

สำนักหอสมุดกลาง พระจอมเกล้าลาดกระบัง

ระบบการจองตั๋วคอนเสิร์ตผ่านโทรศัพท์เคลื่อนที่และพิสูจน์ตัวตนเพื่อการออกตั๋ว
CONCERT TICKET RESERVATION SYSTEM
VIA MOBILE PHONE AND OWNER VERIFICATION

โดย

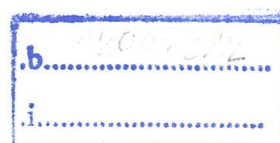
นาย พงศกร คำนวนผล

นาย ภัทรพงษ์ แสงสุข



ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต
สาขาวิชาวิศวกรรมโทรคมนาคม
คณะวิศวกรรมศาสตร์
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
ปีการศึกษา 2551

เลขหมู่.....103006.....
เลขทะเบียน.....
วัน,เดือน,ปี..... 24 ส.ค. 2552



ระบบการจองตั๋วคอนเสิร์ตผ่านโทรศัพท์เคลื่อนที่และพิสูจน์ตัวตนเพื่อการออกตั๋ว

**CONCERT TICKET RESERVATION SYSTEM
VIA MOBILE PHONE AND OWNER VERIFICATION**

โดย

นาย พงศกร คำนวนผล

นาย ภัทรพงษ์ แสงสุข

อาจารย์ที่ปรึกษา

รศ.ดร. ไกรสิน ส่วงวัฒนา

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต

สาขาวิชาวิศวกรรมโทรคมนาคม

คณะวิศวกรรมศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2551

ปริญญาโทปีการศึกษา 2551

ภาควิชาวิศวกรรมโทรคมนาคม

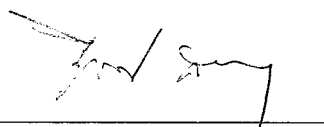
คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เรื่อง ระบบการจองตั๋วคอนเสิร์ตผ่านโทรศัพท์เคลื่อนที่และพิสูจน์ตัวตนเพื่อการออกตั๋ว

**CONCERT TICKET RESERVATION SYSTEM VIA MOBILE PHONE
AND OWNER VERIFICATION**

ผู้จัดทำ

1. นาย พงศกร กำนวนผล 48010555
2. นาย ภัทรพงษ์ แสงสุข 48010657



อาจารย์ที่ปรึกษา

(รศ.ดร. ไกรสิน ส่วงวัฒนา)

โครงการงานเลขที่ 512417

ระบบการจองตั๋วคอนเสิร์ตผ่านโทรศัพท์เคลื่อนที่
และพิสูจน์ตัวตนเพื่อการออกตั๋ว

**CONCERT TICKET RESERVATION SYSTEM
VIA MOBILE PHONE AND OWNER VERIFICATION**

โดย นาย พงศกร คำนวนผล 48010555

นาย กัทรพงษ์ แสงสุข 48010657

อาจารย์ที่ปรึกษา รศ.ดร. ไกรสิน ส่วงวัฒนา

บทคัดย่อ

ปริญญานิพนธ์นี้นำเสนอการออกแบบระบบการจองตั๋วคอนเสิร์ตผ่านโทรศัพท์เคลื่อนที่ ที่ซึ่งมีอาร์เอฟไอดีแท็กติดตั้งอยู่บนตัวโทรศัพท์เคลื่อนที่ และเป็นระบบพิสูจน์ตัวตนเป็นผู้จองตั๋วเพื่อการออกตั๋วโดยอัตโนมัติ เพื่อเพิ่มความสะดวกรสบายให้กับผู้ใช้บริการ โดยที่ไม่ต้องรอเข้าแถวเพื่อจะซื้อตั๋ว ระบบทั้งหมดแบ่งออกเป็น 3 ส่วน คือ (1) การสร้างเว็บแอปพลิเคชันเพื่อใช้ในการติดต่อกับผู้ใช้บริการผ่านทางเว็บเบราว์เซอร์ (2) การออกแบบระบบเพื่อการติดต่อกับฐานข้อมูลของผู้ให้บริการ (3) การออกแบบระบบพิสูจน์ตัวตนผ่านระบบอาร์เอฟไอดีโดยโทรศัพท์เคลื่อนที่

Abstract

This project presents the design of the concert reservation system via mobile phone with Radio Frequency Identification (RFID). The RFID tag is using to authenticate ticket owners and dispense tickets automatically in order to increase convenience for users. The systems are consisted of 3 parts 1. Develop website application to connect with users via mobile phone's web browser. 2. Design system for connecting to database. 3. Design verification system for the concert ticket dispensing automatically.

