

สำนักหอสมุดกลาง พระจอมเกล้าลาดกระบัง

เครื่องระบุตำแหน่ง GPS และบันทึกเส้นทางผ่าน GPRS

GPS Location Display and Record Using GPRS



ภาควิชา
วิศวกรรมโทรคมนาคม

ผ่านการตรวจชิ้นงานแล้ว

(ลงชื่อ).....ผู้ตรวจ

เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์ของสถาบันฯ ไม่ให้นำไปเผยแพร่โดยไม่ได้รับอนุญาต
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เครื่องระบุตำแหน่ง GPS และบันทึกเส้นทางผ่าน GPRS

GPS Location Display and Record Using GPRS



ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต

สาขาวิชาวิศวกรรมโทรคมนาคม

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2549

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ปริญญานิพนธ์ปีการศึกษา 2549

ภาควิชาวิศวกรรมโทรคมนาคม

คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เรื่อง เครื่องระบุตำแหน่ง GPS และบันทึกเส้นทางผ่าน GPRS

GPS Location Display and Record Using GPRS

ผู้จัดทำ

1. นาย ชีรพร ไพทยะทัต 46012012
2. นาย ฤทธิ์ กาไชย 46012026


(ผศ.ดร.พิพัฒน์ พรหมมี)

อาจารย์ที่ปรึกษา



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เครื่องระบุตำแหน่งGPS และบันทึกเส้นทางผ่าน GPRS

GPS LOCATION DISPLAY AND RECORD USING GPRS

โดย นายธีรพร ไพฑูระทัต 46012012
นายฤทธิ์ กาไชย 46012026

อาจารย์ที่ปรึกษา ผศ.ครพิพัฒน์ พรหมมี

บทคัดย่อ

โครงการนี้นำเสนอเครื่องระบุตำแหน่งจีพีเอสและบันทึกเส้นทางผ่านจีพีอาร์เอส ประกอบด้วยแผงวงจรมicrocontroller board, GPS module, data logger and GPRS module. This system operates by receiving position data from GPS module. It can be memorized all position data into data logger in desired period of time. This data transmits over GPRS network to the receiver. The location data is displayed as a map on the computer.

ABSTRACT

This project is presents a GPS location display and record using GPRS. It consists of a microcontroller circuit board, GPS module, data logger and GPRS module. This system operates by receiving position data from GPS module. It can be memorized all position data into data logger in desired period of time. This data transmits over GPRS network to the receiver. The location data is displayed as a map on the computer.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ

	หน้า
บทที่ 1 บทนำ	1
บทที่ 2 ทฤษฎีและหลักการ	2
2.1 ระบบกำหนดตำแหน่งบนพื้นโลก (Global Positioning System: GPS)	2
2.2 ส่วนประกอบหลักของ GPS	4
2.2.1 ส่วนอวกาศ (Space segment)	4
2.2.2 ส่วนสถานีควบคุม (Control segment)	6
2.2.3 ส่วนผู้ใช้ (User segment)	7
2.3 วิธีการหาพิกัดตำแหน่ง โดยใช้ระบบกำหนดตำแหน่งบนโลก	9
2.4 ความถูกต้อง (ACCURACY)	11
2.5 ความแม่นยำของระบบ	11
2.6 โครงสร้างของข้อมูลที่ส่งจากดาวเทียม	13
2.7 การทำงานของเครื่องรับสัญญาณจีพีเอส	14
2.8 การให้บริการจีพีเอส	15
2.9 ทฤษฎีการปฏิบัติงานระบบ จีพีเอส	16
2.10 มาตรฐาน NMEA 0183	18
2.11 มาตรฐานการเชื่อมต่อทางคลื่นวิทยุของเครือข่าย GPRS	23
2.12 โครงสร้างมัลติเฟรมของช่องสื่อสารบนมาตรฐาน GPRS	29
2.13 เซลล์และ Routing Area	32
2.14 กระบวนการแสดงตัวต่อเครือข่าย GPRS	33
2.15 แบบจำลองโปรโตคอลของเครือข่าย GPRS	36
2.16 มาตรฐานการรักษาความปลอดภัยกับเครือข่ายโทรศัพท์เคลื่อนที่ GPRS	37
2.17 ทฤษฎีเกี่ยวกับโมโครบคอนโทรลเลอร์และการประยุกต์ใช้งาน	39
2.18 การใช้โปรแกรม Keil CS1 เบื้องต้น	50
บทที่ 3 การออกแบบและการสร้าง	55
3.1 โครงสร้างการทำงานของระบบ	55
3.2 ส่วนของเครื่องรับสัญญาณ	56
3.3 ส่วนของไมโครคอนโทรลเลอร์	57
3.4 ส่วนของการทำงานของโปรแกรม	59
3.5 การเขียนโปรแกรมย่อย	60

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ(ต่อ)

	หน้า
บทที่ 4 ผลการทดลอง	
4.1 ทดสอบการรับค่าจากโมดูลเครื่องรับ GPS	63
4.2 ทดสอบการรับค่าจากโมดูลเครื่องรับ GPS โดยผ่านทางบอร์ดMcs-51	64
4.3 ทดสอบการรับค่าพิกัดตำแหน่งและทำการบันทึกลงในหน่วยความจำภายนอก	65
4.4 ทดลองนำผลที่ได้จากการบันทึกใน EEPROM ผ่านทางพอร์ต RS-232	66
4.5 ทดลองใช้คำสั่ง AT COMMAND เพื่อเชื่อมต่อกับเครือข่าย	72
4.6 ทดลองส่งค่าพิกัดผ่าน GPRS MODULE ไปยังเครื่องเซิร์ฟเวอร์	72

บทที่ 5 บทวิจารณ์และสรุป	74
--------------------------	----

ภาคผนวก	75
---------	----



สารบัญรูปภาพ

	หน้า
รูปที่ 2.1 แสดงรูปดาวเทียมจีพีเอสบล็อกทู	2
รูปที่ 2.2 แสดงรูปดาวเทียมแนฟสตาร์	3
รูปที่ 2.3 ส่วนประกอบ 3 ส่วนของ GPS	4
รูปที่ 2.4 แสดงกลุ่มดาวเทียมจีพีเอส	5
รูปที่ 2.5 แสดงการโคจรของดาวเทียม GPS รอบโลก	5
รูปที่ 2.6 แสดงที่ตั้งของสถานีควบคุมดาวเทียมจีพีเอส	7
รูปที่ 2.7 แสดงการวัดตำแหน่งบนพื้นโลก	10
รูปที่ 2.8 แสดงลักษณะการตัดกันของระนาบดาวเทียมอันเนื่องมาจากเครื่องรับ	11
รูปที่ 2.9 แสดงการทำงานของระบบ GPS	13
รูปที่ 2.10 โครงสร้างสัญญาณจากดาวเทียมจีพีเอส	17
รูปที่ 2.11 รายละเอียดของจุดเชื่อมต่อและ โพรโตคอลที่เกี่ยวข้อง ระหว่างอุปกรณ์ภายในเครือข่ายสถานีฐานของมาตรฐานโทรศัพท์เคลื่อนที่GPRS	23
รูปที่ 2.12 การจัดสรรและบริหารจัดการช่องเวลาตามมาตรฐาน โทรศัพท์เคลื่อนที่ GSM ให้รองรับการสื่อสารด้วยมาตรฐาน GSM	25
รูปที่ 2.13 การแบ่งกลุ่มของช่องสื่อสารทางตรงตามมาตรฐาน โทรศัพท์เคลื่อนที่ GPRS	26
รูปที่ 2.14 การเกิดปัญหาช่องเวลาไม่ตรงกัน และการแก้ไขโดยกระบวนการ Timing Advance	28
รูปที่ 2.15 โครงสร้างมัลติเฟรมของช่องสื่อสารตามมาตรฐาน GPRS โดยขยายแนวคิดจากโครงสร้างเฟรม ตามมาตรฐานจาก โทรศัพท์เคลื่อนที่ GSM	29
รูปที่ 2.16 ตัวอย่างการจัดโครงสร้างเฟรมข้อมูลGPRSชนิดควบคุมประเภทPBCCH+PCCCH	30
รูปที่ 3.1 โครงสร้างการทำงานของระบบ	55
รูปที่ 3.2 แสดงบล็อกไดอะแกรมของระบบ	56
รูปที่ 3.3 เครื่องรับ GPS ที่เลือกใช้รุ่น GM-82	56
รูปที่ 3.4 วงจรการเชื่อมต่อพอร์ตต่างๆของMCS-51	57
รูปที่ 3.5 วงจรการเชื่อมต่อกับRS-232 ของMCS-51	57
รูปที่ 3.6 วงจรการเชื่อมต่อกับ LCD	58
รูปที่ 3.7 วงจรการเชื่อมต่อกับ EEPROM	58
รูปที่ 3.8 บอร์ดไมโครคอนโทรลเลอร์ที่ใช้ในโครงการ	58
รูปที่ 3.9 แสดงผังการทำงานของส่วนโปรแกรม	59
รูปที่ 3.10 แสดงวงจรภายในของ Timer/Counter#0	60
รูปที่ 3.11 แสดงวงจรภายในของTimer/Counter#1	61

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูป(ต่อ)

	หน้า
รูปที่ 4.1 ผลการรับข้อมูลจากเครื่องรับ GPS	63
รูปที่ 4.2 ผลการรับข้อมูลในส่วนของ RMC โปรโตคอล	64
รูปที่ 4.3 ผลการนำข้อมูลพิกัดตำแหน่งแสดงบน LCD	64
รูปที่ 4.4 แสดงการทดลองติดตั้งจริงในรถยนต์	65
รูปที่ 4.5 แสดงผลการนำค่าพิกัดไปพล็อตลงบนโปรแกรมแผนที่	66-69
รูปที่ 4.6 แสดงการเชื่อมต่อกับเครือข่ายโดยคำสั่ง AT COMMAND	72
รูปที่ 4.7 แสดงผลการส่งค่าพิกัดไปยังเครื่องเซิร์ฟเวอร์	72
รูปที่ 4.8 แสดงรูปอุปกรณ์ภายในกล่องชุดอุปกรณ์	73
รูปที่ 4.9 แสดงกล่องอุปกรณ์เครื่องระบุตำแหน่งและบันทึกเส้นทางที่เสร็จสมบูรณ์	74

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 1

บทนำ

ปัจจุบันยุคสมัยของการสื่อสารข้อมูลจัดได้ว่าเป็นยุคแห่งเทคโนโลยี ซึ่งมีการพัฒนาระบบสื่อสารโทรคมนาคมและเทคโนโลยีต่างๆ ขึ้นมามากมายหลายรูปแบบ สำหรับจีพีเอส(GPS:Global Positioning System) ก็เป็นอุปกรณ์อีกชนิดหนึ่งที่สร้างขึ้นมามาเพื่อใช้ประโยชน์ในการบอกพิกัดตำแหน่ง จีพีเอสมีคุณสมบัติ คือสามารถบอกพิกัดตำแหน่ง,ทิศทาง,ความสูงและความเร็ว รวมทั้งข้อมูลต่างๆของดาวเทียมได้

สำหรับปฏิญานิพนธ์นี้ได้นำจีพีเอสไปประยุกต์ใช้ในการบอกตำแหน่งของรถยนต์ ซึ่งการใช้ประโยชน์นั้นขึ้นอยู่กับผู้ใช้งานว่าจะนำไปประยุกต์ใช้ เช่นนำไปใช้ในการบอกเส้นทางการเดินทางหรือนำประยุกต์เป็นอุปกรณ์กันขโมยและติดตามรถยนต์ ปฏิญานิพนธ์นี้มุ่งเน้นไปทางบอกตำแหน่งของรถยนต์คันนี้อยู่ที่ตำแหน่งใดและค้นหาตำแหน่งของรถ ณเวลาปัจจุบัน และทำการบันทึกเส้นทางที่เคลื่อนที่ของรถ

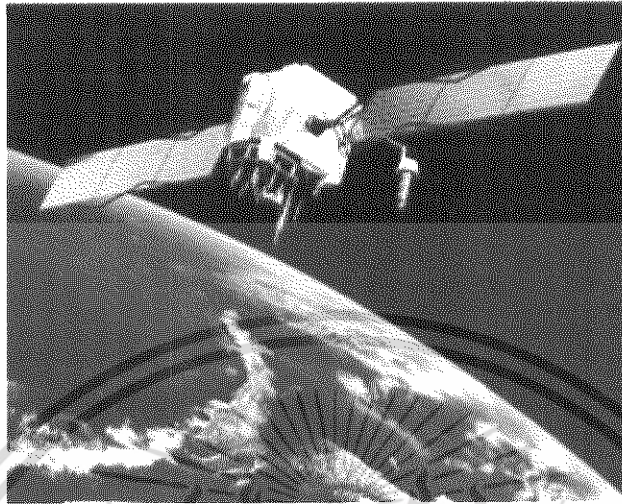


เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 2

ทฤษฎีและหลักการ

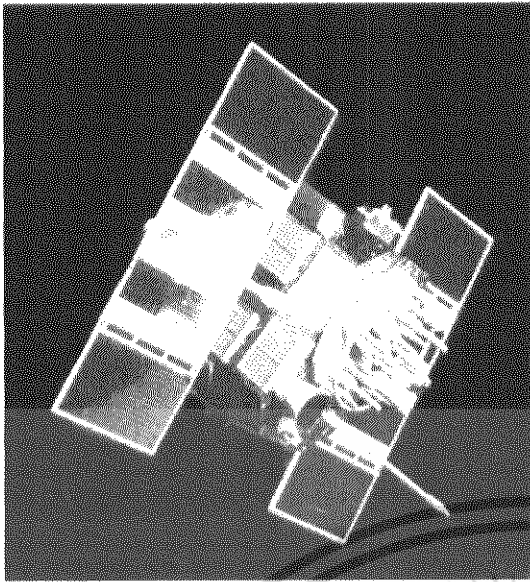
2.1 ระบบพิภักดบนพื้น โลกจีพีเอส (Global Positioning System : GPS)



รูปที่ 2.1 แสดงรูปดาวเทียมจีพีเอสในอวกาศ

จีพีเอสเป็นระบบหาพิภักดบนพื้น โลกโดยการอ้างอิงจากดาวเทียมที่มีความแม่นยำสูงสามารถใช้หาพิภักดใดๆ บนพื้น โลกได้ทุกสภาพอากาศ ซึ่งในปัจจุบันเป็นเทคโนโลยีที่เริ่มเข้ามามีบทบาทในชีวิตประจำวันมากขึ้น มีการนำไปใช้ในกิจกรรมต่างๆ มากมาย เช่น การช่วยวางแผนเดินทางด้วยแผนที่ (โดยรถยนต์) ระบบติดตามแท่งงานพื้นผิวโลก เป็นต้น จึงได้มีการวิจัยและพัฒนาเทคโนโลยีพื้นฐานของจีพีเอส (Core Technology) ซึ่งมุ่งเน้นในการพัฒนาออกแบบวงจรรวมพื้นฐานที่ใช้ในระบบจีพีเอส ทั้งในรูปแบบทรัพย์สินทางปัญญา (Intellectual Property) หรือ ไอพี คอร์ (IP core) และการทำเป็นชิพในขั้นสุดท้ายด้วย จีพีเอสถูกพัฒนาขึ้นมาโดยหน่วยงานต่อต้านการรุกรานของสหรัฐอเมริกาหรือ ดีไอดี (The United States Department of defence : DOD) ใช้ในการหาพิภักดตำแหน่งในที่ต่างๆ บน โลกด้วยดาวเทียม โดยการใช้ส่งสัญญาณคลื่นวิทยุลงมายบน โลก เมื่อเครื่องรับจีพีเอสรับสัญญาณ ได้ จึงเอาข้อมูลต่างๆ ที่มากับสัญญาณจีพีเอส ไปคำนวณหาตำแหน่งเพื่อประโยชน์ทางการทหาร และหน่วยงานราชการบางหน่วยของสหรัฐอเมริกา และพัฒนาระบบจนสมบูรณ์ในปี 1993 ในขณะที่ระบบยังไม่สมบูรณ์ แต่ได้มีการใช้อย่างจริงจังครั้งแรกในช่วงสงครามอ่าวในปี 1991 ต่อมาจึงเปิดโอกาสให้ภาคเอกชน และหน่วยงานต่างๆ นำระบบไปใช้อย่างแพร่หลายทั่วโลก แต่ ดีไอดี ยังเปิดสัญญาณการเลือกที่เหมาะสมหรือเอสเอ (Selective Available : SA) เพื่อลดความถูกต้องของการบอกตำแหน่งเพื่อประโยชน์ทางการทหาร และเปิดสัญญาณเมื่อวันที่ 1 พฤษภาคม 2000 ด้วยเหตุผลทางการตลาดทำให้การบอกตำแหน่งมีความถูกต้องแม่นยำมากขึ้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



GPS Satellites

Name:	NAVSTAR
Manufacturer:	Rockwell International
Altitude:	10,900 nautical miles
Weight:	1900 lbs. (in orbit)
Size:	17 ft with solar panels extended
Orbital Period:	12 hours
Orbital Plane:	65° to equatorial plane
Planned Lifespan:	7.5 years
Number built:	11 Block I prototype satellites 28 Block II production satellites
Constellation:	24 satellites

รูปที่ 2.2 แสดงรูปดาวเทียมเนฟสตาร์

จีพีเอสสามารถหาตำแหน่งได้โดยใช้สัญญาณจากดาวเทียม จีพีเอส ที่ส่งไปโคจรรอบโลก ดาวเทียมจีพีเอสชุดแรกเป็นชุดสำหรับทดลองเรียกว่าบล็อกวัน (Block I) บล็อกวันมีทั้งหมด 10 ดวง ดวงแรกถูกส่งขึ้นในปี 1978 และทยอยส่งจดหมดในปี 1988 จากนั้นในปี 1989 – 1994 ดาวเทียมจีพีเอสที่จะใช้งานจริง (Block II) ก็ถูกส่งโคจรรอบโลกทั้งหมด 24 ดวง (ใช้งาน 21 ดวง สำรองในวงโคจร 3 ดวง) นอกจากนี้ยังมีอีก 4 ดวงเป็นตัวสำรองซึ่งพร้อมที่จะส่งเข้าวงโคจรหากจำเป็น อายุการใช้งานของดาวเทียมประมาณ 7.5 ปี ดาวเทียมทั้งหมดโคจรรอบโลกที่ความสูง 10900 ไมล์ทะเล (20200 km) มีคาบโคจรรอบโลกประมาณ 11 ชั่วโมง 58 นาที โดยเคลื่อนที่ในแนวระนาบ 6 ระนาบ ระนาบละ 4 ดวง มีมุมเอียง (Inclination angle) 55 องศา การจัดวางวงโคจรแบบนี้ทำให้ทุกที่ ที่บนพื้นโลกสามารถรับสัญญาณจากดาวเทียมได้ 6 ดวงเกือบจะ 100% ดาวเทียมเหล่านี้จะเป็นจุดอ้างอิงที่ใช้ในการหาตำแหน่ง มันก็เลยต้องเคลื่อนที่อยู่ในวงโคจรที่คงที่เสถียร บนพื้นโลกก็เลยมีสถานีตรวจวัด ปรับวงโคจรของดาวเทียมให้อยู่ในตำแหน่งที่ถูกต้อง และปรับตั้งนาฬิกาในดาวเทียมให้เดินตรงเสมอสถานีตรวจสอบสำหรับดาวเทียมจีพีเอสมีทั้งหมด 5 แห่ง คือที่ ฮาวาย (Hawaii) , แอสเซนชัน ไอส์แลนด์ (Ascension Island) , ดีเอโก การ์เซีย (Diego Garcia) , ควาจาเลน (Kwajalein) และฟอลคิล เอ เอฟ บี (Falcon AFB) ที่โคโลราโด สปรिंगส์ (Colorado Springs) ซึ่งเป็นสถานีควบคุมหลัก (Master control) ของจีพีเอส

ดาวเทียมแต่ละดวงจะติดตั้งนาฬิกาอะตอมซึ่งตั้งเวลาไว้ให้ตรงกันทุกดวง ดาวเทียมจีพีเอสจะถ่ายทอดสัญญาณในรูปแบบของคลื่นวิทยุตลอดเวลา คลื่นวิทยุที่จะส่งออกมาสองความถี่ คือ $L1 = 1.2276$ GHz และ $L2 = 1.57542$ GHz (ที่เรียก $L1/L2$ เพราะเป็นความถี่ย่าน L-Band) ดาวเทียมทุกดวงจะมีวงจรสำหรับสร้างรหัส ซูด แรนดอม (Pseudo random code) ออกมาด้วยอัตรา 1.023 Mbps แต่รหัสจะแตกต่างกันไปตามอัลกอริทึม (algorithm) ที่เลือกใช้ รหัสนี้จะเอามา modulate กับคลื่นพาห์ (carrier) $L1/L2$ แล้วก็จะมาพื้นโลก การมอดูเลตรหัส ซูด แรนดอม กับ $L1/L2$ ทำให้สัญญาณวิทยุอื่นๆ ไม่สามารถชิงใครในซ์

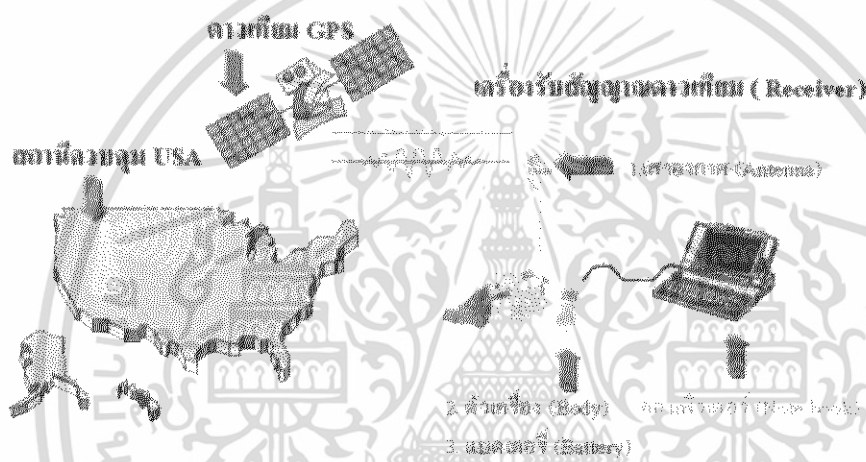
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

(synchronize) กับ L1/L2 ได้ ดาวเทียมจีพีเอสทุกดวงจึงสามารถใช้ความถี่ในการส่งสัญญาณเพียงชุดเดียว เพราะรหัส ชูโค แรนคอม ที่ดาวเทียมแต่ละดวงสร้าง (generate) ขึ้นมาจะเป็นตัวบ่งชี้ได้ว่าสัญญาณที่เครื่องรับ (receiver) ได้รับมาจากดาวเทียมดวงไหน และสัญญาณที่ดาวเทียมจีพีเอสส่งมานี้เองที่เราจะเอาไปคำนวณหาตำแหน่งบนพื้นโลกได้

2.2 ส่วนประกอบของระบบจีพีเอส

ระบบจีพีเอสประกอบไปด้วย 3 ส่วนดังนี้คือ

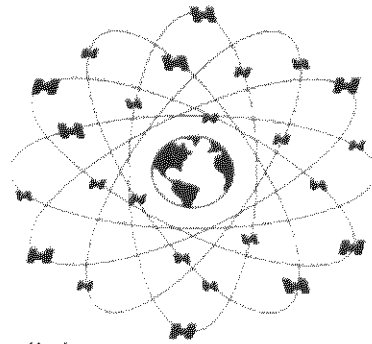
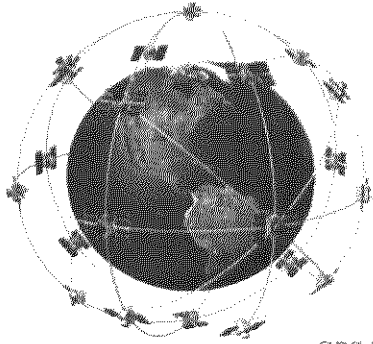
- ส่วนอวกาศ
- ส่วนสถานีควบคุม
- ส่วนผู้ใช้



รูปที่ 2.3 ส่วนประกอบ 3 ส่วนของ GPS

2.2.1 ส่วนอวกาศ (Satellite Constellation) ของระบบจีพีเอส ประกอบไปด้วยดาวเทียม 24 ดวง โดยใช้บรอดพิคค 21 ดวง ส่วนอีก 3 ดวงจะสำรองเอาไว้ซึ่งดาวเทียมทั้ง 24 ดวงนี้จะ โคจรรอบโลกโดยรัศมีวงโคจรจะสูงจากพื้นโลก 20,200 กม. โดยประมาณ ด้วยวงโคจร Non-Geostationary orbit 6 วงโคจร และวงประกอบด้วยดาวเทียม 4 ดวง แต่ละวงโคจรเรียงทำมุม 55 องศาการบอกตำแหน่งบนพื้นโลกจะรับสัญญาณดาวเทียมได้อย่างน้อย 4 ดวงถ้ามีมุมเมย 15 องศาขึ้นไปและสามารถรับสัญญาณนาฬิกาที่มีความเที่ยงตรงสูง และตรงกันระหว่างเครื่องรับจีพีเอสกับดาวเทียมจีพีเอส ซึ่งในจีพีเอสนี้จะใช้สัญญาณนาฬิกาจากอะตอม ในดาวเทียมจีพีเอสบล็อก ที จะมินาฬิกาอะตอมแบบรูบิเดียม 2 เครื่อง และแบบซีเซียม 2 เครื่องซึ่งจะให้ความผิดพลาดต่อวันไม่เกิน 10^{-14} วินาที (เนื่องจากความผิดพลาดของเวลาเพียง 10^{-14} วินาที จะทำให้เกิดความผิดพลาดของตำแหน่งถึง 30 เซนติเมตร) ดาวเทียมจะใช้สัญญาณจากนาฬิกาสร้างเป็นคล็อก (clock) สร้างความถี่พื้นฐาน 10.23 MHz แล้วสร้างความถี่พื้นฐานในแบนด์ L1 และ L2

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



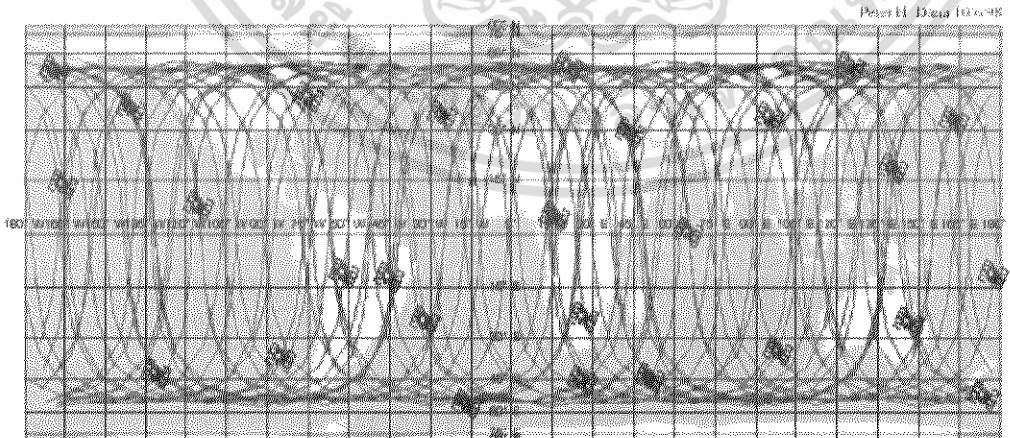
GPS Nominal Constellation

24 Satellites in 6 Orbital Planes

4 Satellites in each Plane

20,200 km Altitude, 55 Degree Inclination

รูปที่ 2.4 แสดงกลุ่มดาวเทียมจีพีเอส โดยมีระนาบโคจร 6 ระนาบ แต่ละระนาบจะมีดาวเทียมอยู่ 4 ดวง โดยการคูณความถี่พื้นฐานด้วย 154 และ 120 ตามลำดับ ดังนั้นความถี่ของคลื่นพาห์ในแบนด์ L1 จะเท่ากับ 1,575.42 MHz และความถี่ของคลื่นพาห์ในแบนด์ L2 คือ 1,227.60 MHz การคำนวณหาระยะทางจากดาวเทียมมาถึงเครื่องรับเรียกว่า ซูโด - เรนจ (Pseudo - Range) จะคำนวณจากระยะเวลาที่สัญญาณทั้งสองที่อยู่ในรูปแบบรบกวน (Pseudorandom Noise : PRN) ที่ผสมกับคลื่นพาห์ทั้งสองแบนด์ เดินทางจากดาวเทียมถึงเครื่องรับสัญญาณรหัสแรกที่ส่งเรียกว่ารหัส ซีเอ (Coares/Acquis - Code : C/A Code) เป็นสัญญาณที่อนุญาตให้บุคคลทั่วไปใช้งานสัญญาณจะผสมออกมาที่แบนด์ L1 เท่านั้น สัญญาณรหัสที่สองคือ พีโค้ด (Precision Code) เป็นสัญญาณที่ใช้ในทางทหารสหรัฐและบุคคลที่ได้รับอนุญาตเท่านั้น สัญญาณที่ผสมจะออกมาทั้ง 2 แบนด์คือ L1 และ L2 ทำให้การระบุตำแหน่งมีความแม่นยำกว่า ซึ่งก็มีการเข้ารหัสไว้เพื่อรักษาความปลอดภัยของการทหารสหรัฐอเมริกา ในคลื่นพาห์ L1 นั้นใช้สำหรับการส่งสัญญาณทั้ง 2 แบบจึงต้องมีการซิงค์เฟสของแคเรียร์ เพื่อมิให้สัญญาณรบกวนกัน หลังจากนั้นรหัสพ็อดร็อนด์ จะมียุคข้อมูลอื่นๆ ที่ผสมมาในสัญญาณด้วยได้แก่ ข้อมูลที่เกี่ยวข้องกับวงโคจรดาวเทียมสัมประสิทธิ์สภาพชั้นบรรยากาศไอโอโนสเฟียร์สภาพการใช้งานของดาวเทียม เวลาของระบบและสัญญาณเวลาของดาวเทียม เป็นต้น รหัสพี และรหัสซีเอ



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 2.5 แสดงการโคจรของดาวเทียม GPS รอบโลก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

นั้นได้ถูกออกแบบมาเพื่อไม่ให้มีการรบกวนกัน โดย พี- โคลด์ จะถูกมอดูเลตกับคลื่นพาห์พร้อมๆ กับ ซีเอ โคลด์ ที่ถูกมอดูเลตกับสัญญาณของคลื่นพาห์ที่ถูกซิมพลิฟายไป 90 องศา โดยสัญญาณของคลื่นพาห์ที่ถูกมอดูเลตไปทั้งสองนี้จะรวมกันด้วยวิธี เฟส ควอดราเจอร์ (Phase Quadrature) และข้อมูลที่ถูกส่งมาจากความถี่นั้นจะถูกมอดูโล - 2 (Modulo - 2) ลงบน พี - โคลด์ หรือลำดับของรหัสซีเอ (C/A Code State Sequence) ของพี โคลด์, ซีเอ โคลด์และข้อมูลนำร่อง (navigation message) คือ P(T), C/A(t) และ D(t) ตามลำดับ โดยการมอดูเลตคลื่นพาห์แสดงได้ดังต่อไปนี้

$$L_1(t) = a_1 P(t) D(t) \cos(f_1 t) + a_1 C/A(t) D(t) \sin(f_1 t)$$

$$L_2(t) = a_2 P(t) D(t) \cos(f_2 t)$$

และการ ดีมอดูเลท (Demodulate) คลื่นพาห์โดย

$$L_1(t) = a_1 \cos(f_1 t)$$

2.2.2 ส่วนสถานีควบคุม (Ground Control Segment) มีส่วนรับผิดชอบการทำงานของดาวเทียมจีพีเอส เช่น การรักษาตำแหน่งการโคจรของดาวเทียม ตรวจสอบสถานะและระบบต่างๆบนดาวเทียม ตรวจสอบแรงเสียดทานระดับพลังงานของแบตเตอรี่ การเปิดดาวเทียมสำรอง ปรับปรุงข้อมูลเวลา ข้อมูลอีฟิเมอร์ซิส (Ephemeris) ข้อมูลอัลมาเนก (Almanac) และตัวชี้ค่าอื่นๆในข่าวสารการนำร่องวันละครั้งหรือตามแต่ความจำเป็น สถานีควบคุมนี้จะมีนาฬิกาอะตอมแบบซีเซียมที่มีความแม่นยำสูงและเครื่องรับที่สามารถคำนวณ พี โคลด์ ซูโดเรนจ์ (P-Code Pseudo-Range) ของดาวเทียมทุกดวงรับสัญญาณได้ ข้อมูลต่างๆ ที่ต้องการจะถูกคำนวณแล้วส่งไปยังสถานีควบคุม ซึ่งสามารถที่จะนำข้อมูลมาหาตัวแปรต่างๆ จากนั้นส่งกลับไปยังดาวเทียมจีพีเอสเพื่อแก้ไขข้อผิดพลาดต่างๆ

2.2.2.1 สถานีสังเกตการณ์ (Monitoring Station) สถานีสังเกตการณ์จะมีเครื่องรับสัญญาณจีพีเอส ทั้งสองความถี่ (L1 & L2) โดยจะทำการวัดค่าซูโดเรนจ์ และเคลตเรนจ์ ของแต่ละดาวเทียมที่ผ่านสถานีและมีนาฬิกาซีเซียม (Cesium) สองตัวที่ตั้งเวลาเพื่อใช้ในการอ้างอิงกับเวลาของระบบจีพีเอส สัญญาณจากดาวเทียมที่ส่งมาถึงสถานีสังเกตการณ์นั้นมีการหักเหและล่าช้าในชั้นบรรยากาศไอโอโนสเฟียร์ และโทรโพสเฟียร์เรียกการล่าช้านี้ว่า ไอโอโนสเฟียร์ดีเลย์ (Ionospheric Delay) และโทรโพสเฟียร์ดีเลย์ (Tropospheric Delay) การล่าช้านี้จะทำให้เกิดการผิดพลาดของข้อมูล ซึ่งการแก้ไขนั้นสถานีสังเกตการณ์จะรวบรวมข้อมูลจากสัญญาณที่ได้รับทั้ง 2 ความถี่ อุณหภูมิ ความดันบรรยากาศ ความชื้นสัมพัทธ์ และจะส่งไปยังสถานีควบคุมหลักโดยกรมอุตุนิยมวิทยาของสหรัฐอเมริกา เพื่อทำการคำนวณหาค่าความผิดพลาดและหาข้อมูลที่ต้องใช้ต่อไป

2.2.2.2 สถานีควบคุมหลัก (Master Control Station) สถานีควบคุมหลักมีหน้าที่ในการประมวลผลข้อมูล ที่ได้รับจากสถานีสังเกตการณ์เพื่อตรวจสอบ และกำหนดค่าสัญญาณนาฬิกาของดาวเทียมข้อมูลอัลมาเนก ข้อมูลอีฟิเมอร์ซิสให้ถูกต้อง โดยเริ่มจากแก้ไขค่าซูโดเรนจ์ที่เกิดจากการล่าช้าเนื่องจากการผ่านชั้นบรรยากาศของทุกๆ สถานีสังเกตการณ์ จากนั้นจึงนำไปผ่านคาลมานฟิลเตอร์ (Kalman Filter) เพื่อนำค่าอีฟิเมอร์ซิสและค่าการเลื่อมของสัญญาณนาฬิกาที่ถูกต้อง โดยฟิลเตอร์จะถูกอัปเดตทุกๆ 15 นาที ด้วยค่าตำแหน่งของดาวเทียมที่ถูกคำนวณ ในระบบ โคออดิเนตแบบเอิร์ธเซนเตอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เอิร์ทฟิฟิกซ์ (Earth-Center Earth-Fixed : ECEF) สถานีควบคุมหลักจะเป็นศูนย์กลางในการทำงานของส่วนควบคุม ตั้งอยู่ที่ฐานทัพอากาศ Falcon, Colorado, Spring, CO. ส่วนสถานีสังเกตการณ์จะกระจายอยู่ตามที่ต่างๆ เพื่อรับสัญญาณดาวเทียมในย่าน L-Band และจะส่งสัญญาณเดือนไปยังสถานีควบคุมหลักภายใน 60 วินาที หากตรวจพบความผิดพลาด

2.2.2.3 สายอากาศภาคพื้นดิน (Ground Uplink Antenna Facility) ส่วนสถานีสายอากาศภาคพื้นดินจะทำการส่งคำสั่ง ข้อมูลการนำร่องและข้อมูลอื่นๆที่เรียกว่า ทีทีแอนด์ซี (Telemetry, Tracking and Command : TT&C) ซึ่งเตรียมพร้อมโดยสถานี ควบคุมหลักสำหรับดาวเทียมแต่ละดวง ข้อมูลเหล่านี้จะถูกส่งไปยังสถานีสายอากาศภาคพื้นดินและเก็บไว้จนกว่าดาวเทียมดวงที่ต้องการส่งข้อมูล จะผ่านมา โดยจะส่งคลื่นความถี่ S-Band โดยสถานีสายอากาศจะตั้งอยู่คู่กับสถานีสังเกตการณ์



Global Positioning System (GPS) Master Control and Monitor Station Network

รูปที่ 2.6 แสดงที่ตั้งของสถานีควบคุมดาวเทียมจีพีเอส

2.2.3 ส่วนผู้ใช้ (User Segment) ในส่วนนี้นั้นส่วนประกอบที่สำคัญคือ เครื่องรับสัญญาณจีพีเอส โดยจะรับสัญญาณ L-Band ที่ถูกส่งมาจากดาวเทียมและนำมาคำนวณเพื่อหาตำแหน่ง ความเร็วและเวลาของเครื่องรับจากนั้นจะนำค่ามาประยุกต์ใช้งานตามลักษณะการใช้งาน ซึ่งเครื่องรับจีพีเอสสามารถแบ่งออกได้เป็น 2 กลุ่ม คือ เครื่องประเภทที่สามารถรับดาวเทียมได้ 4 ดวง หรือมากกว่าได้พร้อมกันทีเดียว กับเครื่องที่มีมีการรับดาวเทียมโดยการเรียงลำดับ และแต่ละกลุ่มยังแบ่งย่อยได้อีกคือ

2.2.3.1 เครื่องรับแบบเรียงลำดับสัญญาณดาวเทียม ปกติเครื่องรับจีพีเอสจะต้องมีข้อมูลจากดาวเทียมอย่างน้อย 4 ดวง จึงสามารถคำนวณหาตำแหน่งที่ได้ เครื่องรับจีพีเอสที่ใช้เรียงลำดับใช้ช่องรับสัญญาณเพียงช่องเดียว รับข้อมูลจากดาวเทียมดวงหนึ่งระยะหนึ่งแล้วเปลี่ยนไปยังอีกดวงหนึ่ง เครื่องประเภทนี้จะมีแผงวงจรเล็ก ดังนั้นจึงมีราคาถูกและใช้กำลังน้อย ข้อเสียของการเรียงลำดับสัญญาณอาจเกิดการขาดตอนและทำให้มีผลต่อความถูกต้องของผลที่ได้น้อย เครื่องรับในกลุ่มนี้ได้แก่

- เครื่องรับช่องเดียวแบบจำกัดพลังงาน (Starved – Power Singler Receivers) เครื่องแบบนี้ ออกแบบให้พกพาได้และสามารถทำงานได้ด้วยอำนาจไฟฉายขนาดเล็ก การจำกัดการใช้กระแสไฟโดยให้ปิดการทำงานตัวเองโดยอัตโนมัติเมื่อแสดงตำแหน่งครั้งสองครั้งใน 1 นาที เหมาะสำหรับใช้งานบอกตำแหน่งส่วนตัว เช่น นักไต่เขาหรือเล่นเรือในเวลากลางวัน โดยไม่ต้องมีแบตเตอรี่ไฟฉายหลายก้อน และทำงานได้ทุกที่บนโลก ข้อเสีย คือ ความถูกต้องของสัญญาณจีพีเอสที่ได้ไม่ดีเท่าที่ควร ต่อเชื่อมกับ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

อุปกรณ์อื่นไม่ได้ และไม่สามารถใช้วัดความเร็วได้ การที่หาความเร็วไม่ได้ เนื่องจากต้องปิดเครื่องเองในระหว่างการวัด เพราะเครื่องใช้แผงวงจรนาฬิกาที่ใช้ไฟน้อย (นาฬิกาจะต้องเดินอยู่ตลอดเวลา) นาฬิกาที่ใช้จึงจะไม่ให้ความถูกต้องเท่าที่ควร

- เครื่องรับช่องเดียว (Single Channel Receivers) เหมือนกับแบบเครื่องรับช่องเดียวแบบจำกัดพลังงาน เป็นเครื่องรับสัญญาณช่องเดียวใช้ทำการหาระยะจากดาวเทียมทุกดวง แต่ที่ไม่เหมือนกันคือเครื่องรับช่องเดียวแบบมาตรฐานไม่จำกัดที่กำลังไฟ ดังนั้นจึงทำการรับสัญญาณต่อเนื่องได้ มีผลทำให้ความถูกต้องสูงกว่า และใช้วัดหาความเร็วได้ จากที่มีเพียงช่องเดียวที่ต้องใช้ทั้งการรับข้อมูลดาวเทียมและคำนวณหาระยะ จึงไม่สามารถหาตำแหน่งต่อเนื่องได้ ยิ่งกว่านั้นตามเหตุผลของวิชาการ ความไม่เที่ยงตรงของนาฬิกามีผลโดยตรงต่อความถูกต้องของการ วัดหาความเร็ว เครื่องราคาถูกบางชนิดใช้นาฬิการาคาถูกเพื่อให้ราคาเครื่องลดลงจึงทำให้ค่าความ

