรองจากไมโครคอนโทรลเลอร์ARM7	41
รูปที่4.5 ข้อมูลที่ทำการกรองจากไมโครคอนโทรลเลอร์ARM7	42
รูปที่4.6 คำสั่ง AT Command	42
รูปที่4.7 เว็บบนการสมัครเซิร์ฟเวอร์	43
รูปที่4.8 เพจแสดงแผนที่รถยนต์สองคัน	44
รูปที่4.9 แผนที่แบบดาวเทียมแสดงเส้นทางที่รถเดินทางผ่าน	44
รูปที่4.10 แผนที่แบบสองมิติแสดงเส้นทางที่รถเดินทางผ่าน	45

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูป(ต่อ)

	หน้า
รูปที่4.11 แสดงข้อมูลในฐานข้อมูลตาราง GPSของรถยนต์คันที่ 1	45
รูปที่4.12 แสดงข้อมูลในฐานข้อมูลตาราง GPSของรถยนต์คันที่ 2	46
รูปที่4.13 แสดงข้อมูลต่างๆ ใน server	46
รูปที่4.14 อุปกรณ์รวมของระบบติดตามรถยนต์ด้วยจีพีเอส	47



สารบัญตาราง

หน้า

ตารางที่ 4.1 ผลการทดลองรับสัญญาณจีพีเอสในแต่ละวัน

39



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 1

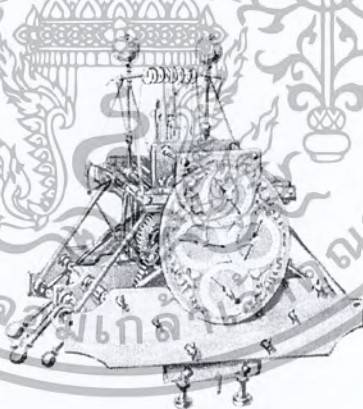
บทนำ

1.1 ความเป็นมาและความสำคัญของปัญหา

ตั้งแต่ในอดีตมนุษย์เราก็มีความพยายามที่จะสร้างเครื่องมือเครื่องมือ เพื่อบอกให้ได้ว่าเรากำลังอยู่ที่ใด เพื่อป้องกันการหลงทางและสามารถกลับไปยังจุดเดิมได้อย่างถูกต้อง ซึ่งในการเดินเรือสมัยแรก ๆ ก็มีการใช้ดวงดาวเป็นการบอกตำแหน่งและทิศทาง

ต่อมาเมื่อเทคโนโลยีทันสมัยมากขึ้น ก็ได้มีการคิดค้นประดิษฐ์เข็มทิศและเครื่องวัดระยะทางหาเส้นรุ้งและเส้นแวง (Sextant) ขึ้นมาโดยเข็มทิศจะชี้ไปทางเหนือเสมอฉะนั้นไม่ว่า เราจะไม่รู้ตำแหน่งของเรา แต่เราจะยังสามารรู้ทิศทางที่กำลังเดินทางไปได้ ส่วนเครื่องวัดระยะทางหาเส้นรุ้งและเส้นแวง นั้นจะช่วยในการวัดมุมระหว่างดวงดาวกับพื้นดิน ในยุคแรก ๆ นั้นเครื่องมือนี้จะใช้ในการเดินเรือและสามารถบอกได้แต่เส้นรุ้งเท่านั้น ไม่สามารถบอกเส้นแวงได้

ต่อมาในศตวรรษที่ 17 ประเทศอังกฤษก็ได้ตั้งกลุ่มนักวิทยาศาสตร์เพื่อทำการสร้าง เครื่องมือเพื่อหาเส้นแวงให้ได้ ซึ่งกลุ่มที่ตั้งขึ้นมาถูกเรียกว่า Board of Longitude โดยมีรางวัลให้กับผู้ที่สามารถสร้างเครื่องมือที่ใช้หาเส้นแวงได้ ซึ่งในปี ค.ศ.1761 John Harrison ได้พัฒนาเครื่องมือที่ใช้หาเส้นแวงได้ ซึ่งเรียกว่า ไชรโนมิเตอร์ (Chronometer) ซึ่งต่อมาก็มีการใช้เครื่องมือ Sextant และ ไชรโนมิเตอร์ ร่วมกันในการเดินทางอย่างแพร่หลาย



รูปที่ 1.1 Chronometer by John Harrison[1]

ในช่วงต้นศตวรรษที่ 20 ได้มีการพัฒนาระบบการส่งสัญญาณวิทยุมาใช้งานกันมากขึ้น จนกระทั่งได้มีการนำมาใช้กันอย่างแพร่หลายในสมัยสงครามโลกครั้งที่ 2 โดยทั้งเรือและเครื่องบิน จะใช้ระบบการรับ-ส่งสัญญาณวิทยุจากสถานีภาคพื้นดินเป็นตัวนำทางการส่งสัญญาณวิทยุนั้น จะสามารถส่งได้ทั้งแบบความถี่สูงและความถี่ต่ำ แต่ข้อเสียก็คือหากส่งสัญญาณ ในช่วงความถี่สูงจะสามารถรับ-ส่งข้อมูลได้อย่างถูกต้องแต่ครอบคลุมได้เพียงพื้นที่จำกัด ส่วนการรับ-ส่งสัญญาณในช่วงความถี่ต่ำสามารถครอบคลุมพื้นที่ได้กว้างไกลกว่า แต่ความถูกต้องต่ำกว่า

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ในศตวรรษที่ 20 ดาวเทียมสปุตนิก (Sputnik) ของประเทศรัสเซียได้ถูกส่งออกสู่อวกาศเมื่อวันที่ 4 ตุลาคม ค.ศ.1957 และทำให้เราเริ่มตระหนักกันว่าเราสามารถใช้ดาวเทียมในการนำทาง ได้เช่นเดียวกับดวงดาวบนท้องฟ้าโดยนักวิจัยจากสถาบัน MIT ได้ติดตามวิถีการโคจรของ ดาวเทียมสปุตนิกและได้สังเกตเห็นว่าสัญญาณวิทยุจากดาวเทียมสปุตนิกจะสูงขึ้น เมื่อดาวเทียมโคจรเข้ามาใกล้และต่ำลงเมื่อดาวเทียมโคจรห่างออกไปจากข้อเท็จจริง ดังกล่าวที่ว่าเราสามารถจะติดตามตำแหน่งของดาวเทียมในขณะที่โคจรรอบโลกได้จาก ภาคพื้นดินนั้นจึงเป็นที่มาของสมมุติฐานที่ว่าในทางกลับกันเราก็น่าจะสามารถติดตาม หรือระบุตำแหน่งของวัตถุใดๆ บนพื้นโลกโดยใช้สัญญาณวิทยุจากดาวเทียมได้เช่นกัน

ต่อมาทางประเทศสหรัฐอเมริกาก็ได้มีการพัฒนาดาวเทียมนำร่องออกสู่อวกาศเช่นกัน โดยทางอเมริกาเรียกระบบนี้ว่า ทรานซิท(Transit) ซึ่งประกอบไปด้วยดาวเทียม 6 ดวงโคจรรอบโลกผ่านขั้วโลก ที่ความสูงประมาณ 1,100 กิโลเมตร โดยใช้สำหรับหาตำแหน่งของเรือเดินสมุทร และเครื่องบิน โดยระบบนี้รัฐบาลอเมริกาอนุญาตให้เอกชนบางรายใช้งานสำรวจเท่านั้น โดยยังไม่เปิด ให้บุคคลทั่วไปใช้งานแต่ระบบนี้ก็ใช้งานกันได้ไม่นานนักเนื่องจากการส่งสัญญาณช้าและมีความถูกต้องต่ำจึงได้เริ่มมีการพัฒนาระบบ จีพีเอส(GPS)เพื่อให้มีการบอกตำแหน่ง ได้อย่างแม่นยำมากขึ้น โดยได้เริ่มมีการพัฒนาอย่างต่อเนื่องจากหน่วยงานที่เกี่ยวข้องส่งผลทำให้ระบบจีพีเอส ที่สมบูรณ์ได้ถูกใช้งานเต็มรูปแบบจากดาวเทียม 24 ดวง ในกลางปี 1990



รูปที่ 1.2 รูปดาวเทียมสปุตนิก1 [2]

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1.2 วัตถุประสงค์

- 1) เพื่อศึกษาระบบการทำงานของ จีพีเอส/จีพีอาร์เอส
- 2) เพื่อศึกษาไมโครคอนโทรลเลอร์อาร์ม7(Microcontroller ARM7) เพื่อประยุกต์ใช้ใน ระบบจีพีเอส/จีพีอาร์เอส
- 3) เพื่อศึกษา อาร์เอส-232(RS-232) และมาตรฐานเอ็นเอ็มอีเอ (NMEA)
- 4) เพื่อศึกษาการส่งข้อมูลแบบจีพีอาร์เอส โดยใช้คำสั่งเอทีคอมมาน(AT Command)
- 5) เพื่อสามารถใช้ภาษาพีเอสพี (PHP) และมายเอสคิวเอล (MySQL)ในการจัดการฐานข้อมูลได้ และสามารถเชื่อมต่อกับ โปรแกรมแผนที่กูเกิ้ลแมพ(Google map)ได้

1.3 ขอบเขตของการดำเนินการของโครงการ

- 1) สามารถรับสัญญาณดาวเทียมด้วยจีพีเอส
- 2) สามารถแสดงผลผ่านระบบอินเทอร์เน็ตได้
- 3) สามารถกรองสัญญาณเพื่อตัดข้อมูลอาบรรทัดที่เราจะใช้
- 4) สามารถสร้างฐานข้อมูลด้วยภาษาพีเอสพีกับเอสคิวเอล (SQL) ได้
- 5) สามารถส่งสัญญาณด้วยจีพีเอสระหว่างจีพีเอสและฐานข้อมูล
- 6) สามารถระบุตำแหน่งบนแผนที่แสดงผลผ่านระบบอินเทอร์เน็ต
- 7) สามารถเพิ่มจำนวนรถยนต์ที่จะใช้ติดตามได้มากกว่าหนึ่งคัน

บทที่ 2

ทฤษฎีและหลักการ

2.1 GPS คืออะไร

ระบบกำหนดตำแหน่งบนโลกหรือจีพีเอส (Global Positioning System - GPS) หมายถึง ระบบบอกตำแหน่งบนพื้นผิวโลก โดยอาศัยการคำนวณพิกัด ยูทีเอ็ม (UTM) จากสัญญาณนาฬิกาที่ส่งมาจากดาวเทียมที่โคจรรอบโลก ซึ่งมีตำแหน่งที่แน่นอน ระบบนี้สามารถบอกตำแหน่ง ณ จุดที่สามารถรับสัญญาณได้ทั่วโลก โดยเครื่องรับสัญญาณจีพีเอส รุ่นใหม่ๆ จะสามารถคำนวณความเร็วและทิศทางนำมาใช้ร่วมกับโปรแกรมแผนที่ เพื่อใช้ในการนำทางได้

แนวคิดในการพัฒนาระบบจีพีเอส เริ่มต้นตั้งแต่ปีค.ศ.1957 เมื่อนักวิทยาศาสตร์ของสหรัฐอเมริกา นำโดย Dr. Richard B. Kershner ได้ติดตามการส่งดาวเทียมสปุตนิกของโซเวียต และพบปรากฏการณ์คอปเปิลอร์ของคลื่นวิทยุที่ส่งมาจากดาวเทียม พวกเขาพบว่าหากทราบตำแหน่งที่แน่นอนบนพื้นผิวโลก ก็สามารถระบุตำแหน่งของดาวเทียมได้จากการตรวจวัดคอปเปิลอร์ และหากทราบตำแหน่งที่แน่นอนของดาวเทียม ก็สามารถระบุตำแหน่งบนพื้นผิวโลกได้ ในทางกลับกัน

กองทัพเรือสหรัฐได้ทดลองระบบนำทางด้วยดาวเทียม ชื่อ TRANSIT เป็นครั้งแรกเมื่อ ค.ศ. 1960 ประกอบด้วยดาวเทียมจำนวน 5 ดวง ส่วนดาวเทียมที่ใช้ในระบบจีพีเอส (GPS Block-I) ส่งขึ้นทดลองเป็นครั้งแรกเมื่อ ค.ศ. 1978 เพื่อใช้ในการทหาร



รูปที่ 2.1 ดาวเทียมทรานซิท 1[2]

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เมื่อ ค.ศ. 1983 หลังจากเกิดเหตุการณ์โคเรียนแอร์ไลน์ เที่ยวบินที่ 007 ของเกาหลีใต้ บินพลัดหลงเข้าไปในน่านฟ้าของสหภาพโซเวียตและถูกยิงตก ผู้โดยสาร 269 คนเสียชีวิตทั้งหมด ประธานาธิบดีโรนัลด์ เรแกนได้ประกาศว่า เมื่อพัฒนาระบบจีพีเอสแล้วเสร็จ จะอนุญาตให้ประชาชนทั่วไปใช้งานได้

ดาวเทียมจีพีเอส เป็นดาวเทียมที่มีวงโคจรต่ำ ที่ระดับความสูง 11,000 ไมล์จากพื้นโลก ใช้การขึ้นยืนตำแหน่งโดยอาศัยพิกัดจากดาวเทียม 3 หรือ 4 ดวง ดาวเทียมจะโคจรรอบโลกเป็นเวลา 12 ชั่วโมงต่อหนึ่งรอบ ที่ความเร็ว 4 กิโลเมตร/วินาที การโคจรแต่ละรอบนั้นสามารถได้เป็น 6 ระนาบๆละ 4 ดวง ทำมุม 55 องศา โดยทั้งระบบจะต้องมีดาวเทียม 24 ดวง หรือมากกว่า เพื่อให้สามารถขึ้นยืนตำแหน่งได้ครอบคลุมทุกจุดบนผิวโลก ปัจจุบัน เป็นดาวเทียม GPS Block-II

2.2 องค์ประกอบที่สำคัญของ GPS

ระบบที่ทำให้จีพีเอสทำงานได้ สามารถแบ่งได้ออกเป็น 3 ส่วนหลัก คือ ส่วนอวกาศ ส่วนควบคุม และส่วนผู้ใช้งาน

2.2.1 ส่วนอวกาศ (Space Segment)

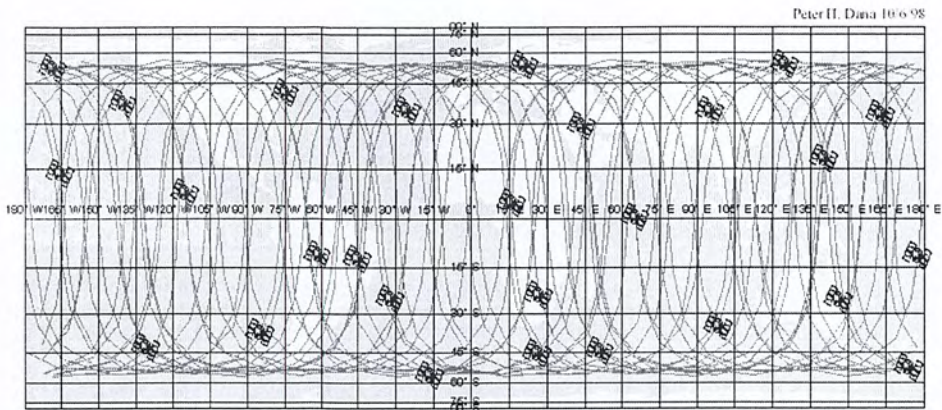
ในระบบดาวเทียมจีพีเอสจะประกอบด้วยดาวเทียมทั้งหมด 24 ดวง โดยดาวเทียมจำนวน 21 ดวง จะใช้ในการบอกค่าพิกัด ส่วนที่เหลือ 3 ดวง จะสำรองเอาไว้ ดาวเทียมทั้ง 24 ดวงนี้จะมียังโคจรอยู่ 6 วงโคจรด้วยกัน โดยแบ่งจำนวนดาวเทียมวงโคจรละ 4 ดวง และมีรัศมีวงโคจรสูงจากพื้นโลกประมาณ 20,200 กิโลเมตร (12,600 ไมล์) วงโคจรทั้ง 6 จะเอียงทำมุมกับเส้นศูนย์สูตร (Equator) เป็นมุม 55 องศา ในลักษณะสานกันคล้ายลูกตะกร้อ ดาวเทียมแต่ละดวงจะใช้เวลาในการโคจรครบรอบ 12 ชั่วโมง นั่น คือ คาบของการโคจรเป็น 12 ชั่วโมง/รอบ ความถี่ที่ใช้ในการบอกตำแหน่งค่าพิกัดของดาวเทียมแต่ละดวงมี 2 ความถี่ คือ ความถี่ L1:1,575.42 MHz และ ความถี่ L2:1,227.60 MHz



GPS Nominal Constellation
24 Satellites in 6 Orbital Planes
4 Satellites in each Plane
20,200 km Altitudes, 55 Degree Inclination

รูปที่ 2.2 ตำแหน่งและการโคจรของดาวเทียม GPS รอบโลก [3]

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



Global Positioning System Satellites and Orbits
for 27 Operational Satellites on September 29, 1998
Satellite Positions at 00:00:00 9/29/98 with 24 hours (2 orbits) of Ground Tracks to 00:00:00 9/30/98

รูปที่ 2.3 การโคจรของดาวเทียม GPS รอบโลก [3]

2.2.2 สถานีควบคุม (Control Station Segment)

ในส่วนของสถานีควบคุมจะประกอบด้วย 5 สถานีย่อย (Monitor Station) ตั้งอยู่ที่เมือง Diego Garcia, Ascension Island, Kwajalein, และ Hawaii ส่วนสถานีควบคุมหลัก (Master Control Station) 1 สถานี ซึ่งเป็นศูนย์กลางการทำงานจากระบบดาวเทียมจีพีเอส ตั้งอยู่ที่เมือง Colorado Springs รัฐ Colorado สหรัฐอเมริกา สถานีควบคุมต่างๆ เหล่านี้มีหน้าที่คอยติดต่อสื่อสาร (Tracking) กับดาวเทียม ทำการคำนวณผล (Computation) เพื่อบอกตำแหน่งของดาวเทียมแต่ละดวงและส่งข้อมูลที่ได้ไปยังดาวเทียมอยู่ตลอดเวลา ทำให้ข้อมูลที่ได้เป็นข้อมูลที่ทันสมัยอยู่เสมอ

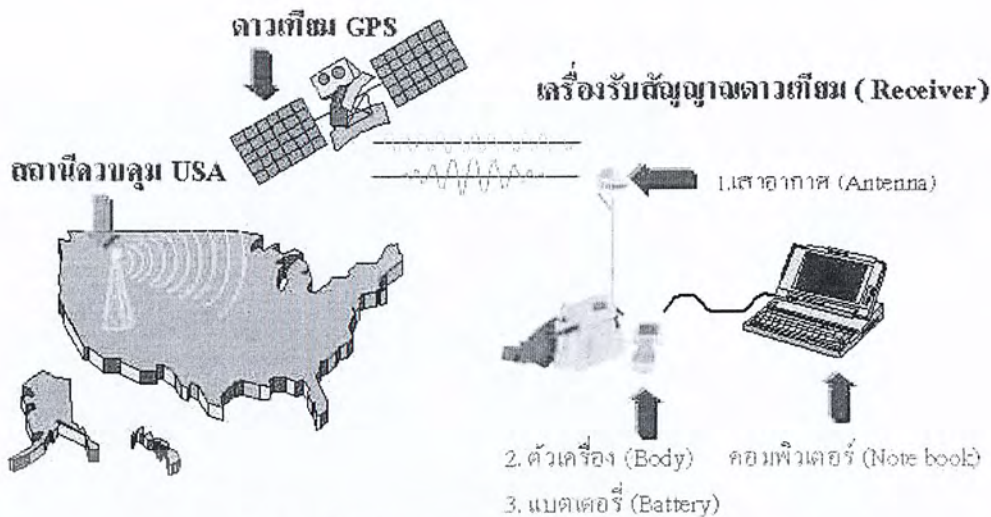
2.2.3 ส่วนผู้ใช้ (User Segment)

ผู้ใช้ประกอบด้วย 2 ส่วนใหญ่ๆ คือ ส่วนที่เกี่ยวข้องกับพลเรือน (Civilian) และส่วนที่เกี่ยวข้องกับทางทหาร (Military) ในส่วนของผู้ใช้จะมีหน้าที่พัฒนาเครื่องรับสัญญาณ (Receiver) ให้ทันสมัยและสะดวกแก่การใช้งาน สามารถที่จะใช้ได้ทุกแห่งในโลก และให้ค่าที่มีความถูกต้องสูง

2.3 ส่วนประกอบของเครื่องรับสัญญาณดาวเทียม GPS

โดยทั่วไปเครื่องรับสัญญาณดาวเทียม (Receiver) ประกอบด้วย 3 ส่วนหลัก คือ

1. ตัวเครื่อง (Body)
2. ส่วนให้พลังงาน (Power Supply)
3. ส่วนเสาอากาศ (Antenna)



รูปที่ 2.4 ส่วนประกอบของระบบดาวเทียม GPS [3]

2.4 การทำงานของ GPS

หลักการพื้นฐานของจีพีเอสเป็นเรื่องง่าย ๆ แต่อุปกรณ์ของเครื่องมือถูกสร้างขึ้นด้วยวิทยาการขั้นสูง การทำงานจีพีเอสแบ่งออกได้เป็น 5 ขั้นตอน คือ

1. การรับสัญญาณจากดาวเทียมโดยหลักการรูปสามเหลี่ยมระหว่างดาวเทียมกับเครื่องรับ
2. จีพีเอสวัดระยะโดยใช้เวลาเดินทางของคลื่นวิทยุ
3. ในดาวเทียมและเครื่องรับจำเป็นจะต้องมีนาฬิกาที่ละเอียดสูงมาก
4. นอกจากระยะทางแล้วจะต้องทราบตำแหน่งของดาวเทียมที่อยู่ในอวกาศด้วย
5. ในชั้นบรรยากาศไอโอโนสเฟียร์ (Ionosphere) และชั้นบรรยากาศโลก (Atmosphere) ความเร็วคลื่นวิทยุเดินทางได้ช้าลง จึงต้องทำการแก้ไขจุดนี้ด้วย

ขั้นตอนที่ 1 การรับสัญญาณจากดาวเทียมเพื่อให้ได้ตำแหน่ง

จีพีเอสจะต้องหาระยะระหว่างดาวเทียมกับเครื่องรับจีพีเอส ดาวเทียมจะเป็นเหมือนหมุดหลักฐานสำหรับการวัดระยะ สิ่งที่เราต้องรู้เพื่อใช้ในการคำนวณ คือ ตำแหน่งดาวเทียมดวงนั้นเพื่อให้ได้ระยะทางที่ต้องการ สมมติว่าเราอยู่ห่างจากดาวเทียม A 11,000 ไมล์ ขณะเดียวกันเราหาได้ว่าระยะจากดาวเทียม B 12,000 ไมล์ ดังนั้นตำแหน่งเรา จึงอยู่ที่ทรงกลมที่มีศูนย์กลางที่ดาวเทียม A กับดาวเทียม B รัศมี 11,000 ไมล์ และ 12,000 ไมล์ ตัดกัน ดังนั้น ถ้าเราได้ระยะจากดาวเทียมดวงที่ 3 ก็บอกตำแหน่งได้แน่นอนยิ่งขึ้น เช่น เรารู้ระยะจากดาวเทียม C 13,000 ไมล์ ก็จะบอกตำแหน่งที่ทรงกลมตัดกันอยู่เพียง 2 จุดเท่านั้น เราวัดหาระยะ ดาวเทียมดวงที่สี่ หรือจากการวัดระยะดาวเทียม 3 ดวงก็สามารถบอกตำแหน่งได้เลย เพราะ 2 ค่าที่ได้จะมีอยู่หนึ่งค่าที่ไม่เป็นจริง เพราะอยู่ห่างจากโลกเรามาก ในกรณีที่เรารู้ค่าความสูงของตำแหน่งที่วัดแน่นอน เช่น ในทะเล ก็ไม่จำเป็นที่จะต้องวัดดาวเทียมดวงที่ 4 เครื่องมือรับวัดบางเครื่องจะให้เลือกใช้วัดแบบ 2D คือ พิกัดราบอย่างเดียว แต่ถ้าต้องการได้ตำแหน่ง 3D ต้องวัดจากดาวเทียม 4 ดวง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ขั้นตอนที่ 2 การวัดระยะจากดาวเทียม

จากการที่จีพีเอส ต้องรู้ระยะทางจากเครื่องรับถึงดาวเทียมจึงต้องมีวิธีการหาระยะ วิธีการหาระยะใช้สมการง่าย ๆ คืออัตราความเร็วคูณด้วยเวลาตัวอย่าง เช่น ถ้ารถยนต์เคลื่อนที่ด้วยความเร็ว 60 กม./ชม. เวลา 2 ชม. รถยนต์จะเคลื่อนที่ได้เป็นระยะทางเท่าใด การคิดใช้ความเร็ว (60 กม./ชม.) ได้ระยะทาง (120 กม.) ระบบจีพีเอส ทำงานเพื่อหาว่าสัญญาณวิทยุที่ส่งมาจากดาวเทียมจนถึงเครื่องรับใช้เวลาเดินทางนานเท่าใด แล้วนำเวลาที่หาได้มาคำนวณระยะทาง คลื่นวิทยุเดินทางด้วยความเร็วแสงคือ 186,000 ไมล์ต่อวินาที ดังนั้นถ้าเรารู้เวลาแน่นอนที่ดาวเทียมเริ่มปล่อยสัญญาณวิทยุ และเวลาที่เรารับสัญญาณนั้นได้ ก็จะได้เวลาที่คลื่นวิทยุเดินทาง ทำเวลาเป็นวินาทีคูณกับ 186,000 ไมล์ ก็จะได้ระยะทางจากเครื่องรับถึงดาวเทียม เราต้องได้ระยะเช่นนี้ 3 ค่าจากดาวเทียมสามดวง จึงจะนำมาหาตำแหน่งได้ เมื่อเป็นเช่นนี้แน่นอนนาฬิกาจะต้องเป็นนาฬิกาที่ดีมาก เพราะเวลาที่วัดได้จะต้องน้อยมากเพราะแสงเดินทางเร็วมาก โดยปกติถ้าดาวเทียมดวงที่ส่งสัญญาณอยู่เหนือศีรษะเราพอที่เวลาที่คลื่นวิทยุจะใช้เวลาเดินทางถึงเราเพียง 0.06 วินาทีเท่านั้น

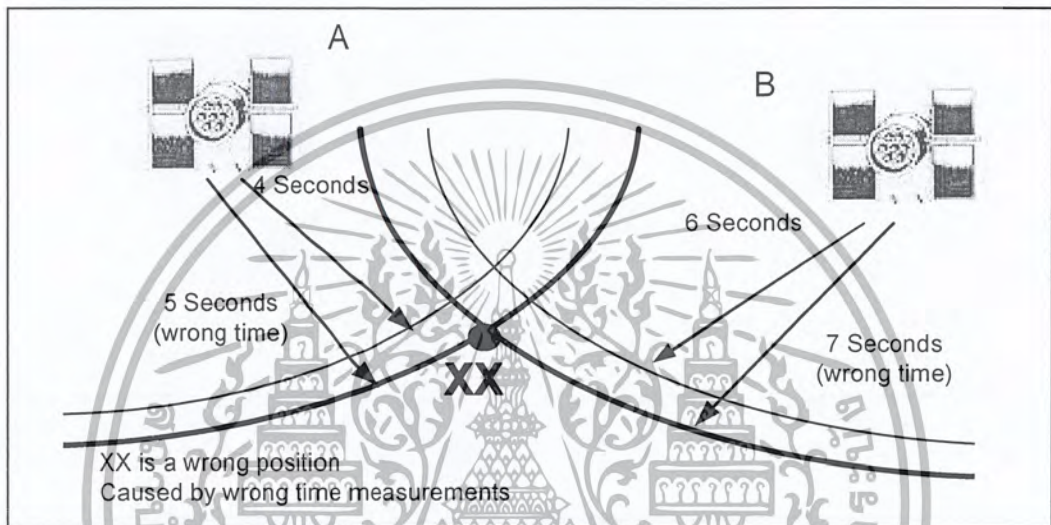
ด้วยเหตุนี้จีพีเอส จึงได้นำเอาวิวัฒนาการทางอิเล็กทรอนิกส์มาใช้เพื่อให้ได้ความถูกต้องของเวลาในระดับที่จีพีเอสต้องการ ต้องใช้นาฬิกาอิเล็กทรอนิกส์ที่มีราคาแพงมาก ซึ่งให้เวลาที่ละเอียดถูกต้องสูง แต่จีพีเอส จำเป็นต้องรู้เวลาที่ละเอียดยิ่งกว่ามาก นาฬิกาดาวเทียมจะอ่านเวลาได้เป็นนาโนเซกกัน หรือ 0.000000001 วินาที เรารู้เวลาที่สัญญาณเริ่มส่งจากดาวเทียมได้อย่างไร เคล็ดลับที่สำคัญในการหาเวลาการเดินทางของคลื่นวิทยุก็คือ ต้องรู้ว่าเวลาที่แน่นอนที่สัญญาณเริ่มออกเดินทางจากดาวเทียม ผู้ออกแบบเครื่องจีพีเอส ใช้หลักการจำลองแบบสัญญาณที่ส่งจากดาวเทียม และสัญญาณที่อยู่ในเครื่องรับให้เป็นแบบเดียวกัน ดังนั้นเครื่องทั้งสองจะต้องสร้างรหัสในเวลาตรงกัน (Pseudo Random Code) ดังนั้นสิ่งที่เราต้องกระทำก็คือการรอรหัสที่ดาวเทียมปล่อยออกมาและมองย้อนกลับ ไปว่าเครื่องของเราได้เริ่มสร้างรหัสที่มีรูปแบบเหมือนกันแล้วเป็นเวลานานเท่าใด เวลาที่แตกต่างกันก็คือเวลาที่คลื่นวิทยุใช้เดินทางมาถึงเครื่องรับ ข้อดีของการใช้รหัสที่ส่งเป็นชุดหรือตัวเลขหลายตัวเราสามารถเปรียบเทียบหาตรงเวลาได้ก็ได้ตามต้องการ ไม่จำเป็นต้องใช้เลขหนึ่งอย่างเดียว ใช้ตัวเลขคู่ใดก็ได้รหัส Pseudo Random ในจีพีเอส ไม่ใช่ตัวเลขตามตัวอย่างที่กล่าวมาทั้งในดาวเทียมและเครื่องรับจะสร้างชุดรหัสเชิงตัวเลขที่ซับซ้อน การที่ต้องสร้างให้ซับซ้อนก็เพื่อสามารถนำรหัสทั้งสองมาเปรียบเทียบกันได้ง่ายและไม่วุ่นวายและยังมีเหตุผลทางวิชาการประกอบด้วย รหัสซ้ำซ้อนนี้ทำให้มองเห็นเหมือนคลื่นวิทยุที่ต่อเนื่องกันยาว ๆ