สารบัญ

เรื่อง	หน้า
บทคัดย่อ	
บทที่ 1. บทนำ	1
บทที่ 2. ทฤษฎีและหลักการ	2
2.1 RFID	2
2.2 Carrier ในระบบ RFID	9
2.3 BARCODE กับ RFID BARCODE หรือ รหัสแท่ง	20
2.4 มาตรฐานพอร์ตอนุกรมแบบ RS-232	23
2.5 รูปแบบการทำงาน (หลักการ) ของ Reader	26
2.6 ภาษาคอมพิวเตอร์	32
2.7 เว็บแอปพลิเคชัน (Web Application)	35
2.8 ASP.NET	36
2.9 Object Oriented Programming (OOP) หรือ การออกแบบและพัฒนาโปรแกรมเชิงวัตถุ	39
2.10 Unified Modeling Language (UML)	41
บทที่ 3 การออกแบบและการสร้าง	43
3.1 การออกแบบและการสร้างระบบการจองตั๋วผ่านเว็บบนโทรศัพท์เคลื่อนที่	44
3.2 การออกแบบและการสร้างระบบการจ่ายเงินผ่านระบบ RFID	45
3.3 การออกแบบและการสร้างเว็บเซิร์ฟเวอร์	48
3.4 การออกแบบและการสร้างฐานข้อมูลโดยใช้ Microsoft SQL Server 2005	52
3.5 การออกแบบและการสร้างเว็บเซอร์วิสโดยใช้ ASP.NET	53
3.6 การออกแบบวงจรควบคุมไฟ LED ด้วยไมโครคอนโทรลเลอร์	54
3.7 การออกแบบวงจรควบคุมไฟ LED ด้วยไมโครคอนโทรลเลอร์	55
บทที่ 4 ผลการทดลอง และวิเคราะห์ผลการทดลอง	57
4.1 ผลการทดลองระบบการจองตั๋วผ่านเว็บเบราว์เซอร์บนโทรศัพท์เคลื่อนที่	57
4.2 ผลการทดลองแคาน์เตอร์เซอร์วิส	60
4.3 ผลการทดลองทางเข้าคอนเสิร์ต	64
4.4 ผลการทดลองวัดสเปคตรัมของเครื่อง RFID reader/writer	67
บทที่ 5 สรุปผลและวิจารณ์การทดลอง	69
ภาคผนวก	