- เครื่องรับช่องเดียวแบบมัลติเพล็กซ์เร็ว (Fast-Multiplexing Single Receiver) เครื่องประเภทนี้เหมือนกับเครื่องทั้งสองประเภทข้างต้นที่กล่าวมาข้างต้นซึ่งรับซ้ำ แต่เครื่องรับนี้สามารถเปลี่ยนดาวเทียมได้เร็วกว่า ข้อดี คือ สามารถทำการวัดได้ในขณะที่กำลังรับข้อมูลจากดาวเทียม ดังนั้นเครื่องทำงานได้อย่างต่อเนื่อง และการที่มีนาฬิกาไม่เที่ยงตรงจึงมีผลต่อเครื่องประเภทนี้ด้วย เครื่องแบบนี้ต้องการใช้แผงวงจรที่ค่อนข้างซับซ้อนและราคาพอๆ กับเครื่องแบบสองช่องรับสัญญาณที่ใช้เครื่องซึ่งให้ความถูกต้องสูงกว่า และมีลักษณะการยืดหยุ่นการใช้งานได้ดีกว่า

- เครื่องรับสัญญาณสองช่องตามลำดับ (Two-Channel Sequencing Receivers) การเพิ่มช่องสัญญาณขึ้นอีกหนึ่งช่องช่วยให้เครื่องเพิ่มขีดความสามารถขึ้นอย่างเห็นได้ชัด ข้อหนึ่งความแรงสัญญาณ สัญญาณต่อสัญญาณรบกวน (Signal - to - Noise) เป็นสองเท่าทันที หมายถึงสามารถจับสัญญาณภายใต้สภาวะที่ไม่ดีและสามารถรับดาวเทียมดวงที่อยู่ค่าใกล้เคียงเส้นของฟ้าได้ จากการที่ช่องหนึ่งสามารถรับข้อมูลตำแหน่งอย่างต่อเนื่องได้ในขณะที่อีกช่องหนึ่งค้นหาดาวเทียมดวงต่อไป เครื่องแบบสองช่องนี้จะทำงานแบบนำร่องได้โดยไม่ต้องมีการขาดตอน และความเร็วก็จะมีค่าที่ถูกต้องขึ้น ความจริงเครื่องรับสองช่องที่คุณภาพดีก็สามารถใช้คำนวณหาและตัดค่าที่เวลาของนาฬิกาเครื่องรับไม่ดีทิ้งเพื่อใช้ในการวัดหาความเร็ว ข้อเสีย ของเครื่องแบบสองช่องคือ มีราคาสูง เครื่องแบบสองช่องยังมีราคาแพงกว่าเครื่องแบบช่องเดียวมาก ทั้งนี้เนื่องจากผู้ใช้เครื่องรับจีพีเอสสองช่องมักต้องการความถูกต้อง ต้องการเครื่องที่แข็งแรง และสามารถควบคุมสังเกตการณ์แสดงผลที่ดีกว่า

2.2.3.2 เครื่องรับแบบต่อเนื่อง (Continuous Receivers) ได้แก่เครื่องรับที่สามารถรับสัญญาณดาวเทียมพร้อมกันได้ ตั้งแต่ 4 ดวงขึ้นไป และสามารถแสดงผลตำแหน่งและความเร็วได้ทันที การรับสัญญาณดาวเทียมทั้ง 4 ดวง พร้อมกับที่มีค่าในการวัดหาในขณะที่มีการเปลี่ยนตำแหน่งรวดเร็วหรือต้องการความถูกต้องสูง ดังนั้นเครื่องแบบนี้จึงนำมาใช้ในงานวัดและทางด้านวิทยุ ซึ่งจะพบว่าจะมีช่องรับสัญญาณทั้ง 4, 5, 8, 10 และ 12 ช่อง นอกจากข้อดีที่ใช้วัดตำแหน่งอย่างต่อเนื่องได้แล้ว เครื่องรับจีพีเอสแบบหลายช่องสามารถช่วยขจัดปัญหา จีดี โอพี (Geometrix Dilution of Precision : GDOP) ได้อีกด้วย คือ แทนที่จะรับดาวเทียม 4 ดวงได้ ก็จะคำนวณหาจีดีโอพีที่ดาวเทียม 4 ดวงของกลุ่มดาวเทียมที่ขึ้นอยู่ และทำการวัดจากดาวกลุ่มที่มีจีดีโอพีต่ำสุด เครื่องรับ 4 ช่องสัญญาณ สามารถให้ค่าสัญญาณต่อสัญญาณ

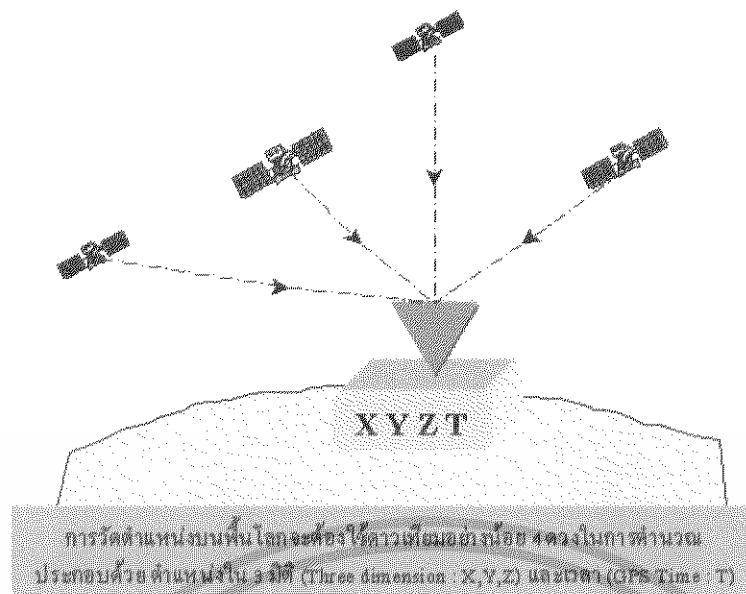
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รบกวนเพิ่มขึ้นเป็นสองเท่าของเครื่อง 2 ช่อง และเป็นสี่เท่าของเครื่องรับช่องเดียว และโดยการเปรียบเทียบค่าการรับสัญญาณของแต่ละช่อง เครื่องสามารถปรับตั้งค่าพิกัดเทียบระหว่างช่องรับสัญญาณ ซึ่งช่วยให้การวัดมีความถูกต้องดีขึ้นนอกเหนือจากข้อดี ข้อเสียที่กล่าวมาแล้ว ยังมีข้อดีอื่นที่น่าสนใจพิจารณาอีกคือ มีเครื่องแบบใหม่ที่ได้ ค่าความถูกต้องสูงมาก โดยการใช้ทั้งรหัส ซูโดเรนดอมที่กล่าวมาแล้ว และใช้ความถี่ของคลื่นพาหะ ซึ่งทำให้เครื่องรับที่ทำงานมีความเที่ยงสูง ที่รหัส ซูโดเรนดอมไม่สามารถให้ได้ และใช้ในการวัดหาวลาได้แม่นยำมากขึ้น ซึ่งช่วยในการบอกตำแหน่งได้ดีขึ้นด้วย และมีบางเครื่องที่ไม่ต้องใส่ค่าประมาณตำแหน่งและเวลาโดยประมาณให้เครื่องก่อนทำการวัด เครื่องรับแบบนี้ใช้ตัวเองใส่ค่าเริ่มตำแหน่งโดยตัวมันเอง ข้อที่ควรพิจารณาคือการต่อเชื่อมกับอุปกรณ์อื่นและความสะดวก บางเครื่องแสดงได้เฉพาะพิกัดภูมิศาสตร์ บางเครื่องไม่สามารถต่อเข้ากับเครื่องมืออื่นหรือคอมพิวเตอร์ได้ และมีข้อสำคัญที่ต้องพิจารณาอีกข้อคือความแข็งแรงทนทานถ้าต้องใช้เครื่องทำงานในพื้นที่ทะเลหรือในพื้นที่ป่าเขา การใช้ไฟและความร้อนที่เกิดขึ้นเป็นตัวชี้สำคัญที่จะต้องเอาใจใส่ ทางสถิติแสดงให้เห็นว่าอัตราของค่าความผิดพลาด จะเพิ่มเป็นสองเท่าของความร้อนในเครื่องเพิ่มทุก 7 องศาฟาเรนไฮด์ เครื่องรับจีพีเอสรุ่นใหม่ปัจจุบัน ได้เพิ่มคุณค่าให้แก่เครื่องรับจีพีเอส อีกหลายประการ เช่น ใช้ในการประมวลผลที่ซับซ้อน แสดงผลด้วยจอภาพรายละเอียด เครื่องรับจีพีเอสอาจแสดงจุดตำแหน่งบนแผนที่ที่วาดไว้แล้วให้เห็นทันที

การหาตำแหน่งบนพื้นโลก

ก่อนจะไปถึงการหาตำแหน่งบนพื้นดินต้องทำความเข้าใจเล็กน้อยเกี่ยวกับภารกิจตำแหน่งของ GPS .. GPS ใช้การอ้างอิงตาม World Geodetic System-84 (WGS-84) ซึ่งจำลองโลกว่าเป็นทรงรี (Ellipsoid) .. ทรงรีที่ว่านี้มีรัศมีตามแกน เส้นศูนย์สูตร (Semi-major axis) = 6378137 เมตร และรัศมีตามแนวขั้วโลก (Semi-minor axis) = 6356752.3124 เมตร ระยะห่างของดาวเทียมที่จริงก็จะเป็นระยะห่างจากศูนย์กลางโลกตาม WGS-84 การจำลองแบบนี้ทำให้สามารถคำนวณตำแหน่งได้ตามหลักของเรขาคณิต ตำแหน่งที่ receiver กำหนดได้ก็จะเป็นจุดที่ระบุอยู่ในระบบ coordinate สามมิติ (X, Y, Z) พอเอามาเทียบกับ ellipsoid ของ WGS-84 ก็จะ ได้เป็นตำแหน่งและความสูงเทียบจาก ellipsoid ค่าเหล่านี้จะเอามาแปลงเป็นพิกัดและความสูงจริงๆ อีกที พูดถึงความสูงนี่เป็นเรื่องค่อนข้างยุ่งยากพอสมควร เพราะว่าโลกเราไม่ได้ราบเรียบ เหมือนกับ ellipsoid ที่เป็น model ..พื้นโลกมีทั้งภูเขา ทะเล หลุมลึก ซึ่งมีช่วงความต่างกว่า 20 กิโลเมตรดังนั้นเราจึงต้องหาจุดอ้างอิงสำหรับความสูงขึ้นมา.. จุดที่วันนั้นก็คือ "ระดับน้ำทะเลกลาง" (จุดอ้างอิงของพิกัดก็คือ prime meridian ที่ผ่านเมือง Greenwich กับเส้นศูนย์สูตร) ระดับน้ำทะเลกลางเป็นตำแหน่งของพื้นผิวซึ่งคิดตามแรงโน้มถ่วงของโลกเรียกว่า "จีโออยด์" (Geoid) ใน WGS-84 มีแผนที่จีโออยด์ของตำแหน่งต่างๆ ทั่วโลกระบุไว้ด้วยว่าต่างจาก ellipsoid ของ WGS-84 มากน้อยแค่ไหน ดังนั้นเราจึงสามารถหาตำแหน่งความสูงของเราเทียบกับระดับน้ำทะเลกลางได้

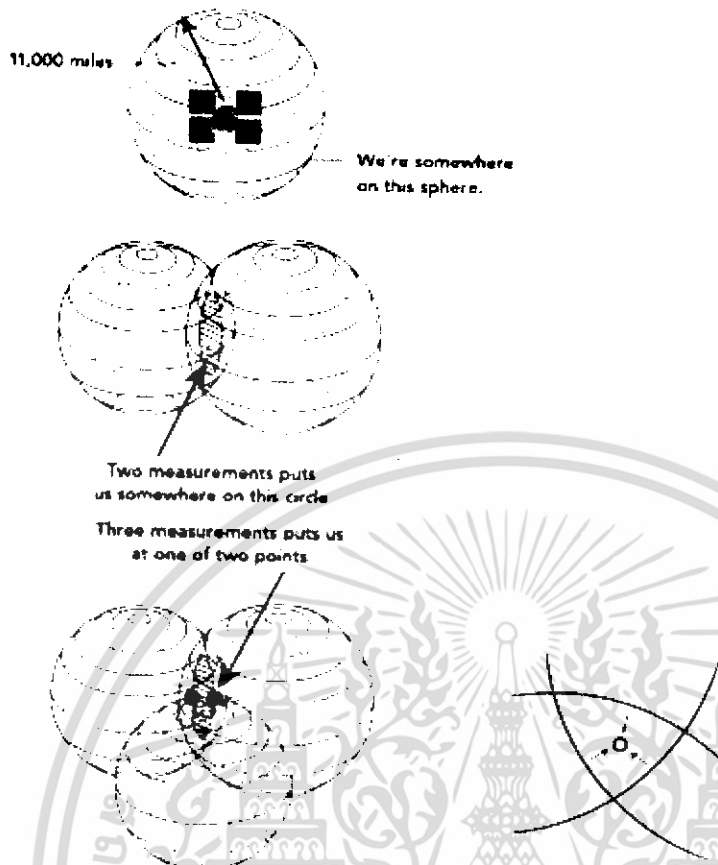
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.7 แสดงการวัดตำแหน่งบนพื้นโลก

สิ่งที่ต้องการในการหาตำแหน่งคือระยะทางระหว่างดาวเทียมกับ receiver แต่การวัดระยะ โดยตรงทำได้ยากมากและไม่ได้ค่าที่ละเอียดพอ ในบรรดาการวัดทั้งหมด "เวลา" เป็นสิ่งที่เราวัดได้แม่นยำที่สุดโดยใช้การสั้นของอะตอมซีเซียม ที่เราเรียกว่านาฬิกาอะตอม ถ้า receiver สามารถวัดเวลาได้แม่นยำพอเราก็จะสามารถวัดระยะเวลาที่คลื่นเดินทางมาจากดาวเทียมมาถึง receiver ได้พอได้ระยะเวลาที่คลื่นเดินทางก็จะสามารถหาระยะทางระหว่างดาวเทียมกับ receiver ได้โดยคำนวณกับความเร็วของแสง การวัดระยะเวลาของ receiver มีจุดสำคัญที่ pseudo random code ที่ receiver และดาวเทียม GPS จะ generate pseudo random code ที่เหมือนกันในขณะเวลาเดียวกัน เมื่อ receiver ได้รับ pseudo random code จากดาวเทียม GPS มันก็จะทำการเปรียบเทียบว่า code ที่ได้รับมีความแตกต่างจาก code ที่ receiver generate ในแกนเวลามากน้อยแค่ไหน ค่าความแตกต่างที่ได้ก็จะเป็นเวลาที่ใช้ในการเคลื่อนที่ของสัญญาณ จากดาวเทียมมาถึง receiver เมื่อรู้ระยะเวลาที่แน่นอนก็จะสามารถหาระยะห่างจาก receiver ถึงดาวเทียมได้ เหตุผลว่าทำไมต้องเป็น 3 ดวงคือ เมื่อเราได้ระยะห่างจากดาวเทียมดวงแรก เราจะได้ว่าตำแหน่งของ receiver จะอยู่บนพื้นผิวของทรงกลมเสมือนอันหนึ่งที่มีรัศมีเท่ากับระยะห่างระหว่างดาวเทียมกับ receiver ทีนี้เมื่อได้ระยะห่างจากดาวเทียมดวงที่สอง เราก็จะได้ว่าตำแหน่งของ receiver จะอยู่บนทรงกลมเสมือนอีกอันหนึ่งด้วย ดังนั้นจะได้ว่าตำแหน่งของ receiver จะอยู่บนเส้นรอบวงกลมที่เกิดจากจุดตัดของทรงกลมเสมือนทั้งสองอัน เมื่อได้ระยะห่างจากดาวเทียมดวงที่สาม เราก็จะได้ทรงกลมเสมือนสามลูกตัดกันซึ่งจะเกิดจุดตัด 2 จุด receiver จะอยู่ที่จุดตัดจุดใดจุดหนึ่งในสองจุดนี้ โดยทั่วไปตำแหน่งของจุดตัดนี้จะมีอันหนึ่งที่ไม่น่าจะเป็นคำตอบที่ถูกต้อง (เช่น คำนวณแล้วได้จุดที่อยู่นอกโลก ในตัวโลก ไม่ใช่บนผิวโลก) ซึ่งจะถูกตัดทิ้งไป และจุดที่เหลือก็คือผลลัพธ์ที่ถูกต้องว่าเราอยู่ที่ตำแหน่งใดบนพื้นโลก วิธีการที่ใช้หาค่าตำแหน่งนี้เรียกว่า "Triangulating"

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.8 แสดงลักษณะการตัดกันของระนาบวงกลมเทียมอันเนื่องมาจากเครื่องรับ

ความแม่นยำของระบบ

ความถูกต้องของตำแหน่งจะอยู่ที่ความถูกต้องในการวัดเวลา ดังนั้นการวัดตรงนี้ต้องทำละเอียดมาก ถ้าลองเทียบจาก pseudo random code ที่ถูก generate ด้วย bit rate 1.023 Mbps คาบของ cycle ก็จะมีประมาณ 1 usec (1/1000000 วินาที) ถึงเวลาจะน้อยนิดเดียว แต่ถ้าวัดผิดเพียง 1 cycle นี้ก็เพิ่มไป 300 เมตร เครื่อง receiver ใดๆ จะสามารถเปรียบเทียบ code ได้ละเอียดระดับ 1-2% ของ cycle ทำให้ผิดพลาดไป 3-6 เมตร แต่ก็ยังไม่ดีพอสำหรับงานบางงาน แต่อย่าลืมว่าสัญญาณจาก GPS ไม่ได้มีเพียง pseudo random code เท่านั้นยังมีสัญญาณ carrier L1/L2 ซึ่งมีความยาวคลื่นประมาณ 20 cm. หากใช้ carrier เข้ามาช่วยในการวัดเวลา ก็จะได้ผลที่แม่นยำขึ้นอีกมาก ตามทฤษฎีจะมี error เพียง 3-4 mm. เท่านั้น อย่างไรก็ตาม การวัดเวลาที่ถูกต้องหมายความว่า receiver จะต้องมيناฬิกาที่เดินตรงกับนาฬิกาบนดาวเทียม GPS และนี่เองที่เป็นปัญหาใหญ่ใน ideal case การใช้อวิธี triangulating จะให้ผลลัพธ์ที่ถูกต้องเพียงจุดเดียว (จากสองจุดที่หามาได้) แล้วถ้าเวลาของ receiver เดินตรงกันกับนาฬิกาของดาวเทียมจริงๆ ผลของการหาจุดตัดจากดาวเทียม GPS มากกว่า 3 ดวงจะต้องได้เป็นจุดเดียวเสมอ ดังนั้นหากการวัดระยะจากดาวเทียมดวงที่สี่ส่งผลให้ไม่ตัดกันจุดตัดดังกล่าว ก็แสดงว่านาฬิกาของ receiver ไม่ตรงกับนาฬิกาของดาวเทียมเราก็จะสามารถ synchronize เวลาของ receiver จะใหม่ให้ตรงได้โดยที่เราปรับนาฬิกาของ receiver จนกระทั่งได้จุดตัดเป็น

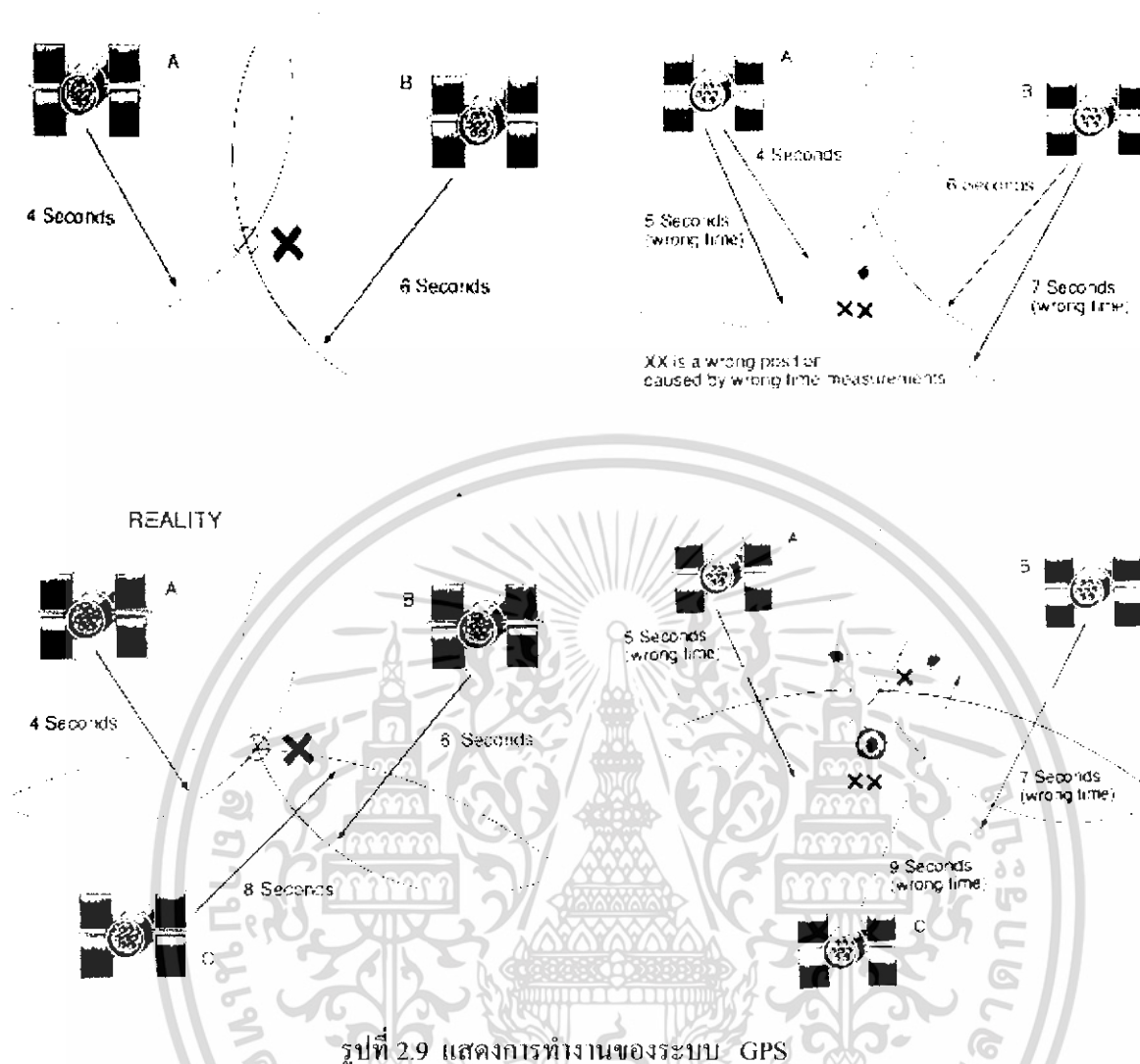
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จุดเดียว ในทางปฏิบัติ สัญญาณจากดาวเทียม GPS มีข้อผิดพลาดไม่น้อยและเป็นตัวทำให้ผลลัพธ์ของตำแหน่งผิดไปด้วย ข้างล่างนี้เป็นตัวอย่างของ error ที่เกิดขึ้น

- สัญญาณจากดาวเทียมจะไม่ได้เคลื่อนที่ด้วยความเร็วแสงจริงๆ แต่จะช้ากว่าแสงเล็กน้อย เนื่องจากต้องเคลื่อนที่ผ่านชั้นบรรยากาศของโลก
- บรรยากาศชั้น ionosphere จะทำให้สัญญาณโคจรบกวนจาก ion ที่มีอยู่หนาแน่น
- บรรยากาศชั้น troposphere ก็ทำให้สัญญาณ โคจรบกวนจากความกดดันและอุณหภูมิที่เปลี่ยนแปลง
- สัญญาณที่ได้รับอาจจะสะท้อนกับสิ่งก่อสร้างทำให้เกิด multipath error ซึ่งเกิดในลักษณะเดียวกับการเกิดภาพซ้อนเป็นเงาใน TV ทำให้ได้ค่าที่ไม่แน่นอน
- นาฬิกาอะตอมของดาวเทียม GPS เองก็มีโอกาสเดินไม่ตรงได้เหมือนกัน (nobody perfect) แม้ว่านาฬิกาอะตอมจะมีการปรับตั้งโดยสถานีควบคุมภาคพื้นดินแต่ บางครั้งอาจจะผิดไปเป็นนาทีได้เหมือนกัน เพราะสถานีควบคุมไม่สามารถตรวจสอบดาวเทียมได้ตลอดเวลา
- ตำแหน่งของดาวเทียมที่ receiver เลือกจะ lock สัญญาณก็มีความสำคัญด้วย เพราะถ้าหากดาวเทียมอยู่ท่ามุมที่ไม่เหมาะสม ก็จะทำให้การตัดกันของทรงกลมให้ผลที่ผิดพลาดมากกว่า

error เหล่านี้พอจะลดลงได้โดยใช้หลักของสถิติในการคำนวณ ปกติ GPS จะไม่ได้วัดค่าเพียงครั้งเดียวเพื่อหาคำตอบ แต่จะวัดค่าซ้ำหลายสิบ หรือหลายร้อยครั้ง แล้วก็หาค่าทางสถิติเพื่อให้ได้ตำแหน่งที่ใกล้เคียงตำแหน่งจริงที่สุด error ที่เกิดจากความตั้งใจเกิดจาก กระทรวงกลาโหมสหรัฐอเมริกา มีนโยบายที่ยังกับให้ GPS มี error เรียกว่า "Selective Availability" (SA) SA error นี้ทำขึ้นเพื่อลดความถูกต้องของภาระบุตำแหน่งของ GPS เพื่อไม่ให้ฝ่ายตรงข้ามสามารถใช้ประโยชน์จาก GPS ในทางทหารได้ วิธีการก็คือดาวเทียม GPS ทุกดวงจะสร้างสัญญาณรบกวนเข้าไปทำให้ความถูกต้องของสัญญาณที่ส่งออกมาตามปกติลดลง ผลก็คือการคำนวณตำแหน่งจะมีความผิดพลาดไปประมาณ 30 เมตรต่อดาวเทียม 1 ดวง นอกจากนี้ดาวเทียมยังอาจจะถูกสั่งให้เคลื่อนที่ออกนอกวงโคจรเพื่อเพิ่มข้อผิดพลาดให้เกิดขึ้นด้วย เพราะมี SA นี้เอง กระทรวงกลาโหมสหรัฐจึงกำหนดให้มีบริการสองแบบครับ คือ Standard Positioning Service (SPS) ซึ่งเป็นบริการฟรีที่ใครๆ ก็ใช้งานได้ สัญญาณที่ส่งมาจากดาวเทียมจะถูกรบกวนโดย SA error receiver ที่มีขายตามท้องตลาดก็จะใช้บริการจาก SPS ตาม requirement ของกระทรวงกลาโหมสหรัฐ กำหนดให้ SPS มีความถูกต้องของพิกัดในแนวตั้ง แนวนอน และเวลา เป็น 100 m/156 m/340 ns (Confidential factor = 95%) ตามลำดับ อีกบริการหนึ่งคือ Precise Positioning Service (PPS) จะใช้ในทางทหาร โดยเฉพาะ อาจจะอนุญาตให้คนทั่วไปมีสิทธิใช้ แต่ก็จะถูกควบคุมเป็นพิเศษ receiver สำหรับ PPS จะมีกุญแจถอดรหัสเพื่อตัด SA errors ออกได้ ทำให้มีความถูกต้อง 22 m/27.7 m/200 ns ที่ confidential factor = 95% .. ส่วนการหาความสูงจะเป็นไปได้ก็ต่อเมื่อรับสัญญาณดาวเทียมได้อย่างน้อย 4 ดวง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.9 แสดงการทำงานของระบบ GPS

2.6 โครงสร้างของข้อมูลที่ส่งจากดาวเทียม GPS

ข้อมูลที่ส่งจากดาวเทียมจีพีเอสประกอบไปด้วยข้อมูลที่ถูกระเบียงเป็นเฟรมย่อยๆ เรียกว่า ซับเฟรม (Subframe) แต่ละซับเฟรมจะแทรกค่าเวลาที่ซับเฟรมนั้นๆ ถูกส่งมาจากดาวเทียมจีพีเอส ไว้ด้วยเพื่อใช้ในการคำนวณหาตำแหน่งที่ติดตั้งตำแหน่งข้อมูลแต่ละเฟรมมีขนาด 1,500 บิต ถูกแบ่งให้อยู่ในรูปเฟรมขนาด 300 บิต จำนวน 5 ซับเฟรม ข้อมูลหนึ่งเฟรมจะถูกส่งมาจากดาวเทียมทุกๆ 30 วินาที ซับเฟรมขนาด 6 วินาที (300 บิต) จะบรรจุไว้ด้วยข้อมูลเส้นทางโคจรและของข้อมูลนาฬิกาโดยข้อมูลในแต่ละเฟรมประกอบด้วยส่วนประกอบย่อยๆ ดังนี้

- ซับเฟรมที่ 1 เป็นข้อมูลในการแก้ไขเวลาดาวเทียมจีพีเอส
- ซับเฟรมที่ 2 และ 3 เป็นข้อมูลเส้นทางโคจรของดาวเทียมจีพีเอส
- ซับเฟรมที่ 4 และ 5 เป็นข้อมูลอื่นๆ ที่เกี่ยวข้องกับระบบ

ข้อมูลจากดาวเทียมซึ่งบรรจุไว้ด้วยข้อมูลในการนำร่อง (Navigation Message) ที่ครบสมบูรณ์จะประกอบด้วยเฟรมข้อมูลจำนวน 25 เฟรม (125 ซับเฟรม) โดยข้อมูลดังกล่าวจะถูกส่งจากดาวเทียมทุกๆ 12.5 นาทีเป็นอย่างน้อย โดยทั่วไปเครื่องรับสัญญาณจะได้รับข้อมูลของตำแหน่งล่าสุดของดาวเทียมทุกเอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ชั่วโมง เพื่อใช้ร่วมกับอัลกอริทึมในการคำนวณพิกัดตำแหน่งและข้อมูลการโคจรของดาวเทียมแต่ละดวง อย่างคร่าวๆ เมื่อเครื่องรับสัญญาณทราบตำแหน่งตำแหน่งคร่าวๆของดาวเทียมแต่ละดวง รวมทั้งปรับ ชดเชยความผิดพลาดของสัญญาณพาหะจากปรากฏการณ์ดอปเปลอร์ของควมถี่พาหะ (Carrier Doppler Frequency) ซึ่งเกิดจากการที่ความถี่พาหะมีการเบนค่าเนื่องจากการเคลื่อนที่ของดาวเทียมนอกจากนั้นชุด ข้อมูลจากดาวเทียมจีพีเอสโดยสมบูรณ์ยังประกอบด้วยข้อมูลแบบจำลองชั้นบรรยากาศไอโอโนสเฟียร์อยู่ ด้วย เพื่อให้เครื่องรับสัญญาณสามารถประเมินค่าในการหน่วงเฟสของสัญญาณ (Phase delay) ที่เกิดจาก สภาพชั้นบรรยากาศไอโอโนสเฟียร์ จากเฟรมของข้อมูลที่กล่าวมาทั้งหมด ส่วนที่เครื่องรับสัญญาณ จำเป็นต้องใช้คือข้อมูลใน 3 ชั้นเฟรมแรก หากสามารถที่ได้รับข้อมูลดังกล่าวจากดาวเทียมตั้งแต่สี่ดวงขึ้นไป ก็สามารถคำนวณพิกัดตำแหน่งปัจจุบันของเครื่องรับสัญญาณได้ ในทางทฤษฎีการรับข้อมูลจากดาว เทียมสี่ดวงจะกินเวลาอย่างน้อย 18 วินาที ก่อนที่จะสามารถนำข้อมูลทั้งหมดมาใช้คำนวณตำแหน่งเนื่องจากชั้น เฟรมจากดาวเทียมแต่ละดวงจะมาถึงเครื่องรับไม่พร้อมกัน อีกทั้งไม่สามารถทราบล่วงหน้าว่าชั้นเฟรมที่ 1 จะได้รับมาเมื่อไหร่ ดังนั้นเพื่อเป็นการประกันว่าจะได้รับข้อมูล 3 ชั้นเฟรมแรกจากดาวเทียมจีพีเอส 4 ดวง อย่างแน่นอน ระยะเวลาที่ใช้ในการรับข้อมูลเท่าที่จำเป็นจะอยู่ที่ 30 วินาที หรือสรุปอย่างง่าย ๆ ได้ว่าการ คำนวณพิกัดตำแหน่งของเครื่องรับสัญญาณจะใช้เวลาอย่างน้อยประมาณ 30 วินาทีนั่นเอง

2.7 การทำงานของเครื่องรับสัญญาณจีพีเอส (GPS Receiver Operation)

2.7.1) การเลือกดาวเทียม (Satellite Tracking Selection)

กระบวนการแทรคสัญญาณจะเริ่ม โดยเครื่องรับจะหาว่าดาวเทียมดวงไหนที่โหนที่โหนที่เป็นไปได้ใน การแทรคสัญญาณ โดยจะทำการค้นหารหัส ซีเอสของดาวเทียมที่อยู่ในวิสัยและลีดสัญญาณดาวเทียมดวง นั้น เมื่อดาวเทียมถูกแทรคเครื่องรับจะสามารถคลิมนอแลตข้อมูลนำร่อง ค่าอัลมันตและข้อมูลแสดง สถานะของดาวเทียมทั้งหมดซึ่งการเลือกดาวเทียมของเครื่องรับจะต้องใช้ข้อมูลจากดาวเทียมอย่างน้อย 4 ดวงในการคำนวณ เครื่องรับอาจรับข้อมูลทุกดวงในพิสัยมาทำการคำนวณเลยก็ได้ แต่เครื่องรับส่วนใหญ่ จะทำการเลือกดาวเทียมไม่มากเกินไป ในการคำนวณเนื่องจากปัญหาการยุ่งยากในการคำนวณและต้นทุน ดังนั้นในการเลือกดาวเทียมที่จะรับข้อมูลสถานะดาวเทียมที่ดีที่สุดที่ได้จากสัญญาณแทรคดาวเทียมดวง แรก หรืออาจเลือกจากดาวเทียมที่ได้สัญญาณก่อนเลยก็ได้ ขึ้นอยู่กับการออกแบบ

2.7.2) การรับสัญญาณดาวเทียม

กำลังของสัญญาณของดาวเทียมที่ส่งออกมาเมื่อมาถึงเครื่องรับ สัญญาณที่ได้จะมีกำลังอ่อน และ จะถูกรบกวนโดย สัญญาณรบกวน (Noise) เครื่องรับจึงจำเป็นต้องทำการจำลองสัญญาณที่ได้รับเข้ามา และนำมาเรียงให้ตรงกับสัญญาณดาวเทียม จากนั้นจึงทำการคอมเพรสกลับมาเป็นสัญญาณต้นแบบ เรียกวิธีการนี้ว่าเทคนิค โค้ดคอร์เรลชัน (Code Correlation) แต่ปัญหาที่เกิดขึ้นคือความล่าช้าของสัญญาณ ในรหัสของเครื่องรับคือเวลาที่ใช้ในการเดินทางของสัญญาณระหว่างดาวเทียมกับเครื่องรับ ทำให้ได้ระยะทางออกมา (ระยะทางจากความล่าช้านี้เรียกว่า พูโดเรนจ์ (Pseudo range) ซึ่งไม่ใช่ระยะทาง ที่แท้จริง เนื่องจากยังไม่ได้ลบค่า ไบอัสของสัญญาณนาฬิกา (Clock Bias) ของเครื่องรับโดยปกติออกไป) เครื่องรับรับ โดยปกติจะใช้เทคนิค เฟสล็อครูปเพื่อซิงค์โคร โนสสัญญาณที่เครื่องรับสร้างขึ้นภายในกับ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สัญญาณที่ได้รับจากดาวเทียมโค้ดแทรคกิ้ง จะใช้คลื่นพาห้แทรคกิ้งทั้งสองจะช่วยกันและกัน เพื่อที่จะได้รับและแทรคสัญญาณดาวเทียม

2.7.3) โค้ดแทรคกิ้ง (Code Tracking)

โค้ดแทรคกิ้ง (Code Tracking) จะใช้เพื่อกำหนดการวัดชุดโคเรนจ์ระหว่างดาวเทียมจีพีเอส กับ เครื่องรับ โดยมันจะสร้างสัญญาณเลียนแบบรหัสซีเอชของดาวเทียมเป้าหมาย ค่าประมาณค่าคอปเปลอร์จะถูกนำออกไปโดยวงจรมอดูเลชันก่อนถึงคอร์รีเลเตอร์ เพื่อที่จะจัดสัญญาณที่ได้รับให้ตรงกับสัญญาณเลียนแบบที่สร้างขึ้น สัญญาณเลียนแบบจะถูกนำมาเทียบกับสัญญาณที่ได้รับมาอย่างเป็นระบบ ตามปกติ ผลลัพธ์ของคอร์รีเลเตอร์จะถูกรวม 1 ถึง 10 มิลลิวินาทีมาสามารถจับการคอร์รีเลตได้เฟสของรหัสที่สร้างขึ้นจะถูกส่งผ่านไป 1 ชิฟ ถ้าไม่สามารถจับการคอร์รีเลตได้หลังจากรหัสทั้งหมดถูกค้นหา แล้วคอปเปลอร์จะถูกปรับและการประมวลผลจะกระทำซ้ำต่อไปเรื่อยๆจนกระทั่งทำการคอร์รีเลตสำเร็จ การชิงโครโมเซชันรหัสจะถูกเริ่มรักษาไว้โดยจะคอร์รีเลตสัญญาณที่ได้รับกับรหัสที่เกิดขึ้นเร็วกว่าปกติ (With Chip and Code) ระบบป้อนกลับง่ายๆนี้จะรักษารหัสให้ตรงไว้ที่ตำแหน่งที่ถูกตั้งในการที่จะนำคลื่นพาห้ซึ่งยังคงมอดูเลตอยู่กับข้อมูลการนำร่องกลับมา (Prompt Code) จะถูกหักออกไปจากสัญญาณที่เข้ามาความล่าช้าที่เครื่องรับต้องเพิ่มให้สัญญาณเลียนแบบเพื่อให้การชิงโครโมเซชันสำเร็จตามความเร็วของแสงคือ การวัดชุดโคเรนจ์ (Pseudo range measurement) เมื่อคลื่นพาห้ถูกสร้างขึ้นใหม่อีกครั้งหนึ่งความถี่ที่แม่นยำและล๊อคกับสัญญาณที่รับเข้ามาซึ่งจะทำให้การวัดชุดโคเรนจ์ที่แม่นยำมากกว่าแบนด์วิดธ์ของโค้ดแทรคกิ้ง โดยปกติมีค่า 0.1 Hz ซึ่งแสดงว่าการวัดอย่างอิสระ จะสามารถทำได้ในช่วงเวลาประมาณ 10 วินาที

2.8 การให้บริการจีพีเอส (GPS Service)

มีอยู่ 2 ระดับ คือ บริการบอกตำแหน่งแบบสมบูร์นหรือพีพีเอส (PPS:Precise Positioning Service)และบริการบอกตำแหน่งแบบมาตรฐานหรือ เอสพีเอส (SPS:Standard Position Service)

2.8.1 บริการบอกตำแหน่งแบบสมบูร์น (Precise Position Service:PPS)

พีพีเอสเป็นบริการที่ให้ตำแหน่ง ความเร็ว และเวลาที่ถูกต้องเฉพาะผู้ที่ได้รับ อนุญาตให้ใช้งานเท่านั้น พีพีเอสจะเน้นไปในทางการทหาร การได้รับอนุญาตให้ใช้พีพีเอสนั้น จะพิจารณาโดย U.S. Military User,NATO Military User และกองทัพอื่นๆ หรือพลเรือนเช่น Australian Defense Force,U.S.Defense Mapping Agency พีพีเอสจะให้ค่าความคลาดเคลื่อนประมาณ 16 เมตร (16 metres Spherical Error Probable – SEP(3-D,50%)Position Accuracy) และความผิดพลาดทางด้านเวลาไม่เกิน 100 นาโนเซค (100 ns (one sigma)) Universal Coordinate Time (UTC)Time Transfer Accuracy) แต่ผู้ใช้งานที่ไม่ได้รับอนุญาตจะให้ค่าประมาณ 37 เมตร(3-D,95%)และ 197 นาโนเซค (95%) ภายใต้เงื่อนไขการทำงานโดยปกติของระบบเครื่องรับแบบพีพีเอส สามารถให้ความถูกต้องของความเร็วมีค่าความคลาดเคลื่อนไม่เกิน 0.2 เมตรต่อวินาที (0.2 m/s 3-D Velocity Accuracy) ได้ แต่ก็จะขึ้นกับการออกแบบเครื่องรับด้วย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การแอคเซส พีพีเอส ถูกควบคุมโดยเทคนิคคริปโตกราฟฟิค 2 อย่างคือ เอสเอ (Sa: Selective Availability) และ เอเอส (A-S: Anti-Spoofing) เอเอสจะใช้ในการลดความถูกต้องของ จีพีเอส ทั้งการบอกตำแหน่ง ความเร็ว เวลา แก่ผู้ใช้ที่ไม่ได้รับอนุญาต เอสเอจะทำงานโดยใส่ค่าผิดพลาดซูโดเรนดอมให้กับสัญญาณดาวเทียมเอเอส จะมีคลื่นดาวเทียมทุกดวงเพื่อป้องกันไม่ให้ผู้ใช้ที่ไม่ได้รับอนุญาตนำสัญญาณไปใช้ชั่วคราวได้อย่างสมบูรณ์ ซึ่งเทคนิคนี้จะนำเอา เอนคริปท์ รหัสพีเป็นรหัสวาย ในรหัสซีเอโค้ดจะไม่มีการทำแบบนี้

เอนคริปท์ชั้นคีย์ และเทคนิคเฉพาะจะถูกจัดให้กับผู้ใช้ พีพีเอส ซึ่งจะทำให้สามารถขจัดผลของ เอสเอ และ เอเอส ออกไป และทำให้ได้รับความแม่นยำสูงสุดของ จีพีเอส พีพีเอสซึ่งไม่มีคริปโตกราฟฟิคที่ถูกต้องจะมีประสิทธิภาพเหมือนเครื่องรับเอสพีเอส และเครื่องรับพีพีเอส สามารถใช้ได้ทั้งรหัสพีวาย หรือ รหัสซีเอโค้ด หรือกับทั้งคู่ ความถูกต้องสูงสุดของ จีพีเอสจะได้รับจากการใช้รหัสพีวาย บนทั้งความถี่ แอสหนึ่ง และแอสสอง เครื่องรับซึ่งสามารถใช้รหัสพีวาย ได้นั้นมักจะใช้รหัสซีเอโค้ด เพื่อเริ่มต้นการรับสัญญาณดาวเทียมจีพีเอส