ขั้นตอนที่ 3 การได้เวลาที่ถูกต้อง

แสงเดินทางด้วยความเร็ว 186,000 ไมล์/วินาที จะเกิดอะไร ถ้าเครื่องรับนับเวลาลดไป 1/100 วินาที ผลคือการวัดเราจะผิดไปถึง 1,860 ไมล์ และเราจะรู้ได้อย่างไรว่าเครื่องรับและดาวเทียมได้สร้างรหัสตรงเวลาเดียวกันหรือไม่ ปัญหานี้สามารถอธิบายได้คือ ในดาวเทียมใช้นาฬิกาอะตอม ซึ่งจะให้เวลาที่ถูกต้อง ในดาวเทียมแต่ละดวงจะมีนาฬิกาอะตอมนี้ติดตั้งอยู่ถึง 4 เครื่อง ทั้งนี้เพื่อให้แน่ใจว่าจะต้องใช้เวลา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากนาฬิกาเครื่องใดเครื่องหนึ่งอย่างแน่นอน นาฬิกาอะตอมไม่ได้เดินด้วยพลังงานอะตอม ที่ให้ชื่ออะตอม เพราะว่าใช้การวัดจังหวะจากอนุภาคของสารเฉพาะ เหมือนเครื่องเคาะจังหวะ อะตอมนี้จะให้เวลาที่แน่นอนและถูกต้องที่สุดที่มนุษย์เราได้ประดิษฐ์มา ดังนั้นถ้านาฬิกาบอกเวลาเที่ยง 12.00 น. ก็หมายถึงเวลาเที่ยง 12.00 จริง โชคดีที่มีวิธีให้หาเวลาได้ถูกต้องใช้งานได้ทั้งที่เครื่องรับจีพีเอส ดินาฬิกาที่มีความถูกต้องธรรมดาเท่านั้น และวิธีนั้นก็คือ จะต้องทำงานวัดระยะจากดาวเทียม สำหรับกรณีนี้เพิ่มอีกหนึ่งดวงเพื่อใช้ในการปรับแก้เวลาของเครื่องรับที่ไม่สมบูรณ์ ซึ่งเป็นเพียงแนวคิดที่ธรรมดาเป็นฐานข้อมูลของการทำงานของเครื่องจีพีเอส จะแสดงให้เห็นว่าการวัดระยะจากดาวเทียมเพิ่มอีกหนึ่งดวงช่วยได้อย่างไร



รูปที่ 2.5 การทำงานของระบบ GPS [3]

ขั้นตอนที่ 4 ต้องรู้ตำแหน่งของดาวเทียมก่อน

ตามที่กล่าวมาทั้งหมดเราสมมติว่ารู้ตำแหน่งของดาวเทียมมาแล้ว จึงสามารถสร้างรูปสามเหลี่ยมขึ้นมาได้ แต่เราจะรู้ตำแหน่งของดาวเทียมที่อยู่สูงถึง 11,000 ไมล์ ได้อย่างไร ความสูงเช่นนี้จะไม่ทำให้มีคลื่นรบกวนจากโลกไปรบกวนได้ วัตถุที่อยู่สูงขึ้นไปเช่นนี้ผ่านพ้นจากชั้นบรรยากาศของโลก หมายความว่าวงโคจรดาวเทียมรอบโลกแสดงได้ด้วยสมการคณิตศาสตร์ ธรรมดาเหมือนกับดวงจันทร์ที่หมุนรอบโลกเป็นเวลาล้าน ๆ ปี โดยไม่มีการเปลี่ยนแปลง ดาวเทียมจีพีเอส เดินตามวงโคจรตามแนวที่กำหนดไว้แน่นอน กองทัพอากาศสหรัฐมีหน้าที่นำดาวเทียมเข้าสู่วงโคจรตามแผนที่กำหนดไว้ในโครงการ และเนื่องจากในอวกาศว่างเปล่าไม่มีแรงเสียดทาน ดาวเทียมก็จะ โคจรอยู่ในวงที่แน่นอนตามกำหนด วงโคจรของดาวเทียมแต่ละดวงถูกกำหนดไว้ล่วงหน้าแล้ว และเครื่องรับจีพีเอส สามารถรับตารางดาวเทียม (Almanac) ไว้ในหน่วยความจำคอมพิวเตอร์ได้ ตารางดาวเทียมจะบอกได้ว่าในท้องฟ้าจะมีดาวเทียมดวงไหนขึ้นลงเวลาใดบ้าง มีการติดตามการโคจรของดาวเทียมทุกดวงอย่างสม่ำเสมอแน่นอน สมการวงโคจรของดาวเทียมจะต้องถูกต้องตามตัวเลขของมันเองอยู่แล้ว แต่เพื่อให้ทุกอย่างถูกต้องสมบูรณ์ กระทรวงกลาโหมสหรัฐจึง

ต้องทำการติดตามการโคจรของดาวเทียมทุกดวงอย่างสม่ำเสมอการที่ต้องติดตามดาวเทียมนี้เป็นเหตุผลหนึ่ง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่นิยมนำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ที่ทำให้ต้องสร้างดาวเทียมจีพีเอส ให้หมุนเร็วกว่าการหมุนของโลก ดาวเทียมหมุนรอบโลกทุก 12 ชั่วโมง และจะโคจรผ่านสถานีติดตามดาวเทียมของดีโอดี (DOD) วันละ 2 ครั้ง ซึ่งทำให้สถานีติดตามนี้สามารถวัด ความสูง ตำแหน่ง และความเร็วของดาวเทียมได้อย่างถูกต้อง ความแปรเปลี่ยนของวงโคจรที่สถานีติดตาม ค้นหา ก็คือ ค่าความคลาดเคลื่อนของอีฟิเมอร์ริส (Ephemeris Error) ปกติจะมีขนาดน้อยมากโดยเกิดจากแรงดึงดูดของดวงจันทร์และดวงอาทิตย์ และเกิดจากการแผ่รังสีดวงอาทิตย์ที่มีต่อดาวเทียม เมื่อดีโอดี วัดหา ตำแหน่งดาวเทียมได้ ค่าตำแหน่งใหม่นี้ก็จะถูกส่งกลับเข้าไปบันทึกไว้ในดาวเทียมอีก ดาวเทียมดวงนั้นก็ จะส่งค่าแก่นี้พร้อมกับข่าวสารเวลาให้เครื่องรับ ข้อสำคัญคือ ดาวเทียมจีพีเอส ไม่เพียงแต่ส่งรหัส Pseudo Random สำหรับการหาเวลาเท่านั้น แต่ส่งข่าวสารข้อมูลเกี่ยวกับตำแหน่งของวงโคจรและค่าความสมบูรณ์ ของระบบด้วยเครื่องรับจีพีเอส ใช้ข่าวสารข้อมูลนี้ควบคู่กับข้อมูลตารางดาวเทียมในเครื่องรับ ในการ คำนวณตำแหน่งที่ถูกต้องของดาวเทียม

ขั้นตอนที่ 5 ความช้าของสัญญาณในการเดินทางผ่านชั้นบรรยากาศ

เราทราบแล้วว่าในระบบจีพีเอส ได้จัดทำทุกส่วนให้ได้รับความถูกต้อง เช่น ใช้นาฬิกาอะตอมใน ดาวเทียม และมีการวัดระยะเพิ่มขึ้นอีกหนึ่งระยะเพื่อใช้จัดความคลาดเคลื่อนของนาฬิกาในเครื่องรับ และ ข้อความส่งจากดาวเทียมจะมีรายงานค่าปรับแก้วงโคจรทุกนาทีก แต่ก็มีสาเหตุของความคลาดเคลื่อนอีก สองสามประการที่กำจัดได้ยาก ถ้าความคลาดเคลื่อนที่เกิดขึ้นได้ชัดที่สุด เกิดจากบรรยากาศชั้นไอโอโนสเฟียร์ ซึ่งเป็นชั้นของอนุภาคประจุไฟฟ้า อยู่สูงจากโลกระหว่าง 80-120 ไมล์ อนุภาคเหล่านี้มีผลต่อความเร็ว ของแสง และความเร็วของสัญญาณวิทยุจากดาวเทียมจีพีเอส เช่นกัน บางคนอาจคิดว่าความเร็วของแสง เป็นค่าคงที่ที่อยู่ตลอดเวลา แต่แสงเดินทางด้วยความเร็วคงที่เมื่ออยู่ในสุญญากาศ ซึ่งอยู่ในชั้นอวกาศที่สูง มาก แต่เมื่อแสงหรือสัญญาณวิทยุเดินทางผ่านตัวกลางที่มีความหนาแน่น เช่น ชั้นที่มีอนุภาคประจุไฟฟ้าที่ หนาหลายไมล์ย่อมทำให้ความเร็วลดลงบ้าง และการที่คลื่นวิทยุเดินทางช้าลงนี้จะทำให้ระยะที่ได้ไม่ ถูกต้อง ถ้าหากว่าใช้ความเร็วของแสงคงที่ มีสองวิธีที่จะใช้ลดความคลาดเคลื่อนของระยะทางจากการที่ สัญญาณเดินทางช้า ทางที่หนึ่งเราต้องรู้ค่าความแปรเปลี่ยนเฉลี่ยรายวันตามสภาพบรรยากาศชั้นไอโอเฟียร์ ความหนาแน่นปานกลางจึงสามารถนำมาเป็นค่าแก้กับทุกค่าที่วัดได้ ซึ่งได้ความถูกต้องสูงขึ้น แต่สภาพ อากาศตามความเป็นจริงจะไม่คงที่ปานกลางตลอดเวลา ดังนั้น การนำค่าเฉลี่ยมาใช้จะไม่ถูกต้องทั้งหมด อีกทางหนึ่งโดยการวัดหาค่าความเร็วของสัญญาณวิทยุ โดยการวัดความเร็วสัมพัทธ์ของสัญญาณ สองแบบที่ส่งมาจากดาวเทียมพร้อมกัน วิธีการนี้เป็นแขนงวิชาฟิสิกส์ที่ผู้ศึกษาจำนวนน้อยมีแนวคิด พื้นฐานดังนี้ เมื่อแสงผ่านชั้นบรรยากาศไอโอโนสเฟียร์จะเดินทางช้าลงเป็นอัตราส่วนกลับกับความถี่ของ สัญญาณยกกำลังสอง ถ้าความถี่ยิ่งต่ำการเดินทางจะยิ่งช้าลง วิธีนี้ใช้การเปรียบเทียบเวลาที่สัญญาณจากจีพี เอส ที่มีความถี่ต่างกันเดินทางถึงเครื่องรับก็จะได้ค่าเวลาที่คลื่นเดินทางเข้าไป วิธีการแก้แบบนี้มักใช้กับ เครื่องจีพีเอส ที่มีความละเอียดถูกต้องสูง ที่เรียกว่าเครื่องรับความถี่คู่ (Dual Frequency) จะช่วยขจัดค่า ความคลาดเคลื่อนจากไอโอโนสเฟียร์ได้เป็นส่วนใหญ่ หลังจากเดินทางผ่านบรรยากาศชั้นไอโอโนสเฟียร์ซึ่งอยู่สูงกว่า ก็ถึงชั้นบรรยากาศโลกที่มีละอองไอน้ำในอากาศซึ่งมีผลต่อความเร็วของสัญญาณเช่นกัน ดังนั้นค่าความคลาดเคลื่อนจึงเกิดเช่นเดียวกับที่เกิดในบรรยากาศชั้นไอโอโนสเฟียร์ แต่ค่าคลาดเคลื่อน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ดังกล่าวยังไม่มีการปรับแก้ได้ซึ่งรวมอยู่ในค่าความคลาดเคลื่อนรวมของการบอกตำแหน่งโดยเครื่องจีพีเอส เป็นระยะประมาณ 25 เมตรเท่านั้น

ยังมีสาเหตุอื่นที่ทำให้เกิดความคลาดเคลื่อนได้อีก ได้แก่ ความเที่ยงของนาฬิกา นาฬิกาอะตอมในดาวเทียมที่ถูกต้องมากแล้ว ก็ยังมีความคลาดเคลื่อนได้เหมือนกับคิโอดี ติดตามความเที่ยงของนาฬิกาอะตอมและทำการปรับแก้แต่ก็ยังคงมีความคลาดเคลื่อนอยู่บ้างเล็กน้อย ซึ่งอยู่ในความคลาดเคลื่อนรวมของเครื่องจีพีเอส เช่นเดียวกัน เครื่องรับที่ตั้งอยู่บนพื้นดินก็มีส่วนทำให้เกิดผลคลาดได้เช่นกัน การคำนวณทางคณิตศาสตร์ในโปรแกรมของเครื่องทำให้ต้องเลือกตัดค่าสังเกตบางค่าทิ้ง บางครั้งเมื่อถูกรบกวนด้วยคลื่นวิทยุอาจทำให้รหัส Pseudo Random มีลักษณะผิดเพี้ยนทำให้โปรแกรมคอมพิวเตอร์ทำงานไม่ถูกต้อง ความคลาดเคลื่อนอาจมีขนาดเล็กมากหรือขนาดใหญ่มากก็ได้ ค่าที่ใหญ่ก็สามารถรู้ได้ง่ายกว่าเพราะเห็นได้ชัด แต่ถ้ามีค่าเล็กน้อยก็เป็นการยากที่จะหาได้พบ ความคลาดเคลื่อนแบบนี้มีผลทำให้การบอกตำแหน่งผิดไปประมาณ 0.5–1 เมตร

ความคลาดเคลื่อนอีกอย่างหนึ่งที่ได้เกิดจากดาวเทียมและเครื่องรับก็คือ Multipath Error ความคลาดเคลื่อนจากจำนวนเส้นทางที่มีมากกว่า 1 เส้นทาง เกิดจากการที่สัญญาณที่ส่งจากดาวเทียมตกกระทบไปยังวัตถุอื่นแล้วจึงสะท้อนเข้าเครื่องรับสัญญาณส่วนนี้ไม่เป็นสัญญาณจากดาวเทียมมีผลต่อการรับเหมือนกับที่เกิดกับการรับสัญญาณที่วัจนเดียวกัน คือทำให้เกิดภาพพร่าซ้อนให้เห็นบนจอจีพีเอส รุ่นใหม่ ใช้วิธีการประมวลผลที่ดีขึ้น และการใช้เสาอากาศที่ป้องกันสัญญาณซ้อนได้ แต่ในบางครั้งถ้าเกิดรุนแรงมาก ก็จะมีผลต่อการวัดจีพีเอส ได้เหมือนกันสาเหตุของการเกิดการคลาดเคลื่อนทั้งหมดที่กล่าวมาเป็นผลทำให้การวัดจีพีเอส ไม่แน่นอน ซึ่งความหมายว่า แทนที่จะกล่าวว่าของอยู่ห่างไป 10 ฟุต (3.5 เมตร) พอดี ต้องกล่าวว่าของอยู่ห่างไป 10 ฟุต บวกหรือลบเศษหนึ่งส่วนสิบนิ้ว เป็นต้นหรือเปรียบเทียบให้เห็นได้อีกอย่างเช่นเหมือนกับเรากำลังใช้ไม้บรรทัดที่ปลายข้างที่ใช้วัดหักหรือลบไป ทำให้ไม่สามารถบอกระยะที่แน่นอนที่เดียวได้ แต่ยัดตีที่ค่าความคลาดเคลื่อนทั้งหมดรวมกันแล้วก็จะจะมีไม่มาก ผลการใช้จีพีเอส จะบอกตำแหน่งได้ถูกต้องอยู่ในระยะไม่เกิน 25 เมตร และจะถูกต้องยิ่งขึ้นถ้าเครื่องรับคุณภาพดี เพื่อให้ได้ค่าความถูกต้องที่ดีที่สุด เครื่องรับที่ดีจะใช้หลักการของวิชาเรขาคณิต ซึ่งเรียกว่า Geometric Dilution of Precision (GDOP)

GDOP เป็นค่าที่ชี้ให้เห็นความถูกต้องของตำแหน่งที่เครื่องรับจีพีเอส แสดงให้เห็นโดยค่าตำแหน่งที่คำนวณได้มาจากการหาระยะจากดาวเทียมหลายดวง ลักษณะการประกอบรวมกลุ่มของดาวเทียมรูปเรขาคณิตหรือขนาดของมุมของดาวเทียมแต่ละดวงภายในกลุ่ม จะมีส่วนทำให้ความคลาดเคลื่อนเพิ่มหรือน้อยลงได้ เหมือนกับคนเล่นสนุกเกอร์ที่ต้องเลือกลูกที่มีมุมแทงลูกให้ลงหลุมได้ง่าย ซึ่งบางลูกอยู่ในมุมที่ดีแทงได้เต็มลูก บางลูกต้องแทงบางมากอาจผิดได้ ดูจากรูปจะเห็นว่าผลที่เกิดจาก GDOP ระยะที่วัดได้อยู่บนเส้นวงกลมที่ดาวเทียมเป็นจุดศูนย์กลาง และเนื่องจากระยะที่วัดได้มีขนาดไม่แน่นอนเส้นรอบวงจึงเป็นเส้นหนา เช่น ระยะแทนที่จะเป็น 10,000 ไมล์ ก็จะเป็นรัศมีระยะ 10,000 +/-

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

0.001 ไมล์ (9,999.999-10,000.001 ไมล์) ดังนั้นเวลาวาดเส้นรอบวงจะมีความกว้างเป็นแถบ ขนาด 0.002 ไมล์ ดังนั้น ตำแหน่ง X ที่เราเคยใช้เป็นแถบสี่เหลี่ยม "X" หรืออาจพูดได้ว่าเนื่องจากผลลัพธ์ไม่แน่นอน ตำแหน่งจึงไม่อยู่เป็นจุด แต่จะอยู่ในที่หนึ่งที่ใดก็ได้ในแถบสี่เหลี่ยมนี้ ดังนั้น GDOP จะได้มาจากมุมระหว่างดาวเทียมที่จะทำให้เกิดรูปแถบสี่เหลี่ยมขึ้น รูปสี่เหลี่ยมที่เกิดขึ้นอาจสวยงามดีได้ฉาก หรือยืดยาว และโตก็ได้ สรุปได้ว่าถ้าจะให้ผลลัพธ์ควรจะเลือกวัดจากกลุ่มดาวเทียมที่มุมระหว่างดาวเทียมมีขนาดใกล้เคียงหนึ่งในเครื่องรับจีพีเอส จะมีโปรแกรมในหีวเคราะห์ตำแหน่งของดาวเทียมที่อยู่บนท้องฟ้า และเลือกวัดจากชุดดาวเทียม 4 ดวง ที่มีค่า GDOP ดีที่สุด ในเครื่องรับประเภทละเอียดก็จะเลือกวัดจากดาวเทียมที่เห็นในท้องฟ้าขณะนั้น โดยวิธีนี้จะทำให้ค่าความคลาดเคลื่อนจาก GDOP เลื่อนน้อยที่สุด

2.5 รหัสสุ่มเทียม (Pseudo Random Code)

การสร้างรหัส Pseudo Random เป็นเรื่องปกติ แต่ระบบนี้ก็ช่วยให้ระบบจีพีเอส ทำงานได้ผล และราคาไม่แพง ซึ่งต่อไปอาจพูดได้ว่า โดยการใช้รหัส Pseudo Random จะทำให้จีพีเอส กลายเป็นเครื่องใช้ทั่วไปที่ทุกคนใช้ได้ ก่อนหน้านี้ได้กล่าวถึงการทำงานโดยการใช้รหัส Pseudo Random มาแล้ว เครื่องจีพีเอส สามารถหาเวลาที่ต่างกันระหว่างเครื่องรับกับดาวเทียมได้อย่างไร แต่นั่นเป็นเพียงส่วนหนึ่งเท่านั้น เหตุผลที่สร้าง Pseudo Random code ก็คือการประหยัด จะเปรียบเทียบให้เห็นชัดต้องดูจากดาวเทียมจีพีเอส ซึ่งกระจายเสียงด้วยสัญญาณกำลังแรงดีมาก แต่ส่วนเครื่องรับบนโลกยังต้องใช้จานรับดาวเทียมทรงโค้งกลมขนาดใหญ่ ลองนึกดูถ้าจีพีเอส ต้องใช้จานรับแบบเดียวกันจะต้องมีความทะอะทะขนาดไหน และยิ่งกว่านั้นดาวเทียมจีพีเอสหนึ่งบนท้องฟ้า แต่ดาวเทียมจีพีเอสโคจร ซึ่งต้องรับโดยหมุนไปหาตำแหน่งดาวเทียมทั้ง 4 ดวง จะต้องเป็นที่ยุงยากมากขึ้น

การใช้ Pseudo Random ช่วยลดความจำเป็นอื่นทั้งหมดในการส่งข้อมูล ดังนั้นการส่งสัญญาณจีพีเอส จึงกินกำลังไฟน้อยและสัญญาณจีพีเอส อ่อนมากที่ไม่รับเอาสัญญาณวิทยุรบกวนอื่น สัญญาณวิทยุรบกวนอื่นเกิดขึ้นเป็นคลื่นไม่เป็นรูปแบบ รหัส Pseudo Random ดูก็คล้ายคลื่นวิทยุรบกวนนี้มาก แต่มีข้อแตกต่างที่สำคัญคือ เราารู้รูปร่างของคลื่นแล้ว ถ้าหาว่าน่าเอาคลื่น Pseudo Random ไปเปรียบเทียบกับคลื่นวิทยุรบกวน การเปรียบเทียบจะแบ่งคลื่นออกเป็นช่วงเวลา ตามรูปให้ส่วนที่เหมือนกันเป็น "X" จะเห็นตามตัวอย่างจากที่คลื่นไม่มีรูปแบบแบบสุ่มโอกาสที่เกิดขึ้นได้เหมือนกันประมาณครึ่งหนึ่ง ถ้าเราให้ค่าคลื่นที่เหมือนกันเป็น 1 และที่ไม่เหมือนกันเป็น -1 จะพบว่าหลังจากเปรียบเทียบนาน ๆ จะได้ค่าสุดท้ายเป็น 0 แต่ถ้าจีพีเอส เริ่มส่งสัญญาณเข้าเครื่องรับที่มีรูปแบบเหมือนกับ Pseudo Random สัญญาณนี้ถึงแม้จะอ่อน จะถูกทำให้แรงขึ้นแล้วนำมาเปรียบเทียบกันได้และถ้าเราเลื่อนรหัสที่ส่งจากดาวเทียม เราก็จะได้คลื่นที่เข้ากันได้มากขึ้นและคะแนนก็จะมากขึ้นเรื่อย ๆ ยิ่งเปรียบเทียบนานตัวเลขจะเพิ่มมากขึ้นและจากที่ผลการเปรียบเทียบให้ผลตรงกันข้ามคลื่นวิทยุรบกวนซึ่งจะมีค่าเกือบศูนย์ ช่วงเวลานี้จะส่งกำลังขยายให้แก่สัญญาณดาวเทียมมากขึ้นเป็นพันเท่า รหัส Pseudo Random ช่วยให้เราจับสัญญาณที่อ่อนมากได้ ซึ่งหมายถึงในเครื่องจีพีเอส ไม่ต้องใช้ไฟมาก และมีวิธีการเพิ่มความแรงสัญญาณ เครื่องรับจึงใช้เสาอากาศขนาดเล็กได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ทำไมดาวเทียมอื่นไม่ใช้วิธีการนี้ ดาวเทียมทั่วไปต้องมีจานรับขนาดใหญ่มาติดตามบ้านที่เห็นทั่วไป ทั้งนี้เพราะสัญญาณจีพีเอส มีข้อมูลน้อย ต้องการเพียงให้หาเวลาได้เท่านั้น ตรงกันข้ามกับสัญญาณทีวีที่ข้อมูลมาก หรือถ้าพูดเชิงวิชาการก็ต้องว่าเป็นสัญญาณที่มีแถบกว้างสูงมาก หลักการของ Pseudo Random Code มีการทำงานในการเปรียบเทียบสัญญาณเหมือนกันหลาย ๆ รอบ การเปรียบเทียบนี้จะทำให้ช้าเมื่อเทียบกับแบบที่ต้องใช้สัญญาณทีวี ดังนั้นวิธีการจีพีเอส จึงนำมาใช้กับดาวเทียมทีวี ไม่ได้ ยังมีเหตุผลอื่นอีกสองข้อของการใช้รหัส Pseudo Random ข้อหนึ่งคือในเวลาสงครามดิโอดี สามารถควบคุมไม่ให้ศัตรูใช้ระบบนี้ได้ ถึงแม้เวลาปกติดิโอดี ยังคงเป็นผู้พิเศษในการควบคุมระบบอยู่ดี รหัส Pseudo Random มี 2 แบบ คือ C/A code และ P-code ดังนั้น บางคนจึงคิดว่า C/A code นี้ถูกต้องน้อยกว่า P-code รหัส P-code เก็บไว้ใช้เฉพาะทางราชการทหารและ P-code นี้จะไม่เกิดการติดขัด

ดิโอดียังสามารถลดความถูกต้องของ C/A code ได้ การทำโดยมาตรการ Selective Availability หรือ S/A วิธี S/A ที่สำคัญก็คือการทำให้นาฬิกาดาวเทียมบอกเวลาคลาดเคลื่อน ถ้าใครนำเวลานี้ไปใช้ก็จะได้ตำแหน่งที่มีความคลาดเคลื่อนมาก ประโยชน์อีกข้อของ Pseudo Random คือ ดาวเทียมทุกดวงสามารถใช้คลื่นความถี่เดียวกันได้ โดยไม่เกิดการรบกวนกัน ดาวเทียมแต่ละดวงจะมี Pseudo Random code เป็นของตัวเองเฉพาะตัว ดังนั้นเวลาเครื่องรับนำรหัสมาใช้ต้องให้ถูกตามหมายเลขดาวเทียมนั้นด้วย

2.6 การวัดด้วยวิธีอนุพันธ์ (Differential)

จีพีเอสเป็นเครื่องมือทำหน้าทีแสดงพิกัดบนโลกเราที่ถูกต้องที่สุดที่เคยมีมา แต่ยังมีวิธีการทำให้ถูกต้องได้ยิ่งขึ้นอีกโดยการใช้วิธีที่เรียกว่า "Differential GPS" แบบนี้จะทำให้ได้ความถูกต้องที่ดีกว่าหนึ่งเมตร และเป็นการนำเครื่องจีพีเอส ไปใช้ประโยชน์อย่างอื่นได้อีกด้วยจีพีเอส สำหรับการสำรวจจริงวัด ช่างสำรวจได้ใช้จีพีเอส มาเป็นเวลานานหลายปีแล้วสำหรับการคำนวณหาตำแหน่งที่ถูกต้องเป็นเซนติเมตร วิธีการนี้เป็นส่วนที่ใช้วิธี Differential การวัดให้ความ ถูกต้องละเอียดมากขึ้น จะต้องตั้งเครื่องเก็บข้อมูลจีพีเอส ที่สถานีเป็นเวลาอย่างน้อย 15 นาที จะต้องรู้จุดอ้างอิงหรือ Reference Point และใช้ซอฟต์แวร์คำนวณงาน การใช้จีพีเอส ทำงานสะดวกและประหยัดช่างสำรวจคนเดียวสามารถทำงานที่ต้องใช้คนหลาย ๆ คนได้ในเวลาเพียงเล็กน้อยเมื่อเทียบกับการรังวัดแบบเดิม ไม่ต้องปีนเขาสูงที่เป็นที่ตั้งของหมุดใน โครงข่ายสามเหลี่ยมเพื่อการส่องกล้องเล็งแนวให้ได้ระยะห่างไกล ๆ อีกต่อไป วิธีการรังวัดด้วยจีพีเอสแบบใหม่ ที่เรียกว่า Kinematics ซึ่งอยู่ในระหว่างการพัฒนา จะทำให้การทำงานเป็นแบบอัตโนมัติมากขึ้น ช่างสำรวจต้องทำเพียงเดินไปยังจุดที่ต้องการวัด แล้วกดปุ่มเท่านั้น เครื่องจีพีเอส ก็จะบันทึกตำแหน่งที่วัดที่แน่นอนได้ เคล็ดลับของการที่ให้ความถูกต้องสูงนี้มีพื้นฐานมาจากแนวคิดที่เราวางเครื่องจีพีเอส เครื่องหนึ่งไว้ที่จุดที่รู้ตำแหน่งบนโลกแล้ว (Reference Point) เราสามารถที่จะหาค่าความคลาดเคลื่อนของค่าตำแหน่งที่ได้จากดาวเทียมได้แน่นอน ดังนั้นที่จุดนี้จีพีเอส ทำหน้าที่เหมือนจุดอ้างอิง และสามารถใช้ค่าปรับแก้เดียวกันนี้กับเครื่องจีพีเอส ตัวอื่นที่อยู่บริเวณเดียวกันได้และสามารถนำไปใช้เป็นค่าแก้สำหรับการคำนวณตำแหน่งใช้เช่นเดียวกัน แนวคิดนี้ใช้ได้เพราะดาวเทียมอยู่สูงมากที่ค่าความคลาดเคลื่อนที่รับได้ในเครื่องรับ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เครื่องหนึ่งจะมีค่าเท่ากับค่าความคลาดเคลื่อนในเครื่องอื่นที่อยู่บริเวณเดียวกัน ทั้งนี้เนื่องจากสัญญาณของดาวเทียมเป็นสัญญาณที่ธรรมชาติ ค่าตัวคงที่ตัวเดียวกันจะมีผลรวมของค่าคลาดเคลื่อนที่เกิดสาเหตุอื่นไม่ว่าเกิดจากนาฬิกาเครื่องรับ นาฬิกาดาวเทียม ตำแหน่งดาวเทียมหรือการที่สัญญาณเดินทางเข้าไปในชั้นบรรยากาศ

รายงานผลคลาดเคลื่อนของเครื่องที่จุดอ้างอิงถูกส่งไปยังเครื่องรับอื่นได้ 2 รูปแบบ คือ วิธีการหนึ่งก็คือการส่งด้วยวิธีผ่านวิทยุหรือระบบสื่อสารไปที่เครื่องรับ แล้วให้คอมพิวเตอร์ประมวลตามความคลาดเคลื่อนที่จุดอ้างอิง เครื่องนั้นก็จะได้คำนวณหาตำแหน่งที่ถูกต้องได้

อีกวิธีหนึ่งโดยการเปลี่ยนเครื่องที่จุดอ้างอิงให้เป็นเหมือนดาวเทียมที่ส่งสัญญาณออกได้แบบเดียวกับดาวเทียม โดยการใช้ Pseudo Random Code แต่จะมีข้อมูลค่าแก้ไขในข่าวสารด้วย เครื่องรับที่จุดอื่นก็จะรับคลื่นจากดาวเทียมนี้ด้วยช่องรับสัญญาณที่วางอยู่ และนำไปแก้โดยวิธีเดียวกับที่ใช้แก้ค่าของอีพีเมอริสที่รับได้จากดาวเทียมจริง ระบบนี้จะไม่ค่อยมีข้อเสียในเรื่องสัญญาณรบกวน ซึ่งมีสถานะ เช่นเดียวกับที่เป็น การรับสัญญาณจากดาวเทียมจริง

2.7 ประเภทเครื่องรับสัญญาณ GPS

เครื่องรับแบ่งออกได้เป็น 2 กลุ่ม คือ เครื่องประเภทที่สามารถรับดาวเทียมได้ 4 ดวง หรือมากกว่าได้พร้อมกันทีเดียว กับเครื่องที่มีการรับดาวเทียมโดยการเรียงลำดับ และแต่ละกลุ่มยังแบ่งย่อยได้อีกคือ

