

สำนักหอสมุดกลาง พระจอมเกล้าลาดกระบัง

ระบบตรวจสอบสถานะและจำนวนสินค้าของผู้หยอดเหรียญอัตโนมัติ

STATUS AND QUANTITY CHECKING SYSTEM

FOR VENDING MACHINE



T104068



เลขหมู่.....
เลขทะเบียน...104068
วัน,เดือน,ปี... 2 8 ต.ค. 2552

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

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต
ภาควิชาวิศวกรรมคอมพิวเตอร์ คณะวิศวกรรมศาสตร์
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
ปีการศึกษา 2551

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ปริญญาโทปีการศึกษา 2551

ภาควิชาวิศวกรรมคอมพิวเตอร์

คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เรื่อง ระบบตรวจสอบสถานะและจำนวนสินค้าของตู้หยอดเหรียญอัตโนมัติ

STATUS AND QUANTITY CHECKING SYSTEM FOR VENDING MACHINE

ผู้จัดทำ

1. นายสุทธิพงษ์ ศรีเกษม รหัสนักศึกษา 49015275
2. นายสมมาตร เพ็ชรชำนาญ รหัสนักศึกษา 49015307
3. นายอนูชิต จิงศรีพิชญ รหัสนักศึกษา 49015313



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ระบบตรวจสอบสถานะและจำนวนสินค้าของผู้หยุดเหรียญอัตโนมัติ

นายสุทธิพงษ์	ศรีเกษม	49015275
นายสมมาตร	เพ็ชรชำนาญ	49015307
นายอนุชิต	จิ่งศรีพิชญ์	49015313
อาจารย์ เจริญ	วงษ์หุ้มเย็น	อาจารย์ที่ปรึกษา
อาจารย์ จิระศักดิ์	สิทธิกร	อาจารย์ที่ปรึกษาร่วม
ปีการศึกษา 2551		

บทคัดย่อ

เนื่องด้วยในปัจจุบันได้มีการนำเทคโนโลยีระบบสมองกลฝังตัวไปประยุกต์ใช้ในงานด้านต่างๆ อย่างแพร่หลาย เช่น ใช้ในอุปกรณ์อิเล็กทรอนิกส์ต่างๆ ใช้ในรถยนต์ซึ่งในปัจจุบันกำลังได้รับความนิยมเป็นอย่างสูง ฯลฯ ซึ่งการนำเทคโนโลยีระบบสมองกลฝังตัวเข้ามาใช้งานนี้ทำให้อุปกรณ์ต่างๆ มีประสิทธิภาพในการทำงานที่สูงขึ้น ปัจจุบันเทคโนโลยีระบบสมองกลฝังตัวกำลังได้รับความนิยมจากผู้พัฒนาเป็นอย่างมาก ทำให้เทคโนโลยีระบบสมองกลฝังตัวถูกพัฒนาไปอย่างรวดเร็วและมีประสิทธิภาพที่สูงขึ้นรวมถึงประสิทธิภาพในการใช้งานที่หลากหลายมากขึ้น ในอนาคตเราอาจจะได้เห็นการใช้งานเทคโนโลยีระบบสมองกลฝังตัวในทุกผลิตภัณฑ์สินค้าเลยก็อาจจะเป็นไปได้

โครงการนี้ได้นำเทคโนโลยีระบบสมองกลฝังตัวไปประยุกต์ใช้งานกับเครื่องขายสินค้าหยุดเหรียญซึ่งจะทำหน้าที่ในการส่งข้อมูลของสินค้าภายในเครื่อง บอกสถานะสินค้าให้กับผู้ให้บริการทราบและนำข้อมูลดังกล่าวไปใช้ประโยชน์ เช่น การจัดทำสถิติในการให้บริการขายสินค้า ซึ่งจะทำให้ผู้ให้บริการทราบข้อมูลเกี่ยวกับการขายและข้อมูลต่างๆ รู้ว่าในหนึ่งวันหรือหนึ่งปีมีอะไรถูกขายไปบ้าง แยกเป็นอะไร สินค้าตัวไหนขายดี ณ ช่วงเวลาใด จากจุดนี้ผู้ให้บริการสามารถรู้แนวทางสำหรับวางแผนการตลาดที่ดีได้ เป็นต้น นอกจากนี้ยังได้มีการใช้เว็บแอปพลิเคชันมาใช้งานรวมด้วย ซึ่งผู้ให้บริการจะสามารถดูแลการทำงานในธุรกิจได้จากทุกที่ ทุกเวลาที่สามารถเชื่อมต่ออินเทอร์เน็ต เพราะสามารถเข้าไปดูสถานการณ์การทำงานของระบบได้ผ่านเว็บเบราว์เซอร์ ทำให้ผู้ให้บริการทำงานอย่างมีประสิทธิภาพและสะดวกสบายมากยิ่งขึ้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดลอกเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Status and quantity checking system for vending machine

Mr.Chutipong	Srikasem	49015275
Mr.Sommart	Bheanchumnan	49015307
Mr.Anuchit	Jungsripisanu	49015313
Mr.Charoen	Vongchumyen	Advisor
Mr.Jirasak	Sittigorn	Co-Advisor

Academic Year 2008

ABSTRACT

At the present, Embedded System Technology has been applied to various kind of work such as electronic equipment, automobiles, which are being popular nowadays. Bringing this technology into work gives equipments higher efficiency. Now, Embedded System Technology is very popular among developer. Therefore, this technology was rapidly developed and has more efficiency in other usage. In the future, all products maybe have embedded system inside.

In this research, we apply the embedded system technology to vending machine by putting microcontroller inside for keeping quantity and detail of goods in this machine. Then the information would be reported to service provider, who uses this thing for, for example, making the trading. Service providers would know the information about goods trading, which was ordered daily, monthly, or yearly, or know which goods were selling well by the time. From this point, service providers are able to plan the marketing strategy or logistic management in easier way. Moreover, web application is also used together, which service providers can supervise business anywhere and anytime just connect to the internet, because we can look into process of system via web browser. That is why the vendors are able to work freely.

กิตติกรรมประกาศ

ปริญญาานิพนธ์ฉบับนี้สำเร็จลุล่วงได้อย่างดี ด้วยคำแนะนำ คำปรึกษาและคอยดูแลจากหลายๆฝ่ายด้วยกัน โดยเฉพาะอย่างยิ่งอาจารย์ที่ปรึกษาทั้งสี่ท่านที่ให้โอกาสข้าพเจ้าได้ทำปริญญาานิพนธ์ฉบับนี้ คอยให้ความเอาใจใส่สอบถามความคืบหน้า ให้คำแนะนำและความช่วยเหลือเสมอ มาคือ อาจารย์ เจริญ วงษ์ชุ่มเย็น ซึ่งต้องกราบขอบพระคุณอย่างสูง

ขอขอบคุณภาควิชาวิศวกรรมคอมพิวเตอร์ที่ได้จัดเตรียมสิ่งอำนวยความสะดวก เพื่อให้การวิจัยและพัฒนา โปรแกรมเป็นไปได้อย่างสะดวกและรวดเร็ว โดยเฉพาะอินเทอร์เน็ตความเร็วสูงที่มีไว้ให้บริการสำหรับการค้นคว้าหาความรู้ต่างๆ ซึ่งทำยสุดแล้วได้ประกอบกันเป็นส่วนหนึ่งของโครงการนี้ และขอขอบคุณรุ่นพี่ที่ให้คำปรึกษาในเรื่องต่างๆที่มีปัญหา รวมถึงรุ่นน้องที่เป็นกำลังใจให้ตลอดเวลา

และที่ขาดไม่ได้เลยต้องขอขอบคุณบุคคลที่มีความสำคัญที่สุดในชีวิตที่ทำให้ข้าพเจ้ามีวันนี้ นั่นคือ บิดา มารดาและครอบครัวอันเป็นที่เคารพรัก ซึ่งได้ให้การเลี้ยงดู คอยอบรมสั่งสอนข้าพเจ้ามาเป็นอย่างดี พร้อมทั้งให้โอกาสในการศึกษาอย่างเต็มที่ อีกทั้งยังให้กำลังใจ มอบความรักให้เสมอมาอย่างหาที่เปรียบไม่ได้ ข้าพเจ้าขอกราบขอบพระคุณมา ณ ที่นี้ด้วย

ชุทธิพงษ์ ศรีเกษม
สมมาตร เพ็ชรชำนาญ
อนุชิต จิ่งศรีพิษณ

สารบัญ

หน้า

บทคัดย่อภาษาไทย.....	I
บทคัดย่อภาษาอังกฤษ.....	II
กิตติกรรมประกาศ.....	III
สารบัญ.....	IV
สารบัญตาราง.....	IX
สารบัญภาพ.....	X
บทที่ 1 บทนำ.....	1
1.1 ความเป็นมาของปัญหา.....	1
1.2 วัตถุประสงค์ของโครงการ.....	1
1.3 ประโยชน์ที่คาดว่าจะได้รับ.....	1
1.4 ขอบเขตของโครงการ.....	2
1.5 ขั้นตอนการทำงาน.....	2
บทที่ 2 ทฤษฎีที่เกี่ยวข้อง.....	3
2.1 ไมโครคอนโทรลเลอร์ PIC16F877.....	3
2.1.1 โครงสร้างของไมโครคอนโทรลเลอร์ PIC16F877.....	3
2.1.2 คุณสมบัติทางเทคนิคของ PIC16F87x/87xA.....	3
2.1.2.1 คุณสมบัติหลัก.....	3
2.1.2.2 คุณสมบัติพิเศษเพิ่มเติม.....	5
2.1.3 ซีซีเอส ซี คอมไพเลอร์ (CCS C Compiler).....	6
2.1.3.1 การใช้งานพอร์ตอินพุตเอาต์พุตดิจิทัลของไมโครคอนโทรลเลอร์ PIC ด้วยโปรแกรมภาษา.....	6
2.1.3.2 การใช้งานไทมเมอร์/เคาเตอร์ ภายในไมโครคอนโทรลเลอร์ PIC.....	9
2.2 จีพีอาร์เอส โมดูล.....	15
2.2.1 อีที-จีเอสเอ็ม ซิม300ซีแซต.....	15
2.2.2 คุณสมบัติของบอร์ด อีที-จีเอสเอ็ม ซิม300ซีแซต วี1.0.....	15
2.2.3 การสั่ง เปิด และ ปิด การทำงานของ โมดูล.....	16
2.2.4 การติดต่อสื่อสารกับ โมดูล ซิม300ซีแซต.....	18
2.2.5 ตัวอย่างการใช้งาน เอที คอมมานเพื่อสั่งงาน โมดูล ซิม300ซีแซต.....	18

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ(ต่อ)

หน้า

2.2.6 การกำหนดโฟลคอนโทรล(Flow Control).....	19
2.3 General Packet Radio Service (จีพีอาร์เอส).....	20
2.3.1 คุณสมบัติเด่นหลักๆ ของระบบ จีพีอาร์เอส มีดังนี้.....	21
2.3.2 ประโยชน์ของ จีพีอาร์เอส.....	21
2.3.3 รูปแบบการให้บริการของ จีพีอาร์เอส.....	22
2.4 การสื่อสารข้อมูล (Data Transmission).....	23
2.4.1 ชนิดของสัญญาณทางอิเล็กทรอนิกส์.....	23
2.4.1.1 สัญญาณแบบอนาลอก (Analog signal).....	23
2.4.1.2 สัญญาณแบบดิจิทัล (Digital signal).....	23
2.4.2 ทิศทางการสื่อสารข้อมูล.....	24
2.4.2.1 แบบทิศทางเดียว (Simplex).....	24
2.4.2.2 แบบกึ่งสองทิศทาง (Half Duplex).....	24
2.4.2.3 แบบสองทิศทาง (Full Duplex).....	24
2.4.3 ประเภทของการสื่อสารข้อมูล.....	25
2.4.3.1 การสื่อสารข้อมูลแบบขนาน (Parallel Transmission).....	25
2.4.3.2 การสื่อสารข้อมูลแบบอนุกรม (Serial Transmission).....	25
2.4.4 มาตรฐานการส่งข้อมูลแบบอนุกรม.....	29
2.4.4.1 พอร์ตอนุกรม อาร์เอส 232 (Serial Port).....	29
2.4.4.2 การเชื่อมต่ออุปกรณ์ภายนอกเข้ากับคอมพิวเตอร์ด้วยสาย ดีบี9.....	30
2.4.4.3 การทำงานของขาสัญญาณ ดีบี9.....	31
2.4.4.4 ไอซี แม็ค232, ไอซีแอล232.....	31
2.5 ระบบฐานข้อมูล (Database System).....	32
2.5.1 การออกแบบฐานข้อมูลด้วยอี - อาร์โมเดล (Entity Relationship Model).....	33
2.5.2 ขั้นตอนในการออกแบบฐานข้อมูลด้วย อี - อาร์โมเดล.....	33
2.5.2.1 การศึกษารายละเอียดและลักษณะหน้าที่งานของระบบ.....	33
2.5.2.2 การกำหนดเอนทิตีที่ควรมีในระบบฐานข้อมูล.....	33
2.5.2.3 การกำหนดความสัมพันธ์ระหว่างเอนทิตี.....	33
2.5.2.4 การกำหนดคุณลักษณะของเอนทิตี.....	33
2.5.2.5 การกำหนดคีย์หลัก (Primary key) ของแต่ละเอนทิตี.....	33

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ(ต่อ)

หน้า

2.5.2.6 การนำสัญลักษณ์ที่ใช้ใน อี – อาร์โมเดล มาอธิบายความสัมพันธ์ระหว่างข้อมูล.....	34
2.5.3 โครงสร้างของภาษาเอสคิวแอล (SQL).....	34
2.5.4 ประเภทของคำสั่งของภาษา เอสคิวแอล.....	34
2.5.4.1 ภาษาสำหรับการนิยามข้อมูล (Data Definition Language: DDL).....	34
2.5.4.2 ภาษาสำหรับการจัดการข้อมูล (Data Manipulation Language: DML).....	35
2.5.4.3 ภาษาควบคุม (Data Control Language: DCL).....	35
2.5.5 ชนิดของข้อมูลที่ใช้ในภาษา เอสคิวแอล.....	35
2.5.5.1 ตัวหนังสือ (Character).....	35
2.5.5.2 จำนวนตัวเลข.....	36
2.5.5.3 ข้อมูลในลักษณะอื่น ๆ.....	36
2.5.6 ลักษณะการใช้งานของภาษา เอสคิวแอล.....	36
2.5.6.1 ภาษา เอสคิวแอล ที่ได้ตอบได้.....	36
2.5.6.2 ภาษา เอสคิวแอล ที่ฝังในโปรแกรม เป็นภาษา เอสคิวแอล.....	36
2.5.7 การบันทึกข้อมูล, การปรับปรุงข้อมูล และการลบข้อมูล.....	37
2.5.7.1 คำสั่งการเพิ่มข้อมูล (INSERT).....	38
2.5.7.2 คำสั่งปรับปรุงแถวข้อมูล (UPDATE).....	38
2.5.7.3 คำสั่งการลบข้อมูลทั้งแถว (DELETE).....	39
2.5.8 การเรียกค้นข้อมูล (SELECT).....	39
2.5.8.1 การเรียกค้นข้อมูลจากตารางหลายตารางในภาษา เอสคิวแอล.....	39
2.5.8.2 การเรียกดูข้อมูลแบบซ้อนกัน (Subqueries).....	39
2.6 ทีซีพี/ไอพี โพรโตคอล.....	40
2.6.1 โครงสร้างของ ทีซีพี/ไอพี แต่ละเลเยอร์.....	40
2.6.1.1 ชั้นโฮสต์-เครือข่าย (Host-to-Network Layer).....	41
2.6.1.2 ชั้นสื่อสารอินเทอร์เน็ต (The Internet Layer).....	41
2.6.1.3 ชั้นสื่อสารนำส่งข้อมูล (Transport Layer).....	43
2.6.1.4 ชั้นสื่อสารการประยุกต์ (Application Layer).....	49
2.7 World Wide Web (WWW).....	49
2.7.1 การทำงานของเว็บเพจ.....	50

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ(ต่อ)

	หน้า
บทที่ 3 การสร้างและการออกแบบระบบ	53
3.1 ส่วน เครื่องหยอดเหรียญ	54
3.1.1 ตัวควบคุม สเตปปีงมอเตอร์	55
3.1.2 ตัวแสดงผล 7 เซกเมนต์	56
3.1.3 การออกแบบวงจรควบคุม จีพีอาร์เอส โมดูล	58
3.1.4 วงจรแหล่งจ่ายไฟ	59
3.1.5 วงจรปุ่มกด แมทริกซ์	59
3.1.6 วงจรแอลซีดี โมดูล	60
3.1.7 วงจร อาร์เอส 232	61
3.1.8 วงจร ดีเอส1307	61
3.2 การสื่อสารระหว่างตัวควบคุม จีพีอาร์เอส โมดูล และ เซิร์ฟเวอร์	63
3.2.1 รูปแบบการสื่อสารไป เซิร์ฟเวอร์ เมื่อมีการเติมสินค้า	63
3.2.2 รูปแบบการสื่อสารไป เซิร์ฟเวอร์ เมื่อมีการซื้อ	63
3.3 ส่วน ฐานข้อมูล และ เว็บแอปพลิเคชัน	65
3.3.1 Entity vending	65
3.3.2 Entity goods_store	65
3.3.3 Entity slot	66
3.3.4 Entity sale	66
3.3.5 Entity fill_goods	66
3.3.6 Entity real_fill	66
3.3.7 Entity truck	67
3.3.8 Entity user	67
3.3.9 การออกแบบเว็บแอปพลิเคชัน	67
3.3.10 ผู้ใช้งานที่ไม่ต้อง ล็อกอิน เข้าสู่ระบบ	68
3.3.11 สโตร์(Store)	69
3.3.12 เมเนเจอร์(Manager)	70
บทที่ 4 การทดลองและผลการทดลอง	71
4.1 การทดลองการใช้งานเครื่องหยอดเหรียญ	71

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ(ต่อ)

	หน้า
4.2 การติดตั้งสวิตช์ลงบน เครื่องหยอดเหรียญ.....	72
4.3 ส่งข้อมูลผ่าน จีพีอาร์เอส ด้วยคำสั่ง เอที-คอมมาน.....	73
4.4 การทดลองเชื่อมต่อไมโครคอนโทรลเลอร์กับ จีพีอาร์เอส โมดูล.....	75
4.5 การทดลองรับข้อมูลของฝั่งเซิร์ฟเวอร์.....	78
4.6 ผลการทดลองการทำงานของระบบ.....	81
บทที่ 5 สรุปและวิจารณ์.....	83
5.1 การดำเนินงาน.....	83
5.2 ผลการดำเนินงาน.....	83
5.3 ปัญหาและอุปสรรค.....	84
5.3.1 ปัญหาและอุปสรรคในการทำงาน.....	84
5.3.2 แนวทางการแก้ไข.....	84
5.4 ข้อสรุปและเสนอแนะ.....	84
บรรณานุกรม.....	86
ภาคผนวก ก.....	87
ภาคผนวก ข.....	111
ภาคผนวก ค.....	125

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญตาราง

ตารางที่	หน้า
2.1 ตัวแปรการทำการรวมกับไทมเมอร์ 0.....	10
2.2 ตัวแปรการทำการรวมกับไทมเมอร์ 1.....	11
2.3 ตัวแปรการทำการรวมกับไทมเมอร์ 2.....	12
2.4 การกำหนดสัญญาณนาฬิกาของโมดูลแปลงสัญญาณอนาล็อกเป็นดิจิตอล.....	13
2.5 สถานะของ แอลอีดี ในโหมดต่างๆ.....	17
2.6 การใช้งานเอที คอมมาน.....	19
2.7 โครงสร้างขาของพอร์ตอนุกรม ดิบี 9.....	30
4.1 ผลการทดลองในการส่งข้อมูลสินค้า.....	81



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญญภาพ

รูปที่	หน้า
2.1 โครงสร้างของไมโครคอนโทรลเลอร์ PIC16F877	5
2.2 จีพีอาร์เอส โมดูล รุ่น อีที-จีเอสเอ็ม ซิม300แซต	15
2.3 จีพีอาร์เอส โมดูล อยู่สถานะพร้อมใช้งาน	17
2.4 จีพีอาร์เอส โมดูล พร้อมใช้งาน	17
2.5 สัญญาณอนาลอก	23
2.6 สัญญาณดิจิตอล	24
2.7 การสื่อสารข้อมูลแบบทิศทางเดียว	24
2.8 การสื่อสารข้อมูลแบบกึ่งสองทิศทาง	24
2.9 การสื่อสารข้อมูลแบบทิศทางเดียว	24
2.10 การส่งข้อมูลแบบขนาน	25
2.11 การส่งข้อมูลแบบอนุกรม	26
2.12 การสื่อสารแบบอะซิงโครนัสที่ไม่ได้ใช้พาร์ตีบิต	26
2.13 การสื่อสารแบบอะซิงโครนัสที่ใช้พาร์ตีบิต	27
2.14 การสื่อสารแบบซิงโครนัส	27
2.15 ตัวอย่างการใช้อักขระซิงในการสื่อสารแบบซิงโครนัส	28
2.16 ตัวอย่างการใช้อักขระซิง 2 ตัวในการสื่อสารแบบซิงโครนัส	28
2.17 การตัดแถวของบิตออกเป็นกลุ่มๆ ละ 8 บิต	29
2.18 พอร์ตอนุกรม ดิบี 9 ตัวผู้ (Male)	29
2.19 พอร์ตอนุกรม ดิบี 9 ตัวเมีย (Female)	30
2.20 ขาของพอร์ตอนุกรม ดิบี 9	30
2.21 การเชื่อมต่ออุปกรณ์ภายนอกผ่าน ดิบี 9 แบบ เนา โมเต็ม	30
2.22 การต่ออุปกรณ์ภายนอกผ่าน ดิบี 9 แบบ 3 เส้น	31
2.23 ตำแหน่งขาของไอซี แม็ค232	31
2.24 การใช้งาน ไอซี แม็ค232	32
2.25 โครงสร้าง ทีซีพี/ไอพี	40
2.26 ไอพี เฮคเตอร์	41
2.27 ไอซีเอ็มพี เฮคเตอร์	43
2.28 ยูดีพี เฮคเตอร์	44
2.29 ทีซีพี เฮคเตอร์	45

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญญภาพ (ต่อ)

รูปที่	หน้า
2.30 ลำดับขั้นตอนการส่ง ทีซีพี เซกเมนต์.....	46
2.31 การเชื่อมต่อระหว่างซ็อกเก็ตคู่เดียวกัน.....	46
2.32 ขั้นตอนการเริ่มต้นการทำงานจากโฮสต์ 1 ไปยังโฮสต์ 2.....	48
2.33 เหตุการณ์ที่แพ็กเก็ต ทีพีดียู.....	49
2.34 การติดต่อระหว่าง ไคลเอ็นต์ และเซิร์ฟเวอร์.....	50
2.35 การทำงานของเว็บเพจ.....	51
2.36 การทำงานของเว็บเพจที่ฝังสคริปต์ภาษา พีเอสพี.....	51
3.1 ระบบโดยรวม.....	53
3.2 การออกแบบแบ่งเป็น 3 ส่วน.....	54
3.3 การควบคุมส่วนต่างๆของ เครื่องหยุดเหรียญ.....	54
3.4 วงจรขับสเตปปีงมอเตอร์ 1.....	55
3.5 วงจรขับสเตปปีงมอเตอร์ 2.....	56
3.6 การต่อวงจร 7-เซกเมนต์.....	57
3.7 การติดตั้ง จีพีอาร์เอส โมดูล ร่วมกับ เครื่องหยุดเหรียญ.....	58
3.8 การควบคุมส่วนต่างๆด้วยไมโครคอนโทรลเลอร์.....	58
3.9 การต่อการควบคุม จีพีอาร์เอส โมดูล.....	59
3.10 วงจรการต่อใช้งานแหล่งจ่ายไฟ 5 โวลท์.....	59
3.11 วงจรปุ่มกด แมทริกซ์.....	60
3.12 การต่อขาเข้ากับ แอลซีดี โมดูล.....	60
3.13 การต่ออาร์เอส 232เข้ากับโมดูล.....	61
3.14 วงจรดีเอส 1307.....	61
3.15 การต่อวงจรรวมกับไมโครคอนโทรลเลอร์.....	62
3.16 การประกอบเสร็จของ จีพีอาร์เอส โมดูล.....	62
3.17 แผนภาพลำดับ ในกรณีเชื่อมต่อกับเซิร์ฟเวอร์ได้.....	64
3.18 แผนภาพลำดับ ในกรณีเชื่อมต่อกับเซิร์ฟเวอร์ไม่ได้.....	64
3.19 Data-Base schema.....	65
3.20 usecase ของ user.....	68
3.21 usecase administrator.....	68
3.22 usecase ของ store.....	69

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญญภาพ (ต่อ)

รูปที่	หน้า
3.23 usecase ของ manager.....	70
4.1 เครื่องหยุดเหรียญ.....	71
4.2 สินค้าเมื่อมีการกดสวิตช์.....	72
4.3 การติดตั้งสวิตช์.....	73
4.4 การเชื่อมต่อ จีพีอาร์เอส โมดูล กับ ไมโครคอนโทรลเลอร์.....	75
4.5 การตั้งค่า จีพีอาร์เอส โมดูล ผ่านไมโครคอนโทรลเลอร์.....	76
4.6 จำนวนสินค้าในแต่ละช่อง.....	76
4.7 การส่งข้อมูลผ่าน จีพีอาร์เอส โมดูล.....	77
4.8 การเซฟข้อมูลรายละเอียดของสินค้า.....	77
4.9 เซิร์ฟเวอร์ เมื่อทำการรับข้อมูลจาก จีพีอาร์เอส โมดูล.....	78
4.10 ข้อมูลผ่านทาง เว็บเบราว์เซอร์.....	78
4.11 ข้อมูลวางแผนก่อนทำการเติมสินค้าจริง.....	79
4.12 ข้อมูลวางแผนก่อนทำการเติมจริงผ่านทาง เว็บเบราว์เซอร์.....	79
4.13 ยอดขายตามเวลา.....	80
4.14 ยอดขายตามสินค้า.....	80
4.15 ยอดขายตามสาขา.....	81
4.16 ยอดขายย้อนหลัง.....	81
ค.1 วงจร เครื่องหยุดเหรียญอัตโนมัติ.....	125
ค.2 วงจรสื่อสาร จีพีอาร์เอส โมดูล ด้วยไมโครคอนโทรลเลอร์.....	126

บทที่ 1

บทนำ

1.1 ความเป็นมาของปัญหา

เนื่องจากการให้บริการผู้หยุดเหรียญในปัจจุบันนั้นในการตรวจสอบจำนวนของสินค้าในตู้จะต้องเดินทางไปยังสถานที่ตั้งของตู้ซึ่งทำให้เสียค่าใช้จ่ายในการเดินทางและเสียเวลา ดังนั้นจึงได้จัดทำระบบนี้ขึ้นมาเพื่อความสะดวกในการตรวจสอบจำนวนของสินค้า

เนื่องจากการให้บริการผู้ขายสินค้าหยุดเหรียญอัตโนมัติในปัจจุบันนี้ ในการตรวจสอบจำนวนสินค้าและการเติมสินค้าของผู้ให้บริการนั้น ผู้ให้บริการจะต้องทำการเดินทางไปยังสถานที่ตั้งของเครื่องขายสินค้าโดยตรง ซึ่งลักษณะการดำเนินธุรกิจแบบนี้อาจเกิดข้อเสีย เช่น การตรวจสอบจำนวนสินค้านั้นไม่สามารถทำได้บ่อยมากนักเนื่องจากปัญหาด้านค่าใช้จ่ายในการเดินทาง เพราะผู้ให้บริการจะทราบว่าเครื่องขายสินค้าใดไปจำนวนเท่าไร ต้องรอจนถึงวันที่มีการเติมสินค้า และยังมีปัญหาในการจัดสินค้าเพื่อนำไปเติมให้กับตู้ขายสินค้าที่ให้บริการได้อย่างเพียงพอซึ่งผู้ให้บริการไม่สามารถทราบได้เลยว่าจะต้องจัดสินค้าไปจำนวนเท่าไร

1.2 วัตถุประสงค์ของโครงการ

- เพื่อศึกษาการทำงานระบบฝังตัว (Embedded System)
- เพื่อศึกษาการทำงานระบบ จีพีอาร์เอส
- เพื่อศึกษาวิธีการส่งข้อมูลแบบ ไร้สาย
- เพื่อศึกษาการทำงานระบบฐานข้อมูล
- สร้างระบบที่สามารถตรวจสอบจำนวนสินค้าภายในเครื่องหยุดเหรียญ
- สร้างระบบติดต่อระหว่างเครื่องหยุดเหรียญกับ เซิร์ฟเวอร์
- สร้าง แอปพลิเคชัน เพื่อนำข้อมูลมาใช้งาน

1.3 ประโยชน์ที่คาดว่าจะได้รับ

- ผู้ผลิตหรือผู้จัดส่งสินค้าสามารถรู้จำนวนสินค้าภายในเครื่องหยุดเหรียญโดยไม่ต้องเดินทางไปตู้ขายสินค้า
- ผู้ผลิตสามารถนำข้อมูลมาใช้ในการวางแผนการตลาด หรือวางแผนในการจัดส่งสินค้า
- สามารถนำไปประยุกต์ใช้กับเครื่องหยุดเหรียญอัตโนมัติที่มีอยู่ในปัจจุบัน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1.4 ขอบเขตของโครงการ

- สร้างเครื่องขายสินค้าหยอดเหรียญจำลอง
- สร้างระบบเก็บจำนวนสินค้าที่ขายได้ ภายในเครื่องหยอดเหรียญ
- สร้างระบบส่งข้อมูลจากเครื่องหยอดเหรียญไปยัง เซิร์ฟเวอร์
- สร้าง แอปพลิเคชัน นำข้อมูลไปใช้งาน

1.5 ขั้นตอนการทำงาน

- เสนอแนวทางและเลือกเทคโนโลยีที่เหมาะสมกับโครงการ
- ศึกษาเทคโนโลยีที่เกี่ยวข้อง
- ศึกษาและออกแบบระบบโดยรวม
- สร้างระบบย่อย ได้แก่
 - แบบจำลองเครื่องขายสินค้าหยอดเหรียญ
 - ระบบติดต่อระหว่างเครื่องหยอดเหรียญและศูนย์กลาง
 - ระบบส่วนกลาง (Server)
 - ระบบฐานข้อมูล (Database)
 - สร้าง โปรแกรมสำเร็จรูปเพื่อนำข้อมูลมาใช้งาน แอปพลิเคชัน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 2

ทฤษฎีที่เกี่ยวข้อง

2.1 ไมโครคอนโทรลเลอร์ PIC16F877

2.1.1 โครงสร้างของไมโครคอนโทรลเลอร์ PIC16F877

แสดงดังในรูปที่ 2.1 ส่วนประกอบหลักคือ ซีพียู, หน่วยความจำโปรแกรม, หน่วยความจำข้อมูลส่วนติดต่oport, ส่วนจัดการสัญญาณนาฬิกาของระบบ, วงจรรีเซ็ต (reset)หลัก, ส่วนควบคุมการตอบสนองอินเทอร์รัปต์ (interrupt) และส่วนจัดการด้านไฟเลี้ยง แต่จะมีโมดูลพิเศษเพิ่มเติมเข้ามาหากพอควรเพื่อช่วยให้ PIC 16F877 มีความสามารถสูงมากขึ้น ได้แก่ วงจรบราวเอาต์รีเซต, ส่วนแก้ไขข้อมูลในวงจรหรือ ดีบั๊กเกอร์ (Debugger), วงจรโปรแกรมข้อมูลด้วยแรงดันต่ำ, ไทเมอร์ที่มีมากถึง 3 ตัว, วงจรแปลงสัญญาณอนาล็อกเป็นดิจิทัลขนาด 10 บิต, โมดูลเชื่อมต่ออุปกรณ์อนุกรม, โมดูลเชื่อมต่ออุปกรณ์ระบบบัส I2C, โมดูลสื่อสารอนุกรมและ โมดูลเปรียบเทียบสัญญาณ-ตรวจจับสัญญาณ วงจรมอดูเลชันทางความกว้างของพัลส์หรือ พีดับเบิลยูเอ็ม (PWM) นอกจากนี้ในอนุกรม PIC 16F87xA จะมีวงจรเปรียบเทียบแรงดันอนาล็อกและ โมดูลสร้างแรงดันอ้างอิงเพิ่มเติมเข้ามาอีกด้วย

2.1.2 คุณสมบัติทางเทคนิคของ Pic16F87x/87xA

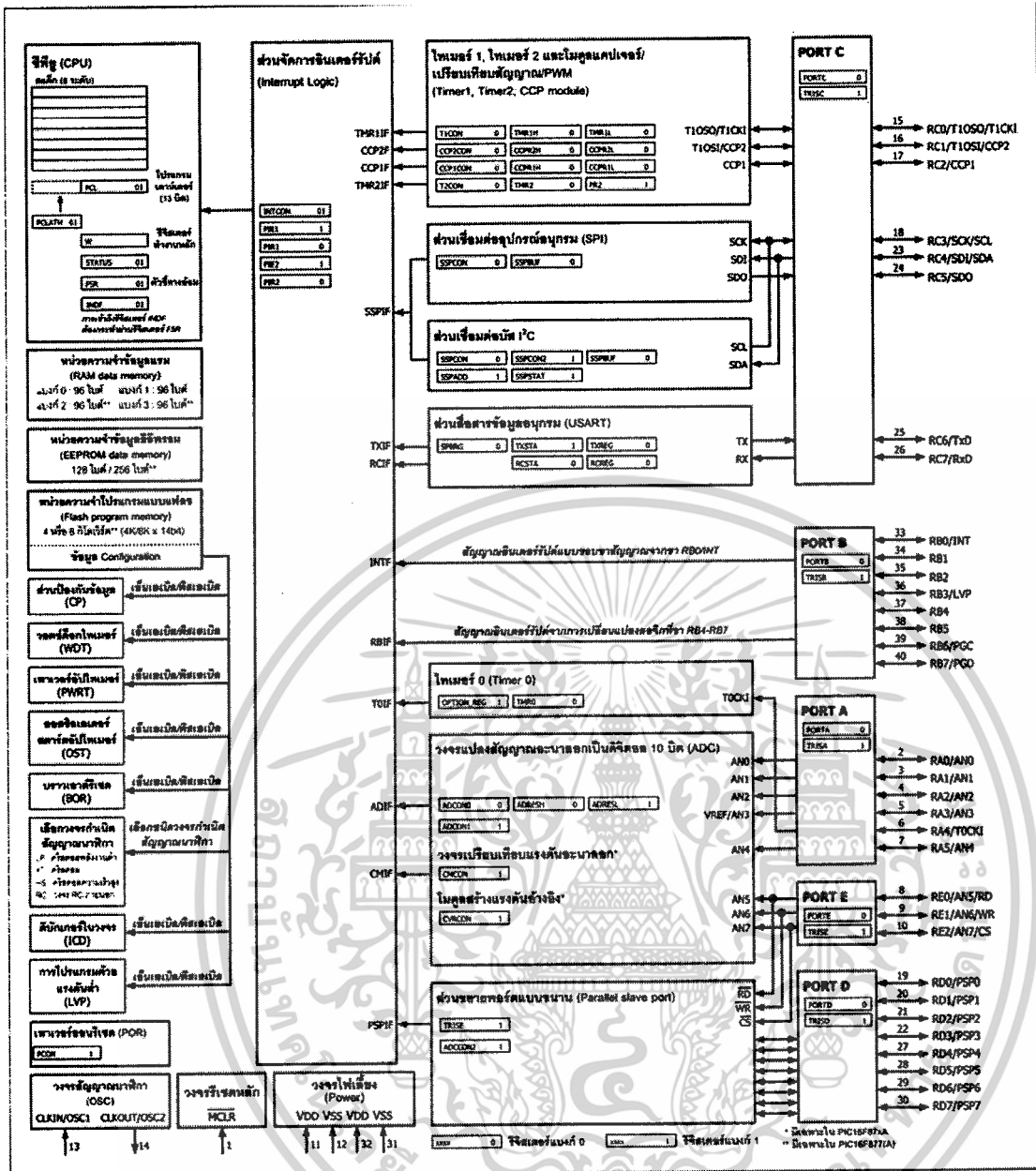
2.1.2.1 คุณสมบัติหลัก

- ซีพียูแบบ อาร์ไอเอสซี (RISC) มีคำสั่งใช้งานเพียง 35 คำสั่ง
- สามารถกระทำคำสั่งโดยใช้สัญญาณเพียงหนึ่งลูก ยกเว้นคำสั่งการกระโดด
- ความถี่สัญญาณนาฬิกา ตั้งแต่ไฟตรงถึง 20 MHz
- หน่วยความจำโปรแกรม
 - 2 กิโลเวิร์ด สำหรับ Pic16F870/871/872
 - 4 กิโลเวิร์ด สำหรับ Pic16F873 (A)/874(A)
 - 8 กิโลเวิร์ด สำหรับ Pic16F876/876A/877/877A
- หน่วยความจำข้อมูลแรมหรือรีจิสเตอร์
 - 128 ไบต์ สำหรับ pic16F870/871/872
 - 192 ไบต์ สำหรับ pic16F873 (A)/874(A)
 - 368 ไบต์ สำหรับ Pic16F876/876A/877/877A
- ขนาดหน่วยความจำอีอีพรอม
 - 64 ไบต์ สำหรับ Pic16F870/871/872

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- 128 ไบต์ สำหรับ pic16F873 (A)/874(A)
- 256 ไบต์ สำหรับ Pic16F876/876A/877/877A
- ตอบสนองแหล่งกำเนิดอินเทอร์รัปต์สูงสุดถึง 15 แหล่งขึ้นกับเบอร์ของไมโครคอนโทรลเลอร์
- มีสแต็ก 4 ระดับ
- มีวงจรเพาเวอร์อนรีเซต (POR)
- มีเพาเวอร์อัปไทมเมอร์ (PWRT) และออสซิลเลเตอร์สตาร์ทอัปไทมเมอร์ (OST)
- มีวงจรวอตช์ด็อกไทมเมอร์ (WDT) ที่มีวงจรออสซิลเลเตอร์ในตัว ทำให้มีความเชื่อถือในการทำงานสูง
- เลือกรหัสคีย์ข้อมูลทั้งในหน่วยความจำโปรแกรมและหน่วยความจำข้อมูลสามารถเลือกระดับการป้องกันได้
- มีโหมดประหยัดพลังงาน
- สามารถโปรแกรมโดยใช้แรงดัน +5v ได้
- แก้ไขข้อมูลในหน่วยความจำโปรแกรมด้วยกระบวนการ ไอซีดี (ICD) ผ่านพอร์ตเพียง 2 ขา
- ซีพียูสามารถอ่านและเขียนหน่วยความจำโปรแกรมได้
- ไฟเลี้ยง +2 ถึง +5.5v
- กระแสซิงก์และซอร์สของพอร์ต 25 mA
- การใช้พลังงานไฟฟ้าในกรณีไม่ขับโหลด
 - น้อยกว่า 2 mA ที่ไฟเลี้ยง +5v และสัญญาณนาฬิกา 4MHz
 - 20 uA ที่ไฟเลี้ยง + 3v และสัญญาณนาฬิกา 32 kHz
 - น้อยกว่า 1 uA ในโหมดประหยัดพลังงานทรอสแตนด์บาย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.1 โครงสร้างของไมโครคอนโทรลเลอร์ PIC16F877

2.1.2.2 คุณสมบัติพิเศษเพิ่มเติม

- ไทเมอร์ 3 ตัว คือ ไทเมอร์ 0 ขนาด 8 บิต มีปริสเกลเลอร์ขนาด 8 บิตในตัว, ไทเมอร์ 1 ขนาด 16 บิต พร้อมปรสเกลเลอร์ และไทเมอร์ 2 ขนาด 8 บิต มีปริสเกลเลอร์, โพสต์สเกลเลอร์ และรีจิสเตอร์คาบเวลา (period register) ขนาด 8 บิตในตัว
- มีโมดูล ซีซีพี (CCP) 2 ชุดโดย
- ส่วนตรวจจับสัญญาณหรือแคปเจอร์ (Capture) มีขนาด 16 บิตความละเอียดสูงสุด 12.5 นาโนวินาที

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่นอนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- ส่วนเปรียบเทียบสัญญาณ (Compare) มีขนาด 16 บิต ความละเอียดสูงสุด 200 นาโนวินาที
- วงจร พีคดับเบิลยูเอ็ม มีความละเอียดสูงสุด 10 บิต
- มีวงจรแปลงสัญญาณอนาลอกเป็นดิจิตอล 10 บิต (5 ช่องสำหรับรุ่น 28 ขาและ 8 ช่องสำหรับรุ่น 40 ขา)
- วงจรเชื่อมต่ออุปกรณ์อนุกรมทั้ง เอสพีไอ(SPI) และ บัส 2 ซี (I2C)
- วงจรสื่อสารข้อมูลอนุกรม (USART) พร้อมการตรวจจับแอดเดรส 9 บิต
- มีวงจรตรวจจับระดับแรงดันไฟเลี้ยง (บราวเอาต์ดีเทกชัน : Brown-out detection) เพื่อการรีเซ็ตชิพหรือ เรียกว่า บราวเอาต์รีเซต (Brown-out reset: BOR)

2.1.3 ซีซีเอส ซี คอมไพเลอร์ (CCS C Compiler)

เป็นซอฟต์แวร์สำหรับการแปลโปรแกรมภาษา C ของไมโครคอนโทรลเลอร์ PIC เป็นรหัสเครื่องผลิตโดย คัสทอม คอมพิวเตอร์ เซอร์วิส (Custom Computer Service) สหรัฐอเมริกา สามารถดูรายละเอียดผ่านทางเว็บไซต์ที่ www.ccsinfo.com สำหรับในประเทศไทยมีตัวแทนจำหน่ายอย่างเป็นทางการคือ บริษัท อิน โนเวตีฟ เอ็กเพอริเมนต์ จำกัด www.inex.co.th

ซีซีเอส ซี (CCS C) คอมไพเลอร์มีหลายรุ่น เพื่อรองรับความต้องการของผู้ใช้งาน ซีซีเอส ซี คอมไพเลอร์ในรุ่นต่างๆและในหนังสือเล่มนี้จะเน้นไปที่รุ่น พีซีเอ็ม (PCM) ซึ่งเป็น คอมไพเลอร์แบบบรรทัดคำสั่งหรือ คอมมานด์ไลน์ รองรับไมโครคอนโทรลเลอร์ PIC รุ่น 14 บิตทั้งหมดและ พีซีดับเบิลยู (PCWH) เป็น C คอมไพเลอร์ที่ผนวกเท็กซ์เอดิเตอร์ คอมไพเลอร์ แอสเซมเบลอร์และดีบักเกอร์ทำงานผ่านส่วนพัฒนาพร้อม (IDE : Integrated Development Environment) ระบบปฏิบัติการวินโดวส์ สามารถรองรับไมโครคอนโทรลเลอร์ PIC ครบทุกรุ่นทั้ง 12, 14 และ 16 บิต

2.1.3.1 การใช้งานพอร์ตอินพุตเอาต์พุตดิจิตอลของไมโครคอนโทรลเลอร์ PIC ด้วยโปรแกรมภาษา C

พอร์ตของไมโครคอนโทรลเลอร์ PIC สามารถทำงานเป็นได้ทั้งพอร์ตอินพุตและพอร์ตเอาต์พุตดิจิตอลและเป็นพอร์ตอินพุตวงจรแปลงสัญญาณอนาลอกเป็นดิจิตอล โดยที่การแปลงสัญญาณอนาลอกนี้จะขึ้นอยู่กับคุณสมบัติของขาพอร์ตนั้น ซึ่งกำหนดไว้แล้วในแต่ละเบอร์ของไมโครคอนโทรลเลอร์ PIC

ไมโครคอนโทรลเลอร์ PIC 16F877 มีขาพอร์ตให้ใช้งานได้มากถึง 34 บิต โดยแบ่งเป็นพอร์ต A จำนวน 6 ช่อง พอร์ต B, C, D อย่างละ 8 ช่อง และพอร์ต E จำนวน 3

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.1.3.1.1 คำสั่งควบคุมพอร์ตไมโครคอนโทรลเลอร์ PIC

ซีซีเอส ซี คอมไพเลอร์ได้เตรียมชุดคำสั่งที่ใช้ในการควบคุมการทำงานของขาพอร์ตไม่ว่าจะเป็นอินพุตหรือเอาต์พุตด้วยฟังก์ชันดังต่อไปนี้ คือ

1. คำสั่งไคเร็กทีฟกำหนดคุณสมบัติของพอร์ต ได้แก่

```
#use fast_io
```

```
#use fixed_io
```

```
#use standard_io
```

2. ฟังก์ชันกำหนดพอร์ตอินพุตเอาต์พุต ได้แก่

```
set_tris_x()
```

3. ฟังก์ชันควบคุมพอร์ตเอาต์พุต ได้แก่

```
output_x()
```

```
output_low()
```

```
output_high()
```

```
output_float()
```

```
output_bit()
```

4. ฟังก์ชันควบคุมพอร์ตอินพุต ได้แก่

```
input_a()
```

```
input_b()
```

```
input_c()
```

```
input_d()
```

```
input_e()
```

```
input()
```

ค่า x คือพอร์ตที่เราจะนำไปใส่ เช่นจะ เซทพอร์ต A ก็ใช้ฟังก์ชัน set_tris_a()

2.1.3.1.2 คำสั่งไคเร็กทีฟ

การควบคุมขาพอร์ตสามารถกำหนดคุณสมบัติในการใช้งานได้ด้วยไคเร็กทีฟต่อไปนี้

1. #use fast_io (port)

การใช้งานไคเร็กทีฟนี้จะมีผลโดยตรงกับรีจิสเตอร์จึงต้องกำหนดการใช้งานพอร์ตด้วยฟังก์ชัน set_tris_x () ก่อนที่จะเรียกใช้งานคำสั่งควบคุมการทำงานของพอร์ตว่าจะทำหน้าที่เป็นพอร์ตอินพุตหรือเอาต์พุตในระหว่างที่ใช้งานพอร์ต

2. #use fixed_io (port_output=pin, pin?)