2.8.2 บริการบอกตำแหน่งแบบมาตรฐาน (Standard Positioning Service: SPS)

เอสพีเอส เป็นบริการที่ให้ความถูกต้องน้อยกว่า แต่ให้ใช้ได้กับผู้ใช้งานทั้งหมดในสภาวะปกติ ระดับ ของ เอสเอ จะถูกควบคุมเพื่อให้ค่าความถูกต้องในแนวเส้นขอบฟ้า 100 เมตร (100m (95% Horizontal Accuracy) มีค่าประมาณ 156 เมตร (3D-95%) เครื่องรับเอสพีเอสมีความสามารถด้านความถูกต้องของเวลาประมาณ 337 นาโนเซค (95% ของ UTC Time Transfer Accuracy) การลดความถูกต้องของระบบสามารถเพิ่มขึ้นได้ถ้ามีความจำเป็น เช่น ในยามมีศึกสงครามโดย ประธานาธิบดีสหรัฐอเมริกาที่มื่ออำนาจ สั่งการผ่านยูเอส เนชันนอล คอมมานด์ ออโตริตี้ (U.S. National Command Authority) เพื่อเปลี่ยนระดับของเอสเอ เป็นระดับอื่นนอกเหนือจากในยามสงบ บริการเอสพีเอส นั้นมีไว้ให้พลเรือนทั่วไปใช้งาน และให้กองทัพใช้ในยามสงบ

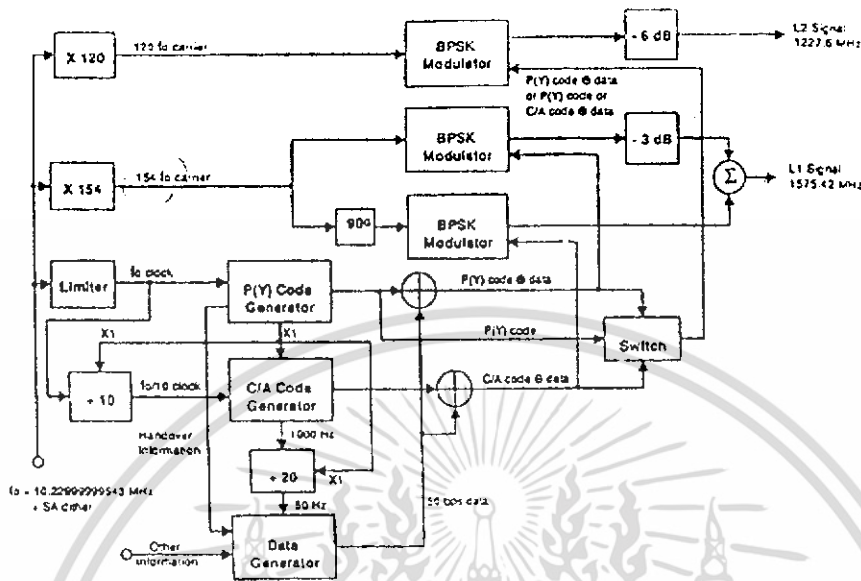
2.9 ทฤษฎีการปฏิบัติงานระบบ จีพีเอส (Global Theory of Operation)

สัญญาณวัดระยะทาง (Rangin code) ที่ส่งมาจากดาวเทียมนั้น จะทำให้เครื่องรับจีพีเอสสามารถวัดเวลาการเดินทางของสัญญาณและทราบระยะทางระหว่างดาวเทียมกับผู้ใช้ ข่าวสารการนำร่องจะให้ข้อมูล เพื่อการคำนวณหาตำแหน่งของดาวเทียมแต่ละดวงที่เวลาของการส่งสัญญาณจากข้อมูลเหล่านี้ โคออร์ดิเนตของตำแหน่งผู้ใช้และค่าเหลือมลิ่งของสัญญาณนาฬิกาของผู้ใช้ (User Clock Offset) จะถูกคำนวณโดยใช้สมการทางคณิตศาสตร์ (Simultaneous Equation) ซึ่งตามปกติจะต้องการดาวเทียมอย่างน้อย 4 ดวง เพื่อการบอกตำแหน่งในสามมิติ ในวิสัย (In view) ของเครื่องรับในเวลาเดียวกัน

2.9.1 ลักษณะของสัญญาณ จีพีเอส

ดาวเทียมจีพีเอส จะส่งสัญญาณมาบนความถี่พาห์ 2 ความถี่ (ย่านแอลแบนด์) เรียกว่า ความถี่แอลหนึ่ง และแอลสอง ความถี่พาห์จะถูกมอดูเลตด้วยวิธีสเปคตรัม กัรบรหัสข้อมูลซึ่งประกอบไปด้วยขบวนการรหัส PRN (Pseudo Random Noise) ที่เป็นเอกลักษณ์ ของดาวเทียมแต่ละดวง และข้อมูลข่าวสารการนำร่อง ดาวเทียมทุกดวงส่งความถี่พาห์ที่เหมือนกันแต่สัญญาณจะไม่รบกวนกันเนื่องเพราะว่าการมอดูเลชันของรหัสพีอาร์เอ็นที่เป็นเอกลักษณ์ของดาวเทียมแต่ละดวง และทุกๆ ขบวนการรหัสพีอาร์เอ็น ของเอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ดาวเทียมแต่ละดวงจะไม่มีความสัมพันธ์กันเลขที่เครื่องรับสัญญาณดาวเทียมสามารถถูกแยกและตีเทคได้ โดยใช้เทคนิคที่เรียกว่า ซีดีเอ็มเอ (CDMA: Code Division Multiple Access)



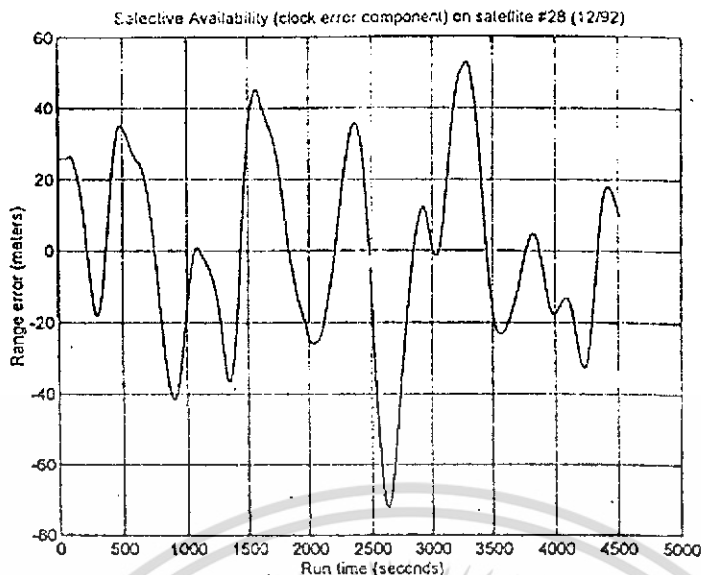
รูปที่ 2.10 โครงสร้างสัญญาณจากดาวเทียมจีพีเอส

2.9.2 Selective Availability (SA)

ต้นกำเนิดความผิดพลาดที่ใหญ่ที่สุดสำหรับผู้ใช้งานจีพีเอส คือ เอสเอ โดย เอสเอจะถูกตั้งใจให้เกิดขึ้นโดยกระทรวงกลาโหมของสหรัฐอเมริกาเพื่อลดความแม่นยำของผู้ใช้งาน เอสเอเริ่มใช้งานอย่างเป็นทางการในวันที่ 25 มีนาคม 2533 การลดความแม่นยำนั้นทำได้โดยการจัดการกับข้อมูลพีซีเอ็มเอสที่ส่งออกมา (ส่วนประกอบทางความคลาดเคลื่อนของวงโคจร) และการสั่นสะเทือนของนาฬิกาบนดาวเทียม (ส่วนประกอบทางความคลาดเคลื่อนของนาฬิกา) เราแทนส่วนประกอบทางความคลาดเคลื่อนของวงโคจรเป็น ϵ และแทนส่วนประกอบทางความคลาดเคลื่อนของนาฬิกาเป็น δ ผลของเอสเอ โดยการสั่นสะเทือนของนาฬิกาบนดาวเทียมจะเห็นได้ในการรบกวนที่แปรผันตามเวลาของซูโดเรนจ์ในรูปที่ 2.10 จะเห็นได้ว่าความคลาดเคลื่อนแสดวงแนวโน้มซึ่งประกอบด้วยการอสซिलเลตที่ไม่แน่นอน ที่มีคาบเวลา 4 ถึง 12 นาที กับการเปลี่ยนแปลงของความคลาดเคลื่อนของซูโดเรนจ์ภายในช่วง 70 เมตร

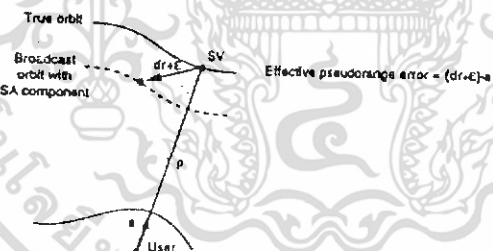
72041

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



เอสเอ (ส่วนประกอบทางความคลาดเคลื่อนของนาฬิกา)

ผลของเอสเอ ที่กระทำโดยการจัดการกับข้อมูลพีเมอริส สามารถทำให้เกิดความคลาดเคลื่อนในการหาตำแหน่งของดาวเทียมของผู้ใช้ ความคลาดเคลื่อนจะสามารถทำให้เกิดขึ้น ได้หลายแบบเนื่องจากข้อมูลพีเมอริส มีพารามิเตอร์อยู่ 15 ส่วนประกอบทางความคลาดเคลื่อนของวงโคจร e จะเป็นผลรวมทางเวกเตอร์ของความคลาดเคลื่อนของพีเมอริส ที่มีอยู่กับส่วนประกอบทางวงโคจร เอสเอที่โปรเจกต์ลงบนเวกเตอร์ที่เกิดจาก ไลน์ออฟไซน์ ของดาวเทียมมายังผู้ใช้งาน ดังรูปที่ 2.11



ผลกระทบของเอสเอ เมื่อรวมกับความคลาดเคลื่อนทางพีเมอริส

ในปัจจุบันนี้ได้พยายามมีการผลักดันให้ยกเลิกการลดค่าความผิดพลาดลงจากเดิม โดยให้เหตุผลว่าความผิดพลาดที่เกิดขึ้นสามารถใช้เทคนิคฟีเฟอเรนเชียลชจัดออกไปได้เกือบหมด นอกจากนั้นระบบกลอนาส ซึ่งเป็นระบบนำร่องของรัสเซียให้ความผิดพลาดเพียง 50 เมตรเท่านั้น

2.10 มาตรฐาน NMEA และโปรโตคอลที่ใช้ในการสื่อสารของ จีพีเอส

2.10.1 มาตรฐาน NMEA (NMEA Standard)

คือการอินเตอร์เฟซทางไฟฟ้าและ โปรโตคอลการสร้างข้อมูลสำหรับการสื่อสารระหว่างอุปกรณ์การเดินเรือ (NMEA ยังมีมาตรฐานสำหรับการสื่อสารในแบบอื่นๆด้วย) โดย NMEA ย่อมาจาก The National Electronic Association ซึ่งการศึกษาและพัฒนาอุตสาหกรรมอิเล็กทรอนิกส์ทางน้ำ เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.10.2 การอินเตอร์เฟซทางไฟฟ้า (Electrical Interface)

มาตรฐานนี้สามารถใช้เป็นระบบที่มีตัวส่ง (talker) เดียวและตัวรับ (listener) สายที่แนะนำให้ใช้เป็นแบบชิลด์ทวิสต์เพอร์ โดยต่อกราวด์ที่ตัวส่งเท่านั้นมาตรฐานไม่ได้กำหนดชนิดคอนเนคเตอร์ (connector) เฉพาะ

2.10.3 มาตรฐาน NMEA – 0183

มาตรฐานที่เอาท์พุทจะเป็นแบบ อีไอเอ – 4221 และมีสายสัญญาณ 2 เส้น เอ และ บี โวลเตจบนเส้นเอจะเป็นเหมือนกับสาย ทีทีแอล เดียวแบบเดิม ขณะที่ บี โวลเตจจะกลับทางกันกับ เอ เช่น เอเป็น +5 บีจะเป็นกราวด์ ในการใช้งาน, สายเพียงสายเดียวคือ สายเอ ใน อีไอเอ – 4221 อาจจะถูกใช้เชื่อมต่อกับ อาร์เอส – 232 อินพุทของเครื่องคอมพิวเตอร์ในมาตรฐาน เอ็นเอ็มอีเอ – 0183 , ตัวอักษรที่ใช้คือแอสกี ซึ่งสามารถพิมพ์ได้ เอ็นเอ็มอีเอ – 0183 นั้นข้อมูลจะถูกส่งด้วยอัตรา 4800 บอริต ข้อมูลจะถูกส่งในรูปของประโยค แต่ละประโยคเริ่มด้วย \$ ตัวอักษรที่ตามมาอีกสองตัวคือ เซ็นเทนไอดีหรือตัวกำหนดรูปแบบประโยคหรือจะเรียกว่าชื่อประโยค (sentence name) ตามด้วยฟิลด์ข้อมูลจำนวนหนึ่งซึ่งถูกแบ่งแยกโดยเครื่องหมายคอมม่า (,) และสิ้นสุดด้วยเครื่องหมายเช็คซัม (check sum) ที่สามารถเลือกได้ว่าจะมีหรือไม่ และจบลงด้วยแครีเรียเทอร์นและไลน์ฟีด (carrier return/line feed) ซึ่งประโยคจะมีตัวอักษรถึง 82 ตัวรวม \$ และแครีเรียเทอร์นและไลน์ฟีด

ถ้าข้อมูลสำหรับฟิลด์ไม่สามารถหาได้ ฟิลด์จะถูกเว้นข้ามไปแต่คอมม่าที่ทำหน้าที่แบ่งฟิลด์ยังคงถูกส่งไปโดยไม่เว้นช่องว่าง เพราะว่าในแต่ละฟิลด์มีความยาวไม่คงที่หรือไม่มีข้อมูลเครื่องรับจะระบุตำแหน่งของฟิลด์ของข้อมูลที่ต้องการ โดยการนับเครื่องหมายคอมม่าเช็คซัมที่เลือกได้ว่าจะมีหรือไม่ ประกอบด้วย “ * ” และ 2 บิตของเลขฐาน 16 แทนการ เอกซ์คูซีฟออร์ของตัวอักษรทั้งหมดแต่ไม่รวม “ \$ ” และ “ * ” ในการใช้งานจะมีความต้องการใช้ เช็คซัมในบางประโยค ในมาตรฐานจะอนุญาตแต่ละผู้ผลิตในการนิยามรูปแบบประโยค ประโยคเหล่านี้เริ่มต้นด้วย “ \$PG ” และตัวอักษรสามตัว ที่ตามมาเป็นไอดีที่ถูกกำหนดมาจากโรงงานตามด้วยข้อมูลซึ่งเป็นไปตามรูปแบบทั่วไปของประโยคมาตรฐาน

2.10.4 โปรโตคอล NMEA 183

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

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.10.5 รูปแบบประโยคของ NMEA

RMB

\$GPRMB,A,x.x,a,c--cc,d--d,lll.ll,c,yyyy.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 (Crosstrack 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) ในหน่วยน็อต
- 13 = สถานะเมื่อเข้ามาถึง (Arrival status) (A=เข้าถึงหรือผ่านไปอย่างตั้งฉาก)
- 14 = เช็คซัม

RMC

\$GPRMC,hhmmss.ss,A,lll.ll,a,yyyy.yy,a,x.x,x.x,ddmmyy,x.x,a*hh

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

- 1 = เวลาปัจจุบันในระบบ ยูทีซี
- 2 = สถานะของข้อมูล
- 3 = ละติจูด
- 4 = ทิศเหนือหรือทิศใต้
- 5 = ลองจิจูด
- 6 = ตะวันออกตะวันตก
- 7 = ความเร็วในหน่วยน็อต
- 8 = มุมที่วัดจากทิศเหนือ
- 9 = วันเดือนปี
- 10 = ความแปรปรวนเนื่องจากสนามแม่เหล็ก
- 11 = ตะวันออกหรือตะวันตก
- 12 = เช็คซัม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

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 = ตำแหน่งของยูทิลิตี้
- 2 = ละติจูด
- 3 = ทิศเหนือหรือใต้
- 4 = ลองจิจูด
- 5 = ตะวันออกหรือตะวันตก
- 6 = ซีดีคุณภาพของระบบจีพีเอส (0 = invalid;1 = จีพีเอส fix;2 = diff.จีพีเอส fix)
- 7 = จำนวนดาวเทียมที่ใช้
- 8 = เฮสติโอพี(Horizontal dilution of position)
- 9 = ความสูงของสายอากาศ เหนือกว่าหรือต่ำกว่า ระดับน้ำทะเล
- 10 = หน่วยความสูงของสายอากาศ (เมตร)
- 11 = ค่าความแตกต่างระหว่างระบบ คับเบิลยูจีเอส-84 กับระบบจีอออยด์
- 12 = ความต่างของจีอออยด์
- 13 = ระยะเวลาตั้งแต่อัปเดตข้อมูลครั้งสุดท้ายจากสถานีอ้างอิง
- 14 = หมายเลขประจำสถานีอ้างอิง
- 15 = เช็คซัม

VGT

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

VGT = มุมที่ทำกับทิศเหนือจริงๆและความเร็ว (Actual track made good and speed over ground)

- 1 = มุมที่ทำกับทิศเหนือ
- 2 = อักษร 'T' แสดงว่ามุมที่วัดเทียบกับทิศเหนือ
- 3 = ไม่ใช่
- 4 = ไม่ใช่
- 5 = ความเร็วหน่วยน็อต
- 6 = อักษร 'N' แสดงความเร็วในหน่วยน็อต
- 7 = ความเร็วหน่วย กิโลเมตร/ชั่วโมง
- 8 = อักษร 'T' แสดงความเร็วในหน่วย กิโลเมตร/ชั่วโมง
- 9 = เช็คซัม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

GSA

\$GPGSA,A,3,19,28,14,18,27,22,31,39,,,,,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

3-14 = หมายเลขของดาวเทียมที่ใช้ในการระบุตำแหน่ง (เป็น 0 สำหรับฟิลด์ที่ไม่ใช้)

15 = พีดีไอพี

16 = เฮสติไอพี

17 = วีดีไอพี

GSV

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

GSV = จำนวนของดาวเทียมที่มองเห็น, หมายเลข พรีเออร์เอ็น, เอเลเวชัน, อาซิมูท และค่าเอสเอ็นอาร์ (Number

of SVs in view, PRN number, elevation, azimuth & SNR value.)

1 = จำนวนหมายเลขของข่าวสารทั้งหมดของมูลนี้ใน 1 รอบ

2 = หมายเลขข่าวสาร

3 = จำนวนทั้งหมดของดาวเทียมที่อยู่ในพิสัย

4 = หมายเลข พีเอ็นอาร์ ของดาวเทียม

5 = มุมเอเลเวชันหรือมุมเงยมีค่าสูงสุด 90 องศา

6 = มุมอาซิมูททำกับขั้วเหนือมีค่า 0 - 359 องศา

7 = ค่าเอสเอ็นอาร์ มีค่า 0 - 99 เดซิเบล (เมื่อไม่มีการแทรกคิ่ง)

8-11 = ข่าวสารเกี่ยวกับดาวเทียมดวงที่สอง, เหมือนกับฟิลด์ 4-7

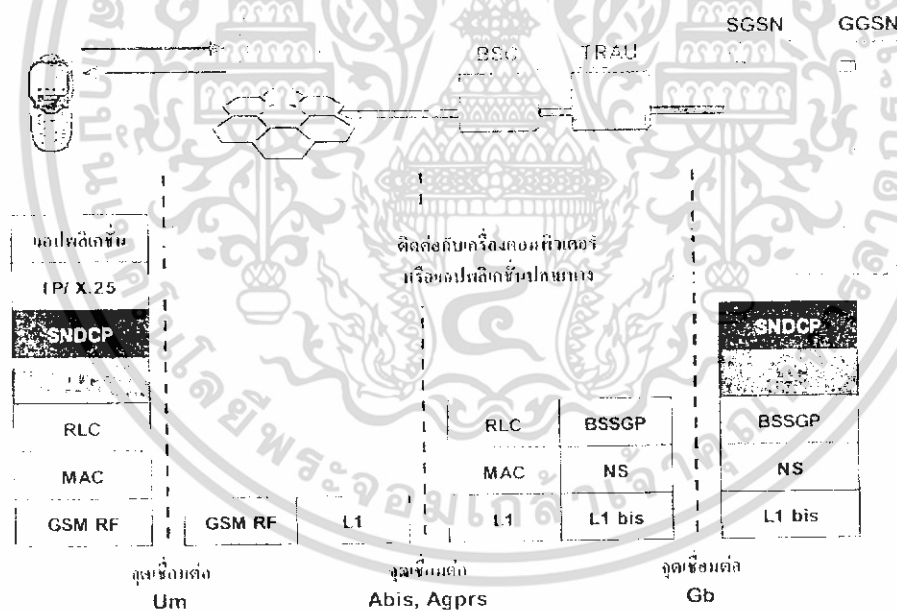
12-15 = ข่าวสารเกี่ยวกับดาวเทียมดวงที่สาม, เหมือนกับฟิลด์ 4-7

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.11 มาตรฐานการเชื่อมต่อทางคลื่นวิทยุของเครือข่ายGPRS

เนื่องจากGPRSถือเป็นมาตรฐานการสื่อสารข้อมูลประเภทหนึ่ง ดังนั้นจึงสามารถแสดงภาพการ จัดแบ่งหน้าที่การทำงานของวงจรทางสาร์คว์แวร์และซอฟต์แวร์ โดยอ้างอิงตามแบบจำลองOSI (Open System Integration) ดังรูปที่11.1 ซึ่งจะเห็นได้ว่า หน้าที่การทำงานของสถานีฐานหรืออุปกรณ์ BTS ในการรองรับเทคโนโลยี GPRS นั้นเป็นเพียงการส่งผ่านข้อมูลทั้งที่เป็นสัญญาณควบคุม (Control Signaling) และสัญญาณข่าวสารข้อมูล(Data Signaling) ระหว่างเครื่องลูกข่ายGPRS กับอุปกรณ์ BSC และ TRAU เท่านั้น โดยข้อมูลที่ถูกรับส่งผ่านจุดเชื่อมต่อ Um และจุดเชื่อมต่อ Abis หรือ Agprs ซึ่งล้วนแล้ว เกี่ยวข้องกับสถานีฐานจะถูกปรับเปลี่ยนรูปแบบของสัญญาณจากคลื่นความถี่วิทยุไปเป็นสัญญาณทาง ไฟฟ้าเพื่อส่งต่อไปให้กับอุปกรณ์BSCเท่านั้น สถานีฐานจะไม่ทำหน้าที่ใดๆในการอ่านหรือแก้ไขเนื้อหา ข้อมูลที่เป็นการรับส่งตามมาตรฐาน GPRS ซึ่งจะถูกห่อหุ้มโดยบล็อกข้อมูลประเภท MAC(Medium Access Control) และ RLC(Radio Link Control)

ซึ่งจะถูกนำไปประมวลผลในเครื่องลูกข่ายโทรศัพท์เคลื่อนที่และอุปกรณ์ BSC/TRAU เท่านั้น



รูปที่2.11 รายละเอียดของจุดเชื่อมต่อและโปรโตคอลที่เกี่ยวข้อง ระหว่างอุปกรณ์ภายในเครือข่าย สถานีฐานของมาตรฐานโทรศัพท์เคลื่อนที่GPRS

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

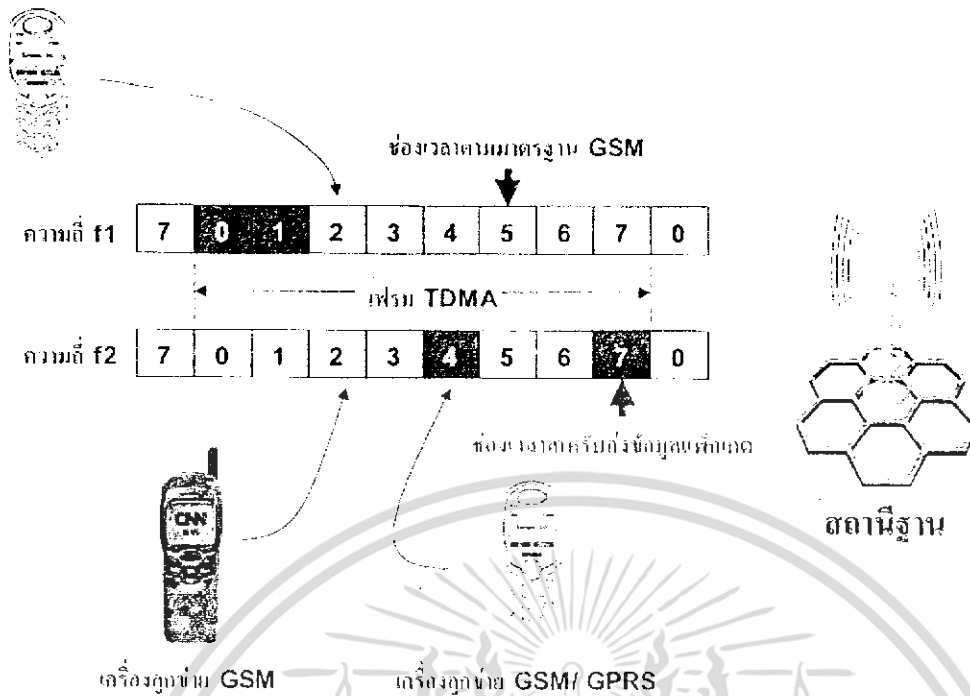
การสร้างและการจัดการช่องสื่อสารทางตรรก (Logical Channel) ซึ่งจะได้กล่าวถึงต่อไป เป็นการดำเนินการร่วมกันระหว่างอุปกรณ์BSC กับเครื่องลูกข่ายโทรศัพท์เคลื่อนที่GPRSโดยการกำหนดค่าบิตข้อมูลในระดับของโปรโตคอล RLC โดยมีการใช้โปรโตคอล MAC ในการควบคุมการส่งผ่านข้อมูลทุกๆ บิตที่เป็นการสื่อสารแบบ GPRS ให้มีความถูกต้อง ไม่เกิดความผิดพลาดของข้อมูล การจัดการกำหนดรายละเอียดของการใช้ช่วงเวลา (Timeslot) ในแต่ละความถี่ของสถานีฐานให้เป็นช่องสื่อสารสำหรับโทรศัพท์เคลื่อนที่ GSM หรือGPRS ก็เป็นอีกภาระหน้าที่หนึ่งของอุปกรณ์ เช่นเดียวกัน สำหรับอุปกรณ์ ซึ่งติดตั้ง ไว้ภายในจะทำหน้าที่ปรับรูปแบบข้อมูลเพื่อทำการติดต่อสื่อสารแบบแพ็คเกจ ซึ่งได้แก่อุปกรณ์ SGSN และ GGSN โดยผ่านโปรโตคอลชนิดใหม่ที่มีชื่อเรียกว่า ในกรณีของการเชื่อมต่อวงจรแบบสวิตซ์วงจรระหว่างอุปกรณ์ กับชุมสายโทรศัพท์เคลื่อนที่มาตรฐาน

อุปกรณ์ จะทำการติดต่อสื่อสารกับเครื่องลูกข่ายโทรศัพท์เคลื่อนที่ โดยตรวจสอบและควบคุมตำแหน่งที่อยู่ของเครื่องลูกข่าย และบริหารการเชื่อมต่อวงจรแบบสวิตซ์แพ็คเกจ เพื่อรับส่งข้อมูลไปยังโทรศัพท์เคลื่อนที่ เครื่องอื่น หรือไปยังเครือข่ายสื่อสารข้อมูลภายนอก เช่น เครือข่ายอินเทอร์เน็ต ทั้งนี้การติดต่อสื่อสารดังกล่าวจะอยู่ในรูปแบบของบล็อกข้อมูลที่ถูกควบคุมโดยโปรโตคอลที่ถูกส่งมาจากอุปกรณ์ รวมถึงการเข้ารหัสข้อมูลเพื่อป้องกันการดักจับโดยบุคคลที่สาม สำหรับการส่งสัญญาณควบคุมเพื่อดำเนินการต่างๆเกี่ยวกับการติดต่อสื่อสารระหว่างอุปกรณ์ กับเครื่องลูกข่าย นั้นจะเป็นหน้าที่ของโปรโตคอลที่มีชื่อว่า SNDCP (Subnet work-Dependent Convergence Protocol)

ข้อมูลที่ถูกถอดออกจากโปรโตคอล SDNCP แล้วจะเป็นข้อมูลที่อยู่ในบล็อกข้อมูลมาตรฐาน IP (Internet Protocol) หรือ X.25 ซึ่งเป็นมาตรฐานการสื่อสารที่สามารถใช้เชื่อมต่อกับเครือข่ายสื่อสารข้อมูลประเภทต่างๆได้ทั่วโลก เนื้อหาภายในบล็อกข้อมูลแบบ IP หรือ X.25 นั้นก็จะเข้าแอปพลิเคชันใช้งานต่างๆตามมาตรฐานการสื่อสารทั่วไป ไม่ว่าจะเป็น HTTP,FTP,Telnet ฯลฯ ข้อมูลเหล่านี้ซึ่งถือว่าอยู่ในระดับชั้นบนสุดของแบบจำลอง OSI ที่มีการรับส่งจากเครื่องลูกข่าย GPRS จะถูกเชื่อมต่อเพื่อการสื่อสารกับเครื่องลูกข่าย GPRS เครื่องอื่น หรือกับเครือข่ายภายนอก ตัวอย่างเช่น เครือข่ายอินเทอร์เน็ต เป็นต้น

เนื่องจาก GPRS เป็นเทคโนโลยีที่ได้รับการออกแบบขึ้นบนมาตรฐานโทรศัพท์เคลื่อนที่GSM ดังนั้นจึงไม่มีการเปลี่ยนแปลงรายละเอียดของมาตรฐานการเชื่อมต่อทางวิทยุในระดับชั้นกายภาพแต่ประการใด รูปที่ 11.2 แสดงโครงสร้างของการเชื่อมต่อทางคลื่นวิทยุระหว่างสถานีฐานและเครื่องลูกข่ายโทรศัพท์เคลื่อนที่ทั้งชนิด GSM และ GPRS โดยที่จะต้องมีการกำหนดความถี่ใช้งานอย่างน้อย 1 คู่สำหรับการติดต่อสื่อสารซึ่งความถี่ต่ำหนึ่งมีค่าความถี่ต่ำกว่าจะถูกใช้ในการส่งสัญญาณจากเครื่องลูกข่ายไปยังสถานีฐาน เรียกว่าความถี่ขาขึ้น หรือ Uplink Frequency และความถี่ที่มีค่าสูงกว่าจะใช้ส่งสัญญาณในทิศทางกลับกัน เรียกว่า ความถี่ขาลง หรือ Downlink Frequency มาตรฐานโทรศัพท์เคลื่อนที่ในตระกูล GSM ทั้งย่านความถี่ 900 MHz 1800 MHz และ 1900 MHz ต่างก็มีการกำหนดค่าความแตกต่างของความถี่ไว้ กล่าวคือ GSM 900 กำหนดให้ค่าความถี่ขาขึ้นและขาลงแต่ละคู่อยู่ห่างกัน 45 MHz

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

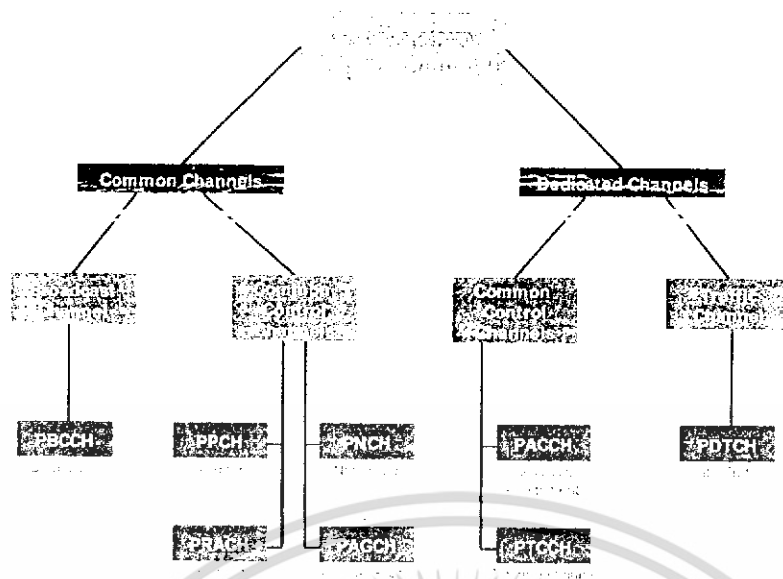


รูปที่ 2.12 การจัดสรรและบริหารจัดการช่วงเวลาตามมาตรฐานโทรศัพท์เคลื่อนที่ GSM ให้รองรับการสื่อสารด้วยมาตรฐาน GSM โดยเป็นไปในลักษณะการร่วมใช้ความถี่เดียวกัน

จากรูปที่ 2.12 สถานีฐานมีการใช้งานความถี่ 2 คู่ คือ F1 และ F2 สำหรับการรับส่งข้อมูลกับเครื่องลูกข่ายโทรศัพท์เคลื่อนที่ในพื้นที่แห่งหนึ่ง แต่ละความถี่จะมีการกำหนดโครงสร้างโดยแบ่งออกเป็นช่วงเวลาย่อยๆ เรียกว่า ช่องเวลา สำหรับให้เครื่องลูกข่ายแต่ละเครื่องสลับกันรับส่งข้อมูลเพื่อติดต่อสื่อสารกับสถานีฐาน โดยกำหนดให้มีโครงสร้างที่เรียกว่า เฟรม แบบ TDMA (Time Division Multiple Access) ซึ่งแต่ละเฟรมประกอบไปด้วยช่วงเวลาจำนวน 8 ช่อง ในกรณีของสถานีฐานสนับสนุนเทคโนโลยี GPRS นั้นทั้งเครื่องลูกข่ายที่เป็นแบบ GSM และเครื่องลูกข่ายแบบ GPRS ต่างก็สามารถติดต่อสื่อสารกับเครื่องข่ายโทรศัพท์เคลื่อนที่ผ่านทางช่วงเวลาได้เท่าเทียมกัน

ในกรณีของมาตรฐานโทรศัพท์เคลื่อนที่ GSM จะมีการกำหนดนิยามช่องสื่อสารทางตรรกะขึ้นซ้อนบนช่วงเวลา โดยที่ช่องสื่อสารทางตรรกะแต่ละประเภทจะมีรูปแบบการใช้งานที่แตกต่างกันสำหรับมาตรฐาน GPRS ก็มีการกำหนดช่องสื่อสารทางตรรกะสำหรับการติดต่อสื่อสารแบบแพ็กเก็ต ซึ่งเป็นไปในลักษณะเดียวกันกับในมาตรฐาน GSM อย่างไรก็ตามรายละเอียดของการจัดการช่องสื่อสารทางตรรกะและรูปแบบการใช้งานในมาตรฐาน GPRS ก็มีรายละเอียดพอสังเขปในรูปที่ 1.3 ซึ่งสามารถจัดกลุ่มในการอธิบายได้ 3 ประเภทด้วยกันดังนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.13 การแบ่งกลุ่มของช่องสื่อสารทางตรรกตามมาตรฐาน โทรศัพท์เคลื่อนที่ GPRS

กลุ่มที่ 1: ช่องสื่อสาร Packet Common Control Channel หรือ PCCCH มีหน้าที่การทำงานโดยทั่วไปคล้ายคลึงกับกลุ่มช่องสื่อสาร Common Control Channel (PCCCH) ของระบบโทรศัพท์เคลื่อนที่ GSM ในกลุ่มนี้สามารถแบ่งช่องสื่อสารทางตรรกออกได้เป็น 4 ชนิดดังนี้

1. Packet Random Access channel หรือ PRACH ใช้สำหรับการส่งข้อมูลจากเครื่องลูกข่าย GPRS ไปยังสถานีฐานในทิศทางขาขึ้น สำหรับเริ่มต้นการร้องขอสร้างวงจรเชื่อมต่อกับเครือข่ายโทรศัพท์เคลื่อนที่ GSM ในกรณีของการรับส่งข้อมูลแบบแพ็คเกจ ช่องสื่อสารชนิดนี้มีความคล้ายคลึงกับช่องสื่อสารแบบ RACH ในกรณีของมาตรฐาน GSM

2. Packet Paging Channel หรือ PPCH ใช้สำหรับการส่งข้อมูลจากสถานีฐานไปยังเครื่องลูกข่าย GPRS ในกรณีที่มีการขอเชื่อมต่อผ่านเครือข่าย โทรศัพท์เคลื่อนที่ไปยังเครื่องลูกข่าย GPRS เครื่องใดเครื่องใดเครื่องหนึ่งในเครือข่าย สิ่งที่ต้องทราบก็คือเครือข่าย GPRS จะใช้ช่องสื่อสาร PPCH นี้สำหรับการส่งสัญญาณเรียกติดต่อเฉพาะกรณีของเครื่องลูกข่าย GPRS คลาส A และ B เท่านั้น

3. Packet Access Grant Channel หรือ PACGH ใช้ในทิศทางการสื่อสารขาลง โดยสถานีฐานจะใช้ในการแจ้งยืนยันไปยังเครื่องลูกข่าย GPRS ว่าสามารถเริ่มต้นการสร้างวงจรเชื่อมต่อเพื่อทำการสื่อสารข้อมูลได้ ช่องสื่อสารนี้มีรูปแบบการใช้งานคล้ายคลึงกับช่องสื่อสารควบคุม AGCH ในมาตรฐาน GSM

4. Packet Notification Channel หรือ PNCH เป็นช่องสื่อสารชนิดใหม่สำหรับการสื่อสารแบบกระจายระหว่างสถานีฐานไปยังเครื่องลูกข่าย GPRS หลายๆเครื่องพร้อมๆกัน สำหรับการส่งข่าวสารแจ้งรายละเอียดการจัดสรรช่องสื่อสารให้กับโทรศัพท์เคลื่อนที่ GPRS แต่ละเครื่อง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กลุ่มที่ 2: ช่องสื่อสาร Packet Broadcast Control Channel หรือ PBCCH ในกลุ่มนี้มีช่องสื่อสารควบคุมเพียงชนิดเดียว PBCCH เป็นช่องสื่อสารในทิศทางขาลง ซึ่งสถานีฐานจะทำการส่งข้อมูลที่สำคัญเกี่ยวกับรายละเอียดการเชื่อมต่อและติดต่อสื่อสารกับเครือข่ายให้กับเครื่องลูกข่าย GPRS ทั้งนี้เครื่องลูกข่าย GPRS จะทราบว่าสามารถอ่านค่าข้อมูลที่อยู่ในช่องสื่อสารควบคุม PBCCH ในช่วงเวลาใดๆ ได้จากการส่งข่าวจากช่องสื่อสารควบคุม BCCH ซึ่งอยู่ในช่องเวลามาตรฐานของการสื่อสารแบบ GSM ผู้ให้บริการระบบเครือข่ายสามารถเลือกกำหนดได้ว่า จะรวมให้มีการกระจายข่าวสารที่จำเป็นในการติดต่อสื่อสารแบบแพ็คเกจ (GPRS) และสวิตซ์วงจร (GSM) โดยรวมส่ง BCCH ก็ได้

กลุ่มที่ 3: ช่องสื่อสาร Packet Traffic Channel หรือ PTCH เป็นกลุ่มของช่องสื่อสารที่ทำหน้าที่เกี่ยวข้องกับการรับส่งข้อมูลของผู้ใช้บริการซึ่งมีการรับส่งในรูปแบบของแพ็คเกจ ประกอบไปด้วยช่องสื่อสารควบคุม 3 ชนิด ดังนี้

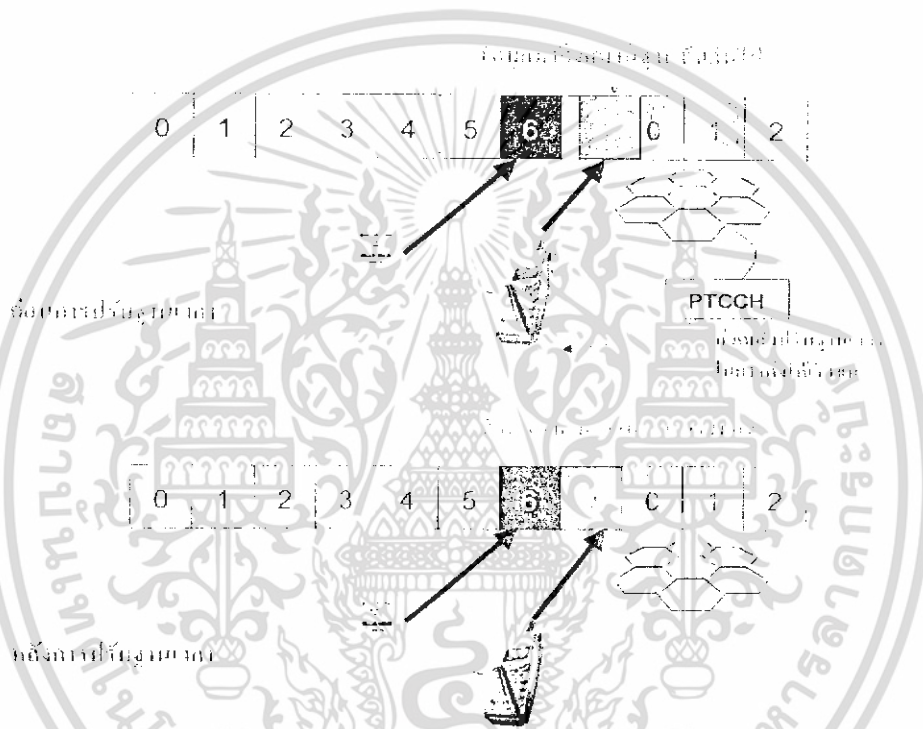
1. Packet Data Traffic Channel (PDTCH) เป็นช่องสื่อสารที่มีอยู่ทั้งในทิศทางขาขึ้นและขาลง โดยทำหน้าที่เป็นช่องทางในการรับส่งข้อมูลของผู้ใช้บริการระหว่างเครื่องลูกข่าย GPRS และสถานีฐาน ตำแหน่งการจัดสรรช่องสื่อสาร PDTCH ในช่องเวลาบนมาตรฐานเฟรมแบบ TDMA นั้นจะเป็นไปตามรูปแบบซึ่งจะได้กล่าวถึงต่อไป แต่หากจะกล่าวโดยง่ายแล้วก็คือช่องสื่อสาร PDTCH หนึ่งช่องจะถูกจัดสรรบนช่วงเวลาหนึ่งช่อง เครื่องลูกข่าย GPRS 1 เครื่อง จะสามารถร้องขอจากระบบเครือข่ายเพื่อทำการรับส่งข้อมูล เพื่อทำการรับส่งข้อมูลผ่านช่องสื่อสาร PDTCH เพียงหนึ่งช่องหรือมากกว่าหนึ่งช่องได้ในเวลาเดียวกัน นอกจากนั้นมาตรฐาน GPRS ยังสามารถกำหนดให้เครื่องลูกข่าย GPRS หลายๆ เครื่อง แบ่งกันใช้ช่องสื่อสาร PDTCH ได้มากกว่า 1 ช่องพร้อมๆกัน โดยให้อยู่ภายใต้การควบคุมจากระบบเครือข่ายผ่านทางช่องสื่อสารควบคุม PNCH ถือเป็นช่องสื่อสารที่มีความคล้ายคลึงกันกับช่องสื่อสาร TCH ในกรณีของมาตรฐาน GSM