สารบัญรูปภาพ

	หน้า
รูปที่ 2.1 ภาพการทำงานของระบบ RFID	3
รูปที่ 2.2 ระบบการอ่าน / เขียนข้อมูลอย่างง่ายของ RFID	4
รูปที่ 2.3 แสดงตัวอย่างการใช้งานแท็กและตัวอ่านข้อมูล (Reader)	5
รูปที่ 2.4 แสดง RFID tag ในรูปแบบต่างๆ	6
รูปที่ 2.5 แสดงบล็อกไดอะแกรมของ Passive tag	6
รูปที่ 2.6 แสดงรูปตัวอย่าง Active tag ที่มีแบตเตอรี่ lithium, 2 ก้อนอยู่ภายนอก	7
รูปที่ 2.7 แสดงโครงสร้างภายในตัวอ่าน	7
รูปที่ 2.8 แสดงส่วนประกอบของเครื่องอ่าน RFID	8
รูปที่ 2.9 ตัวอย่างรูปร่างของ Reader ต่างๆ	9
รูปที่ 2.10 แสดงความถี่ที่ระบบ RFID ถูกใช้งาน	9
รูปที่ 2.11 แสดงแผนผังการทำงานของระบบ RFID	10
รูปที่ 2.12 แสดงการการส่งข้อมูลระหว่างตัวอ่านและ Tag รูปแบบต่างๆ	13
รูปที่ 2.13 การสื่อสารระหว่าง Reader กับ Tag	13
รูปที่ 2.14 แสดงการสร้าง Power supply ให้กับ tag โดย reader	14
รูปที่ 2.15 การ Generation of Load Modulation	15
รูปที่ 2.16 วงจร Generation of Load Modulation with Sub Carrier	15
รูปที่ 2.17 การเข้ารหัสข้อมูลในรูปแบบต่างๆ	16
รูปที่ 2.18 การผสมข้อมูลโดยอาศัยการเปลี่ยนแปลงของยอดคลื่นพาหะ	17
รูปที่ 2.19 การผสมข้อมูลโดยอาศัยการเปลี่ยนแปลงความถี่	18
รูปที่ 2.20 การผสมข้อมูลโดยอาศัยการเปลี่ยนแปลงเฟส	18
รูปที่ 2.21 การ Modulation แบบ ASK Sub carrier	19
รูปที่ 2.22 รูปแบบของ Barcode ทั่วไป	21
รูปที่ 2.23 ระบบ RFID	22
รูปที่ 2.24 แสดงการจัดขาของคอนเน็คเตอร์พอร์ตอนุกรมตามมาตรฐาน RS-232	24
รูปที่ 2.25 แสดงการต่ออุปกรณ์ภายนอกกับพอร์ตอนุกรมของคอมพิวเตอร์ในลักษณะต่างๆ	25
รูปที่ 2.26 Master – Slave ระหว่าง Application Software, เครื่องอ่านและ Tag	26
รูปที่ 2.27 Block Diagram ของเครื่องอ่านที่ประกอบด้วยส่วนควบคุมและ HF Interface	27
รูปที่ 2.28 Block Diagram ของ HF Interface สำหรับ Inductive Couple	28
รูปที่ 2.29 Block Diagram ของ Control Unit ของ Reader และสื่อสารกับ Application Software	29
รูปที่ 2.30 สัญญาณ Coding และ Decoding ถูกกระทำโดย Control Unit ของ Reader	30
รูปที่ 2.31 Block Diagram ของ RFID Data Carrier ของ Memory Function	30

สารบัญรูปภาพ (ต่อ)