2.7.1 เครื่องรับแบบเรียงลำดับสัญญาณดาวเทียม

ปกติเครื่องรับจีพีเอส จะต้องรับข้อมูลจากดาวเทียมอย่างน้อย 4 ดวง จึงสามารถคำนวณหาตำแหน่งที่ได้ เครื่องรับที่ใช้เรียงลำดับใช้ช่องรับสัญญาณเพียงช่องเดียว รับข้อมูลจากดาวเทียมดวงหนึ่งระยะหนึ่งแล้วเปลี่ยนไปยังอีกดวงหนึ่ง เครื่องประเภทนี้จะมีแฉวงจรรเล็ก ดังนั้นจึงมีราคาถูกลงและใช้กำลังน้อยกว่า ข้อเสียของการเรียงลำดับสัญญาณอาจเกิดขาดตอนและทำให้มีผลต่อความถูกต้องของผลที่ได้ ในกลุ่มนี้จะมี "Starved Power" Single-Channel Receivers, Two Channel Receivers, และเครื่องแบบเก่า Fast-Multiplexing Single Receivers

Starved-Power Single Receivers เครื่องแบบนี้ออกแบบให้พกพาได้และสามารถ ทำงานได้ด้วยถ่านไฟฉายขนาดเล็ก การจำกัดการใช้กระแสไฟโดยให้ปิดการทำงานตัวเองโดยอัตโนมัติ เมื่อแสดงตำแหน่งครั้งสองครั้งใน 1 นาที เหมาะสำหรับใช้งานบอกตำแหน่งส่วนตัว เช่น นักไต่เขาหรือเล่นเรือในเวลากลางวัน โดยไม่ต้องมีถ่านไฟฉายหลายก้อน นับว่าเป็นเครื่องที่ใช้การได้ สามารถให้ความถูกต้องที่ดีกว่าระบบ LORAN และทำงานได้ทุกที่บนโลก ข้อเสีย คือ ความถูกต้องของจีพีเอส ไม่ดีและต้องเชื่อมกับอุปกรณ์อื่นไม่ได้ และไม่สามารถใช้วัดหาความเร็วได้ การที่หาความเร็วไม่ได้ เนื่องจากต้องปิดเครื่อง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เองในระหว่างการวัด เพราะว่าเครื่องใช้แผงวงจรนาฬิกาที่กินไฟน้อย (นาฬิกาจะต้องเดินอยู่ตลอดเวลา) นาฬิกาที่ใช้จึงไม่ให้ความถูกต้องเท่าที่ควร

Single Channel Receivers เหมือนกับแบบค่าข้างบน เป็นเครื่องรับสัญญาณห้องเดียวใช้ทำงานหาระยะจากดาวเทียมทุกดวง แต่ที่ไม่เหมือนคือเครื่องรับช่องเดียวแบบมาตรฐานไม่จำกัดที่กัลังไฟ ดังนั้นจึงทำการรับต่อเนื่องได้ มีผลทำให้ความถูกต้องสูงกว่า และใช้วัดหาความเร็วได้ จากที่มีเพียงช่องเดียวที่ต้องใช้ทั้งการรับข้อมูลดาวเทียมและคำนวณหาระยะ จึงไม่สามารถหาตำแหน่งต่อเนื่องได้ ยิ่งกว่านั้นตามเหตุผลของวิชาการ ความไม่เที่ยงตรงของนาฬิกามีผลโดยตรงต่อความถูกต้องของการวัดหาความเร็ว เครื่องราคาถูกบางชนิดใช้นาฬิการาคาถูกเพื่อให้ราคาเครื่องลดลง จึงทำให้ค่าความเร็วที่แสดงมาเชื่อถือไม่ได้

Fast-Multiplexing Single Receivers เครื่องประเภทนี้เหมือนกับเครื่องทั้งสองประเภทข้างบนซึ่งรับซ้ำ แต่เครื่องรับนี้สามารถเปลี่ยนดาวเทียมได้เร็วกว่ามาก ข้อดี คือ สามารถทำการวัดได้ในขณะที่กำลังรับข้อมูลจากดาวเทียม ดังนั้นเครื่องทำงานได้อย่างต่อเนื่อง และกรณีที่มิมีนาฬิกาไม่เที่ยงจึงมีผลต่อเครื่องประเภทนี้น้อย เครื่องแบบนี้ต้องการใช้แผงวงจรที่ค่อนข้างซับซ้อนและราคาพอ ๆ กับเครื่องแบบสองช่องรับสัญญาณที่ใช้เครื่องซึ่งให้ความถูกต้องสูงกว่าและมีลักษณะการปิดหน่วยการใช้งานได้ดีกว่า

Two-Channel Sequencing Receivers การเพิ่มช่องรับสัญญาณขึ้นอีกหนึ่งช่องช่วยให้เครื่องเพิ่มขีดความสามารถขึ้นอย่างเห็นได้ชัด ข้อหนึ่งความแรงสัญญาณ Signal-to-Noise เป็นสองเท่าทันที หมายถึงสามารถจับสัญญาณภายใต้สภาวะที่ไม่ดีได้และสามารถรับดาวเทียมดวงที่อยู่ระดับต่ำใกล้เส้นขอบฟ้าได้จากการที่ช่องหนึ่งสามารถรับข้อมูลตำแหน่งอย่างต่อเนื่องได้ในขณะที่อีกช่องหนึ่งค้นหาดาวเทียมดวงต่อไป เครื่องแบบสองช่องนี้จะทำงานแบบนาร่องได้โดยไม่ต้องมีการขาดตอน และความเร็วก็จะมีค่าที่ถูกต้องขึ้น ความจริงเครื่องรับสองช่องที่มีคุณภาพดีก็สามารถใช้คำนวณหาและตัดค่าที่เวลาของนาฬิกาเครื่องรับไม่ดีทิ้งเพื่อใช้ในการวัดหาความเร็ว ข้อเสียของเครื่องแบบสองช่องคือ มีราคาสูงกว่าและกินไฟมากกว่า ในเครื่องรับรุ่นใหม่มักใช้แบบ IC ที่สามารถเพิ่มช่องรับสัญญาณที่สองในราคาที่ไมต่างกับราคานาฬิกาดี ๆ หนึ่งเรือน แต่กระนั้นเครื่องแบบสองช่องยังมีราคาแพงกว่าเครื่องแบบช่องเดียวมาก ทั้งนี้ เนื่องจากผู้ใช้สองช่องมักต้องการความถูกต้อง และต้องการเครื่องที่แข็งแรงและสามารถควบคุมสังเกตการณ์แสดงผลที่ดีกว่า

Continuous Receivers ได้แก่เครื่องรับที่สามารถรับสัญญาณดาวเทียมพร้อมกันได้ตั้งแต่ 4 ดวงขึ้นไป และสามารถแสดงผลค่าตำแหน่งและความเร็วได้ทันที การรับดาวเทียมได้ทั้ง 4 ดวง พร้อมกับที่มีค่าในการวัดหาในขณะที่มีการเปลี่ยนตำแหน่งรวดเร็วหรือต้องการความถูกต้องสูง ดังนั้นเครื่องแบบนี้จึงนำมาใช้งานรังวัดและทางด้านวิทยุ ซึ่งจะพบว่าจะมีช่องรับสัญญาณทั้ง 4, 5, 8 10 และ 12 ช่อง นอกจากข้อดีที่ไว้วัดตำแหน่งอย่างต่อเนื่องได้แล้ว เครื่องรับจีพีเอส แบบหลายช่องสามารถช่วยขจัดปัญหา GDOP

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ได้อีกด้วย คือ แทนที่จะรับดาวเทียม 4 ดวงได้ก็จะได้จำนวนค่า GDOP ดาวเทียม 4 ดวงของกลุ่มดาวเทียมที่ขึ้นอยู่ และทำการวัดจากดาวกลุ่มที่มีค่า GDOP ต่ำสุด เครื่องรับ 4 ช่องสัญญาณ สามารถให้ค่า Signal to Noise Ratio เพิ่มขึ้นเป็นสองเท่าของเครื่อง 2 ช่อง และเป็นสี่เท่าของเครื่องรับแบบช่องเดียว และโดยการเปรียบเทียบค่าการรับของแต่ละช่อง เครื่องสามารถปรับตั้งค่าพิกัดเทียมระหว่างช่องรับสัญญาณ ซึ่งช่วยทำให้การวัดมีความถูกต้องดีขึ้น นอกเหนือจากข้อดี ข้อเสียที่กล่าวแล้ว ยังมีข้ออื่นมาพิจารณาอีก คือ มีเครื่องแบบใหม่สามารถได้ค่าความถูกต้องสูงมาก โดยการใช้รหัส Pseudo Random ที่กล่าวมาแล้ว และใช้ความถี่ของคลื่นพาห์ (Carrier Frequency) ซึ่งทำให้เครื่องรับทำงานมีความเที่ยงสูง ที่รหัส Pseudo Random ไม่สามารถให้ได้ และใช้ในการวัดหาเวลาได้แม่นยำมากขึ้น ซึ่งช่วยในการบอกตำแหน่งได้ดีขึ้นด้วย และมีบางเครื่องที่ไม่ต้องใส่ค่าประมาณตำแหน่งและเวลา โดยประมาณให้เครื่องก่อนทำการวัด เครื่องรับแบบนี้ใช้ตัวเองใส่ค่าเริ่มตำแหน่งได้โดยตัวมันเอง ข้อที่ควรพิจารณา คือ การต่อเชื่อมกับอุปกรณ์อื่นและความสะดวกบางเครื่องแสดงได้เฉพาะพิกัดภูมิศาสตร์ บางเครื่องไม่สามารถต่อเข้ากับเครื่องมืออื่นหรือคอมพิวเตอร์ขนาดเล็ก (PC) ได้ และข้อใหญ่ที่ต้องพิจารณา ความแข็งแรงทนทานถ้าต้องใช้เครื่องทำงานในพื้นที่ทะเล หรือในพื้นที่ป่าเขา การใช้ไฟและความร้อนที่เกิดขึ้นเป็นตัวชี้สำคัญที่จะต้องเอาใจใส่ ทางสถิติแสดงให้เห็นว่าอัตราของค่าความผิดพลาดจะเพิ่มเป็นสองเท่าของความร้อนในเครื่องเพิ่มทุก 7 องศาเซลเซียส เครื่องรับรุ่นใหมปัจจุบันได้เพิ่มคุณค่าให้แก่เครื่องรับ GPS อีกหลายประการ เช่น ใช้การประมวลผลที่ซับซ้อน แสดงผลด้วยจอภาพรายละเอียด เครื่องรับจีพีเอสอาจแสดงจุดตำแหน่งบนแผนที่ที่ได้วาดไว้แล้วให้เห็นทันที

2.8 มาตรฐาน NMEA และโปรโตคอลที่ใช้ในการสื่อสารของ จีพีเอส

2.8.1 มาตรฐาน NMEA (NMEA Standard)

คือการอินเทอร์เฟซทางไฟฟ้าและโปรโตคอลการสร้างข้อมูลสำหรับการสื่อสารระหว่างอุปกรณ์การเดินเรือ (NMEA ยังมีมาตรฐานสำหรับการสื่อสารในแบบอื่นๆด้วย) โดย NMEA ย่อมาจากคือ National Marine Electronics Association เริ่มแรกถูกพัฒนาขึ้นเพื่อใช้ในการแลกเปลี่ยนข้อมูลกับเซนเซอร์อิเล็กทรอนิกส์ที่ใช้ในการเดินเรือเป็นหลักต่อมาเมื่อระบบจีพีเอสเข้ามามีบทบาทในการเดินเรือมากขึ้นจึงทำให้โปรโตคอล NMEA-0183 ถูกพัฒนามาเป็นมาตรฐานกลางสำหรับใช้สื่อสารระหว่างอุปกรณ์รับพิกัดจีพีเอสและอุปกรณ์ต่อพ่วงอื่นๆ (Terminal equipment) สำหรับข้อมูลที่อ่านได้จากโมดูลรับพิกัดจีพีเอสสามารถแบ่งได้เป็นเรคอร์ด (Record) หรือฟิลด์ (Field) ย่อยโดยแต่ละเรคอร์ดจะประกอบด้วยอักขระแอสกี (ASCII) ซึ่งมีความยาวรวมไม่เกิน 80 ตัวอักษร เรคอร์ดหลักในโปรโตคอล NMEA-0183 จะมี 6 เรคอร์ดหลักคือ GGA (Global positioning system fixed data) , GLL (Geographic position latitude/longitude) , GSA (GNSS DOP and active satellites) , GSV (GNSS Satellites in view) , RMC (Recommended minimum specific GNSS data) และ VTG (Course over ground and ground speed) ซึ่งมีรายละเอียดของค่าต่างๆ เช่น พิกัดตำแหน่ง ละติจูด ลองจิจูด เวลา จำนวนดาวเทียมที่ใช้ในการคำนวณพิกัด และความสูงจากระดับน้ำทะเล (MSL Altitude) เป็นต้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.8.2 การอินเตอร์เฟซทางไฟฟ้า (Electrical Interface)

มาตรฐานนี้สามารถใช้เป็นระบบที่มีตัวส่ง (talker) เดียวและตัวรับ (Listener) สายที่แนะนำให้ใช้เป็นชีลด์ทวิสต์แพร์โดยต่อกราวด์ที่ตัวส่งเท่านั้นมาตรฐานไม่ได้กำหนดชนิดคอนเนคเตอร์(connector) เจาะจง

2.8.3 มาตรฐาน NMEA – 0183

มาตรฐานที่เอาท์พุทจะเป็นแบบ อีไอเอ – 4221 และมีสายสัญญาณ 2 เส้น, เอ และบี โวลเตจบนเส้นเอจะเป็นเหมือนกับสาย ทีทีแอล เดียวแบบเดิม ขณะที่บี โวลเตจจะกลับทางกันกับ เอ เช่น เอเป็น +5 บีจะเป็นกราวด์ ในการใช้งาน, สายเพียงสายเดียวคือ สายเอ ใน อีไอเอ – 4221 อาจจะถูกใช้เชื่อมต่อกับอาร์เอส – 232 อินพุทของเครื่องคอมพิวเตอร์ในมาตรฐาน เอ็นเอ็มอีเอ – 0183, ตัวอักษรที่ใช้คือแอสกี ซึ่งสามารถพิมพ์ได้ เอ็นเอ็มอีเอ – 0183 นั้นข้อมูลจะถูกส่งด้วยอัตรา 4800 บอร์ด ข้อมูลจะถูกส่งในรูปของประโยค แต่ละประโยคเริ่มด้วย \$ ตัวอักษรที่ตามมาสองตัวคือ เซ็นเทนไอดีหรือตัวกำหนดรูปแบบประโยค หรือจะเรียกว่าชื่อประโยค (sentence name) ตามด้วยฟิลด์ข้อมูลจำนวนหนึ่งซึ่งถูกแบ่งแยกโดยเครื่องหมายคอมม่า (,) และสิ้นสุดด้วยเครื่องหมายเช็คซัม (check sum) ที่สามารถเลือกได้ว่าจะมีหรือไม่ และจบลงด้วยแครีเรียเทอร์นและไลน์ฟีด (carrier return/line feed) ซึ่งประโยคจะมีตัวอักษรถึง 82 ตัวรวม \$ และแครีเรียเทอร์นและไลน์ฟีด

ถ้าข้อมูลสำหรับฟิลด์ไม่สามารถหาได้ ฟิลด์จะถูกเว้นข้ามไปแค่คอมม่าที่ทำหน้าที่แบ่งฟิลด์ยังคงถูกส่งไปโดยไม่เว้นช่องว่าง เพราะว่าในแต่ละฟิลด์มีความยาวไม่คงที่หรือไม่มีข้อมูลเครื่องรับจะระบุตำแหน่งของฟิลด์ของข้อมูลที่ต้องการ โดยการนับเครื่องหมายคอมม่าเช็คซัมที่เลือกได้ว่าจะมีหรือไม่มีประกอบด้วย “*” และ 2 บิตของเลขฐาน 16 แทนการ เอ็กซ์คูซีฟอ์ของตัวอักษรทั้งหมดแต่ไม่รวม “\$” และ “*” ในการใช้งานจะมีความต้องการให้เช็คซัมในบางประโยคในมาตรฐานจะอนุญาตแต่ละผู้ผลิตในการนิยามรูปแบบประโยค ประโยคเหล่านี้เริ่มต้นด้วย “\$PG” และตัวอักษรสามตัว ที่ตามมาเป็นไอดีที่ถูกกำหนดมาจากโรงงานตามโรงงานตามข้อมูลซึ่งเป็นไปตามรูปแบบทั่วไปของประโยคมาตรฐาน

เอ็นเอ็มอีไอ คือ โปรโตคอลมาตรฐาน ถูกนำมาใช้โดยเครื่องรับ จีพีเอส เพื่อส่งข้อมูล เอ็นเอ็มอีเอ เอาท์พุทจะเป็นโปรโตคอล อีไอเอ – 422 เอ แต่เราสามารถนำไปใช้ร่วมงานกับ อาร์เอส -232 ได้โดยอัตรา การส่งข้อมูลเป็น 4800 บิตต่อวินาที, 8 คาตาบิต, ไม่มีพาริตีบิต แต่มีหนึ่งสตอปบิต และประโยคของ เอ็นเอ็มอีเอ 0183 จะเป็นแอสกี ทั้งหมดแต่ละประโยคจะเริ่มต้นด้วย “\$” และจบลงด้วย “<CR><LF>” และข้อมูลจะถูกแบ่งขึ้นด้วย “,” เครื่องรับจีพีเอส บางตัวไม่ส่งฟิลด์ข้อมูลเช็คซัม ถูกเพิ่มเข้าไป(ในบางกรณี) ส่วนที่ตามหลัง \$ คือแอดเดรสฟิลด์ aaccc aa คือ device id.GP ใช้เพื่อบ่งชี้ว่าเป็นข้อมูล จีพีเอส การส่ง device id. ตามปกติแล้วเลือกได้ ccc คือรูปแบบประโยค

2.8.4 รูปแบบประโยคของ NMEA

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

RMB

SGPRMB,A,x.x,a,c--cc,d--d,III,II,e,yyyyy.yy,f,g,g,h,h,i,i,j*kk

RMB = เป็นข่าวสารที่น้อยที่สุดในการนำร่อง (Recommended Minimum Navigation information)

- 1 = สถานะของข้อมูล (v = การเตือนเครื่องรับในการนำร่อง (navigation receiver warning))
- 2 = ความคลาดเคลื่อนของครอสแทรค (Cross track error) ในหน่วยไมล์
- 3 = ทิศทางที่ต้องเบนเข็มไป (Direction to steer)(L or R) เพื่อแก้ไขความคลาดเคลื่อน
- 4 = หมายเลขเริ่มต้นของเวย์พอยท์ (waypoint)
- 5 = หมายเลขเวย์พอยท์ปลายทาง
- 6 = ละติจูดของเวย์พอยท์ปลายทาง
- 7 = เหนือหรือใต้
- 8 = ลองจิจูดของเวย์พอยท์เป้าหมาย
- 9 = ตะวันออกหรือตะวันตก
- 10 = ระยะทางไปยังปลายทางในหน่วยไมล์
- 11 = แบริ่ง(มุมที่ทำกับปลายทาง) ในหน่วยองศา
- 12 = ความเร็วในการเข้าถึงเป้าหมาย (Destination closing velocity) ในหน่วยนอต
- 13 = สถานะเมื่อเข้าถึง (Arrival status)(A=เข้าถึงหรือผ่าน ไปยังดังกล่าว)
- 14 = เช็คซัม

RMC

\$GPRMC,hhmmss.ss,A,III,II,a,yyyyy.yy,a,x.x,x.x,ddmmyy,x.x,a*hh

RMC = ข้อมูลเฉพาะของจีพีเอส / ทรานสมิตเตอร์ที่น้อยที่สุดที่จำเป็น (Recommended Minimum Specific GPS / TRANSIT Data)

- 1 = เวลาปัจจุบันในระบบยูทีซี
- 2 = สถานะของข้อมูล
- 3 = ละติจูด
- 4 = ทิศเหนือหรือทิศใต้
- 5 = ลองจิจูด
- 6 = ตะวันออกหรือตะวันตก
- 7 = ความเร็วในหน่วยนอต
- 8 = มุมที่วัดจากทิศเหนือ
- 9 = วันเดือนปี

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

10 = ความแปรปรวนเนื่องจากสนามแม่เหล็ก

11 = ตะวันออกหรือตะวันตก

12 = เช็คซัม

GGA

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

GGA = ข้อมูลเฉพาะของระบบจีพีเอส(Global Positioning System Fix Data)

1 = ตำแหน่งของยูทิส

2 = ละติจูด

3 = ทิศเหนือหรือใต้

4 = ลองติจูด

5 = ตะวันออกหรือตะวันตก

6 = ชี้อ้างคุณภาพของระบบจีพีเอส(0 = invalid; 1 = จีพีเอส fix; 2 = diff.จีพีเอส fix)

7 = จำนวนดาวเทียมที่ใช้

8 = โฮริซอน ดิลูชัน ออฟ พозиชัน (Horizon dilution of position)

9 = ความสูงของสายอากาศ เหนือกว่าหรือต่ำกว่าระดับน้ำทะเล

10 = หน่วยความสูงของสายอากาศ(เมตร)

11 = ค่าความแตกต่างระหว่างระบบ คัมเบิ้ลยูจีเอส - s4 กับระบบจีออซด์

12 = ความแตกต่างของจีออซด์

13 = ระยะเวลาตั้งแต่อัปเดตข้อมูลครั้งสุดท้ายจากสถานีอ้างอิง

14 = หมายเลขประจำตัวสถานีอ้างอิง

15 = เช็คซัม

VGT

\$GPVGT,t,T,,,s.ss,N,s.ss,k*hh

VGT = มุมที่ทำกับทิศเหนือจริงๆและความเร็ว (Accualtrack madegood and speedover ground)

1 = มุมที่ทำกับทิศเหนือ

2 = อักษร 'T' แสดงว่ามุมที่วัดเทียบกับทิศเหนือ

3 = ไม่ใช่

4 = ไม่ใช่

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5 = ความเร็วหน่วยนอต

6 = อักษร 'N' แสดงถึงความเร็วในหน่วยนอต

7 = ความเร็วหน่วย กิโลเมตร/ชั่วโมง

8 = อักษร 'T' แสดงถึงความเร็วในหน่วย กิโลเมตร/ชั่วโมง

9 = เช็คซัม

GSA

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

GSA = โหมดการทำงานของเครื่องรับจีพีเอส, ดาวเทียมที่ใช้ในการนำร่องและค่าดีไอพี (GPS receiver operating, Mode, SVS used for navigation, DOP values)

1 = โหมด

M = Manual, ให้ผู้ใช้เลือกว่าจะใช้ 2 มิติ หรือ 3 มิติ

A = Automatic, 2 มิติ/3 มิติ

2 = โหมด

1 = ไม่สามารถระบุตำแหน่งได้

2 = 2 มิติ

3 = 3 มิติ

4-14 = หมายเลขของดาวเทียมที่ใช้ในการระบุตำแหน่ง (เป็น 0 สำหรับฟิลด์ที่ไม่ใช้)

15 = พีดีไอพี

16 = เฮสติไอพี

17 = วีดีไอพี

GSV

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

GVS = จำนวนของดาวเทียมที่มองเห็น, หมายเลข พีอาร์เอ็น, เอลเวชัน, อาซิมุทและค่าเอสเอ็นอาร์ (Number of SVs in view, PRN number, elevation, azimuth & SNR value.)

1 = จำนวนหมายเลขของข่าวสารทั้งหมดของมูลี่ใน 1 รอบ

2 = หมายเลขข่าวสาร

3 = จำนวนทั้งหมดของดาวเทียมที่อยู่ในพิสัย

4 = หมายเลข พีเอ็นอาร์ ของดาวเทียม

5 = มุมเอลเวชันหรือมุมเงยมีค่าสูงสุด 90 องศา

6 = มุมอาซิมุททำกับหัวเหนือมีค่า 0-359 องศา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

7 = ค่าเอสเอ็นอาร์ มีค่า 0 – 99 เดซิเบล (เมื่อไม่มีการแทรกคั้ง)

8 – 11 = ข่าวสารเกี่ยวกับดาวเทียมดวงที่สอง, เหมือนกับฟิลด์ 4 – 7

12 = ข่าวสารเกี่ยวกับดาวเทียมดวงที่สาม, เหมือนกับฟิลด์ 4 – 7

2.9 จีพีอาร์เอส (GPRS)

GPRS (General Packet Radio Service) คือเทคโนโลยีการเชื่อมต่อข้อมูลความเร็วสูง ผ่านเครือข่ายสัญญาณโทรศัพท์มือถือ สามารถเชื่อมต่อการใช้งานได้ ทุกที่ทุกเวลา เพียงสถานที่นั้นมีสัญญาณโทรศัพท์มือถือโดย O2 PDA Phones ทุกุ่นสามารถรองรับการ ใช้งานจีพีอาร์เอสได้อย่างมีประสิทธิภาพ และในปัจจุบัน ผู้ให้บริการเครือข่ายโทรศัพท์มือถือในระบบจีพีอาร์เอส ทุกรายในประเทศไทย ต่างก็มีระบบจีพีอาร์เอส ให้ลูกค้าเลือกใช้หลากหลายรูปแบบตามลักษณะการใช้งานของผู้ใช้แต่ละรายละเอียดอีก ค่าย

เทคโนโลยี EDGE หรือ Enhanced Data rate for GSM/GPRS Evolution เป็นข้อกำหนดทางเทคนิค ที่ได้รับการออกแบบและกำหนดขึ้นโดยหน่วยงาน ETSI (European Telecommunications Standard Institute) ซึ่งเป็นผู้สร้างมาตรฐานโทรศัพท์เคลื่อนที่จีเอสเอ็ม (GSM) และจีพีอาร์เอส โดยมีเป้าหมายเพื่อพัฒนาขีดความสามารถของเครือข่ายโทรศัพท์เคลื่อนที่ GSM/GPRS ให้รองรับการสื่อสารข้อมูลด้วยอัตราเร็วที่สูงขึ้นเทียบเท่ากับมาตรฐานโทรศัพท์เคลื่อนที่ UMTS แบบ W-CDMA แต่ใช้งบประมาณในการลงทุนที่ต่ำ ที่สำคัญคือผู้ให้บริการโทรศัพท์เคลื่อนที่จีเอสเอ็มทั่วโลก ไม่ว่าจะใช้ความเร็ว 900, 1800 หรือ 1900 เมกะเฮิร์ตซ์ ต่างก็สามารถพัฒนาเครือข่าย ให้รองรับเทคโนโลยี EDGE ได้โดยไม่ต้องใช้ความถี่ในย่าน 3G แต่อย่างใด ถือเป็นการเพิ่มขีดความสามารถในการสื่อสารข้อมูลแบบมัลติมีเดียให้กับเครือข่าย GSM/GPRS โดยใช้เวลาน้อยลง และด้วยงบประมาณที่ต่ำมาก

พิจารณาในแง่ของผลประโยชน์ที่ผู้ให้บริการ โทรศัพท์เคลื่อนที่พึงได้รับจากเทคโนโลยี EDGE ก็คือความรวดเร็วในการรับส่งข้อมูล ซึ่งเมื่อเปรียบเทียบประสิทธิภาพในการสื่อสารข้อมูลผ่านเครือข่ายโทรคมนาคมชนิดต่าง ๆ แล้วก็จะพบว่าทั้งเครือข่าย EDGE, UMTS แบบ W-CDMA หรือแม้กระทั่ง cdma2000 ซึ่งต่างมีขีดความสามารถในการรองรับการสื่อสารข้อมูลที่เทียบเท่ากัน จะสามารถตอบสนองความพึงพอใจของผู้ใช้บริการ ได้ใกล้เคียงกัน ผู้ใช้บริการสามารถใช้โทรศัพท์เคลื่อนที่มาตรฐาน 3G ดาวน์โหลดข้อมูลที่เป็น Video Clip ได้ในเวลาเพียงสิบกว่าวินาทีเท่านั้น ในขณะที่ต้องใช้เวลาในการดาวน์โหลดข้อมูลดังกล่าวผ่านเครือข่ายจีพีอาร์เอส นานถึง 2-3 นาที ยิ่งกว่านั้นในกรณีของการดาวน์โหลดข้อมูลผ่านเครือข่ายจีพีเอส ซึ่งรองรับการสื่อสารข้อมูลได้ด้วยอัตราเร็วสูงสุดเพียง 9.6 กิโลบิตต่อวินาที อีกทั้งยังเป็นการสื่อสารแบบสวิตช์วงจร ก็จะต้องใช้เวลานานถึงเกือบ 1 ชั่วโมง ทำให้มองเห็นได้ชัดเจนว่า ผู้ให้บริการเครือข่ายโทรศัพท์เคลื่อนที่ EDGE สามารถใช้ประโยชน์จากขีดความสามารถของเครือข่ายในการเพิ่มรายได้ให้กับตนเอง ไม่ว่าจะเป็นการนำเสนอบริการสื่อสารข้อมูลความเร็วสูง เช่น การรับส่งสัญญาณวิดีโอ ฯลฯ ไปจนถึงการแบ่งสรรทรัพยากรของสื่อสารอัตราเร็วสูงดังกล่าวให้กับผู้ให้บริการหลาย ๆ รายได้ใช้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

งานร่วมกัน โดยผู้ให้บริการแต่ละรายอาจไม่ต้องการความเร็วในการสื่อสารข้อมูลมากนัก ซึ่งอาจเป็นการรับส่งอีเมล, การเข้าชมเว็บไซต์ หรือการรับส่งไฟล์รูปภาพต่าง ๆ ฯลฯ

2.10 เอสคิวแอล (SQL)

เอสคิวแอล (SQL) คือ ภาษาสอบถามข้อมูล หรือภาษาจัดการข้อมูลอย่างมีโครงสร้าง มีการพัฒนาภาษาคอมพิวเตอร์ และ โปรแกรมฐานข้อมูลที่รองรับมากมาย เพราะจัดการข้อมูลได้ง่าย เช่น MySQL, MsSQL, PostgreSQL หรือ MS Access เป็นต้น สำหรับโปรแกรมฐานข้อมูลที่ได้รับความนิยมคือ MySQL เป็น Open Source ที่ใช้งานได้ทั้งใน Linux และ Windows SQL เป็นภาษาที่ใช้ในการเขียนโปรแกรมเพื่อจัดการกับฐานข้อมูลโดยเฉพาะ เราสามารถแบ่งการทำงานได้เป็น 4 ประเภท ดังนี้

1. Select query ใช้สำหรับดึงข้อมูลที่ต้องการ
2. Update query ใช้สำหรับแก้ไขข้อมูล
3. Insert query ใช้สำหรับการเพิ่มข้อมูล
4. Delete query ใช้สำหรับลบข้อมูลออกไป

Select query

ใช้ในการดึงข้อมูลในฐานข้อมูล จะมีการค้นหารายการจากตารางในฐานข้อมูล ตั้งแต่หนึ่งตารางขึ้นไป ตามเงื่อนไขที่สั่ง ผลลัพธ์ที่ได้จะเป็นเซตของข้อมูลที่สามารถสร้างเป็นตารางใหม่ หรือใช้แสดงออกมาทางจอภาพเท่านั้น โดยมีรูปแบบดังนี้ Select รายละเอียดที่เลือก From ตารางแหล่งที่มา Where กำหนดเงื่อนไขฐานข้อมูลที่เลือก Group by ชื่อคอลัมน์