การใช้งานไคร์เร็กตีฟนี้ เป็นการบอกให้คอมไพเลอร์รับรู้ว่าจะใช้พอร์ตดังกล่าว เป็นพอร์ตอินพุต หรือ พอร์ตเอาพุตตลอดการใช้งานจนกว่าจะมีการเปลี่ยนแปลงเป็นอย่างอื่น

3. #use standard_io (port)

การใช้งานไคร์เร็กตีฟนี้ เป็นการกำหนดให้พอร์ตเป็นอินพุตหรือเอาพุตตลอดการใช้งาน

ขาพอร์ต ซึ่งเป็นค่าปกติ (default I/O) ของการใช้งาน

2.1.3.1.3 ฟังก์ชันกำหนดพอร์ตอินพุตเอาพุต

ก่อนการใช้งานขาพอร์ตของไมโครคอนโทรลเลอร์ PIC จะต้องกำหนดสถานะของขาพอร์ตว่า ต้องการให้ขาใดเป็นอินพุตหรือเอาพุตด้วยฟังก์ชัน `set_tris_x ()` เมื่อมีการใช้งานไคร์เร็กตีฟ `# use fast_io`

2.1.3.1.4 ฟังก์ชันควบคุมพอร์ตเอาต์พุต

1. การใช้งาน `output_x ()`

การใช้งาน ค่า x คือพอร์ตที่จะ output ออกไปส่วนค่าในวงเล็บมีขนาด 8 บิต

2. การใช้งาน `output_low ()` และ `output_high ()`

เป็นฟังก์ชันที่ใช้ในการควบคุมเอาพุตในระดับลอจิกสูง (High) หรือระดับต่ำ (Low) ของขาพอร์ตที่ต้องการใช้งาน

โดยที่ ขา (Pin) คือค่าที่ถูกกำหนดไว้แล้วในเฮดเดอร์ไฟล์ของไมโครคอนโทรลเลอร์ PIC ที่ใช้งาน เช่น PIC 16F877 ใช้ไฟล์ 16f877.h ซึ่งกำหนดให้แต่ละพอร์ตของ A เป็นดังนี้

```
# define PIN_A0 40
```

```
# define PIN_A1 41
```

```
# define PIN_A2 42
```

```
# define PIN_A3 43
```

```
# define PIN_A4 44
```

```
# define PIN_A5 45
```

3. การใช้งาน `output_float ()`

ฟังก์ชัน `output_float ()` จะทำให้ขาพอร์ตเป็นแบบคอลเล็กเตอร์เปิด หรือแบบเดรน นั่นคือ เมื่อมีการใช้งานขาพอร์ตที่เป็นคอลเล็กเตอร์เปิด จึงต้องต่อตัวต้านทานพูลอัพไว้ ดังนั้นในการใช้งานในลักษณะนี้ต้องให้ความระวังเป็นพิเศษ

4. การใช้งาน `output_bit ()`

เป็นฟังก์ชันที่ใช้ในการควบคุมเอาพุตในระดับบิตของขาพอร์ต

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.1.3.1.5 ฟังก์ชันควบคุมพอร์ตอินพุต

- การใช้งาน `input_x ()`

เป็นกลุ่มฟังก์ชันที่เกี่ยวข้องกับการอ่านค่าของขาพอร์ตหรืออ่านค่าอินพุตของขาพอร์ต ได้แก่ `input_a ()`, `input_b ()`, `input_c ()`

2.1.3.2 การใช้งานไทมเมอร์/เคาเตอร์ ภายในไมโครคอนโทรลเลอร์ PIC

ไทมเมอร์/เคาเตอร์ เป็นส่วนสำคัญสำหรับไมโครคอนโทรลเลอร์ PIC เพื่อนำมาสร้างฐานเวลา โดยไทมเมอร์หรือตัวตั้งเวลาและเคาเตอร์หรือตัวนับ มีทั้งหมด 8 และ 16 บิตสามารถใช้กำเนิดสัญญาณอินเตอร์รัปต์เมื่อเกิดการนับค่าเกินหรือโอเวอร์โฟล นอกจากนี้ยังสามารถนำมาใช้งานเป็นตัวสร้างฐานเวลาให้กับโมดูลตรวจจับ/เปรียบเทียบ/สร้างสัญญาณ พัลส์เบรียเอม หรือ โมดูล ซีซีพี เมื่อกำหนดให้ทำงานในโหมดกำเนิดสัญญาณ พัลส์เบรียเอม

ในไมโครคอนโทรลเลอร์ PIC ทั้งเบอร์ PIC 16F877 และ PIC 16F628 จะมีไทมเมอร์ให้นำมาใช้งานได้มากถึง 3 ตัวคือ ไทมเมอร์ 0, ไทมเมอร์ 1, ไทมเมอร์ 2 รายละเอียดการทำงานของไทมเมอร์ทั้งสามตัวมีดังนี้

1. ไทมเมอร์ 0 เป็นตัวตั้งเวลาและตัวนับ ขนาด 8 บิต และมีปริสเกลขนาด 8 บิต
2. ไทมเมอร์ 1 เป็นตัวตั้งเวลาและตัวนับขนาด 16 บิตและมีปริสเกลขนาด 16 บิตที่พิเศษคือเมื่อทำงานอยู่ในโหมดสลีป ไทมเมอร์ 1 ก็ยังคงทำงานอยู่ กำหนดให้ทำงานแบบซิงโครนัสและอะซิงโครนัส และกำเนิดสัญญาณอินเตอร์รัปต์เนื่องจากโอเวอร์โฟลได้
3. ไทมเมอร์ 2 มีขนาด 8 บิต มีทั้งไทมเมอร์ , ปริสเกลเลอร์ , โปสต์สเกลเลอร์ และรีจิสเตอร์คาบเวลา นอกจากนี้ยังเป็นตัวกำหนดมาตรฐานเวลาของโมดูล ซีซีพี เมื่อทำงานในโหมด พัลส์เบรียเอม(PWM)

การกำหนดให้ไทมเมอร์ทำงานในโหมดไทมเมอร์หรือเคาเตอร์จะขึ้นอยู่กับการทำงานนับสัญญาณ ถ้าเป็นการนับสัญญาณนาฬิกาจากภายในก็จะทำงานเป็นตัวตั้งเวลาหรือไทมเมอร์ แต่ถ้าเป็นการนับสัญญาณจากภายนอกก็จะเป็นเคาเตอร์

2.1.3.2.1 การใช้งานไทมเมอร์เคาเตอร์

ซีซีเอส ซี ได้เตรียม บิวอินฟังก์ชัน (Built-in functions) เพื่อใช้งานกับไทมเมอร์/เคาเตอร์และวอตช์ด็อกไทมเมอร์ ในการใช้งานไทมเมอร์ทั้งสามตัวนั้นจะต้องมีการจัดเตรียมไทมเมอร์ก่อนที่จะใช้งาน โดยอาศัยฟังก์ชัน `setup_timer_x()` (x คือค่าแสดงถึงตัวเลขของไทมเมอร์) ซึ่งมีรายละเอียดดังนี้

2.1.3.2.1.1 ฟังก์ชัน `setup_timer_0()` กับการใช้งานในโหมดไทมเมอร์ 0

รูปแบบการใช้งานโดยที่

`Setup_timer_0(mode)`

Mode คือค่าคงที่ที่กำหนดในไฟล์เฮดเดอร์ มีรายละเอียดดังตารางที่ 2.1

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่นิยมนำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 2.1 ตัวแปรการทำงานร่วมกับไทมเมอร์ 0

Mode	ความหมาย
RTCC_INTERNAL	รับสัญญาณพิกจากภายในไมโครคอนโทรลเลอร์
RTCC_EXT_L_TO_H	รับสัญญาณพิกที่เปลี่ยนจากลอจิกต่ำไปสูงจากภายนอก
RTCC_EXT_H_TO_L	รับสัญญาณพิกที่เปลี่ยนจากลอจิกสูงไปต่ำจากภายนอก
RTCC_DIV_2	อัตราส่วนการทำงานของปริสเกลเลอร์ 1: 2
RTCC_DIV_4	อัตราส่วนการทำงานของปริสเกลเลอร์ 1: 4
RTCC_DIV_16	อัตราส่วนการทำงานของปริสเกลเลอร์ 1: 16
RTCC_DIV_32	อัตราส่วนการทำงานของปริสเกลเลอร์ 1: 32
RTCC_DIV_64	อัตราส่วนการทำงานของปริสเกลเลอร์ 1: 64
RTCC_DIV_128	อัตราส่วนการทำงานของปริสเกลเลอร์ 1: 128
RTCC_DIV_256	อัตราส่วนการทำงานของปริสเกลเลอร์ 1: 256

หมายเหตุ ปริสเกลเลอร์ คือตัวหาร-ตัวลดทอนของโมดูลไทมเมอร์ 0 สามารถกำหนดค่าคงที่ได้ 1 หรือ 2 ค่าโดยใช้ตัวปฏิบัติการ OR “|” ในการเชื่อมค่าคงที่

2.1.3.2.1.2 ฟังก์ชัน set_timer_1() กับการใช้งานในโหมดไทมเมอร์ 1

รูปแบบการใช้งาน โดยมี

Setup_timer_1()

Mode คือค่าคงที่ที่กำหนดในไฟล์เฮดเดอร์ มีรายละเอียดดังตารางที่ 2.2

ตารางที่ 2.2 ตัวแปรการทำงานร่วมกับไทมเมอร์ 1

Mode	ความหมาย
T1_DISABLED	หยุดการทำงานของไทมเมอร์1
T1_INTERNAL	ไทมเมอร์1 ทำงานใช้สัญญาณนาฬิกาภายในความถี่ ¼ เท่าของความถี่หลัก
T1_EXTERNAL	ไทมเมอร์1 ทำงานใช้สัญญาณนาฬิกาภายนอก, ไม่ต้องการให้สัญญาณกระตุ้นจากภายนอกสัมพันธ์กับสัญญาณนาฬิกาภายในของไมโครคอนโทรลเลอร์หรือไม่เกิดการซิงโครไนซ์
T1_EXTERNAL_SYNC	ไทมเมอร์1 ทำงาน, ใช้สัญญาณนาฬิกาภายนอก, กำหนดให้สัญญาณกระตุ้นจากภายนอกสัมพันธ์กับ นาฬิกาภายในหรือการซิงโครไนซ์
T1_CLK_OUT	เอ็นเอเบิลการส่งสัญญาณออกไปทางเอาพุตของวงจรถ้าเน็ดสัญญาณนาฬิกา
T1_DIV_BY_1	อัตราส่วนการทำงานของปริสเกลเลอร์ 1:1
T1_DIV_BY_2	อัตราส่วนการทำงานของปริสเกลเลอร์ 1:2
T1_DIV_BY_4	อัตราส่วนการทำงานของปริสเกลเลอร์ 1:4
T1_DIV_BY_8	อัตราส่วนการทำงานของปริสเกลเลอร์ 1:8สามารถกำหนดค่าคงที่เป็นกลุ่มได้โดยใช้ตัวปฏิบัติการOR “ ”

2.1.3.2.1.3 ฟังก์ชัน setup_timer_20 กับการใช้งานในโหมดไทมเมอร์ 2 รูปแบบการใช้งาน

Setup_timer_2 (mode, period, postscale)

โดยที่

Mode คือค่าคงที่กำหนดใน ไฟล์เฮดเดอร์มีรายละเอียดดังตารางที่ 2.3

ตารางที่ 2.3 ตัวแปรการทำงานร่วมกับไทมเมอร์ 2

Mode	ความหมาย
T2_DISABLED	หยุดการทำงานของไทมเมอร์
T2_DIV_BY_1	อัตราส่วนการทำงานของปริสเกลเลอร์ 1:1
T2_DIV_BY_4	อัตราส่วนการทำงานของปริสเกลเลอร์ 1:4
T2_DIV_BY_16	อัตราส่วนการทำงานของปริสเกลเลอร์ 1:16

Period คือ คาบเวลาช่วง 0-255 เพื่อกำหนดหารรีเซตก่อนที่จะเกิดอินเตอร์รัปต์เพื่อกระตุ้นการเกิดอินเตอร์รัปต์เมื่อมีการเอ็นเอเบิลไว้

Postscale ตัวลดทอนการส่งออก มีค่า 1-16, ตัวหารความถี่ออสซิลเลเตอร์ (prescaler) สามารถเลือกอัตราส่วนเป็น 1,4,16 และ รีจิสเตอร์คาบเวลา (period register)

2.1.3.2.1.4 เซตไทมเมอร์ด้วยฟังก์ชัน set_timer0(), set_timer1(), set_timer2()

รูปแบบการใช้งาน

Set_timer0(value)

Set_timer1(value)

Set_timer2(value)

โดยบิตที่ Value คือการกำหนดค่าเริ่มต้นในการนับให้กับไทมเมอร์สำหรับ สำหรับ timer0 และ timer2 เป็นแบบ int 8 บิตและ 16 บิต สำหรับไทมเมอร์ 1

$$\text{สูตรในการคำนวณ} \quad TMR0 = 256 - \left(\frac{T_{delay}}{PR} \right) \times \left(\frac{f_{osc}}{4} \right) \quad (2.1)$$

เมื่อ T_{delay} คือ ค่าหน่วยเวลาในหน่วยวินาที

PR คือ ค่าของปริสเกล

f_{osc} คือ ความถี่ของสัญญาณนาฬิกาหลักในหน่วยเฮิร์ตซ์

2.1.3.2.2 ใช้งานโมดูลแปลงสัญญาณอะนาลอกเป็นดิจิตอลในไมโครคอนโทรลเลอร์

ไมโครคอนโทรลเลอร์ PIC ในปัจจุบันมักจะผนวกโมดูลแปลงสัญญาณอะนาลอกเป็นดิจิตอลไว้ด้วยภายใน ส่วนจำนวนช่องและความละเอียดของการแปลงนั้นจะขึ้นอยู่กับเบอร์ของ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ไมโครคอนโทรลเลอร์เช่น ไมโครคอนโทรลเลอร์ PIC 16F877 จะมีความละเอียด 10 บิต 5 ช่อง สำหรับรุ่น 28 ขาและ 8 ช่องในรุ่น 40 ขา โดยขาพอร์ตที่ใช้งานร่วมด้วยนั้นประกอบด้วย RA0-RA3 และ RA5 ในรุ่น 40 ขา จะมี RE0-RE2 รวมอยู่ด้วย

การใช้งานโมดูลแปลงสัญญาณอะนาลอกเป็นดิจิทัลของไมโครคอนโทรลเลอร์ PIC ด้วยโปรแกรมภาษา C นั้น ซีซีเอส ซี คอมไพเลอร์ ได้เตรียมฟังก์ชันไว้ให้พร้อมแล้วไม่ว่าจะเป็น ส่วนเตรียมความพร้อมและการเรียกใช้งานโมดูลแปลงสัญญาณอะนาลอกเป็นดิจิทัล รวมถึง ฟังก์ชันในการอ่านค่าสัญญาณอะนาลอกเป็นดิจิทัลดังนี้

1. ฟังก์ชันเกี่ยวกับการกำหนดขาพอร์ตอะนาลอก ได้แก่ `setup_adc()`, `setup_adc_ports()` และ `set_adc_channel()`
2. ฟังก์ชันในการอ่านค่าสัญญาณอะนาลอก ได้แก่ `read_adc()`

ค่าคงที่ที่เกี่ยวข้องกับฟังก์ชันของโมดูลแปลงสัญญาณอะนาลอกเป็นดิจิทัลใน ไมโครคอนโทรลเลอร์ PIC 16F877 ที่ได้กำหนดไว้แล้วในไฟล์เฮดเดอร์ของไมโครคอนโทรลเลอร์ เบอร์นั้นๆ

2.1.3.2.2.1 ฟังก์ชันเกี่ยวกับการกำหนดขาพอร์ตอะนาลอก

2.1.3.2.2.1.1 `setup_adc()`

ใช้กำหนดสัญญาณนาฬิกาสำหรับ โมดูลแปลงสัญญาณอะนาลอกเป็นดิจิทัล ในตารางที่ 2.4 แสดงการกำหนดสัญญาณนาฬิกาของ โมดูลแปลงสัญญาณอะนาลอกเป็นดิจิทัล

ตารางที่ 2.4 การกำหนดสัญญาณนาฬิกาของโมดูลแปลงสัญญาณอะนาลอกเป็นดิจิทัล

ความถี่สัญญาณนาฬิกา	โหมดการทำงานของ โมดูลแปลงสัญญาณอะนาลอกเป็นดิจิทัล			
	ADC_CLO	ADC_CLOCK_	ADC_CLOC	ADC_CLOCK_
ไมโครคอนโทรลเลอร์ PIC 16F877	CK_DIV_2	DIV_8	K_DIV_32	INTERNAL
20 MHz	10 MHz	2.5 MHz	625kHz	166.67-500 kHz
5 MHz	2.5 MHz	625 kHz	156.25 kHz	166.67-500 kHz
1 MHz	500 kHz	125 kHz	31.25 kHz	166.67-500 kHz
333.33 kHz	166.67 kHz	41.67 kHz	10.42 kHz	166.67-500 kHz

รูปแบบการใช้งาน

`Setup_adc(mode);`

โดยที่ โหมด เป็นค่าที่ใช้ในการกำหนดโหมดทำงานของ โมดูลแปลง

สัญญาณอะนาลอกเป็นดิจิทัล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หมายเหตุ โหมด (Mode) เป็นค่าคงที่ ที่ได้กำหนดไว้แล้วในไฟล์เฮดเดอร์ของไมโครคอนโทรลเลอร์เบอร์นั้น

`setup_adc_port()`

เป็นการกำหนดคุณสมบัติของขาพอร์ตที่ใช้งาน ว่าต้องการให้ขาพอร์ตใดบ้างเป็นอินพุตอนาลอกและขาพอร์ตใดใช้งานในการอ้างอิงหรือกำหนดให้ทำงานกับแรงดันอ้างอิง

รูปแบบการใช้งาน

`Setup_adc_ports(value)`

โดยที่ Value เป็นค่าคงที่ที่ใช้การกำหนดคุณสมบัติของขาพอร์ตที่ใช้งาน เช่น

- ALL_ANALOG
- NO_ANALOG
- ANALOG_RA3_REF
- RA0_RA1_RA3_ANALOG

เป็นต้น

2.1.3.2.2.1.2 `setup_adc_channel()`

ใช้กำหนดขาพอร์ตอะนาลอกที่ต้องการใช้งานและอ่านค่ากลับมาด้วยฟังก์ชัน

`read_adc()`

รูปแบบการใช้งาน

`Set_adc_channel(chan)`

โดยที่ chan คือ ขาพอร์ตแปลงสัญญาณอะนาลอกเป็นดิจิตอลแต่ละช่อง มีค่า

เริ่มต้นที่ 0 เท่ากับ AN0

2.1.3.2.2.2 ฟังก์ชันการอ่านค่าอนาลอก

มี 1 ฟังก์ชันคือ `read_adc()` ใช้อ่านค่าจากขาพอร์ตอนาลอก

รูปแบบการใช้งาน

`Value = read_adc()`

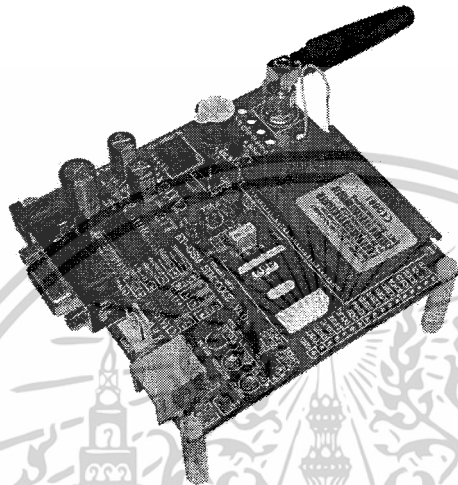
ค่าส่งกลับ Value เป็นข้อมูลดิจิตอลที่ได้จากการแปลงสัญญาณแล้ว จะมีขนาด 8 หรือ 16 บิตขึ้นอยู่กับเบอร์ของไมโครคอนโทรลเลอร์ที่ใช้และการกำหนดไคเร็กทีฟ #device

`ADC=nm`

2.2 จีพีอาร์เอส โมดูล

การออกแบบ จีพีอาร์เอส โมดูล ในส่วนนี้สามารถอธิบายได้ ส่วนใหญ่ๆสองส่วน ด้วยกันคือ ส่วนที่เป็น จีพีอาร์เอส โมดูล และส่วนที่เป็น ไมโครคอนโทรลเลอร์ซึ่งจะอธิบายรายละเอียดของตัว จีพีอาร์เอส โมดูล ในการทดลองนี้ได้ใช้ อีที-จีเอสเอ็ม ซิม300ซีแซต

2.2.1 อีที-จีเอสเอ็ม ซิม300ซีแซต



รูปที่ 2.2 จีพีอาร์เอส โมดูล รุ่น อีที-จีเอสเอ็ม ซิม300ซีแซต

รุ่น ซิม300ซีแซต ของ “ซิมคอม-แอลทีดี (SIMCOM Ltd.)” เป็นอุปกรณ์หลัก ซึ่ง ซิม300ซีแซต เป็นโมดูลสื่อสารระบบ จีเอสเอ็ม/จีพีอาร์เอสขนาดเล็ก รองรับระบบสื่อสาร จีเอสเอ็ม ความถี่ 900/1800/1900MHZ โดยส่งงานผ่านทางพอร์ตสื่อสารอนุกรม อาร์เอส232 ด้วยชุดคำสั่ง เอที คอมมาน (AT Command) สามารถประยุกต์ใช้งานได้มากมายหลายรูปแบบ ไม่ว่าจะเป็นการรับส่ง สัญญาณแบบ วอยซ์ (Voice), เอสเอ็มเอส (SMS), ข้อมูล(Data), แฟ็ก (FAX) และยังรวมถึงการสื่อสารด้วย โพรโทคอล ทีซีพี/ไอพี ด้วย ซึ่งตามปรกติแล้ว ถึงแม้ว่าโมดูล ซิม300ซีแซต จะมีวงจร และ เฟิร์มแวร์ (Firmware) บรรจุไว้ภายในตัวเป็นที่เรียบร้อยแล้วก็ยังไม่สามารถนำไปใช้งานได้ โดยตรงทันทีเนื่องจากในการใช้งานจริงๆนั้น ผู้ใช้งานเองจำเป็นต้องออกแบบวงจรรอบนอกที่จำเป็นมาเชื่อมต่อกับขาสัญญาณของตัวโมดูลอีกในบางส่วน ไม่ว่าจะเป็นวงจรภาค เพาเวอร์ซัพพลาย, วงจรเชื่อมต่อกับ ซิม การ์ดรวมไปถึงวงจร ไล ไลน์เวอร์ ของ อาร์เอส232 เป็นต้น

2.2.2 คุณสมบัติของบอร์ด อีที-จีเอสเอ็ม ซิม300ซีแซต วี1.0

- มีสวิตช์แบบ Push-Button สำหรับใช้สั่ง เปิด-ปิด การทำงานของโมดูลภายในบอร์ด
- มี ซ็อกเกต ซิม รองรับ ซิม การ์ด พร้อมวงจร อีเอสดี (ESD) ป้องกัน ซิม เสียหาย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- มีวงจรถูกดูแล (Regulate) แยกอิสระ จำนวน 2 ชุด สามารถใช้กับแหล่งจ่ายภายนอก (Adaptor) ขนาดตั้งแต่ +5V ขึ้นไป สามารถจ่ายกระแสให้กับ โมดูล ซิม300ซีแซต (SIM300CZ) อย่างเพียงพอ
- มีวงจรถูกดูแล ขนาด 4.2V / 3A สำหรับจ่ายให้กับ โมดูล ซิม300ซีแซต ได้อย่างเพียงพอ สามารถใช้กับ SIM ของระบบ จีเอสเอ็ม 900MHz แบบ 2-Watt ได้อย่างไม่เกิดปัญหา
- มีวงจรถูกดูแล ขนาด 3.3V / 1A สำหรับจ่ายให้กับวงจรเชื่อมต่อภายนอกโดยไม่ต้องไปดึงไฟจากตัวโมดูลมาใช้ ป้องกันปัญหาโมดูลเสียหายจากวงจรภายนอกดึงกระแสเกินพิกัดและสะดวกต่อการออกแบบวงจรเชื่อมต่อเพิ่มเติม โดยไม่ต้องกังวลว่ากระแสจะไม่พอจ่ายให้กับอุปกรณ์
- มีวงจรถูกดูแล Line Driver สำหรับแปลงระดับสัญญาณลอจิกจากโมดูล ซิม300ซีแซต ให้เป็น อาร์เอส232 ระดับมาตรฐานครบทุกเส้นสัญญาณ ทั้งพอร์ตที่ใช้ในการสื่อสารสำหรับสั่งงานโมดูล และ พอร์ตสำหรับใช้ในการพัฒนาโปรแกรม (Debug) สามารถเชื่อมต่อกับพอร์ต อาร์เอส232 มาตรฐานได้ทันที
- มี แอลอีดี แสดงสถานะพร้อมในบอร์ด สำหรับแสดงสถานะของแหล่งจ่ายไฟ สถานะพร้อมทำงานของโมดูล สถานะในการเชื่อมต่อกับ เน็ตเวิร์ก (Network) และสถานะทำงาน (Power-On)/หยุดทำงาน (Power-OFF) ของโมดูล
- มีขั้วสำหรับเชื่อมต่อกับ แฮนด์เซต (Handset) (ชุดปากพูด และหูฟัง ของ โทรศัพท์บ้าน) โดยใช้ขั้วต่อแบบ RJ11มาตรฐาน พร้อมวงจร ฟิลเตอร์ (Voice Filter) สามารถนำชุด แฮนด์เซตของ โทรศัพท์บ้าน ต่อเข้ากับบอร์ดทางขั้วต่อแบบ อาร์เจ11 สำหรับใช้พูดคุย โทรออก และรับสายได้โดยสะดวก
- มี บัสเซอร์ พร้อมวงจรขับเพื่อสร้างสัญญาณเสียง ในกรณีมีการ โทรเรียกเข้ามายัง โมดูล
- มีจุดยึดเสาอากาศ สำหรับ ใช้เป็นจุดพักสำหรับเชื่อมต่อกับเสาอากาศแบบต่างๆ ได้โดยสะดวก
- มีขั้วต่อสำหรับติดตั้ง โมดูล ซิม300ซีแซต พร้อมเสารองและสกรูยึดโมดูลกับตัวบอร์ด
- มีจุดต่อสัญญาณอื่นๆที่เหลือจากโมดูล เช่น คีย์บอร์ด (Keyboard), ดิสเพลย์ (Display), จีพีไอโอ (GPIO), แบตเตอรี่ ชาร์จเจอร์ (Battery Charger) ฯลฯ สำหรับให้ผู้ใช้ต่อขยายไปยังวงจรที่ออกแบบเพิ่มเติมได้โดยง่ายและสะดวก

2.2.3 การสั่ง เปิด และ ปิด การทำงานของโมดูล

ตามปรกติแล้ว โมดูล ซิม300ซีแซต จะมีโหมดการทำงานอยู่หลายโหมด สามารถทำงานสั่งเปิดและ ปิดการทำงานของโมดูลได้ หลายวิธี

- สวิตช์ เปิด/ปิด เป็นการสั่ง เปิด และ ปิด การทำงานของ โมดูล ซิม300ซีแซต ด้วยการกด สวิตช์โดยสวิตช์ตัวนี้ จะเป็นแบบ Push-Button Switch (สวิตช์ กดติด-ปล่อยดับ) โดยเป็นการกำหนดสถานะทางลอจิกให้กับขาสัญญาณ PWRKEY(Pin17) ของ โมดูล โดยเมื่อกดสวิตช์จะเอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สำนักหอสมุดกลาง พระจอมเกล้าลาดกระบัง

เป็นโลจิก “0”เมื่อปล่อยสวิตช์จะเป็นโลจิก “1” โดยการทำงานของสวิตช์จะต้องทำการกด สวิตช์ต่อเนื่องกันเป็นเวลานานอย่างน้อย 2000ms (2 วินาที) จึงจะมีผลต่อการทำงานของ โมดูล โดยลักษณะการทำงานของสวิตช์ จะเป็นแบบ ทริกเกอร์(Toggle) กล่าวคือ ถ้า โมดูลอยู่ในสถานะของ หยุดทำงาน อยู่แล้วทำการกดสวิตช์ เป็นเวลาอย่างน้อย 2000ms (2 วินาที) จะเป็นการสั่งให้โมดูลกลับเข้าสู่ ทำงาน หรือพร้อมทำงาน แต่ถ้าหากว่าโมดูลอยู่ในสถานะของ ทำงาน อยู่แล้วทำการกดสวิตช์ เป็นเวลาอย่างน้อย 2000ms (2 วินาที) แล้วปล่อยจะเป็นการสั่งให้ โมดูลหยุดทำงานและกลับเข้าสู่สถานะ หยุดทำงาน

ตารางที่ 2.5 สถานะของ LED ในโหมดต่างๆ

LED สถานะ	Power-ON	Power-OFF
VBAT (แดง)	ติดสว่าง	ติดสว่าง
POWER (แดง)	ติดสว่าง	ดับ
NETLIGHT (เหลือง)	กระพริบ	ดับ
STATUS (เขียว)	ติดสว่าง	ดับ

หลังจากทำการสั่ง ทำงาน ในครั้งแรกนั้น ก่อนที่จะเริ่มต้นส่งคำสั่งใดๆให้กับ โมดูล ควรรอให้ ตัวโมดูลพร้อมเสียก่อน โดยจะมีข้อความ “Call Ready” ปรากฏให้เห็น ในกรณีที่กำหนด บอร์ดเรต เป็นแบบ ออโต้ บอร์ดเรต ไว้ (AT+IPR=“0”) เมื่อทำการ ทำงานจะได้ผลดังตัวอย่าง

Call Ready

รูปที่ 2.3 จีพีอาร์เอส โมดูล อยู่สถานะพร้อมใช้งาน

ในกรณีที่กำหนด บอร์ดเรต (Baudrate) เป็นแบบ Fix บอร์ดเรต ไว้ (AT+IPR= board rate) เมื่อทำการ สั่งให้โมดูล ทำงาน แต่ละครั้งจะได้ผลดังตัวอย่าง

```
RDV
+CFUN: 1
+CPIN: READY
Call Ready
```

รูปที่ 2.4 จีพีอาร์เอส โมดูล พร้อมใช้งาน

2.2.4 การติดต่อสื่อสารกับโมดูล ชิม300ซีแซต

การติดต่อสื่อสารกับ โมดูล ชิม300ซีแซต ของบอร์ด อีที-จีเอสเอ็ม ชิม300ซีแซต นั้นจะเชื่อมต่อผ่านพอร์ตสื่อสารอนุกรม อาร์เอส232 โดยใช้ขั้วต่อแบบ ดิบี 9 ตัวเมีย จัดเรียงสัญญาณตามมาตรฐาน อาร์เอส232-ดีซีอี สามารถนำไปเชื่อมต่อกับสัญญาณ อาร์เอส232-ดีทีอี มาตรฐาน โดยใช้สาย ดิบี 9 แบบต่อตรง ได้ทันที โดยสัญญาณทั้งหมดที่ ดิบี 9 นี้ได้ผ่านวงจร ไล ไคร์เวอร์ เพื่อแปลงสัญญาณระดับ โลจิก จากโมดูล ให้เป็นสัญญาณระดับมาตรฐาน อาร์เอส232 เป็นที่เรียบร้อยแล้ว ซึ่งถ้าต้องการนำไปเชื่อมต่อกับ อาร์เอส232 (Com Port) ของคอมพิวเตอร์ PC ก็สามารถทำการเชื่อมต่อกันโดยตรงได้ทันที โดยไม่ต้องทำการสลบสายสัญญาณใดๆทั้งสิ้น โดยสัญญาณเชื่อมต่อทางด้าน โมดูล ชิม300ซีแซต นั้น จะมีทั้งหมด 8 เส้นสัญญาณ ซึ่งในการเชื่อมต่อใช้งานนั้น จะต่อให้ครบทั้ง 8 เส้น หรือ จะเลือกต่อเพียง 3 เส้น (RXD, TXD และ GND) ก็ได้เช่นเดียวกัน โดยสามารถกำหนดได้จากการ Setup ค่า Configuration และคำสั่งใช้งาน โดยสัญญาณการเชื่อมต่อ อาร์เอส232

2.2.5 ตัวอย่างการใช้งาน เอที คอมมานเพื่อสั่งงานโมดูล ชิม300ซีแซต

โมดูล จีเอสเอ็ม/จีพีอาร์เอส รุ่น ชิม300ซีแซต ถูกออกแบบให้ทำหน้าที่เหมือน โมเต็ม โดยจะใช้การติดต่อสั่งงานและสื่อสารกับ โมดูล ผ่านทางพอร์ตสื่อสาร อาร์เอส 232 รองรับ บอร์ดเรต ตั้งแต่ 1200-115200 BPS โดยใช้ชุดคำสั่งแบบ เอที คอมมานซึ่งจะมีรูปแบบการใช้งานเหมือนกับ โมเต็ม มาตรฐานทั่วไปเพียงแต่จะมีการเพิ่มเติม อ็อพชั่น และคำสั่งพิเศษอื่นๆเพิ่มเติมขึ้นมาอีก เพื่อให้เหมาะสมและสอดคล้องกับความสามารถในการทำงานของ โมดูล ได้อย่างครบถ้วนสำหรับ รายละเอียดการใช้คำสั่ง เอที คอมมานที่จะใช้ สำหรับติดต่อสั่งงาน โมดูล ชิม300ซีแซต ไม่ว่าจะเป็นรูปแบบคำสั่ง และ หน้าที่การทำงานของแต่ละคำสั่ง ผู้ใช้สามารถศึกษารายละเอียดต่างๆได้จากคู่มือคำสั่ง เอที คอมมาน(ไฟล์เอกสารชื่อ SIM300C_ATC_V1.06.PDF) ในแผ่น ซีดีรอมซึ่งในที่นี้จะขอแนะนำถึงวิธีการและรูปแบบการใช้งานคำสั่งแบบย่อๆ แบบพอสังเขป เพื่อเป็นแนวทางให้ผู้เริ่มต้นได้ใช้เป็นแนวทางและประกอบความเข้าใจในการศึกษาการทำงาน of คำสั่งต่างๆต่อไป โดยรูปแบบของคำสั่งต่างๆที่เป็น เอที คอมมานนั้น จะเริ่มต้นคำสั่งด้วยรหัส แอสกี(ASCII) ของตัวอักษร 2 ตัว คือ “A” และ “T” ซึ่งจะใช้ตัวอักษรแบบพิมพ์เล็ก หรือ พิมพ์ใหญ่ก็ได้ มีความหมายเหมือนกัน จากนั้นก็จะตามด้วยรหัสคำสั่ง และ อ็อพชั่น(Oftion) ต่างๆของคำสั่ง(ถ้ามี) โดยทุกๆคำสั่งจะต้องจบด้วยรหัส Enter หรือ 0DH (13) เสมอ เช่นคำสั่ง รีเซ็ตจะใช้รูปแบบคำสั่งเป็น “ATZ” หรือ “atz” ก็สามารถใช้งานได้ถูกต้องเหมือนกัน โดยรูปแบบคำสั่งทั้งหมดจะแบ่งออกเป็น 4 แบบด้วยกัน คือ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 2.6 การใช้งาน เอที-คอมมาน

การใช้งาน	รูปแบบคำสั่ง	รายละเอียด
ทดสอบคำสั่ง	AT+<x>=?	รูปแบบการใช้คำสั่งแบบนี้ จะใช้สำหรับสั่งอ่านค่ารูปแบบและพารามิเตอร์ต่างๆของคำสั่ง โดยถ้าคำสั่งนั้นมีอยู่จริง โมดูลจะตอบรับด้วยการพิมพ์ค่าของพารามิเตอร์ต่างๆของคำสั่งที่มีอยู่ทั้งหมดให้ทราบ
อ่านค่าพารามิเตอร์	AT+<x>?	รูปแบบการใช้คำสั่งแบบนี้ จะใช้สำหรับสั่งอ่านค่าพารามิเตอร์ที่กำหนดไว้แล้วของคำสั่งนั้นๆ โดยโมดูลจะตอบรับด้วยการพิมพ์ค่าพารามิเตอร์ปัจจุบันที่กำหนดไว้แล้วให้ทราบ
กำหนดค่าการทำงาน	AT+<x>=<...>	รูปแบบการใช้คำสั่งแบบนี้ จะใช้สำหรับสั่งเขียนหรือกำหนดค่าพารามิเตอร์ให้กับคำสั่ง เช่น การกำหนดค่า บอร์ดเรต
สั่งให้ทำงาน	AT+<x>	รูปแบบการใช้คำสั่งแบบนี้ จะใช้สำหรับสั่งงานให้โมดูลปฏิบัติตามคำสั่งที่ต้องการ เช่น การสั่งรีเซ็ต (ATZ)