	หน้า
รูปที่ 2.32 Block Diagram ของ HF Interface ของ Inductive Coupled	31
รูปที่ 2.33 การสร้าง Load Modulation กับ Sub carrier Modulated	32
รูปที่ 2.34 การติดต่อกับฐานข้อมูลโดยใช้ซีเควล (SQL)	34
รูปที่ 2.35 โครงสร้างสถาปัตยกรรมของ .NET	36
รูปที่ 3.1 แสดงมุมมองทั้งหมดของผู้ใช้บริการ	43
รูปที่ 3.2 บล็อกไดอะแกรมการทำงานของโปรแกรมเพื่อการจองตั๋ว	44
รูปที่ 3.3 บล็อกไดอะแกรมการทำงานของโปรแกรมเพื่อการจ่ายเงินผ่านระบบ RFID	45
รูปที่ 3.4 บล็อกไดอะแกรมการทำงานของระบบ RFID มาตรฐาน Mifare	46
รูปที่ 3.5 Memory ของ RFID Tag มาตรฐาน Mifare ขนาด 1024 X 8 บิต	47
รูปที่ 3.6 บล็อกไดอะแกรมบริการเติมเงินให้กับผู้ใช้บริการ	48
รูปที่ 3.7 การออกแบบฐานข้อมูลบนเว็บเซิร์ฟเวอร์	52
รูปที่ 3.8 เว็บเซอร์วิสเพื่อใช้บริการในชั้นตอนต่าง ๆ	53
รูปที่ 3.9 บล็อกไดอะแกรมในส่วนของการแสดงตัวตนเพื่อเข้างาน	54
รูปที่ 3.10 วงจรรวมจำลองเปิด-ปิดประตูโดยใช้ LED	55
รูปที่ 3.11 โมดูล GSM/GPRS รุ่น 300SIMCZ	55
รูปที่ 3.12 เซ็ตบิตเรทติดต่อกับ โมดูลผ่าน HyperTerminal	56
รูปที่ 3.13 แสดงคำสั่ง AT command ใช้ติดต่อกับ โมดูล เพื่อส่ง SMS	56
รูปที่ 4.1 แสดงเว็บเซอร์วิส IIS	57
รูปที่ 4.2 แสดงผลการกรอก URL เพื่อเข้าเว็บการจองตั๋วคอนเสิร์ต	58
รูปที่ 4.3 แสดงหน้าเว็บบนโทรศัพท์เคลื่อนที่เพื่อการจองตั๋ว	59
รูปที่ 4.4 แสดงฐานข้อมูลเมื่อยังไม่ได้ทำการจองตั๋ว	59
รูปที่ 4.5 แสดงฐานข้อมูลเมื่อทำการจองตั๋วคอนเสิร์ต	59
รูปที่ 4.6 แสดงการออกแบบ Window Applications	60
รูปที่ 4.7 แสดงหน้าต่างค่าเงินใน Touch SIM เป็นเงิน 0 บาท	61
รูปที่ 4.8 แสดงหน้าต่างเมื่อทำการเพิ่มเงินในราคา 5,000 บาท	61
รูปที่ 4.9 แสดงหน้าต่างเมื่อทำการอ่านค่าเงินจาก Touch SIM หลังจากเติมเงินเรียบร้อยแล้ว	62
รูปที่ 4.10 แสดงหน้าต่างต้อนรับผู้ใช้บริการเพื่อให้ผู้ใช้บริการ Touch SIM	62
รูปที่ 4.11 แสดงรายละเอียดของผู้ใช้บริการที่ทำการจองคอนเสิร์ตไว้	63
รูปที่ 4.12 แสดงยอดเงินคงเหลือหลังทำการหักเงินตามยอดการจองตั๋วคอนเสิร์ต	63
รูปที่ 4.13 แสดงการออกแบบ Window Applications ในส่วนหน้าคอนเสิร์ต	64
รูปที่ 4.14 ส่วนประกอบรวมในภาคหน้าคอนเสิร์ต	65

สารบัญรูปภาพ (ต่อ)

	หน้า
รูปที่ 4.15 แสดง LED เมื่อเริ่มทำงาน	65
รูปที่ 4.16 แสดง LED เมื่อสมมุติให้ประตูเปิด	66
รูปที่ 4.17 แสดงหลอด LED เมื่อไม่ได้อนุญาตให้เข้าประตูได้	66
รูปที่ 4.18 แสดง SMS บอกโซนที่นิ่งตามราคา	67
รูปที่ 4.19 แสดงผลการวัดสเปกขณะเปิดเครื่อง RFID reader/writer	67
รูปที่ 4.20 แสดงสเปกตรัมเมื่อทำการส่ง-รับข้อมูล ของ RFID reader/writer	68

สารบัญตาราง

	หน้า
ตารางที่ 2.1 กระบวนการส่งข้อมูลของ RFID	19
ตารางที่ 2.2 เปรียบเทียบระบบ Barcode กับระบบ RFID	22
ตารางที่ 2.3 ตัวอย่างการดำเนินการอ่านข้อมูลด้วย Application Software เครื่องอ่านและ Tag	27
ตารางที่ 2.4 ความถี่ Clock ที่ต้องการใน HF Interface ถูกสร้าง โดย Binary ของสัญญาณ Carrier 13.56 MHz	32