ตัวอย่างการใช้งาน

1. Select fname, lname From stdinfo หมายถึง ให้แสดงเฉพาะคอลัมน์ fname คือ ชื่อ และคอลัมน์ lname คือ นามสกุล จากตาราง stdinfo
2. Select fname, lname From stdinfo Where programe= "สังคมศึกษา" หมายถึง ให้แสดงชื่อ และ นามสกุลจากตาราง stdinfo ซึ่งมีโปรแกรมวิชาเป็นสังคมศึกษา
3. Select fname From stdinfo Where fname Like 'ส%' หมายถึง ให้เลือกรายชื่อ นักศึกษาที่มีอักษรนำหน้าเป็น "ส" ขึ้นมาแสดงทั้งหมด
4. Select id, fname, lname From stdinfo Where id= "001" AND id= "005" หมายถึง ให้แสดงรหัสประจำตัวนักศึกษา, ชื่อ และ นามสกุล ที่มีรหัสเป็น 001 และ 005

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Update query

ใช้สำหรับการแก้ไขข้อมูลในตาราง โดยแก้ไขคอลัมน์ที่มีค่าตรงตามเงื่อนไข มีรูปแบบดังนี้
 Update ชื่อตาราง Set [ชื่อคอลัมน์=ค่าที่จะใส่เข้าไปในคอลัมน์นั้น ๆ] Where เงื่อนไขเช่น จากตารางแสดง
 รายชื่อนักศึกษากรณีที่นักศึกษาชื่อ สมบัติ มکنون ย้ายโปรแกรมวิชา จาก สังคมศึกษา ไปเป็นภาษาไทย
 เราใช้คำสั่งดังนี้ Select stdinfo Set program = 'ภาษาไทย' Where Fname = 'สมบัติ' and Lname = 'มکنون'

Insert query

ใช้ในการเพิ่มข้อมูลใหม่ ๆ เข้าไปในฐานข้อมูล มีรูปแบบดังนี้ Insert Into ชื่อตาราง [=ชื่อ
 คอลัมน์1,2,...] Values [ค่าที่จะใส่ลงในคอลัมน์ 1,2,...] เช่น ต้องการเพิ่มรายชื่อนักศึกษา ที่มีรหัสประจำตัว
 เป็น 007 ชื่อ กมลวรรณ ศิริกุล โปรแกรมวิชา วิทยาศาสตร์ เราสามารถใช้คำสั่งดังนี้ Insert into stdinfo
 (id,fname,lname,program) Values ('007', 'กมลวรรณ', 'ศิริกุล', 'วิทยาศาสตร์')

Delete query

ใช้ลบข้อมูลออกจากตาราง มีรูปแบบดังนี้ Delete From ชื่อตาราง Where เงื่อนไข เช่น ต้องการ
 ลบรหัสประจำตัวนักศึกษา 005 ออกจากฐานข้อมูล เราใช้คำสั่งดังนี้ Delete From stdinfo Where id= '005'

2.11 MySQL (มายเอสคิวแอล)

MySQL (มายเอสคิวแอล) เป็นระบบจัดการฐานข้อมูลโดยใช้ภาษาเอสคิวแอล แม้ว่ามายเอสคิว
 แอล เป็นซอฟต์แวร์โอเพนซอร์ส แต่มันแตกต่างจากซอฟต์แวร์โอเพนซอร์สทั่วไป โดยมีการพัฒนาภายใต้
 บริษัท MySQL AB ในประเทศสวีเดน โดยจัดการมายเอสคิวแอล ทั้งในรูปแบบที่ให้ใช้ฟรี และแบบที่ใช้ใน
 เชิงธุรกิจมายเอสคิวแอล สร้างขึ้นโดยชาวสวีเดน 2 คน และชาวฟินแลนด์ ชื่อ David Axmark, Allan
 Larsson และ Michael "Monty" Widenius. ปัจจุบันบริษัทซันไมโครซิสเต็มส์ (Sun Microsystems, Inc.)
 เข้าซื้อกิจการของ MySQL AB เรียบร้อยแล้ว ฉะนั้นผลิตภัณฑ์ภายใต้ MySQL AB ทั้งหมดจะตกเป็นของ
 ซัน ชื่อ "MySQL" อ่านออกเสียงว่า "มายเอสคิวแอล" หรือ "มายเอสคิวแอล" (ในการอ่านอักษร L ใน
 ภาษาไทย) ซึ่งทางซอฟต์แวร์ไม่ได้อ่าน มายซีเควล หรือ มายซีควล เหมือนกับซอฟต์แวร์จัดการฐานข้อมูล
 ตัวอื่นมายเอสคิวแอล เป็นที่นิยมใช้กันมากสำหรับฐานข้อมูลสำหรับเว็บไซต์ เช่น มีเดียวิกิ และ phpBB
 และนิยมใช้งานร่วมกับภาษาโปรแกรมพีเอสพี ซึ่งมักจะได้อีกว่าเป็นคู่ จะเห็นได้จากคู่มือคอมพิวเตอร์
 ต่างๆ ที่จะสอนการใช้งานมายเอสคิวแอล และพีเอสพี ควบคู่กันไป นอกจากนี้หลายภาษาโปรแกรมที่
 สามารถทำงานร่วมกับฐานข้อมูลมายเอสคิวแอล ซึ่งรวมถึง ภาษาซี ซีพลัสพลัส ปาสคาล ซีชาร์ป ภาษาจา
 วา ภาษาเพิร์ล พีเอชพี ไพทอน รูบี และภาษาอื่น ใช้งานผ่าน API สำหรับโปรแกรมที่ติดต่อผ่าน ODBC

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
 ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หรือ ส่วนเชื่อมต่อกับภาษาอื่น (database connector) เช่น เอเอสพี สามารถเรียกใช้มายเอสคิวแอล ผ่านทาง MyODBC,ADO,ADO.NET เป็นต้น

2.12 พีเอชพี (PHP)

(PHP) Hypertext Preprocessor หรือชื่อเดิม Personal Home Page คือ ภาษาคอมพิวเตอร์ใน ลักษณะเซิร์ฟเวอร์-ไซด์ สคริปต์ โดยลิขสิทธิ์อยู่ในลักษณะโอเพนซอร์ส ภาษาพีเอชพีใช้สำหรับจัดทำ เว็บไซต์ และแสดงผลออกมาในรูปแบบเอชทีเอ็มแอล(HTML) โดยมีรากฐานโครงสร้างคำสั่งมาจากภาษา ภาษาซี ภาษาจาวา และ ภาษาเพิร์ล ซึ่ง ภาษาพีเอชพี นั้นง่ายต่อการเรียนรู้ ซึ่งเป้าหมายหลักของภาษานี้คือ ให้นักพัฒนาเว็บไซต์สามารถเขียน เว็บเพจ ที่มีความตอบโต้ได้อย่างรวดเร็วของพีเอชพี

ตัวอย่างภาษาพีเอชพี

ภาษาพีเอชพี จะเป็นส่วนประกอบภายในเว็บเพจ โดยคำสั่งจะปรากฏระหว่าง <?php ... ?> เช่น

```
<?php
echo "Hello, World!";
?>
<?
echo "Hello World.";
?>
<SCRIPT LANGUAGE = 'php'>
echo "Hello World.";
</SCRIPT>
%
<%
echo "Hello World.";
%>
```

การแสดงผลของพีเอชพี จะปรากฏในลักษณะเอชทีเอ็มแอล ซึ่งจะไม่ต้องแสดงคำสั่งที่ผู้ใช้เขียน ซึ่งเป็นลักษณะเด่นที่พีเอชพีแตกต่างจากภาษาในลักษณะไคลเอนต์-ไซด์ สคริปต์ เช่น ภาษาจาวาสคริปต์ ที่ ผู้ชมเว็บไซต์สามารถอ่าน ดูและคัดลอกคำสั่งไปใช้เองได้ นอกจากนี้พีเอชพียังเป็นภาษาที่เรียนรู้และ เริ่มต้นได้ไม่ยาก โดยมีเครื่องมือช่วยเหลือและคู่มือที่สามารถหาอ่านได้ฟรีบนอินเทอร์เน็ต ความสามารถ การประมวลผลหลักของพีเอชพี ได้แก่ การสร้างเนื้อหาอัตโนมัติจัดการคำสั่ง การอ่านข้อมูลจากผู้ใช้และ ประมวลผล การอ่านข้อมูลจากคาส์เบส ความสามารถจัดการกับคุกกี้ ซึ่งทำงานเช่นเดียวกับโปรแกรมใน ลักษณะ CGI คุณสมบัติอื่นเช่น การประมวลผลตามบรรทัดคำสั่ง (command line scripting) ทำให้ผู้เขียน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โปรแกรมสร้างสคริปต์พีเอชพี ทำงานผ่านพีเอชพี พาร์เซอร์ (PHP parser) โดยไม่ต้องผ่านเซิร์ฟเวอร์หรือ เบราวเซอร์ ซึ่งมีลักษณะเหมือนกับ Cron (ใน ยูนิกซ์หรือลินุกซ์) หรือ Task Scheduler (ในวินโดวส์) สคริปต์เหล่านี้สามารถนำไปใช้ในแบบ Simple text processing tasks ได้

การแสดงผลของพีเอชพี ถึงแม้ว่าจุดประสงค์หลักใช้ในการแสดงผลเอชทีเอ็มแอล แต่ยังสามารถ สร้างเอ็กซ์เอชทีเอ็มแอล (XHTML) หรือเอ็กซ์เอ็มแอล (XML) ได้ นอกจากนี้สามารถทำงานร่วมกับคำสั่ง เสริมต่างๆ ซึ่งสามารถแสดงผลข้อมูลหลักพีดีเอฟ (PDF) แฟลช (โดยใช้ libswf และ Ming) พีเอชพีมี ความสามารถอย่างมากในการทำงานเป็นประมวลผลข้อความ จาก POSIX Extended หรือ รูปแบบ Perl ทั่วไป เพื่อแปลงเป็นเอกสารเอ็กซ์เอ็มแอล ในการแปลงและเข้าสู่เอกสารเอ็กซ์เอ็มแอล เรารองรับ มาตรฐานเอสเอเอ็กซ์ (SAX) และดีโอเอ็ม (DOM) สามารถใช้รูปแบบเอ็กซ์เอสแอลที (XSLT) ของเรา เพื่อแปลงเอกสารเอ็กซ์เอ็มแอล เมื่อใช้พีเอชพีในการทำอีคอมเมิร์ซ สามารถทำงานร่วมกับโปรแกรมอื่น เช่น Cybercash payment, CyberMUT, VeriSign Payflow Pro และ CCVS functions เพื่อใช้ในการสร้าง โปรแกรมทำธุรกรรมทางการเงิน

คำสั่งของพีเอชพี สามารถสร้างผ่านทางโปรแกรมแก้ไขข้อความทั่วไป เช่น โน้ตแพด หรือ vi ซึ่ง ทำให้การทำงานพีเอชพี สามารถทำงานได้ในระบบปฏิบัติการหลักเกือบทั้งหมด โดยเมื่อเขียนคำสั่งแล้ว นำมาประมวลผล Apache, Microsoft Internet Information Services (IIS), Personal Web Server, Netscape และ iPlanet servers, O'Reilly Website Pro server, Caudium, Xitami, OmniHTTPd, และอื่นๆ อีกมากมาย. สำหรับส่วนหลักของพีเอชพี ยังมีโมดูล (Module) ในการรองรับซีจีไอ (CGI) มาตรฐาน ซึ่งพีเอชพี สามารถ ทำงานเป็นตัวประมวลผลซีจีไอด้วย และด้วยพีเอชพี, คุณมีอิสรภาพในการเลือก ระบบปฏิบัติการ และ เว็บ เซิร์ฟเวอร์ นอกจากนี้คุณยังสามารถใช้สร้างโปรแกรมโครงสร้าง สร้างโปรแกรมเชิงวัตถุ (OOP) หรือสร้าง โปรแกรมที่รวมทั้งสองอย่างเข้าด้วยกัน แม้ความสามารถของคำสั่ง โปรแกรมเชิงวัตถุ มาตรฐานใน เวอร์ชันนี้ยังไม่สมบูรณ์ แต่ตัวไลบรารีทั้งหลายของ โปรแกรม และตัวโปรแกรมประยุกต์ (รวมถึง PEAR library) ได้ถูกเขียนขึ้นโดยใช้รูปแบบการเขียนแบบ โปรแกรมเชิงวัตถุเท่านั้น พีเอชพีสามารถทำงาน ร่วมกับฐานข้อมูลได้หลายชนิด ซึ่งฐานข้อมูลส่วนหนึ่งที่รองรับ ได้แก่ ออราเคิล dBase PostgreSQL IBM DB2 MySQL Informix ODBC โครงสร้างของฐานข้อมูลแบบดบีเอ็กซ์ (DBX) ซึ่งทำให้พีเอชพีใช้กับ ฐานข้อมูลอะไรก็ได้ที่รองรับรูปแบบนี้ และพีเอชพี ยังรองรับ ODBC (Open Database Connection) ซึ่งเป็น มาตรฐานการเชื่อมต่อฐานข้อมูลที่ใช้กันแพร่หลายอีกด้วย คุณสามารถเชื่อมต่อกับฐานข้อมูลต่างๆ ที่ รองรับมาตรฐานโลกนี้ได้ พีเอชพียังสามารถรองรับการสื่อสารกับการบริการในโพรโทคอลต่างๆ เช่น LDAP IMAP SNMP NNTP POP3 HTTP COM (บนวินโดวส์) และอื่นๆ อีกมากมาย คุณสามารถเปิด Socket บนเครือข่ายโดยตรง และ ตอบโต้โดยใช้ โพรโทคอลใดๆ ก็ได้พีเอชพี มีการรองรับสำหรับการ แลกเปลี่ยนข้อมูลแบบ WDDX Complex กับ Web Programming อื่นๆ ทั่วไปได้ พุดถึงในส่วน Interconnection, พีเอชพีมีการรองรับสำหรับ Java objects ให้เปลี่ยนมันเป็น PHP Object แล้วใช้งาน คุณยัง สามารถใช้รูปแบบ CORBA เพื่อเข้าสู่ Remote Object ได้เช่นกัน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.13 จาวาสคริปต์ (JavaScript)

จาวาสคริปต์ (JavaScript) เป็นภาษาสคริปต์ ที่มีลักษณะการเขียนแบบโปรโตไทป์ (Prototyped-based Programming) ส่วนมากใช้ในหน้าเว็บเพื่อประมวลผลข้อมูลที่ฝั่งของผู้ใช้งาน แต่ก็ยังมีใช้เพื่อเพิ่มเติมความสามารถในการเขียนสคริปต์โดยฝั่งอยู่ในโปรแกรมอื่นๆ เช่น ไมโครซิสเต็มส์เป็นเจ้าของเครื่องหมายการค้า "JavaScript" โดยมันถูกนำไปใช้ภายใต้สัญญาอนุญาตเพื่อการพัฒนาเทคโนโลยีโดย

ประวัติ เริ่มพัฒนาโดย Brendan Eich พนักงานบริษัทเน็ตสเคป โดยขณะนั้นจาวาสคริปต์ใช้ชื่อว่า โมคา และภายหลังได้เปลี่ยนชื่อมาเป็น โลฟีสคริปต์ และเป็น จาวาสคริปต์ในปัจจุบัน รูปแบบการเขียนภาษาที่ใช้ คล้ายคลึงกับภาษาซี รุ่นล่าสุดของจาวาสคริปต์คือ 2.0 ซึ่งตรงกับมาตรฐานของ ECMAScript ภาษาจาวาสคริปต์ไม่มีความสัมพันธ์กับ ภาษาจาวา (Java) และ เจสคริปต์ (JScript) แต่อย่างใด ยกเว้นแต่โครงสร้างภาษาที่มีลักษณะคล้ายคลึงกัน เนื่องมาจากได้รับการพัฒนาต่อมาจากภาษาซีเหมือนกัน และมีชื่อที่คล้ายคลึงกันเท่านั้น

สำหรับเจสคริปต์ (JScript) หลังจากที่จาวาสคริปต์ประสบความสำเร็จ โดยมีเว็บเบราว์เซอร์จากหลายๆ บริษัทนำมาใช้งาน ทางไมโครซอฟท์จึงได้พัฒนาภาษาโปรแกรมที่ทำงานในลักษณะคล้ายคลึงกับจาวาสคริปต์ขึ้น และตั้งชื่อว่าเจสคริปต์ ซึ่งทำงานได้กับเบราว์เซอร์อินเทอร์เน็ตเอกซ์พลอเรอร์ (Internet Explorer) เท่านั้น เริ่มใช้ครั้งแรกใน อินเทอร์เน็ตเอกซ์พลอเรอร์ 3.0 เมื่อ สิงหาคม พ.ศ. 2539 โดยสร้างตามมาตรฐาน ECMA 262

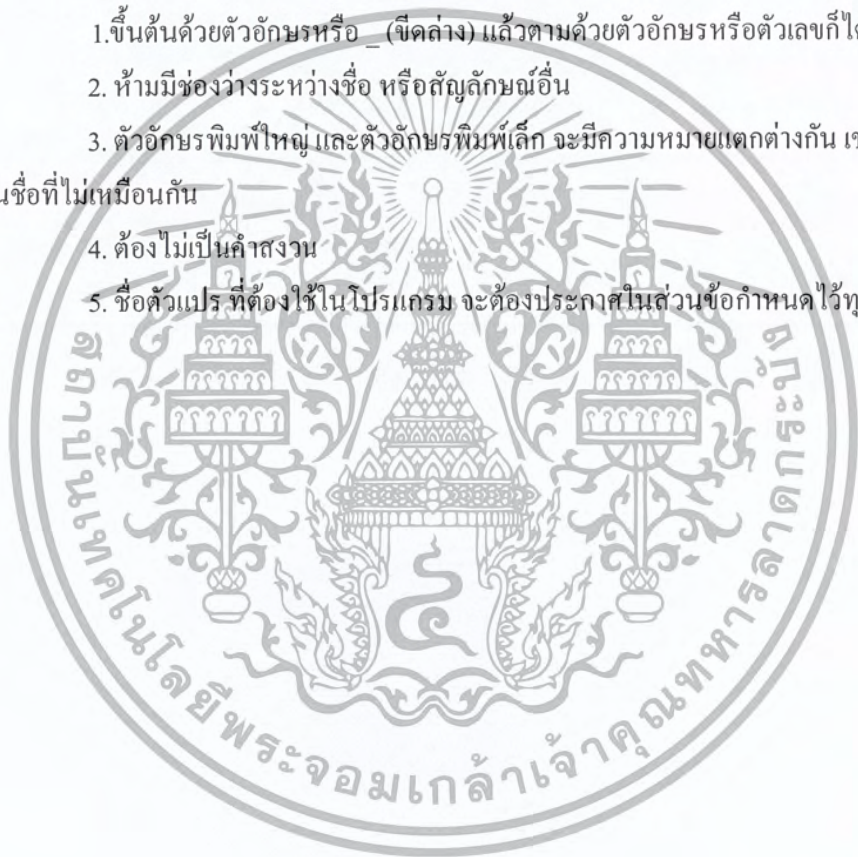
การใช้งาน จาวาสคริปต์ เป็นภาษาในรูปแบบของภาษาโปรแกรมแบบโปรโตไทป์ โดยมีโครงสร้างของภาษาและไวยากรณ์อยู่บนพื้นฐานของภาษาซี ปัจจุบันมีการใช้จาวาสคริปต์ที่ฝังอยู่ในเว็บเบราว์เซอร์ในหลายรูปแบบ เช่น ใช้เพื่อสร้างเนื้อหาที่เปลี่ยนแปลงเสมอภายในเว็บเพจ, ใช้เพื่อตรวจสอบความถูกต้องของข้อมูลที่ผู้ใช้กรอกก่อนนำเข้าระบบ, ใช้เพื่อเข้าถึงข้อมูลที่อยู่ภายใต้โครงสร้างแบบ Document Object Model (DOM) เป็นต้น

นอกจากนี้จาวาสคริปต์ยังถูกฝังอยู่ในแอปพลิเคชันต่างๆ นอกเหนือจากเว็บเบราว์เซอร์ได้อีกด้วย เช่น widget ของ ยาฮู! เป็นต้น โดยรวมแล้วจาวาสคริปต์ถูกใช้เพื่อให้นักพัฒนาโปรแกรม สามารถเขียนสคริปต์เพื่อสร้างคุณสมบัติพิเศษต่างๆ เพิ่มเติมจากที่มีอยู่บนแอปพลิเคชันดั้งเดิม

โปรแกรมใดๆ ที่สนับสนุนจาวาสคริปต์จะมีตัวขับเคลื่อนจาวาสคริปต์ (JavaScript Engine) ของตัวเอง เพื่อเรียกใช้งานโครงสร้างเชิงวัตถุของโปรแกรมหรือแอปพลิเคชันนั้นๆ เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตัวแปร คือ ชื่อที่ตั้งขึ้นเพื่อใช้เป็นชื่อของหน่วยความจำที่ต้องการเก็บข้อมูล เพื่อให้สามารถนำข้อมูลนั้นกลับมาใช้ได้อีกตามที่ต้องการ ตัวแปรที่ต้องการใช้ในโปรแกรม จะต้องประกาศไว้ในโปรแกรมทุกครั้ง ตัวแปรที่กำหนดขึ้นสามารถเก็บข้อมูลได้เพียงค่าเดียวเท่านั้น เมื่อมีการกำหนดข้อมูลซ้ำกันในชื่อตัวแปรเดียวกัน ข้อมูลที่กำหนดครั้งหลังสุดจะถูกเก็บแทนที่ข้อมูลเดิม รูปแบบของการประกาศตัวแปรเพื่อใช้ในโปรแกรมนี้นี้ VAR ชื่อตัวแปร; หรือ VAR ชื่อตัวแปร = ค่าข้อมูล; ชื่อตัวแปร หมายถึงตัวแปรที่ประกาศไว้ในส่วนการกำหนดตัวแปร เพื่อใช้เก็บข้อมูล ค่าข้อมูล หมายถึงค่าข้อมูลที่ต้องการนำไปเก็บไว้ที่ชื่อตัวแปรที่กำหนด = หมายถึงการนำค่าที่อยู่ด้านขวามือไปเก็บไว้ในชื่อตัวแปรที่อยู่ด้านซ้ายมือ หรือ หมายถึงการกำหนดค่าให้กับตัวแปร กฎเกณฑ์ในการตั้งชื่อตัวแปร เพื่อใช้ในโปรแกรม จะมีหลักในการตั้งชื่อตัวแปรดังนี้

1. เริ่มต้นด้วยตัวอักษรหรือ _ (ขีดล่าง) แล้วตามด้วยตัวอักษรหรือตัวเลขก็ได้
2. ห้ามมีช่องว่างระหว่างชื่อ หรือสัญลักษณ์อื่น
3. ตัวอักษรพิมพ์ใหญ่ และตัวอักษรพิมพ์เล็ก จะมีความหมายแตกต่างกัน เช่น name กับ Name เป็นชื่อที่ไม่เหมือนกัน
4. ต้องไม่เป็นคำสงวน
5. ชื่อตัวแปร ที่ต้องใช้ในโปรแกรม จะต้องประกาศในส่วนข้อกำหนดไว้ทุกครั้ง

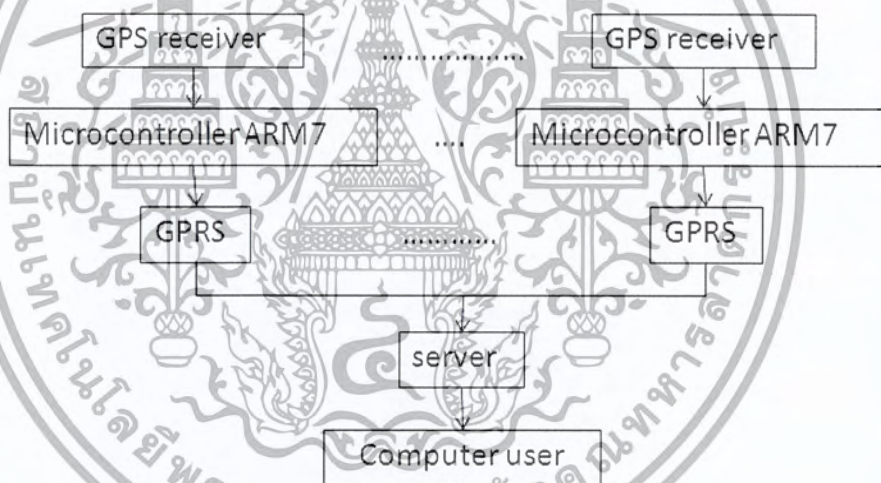


บทที่ 3

หลักการออกแบบและการสร้าง

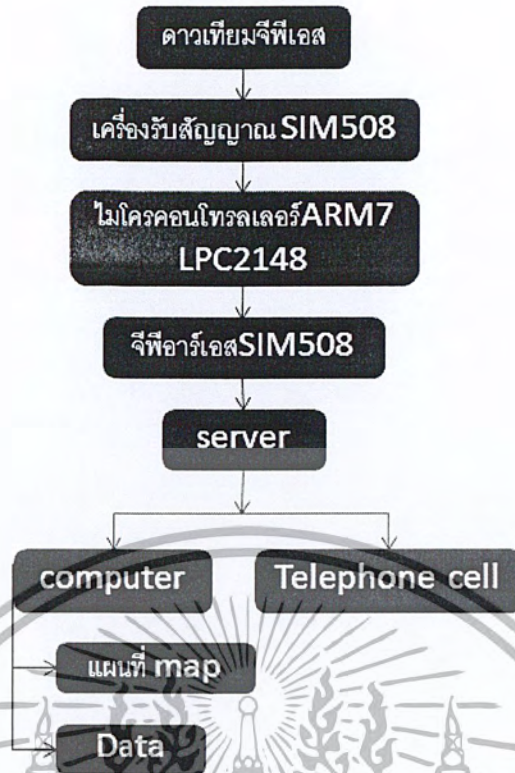
3.1 โครงสร้างการทำงานของระบบ

การทำงานของระบบ ส่วนจีพีเอสจะรับค่ามาจากดาวเทียมโดยเป็นโปรโตคอลเอ็นเอ็มอีเอ หลังจากนั้นจะนำข้อมูลที่ได้นั้นไปคัดกรองเอาเฉพาะประโยชน์จีพีอาร์เอ็มซี ซึ่งจะคัดกรองเอาตำแหน่งที่เป็น วันเดือนปี เวลา ละติจูด และความเร็ว การเชื่อมต่อส่วนจีพีเอสกับไมโครคอนโทรเลอร์อาร์ม7 LPC2148 และ จีพีอาร์เอส จะเชื่อมต่อกันแบบอนุกรม ซึ่งจีพีเอสกับจีพีอาร์เอสจะอยู่ในฮาร์ดแวร์เดียวกัน โดยไมโครคอนโทรเลอร์จะนำข้อมูลที่ได้จากจีพีเอส มาคัดกรองเอาข้อมูลที่ต้องการส่งไปยังจีพีอาร์เอสโมดูล และสุดท้ายส่งไปที่เซิร์ฟเวอร์ที่จะเก็บข้อมูลไว้ และแสดงผลเป็นแผนที่โดยแสดงออกที่เว็บเบราว์เซอร์(web browser) การแสดงผลบนแผนที่จะใช้จุดเกิดแผนที่ โดยดึงข้อมูลพิกัดจากฐานข้อมูล จากบล็อกรวมด้านล่างเราสามารถแสดงข้อมูลของยานพาหนะที่เราจะติดตามได้มากกว่าหนึ่งคัน โดยแสดงบนฐานข้อมูลเดียวกัน



รูปที่ 3.1 บล็อกรวมของระบบ

จากรูปที่ 3.2 เป็นการทำงานของระบบในส่วนย่อยๆของแต่ละระบบ โดยเริ่มจากการรับสัญญาณจากดาวเทียม ส่งเข้ามายังอุปกรณ์รับสัญญาณ SIM508 จากนั้นก็จะส่งไมโครคอนโทรเลอร์อาร์ม7 LPC2148 ทำการกรองข้อมูลแล้วส่งไปยังเซิร์ฟเวอร์โดยใช้จีพีอาร์เอส SIM508 ข้อมูลก็จะไปปรากฏอยู่บนเว็บเซิร์ฟเวอร์

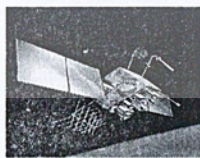


รูปที่ 3.2 โครงสร้างการทำงานของระบบ

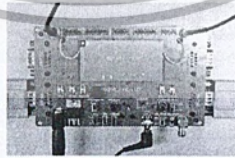
3.2 ส่วนของเครื่องรับสัญญาณ

3.2.1 เครื่องรับระบบพิกัดตำแหน่ง (GPS Receiver)

ส่วนประกอบของการรับสัญญาณตำแหน่งปัจจุบันประกอบด้วย 2 ส่วน คือ เครื่องรับ GPS และชุดรับส่งข้อมูลระหว่างเครื่องรับ GPRS และในการเชื่อมต่อตัวเครื่องรับ GPS เข้ากับระบบนั้นเราจะออกแบบให้ขา TX ของเครื่องรับสัญญาณ GPS ไปยังขา RX ของ Microship



ดาวเทียม



sim 508



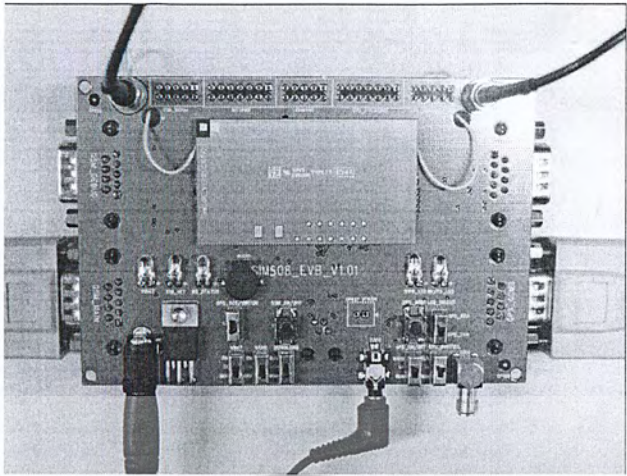
computer

รูปที่ 3.3 บล็อกการรับสัญญาณ GPS

โครงสร้างนี้ของระบบนี้จะถูกติดอยู่กับรถ โดยจะมีส่วนของการประมวลผลทั้งหมดและจัดการ

ข้อมูลต่างๆ โดยจะมีโครงสร้างโดยรวมแสดงดังรูป 3.4

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.4 เครื่องรับ GPS และ GPRS รุ่น SIM508_EVB

3.2.2 คุณสมบัติของ SIM508

- 1. ย่านความถี่ GSM 900, DCS 1800, PCS 1900 โดยปกติจะใช้ GSM 900 และ DSC 1800
- 2. Transmit power – Class 4 (2W) ที่ EGSM 900; Class 1 (1W) ที่ DCS 1800 และ PCS 1900
- 3. ข้อกำหนดเกี่ยวกับการส่งข้อมูล – อัตราในการส่งข้อมูลได้สูงสุด 85.6 Kbps, อัตราในการรับข้อมูลได้สูงสุด 42.8 Kbps, สนับสนุนโปรโตคอลแบบ PAP (Password Authentication Protocol)
- 4. ควบคุมการทำงาน GPRS โมดูลโดยใช้คำสั่ง AT Command
- 5. รองรับขั้วการวัดที่มีขนาด 1.8 V และ 3 V
- 6. ลักษณะทางกายภาพ มีขนาด 55±0.15×34±0.15×2.9±0.3 mm น้ำหนัก 11 กรัม