2. Packet Associated Control Channel (PACCH) เป็นช่องสื่อสารควบคุมที่ถูกจัดให้ฝังตัวอยู่ร่วมกับช่องสื่อสารรับส่งข้อมูล ซึ่งมีรายละเอียดการจัดวางช่องสื่อสารดังแสดงในรูปที่ 11.10 โดยจะได้กล่าวต่อไปมีอยู่ทั้งในทิศทางขาขึ้นและขาลง โดยทั้งเครื่องลูกข่าย GPRS และสถานีฐานจะใช้งานช่องสื่อสารควบคุมดังกล่าว ในกรณีของการควบคุมกำลัง (Power Control) และการติดต่อสื่อสารในขณะที่เครื่องลูกข่าย GPRS เครื่องนั้นๆ อยู่ในช่วงของการทำงานรับส่งข้อมูลแบบแพ็คเกจกับสถานีฐาน

3. Packet Timing-Advance Control Channel (PTCCH) เป็นช่องสื่อสารควบคุมชนิดใหม่ที่ถูกกำหนดให้มีการใช้งานในการปรับแก้ไขฐานเวลาภายในเครื่องลูกข่าย GPRS โดยมีวัตถุประสงค์เพื่อป้องกันปัญหาการชนกันของข้อมูล (Collision) ที่ถูกส่งจากเครื่องลูกข่าย GPRS แต่ละเครื่องมายังภาครับของสถานีฐาน โดยเครื่องลูกข่าย GPRS แต่ละเครื่องจะทำการส่งข้อมูลประมาณค่าความแตกต่างของช่วงเวลา (Timing advance หรือ TA) ที่คำนวณได้กลับมายังสถานีฐาน เรื่องของ TA นั้นเป็นมาตรฐานดั้งเดิมที่ถูกกำหนดขึ้นในระบบโทรศัพท์เคลื่อนที่ GSM ใช้ในการแสดงระยะห่างระหว่างเครื่องลูกข่ายแต่ละเครื่องกับสถานีฐาน

เครื่องลูกข่ายที่อยู่ห่างจากสถานีฐานย่อมมีโอกาสส่งข้อมูลกลับมายังฐานเวลาในลักษณะเวลาที่เหลื่อมกับตำแหน่งช่องเวลาที่ควรจะเป็นได้มากกว่าเครื่องลูกข่ายที่อยู่ใกล้สถานีฐานมากกว่า ทั้งนี้เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ในการป้องกันการเกิดปัญหาของข้อมูลเหลือทิ้ง ซึ่งทำให้เกิดการปะทะกันของข้อมูลจากเครื่องลูกข่ายที่มีการรับส่งข้อมูลในช่องเวลาที่ติดกันนั้น สำหรับมาตรฐาน GPRS สถานีฐานจะทำการคำนวณฐานเวลาที่เหมาะสมให้กับเครื่องลูกข่ายแต่ละเครื่อง พร้อมทั้งทำการส่งสัญญาณแจ้งเพื่อขอปรับเปลี่ยนฐานเวลาไปยังเครื่องลูกข่ายแต่ละเครื่องโดยผ่านทางช่องสื่อสาร PTCCCH เครื่องลูกข่ายที่อยู่ห่างจากสถานีฐานมาก ซึ่งมีโอกาสจะส่งข้อมูลมาถึงสถานีฐานล่าช้าก็จะได้รับการปรับฐานเวลาภายในตัวเครื่องให้ทำงานเร็วขึ้นกว่าเครื่องลูกข่ายที่อยู่ใกล้สถานีฐานมากกว่าดังตัวอย่างซึ่งแสดงได้ดังรูปที่ 11.4 ซึ่งเป็นการปรับปัญหาเวลาของเครื่องลูกข่าย GPRS ที่มีการรับส่งข้อมูลในช่องเวลาที่ 7 ให้ส่งข้อมูลกลับมายังสถานีฐานในเวลาตรงกับช่วงเวลาของตนพอดี



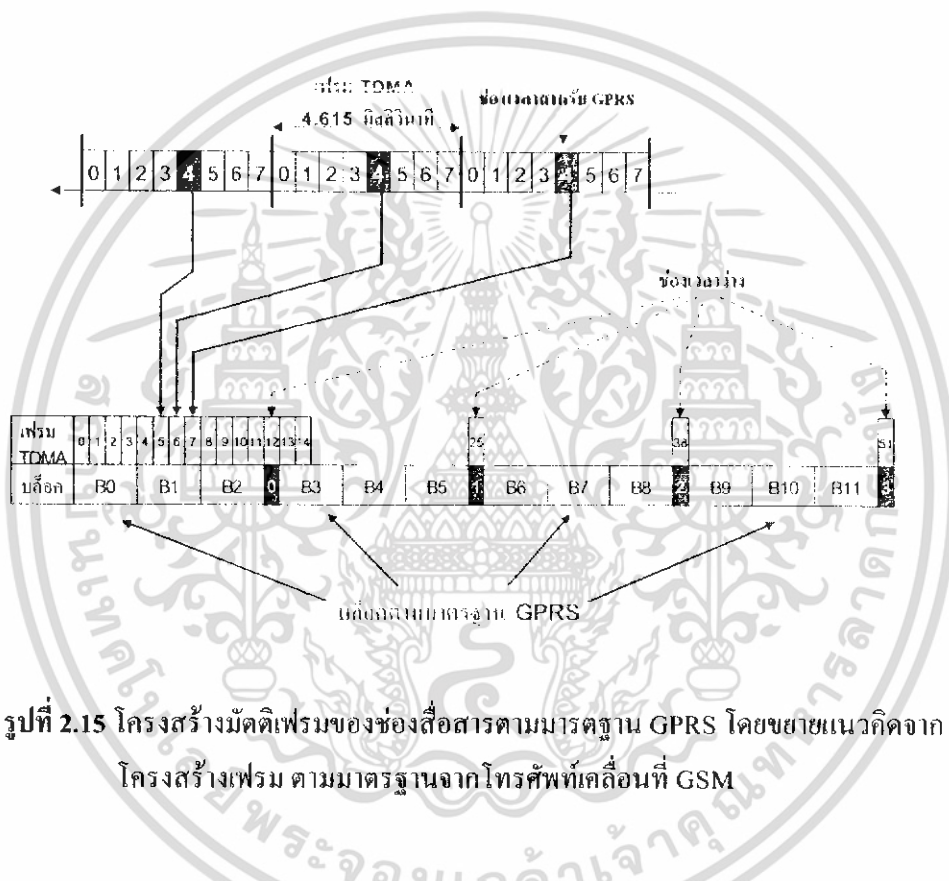
รูปที่ 2.14 การเกิดปัญหาช่วงเวลาไม่ตรงกัน และการแก้ไขโดยกระบวนการ Timing Advance

ตามมาตรฐาน GSM กำหนดไว้ว่าสถานีฐานแต่ละแห่งจะมีรัศมีครอบคลุมได้ไกลที่สุดไม่เกิน 35 กิโลเมตร ทั้งนี้ข้อกำหนดดังกล่าวเกี่ยวข้องกับารออกแบบรูปแบบของ Burst แบบ Random Access ก็คือปัจจัยในเรื่องของการกำหนดขอบเขตของเซลล์สถานีฐานก็นับเป็นอีกหนึ่งปัจจัยต่อค่า TA ของเครื่องลูกข่าย กลไกในการปรับฐานเวลาของโทรศัพท์เคลื่อนที่ GSM และ GPRS จะทำงานได้ดีหรือไม่ จึงขึ้นอยู่กับแนวทางในการออกแบบวางเครือข่ายของผู้ให้บริการเป็นสำคัญ

สำหรับแนวทางในการใช้งานช่องสื่อสารทางตรรกะที่มีอยู่เป็นจำนวนมากนี้ มาตรฐานระบบโทรศัพท์เคลื่อนที่ GPRS ได้แบ่งแนวทางในการจัดสรรใช้งานออกเป็น 2 ระยะ คือ ระยะที่ 1 ซึ่งอยู่ในช่วงปัจจุบันที่มาตรฐาน GPRS ถือว่าอยู่ในช่วงเริ่มต้น กำหนดให้การใช้งานช่องสื่อสารในกลุ่ม CCCH ตาม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

มาตรฐาน GSM สำหรับทำหน้าที่แทนกลุ่มช่องสื่อสาร BCCH ตามมาตรฐาน GSM แทนช่องสื่อสาร PBCCH จนกว่าการพัฒนาเครื่องลูกข่ายรุ่นใหม่ๆ และสถานีฐาน GSM จะมีความก้าวหน้ามากขึ้น จึงจะประกาศใช้งานช่องสื่อสารควบคุมตามมาตรฐาน GPRS นั้นหมายความว่า ETSI ซึ่งเป็นผู้ว่ามาตรฐาน GPRS ได้มีการพัฒนาข้อกำหนดของช่องสื่อสารตามมาตรฐาน GSM ให้มีศักยภาพมากขึ้นเพื่อรองรับการทำงานตามมาตรฐาน GPRS ตัวอย่างเช่น ในช่องสื่อสารควบคุมแบบ BCCH นั้น ก็ได้มีการเพิ่มเติมประเภทของการส่งข้อมูลแจ้งสถานะและรายละเอียดการใช้งานชนิดใหม่ ที่มีชื่อว่า System Information Type13 หรือ SI13ขึ้นสำหรับใช้แจ้งข่าวสารไปยังเครื่องลูกข่าย GPRS โดยเฉพาะ



รูปที่ 2.15 โครงสร้างมัลติเฟรมของช่องสื่อสารตามมาตรฐาน GPRS โดยขยายแนวคิดจากโครงสร้างเฟรม ตามมาตรฐานจากโทรศัพท์เคลื่อนที่ GSM

โครงสร้างมัลติเฟรมของช่องสื่อสารบนมาตรฐาน GPRS

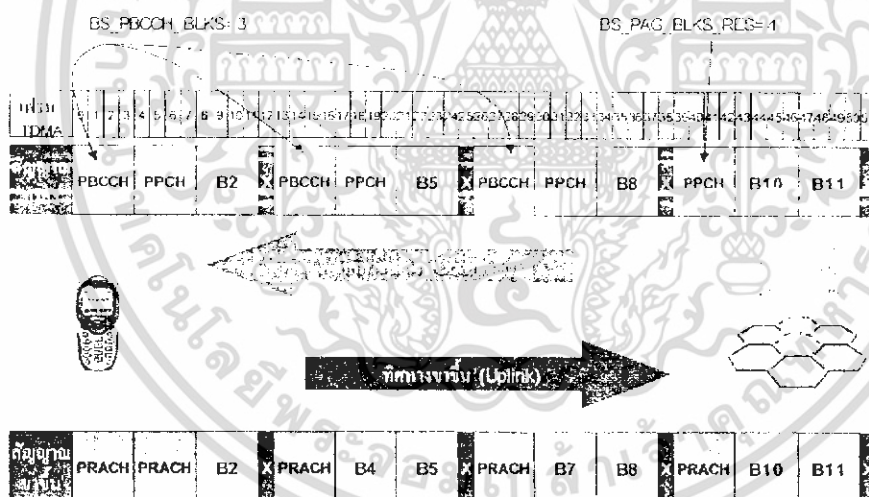
เราได้กล่าวถึงชื่อของช่องสื่อสารทางตรรกตามมาตรฐาน GPRS ไปหลายครั้ง แต่เชื่อว่าคงยังมองไม่เห็นความข้องเกี่ยวข้องระหว่างการใช้งานช่องสื่อสารดังกล่าว กับโครงสร้างของเฟรมแบบ TDMA ตามมาตรฐาน GSM ซึ่งเป็นที่คุ้นเคยกันตามปกติจึงขอแสดงความสัมพันธ์ของการจัดโครงสร้างมัลติเฟรม (Multiframe) ของมาตรฐาน GPRSซึ่งมีความคล้ายคลึงกับการจัดมัลติเฟรมตามมาตรฐาน GSM ดังแสดงรูปที่ 6 สาเหตุของการกำหนดนิยามของมัลติเฟรมขึ้นก็เพื่อเป็น ข้อตกลงร่วมกันระหว่างเครื่องลูกข่าย GPRS กับสถานีฐานว่าจะสามารถรับหรือส่งข้อมูลในช่องสื่อสารทางตรรกแต่ละกลุ่มในช่วงเวลาใด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตัวอย่างการจัดโครงสร้างมัลติเฟรมของกลุ่มช่องสื่อสารควบคุมและช่องสื่อสารสนทนา

สิ่งที่ควรให้ความสนใจก็คือ แนวทางในการมัลติเพล็กซ์ช่องสื่อสารทางตรรกษณ์ค่างๆ บนมัลติเฟรม GPRS นั้นมีความแตกต่างจากมัลติเพล็กซ์ช่องสื่อสารบนมัลติเฟรม GSM ค่อนข้างมากกล่าวคือ ในกรณีของมาตรฐานเครือข่าย GPRS จะมีพารามิเตอร์หลายชนิดทำหน้าที่ควบคุมและแสดงรายละเอียดของการมัลติเพล็กซ์ ซึ่งทำให้สามารถปรับเปลี่ยนรายละเอียดของการมัลติเพล็กซ์ได้หลายรูปแบบ แตกต่างจากในกรณีของเครือข่าย GSM ที่มีกรมัลติเพล็กซ์แบบค่อนข้างยึดติดตายตัว นอกจากนี้ลำดับของการอ่านข้อมูลของช่องสื่อสารที่มีการกำหนดไว้เป็นบล็อกก็จะเป็นไปตามก็จะเป็นไปตามลำดับดังต่อไปนี้ คือ B0,B6,B3,B9,B1,B7,B4,B10,B2,B8,B5 และB11

รูปที่ 11.8 เป็นตัวอย่างของการจัดโครงสร้างช่องสื่อสารควบคุมแบบ PBCCH+PCCCH ซึ่งเราจะเห็นได้ว่าความสัมพันธ์ระหว่างพารามิเตอร์ BS_PBCCH_BLK=3 และ BS_PAG_BLK_RES=1 ใช้ระบุโครงสร้างของการจัดมัลติเฟรมได้เป็นอย่างดี อย่างไรก็ตามเครื่องลูกข่ายโทรศัพท์เคลื่อนที่ที่เปรียบเสมือนคนตาบอดที่ไม่เคยทราบมาก่อนว่าโครงสร้างของมัลติเฟรม GPRS สำหรับช่องสื่อสารควบคุมมีลักษณะอย่างไร พารามิเตอร์ต่างๆ เหล่านี้จึงเปรียบเสมือนกับแผนที่ให้กับเครื่องลูกข่ายโทรศัพท์เคลื่อนที่ GPRS ในการทำความเข้าใจรายละเอียดของโครงสร้างมัลติเฟรมได้เป็นอย่างดี

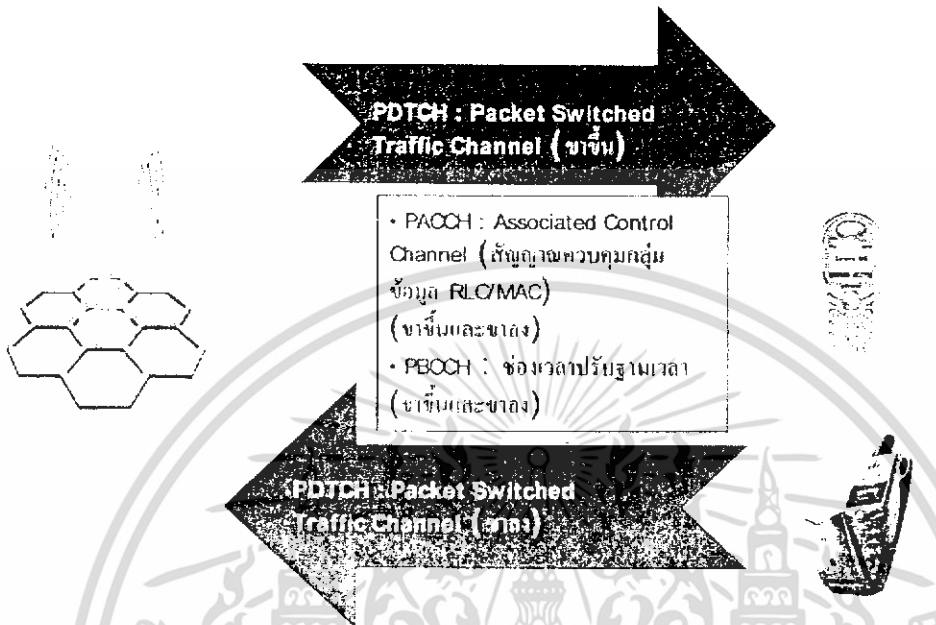


รูปที่ 2.16 ตัวอย่างการจัดโครงสร้างเฟรมข้อมูล GPRS ชนิดควบคุมประเภท PBCCH+PCCCH

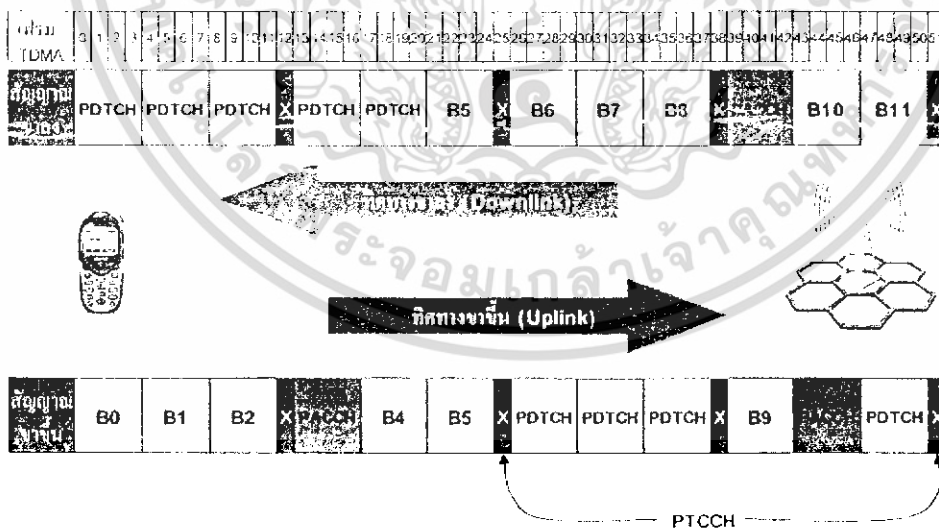
รูปที่ 2.16 เป็นการสรุปรายละเอียดของช่องสื่อสารทางตรรกษณ์ที่เกี่ยวข้องกับการรับส่งข้อมูลแบบแพ็กเกต ในกลุ่มของช่องสื่อสารแบบ PTCH ซึ่งเป็นไปตามข้อมูลที่ได้อ้างไว้ไปแล้ว สิ่งที่เราต้องการขยายความเพิ่มเติมเกี่ยวกับการจัดสรรใช้งานช่องสื่อสารแบบ PDTCH ก็คือมาตรฐาน GPRS เปิดโอกาสให้เครื่องลูกข่ายโทรศัพท์เคลื่อนที่ GPRS เครื่องสามารถใช้งานช่วงเวลา 8 ช่องบนเฟรมแบบ TDMA เดียวกัน (ที่รับส่งขนาดความถี่เดียวกัน) สำหรับการติดต่อสื่อสารแบบแพ็กเกตได้พร้อมๆ กันในเวลาเดียวกัน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

และผู้ให้บริการโทรศัพท์เคลื่อนที่ GPRS สามารถแบ่งกันใช้ช่องสื่อสารในกลุ่ม PDCH ได้พร้อมๆกันเวลาเดียวกันสูงสุดถึง 8 ราย



รูปที่ 2.17 ช่องสื่อสารทางตรรกที่เกี่ยวกับการรับส่งข้อมูลแบบเพิกเกิดระหว่างเครื่องข่ายโทรศัพท์เคลื่อนที่ GPRS กับผู้ให้บริการผ่านการเชื่อมต่อทางวิทยุ



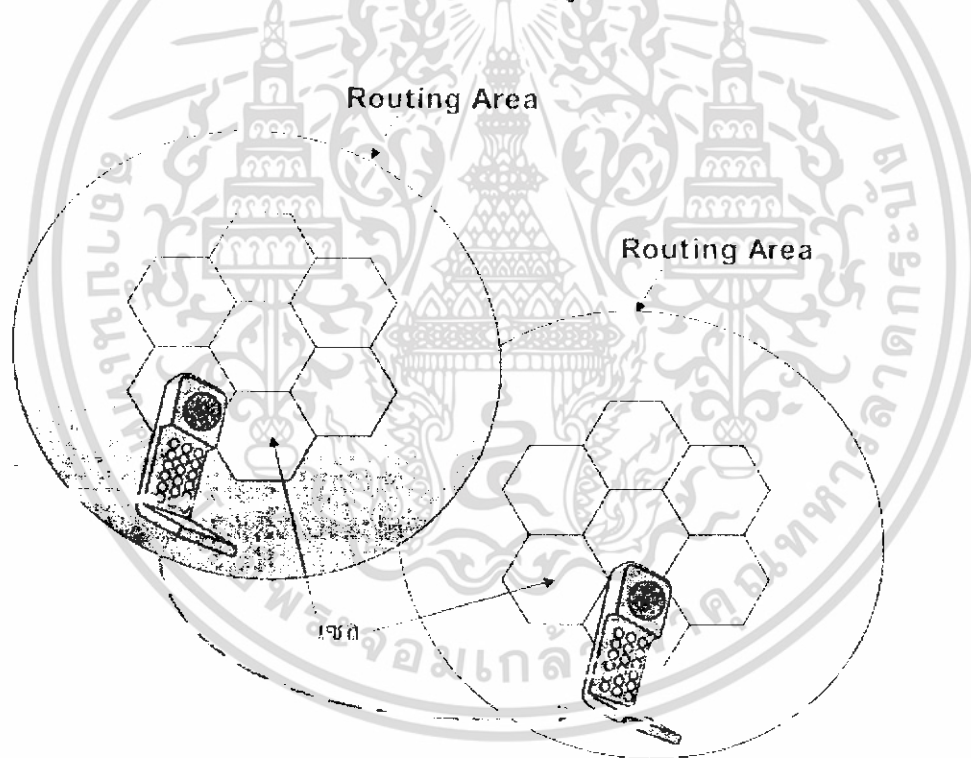
รูปที่ 2.18 ตัวอย่างการจัดโครงสร้างเฟรมข้อมูล GPRS ชนิดสื่อสาร ประเภท PDTCH

รูปที่ 2.18 เป็นการจัดโครงสร้างมัลติเฟรมสำหรับช่องสื่อสารแบบ PDTCH ซึ่งจะเห็นว่ามีการรวมช่องสื่อสารควบคุมแบบ PACCH ซึ่งมีหน้าที่การทำงานที่เทียบเคียงได้กับช่องสื่อสารควบคุมแบบเอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

SDCCH+SACCH+FACCH ในกรณีของมาตรฐานโทรศัพท์เคลื่อนที่ GSM ไปด้วยกันกับช่องสื่อสารแบบ PDTCH นอกจากนั้น ขอให้สังเกตตำแหน่งที่อยู่ของช่องสื่อสารแบบ PDTCH ทั้งในทิศทางการสื่อสารขาขึ้นและขาลงซึ่งใช้สำหรับปรับแต่งฐานเวลาให้กับเครื่องลูกข่ายโทรศัพท์เคลื่อนที่ ตามกลไกการแก้ไขปรากฏการณ์ TA ตามกระบวนการที่ได้กล่าวถึงไปแล้ว

เขตและ Routing Area

ในการสื่อสารข้อมูลระหว่างโทรศัพท์เคลื่อนที่ GPRS กับเครือข่ายสถานีฐานนั้น ผู้ให้บริการระบบเครือข่ายจำเป็นต้องทำการกำหนดพื้นที่ของการให้บริการออกเป็นส่วนๆ เรียกว่า Routing Area (RA) โดยพื้นที่ของแต่ละ RA จะครอบคลุมจำนวนเซลล์หลายๆเซลล์ มากหรือน้อยขึ้นอยู่กับเงื่อนไขเกี่ยวกับความบ่อยของการเคลื่อนที่เข้าข้าม RA แนวคิดของการกำหนด RA ทั่วไปในลักษณะเดียวกันกับการกำหนดพื้นที่ที่เรียกว่า Location Area หรือ LA ในกรณีของการสื่อสารผ่านเครือข่าย GSM ที่นำสังเกตก็คือพื้นที่ของ RA อาจจะเท่าหรือไม่เท่ากับ LA ก็ได้ เนื่องจากพฤติกรรมการสื่อสารข้อมูลแบบเครือข่าย GPRS เทียบกับการสนทนาผ่านเครือข่าย GSM ซึ่งซ้อนทับกันอยู่นั้นมีความแตกต่างกันอย่างสิ้นเชิง



รูปที่ 2.19 เปรียบเทียบความแตกต่างระหว่างเซลล์และ Routing Area (RA)

รูปที่ 2.19 เป็นการแสดงความหมายของ RA กับเซลล์ผ่านในเครือข่าย GSM เมื่อมองในมุมมองของการสื่อสารแบบ GPRS ถ้าถามที่ว่า จะออกแบบ RA ให้ครอบคลุมพื้นที่กี่เซลล์จึงจะเหมาะสมสามารถตอบได้ง่ายโดยเราพึงเข้าใจว่าอุปกรณ์ SGSN จะทราบตำแหน่งที่อยู่เครื่องลูกข่าย GPRS โดยอ้างอิงในหน่วยย่อยที่สุดคือ RA หากผู้เขียนถือเครื่องลูกข่ายอยู่ในบริเวณสยามสแควร์ เมื่อมีการส่งข้อมูลผ่านเอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

อุปกรณ์ SGSN มายังเครื่องลูกข่ายของเรา SGSN จะทำการส่งสัญญาณเรียก ค้นหา หรือที่เรียกว่าเพจใน ทุกๆเซลล์ที่ประกอบขึ้นเป็นพื้นที่ RA เดียวกับที่เราใช้เครื่องลูกข่ายอยู่ ดังนั้นจึงไม่แปลกกว่าอาจมีสัญญาณ เพจปรากฏในเซลล์ที่เราใช้งานพร้อมๆกับในเซลล์อื่นๆ ซึ่งการส่งสัญญาณเพจนั้น จะกระทำผ่าน ช่องสื่อสารทางตรรก PPCH (Packet Paging Channel)

ถ้าผู้ให้บริการกำหนดพื้นที่ของ RA ให้กว้างเกินไปซึ่งหมายถึงว่าครอบคลุมจำนวนเซลล์มาก เกินไปปริมาณของข้อมูลเพจที่ถูกส่งไปยังเครื่องลูกข่าย GPRS แต่ละเครื่องก็จะเกิดขึ้นในวงกว้าง ส่งผล กระทบต่อปริมาณข้อมูล GPRS ที่เกิดขึ้นในเซลล์อื่นๆ แต่อย่างไรก็ตามผู้ให้บริการเครือข่ายก็พึง ระวังไม่กำหนดให้พื้นที่ของ RA เล็กเกินไป เนื่องจากเมื่อเครื่องลูกข่าย GPRS มีการเคลื่อนที่จาก พื้นที่เซลล์ของ RA ไปยังอีก RA หนึ่งจะส่งผลให้เครื่องลูกข่ายต้องส่งข้อมูลปรับเปลี่ยนตำแหน่งพื้นที่ RA ไปให้กับอุปกรณ์ SGSN โดยใช้ช่องสื่อสาร PACCH (Packet Associated Control Channel) ยิ่งพื้นที่ RA เล็กมากเท่าใด โอกาสที่เครื่องลูกข่ายจะมีการเคลื่อนที่ข้าม RA ก็ยิ่งมากขึ้น ส่งผลกระทบต่อปริมาณข้อมูล GPRS อีกเช่นเดียวกัน การกำหนดขนาดของ RA จึงเป็นการลองผิดลองถูกในระยะแรกของการเปิด ให้บริการ จนกว่าผู้ให้บริการจะทราบถึงพฤติกรรมของการใช้บริการภายในเครือข่ายของตน

กระบวนการแสดงตัวต่อเครือข่าย GPRS

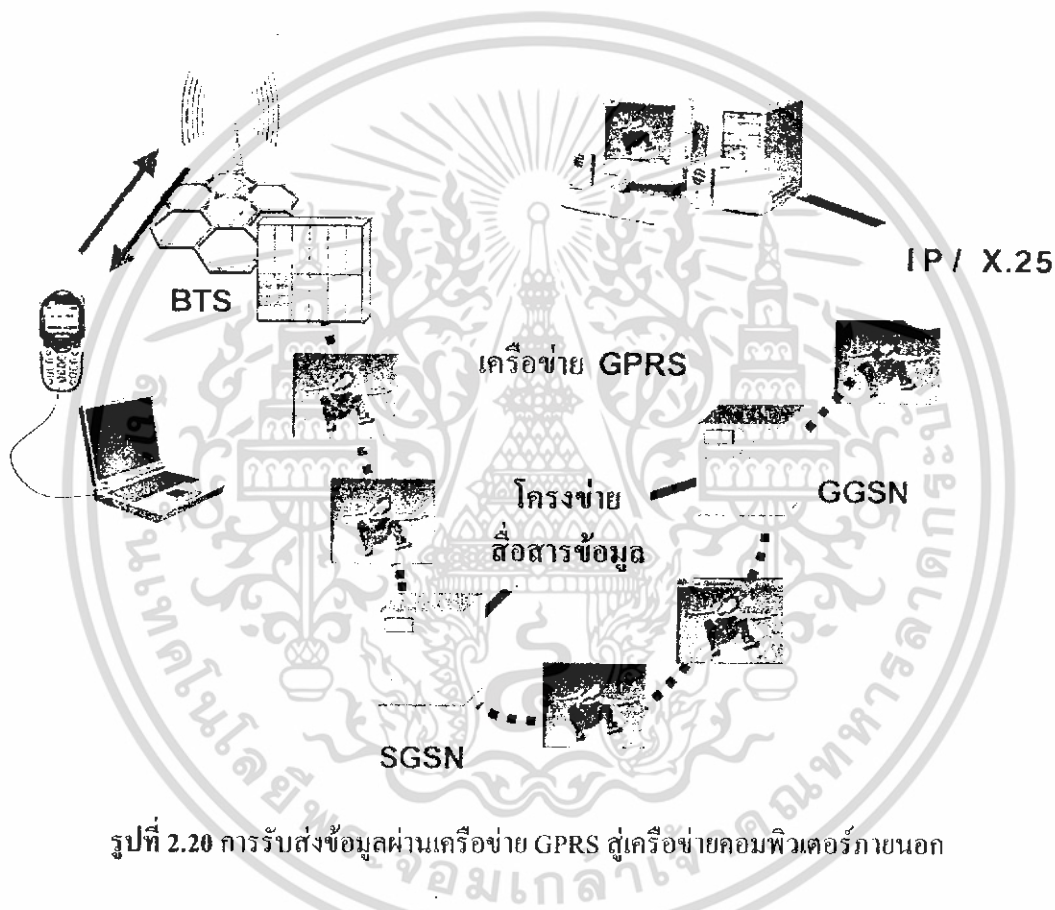
เมื่อใดก็ตามที่เครื่องลูกข่ายโทรศัพท์เคลื่อนที่ GPRS ต้องการมีใช้บริการสื่อสารข้อมูลผ่าน เครือข่ายโทรศัพท์เคลื่อนที่ GPRS สิ่งที่ต้องเกิดขึ้นเป็นอันดับแรกก็คือ เครื่องลูกข่ายจะทำการแสดงตัวต่อ อุปกรณ์ SGSN โดยที่อุปกรณ์ SGSN จะทำการตรวจสอบเงื่อนไขสำคัญ 3 ประการจากเครื่องลูกข่ายเพื่อ เป็นการยืนยันสิทธิในการใช้บริการของเครื่องลูกข่าย พร้อมกับจัดเตรียมระดับคุณภาพของการให้บริการ ไปในเวลาเดียวกัน เงื่อนไขทั้ง 3 ประการได้แก่

1. Authorization เป็นการตรวจสอบว่าเลขหมายโทรศัพท์เคลื่อนที่ซึ่งแสดงตัวด้วยแผ่นซิม ภายในตัวเครื่องลูกข่าย ได้รับอนุญาตให้ใช้บริการ GPRS หรือไม่ ผู้ใช้บริการที่ยังไม่ได้เปิดบริการ GPRS จะไม่ได้รับสิทธิในการใช้บริการดังกล่าว โดยการตรวจสอบตั้งแต่ขั้นตอนนี้
2. Authentication เป็นการตรวจสอบความถูกต้องของเลขหมายผู้ให้บริการ ว่าเป็นผู้ให้บริการที่ ลงทะเบียนภายในเครือข่ายโทรศัพท์เคลื่อนที่ GSM ถูกต้องหรือไม่ ในกรณีนี้อุปกรณ์ SGSN จะทำหน้าที่ ตรวจสอบความถูกต้องของแผ่นซิมโดยมีการทำงานร่วมกับอุปกรณ์ HLR/AuC ภายในเครือข่าย GSM ที่ เก็บข้อมูลเกี่ยวกับผู้ให้บริการ (เลขหมาย IMSI รหัส KI) เพื่อใช้กลไกการตรวจยืนยันผู้ให้บริการตาม ขั้นตอนที่หนึ่งของเครือข่าย GSM ตามปกติ 1
3. Quality of Service เพื่อตรวจสอบว่าผู้ให้บริการที่ขอใช้บริการ GPRS แต่ละรายนั้นมีความจำเป็นขอ ใช้บริการรับส่งข้อมูลภายใต้การควบคุมคุณภาพในระดับใดเครือข่ายสามารถให้บริการภายใต้ระดับ คุณภาพที่ต้องการนั้นได้หรือไม่ และหากสามารถให้บริการได้แล้ว จะทำให้คุณภาพในการสื่อสารข้อมูล ของผู้บริการรายอื่นภายในเซลล์นั้นๆ ลดต่ำลงหรือไม่

หากอุปกรณ์ SGSN ตัดสินใจยอมรับเงื่อนไขข้างต้นทั้ง 3 ประการ และเปิดโอกาสให้เครื่องลูกข่าย สามารถใช้บริการ GPRS ได้ อุปกรณ์ SGSN จะทำการติดตามเครื่องลูกข่ายนั้นอยู่ตลอดเวลาทราบเท่าที่ เครื่องลูกข่ายดังกล่าวยังเปิดเครื่องใช้งานอยู่ภายในพื้นที่ภายใต้การควบคุมของอุปกรณ์ SGSN นั้น ซึ่งอาจ เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ประกอบด้วยพื้นที่ RA จำนวนหลายๆ ชุด ทั้งนี้ก็เพื่อจะสามารถทราบได้ว่าเครื่องลูกข่ายดังกล่าวต้องการรับส่งข้อมูลกับเครือข่ายเมื่อใด การแสดงตัวต่ออุปกรณ์ SGSN นั้น มิใช่การรับประกันว่าเครื่องลูกข่าย GPRS ดังกล่าวจะสามารถรับส่งข้อมูลกับเครือข่ายใดเมื่อใดก็ได้ โดยไม่มีการแจ้งต่ออุปกรณ์ SGSN อีก ทุกครั้งที่ต้องการรับส่งข้อมูลเครื่องลูกข่ายดังกล่าวจะต้องทำการส่งข้อมูล PDP เพื่อแสดงรายละเอียดต่างๆ เกี่ยวกับการเชื่อให้กับอุปกรณ์ SGSN เป็นอีกครั้งหนึ่ง

นั่นหมายความว่า ทุกครั้งที่ผู้ใช้ให้บริการนำเครื่องลูกข่ายโทรศัพท์เคลื่อนที่ GPRS มาเลือกรายการเปิดใช้บริการ GPRS นั้น เครื่องลูกข่ายทำการแสดงตัวต่อเครือข่าย ซึ่งก็หมายถึงอุปกรณ์ SGSN ที่ควบคุม ดูแลพื้นที่ที่ใช้งานนั้น เมื่อใดที่เครื่องลูกข่ายมีการเคลื่อนที่ย้ายจากพื้นที่ใช้งานภายใน SGSN หนึ่งไปสู่อีก SGSN หนึ่ง ก็จะมีการโอนย้ายข้อมูลผู้ใช้บริการที่เก็บไว้ใน SGSN ต้นทางไปยัง SGSN ปลายทาง



รูปที่ 2.20 การรับส่งข้อมูลผ่านเครือข่าย GPRS สู่เครือข่ายคอมพิวเตอร์ภายนอก

สำหรับการสื่อสารระหว่างอุปกรณ์ GGSN และ SGSN นั้นมาตรฐาน GPRS ได้มีการกำหนดเทคนิคการสื่อสารชนิดพิเศษขึ้นเรียกว่า การส่งผ่านอุโมงค์ หรือ tunneling เพื่อช่วยให้ผู้ใช้บริการเครือข่าย GSM/GPRS สามารถใช้ประโยชน์ของโครงข่ายสื่อสารข้อมูลที่มีอยู่แล้ว หรือที่ต้องดำเนินการเช่าใช้งานจากผู้อื่นในการเชื่อมต่ออุปกรณ์ GSN ต่างๆ ภายในเครือข่าย GPRS เข้าด้วยกัน ตัวอย่างเช่น เช่าโครงข่าย X.25 จากบริษัท A เพื่อใช้เชื่อมต่อกับอุปกรณ์ GGSN ที่อยู่ที่กรุงเทพฯ กับอุปกรณ์ SGSN 4 ชุด ที่ติดตั้งอยู่ที่ กรุงเทพฯ, เชียงใหม่, ภูเก็ต, ขอนแก่น และชลบุรี ซึ่งเมื่อพิจารณาถึงการทำงานของโครงข่ายภายนอกที่ ต้องมีการจัดรูปแบบแพ็คเกจข้อมูลตามมาตรฐานปกติแล้ว จะทำให้เกิดความสับสนต่อการรับส่งข้อมูล แพ็คเกจ GPRS ระหว่างอุปกรณ์ GSN ด้วยกันได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ทั้งนี้เนื่องจากการรับส่งข้อมูลแพ็คเก็ตโดยทั่วไปนั้น โครงข่ายเข้าภายนอกก็จะต้องตรวจสอบ แอดเดรสต้นทางและปลายทางเพื่อตัดสินใจเลือกเส้นทางในการรับส่งข้อมูลนั้นๆ แต่เนื่องจากแอดเดรสต้นทางและปลายทางที่ปรากฏภายในแพ็คเก็ตข้อมูลของเครือข่าย GPRS อาจเป็นแอดเดรส PDP ของเครื่องลูกข่ายแอดเดรส PDP ของอุปกรณ์ GGSN หรือแอดเดรส IP/X.25 ของเครื่องคอมพิวเตอร์ปลายทางที่เชื่อมต่ออยู่กับเครือข่ายภายนอก ความหวังดีที่จะประหยัดค่านเชื่อมต่ออุปกรณ์ภายใน GSN ภายในเครือข่าย GPRS โดยอาศัยโครงข่ายเข้าภายนอก ย่อมก็ให้เกิดปัญหาต่อการส่งผ่านแพ็คเก็ตข้อมูลในทันทีเนื่องจากโหนดการสื่อสารต่างๆ ภายในโครงข่ายสื่อสารข้อมูลย่อมไม่อาจรู้จักแอดเดรส PDP ที่มีการระบุไว้ในแอดเดรสต้นทางและปลายทางของแพ็คเก็ตข้อมูลอย่างแน่นอน ทำให้ไม่สามารถส่งผ่านข้อมูลภายในโครงข่ายสื่อสารข้อมูลที่ทำกรเข้าใช้งาน หรือแม้จะสร้างขึ้นมาใช้เองเป็นการภายใน