บทที่ 1

บทนำ

1.1 ที่มาและความสำคัญ

ปัจจุบันเทคโนโลยีการติดต่อสื่อสารได้เข้ามามีบทบาทต่อชีวิตประจำวันมากขึ้น จะเห็นได้ว่ามีธุรกิจหลายรูปแบบที่มุ่งเน้น และมีความพยายามที่จะจัดหาอุปกรณ์และบริการเพื่อตอบสนองต่อความต้องการ พร้อมทั้งอำนวยความสะดวกให้กับผู้ใช้ได้ทันที่ ซึ่งการให้บริการลูกค้าผ่านทางโทรศัพท์เคลื่อนที่เป็นการบริการอีกรูปแบบหนึ่งที่เป็นที่นิยม ดังนั้นปริญญาานิพนธ์นี้จึงทำการศึกษาและออกแบบระบบการจองตั๋วคอนเสิร์ตผ่านทางโทรศัพท์เคลื่อนที่ ที่ใช้ร่วมกับเทคโนโลยี RFID เพื่อนำมาแทนเงินสดในการชำระค่าบริการ ซึ่งเป็นแนวทางนำไปประยุกต์ใช้ในการทำธุรกิจต่างๆ เพื่อให้ผู้ใช้บริการสามารถเข้าถึงข้อมูลและบริการต่างๆ ได้อย่างรวดเร็วยิ่งขึ้น

1.2 วัตถุประสงค์ของปริญญาานิพนธ์

1. เพื่อศึกษาและสร้างระบบจำลองของการจองตั๋วคอนเสิร์ตผ่านเว็บบนโทรศัพท์เคลื่อนที่
2. เพื่อศึกษาและสร้างระบบจำลองของการจ่ายเงินผ่านระบบ RFID
3. ศึกษาเรื่องเว็บเซอร์วิส โดยใช้ .NET Technology

1.3 ขอบเขตของโครงการ

1. ออกแบบและสร้างเว็บแอปพลิเคชันเพื่อใช้งานผ่านโทรศัพท์เคลื่อนที่
2. ออกแบบและสร้างเว็บเซอร์วิส ร่วมกับการใช้ระบบ RFID

1.4 วิธีการดำเนินการของโครงการ

1. ศึกษาทฤษฎีของ XML SOAP และส่วนประกอบในเว็บเซอร์วิส
2. ศึกษาการพัฒนาคอมโพเนนต์และเว็บเซอร์วิสใน .NET
3. ออกแบบระบบเว็บแอปพลิเคชันผ่านเว็บบนโทรศัพท์เคลื่อนที่
4. ออกแบบระบบ RFID เพื่อติดต่อกับเว็บเซอร์วิส

บทที่ 2

ทฤษฎีและหลักการ

2.1 RFID

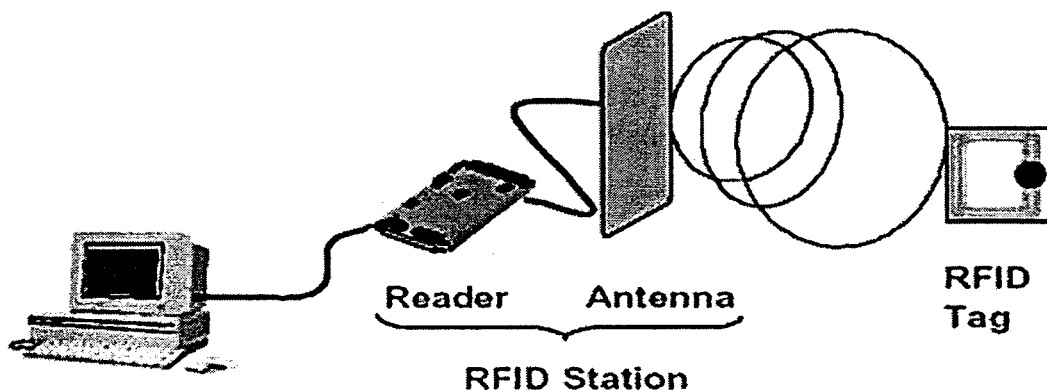
คือระบบที่เฉพาะอัตโนมัติ (Automatic Identification) แบบไร้สาย (Wireless) เช่นเดียวกับระบบ Barcode แต่ใช้ความถี่วิทยุในการติดต่อสื่อสาร ระบบนี้จะประกอบด้วยอุปกรณ์สองส่วน คือ ส่วนเครื่องอ่าน (Reader) และส่วนป้ายชื่อ (Tag) โดยการทำงานนั้นเครื่องอ่านจะทำหน้าที่จ่ายกำลังงานในรูปแบบคลื่นความถี่วิทยุให้กับ Tag ยังผลให้วงจรอิเล็กทรอนิกส์ภายในสามารถส่งข้อมูลจำเพาะที่แสดงถึง “Identity” กลับมาประมวลผลที่ตัวอ่านได้ ข้อดีของระบบ RFID คือเป็นระบบไร้สายจึงไม่จำเป็นต้องนำ Tag ไปสัมผัสกับเครื่องอ่านซึ่งช่วยเพิ่มความสะดวกและเป็นอัตโนมัติให้กับระบบที่เฉพาะ โดยระบบ RFID สามารถนำไปประยุกต์ใช้งานได้อย่างกว้างขวาง เช่น การใช้บอกรหัสสินค้า การใช้บอกรหัสประจำตัวพนักงาน หรือ การใช้งานควบคุมการเข้าออก เป็นต้น