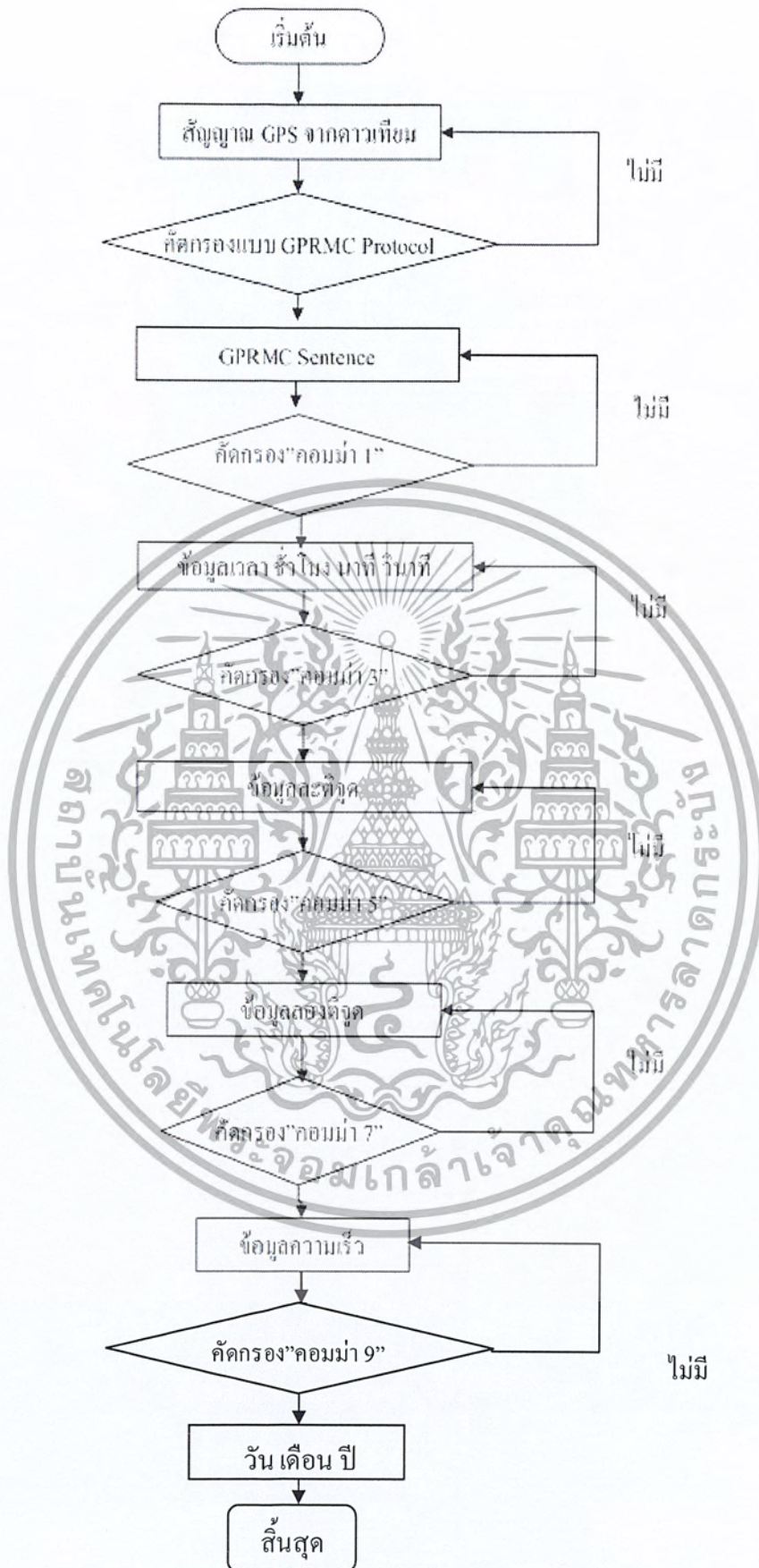
3.3 ส่วนของไมโครคอนโทรลเลอร์

ระบบนี้จะทำการนำข้อมูลที่ได้จากจีพีเอสโมดูลมาคัดกรอง (Filter) โดยข้อมูลที่ต้องการ มีดังนี้ 1.วันเดือนปี 2.เวลา ณ ปัจจุบัน 3.ละติจูด 4.ลองจิจูด 5.ความเร็ว ลำดับแรกคัดกรองสตริง RMC เพื่อหา GPRMC Sentence หลังจากนั้นคัดเอาเฉพาะหลังคอมม่าที่ 1 เป็นเวลาที่เป็นชั่วโมง นาที วินาที หลังคอมม่าที่ 3 เป็นละติจูด หลังคอมม่าที่ 5 เป็นลองจิจูด หลังคอมม่าที่ 7 เป็นความเร็วในหน่วยนอต (1 นอต = 1.825 กิโลเมตร) หลังคอมม่าที่ 9 เป็นวันเดือนปี แล้วเก็บค่าที่ได้มาในตัวแปร ดังนี้ dat คือ วันเดือนปี tim คือเวลา lat คือ ละติจูด lng คือลองจิจูด spd คือความเร็วเป็นไปตามแผนผังที่ 3.4



รูปที่ 3.5 บล็อกการทำงานส่วนไมโครคอนโทรลเลอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.6 แผนผังระบบการคัดกรองข้อมูลจากสัญญาณจีพีเอส

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

CP-JR ARM7 USB-LPC2148 เป็นบอร์ดไมโครคอนโทรลเลอร์ในตระกูล ARM7TDMI-S Core ซึ่งเลือกใช้ไมโครคอนโทรลเลอร์ 16/32-Bit ขนาด 64 Pin แบบใช้พลังงานต่ำเป็น MCU ประจำบอร์ด ซึ่งบอร์ดนี้เลือกใช้ MCU เบอร์ LPC2148 ของ Philips โดยการออกแบบโครงสร้างของบอร์ดนั้นจะเน้นเรื่องของ การจัดวางอุปกรณ์พื้นฐานที่จำเป็นต่อการศึกษาและทดลองขั้นพื้นฐาน อย่างเป็นสัดส่วน และเป็นอิสระต่อกันเช่น LED แสดงสถานะของ Output Logic และ Push Button Switch สำหรับสร้างสัญญาณ Logic เพื่อทดสอบการทำงานของ Input หรือ Volume ปรับค่าแรงดัน เพื่อใช้ทดสอบการทำงานของ A/D เป็นต้น โดยวงจรพื้นฐานต่างๆเหล่านี้จะถูกออกแบบและจัดวางเป็นจุดเชื่อมต่อแบบอิสระไว้ โดยยังไม่ได้อัปเดตสัญญาณเข้ากับขาสัญญาณใดๆของซีพียู(CPU) เพื่อให้ผู้ใช้สามารถปรับเลือกใช้งานวงจร ประกอบการทดลองที่มีอยู่ในบอร์ดให้สอดคล้องกับทรัพยากรภายในตัวซีพียู ได้อย่างเหมาะสมและเป็นอิสระ โดยไม่ยึดติดกับวงจรใดวงจรหนึ่งเพียงอย่างเดียว ตัวอย่างเช่น ขาสัญญาณ GPIO0.8 ของ LPC2148 สามารถเลือกกำหนดการทำงานได้มากถึง 4 แบบ คือ ใช้งานเป็น Input/Output, TXD1, PWM และ ADC ดังนั้นเมื่อผู้ใช้งานต้องการทดลองการใช้งานขา GPIO0.8 ในโหมดใดก็สามารถเลือกต่อสัญญาณจาก GPIO0.8 ไปยังวงจรสนับสนุนที่เหมาะสมได้ตามต้องการเช่น

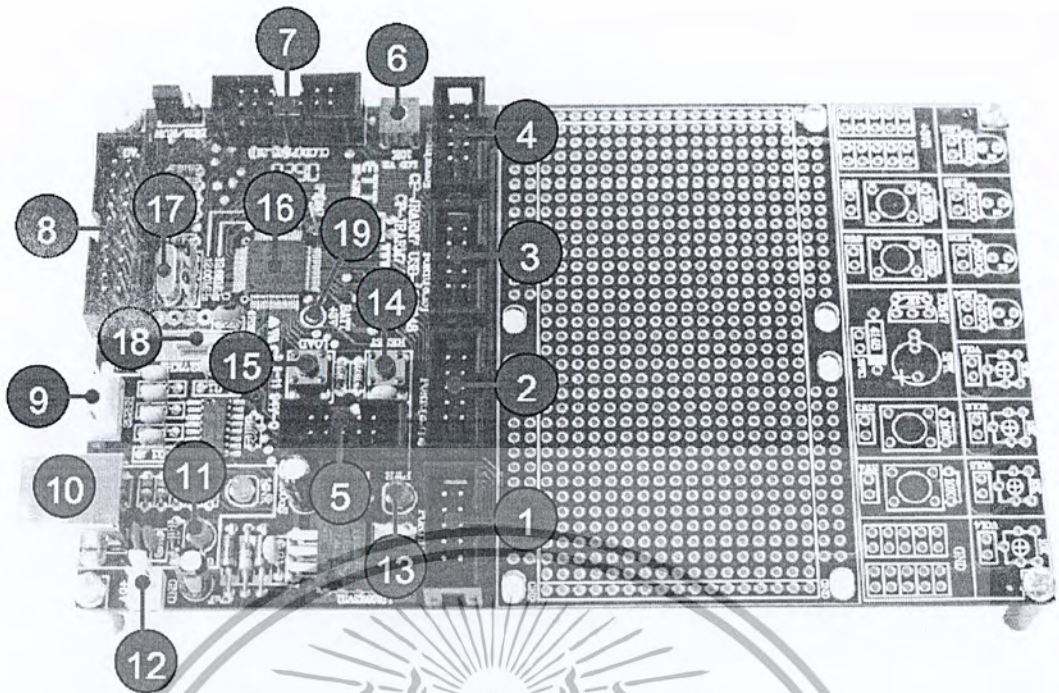
- เมื่อต้องการทดลองการใช้งาน GPIO0.8 เป็น Input ก็สามารถเลือกต่อสัญญาณจาก GPIO0.8 ไปยังวงจรของ Push Button Switch เพื่อทดลองการทำงานกับ Input
- เมื่อต้องการทดลองการใช้งาน GPIO0.8 เป็น Output ก็สามารถเลือกต่อสัญญาณจาก GPIO0.8 ไปยังวงจรของ LED เพื่อดูสถานะการทำงานของ Output
- เมื่อต้องการทดลองการใช้งาน GPIO0.8 เป็น ADC ก็สามารถเลือกต่อสัญญาณจาก GPIO0.8 ไปยังวงจรของ Volume ปรับค่าแรงดัน เพื่อทดสอบอ่านค่าแรงดัน Analog
- เมื่อต้องการทดลองการใช้งาน GPIO0.8 เป็น TXD1 ก็สามารถเลือกต่อสัญญาณจาก GPIO0.8 ไปยังวงจรของ Line Driver ของ RS232 เพื่อทดลองการส่งข้อมูลแบบ RS232

ซึ่งจะเห็นได้ว่าผู้ใช้สามารถเลือกกำหนดและปรับเปลี่ยนการใช้งานขาสัญญาณต่างๆของ CPU ได้อย่างครบถ้วนทุกฟังก์ชันการทำงานได้โดยอิสระอย่างแท้จริง โดยโครงสร้างของบอร์ดจะนำ MCU มาจัดวงจรร่วมกับอุปกรณ์พื้นฐานที่จำเป็นและจัดขาออกมาให้ใช้งานภายนอกผ่านหัวต่อ Connector แบบ IDE10 Pin ซึ่งการจัดเรียงขาสัญญาณจะทำการจัดเรียงอย่างเป็นระเบียบตามมาตรฐานของ อีทีที เพื่อให้สามารถต่อใช้งานร่วมกับอุปกรณ์สนับสนุนต่างๆของ อีทีที ได้โดยสะดวก ตัวบอร์ดใช้ไฟ +5VDC จากภายนอก หรือ USB Port (ในกรณี ใช้กระแสไม่เกิน 100 mA ขณะยังไม่ Enumerate USB และไม่เกิน 500 mA ขณะ Enumerate USB แล้ว) ซึ่งภายในบอร์ดจะมีวงจร Regulate ขนาด 3.3V/1A จัดเตรียมไว้ให้เรียบร้อยแล้ว และภายในบอร์ดจะมี Connector RS232(UART0) จำนวน 1 Port สำหรับทำการ Download Hex File หรือใช้งานในการสื่อสาร RS232 ในโปรแกรม Application ที่เขียนขึ้นเอง

คุณสมบัติของบอร์ด

1. ใช้ MCU ตระกูล ARM7TDMI-S เบอร์ LPC2148 ของ Philips ซึ่งเป็น MCU ขนาด 16/32-Bit
2. ใช้ Crystal 12.00 MHz โดย MCU สามารถประมวลผลด้วยความเร็วสูงสุดที่ 60 MHz เมื่อใช้งานร่วมกับ Phase-Locked Loop (PLL) ภายในตัว MCU เอง
3. รองรับการโปรแกรมแบบ In-System Programming (ISP) และ In-Application Programming (IAP) ผ่านทาง On-Chip Boot-Loader Software ทางพอร์ต UART0 (RS232)
4. Power Supply ใช้แรงดันไฟฟ้า +5VDC โดยใช้ ขั้วต่อแบบ CPA-2PIN จากภายนอก หรือ ใช้พลังงานจาก USB Port ได้ (ในกรณีใช้กระแสไม่เกิน 500mA)
5. ภายใน MCU มีหน่วยความจำโปรแกรมแบบ Flash ขนาด 512KB, Static RAM ขนาด 40KB
6. มีวงจร USB มาตรฐาน 2.0 แบบ Full Speed ภายในตัว (USB Function มี 32 End Point)
7. จำนวน GPIO สูงสุดถึง 47 I/O Pins สามารถเชื่อมต่อกับระบบ I/O ที่เป็นสัญญาณ 5V ได้ ซึ่งขาสัญญาณ GPIO จะมีการใช้งานร่วมกันของ Function อื่นๆอีกดังนี้
 - วงจรสื่อสารอนุกรมแบบ SPI จำนวน 2 ช่อง และ วงจรสื่อสารอนุกรมแบบ I2C จำนวน 2 ช่อง
 - วงจร ADC ขนาด 10 Bit จำนวน 14 ชุด และ วงจร DAC ขนาด 10 Bit จำนวน 1 ชุด
 - วงจร UART แบบ Full-Duplex จำนวน 2 ช่อง คือ UART-0 มาตรฐาน 4 Pin ETT เป็นสัญญาณระดับ RS232 Level และ UART-1 เป็นสัญญาณระดับ TTL Level
 - Timer 32-bit จำนวน 2 ช่อง (4 Input Capture / 4 Output Compare), 6-Channel PWM
 - Output, Watchdog Timer และ Real Time Clock
8. มีวงจรเชื่อมต่อกับ Character LCD โดยใช้วงจรการเชื่อมต่อแบบ 4 บิต จาก GPIO[25..31] พร้อมวงจรปรับความสว่างหน้าจอ
9. มีวงจรเชื่อมต่อกับ JTAG ARM ขนาด 20 Pin มาตรฐาน เพื่อทำการ Debug แบบ Real Time ได้
10. มีวงจรทดลองขั้นพื้นฐานสำหรับสนับสนุนการใช้งานและทดลองเรียนรู้ขั้นพื้นฐานอย่างครบถ้วน จัดเตรียมไว้ภายในบอร์ด (ติดตั้งไว้เฉพาะรุ่น CP-JR ARM7 USB-LPC2148 EXP) ซึ่ง ได้แก่
 - LED Output (แบบ Sink Current) สำหรับแสดงสถานะของ Output จำนวน 4 ชุด
 - Push Button Switch แบบ Active Logic "0" สำหรับทดสอบ Input Logic จำนวน 4 ชุด
 - Volume ปรับค่าแรงดัน 0-3.3V สำหรับทดสอบการทำงานของ ADC จำนวน 4 ชุด
 - ชุดกำเนิดสัญญาณเสียง Mini Speaker สำหรับทดสอบการเสียงแบบต่างๆ จำนวน 1 ชุด
 - แผงต่อวงจร Project Board รุ่น AD-100 ขนาด 360 จุด สำหรับเป็นพื้นที่ต่อทดลองวงจรขนาดเล็กๆ เพื่อใช้งานร่วมกับ CPU ได้อย่างอิสระ
 - จุดต่อแหล่งจ่ายไฟ +3.3V และ GND สำหรับเชื่อมต่อไปยังวงจรภายนอกอื่นๆ
11. ทนอุณหภูมิใช้งานระหว่าง -40 to +85°C

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



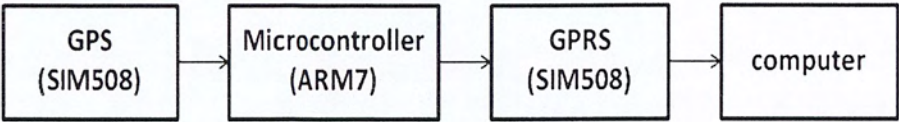
รูปที่ 3.7 ลักษณะของบอร์ด CP-JR ARM7 USB-LPC2148

- หมายเลข 1 คือ ขั้วต่อ Port1[16..23] จำนวน 8 บิต
- หมายเลข 2 คือ ขั้วต่อ Port0[2..7] จำนวน 6 บิต
- หมายเลข 3 คือ ขั้วต่อ Port0[8..15] จำนวน 8 บิต
- หมายเลข 4 คือ ขั้วต่อ Port0[16..23] จำนวน 8 บิต
- หมายเลข 5 คือ ขั้วต่อ Port0[25..31] จำนวน 7 บิต
- หมายเลข 6 คือ ตัวต้านทานสำหรับปรับค่าความสว่าง (Contrast) ของหน้าจอ LCD
- หมายเลข 7 คือ ขั้วต่อ Character LCD โดยใช้สัญญาณ Port1[25..31] ในการเชื่อมต่อ
- หมายเลข 8 คือ ขั้วต่อ JTAG โดยใช้สัญญาณ Port1[26..31] และ Reset ของ CPU
- หมายเลข 9 คือ ขั้วต่อ RS232 สำหรับใช้งาน และ Download Hex File ให้ CPU
- หมายเลข 10 คือ ขั้วต่อ USB สำหรับเชื่อมต่อกับ USB Hub รุ่น 2.0
- หมายเลข 11 คือ LED แสดงสถานะ Power จาก USB และ สถานะของการเชื่อมต่อกับ USB
- หมายเลข 12 คือ ขั้วต่อ Power ขนาด +5VDC และ GND เพื่อจ่ายให้กับบอร์ด
- หมายเลข 13 คือ LED แสดงสถานะของแหล่งจ่ายไฟ Power ของบอร์ด
- หมายเลข 14 คือ Switch RESET สำหรับสั่ง Reset การทำงานของ CPU
- หมายเลข 15 คือ Switch LOAD ใช้ร่วมกับ Switch RESET เพื่อ Download Hex ให้ CPU
- หมายเลข 16 คือ CPU เบอร์ LPC2148 ของ Philips ซึ่งเป็น CPU ประจำบอร์ด
- หมายเลข 17 คือ Crystal 12.00 MHz สำหรับป้อนให้เป็นสัญญาณนาฬิกาของ LPC2148
- หมายเลข 18 คือ Crystal 32.768 KHz สำหรับ Real Time Clock (RTC) ในตัวของ LPC2148

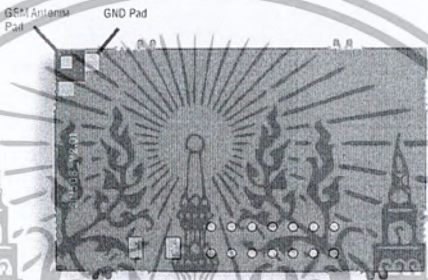
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- **หมายเลข 19** คือ จุดเชื่อมต่อด้าน Battery ขนาด +3V (อยู่ด้านใต้บอร์ด) สำหรับต่อให้กับ RTC เพื่อเก็บรักษาค่าเวลาของ RTC ในกรณีที่ไม่ได้จ่ายไฟเลี้ยงให้กับบอร์ด

3.4 ส่วนของการส่งข้อมูลผ่าน GPRS module



รูปที่ 3.8 บล็อกการส่งข้อมูลผ่าน GPRS ไปยัง SERVER



รูปที่ 3.9 GPRS module

ใช้คำสั่ง AT COMMAND ในการติดต่อกับระบบ GPRS

- คำสั่งการขอเปิดการติดต่อ GPRS

รูปแบบคำสั่ง AT+CGATT=<mode><CR>

ให้ตั้งค่าAT + CGATT = 1

; 0 คือ ไม่ใช้งานและ 1 คือใช้งาน

- คำสั่งติดต่อกับศูนย์บริการ

รูปแบบคำสั่ง AT#APNSERV= www.AIS.co.th<CR>

โดยเป็นการเริ่มเปิดการติดต่อกับระบบ GPRS

- คำสั่งเชื่อมต่อกลับเครือข่าย

รูปแบบคำสั่ง AT#TCPSE V = “161.246.xxx.xxx”<CR>

โดยเป็นการเชื่อมต่อกลับเครือข่ายผ่านทาง IP Address

- คำสั่งติดต่อทางพอร์ต

รูปแบบคำสั่ง AT#TCPPORT = “1000”<CR>

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โดยเป็นการส่งข้อมูลผ่านทางพอร์ต 1000 ของเครื่อง server

- คำสั่งเปิดการส่งข้อมูล/ปิดการส่งข้อมูล

รูปแบบคำสั่ง AT#OTCP<CR>

โดยเป็นการเริ่มส่งข้อมูลผ่านระบบ GPRS

- คำสั่งหยุดการติดต่อกับระบบเครือข่าย

รูปแบบคำสั่ง AT#CONNECTIONSTOP

โดย เป็นหยุดติดต่อกับระบบเครือข่าย

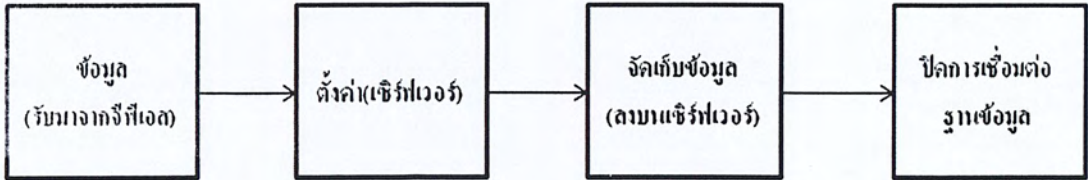
3.5 SERVER หรือฐานข้อมูล

ระบบการทำงานของฐานข้อมูล มีการสร้างข้อมูลของรถแต่ละคัน เมื่อได้ข้อมูลแล้วเราจะนำไปเก็บไว้บนเซิร์ฟเวอร์อันเดียวกัน เราจะสามารถดูข้อมูลจากที่เดียวกันได้



รูปที่ 3.10 บล็อกการทำงานของฐานข้อมูล

การสร้างฐานข้อมูลบนเว็บเซิร์ฟเวอร์เมื่อเราต้องการเก็บและเรียกดูข้อมูลต่างๆ จะรับข้อมูลที่ได้ไปเก็บไว้บนเซิร์ฟเวอร์ โดยจะมีการตั้งค่าต่างๆบนเซิร์ฟเวอร์ก่อนที่จะนำข้อมูลไปเก็บไว้ ดังบล็อกการทำงานด้านล่าง



รูปที่3.11 บล็อกการทำงานของฐานข้อมูล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ในการรับข้อมูลจะต้องประกอบไปด้วยเซิร์ฟเวอร์ที่มีฐานข้อมูลใช้ในการจัดเก็บข้อมูล เซิร์ฟเวอร์ที่ใช้ในการเป็นแบบ free server ได้ทำการสมัครดังนี้

1. ลงทะเบียนเป็นสมาชิกที่ <http://www.freetzi.com>
2. สร้างตาราง gpsa ในฐานข้อมูล 272548
3. อัปโหลดลงบนเซิร์ฟเวอร์ของเรา



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 4

การทดลองและผลการทดลอง

4.1 การทดลองส่วน GPS Module

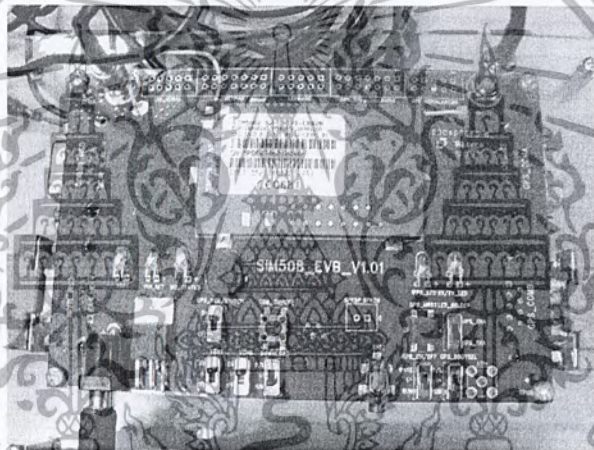
การทดลอง

จุดประสงค์การทดลอง

- 1.เพื่อดูว่าข้อมูลที่รับมาจากเครื่องรับ GPS Module มีลักษณะเป็นอย่างไร

ขั้นตอนการทดลอง

- 1.ต่อ GPS Module เข้ากับเครื่องคอมพิวเตอร์ผ่านทางพอร์ตอนุกรม
- 2.ใช้โปรแกรมไฮเปอร์เทอร์มินอล(Hyper Terminal) เพื่อดูว่าข้อมูลที่รับมาจะมีลักษณะเป็นอย่างไร



รูปที่ 4.1 รูปเครื่อง GPS Module

ผลการทดลอง

```
$GPRMC,185045.600,A,1343.6533,N,10046.5540,E,3.98,358.07,160211,,A*69
$GPGGA,185046.599,1343.6557,N,10046.5538,E,1.03,4.8,-17.8,M,-27.2,M,0000*60
$GPGSA,A,2,07,08,19,,,,,4.9,4.8,1.0*34
$GPGSV,3,1,12,11,68,112,16,07,60,348,24,24,52,123,17,13,42,214,19*77
$GPGSV,3,2,12,19,28,028,30,08,28,332,20,23,22,179,,17,22,238,*73
$GPGSV,3,3,12,28,16,305,17,20,10,162,,32,03,138,,03,02,043,23*7E
$GPRMC,185046.599,A,1343.6557,N,10046.5538,E,4.19,3.65,160211,,A*60
$GPGGA,185047.603,1343.6570,N,10046.5545,E,1.03,4.8,-17.9,M,-27.2,M,0000*62
$GPGSA,A,2,07,08,19,,,,,4.9,4.8,1.0*34
$GPRMC,185047.603,A,1343.6570,N,10046.5545,E,3.94,9.63,160211,,A*60
$GPGGA,185048.603,1343.6578,N,10046.5554,E,1.03,4.8,-18.1,M,-27.2,M,0000*62
$GPGSA,A,2,07,08,19,,,,,4.9,4.8,1.0*34
$GPRMC,185048.603,A,1343.6578,N,10046.5554,E,3.41,13.35,160211,,A*57
$GPGGA,185049.603,1343.6606,N,10046.5555,E,1.03,4.8,-18.1,M,-27.2,M,0000*68
$GPGSA,A,2,07,08,19,,,,,4.9,4.8,1.0*34
$GPRMC,185049.603,A,1343.6606,N,10046.5555,E,3.89,7.13,160211,,A*68
$GPGGA,185050.599,1343.6595,N,10046.5561,E,1.03,4.8,-17.9,M,-27.2,M,0000*69
$GPGSA,A,2,07,08,19,,,,,4.9,4.8,1.0*34
$GPRMC,185050.599,A,1343.6595,N,10046.5561,E,2.98,8.33,160211,,A*62
$GPGGA,185051.603,1343.6568,N,10046.5560,E,1.03,4.8,-17.7,M,-27.2,M,0000*65
$GPGSA,A,2,07,08,19,,,,,4.9,4.8,1.0*34
```

รูปที่ 4.2 ข้อมูลที่ส่งออกมาจากเครื่องรับ GPS เมื่อเชื่อมต่อกับคอมพิวเตอร์ผ่านทางพอร์ตอนุกรม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สรุปผลการทดลอง

จากการทดลองจะได้ข้อมูลที่ส่งมาอยู่ในรูปประโยคของ NMEA ซึ่งข้อมูลที่เราส่งใจและครบหมดทุกอย่างก็จะอยู่ในบรรทัด RMC ซึ่งจากการทดลองบรรทัดที่เลือกก็คือ

\$GPRMC,185049.603,A,1343.6606,N,10046.5555,E,3.89,7.13,160211...A*68

คำอธิบาย GPRMC คือ ใช้ protocol RMC

- 185049.000 คือ เวลาปัจจุบันในระบบยูทีซี
- A คือ สถานะข้อมูล
- 1343.6602 คือ ละติจูด
- N คือ ทิศเหนือหรือใต้ (N or S)
- 10046.5555 คือ ลองจิจูด
- E คือ ทิศตะวันออกหรือ ตก (E or W)
- 3.89 คือ ความเร็ว นอต
- 160211 คือ วัน เดือน ปี

4.2 การทดลองเปรียบเทียบการรับสัญญาณดาวเทียมด้วยเครื่องรับจีพีเอส

การทดลอง

จุดประสงค์การทดลอง

- 1.เพื่อเปรียบเทียบความคลาดเคลื่อนของเครื่องรับสัญญาณจีพีเอส

ขั้นตอนการทดลอง

- 1.นำเครื่องรับสัญญาณจีพีเอสมาต่อเข้ากับคอมพิวเตอร์รับสัญญาณ โดยรับสัญญาณในตำแหน่งเดียวกัน ทำการทดลองที่บริเวณตึกภาคโทรคมนาคม ชั้นที่ 2 ห้อง T-211 เป็นประจำ ทดลองทั้งหมด 20 วัน
- 2.ใช้โปรแกรมไฮเปอร์เทอร์มินอล เพื่อดูว่าข้อมูลที่รับมาจะมีลักษณะเป็นอย่างไร

ผลการทดลอง

ตารางที่4.1 ผลการทดลองรับสัญญาณจีพีเอสในแต่ละวัน

ลำดับ	วัน/เดือน/ปี	ละติจูด	ลองจิจูด
1	07/01/2011	13.728267771	100.776418000
2	08/01/2011	13.728265518	100.776343222
3	09/01/2011	13.728400121	100.776441400

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4	11/01/2011	13.728265000	100.776418000
5	12/01/2011	13.728265220	100.776445444
6	13/01/2011	13.722515011	100.776560000
7	15/01/2011	13.728850001	100.776400870
8	17/01/2011	13.728266678	100.775600100
9	18/01/2011	13.728353530	100.776440015
10	19/01/2011	13.728200100	100.777700100
11	20/01/2011	13.728266666	100.776001000
12	21/01/2011	13.728445404	100.776386010
13	25/01/2011	13.728222022	100.776304011
14	26/01/2011	13.728102207	100.777872000
15	27/01/2011	13.728200000	100.776200178
16	28/01/2011	13.730019801	100.775622001
17	29/01/2011	13.729902000	100.776423230
18	30/01/2011	13.728233232	100.776400000
19	31/01/2011	13.728274700	100.776022222
20	01/02/2011	13.728268601	100.776424442
	เฉลี่ย	13.72817918	100.776421100