เครื่องลูกข่ายGPRS

มาตรฐาน GPRS กำหนดให้ทั้งเครือข่ายสถานีฐานและเครื่องลูกข่าย สามารถติดต่อสื่อสารเพื่อรับส่งข้อมูล โดยมีการใช้ช่วงเวลา (Timeslot)บนเฟรมมาตรฐาน ได้หลากหลายรูปแบบ โดยมีกำหนดคลาสการใช้งานหลายช่วงเวลา (Multislot Class) ไว้ทั้งสิ้น 29 คลาส เครื่องลูกข่ายโทรศัพท์เคลื่อนที่ GPRS แต่ละรุ่นที่ได้รับการออกแบบมาสำหรับใช้งานในเชิงพาณิชย์ จะได้รับระบุว่ามีสนับสนุนคลาสการใช้งานหลายช่วงเวลาในระดับใด ตารางที่ 11.1 เป็นการแสดงรายละเอียดทางเทคนิคของเครื่องลูกข่ายทั้ง 29 คลาส โดยระบุจำนวนช่วงเวลาสูงสุดที่เครื่องลูกข่ายสามารถใช้ส่งข้อมูลไปยังสถานีฐาน (Tx)โดยในทางปฏิบัติเครื่องลูกข่ายสามารถใช้ส่งข้อมูลไปยังสถานีฐาน (Rx)โดยในทางปฏิบัติเครื่องลูกข่ายสามารถใช้ช่วงเวลาได้ตั้งแต่ 0 ช่องไปจนถึง Tx และ Rx จะต้องไม่เกินค่าผลรวมในตารางช่องที่ 3

ตัวอย่างเช่นเครื่องลูกข่าย Siemens รุ่น 45 เป็นเครื่องลูกข่าย GPRS คลาส 8 หมายความว่า เครื่อง S45 มีความสามารถใช้ช่วงเวลาบนเฟรม TDMA ขาขึ้น (Uplink) สำหรับการส่งข้อมูลไปยังเครือข่าย GPRS ได้พร้อมกันสูงสุด 1 ช่องเวลา และรับข้อมูลจากเครือข่ายได้โดยใช้ช่วงเวลาบนเฟรม TDMA ขอลงได้สูงสุดถึง 4 ช่อง แต่อย่างไรก็ตามในทางปฏิบัติผลรวมของค่า Tx และ Rx จะต้องไม่เกินค่าไม่เกินกว่า 5 ช่องเวลา นั่นเอง เครื่องลูกข่ายดังกล่าวอาจมีจำนวนช่วงเวลาในเฟรมข้อมูลขาขึ้นและขาลงในสัดส่วนเท่ากัน 0 ต่อ 4, 0 ต่อ 3, 0 ต่อ 2, 0 ต่อ 1, 1 ต่อ 4, 1 ต่อ 3, 1 ต่อ 2, 1 ต่อ 1, หรือ 1 ต่อ 0 ในช่วงเวลาใดเวลาหนึ่งก็ได้ อย่างไรก็ตามผู้ให้บริการเครือข่ายจะมีกำหนดจำนวนช่องได้มากตามขีดความสามารถของเครื่องลูกข่าย ทั้งนี้ก็เพื่อให้เกิดความมั่นใจว่ายังมีช่องเวลาเหลือพอสำหรับให้บริการเชื่อมต่อวงจรเพื่อกรสนทนาบนเครือข่าย GSM ได้เพียงพอ ป้องกันการเกิดปัญหาวงจรเชื่อมต่อเต็มจนเกิดผลกระทบต่อความปลอดภัยในการใช้บริการของผู้บริโภค

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

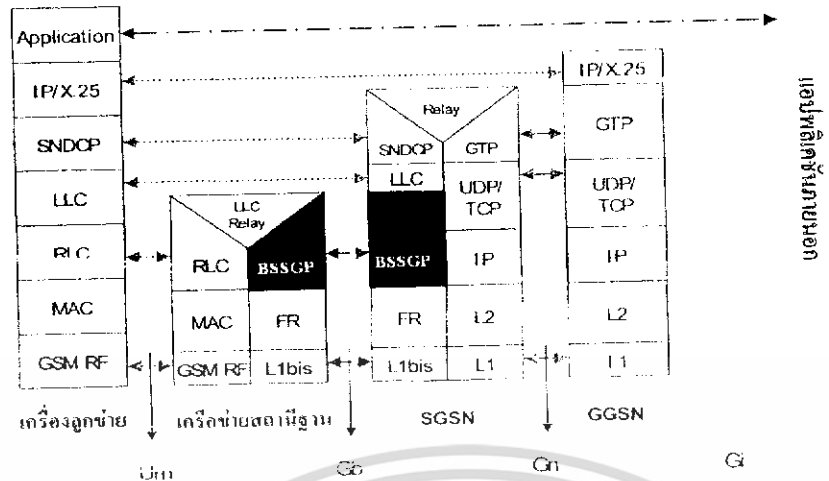
แบบจำลองโปรโตคอลของเครือข่าย GPRS

เราจะเห็นได้ว่าการทำงานของมาตรฐาน GPRS นั้นได้ออกแบบให้แยกออกจากเครือข่าย GSM ก่อนข้างมาก ที่เกี่ยวข้องกันบ้างก็ได้แก่การร่วมใช้ช่องสื่อสารทางกายภาพ (Physical Channel) ระหว่างเครื่องลูกข่ายกับสถานีฐาน (Air Interface) และการเชื่อมต่อระหว่างสถานีฐานกับอุปกรณ์ BSC เมื่อกล่าวถึงจุดนี้เราจะนำเสนอแบบจำลองโปรโตคอลภายในมาตรฐานเครือข่าย GPRS เพื่อให้เกิดความเข้าใจในการทำงานของเครือข่าย GPRS ทั้งหมด และยังเป็นพื้นฐานสำหรับการกล่าวถึงมาตรฐานถึงรายละเอียดเบื้องต้นทางด้านโปรโตคอล รูปที่ 11.18 เป็นแบบจำลองโปรโตคอลสำหรับการเชื่อมต่อทั้งหมดภายในเครือข่าย GPRS นับตั้งแต่จุดเชื่อมต่อระหว่างเครื่องลูกข่ายกับสถานีฐาน ไปจนถึงการเชื่อมต่อระหว่างอุปกรณ์ GGSN กับเครือข่ายสื่อสารข้อมูลภายนอก

เครื่องลูกข่าย

GPRS เชื่อมต่อกับสถานีฐานโดยผ่านทางคลื่นความถี่วิทยุ ข้อมูลหน่วยย่อยที่สุดจะอยู่ในรูปของ Burst ซึ่งเป็นไปตามข้อกำหนดมาตรฐานของเครือข่าย GSM โดยเฟรมข้อมูลที่ผ่านการเข้ารหัสช่องสื่อสารความยาว 456 บิต จะถูกแบ่งออกเป็น 4 ส่วนย่อยๆ ส่วนละ 57 บิตเพื่อนำมาผ่านกระบวนการอินเตอร์ลีฟ (Interleaving) การแยกส่วนและการรวมกลับคืนข้อมูลทั้ง 456 บิตเข้าด้วยกันถือเป็นการใช้กระบวนการปกติของเครือข่าย GSM ที่เรียกว่า GSM RF สำหรับเนื้อหาภายในเฟรมข้อมูลแต่ละชุดนั้นจะมีโปรโตคอล MAC และ RLC ฝังอยู่เพื่อทำหน้าที่ปรับแต่งแบบการเข้ารหัสตามขบวนการที่ได้กล่าวไปแล้ว เมื่อหักข้อมูล MAC/RLC ออกแล้ว ข้อมูลส่วนที่เหลือจะมีส่วนของโปรโตคอล LLC (Logical Link Control) และ SNDCP ซึ่งไม่ถูกนำไปใช้งานโดยเครือข่ายสถานีฐาน แต่จะเป็นการสื่อสารโดยตรงระหว่างเครื่องลูกข่ายกับอุปกรณ์ SGSN โปรโตคอล LLC และ SNDCP ถือเป็นเรื่องสำคัญของการตรวจสอบตำแหน่งของเครื่องลูกข่ายโดยอุปกรณ์ SGSN และ GGSN รวมถึงการส่งสัญญาณควบคุมอื่นๆ อีกหลายชนิดระหว่างเครือข่ายกับเครื่องลูกข่าย GPRS การออกแบบโปรโตคอล LLC และ SNDCP ขึ้นทำให้สามารถนำเทคโนโลยี GPRS ไปใช้ได้บนเครือข่ายโทรศัพท์เคลื่อนที่ชนิดอื่นๆ เช่น เครือข่าย CDMA หรือการพัฒนาเทคโนโลยี GPRS ไปสู่เครือข่ายโทรศัพท์เคลื่อนที่ยุคที่ 3 อันได้แก่ มาตรฐาน UMTS ข้อมูลส่วนที่เหลือจะมีส่วนของโปรโตคอล IP หรือ X.25 ซึ่งเครื่องลูกข่ายใช้ในการติดต่อสื่อสารเพื่อดำเนินการจัดการโปรไฟล์ PDP กับอุปกรณ์ GGSN การเลือกจะใช้โปรโตคอล IP หรือ X.25 นั้นก็ขึ้นอยู่กับกรอบการออกแบบทางด้านวิศวกรรมเครือข่ายของผู้ให้บริการแต่ละราย ข้อมูลหลังจากโปรโตคอล IP/X.25 ก็จะเป็นเนื้อหาข้อมูลจริงของผู้ใช้บริการ หรือ แอปพลิเคชันที่ต้องการให้มีการรับส่งระหว่างเครื่องลูกข่าย GPRS กับแหล่งข้อมูลทั้งที่อยู่ภายในและภายนอกเครือข่าย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.21 โครงสร้างการจัดโปรโตคอลภายในเครือข่าย GPRS

เครือข่ายสถานีฐาน

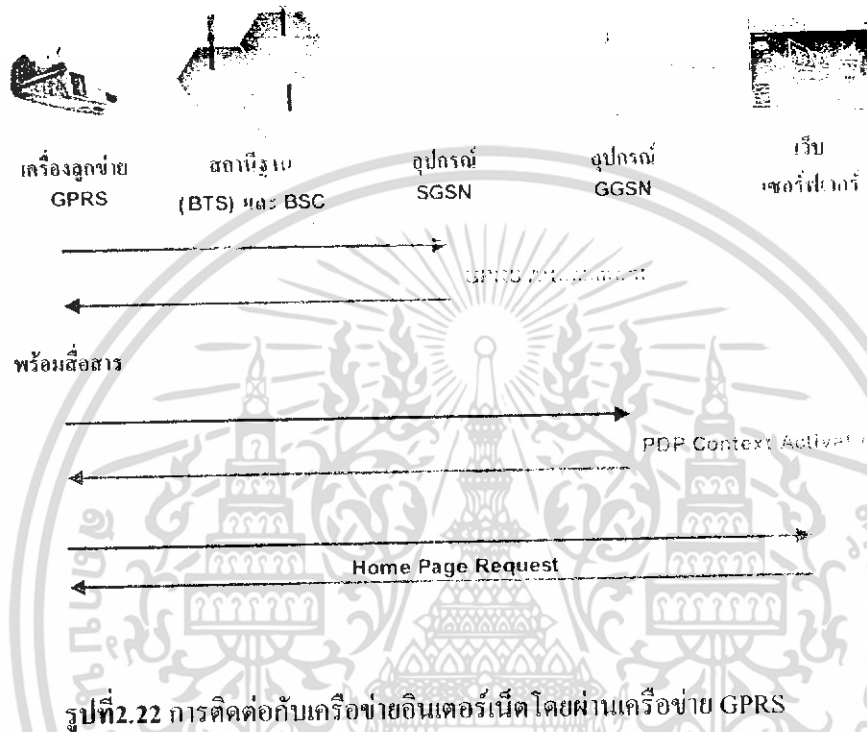
ทำหน้าที่เป็นทั้งตัวทวนสัญญาณ (Relay Function) สำหรับข้อมูลที่เครื่องลูกข่ายต้องการสื่อสารกับอุปกรณ์ SGSN ขึ้นไปโดยตรง โดยส่งผ่านข้อมูลที่ถูกรหัสด้วยโปรโตคอล LLC ต่อไปยังอุปกรณ์ SGSN และยังเป็นอุปกรณ์ปลายทางสำหรับควบคุมการทำงานเกี่ยวกับการจัดสรรช่องสื่อสารทางด้านความถี่วิทยุให้กับเครื่องลูกข่าย GPRS โดยผ่านการทำงานของโปรโตคอล GSM RF, MAC และ RLC ในขณะเดียวกันเครือข่ายสถานีฐาน (ซึ่งประกอบด้วยสถานีฐานและอุปกรณ์ BSC) ยังต้องมีการส่งสัญญาณควบคุมเพื่อติดต่อสื่อสารโดยตรงระหว่างเครือข่ายสถานีฐานเองกับอุปกรณ์ SGSN ซึ่งจำเป็นต้องมีการออกแบบโปรโตคอลอีกชุดหนึ่งเพื่อรองรับการติดต่อสื่อสารดังกล่าว รวมถึงเป็นรากฐานในการรับส่งข้อมูล LLC ที่ถูกส่งผ่านไปยังอุปกรณ์ SGSN ไปพร้อมๆกัน โปรโตคอลในจุดเชื่อมต่อ Gb ดังกล่าวประกอบไปด้วย ชั้นล่างสุดคือ L1bis ซึ่งสนับสนุนการรับส่งข้อมูลผ่านเครือข่ายสื่อสารสารพัสดุแบบไม่ว่าจะเป็นวงจรสื่อสารแบบ TDM เช่น E1 ผ่านเครือข่าย ATM หรือผ่านเครือข่ายเฟรมรีเลย์ เป็นต้น โดยในระดับชั้นที่สองเป็นโปรโตคอลแบบ Frame Relay (FR) ทำหน้าที่แก้ไขความผิดพลาดของข้อมูลที่อาจเกิดขึ้นบนจุดเชื่อมต่อ Gb สำหรับเนื้อหาของสัญญาณควบคุมระหว่างเครือข่ายสถานีฐานกับอุปกรณ์ SGSN จะถูกบรรจุอยู่ในโปรโตคอล BSSGP (Base Station Subsystem GPRS Protocol) ซึ่งเป็นช่องทางในการบรรจุข้อมูล LLC ระหว่างเครื่องลูกข่าย GPRS กับอุปกรณ์ SGSN ด้วยในเวลาเดียวกัน

มาตรฐานการรักษาความปลอดภัยกับเครือข่ายโทรศัพท์เคลื่อนที่ GPRS

กลไกการรักษาความปลอดภัยภายในเครือข่ายโทรศัพท์เคลื่อนที่ GPRS ประกอบไปด้วยมาตรฐานทางเทคนิคที่สำคัญ 3 ประเภท คือการตรวจยืนยันผู้ใช้บริการ (Authentication), การเข้ารหัสสัญญาณข้อมูลที่ถูกส่งผ่านทางคลื่นวิทยุ (Ciphering) และการตรวจสอบความถูกต้องของผู้ใช้บริการ (Authentication the User) โดย 2 มาตรฐานแรกเป็นกระบวนการปกติที่มีการกำหนดใช้งานในมาตรฐาน GSM จึงไม่ขอนำมากล่าวถึงอีก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

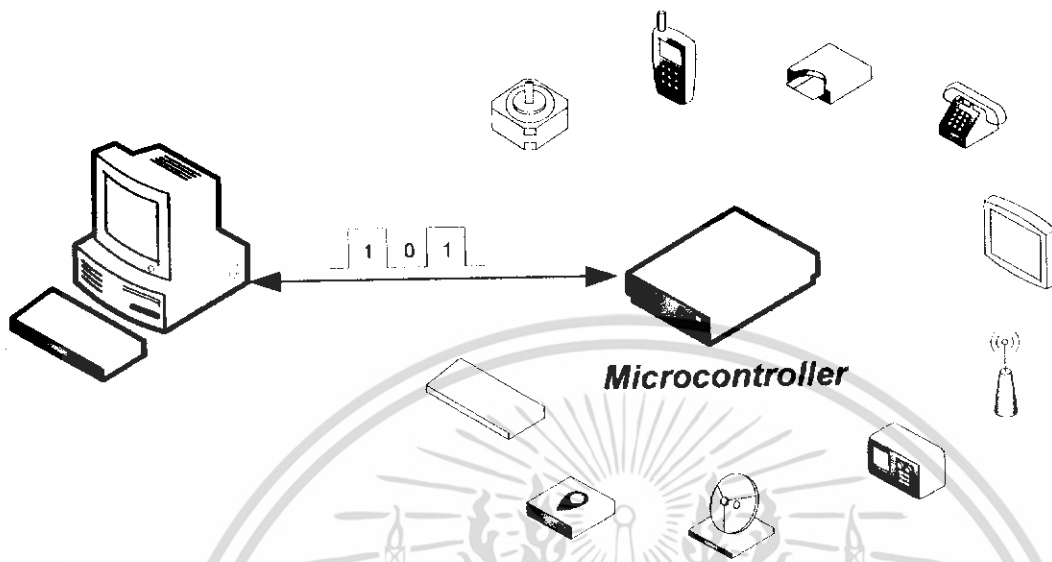
สิ่งที่เป็นเทคนิคใหม่ในการรักษาความปลอดภัยให้กับผู้ใช้บริการ โทรศัพท์เคลื่อนที่ GPRS ในการสื่อสารข้อมูลแสดงในรูปที่ 2.22 ซึ่งเป็นตัวอย่างของการใช้โทรศัพท์เคลื่อนที่ GPRS เข้าชมเว็บไซต์บนโลกของเครือข่ายอินเทอร์เน็ต ก่อนที่ผู้ใช้จะสามารถรับชมและติดต่อกับเว็บไซต์ได้ เครื่องลูกข่าย GPRS จะต้องดำเนินการกระบวนการ GPRS Attach เพื่อให้เครื่องลูกข่ายอยู่ในสถานะพร้อมสื่อสารก่อน จากนั้นจึงดำเนินการกระบวนการ PDP Context Activation เพื่อสร้างการเชื่อมโยงและติดต่อสื่อสารระหว่างเครื่องลูกข่ายโทรศัพท์เคลื่อนที่ GPRS กับเว็บไซต์ดังกล่าว



ในการติดต่อสื่อสารแลกเปลี่ยนข้อมูล อุปกรณ์ SGSN จะทำการจัดเรียงข้อมูลที่ใช้ในกระบวนการให้อยู่ในรูปของแพ็กเก็ต หรือที่นิยมเรียกกันว่า “แคปซูล” (Capsule) ก่อนที่จะจัดเส้นทางการส่งให้กับก้อนข้อมูลนั้นไปให้กับอุปกรณ์ GGSN โดยผ่านเส้นทางการสื่อสารแบบเอโมงค์ ภายใต้กลไกโปรโตคอลแบบ GTP ซึ่งในท้ายที่สุดอุปกรณ์ GGSN ก็จะลำเลียงข้อมูลนั้นส่งต่อไปยังเครือข่ายอินเทอร์เน็ต และในทิศทางกลับกันข้อมูลที่ได้รับจากเครือข่ายอินเทอร์เน็ตก็จะถูกควบคุมและจัดเส้นทางการส่งภายใต้ความร่วมมือระหว่างอุปกรณ์ GGSN และ SGSN

กระบวนการในการรับส่งข้อมูลต่าง ๆ เหล่านี้ถือเป็นมาตรการอันรัดกุม ที่ระบบเครือข่ายโทรศัพท์เคลื่อนที่ GPRS จัดการบริการทั้งเส้นทางการสื่อสาร การเข้ารหัสก่อนข้อมูล ให้กับผู้ใช้บริการโทรศัพท์เคลื่อนที่แต่ละราย ถือเป็นการสร้างความน่าเชื่อถือในแง่ของการรักษาความปลอดภัยให้กับผู้ใช้บริการโทรศัพท์เคลื่อนที่ GPRS นอกเหนือไปจากการรักษาความปลอดภัยในการติดต่อสื่อสารโดยกลไกหลักของเครือข่าย GSM ที่เป็นเสมือนกับรากฐานของเครือข่าย GPRS

ระบบไมโครคอนโทรลเลอร์



ระบบไมโครคอนโทรลเลอร์

ระบบไมโครคอนโทรลเลอร์ มีขนาดเล็ก พร้อมความสามารถในการควบคุมและติดต่ออุปกรณ์ต่างๆ ได้อย่างมากมาย เพียงแค่เราเรียนรู้และทำความเข้าใจมัน เราก็สามารถใช้ไมโครคอนโทรลเลอร์ควบคุม อุปกรณ์ต่างๆ ตามที่เราต้องการ

การพัฒนาะบบไมโครคอนโทรลเลอร์

การพัฒนาะบบไมโครคอนโทรลเลอร์นั้น จะประกอบด้วยองค์ประกอบที่สำคัญหลัก ๆ อยู่ 4 ส่วน ดังนี้

- Software เป็นโปรแกรมที่ใช้ในการเขียนโปรแกรม เพื่อควบคุมการทำงานของไมโคร ฯ
- Computer ที่เป็นเครื่องมือในการเขียน
- Project Board เป็นบอร์ดของระบบไมโครคอนโทรลเลอร์ เพื่อการควบคุมอุปกรณ์ต่าง ๆ
- Device อุปกรณ์ที่เราจะควบคุม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ที่มาของไมโครคอนโทรลเลอร์

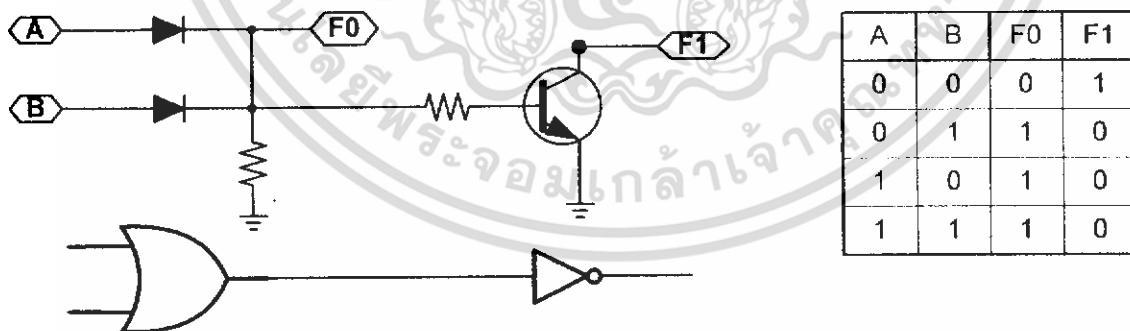
เราจะมาศึกษาว่ากว่าจะมาเป็นไมโครคอนโทรลเลอร์ ให้เราได้ใช้นั้นมันมีความเป็นมาอย่างไร เพื่อที่จะทำให้เราเข้าใจถึงการทำงานของไมโครคอนโทรลเลอร์ ได้ดียิ่งขึ้น เราเริ่มต้นที่การเปลี่ยนแปลงครั้งยิ่งใหญ่ของมนุษยชาติก็แล้วกัน นั่นคือการสร้าง สารกึ่งตัวนำเป็นผลสำเร็จ ซึ่งอุปกรณ์ตัวแรกของสารกึ่งตัวนำก็คือ ไดโอด (Diode) นั่นเอง



ไดโอดนั้นเป็นอุปกรณ์จากที่เกิดสารกึ่งตัวนำ 2 ชนิด คือสาร P และสาร N มาเชื่อมต่อกัน ซึ่งคุณสมบัติของสารกึ่งตัวนำนั้น สามารถนำกระแสได้ ในกรณีที่เราย้ายแรงดันให้กับสาร กึ่งตัวนำได้ถูกขั้ว และจะไม่นำกระแสเมื่อเราย้ายไฟกลับขั้วนั่นเอง จากนั้นก็ได้มีการพัฒนา จากไดโอดมาเป็นอุปกรณ์ตัวใหม่ที่เรียกว่าทรานซิสเตอร์ ซึ่งสามารถต่อได้ 2 แบบตามรูปข้างล่าง

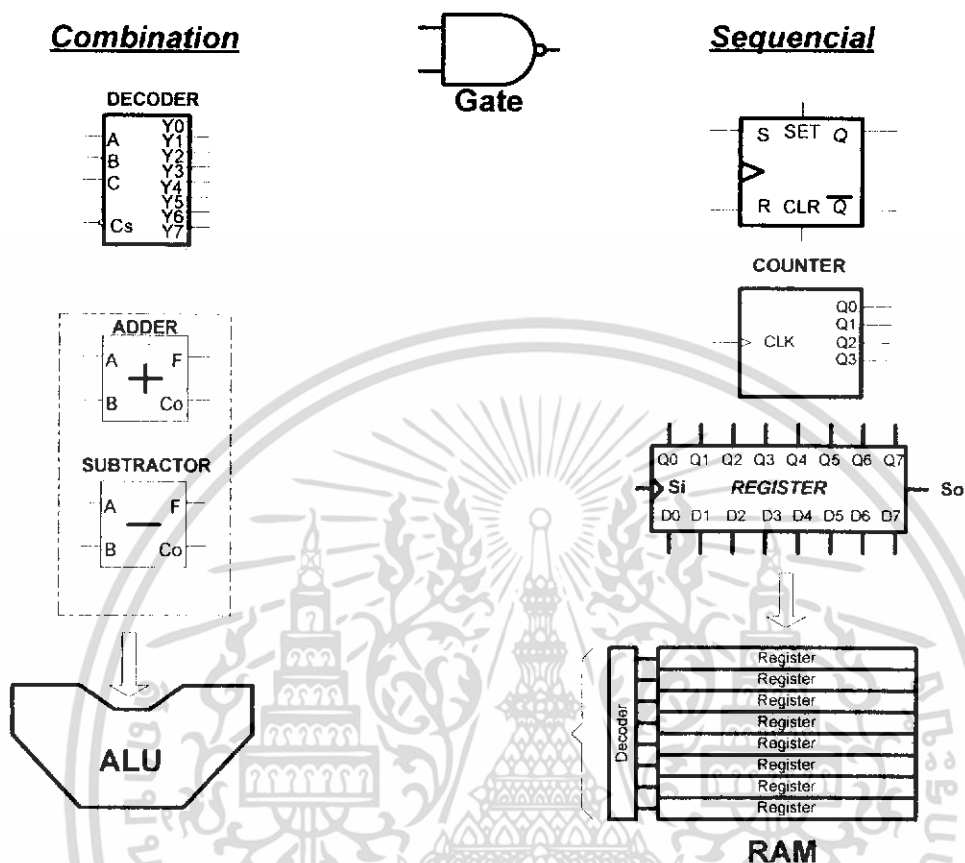


ด้วยลักษณะการการเชื่อมต่อของสารกึ่งตัวนำ ทรานซิสเตอร์ จึงมี 2 ชนิดคือแบบ PNP และ NPN หน้าทีโดยหลักของทรานซิสเตอร์คือ ขยายสัญญาณและเปิดปิดวงจร ด้วยลักษณะ การทำงานของไดโอดและทรานซิสเตอร์นี้ เราจึงสามารถนำเอาอุปกรณ์ทั้ง 2 ตัว มาเชื่อมต่อ เป็นวงจรทางด้านดิจิตอลได้ดังนี้



วงจรข้างบนเป็นการนำเอาอุปกรณ์ต่าง ๆ มาต่อกันเป็น วงจรเกต (Gate) ตามรูปด้าน บน จัดเป็นวงจรแม่งออกเป็น 2 ส่วน โดยส่วนแรกประกอบด้วย ไดโอดและความต้านทาน เชื่อมต่อกันเป็นวงจร OR Gate และส่วนที่ 2 ประกอบด้วยทรานซิสเตอร์และความต้านทาน เชื่อมต่อกันเป็น วงจร NOT Gate เมื่อนำ 2 วงจรนี้มาต่อกัน ก็จะได้ NOR Gate นั่นเอง เป็นการแสดงให้เห็นถึง ที่มา ในการสร้างวงจรเกตต่าง ๆ มากมาย ไม่ว่าจะเป็น AND Gate , NAND Gate , EX-OR Gate หลังจากที่เรานำเอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สามารถสร้างวงจรเกิดต่าง ๆ ได้แล้ว ก็ได้มีการพัฒนาโดยเอาวงจรเกิด มาเชื่อมกันเป็นวงจรต่าง ๆ หรือวงจรดิจิทัล ซึ่งสามารถแบ่งลักษณะการต่อวงจรออกเป็น 2 แบบด้วยกัน



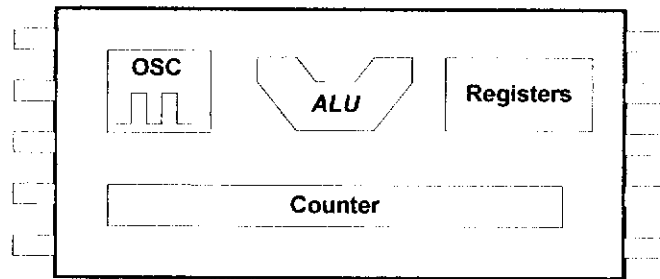
1. วงจร Combination

คือวงจรที่ประกอบด้วยเกตต่าง ๆ ต่อเข้าด้วยกัน โดยไม่จำเป็นต้องมีสัญญาณพัลส์มากระตุ้น วงจรประเภทนี้ได้แก่ วงจร Decoder , Adder , Subtractor , Multiplex , Demultiplex , And , Or วงจรเหล่านี้ถ้านำมารวมกันแล้วก็จะกลายเป็น ALU (Arithmetic Logic Unit) ที่อยู่ใน CPU ทั่วไป ทำหน้าที่คำนวณทางด้านคณิตศาสตร์และลอจิก

2. วงจร Sequential

คือวงจรที่ประกอบจากพวก Flip - Flop ต่อเข้าด้วยกัน ซึ่งต้องมีสัญญาณมากระตุ้น (Synchronous) หรือมีสัญญาณนาฬิกา เช่นวงจร Counter, Register, Latch ถ้าเรานำวงจร Register เหล่านี้มาต่อพ่วงกันหลายๆ ตัว และมีวงจร Decoder เป็นตัวเลือก Register มันก็จะ กลายเป็น หน่วยความจำ (RAM) เพราะฉะนั้นรีจิสเตอร์ก็เป็นส่วนหนึ่งของหน่วยความจำ (RAM) นั่นเอง ถ้าเราเอาวงจรทั้งหมดนี้มาจัดเรียง เชื่อมต่อซึ่งกันและกัน และเพิ่มเติมวงจรสร้าง สัญญาณนาฬิกา เพื่อ กำหนดจังหวะการทำงาน เราก็จะได้อุปกรณ์ตัวใหม่ที่เรียกว่า CPU

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



หน่วยประมวลผลกลาง CPU (Central Processing Unit)

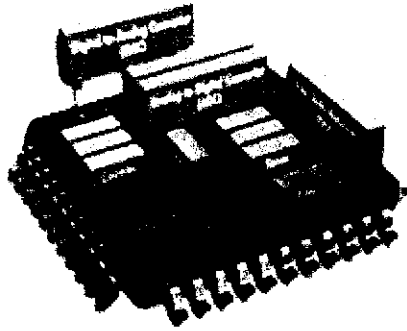
และนี่ก็คือที่มาของ CPU ที่เราเรียนรู้กันมานี้เอง ซึ่งล้วนเกิดจากวงจรลอจิกต่าง ๆ มารวมกัน ให้สามารถทำงานได้หลาย ๆ อย่าง นอกจากความสามารถของ Gate ที่กล่าวมาแล้ว เรายังสามารถดัดแปลงพัฒนาเป็นวงจรต่าง ๆ มากมาย ตามวัตถุประสงค์ของผู้ออกแบบ ไม่ว่าจะเป็น

- วงจรพอร์ตขนาน (Parallel Port)
- วงจรพอร์ตอนุกรม (Serial Port)
- วงจร Timer/Counter
- วงจร A/D (Analog to Digital)
- ฯลฯ

CPU นั้นถ้าจะเปรียบกับร่างกายของเรา ก็เปรียบเสมือนสมองของเรานั้นเอง ที่มีความสามารถในการคำนวณและจดจำข้อมูล แต่ไม่สามารถที่จะควบคุมอะไรได้ เปรียบเสมือนว่าเรามีแต่สมองแต่ไม่มีแขนขา, ตา, หู, จมูก, ปาก ดังนั้น CPU จึงไม่สามารถทำงานเพียงลำพังตัวเดียวได้ จะต้องมียังวงจรหรือวงจรรวมสนับสนุน (Chip Support) มาต่อเข้ากับ CPU ทำหน้าที่เป็น อินพุต, เอาท์พุท ให้กับ CPU จึงสามารถที่จะควบคุมอุปกรณ์ต่าง ๆ ได้

Chip Support ที่เชื่อมต่อกับ CPU สามารถเลือกที่จะมาเชื่อมต่อได้ตามต้องการ เพราะฉะนั้น การที่เราจะสร้างวงจรเพื่อมาควบคุมอะไรสักอย่างนั้น ในบอร์ดควบคุมนั้นจะมีเพียง CPU อย่างเดียวไม่ได้ จะต้องมีการต่อ EPROM ภายนอกซึ่งทำหน้าที่เก็บคำสั่งต่าง ๆ ที่เราเขียนโปรแกรมไว้ และต้องต่อพอร์ตขนานเพื่อควบคุมอุปกรณ์ภายนอก หรือถ้ามีการใช้งานที่เกี่ยวข้องเวลาก็อาจจะต้องใช้วงจร Timer/Counter เข้ามาด้วยจะเห็นว่าอุปกรณ์ที่ใช้นั้น จะเริ่มมากขึ้น ทำให้บอร์ดควบคุมนั้นใหญ่ขึ้นเป็น เรตามตัวไปด้วย ด้วยเหตุนี้จึงทำให้ต้นทุนที่ใช้ในการผลิต นั้นสูงขึ้นไปด้วย อีกทั้งยังมีขนาดใหญ่ไม่สะดวกต่อการพกพา นี้ยังไม่รวมถึงพลังงานที่ต้องจ่ายให้กับวงจร ทำให้ต้องใช้แบตเตอรี่ที่มีขนาดใหญ่ขึ้น หรือไม่ก็ใช้งานได้ไม่นานแบตเตอรี่ก็หมดซะแล้ว จึงได้มีการคิดค้นกรออกแบบ Chip ใหม่โดยทำการรวมเอาวงจรต่าง ๆ ที่กล่าวมารวมกับ CPU ไว้ใน Chip ตัวเดียวกัน เพื่อลดขนาด ต้นทุน พลังงาน แล้วตั้งชื่อใหม่ว่า MCU

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



การออกแบบวงจรพื้นฐานของ MCS-51

การทำงานของ ขาต่างๆ ของ Mcs-51 และในขณะเดียวกัน เราก็จะทำการศึกษาการออกแบบวงจร ใช้งานให้กับ MCS - 51 เพื่อให้ MCS - 51 พร้อม ที่จะทำงานตามคำสั่ง ที่เราโปรแกรมได้

P1.0	□ 1	40	□ VCC
P1.1	□ 2	39	□ P0.0 (AD0)
P1.2	□ 3	38	□ P0.1 (AD1)
P1.3	□ 4	37	□ P0.2 (AD2)
P1.4	□ 5	36	□ P0.3 (AD3)
P1.5	□ 6	35	□ P0.4 (AD4)
P1.6	□ 7	34	□ P0.5 (AD5)
P1.7	□ 8	33	□ P0.6 (AD6)
RST	□ 9	32	□ P0.7 (AD7)
(RXD) P3.0	□ 10	31	□ EA/VPP
(TXD) P3.1	□ 11	30	□ ALE/PROG
(INT0) P3.2	□ 12	29	□ PSEN
(INT1) P3.3	□ 13	28	□ P2.7 (A15)
(T0) P3.4	□ 14	27	□ P2.6 (A14)
(T1) P3.5	□ 15	26	□ P2.5 (A13)
(WR) P3.6	□ 16	25	□ P2.4 (A12)
(RD) P3.7	□ 17	24	□ P2.3 (A11)
XTAL2	□ 18	23	□ P2.2 (A10)
XTAL1	□ 19	22	□ P2.1 (A9)
GND	□ 20	21	□ P2.0 (A8)

รูปที่ 2.23 แสดงรูปร่างและขาต่างของ MCS-51

หน้าที่ของแต่ละขา

P0.0 - P0.7 (ขา 39 - 32) เป็น Port ใช้งานทั่วไป และพอร์ท Data และ พอร์ท Address ไบท์ต่ำ

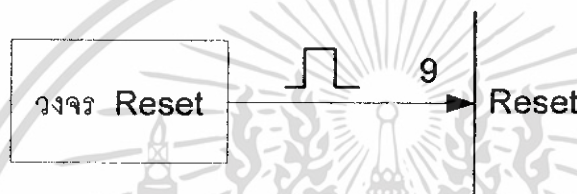
P1.0 - P1.7 (ขา 1 - 8) เป็น Port ใช้งานทั่วไป

P2.0 - P2.7 (ขา 21 - 28) เป็น Port ใช้งานทั่วไป และพอร์ท Address ไบท์สูง

P3.0 - 3.7 (ขา 10 - 17) เป็นพอร์ทใช้งานทั่วไป นอกจากนั้นยังสามารถทำหน้าที่อื่น ๆ ได้ดังนี้

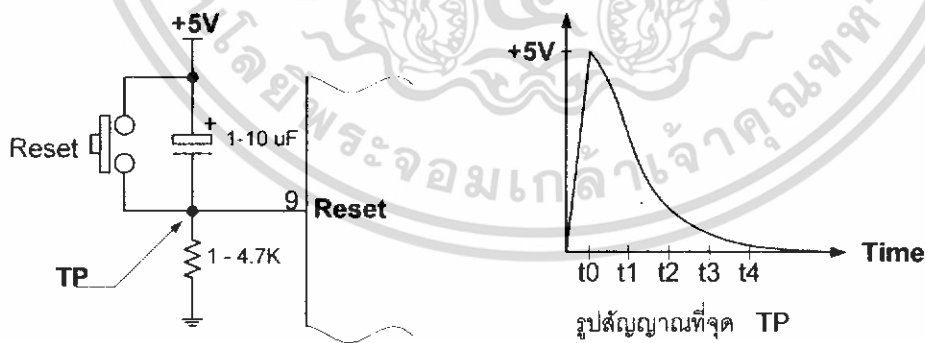
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

P3.0 , RXD	สามารถเป็นขารับสัญญาณ Serial Port
P3.1 , TXD	สามารถเป็นขาส่งสัญญาณ Serial Port
P3.2 , INT0	สามารถเป็นขารับสัญญาณ Interrupt หมายเลข 0
P3.3 , INT1	สามารถเป็นขารับสัญญาณ Interrupt หมายเลข 1
P3.4 , T0	สามารถเป็นขาที่รับสัญญาณพัลส์ หมายเลข 0 เพื่อเข้าวงจร Counter
P3.5 , T1	สามารถเป็นขาที่รับสัญญาณพัลส์ หมายเลข 1 เพื่อเข้าวงจร Counter
P3.6 , WR	สามารถเป็นขาสัญญาณเขียนข้อมูลไปยังอุปกรณ์ภายนอก
P3.7 , RD	สามารถเป็นขาสัญญาณอ่านข้อมูลจากอุปกรณ์ภายนอก



Reset (ขา 9)

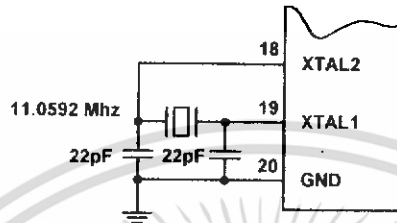
ขานี้มีไว้สำหรับรีเซ็ตการทำงานของ MCS-51 ให้เริ่มต้นทำงานใหม่ ขานี้ทำงานที่ลอจิก "1" สังกัดว่าไม่มีเครื่องหมาย บนชื่อของขา ถ้ามีแสดงว่าทำงานที่ลอจิก "0" สิ่งที่เราต้อง ออกแบบวงจรก็คือ ต้องสร้างสัญญาณพัลส์บวกที่ขา RST ในช่วงเวลาที่เราย้ายไฟให้กับ MCS-51 ตอนเปิดเครื่อง เพราะฉะนั้นเมื่อเราเปิดเครื่อง ไมโครคอนโทรลเลอร์จะรีเซ็ตตัวเองโดยอัตโนมัติ วงจรนี้เราสามารถสร้างโดยใช้ R และ C เพื่อสร้างสัญญาณพัลส์บวก ได้ดังวงจรข้างล่างนี้



จากวงจรข้างบน เราจะใช้วงจร RC สร้างสัญญาณ Reset ให้กับ MCS-51 เรามาดูกันว่ามันทำงานกันอย่างไร โดยคุณสมบัติของ C คือเป็นตัวเก็บประจุไฟฟ้า เมื่อเริ่มจ่ายไฟเข้าไปในวงจร C จะเริ่มชาร์จตัวเอง ตอนนี่ C เปรียบเสมือนช็อต แรงดันตกคร่อมที่ C (V_C) = 0V แรงดันที่ตกคร่อมความต้านทาน (V_R) = 5V จากนั้นเมื่อเวลาผ่านไป C ก็เริ่มชาร์จไฟมากขึ้น ทำให้มีแรงดัน (V_C) ตกคร่อม ที่ตัวมันมากขึ้น แรงดันที่ความต้านทาน (V_R) ก็จะน้อยลงจนในที่สุด C ชาร์จไฟจนเต็มแรงดัน $V_C = 5V$ และแรงดันที่ความต้านทาน $V_R = 0V$ ตามรูปทางขวามือ เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เวลาของสัญญาณพัลส์หรือความกว้างของสัญญาณนั้นขึ้นอยู่กับค่า R , C ค่าที่เหมาะสมของ R ประมาณ 1K - 10K , C ประมาณ 1 uF - 10 uF

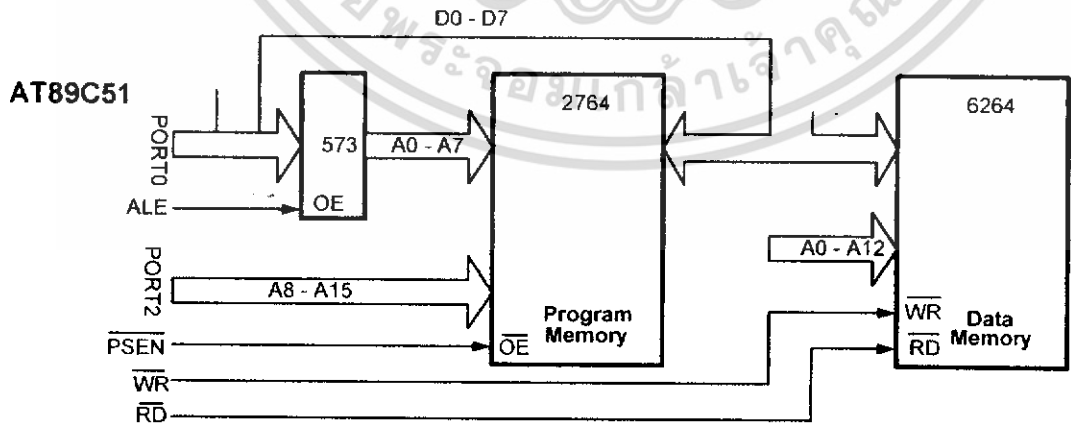
$$\begin{aligned}
 \text{เราสามารถหาค่าเวลาที่ได้จากวงจร RC ได้จากสูตร} &= 4 \times RC \\
 \text{สมมติ ค่า } R = 1K , C = 1 \mu F &= 4 \times 10000 \times 0.000001 \\
 &= 0.004 \text{ S} \\
 &= 4 \text{ mS}
 \end{aligned}$$