2.1.1 ความหมายของ RFID (Radio frequency Identification System)

RFID จะมีลักษณะใกล้เคียงกับสมาร์ทการ์ดที่สุดคือข้อมูลจะถูกเก็บไว้ในการ์ดหรือ Tags เหมือนกัน แต่ข้อแตกต่างที่เห็นได้ชัดคือการอ่าน/เขียน ข้อมูลสามารถทำได้โดยไม่ต้องสัมผัส เหมือนกับสมาร์ทการ์ด ซึ่งจะอาศัยคลื่นแม่เหล็กไฟฟ้าในช่วงความถี่วิทยุ (Radio Frequency) ในการติดต่อสื่อสารจึงเป็นสาเหตุให้ RFID ได้เปรียบระบบ Auto ID ทุกชนิดที่กล่าวมาแล้ว และมีแนวโน้มจะถูกใช้มากขึ้นเรื่อยๆ **Radio frequency Identification (RFID)** RFID ย่อมาจาก Radio Frequency Identification เป็นระบบระบุเอกลักษณ์ของวัตถุ ด้วยคลื่นความถี่วิทยุ ที่ได้ถูกพัฒนามาตั้งแต่ปี ค.ศ. 1980 เพื่อวัตถุประสงค์หลักเพื่อนำไปใช้งานแทนระบบบาร์โค้ด (Barcode) โดยจุดเด่นของ RFID ก็อยู่ตรงการอ่านข้อมูลจาก Tag ได้หลาย ๆ Tag แบบไร้สัมผัส และสามารถอ่านค่าได้แม้ในสภาพที่ทัศนวิสัย ทนต่อความเปียกชื้นแรงสั่นสะเทือน การกระทบกระแทกและสามารถจะอ่านข้อมูลได้ด้วยความเร็วสูง โดยข้อมูลจะถูกเก็บไว้ในไมโครชิพที่อยู่ใน Tag

Radio Frequency Identification (RFID) ประกอบด้วย

1. Label หรือ RFID Tag คือแผงวงจรรีวิทยุขนาดเล็กบรรจุข้อมูลความจำ (Memory chip) บนแผ่นกระดาษขนาด 2 ตารางนิ้ว
2. Reader เครื่องอ่านสัญญาณจะสร้างคลื่นวิทยุขึ้นแล้วส่งไปยังสายอากาศ
3. Antenna จะเชื่อมต่อกับ Reader เพื่อส่งคลื่นวิทยุที่กำเนิดจากตัว Reader ไปยัง Tag เพื่อกระตุ้นให้ Tag ส่งข้อมูลกลับมาให้ตัว Reader



รูปที่ 2.1 ภาพการทำงานของระบบ RFID

2.1.2 ส่วนประกอบของระบบ RFID

โดยทั่วไประบบ RFID จะประกอบไปด้วย 2 ส่วนสำคัญคือ

ฉลากอิเล็กทรอนิกส์ (Transponder) หรือ Data Carrier หรือเรียกง่ายๆ ว่า Tags จะเป็นส่วนของการเก็บข้อมูลและถูกติดตั้งอยู่กับวัตถุที่เราต้องการบ่งชี้ (Identify)