2.2.6 การกำหนด โฟลคอนโทรล

โมดูล ซิม300ซีแฮต สามารถกำหนด โฟลคอนโทรล (Flow Control) หรือ รูปแบบการตรวจสอบความพร้อมในการสื่อสารและรับส่งข้อมูลได้ด้วย ซึ่ง โฟลคอนโทรลจะมีความสำคัญเป็นอย่างมาก เนื่องจากการประมวลผลของอุปกรณ์ต่างๆจะมีความช้าเร็วที่แตกต่างกัน เมื่อมีการรับส่งข้อมูลที่มีจำนวนข้อมูลมากๆแบบต่อเนื่องนั้น ถ้าฝ่ายรับไม่พร้อมรับข้อมูลแต่ฝ่ายส่งยังคงส่งข้อมูลออกไป ก็จะทำให้ข้อมูลสูญหายและเกิดความผิดพลาดขึ้นได้ โดย ซิม300ซีแฮต เองรองรับการตรวจสอบความพร้อมหรือ โฟลคอนโทรลได้ 2 แบบ คือ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- ซอร์ฟแวร์ โฟลคอนโทรล (XON/XOFF Flow Control) เป็นการตรวจสอบความพร้อมด้วยซอร์ฟแวร์โดยจะใช้รหัส XOFF (13H) เป็นตัวส่งหยุดการส่งข้อมูลจากฝ่ายส่ง และใช้รหัส XON (11H) เพื่อบอกหรืออนุญาตให้ฝ่ายส่งเริ่มต้นส่งข้อมูลลำดับต่อไปมายังโมดูลได้ โดยการใช้ โฟลคอนโทรลแบบนี้เหมาะกับการเชื่อมต่อกับอุปกรณ์ที่ไม่มีสัญญาณตรวจสอบความพร้อม เช่น ไมโครคอนโทรลเลอร์หรืออุปกรณ์ที่ใช้การต่อสายสัญญาณเพียง 3 เส้น (RXD, TXD และ GND)
 - ฮาร์ดแวร์ ซอร์ฟแวร์ โฟลคอนโทรล (อาทีเอส (RTS)/ซีทีเอส โฟลคอนโทรล (CTS Flow Control)) เป็นการตรวจสอบความพร้อมด้วยสัญญาณทางฮาร์ดแวร์ โดยการใช้การ เอ็กทีฟ (“LOW”) สัญญาณ ซีทีเอส เพื่อบอกให้ฝ่ายส่งหยุดการส่งข้อมูลเมื่อ โมดูลไม่พร้อมรับข้อมูล และในทางกลับกันก่อนการส่งข้อมูลกลับออกไปมันจะตรวจสอบสถานะของ อาทีเอส ว่า เอ็กทีฟ อยู่หรือไม่ ถ้า เอ็กทีฟ (“LOW”) แสดงว่าฝ่ายรับยังไม่พร้อมรับมันจะหยุดรอจนกว่า อาทีเอส จะเป็น “HIGH”
- การกำหนด โฟลคอนโทรล เป็น XON/XOFF จะใช้คำสั่ง “AT+IFC=1, 1”
 การกำหนด โฟลคอนโทรล เป็น RTS/CTS จะใช้คำสั่ง “AT+IFC=2, 2”

2.3 General packet Radio Service (จีพีอาร์เอส)

ตั้งแต่อดีตจนถึงปัจจุบัน โทรศัพท์มือถือได้มีการพัฒนาเทคโนโลยีมาอย่างต่อเนื่อง เพื่อให้ติดต่อสื่อสารได้อย่างไร้ขีดจำกัด ทำให้ผู้ให้บริการเครือข่ายโทรศัพท์ต้องพัฒนาเทคโนโลยีที่ใช้ในการติดต่อสื่อสารให้มีประสิทธิภาพสูงขึ้นจนในปัจจุบันมีเทคโนโลยีที่เรียกว่า จีพีอาร์เอส

จีพีอาร์เอส ย่อมาจากคำว่า General Packet Radio Service เป็นวิวัฒนาการของการสื่อสารข้อมูลไร้สายแบบ แพค สวิตซ์ คือ การแบ่งข้อมูลออกเป็นส่วนเล็ก ๆ ที่เรียกว่า แพ็กเกจ (Packet) ซึ่งมีความสามารถในการส่งผ่านข้อมูลตรงเข้าได้ดีกว่าแบบเดิม ทำให้สามารถตรวจสอบความผิดพลาดในการส่ง และยังช่วยเพิ่มอัตราการส่งข้อมูลสูงขึ้นอีกด้วย เทคโนโลยีจีพีอาร์เอสนี้สร้างขึ้นมาเพื่อช่วยให้สามารถทำธุรกรรมต่างๆ ผ่านโทรศัพท์เคลื่อนที่ได้ง่ายและสะดวกขึ้น

อย่างไรก็ดี จีพีอาร์เอส ไม่ได้เป็นลักษณะที่จะสามารถให้บริการได้ด้วยตัวของระบบเอง แต่ตัวมันเองเป็นเพียงแค่ Bearer ให้กับ แอปพลิเคชัน ต่างๆ ที่ต้องการใช้ความเร็วที่เพิ่มมากกว่าปกติในระบบ จีเอสเอ็ม ที่เคยรองรับอยู่เดิมมาก่อน และระบบ จีพีอาร์เอส จะต้องต่อไปยัง แพ็กเกจ คาด้า เน็ตเวิร์ค (Packet data network) ที่เป็น ไอพี เน็ตเวิร์ค (IP Network) อีกต่อหนึ่งดังนั้นผู้ให้บริการเครือข่ายโทรศัพท์เคลื่อนที่ ที่จะเปิดให้ใช้ในระบบ จีพีอาร์เอส ได้นั้นจะต้องทำการติดตั้งระบบเครือข่าย ที่ประกอบด้วยหน่วยหลักๆ 2 หน่วยด้วยกันคือ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
 ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1. SGSN (เซิร์ฟวิ้ง จีพีอาร์เอส ซัพพอร์ต อาทีเอส โหนด)
2. GGSN (เกตเวย์ จีพีอาร์เอส ซัพพอร์ต โหนด)

โดยทั้งสองหน่วยหลักขององค์ประกอบนี้จะถูกเชื่อมต่อเข้าด้วยกัน โดยมีอุปกรณ์อื่นๆ เป็นตัวช่วยเพื่อไปร่วมใช้ Radio interface จาก Base Station โดยผ่านตัวควบคุมที่เรียกว่า PCU (Packet Control Unit) ที่ติดตั้งไว้ที่ BSC (Base Station Controller) อันทั้งนี้อาจมองได้ว่า จีพีอาร์เอส เน็ตเวิร์ก เป็นอีกระบบหนึ่ง ซึ่งเข้าถึงโทรศัพท์มือถือผ่านทาง Radio Interface ของระบบ จีเอสเอ็ม เน็ตเวิร์ก เดิม โดยเป็นบริการที่เกี่ยวข้องเนื่องกับการรับส่งข้อมูลเป็น แพค อีทีโดยตรง

2.3.1 คุณสมบัติเด่นหลักๆ ของระบบ จีพีอาร์เอส มีดังนี้

การโอนถ่ายข้อมูลที่มีความสามารถในการ รับ-ส่งผ่านเครือข่ายอินเทอร์เน็ตได้สูงถึง 9 - 40 kbps ซึ่งจะทำให้สามารถรับ-ส่งข้อมูลที่เป็น วิดีโอ เมล์ (VDO Mail) หรือ ภาพเคลื่อนไหวต่างๆ ได้ พร้อมทั้งเชื่อมต่อกับเครือข่ายอินเทอร์เน็ตได้เร็วและมีประสิทธิภาพมากกว่าเดิมรวมถึงการ ดาวน์โหลด (Down load) และ อัปโหลด (Upload) ได้ง่ายยิ่งขึ้น

Always On การเชื่อมต่อเครือข่ายและโอนถ่ายข้อมูลสามารถดำเนินต่อไป แม้ในขณะที่มีสายติดต่อเข้ามาก็ตาม จึงทำให้การ โอนถ่ายข้อมูล ไม่ขาดตอนลง

Wireless Internet ที่เชื่อมต่อเข้ากับ เทอร์มินอล (Terminal) เช่น พีดีเอ หรือ โน้ตบุค (Notebook) สามารถที่จะ โอนถ่ายข้อมูลได้เร็วขึ้นจากที่เคยเป็นอยู่

2.3.2 ประโยชน์ของ จีพีอาร์เอส

● ประหยัดค่าใช้จ่าย

เทคโนโลยี จีพีอาร์เอส จะทำให้การคิดค่าบริการในการใช้อินเทอร์เน็ต ขึ้นอยู่กับช่วงเวลาในการรับ และส่งข้อมูล ไม่ใช่ช่วงเวลาในการเชื่อมต่อ ซึ่งจะทำให้ผู้ใช้ จ่ายเพียงแค่อัตราค่าบริการในการดาวน์โหลด และอัปโหลดเท่านั้น

● รวดเร็วยิ่งขึ้น

จีพีอาร์เอส จะช่วยให้ท่านเชื่อมต่อ และรับข้อมูลจ่าง ๆ ผ่านอินเทอร์เน็ตด้วยระยะเวลาที่รวดเร็วกว่าโทรศัพท์เคลื่อนที่ในระบบ จีเอสเอ็ม ทั่วไป ทำให้การเข้าสู่ เว็บ หรือการรับส่ง อีเมล เป็นไปอย่างสะดวก และง่ายดาย

● คุ่มค่า

เพราะมีค่าใช้จ่ายน้อย แต่รับผลตอบแทนจากการรับ-ส่งข้อมูลอย่างมากมาย

● นำใช้

จีพีอาร์เอส ทำให้ท่านได้รับข้อมูลในทุกรูปแบบไม่ว่าจะเป็นในรูปแบบข้อความ หรือรูปแบบมัลติมีเดีย ซึ่งประกอบไปด้วยรูปภาพ เสียง และวิดีโอ ทำให้การติดต่อสื่อสารของคุณผ่านโทรศัพท์เคลื่อนที่ไม่น่าซ้ำซากอีกต่อไป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.3.3 รูปแบบการให้บริการของ จีพีอาร์เอส

- **Textual And Visual Information**

บริการนี้เป็นจุดแตกต่างอย่างแรกที่จีพีอาร์เอสเหนือกว่าจีเอสเอ็มทั่วไป โดยสามารถส่งข้อมูลที่เป็นตัวอักษร หรือรูปภาพกราฟิกไปยังโทรศัพท์มือถือได้อย่างสะดวกรวดเร็วซึ่งจะทำให้จีพีอาร์เอสแทรกซึมเข้าสู่การใช้งานของคนทั่วไป ได้ทั้งข่าวความเคลื่อนไหวและข้อมูลที่คนส่วนใหญ่สนใจ รวมทั้งบริการต่างๆ ที่จะเสริมเข้ามาในอนาคต

- **Still Images**

เป็นการส่งภาพนิ่งความละเอียดสูงไปมาระหว่างเครื่องด้วยกันได้ ทำให้สามารถส่งผ่านความรู้สึกดีๆ ผ่านภาพถ่าย หรือการ์ตูนสวยๆ ได้เลย รวมทั้งภาพที่ถ่ายได้จากกล้องดิจิทัล ก็สามารถโอนแล้วส่งต่อไปได้ทันที

- **Moving Images**

นอกเหนือจากภาพนิ่งแล้วภาพเคลื่อนไหวก็สามารถส่งต่อกันไปได้เช่นกัน เช่น การประชุมทางไกล หรือ การส่งภาพจากกล้องวงจรปิดไปยังโทรศัพท์มือถือในกรณีประยุกต์ใช้กับระบบรักษาความปลอดภัย

- **chat**

เป็นคุณสมบัติที่คงจะถูกใจของผู้รักการคุยแบบไม่ใช้เสียง ซึ่งสามารถสนทนากันได้ทั้งแบบเป็นคู่ หรือเป็นกลุ่มได้อย่างสบายใจ ซึ่งจุดเด่นที่สำหรับ สามารถ แชท (Chat) ได้ทุกที่ที่อยากจะ แชท

- **Web Browsing**

เป็นการเข้าสู่ World Wide Web ด้วยการ ใช้โทรศัพท์มือถือ ซึ่งความเร็วมีให้เลือกตั้งแต่ 56 Kbps ไปจนถึง 112 Kbps การท่องเว็บจึงไม่ใช่เรื่องยากอีกต่อไป แม้รูปแบบการแสดงผลจะแตกต่างจากการท่องเว็บโดยใช้เครื่องคอมพิวเตอร์อยู่บ้าง

- **E-Mail**

เป็นบริการพื้นฐานที่มีคนนิยมใช้งานมากที่สุดสำหรับการส่งข้อความ โดยจะมีการใช้ในรูปแบบของ เอสเอ็มเอส (Short Message Service) ที่เรารู้กันดีอยู่แล้ว

- **File Transfer**

เป็นบริการ โอนถ่ายไฟล์ข้อมูลซึ่งน่าจะใช้งานกันอย่างแพร่หลายขึ้น จีพีอาร์เอส เพราะความเร็วจะเหนือกว่าการใช้งานผ่านโมเด็มกับโทรศัพท์พื้นฐานที่เราใช้กันอยู่ในปัจจุบันมาก โดยจะรองรับกับ โพร โตคอล เอฟทีพี และแอปพลิเคชันที่อ่านข้อความอย่าง อะ โคเบท ลิตเตอร์ (Acrobat Reader)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- **Audio**

แน่นอนว่าโทรศัพท์ต้องมีเสียง แต่บริการด้านเสียงของ จีพีอาร์เอส จะเหนือกว่าโทรศัพท์มือถือเดิม ๆ ที่เรารู้จัก เนื่องจากความคมชัดของสัญญาณเสียงที่เหนือกว่า และยังประยุกต์ใช้ในการเก็บไฟล์เสียงเพื่อนำไปใช้งานในด้านต่างๆ ด้วย เช่น การวิเคราะห์รายละเอียดของเสียงในงานของตำรวจ เป็นต้น

- **Remote LAN Access**

เราสามารถเข้าถึงเครือข่ายคอมพิวเตอร์ โดยใช้โทรศัพท์มือถือแทนเบอร์โทรศัพท์กับเครื่องคอมพิวเตอร์ที่บ้านได้อย่างง่ายดาย ซึ่งความเร็วในการส่งถ่ายข้อมูลจะเหนือกว่าโทรศัพท์พื้นฐานทั่วไป

- **Vehicle Positioning**

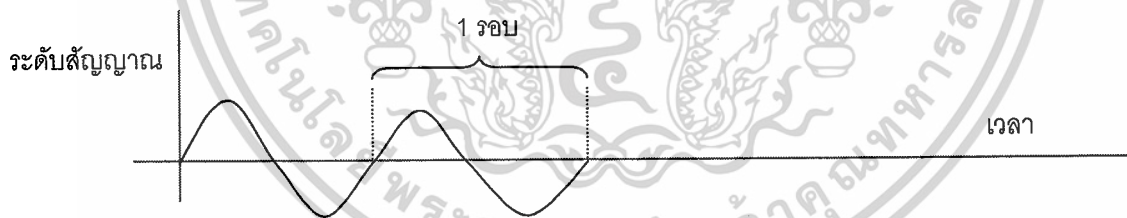
เป็นความสามารถในการบอกตำแหน่งของยานพาหนะที่เราใช้อยู่ โดยจะสามารถเชื่อมต่อกับดาวเทียม ซึ่งจะสามารถบอกตำแหน่งที่เราอยู่โดยอ้างอิงกับเครื่องโทรศัพท์มือถือได้อย่างแม่นยำ

2.4 การสื่อสารข้อมูล (Data Transmission)

2.4.1 ชนิดของสัญญาณทางอิเล็กทรอนิกส์

2.4.1.1 สัญญาณแบบอนาลอก (Analog signal)

เป็นสัญญาณแบบต่อเนื่อง มีหน่วยเป็น Hertz (Hz) เช่น สัญญาณเสียงในสายโทรศัพท์

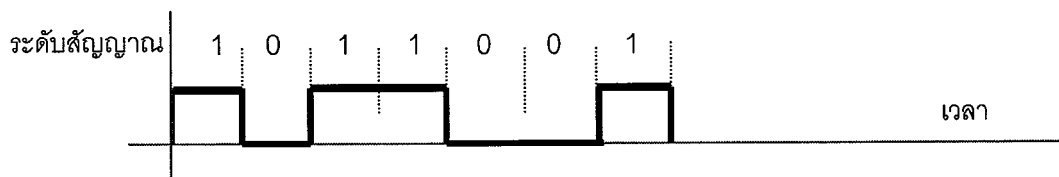


รูปที่ 2.5 สัญญาณอนาลอก

2.4.1.2 สัญญาณแบบดิจิทัล (Digital signal)

เป็นสัญญาณแบบไม่ต่อเนื่องประกอบด้วยระดับสัญญาณสูงสุด และระดับสัญญาณต่ำสุด มีหน่วยเป็น Bit per second (bps) เช่น สัญญาณที่คอมพิวเตอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.6 สัญญาณดิจิทัล

2.4.2 ทิศทางการสื่อสารข้อมูล

2.4.2.1 แบบทิศทางเดียว (Simplex)

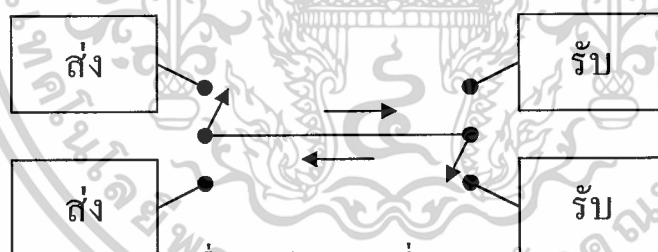
ข้อมูลจะถูกส่งจากทิศทางหนึ่งไปยังอีกทิศทางโดยจะส่งย้อนกลับไม่ได้ เช่น สถานีโทรทัศน์กับโทรทัศน์



รูปที่ 2.7 การสื่อสารข้อมูลแบบทิศทางเดียว

2.4.2.2 แบบกึ่งสองทิศทาง (Half Duplex)

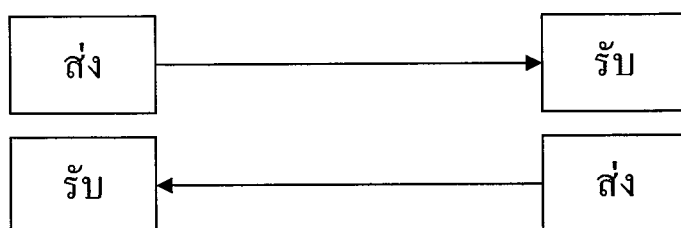
ข้อมูลสามารถส่งสลับกันได้ แต่ต้องคนละเวลากัน เช่น วิทยุสื่อสาร



รูปที่ 2.8 การสื่อสารข้อมูลแบบกึ่งสองทิศทาง

2.4.2.3 แบบสองทิศทาง (Full Duplex)

ข้อมูลสามารถส่งสลับกันได้พร้อมๆ กันทั้ง 2 ทิศทางในเวลาเดียวกัน เช่น โทรศัพท์



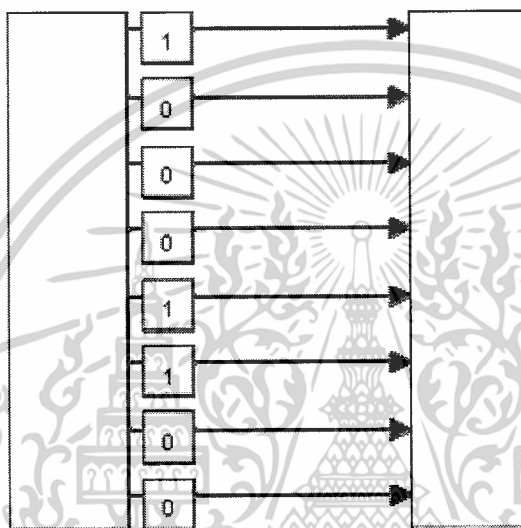
รูปที่ 2.9 การสื่อสารข้อมูลแบบทิศทางเดียว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.4.3 ประเภทของการสื่อสารข้อมูล

2.4.3.1 การสื่อสารข้อมูลแบบขนาน (Parallel Transmission)

ในรูปแบบการส่งผ่านข้อมูลในลักษณะนี้ทุกบิตที่แทนข้อมูลหนึ่งตัวอักษร จะถูกส่งผ่านไปทางสายส่ง หลายๆ สายขนานกันไป ดังนั้นทุกบิตจะเดินทางถึงผู้รับพร้อมๆ กัน และจำนวนสายส่งเพื่อให้เกิดช่องทางการส่งจะต้องมีจำนวนเท่ากับจำนวนบิตที่เข้ารหัสแทนตัวอักษรในแต่ละระบบ เช่นการส่งผ่านข้อมูลที่เข้ารหัสแบบ แอสกี ก็จำเป็นต้องใช้ช่องทางการส่ง 8 ช่องทาง จึงจะทำให้ทุกบิตวิ่งผ่านสายส่งขนานกันไปได้อย่างรูปที่ 2.10



รูปที่ 2.10 การส่งข้อมูลแบบขนาน

จากรูปแสดงให้เห็นว่าทุกบิตที่เข้ารหัสตัวอักษรตัวหนึ่งๆ จะเดินทางขนานกันไป โดยเริ่มจากต้นทางผ่านสายส่งสัญญาณที่มีอยู่ด้วยกัน 8 เส้น ไปยังปลายทาง ดังนั้นปลายทางจะได้รับทุกบิตของตัวอักษรพร้อมๆ กัน การส่งผ่าน ข้อมูลแบบขนานนี้ ส่วนมากจะใช้ในการส่งผ่านข้อมูลในระยะใกล้ อันได้แก่การเคลื่อนย้ายข้อมูลระหว่างเครื่องคอมพิวเตอร์กับอุปกรณ์รอบข้างของมันเช่นระหว่างเครื่องคอมพิวเตอร์กับเครื่องพิมพ์ เพราะสามารถทำได้ด้วยความเร็วสูง และถ้านำไปใช้ในระยะไกลๆ จะต้องเสียค่าใช้จ่ายสูง เนื่องจากต้องใช้สายส่งเท่ากับจำนวนบิตที่ใช้เข้ารหัสแทนข้อมูลตัวอักษร

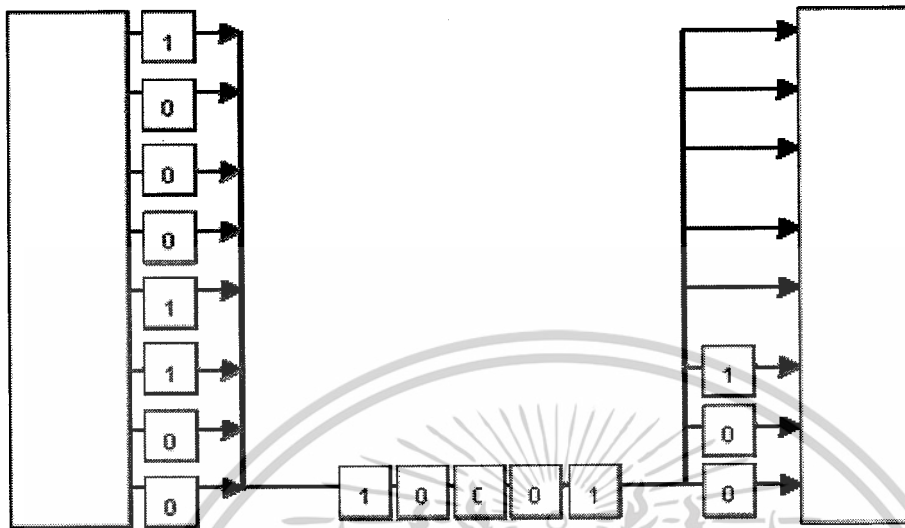
2.4.3.2 การสื่อสารข้อมูลแบบอนุกรม (Serial Transmission)

รูปแบบการส่งผ่านข้อมูลในลักษณะนี้ทุกบิตที่เข้ารหัสแทนข้อมูลหนึ่งตัวอักษรจะถูกส่งผ่านไปทางสายส่งเรียงลำดับกันไปทีละบิตในสายส่งเพียงเส้นเดียว ดังรูปที่ 2.11

จากรูปตัวอักษรจะประกอบด้วย 8 บิต เรียงเป็นลำดับ ข้อมูลจะถูกส่งออกมาทีละบิตระหว่างต้นทาง และปลายทาง และปลายทางจะรวบรวมบิตเหล่านี้ทีละบิตจนครบ 8 บิต เป็น 1

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตัวอักษร จะเห็นว่าการส่งข้อมูลแบบนี้จะช้ากว่าแบบขนาน แต่ค่าใช้จ่ายจะถูกกว่าแบบขนาน ซึ่งเหมาะสำหรับการส่งระยะทางไกลๆ

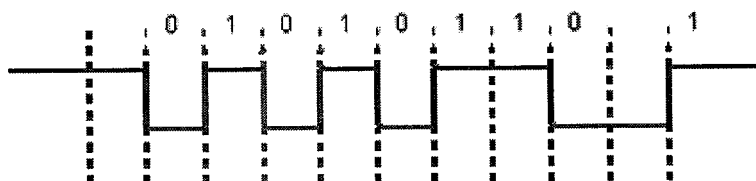


รูปที่ 2.11 การส่งข้อมูลแบบอนุกรม

โดยทั่วไปแล้วการส่งข้อมูลนั้นจะประกอบไปด้วยกลุ่มของตัวอักษร ดังนั้นในการส่งข้อมูลแบบอนุกรมนี้จึงเกิดปัญหาขึ้นว่า แล้วต้นทางและปลายทางจะทราบได้อย่างไรว่า จะแบ่งแต่ละตัวอักษรตรงบิดใด จึงเกิดวิธีการสื่อสารข้อมูลขึ้น 2 แบบคือ การสื่อสารแบบอะซิงโครนัส (Asynchronous Transmission) และการสื่อสารแบบซิงโครนัส (Synchronous Transmission)

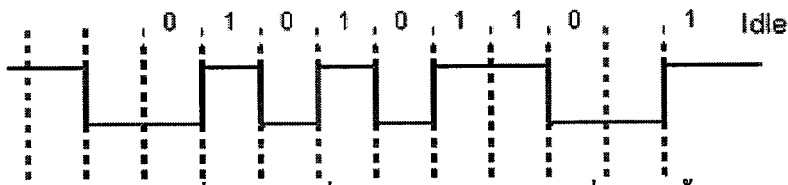
2.4.3.2.1 การสื่อสารแบบอะซิงโครนัส (Asynchronous Transmission)

หรือเรียกอีกอย่างหนึ่งว่าเป็น การสื่อสารแบบระบุจุดเริ่มต้น และจุดสิ้นสุด (Start-Stop Transmission) ลักษณะของสัญญาณที่ใช้ในการติดต่อสื่อสารกันจะประกอบไปด้วย บิตเริ่มต้น (start bit) บิตของข้อมูลที่สื่อสาร (transmission data) จำนวน 8 บิต บิตตรวจข้อผิดพลาด (parity bit) และบิตสิ้นสุด (stop bit) สำหรับบิตตรวจข้อผิดพลาดจะใช้หรือไม่ใช้ก็ได้ ดังนั้นสัญญาณจึงต้องประกอบด้วยส่วนประกอบอย่างน้อย 3 ส่วน ดังรูปที่ 2.12 และ 2.13



รูปที่ 2.12 การสื่อสารแบบอะซิงโครนัสที่ไม่ได้ใช้พาริตีบิต

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



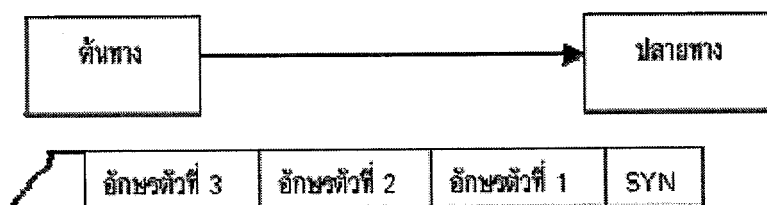
รูปที่ 2.13 การสื่อสารแบบอะซิงโครนัสที่ใช้พาริตีบิต

จากรูปจะเห็นว่าขณะที่ไม่มีข้อมูลส่งออกมาสถานะของการส่งจะเป็นแบบว่าง (Idle) ซึ่งจะมีระดับของสัญญาณเป็น 1 ตลอดเวลา เพื่อความแน่ใจว่าปลายทาง หรือฝ่ายรับยังคงติดต่อกับต้นทาง หรือฝ่ายส่งอยู่ เมื่อเริ่มจะส่งข้อมูลสัญญาณของอะซิงโครนัสจะเป็น 0 หนึ่งช่วงสัญญาณนาฬิกา ซึ่งบิตนี้เราเรียกว่าบิตเริ่มต้น ตามหลังของบิตเริ่มต้นจะเป็นบิตข้อมูลสำหรับ 1 ตัวอักษร ตามหลังบิตข้อมูลก็จะเป็นบิตตรวจข้อผิดพลาด แล้วจะตามด้วยบิตสิ้นสุด ถ้าไม่ใช่บิตตรวจข้อผิดพลาด ตามหลังบิตข้อมูลก็จะเป็นบิตสิ้นสุดเลย หลังจากนั้นถ้าไม่มีข้อมูลส่งออกมาสัญญาณจะกลับไปอยู่ที่สถานะแบบว่างอีก เพื่อรอการส่งข้อมูลต่อไป

จะเห็นว่า การสื่อสารแบบอะซิงโครนัส มีลักษณะเป็น ไปทีละตัวอักษร และสัญญาณที่ส่งออกมา มีบางส่วนเป็นบิตเริ่มต้น บิตสิ้นสุด และบิตตรวจข้อผิดพลาด ทำให้ความเร็วในการส่งข้อมูลต่อวินาทีน้อยลงไป เนื่องจากต้อง สูญเสียช่องทางการสื่อสารให้กับ บิตเริ่มต้น บิตสิ้นสุด และบิตตรวจข้อผิดพลาด (ถ้ามีใช้) ตลอดเวลา การสื่อสารแบบอะซิงโครนัสนี้มักใช้ในการติดต่อระหว่างคอมพิวเตอร์กับอุปกรณ์รอบข้าง

2.4.3.2.2 การสื่อสารแบบซิงโครนัส (Synchronous Transmission)

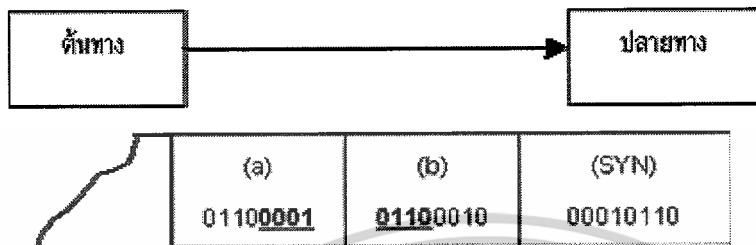
การสื่อสารแบบซิงโครนัสนี้มักใช้ในการติดต่อกันระหว่างคอมพิวเตอร์ ซึ่งจะทำการจัดกลุ่มของข้อมูลเป็นกลุ่มๆ และทำการส่งข้อมูลทั้งกลุ่มไปพร้อมกันในที่เดียว เราเรียกกลุ่มของข้อมูลนี้ว่า บล็อกของข้อมูล (Block of Data) ซึ่งตัวอักษรตัวแรก และตัวถัดไปที่อยู่ในบล็อกเดียวกันจะไม่มีอะไรมาคั่นเหมือนอย่างแบบอะซิงโครนัส ที่ต้องใช้บิตเริ่มต้น และบิตสิ้นสุดคั่นทุกๆ ตัวอักษร แต่จะมีข้อมูลเริ่มต้นซึ่งเป็นลักษณะของบิตพิเศษที่ส่งมาเพื่อให้รู้ว่านั่นคือ จุดเริ่มต้นของกลุ่มตัวอักษรที่กำลังส่งเรียงกันเข้ามา เช่น อักขระซิง (SYN character) โดยที่อักขระซิงมีรูปแบบบิต คือ 00010110 ตัวอย่างของการส่งแสดงได้ดังรูปที่ 2.14



รูปที่ 2.14 การสื่อสารแบบซิงโครนัส

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากรูปที่ 2.15 เมื่อสายทางตรวจพบอักขระซิง หรือ 00010110 แล้วจะทราบได้ทันทีว่า บิตที่ตามมาคือบิตตัวอักษรแต่ละตัว แต่การใช้อักขระซิงเพียงตัวเดียวอาจเกิดข้อผิดพลาดได้ เช่น ถ้าเราส่งตัวอักษร b และตัวอักษร a ติดต่อกันไป ซึ่งตัวอักษร b มีรูปแบบบิตคือ 01100010 และตัวอักษร a มีรูปแบบบิตคือ 01100001 การส่งจะแสดงได้ดังรูปที่ 2.15

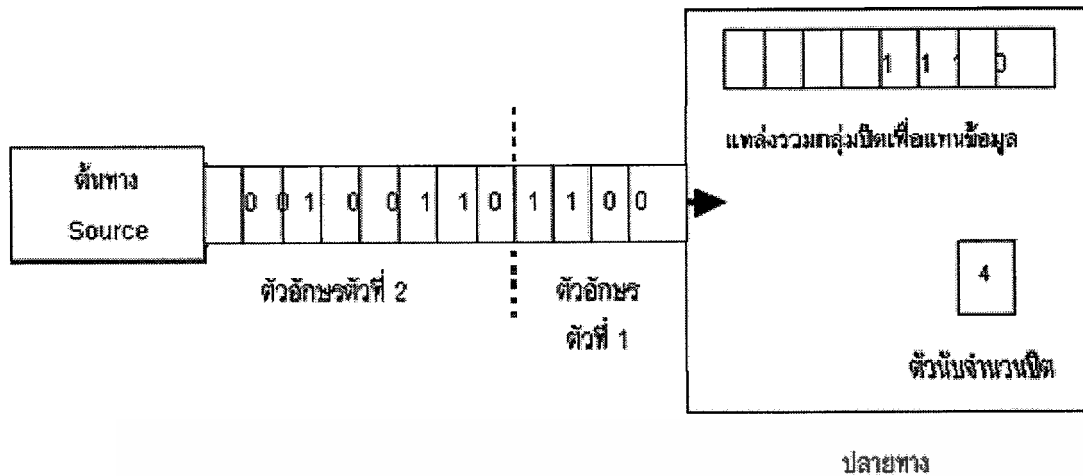


รูปที่ 2.15 ตัวอย่างการใช้อักขระซิงในการสื่อสารแบบซิงโครนัส

จะเห็นว่าเครื่องปลายทางจะตรวจพบอักขระซิงระหว่างบิตของตัวอักษร b และตัวอักษร a ทำให้เข้าใจว่าบิตต่อไปจะเป็นบิตของกลุ่มข้อมูล ซึ่งจะทำให้การรับข้อมูลนั้นเกิดผิดพลาดขึ้นได้ ดังนั้นจึงแก้ปัญหาด้วยการใช้อักขระซิง 2 ตัวต่อกันเป็นลักษณะของบิตพิเศษที่บอกให้ทราบว่าเป็นจุดเริ่มต้นบิตของกลุ่มข้อมูล ตัวอย่างของการใช้อักขระซิง 2 ตัวในการสื่อสารแบบซิงโครนัส และการตัดแฉของบิตข้อมูลออกเป็นกลุ่มทีละ 8 บิต เพื่อแทนข้อมูลแสดงได้ดังรูปที่ 2.16 และ 2.17



รูปที่ 2.16 ตัวอย่างการใช้อักขระซิง 2 ตัวในการสื่อสารแบบซิงโครนัส



รูปที่ 2.17 การตัดแฉวของบิตออกเป็นกลุ่มๆ ละ 8 บิต

2.4.4 มาตรฐานการส่งข้อมูลแบบอนุกรม

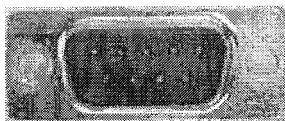
การสื่อสารแบบอนุกรม นับว่ามีความสำคัญ ต่อการใช้งาน ไมโครคอนโทรลเลอร์มาก เพราะสามารถใช้เป็นพิน์และจอภาพ เป็นอินพุตและเอาต์พุต ในการติดต่อหรือควบคุม ไมโครคอนโทรลเลอร์ ด้วยสัญญาณอย่างน้อย เพียง 3 เส้น คือ

- สายส่งสัญญาณ TX
- สายรับสัญญาณ RX
- สายกราวด์ (GND)

2.4.4.1 พอร์ตอนุกรม อาร์เอส232(RS232) (Serial Port)

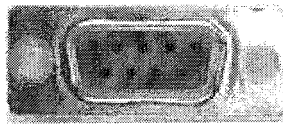
พอร์ตที่ได้รับความนิยมในการสื่อสารแบบอนุกรม คือ พอร์ตอนุกรม อาร์เอส 232 ซึ่งแบ่งได้ 2 แบบ ตามลักษณะการใช้งาน ดังนี้

- พอร์ตอนุกรมของคอมพิวเตอร์ จะเป็นคอนเนคเตอร์แบบ ดีบี 9 ตัวผู้ (Male) ดังรูปที่ 2.15
- พอร์ตอนุกรมของอุปกรณ์ภายนอก จะเป็นคอนเนคเตอร์แบบ ดีบี 9 ตัวเมีย (Female) ดังรูปที่ 2.16

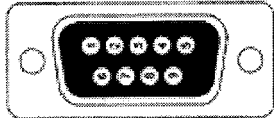


รูปที่ 2.18 พอร์ตอนุกรม ดีบี 9 (DB9) ตัวผู้ (Male)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่นุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.19 พอร์ตอนุกรม ดิปี 9 ตัวเมีย (Female)

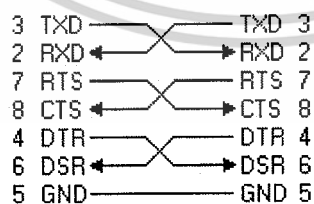


รูปที่ 2.20 ขาของพอร์ตอนุกรม ดิปี 9

ตารางที่ 2.7 โครงสร้างขาของพอร์ตอนุกรม ดิปี 9

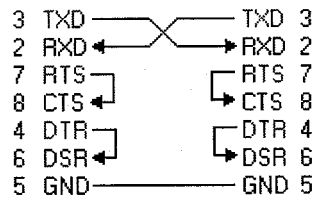
ขา	คำอธิบาย	ชนิด
1	Data Carrier Detect (DCD)	Input
2	Received Data (RXD)	Input
3	Transmitted Data (TXD)	Output
4	Data Terminal Ready (DTR)	Output
5	Signal Ground (GND)	Input
6	Data SetReady (DSR)	Input
7	Request To Send (RTS)	Output
8	Clear to Send (CTS)	Input
9	Ring Indicator (RI)	Input

2.4.4.2 การเชื่อมต่ออุปกรณ์ภายนอกเข้ากับคอมพิวเตอร์ด้วยสาย ดิปี 9



รูปที่ 2.21 การเชื่อมต่ออุปกรณ์ภายนอกผ่าน ดิปี 9 แบบ เนา โมเต็ม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



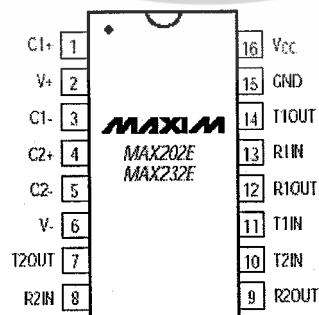
รูปที่ 2.22 การต่ออุปกรณ์ภายนอกผ่าน ดีบี 9 แบบ 3 เส้น

2.4.4.3 การทำงานของขาสัญญาณ ดีบี 9

- TXD เป็นขาที่ใช้ส่งข้อมูล
- RXD เป็นขาที่ใช้รับข้อมูล
- ดีทีอาร์ แสดงสถานะพอร์ตว่าเปิดใช้งาน, ดีทีอาร์ ตรวจสอบว่าพอร์ต ที่ติดต่อกับ เปิดอยู่หรือไม่
- เมื่อเปิดพอร์ตอนุกรม ขา ดีทีอาร์ จะ เปิด เพื่อให้อุปกรณ์ได้รับทราบว่าการติดต่อกับ
- ในขณะที่เดียวกันก็จะตรวจสอบขา ดีเอสอาร์ ว่าอุปกรณ์พร้อมหรือไม่
- อาทีเอส แสดงสถานะพอร์ตว่าต้องการส่งข้อมูล, ซีทีเอส ตรวจสอบว่าพอร์ตที่ติดต่อกับ ต้องการส่งข้อมูลหรือไม่
- เมื่อต้องการส่งข้อมูลขา อาทีเอส จะ เปิด และจะส่งข้อมูลออกที่ขา TXD เมื่อส่งเสร็จก็จะ ปิด
- ในขณะที่เดียวกันก็จะตรวจสอบขา ซีทีเอส ว่าอุปกรณ์ต้องการที่จะส่งข้อมูลหรือไม่
- กราวนด์ ขา ground