XTAL2,1 (ขา 18,19)

ไว้สำหรับต่อ X - TAL เพื่อสร้างสัญญาณนาฬิกาให้กับ MCS - 51 เพราะการทำงานของไมโครคอนโทรลเลอร์ ไม่ว่าเบอร์ก็ต้องมีวงจรสร้างสัญญาณนาฬิกา (Oscillator) เพื่อ กำหนดจังหวะการทำงานของไมโครคอนโทรลเลอร์ วงจรสร้างสัญญาณนาฬิกานั้นจึงต้องการ X-Tal เพื่อสร้างสัญญาณ ซึ่งค่าความถี่ X - Tal ที่เราใช้คือ 11.0592 Mhz การที่เราใช้ค่านี เพราะใน MCS - 51 นั้นมีวงจร Serial Port ซึ่งทำหน้าที่ในการส่งข้อมูลอนุกรม ซึ่งจะต้องมีอัตรา การส่งข้อมูลหรือความถี่ในการส่งข้อมูล เป็นไปตามมาตรฐานของ Serial Port ซึ่ง Serial Port ที่เรารู้จักก็มีอยู่ในคอมพิวเตอร์ทั่วไปที่พวกเราชอบเรียกว่า Com1, Com2 นั้นเอง ความถี่ในการส่งข้อมูลนั้น จะได้จากอัตรา ความถี่จาก X-Tal ซึ่งค่าความถี่ X-Tal คำนี้นี้เมื่อหารความถี่แล้ว เราจะได้ความถี่ที่ตรงตามความถี่มาตรฐานพอดี เราจึงเลือกใช้ค่านี

สำหรับ C ทั้ง 2 ตัวนี้ทำหน้าที่ เอาสัญญาณรบกวนที่เกิดจากวงจร OSC ลงกราวด์ ค่าที่เราใช้นั้นจะมีค่าประมาณ 6 - 30 p ซึ่งเราได้เลือกใช้ค่า 22 pF



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

PSEN (Program Status Enable) (ขา 29)

ขานี้จะทำงานที่ลอจิก "0" (Active low) ขานี้ Active เมื่อ MCS - 51 ต้องการติดต่อกับ Program Memory ซึ่ง Program Memory จะเป็นหน่วยความจำที่ทำหน้าที่ เก็บคำสั่งที่เรา เขียนนั่นเอง หน่วยความจำส่วนนี้ส่วนใหญ่จะเป็น EPROM ที่ต่ออยู่ภายนอกหรือ FLASH ROM ที่อยู่ภายในบอร์ด ดังนั้นขานี้จะต้องเข้ากับขา OE ของ EPROM ในกรณีที่เราไม่ได้ต่อ EPROM ภายนอก เราก็ปล่อยให้ลอย ๆ ไว้ อย่างนั้น

ALE (Address Latch Enable) (ขา 30)

เนื่องจาก MCS-51 มีความสามารถในการอ้างหน่วยความจำและอุปกรณ์ภายนอก จึงใช้ พอร์ต 2 เป็น Address ไบท์สูง (A8 - A15) และให้พอร์ต 0 เป็น Data Bus และ Address Bus ไบท์ต่ำ เพื่อเป็นการประหยัดพอร์ต ดังนั้นเมื่อพอร์ต 0 มีถึงสองหน้าที่ จึงต้องมีสัญญาณมา บอกให้รู้ว่าตอนนี้ข้อมูลที่อยู่บนพอร์ต 0 เป็น Address หรือ Data จึงใช้ขา ALE เป็นตัวบอก ว่าที่พอร์ต 0 เป็น Address ขา ALE จะเป็น "1" ด้วยเหตุนี้ที่พอร์ต 0 จึงมีไอซีที่ทำหน้าที่ เก็บข้อมูล (Latch) เช่นเบอร์ 74HC373, 74HC573 ในกรณีที่ต่อหน่วยความจำภายนอก โดย ขา ALE จะต่อเข้ากับขา OE เพื่อส่งสัญญาณให้กับไอซี เอาข้อมูลที่พอร์ต 0 ซึ่งเป็น Address ไบท์ต่ำ (A0 - A7) ออกไปที่เอาต์พุตของไอซีตัวนี้ ดังนั้นเอาต์พุตของไอซี Latch จึงเป็น Address (A0-A7) นั่นเอง

EA (Enable Access) (ขา 31)

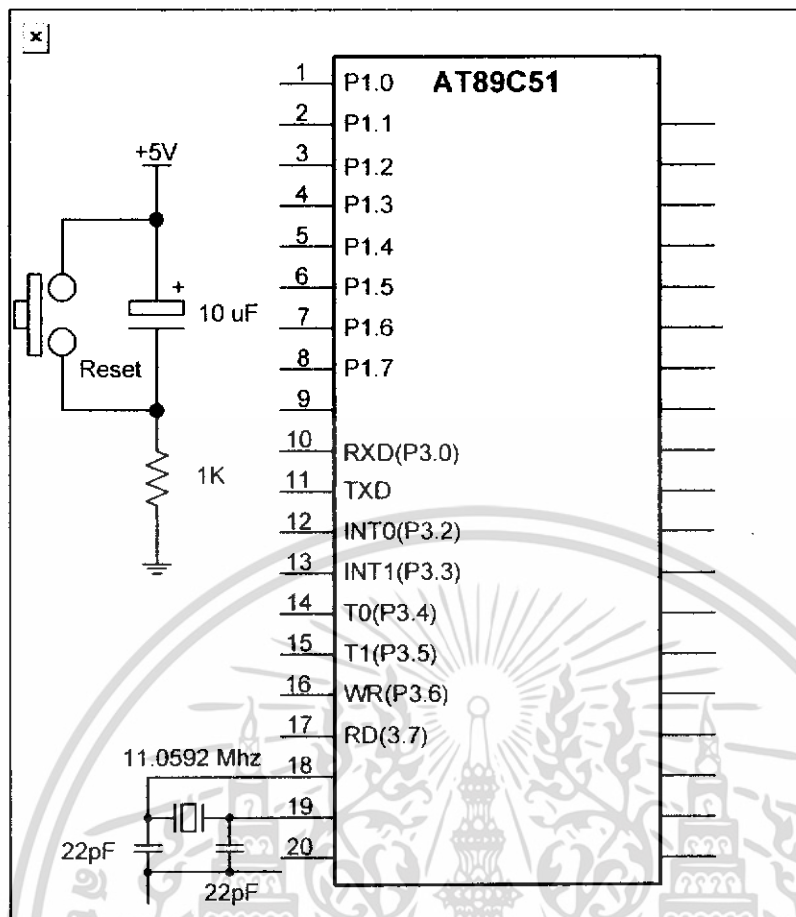
MCS-51 สามารถติดต่อกับหน่วยความจำได้ 3 แบบคือ

- หน่วยความจำภายใน (Internal Memory)
- หน่วยความจำข้อมูลภายนอก (Data Memory) ส่วนที่ทำหน้าที่เก็บข้อมูลจะเป็น

หน่วยความจำประเภท RAM

- หน่วยความจำโปรแกรมภายนอก (Program Memory) สามารถอ้างข้อมูลได้ 64 Kbyte เป็นส่วนที่ทำหน้าที่เก็บคำสั่งที่เราเขียน ส่วนใหญ่ถูกเก็บไว้ที่ EPROM ภายนอก แต่ปัจจุบันนี้ ได้มีการพัฒนาจนสามารถนำเอา Program Memory อีกแบบหนึ่งซึ่งเรียกว่า Flash Rom เข้าไปในไมโครคอนโทรลเลอร์ได้แล้ว Flash ROM นั้นมีข้อดีกว่า EPROM เพราะสามารถลบได้ด้วย ไฟฟ้า และเขียนได้เร็วกว่าด้วย ด้วยเหตุนี้จึงต้องมีขา EA เป็นตัวเลือกว่าจะใช้ Program Memory ภายนอกหรือภายใน ซึ่งในที่นี้เราใช้เบอร์ AT89C51 ซึ่งมี Program Memory ภายใน ที่เป็น Flash ROM เราจึงให้ขา EA ต่อ VCC ถ้าเราใช้ Program Memory ภายนอกขา EA จะต่อลงกราวด์ เพราะฉะนั้นเราจึงได้วงจรใช้งานจริงของ MCS - 51 ดังรูป

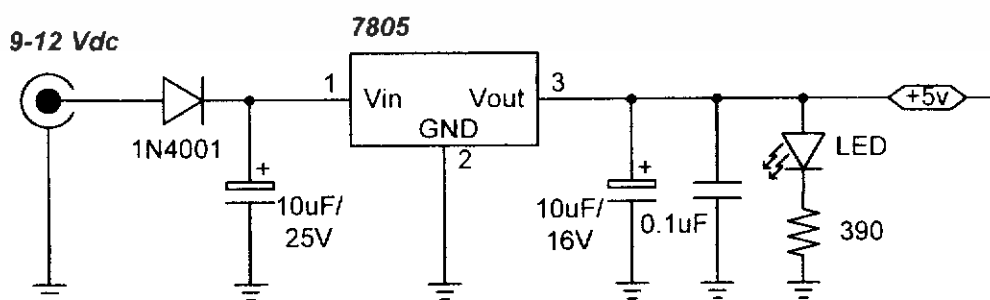
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



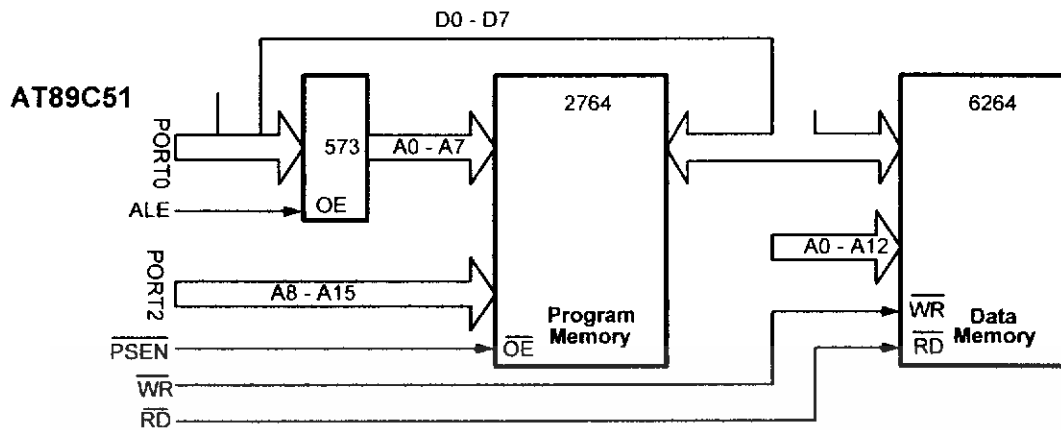
รูป 2.24 แสดงวงจรการใช้งานของ MCS-51

วงจร Regulator

เป็นวงจรที่จำกัดแรงดันให้เหลือเพียง 5 V เพื่อที่จะเป็นแหล่งจ่ายไฟให้กับ วงจรไมโครคอนโทรลเลอร์ของเรา ซึ่งตัวที่ทำหน้าที่นี้คือไอซี 7805



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



หน่วยความจำและรีจิสเตอร์

หน่วยความจำโปรแกรมภายนอก (Program Memory)

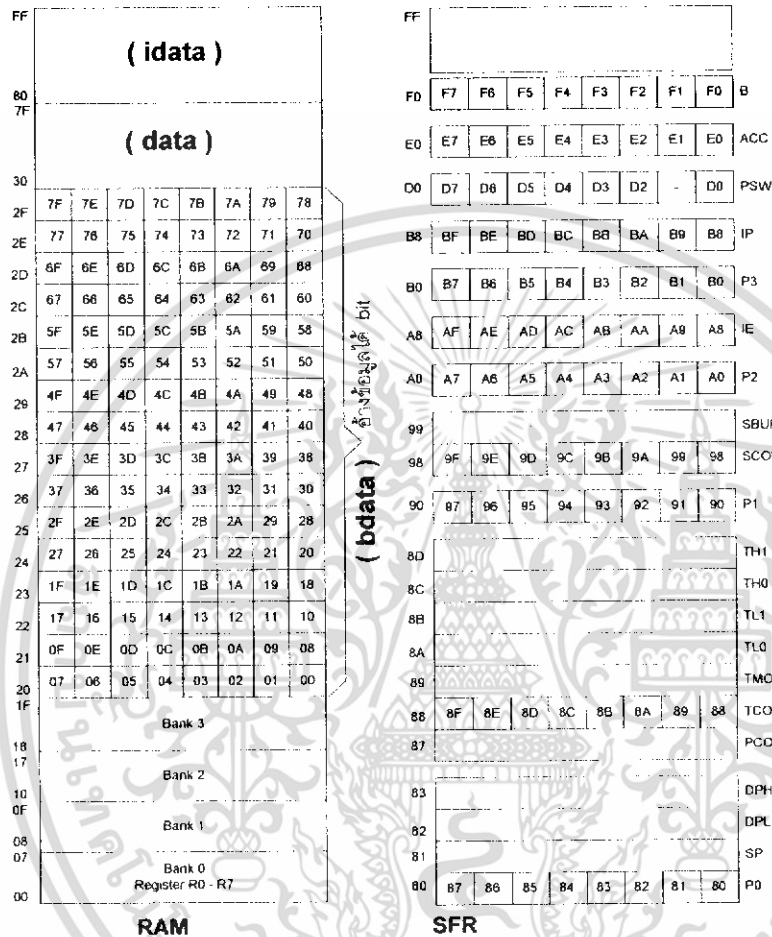
หน่วยความจำชนิดทำหน้าที่เก็บคำสั่งของผู้เขียน โปรแกรมส่วนใหญ่จะใช้ EPROM เป็นตัวเก็บ โดยมีสัญญาณ PSEN เป็นขาสัญญาณเพื่อติดต่อกับ EPROM สามารถอ้างหน่วย ความจำ ได้ 64 Kbyte ชื่อของ EPROM นั้นส่วนใหญ่จะขึ้นด้วย 27 แล้วตามด้วยขนาดของหน่วย ความจำเช่น 2764 หมายเลข 64 คือขนาดของหน่วยความจำ โดยเอาค่า $64/8 = 8$ Kbyte เราก็ จะได้ขนาดของ EPROM ตัวนี้ จากนั้นได้มีการ พัฒนา MCS-51 โดยเอา EPROM ใส่เข้าไปในตัว MCS - 51 แล้วตั้งชื่อ เบอร์ใหม่ว่า 8751 และก็มีการพัฒนาขึ้นไปอีกคือการนำเอา Flash ROM ซึ่งสามารถลบและเขียนด้วย ไฟฟ้าใส่เข้าไปใน MCS-51 แล้วตั้งชื่อใหม่ว่า 8951 จนเป็นที่นิยม ในปัจจุบัน คำสั่งที่อ้างข้อมูลใน Program Memory ในภาษา C คือ code

หน่วยความจำข้อมูลภายนอก (Data Memory)

หน่วยความจำชนิดนี้ทำหน้าที่ เก็บข้อมูลต่าง ๆ ที่เราต้องการ ซึ่งก็คือ RAM ที่ต่ออยู่ ภายนอกสามารถเข้าถึงข้อมูลโดยใช้ขา RD , WR ในการเขียนและอ่าน Data จาก RAM MCS-51 สามารถอ้างหน่วยความจำข้อมูลภายนอก ได้ 64 Kbyte เช่นกัน ส่วนใหญ่แล้วหน่วยความจำ นี้จะถู กจัด Address ให้ถัดจาก Program Memory (EPROM) RAM ที่ใช้ส่วนใหญ่จะเป็น Static RAM เบอร์จะ ขึ้นด้วย หมายเลข 61,62,64,68 แล้วตามด้วยขนาดของหน่วยความจำเช่น 6264 หมายเลข 64 คือขนาด ของหน่วยความจำ โดยเอาค่า $64/8 = 8$ Kbyte เราก็จะได้ขนาดของ RAM คำสั่งที่ใช้ในการอ้างข้อมูล ของ Data Memory คือ xdata

หน่วยความจำภายใน Internal RAM

เป็นหน่วยความจำภายใน ซึ่งเป็น RAM ในหน่วยความจำนั้น ก็จะประกอบด้วย Register ซึ่งจะมี Address ประจำตัวทำหน้าที่ใช้งานแตกต่างกันไปตามการใช้งาน รูปข้างล่างนี้แสดงรีจิสเตอร์ของหน่วยความจำภายใน

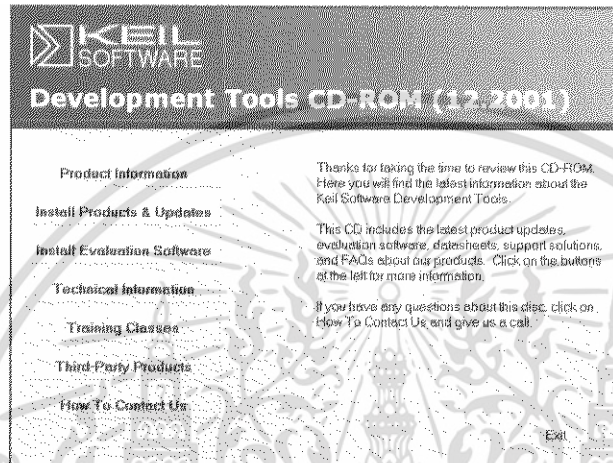


เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

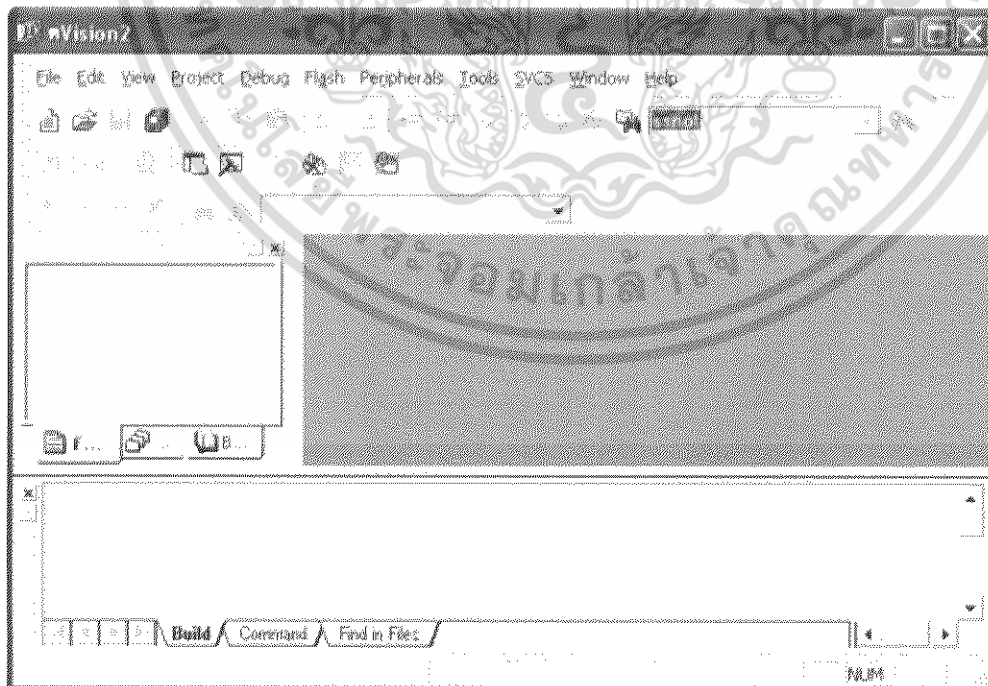
การใช้โปรแกรม Keil C51 เบื้องต้น

Keil C51

เป็นโปรแกรมที่ใช้ในการเขียนโปรแกรมภาษา C สำหรับไมโครคอนโทรลเลอร์ ต่าง ๆ ของแต่ละบริษัท ซึ่งสามารถทำงานบนระบบปฏิบัติการ Windows 95 ขึ้นไปได้

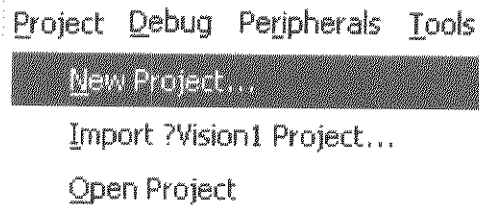


ทำการ Run โปรแกรมโดยการเลือกที่  Keil uVision2 หน้าจอจะแสดงโปรแกรมดังรูปข้างล่างนี้

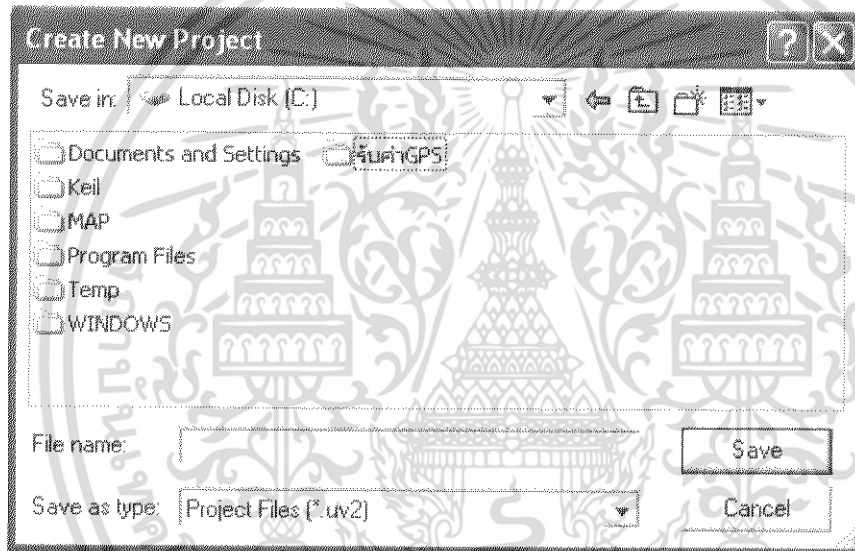


เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

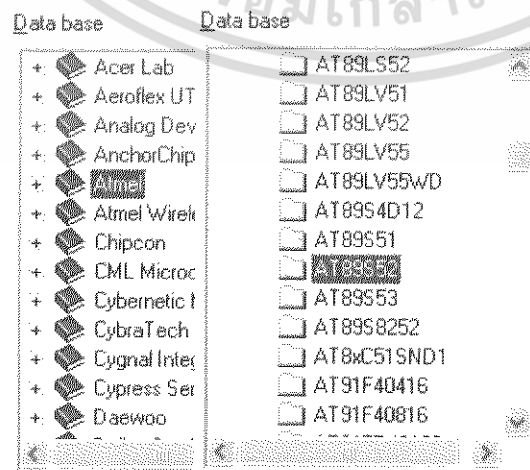
สร้าง Project ใหม่ โดยเข้าไปที่เมนู Project / New Project ตามรูป



หน้าจอจะแสดงดังรูปข้างล่าง ให้เราทำการสร้างชื่อโฟลเดอร์ใหม่ในไดรฟ์ C โดยให้ตั้งชื่อว่า รับค่า GPS จากนั้นให้เราตั้งชื่อ Project เป็น รับค่าGPS แล้วคลิกปุ่ม Save ตามรูปข้างล่าง

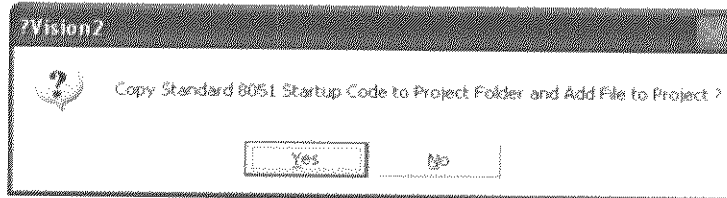


จากนั้นให้เราทำการเลือกบริษัทและเบอร์ไมโครคอนโทรลเลอร์ ที่เราเลือกใช้ตามรูปข้างล่างแล้วกดปุ่ม OK

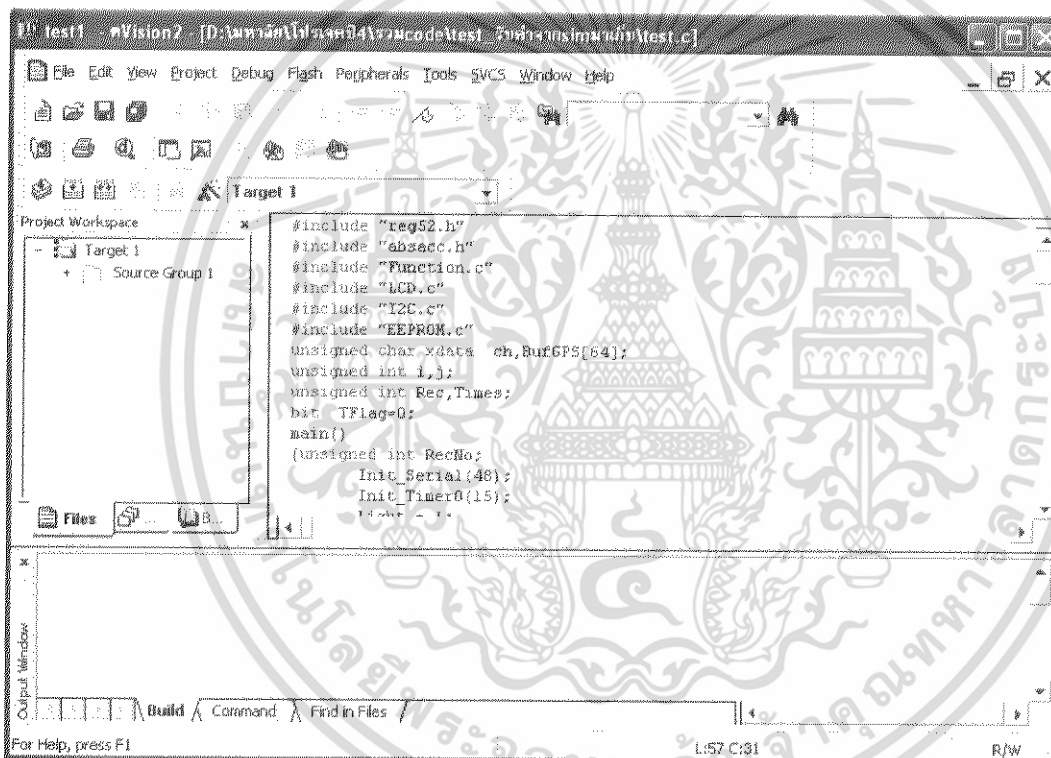


เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

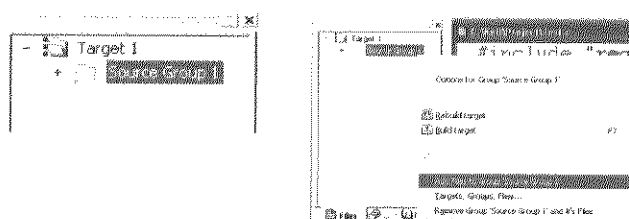
โปรแกรมจะขึ้นหน้าต่างขึ้นมาถามว่า ต้องการ Copy ไฟล์ Startup หรือไม่ให้เราคลิกปุ่ม Yes



สร้างไฟล์โปรแกรมใหม่โดยเลือกเมนูที่ File / New จากนั้นจะมีหน้าต่างสำหรับเขียนโปรแกรม ก็ให้เราเขียนโปรแกรมภาษา C สักโปรแกรมตามรูปข้างล่าง แล้วทำการบันทึกไฟล์โดยไปที่เมนู File / Save แล้วตั้งชื่อไฟล์ว่า test1 **อย่าลืมต้องใส่นามสกุล .C ด้วย**

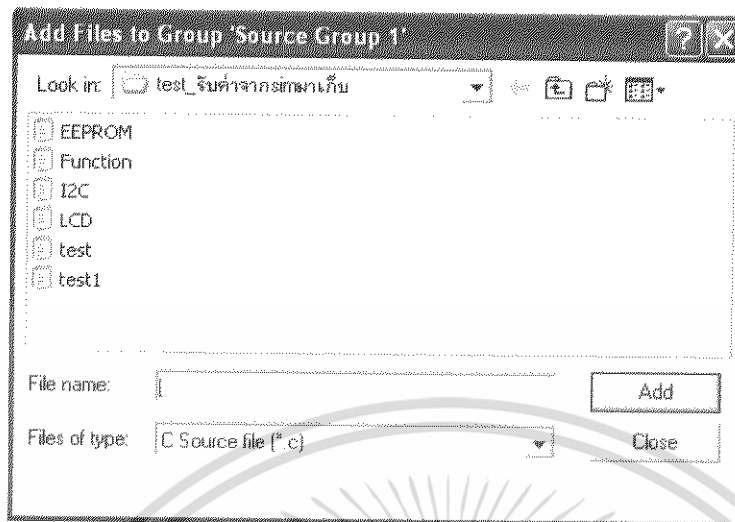


ทำการเพิ่มไฟล์ test.c เข้าไปใน Project ของเราโดยการเลื่อนเมาส์เครื่องหมาย + หน้า Target 1 จะแสดงชื่อ Source Group1 แล้วคลิกขวาที่ชื่อ Source Group 1 จากนั้นเลือก Add files to Group 'Source Group1' ตามรูป

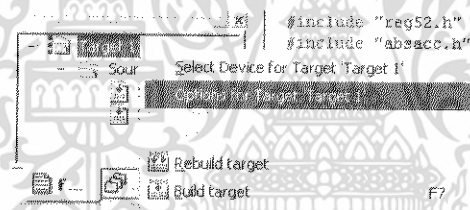


เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

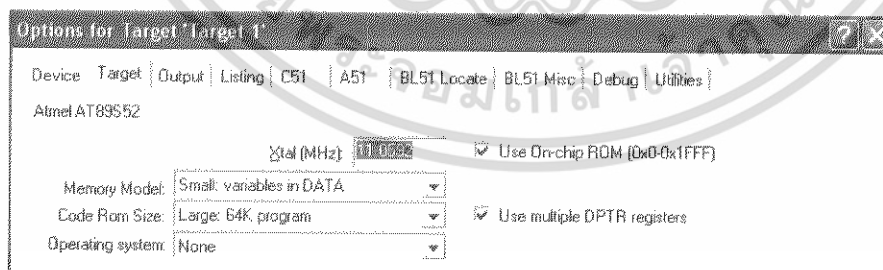
โปรแกรมจะไปที่โฟลเดอร์ที่เราได้สร้างไว้ เลือกไฟล์ชื่อ test.c ที่เราได้สร้างไว้ แล้วคลิกปุ่ม Add เพียงครั้งเดียว จากนั้นกดปุ่ม Close



ทำการเซต Option ของ Project โดยให้ไปคลิกปุ่มขวาของเมาส์ที่เครื่องหมาย + หน้า Target 1 แล้วเลือกหัวข้อ Option for Target 'Target 1' ตามรูป



โปรแกรมจะแสดงหน้าต่างรูปข้างล่าง ให้เราทำการเซตตามรูป



ในส่วนของ Target เราสามารถเซตค่าต่างๆ ได้ดังนี้

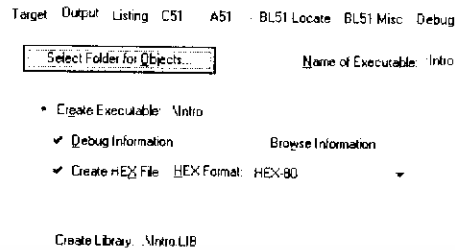
ให้เราเปลี่ยนความถี่ X-Tal ที่อยู่บนบอร์ดของเราเป็นหน่วย Mhz

เลือก Memory Model เป็น Small

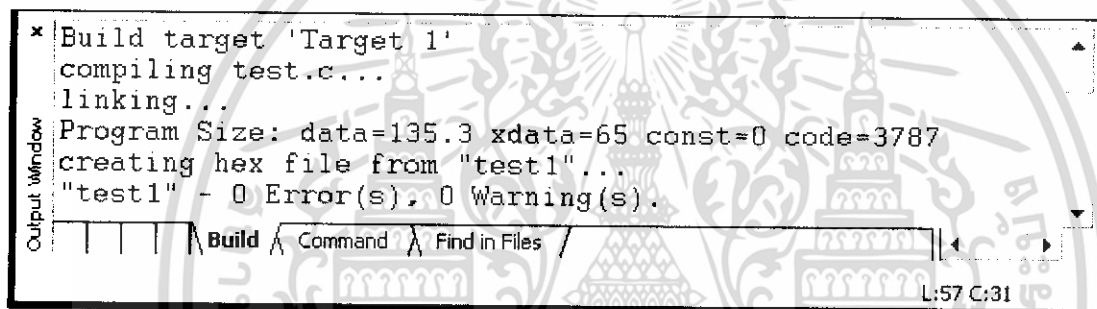
เลือก Code Rom Size เป็น Large

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Operating System เป็น None
 คลิกเลือก Use Onchip Rom , Use multiple DPTR
 ตามรูป อันนี้ขึ้นอยู่กับเบอร์ที่เราเลือกใช้



หลังจากที่เราทำการเขียนโปรแกรมแล้ว เราก็จะต้องทำการ Compile โปรแกรมที่เราเขียน ขึ้นมาว่า ถูกต้องตามรูปแบบของคำสั่ง หรือไม่โดยการคลิกที่ ไอคอน  ที่ Toolbar



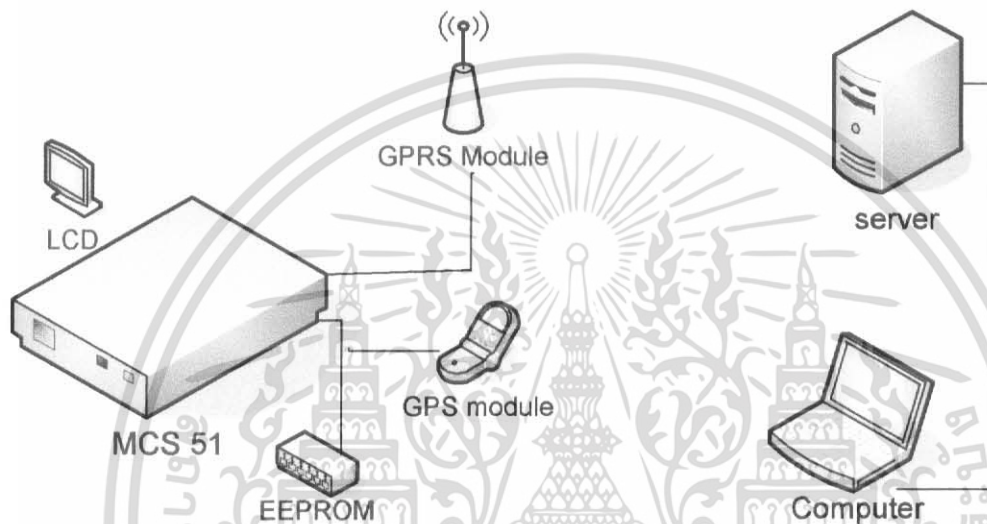
ผลที่ได้จากการ Compile จะแสดงที่หน้าต่าง Output โปรแกรมก็จะบอกว่าเราคำสั่งที่เราเขียนถูกต้องหรือไม่ และถ้าไม่ถูกต้องโปรแกรมก็จะบอกตำแหน่งของคำสั่งที่ผิดพลาด ผ่านทางหน้าต่าง Output ถ้าผิดพลาดก็ให้เราทำการ แก้ไขให้ถูกต้อง หลังจากที่เรา Compile ผ่านแล้วเราก็จะได้ไฟล์นามสกุล HEX ใน Folder ของโปรแกรมเราก็สามารถเอา HEX ไฟล์ไป โปรแกรมลงบนไมโครคอนโทรลเลอร์อีกที

บทที่ 3

หลักการออกแบบและการสร้าง

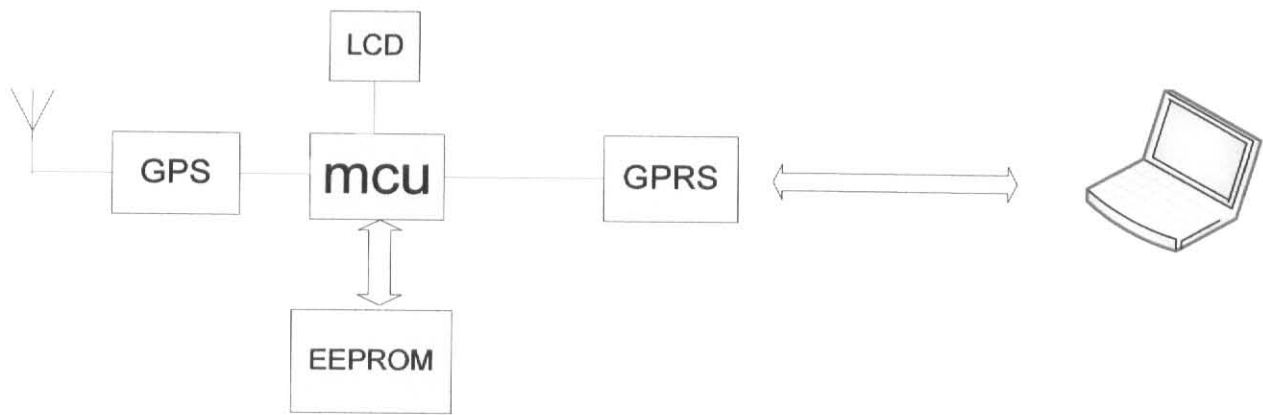
3.1 โครงสร้างการทำงานของระบบ

โครงสร้างนี้ของระบบนี้จะถูกคิดอยู่กับรถ โดยจะมีส่วนของการประมวลผลทั้งหมดและจัดการข้อมูลต่างๆ โดยจะมีโครงสร้างโดยรวมแสดงดังรูป



รูปที่ 3.1 โครงสร้างการทำงานของระบบ

หลักการการทำงานของระบบแสดงตำแหน่งของรถ เป็นการนำเอาเทคโนโลยีของ ระบบ GPS มาประยุกต์ใช้ในการประมวลผลข้อมูลต่างๆ เพื่อที่จะสามารถแสดงตำแหน่งของรถ ณ ขณะนั้น ว่าอยู่ ณ ตำแหน่งใด โดยผ่านระบบจีพีอาร์เอส และระบบสามารถบันทึกตำแหน่งละติจูด ลองจิจูดและเวลา หรือสามารถเลือกเก็บค่าต่างๆ ได้ตามต้องการและนำมาเก็บไว้ใน หน่วยความจำ เพื่อที่จะสามารถดูข้อมูลของการเดินทางแต่ละครั้งได้

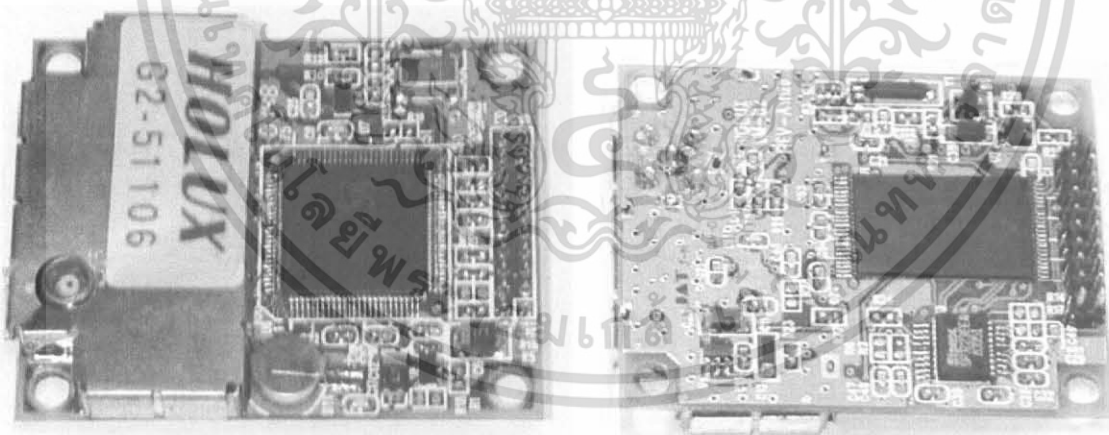


รูปที่ 3.2 แสดงบล็อกโคะแกรมของระบบ

3.2 ส่วนของเครื่องรับสัญญาณ

3.2.1 เครื่องรับระบบพิกัดตำแหน่ง (GPS Receiver)