สรุปผลการทดลอง

จากการนำเครื่องรับสัญญาณจีพีเอส SIM508 มารับสัญญาณที่ตำแหน่งเดิม บริเวณตึกภาค
โทรคมนาคม สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง เป็นระยะเวลาติดต่อกัน โดยตั้งค่า
เครื่อง เซตเครื่องเหมือนเดิมทุกครั้ง ผลที่ได้ดังตารางการทดลองค่าละติจูดเฉลี่ยได้ เท่ากับ 13.72817918 ค่า
ลองจิจูดเฉลี่ยเท่ากับ 100.776421100 จะเห็นว่าถึงแม้เราจะทำการรับสัญญาณในตำแหน่งเดิม แต่ค่าผลการ
ทดลองที่ได้มีค่าไม่เท่ากัน บ้างครั้งได้ค่าที่เหมือนกันแต่บ้างครั้งก็ได้ค่าที่ต่างออกไป เมื่อเรานำตำแหน่งที่
เราทำการทดลองไปค้นในกูเกิ้ลแมพได้ ละติจูดเท่ากับ 13.727785000 ลองจิจูดเท่ากับ 100.776386000
ดังนั้นทำให้ทราบว่าเครื่องรับจีพีเอส รับตำแหน่งคลาดเคลื่อนไป 0.003 % ถ้าคิดเป็นระยะทางประมาณ 5
เมตรจากค่าที่ทำการทดลอง

4.3 การทดลองการกรองข้อมูลด้วย ARM7

การทดลอง

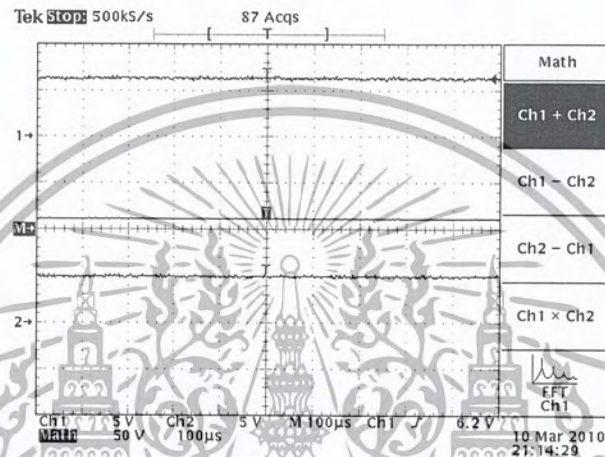
จุดประสงค์การทดลอง

1. เพื่อรู้ว่าผลที่ผ่านการกรองข้อมูลด้วย ARM7 มีลักษณะเป็นอย่างไร

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ขั้นตอนการทดลอง

1. ต่อ GPS Module เข้ากับเครื่องคอมพิวเตอร์ผ่านทางพอร์ตอนุกรม
2. ใช้โปรแกรม Hyper Terminal เพื่อดูว่าข้อมูลที่รับมาจะมีลักษณะเป็นอย่างไร
3. ใช้ไมโครคอนโทรลเลอร์ตัดเอาเฉพาะข้อมูลที่ต้องการ ไมโครคอนโทรลเลอร์ใช้แรงดันไฟฟ้า 5 V จึงต้องสร้างวงจรแปลงแรงดันจากแบตเตอรี่ 6V ใช้เครื่องวัดสัญญาณเทียบสัญญาณขาเข้าและสัญญาณขาออกได้ดังนี้



รูปที่ 4.3 แสดงผลเทียบระหว่างสัญญาณขาเข้า 6V กับสัญญาณขาออก 5V

ผลการทดลอง

```

$GPRMC,185045.603,A,1343.6593,N,10046.5540,E,3.98,358.07,160211,,,A*69
$GPGGA,185046.599,1343.6557,N,10046.5538,E,1.03,4.8,-17.8,M,-27.2,M,0000*6D
$GPGSA,A,2,07,08,19,,,,,4.9,4.8,1.0*34
$GPGSV,3,1,12,11,68,112,16,07,60,348,24,24,52,123,17,13,42,214,19*77
$GPGSV,3,2,12,19,28,028,30,08,28,332,20,23,22,179,17,22,238,*73
$GPGSV,3,3,12,28,16,305,17,20,10,162,32,03,138,03,02,043,23*7E
$GPRMC,185046.599,A,1343.6557,N,10046.5538,E,4.19,3.65,160211,,,A*6D
$GPGGA,185047.603,1343.6570,N,10046.5545,E,1.03,4.8,-17.9,M,-27.2,M,0000*62
$GPGSA,A,2,07,08,19,,,,,4.9,4.8,1.0*34
$GPRMC,185047.603,A,1343.6570,N,10046.5545,E,3.94,9.63,160211,,,A*60
$GPGGA,185048.603,1343.6578,N,10046.5554,E,1.03,4.8,-18.1,M,-27.2,M,0000*62
$GPGSA,A,2,07,08,19,,,,,4.9,4.8,1.0*34
$GPRMC,185048.603,A,1343.6578,N,10046.5554,E,3.41,13.35,160211,,,A*57
$GPGGA,185049.603,1343.6606,N,10046.5555,E,1.03,4.8,-18.1,M,-27.2,M,0000*68
$GPGSA,A,2,07,08,19,,,,,4.9,4.8,1.0*34
$GPRMC,185049.603,A,1343.6606,N,10046.5555,E,3.89,7.13,160211,,,A*68
$GPGGA,185050.599,1343.6595,N,10046.5561,E,1.03,4.8,-17.9,M,-27.2,M,0000*69
$GPGSA,A,2,07,08,19,,,,,4.9,4.8,1.0*34
$GPRMC,185050.599,A,1343.6595,N,10046.5561,E,2.98,8.33,160211,,,A*62
$GPGGA,185051.603,1343.6568,N,10046.5560,E,1.03,4.8,-17.7,M,-27.2,M,0000*65
  
```

รูปที่ 4.4 ข้อมูลที่ยังไม่ได้ทำการกรองจากไมโครคอนโทรลเลอร์ ARM7

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
AT+CIPSTART="TCP","mpgps.freetzi.com","80"
SHUT OK
```

OK

+PDP: DEACT

CONNECT OK

AT+CIPSEND

```
> GET /conv1.php?Dat=260211&Time=005536.000&Lat=13.436585&Lng=100.465671&Spd=1.0
8 HTTP/1.1
host:mpgps.freetzi.com
```

SEND OK

AT+CIPSHUT

SHUT OK

AT+CIPSTART="TCP","mpgps.freetzi.com","80"

OK

+PDP: DEACT

CONNECT OK

รูปที่ 4.5 ข้อมูลที่ทำการกรองจากไมโครคอนโทรลเลอร์ ARM7

สรุปผลการทดลอง

จากการทดลองจะได้ข้อมูลที่ส่งมาอยู่ในรูปประโยคของ NMEA ซึ่งข้อมูลที่ได้จากการทดลองในการทดลองนี้เมื่อผ่านกรองด้วย ARM7 จะบอกวันเดือนปี เวลา ละติจูด ลองจิจูด และความเร็ว ข้อมูลที่รับมามีความหมายดังนี้ Dat = 260211 แสดงวันที่ 26 เดือนกุมภาพันธ์ ค.ศ. 2011 Tim = 005536.000 แสดงว่าตอนนี้เวลา 24 นาฬิกา 55 นาที 36 วินาที Lat = 13.436585 ละติจูดที่ 13.46585 Lng = 100.465671 ลองจิจูดที่ Lng = 100.465671 และ Spd = 1.00 ความเร็วเท่ากับ 1.00 นอต ข้อมูลในที่นี้เป็นเวลาสากลความเร็วเป็นหน่วยนอต และขณะทำการทดลองไม่ได้มีการเคลื่อนที่ แสดงว่าในการทดลองมีการคาดเคลื่อน การทดลองนี้เป็นการคัดกรองข้อมูลจีพีเอสโดยเฉพาะข้อมูลที่ต้องการจากข้อมูลจีพีเอสที่ประโยค RMC ที่รับได้

4.4 การใช้คำสั่ง AT Command กับโมดูลจีพีอาร์เอส

โมดูล SIM508 มีส่วนที่เป็นจีพีเอสและจีพีอาร์เอส การใช้จีพีอาร์เอสจะใช้คำสั่งที่เรียกว่า AT Command ในการสั่งการจีพีอาร์เอส โดยจะใช้คำสั่งในโครงงานนี้ดังนี้

```
CONNECT OK
AT+CIPSEND
> GET /conv1.php?Dat=210211&Time=022714.000&Lat=13.436698&Lng=100.465682&Spd=0.2
6 HTTP/1.0
host:mpgps.freetzi.com

SEND OK
HTTP/1.0 200 OK
Date: Sun, 20 Feb 2011 19:27:40 GMT
Server: Apache
X-Powered-By: PHP/5.2.13
Content-Length: 7
Content-Type: text/html
X-Cache: MISS from paul-suk3
Via: 1.1 paul-suk3:3128 (squid/2.7.STABLE9)
Connection: close

ok<br>
CLOSED
AT+CIPSHUT
SHUT OK
AT+CIPSTART="TCP","mpgps.freetzi.com","80"
```

รูปที่ 4.6 คำสั่ง AT Command

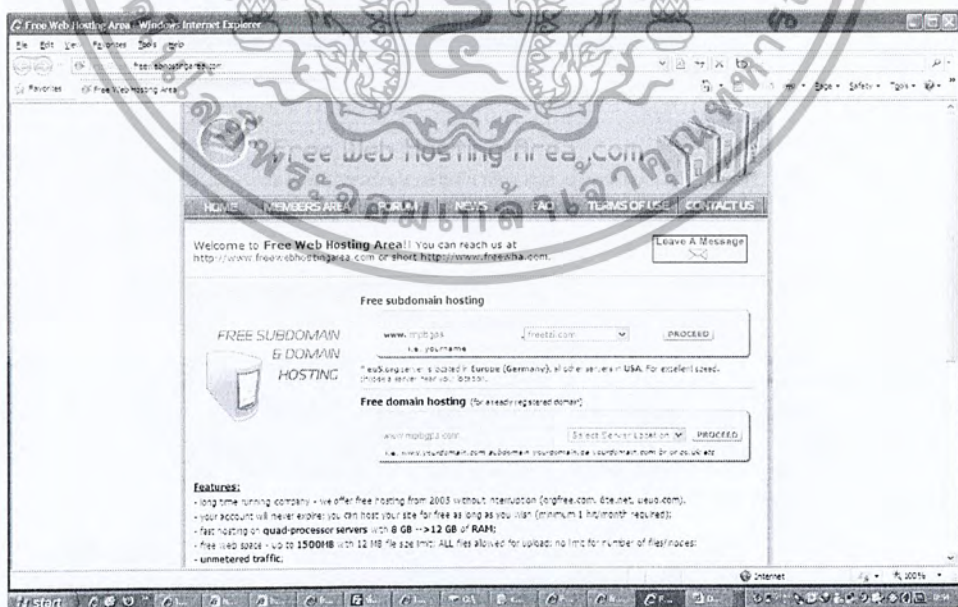
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

คำสั่ง Network registration คือ AT+CERG=1 ผลออกมาคือ OK แสดงว่าสามารถใช้ซิมการ์ดนี้ได้ เป็นซิมการ์ดที่สามารถติดต่อกับเครือข่ายได้ คำสั่ง Define PDP (Packet data Protocol) context คือ AT+CGDCONT=1,"IP",www.ais.co.th ผลออกมาคือ OK สามารถใช้ IP protocol ได้ คำสั่ง Bring up wireless connection with GPRS คือ AT+CIICR ผลออกมาคือ OK สามารถใช้การติดต่อสื่อสารแบบ GPRS ได้ คำสั่งต่อมาคือ Get local IP address คือ AT+CIFSR เป็นการขอไอพี ผลออกมาคือ 155.67.183.131 คือ IP address มาจาก assigned from GPRS เป็น IP ที่ถูกกำหนดโดยบริษัท ais ในแต่ละครั้งที่ใช้คำสั่งนี้ผลที่ได้จะต่างกันไปเพราะจะเป็น IP ที่สุ่มขึ้นมาโดย ais คำสั่ง Configure domain name sever คือ AT+CDNSCFG= 202.183.255.20 , 202.183.255.21 ผลออกมาคือ OK การกำหนดไอพีหลักและไอพีรองของเอไอเอสได้สำเร็จ คำสั่ง Connect with IP address or DNS AT+CDNSORIP=1 ผลออกมาคือ OK เป็นการบอกตำแหน่งปลายทางเป็น DNS domain name server คำสั่ง Query current connect status ระบุตำแหน่งปลายทางคือ AT+CIPSTART= "TCP","mpgps.freetzi.com",80" ผลออกมาคือ OK คำสั่ง ข้อมูลคือ AT+CIPSEND ผลออกมาคือให้เราส่งข้อมูลไปในรูปแบบการส่งแบบ GET

4.5 การสร้างเซิร์ฟเวอร์และฐานข้อมูล

ในการรับข้อมูลจะต้องประกอบไปด้วยเซิร์ฟเวอร์ที่มีฐานข้อมูลให้ในการจัดเก็บ เซิร์ฟเวอร์ที่ใช้บริการได้สมัครดังนี้

- 1.ลงทะเบียนเป็นสมาชิกที่ <http://www.freetzi.com>
- 2.อัปโหลดที่เขียนลงบนเซิร์ฟเวอร์ของเรา



รูปที่ 4.7 เว็บไซต์ในการสมัครเซิร์ฟเวอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.8 เพื่แสดงแผนที่รถยนต์สองคัน

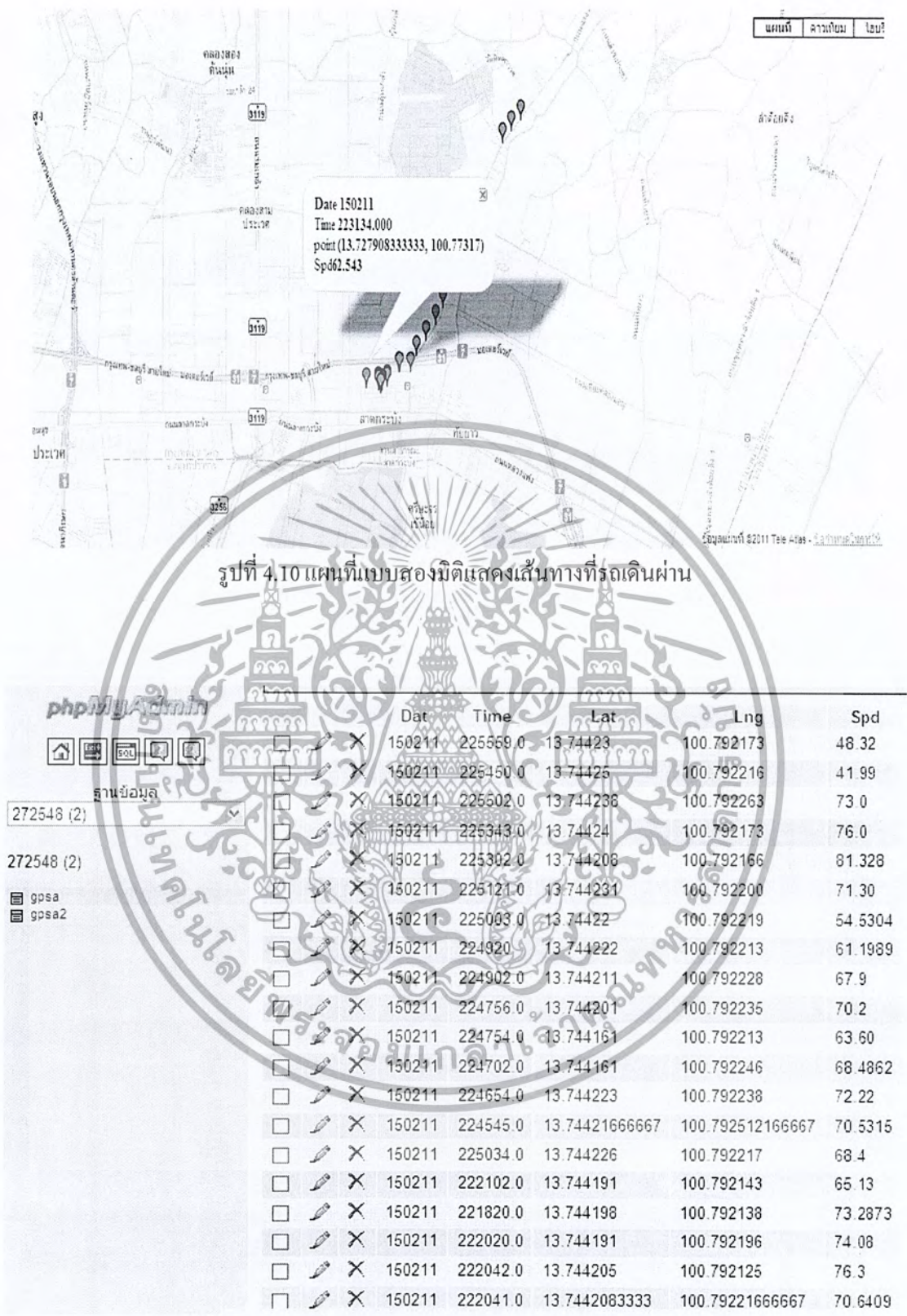
4.6 ทดลองโดยนำอุปกรณ์ไบนารีรถยนต์แล้วเคลื่อนที่

นำอุปกรณ์ที่ประกอบเสร็จ วางบนรถยนต์แล้วเคลื่อนที่จากตึกโทรคมนาคมไปตามถนนทดลอง
กรุงจะได้ผลการเดินทางดังรูป



รูปที่ 4.9 แผนที่แบบดาวเทียมแสดงเส้นทางที่รถเดินทาง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.11 แสดงข้อมูลในฐานข้อมูลตาราง GPSของรถยนต์คันที่ 1

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

phpMyAdmin

ฐานข้อมูล: 272548 (2)

272548 (2)

gpsa

gpsa2

			Dat	Time	Lat	Lng	Spd
☐			150211	213838.0	13.727661666667	100.776131666667	81.328
☐			150211	213856.0	13.7278	100.776096666667	68.068
☐			150211	213915.0	13.72779	100.776091666667	68.51
☐			150211	213934.0	13.727901666667	100.776005	78.234
☐			150211	213953.0	13.728026666667	100.776045	65.858
☐			150211	214011.0	13.72804	100.776035	40.832
☐			150211	214030.0	13.727876666667	100.776075	64.753
☐			150211	214049	13.727641666667	100.776003333333	87.958
☐			150211	214108.0	13.727648333333	100.776843333333	82.433
☐			150211	214126	13.727645	100.776773333333	76.908
☐			150211	214145	13.727616666667	100.776905	59.891
☐			150211	214204.0	13.727721666667	100.776885	60.912
☐			150211	214223.0	13.727695	100.776948333333	62.853
☐			150211	214241.0	13.727633333333	100.776958333333	62.842
☐			150211	214300.0	13.727508333333	100.776983333333	58.565
☐			150211	214319.0	13.727498333333	100.776108333333	30.352
☐			150211	214338.0	13.727498333333	100.776108333333	40.893
☐			150211	214356.0	13.727498333333	100.776108333333	54.765
☐			150211	214415.0	13.727498333333	100.776108333333	63.538
☐			150211	214634.0	13.727498333333	100.776108333333	54.873

รูปที่ 4.11 แสดงข้อมูลในฐานข้อมูลตาราง GPS ของรถยนต์คันที่ 2

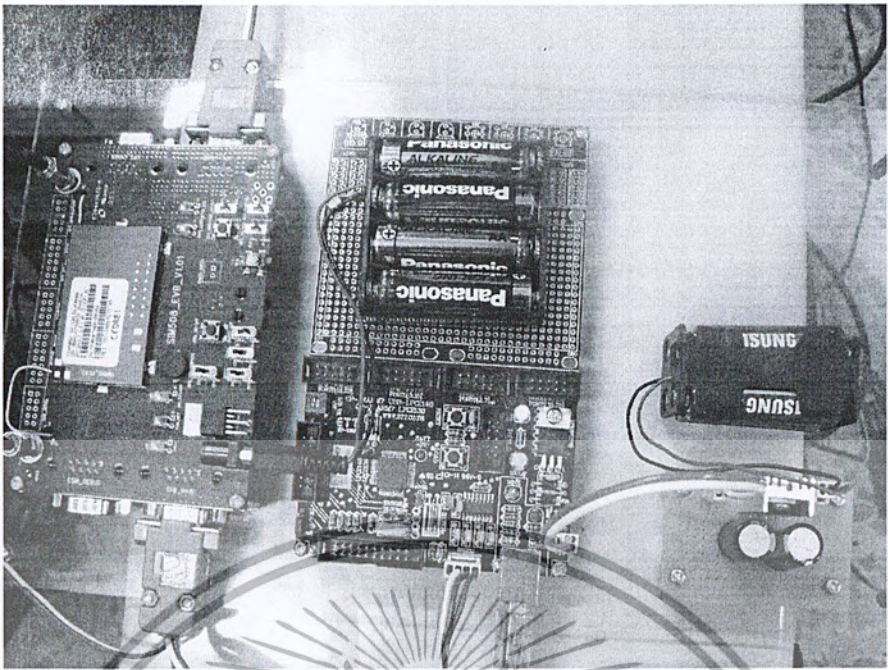
```