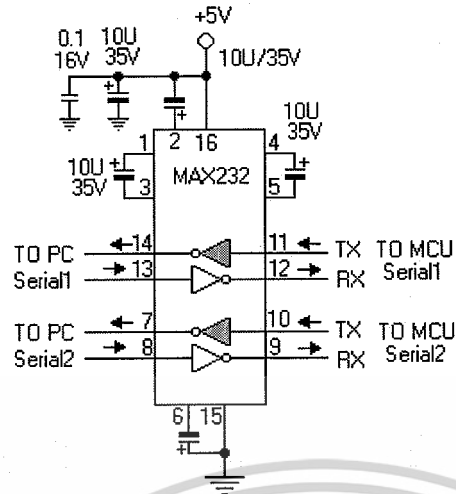
2.4.4.4 ไอซี แม็ค232, ไอซีแอล232

แม็ค 232, ไอซีแอล 232 เป็น ไอซีที่แปลงระดับสัญญาณของ อาร์เอส-232 มาเป็นระดับ ทีทีแอล (TTL) และในทำนองเดียวกันก็แปลงระดับสัญญาณ ทีทีแอล ไปเป็นระดับสัญญาณ อาร์เอส-232



รูปที่ 2.23 ตำแหน่งขาของไอซี แม็ค232

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.24 การใช้งานไอซี แม็ค232

จากรูปที่ 2.24 เป็นการต่อใช้งาน แม็ค-232 ติดต่อระหว่างคอมพิวเตอร์กับ ไมโครคอนโทรลเลอร์ ดังนี้

- ขา 7 และ 8 ต่อเข้าพอร์ตอนุกรมของคอมพิวเตอร์ ส่วนขา 9 และ 10 ต่อเข้าพอร์ตอนุกรมของไมโครคอนโทรลเลอร์ เป็นชุดติดต่อกับ 1 พอร์ต
- ขา 13 และ 14 ต่อเข้าพอร์ตอนุกรมของคอมพิวเตอร์ ส่วนขา 11 และ 12 ต่อเข้าพอร์ตอนุกรมของไมโครคอนโทรลเลอร์ เป็นชุดติดต่อกับอีก 1 พอร์ต
- สรุปได้ว่า ไอซี แม็ค-232 1 ตัว สามารถต่อพอร์ตอนุกรมได้ 2 พอร์ต

2.5 ระบบฐานข้อมูล (Database System)

ฐานข้อมูลถือได้ว่าเป็นแอปพลิเคชัน (Application) หรือ โปรแกรมตัวหนึ่งซึ่งทำหน้าที่ในการเก็บข้อมูลต่าง ๆ ด้วยวิธีการและรูปแบบที่เหมาะสม เพื่อให้ผู้ใช้งานสามารถเก็บข้อมูล, ดูแลรักษาข้อมูล และนำข้อมูลมาใช้งานได้ง่ายกว่าการเก็บข้อมูลในรูปแบบไฟล์

ในความเป็นจริงคำว่า “ระบบฐานข้อมูล” มีความหมายแตกต่างกับคำว่า “ฐานข้อมูล” โดยระบบฐานข้อมูล (Database System) จะประกอบไปด้วย 4 ส่วนหลักคือ ฐานข้อมูล (Database), ซอฟต์แวร์จัดการระบบฐานข้อมูล (DBMS), โปรแกรมใช้งานฐานข้อมูล (Application Program) และผู้ใช้งาน (User)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.5.1 การออกแบบฐานข้อมูลด้วยอี - อาร์โมเดล (Entity Relationship Model)

ในการออกแบบฐานข้อมูลจำเป็นต้องทำการศึกษาลักษณะสมบัติ และความสัมพันธ์ระหว่างข้อมูลที่มีอยู่ในระบบเพื่อให้ได้มาซึ่งโครงสร้างพื้นฐานของฐานข้อมูลโดยทั่วไป มักดำเนินการโดยใช้แบบจำลองข้อมูล

อี - อาร์โมเดลเป็นแบบจำลองข้อมูลที่ได้รับความนิยมมากในการใช้เป็นเครื่องมือสำหรับงานออกแบบฐานข้อมูลด้วยอี อาร์ โมเดล จะเสนอ โครงสร้างของฐานข้อมูลในระดับแนวคิดออกมาในรูปแบบของแผนภาพที่มีโครงสร้างง่ายต่อการทำความเข้าใจ ทำให้เห็นภาพรวมของเอนทิตีทั้งหมดและความสัมพันธ์ระหว่างเอนทิตีในระบบฐานข้อมูล

2.5.2 ขั้นตอนในการออกแบบฐานข้อมูลด้วยอี – อาร์โมเดล ประกอบด้วยขั้นตอนต่างๆ

ดังนี้คือ

2.5.2.1 การศึกษารายละเอียดและลักษณะหน้าที่งานของระบบ

การศึกษารายละเอียดและลักษณะหน้าที่งานของระบบเป็นการศึกษาและรวบรวมเอารายละเอียดที่เกี่ยวกับลักษณะหน้าที่งานของระบบข้อมูลที่เกี่ยวข้องกับขั้นตอนในการทำงาน ตลอดจนข้อกำหนดและสมมติฐานต่างๆ ซึ่งทำได้ด้วยการสัมภาษณ์ หรือ ศึกษาจากแบบฟอร์มต่างๆ ที่มีการใช้งานอยู่ในระบบงานขณะนั้น

2.5.2.2 การกำหนดเอนทิตีที่ควรมีในระบบฐานข้อมูล

เนื่องจากฐานข้อมูลหนึ่ง ๆ อาจประกอบด้วยเอนทิตีต่าง ๆ ได้จำนวนมาก ดังนั้น ในขั้นตอนนี้จึงเป็นการนำรายละเอียดในข้อกำหนดหน้านี้มาทำการกำหนด เอนทิตี ที่จำเป็นต้องมีในระบบฐานข้อมูล โดยคำนึงถึงการเป็นเอนทิตีประเภทอ่อนแอ ตลอดจน Super type หรือ Subtype ด้วย

2.5.2.3 การกำหนดความสัมพันธ์ระหว่างเอนทิตี

การกำหนดความสัมพันธ์ระหว่างเอนทิตี จะเป็นการกำหนดประเภทของความสัมพันธ์ระหว่างเอนทิตี โดยพิจารณาจากข้อกำหนดและสมมติฐานต่าง ๆ

2.5.2.4 การกำหนดคุณลักษณะของเอนทิตี

การกำหนดคุณลักษณะของเอนทิตีเป็นการกำหนดว่าในแต่ละเอนทิตี ควรจะประกอบด้วย Property ใดบ้าง Property ใดที่มีคุณสมบัติเป็น Key Property หรือ Composite Property หรือ Derived Property

2.5.2.5 การกำหนดคีย์หลัก (Primary key) ของแต่ละเอนทิตี

การกำหนดคีย์หลักของแต่ละเอนทิตีนั้น เป็นการกำหนด Key Property ของแต่ละเอนทิตี เพื่อให้แต่ละสมาชิกในเอนทิตีสามารถมีคุณสมบัติที่เป็นเอกลักษณ์เฉพาะได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.5.2.6 การนำสัญลักษณ์ที่ใช้ใน อี – อาร์โมเดล มาอธิบายความสัมพันธ์ระหว่าง

ข้อมูล

การนำสัญลักษณ์ที่ใช้ใน อี – อาร์โมเดล มาอธิบายความสัมพันธ์ระหว่างข้อมูลเป็นการนำรายละเอียดในชั้นตอนต่าง ๆ มาพิจารณาทบทวนเพื่อเพิ่มหรือ ลด Entity Property และความสัมพันธ์ ต่าง ๆ จากนั้นจึงนำข้อมูลที่ได้จากชั้นตอนทั้งหมดมาเขียนเป็นแบบจำลองเพื่ออธิบายความสัมพันธ์ระหว่างข้อมูลด้วยสัญลักษณ์ต่าง ๆ หรือ อี-อาร์โมเดแกรม ดังนั้น แบบจะลงข้อมูลที่เกิดขึ้นจึงมีความชัดเจน สอดคล้อง ถูกต้องและเหมาะสมกับองค์ประกอบของงานที่กำลังศึกษาทำให้เป็นที่ยอมรับของทุกฝ่ายที่เกี่ยวข้อง

2.5.3 โครงสร้างของภาษาเอสคิวแอล (SQL)

ภาษา เอสคิวแอล ย่อมาจาก Structured Query language หรือภาษาในการสอบถามข้อมูล เป็นภาษาทางด้านฐานข้อมูลที่สามารถสร้างและปฏิบัติการกับฐานข้อมูลแบบสัมพันธ์ (Relational Database) โดยเฉพาะ และเป็นภาษาที่มีลักษณะคล้ายกับภาษาอังกฤษ ภาษา เอสคิวแอล ถูกพัฒนาขึ้นจากแนวคิดของ Relational Calculus หรือ Relational Algebra เป็นหลัก ภาษา เอสคิวแอล เริ่มพัฒนาครั้งแรกโดย Almaden Research Center ของบริษัท ไอบีเอ็ม (IBM) โดยมีชื่อเริ่มแรกว่า “ซีเควล” (Sequel) ต่อมาได้เปลี่ยนชื่อเป็น “เอสคิวแอล” หลังจากนั้นภาษา เอสคิวแอล ได้ถูกพัฒนาโดยผู้ผลิตซอฟต์แวร์ด้านระบบจัดการฐานข้อมูลเชิงสัมพันธ์จนเป็นที่นิยมกันอย่างแพร่หลายในปัจจุบัน โดยผู้ผลิตแต่ละรายก็ได้พยายามที่จะพัฒนาระบบจัดการฐานข้อมูลของตนเองให้มีลักษณะเด่นเฉพาะขึ้นมา ทำให้รูปแบบการใช้คำสั่ง เอสคิวแอล มีรูปแบบที่แตกต่างกันไปบ้าง เช่น PRACLE ACCESS SQL Base ของ Sybase INGRES หรือ เอสคิวแอล เซิร์ฟเวอร์ ของ Microsoft เป็นต้น ดังนั้นในปี ค.ศ. 1986 American National Standards Institute (ANSI) จึงได้กำหนดมาตรฐานของ เอสคิวแอล ขึ้น อย่างไรก็ตามโปรแกรมฐานข้อมูลที่ขายในท้องตลาดได้ขยาย เอสคิวแอล ออกไปจนเกินข้อกำหนดของ เอเอ็นเอสไอ โดยเพิ่มคุณสมบัติอื่น ๆ ที่คิดว่าเป็นประโยชน์เข้าไปอีกแต่โดยหลักทั่วไปแล้วก็ยังปฏิบัติตามมาตรฐานของ เอเอ็นเอสไอ ในการอธิบายคำสั่งต่าง ๆ ของภาษา เอสคิวแอล ในหนังสือเล่มนี้จะอธิบายคำสั่งที่เป็นรูปแบบคำสั่งมาตรฐานของภาษา เอสคิวแอล โดยทั่วไป

2.5.4 ประเภทของคำสั่งของภาษา เอสคิวแอล

ภาษา เอสคิวแอล เป็นภาษาที่ใช้งานได้ตั้งแต่ระดับเครื่องคอมพิวเตอร์ส่วนบุคคล (PC) ไปจนถึงระดับเมนเฟรม ประเภทของคำสั่งในภาษา เอสคิวแอล (The subdivision of SQL) แบ่งออกเป็น 3 ประเภท คือ

2.5.4.1 ภาษาสำหรับการนิยามข้อมูล (Data Definition Language: DDL)

ประกอบด้วยคำสั่งที่ใช้ในการกำหนดโครงสร้างข้อมูลว่ามีคอลัมน์อะไร แต่ละคอลัมน์เก็บข้อมูลประเภทใด รวมถึงการเพิ่มคอลัมน์ การกำหนดวิวหรือตารางเสมือนของผู้ใช้ เป็นต้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.5.4.2 ภาษาสำหรับการจัดการข้อมูล (Data Manipulation Language: DM)

ประกอบด้วยคำสั่งที่ใช้ในการเรียกใช้ข้อมูล การเปลี่ยนแปลงข้อมูล การเพิ่มหรือลบข้อมูล เป็นต้น

2.5.4.3 ภาษาควบคุม (Data Control Language: DCL)

ประกอบด้วยคำสั่งที่ใช้ในการควบคุม การเกิดภาวะพร้อมกัน หรือป้องกันการเกิดเหตุการณ์ที่ผู้ใช้หลายคนเรียกใช้ข้อมูลพร้อมกัน และคำสั่งที่เกี่ยวข้องกับการควบคุมความปลอดภัยของข้อมูลด้วยการกำหนดสิทธิ์ของผู้ใช้ที่แตกต่างกัน เป็นต้น

2.5.5 ชนิดของข้อมูลที่ใช้ในภาษา เอสคิวแอล

ในภาษา เอสคิวแอล การบรรจุข้อมูลลงในคอลัมน์ต่าง ๆ ของตารางจะต้องกำหนดชนิดของข้อมูล (Data type) ให้แต่ละคอลัมน์ ชนิดของข้อมูลนี้จะแสดงชนิดของค่าที่อยู่ในคอลัมน์ ค่าทุกค่าในคอลัมน์ที่กำหนดจะต้องเป็นชนิดเดียวกัน เช่น ในตารางลูกค้าคอลัมน์ที่เป็นรายชื่อลูกค้า จะต้องเป็นหนังสือ ในขณะที่คอลัมน์จำนวนเงินที่ถูกค้าซื้อเป็นตัวเลข

ชนิดของข้อมูลของแต่ละคอลัมน์จะขึ้นกับลักษณะของข้อมูลแต่ละคอลัมน์ ซึ่งแบ่งได้ดังนี้

2.5.5.1 ตัวหนังสือ (Character) ในภาษา เอสคิวแอล จะใช้

2.5.5.1.1 ตัวหนังสือแบบความยาวคงที่ (Fixed-length character)

จะใช้ char (n) หรือ character(n) แทนประเภทของข้อมูลที่เป็นตัวหนังสือใด ๆ ที่มีความยาวของข้อมูลคงที่โดยมีความยาวเป็น n ตัวหนังสือ ประเภทนี้จะมีการจองเนื้อที่ตามความยาวที่คงที่ตามที่กำหนดไว้ ชนิดของข้อมูลประเภทนี้จะเก็บความยาวของข้อมูลได้มากที่สุด 255 ตัวอักษร

2.5.5.1.2 ตัวหนังสือแบบความยาวไม่คงที่ (Variable-length character)

จะใช้ Varchar(n) แทนประเภทของข้อมูลที่เป็นตัวหนังสือใด ๆ ที่มีความยาวของข้อมูลไม่คงที่โดยมีความยาว n ตัวหนังสือ ประเภทนี้จะมีการจองเนื้อที่ตามความยาวของข้อมูล ชนิดของข้อมูลประเภทนี้จะเก็บความยาวของข้อมูลได้มากที่สุด 4000 ตัวอักษร

2.5.5.2 จำนวนตัวเลข

2.5.5.2.1 จำนวนเลขจุดทศนิยม (Decimal)

ในภาษา เอสคิวแอล จะใช้ Dec (m, n) หรือ decimal (m, n) เป็นประเภทข้อมูลที่เป็นจำนวนเลขที่มีจุดทศนิยม โดย m คือจำนวนตัวเลขทั้งหมด (รวมจุดทศนิยม) และ n คือจำนวนตัวเลขหลังจุดทศนิยม

2.5.5.2.2 จำนวนเลขที่ไม่มีจุดทศนิยม

ในภาษา เอสคิวแอล จะใช้ Int หรือ Integer เป็นเลขจำนวนเต็มบวกหรือลบขนาดใหญ่เป็นตัวเลข 10 หลัก และในภาษา เอสคิวแอล จะใช้ Smallint เป็นประเภทข้อมูลที่เป็นเลขจำนวน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เต็มบวกหรือลบขนาดเล็ก เป็นเลข 5 หลัก ที่มีตั้งแต่ -32,768 ถึง +32,767 ตัวเลขจำนวนเต็มประเภทนี้จะมีการจองเนื้อที่น้อยกว่าแบบ Integer

2.5.5.2.3 เลขจำนวนจริง

ในภาษา เอสคิวแอล อาจใช้ number (n) แทนจำนวนเลขที่ไม่มีจุดทศนิยมและจำนวนเลขที่มีจุดทศนิยม

2.5.5.3 ข้อมูลในลักษณะอื่น ๆ

2.5.5.3.1 วันที่และเวลา (Date/Time)

เป็นชนิดวันที่หรือเวลาในภาษา เอสคิวแอล จะใช้ ข้อมูล เป็นข้อมูลวันที่ ซึ่งจะมรหลายรูปแบบให้เลือกใช้ เช่น yyyy-mm-dd (1999-10-31), dd.mm.yyyy (31.10.1999) หรือ dd/mm/yyyy (31/10/1999)

2.5.6 ลักษณะการใช้งานของภาษา เอสคิวแอล

ภาษา เอสคิวแอล เป็นส่วนประกอบหนึ่งของ ดีบีเอ็มเอส มักพบใน ดีบีเอ็มเอส เิงส์ัมพันธ์หลายตัวและเป็นที่ยอมรับในปัจจุบัน ภาษา เอสคิวแอล ง่ายต่อการเรียนรู้ การใช้งานในภาษา เอสคิวแอล แบ่งเป็น 2 ลักษณะ คือ ภาษา เอสคิวแอล ที่โต้ตอบได้ (inter active SQL) และภาษา เอสคิวแอล ที่ฝังในโปรแกรม (embedded SQL)

2.5.6.1 ภาษา เอสคิวแอล ที่โต้ตอบได้

ใช้เพื่อปฏิบัติงานกับฐานข้อมูล โดยตรง เป็นการ ใช้คำสั่งภาษา เอสคิวแอล สั่งงานบนจอภาพ โดยเรียกดูข้อมูลได้โดยตรงในขณะที่ทำงาน เพื่อให้ได้ผลลัพธ์ที่นำไปใช้งานได้ ตัวอย่างเช่น ต้องการเรียกดูข้อมูลในคอลัมน์ SALENAME และ SALECOM จากตาราง SALESTAR ใช้คำสั่งของภาษา ดังนี้

```
SELECT SALENAME, SALECOM
FROM SALESATB;
```

2.5.6.2 ภาษา เอสคิวแอล ที่ฝังในโปรแกรม

เป็นภาษา เอสคิวแอล ที่ประกอบด้วยคำสั่งต่าง ๆ ของภาษา เอสคิวแอล ที่ใส่ไว้ในโปรแกรมที่ส่วนมากแล้วเขียนด้วยภาษาอื่น เช่น โคบอล ปาสคาล ภาษาซี ลักษณะของคำสั่ง เอสคิวแอล จะแตกต่างจากภาษาอื่น ๆ ในแง่ที่ว่า เอสคิวแอล ไม่มีคำสั่งเกี่ยวกับควบคุม (control statement) เหมือนภาษาอื่น เช่น if...then...else for...do หรือ loop หรือ while ทำทำให้มีข้อจำกัดในการเขียนชุดคำสั่ง การใช้งานภาษา เอสคิวแอล ฝังในโปรแกรมอื่นจะทำให้ภาษา เอสคิวแอล มีความสามารถ และมีประสิทธิภาพมากยิ่งขึ้น ผลลัพธ์ของคำสั่งที่เกิดจากภาษา เอสคิวแอล ที่ฝังใน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โปรแกรมจะถูกส่งผ่านไปให้กับตัวแปรหรือพารามิเตอร์ที่ใช้ โดยโปรแกรมที่ภาษา เอสคิวแอล ไปฟังตัวอยู่ เช่น

```
While not end-of-file(input)do
Begin
Reading (id-num, salesperson, loc, comm);
EXEC SQL INSERT INTO SALESTAB
VALUES (:id-num, :salesperson, :loc, :comm );
End;
```

ทั้งภาษา เอสคิวแอล ที่ได้ตอบได้และภาษา เอสคิวแอล ที่ฟังใน โปรแกรมจะมีลักษณะของคำสั่งที่ใช้งานเหมือนกัน จะต่างกันแต่เพียงภาษา เอสคิวแอล ที่ฟังใน โปรแกรมจะมีวิธีการเชื่อมโยงกับภาษาอื่น ๆ

2.5.7 การบันทึกข้อมูล, การปรับปรุงข้อมูล และการลบข้อมูล

ในระบบฐานข้อมูล การบันทึกข้อมูล การปรับปรุงข้อมูลและการลบข้อมูลถือเป็นสิ่งสำคัญ ในภาษา เอสคิวแอล มีภาษาสำหรับการจัดการข้อมูล (Data manipulation Language : DML) ซึ่งเป็นภาษาที่ใช้ในการบันทึกข้อมูล การปรับปรุงข้อมูลการลบข้อมูล ภาษาสำหรับการจัดการข้อมูลเป็นส่วนประกอบหนึ่งในภาษา เอสคิวแอล โดยภาษาสำหรับการจัดการข้อมูลใช้สำหรับการจัดการข้อมูลในตารางของฐานข้อมูล ในการใช้คำสั่งที่เป็นภาษาสำหรับนิยามข้อมูลของภาษา เอสคิวแอล เช่น CREATE TABLE จะทำให้ได้โครงสร้างตารางว่าง ๆ ที่ยังไม่มีข้อมูลใด ๆ เก็บอยู่ คำสั่งในภาษา เอสคิวแอล สำหรับการจัดการข้อมูลจะเป็นคำสั่งที่ช่วยในการจัดการข้อมูลภายใน โครงสร้างตารางที่สร้างขึ้นตัวอย่างของคำสั่งในภาษาสำหรับการจัดการข้อมูล จะเป็นคำสั่งการปรับปรุงข้อมูล ได้แก่ การเพิ่มข้อมูล (INSERT) การปรับปรุง (UPDATE) และ การลบข้อมูล (DELETE) และคำสั่งการเรียกคืนข้อมูลได้แก่คำสั่งการเรียกคืนข้อมูล (SELECT)

คำสั่งที่ใช้ในการปรับปรุงข้อมูลของภาษา เอสคิวแอล คือ การเพิ่มข้อมูล การปรับปรุงข้อมูล และการลบข้อมูล เป็นคำสั่งในภาษาการจัดการข้อมูล เมื่อโครงสร้างหลักของตารางได้ถูกกำหนดขึ้นเรียบร้อยแล้ว ก็จะทำการบันทึกข้อมูลลงในตารางหลักหรืออาจทำการปรับปรุง หรือลบข้อมูลในภายหลัง คำสั่งทั้ง 3 นี้ เมื่อดำเนินการในภาษา เอสคิวแอลจะไม่แสดงผลลัพธ์ออกทางหน้าจอ แต่ผลของคำสั่งจะมีผลต่อข้อมูล ผู้ใช้สามารถดูผลของการใช้คำสั่งในการเพิ่มข้อมูล การปรับปรุงและการลบข้อมูล โดยใช้คำสั่งการเรียกคืนข้อมูล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.5.7.1 คำสั่งการเพิ่มข้อมูล (INSERT)

คำสั่งการเพิ่มข้อมูลในตารางจะใช้คำสั่ง เพิ่มข้อมูล จะมีอยู่ 2 รูปแบบคือ การเพิ่มข้อมูลเข้าไปทีละแถว และการเพิ่มข้อมูลโดยโดยการดึงกลุ่มข้อมูลด้วยคำสั่งค้นหาข้อมูล

คำสั่งการเพิ่มข้อมูลทีละแถวโดยระบุข้อมูลที่เพิ่มข้อมูล เข้าไปโดยตรง รูปแบบของคำสั่งเป็นดังนี้

```
เพิ่มข้อมูล INTO <Table_name>[(column 1, column 2, ...)]
```

```
VALUE (<value1, value2, ...>);
```

เพิ่มข้อมูล INTO เป็นคำสั่งที่ต้องมีทุกครั้งที่ต้องการเพิ่มข้อมูล

Table_name ชื่อตารางที่จะเพิ่มข้อมูล

Column 1, Column 2, คอลัมน์ที่ต้องการเพิ่มข้อมูล

Value1, value2, ค่าข้อมูลของแต่ละคอลัมน์ที่ต้องการเพิ่ม

- คำสั่งการเพิ่มข้อมูลโดยการดึงกลุ่มข้อมูลด้วยคำสั่งค้นหาข้อมูล ในภาษา เอสคิวแอลสามารถใช้คำสั่ง เพิ่มข้อมูล ในการนำค่าหรือหาค่าจากตารางหนึ่งแล้วไปใส่ไว้ในอีกตารางหนึ่งได้ โดยได้ค่านั้นมาจากการสอบถามข้อมูล รูปแบบเป็นดังนี้

```
เพิ่มข้อมูล INTO <Table_name>[(column 1, column 2,...)]
```

```
SELECT statement;
```

เพิ่มข้อมูล INTO เป็นคำสั่งที่ต้องมีทุกครั้งที่ต้องการเพิ่มข้อมูล

Table_name ชื่อตารางที่จะเพิ่มข้อมูล

SELECT statement เป็นคำสั่ง การเรียกค้นข้อมูลที่ต้องการข้อมูลอีกตารางหนึ่ง

2.5.7.2 คำสั่งปรับปรุงแถวข้อมูล (UPDATE)

หลังจากที่ป้อนข้อมูลเข้าไปเก็บไว้ในตารางแล้ว ในกรณีที่ต้องการจะปรับปรุงแก้ไขข้อมูลสามารถทำได้ด้วยภาษา เอสคิวแอล การปรับปรุงแถวข้อมูลเป็นการปรับปรุงหรือแก้ไขค่าคอลัมน์ซึ่งในคำสั่งปรับปรุงข้อมูลอาจมีมากกว่า 1 คอลัมน์ในแถวทุกแถวที่มีเงื่อนไขสอดคล้องกับระบุไว้

หลังคำว่า WHERE

รูปแบบของคำสั่งปรับปรุงแถวข้อมูลดังนี้

```
UPDATE <table name>SET<column 1>[,<column 2,>...]=
```

```
<expression |subquery >
```

```
[WHERE<condition>];
```

UPDATE เป็นคำสั่งที่ต้องมีทุกครั้งที่ต้องการปรับปรุงข้อมูล

Table_name ชื่อตารางที่ต้องการปรับปรุง

SET<column>ชื่อคอลัมน์ที่ต้องการปรับปรุง

Expression ค่าข้อมูลที่ต้องการปรับปรุง

เอกสารนี้เป็นเอกสารทสวจนวสวสำหรับการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

WHERE<condition> เงื่อนไขในการปรับปรุง

2.5.7.3 คำสั่งการลบข้อมูลทั้งหมด (DELETE)

คำสั่งในการลบแถวข้อมูลเป็นคำสั่งที่ใช้ในการลบแถวข้อมูลทุกแถวที่มีเงื่อนไขสอดคล้องกับที่ระบุไว้หลัง WHERE คำสั่งการลบข้อมูลมีรูปแบบทั่วไปดังนี้

```
DELETE FROM <Table_name>
```

```
[WHERE<condition>];
```

DELETE FROM เป็นคำสั่งที่ต้องมีทุกครั้งที่ต้องการลบข้อมูล

Table_name ชื่อตารางที่ต้องการลบข้อมูล

WHERE<condition> เงื่อนไขในการลบข้อมูล

2.5.8 การเรียกค้นข้อมูล (SELECT)

การจัดทำฐานข้อมูลในรูปแบบตารางนั้น เกิดจากการที่ข้อมูลได้ออกแบบมาเพื่อลดความซ้ำซ้อน (Normalization) ดังนั้นข้อมูลที่มีรายละเอียดของข้อมูลมากอาจจะถูกเก็บไว้ในหลาย ๆ ตารางแยกออกมาต่างหาก เช่น ตารางข้อมูลที่เป็นตารางหลัก (master table) และ ตารางข้อมูลที่เป็นตารางเชิงรายการ (transaction table) และตารางข้อมูลที่เป็นตารางอยู่ (address table) เป็นต้น การแยกออกเป็นตารางย่อย ๆ นี้ นอกจากลดความซ้ำซ้อนแล้ว ยังช่วยในการประหยัดเนื้อที่และยังเพิ่มประสิทธิภาพของฐานข้อมูล

2.5.8.1 การเรียกค้นข้อมูลจากตารางหลายตารางในภาษา เอสคิวแอล

เป็นการกำหนดความสัมพันธ์ระหว่างตารางทั้งหลาย โดยที่จะสามารถเอาข้อมูลในตารางก็ตารางก็ได้ให้มาสัมพันธ์กันดังนั้นจึงสามารถเชื่อมต่อข้อมูลที่แตกต่างกันได้โดยการใช้คำสั่ง WHERE คำสั่ง WHERE เป็นคำสั่งในการกำหนดเงื่อนไขในการเรียกดูข้อมูลใช้คู่กับคำสั่ง SELECT และ FROM

```
SELECT * FROM TABLE1, TABLE2
```

2.5.8.2 การเรียกดูข้อมูลแบบซ้อนกัน (subqueries)