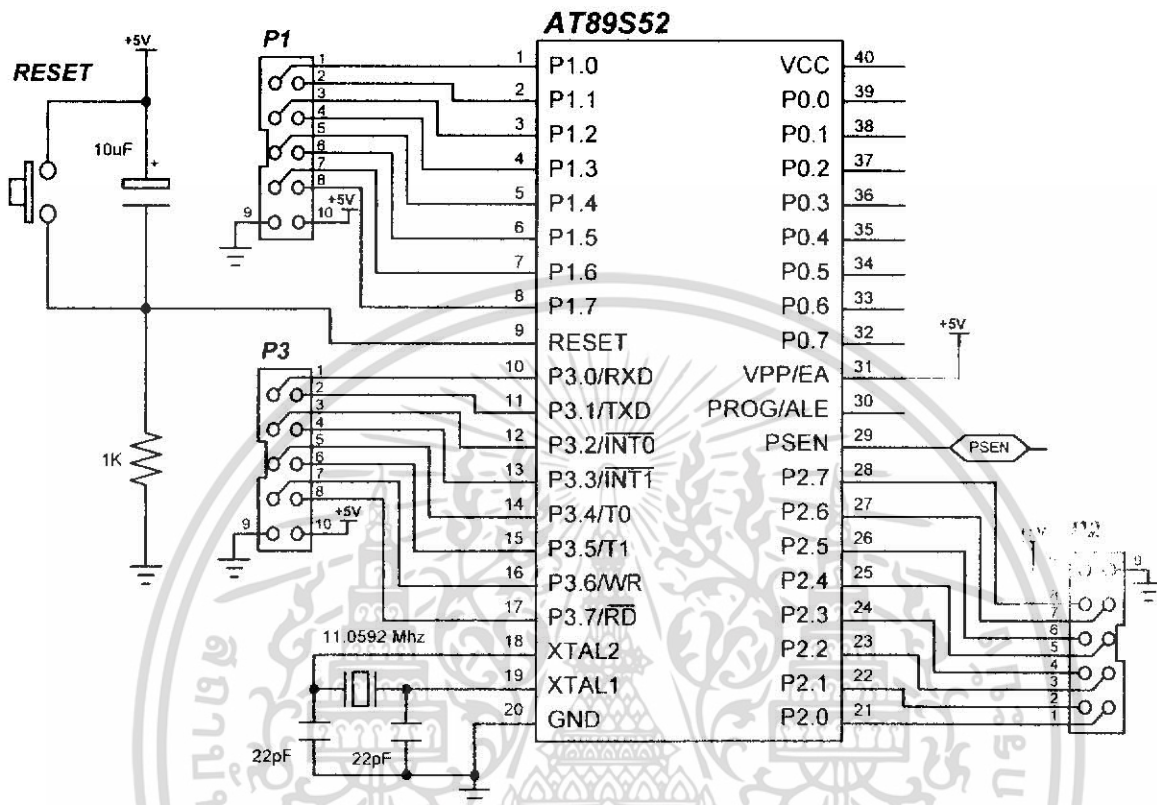
ส่วนประกอบของการรับสัญญาณตำแหน่งปัจจุบัน ประกอบด้วย 2 ส่วน คือ เครื่องรับ GPS และ ชุดรับส่งข้อมูลระหว่าง เครื่องรับ GPS และ ในการเชื่อมต่อตัวเครื่องรับ GPS เข้ากับระบบนั้นเราจะออกแบบให้ขา TX ของเครื่องรับสัญญาณ GPS ไปยังขา RX ของ MCS-51



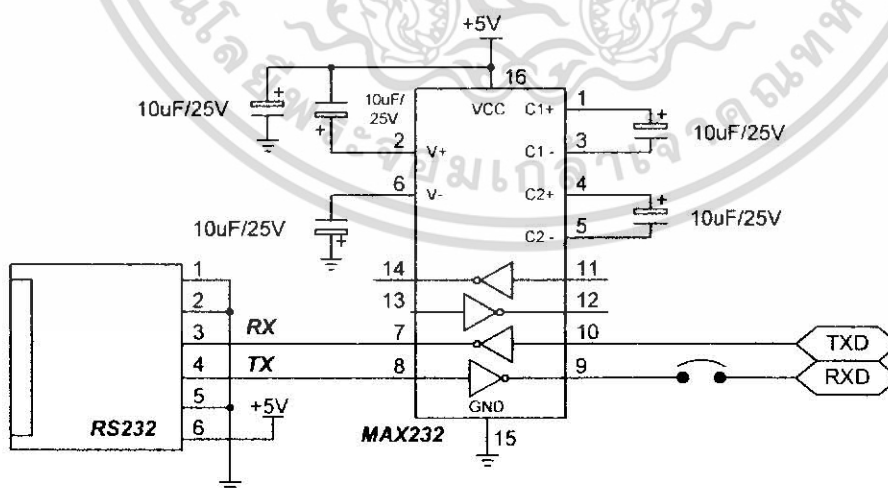
รูปที่ 3.3 เครื่องรับ GPS ที่เลือกใช้รุ่น GM-82

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.3 ส่วนของไมโครคอนโทรลเลอร์



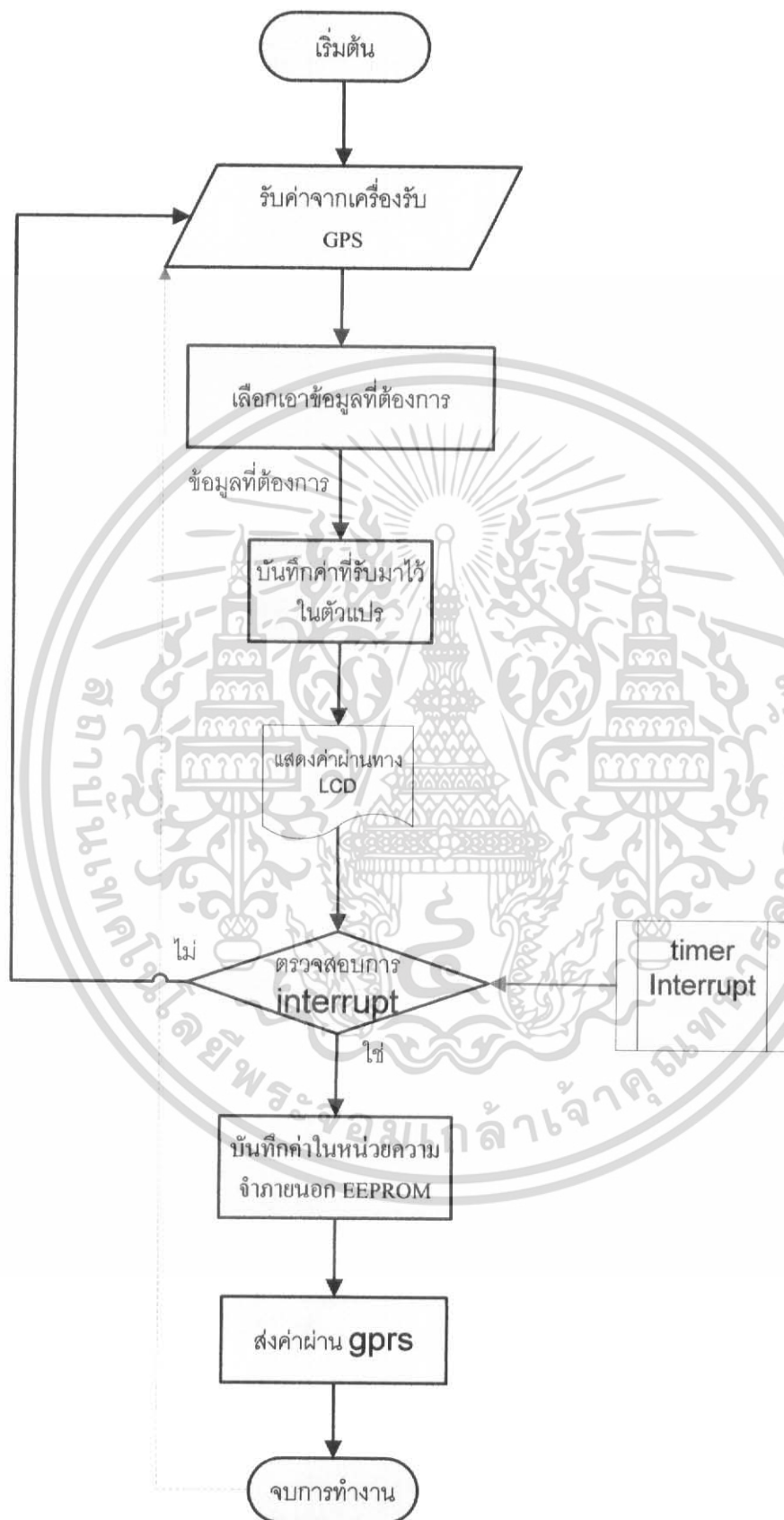
รูปที่ 3.4 วงจรการเชื่อมต่อพอร์ตต่างๆของMCS-51



รูปที่ 3.5 วงจรการเชื่อมต่อกับRS-232 ของMCS-51

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.4 ส่วนของการทำงานของโปรแกรม



รูปที่ 3.9 แสดงผังการทำงานของส่วนโปรแกรม

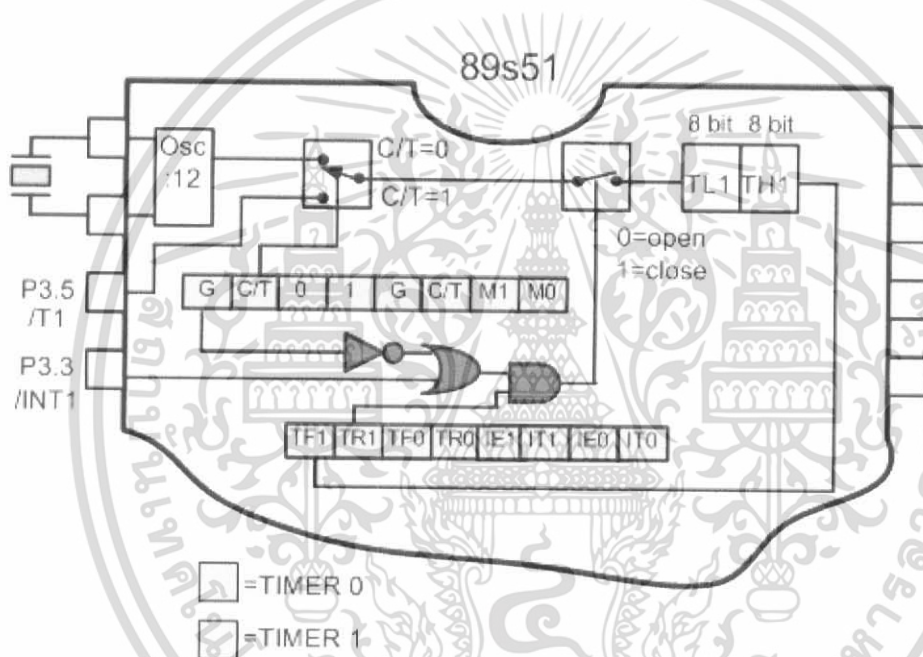
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- การทำงานของโปรแกรมหลัก

เริ่มจากการรอรับค่าพิกัดตำแหน่งจากเครื่องรับสัญญาณGPS เราจะเลือกเฉพาะข้อมูลที่ต้องการ โดยในโครงการนี้จะเลือกในส่วนของตำแหน่งพิกัดละติจูด ลองติจูดเท่านั้น จากนั้นจะเก็บค่าลงในตัวแปรและแสดงผลผ่านทางจอแอลซีดี หลังจากนั้นจะมีการตรวจสอบสัญญาณการขัดจังหวะของโปรแกรม เนื่องจากว่าโครงการนี้มีการบันทึกค่าในแต่ละช่วงเวลาได้ หากไม่มีการขัดจังหวะโปรแกรม จะเริ่มกลับไปรอรับค่าใหม่อีกครั้ง

3.4. การเขียนโปรแกรมย่อย

- **Timer Interrupt** ส่วนนี้จะใช้ในการตั้งเวลาในการเก็บ ระยะเวลาใดๆ ตัวอย่างเช่นต้องการเก็บค่าพิกัดตำแหน่ง ทุกๆ 5 วินาที พอถึง 5วินาที โปรแกรมย่อยนี้จะส่งค่ามายังโปรแกรมหลัก



รูปที่ 3.10 แสดงวงจรภายในของ Timer/Counter#0

นี่คือวงจรภายใน ซึ่งแสดงความสัมพันธ์ระหว่างวงจรกับรีจิสเตอร์ว่ามีความสัมพันธ์กันอย่างไร ซึ่งวงจรข้างบนนี้สามารถจัดให้มีการทำงานแบบTimer หรือ Counter ก็ได้ แต่โครงการนี้เราจะใช้Timer มากกว่า เป็นการจับวงจรหารแบบ16บิต โดยมี TH0 และ TH1 เป็นรีจิสเตอร์ในการหารความถี่ ซึ่งเราสามารถอธิบายการทำงานของวงจรข้างบนได้ดังนี้

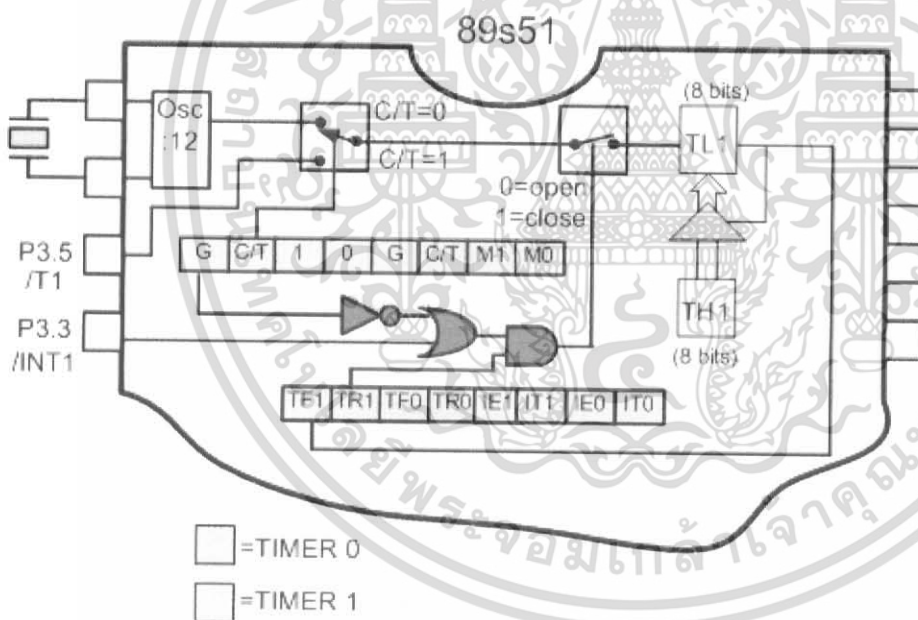
การทำงานของวงจรTimer/Counter#0

- วงจรosc ซึ่งวงจรนี้จะเชื่อมต่อกับx-tal ภายนอกเพื่อสร้างสัญญาณนาฬิกา 11.0592Mhz
- สัญญาณนาฬิกาที่ได้จะส่งไปให้วงจรหารความถี่ด้วยค่า 12เราก็จะได้ความถี่ 921600 Hz
- ความถี่ที่ได้ก็จะมาเจอสวิตตัวหนึ่งทำหน้าที่เลือกสัญญาณนาฬิกาจากวงจรoscหรือจากขาT0 ของMCS-

51

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- เราต้องให้วงจรทำงานเป็นแบบTimer ฉะนั้นเราจึงให้บิต C/T เป็น "0" เพื่อเลือกสัญญาณนาฬิกาของวงจรOsc เพราะฉะนั้นรีจิสเตอร์ TMOD = 01H
- จากนั้นจะเจอสวิทอีกตัวหนึ่ง ซึ่งทำหน้าที่เปิดปิดสัญญาณนาฬิกาที่จะเข้าไปยังวงจรหาร โดยการคุมของบิต TR0 ของรีจิสเตอร์ TCON กับบิตGate ของรีจิสเตอร์ TMOD
- บิตGate จะทำหน้าที่ป้องกันไม่ให้สัญญาณจากขาINT0 ควบคุมการเปิดปิดของสัญญาณนาฬิกาที่จุดa เพราะฉะนั้นเราจึงเซตให้บิตนี้เป็น 0
- เมื่อสัญญาณนาฬิกาเข้ามาถึงวงจรหาร โดยมีรีจิสเตอร์TH0 ,TL0เป็นตัวหาร
- รีจิสเตอร์จะทำหน้าที่เป็นวงจรcounter แบบนับขึ้น
- เมื่อนับจนถึง0000บิต TF0 จะถูกเซตเป็น "1" ทันที จากนั้น Mcs-51จะทำการตรวจสอบบิตEAและบิตET0 ของรีจิสเตอร์IEว่าเป็น 1ทั้งคู่หรือไม่
- ถ้าเป็นมันจะทำโปรแกรมจะทำการกระโดดไปยังโปรแกรมย่อย interrupt 1
- **Serial Port** ในโครงงานนี้จะทำการเชื่อมต่อระหว่างMCUกับ คอมพิวเตอร์ผ่านทางพอร์ตอนุกรมมาตรฐานRS-232 จึงต้องเขียน โปรแกรมควบคุมพอร์ตอนุกรม



รูปที่ 3.11 แสดงวงจรภายในของTimer/Counter#1

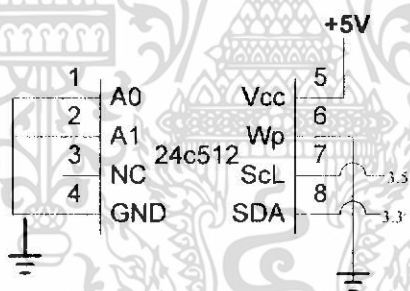
การทำงานของวงจรพอร์ตอนุกรม เริ่มต้นจากสร้างความเร็วการสื่อสาร (Baud rate) ก่อน โดยใน Mcs-51 จะบังคับให้ใช้ Timer 1 เท่านั้น ซึ่งเราจะต้องกำหนดให้Timer 1 ทำงานที่โหมด2จะทำงานแบบ 8บิตAuto Reload หมายความว่าเมื่อสัญญาณนาฬิกาที่ได้จากวงจรOsc ถูกหารด้วย12แล้วป้อนเข้าที่รีจิสเตอร์ TL1ค่าของTL1จะถูกเพิ่มค่าไปเรื่อยๆจนกลายเป็น00H ค่าที่อยู่ในรีจิสเตอร์TH1จะถูกใส่เข้าไปที่รีจิสเตอร์TL1ทันทีแล้วTL1ก็จะถูกเพิ่มไปเรื่อยๆจนเป็น00H ใหม่ทำให้เอาที่พุดของTL1เป็นสัญญาณนาฬิกาค่าหนึ่ง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หลังจากที่เราได้สัญญาณนาฬิกาจาวงจร Timer 1 แล้วสัญญาณจะถูกส่งไปยังวงจรหารหารความถี่ด้วย 32กับ16 โดยมีทSMOD ของรีจิสเตอร์ PCON เป็นตัวเลือกวงจรหาร หลังจากนั้นก็จะได้ความถี่ Baud Rate ป้อนให้กับวงจรSift Register ในการส่งข้อมูลกับข้อมูลจากSerial Port ผ่านทางขา TXD และRXD

- ขั้นตอนการเขียนโปรแกรม Serial Port
- กำหนดค่ารีจิสเตอร์ TMOD ให้ Timer 1 ทำงานที่โหมด 2 8bit Auto Reload ฉะนั้น Tmod = 20H
 - กำหนดค่ารีจิสเตอร์ SCON โดยทำงานที่โหมด 1 8bit UART เซตบิต REN เพื่อให้สามารถรับข้อมูลได้คั้งนั้น SCON = 50H
 - กำหนดค่าTH1 ตามความถี่Baud Rate ที่เรากำหนด(TH1=FDH, 9600Hz)
 - เปิดสัญญาณนาฬิกาให้เข้าไปที่ T1L โดยการเซตบิตTR1
 - ถ้าต้องการให้สามารถ Interrupt จากSerial Port ได้ ให้ทำการเซตบิต EA1=1 และ ES=1 ของรีจิสเตอร์ IE

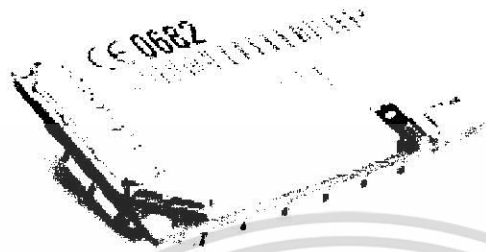
-EEPROM เป็นอุปกรณ์ตัวหนึ่งที่สำคัญกับการสร้างโครงการนี้ซึ่งทำหน้าที่เก็บพิกัดตำแหน่งGPSในการเคลื่อนที่ของรถ ซึ่งสามารถเก็บข้อมูลได้แม้ไม่มีไฟเลี้ยง ในโครงการนี้จะใช้EEPROM แบบอนุกรมเพราะใช้พอร์ตในการควบคุมน้อยเพียงไม่กี่บิต ซึ่งการเชื่อมต่อกับMCU ซึ่งจะเป็นลักษณะแบบI2C BUS



- SCI = เป็นขาสัญญาณนาฬิกา(clock) ของ I2C Bus
- SDA = เป็นขาข้อมูล(Data) ของ I2C Bus
- WP = เป็นขาป้องกันการเขียนข้อมูลเมื่อต่อ Vcc
- A1-A0 = เป็นขาAddress เราสามารถให้ไอซีเบอร์นี้อยู่Addressไหนก็สามารถกำหนดได้ที่งานนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.5 ส่วนของการส่งข้อมูลผ่าน GPRS module



คำสั่ง AT COMMAND ในการติดต่อกับระบบ GPRS

- คำสั่งการขอเปิดการติดต่อ GPRS

รูปแบบคำสั่ง

```
AT+CGATT=<mode><CR>
```

.ให้ตั้งค่า AT+CGATT=1

; 0 คือ ไม่ใช้งานและ 1 คือ ใช้งาน

- คำสั่งติดต่อกับศูนย์บริการ

รูปแบบคำสั่ง

```
AT#APNSERV="www.dtac.co.th"<CR>
```

โดย เป็นการเริ่มเปิดการติดต่อระบบ GPRS

- คำสั่งเชื่อมต่อกลับเครือข่าย

รูปแบบคำสั่ง

```
AT#ICPSERV="161.246.xxx.xxx"<CR>
```

โดย เป็นการเชื่อมต่อกลับเครือข่ายผ่านทาง IP Address

- คำสั่งติดต่อผ่านพอร์ต

รูปแบบคำสั่ง

```
AT#ICPPORT="1000"<CR>
```

โดย เป็นการส่งข้อมูลผ่านทางพอร์ต1000 ของเครื่องserver

- คำสั่งเปิดการส่งข้อมูล/ปิดการส่งข้อมูล

รูปแบบคำสั่ง

```
AT#OTCP<CR>
```

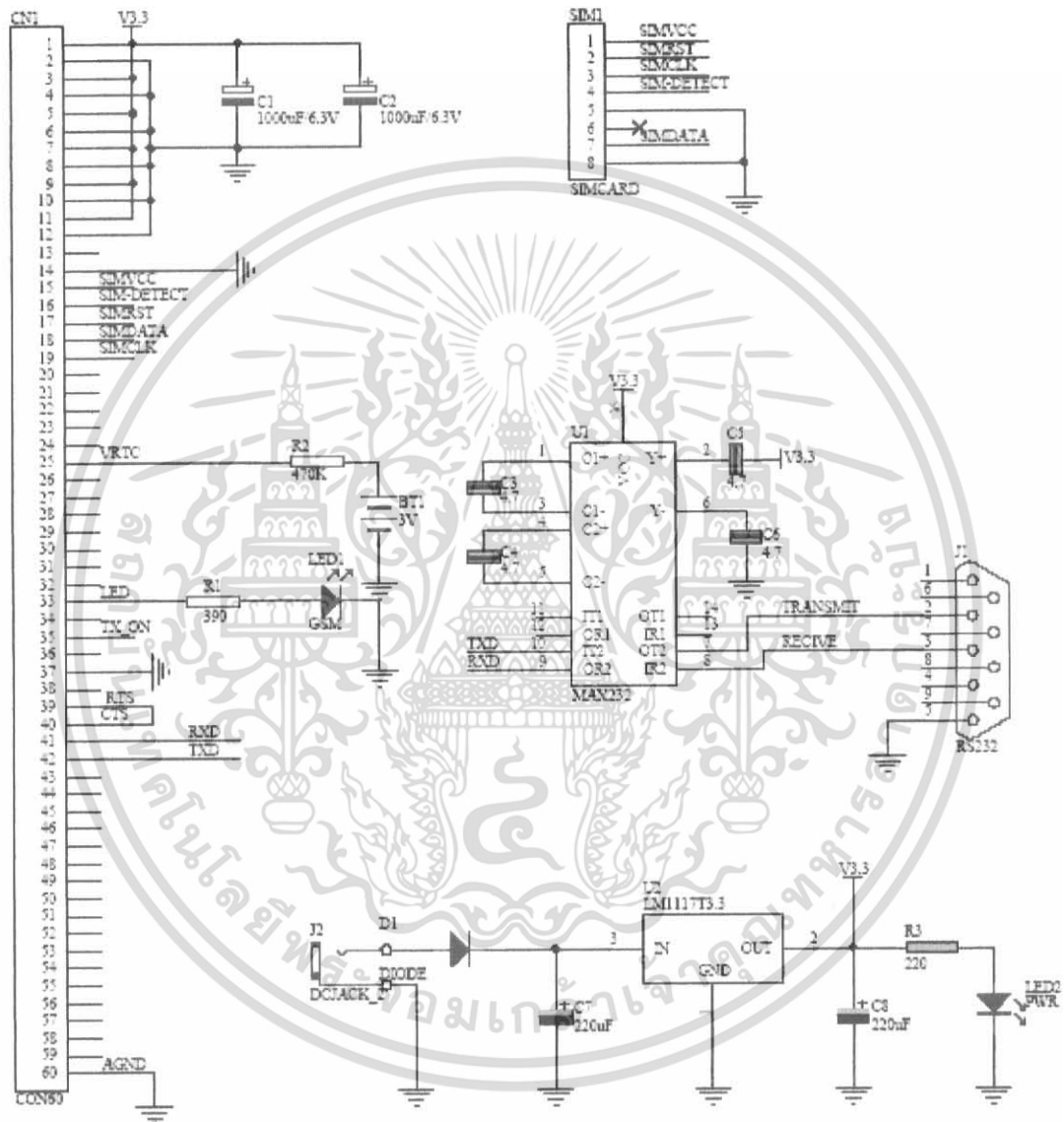
โดย เป็นการเริ่มส่งข้อมูลผ่านระบบ GPRS

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- คำสั่งหยุดการติดต่อกับระบบเครือข่าย
รูปแบบคำสั่ง

AT#CONNECTIONSTOP<CR>

โดย เป็นหยุดการติดต่อกับระบบเครือข่าย



รูปที่3.12 วงจรการเชื่อม GPRS Module กับบอร์ดควบคุมไมโครคอนโทรลเลอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 4

ผลการทดลอง

ผลการทดลองกล่าวถึงการทำงานส่วนต่างๆ ที่ได้พัฒนาขึ้นในโครงการนี้ โดยแบ่งเป็นหัวข้อการทดลองหลักๆ คือ การทดสอบการรับค่าจากโมดูลเครื่องรับ GPS ส่วนของการทำงานของโปรแกรม การบันทึกค่าพิกัดณ.เวลาต่างๆ การนำพิกัดที่ได้ทำการบันทึกไปพล็อตบนโปรแกรมแผนที่

4.1 ทดสอบการรับค่าจากโมดูลเครื่องรับ GPS

วัตถุประสงค์ของการทดลอง

1. ทดสอบการทำงานของโมดูลเครื่องรับ GPS
2. สามารถนำข้อมูลที่รับไปใช้งานได้

```

123 - HyperTerminal
File Edit View Call Transfer Help
$GPGGA,154030.892,1342.0098,N,10030.0524,E,1,05,1.8,-22.3,M,-27.5,M,0.0,0000*49
$GPGSA,A,3,28,04,24,17,08,3,3,1.8,2.6*39
$GPRMC,154030.892,A,1342.0098,N,10030.0524,E,0.00,130806.0,15
$GPGGA,154031.892,1342.0098,N,10030.0524,E,1,05,1.8,-21.8,M,-27.5,M,0.0,0000*40
$GPGSA,A,3,28,04,24,17,08,3,3,1.8,2.6*39
$GPRMC,154031.892,A,1342.0098,N,10030.0524,E,0.00,130806.0,14
$GPGGA,154032.892,1342.0098,N,10030.0523,E,1,05,1.8,-21.3,M,-27.5,M,0.0,0000*4F
$GPGSA,A,3,28,04,24,17,08,3,3,1.8,2.6*39
$GPRMC,154032.892,A,1342.0098,N,10030.0523,E,0.00,130806.0,10
$GPGGA,154033.892,1342.0098,N,10030.0523,E,1,05,1.8,-20.8,M,-27.5,M,0.0,0000*44
$GPGSA,A,3,28,04,24,17,08,3,3,1.8,2.6*39
$GPRMC,154033.892,A,1342.0098,N,10030.0523,E,0.00,130806.0,11
$GPGGA,154034.892,1342.0099,N,10030.0523,E,1,05,1.8,-20.4,M,-27.5,M,0.0,0000*4E
$GPGSA,A,3,28,04,24,17,08,3,3,1.8,2.6*39
$GPGSV,2,1,08,28,65,108,47,04,63,304,46,24,57,358,46,17,38,013,36*77
$GPGSV,2,2,08,02,37,251,00,08,22,179,37,20,17,049,00,09,09,303,00*7B
$GPRMC,154034.892,A,1342.0099,N,10030.0523,E,0.00,130806.0,17
$GPGGA,154035.892,1342.0099,N,10030.0523,E,1,05,1.8,-20.2,M,-27.5,M,0.0,0000*49
$GPGSA,A,3,28,04,24,17,08,3,3,1.8,2.6*39
$GPRMC,154035.892,A,1342.0099,N,10030.0523,E,0.00,130806.0,16
$GPGGA,154036.892,1342.0100,N,10030.0523,E,1,05,1.8,-20.0,M,-27.5,M,0.0,0000*49
$GPGSA,A,3,28,04,24,17,08,3,3,1.8,2.6*39
$GPRMC,154036.892,A,1342.0100,N,10030.0523,E,0.00,130806.0,14
  
```

Connected 0:02:33 Auto detect 4800 8-N-1 SCROLL CAPS NUM Capture Print echo

รูปที่ 4.1 ผลการรับข้อมูลจากเครื่องรับ GPS

จากรูป เป็นการทดลองการรับข้อมูลจากเครื่องรับสัญญาณ GPS โดยต่อผ่านวงจร MAX232

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.2 การรับค่าจากโมดูลเครื่องรับ GPS โดยผ่านทางบอร์ดไมโครคอนโทรลเลอร์

วัตถุประสงค์ของการทดลอง

1. เพื่อต้องการในส่วนของข้อมูลที่จำเป็นต้องใช้ในโครงการ คือ ส่วนของพิกัดตำแหน่ง
2. การทดลองนี้ต้องการในส่วนของ RMC โปรโตคอลเท่านั้น
3. นำค่าที่ได้ไปแสดงผลบนจอ LCD

```

$GPRMC,024.036,A,1341.3118,N,10031.3216,E,40.66,25.36,211006.0,04U
$GPRMC,024.041,A,1341.3687,N,10031.3418,E,43.39,20.12,211006.0,07U
$GPRMC,024.046,A,1341.4212,N,10031.3717,E,43.03,35.78,211006.0,03U
$GPRMC,024.051,A,1341.4701,N,10031.4083,E,42.42,36.23,211006.0,06U
$GPRMC,024.056,A,1341.5169,N,10031.4430,E,40.32,35.97,211006.0,00U
$GPRMC,024.101,A,1341.5579,N,10031.4786,E,37.75,47.19,211006.0,08U
$GPRMC,024.106,A,1341.5862,N,10031.5233,E,37.26,65.56,211006.0,00U
$GPRMC,024.111,A,1341.5993,N,10031.5771,E,40.93,83.75,211006.0,00U
$GPRMC,024.116,A,1341.6001,N,10031.6354,E,39.09,88.61,211006.0,04U
$GPRMC,024.121,A,1341.6070,N,10031.6882,E,36.75,72.70,211006.0,07U
$GPRMC,024.126,A,1341.6257,N,10031.7430,E,45.85,70.62,211006.0,09U
$GPRMC,024.131,A,1341.6489,N,10031.8039,E,41.06,69.06,211006.0,0EU
$GPRMC,024.136,A,1341.6643,N,10031.8448,E,24.33,69.94,211006.0,01U
$GPRMC,024.141,A,1341.6723,N,10031.8664,E,11.37,70.36,211006.0,08U
$GPRMC,024.146,A,1341.6740,N,10031.8714,E,0.00,68.56,211006.0,37
$GPRMC,024.151,A,1341.6742,N,10031.8702,E,0.00,68.56,211006.0,35
$GPRMC,024.156,A,1341.6755,N,10031.8693,E,0.00,68.56,211006.0,3D
$GPRMC,024.201,A,1341.6752,N,10031.8691,E,0.00,68.56,211006.0,39
$GPRMC,024.206,A,1341.6748,N,10031.8693,E,0.00,68.56,211006.0,30
$GPRMC,024.211,A,1341.6752,N,10031.8717,E,8.54,70.37,211006.0,37
$GPRMC,024.216,A,1341.6826,N,10031.8922,E,16.72,69.32,211006.0,02U
$GPRMC,024.221,A,1341.6923,N,10031.9165,E,16.11,65.73,211006.0,05U
$GPR_
  
```

รูปที่ 4.2 ผลการรับข้อมูลในส่วนของ RMC โปรโตคอล

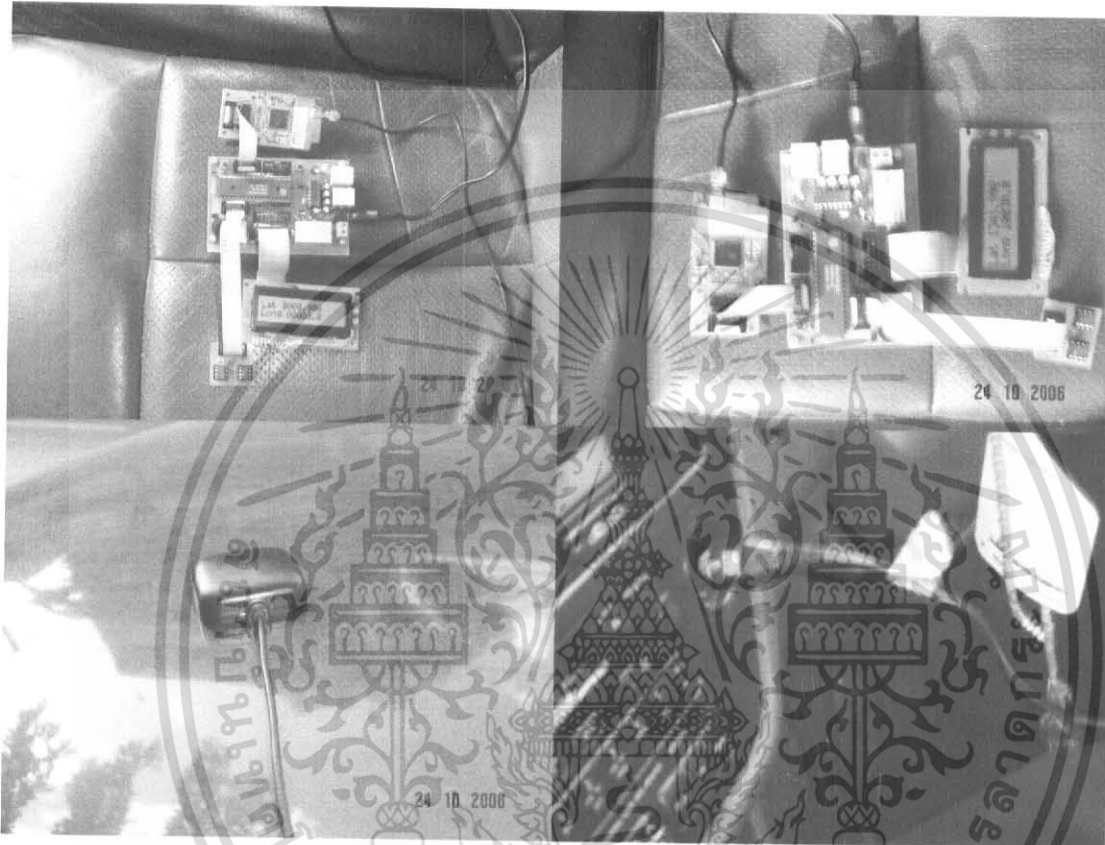


รูปที่ 4.3 ผลการนำข้อมูลพิกัดตำแหน่งแสดงบน LCD

เอกสารนี้เป็นเอกสารที่สวอนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.2 ทดสอบการรับค่าพิกัดตำแหน่งและทำการบันทึกลงในหน่วยความจำภายนอก วัตถุประสงค์ของการทดลอง

1. ทดสอบการรับค่าพิกัดและบันทึกลงในหน่วยความจำ EEPROM
2. ทดลองติดตั้งจริงในรถยนต์โดยทำการเคลื่อนไปยังเส้นทางต่างๆ
3. สามารถนำค่าที่ได้บันทึกไปประยุกต์ใช้ประโยชน์ได้



รูปที่ 4.4 แสดงการทดลองติดตั้งจริงในรถยนต์

การทดลองในครั้งนี้จะทดสอบการเก็บค่าพิกัดแล้วจะนำค่าที่บันทึกในEEPROM ไปใช้ในการทดสอบในส่วนอื่นต่อไป โดยได้ทำการเคลื่อนที่รถยนต์ไปหลากหลายเส้นทาง รวมเป็นระยะทางประมาณ 50 กิโลเมตร และ ทำการบันทึกค่าพิกัดในทุกๆ 5 วินาที

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.3 ทดลองนำผลที่ได้จากการบันทึกใน EEPROM ผ่านทางพอร์ต RS-232

วัตถุประสงค์ของการทดลอง

1. ตรวจสอบค่าที่บันทึกอยู่ใน EEPROM ว่าถูกต้องหรือไม่
2. นำค่าที่ได้นี้ ไปพล็อตลงบนแผนที่เพื่อดูบันทึกการเดินทางของรถว่าไปในเส้นทางใดบ้าง ณ เวลาใด

Copyright 2002 © MappointAsia (Thailand) Company Limited

SmartMAP

Viewer 3.0

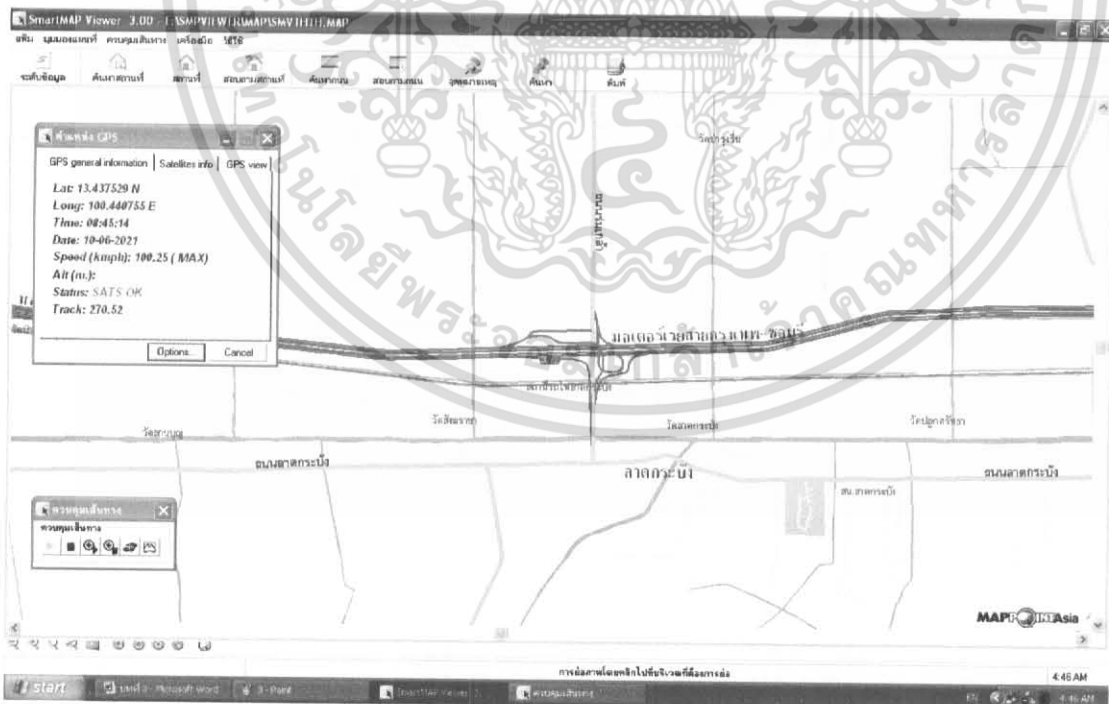
NEXT
generation of mapping software

Thailand Map

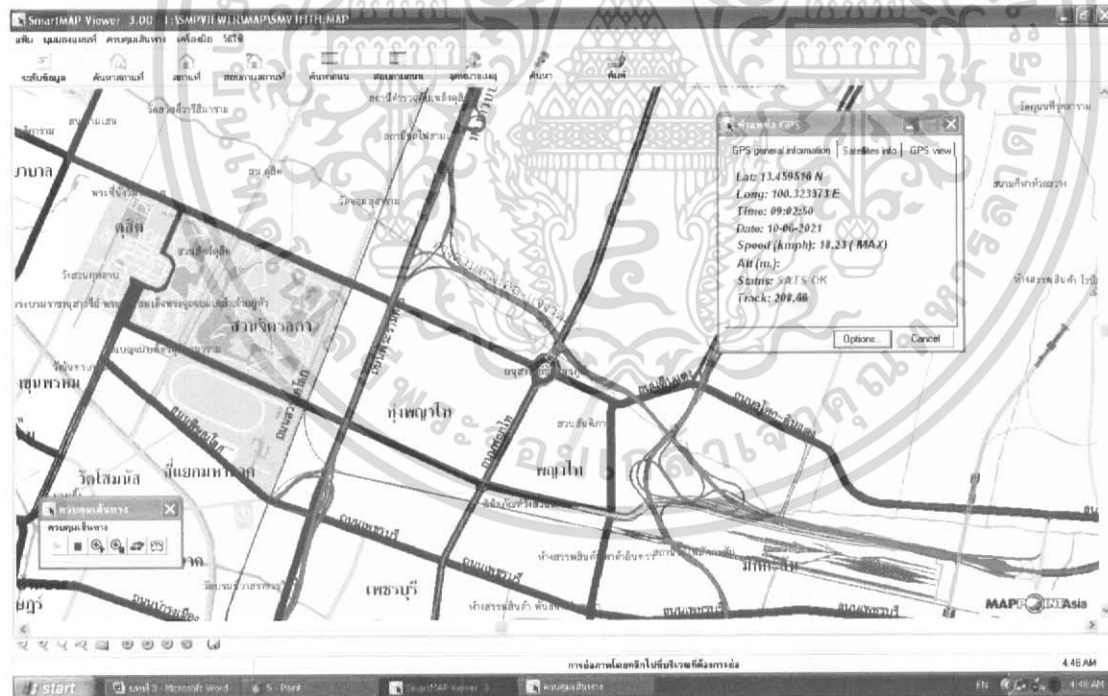
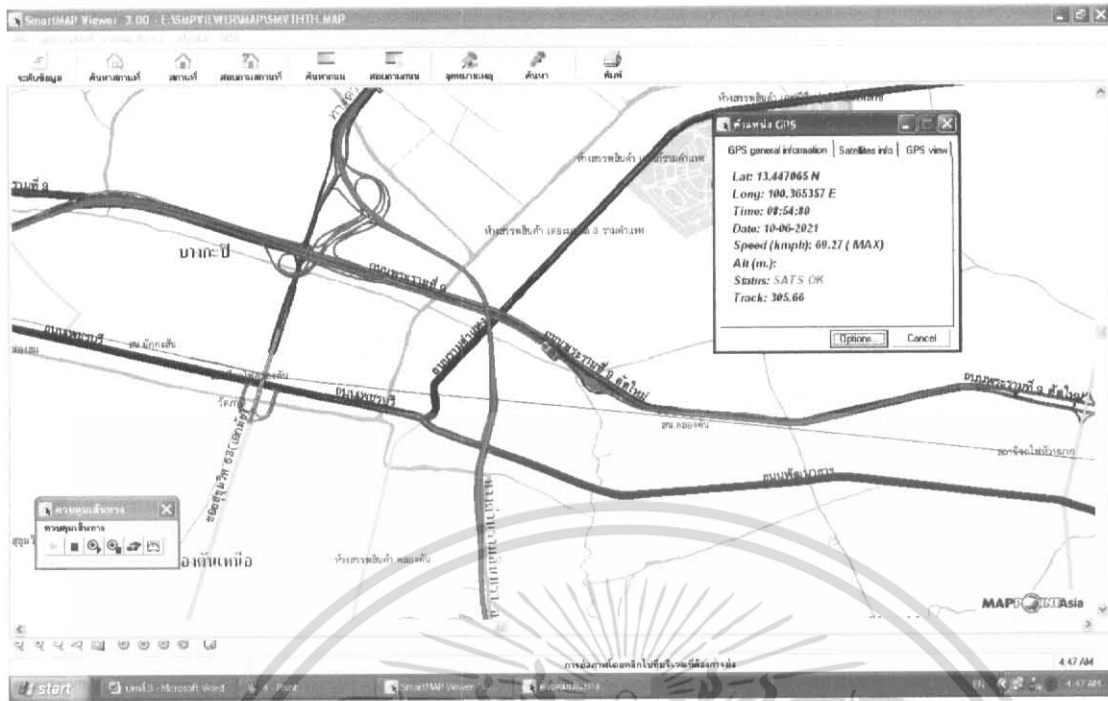
MappointAsia (Thailand) Company Limited
 128/188 Payatai Plaza Building 17 th Floor
 Payathai Road, Rachathevee, Bangkok 10400
 Thailand Tel : (662) 209-9112 Fax : (662) 209-9113