- <markers>
<marker type="2" Dat="150211" Time="213838.0" Spd="81.328" lat="13.727661666667" lng="100.776131666667" />
<marker type="2" Dat="150211" Time="213856.0" Spd="68.068" lat="13.7278" lng="100.776096666667" />
<marker type="2" Dat="150211" Time="213915.0" Spd="68.51" lat="13.72779" lng="100.776091666667" />
<marker type="2" Dat="150211" Time="213934.0" Spd="78.234" lat="13.727901666667" lng="100.776005" />
<marker type="2" Dat="150211" Time="213953.0" Spd="65.858" lat="13.728026666667" lng="100.776045" />
<marker type="2" Dat="150211" Time="214011.0" Spd="40.832" lat="13.72804" lng="100.776035" />
<marker type="2" Dat="150211" Time="214030.0" Spd="64.753" lat="13.727876666667" lng="100.776075" />
<marker type="2" Dat="150211" Time="214049" Spd="87.958" lat="13.727641666667" lng="100.776003333333" />
<marker type="2" Dat="150211" Time="214108.0" Spd="82.433" lat="13.727648333333" lng="100.776843333333" />
<marker type="2" Dat="150211" Time="214126" Spd="76.908" lat="13.727645" lng="100.776773333333" />
<marker type="2" Dat="150211" Time="214145" Spd="59.891" lat="13.727616666667" lng="100.776905" />
<marker type="2" Dat="150211" Time="214204.0" Spd="60.912" lat="13.727721666667" lng="100.776885" />
<marker type="2" Dat="150211" Time="214223.0" Spd="62.853" lat="13.727695" lng="100.776948333333" />
<marker type="2" Dat="150211" Time="214241.0" Spd="62.842" lat="13.727633333333" lng="100.776958333333" />
<marker type="2" Dat="150211" Time="214300.0" Spd="58.565" lat="13.727508333333" lng="100.776983333333" />
<marker type="2" Dat="150211" Time="214319.0" Spd="30.352" lat="13.727498333333" lng="100.776108333333" />
<marker type="2" Dat="150211" Time="214338.0" Spd="40.893" lat="13.727498333333" lng="100.776108333333" />
<marker type="2" Dat="150211" Time="214356.0" Spd="54.765" lat="13.727498333333" lng="100.776108333333" />
<marker type="2" Dat="150211" Time="214415.0" Spd="63.538" lat="13.727498333333" lng="100.776108333333" />
<marker type="2" Dat="150211" Time="214634.0" Spd="54.873" lat="13.727498333333" lng="100.776108333333" />
<marker type="2" Dat="150211" Time="214853.0" Spd="65.432" lat="13.727498333333" lng="100.776108333333" />
<marker type="2" Dat="150211" Time="215011.0" Spd="74.876" lat="13.727498333333" lng="100.776108333333" />
<marker type="2" Dat="150211" Time="215230.0" Spd="53.438" lat="13.727168333333" lng="100.776766666667" />
<marker type="2" Dat="150211" Time="215449.0" Spd="65.476" lat="13.727126666667" lng="100.77657" />
<marker type="2" Dat="150211" Time="221846.000" Spd="70.0884" lat="13.742001666667" lng="100.791521666667" />
<marker type="2" Dat="150211" Time="222020.000" Spd="81.328" lat="13.748971666667" lng="100.793646666667" />
<marker type="2" Dat="150211" Time="222058.000" Spd="68.068" lat="13.754058333333" lng="100.79673" />

```

รูปที่ 4.13 แสดงข้อมูลต่างๆ ใน server

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.14 อุปกรณ์รวมของระบบติดตามรถยนต์ด้วยจีพีเอส

สรุปผลการทดลอง

จากการนำอุปกรณ์ที่ประกอบเพื่อใช้ติดตามรถยนต์ 2 ชุดมาทดลองโดยการนำไปวางไว้บนรถยนต์สองคัน แล้วทำการติดตามพร้อมกัน ทำให้สามารถทราบข้อมูลของรถยนต์ทั้งสองคัน โดยการดูผ่านเว็บเซิร์ฟเวอร์ได้

บทที่ 5

สรุปผลและข้อเสนอแนะ

5.1 สรุปผล

จากการทำโครงงานชิ้นนี้ เราทำการรับข้อมูล PSEUDO RANDOM CODE จากเครื่องรับ GPS Module ที่เชื่อมต่อกับคอมพิวเตอร์ผ่านทางพอร์ตอนุกรมโดยใช้โปรแกรม Hyper Terminal โดยค่าที่ได้จะเป็นค่า NMEA Protocol ซึ่งจะประกอบไปด้วยประโยค GPGSA ,GPGSV ,GPRMC ,GPRMB, GPGLA ,GPVGT โดยในแต่ละประโยคจะแสดงข้อมูลที่แตกต่างกันไป ในโครงงานนี้เลือกใช้ประโยค GPRMC เพราะมีข้อมูลที่เราต้องการใช้งาน จากนั้นก็ใช้ไมโครคอนโทรลเลอร์อาร์ม7 กรองข้อมูล แล้วจะได้ข้อมูลเฉพาะประโยค GPRMC ที่มี วันเดือนปี, เวลา, ชั้วโมงนาที่วินาที, พิกัดละติจูด, ลองจิจูด,และความเร็ว จากนั้นก็ส่งข้อมูลไปยังเซิร์ฟเวอร์ด้วยจีพีอาร์เอส โดยใช้เครือข่ายการสื่อสารระบบเอไอเอส เว็บเซิร์ฟเวอร์ที่ใช้คือ <http://www.freetzi.com> เนื่องจากเป็นเว็บที่ไม่ต้องเสียค่าใช้จ่ายและสะดวกต่อการใช้งาน เมื่อเราต้องการติดตามดูข้อมูลของยานพาหนะ สามารถดูข้อมูลได้ทั้งโทรศัพท์มือถือและคอมพิวเตอร์ที่สามารถใช้งานอินเทอร์เน็ตได้ โครงงานนี้ติดตามดูตำแหน่งและข้อมูลต่างๆผ่านบนแผนที่กูเกิ้ลแมพ ที่แสดงบนเว็บเซิร์ฟเวอร์ได้ทั้งหมด อีกทั้งยังสามารถติดตามยานพาหนะได้มากกว่าหนึ่งคัน ในเวลาพร้อมๆกันบนเว็บเซิร์ฟเวอร์เดียวกันได้

ปัญหาที่พบ คือ เมื่อทดสอบอุปกรณ์รับสัญญาณจีพีเอส จะไม่สามารถรับสัญญาณได้เมื่อรถเคลื่อนที่เข้าอาคาร รอดสะพานหรือมีสิ่งกีดขวางบังอยู่ แต่เมื่อรถยนต์ออกมาสู่ที่โล่งก็สามารถรับสัญญาณได้อีกครั้ง และเมื่อทดสอบลองวัดความแม่นยำของสัญญาณรับดาวเทียมแล้ว ปรากฏว่าข้อมูลที่ได้รับจากดาวเทียมมีความคลาดเคลื่อนจากความเป็นจริงมาก ความละเอียดในการระบุตำแหน่งจึงมีน้อยลง

5.2 ข้อเสนอแนะ

แนวทางในการพัฒนาโครงงานต่อไปคือ อุปกรณ์ที่ใช้ติดตั้งในรถยนต์มีขนาดใหญ่ ทำให้ไม่สะดวกในการติดตั้ง จะพัฒนาให้มีขนาดกะทัดรัดให้เหมาะกับการติดตั้ง ไม่เฉพาะแค่ใช้ติดตามรถยนต์เท่านั้น สามารถที่จะใช้ติดตามวัตถุอื่นๆได้

บรรณานุกรม

- [1] John HARRISON'S CHRONOMETER ILLUSTRATION.14,01,2010, From <http://www.fotolibra.com/gallery/49403/john-harrisons-chronometer-illustration/>.
- [2] การสำรวจอวกาศ SpaceExploration Time Line .14,01,2010,From <http://sci4fun.com/spaceexploration/timeline.html>
- [3] GLOBAL POSITIONING SYSTEM , From http://www.thaitechnics.com/nav/gps_t.html .
- [4] ไพโรจน์ ไวนิชขกิจ.มือถือไทย ... อดีต ปัจจุบัน กับพัฒนาการสู่อนาคต.14,02,2010, From http://www.siamphone.com/news/aritcle/2004/00001_2.htm
- [5] พิชญ จุลศิริ. ครอบเครื่องเรื่อง GPS. พิมพ์ครั้งที่ 1. กรุงเทพมหานคร:บริษัท ซีเอ็ดดูเคชั่นจำกัด (มหาชน), 2549.
- [6] โอภาส ศิริกรรชิตถาวร. เรียนรู้และพัฒนาไมโครคอนโทรลเลอร์ ARM7 LPC2148. พิมพ์ครั้งที่ 1. กรุงเทพมหานคร:โรงพิมพ์เจริญพรสาธิต,2549.
- [7] ชานูชัย สุภอรรถกร.คู่มือเรียนเขียนเว็บอีคอมเมิร์ซด้วยPHP+MySQL ฉบับสมบูรณ์.พิมพ์ครั้งที่1. กรุงเทพมหานคร:บริษัท รีไวน์ จำกัด, 2552.





เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การใช้งานบอร์ดไมโครคอนโทรลเลอร์รุ่น CP-JR ARM7 USB-LPC2148

CP-JR ARM7 USB-LPC2148 เป็นบอร์ดไมโครคอนโทรลเลอร์ในตระกูล ARM7TDMI-S Core ซึ่งเลือกใช้ไมโครคอนโทรลเลอร์ 16/32-Bit ขนาด 64 Pin แบบใช้พลังงานต่ำเป็น MCU ประจำบอร์ด ซึ่งบอร์ดนี้เลือกใช้ MCU เบอร์ LPC2148 ของ Philips โดยการออกแบบโครงสร้างของบอร์ดนั้นจะเน้นเรื่องของการจัดวางอุปกรณ์พื้นฐานที่จำเป็นต่อการ ศึกษา และ ทดลอง ขึ้นพื้นฐาน อย่างเป็นสัดส่วน และเป็นอิสระต่อกันเช่น LED แสดงสถานะของ Output Logic และ Push Button Switch สำหรับสร้างสัญญาณ Logic เพื่อ ทดสอบการทำงานของ Input หรือ Volume ปรับค่าแรงดัน เพื่อให้ทดสอบการทำงานของ A/D เป็นต้น โดยวงจรพื้นฐานต่างๆเหล่านี้จะถูกออกแบบและจัดวางเป็นจุดเชื่อมต่อแบบอิสระไว้ โดยยังไม่ได้เชื่อมต่อสัญญาณเข้ากับขาสัญญาณใดๆของ CPU เพื่อให้ผู้ใช้สามารถปรับเลือกใช้งานวงจรประกอบทดสอบที่มีอยู่ในบอร์ดให้สอดคล้องกับทรัพยากรภายในตัว CPU ได้อย่างเหมาะสมและเป็นอิสระ โดยไม่ยึดติดกับ วงจรใดวงจรหนึ่งเพียงอย่างเดียว ตัวอย่างเช่น ขาสัญญาณ GPIO0.8 ของ LPC2148 สามารถเลือกกำหนด การทำงานได้มากถึง 4 แบบ คือ ใช้งานเป็น Input/Output, TXD1, PWM และ ADC ดังนั้นเมื่อผู้ใช้ต้องการทดสอบการใช้งานขา GPIO0.8 ในโหมดใดก็สามารถเลือกต่อสัญญาณจาก GPIO0.8 ไปยังวงจรสนับสนุนที่เหมาะสมได้ตามต้องการเช่น

เมื่อต้องการทดสอบการใช้งาน GPIO0.8 เป็น Input ก็สามารถเลือกต่อสัญญาณจาก GPIO0.8 ไปยังวงจรของ Push Button Switch เพื่อทดสอบการทำงานกับ Input

เมื่อต้องการทดสอบการใช้งาน GPIO0.8 เป็น Output ก็สามารถเลือกต่อสัญญาณจาก GPIO0.8 ไปยังวงจรของ LED เพื่อทดสอบสถานะการทำงานของ Output

เมื่อต้องการทดสอบการใช้งาน GPIO0.8 เป็น ADC ก็สามารถเลือกต่อสัญญาณจาก GPIO0.8 ไปยังวงจรของ Volume ปรับค่าแรงดัน เพื่อทดสอบอ่านค่าแรงดัน Analog

เมื่อต้องการทดสอบการใช้งาน GPIO0.8 เป็น TXD1 ก็สามารถเลือกต่อสัญญาณจาก GPIO0.8 ไปยังวงจรของ Line Driver ของ RS232 เพื่อทดสอบการส่งข้อมูลแบบ RS232 ซึ่งจะเห็นได้ว่า ผู้ใช้สามารถเลือก กำหนด และ ปรับเปลี่ยน การใช้งานขาสัญญาณต่างๆของ CPU ได้อย่างครบถ้วนทุกฟังก์ชันการทำงาน ได้โดยอิสระอย่างแท้จริง โดยโครงสร้างของบอร์ดจะนำ MCU มาจัด วงจรร่วมกับอุปกรณ์พื้นฐานที่จำเป็นและจัดขาออกมาให้ใช้งานภายนอกผ่านหัวต่อ Connector แบบ IDE 10 Pin ซึ่งการจัดเรียงขาสัญญาณจะทำการจัดเรียงอย่างเป็นระเบียบตามมาตรฐานของ อีทีที เพื่อให้สามารถต่อใช้งานร่วมกับอุปกรณ์สนับสนุนต่างๆของ อีทีที ได้โดยสะดวก ตัวบอร์ดใช้ไฟ +5VDC จากภายนอก หรือ USB Port (ในกรณีใช้กระแสไม่เกิน 100mA ขณะยังไม่ Enumerate USB และไม่เกิน 500mA ขณะ Enumerate USB แล้ว) ซึ่งภายในบอร์ดจะมีวงจร Regulate ขนาด 3.3V/1A จัดเตรียมไว้ให้เรียบร้อยแล้ว และภายในบอร์ดจะมี Connector RS232(UART0) จำนวน 1 Port สำหรับทำการ Download Hex File หรือใช้งานในการสื่อสาร RS232 ในโปรแกรม Application ที่เขียนขึ้นเอง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

คุณสมบัติของบอร์ด

1. ใช้ MCU ตระกูล ARM7TDMI-S เบอร์ LPC2148 ของ Philips ซึ่งเป็น MCU ขนาด 16/32-Bit
2. ใช้ Crystal 12.00 MHz โดย MCU สามารถประมวลผลด้วยความเร็วสูงสุดที่ 60 MHz เมื่อใช้งานร่วมกับ Phase-Locked Loop (PLL) ภายในตัว MCU เอง
3. รองรับการโปรแกรมแบบ In-System Programming (ISP) และ In-Application Programming (IAP) ผ่านทาง On-Chip Boot-Loader Software ทางพอร์ต UART0 (RS232)
4. Power Supply ใช้แรงดันไฟฟ้า +5VDC โดยใช้ขั้วต่อแบบ CPA-2PIN จากภายนอก หรือ ใช้พลังงานจาก USB Port ได้ (ในกรณีใช้กระแสไม่เกิน 500mA)
5. ภายใน MCU มีหน่วยความจำโปรแกรมแบบ Flash ขนาด 512KB, Static RAM ขนาด 40KB
6. มีวงจร USB มาตรฐาน 2.0 แบบ Full Speed ภายในตัว (USB Function มี 32 End Point)
7. จำนวน GPIO สูงสุดถึง 47 I/O-Pins สามารถเชื่อมต่อกับระบบ I/O ที่เป็นสัญญาณ 5V ได้ ซึ่งขาสัญญาณ GPIO จะมีการใช้งานร่วมกันของ Function อื่นๆ อีกดังนี้ วงจรสื่อสารอนุกรมแบบ SPI จำนวน 2 ช่อง และ วงจรสื่อสารอนุกรมแบบ I2C จำนวน 2 ช่อง วงจร ADC ขนาด 10 Bit จำนวน 14 ชุด และ วงจร DAC ขนาด 10 Bit จำนวน 1 ชุด วงจร UART แบบ Full-Duplex จำนวน 2 ช่อง คือ UART-0 มาตรฐาน 4 Pin ETT เป็นสัญญาณระดับ RS232 Level และ UART-1 เป็นสัญญาณระดับ TTL Level Timer 32-bit จำนวน 2 ช่อง (4 Input Capture / 4 Output Compare), 6-Channel PWM Output, Watchdog Timer และ Real Time Clock
8. มีวงจรเชื่อมต่อกับ Character LCD โดยใช้วงจรการเชื่อมต่อแบบ 4 บิต จาก GPIO1[25..31] พร้อมวงจรปรับความสว่างหน้าจอ
9. มีวงจรเชื่อมต่อกับ JTAG ARM ขนาด 20 Pin มาตรฐาน เพื่อทำการ Debug แบบ Real Time ได้
10. มีวงจรทดลองขั้นพื้นฐานสำหรับสนับสนุนการใช้งานและทดลองเรียนรู้ ขั้นพื้นฐานอย่างครบถ้วน จัดเตรียมไว้ภายในบอร์ด (ติดตั้งไว้เฉพาะรุ่น CP-JR ARM7 USB-LPC2148 EXP) ซึ่ง ได้แก่

-LED Output แบบ Sink Current สำหรับแสดงสถานะของ Output จำนวน 4 ชุด

-Push Button Switch แบบ Active Logic "0" สำหรับทดสอบ Input Logic จำนวน 4 ชุด

-Volume ปรับค่าแรงดัน 0-3.3V สำหรับทดสอบการทำงานของ ADC จำนวน 4 ชุด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

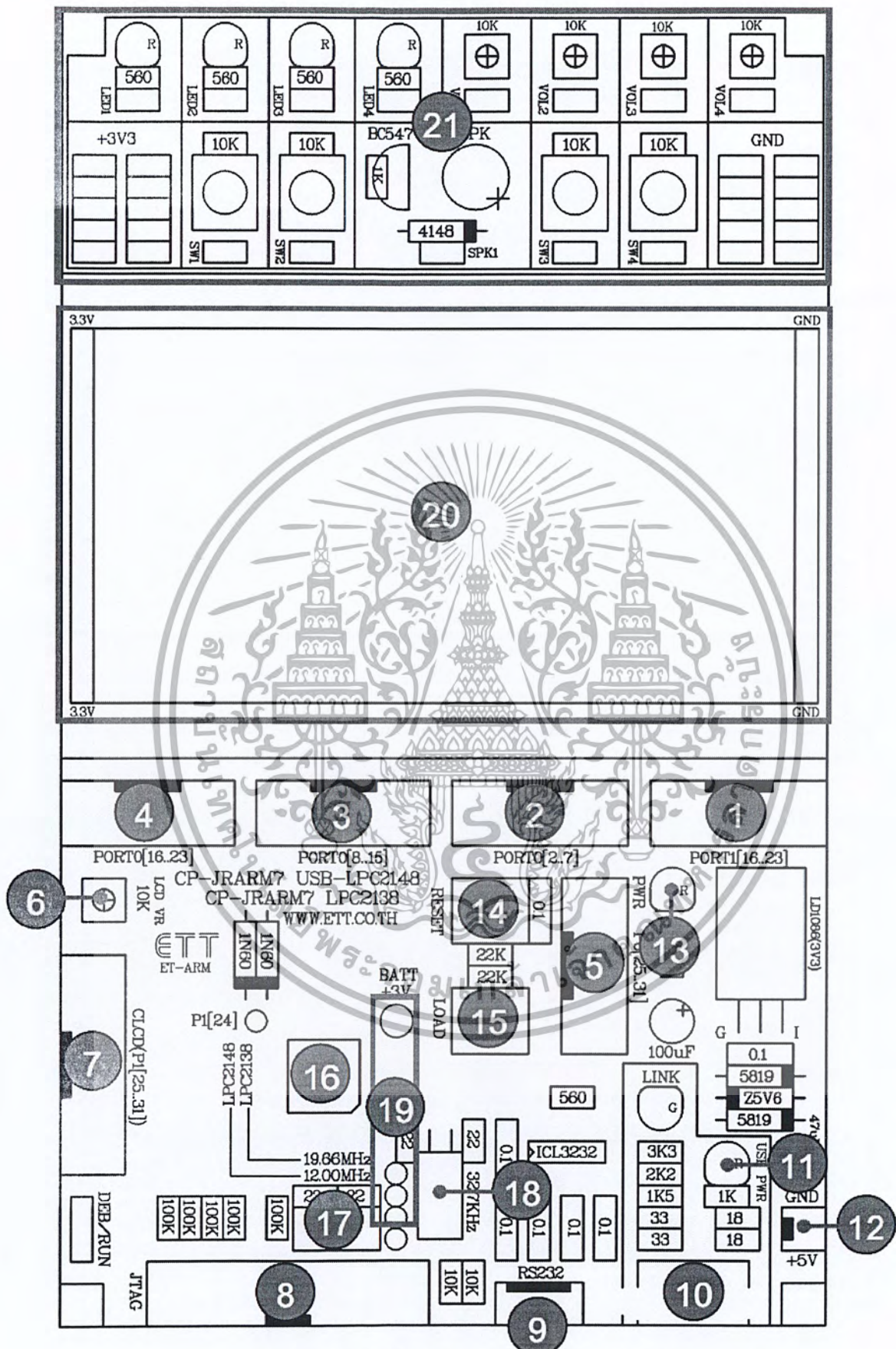
- ชุดกำเนิดสัญญาณเสียง Mini Speaker สำหรับทดสอบการเสียงแบบต่างๆ จำนวน 1 ชุด
- แผงต่อวงจร Project Board รุ่น AD-100 ขนาด 360 จุด สำหรับเป็นพื้นที่ต่อทดลองวงจรขนาดเล็กๆ เพื่อใช้งานร่วมกับ CPU ได้อย่างอิสระ
- จุดต่อแหล่งจ่ายไฟ +3.3V และ GND สำหรับเชื่อมต่อไปยังวงจรภายนอกอื่นๆ

11. ทนอุณหภูมิใช้งานระหว่าง -40 to +85°C

โครงสร้างบอร์ด CP-JR ARM7 USB-LPC2148 / EXP



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

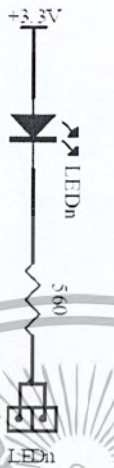


รูป ลักษณะของบอร์ด CP-JR ARM7 USB-LPC2148

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การใช้งาน LED แสดงผล

LED แสดงผลของบอร์ด จะต้องวงจรแบบรับกระแส (Sink Current) โดยใช้กับแหล่งจ่าย +3.3V ทำงานด้วยลอจิก “0” (0V) และหยุดทำงานด้วยลอจิก “1” (+3.3V) โดยมีทั้งหมด 4 ชุด โดยวงจรในส่วนนี้จะใช้สำหรับทดสอบการทำงานของ Port I/O ต่างๆที่ทำงานให้ผลเป็น Output แบบลอจิก



การใช้งานวงจรปรับแรงดัน (0V-3.3V)

วงจรปรับแรงดันของบอร์ดจะใช้ตัวต้านทานปรับค่าได้แบบเกอ๊กน้ำ ชนิดมีแกนหมุนสำหรับปรับค่าโดยวงจรนี้ใช้กับแหล่งจ่าย +3.3V โดยจะให้ Output เป็นแรงดันซึ่งมีค่าระหว่าง 0V ถึง +3.3V ตามการปรับค่าของตัวต้านทาน ซึ่งมีทั้งหมด 4 ชุด ใช้สำหรับสร้างแรงดัน Input เพื่อทดสอบการทำงานของวงจร A/D

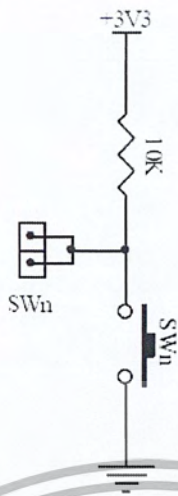


ใช้งานวงจร Push Button Switch

วงจร Push Button Switch จะใช้วงจร Switch แบบ กดติด-ปล่อยดับ (Push Button) พร้อมวงจร Pull-Up ใช้กับแหล่งจ่าย +3.3V โดยในขณะที่สวิตช์ยังไม่ถูกกดจะให้ค่าสถานะเป็นลอจิก “1” แต่เมื่อสวิตช์ถูกกดอยู่จะให้สถานะเป็นลอจิก “0” โดยวงจรส่วนนี้จะมีอยู่ด้วยกัน 4 ชุด ใช้สำหรับทดสอบการทำงานของ

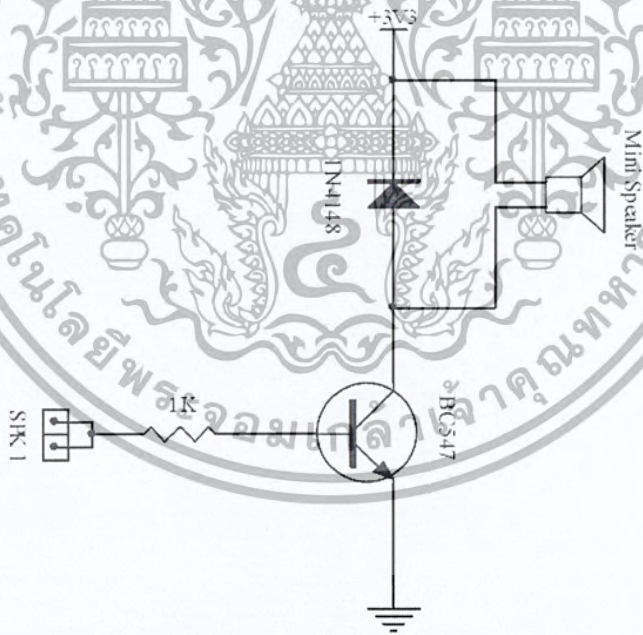
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

I/O ต่างๆที่ต้องควบคุมการทำงานของวงจรด้วย Input แบบลอจิก



การใช้งาน วงจรกำเนิดเสียง

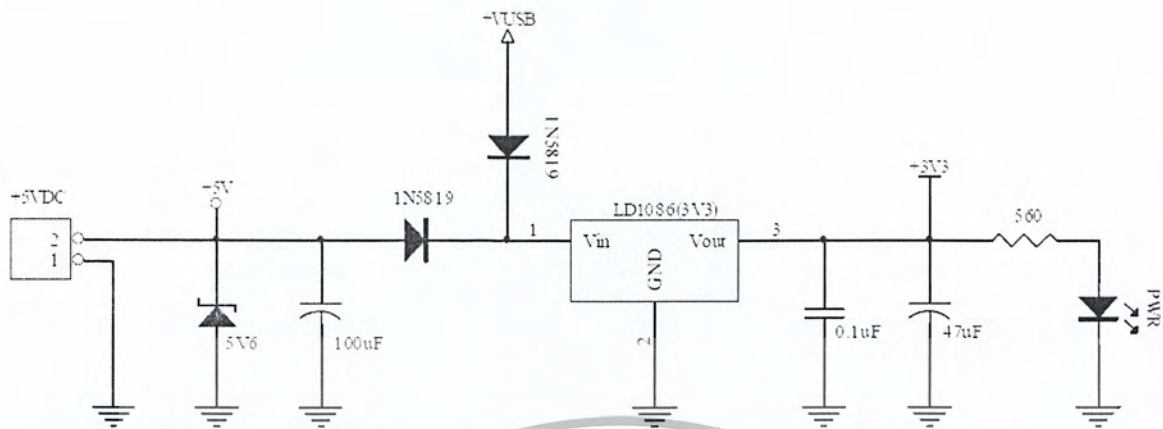
วงจรกำเนิดเสียง จะใช้ลำโพงขนาดเล็ก (Mini Speaker) พร้อมด้วยวงจรทรานซิสเตอร์แบบ NPN สำหรับขับกระแสให้กับลำโพง ใช้กับแหล่งจ่ายขนาด +3.3V ทำงานด้วยลอจิก “1” และหยุดทำงานด้วยลอจิก “0” โดยในการทำงานนั้นต้องส่งสัญญาณลอจิกที่เป็นความถี่ต่างๆ ให้กับลำโพงเพื่อสร้างเป็นความถี่เสียงย่านต่างๆ ตามต้องการ



วงจรแหล่งจ่ายไฟ

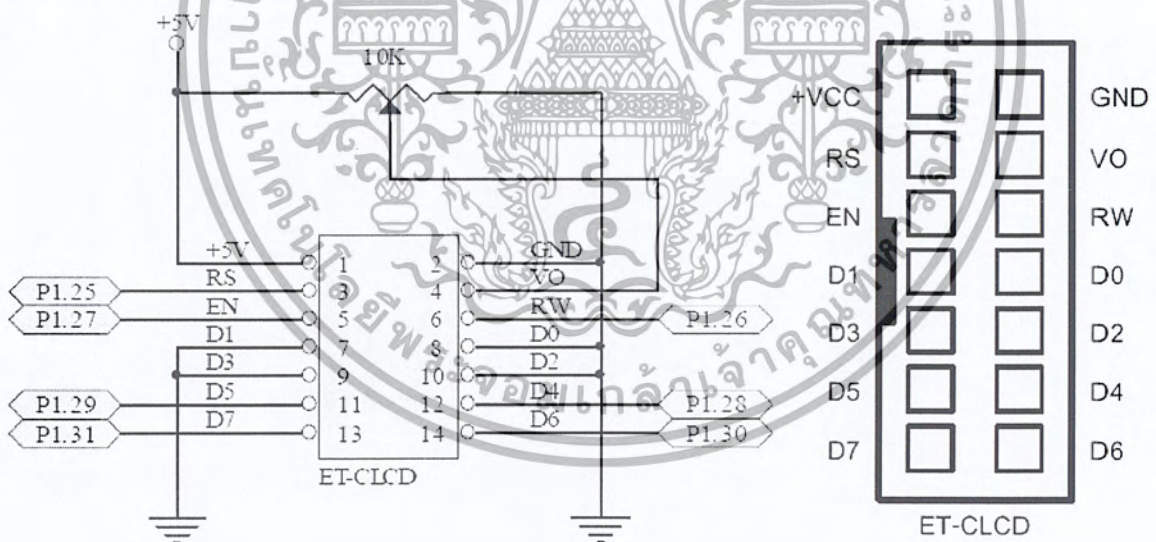
วงจรแหล่งจ่ายไฟสามารถใช้งานได้กับไฟ DC ขนาด +5V ได้ทันที โดยวงจรภาคแหล่งจ่ายไฟในส่วนที่เป็นวงจร Regulate ขนาด 3.3V นั้นจะจ่ายให้กับ CPU และวงจร I/O ของบอร์ดทั้งหมด ยกเว้น LCD แบบ Character ซึ่งจะใช้แหล่งจ่ายไฟขนาด +5VDC จากขั้ว Terminal ขนาด 2 Pin ที่จ่ายให้กับบอร์ด ในกรณีที่ไม่ได้ใช้งาน Character LCD ผู้ใช้สามารถใช้แหล่งจ่ายจาก USB Port เพียงอย่างเดียวได้ แต่ต้องไม่นำอุปกรณ์อื่นๆภายนอกมาเชื่อมต่อกับบอร์ด โดย USB Port สามารถจ่ายพลังงานให้กับบอร์ดได้ เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ไม่เกิน 100mA ถ้ายังไม่ทำการ Enumerate USB Function และสามารถจ่ายกระแสให้บอร์ดได้สูงสุดไม่เกิน 500mA หลังจากทำการ Enumerate USB Function เรียบร้อยแล้ว



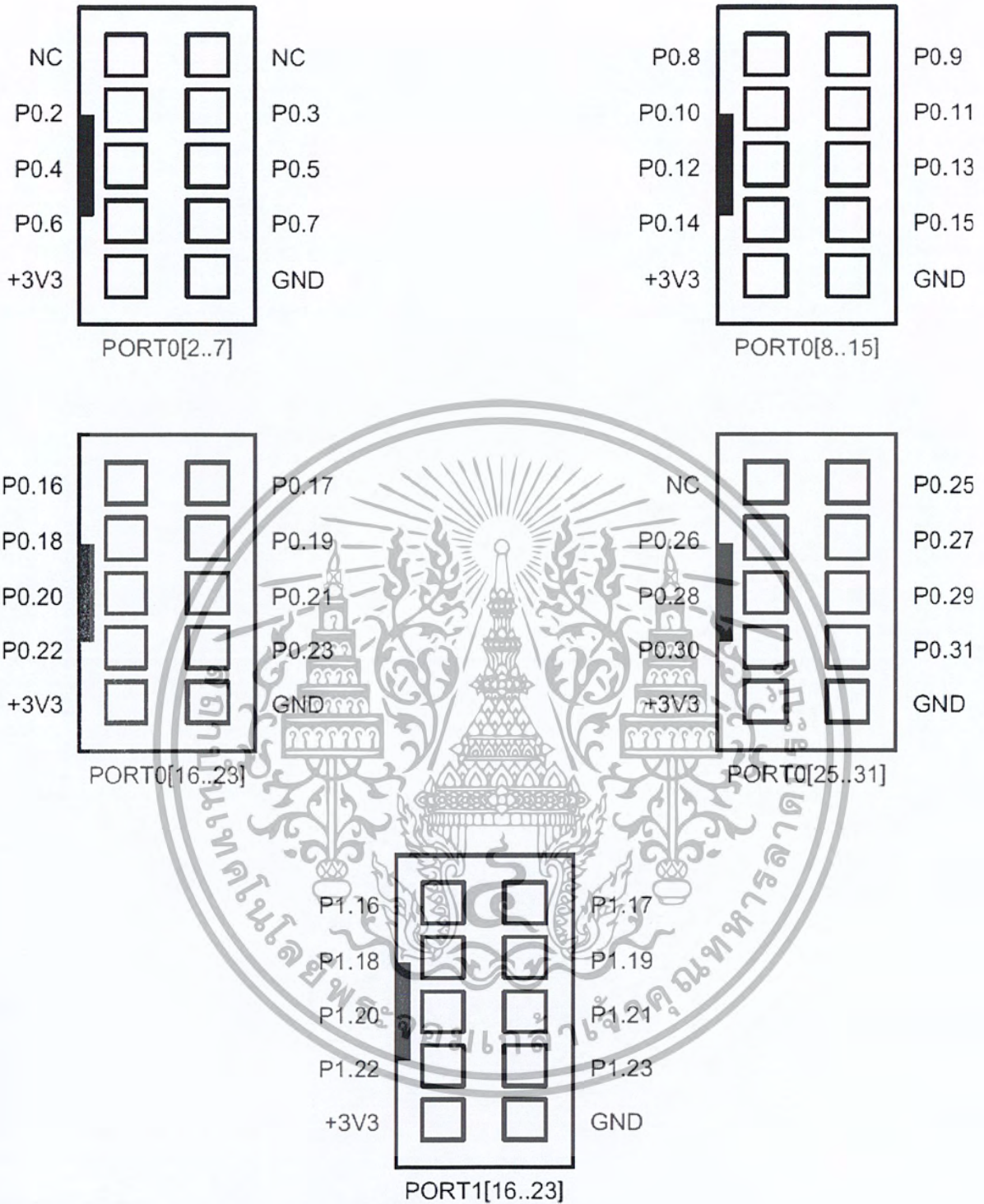
การเชื่อมต่อกับ Character LCD

สำหรับการเชื่อมต่อ LCD นั้นจะสามารถใช้ได้กับ LCD แบบ Character LCD เท่านั้น โดยเชื่อมต่อแบบ 4 บิต โดยสัญญาณที่ใช้เชื่อมต่อกับ LCD จะเป็นสัญญาณจาก GPIO1.25 ถึง GPIO1.31 จำนวน 7 บิต ซึ่งจะมีสัญญาณบางบิตถูกต่อไปยังหัวต่อ JTAG ด้วย โดยในการเชื่อมต่อสายสัญญาณจากหัวต่อของพอร์ต LCD ไปยังจอแสดงผล LCD นั้น ให้ยึดชื่อสัญญาณเป็นจุดอ้างอิง โดยให้ต่อสัญญาณที่มีชื่อตรงกันเข้าด้วยกันให้ครบทั้ง 14 เส้น ดังรูป



ข้อต่อ Port I/O ต่างๆของบอร์ด

สำหรับขั้วต่อ Port I/O ของ CPU นั้น จะจัดเรียงออกมาเป็นขั้วต่อแบบ IDE 10 Pin จำนวน 5 ชุด ชุดละ 8 บิต ดังนี้



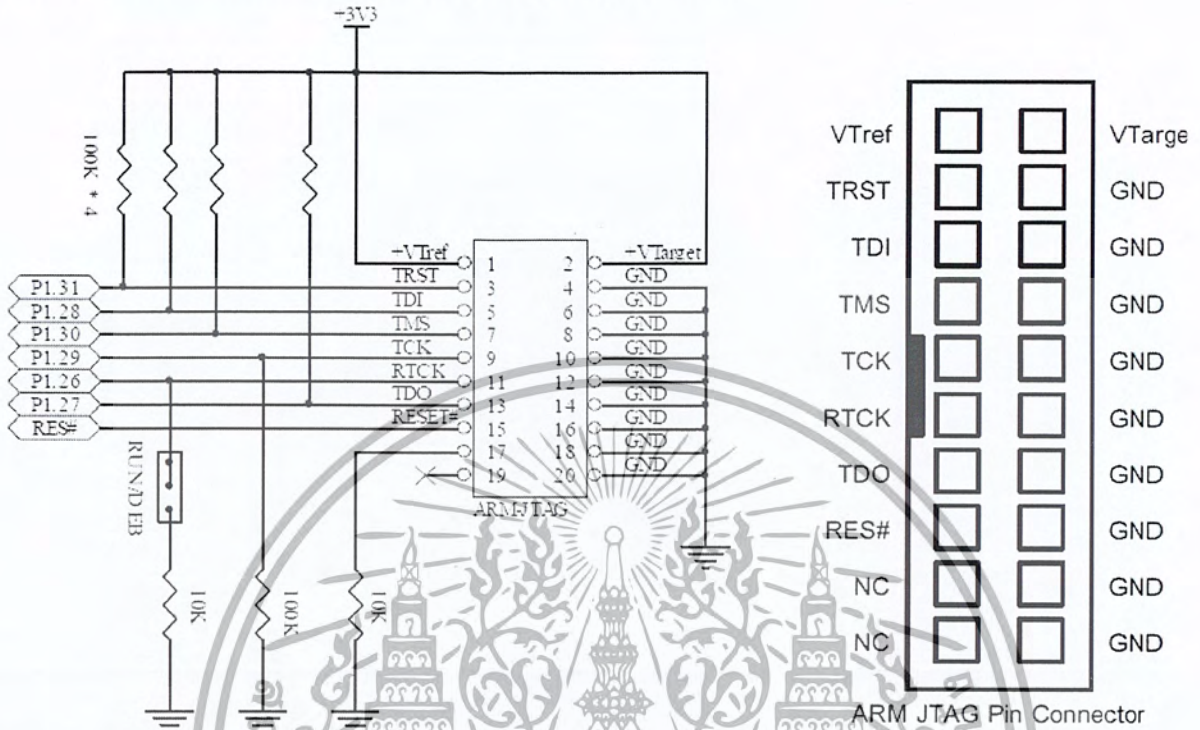
หมายเหตุ

- **GPIO0.0** ถูกสงวนไว้ใช้เป็น TXD ของ UART0 (RS232)
- **GPIO0.1** ถูกสงวนไว้ใช้เป็น RXD ของ UART0 (RS232)
- **GPIO1.25 ถึง GPIO1.31** ถูกสงวนไว้ใช้ เชื่อมต่อกับ Character LCD
- **GPIO.23, GPIO0.26, GPIO0.27 และ GPIO0.31** สงวนไว้สำหรับการเชื่อมต่อกับ USB

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ข้อต่อ JTAG ARM

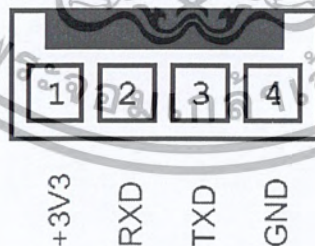
JTAG หรือ JTAG ARM จะเป็น Connector แบบ IDE 20 Pin สำหรับ Interface กับ JTAG Debugger โดยมีการจัดวงจรและสัญญาณตามมาตรฐานของ JTAG ดังนี้



ข้อต่อ RS232

พอร์ต RS232 เป็นสัญญาณ RS232 ซึ่งผ่านวงจรแปลงระดับสัญญาณ MAX232 เรียบร้อยแล้ว สามารถใช้เชื่อมต่อกับสัญญาณ RS232 เพื่อรับส่งข้อมูล นอกจากนี้แล้วยังสามารถใช้งาน ร่วมกับ Switch LOAD และ Switch RESET เพื่อทำการ Download Hex File ให้กับ CPU ได้ด้วย

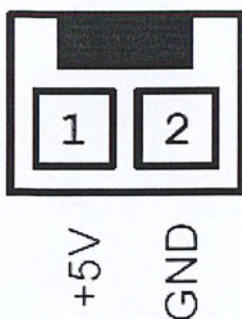
RS232



ข้อต่อ Power Input (+5V)

เป็นจุดต่อไฟเลี้ยงบอร์ด +5V และ GND ซึ่งใช้สำหรับเป็นแหล่งจ่ายไฟให้กับบอร์ด โดยต้องป้อนเป็นไฟกระแสตรง DC ขนาด +5V เท่านั้น โดยมีการจัดขั้วดังนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

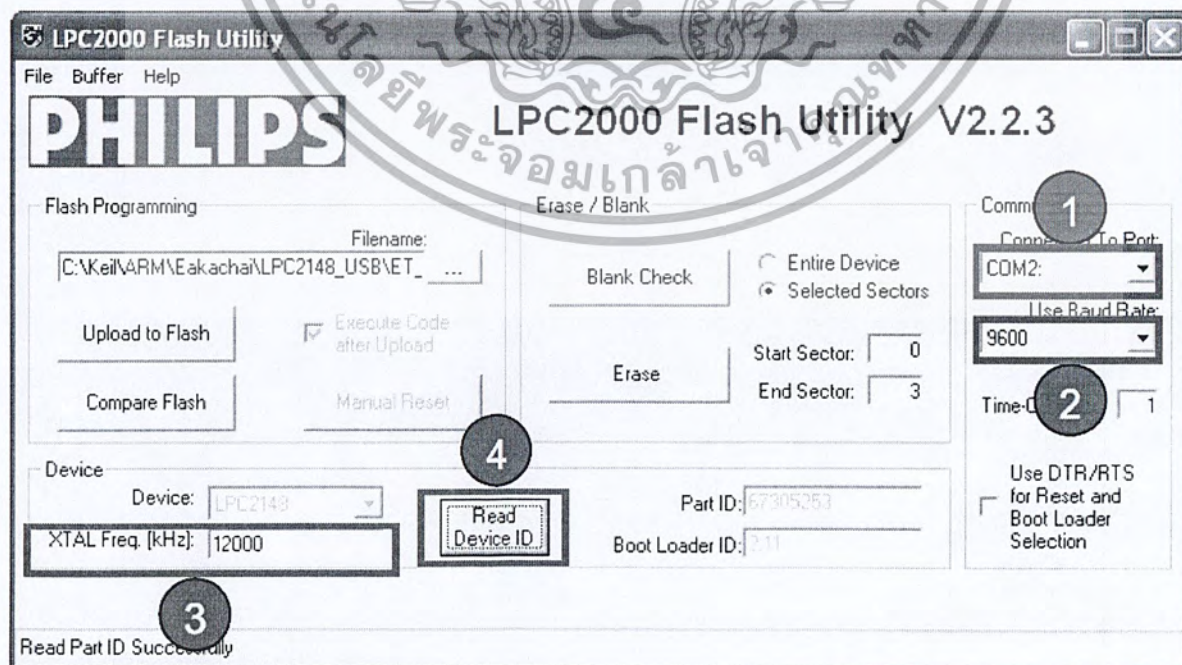


การ Download Hex file ให้กับ MCU ของบอร์ด

การ Download Hex File ให้กับหน่วยความจำ Flash ของ MCU ในบอร์ดนั้น จะใช้โปรแกรมชื่อ LPC2000 Flash Utility ของ Philips ซึ่งจะติดต่อกับ MCU ผ่าน Serial Port ของคอมพิวเตอร์ PC โดยโปรแกรมดังกล่าวสามารถดาวน์โหลดฟรีได้ที่ www.semiconductors.philips.com

ขั้นตอนการ Download HEX File ให้กับ MCU

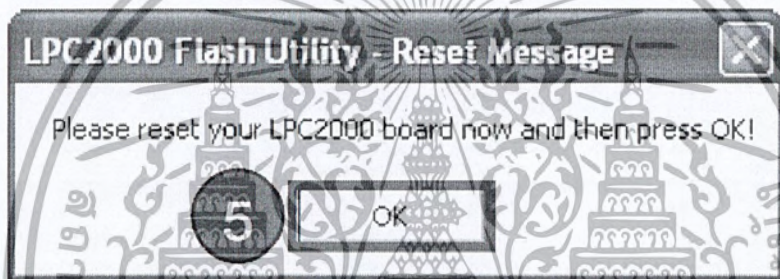
1. ต่อสายสัญญาณ RS232 ระหว่างพอร์ตสื่อสารอนุกรม RS232 ของ PC และบอร์ด (RS232)
2. จ่ายไฟเลี้ยงวงจร +5VDC ให้กับบอร์ด ทางขั้วต่อ CPA ขนาด 2 Pin (หรือใช้การผ่าน USB Port ได้ถ้า USB Hub สามารถจ่ายพลังงานได้) ซึ่งจะสังเกตเห็น LED สีแดง (PWR) ติดสว่างให้เห็น
3. สั่ง Run โปรแกรม LPC2000 Flash Utility ของ Philips ซึ่งจะ ได้ผลดังรูป



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4. เริ่มต้นกำหนดค่าตัวเลือกต่างๆ ให้กับโปรแกรมตามต้องการ ซึ่งในกรณีนี้ใช้กับ LPC2148 ของบอร์ด CP-JR ARM7 USB-LPC2148 ของ อีทีที ให้เลือกกำหนดค่าต่างๆ ให้โปรแกรมดังนี้

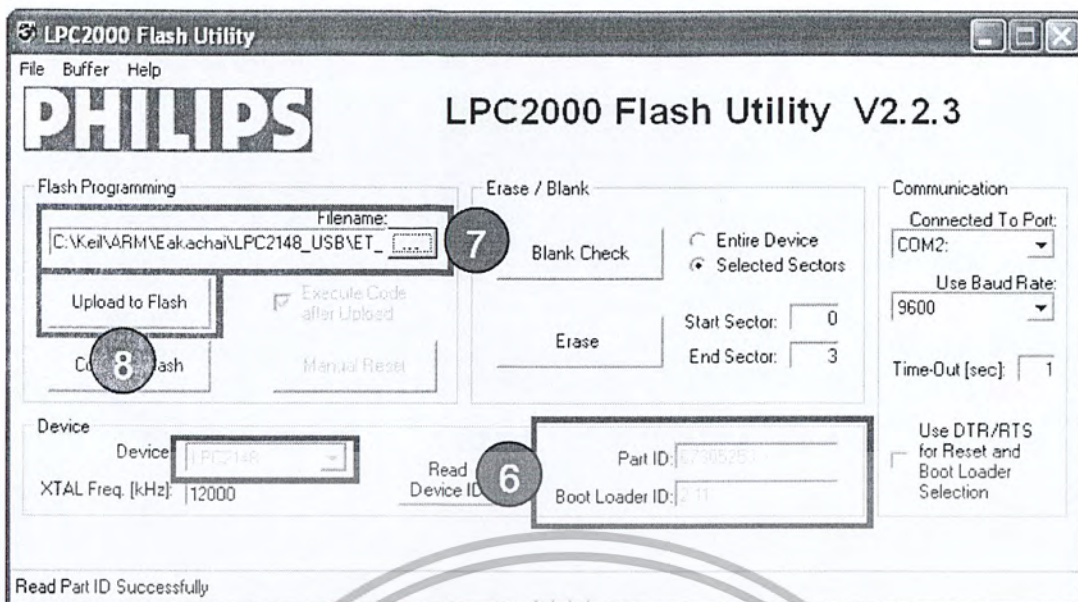
- 1) เลือก COM Port ให้ตรงกับหมายเลข COM Port ที่ใช้งานจริง (ในตัวอย่างใช้ COM1)
- 2) ตั้งค่า Baud Rate อยู่ที่ระหว่าง 4800 - 38400 ซึ่งเป็นค่าที่ทดสอบแล้วใช้ได้โดยไม่เกิดปัญหา หรือใช้ค่าความเร็วมาตรฐานคือ 9600
- 3) กำหนดค่าคริสตอล ออสซิลเลเตอร์ ให้ตรงกับที่ใช้ในจริงภายในบอร์ด โดยกำหนดให้มีหน่วยเป็น KHz และห้ามใส่ค่าเกิน 5 หลัก ในที่นี้ใช้ค่า 12.000MHz ซึ่งเท่ากับ 12000
- 4) คลิกเมาส์ที่ปุ่มคำสั่ง Read Device ID เพื่อติดต่อกับ CPU ซึ่งจะมีข้อความขึ้นมาเตือนให้เข้าสู่ Boot Mode ดังแสดงในรูป



5) ให้กดสวิตช์ RESET และ LOAD ที่บอร์ด CP-JR ARM7 USB-LPC2148 เพื่อทำการReset ให้ MCU ทำงานใน Boot Loader ตามขั้นตอนดังต่อไปนี้

- กดสวิตช์ LOAD ค้างไว้
- กดสวิตช์ RESET โดยที่สวิตช์ LOAD ยังกดค้างอยู่
- ปลดสวิตช์ RESET โดยที่สวิตช์ LOAD ยังกดค้างอยู่
- ปลดสวิตช์ LOAD เป็นลำดับสุดท้าย เสร็จแล้วจึงคลิกเมาส์ที่ "OK"

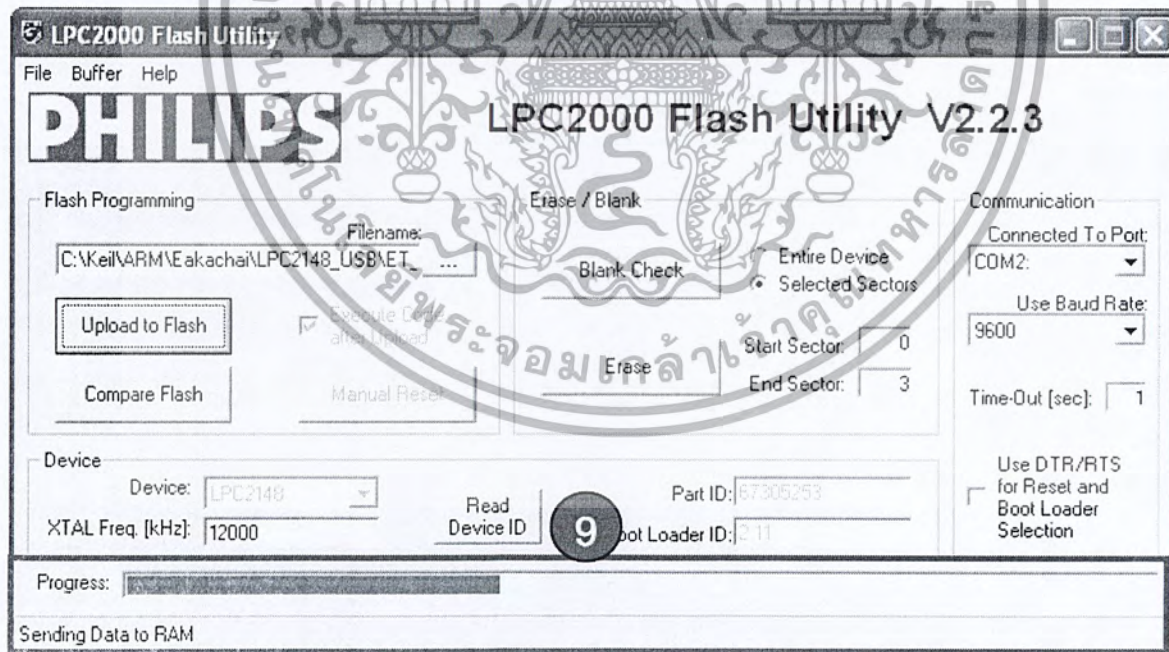
6) เมื่อติดต่อกับ CPU ได้ จะปรากฏรายละเอียด Part ID และ Boot Loader ID ดังรูป



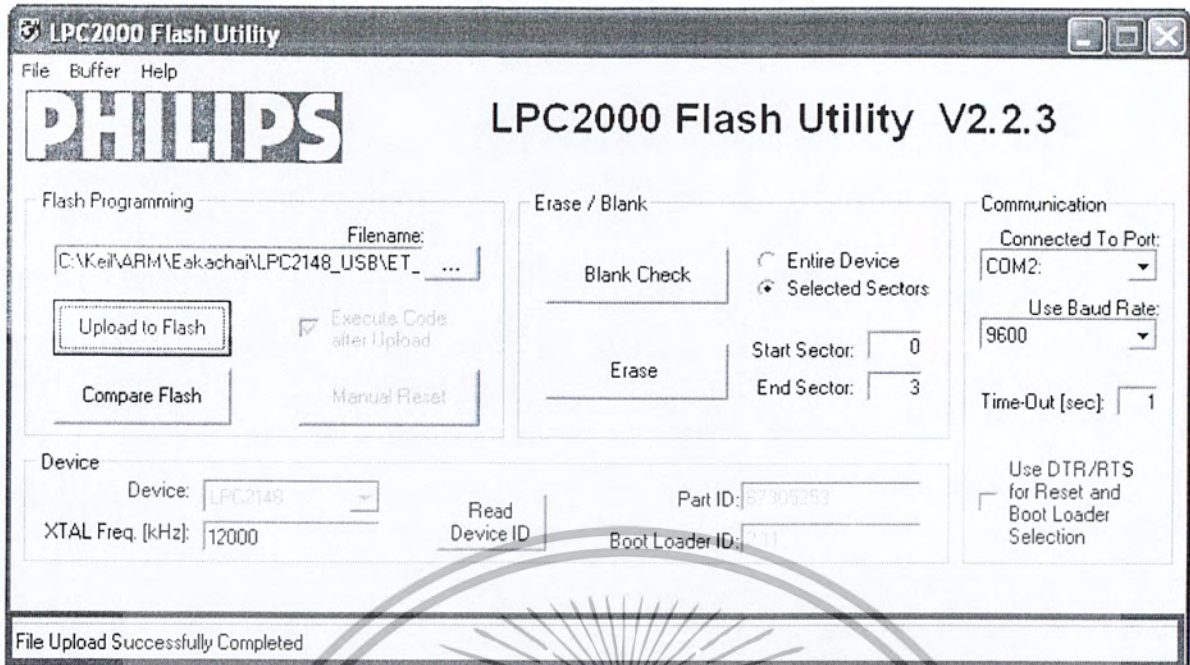
7) ให้ทำการเลือกกำหนด HEX File ที่จะทำการส่งโปรแกรม

8) ให้ทำการคลิกเมาส์ที่ “Upload to Flash” ซึ่งโปรแกรม LPC2000 จะเริ่มต้นทำการ

Download ข้อมูลให้กับ MCU ทันที โดยสังเกตที่ Status “Uploading to LPC2000 RAM and Copying to Flash Memory” ดังรูป โดยในขั้นตอนนี้ให้รอก่อนกว่าการทำงานของโปรแกรมจะเสร็จสมบูรณ์ ซึ่งให้สังเกตที่ Status “File Upload Successfully Completed”



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



9) เมื่อการทำงานของโปรแกรมเสร็จเรียบร้อยแล้ว ให้กดสวิทช์ Reset ที่บอร์ด ซึ่ง MCU จะเริ่มต้นทำงานตามโปรแกรมที่สั่ง Download ให้ทันที

Code Part Microcontroller