เป็นการสร้างคำสั่ง การเรียกค้นข้อมูล ซ้อนกันการเรียกดูข้อมูลแบบซ้อนกัน มีจุดประสงค์ก็เพื่อลดภาระในการเชื่อมตารางที่ต้องการใช้หน่วยความจำเป็นเป็นจำนวนมาก คำสั่งย่อยนี้สามารถสร้างหลังคำสั่ง WHERE มีรูปแบบดังนี้

```
SELECT [*] <column 1, column 2,...>
```

```
FROM<table_name>
```

```
[WHERE<column list = <Select Statement>]
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

SELECT คำสั่งที่ต้องมีทุกครั้งที่ต้องการเรียกค้นข้อมูล
 Column 1, column 2,... คอลัมน์ที่ต้องการเรียกค้น
 FROM การกำหนดค่าให้เรียกดูข้อมูล ได้จากตารางใดบ้าง
 Table_name ชื่อตารางที่ต้องการเรียกค้นข้อมูล
 WHERE<condition> ส่วนของคำสั่งที่บอกเงื่อนไขที่จะใช้ในการค้นหาข้อมูล
 Select Statement ส่วนของคำสั่งที่เรียกค้นข้อมูลตามเงื่อนไข

การทำงานของคำสั่งย่อยที่ใช้ในการระบุเงื่อนไขหรือเรียกข้อมูลจะทำจากคำถามย่อค้ำน
 ในสุดผลที่ได้จะเป็นค่ากลับมาให้กับค่าที่อยู่หน้าเครื่องหมาย (=) เพื่อเรียกค้นข้อมูลตามที่ต้องการ

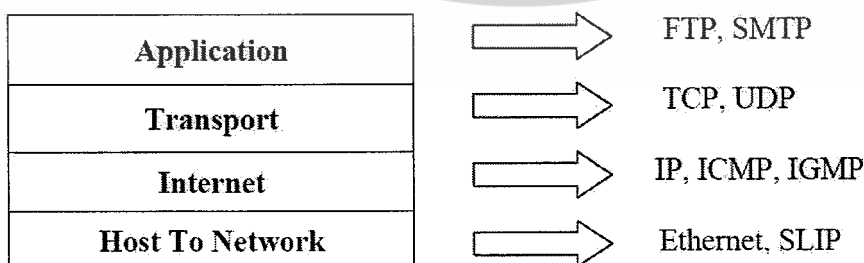
2.6 ทีซีพี/ไอพี โพรโตคอล

ทีซีพี/ไอพี มีจุดประสงค์ของการสื่อสารตามมาตรฐาน 3 ประการคือ

1. เพื่อใช้ติดต่อสื่อสารระหว่างระบบที่มีความแตกต่างกัน
2. ความสามารถในการแก้ไขปัญหาที่เกิดขึ้นในระบบเครือข่าย เช่น ในกรณีที่ผู้ส่งและผู้รับ
 ยังคงมีการติดต่อกันอยู่ แต่โหนดกลางที่ใช้เป็นผู้ช่วยรับ-ส่งเกิดเสียหายใช้การไม่ได้
 หรือสายสื่อสารบางช่วงถูกตัดขาด กฎการสื่อสารนี้จะต้องสามารถจัดหาทางเลือกอื่น
 เพื่อให้การสื่อสารดำเนินต่อไปได้โดยอัตโนมัติ
3. มีความคล่องตัวต่อการสื่อสารข้อมูลได้หลายชนิดทั้งแบบที่ไม่มีความเร่งด่วน เช่น การ
 จัดส่งแฟ้มข้อมูล และแบบที่ต้องการรับประกันความเร่งด่วนของข้อมูล เช่น การ
 สื่อสาร

แบบ Real-time และทั้งการสื่อสารแบบเสียง (Voice) และข้อมูล (data)

2.6.1 โครงสร้างของ ทีซีพี/ไอพี แต่ละเลเยอร์



รูปที่ 2.25 โครงสร้าง ทีซีพี/ไอพี

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
 ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.6.1.1 ชั้นโฮสต์-เครือข่าย (Host-to-Network Layer)

โพรโทคอลสำหรับการควบคุมการสื่อสารในชั้นนี้เป็นสิ่งที่ไม่มีการกำหนดรายละเอียดอย่างเป็นทางการ หน้าที่หลักคือการรับข้อมูลจากชั้นสื่อสาร ไอพี มาแล้วส่งไปยังโหนดที่ระบุไว้ในเส้นทางเดินข้อมูลทางด้านผู้รับก็จะทำงานในทางกลับกัน คือรับข้อมูลจากสายสื่อสารแล้วนำส่งให้กับโปรแกรมในชั้นสื่อสาร

2.6.1.2 ชั้นสื่อสารอินเทอร์เน็ต (The Internet Layer)

ใช้ประเภทของระบบการสื่อสารที่เรียกว่า ระบบเครือข่ายแบบสลับช่องสื่อสารระดับแพ็กเก็ต (Packet-switching network) ซึ่งเป็นการติดต่อแบบไม่ต่อเนื่อง (Connectionless) หลักการทำงานคือการปล่อยให้ข้อมูลขนาดเล็กที่เรียกว่า แพ็กเก็ต (packet) สามารถไหลจากโหนดผู้ส่งไปตามโหนดต่างๆ ในระบบจนถึงจุดหมายปลายทางได้โดยอิสระ หากว่ามีกรส่งแพ็กเก็ตออกมาเป็นชุด โดยมีจุดหมายปลายทางเดียวกัน ในระหว่างการเดินทางในเครือข่าย แพ็กเก็ตแต่ละตัวในชุดนี้ก็จะไปอิสระแก่กันและกัน ดังนั้น แพ็กเก็ตที่ส่งไปถึงปลายทางอาจจะไม่เป็นไปตามลำดับก็ได้

2.6.1.2.1 ไอพี (Internet Protocol)

ไอพี เป็นโพรโทคอลในระดับเน็ตเวิร์กเลเยอร์ ทำหน้าที่จัดการเกี่ยวกับแอดเดรสและข้อมูล และควบคุมการส่งข้อมูลบางอย่างที่ใช้ในการหาเส้นทางของแพ็กเก็ต ซึ่งกลไกในการหาเส้นทางของ ไอพี จะมีความสามารถในการหาเส้นทางที่ดีที่สุด และสามารถเปลี่ยนแปลงเส้นทางได้ในระหว่างการส่งข้อมูล และมีระบบการแยกและประกอบดาต้าแกรม (datagram) เพื่อรองรับการส่งข้อมูลระดับ data link ที่มีขนาด เอ็มทียู (Maximum Transmission Unit) ที่แตกต่างกัน ทำให้สามารถนำ ไอพี ไปใช้บนโพรโทคอลอื่นได้หลากหลาย เช่น อีเทอร์เน็ต (Ethernet), โทเคนริง (Token Ring) หรือ แอปเปิล ทอล์ค (Apple Talk) การเชื่อมต่อของ ไอพี เพื่อทำการส่งข้อมูล จะเป็นแบบ connectionless หรือเกิดเส้นทางการเชื่อมต่อในทุกๆ ครั้งของการส่งข้อมูล 1 ดาต้าแกรม โดยจะไม่ทราบถึงข้อมูลดาต้าแกรมที่ส่งก่อนหน้าหรือส่งตามมา แต่การส่งข้อมูลใน 1 ดาต้าแกรม อาจจะมีการส่งได้หลายครั้งในกรณีที่มีการแบ่งข้อมูลออกเป็นส่วนย่อยๆ (fragmentation) และถูกนำไปรวมเป็นดาต้าแกรมเดิมเมื่อถึงปลายทาง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4-bit Version	Header Length	8-bit Type of Service	16-bit Total Length in Byte	
16-bit Identification			3-bit Flag	16-bit Fragment Checksum
8-bit Time to Live		8-bit Protocol	16-bit Header Checksum	
32-bit Source IP Address				
32-bit Destination IP Address				
option				
Data				

รูปที่ 2.26 ไอพี เฮดเดอร์

เฮดเดอร์ของ ไอพี โดยปกติจะมีขนาด 20 bytes ยกเว้นในกรณีที่มีการเพิ่ม อีพชั่น บางอย่าง 필ด์ของเฮดเดอร์ ไอพี จะมีความหมายดังนี้

- **Version** : หมายเลขเวอร์ชันของ โพร โทคอล ที่ใช้งานในปัจจุบันคือ เวอร์ชัน 4 (ไอพี v4) และเวอร์ชัน 6 (IP v6)
- **Header Length** : ความยาวของเฮดเดอร์ โดยทั่วไปถ้าไม่มีส่วน อีพชั่น จะมีค่าเป็น 5 (5*32 bit)
- **Type of Service (TOS)** : ใช้เป็นข้อมูลสำหรับเราเตอร์ในการตัดสินใจเลือกการเราต์ ข้อมูลในแต่ละคาต้าแกรม แต่ในปัจจุบันไม่ได้มีการนำไปใช้งานแล้ว
- **Length** : ความยาวทั้งหมดเป็นจำนวน ไบต์ของคาต้าแกรม ซึ่งด้วยขนาด 16 บิตของ 필ด์ จะหมายถึงความยาวสูงสุดของคาต้าแกรม คือ 65535 byte (64k) แต่ในการส่ง ข้อมูลจริง ข้อมูลจะถูกแยกเป็นส่วนๆตามขนาดของ เอ็มทียู ที่กำหนดในลิงค์เลเยอร์ และนำมารวมกันอีกครั้งเมื่อส่งถึงปลายทาง แอปพลิเคชันส่วนใหญ่จะมีขนาดของ คาต้าแกรมไม่เกิน 512 byte
- **Identification** : เป็นหมายเลขของคาต้าแกรมในกรณีที่มีการแยกคาต้าแกรมเมื่อ ข้อมูลส่งถึงปลายทางจะนำข้อมูลที่มี identification เดียวกันมารวมกัน
- **Flag** : ใช้ในกรณีที่มีการแยกคาต้าแกรม
- **Fragment offset**: ใช้ในการกำหนดตำแหน่งข้อมูลในคาต้าแกรมที่มีการแยกส่วน เพื่อให้สามารถนำกลับมาเรียงต่อกันได้อย่างถูกต้อง
- **Time to live (TTL)** : กำหนดจำนวนครั้งที่มากที่สุดที่คาต้าแกรมจะถูกส่งระหว่าง hop (การส่งผ่านข้อมูลระหว่างเน็ตเวิร์ค) เพื่อป้องกันไม่ให้เกิดการส่งข้อมูลโดยไม่ สิ้นสุด โดยเมื่อข้อมูลถูกส่งไป 1 hop จะทำการลดค่า ทีทีแอล ลง 1 เมื่อค่าของ ทีทีแอล เป็น 0 และข้อมูลยังไม่ถึงปลายทาง ข้อมูลนั้นจะถูกยกเลิก และเราเตอร์

เอกสารนี้เป็นเอกสารสงวนลิขสิทธิ์ของสถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สุดท้ายจะส่งข้อมูล ไอซีเอ็มพี แจกกลับมายังต้นทางว่าเกิด time out ในระหว่างการส่งข้อมูล

- **Protocol** : ระบุโปรโตคอลที่ส่งในดาต้าแกรม เช่น ทีซีพี ,ยูดีพี หรือ ไอซีเอ็มพี
- **Header checksum** : ใช้ในการตรวจสอบความถูกต้องของข้อมูลในเฮดเดอร์
- **Source IP address** : หมายเลข ไอพี ของผู้ส่งข้อมูล
- **Destination IP address** : หมายเลข ไอพี ของผู้รับข้อมูล
- **Data** : ข้อมูลจากโปรโตคอลระดับบน

2.6.2.1.2 ไอซีเอ็มพี (Internet Control Message Protocol)

ไอซีเอ็มพี เป็นโปรโตคอลที่ใช้ในการตรวจสอบและรายงานสถานภาพของดาต้าแกรม (Datagram) ในกรณีที่เกิดปัญหาที่ดาต้าแกรม เช่น เราเตอร์ไม่สามารถส่งดาต้าแกรมไปถึงปลายทางได้ ไอซีเอ็มพี จะถูกส่งออกไปยังโฮสต์ต้นทางเพื่อรายงานข้อผิดพลาดที่เกิดขึ้น อย่างไรก็ตาม ไม่มีอะไรรับประกันได้ว่า ไอซีเอ็มพีเมสเสจที่ส่งไปจะถึงผู้รับจริงหรือไม่ หากมีการส่งดาต้าแกรมออกไปแล้วไม่มี ไอซีเอ็มพีเมสเสจที่ส่ง Error กลับมา ก็แปลความหมายได้สองกรณีคือ ข้อมูลถูกส่งไปถึงปลายทางอย่างเรียบร้อย หรืออาจจะมีปัญหาในการสื่อสารทั้งการส่งดาต้าแกรมและ ไอซีเอ็มพีเมสเสจที่ส่งกลับมาก็มีปัญหาห้วงทางก็ได้ ไอซีเอ็มพี จึงเป็นโปรโตคอลที่ไม่มีความน่าเชื่อถือ (unreliable) ซึ่งจะเป็นหน้าที่ของ โปรโตคอลในระดับสูงกว่า เน็ตเวิร์ก เลเยอร์ (Network Layer) ในการจัดการให้การสื่อสารนั้นๆ มีความน่าเชื่อถือ ในส่วนของ ไอซีเอ็มพี เมสเสจ จะประกอบด้วย Type ขนาด 8 บิต เช็คซั่ม ขนาด 16 บิต และส่วนของ Content ซึ่งจะมีขนาดแตกต่างกันไปตาม Type และ ใค้ด ดังรูป

8-Bit Type	8-bit Code	16-bit Checksum
ICMP Content		

รูปที่ 2.27 ไอซีเอ็มพี เฮดเดอร์

2.6.1.3 ชั้นสื่อสารนำส่งข้อมูล (Transport Layer)

แบ่งเป็นโปรโตคอล 2 ชนิดตามลักษณะ ลักษณะแรกเรียกว่า Transmission Control Protocol (ทีซีพี) เป็นแบบที่มีการกำหนดช่วงการสื่อสารตลอดระยะเวลาการสื่อสาร (connection-oriented) ซึ่งจะยอมให้มีการส่งข้อมูลเป็นแบบ Byte stream ที่ไว้วางใจได้โดยไม่มีข้อผิดพลาด ข้อมูลที่มีปริมาณ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

มากจะถูกแบ่งออกเป็นส่วนเล็กๆ เรียกว่า ข้อความ ซึ่งจะถูกส่งไปยังผู้รับผ่านทางชั้นสื่อสารของ อินเทอร์เน็ต ทางฝ่ายผู้รับจะนำ ข้อความ มาเรียงต่อกันตามลำดับเป็นข้อมูลตัวเดิม ทีซีพี ยังมี ความสามารถในการควบคุมการไหลของข้อมูลเพื่อป้องกันไม่ให้ผู้ส่ง ส่งข้อมูลเร็วเกินกว่าที่ผู้รับจะ ทำงานได้ทันอีกด้วย โพรโทคอลการนำส่งข้อมูลแบบที่สองเรียกว่า ยูดีพี (User Datagram Protocol) เป็นการติดต่อแบบไม่ต่อเนื่อง (connectionless) มีการตรวจสอบความถูกต้องของข้อมูลแต่จะไม่มี การแจ้งกลับไปยังผู้ส่ง จึงถือได้ว่าไม่มีการตรวจสอบความถูกต้องของข้อมูล ใดๆก็ตาม วิธีการ นี้มีข้อดีในด้านความเร็วในการส่งข้อมูล จึงนิยมใช้ในระบบผู้ให้และผู้ให้บริการ (client/server system) ซึ่งมีการสื่อสารแบบ ถาม/ตอบ (request/reply) นอกจากนั้นยังใช้ในการส่งข้อมูลประเภท ภาพเคลื่อนไหวหรือการส่งเสียง (voice) ทางอินเทอร์เน็ต

2.6.1.3.1 ยูดีพี (User Datagram Protocol)

เป็นโพรโทคอลที่อยู่ใน Transport Layer เมื่อเทียบกับ โมเดล โอเอสไอ (OSI) โดยการ ส่งข้อมูลของ ยูดีพี นั้นจะเป็นการส่งครั้งละ 1 ชุดข้อมูล เรียกว่า ยูดีพี ไคอะแกรม ซึ่งจะไม่มีความสัมพันธ์กันระหว่างดาต้าแกรมและจะไม่มีกลไกการตรวจสอบความสำเร็จในการรับส่งข้อมูล กลไกการตรวจสอบ โดย เช็คซั่ม ของ ยูดีพี นั้นเพื่อเป็นการป้องกันข้อมูลที่จะถูกแก้ไข หรือมีความผิดพลาดระหว่างการส่ง และหากเกิดเหตุการณ์ดังกล่าว ปลายทางจะรู้ว่าว่ามีข้อผิดพลาด เกิดขึ้น แต่มันจะเป็นการตรวจสอบเพียงฝ่ายเดียวเท่านั้น โดยในข้อกำหนดของ ยูดีพี หากพบว่า Checksum Error ก็ให้ผู้รับปลายทางทำการทิ้งข้อมูลนั้น แต่จะไม่มีแจ้งกลับไปยังผู้ส่งแต่อย่างใด การรับส่งข้อมูลแต่ละครั้งหากเกิดข้อผิดพลาดในระดับ ไอพี เช่น ส่งไม่ถึง, หมดเวลา ผู้ส่งจะ ได้รับ Error Message จากระดับ ไอพี เป็น Icmp Error Message แต่เมื่อข้อมูลส่งถึงปลายทางถูกต้อง แต่เกิดข้อผิดพลาดในส่วนของ ยูดีพี เอง จะไม่มีการยืนยัน หรือแจ้งให้ผู้ส่งทราบแต่อย่างใด

16-bit Source Port	16-bit Destination Port
Length	Checksum
Data	

รูปที่ 2.28 ยูดีพี เฮสเตอร์

มีรายละเอียด ดังนี้

- **Source Port Number** : หมายเลขพอร์ตต้นทางที่ส่งดาต้าแกรมนี้
- **Destination Port Number** : หมายเลขพอร์ตปลายทางที่จะเป็นผู้รับดาต้าแกรม
- **UDP Length** : ความยาวของดาต้าแกรม ทั้งส่วน เฮสเตอร์ และ data นั้นหมายความ

เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์ไว้เพื่อใช้ในการศึกษาเท่านั้น เมื่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- **Checksum** : เป็นตัวตรวจสอบความถูกต้องของ ยูดีพี โดอะแกรม และจะนำข้อมูลบางส่วนใน ไอพี เฮดเดอร์ มาคำนวณด้วย

2.6.1.3.2 ทีซีพี (Transmission Control Protocol)

อยู่ใน Transport Layer เช่นเดียวกับ ยูดีพี ทำหน้าที่จัดการและควบคุมการรับส่งข้อมูล ซึ่งมีความสามารถและรายละเอียดมากกว่า ยูดีพี โดยค้ำประกันของ ทีซีพี จะมีความสัมพันธ์ต่อเนื่องกัน และมีกลไกควบคุมการรับส่งข้อมูลให้มีความถูกต้อง (Reliable) และมีการสื่อสารอย่างเป็นกระบวนการ (Connection-oriented)

16-bit Source Port Number				16-bit Source Destination Port				
32-bit Sequence Number								
32-bit Acknowledge Number								
Header Length	16-bit Reserved	URG	ACK	PUSH	RESET	SYN	FIN	16-bit Windows Size
16-bit TCP Checksum				16-bit Urgent Pointer				
TCP Option								
Data								

รูปที่ 2.29 แสดง ทีซีพี เฮดเดอร์

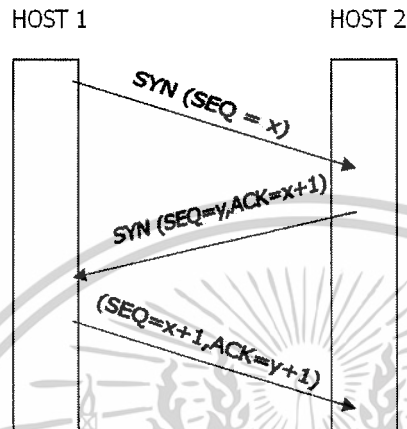
มีรายละเอียด ดังนี้

- **Source Port Number** : หมายเลขพอร์ตต้นทางที่ส่งค้ำประกันนี้
- **Destination Port Number** : หมายเลขพอร์ตปลายทางที่จะเป็นผู้รับค้ำประกัน
- **Sequence Number** : ฟิลด์ที่ระบุหมายเลขลำดับอ้างอิงในการสื่อสารข้อมูลแต่ละครั้ง เพื่อใช้ในการแยกแยะว่าเป็นข้อมูลของชุดใด และนำมาจัดลำดับได้ถูกต้อง
- **Acknowledgment Number** : ทำหน้าที่เช่นเดียวกับ แผนภาพลำดับ แต่จะใช้ในการตอบรับ
- **Header Length** : โดยปกติความยาวของเฮดเดอร์ ทีซีพี จะมีความยาว 20 ไบต์มากกว่านั้น ถ้ามีข้อมูลในฟิลด์ อ็อพชั่น แต่ต้องไม่เกิน 60 ไบต์

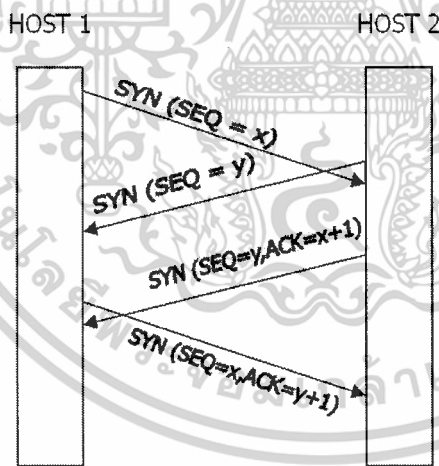
เพื่อ ในเฮดเดอร์ของ ทีซีพี มีความสำคัญในการกำหนดการทำงานของ ทีซีพี เซกเมนต์ เนื่องจากข้อมูลในเฮดเดอร์ของ ทีซีพี จะมีข้อมูลครบถ้วนทั้งการรับและการส่งข้อมูล ซึ่งในการทำงานแต่ละอย่างจะมีการใช้งานฟิลด์ไม่เหมือนกัน เพื่อ จะเป็นตัวกำหนดว่าให้ใช้งานฟิลด์ไหน เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่นิยมนำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เช่น Field Acknowledgment number จะไม่ถูกใช้ในขั้นตอนการเริ่มต้นการเชื่อมต่อ แต่จะมีข้อมูลในฟิลด์ ซึ่งเป็นข้อมูลที่ไม่มีคามหมายใดๆ ซึ่งถ้าไม่มี แพ็ค เป็นตัวกำหนดก็อาจจะมีกรนำข้อมูลมาใช้ และก่อให้เกิดความผิดพลาดได้

2.6.1.3.2.1 การสื่อสารของ ทีซีพี



รูปที่ 2.30 ลำดับขั้นตอนการส่ง ทีซีพี เซกเมนต์



รูปที่ 2.31 การเชื่อมต่อระหว่างซ็อกเก็ตคู่เดียวกัน

เมื่อเซกเมนต์ ติดต่อ(CONNECT) (SYN = "1" และ ACK = "0") เดินทางมาถึงเอ็นทิตี ทีซีพี ที่โฮสต์ปลายทางจะค้นหาโปรเซสตามหมายเลขพอร์ตที่กำหนดในเขตข้อมูล Destination port ซึ่งถ้าหากไม่พบก็จะตอบปฏิเสธด้วยเซกเมนต์ที่มี อาร์เอสที = "1" กลับไปยังผู้ส่ง เซกเมนต์ ติดต่อ ของผู้ส่งจะถูกส่งต่อไปยังโปรเซส ตามพอร์ตที่ระบุซึ่งอาจจะตอบรับหรือตอบเอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่นุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ปฏิเสธก็ได้ ถ้าโพรเซสนั้นต้องการสื่อสารด้วยก็จะส่งเซกเมนต์ตอบรับกลับไป รูปที่ 2.30 แสดงลำดับขั้นตอนการส่ง ทีซีพี เซกเมนต์ในการสร้างการเชื่อมต่อในสถานะปกติระหว่างผู้ส่งและผู้รับ

ในกรณีที่โฮสต์สองแห่งพยายามสร้างการเชื่อมต่อระหว่างซ็อกเก็ตคู่เดียวกันจะเกิดเป็นลำดับขั้นตอนแสดงในรูปที่ 2.31 ผลสุดท้ายจะมีการเชื่อมต่อเกิดขึ้นเพียงหนึ่งช่องทางเท่านั้นเนื่องจากการเชื่อมต่อในแต่ละช่องทางจะถูกกำหนดขึ้นโดยใช้หมายเลขซ็อกเก็ตผู้ส่งและผู้รับ ถ้าการเชื่อมต่อลำดับแรกสำเร็จก็จะถูกบันทึกไว้ในตารางการสื่อสาร เช่น (x, y) ถ้าการเชื่อมต่อลำดับสองสำเร็จในเวลาต่อมา ข้อมูลนี้ก็จะถูกบันทึกไว้ที่เดียวกันคือ (x, y)

ขั้นตอนในการสร้างการเชื่อมต่อและการยกเลิกสามารถเขียนอธิบายด้วยไฟไนท์สเตตแมชชีนที่มีการทำงาน 11 สถานะ ดังแสดงในตารางข้างล่าง ในแต่ละสถานะจะมีเหตุการณ์บางอย่างที่เป็นไปได้ซึ่งจะได้รับการตอบสนองด้วยการกระทำที่เหมาะสม ในทางตรงกันข้ามเหตุการณ์ที่เป็นไปไม่ได้จะกลายเป็นข้อผิดพลาดที่จะต้องรายงานให้ทราบ

การเชื่อมต่อเริ่มต้นจากสถานะ ปิด (CLOSED) เมื่อเรียกใช้บริการ LISTEN หรือติดต่อก็จะมีการเปลี่ยนสถานะไปจากเดิม และถ้าอีกฝ่ายต้องการเชื่อมต่อด้วย การเชื่อมต่อก็จะเกิดขึ้นและย้ายไปอยู่ในสถานะ ESTABLISHED คือการเชื่อมต่อสมบูรณ์ และเมื่อยกเลิกการติดต่อก็จะกลับไปสู่สถานะ ปิด อย่างเดิม

2.6.3.2.1 การเริ่มต้นการสื่อสารของ ทีซีพี โดยใช้การบันทึกเวลาแบบ Three-way handshake

Three-way Handshake เป็นวิธีการส่งแพ็กเก็ตที่สามารถช่วยแก้ปัญหาในเรื่องแพ็กเก็ตซ้ำซ้อนได้ดี แต่วิธีนี้จำเป็นจะต้องสร้างช่องสื่อสารให้ได้ก่อนที่จะเริ่มรับ-ส่งข้อมูล อย่างไรก็ตามแพ็กเก็ตความถี่ที่ใช้ในการต่อรองค่าตัวแปรสำหรับการสื่อสารต่างๆ อาจเกิดการตกค้างอยู่ในระบบได้ ทำให้การกำหนดค่าหมายเลขลำดับมีปัญหาไปด้วย เช่นการสร้างช่องสื่อสารระหว่างโฮสต์1 และ โฮสต์2 เริ่มจาก โฮสต์1 ขอเริ่มการเชื่อมต่อด้วยการส่งแพ็กเก็ต CR (Connection Request) ไปยังโฮสต์2 ซึ่งจะมีค่าตัวแปรต่างๆสำหรับการสื่อสารรวมทั้งหมายเลขลำดับและหมายเลขช่องสื่อสารไปด้วย ผู้รับคือโฮสต์2 ก็จะส่ง ACK (Acknowledge) กลับมายังโฮสต์1 แต่ถ้าแพ็กเก็ตจากผู้ส่งเกิดสูญหายระหว่างทางและสำเนาแพ็กเก็ตที่ยังค้างอยู่ระบบเกิดเดินทางไปถึงผู้รับในภายหลังก็จะทำให้การสร้างช่องสื่อสารใช้การไม่ได้เนื่องจากมีค่าตัวแปรต่างๆไม่ตรงกัน

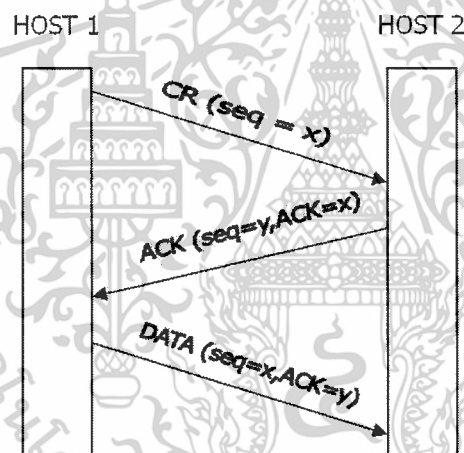
การใช้ ตรีเวย์แฮนด์เชก (Three-way handshake) เป็นการไม่บังคับให้ผู้ส่งและผู้รับข้อมูลจะต้องกำหนดค่าเริ่มต้นของหมายเลขลำดับเป็นเลขเดียวกัน ทำให้สามารถนำวิธีนี้มาใช้ร่วมกับวิธีการจัดจังหวะการทำงานให้พร้อมกัน (Synchronization) แบบต่างๆ ได้ แทนที่จะเป็นการใช้วิธีการบันทึกเวลา ดังรูปที่ 2.32 แสดงขั้นตอนการเริ่มต้นการทำงานจาก โฮสต์ 1 ไปยังโฮสต์ 2 สมมุติให้โฮสต์ 1 เลือกหมายเลขลำดับเป็น “x” และส่งแพ็กเก็ต CONNECTION REQUEST ไปยังโฮสต์ 2 โฮสต์ 2 ตอบรับด้วยแพ็กเก็ต CONNECTION ACCEPTED ซึ่งจะยอมรับหมายเลขลำดับ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษเท่านั้น เมื่อผู้ผู้ดูเห็นหน้าไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

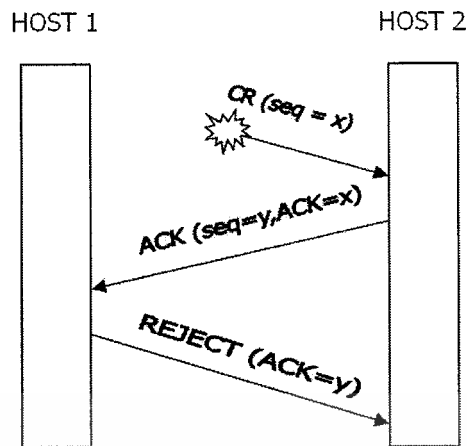
“x” พร้อมกับประกาศหมายเลขลำดับ “y” ที่เป็นของตนเอง จากนั้นโฮสต์ 1 ก็จะตอบรับค่าตัวเลขของโฮสต์ 2 ผ่านทางเซตข้อมูลสำหรับการควบคุมในแพ็กเก็ตข้อมูลแรกที่ส่งมา

สมมติว่าได้เกิดปัญหาการสูญหายของแพ็กเก็ตในขณะทีสำเนาแพ็กเก็ตที่ค้างในระบบเดินทางไปถึงผู้รับแทน รูปที่ 2.33 แสดงเหตุการณ์ที่แพ็กเก็ตที่ผิดพลาด (ตัวแรกในรูป) เป็นสำเนาแพ็กเก็ตเก่าที่เพิ่งจะเดินทางไปถึงโฮสต์ 2 โดยที่โฮสต์ 1 ไม่ทราบ โฮสต์ 2 ก็จะทำงานตามปกติคือจะตอบรับด้วยการส่งแพ็กเก็ต CONNECTION ACCEPTED TPDU กลับมา ที่โฮสต์ 1 ซึ่งโฮสต์ 1 จะสามารถตรวจสอบได้ว่า หมายเลขลำดับ โฮสต์ 2 ตอบกลับมานั้นเป็นหมายเลขลำดับที่ได้เลิกใช้ไปแล้ว จึงมีการส่งแพ็กเก็ต รีเจก (REJECT) กลับมายังโฮสต์ 2 เพื่อบอกยกเลิกการทำงาน จะเห็นว่าวิธีการนี้อาศัยการสื่อสารผ่านแพ็กเก็ต 3 ตัวซึ่งเป็นที่มาของคำว่า “การจับมือร่วมสามชั้นตอน” ผลสุดท้าย ทั้งโฮสต์ 1 และโฮสต์ 2 ก็จะไม่มีการสร้างช่องสื่อสารขึ้นมาจากข้อมูลในสำเนาแพ็กเก็ตเก่าแต่อย่างใด



รูปที่ 2.32 แสดงขั้นตอนการเริ่มต้นการทำงานจากโฮสต์ 1 ไปยังโฮสต์ 2

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.33 เหตุการณ์ที่แพ็กเก็ต ทีพีดียู

2.6.1.4 ชั้นสื่อสารการประยุกต์ (Application Layer)

มีโพรโตคอลสำหรับสร้างจอตอร์มินัลเสมือน เรียกว่า เทลเน็ต โพรโตคอล (TELNET Protocol) สำหรับการจัดการเพิ่มข้อมูล เรียกว่า เอฟทีพี และ โพรโตคอลสำหรับการให้บริการจดหมายอิเล็กทรอนิกส์ เรียกว่า เอสเอ็มทีพี โดยโพรโตคอลสำหรับสร้างจอตอร์มินัลเสมือนช่วยให้ผู้ใช้สามารถติดต่อกับเครื่องโฮสต์ที่อยู่ไกลออกไปโดยผ่านอินเทอร์เน็ต และสามารถทำงานได้เสมือนกับว่ากำลังนั่งทำงานอยู่ที่เครื่องโฮสต์นั้น โพรโตคอลสำหรับการจัดการเพิ่มข้อมูลช่วยในการคัดลอกเพิ่มข้อมูลจากเครื่องอื่นที่อยู่ในระบบเครือข่ายหรือส่งสำเนาเพิ่มข้อมูลไปยังเครื่องใดก็ได้ โพรโตคอลสำหรับให้บริการจดหมายอิเล็กทรอนิกส์ช่วยในการจัดส่งข้อความไปยังผู้ใช้ในระบบ หรือรับข้อความที่มีผู้ส่งเข้ามา

2.7 World Wide Web (WWW)

World Wide Web (WWW) เป็นบริการหนึ่งของอินเทอร์เน็ต ซึ่งมีการพัฒนาขึ้นมาในช่วงปลายปี 1989 โดยทีมงานจาก ห้องปฏิบัติการทางจุลภาคฟิสิกส์แห่งยุโรป (European Particle Physics Labs) หรือที่รู้จักกันในนาม CERN (Conseil European pour la Recherche Nucleaire) ประเทศสวิตเซอร์แลนด์ และได้มีการพัฒนาภาษาที่ใช้สนับสนุน การเผยแพร่เอกสารของนักวิจัยหรือเอกสารเว็บ (Web Document) จากเครื่องแม่ข่าย (Server) ไปยังสถานที่ต่างๆ ในระบบ WWW เรียกว่า ภาษา เซชทีเอ็มแอล (HyperText Markup Language)

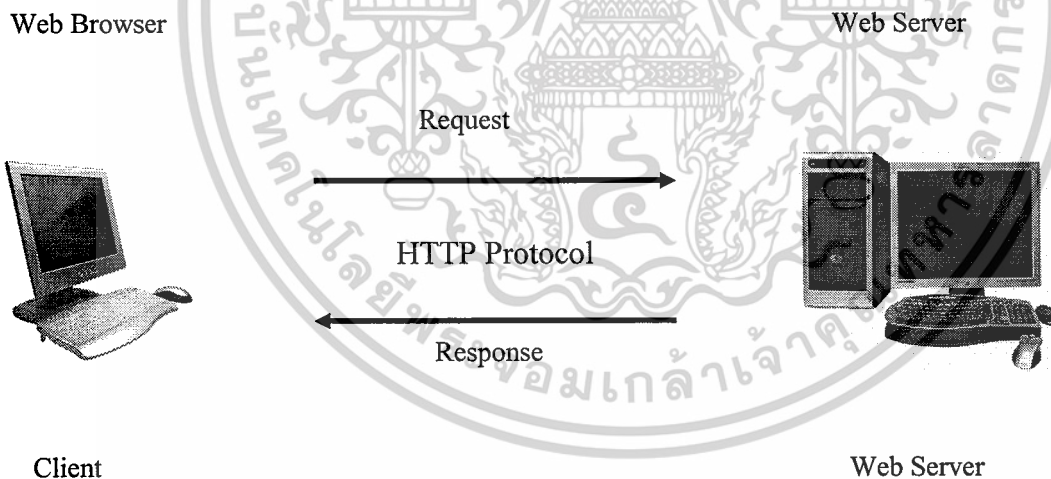
การเผยแพร่ข้อมูลทางอินเทอร์เน็ต ผ่านสื่อประเภทเว็บเพจ (Web Page) เป็นที่นิยมกันอย่างสูงในปัจจุบัน ไม่เฉพาะข้อมูลโฆษณาสินค้า ยังรวมถึงข้อมูลทางการแพทย์ การเรียน งานวิจัยต่างๆ เพราะเข้าถึงกลุ่มผู้สนใจได้ทั่วโลก ตลอดจนข้อมูลที่น่าสนใจนำเสนอออกไป สามารถเผยแพร่ ได้ทั้งข้อมูลตัวอักษร ข้อมูลภาพ ข้อมูลเสียง และภาพเคลื่อนไหว มีลูกเล่นและเทคนิคการนำเสนอที่

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หลากหลาย อันส่งผลให้ระบบ WWW เติบโตเป็นหนึ่งในรูปแบบบริการ ที่ได้รับความนิยมสูงสุดของระบบอินเทอร์เน็ต

ลักษณะเด่นของการนำเสนอข้อมูลเว็บเพจ คือ สามารถเชื่อมโยงข้อมูลไปยังจุดอื่นๆ บนหน้าเว็บได้ ตลอดจนสามารถเชื่อมโยงไปยังเว็บอื่นๆ ในระบบเครือข่าย อันเป็นที่มาของคำว่า HyperText หรือข้อความที่มีความสามารถ มากกว่าข้อความปกติ นั่นเอง จึงมีลักษณะคล้ายกับว่าผู้อ่านเอกสารเว็บ สามารถได้ตอบกับเอกสารนั้นๆ ด้วยตนเอง ตลอดเวลาที่มีการใช้งานนั่นเอง

การทำงานของบริการ WWW นี้จะมีลักษณะเช่นเดียวกับบริการอื่นๆ ของอินเทอร์เน็ต คืออยู่ในรูปแบบไคลเอ็นต์-เซิร์ฟเวอร์ (Client-Server) โดยมีโปรแกรม เว็บไคลเอ็นต์ (Web Client) ทำหน้าที่เป็นผู้ร้องขอบริการ และมีโปรแกรม เว็บเซิร์ฟเวอร์ (เว็บเซิร์ฟเวอร์ หรือบางครั้งถูกเรียกว่า แอสทีทีพี เซิร์ฟเวอร์) ทำหน้าที่เป็นผู้ให้บริการ โปรแกรมเว็บไคลเอ็นต์ก็คือโปรแกรมเว็บเบราว์เซอร์ (Web browser) ในเครื่องของผู้ใช้นั่นเอง เช่น ไมโครซอฟท์ อินเทอร์เน็ต เอ็กพลอเรอร์ (Microsoft Internet Explorer), มอซิลลา ฟายร์ฟ็อกซ์ (Mozilla Firefox), โอเปรา (Opera), ซาฟารี (Safari), etc สำหรับโปรแกรมเว็บเซิร์ฟเวอร์นั้นจะถูกติดตั้งไว้ในเครื่องของผู้ให้บริการเว็บไซต์ การติดต่อระหว่าง โปรแกรมเว็บเบราว์เซอร์กับ โปรแกรมเว็บเซิร์ฟเวอร์จะกระทำผ่าน โพรโทคอล เซชทีทีพี (Hypertext Transfer Protocol) ดังรูป 2.34



รูปที่ 2.34 การติดต่อระหว่าง ไคลเอ็นต์ และเซิร์ฟเวอร์

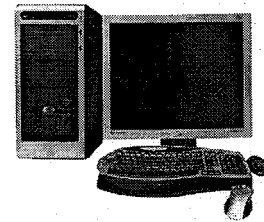
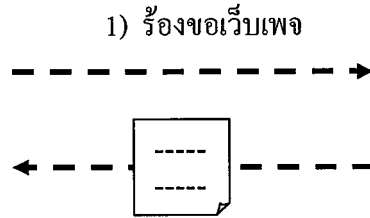
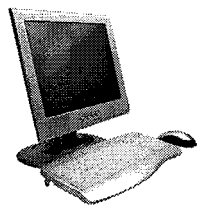
2.7.1 การทำงานของเว็บเพจ

สำหรับเว็บเพจธรรมดาที่โดยปกติแล้วจะมีนามสกุลของไฟล์เป็น เซชทีเอ็ม หรือ เซชทีเอ็มแอล นั้น เมื่อเราใช้เว็บเบราว์เซอร์ เปิดดูเว็บเพจใด เว็บเซิร์ฟเวอร์ก็จะส่งเว็บเพจนั้นกลับมายังเบราว์เซอร์ จากนั้นเบราว์เซอร์จะแสดงผลไปตามคำสั่งภาษา เซชทีเอ็มแอล (Hypertext Markup Language) ที่อยู่ในไฟล์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Web Browser

Web Server



2) ส่งเว็บเพจนั้นกลับไปยัง
เว็บเบราว์เซอร์

Client

Web Server

รูปที่ 2.35 การทำงานของเว็บเพจ

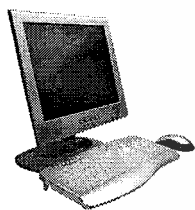
จะเห็นได้ว่าเว็บเพจดังรูป 2.35 เป็นเว็บเพจที่มีลักษณะ สเตติก(static) กล่าวคือผู้ใช้จะพบกับเว็บเพจหน้าตาเดิมๆ ทุกครั้งจนกว่าผู้ดูแลเว็บจะทำการปรับปรุงเว็บเพจนั้น นี่คือข้อจำกัดอันมีสาเหตุมาจากภาษา เฮชทีเอ็มแอล ซึ่งเป็นภาษาที่ใช้อธิบายหน้าตาของเว็บเพจ (HTML จัดเป็นภาษาในกลุ่มที่เรียกว่า page description language) หรือกล่าวอีกนัยหนึ่งคือ เฮชทีเอ็มแอล สามารถกำหนดให้เว็บเพจมีหน้าตาอย่างที่เราต้องการได้ แต่ไม่ช่วยให้เว็บเพจมีความฉลาดได้

การสร้างเว็บเพจที่มีความฉลาดนั้นสามารถทำได้หลายวิธีด้วยกัน หนึ่งในนั้นคือการฝังสคริปต์หรือชุดคำสั่งที่ทำงานทางฝั่งเซิร์ฟเวอร์ (Server-side script) ไว้ในเว็บเพจ

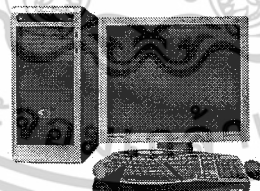
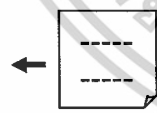
Web Browser

Web Server

2) เรียก PHP Engine
ขึ้นมาประมวลผล
ไฟล์ PHP นั้น



1) ร้องขอไฟล์ PHP



PHP

4) ส่งผลลัพธ์เป็น
HTML กลับไป
เว็บเบราว์เซอร์

3) ติดต่อกับฐานข้อมูล (ถ้ามี)



รูปที่ 2.36 การทำงานของเว็บเพจที่ฝังสคริปต์ภาษา พิเอสพี

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากรูป 2.36 เป็นการทำงานของเว็บเพจที่ฝังสคริปต์ภาษา พีเอสซี ไ่ว์ เมื่อเว็บเบราว์เซอร์ร้องขอไฟล์ พีเอสซี ไฟล์ใด เว็บเซิร์ฟเวอร์จะเรียก พีเอสซี เอ็นจิน(Engine) ขึ้นมาแปล (interpret) และประมวลผลคำสั่งที่อยู่ในไฟล์ พีเอสซี นั้น โดยอาจจะมีการดึงข้อมูล เซชที่เอ็มแอล (และสคริปต์ที่ทำงานทางฝั่งเบราว์เซอร์ เช่น client-side JavaScript) จะถูกส่งกลับไปยังเบราว์เซอร์ เบราวเซอร์ก็จะแสดงผลตามคำสั่ง เซชที่เอ็มแอล ที่ได้รับมา ซึ่งย่อมน่าจะมีคำสั่ง พีเอสซี ใดๆ หลงเหลืออยู่ เนื่องจากถูกแปลและประมวลผลโดย พีเอสซี เอ็นจิน ที่ฝั่งเซิร์ฟเวอร์ไปหมดแล้ว

ให้สังเกตว่าการทำงานของเบราว์เซอร์ในกรณีนี้ไม่แตกต่างจากกรณีของเว็บเพจธรรมดาที่ได้อธิบายไปก่อนหน้านี้เลย เพราะสิ่งที่เบราว์เซอร์ต้องกระทำก็คือการร้องขอไฟล์จากเว็บเซิร์ฟเวอร์ จากนั้นก็รอรับผลลัพธ์กลับมาแล้วแสดงผล ความแตกต่างจริงๆ อยู่ที่การทำงานทางฝั่งเซิร์ฟเวอร์ ซึ่งกรณีหลังนี้ เว็บเพจ (ไฟล์ พีเอสซี) จะผ่านการประมวลผลก่อน แทนที่จะถูกส่งไปยังเบราว์เซอร์เลยทันที

การฝังสคริปต์ พีเอสซี ไ่ว์ในเว็บเพจ ช่วยให้เราสร้างเว็บเพจแบบไดนามิกได้ ซึ่งหมายถึงเว็บเพจที่มีเนื้อหาสาระหรือหน้าตาเปลี่ยนแปลงไปได้ในแต่ละครั้งที่ผู้ใช้เปิดดู โดยขึ้นอยู่กับเงื่อนไขต่างๆ เช่น ข้อมูลที่ผู้ใช้ส่งมา, ข้อมูลในฐานข้อมูล ฯลฯ เป็นต้น

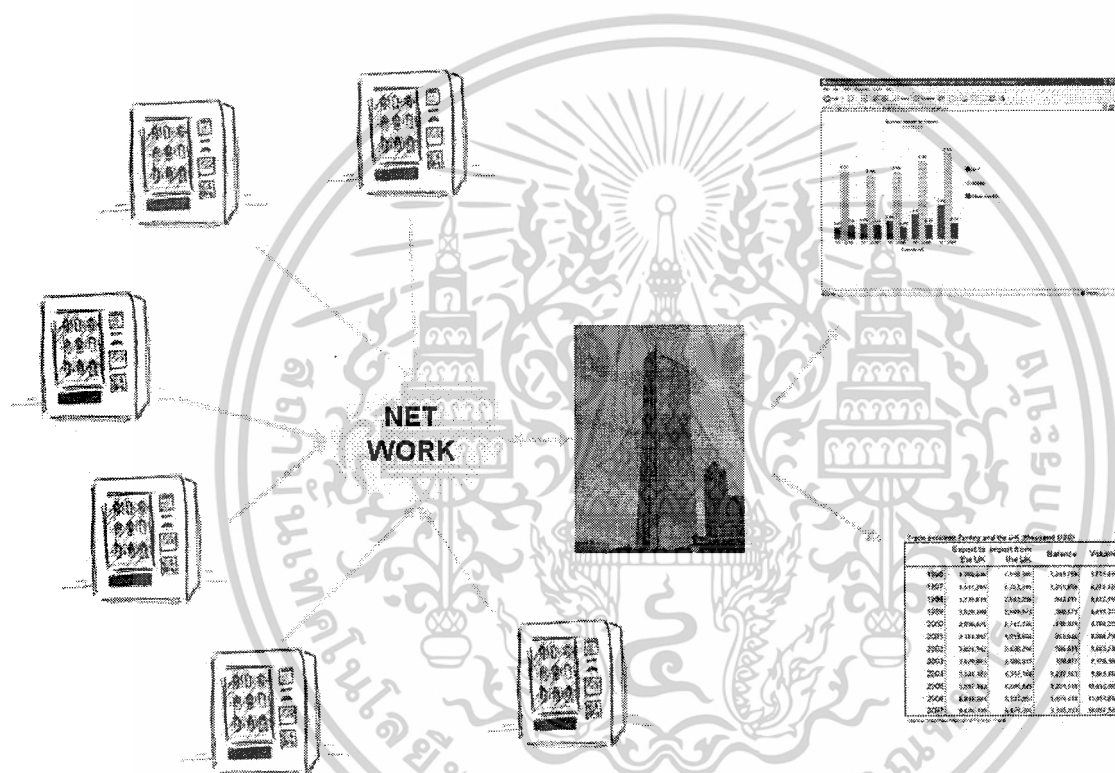


เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 3

การสร้างและการออกแบบระบบ

ในการออกแบบระบบ สามารถมองในรูปแบบโดยรวมของภาพก่อนก็จะมี เครื่องหยุดเหรียญอัตโนมัติซึ่งแต่ละตัวจะต่างสาขาและต่างสถานที่กันออกไป และจะส่งข้อมูลให้กับเซิร์ฟเวอร์ ทุกๆ 1 ชั่วโมง และเซิร์ฟเวอร์จะทำการเก็บข้อมูลไว้และแสดงผลพร้อมออกมาเมื่อมีการเรียกใช้

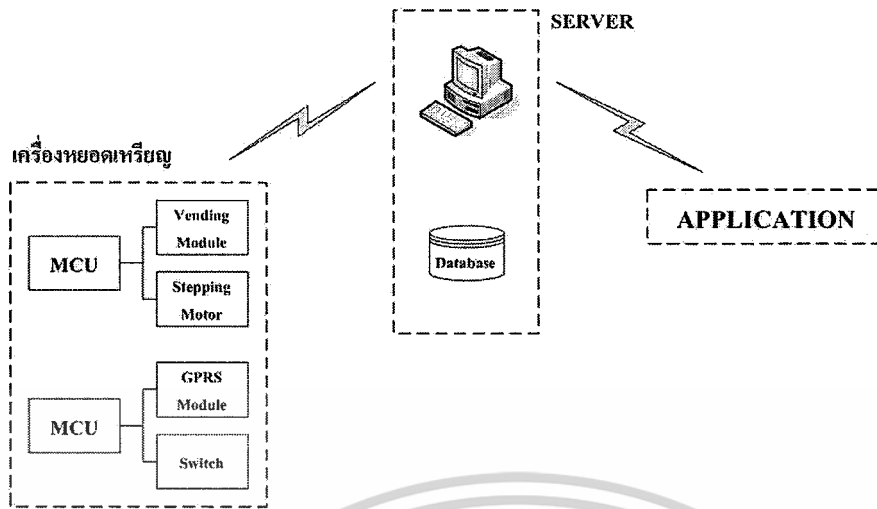


รูปที่ 3.1 แสดงระบบโดยรวม

สำหรับผลลัพธ์ที่ออกมานั้นจะเป็นในรูปแบบของ เว็บเบราว์เซอร์ ซึ่งจะดูว่าสินค้าที่ขายออกไปนั้นขายไปจำนวนเท่าไร ซึ่งสามารถระบุ จำนวนสินค้าในการเติมครั้งต่อไปได้ด้วยเช่นกัน รวมถึงสามารถดูรายละเอียดในการขายสินค้าในแต่ละวันแต่ละเดือน ซึ่งจะแสดงออกมาเป็นในรูปแบบของกราฟ

การออกแบบระบบระบบตรวจสอบสถานะและจำนวนสินค้าของผู้หยุดเหรียญอัตโนมัติสามารถแบ่งการออกได้เป็น 3 ส่วนด้วยกันคือ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

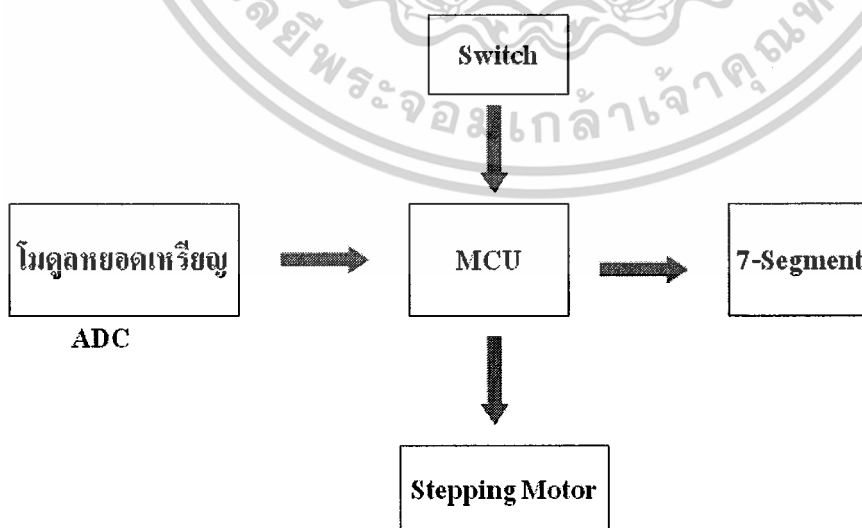


รูปที่ 3.2 การออกแบบแบ่งเป็น 3 ส่วน

- ส่วน เครื่องหยอดเหรียญอัตโนมัติ
- ส่วน จีพีอาร์เอส โมดูล
- ส่วนฐานข้อมูล และ เว็บแอปพลิเคชัน

โดยการแยกแต่ละส่วนนั้นสามารถดูได้จากภาพ โดยที่การทำงานของ เครื่องหยอดเหรียญอัตโนมัตินั้นจะถูกติดตั้งโดย จีพีอาร์เอส โมดูล และจะส่งข้อมูลออกไปให้เซิร์ฟเวอร์เมื่อครบกำหนดเวลาโดยข้อมูลที่ส่งออกไปนั้นจะเป็นเวลา เครื่อง และช่องของสินค้า

3.1 ส่วน เครื่องหยอดเหรียญอัตโนมัติ



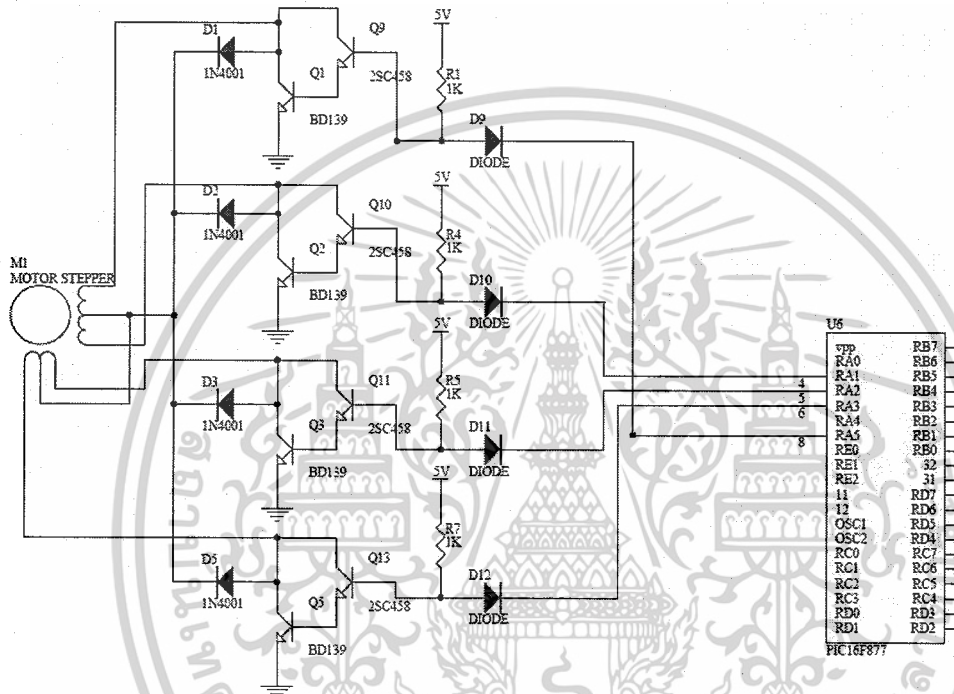
รูปที่ 3.3 การควบคุมส่วนต่างๆของ เครื่องหยอดเหรียญอัตโนมัติ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การออกแบบ วงจรออกเป็นวงจร โดยการเชื่อมต่อกับ โมดูลหอยดหริยญ ซึ่งจะส่ง พัลส์ ออกมาจากตัวโมดูลเป็นจำนวนพัลส์เท่ากับจำนวนหริยญซึ่งจะไปควบคุม สเตปปีงมอเตอร์อีกที โดยค่าเหล่านี้จะไปแสดงผลออกมาที่ 7-เซ็กเมนต์ และเมื่อมีสวิชต์กดเข้ามาก็จะบวกลบกับจำนวนเงินที่หอยดหริยญ ก็จะไปควบคุมสเตปปีงมอเตอร์ให้จ่ายสินค่านั่นเอง

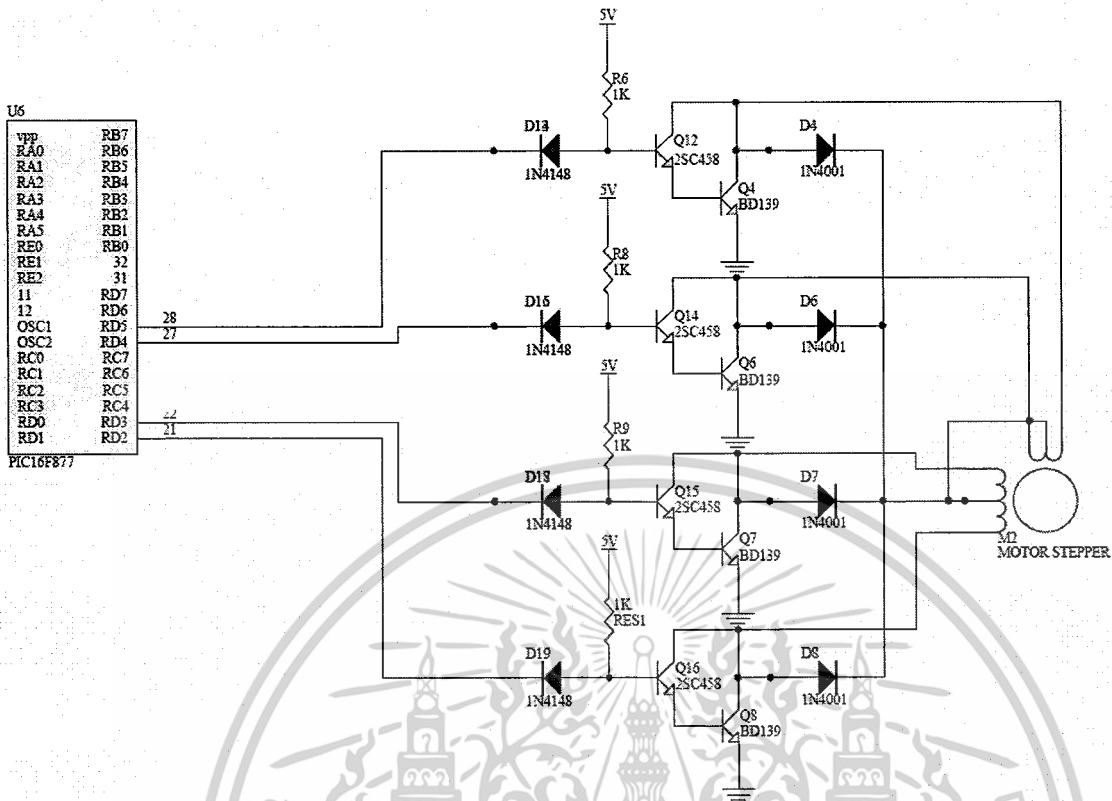
โดยแบ่งการต่อวงจรต่างๆ ได้ดังนี้

3.1.1 ตัวควบคุม สเตปปีงมอเตอร์



รูปที่ 3.4 วงจรขับสเตปปีงมอเตอร์ 1

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



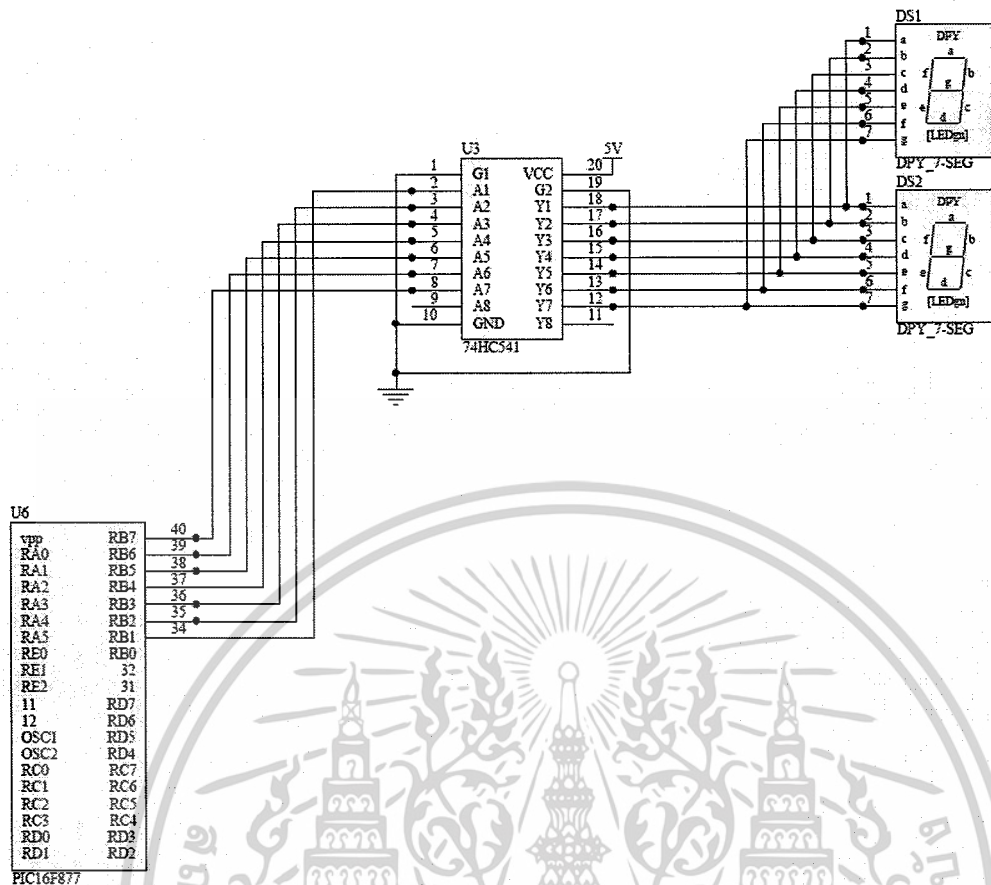
รูปที่ 3.5 วงจรขับสเต็ปมอเตอร์ 2

วงจรขับ Q1-Q4 เป็นตัวขับร่วมกับ Q5-Q8 โดยการต่อแบบคาร์ลิงตัน วงจรนี้สามารถใช้กับมอเตอร์ที่ดึงกระแสมากๆแต่ไม่ควรเกิน 2A เพราะ Q5-Q8 อาจจะร้อนซึ่งวงจรที่ให้นานี้ออกแบบไว้สำหรับรุ่นที่ให้กระแสไม่เกิน 600 mA ถ้าจะใช้ กับรุ่นที่ดึงกระแสมากๆก็เพียงแค่เปลี่ยน Q5-Q8 เป็นเบอร์ที่สามารถทนกระแสได้มากขึ้น D5-D8 เป็นตัวป้องกัน ความเสียหาย ทรานซิสเตอร์อื่นเกิดจากการเปลี่ยนแปลงของสนามแม่เหล็ก แรงดัน +V นั้นควรเลือกให้เหมาะสมกับสเต็ปมอเตอร์ที่จะใช้ด้วยคือ ประมาณ $V+2.5$ โดย V คือค่าแรงดันที่กำกับไว้ที่ข้างตัวมอเตอร์ เช่น 12 V ก็จะใช้แรงดันเท่ากับ $12+2.5 = 14.5$ โวลต์ เป็นต้น

3.1.2 ตัวแสดงผล 7-เซกเมนต์