MAPPPOINTAsia
<http://www.mappointasia.com>

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
 ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



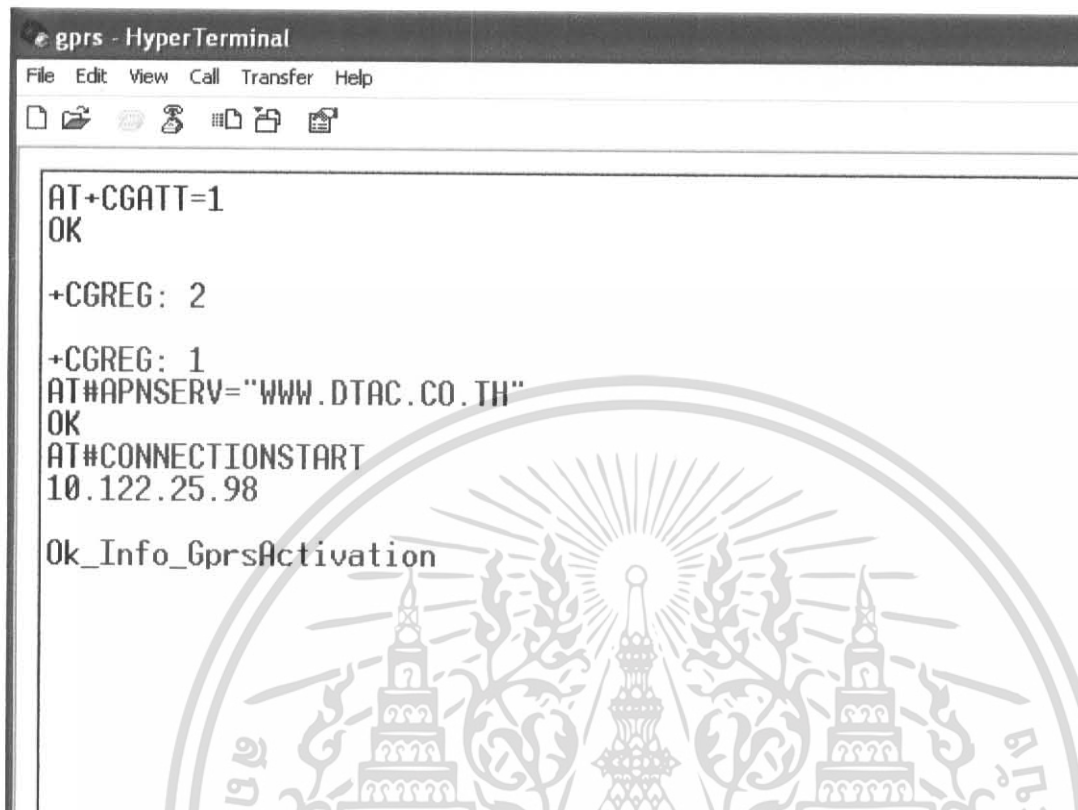
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.5 แสดงผลการนำค่าพิกัดไปพล็อตลงบน โปรแกรมแผนที่

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.4 ทดลองใช้คำสั่ง AT COMMAND เพื่อเชื่อมต่อกับเครือข่าย



```

gprs - HyperTerminal
File Edit View Call Transfer Help
AT+CGATT=1
OK
+CGREG: 2
+CGREG: 1
AT#APNSERV='WWW.DTAC.CO.TH'
OK
AT#CONNECTIONSTART
10.122.25.98
Ok_Info_GprsActivation
  
```

รูปที่ 4.6 แสดงการเชื่อมต่อกับเครือข่ายโดยใช้คำสั่ง AT COMMAND

4.5 ทดลองส่งค่าพิกัดผ่าน GPRS MODULE

วัตถุประสงค์ของการทดลอง

1. ทดสอบการเชื่อมต่อGPRS เข้าสู่เครือข่ายอินเทอร์เน็ต
2. สามารถส่งค่าพิกัดต่างๆไปยังเครื่องเซิร์ฟเวอร์ปลายทางได้
3. ทดสอบการเชื่อมต่อกันระหว่างบอร์ดควบคุมกับGPRS module

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

test_com1 - HyperTerminal
AT+CGATT=1
AT#APNSERV="WWW.DTAC.CO.TH"
AT#CONNECTIONSTART
AT#TCPSEV="161.246.65.110"
AT#TCPPORT=1500
AT#OTCP
$GPRMC,000029.997,V,0000.0000,N,00000.0000,E,0.00,180721,0B
$GPRMC,000034.997,V,0000.0000,N,00000.0000,E,0.00,180721,07
$GPRMC,000039.997,V,0000.0000,N,00000.0000,E,0.00,180721,0A
$GPRMC,000044.997,V,0000.0000,N,00000.0000,E,0.00,180721,00
$GPRMC,000049.996,V,0000.0000,N,00000.0000,E,0.00,180721,0C
$GPRMC,000054.996,V,0000.0000,N,00000.0000,E,0.00,180721,00
$GPRMC,000059.996,V,0000.0000,N,00000.0000,E,0.00,180721,0D
$GPRMC,000104.995,V,0000.0000,N,00000.0000,E,0.00,180721,07
$GPRMC,000109.995,V,0000.0000,N,00000.0000,E,0.00,180721,0A
$GPRMC,000114.995,V,0000.0000,N,00000.0000,E,0.00,180721,06
$GPRMC,000119.994,V,0000.0000,N,00000.0000,E,0.00,180721,0A
$GPRMC,000124.994,V,0000.0000,N,00000.0000,E,0.00,180721,04
$GPRMC,000129.994,V,0000.0000,N,00000.0000,E,0.00,180721,09
$GPRMC,000134.993,V,0000.0000,N,00000.0000,E,0.00,180721,02
$GPRMC,000139.993,V,0000.0000,N,00000.0000,E,0.00,180721,0F
$GPRMC,000144.993,V,0000.0000,N,00000.0000,E,0.00,180721,05

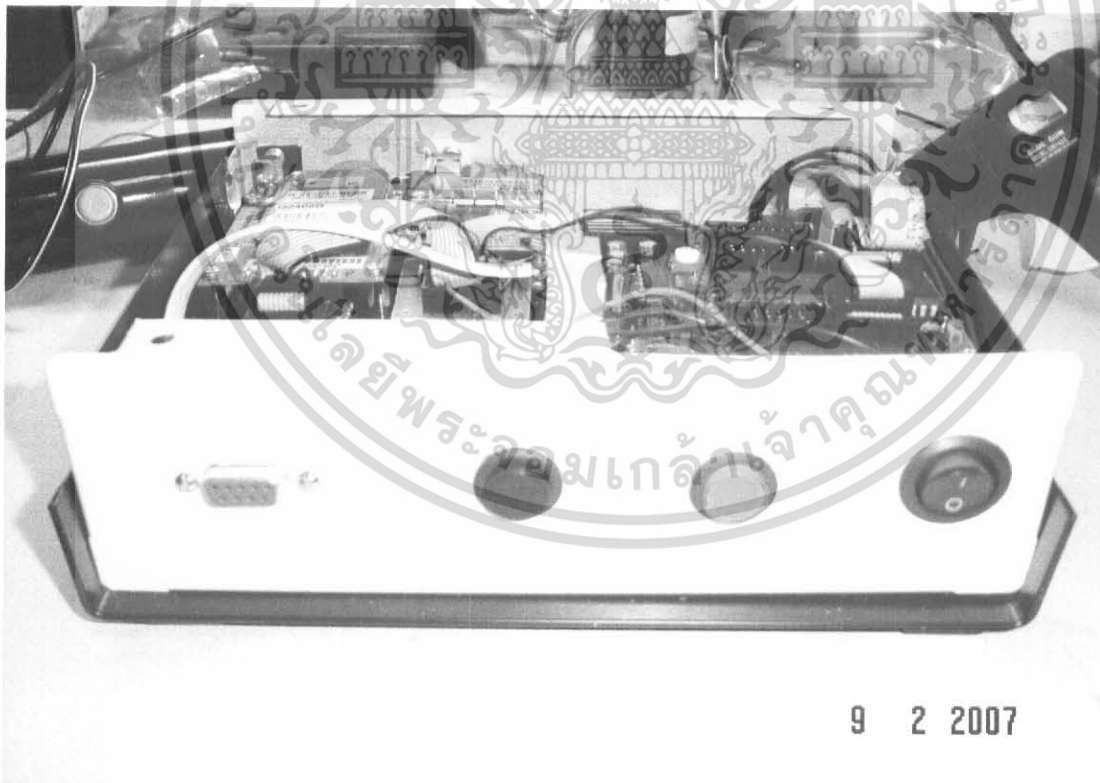
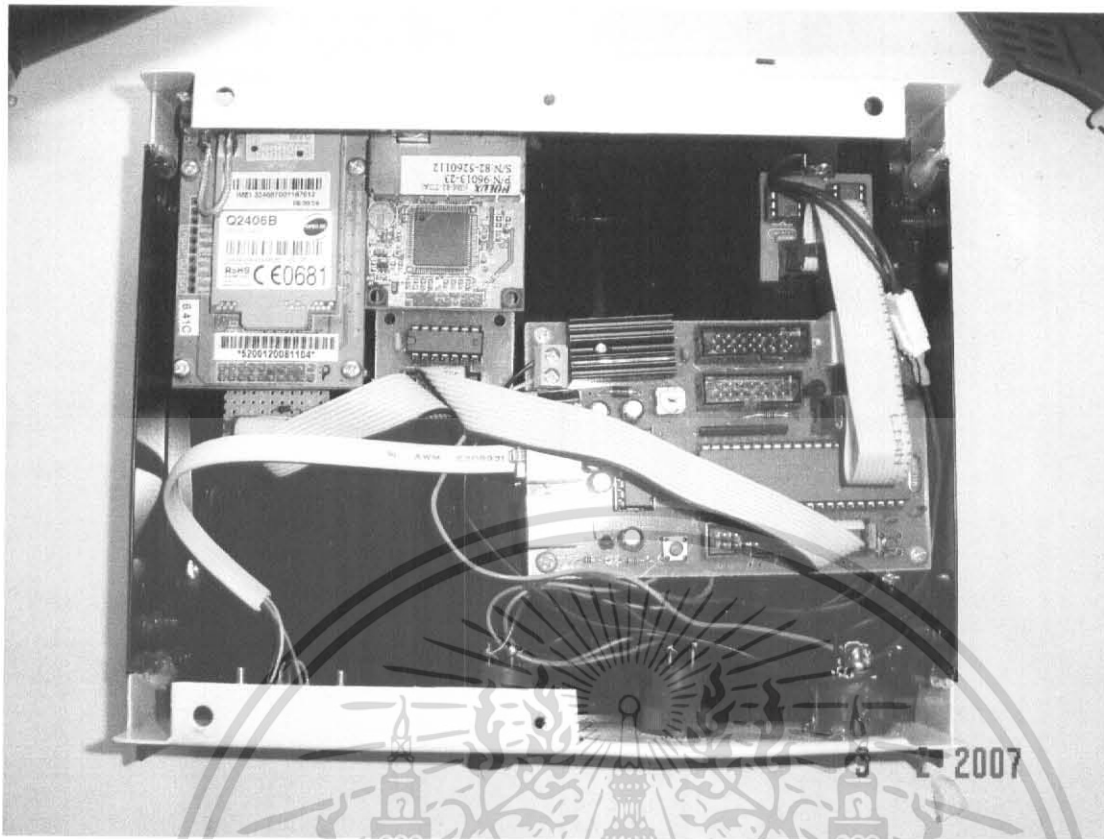
Connected 0:03:32 Auto detect 4800 8-N-1

```

รูปที่ 4.7 แสดงผลการส่งค่าพิกัดไปยังเครื่องเซิร์ฟเวอร์

ในการทดลองนี้ ได้ทำการเขียนโปรแกรมเพื่อทำการทดสอบการรับค่าพิกัดต่างๆ โดยติดตั้ง วิทยุเครื่องเซิร์ฟเวอร์ โดยทำการส่งข้อมูลผ่านทาง GPRS MODULE ไปให้เพื่อนำค่าพิกัดที่ได้ไปแสดงตำแหน่งลงบนแผนที่

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.8 แสดงรูปอุปกรณ์ภายในกล่องชุดอุปกรณ์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.9 แสดงกล่องอุปกรณ์เครื่องระบุตำแหน่งและบันทึกเส้นทางที่เสร็จสมบูรณ์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 5 บทวิจารณ์และบทสรุป

สรุปผลการพัฒนาโครงการ

โครงการนี้ได้สำเร็จตามจุดประสงค์ตามที่ได้ตั้งไว้คือ สามารถรับค่าพิกัดตำแหน่งที่ได้จากเครื่องรับสัญญาณจีพีเอส และนำค่าไปบันทึกยังหน่วยความจำ และยังสามารถนำค่าพิกัดนั้นพล็อตลงบนแผนที่ได้

โครงการที่ได้จัดทำขึ้นนี้สามารถทำให้สำเร็จตรงตามวัตถุประสงค์ที่ได้ตั้งไว้ และผลที่ได้รับเวลาการปฏิบัติงานจริงสามารถที่จะดำเนินการตามที่กำหนดได้

ปัญหาที่เกิดขึ้น

สัญญาณที่ส่งมายังเครื่องรับ GPS ไม่สามารถทะลุผ่านของแข็ง เช่นอาคาร ได้ดีเท่าที่ควร จึงต้องสายอากาศช่วยในการทดสอบในอาคาร และในการทดสอบการเคลื่อนที่ของรถเมื่อเวลาเคลื่อนที่ด้วยความเร็วพบส่วนข้อมูลที่เก็บลงในหน่วยความจำนั้นมีความคลาดเคลื่อน มีข้อมูลอื่นปะปนมาด้วย

ข้อเสนอแนะ

ในการนำไปใช้งานทางด้านธุรกิจนอกจากระบบต้องมีประสิทธิภาพที่ดีแล้วค่าใช้จ่ายก็เป็นส่วนสำคัญ หน่วยความจำควรมีขนาดใหญ่ เพียงพอสำหรับเก็บข้อมูลต่าง ที่ใช้เก็บข้อมูลการเดินทางด้วย ควรเพิ่มประสิทธิภาพการทำงานของโปรแกรมให้มีการจัดการหน่วยความจำให้ดีขึ้น เพื่อเพิ่มความเร็วของการทำงาน ส่วนของการแสดงผล สามารถพัฒนาให้การแสดงภาพของแผนที่แสดงออกมาได้สวยงามยิ่งขึ้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก

ส่วนโปรแกรมหลัก

```
#include "reg52.h"
#include "absacc.h"
#include "Function.c"
#include "LCD.c"
#include "I2C.c"
#include "EEPROM.c"
unsigned char xdata ch,BufGPS[64];
unsigned int i,j;
unsigned int Rec,Times;
sbit Gate = P3^4;
bit TFlag=0;
main()
{unsigned int RecNo;
  if(INT0 == 0){Rec = 0;}
  else{Rec = (Rd_EEPROM(62)*256)+Rd_EEPROM(63);}

  while(INT0==0){}Delay(300);

  Init_Serial(48);

  Light = 1;
  Init_LCD();
  Wr_String("Lat = ",1);
  Wr_String("Long = ",2);

  Delay(14000);
  // Tx_String("AT+IPR=4800");Tx_String("AT+IPR=4800");
  Tx_Byte(0x0D);Delay(1000);

  Tx_String("AT+CGATT=1");
  Tx_Byte(0x0D);Tx_Byte(0x0A);Delay(7000);

  Tx_String("AT#APNSERV=");Tx_Byte(0x22);Tx_String("WWW.DTAC.CO.TH");Tx
  _Byte(0x22);Tx_Byte(0x0D);Tx_Byte(0x0A);Delay(2000);
  Tx_String("AT#CONNECTIONSTART");
  Tx_Byte(0x0D);Tx_Byte(0x0A);Delay(3000);

  Tx_String("AT#TCPSERV=");Tx_Byte(0x22);Tx_String("161.246.65.110");Tx
  _Byte(0x22);Tx_Byte(0x0D);Tx_Byte(0x0A);Delay(3000);
  Tx_String("AT#TCPPOORT=1500");
  Tx_Byte(0x0D);Tx_Byte(0x0A);Delay(3000);
  Tx_String("AT#OTCP");
  Tx_Byte(0x0D);Tx_Byte(0x0A);Delay(5000);

  Gate= 0;
  Init_Timer0(15);
while(1)
{
  if(INT0 == 0)
  {
    RecNo = (Rd_EEPROM(62)*256)+Rd_EEPROM(63);
    for(i=0;i<RecNo;i++)
      { if(i<1024){Rp_EEPROM(i*64,61);}
        else{Rp_EEPROM2((i-1024)*64,61);}
      }
  }
}
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

        Delay(100);

        Tx_Byte('$');
        Tx_Byte('G');
        Tx_Byte('P');
        Tx_Byte('R');
        Tx_Byte('M');
        Tx_Byte('C');
        Tx_Byte(',');
    j = 0;
    while((j<62)&&(BufEEP[j]!= 0x0D))
        {Tx_Byte(BufEEP[j]);j++;}
        Tx_Byte(0x0D);
        Tx_Byte(0x0A);
    }
}
if(Rx_Byte()=='$')
if(Rx_Byte()=='G')
if(Rx_Byte()=='P')
if(Rx_Byte()=='R')
if(Rx_Byte()=='M')
if(Rx_Byte()=='C')
if(Rx_Byte()=='')
{ ch = Rx_Byte();
i = 0;
while((i<62)&&(ch != 0x0D))
{
    BufGPS[i] = ch;
    i++;
    ch = Rx_Byte();
}
    BufGPS[i] = ch;
    Goto_XY(1,5);

Wr_Char(BufGPS[13]);Wr_Char(BufGPS[14]);Wr_Char(BufGPS[15]);
Wr_Char(BufGPS[16]);Wr_Char(BufGPS[17]);Wr_Char(BufGPS[18]);
    Wr_Char(BufGPS[19]);Wr_Char(BufGPS[20]);
Wr_Char(BufGPS[21]);

    Goto_XY(2,6);

Wr_Char(BufGPS[25]);Wr_Char(BufGPS[26]);Wr_Char(BufGPS[27]);
Wr_Char(BufGPS[28]);Wr_Char(BufGPS[29]);Wr_Char(BufGPS[30]);
Wr_Char(BufGPS[31]);Wr_Char(BufGPS[32]);Wr_Char(BufGPS[33]);
    Wr_Char(BufGPS[34]);
    if(TFlag==1)
        {
        for(i=0;i<64;i++)
            {BufEEP[i] = BufGPS[i];
            }
        Delay(100);
        if(Rec < 1024){ Wp_EEPROM(Rec*64,61);}
            else{Wp_EEPROM2((Rec-
1024)*64,61);}
        Rec++;

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

        if (Rec>=2048) {Rec=2047;}
        Wr_EEPROM(62,Rec>>8);
        Wr_EEPROM(63,Rec & 0x00FF);
        TFlag = 0;
        Gate = 1;
        Tx_Byte('$');
        Tx_Byte('G');
        Tx_Byte('P');
        Tx_Byte('R');
        Tx_Byte('M');
        Tx_Byte('C');
        Tx_Byte(',');
        for(i=0;i<61;i++)
        {
if (BufEEP[i]==0x0D) {break;}else{Tx_Byte(BufEEP[i]);}
        }
        Tx_Byte(0x0D);
        Tx_Byte(0x0A);
        Gate = 0;
    }
}

}
}
void Timer0_Int(void) interrupt 1
{
    TH0 = _TH0;
    TLO = _TLO;
    Times++;
    if(Times == 15*5)
    {
        Times = 0;
        TFlag = 1;
    }
}

}

unsigned char idata BufEEP[64];
void Wr_EEPROM(unsigned int Addr,unsigned char Data)
{
    I2c_Start();
    I2c_Write(0xA0);
    I2c_Write(Addr >> 8);
    I2c_Write(Addr & 0x00FF);
    I2c_Write(Data);
    I2c_Stop();
    Delay(50);
}
}

```

```

unsigned char Rd_EEPROM(unsigned int Addr)
{ unsigned char Temp;

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

    I2c_Start();
    I2c_Write(0xA0);
    I2c_Write(Addr >> 8);
    I2c_Write(Addr & 0x00ff);
    I2c_Start();
    I2c_Write(0xA1);
    Temp = I2c_Read();
    I2c_Nack();
    I2c_Stop();
    return(Temp);
}

void Wp_EEPROM (unsigned int Addr,unsigned char Count)
{ unsigned char i;
  I2c_Start();
  I2c_Write(0xA0);
  I2c_Write(Addr >> 8);
  I2c_Write(Addr & 0x00ff);
  for (i=0;i<Count;i++)
  {
    I2c_Write(BufEEP[i]);
  }
  I2c_Stop();
  Delay(100);
}

void Rp_EEPROM(unsigned int Addr,unsigned char Count)
{unsigned char i;
  I2c_Start();
  I2c_Write(0xA0);
  I2c_Write(Addr >> 8);
  I2c_Write(Addr & 0x00ff);
  I2c_Start();
  I2c_Write(0xA1);
  for (i=0;i<Count-1;i++)
  { BufEEP[i] = I2c_Read();
    I2c_Ack();
  }
  BufEEP[Count-1] = I2c_Read();
  I2c_Nack();

  I2c_Stop();
}

//-----
void Wr_EEPROM2(unsigned int Addr,unsigned char Data)
{
  I2c_Start();
  I2c_Write(0xA2);
  I2c_Write(Addr >> 8);
  I2c_Write(Addr & 0x00FF);

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

    I2c_Write(Data);
    I2c_Stop();
    Delay(50);
}

unsigned char Rd_EEPROM2(unsigned int Addr)
{ unsigned char Temp;
  I2c_Start();
  I2c_Write(0xA2);
  I2c_Write(Addr >> 8);
  I2c_Write(Addr & 0x00ff);
  I2c_Start();
  I2c_Write(0xA3);
  Temp = I2c_Read();
  I2c_Nack();
  I2c_Stop();
  return(Temp);
}

void Wp_EEPROM2(unsigned int Addr,unsigned char Count)
{ unsigned char i;
  I2c_Start();
  I2c_Write(0xA2);
  I2c_Write(Addr >> 8);
  I2c_Write(Addr & 0x00ff);
  for (i=0;i<Count;i++)
  {
    I2c_Write(BufEEP[i]);
  }
  I2c_Stop();
  Delay(100);
}

void Rp_EEPROM2(unsigned int Addr,unsigned char Count)
{unsigned char i;
  I2c_Start();
  I2c_Write(0xA2);
  I2c_Write(Addr >> 8);
  I2c_Write(Addr & 0x00ff);
  I2c_Start();
  I2c_Write(0xA3);
  for (i=0;i<Count-1;i++)
  { BufEEP[i] = I2c_Read();
    I2c_Ack();
  }
  BufEEP[Count-1] = I2c_Read();
  I2c_Nack();

  I2c_Stop();

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

}

sbit SDA = P2^4;
sbit SCL = P2^3;

void I2c_Wait(void)
{ unsigned char i;
  for (i=0;i<30;i++) { }
}

void I2c_Clk(void)
{ SCL = 1;
  I2c_Wait();
  SCL = 0;
  I2c_Wait();
}

void I2c_Ack(void)
{ SDA = 0;
  I2c_Clk();
}

void I2c_Nack(void)
{ SDA = 1;
  I2c_Clk();
}

void I2c_Start(void)
{ SCL = 0;
  SDA = 1;
  SCL = 1;
  I2c_Wait();
  SDA = 0;
  I2c_Wait();
}

void I2c_Stop(void)
{ SCL = 0;
  SDA = 0;
  SCL = 1;
  I2c_Wait();
  SDA = 1;
  I2c_Wait();
}

void I2c_Write(unsigned char Data )
{ unsigned char i;

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

bit out;
    SCL = 0;
    for (i=0;i<8;i++)
    {
        out = Data & 0x80;
        Data = Data << 1;
        SDA = out;
        I2c_Clk();
    }
    SDA = 1;
    I2c_Clk();
}

```

```

unsigned char I2c_Read(void)
{ unsigned char Data,i;
  bit out;
  SCL = 0;
  SDA = 1;
  I2c_Wait();
  for (i=0;i<8;i++)
  {
    SCL = 1;
    I2c_Wait();
    out = SDA;
    Data = Data << 1;
    Data = Data | out;
    SCL = 0;
    I2c_Wait();
  }
  return(Data);
}

sbit EN_LCD = P0^3;
sbit RS_LCD = P0^2;
sbit BLIGHT = P0^1;
bit Light;

```

```

void Wr_Register(unsigned char Data)
{unsigned char Temp;
  Temp = Data;
  Temp = Temp | 0x0F;
  P0 = Temp;
  if (Light) { BLIGHT = 0;}
  RS_LCD = 0;
  EN_LCD = 0;
  Delay(1);
  EN_LCD = 1;
}

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

    Temp = Data;
    Temp = Temp << 4;
    Temp = Temp | 0x0F;
    Po = Temp;
    if (Light) { BLIGHT = 0;}
    RS_LCD = 0;
    EN_LCD = 0;
    Delay(1);
    EN_LCD = 1;

```

```

}
void Wr_Char(unsigned char Data)
{unsigned char Temp;

```

```

    Temp = Data;
    Temp = Temp | 0x0F;
    Po = Temp;
    if (Light) { BLIGHT = 0;}
    RS_LCD = 1;
    EN_LCD = 0;
    Delay(1);
    EN_LCD = 1;

```

```

    Temp = Data;
    Temp = Temp << 4;
    Temp = Temp | 0x0F;
    Po = Temp;
    if (Light) { BLIGHT = 0;}
    RS_LCD = 1;
    EN_LCD = 0;
    Delay(1);
    EN_LCD = 1;

```

```

}
void Init_LCD(void)
{

```

```

    Delay(200);
    Wr_Register(0x33);
    Wr_Register(0x32);
    Wr_Register(0x28);
    Wr_Register(0x0C);
    Wr_Register(0x06);
    Wr_Register(0x01);
    Delay(100);

```

```

}
void Goto_XY(unsigned char x,y)
{

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

    if(x == 1)
    {
        Wr_Register(0x80+(y-1));
    }
    if(x == 2)
    {
        Wr_Register(0xC0+(y-1));
    }
    if(x == 3)
    {
        Wr_Register(0x94+(y-1));
    }
    if(x == 4)
    {
        Wr_Register(0xD4+(y-1));
    }
}

void Wr_Page(unsigned char *dptr)
{unsigned char i;
  Goto_XY(1,1);
  for (i=0;i<20;i++)
  { Wr_Char(*dptr);
    dptr++; }
  Goto_XY(2,1);
  for (i=0;i<20;i++)
  { Wr_Char(*dptr);
    dptr++; }
}

void Wr_String(unsigned char *dptr,Row)
{unsigned char i,Count;
  Count = strlen(dptr);
  Goto_XY(Row,1);
  for (i=0;i<Count;i++)
  { Wr_Char(*dptr);
    dptr++; }
}

void Cursor_Off(void)
{
  Wr_Register(0x0C);
}

void Cursor_On(void)
{
  Wr_Register(0x0F);
}

void SlideNum(void)
{

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

BufDisp[6] = BufDisp[5];
BufDisp[5] = BufDisp[4];
BufDisp[4] = BufDisp[3];
BufDisp[3] = BufDisp[2];
BufDisp[2] = BufDisp[1];
BufDisp[1] = BufDisp[0];
}
void CountToLCD( unsigned long Count)
{
    BufDisp[6]=      (Count / 1000000)+0x30;
    if (BufDisp[6]==0x30) BufDisp[0] = 0x20;

    Count      = Count % 1000000;
    BufDisp[5] = (Count / 100000)+0x30;
    if ((BufDisp[5]==0x30) && (BufDisp[6]==0x20)) BufDisp[5] = 0x20;

    Count      = Count % 100000;
    BufDisp[4] = (Count / 10000)+0x30;
    if ((BufDisp[4]==0x30) && (BufDisp[5]==0x20)) BufDisp[4] = 0x20;

    Count      = Count % 10000;
    BufDisp[3] = (Count / 1000)+0x30;
    if ((BufDisp[3]==0x30) && (BufDisp[4]==0x20)) BufDisp[3] = 0x20;

    Count      = Count % 1000;
    BufDisp[2] = (Count / 100)+0x30;
    if ((BufDisp[2]==0x30) && (BufDisp[3]==0x20)) BufDisp[2] = 0x20;

    Count      = Count % 100;
    BufDisp[1] = (Count / 10)+0x30;
    if ((BufDisp[1]==0x30) && (BufDisp[2]==0x20)) BufDisp[1] = 0x20;

    Count      = Count % 10;
    BufDisp[0] = Count+0x30;
}
void ShowNum(unsigned long Count,unsigned char Index)
{unsigned char i;
  CountToLCD(Count);
  for(i=0;i<Index;i++)
  {
    Wr_Char(BufDisp[Index-(i+1)]);
  }
}
unsigned long GetNum(unsigned char Count,X,Y)
{unsigned char i,j,Key;
  unsigned long Sum;
  for(i=0;i<Count;i++){BufDisp[i] = 0x20;}
}

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

i = 0;
while(Key != 11)
{Goto_XY(X,Y+(Count-1));
  Cursor_On();
  Key = KeyPress();
  Sound();

  if((Key < 10)&&(i<Count))
  {  SlideNum();
    BufDisp[0]=Key;

    i++;
  }
  if(Key == 10)
  {  i = 0;
    for(j=0;j<Count;j++){BufDisp[j] = 0x20;}
  }
  Cursor_Off();
  Goto_XY(X,Y);
  for(j=0;j<Count;j++)
  {
    if(BufDisp[Count-(j+1)]!=0x20){Wr_Char(BufDisp[Count-(j+1)+0x30]);}
    else
{Wr_Char(BufDisp[Count-(j+1)]);}
  }

  Key_Off();
}

Sum = 0;
for(i=0;i<Count;i++)
{
  if(BufDisp[Count-(i+1)]==0x20){BufDisp[Count-(i+1)]=0 ;}
  Sum = (Sum*10)+(BufDisp[Count-(i+1)]);
}
Cursor_Off();
return(Sum);
}

```

```

unsigned char idata BufEEP[64];
void Wr_EEPROM(unsigned int Addr,unsigned char Data)
{
  I2c_Start();
  I2c_Write(0xA0);
  I2c_Write(Addr >> 8);
  I2c_Write(Addr & 0x00FF);
  I2c_Write(Data);
  I2c_Stop();
  Delay(50);
}

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

}

unsigned char Rd_EEPROM(unsigned int Addr)
{ unsigned char Temp;
  I2c_Start();
  I2c_Write(0xA0);
  I2c_Write(Addr >> 8);
  I2c_Write(Addr & 0x00ff);
  I2c_Start();
  I2c_Write(0xA1);
  Temp = I2c_Read();
  I2c_Nack();
  I2c_Stop();
  return(Temp);
}

void Wp_EEPROM (unsigned int Addr,unsigned char Count)
{ unsigned char i;
  I2c_Start();
  I2c_Write(0xA0);
  I2c_Write(Addr >> 8);
  I2c_Write(Addr & 0x00ff);
  for (i=0;i<Count;i++)
  {
    I2c_Write(BufEEP[i]);
  }
  I2c_Stop();
  Delay(100);
}

void Rp_EEPROM(unsigned int Addr,unsigned char Count)
{unsigned char i;
  I2c_Start();
  I2c_Write(0xA0);
  I2c_Write(Addr >> 8);
  I2c_Write(Addr & 0x00ff);
  I2c_Start();
  I2c_Write(0xA1);
  for (i=0;i<Count-1;i++)
  { BufEEP[i] = I2c_Read();
    I2c_Ack();
  }
  BufEEP[Count-1] = I2c_Read();
  I2c_Nack();

  I2c_Stop();
}

//-----
void Wr_EEPROM2(unsigned int Addr,unsigned char Data)
{
  I2c_Start();
  I2c_Write(0xA2);

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

    I2c_Write(Addr >> 8);
    I2c_Write(Addr & 0x00FF);
    I2c_Write(Data);
    I2c_Stop();
    Delay(50);
}

unsigned char  Rd_EEPROM2(unsigned int  Addr)
{  unsigned char Temp;
    I2c_Start();
    I2c_Write(0xA2);
    I2c_Write(Addr >> 8);
    I2c_Write(Addr & 0x00ff);
    I2c_Start();
    I2c_Write(0xA3);
    Temp = I2c_Read();
    I2c_Nack();
    I2c_Stop();
    return(Temp);
}

void  Wp_EEPROM2 (unsigned  int Addr,unsigned char Count)
{  unsigned char i;
    I2c_Start();
    I2c_Write(0xA2);
    I2c_Write(Addr >> 8);
    I2c_Write(Addr & 0x00ff);
    for (i=0;i<Count;i++)
    {
        I2c_Write(BufEEP[i]);
    }
    I2c_Stop();
    Delay(100);
}

void  Rp_EEPROM2(unsigned int Addr,unsigned char Count)
{unsigned char i;
    I2c_Start();
    I2c_Write(0xA2);
    I2c_Write(Addr >> 8);
    I2c_Write(Addr & 0x00ff);
    I2c_Start();
    I2c_Write(0xA3);
    for (i=0;i<Count-1;i++)
        {  BufEEP[i] = I2c_Read();
            I2c_Ack();
        }
    BufEEP[Count-1] = I2c_Read();
    I2c_Nack();

    I2c_Stop();
}

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

#include "string.h"
sbit Row0      = P1^0;
sbit Row1      = P1^1;
sbit Row2      = P1^2;
sbit Row3      = P1^3;
unsigned char data BufDisp[16];
void Wait(unsigned int x)
{unsigned int i;
  for(i=0;i<x;i++) {}
}
sbit Piezo     = P3^6;
void Sound(void)
{unsigned char i;
  for (i=0;i<50;i++)
  {
    Piezo = ~Piezo;
    Wait(50);
  }
}
sbit Piezo2    = P2^5;
void SoundSmart(void)
{unsigned char i;
  for (i=0;i<50;i++)
  {
    Piezo2 = ~Piezo2;
    Wait(50);
  }
}
/* ----- Scan Key Sw. ----- */
unsigned char Scan_Key(void)
{
  P1 = 0xEF;
  if (Row0 == 0) {return(11);}
  if (Row1 == 0) {return(9);}
  if (Row2 == 0) {return(6);}
  if (Row3 == 0) {return(3);}

  P1 = 0xDF;
  if (Row0 == 0) {return(0);}
  if (Row1 == 0) {return(8);}
  if (Row2 == 0) {return(5);}
  if (Row3 == 0) {return(2);}

  P1 = 0xBF;
  if (Row0 == 0) {return(10);}
  if (Row1 == 0) {return(7);}
  if (Row2 == 0) {return(4);}
  if (Row3 == 0) {return(1);}
  return(0xff);
}

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

P1    =    0x7F;
if (Row0 == 0) {return(15);}
if (Row1 == 0) {return(14);}
if (Row2 == 0) {return(13);}
if (Row3 == 0) {return(12);}
return(0xff);

}
/* --- Delay ----- */
void Delay(unsigned int x )
{ unsigned int  i,j;
  for (i=0;i<x;i++)
  {
    for (j=0;j<115;j++) { }
  }
}

void Key_Off(void)
{
  while (Scan_Key() != 0xFF) { }
  Delay(50);
}

unsigned char KeyPress(void)
{unsigned char Key;
  while (Scan_Key() == 0xFF) { }
  Key = Scan_Key();
  Key_Off();
  return(Key);
}

code unsigned char BufSEG[] = {0xC0,0xF9,0xA4,0xB0,0x99,0x92,
                               0x82,0xF8,0x80,0x90,0xFF,0xC6};

void Display (void)
{
  P2 = 0xFF; /* Display off */
  P0 = BufSEG[BufDisp[3]];
  P2 = 0xEF; /* Scan 7 - Segment */
  Delay(4);

  P2 = 0xFF; /* Display off */
  P0 = BufSEG[BufDisp[2]];
  P2 = 0xDF; /* Scan 7 - Segment */
  Delay(4);

  P2 = 0xFF; /* Display off */
  P0 = BufSEG[BufDisp[1]];
  P2 = 0xBF; /* Scan 7 - Segment */
  Delay(4);
}

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

    P2 = 0xFF;          /* Display off */
    P0 = BufSEG[BufDisp[0]];
    P2 = 0x7F;        /* Scan 7 - Segment */
    Delay(4);
    P2 = 0xFF;
}

void CountToDisp( unsigned long Count)
{
    BufDisp[6] =      Count / 1000000;
    if(BufDisp[6] == 0) BufDisp[6] = 10;

    Count  = Count % 1000000;
    BufDisp[5] =      Count / 100000;

    Count  = Count % 100000;
    BufDisp[4] =      Count / 10000;

    Count  = Count % 10000;
    BufDisp[3] =      Count / 1000;

    Count  = Count % 1000;
    BufDisp[2] =      Count / 100;

    Count  = Count % 100;
    BufDisp[1] =      Count / 10;

    BufDisp[0] =      Count % 10;
}

void CountToLCD2( unsigned long Count)
{
    Count  = Count % 1000;
    BufDisp[2] =      (Count / 100)+0x30;
    if((BufDisp[2] == 0x30) ) BufDisp[2] = 0x20;

    Count  = Count % 100;
    BufDisp[1] =      (Count / 10)+0x30;
    if((BufDisp[1] == 0x30) && (BufDisp[2] == 0x20))
BufDisp[1] = 0x20;

    Count  = Count % 10;

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

        BufDisp[0] =      Count+0x30;
    }
void ShowVal(unsigned int Num)
{
    CountToDisp(Num);
    Display();
}
void Tx_Byte(unsigned char Data)
{
    TI    =    1;
    SBUF  =    Data;
    TI    =    0;
    while (!TI) {}
    TI    =    0;
}
unsigned char Rx_Byte(void)
{unsigned char Data;
    while (!RI) {}
    RI    =    0;
    Data  =    SBUF;
    return(Data);
}
void Init_Serial(unsigned char br)
{
    SCON  =    0x50;
    TMOD  =    (TMOD & 0x0F)+0x20;
    TH1   =    0xFD;
    TR1   =    1;
    EA    =    1;

    if (br == 48)    TH1 = 0xFA;
    if (br == 96)    TH1 = 0xFD;
    if (br == 192)  {PCON = 0x80;} else {PCON = 0x00;}
}
void Tx_String(unsigned char *dptr)
{unsigned char i,Count;
    Count = strlen(dptr);
    for(i=0;i<Count;i++)
    {
        Tx_Byte(*dptr);
        dptr++;
    }
}
unsigned char _TH0,_TLO;
void Init_Timer0(unsigned int Freq)
{unsigned int Count;

    Count = 65536 - (921600 / Freq);
    TH0  =  Count >> 8;
    TLO  =  Count & 0xFF;
    _TH0 =  TH0;
}

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

    _TLO = TLO;
    TMOD = (TMOD & 0xF0)+0x01;
    TR0 = 1;
    ETO = 1;
    EA = 1;
}
void Timer0_IntOn(void)
{ ETO = 1;
}
void Timer0_IntOff(void)
{
    ETO = 0;
}
void Timer0_On(void)
{
    TR0 = 1;
}
void Timer0_Off(void)
{
    TR0 = 0;
}
void Serial_IntOn(void)
{ ES = 1;}
void Serial_IntOff(void)
{ ES = 0;
}

```



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้