```

/*****
/* Target MCU : Philips ARM7-LPC2148 */
/*      : X-TAL : 12.000 MHz */
/*      : Run Speed 58.9824MHz (With PLL) */
/* Compiler : Keil ARM V2.32a */
/* Function : Example ....(ENTER YOUR PROJECT) */
*****/

#include <LPC214x.H> // LPC21xx MPU Register
#include <serial1.h> /* pototype section */

void serial0_init (unsigned long baud); // Initil UART-0

int serial0_putc (int ch);

int serial0_getc(void);

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

void serial0_puts(unsigned char *st);
void serial0_gets(void);
void http_post2(void); // GET message
void estab_con1(void); // establish connection past_1
void estab_con2(void);
void delay(unsigned long int); // Delay time Function
void detect_data(void);
void keep_rmc2(void);
void edit_time(void);
void edit_date(void);
unsigned int i, ii, m,v,sel;
unsigned char st[50];
unsigned char j1, j2, jj, trmc, buff_ori[500],buff_item[20];
unsigned char find_rmc, res_rmc;
unsigned char sta_wri, sta_rea, buff_stp[60], buff_upd[60];
unsigned char update_rmc, t_end, often_t;
unsigned char lat[12], lng[12], spd[6], dat[12], tim[12], k1, k2, k3, k4, k5;
unsigned char lat_1, lat_2, lng_1, lng_2, hour, time1;
void main(void)
{
    for(i=0; i<=20; i++) // clr data
    {
        dat[i] = 'x';
        tim[i] = 'x';
        lat[i] = 'x';
        lng[i] = 'x';
        spd[i] = 'x';
    }

    trmc = 0; // reset
    i=0;
    often_t = 0;
    serial0_init(4800);
    estab_con1();

    while(1) // Loop Continue

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

{
    buff_ori[i] = serial0_getc();           // receive gps
    detect_data();                         // detect data
    if(find_rmc == 'C')                    // find RMC
    {
        if(res_rmc == 0)
        {
            delay(10);
            keep_rmc2();                   // keep rmc in array
        }
        else if(res_rmc == 1)              // shift 1 data
        {
            res_rmc = 0;                  // reset rmc for keep data rmc
            j2=7;
            k1 = k2 = k3 = k4 = k5 = 0;    // reset value k1,k2,k3,k4,k5
        }
    }
    // end find_rmc
    if(update_rmc==1)
    {
        {
            estab_con2();                 // establish con2
            http_post2();
            ofien_t = 0;
            update_rmc = 0;
        }
    }
    // end loop update_rmc == 1
}
// end loop while(1)
}

int *t_st;

/*****/
/* Initial UART0 = 4800,N,8,1          */
/* VPB(pclk) = 29.4912 MHz              */
/*****/

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

void serial0_init (unsigned long baud)
{
    unsigned long temp;

    PINSEL0 &= 0xFFFFFFF0;           // Reset P0.0,P0.1 PinConfig
    PINSEL0 |= 0x00000001;           // Select P0.0 = TxD
(UART0)
    PINSEL0 |= 0x00000004;           // Select P0.1 = RxD
(UART0)
    U0LCR &= 0xFC;                   // Reset Word Select(1:0)
    U0LCR |= 0x03;                   // data Bit = 8 Bit
    U0LCR &= 0xFB;                   // Stop Bit = 1 Bit
    U0LCR &= 0xF7;                   // Parity = Disable
    U0LCR &= 0xBF;                   // Disable Break Control
    U0LCR |= 0x80;                   // Enable Programming of Divisor latches
    temp = 60000000/16/baud;          // for 2148
    U0DLM = temp / 256;              // Program Divisor latch
(192) for 9600 Baud
    U0DLL = temp % 256;

    U0LCR &= 0x7F;                   // Disable Programming of
Divisor latches
    U0FCR |= 0x01;                   // FIFO Enable
    U0FCR |= 0x02;                   // RX FIFO Reset
    U0FCR |= 0x04;                   // TX FIFO Reset
    U0FCR &= 0x3F;
}

/*****

/* Write Character To UART0 */

*****/

int serial0_putc (int ch)
{
    while (!(U0LSR & 0x20));         // Wait TXD Buffer Empty
    return (U0THR = ch);             // Write Character
}

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

/*****/
/* Read Character From UART0 */
/*****/

int serial0_getc (void)
{
    while (!(U0LSR & 0x01));           // Wait RXD Receive data Ready
    return (U0RBR);                     // Get Receive data & Return
}

void serial0_puts(unsigned char *st)
{
    unsigned char i=0;
    while(st[i] != '\0')
        serial0_putc(st[i++]);
}

void serial0_gets(void)
{
    unsigned char i=0;
    do
    {
        i++;
        t_st[i] = serial0_getc();
    }
    while(t_st[i] != 0x13);
}

/*****/
/* time Function */
/* 1-4294967296 */
/*****/

void delay(unsigned long int count1)
{
    unsigned char t_delay;
    while(count1--)
        for(t_delay=0; t_delay<=200; t_delay++);
}

void detect_data(void)

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

{
    if(buff_ori[i] == 'R')
    {
        find_rmc = 'A';
    }
    if(find_rmc == 'A')
    {
        if(buff_ori[i] == 'M')
        {
            find_rmc = 'B';
        }
    }
    if(find_rmc == 'B')
    {
        if(buff_ori[i] == 'C')
        {
            find_rmc = 'C';
            res_rmc = 1;
            update_rmc = 0;
        }
    }
}

void keep_rmc2(void)
{
    if(buff_ori[i] == ',')
        trmc++;
    if(trmc == 1)
    {
        tim[k1] = buff_ori[i];
        k1++;
    }
    else if(trmc == 3)
    {
        lat[k2] = buff_ori[i];

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

k2++;
if(k2==10)
{
    lat_1 = lat[3];
    lat[3] = lat[5];
    lat_2 = lat[4];
    lat[4] = lat_1;
    lat[5] = lat_2;
}
}

else if(trmc == 5) // '=' 5 keep Longitude
{
    lng[k3] = buff_ori[i];
    k3++;
    if(k3==11)
    {
        lng_1 = lng[4];
        lng[4] = lng[6];
        lng_2 = lng[5];
        lng[5] = lng_1;
        lng[6] = lng_2;
    }
}

else if(trmc == 7) // '=' 7 keep Speed
{
    spd[k4] = buff_ori[i];
    k4++;
}

else if(trmc == 9) // '=' 9 keep date
{
    dat[k5] = buff_ori[i];
    k5++;
}

else if(trmc == 10)

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

{
    often_t++;                                // count round
    trmc = 0;
    find_rmc = 'X';
    update_rmc = 1;                            // rmc = update
}
}

void edit_time(void)
{
    if(tim[1] == '0')
    {
        if(tim[2] <= '2')
        {
            tim[2] = tim[2]+7;
        }
        else
        {
            tim[2] = (tim[2]+7)-10;
            tim[1] = tim[1]+1;
        }
    }
    else if(tim[1] == '1')
    {
        if(tim[2] <= '2')
        {
            tim[2] = tim[2]+7;
        }

        else if(tim[2] <= '6')
        {
            tim[2] = (tim[2]+7)-10;
            tim[1] = tim[1]+1;
        }

        else if(tim[2] > '6')
        {

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

        tim[2] = (tim[2]-7);
        tim[1] = '0';
        edit_date();
    }
}

else if(tim[1] == '2')
{
    tim[2] = tim[2]+3;
    tim[1] = '0';
    edit_date();
}
}

void edit_date(void)
{
    if(dat[2] == '9')
    {
        dat[1] = dat[1]+1;
        dat[2] = '0';
    }

    else    dat[2] = dat[2]+1;
}

void estab_con2(void)
{
    serial0_puts("AT+CIPSHUT");
    serial0_putc(0x0d);
    delay(20000);
    serial0_puts("AT+CIPSTART=\"TCP\", \"mpgps.freetzi.com\", \"80\"");
    serial0_putc(0x0d);
    delay(340000); // wait for make connection
    serial0_puts("AT+CIPSEND");
    serial0_putc(0x0d);
    delay(80000);
}

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

void http_post2(void)
{
    edit_time();
    serial0_puts("GET /conv1new.php?Dat=");
        for(jj=1; jj<=6; jj++){
            serial0_putc(dat[jj]);
        }
    serial0_puts("&Time=");
        for(jj=1; jj<=10; jj++){
            serial0_putc(tim[jj]);
        }
    serial0_puts("&Lat=");
        for(jj=1; jj<=9; jj++){
            serial0_putc(lat[jj]);
        }
    serial0_puts("&Lng=");
        for(jj=1; jj<=10; jj++){
            serial0_putc(lng[jj]);
        }
    serial0_puts("&Spd=");
        for(jj=1; jj<=4; jj++){
            serial0_putc(spd[jj]);
        }
    serial0_puts(" HTTP/1.1");
    serial0_putc(0x0d);                serial0_putc(0x0a);
    serial0_puts("host:mpgps.freetzi.com");
    serial0_putc(0x0d);                serial0_putc(0x0a);
    serial0_putc(0x0d);                serial0_putc(0x0a);
    serial0_putc(0x1a);
    delay(80000);
}

void estab_con1(void)
{
    serial0_puts("AT");

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

serial0_putc(0x0d);
delay(10000);
serial0_puts("AT+CIPCLOSE");
serial0_putc(0x0d);
delay(20000);
serial0_puts("AT+CIPSHUT");
serial0_putc(0x0d);
delay(20000);
serial0_puts("AT+CREG=1");
serial0_putc(0x0d);
delay(20000);
serial0_puts("AT+CGDCONT=1,\"IP\",\"WWW.AIS.CO.TH\"");
serial0_putc(0x0d);
delay(20000);
serial0_puts("AT+CSTT=\"WWW.AIS.CO.TH\"");
serial0_putc(0x0d);
delay(20000);
serial0_puts("AT+CIICR");
serial0_putc(0x0d);
delay(20000);
serial0_puts("AT+CIFSR");
serial0_putc(0x0d);
delay(20000);
serial0_puts("AT+CDNSCFG=\"202.183.255.20\",\"202.183.255.21\"");
serial0_putc(0x0d);
delay(20000);
serial0_puts("AT+CDNSORIP=1");
serial0_putc(0x0d);
delay(20000);
serial0_puts("AT+CIPSTATUS");
serial0_putc(0x0d);
delay(10000);
}

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Code Part Server

1. Code ติดต่อกับฐานข้อมูลกับเว็บเซิร์ฟเวอร์

```
<?
$gd  = $_GET["Dat"];
$gt  = $_GET["Time"];
$gl  = $_GET["Lat"];
$gn  = $_GET["Lng"];
$gs  = $_GET["Spd"];

$z = intval("$gl");
$y = intval("$gn");
$gl -= $z;
$gl *= 100;
$gl /= 60;
$gl += $z;
$gn -= $y;
$gn *= 100;
$gn /= 60;
$gn += $y;
$gs *= 2.21;
$a=mysql_connect("localhost","272548","043443174");
if (!$a)
{
    print("error connect");
}
else
{
    print("ok<br>");
    mysql_select_db("272548",$a);
    $bat="insert into gpsa (Dat,Time,Lat,Lng,Spd)
        values($gd,$gt,$gl,$gn,$gs)";

    mysql_query ($bat,$a);
    mysql_close($a);
}
?>
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.แสดงหมุดและแสดงแผนที่

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN"
```

```
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd">
```

```
<html xmlns="http://www.w3.org/1999/xhtml">
```

```
<head>
```

```
<meta http-equiv="content-type" content="text/html; charset=utf-8"/>
```

```
<title>Maps</title>
```

```
<script
```

```
src="http://maps.google.com/maps?file=api&v=2&key=ABQIAAAQmi21RJUUpFG9ZUPakqaYBTN4xzo8W  
BnmLiLO_LnzcZk8yu1IBRP1ldXT6R76G-zOqAlguziekdhQw"
```

```
type="text/javascript"></script>
```

```
<script type="text/javascript">
```

```
var iconblue = new GIcon();
```

```
iconblue.image = 'http://labs.google.com/ridefinder/images/mm_20_blue.png';
```

```
iconblue.iconSize = new GSize(12, 20);
```

```
iconblue.shadowSize = new GSize(22, 20);
```

```
iconblue.iconAnchor = new GPoint(6, 20);
```

```
iconblue.infoWindowAnchor = new GPoint(5, 1);
```

```
var iconred = new GIcon();
```

```
iconred.image = 'http://labs.google.com/ridefinder/images/mm_20_red.png';
```

```
iconred.iconSize = new GSize(12, 20);
```

```
iconred.shadowSize = new GSize(22, 20);
```

```
iconred.iconAnchor = new GPoint(6, 20);
```

```
iconred.infoWindowAnchor = new GPoint(5, 1);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

var customIcons = [];

customIcons["1"] = iconblue;

customIcons["2"] = iconred;

function load() {

    if (GBrowserIsCompatible()) {

        var map = new GMap2(document.getElementById("map"));

        map.addControl(new GSmallMapControl());

        map.addControl(new GMapTypeControl());

        map.setCenter(new GLatLng(13.7225, 100.76328), 15);

        GDownloadUrl("xmlallv1.php", function(data) {

            var xml = GXml.parse(data);

            var markers = xml.documentElement.getElementsByTagName("marker");

            for (var i = 0; i < markers.length; i++) {

                var Dat = markers[i].getAttribute("Dat");

                var Time = markers[i].getAttribute("Time");

                var Spd = markers[i].getAttribute("Spd");

                var type = markers[i].getAttribute("type");

                var point = new GLatLng(parseFloat(markers[i].getAttribute("lat")),

                    parseFloat(markers[i].getAttribute("lng")));

                var marker2 = createMarker(point, Dat, Time, Spd, type);

                map.addOverlay(marker2);

            }

        });
    }
}

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

GDownloadUrl("xmlallv1new.php", function(data) {

    var xml = GXml.parse(data);

    var markers = xml.documentElement.getElementsByTagName("marker");

    for (var i = 0; i < markers.length; i++) {

        var Dat = markers[i].getAttribute("Dat");

        var Time = markers[i].getAttribute("Time");

        var Spd = markers[i].getAttribute("Spd");

        var type = markers[i].getAttribute("type");

        var point = new GLatLng(parseFloat(markers[i].getAttribute("lat")),
                                parseFloat(markers[i].getAttribute("lng")));

        var marker3 = createMarker(point, Dat, Time, Spd, type);

        map.addOverlay(marker3);

    }

});

}

}

function createMarker(point, Dat, Time, Spd, type) {

    var marker2 = new GMarker(point, customIcons[type]);

    var html = "<b> Date "+Dat+"</b><br>"+Time+"<b>"+Time+"</b><br>"+point
    "+"<b>"+point+"</b><br>"+Spd+"<b>"+Spd+"</b>";

    GEvent.addListener(marker2, 'click', function() {

        marker2.openInfoWindowHtml(html);

    });

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

return marker2;

}

</script>

<META HTTP-EQUIV="Refresh" Content="5;URL=test.php">

</head>

<body onload="load()" onunload="GUnload()">

<div id="map" style="width: 1300px; height: 500px"></div>

</body>

</html>

```

2. Code ดึงข้อมูลจากเซิร์ฟเวอร์มาแสดงบนเว็บเซิร์ฟเวอร์

```

<?php

$username="272548";

$password="043443174";

$dbase="272548";

function parseToXML($htmlStr)

{

$xmlStr=str_replace('<','&lt;',$htmlStr);

$xmlStr=str_replace('>','&gt;',$htmlStr);

$xmlStr=str_replace('"','&quot;',$htmlStr);

$xmlStr=str_replace("'",'&apos;',$htmlStr);

$xmlStr=str_replace('&','&amp;',$htmlStr);

return $xmlStr;

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

}          // Opens a connection to a MySQL server

$connection=mysql_connect (localhost, $username, $password);

if (!$connection) {

    die('Not connected : ' . mysql_error());

}          // Set the active MySQL database

$db_selected = mysql_select_db("272548", $connection);

if (!$db_selected) {

    die ('Can\'t use db : ' . mysql_error());

}          // Select all the rows in the markers table

$query = "SELECT * FROM gpsa";
$result = mysql_query($query);
if (!$result) {

    die('Invalid query: ' . mysql_error());

}

header("Content-type: text/xml; charset=utf-8"); // Start XML file, echo parent node

echo '<markers>';          // Iterate through the rows, printing XML nodes for each

while ($row = @mysql_fetch_assoc($result)){ // ADD TO XML DOCUMENT NODE

    echo '<marker '

    echo 'type="1" '

    echo 'Dat="' . $row['Dat'] . '" '

    echo 'Time="' . $row['Time'] . '" '

    echo 'Spd="' . $row['Spd'] . '" '

    echo 'lat="' . $row['Lat'] . '" '

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
echo 'lng="" . $row['Lng'] . "" ';
```

```
echo '>';
```

```
} // End XML file
```

```
echo '</markers>';
```

```
?>
```



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้