เป็นวงจรที่สามารถขับ 7-เซกเมนต์ ให้แสดงผลตามต้องการ

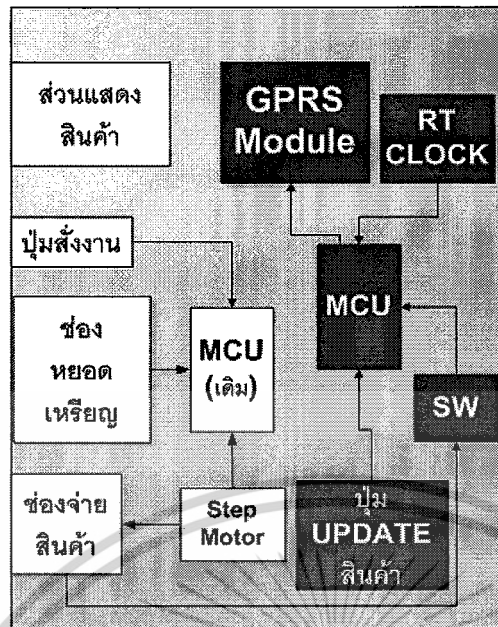
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.6 แสดงการต่อวงจร 7-เซกเมนต์

การออกแบบ เครื่องหยุดเหรียญอัตโนมัติจะแยกกับส่วนที่เป็น จีพีอาร์เอส โมดูล โดยที่ จะติดตั้ง จีพีอาร์เอส โมดูล เข้าไปที่เครื่องเท่านั้น โดยที่ไม่ไปยุ่งเกี่ยวกับระบบ เดิมของตัว เครื่องหยุดเหรียญอัตโนมัติดังกล่าว

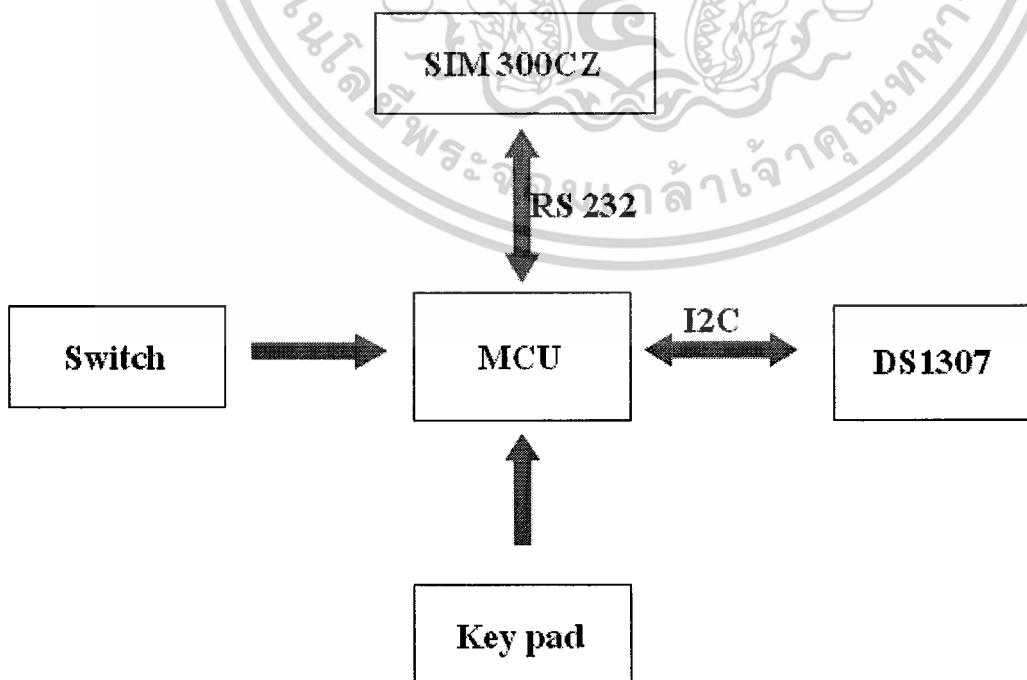
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.7 การติดตั้ง จีพีอาร์เอส โมดูล ร่วมกับ เครื่องหยอดเหรียญอัตโนมัติ

3.1.3 การออกแบบวงจรควบคุม จีพีอาร์เอส โมดูล

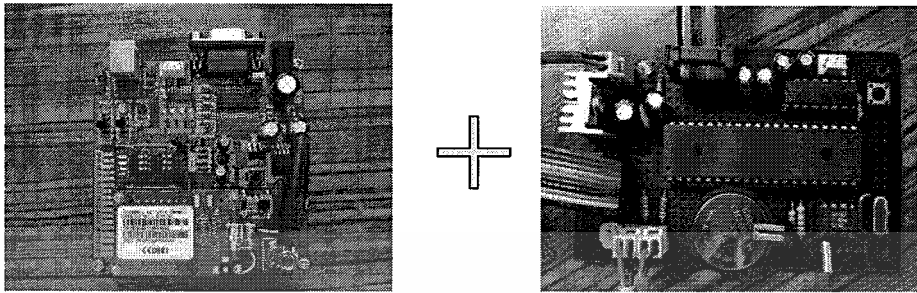
การออกแบบการควบคุม จีพีอาร์เอส โมดูล นั้นทำการเชื่อมต่อกับ เอ็มซียู (MCU) โดยที่มี อินพุทคือ สวิตช์ และ คีย์แพด (Keypad) เป็นอินพุท ในการรับค่าที่จะใส่ข้อมูลลงไปเช่น จำนวน สินค้าที่ทำการเติม, ช่องของสินค้า โดยที่ 300ซีมิลิเมตร นั้นคือการส่งข้อมูลและมีตัวนับเวลาคือ DS1307 โดยช่องทางการสื่อสาร ใช้การสื่อสาร 2 เส้นทางคือ อาร์เอส232, ไอ 2 ซี



รูปที่ 3.8 แสดงการควบคุมส่วนต่างๆด้วยไมโครคอนโทรลเลอร์

เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์โดย บริษัท อีทีเอส จำกัด หากมีผู้ใดนำเนื้อหาไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การใช้ไมโครคอนโทรลเลอร์ในการควบคุมคำสั่ง จีพีอาร์เอส โมดูล สำหรับการใช้งาน จีพีอาร์เอส โมดูล ใช้โมดูล อีที-จีเอสเอ็ม ซิม300ซีแซดใช้ต่อกับ PIC16F877



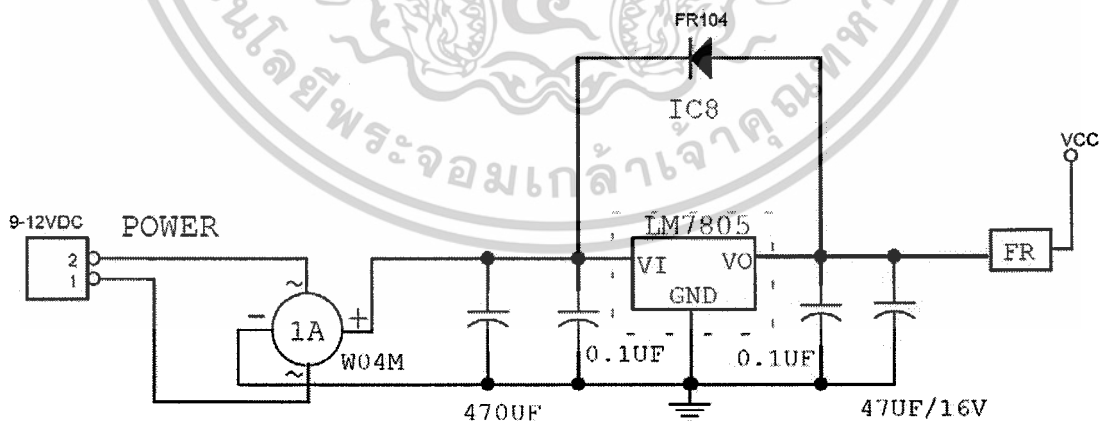
รูปที่ 3.9 การต่อการควบคุม จีพีอาร์เอส โมดูล

ซึ่งสามารถออกแบบวงจรไมโครคอนโทรลเลอร์ในการควบคุม จีพีอาร์เอส โมดูล ได้

- วงจรแหล่งจ่ายไฟ
- วงจรปุ่มกด แมททริกซ์
- วงจร แอลซีดี โมดูล
- วงจร อาร์เอส232
- วงจร ดีเอส1307

3.1.4 วงจรแหล่งจ่ายไฟ

เป็นวงจรแหล่งจ่ายไฟ 5 โวลต์ ให้กับไมโครคอนโทรลเลอร์โดยอินพุตจะเป็นไฟ 9-12 โวลต์

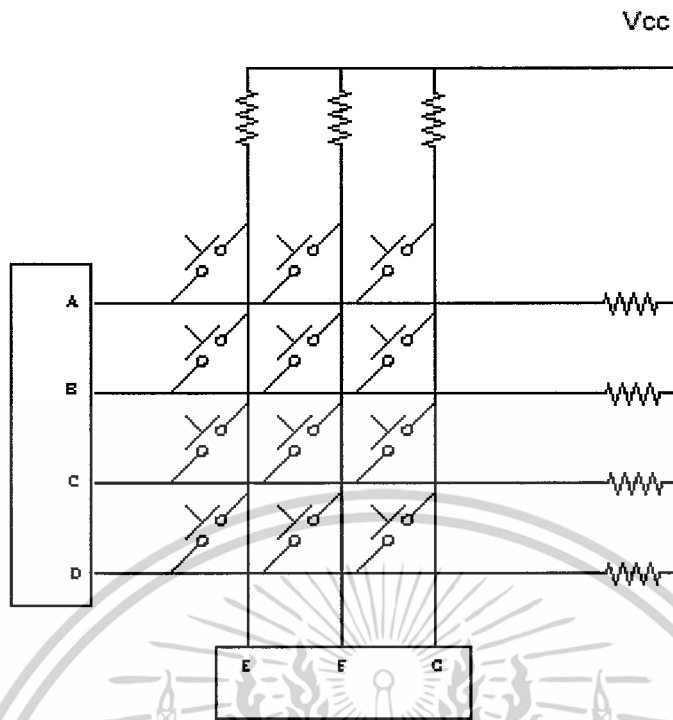


รูปที่ 3.10 วงจรการต่อใช้งานแหล่งจ่ายไฟ 5 โวลต์

3.1.5 วงจรปุ่มกด แมททริกซ์

เป็นวงจรปุ่มกด โดยอินพุตที่ต่อเข้ากับไมโครคอนโทรลเลอร์จะเป็น 7 อินพุต

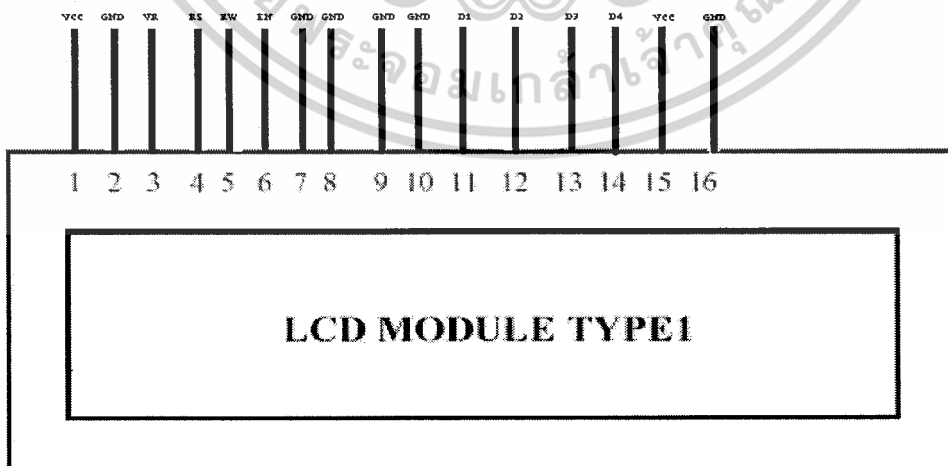
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่นิยมนำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.11 วงจรปุ่มกด เมทริกซ์

3.1.6 วงจร แอลซีดี โมดูล

วงจรแสดงผลออกทางหน้าจอของ แอลซีดี โดยการต่อเข้ากับไมโครคอนโทรลเลอร์ สำหรับอินพุตที่มาจากไมโครคอนโทรลเลอร์จะมี 7 อินพุต และที่เหลือคือ จากภายนอก

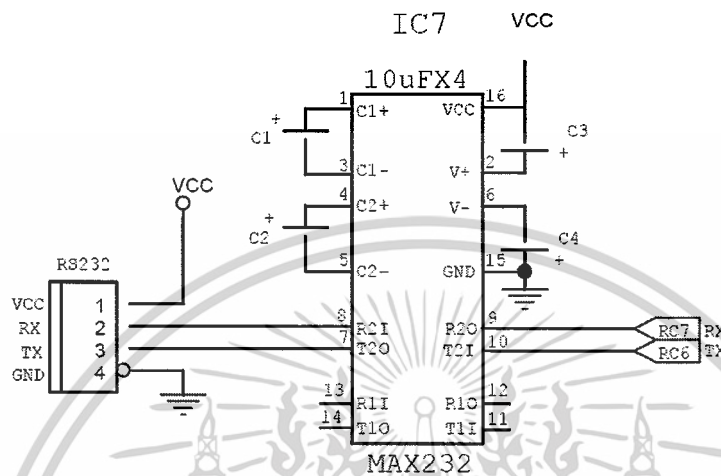


รูปที่ 3.12 การต่อขาเข้ากับ แอลซีดี โมดูล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.1.7 วงจร อาร์เอต 232

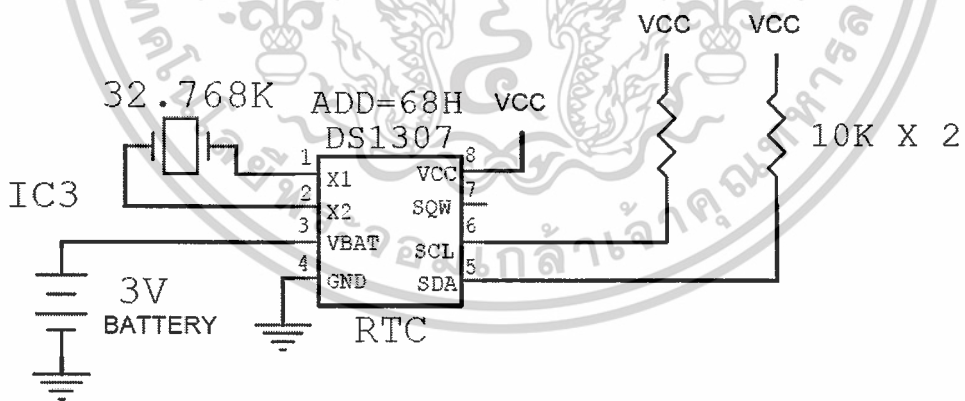
วงจรที่สำหรับสื่อสาร โมดูล จีพีอาร์เอต เข้ากับไมโครคอนโทรลเลอร์โดยการต่อใช้งานจะได้ จำนวน 3 ขา



รูปที่ 3.13 การต่ออาร์เอต 232เข้ากับโมดูล

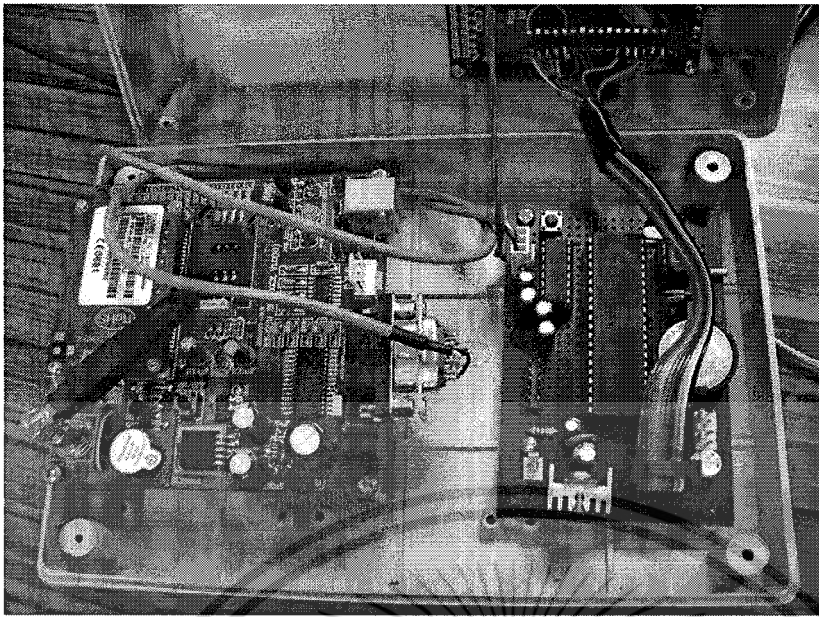
3.1.8 วงจร ดีเอส 1307

เป็นวงจรที่เชื่อมต่อเข้ากับ นาฬิกา โดยการเชื่อมต่อจะเป็นแบบ I²C ซึ่งสามารถต่อได้ตามวงจร



รูปที่ 3.14 วงจร ดีเอส 1307

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.15 การต่อวงจรร่วมกับไมโครคอนโทรลเลอร์

สำหรับการต่อการใช้งานของ จีพีอาร์เอส โมดูล นั้นการเชื่อมต่อไปไมโครคอนโทรลเลอร์นั้นจะเชื่อมต่อผ่านทาง อาร์เอส 232 ซึ่งเมื่อประกอบสำเร็จแล้วนั้นจะได้ดังภาพ



รูปที่ 3.16 การประกอบเสร็จของ จีพีอาร์เอส โมดูล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.2 การสื่อสารระหว่างตัวควบคุม จีพีอาร์เอส โมดูล และ เซิร์ฟเวอร์

การสื่อสารระหว่าง เซิร์ฟเวอร์ กับตัวควบคุมไมโครคอนโทรลเลอร์จะเป็นการสื่อสารแบบ ทีซีพี/ไอพี ที่พอร์ต 2020 โดยมีการรับรูปแบบที่เป็นตัวแปรของรูปแบบจำนวน 2 รูปแบบคือรูปแบบที่ส่งไปเมื่อมีการเติมสินค้า และการส่งข้อมูลเมื่อมีการซื้อสินค้า สามารถเขียนรูปแบบได้ดังนี้

3.2.1 รูปแบบการสื่อสารไป เซิร์ฟเวอร์ เมื่อมีการเติมสินค้า

GET /FILL.PHP?T1=A0&T2=A1&T3=A2&T4=A3&T5=A4

โดยที่ A0 คือ หมายเลข เครื่องหยอดเหรียญอัตโนมัติ

A1 คือ ช่องของสินค้า

A2 คือ จำนวนสินค้าที่ทำการเติม

A3 คือ วันในการเติมซึ่งจะใช้เองโดยไมโครคอนโทรลเลอร์

A4 คือ เวลาการเติมซึ่งจะใช้เองโดยไมโครคอนโทรลเลอร์

ตัวอย่างในการส่ง เครื่องที่หมายเลข 1542 ช่องที่ 1 จำนวนสินค้าที่เติมคือ 44 ชิ้น ณ เวลา 10: 11:21 เวลา 2009-02-13 คือ

GET /FILL.PHP?T1=1542&T2=1&T3=44&T4=2009-02-13&T5=10:11:21

3.2.2 รูปแบบการสื่อสารไป เซิร์ฟเวอร์ เมื่อมีการซื้อสินค้า โดยที่ไมโครคอนโทรลเลอร์จะส่งออกไปเป็นชุด ต่อ หนึ่งกระป๋อง

GET /FILL.PHP?T1=A0&T2=A1&T3=A2&T4=A3

โดยที่ A0 คือ หมายเลข เครื่องหยอดเหรียญอัตโนมัติ

A1 คือ ช่องของสินค้า

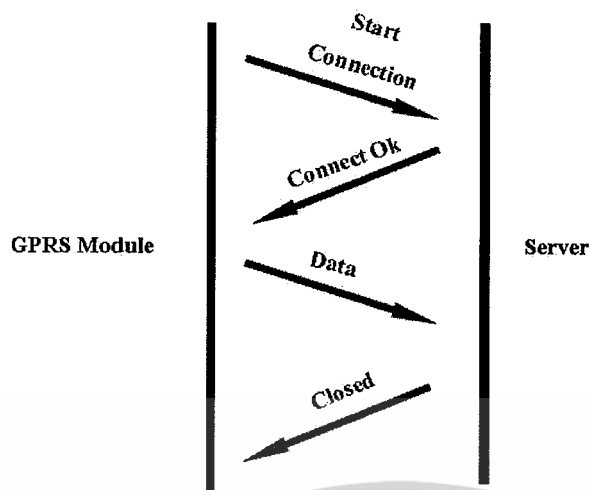
A2 คือ วันในการเติมซึ่งจะใช้เองโดยไมโครคอนโทรลเลอร์

A3 คือ เวลาการเติมซึ่งจะใช้เองโดยไมโครคอนโทรลเลอร์

ตัวอย่างในการส่ง เครื่องที่หมายเลข 1187 ช่องที่ 2 ณ เวลา 14: 07:45 เวลา 2009-07-12 คือ

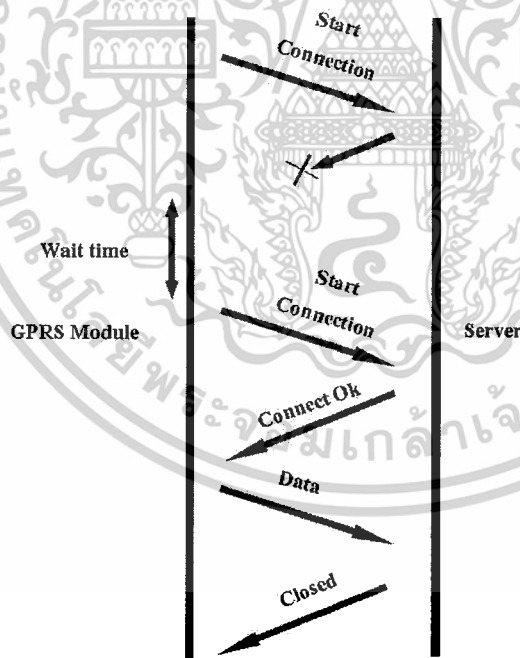
GET /FILL.PHP?T1=1187&T2=2&T3=2009-07-12&T4=14:07:45

การสื่อสารระหว่าง จีพีอาร์เอส โมดูล กับเซิร์ฟเวอร์นั้นสื่อสารได้ 2 กรณีคือ กรณีที่สามารถเชื่อมต่อกันได้และกรณีที่ไม่สามารถส่งได้



รูปที่ 3.17 แผนภาพลำดับในกรณีเชื่อมต่อกับเซิร์ฟเวอร์ได้

สำหรับการสื่อสารในกรณีที่ ไม่สามารถเชื่อมต่อได้ทางฝั่ง จีพีอาร์เอส โมดูล นั้นจะทำการเก็บข้อมูลเพื่อที่จะรอเวลาการส่งข้อมูลครั้งต่อไป โดยสามารถดูได้จากภาพ แผนภาพลำดับ (Sequence Diagram) ข้างล่าง

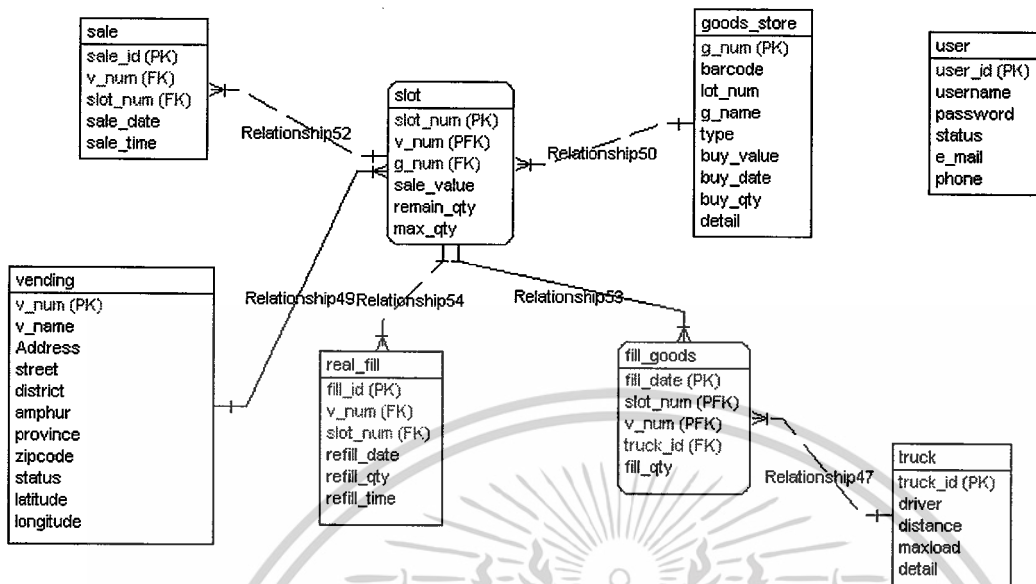


รูปที่ 3.18 แผนภาพลำดับในกรณีเชื่อมต่อกับเซิร์ฟเวอร์ไม่ได้

เมื่อมีการเชื่อมต่อผิดพลาด จีพีอาร์เอส โมดูล จะเก็บข้อมูลครั้งที่ไม่สามารถส่งได้เก็บลงไมโครคอนโทรลเลอร์เพื่อที่จะรอส่งข้อมูลต่อ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.3 ส่วนฐานข้อมูล และ เว็บแอปพลิเคชัน



รูปที่ 3.19 Data-Base schema

ในส่วนของการออกแบบฐานข้อมูล ประกอบด้วย 8 Entity ดังนี้

3.3.1 Entity vending ประกอบด้วย Attribute

- V_num – หมายเลขประจำเครื่อง ใช้เป็น Primary key
- V_name – ชื่อสาขาของเครื่อง ซึ่งจะไม่ซ้ำ (Unique)
- Address – ที่อยู่ของเครื่อง
- Street – ชื่อถนน หรือ ซอย
- District – ตำบล หรือ แขวง
- Amphur – อำเภอ หรือ เขต
- Province – จังหวัด
- Zipcode – รหัสไปรษณีย์
- Status – ระยะทางจากคลังสินค้าไปยังเครื่อง
- Latitude – ละติจูดของเครื่อง
- Longitude – ลองจิจูดของเครื่อง

3.3.2 Entity goods_store ประกอบด้วย Attribute

- Gnum - หมายเลขประจำสินค้า ใช้เป็น Primary key
- Barcode – บาร์โค้ดของสินค้า
- Lot_num – เลขของ Lot ที่ตั้งชื่อ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับใช้เพื่อการศึกษเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- G_name – ชื่อสินค้า
- Type – ชนิดของสินค้า
- Buy_value – ราคารวมที่ซื้อสินค้าต่อครั้ง (Lot)
- Buy_date – วัน / เดือน / ปี ที่ซื้อสินค้า
- Buy_qty – จำนวนสินค้าที่ซื้อต่อครั้ง (Lot)
- Detail – รายละเอียดต่างๆ ของสินค้า

3.3.3 Entity slot ประกอบด้วย Attribute

- Slot_num – เลขช่องในเครื่อง ใช้เป็น primary key ร่วม
- V_num – หมายเลขประจำเครื่อง เป็น foreign key จาก entity vending ใช้เป็น primary key ร่วม
- G_num – หมายเลขประจำสินค้า เป็น foreign key จาก entity goods_store
- Sale_value – ราคาขายสินค้าต่อหน่วย
- Remain_qty – สินค้าคงเหลือในเครื่อง
- Max_qty – สินค้าจำนวนที่บรรจุได้สูงสุดต่อช่อง

3.3.4 Entity sale ประกอบด้วย Attribute

- Sale_id – เลขของสินค้าที่ขายไป ใช้เป็น primary key
- V_num – หมายเลขประจำเครื่อง เป็น foreign key จาก entity vending
- Slot_num – เลขช่องในเครื่อง เป็น foreign key จาก entity slot
- Sale_date – วัน / เดือน / ปี ที่ขายสินค้า
- Sale_time – เวลาที่ขายสินค้า

3.3.5 Entity fill_goods ประกอบด้วย Attribute

- Fill_date – วัน / เดือน / ปี ในการวางแผนการเติมสินค้า ใช้เป็น primary key ร่วม
- Slot_num – เลขช่องในเครื่อง เป็น foreign key จาก entity slot ใช้เป็น primary key ร่วม
- V_num – หมายเลขประจำเครื่อง เป็น foreign key จาก entity vending ใช้เป็น primary key ร่วม
- Truck_id – หมายเลขของรถขนส่งสินค้า เป็น foreign key จาก entity truck
- Fill_qty – จำนวนเติมในการวางแผนเติมสินค้า

3.3.6 Entity real_fill ประกอบด้วย Attribute

- Fill_id – หมายเลขการเติมสินค้า ใช้เป็น primary key

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น เมื่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- V_num – หมายเลขประจำเครื่อง เป็น foreign key จาก entity vending
- Slot_num – เลขช่องในเครื่อง เป็น foreign key จาก entity slot
- Refill_date – วัน / เดือน / ปี ที่มีการเติมสินค้า
- Refill_qty – จำนวนที่มีการเติมสินค้า
- Refill_time – เวลาที่เติมสินค้า

3.3.7 Entity truck ประกอบด้วย Attribute

- Truck_id – หมายเลขของรถขนส่งสินค้า ใช้เป็น primary key
- Driver – ชื่อผู้ขับรถ
- Distance – ระยะทางที่รถสามารถเดินทางได้สูงสุด
- Maxload – จำนวนที่รถสามารถบรรทุกสินค้าได้สูงสุด (ตัน)
- Detail – รายละเอียดต่างๆ ของรถ

3.3.8 Entity user ประกอบด้วย Attribute

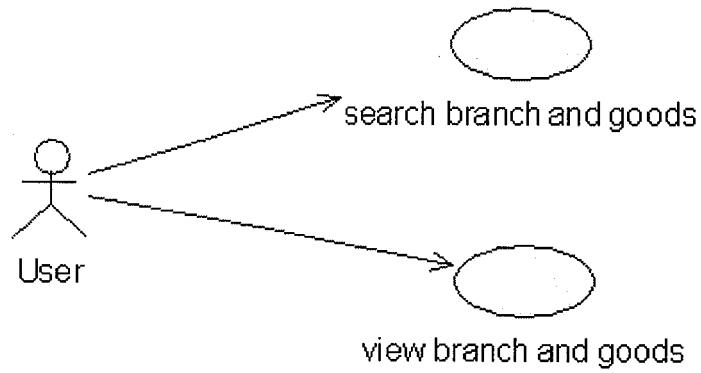
- User_id – หมายเลขของผู้ใช้งาน ใช้เป็น primary key
- Username – ชื่อผู้ใช้งาน ซึ่งจะไม่ซ้ำ (Unique)
- Password – รหัสของผู้ใช้งาน
- Status – สถานะของผู้ใช้งาน สามารถแบ่งได้เป็น 3 ประเภท ดังนี้
 - Status = 1 หมายถึง Administrator
 - Status = 2 หมายถึง Store
 - Status = 3 หมายถึง Manager
- E_mail – e-mail ของผู้ใช้งาน
- Phone – เบอร์โทรศัพท์ของผู้ใช้งาน

3.3.9 การออกแบบ เว็บแอปพลิเคชัน

ในการออกแบบ เว็บแอปพลิเคชัน จะแบ่งเป็น 4 ส่วน แยกตามสถานะของผู้ใช้งาน ซึ่งมี 2 ระดับด้วยกัน คือ

ผู้ใช้งานที่ไม่ต้อง ล็อกอิน ได้แก่

ยูสเซอร์ คือ ผู้ใช้งานทั่วไป หรือ ลูกค้า ซึ่งมีสิทธิต่าง ๆ ดังนี้



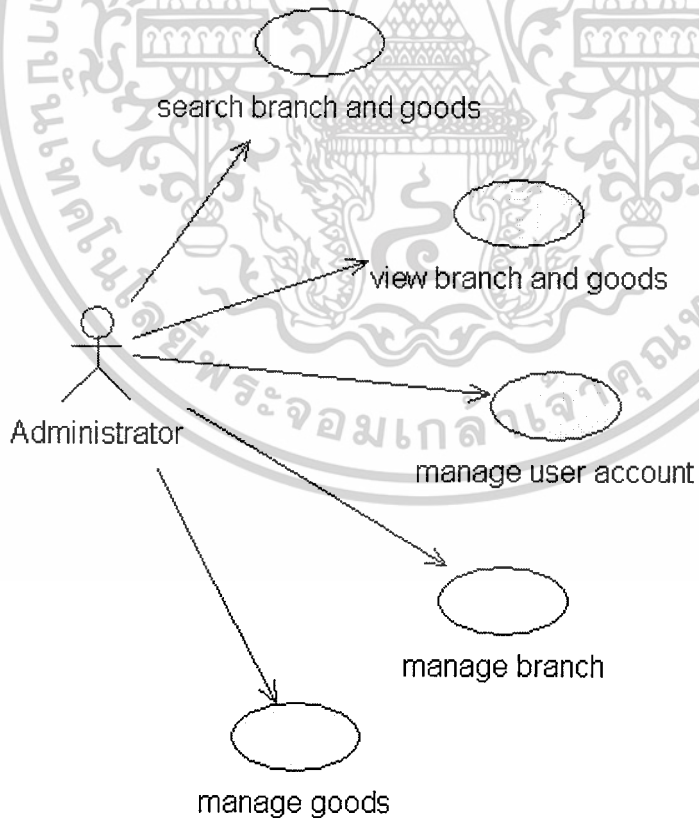
รูปที่ 3.20 usecase ของ user

search branch and goods – สามารถค้นหาสินค้าและสาขาได้

view branch and goods – สามารถดูสินค้าและสาขาได้

3.3.10 ผู้ใช้งานที่ไม่ต้อง ล็อกอิน เข้าสู่ระบบ

Administrator คือ ผู้ดูแลระบบ และจัดการข้อมูลต่างๆ ซึ่งมีสิทธิ์ต่าง ๆ ดังนี้



รูปที่ 3.21 usecase administrator

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

search branch and goods – สามารถค้นหาสินค้าและสาขาได้

view branch and goods – สามารถดูสินค้าและสาขาได้

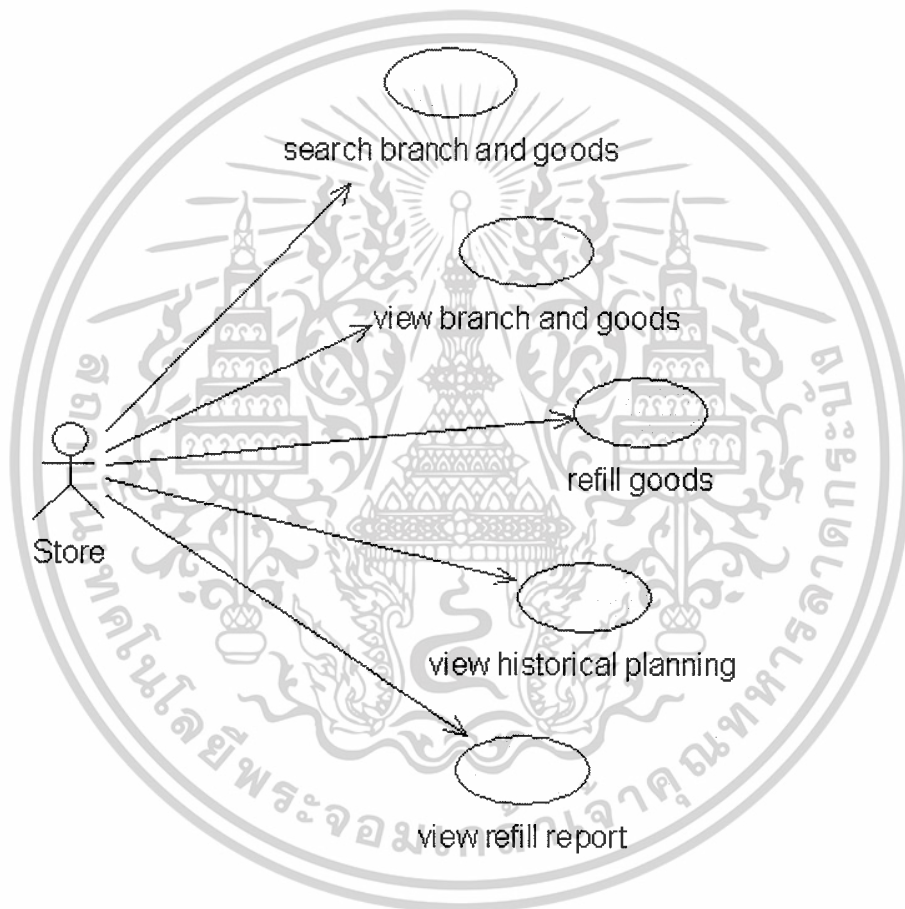
manage user account – จัดการเกี่ยวกับระบบสมาชิก

manage branch – จัดการเกี่ยวกับสาขา

manage goods – จัดการเกี่ยวกับสินค้า

3.3.11 สตอร์ (Store)

คือ ผู้จัดการเกี่ยวกับสินค้า ซึ่งมีสิทธิต่าง ๆ ดังนี้



รูปที่ 3.22 usecase ของ store

search branch and goods – สามารถค้นหาสินค้าและสาขาได้

view branch and goods – สามารถดูสินค้าและสาขาได้

refill goods – วางแผนในการเติมสินค้า

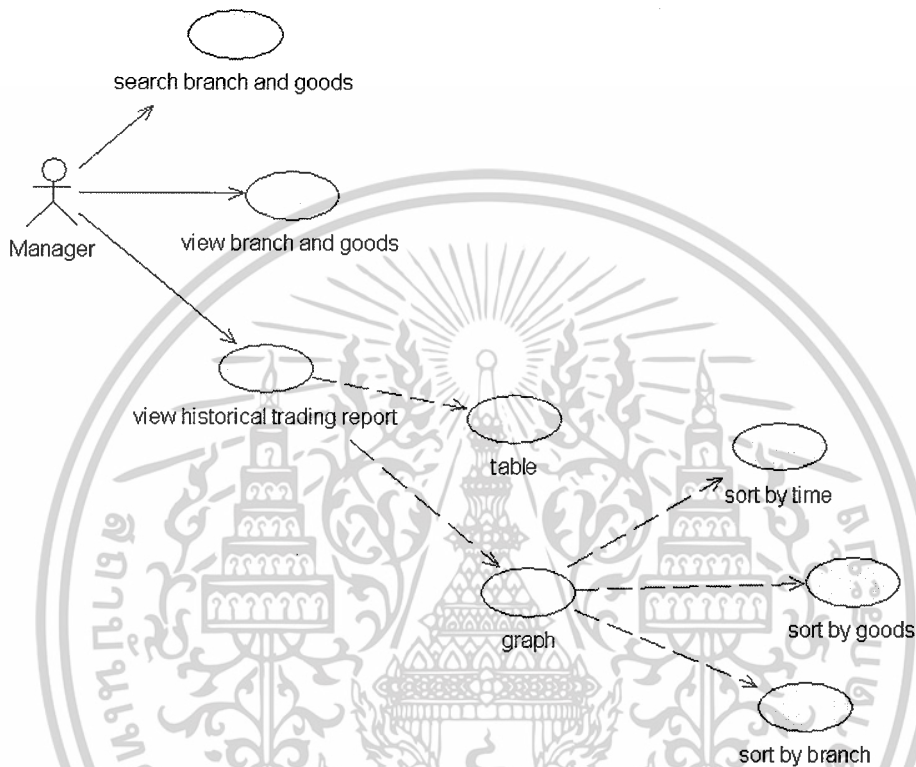
view historical planning - สามารถดูการวางแผนการเติมสินค้าย้อนหลัง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

view refill report – สามารถดูค่าความคลาดเคลื่อนของการวางแผนการเติมสินค้ากับการเติมสินค้าจริงได้

3.3.12 เมเนเจอร์ (Manager)

คือ ผู้จัดการองค์กร ซึ่งมีสิทธิต่าง ๆ ดังนี้



รูปที่ 3.23 usecase ของ manager

search branch and goods – สามารถค้นหาสินค้าและสาขาได้

view branch and goods – สามารถดูสินค้าและสาขาได้

view historical trading report – สามารถดูรายการขายสินค้าย้อนหลังได้ ผ่านทางตาราง แสดงยอดขายย้อนหลัง

กราฟ

- แสดงยอดขายย้อนหลัง จำแนกตามเวลาขาย
- แสดงยอดขายย้อนหลัง จำแนกตามยอดขายของแต่ละสาขา
- แสดงยอดขายย้อนหลัง จำแนกตามจำนวนสินค้าที่ขายได้

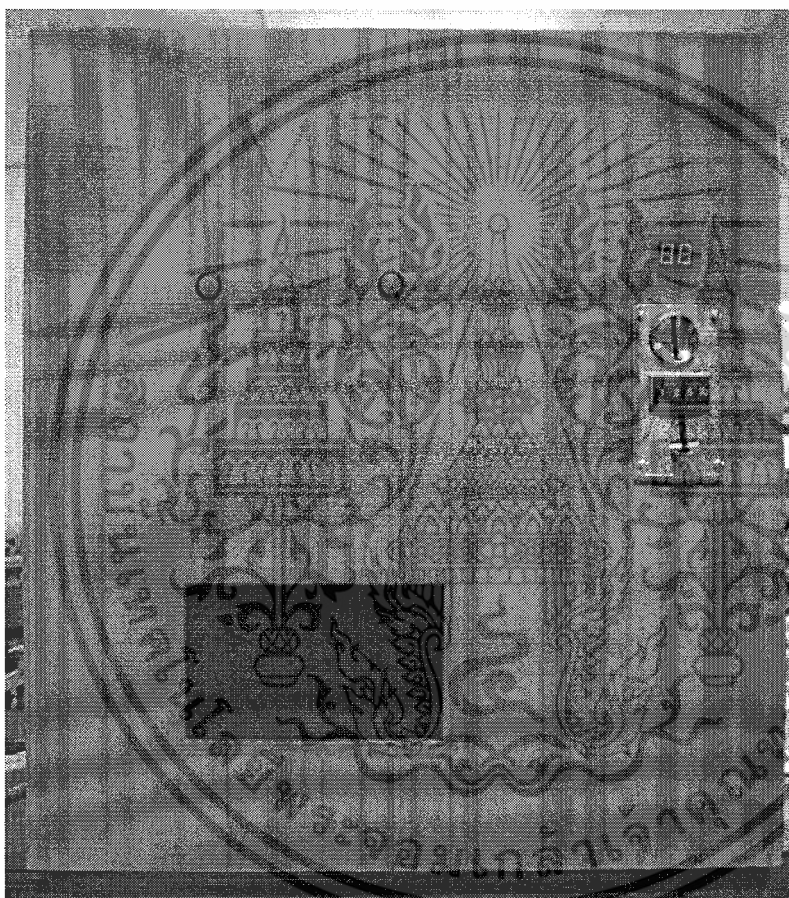
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 4

การทดลองและผลการทดลอง

4.1 การทดลองการใช้งานเครื่องหยอดเหรียญอัตโนมัติ

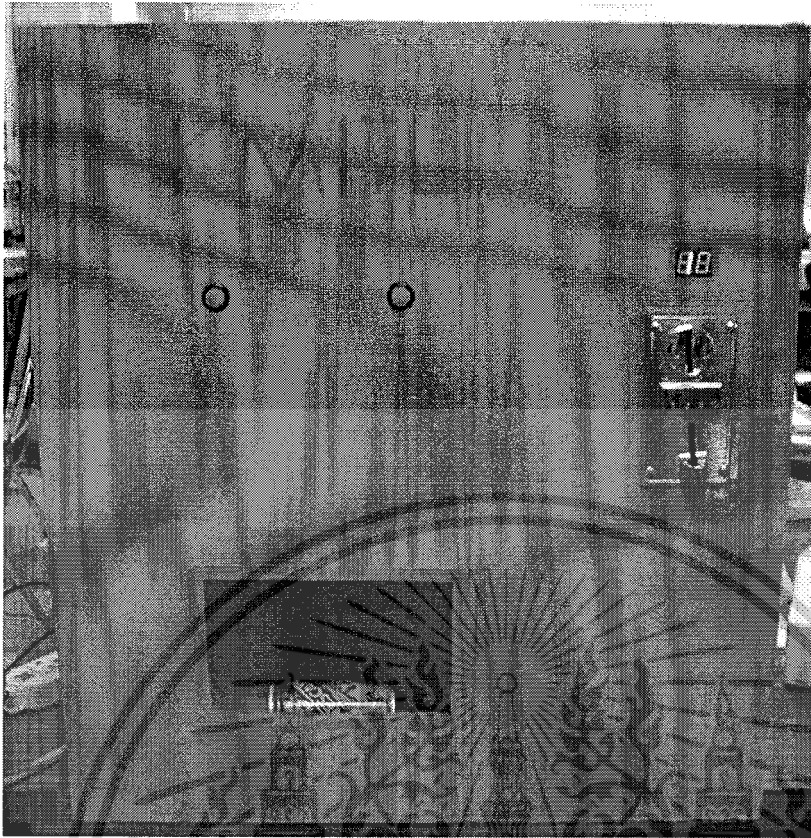
ในการทดลองได้ทำการทดลองโดยการหยอดเหรียญ 1 บาท 5 บาท และ 10 บาท ตามลำดับ โดยที่การทดลองนั้นสามารถทำออกมาได้ตามที่ทำการหยอดเหรียญตามมูลค่าของเหรียญ



รูปที่ 4.1 เครื่องหยอดเหรียญอัตโนมัติ

เมื่อทำการหยอดเหรียญ โดยที่ หยอดเหรียญ 10 จำนวน 2 เหรียญและกดปุ่ม สวิตซ์ในการเลือกสินค้าก็จะได้รับสินค้าตามที่ต้องการไว้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.2 สิ้นค้าเมื่อมีการกดสวิทช์

4.2 การติดตั้งสวิทช์ลงบน เครื่องหยุดเหรียญอัตโนมัติ

ในการติดตั้งสวิทช์บน เครื่องหยุดเหรียญอัตโนมัติ นั้นจะทำการติดตั้งสวิทช์ตรงช่องจ่าย สิ้นค้าเพราะเมื่อมีการจำหน่ายสินค้าออกไปจะทำให้สินค้าที่จ่ายออกมาจะไปสัมผัสกับสวิทช์แล้ว ไมโครคอนโทรลเลอร์ก็จะทำการเก็บข้อมูลของการขายสินค้าลงใน EEPROM และจะทำการส่ง ข้อมูลไปยัง เซิร์ฟเวอร์ ในเวลาถัดไป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.3 การติดตั้งสวิตช์

4.3 ส่งข้อมูลผ่าน จีพีอาร์เอส ด้วย คำสั่ง เอที คอมมาน

คำสั่งที่ใช้ในการส่งข้อมูลจะคำสั่งต่อไปนี้ โดยในการทดลองครั้งนี้ได้ใช้ ซิม ดีเทค ในการส่งข้อมูล ต่อ 1 เรคคอร์ด

หมายเหตุ

- ตัวหนาคือคำสั่งของโมดูล
>
- ขีดเส้นใต้คือข้อมูลที่แสดงตอบกลับมากโมดูล
- ตัวเอียงคือข้อมูลที่ใช้ในการส่ง

at

OK

ati

SIMCOM_Ltd

SIMCOM_SIM300

Revision:1008B08SIM300M32_SPANSION

OK

at+cpin? (ตรวจสอบสถานะของ SIM Card)

+CPIN: READY

OK

at+cops? (ตรวจสอบเครือข่าย)

+COPS: 0,0,"DTAC"

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

OK

at+creg? (ตรวจสอบสถานะในการเข้าร่วมเครือข่าย)

+CREG: 0,1

OK

at+creg=1

OK

at+cgdcont=1,"IP",www.dtac.co.th (ติดต่อกับเครือข่าย)

OK

at+cstt="www.dtac.co.th",",","

OK

at+ciicr (การร้องขอ IP)

OK

at+cifsr (แสดงหมายเลข IP)

10.122.148.114

at+cipstatus (ตรวจสอบว่ามีหมายเลข IP หรือไม่)

OK

STATE: IP STATUS

at+cdnsfcfg="203.155.33.1","202.44.144.33" (ตั้งค่า DNS เซิร์ฟเวอร์)

OK

at+cipstatus

OK

STATE: IP STATUS

at+cdnsorip=0 (ตั้งค่าในการใช้ IP)

OK

at+cipstart="UDP","58.10.84.53","2020" (ทำการติดต่อไปยัง เซิร์ฟเวอร์)

CONNECT OK

OK

at+cipsend (ทำการส่งข้อมูล)

> **GET /TEST.PHP?T1=1&T2=2&T3=2009-01-20&T4=12:23:00** (กด Ctrl+Z เพื่อทำการส่งข้อมูล)

SEND OK

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

at+cipclose (ปิด Connection)

CLOSE OK

at+cipshut (ยกเลิกการติดต่อทั้งหมด)

SHUT OK

at+cipstatus

OK

STATE: IP INITIAL

at+cpowd=1

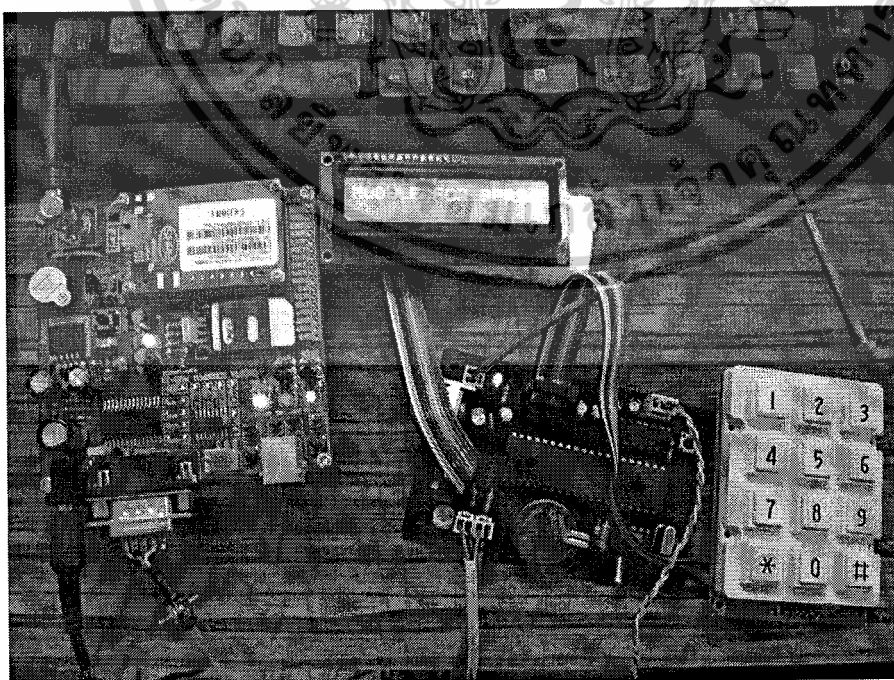
+CREG: 0

NORMAL POWER DOWN

สำหรับข้อมูลที่จะใช้ในการส่งจะเป็น วันและเวลา ที่ใช้ในการเก็บจาก ไมโครคอนโทรลเลอร์ โดยคำสั่งที่จะทำการส่งไปจะเป็น "GET /TEST.PHP?T1=เครื่อง&T2=ช่อง&T3=วัน&T4=เวลา" สำหรับการส่งข้อมูลเป็นชั่วโมง และ "GET /FILL.PHP?T1=เครื่อง&T2=ช่อง&T3=จำนวนที่เหลือ&T4=วัน&T5=เวลา"

4.4 การทดลองเชื่อมต่อไมโครคอนโทรลเลอร์ กับ จีพีอาร์เอส โมดูล

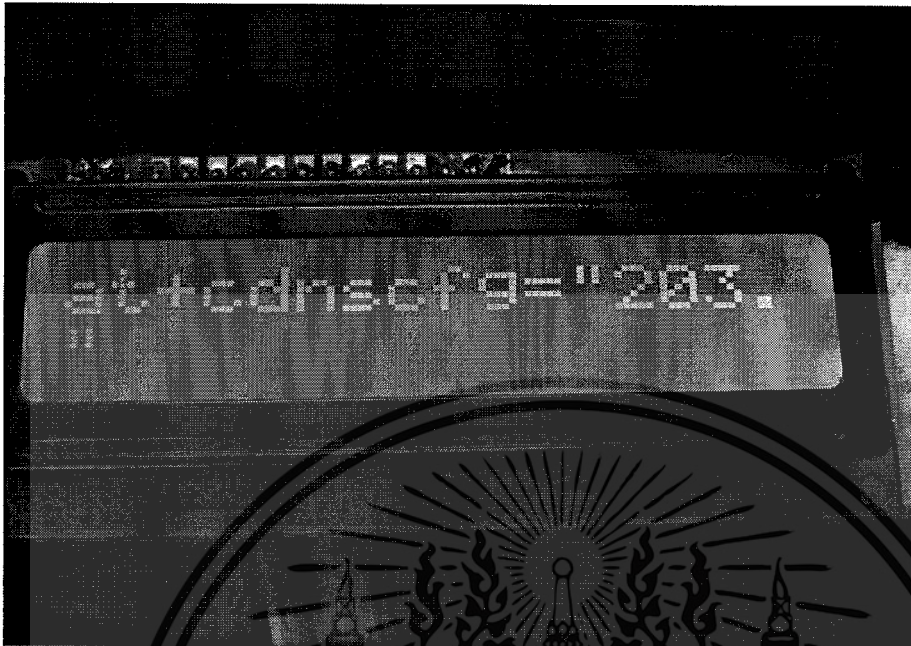
เริ่มจากการเขียนโปรแกรมเพื่อรองรับ จีพีอาร์เอส โมดูล โดยทำการตั้งค่าของเครื่องหยอดเหรียญอัตโนมัติ



รูปที่ 4.4 การเชื่อมต่อ จีพีอาร์เอส โมดูล กับ ไมโครคอนโทรลเลอร์

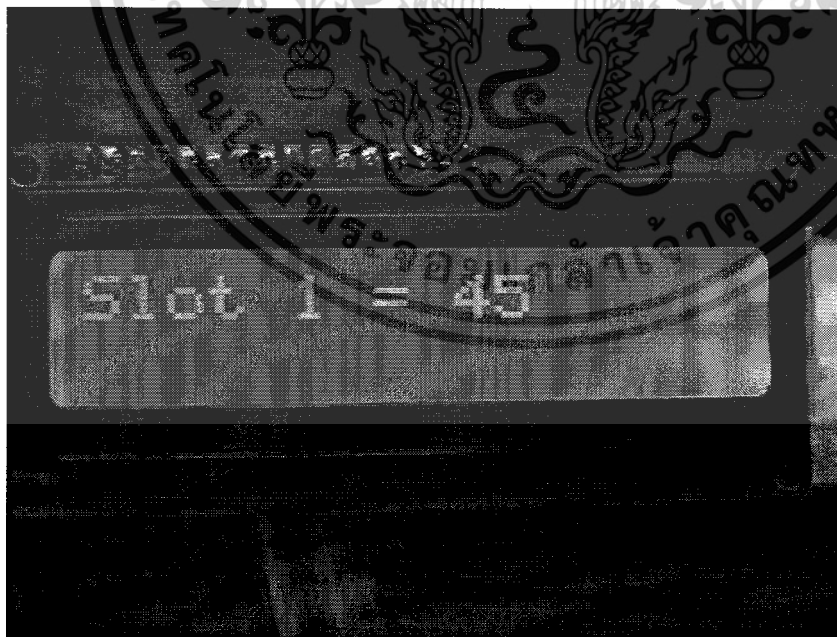
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เมื่อทำการเชื่อมต่อเสร็จแล้วจากนั้นทำการตั้งค่าเริ่มต้น จีพีอาร์เอส โมดูล โดยการกดหมายเลข 3



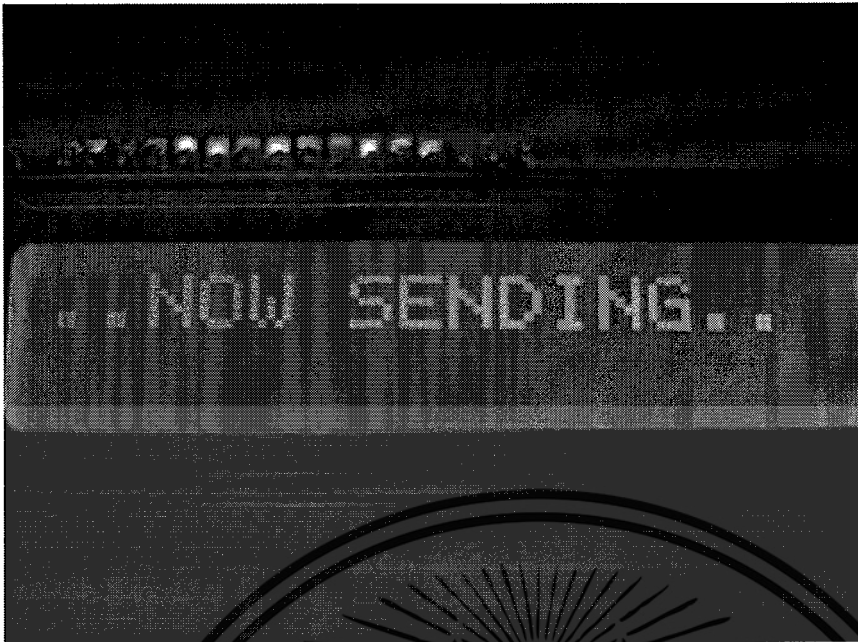
รูปที่ 4.5 การตั้งค่า จีพีอาร์เอส โมดูล ผ่านไมโครคอนโทรลเลอร์

เมื่อทำการตั้งค่า จีพีอาร์เอส โมดูล แล้วจากนั้นทำการตั้งค่า ช่อง ของตู้หยอดเหรียญอัตโนมัติแต่ละตู้ พร้อมทำการส่งข้อมูลออกไป



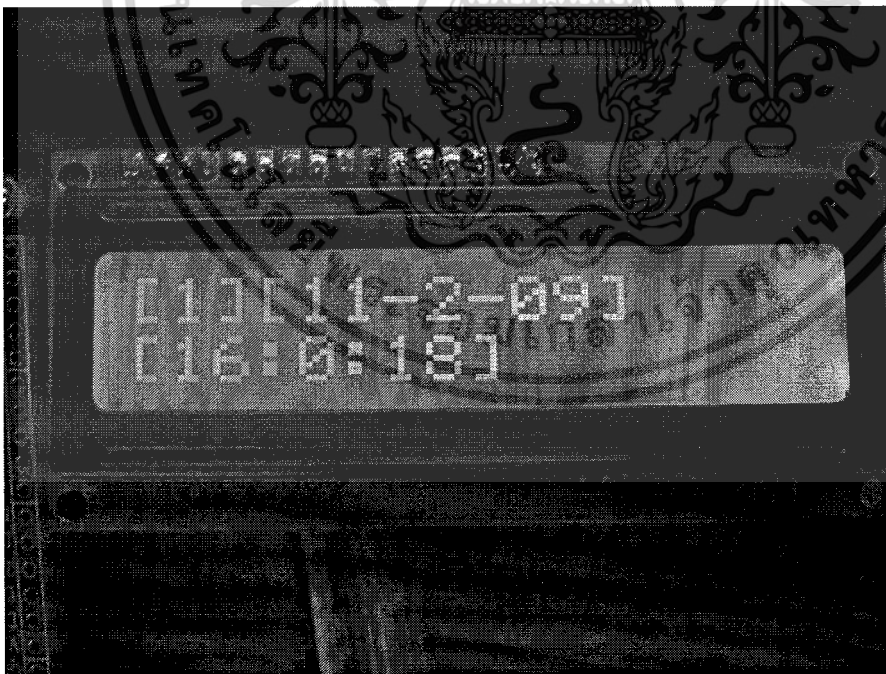
รูปที่ 4.6 จำนวนสินค้าในแต่ละช่อง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.7 การส่งข้อมูลผ่าน จีพีอาร์เอส โมดูล

เมื่อทำการส่งข้อมูลออกไปยัง เซิร์ฟเวอร์ แล้ว ต่อจากนั้นกดปุ่ม '*' เพื่อทำการเข้าสู่โหมดสแตนด์บายล์ โดยในโหมดนี้นั้น เมื่อมีสินค้าที่หล่นมากระทบกับ สวิตช์ แล้วจะมีการ เซฟข้อมูลแต่ละครั้งเพื่อที่จะทำการส่งออกไปทุกๆ ชั่วโมง



รูปที่ 4.8 การเซฟข้อมูลรายละเอียดของสินค้า

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.5 การทดลองรับข้อมูลของฝั่งเซิร์ฟเวอร์

เมื่อทุกครั้งที่มีการส่งสินค้าข้อมูลจะถูกบันทึกไว้และนำไปส่งต่อในภายหลังเมื่อทำการส่งไปที่เซิร์ฟเวอร์ แล้ว เซิร์ฟเวอร์ จะทำการเก็บข้อมูล ลงฐานข้อมูล

	fill_id	v_num	slot_num	refill_date	refill_qty	refill_time
<input type="checkbox"/>	69	2	1	2009-02-11	30	15:58:29
<input type="checkbox"/>	70	2	2	2009-02-11	28	15:58:13
<input type="checkbox"/>	71	1	2	2009-02-11	42	15:58:13
<input type="checkbox"/>	72	1	1	2009-02-11	45	15:58:13

รูปที่ 4.9 เซิร์ฟเวอร์ เมื่อทำการรับข้อมูลจาก จีพีอาร์เอส โมดูล

เมื่อได้ข้อมูลแล้ว ก็จะสามารถนำข้อมูลที่ได้นั้นมาแสดงบน เว็บเบราว์เซอร์ ได้ทุกที่ที่มีการเชื่อมต่อ อินเทอร์เน็ต

Vending Machine

เครื่องขายสินค้าอัตโนมัติ

MAIN Store Index Branch Status Refill Goods Historical Planning Refill Report Log Out

ค้นหาสินค้า

ยืนยันการส่งสินค้า

สินค้ารวม 136 กระป๋อง
รถส่งสินค้า สท-4567 กรุงเทพ

ลำดับที่	สาขา	สินค้า	จำนวน
1	ลาดพร้าว	nescafe latte	44
2	ลาดพร้าว	nescafe espresso	42
3	หัวหมาก	nescafe latte	28
4	หัวหมาก	nescafe espresso	22

รูปที่ 4.10 ดูข้อมูลผ่านทาง เว็บเบราว์เซอร์

โดยที่ข้อมูลต่างๆได้ทำการวางแผนลง ฐานข้อมูล ที่จะทำการเติมสินค้าออกไปจริง โดยข้อมูลนี้ได้ถูกบันทึกที่ เซิร์ฟเวอร์ เอง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Show: 200 row(s) starting from record # 0
in horizontal mode and repeat headers after 100 cells

Sort by key: None Go

	fill_date	slot_num	v_num	truck_id	fill_qty
<input type="checkbox"/> ✎ ✕	2009-02-12 01:37:00	1	1	สท-4567 กรุงเทพ	44
<input type="checkbox"/> ✎ ✕	2009-02-12 01:37:00	1	2	สท-4567 กรุงเทพ	28
<input type="checkbox"/> ✎ ✕	2009-02-12 01:37:00	2	1	สท-4567 กรุงเทพ	42
<input type="checkbox"/> ✎ ✕	2009-02-12 01:37:00	2	2	สท-4567 กรุงเทพ	22

Check All / Uncheck All With selected: ✎ ✕

Show: 200 row(s) starting from record # 0
in horizontal mode and repeat headers after 100 cells

รูปที่ 4.11 ข้อมูลวางแผนก่อนทำการเติมสินค้าจริง

ซึ่งสามารถดูข้อมูลนี้ได้จากทาง เว็บเบราว์เซอร์ เองเช่นกัน

Vending Machine

เครื่องขายสินค้าอัตโนมัติ

MAIN Store Index Branch Status Refill Goods Historical Planning **Refill Report** Log Out

ค้นหาสินค้า

รายงานการเติมสินค้า

วันที่	สาขา	สินค้า	ช่อง ที่	เติม สินค้า	เติม จริง	คลาด เคลื่อน
2009-02-11	ลาดพร้าว	nescafe latte	1	44	45	1
2009-02-11	หัวหมาก	nescafe latte	1	28	30	2
2009-02-11	หัวหมาก	nescafe espresso	2	22	28	6

รูปที่ 4.12 ข้อมูลวางแผนก่อนทำการเติมจริงผ่านทาง เว็บเบราว์เซอร์

จากเว็บเบราว์เซอร์สามารถแสดงข้อมูลต่างๆผ่านทางเว็บ โดยสามารถแสดงเป็นกราฟ รายละเอียดดังนี้

- ยอดขายสินค้าเรียงตามเวลา
- ยอดขายสินค้าเรียงตามสินค้า
- ยอดขายสินค้าเรียงตามสาขา
- ดูยอดขายย้อนหลัง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

MAIN

Historical Trading

Sort By Time

Sort By Branch

Sort By Goods

Log Out

ค้นหาสินค้า

Search

ราชนาฬิกา

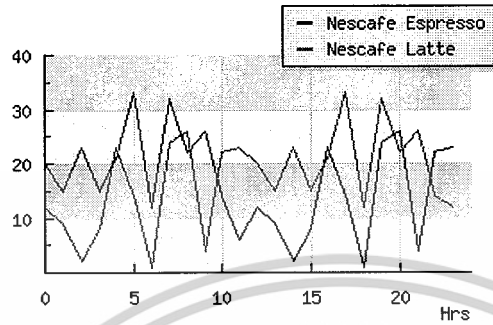
1

กุมภาพันธ์

2009

GO

ยอดขายสินค้าเรียงตามเวลา



รูปที่ 4.13 ยอดขายตามเวลา

MAIN

Historical Trading

Sort By Time

Sort By Branch

Sort By Goods

Log Out

ค้นหาสินค้า

Search

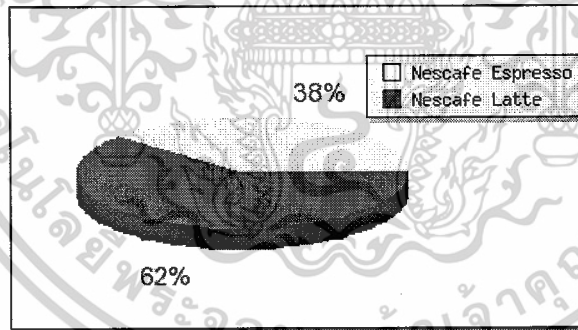
ลาดพร้าว

กุมภาพันธ์

2009

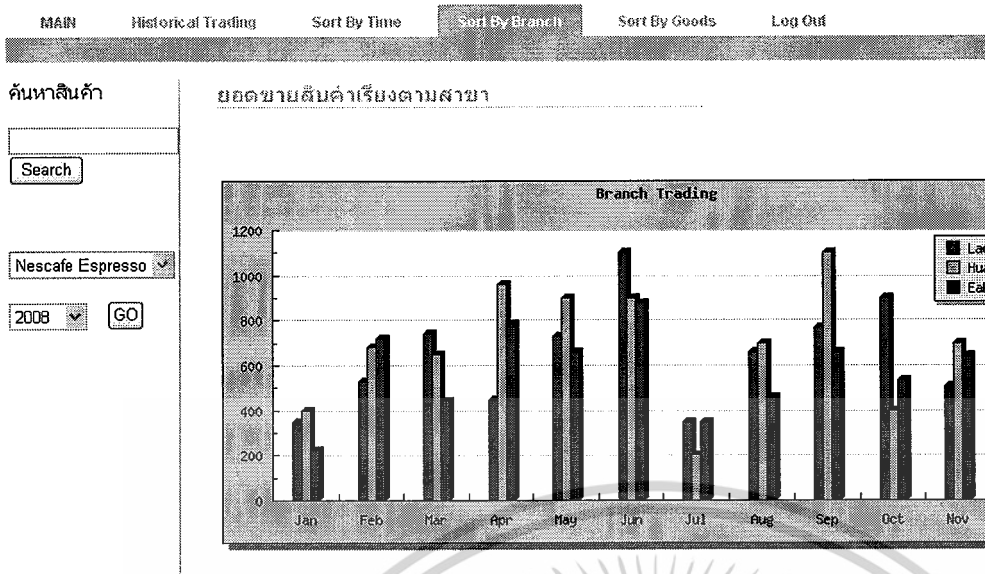
GO

ยอดขายสินค้าเรียงตามสินค้า



รูปที่ 4.14 ยอดขายตามสินค้า

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.15 ยอดขายตามสาขา

Vending Machine

เครื่องขายสินค้าอัตโนมัติ

MAIN Historical Trading Sort By Time Sort By Branch Sort By Goods Log Out

ค้นหาสินค้า

Search

บางนา

ยอดขายสินค้าย้อนหลัง

สาขา	สินค้า	วันที่	เวลา
บางนา	nescafe latte	2009-01-10	09:00:05
บางนา	nescafe latte	2009-01-10	09:10:05
บางนา	nescafe latte	2009-01-11	10:00:05
บางนา	nescafe latte	2009-01-11	11:10:48
บางนา	nescafe latte	2009-01-11	15:00:05
บางนา	nescafe latte	2009-01-11	17:28:05
บางนา	nescafe latte	2009-01-12	09:00:05
บางนา	nescafe latte	2009-01-13	09:10:05
บางนา	nescafe latte	2009-01-13	10:00:05
บางนา	nescafe espresso	2009-01-13	11:00:05
บางนา	nescafe latte	2009-01-14	12:00:05
บางนา	nescafe espresso	2009-01-14	13:00:05
บางนา	nescafe latte	2009-01-16	14:00:05
บางนา	nescafe espresso	2009-01-24	15:00:05

รูปที่ 4.16 ยอดขายย้อนหลัง

4.6 ผลการทดลองการทำงานของระบบ

จากการทดลองสามารถสรุปได้ว่าการส่งข้อมูลไปยัง เซิร์ฟเวอร์ นั้นมีข้อเสียในเรื่องของการเชื่อมต่อซึ่งจะทำให้การส่งข้อมูลนั้นเกิดการผิดพลาดซึ่งทำให้ข้อมูลที่ได้รับไม่ถูกต้อง โดยที่ทำการทดลองจำหน่ายสินค้าจากตู้หยอดเหรียญและทำการส่งข้อมูลของสินค้าไปยัง เซิร์ฟเวอร์ ด้วย

เอกสารนี้เป็นเอกสารที่ส่งในวันเสาร์หรือการชำระเงินเพื่อการศึกษาเท่านั้น เมื่อนูญตาเดินทางไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จำนวนสินค้าที่แตกต่างกัน ซึ่งจากการทดลองสามารถสรุปได้ดังนี้

ตารางที่ 4.1 ผลการทดลองในการส่งข้อมูลสินค้า

จำนวนสินค้าที่ขายจริง	จำนวนสินค้าที่ส่งได้	เปอร์เซ็นต์การส่ง	เวลาในการส่ง	เวลาเฉลี่ยต่อกระป๋อง
10 กระป๋อง	9 กระป๋อง	90 %	10.09 นาที	1.12 นาที
20 กระป๋อง	20 กระป๋อง	100 %	5.33 นาที	0.27 นาที
30 กระป๋อง	16 กระป๋อง	53.3 %	5.56 นาที	0.34 นาที

จากตารางแสดงผลการทดลองนั้นจะเห็นได้ว่าการส่งข้อมูลของการขายสินค้านั้นเปอร์เซ็นต์ของการส่งสินค้าที่สามารถส่งได้ในแต่ละครั้งนั้นจะไม่ได้ขึ้นอยู่กับจำนวนสินค้าที่ได้ทำการส่งแต่ละครั้งแต่จะขึ้นอยู่กับส่วนของ ไมโครคอนโทรลเลอร์อาจเกิดการ Error หรืออาจเกิดจากการติดต่อรหว่าง จีพีอาร์เอส โมดูล กับ เซิร์ฟเวอร์ ซึ่งอาจจะทำให้การส่งข้อมูลนั้นหยุดทำการส่งข้อมูลไปเลยหรืออาจเกิดการหายไปของข้อมูลบางส่วน สำหรับในเรื่องของเวลาในการส่งแต่ละครั้งนั้นจะใช้เวลาในการส่งที่ไม่เท่ากันเนื่องจากการติดต่อรหว่าง จีพีอาร์เอส โมดูล กับ เซิร์ฟเวอร์เกิดการ ตีเลย์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 5

สรุปและวิจารณ์

5.1 การดำเนินงาน

การทดลองจะแบ่งการทำงานเป็น 3 ส่วนคือ ส่วนตู้หยอดเหรียญ ส่วนส่งข้อมูลและเซิร์ฟเวอร์ ส่วนตู้หยอดเหรียญได้ทดลองการหยอดเหรียญ การตรวจนับเหรียญและการจ่ายสินค้าเหมือนกับ การใช้งานจริง ส่วนส่งข้อมูลได้ทดลองการรับข้อมูลจากเซนเซอร์จากช่องสินค้า เก็บข้อมูลลง หน่วยความจำ ตั้งเวลาในการส่งข้อมูล โดยใช้ จีพีอาร์เอส รวมถึงรองรับกรณีผิดพลาดต่างๆ ที่ อาจเกิดขึ้นได้ ส่วนเซิร์ฟเวอร์จะทดลองรับข้อมูล ติดต่อกับฐานข้อมูล และแสดงข้อมูลต่างๆ ผ่านเว็บเบราว์เซอร์ เมื่อทุกส่วนทำงานได้จึงนำมารวมเพื่อทดลองการใช้งานระบบทั้งหมด โดยทำ การหยอดเหรียญเพื่อทดสอบการจ่ายสินค้าและการเก็บข้อมูลการขาย พร้อมทั้งการส่งข้อมูล ดังกล่าวไปยังเซิร์ฟเวอร์ นอกจากนั้นจะส่งข้อมูลการเติมสินค้าเมื่อมีการเติมสินค้าอีกด้วย ด้าน เซิร์ฟเวอร์จะทำการรับข้อมูลมาจากเครื่องขายสินค้าเพื่อเก็บลงฐานข้อมูล จากนั้นจะแสดงผลผ่าน เว็บเบราว์เซอร์

5.2 ผลการดำเนินงาน

ผลการดำเนินงานเริ่มจากการเขียน โปรแกรมเพื่อการรองรับการใช้งานเราสามารถใช้งานได้ ดังนี้

- เครื่องควบคุม จีพีอาร์เอส โมดูล สามารถรองรับเครื่องขายสินค้าได้ไม่เกิน 65,535 เครื่อง
- การใช้งานเครื่องควบคุม จีพีอาร์เอส โมดูล ต้องการหน่วยเวลาในการส่งทุกๆ ชั่วโมง ขึ้นอยู่ กับจำนวนสินค้า
- เครื่องควบคุม จีพีอาร์เอส โมดูล สามารถเก็บข้อมูลได้ 32 ชุดข้อมูลต่อการส่งทุกๆ ชั่วโมง
- สามารถใช้กับเครื่องขายสินค้าที่มีจำนวนช่องใส่สินค้าไม่เกิน 10 ช่อง
- เวลาเฉลี่ยในการส่ง 35 วินาทีต่อหนึ่งชุดข้อมูล
- เปอร์เซนต์ของการส่งข้อมูลประมาณ 95%
- ถ้าเกิดข้อผิดพลาดขณะการส่งหรือเกิดเหตุการณ์ฉุกเฉิน เช่น ไฟดับ จะเก็บข้อมูลไว้ใน หน่วยความจำและจะส่งข้อมูลในชั่วโมงถัดไปหรือเมื่อระบบพร้อมทำงาน
- เว็บแอปพลิเคชันรองรับการทำงานของผู้ใช้งานต่างๆ ดังนี้
 - ผู้ใช้งานทั่วไป สามารถดูและค้นหาสินค้า จำนวนสินค้าคงเหลือและตู้สาขาที่ ให้บริการ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- ผู้ใช้งานทั่วไป สามารถดูและค้นหาสินค้า จำนวนสินค้าคงเหลือและตู้สาขาที่ให้บริการ
- ผู้ดูแลระบบ สามารถจัดการเพิ่ม ลบหรือเปลี่ยนแปลงรายละเอียดต่างๆ ของระบบ สมาชิก สินค้า และตู้สาขา
- ฝ่ายจัดการสินค้า สามารถดูสถานะสินค้าในตู้สาขา วางแผนการเติมสินค้า ดูรายงานการเปรียบเทียบระหว่างการวางแผนและการเติมสินค้าจริง นอกจากนั้นระบบยังมีการแจ้งเตือนหากมีสินค้าใกล้จะหมด
- ผู้บริหารหรือฝ่ายการตลาด สามารถดูยอดขายสินค้าย้อนหลัง แสดงเป็นตารางและกราฟ โดยเรียงตามยอดขายเปรียบเทียบรายวัน, ยอดขายเปรียบเทียบรายเดือน, ยอดขายเปรียบเทียบรายปี

5.3 ปัญหา อุปสรรค และแนวทางการแก้ไข

5.3.1 ปัญหาและอุปสรรคในการทำงาน

- มีปัญหาเกี่ยวกับตู้หยอดเหรียญอัตโนมัติ ซึ่งเราไม่สามารถควบคุมการให้แหล่งจ่ายไฟกับตัวสเตปป์มอเตอร์รวมถึงการแสดงผลไฟออกทาง 7-Segment
- การส่งข้อมูลผ่านทาง จีพีอาร์เอส โมดูล นั้นมีปัญหาในเรื่องของการส่งข้อมูลแต่ละครั้งเมื่อมีการ เชื่อมต่อ ครั้งแรก ต้องขึ้นอยู่กับ เซิร์ฟเวอร์ รวมถึงโปรโตคอล รวมถึงการ ดีเลย์ ซึ่งเราไม่สามารถควบคุมเวลาในการส่งข้อมูล และจะทำให้เสียเวลาในการส่งข้อมูลทุกครั้งที่ทำ การส่ง และการทำงานของไมโครคอนโทรลเลอร์เมื่อมีการส่งเกิดการผิดพลาดไปนั้น จะมีการ Error ของไมโครคอนโทรลเลอร์
- การเขียนโปรแกรมไม่สามารถรองรับในกรณีที่เกิดการส่งข้อมูลและการซื้อสินค้าพร้อมๆกัน

5.3.2 แนวทางการแก้ไข

- การแก้ไขสำหรับแหล่งจ่ายไฟให้กับสเตปป์มอเตอร์เราควรเปลี่ยนตัวสเตปป์มอเตอร์ไปใช้ 24 โวลท์และแหล่งจ่ายไฟที่ไฟที่ไม่มีความร้อนมากไป
- การแก้ไขในการส่งข้อมูลเราไม่สามารถแก้ไขในเรื่องของ โปรโตคอลได้ แต่เราสามารถใช้การเก็บข้อมูลที่มากขึ้นได้เพื่อรองรับส่งที่ช้าเกินไปและการส่งครั้งถัดไป
- เปลี่ยนตัวไมโครคอนโทรลเลอร์ใหม่เพื่อรองรับการเขียน โปรแกรมที่มีขนาดใหญ่กว่า

5.4 ข้อเสนอแนะ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ถ้าสำหรับระบบในการเก็บข้อมูลนั้น สามารถใช้งานได้จริง ซึ่งแนวทางพัฒนาต่อไปนั้น เราสามารถเลือกการเชื่อมต่อที่สามารถลด คิเลย์ ในการเชื่อมต่อและการส่งข้อมูลทุกครั้งเพื่อ ลดเวลาการเชื่อมต่อ รวมถึงเพิ่มขนาดของการบันทึกข้อมูลเมื่อมีการซื้อสินค้า ให้มีขนาดเพียงพอต่อความต้องการ และการออกแบบตู้หยอดเหรียญอัตโนมัตินั้นสามารถออกแบบโดยใช้วัสดุที่แข็งแรงและทนทานต่อการใช้งานนานและการออกแบบ เซิร์ฟเวอร์ นั้นสามารถออกแบบฐานข้อมูลได้ตามขนาดของข้อมูลที่เราต้องการ และการส่งข้อมูลนั้นเราสามารถกำหนดข้อมูลในการส่งได้แล้วแต่ความต้องการของระบบ ซึ่งสามารถเป็นแนวทางในการพัฒนาต่อไปได้เช่นกัน



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บรรณานุกรม

ชัยวัฒน์ ลิ้มพรจิตรวิไล และ ณีฐฎพล วงศ์สุนทรชัย. 2521. **เรียนรู้และปฏิบัติการไมโครคอน**

โทรเตอร์ PIC16F877. กรุงเทพฯ : วิเจ.พริ้นติ้ง

ประจัน พลังสันติคุณ. 2521. **เรียนรู้และใช้งาน CCS C คอมไพเลอร์ เขียนโปรแกรมภาษา**

ควบคุมไมโครคอนโทรเลอร์ PIC. กรุงเทพฯ : อินโนเวตีฟ เอ็กเพอริเมนต์

สมศักดิ์ โชคชัยชุตikulm. 2547. **อินไซต์ PHP 5.** กรุงเทพฯ : โปรวิชั่น

Ali Mazidi, M. and Gillispie Mazidi, J. 1998. **The 80x86 Ibm Pc And Comp Atible**

Computers Volume II. New Jersey : Simon & Schuster



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก ก.

โปรแกรม ควบคุม จีพีอาร์เอส โมดูล

```
#define _PIC16F877_  
#ifdef _PIC16F877_  
#include <16F877.h>  
#define CLOCK_SP 4000000  
#endif  
#fuses HS  
#fuses NOLVP,NOWDT  
#fuses NOPROTECT  
#use delay (clock = CLOCK_SP)  
#use i2c(master,fast,sda=PIN_C4,scl=PIN_C3)  
#use rs232(baud=9600 ,xmit=PIN_C6,rcv=PIN_C7,stream=NOVEL)  
int forsave = 0;  
int item[5]={0,0,0,0,0},temp,itemselect=0;  
#define BUFFER_SIZE 32  
BYTE buffer[BUFFER_SIZE];  
BYTE next_in = 0;  
BYTE next_out = 0;  
byte slot;  
int vending=2;  
boolean for_tcp=false;  
#include <string.h>  
#include <DS1307_2.c>  
#include <LCD.c>  
#include <Maxtrix.c>  
#DEFINE ENDPROM 0x0D  
#DEFINE NEWROW 0x0A  
#DEFINE END 0x13
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

void clear_buff(char *st);
void LCD_String(char *st,int dly);
void ini_gprs();
void chkfor_close();
void beginsend();
void endsend();
void senddata_ini1();
void senddata_ini2();
void chkfor_eeprom();
void save_eeprom(int slot,int date,int month,int yr,int down,int hr,int min,int sec);
void send_eeprom();
void send_dataitem1();
void send_dataitem2();
boolean rb6=false,rb7=false;
#priority rb,rda
#int_rb
void rb_int(){
if(input(PIN_B7)){
  delay_ms(200);
  --item[0];
  slot=1;
  rb6=true;
}
else if(input(PIN_B6)){
  delay_ms(200);
  --item[1];
  slot=2;
  rb7=true;
}
}
}
}
#int_rda

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

void serial_isr() {
    int t;
    buffer[next_in]=getc();
    t=next_in;
    next_in=(next_in+1) % BUFFER_SIZE;
    if(next_in==next_out)
        next_in=t;    // Buffer full !!
}

```

```

#define bkbhit (next_in!=next_out)

```

```

BYTE bgetc() {

```

```

    BYTE c;

```

```

    while(!bkbhit);

```

```

    c=buffer[next_out];

```

```

    next_out=(next_out+1) % BUFFER_SIZE;

```

```

    return(c);
}

```

```

void chk_cha(){

```

```

    BYTE c;

```

```

    printf(lcd_putc,"\f");

```

```

do{

```

```

    c=bgetc();

```

```

    if(c=='O'){

```

```

        c=bgetc();

```

```

        if(c=='K'){

```

```

            printf(lcd_putc,"\f>>>>>>>OK<<<<<<<<");

```

```

            delay_ms(1000);

```

```

            return;

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

}
if(c == 'E'){
    c=bgetc();
    if(c=='R'){
        printf(lcd_putc, "\fERROR....");
        delay_ms(1000);
        return;
    }
}
if(c == 'F' | c == 'P'){
    c=bgetc();
    if(c=='A' | c == 'D'){
        printf(lcd_putc, "\fFAIL NOW SAVING..");
        delay_ms(1000);
        return;
    }
}
}while(c != NULL);
printf(lcd_putc, "\f");
clear_buff(buffer);
delay_ms(200);
}
void chk_cha_tcp(){

```

```

BYTE c;

```

```

printf(lcd_putc, "\f");

```

```

do{

```

```

c=bgetc();

```

```

if(c=='O'){

```

```

c=bgetc();

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

if(c=='K'){
    printf(lcd_putc,"\fCONNECT OK.");
    delay_ms(250);
    return;
}

}

if(c == 'E'){
    c=bgetc();
    if(c=='R'){
        printf(lcd_putc,"\fERROR..");
        delay_ms(250);
        return;
    }
}

if(c == 'F'){
    c=bgetc();
    if(c=='A'){
        printf(lcd_putc,"\fFAIL NOW SAVING..");
        delay_ms(250);

        for_tcp=true;
        return;
    }
}

}while(c != NULL);
printf(lcd_putc,"\f");
clear_buff(buffer);
delay_ms(200);
}

char i=49;

void main(){

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

enable_interrupts(GLOBAL);
enable_interrupts(INT_RDA);
enable_interrupts(INT_RB);
lcd_init();
ds1307_init();
inimaxtrix();
ds1307_set_date_time(11,2,9,20,15,58,11);
ds1307_get_date(day,month,yr,dow);
ds1307_get_time(hrs,min,sec);
label1:
printf(lcd_putc,"\fMODULE FOR GPRS");
i=number();
delay_ms(500);
switch (i){
case '1' : lcd_putc("\fInsert slot\n");
            itemselect=get_item1();
            switch_item(itemselect);
            goto label1;
case '2' : break;
case '3': ini_gprs();
            goto label1;
case '4': senddata_ini1();
            goto label1;
case '5': senddata_ini2();
            goto label1;
case '6': printf(lcd_putc,"\fSlot 1 = %d",item[0]);
            delay_ms(1500);
            goto label1;
case '7': printf(lcd_putc,"\fSlot 2 = %d",item[1]);
            delay_ms(1500);
            goto label1;
case '8': printf("%c",END);

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

printf("at+cipclose");
printf(lcd_putc,"\fat+cipclose");
delay_ms(300);
printf("%c",ENDPROM);
printf("%c",BEGIN);
delay_ms(500);
chk_cha();
break;
case '9': printf("ATA");
printf("%c",ENDPROM);
printf(lcd_putc,"\f");
break;
case '*': break;
default : goto label1;
}
while(true){
ds1307_get_date(day,month,yr,dow);
ds1307_get_time(hrs,min,sec);
printf(lcd_putc,"\f%02d/%%02d/%d%\n",day,month,dow,yr);
printf(lcd_putc,"%02d:%%02d:%%02d", hrs,min,sec);
delay_ms(150);
if(get_standby()=='#'){
goto label1;
}

else if(min == ';'){
send_eeprom();
}

else if(rb7==true){
delay_ms(500);
save_eeprom(slot,day,month,dow,yr,hrs,min,sec);
rb7=false;
}

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

}

else if(rb6==true){

    save_eeprom(slot,day,month,dow,yr,hrs,min,sec);

    delay_ms(500);

    rb6=false;

}

}

}

void send_dataitem1(){

    beginsend();

    printf("GET /TEST.PHP?T1=%d&T2=1&T3=%d%d-%d-

%d&T4=%d:%d:%d",VENDING,dow,yr,month,day,hrs,min,sec);

    endsend();

}

void send_dataitem2(){

    beginsend();

    printf("GET /TEST.PHP?T1=%d&T2=2&T3=%d%d-%d-

%d&T4=%d:%d:%d",VENDING,dow,yr,month,day,hrs,min,sec);

    endsend();

}

void senddata_ini1(){

    beginsend();

    printf("GET /FILL.PHP?T1=%d&T2=1&T3=%d&T4=%d%d-%d-

%d&T5=%d:%d:%d",VENDING,item[0],dow,yr,month,day,hrs,min,sec);

    endsend();

}

void senddata_ini2(){

    beginsend();

    printf("GET /FILL.PHP?T1=%d&T2=2&T3=%d&T4=%d%d-%d-

%d&T5=%d:%d:%d",VENDING,item[1],dow,yr,month,day,hrs,min,sec);

```

เอกสารนี้เป็นเอกสารที่ลงวันเวลาสำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

endsend();
}

void save_eeprom(int slot,int date,int month,int yr,int down,int hr,int min,int sec){
    printf(lcd_putc,"\f[%d][%d-%d-%d%]\n[%d:%d:%d]",slot,date,month,yr,down,hr,min,sec);
    delay_ms(5000);
    write_eeprom (forsave,slot);
    delay_ms(100);
    ++forsave;
    write_eeprom (forsave,date);
    delay_ms(100);
    ++forsave;
    write_eeprom (forsave,month);
    delay_ms(100);
    ++forsave;
    write_eeprom (forsave,yr);
    delay_ms(100);
    ++forsave;
    write_eeprom (forsave,down);
    delay_ms(100);
    ++forsave;
    write_eeprom (forsave,hr);
    delay_ms(100);
    ++forsave;
    write_eeprom (forsave,min);
    delay_ms(100);
    ++forsave;
    write_eeprom (forsave,sec);
    delay_ms(100);
    //++forsave;
}

```

```
void send_eeprom(){
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

while(forsave!=0){
sec = read_eeprom(forsave);
--forsave;
min = read_eeprom(forsave);
--forsave;
hrs = read_eeprom(forsave);
--forsave;
dow = read_eeprom(forsave);
--forsave;
yr = read_eeprom(forsave);
--forsave;
month = read_eeprom(forsave);
--forsave;
day = read_eeprom(forsave);
--forsave;
slot = read_eeprom(forsave);
printf(lcd_putc, "\f[%d][%d-%d-%d-
%d-%d]\n[%d:%d:%d]", VENDING, slot, day, month, yr, dow, hrs, min, sec);
delay_ms(5000);
beginsend();
printf("GET /TEST.PHP?T1=%s&T2=%d&T3=%d-%d-
%d&T4=%d:%d:%d", VENDING, slot, day, month, yr, dow, hrs, min, sec);
endsend();
}
}

void beginsend(){
printf(lcd_putc, "\fNow..Sendind..20Sec");
printf("%c", END);
printf("at+cipclose");
printf(lcd_putc, "\fat+cipclose");
delay_ms(300);
printf("%c" BEGIN);

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

printf("%c",END);
printf("%c",ENDPROM);
printf("%c",BEGIN);
chk_cha();
ds1307_get_date(day,month,yr,dow);
ds1307_get_time(hrs,min,sec);
printf("%c",END);
printf("at+cipstart=\\"TCP\\",\\"161.246.6.46\\",\\"2020\\");
printf(lcd_putc, "\fat+cipstart=\\"TCP\\",\\"161.246.6.46\\",\\"2020\\");
delay_ms(300);
printf("%c",BEGIN);
printf("%c",END);
printf("%c",ENDPROM);
printf("%c",BEGIN);
chk_cha();
chk_cha_tcp();
if(for_tcp == true){
    return;
}
printf("%c",END);
printf("at+cipsend");
printf("%c",BEGIN);
printf("%c",END);
printf("%c",ENDPROM);
printf("%c",BEGIN);
printf(lcd_putc, "\f..NOW SENDING..");
}
void endsend(){
    if(for_tcp == true){
        for_tcp = false;
        return;
    }

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

delay_ms(250);
printf("%c",END);
printf("%c",NEWROW);
printf("%c",BEGIN);
delay_ms(250);
printf("%c",END);
printf("%c",ENDPROM);
printf("%c",BEGIN);
delay_ms(250);
printf("%c",END);
printf("%c",NEWROW);
printf("%c",BEGIN);
delay_ms(250);
printf("%c",END);
printf("%c",CTZ);
printf("%c",BEGIN);
delay_ms(250);
}

```

```

void ini_gprs(){
printf("%c",END);
printf("at+creg=1");
printf(lcd_putc,"\fat+creg=1");
delay_ms(500);
printf("%c",ENDPROM);
printf("%c",BEGIN);
chk_cha();
printf("%c",END);
printf("AT+CGDCONT=1,\"IP\", \"www.dtac.co.th\");
printf(lcd_putc,"\fAT+CGDCONT=1,\"IP\", \"www.dtac.co.th\");
delay_ms(500);
printf("\f%c",ENDPROM);

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้




```

delay_ms(500);
printf("%c",ENDPROM);
printf("%c",BEGIN);
chk_cha();
}

void clear_buff(char *st){
int i;
while(i <= 32)
{
*st[i]=NULL;
i++;
}
}

/*****DS1307_2.h*****/
#define RTC_SDA PIN_C4
#define RTC_SCL PIN_C3

//use i2c(master, sda=RTC_SDA, scl=RTC_SCL)

//BYTE bin2bcd(BYTE binary_value);
//BYTE bcd2bin(BYTE bcd_value);

BYTE sec;
BYTE min;
BYTE hrs;
BYTE day;
BYTE month;
BYTE yr;
BYTE dow;

int8 bin2bcd(int8 binary_value)
{

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

char retval;

temp = binary_value;
retval = 0;

while(1)
{
    // Get the tens digit by doing multiple subtraction
    // of 10 from the binary value.
    if(temp >= 10)
    {
        temp -= 10;
        retval += 0x10;
    }
    else // Get the ones digit by adding the remainder.
    {
        retval += temp;
        break;
    }
}

return(retval);
}

// Input range - 00 to 99.
char bcd2bin(char bcd_value)
{
    char temp;

    temp = bcd_value;
    // Shifting upper digit right by 1 is same as multiplying by 8.
    temp >= 1;

```



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

// Isolate the bits for the upper digit.
temp &= 0x78;

// Now return: (Tens * 8) + (Tens * 2) + Ones

return(temp + (temp >> 2) + (bcd_value & 0x0f));
}

```

```

void ds1307_init(void)
{
    BYTE seconds = 0;

    i2c_start();
    i2c_write(0xD0); // WR to RTC
    i2c_write(0x00); // REG 0
    i2c_start();
    i2c_write(0xD1); // RD from RTC
    seconds = bcd2bin(i2c_read(0)); // Read current "seconds" in DS1307
    i2c_stop();
    seconds &= 0x7F;

    delay_us(3);

    i2c_start();
    i2c_write(0xD0); // WR to RTC
    i2c_write(0x00); // REG 0
    i2c_write(bin2bcd(seconds)); // Start oscillator with current "seconds value
    i2c_start();
    i2c_write(0xD0); // WR to RTC
    i2c_write(0x07); // Control Register
    i2c_write(0x80); // Disable squarewave output pin
    i2c_stop();

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

}

```
void ds1307_set_date_time(BYTE day, int8 mth, BYTE year, BYTE dow, BYTE hr, BYTE min,
BYTE sec)
```

```
{
    sec &= 0x7F;
    hr &= 0x3F;

    i2c_start();
    i2c_write(0xD0);    // I2C write address
    i2c_write(0x00);    // Start at REG 0 - Seconds
    i2c_write(bin2bcd(sec));    // REG 0
    i2c_write(bin2bcd(min));    // REG 1
    i2c_write(bin2bcd(hr));    // REG 2
    i2c_write(bin2bcd(dow));    // REG 3
    i2c_write(bin2bcd(day));    // REG 4
    i2c_write(bin2bcd(mth));    // REG 5
    i2c_write(bin2bcd(year));    // REG 6
    i2c_write(0x80);    // REG 7 - Disable squarewave output pin
    i2c_stop();
}
```

```
void ds1307_set_time(byte hr, byte min, byte sec)
```

```
{
    sec &= 0x7F;
    hr &= 0x3F;

    i2c_start();
    i2c_write(0xD0);    // I2C write address
    i2c_write(0x00);    // Start at REG 0 - Seconds
    i2c_write(bin2bcd(sec));    // REG 0
    i2c_write(bin2bcd(min));    // REG 1
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

i2c_write(bin2bcd(hr)); // REG 2

//i2c_write(0x80); // REG 7 - Disable squarewave output pin

i2c_stop();
}

void ds1307_set_date(byte day, byte mth, byte year, byte dow)
{

i2c_start();
i2c_write(0xD0); // I2C write address
i2c_write(0x03); // Start at REG 0 - Seconds
i2c_write(bin2bcd(dow));
i2c_write(bin2bcd(day));
i2c_write(bin2bcd(mth));
i2c_write(bin2bcd(year));
i2c_write(0x80); // REG 7 - Disable squarewave output pin
i2c_stop();
}

void ds1307_get_date(BYTE &day, BYTE &mth, BYTE &year, BYTE &dow)
{
i2c_start();
i2c_write(0xD0);
i2c_write(0x03); // Start at REG 3 - Day of week
i2c_start();
i2c_write(0xD1);
dow = bcd2bin(i2c_read() & 0x7f); // REG 3
day = bcd2bin(i2c_read() & 0x3f); // REG 4
mth = bcd2bin(i2c_read() & 0x1f); // REG 5
year = bcd2bin(i2c_read(0)); // REG 6
i2c_stop();
}

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
void ds1307_get_time(BYTE &hr, BYTE &min, BYTE &sec)
```

```
{
  i2c_start();
  i2c_write(0xD0);
  i2c_write(0x00);      // Start at REG 0 - Seconds
  i2c_start();
  i2c_write(0xD1);
  sec = bcd2bin(i2c_read() & 0x7f);
  min = bcd2bin(i2c_read() & 0x7f);
  hr = bcd2bin(i2c_read(0) & 0x3f);
  i2c_stop();
}
```

```
/******Matrix.h*****
```

```
-number()
-inimatrix()
-LCD_String()
-stncpy(char *p,char *s)
-get_num()
-get_frist()
-get_item1()
-get_item1(long int itemA)
```

```
*****/
```

```
#byte TRIS_A = 0x85
```

```
#byte TRIS_B = 0x86
```

```
#bit TRIS_A0 = TRIS_A.0
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
#bit TRIS_A1 = TRIS_A.1
#bit TRIS_A2 = TRIS_A.2
#bit TRIS_A3 = TRIS_A.3
#bit TRIS_A4 = TRIS_A.4
#bit TRIS_A5 = TRIS_A.5
#bit TRIS_B0 = TRIS_B.0
#bit TRIS_B4 = TRIS_B.4
#bit TRIS_B5 = TRIS_B.5
#bit TRIS_B6 = TRIS_B.6
#bit TRIS_B7 = TRIS_B.7
```

```
void inimatrix(){
    set_tris_a(0x03);
    TRIS_B6=1;
    TRIS_B7=1;
    TRIS_B0=1;
    TRIS_B4=0;
    TRIS_B5=0;
    output_a(0);
    output_low(PIN_B0);
```

```
}
char number(){
    while(!input(PIN_A4)||input(PIN_A5)||input(PIN_B0))
```

```
{
    output_a(0);
    output_low(PIN_B0);
    output_high(PIN_A0);
    if(input(PIN_A4))
    {
```

return '1'; นี่เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



```

}
else if(input(PIN_A5))
{
    return '2';
}
else if(input(PIN_B0))
{
    return '3';
}
output_a(0);
output_low(PIN_B0);
output_high(PIN_A1);
if(input(PIN_A4))
    return '4';
else if(input(PIN_A5))
    return '5';
else if(input(PIN_B0))
    return '6';
output_a(0);
output_low(PIN_B0);
output_high(PIN_A2);
if(input(PIN_A4))
    return '7';
else if(input(PIN_A5))
    return '8';
else if(input(PIN_B0))
    return '9';
output_a(0);
output_low(PIN_B0);
output_high(PIN_A3);
if(input(PIN_A4))

```

return '*';

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้




```

while(*p++=*s++)
;
}

```

```

int get_item1() {
    int itemA[3]={0,0,0};
    itemA[0] = (int)number();
    lcd_putc(itemA[0]);
    itemA[0]=(itemA[0]-48)*100;
    delay_ms(250);
    itemA[1] = (int)number();
    lcd_putc(itemA[1]);
    itemA[1]=(itemA[1]-48)*10;
    delay_ms(250);
    itemA[2] = (int)number();
    lcd_putc(itemA[2]);
    itemA[2]=(itemA[2]-48)*1;
    delay_ms(250);
    temp = itemA[0]+itemA[1]+itemA[2];
    delay_ms(1000);
    return temp;
}

```

```

char get_standby(){
    output_a(0);
    output_low(PIN_B0);
    output_high(PIN_A3);
    if(input(PIN_A4))
        return '*';
    else if(input(PIN_A5))
        return '0';
    else if(input(PIN_B0))

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

return '#';
}
void switch_item(int slot){
switch(slot){
case 1 : lcd_putc("\fInsert Number\n");
item[0]+=get_item1();
break;
case 2 : lcd_putc("\fInsert Number\n");
item[1]+=get_item1();
break;
case 3 :
break;
case 4 :
break;
case 5 :
break;
default : printf(lcd_putc,"Out of range");
break;
}
}
}

```



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก ข.

โปรแกรม เครื่องหยุดเหรียญอัตโนมัติ

```
#define _PIC16F877A
```

```
#include <16F877A.h>
```

```
#define CLOCK_SP 4000000
```

```
#device ADC=10
```

```
#fuses XT
```

```
#fuses NOLVP, NOWDT
```

```
#fuses NOPROTECT
```

```
#use delay (clock=CLOCK_SP)
```

```
#byte TRIS_A = 0x85
```

```
#byte TRIS_D = 0x88
```

```
#bit TRIS_A0 = TRIS_A.0
```

```
#bit TRIS_A1 = TRIS_A.1
```

```
#bit TRIS_A2 = TRIS_A.2//GND
```

```
#bit TRIS_A3 = TRIS_A.3//VCC
```

```
#bit TRIS_A5 = TRIS_A.5
```

```
#bit TRIS_D2 = TRIS_D.2//GND
```

```
#bit TRIS_D3 = TRIS_D.3//VCC
```

```
#bit TRIS_D4 = TRIS_D.4
```

```
#bit TRIS_D5 = TRIS_D.5
```

```
#include <number.c>
```

```
#include <control.c>
```

```
static float f=0;
```

```
void main (){
```

```
//*****-----SETUP-----*****
```

```
TRIS_A0=1;
```

```
TRIS_A1=0;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

TRIS_A2=0;
TRIS_A3=0;
TRIS_A5=0;
TRIS_D2=0;
TRIS_D3=0;
TRIS_D4=0;
TRIS_D5=0;

setup_adc_ports(RA0_ANALOG);
setup_adc(ADC_CLOCK_INTERNAL);
set_tris_b(0b00000001);
set_adc_channel(0);
//set_tris_a(0b00000001);
set_tris_c(0b00000011);
//set_tris_e(0b00000110);

//*****-----SETUP-----*****
//*****-----RUN-----*****

while(true)
{

selcitem=0;
f=read_adc();
f=f*0.0048f;
chkp(f);
mod10();
show();
check_control();
ctrol1();
ctrol2();

}
}

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
/**-----RUN-----**
```

```
/**-----**
```

```
*****
```

```
void num1-0
```

```
void show()
```

```
*****
```

```
*****/
```

```
static int b=00,m=0,n=0;
```

```
void numf_1(){
```

```
  output_d(0xfe);
```

```
  output_b(0xf3);
```

```
  delay_us(500);
```

```
}
```

```
void numf_2(){
```

```
  output_d(0xfe);
```

```
  output_b(0x48);
```

```
  delay_us(500);
```

```
}
```

```
void numf_3(){
```

```
  output_d(0xfe);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

output_b(0x61);
delay_us(500);
}

void numf_40{
output_d(0xfe);
output_b(0x33);
delay_us(500);
}

void numf_50{
output_d(0xfe);
output_b(0x24);
delay_us(500);
}

void numf_60{
output_d(0xfe);
output_b(0x04);
delay_us(500);
}

void numf_70{
output_d(0xfe);
output_b(0xf0);
delay_us(500);
}

void numf_80{
output_d(0xfe);
output_b(0x00);
delay_us(500);
}

void numf_90{
output_d(0xfe);
output_b(0x20);
delay_us(500);
}

```



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

}

void numf_00{
  output_d(0xfe);
  output_b(0x80);
  delay_us(500);
}

/*****
*****/

void nums_10{
  output_d(0xfd);
  output_b(0xf3);
  delay_us(500);
}

void nums_20{
  output_d(0xfd);
  output_b(0x48);
  delay_us(500);
}

void nums_30{
  output_d(0xfd);
  output_b(0x61);
  delay_us(500);
}

void nums_40{
  output_d(0xfd);
  output_b(0x33);
  delay_us(500);
}

void nums_50{
  output_d(0xfd);
  output_b(0x24);
  delay_us(500);
}

```



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

}

void nums_6(){
    output_d(0xfd);
    output_b(0x04);
    delay_us(500);
}

void nums_7(){
    output_d(0xfd);
    output_b(0xf0);
    delay_us(500);
}

void nums_8(){
    output_d(0xfd);
    output_b(0x00);
    delay_us(500);
}

void nums_9(){
    output_d(0xfd);
    output_b(0x20);
    delay_us(500);
}

void nums_0(){
    output_d(0xfd);
    output_b(0x80);
    delay_us(500);
}

void show(){
    set_tris_d(0xff);
    if(m==0){
        numf_0();
    }

    else if(m==1){

```



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

numf_10);
}
else if(m==2){
numf_20);
}

else if(m==3){
numf_30);
}
else if(m==4){
numf_40);
}
else if(m==5){
numf_50);
}
else if(m==6){
numf_60);
}
else if(m==7){
numf_70);
}
else if(m==8){
numf_80);
}
else if(m==9){
numf_90);
}
if (n==0){
nums_00);
}
else if(n==1){
nums_10);

```



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

}
else if(n==2){
nums_2();
}

else if(n==3){
nums_3();
}

else if(n==4){
nums_4();
}

else if(n==5){
nums_5();
}

else if(n==6){
nums_6();
}

else if(n==7){
nums_7();
}

else if(n==8){
nums_8();
}

else if(n==9){
nums_9();
}

}

void mod10()
{
m = b%10;
n = b/10;

```



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

}

static int selcitem =0,ctro1=0,ctro2=0,temp=0;

static boolean differ=false,differ2=false;

int i=0,l=6;

void ne_d();
void nw_d();
void sw_d();
void se_d();
void ne_a();
void nw_a();
void sw_a();
void se_a();
void buffer(){
while(differ){
if(ctro1==0){
/*****/
for(i=0;i<=(l-1);++i){
se_d();
sw_d();
nw_d();
ne_d();
}
//i=0;
}
else if(ctro1==1){
for(i=0;i<=(l-1);++i){
ne_d();
se_d();
sw_d();
nw_d();
}
//i=0;

```



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

}

differ=false;

}/******
*****/

while(differ2){
if(ctro2==0){
for(i=0;i<=(l-1);i++){
nw_a();
sw_a();
se_a();
ne_a();
}
}
else if(ctro2==1){
for(i=0;i<=(l-1);i++){
ne_a();
nw_a();
sw_a();
se_a();
}
}

differ2=false;
}
}

void ne_d(){
output_high(PIN_D2);
output_high(PIN_D3);
output_low(PIN_D4);
output_low(PIN_D5);
}

```



ยกเว้นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

delay_ms(100);
}

void nw_d(){
  output_high(PIN_D4);
  output_high(PIN_D3);
  output_low(PIN_D2);
  output_low(PIN_D5);
  delay_ms(100);
}

void sw_d(){
  output_high(PIN_D4);
  output_high(PIN_D5);
  output_low(PIN_D2);
  output_low(PIN_D3);
  delay_ms(100);
}

void se_d(){
  output_high(PIN_D2);
  output_high(PIN_D5);
  output_low(PIN_D3);
  output_low(PIN_D4);
  delay_ms(100);
}

void ne_a(){
  output_high(PIN_A5);
  output_high(PIN_A1);
  output_low(PIN_A2);
  output_low(PIN_A3);
  delay_ms(100);
}

void nw_a(){
  output_high(PIN_A1);

```



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

output_high(PIN_A2);
output_low(PIN_A5);
output_low(PIN_A3);
delay_ms(100);
}

void sw_a(){
output_high(PIN_A3);
output_high(PIN_A2);
output_low(PIN_A1);
output_low(PIN_A5);
delay_ms(100);
}

void se_a(){
output_high(PIN_A5);
output_high(PIN_A3);
output_low(PIN_A2);
output_low(PIN_A1);
delay_ms(100);
}

void ctrol1(){
if(b>=10&&selcitem==1){
delay_ms(400);
b=b-10;
selcitem=0;
differ = true;
buffer();
ctrol1++;
}
else if(ctrol1==0){
se_d();
}
}

```



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

else if(ctro1==1){
  ne_d();
}
else if(ctro1>1){ctro1=0;}
}
void ctrol2(){
if(b>=10&&selcitem==2){
  delay_ms(400);
  b=b-10;
  selcitem=0;
  differ2 = true ;
  buffer();
  ++ctro2;
}
else if(ctro2==0){
  nw_a();
}
else if(ctro2==1){
  ne_a();
}
else if(ctro2>1)ctro2=0;
}
void chkp(float l){
  if(l>0.5f){
    b++;
    delay_ms(110);
  }
}
void check_control(){

```



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

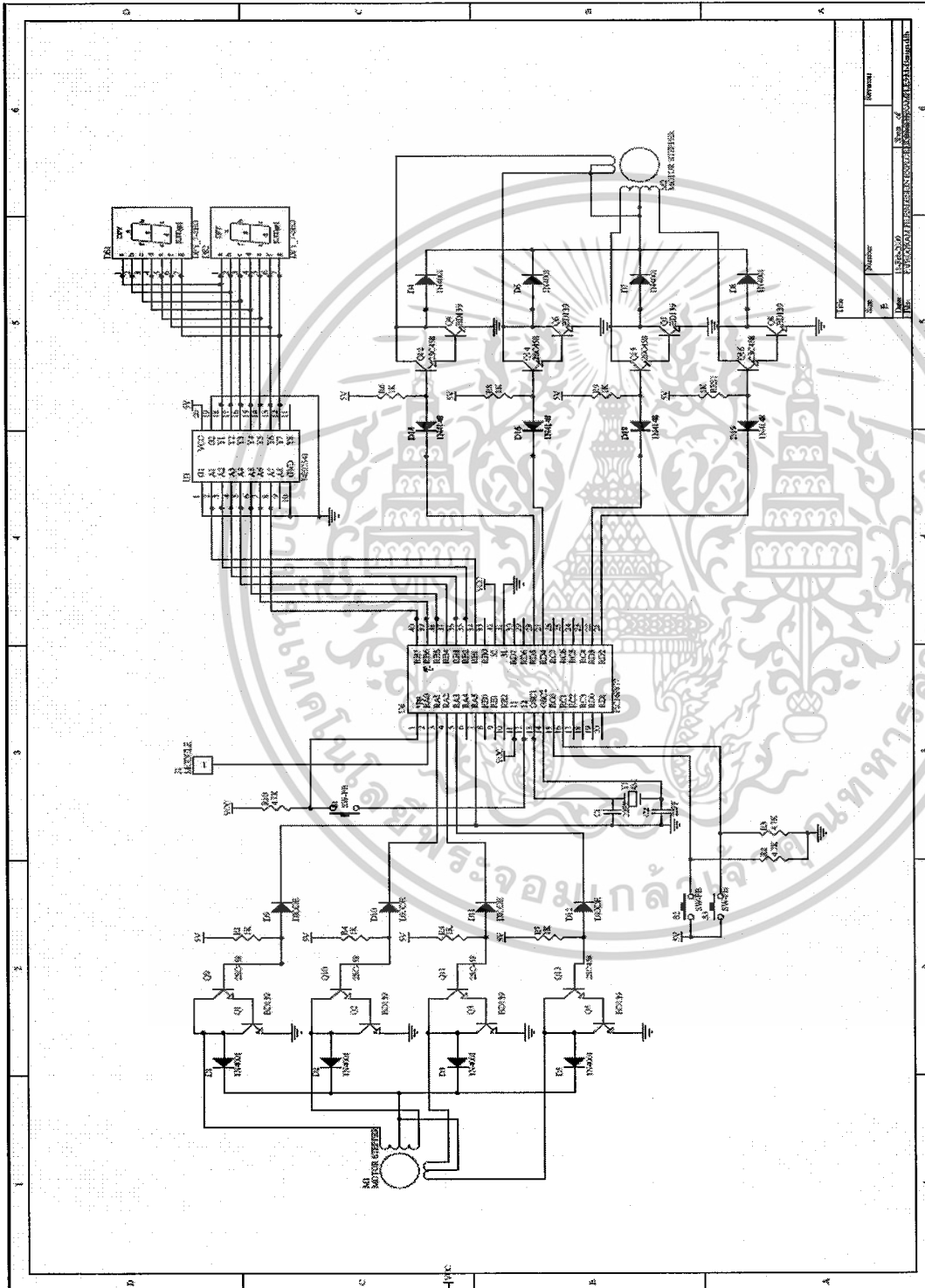
```
delay_us(100);  
if(input(PIN_C0))    // SELECT ITEM 1  
  selcitem=1;  
else if(input(PIN_C1))  // SELECT ITEM 2  
  selcitem=2;  
delay_us(100);  
}
```



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก ก

วงจรรวมของ เครื่องหยอดเหรียญอัตโนมัติ



รูปที่ ก.1 วงจร เครื่องหยอดเหรียญอัตโนมัติ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