Tags จะประกอบไปด้วยเสาอากาศทำหน้าที่คล่องสัญญาณที่ส่งมาจากเครื่องอ่านและส่วนขงไมโครชิพ ในกรณีที่ Tags ไม่มีแบตเตอรี่ในตัวอยู่นอกพื้นที่ที่มีสัญญาณจะไม่มีการทำงานเกิดขึ้น Tags จะทำงานก็ต่อเมื่อ Tags เข้ามาในพื้นที่ที่มีสัญญาณซึ่ง Tags จะได้รับพลังงานจากการคล่องของสัญญาณสนามแม่เหล็กไฟฟ้าและสร้างแรงดันไฟฟ้าขึ้นจำนวนหนึ่ง ปริมาณเพียงพอที่จะใช้ในการทำงานของ Tags

เครื่องอ่าน (Reader) จะประกอบไปด้วยภาครับ/ส่งสัญญาณวิทยุ ส่วนควบคุมและเสาอากาศ (Antenna) ทำหน้าที่คล่องสัญญาณกับ Tags และส่วนของการติดต่อสื่อสารกับอุปกรณ์ควบคุมภายนอก (RS232C, RS485 หรือ RS422) ขึ้นอยู่กับลักษณะการสื่อสารที่เราต้องการ

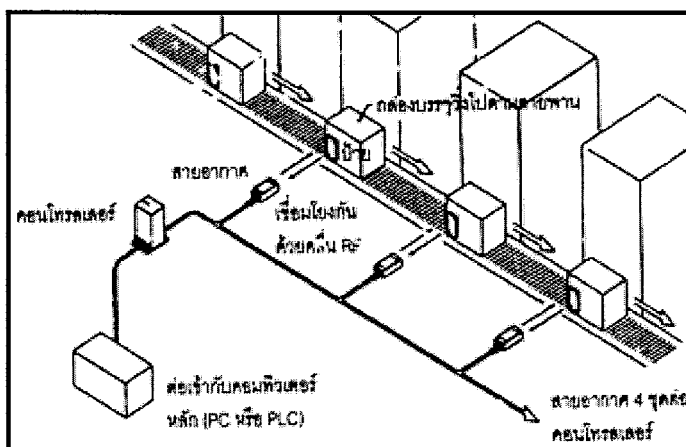
Tag หรือ Transponder

แท็ก (Tag) นั้นเรียกอีกชื่อหนึ่งว่าทรานสปอนเดอร์ (Transponder) มาจากคำว่าทรานสมิตเตอร์ (Transmitter) ผสมกับคำว่าเรสปอนเดอร์ (Responder) ถ้าจะแปลให้ตรงตามศัพท์ แท็กก็จะทำหน้าที่ส่งสัญญาณหรือข้อมูลที่บันทึกอยู่ในแท็กตอบสนองไปที่ตัวอ่านข้อมูล การสื่อสารระหว่างแท็กและตัวอ่านข้อมูลจะเป็นการสื่อสารกันโดยอาศัยช่องความถี่วิทยุผ่านอากาศ โครงสร้างภายในแท็กจะประกอบไปด้วย 2 ส่วนใหญ่ๆ ได้แก่ ส่วนของไอซีซึ่งเป็นชิพสารกึ่งตัวนำ (Semiconductor Chip) และส่วนของขดลวดซึ่งทำหน้าที่เป็นเสาอากาศสำหรับรับส่งข้อมูล โดยทั้งสองส่วนนี้จะเชื่อมต่ออยู่ด้วยกัน ไอซีของแท็กที่มีการผลิตออกมาจะมีทั้งขนาดและรูปร่างเป็นได้ตั้งแต่แท่งหรือแผ่นขนาดเล็กจนแทบไม่สามารถมองเห็น หรือไปจนถึงขนาดใหญ่จะสะดุดตา ซึ่งต่างก็มีความเหมาะสมกับชนิดงานที่แตกต่างกัน โดยทั่วไปโครงสร้างภายในส่วนที่เป็นไอซีของแท็กนั้นก็จะประกอบด้วย 2 ส่วนหลัก ๆ ได้แก่

1. ส่วนของการควบคุมภาครับส่งสัญญาณวิทยุ สำหรับโครงสร้างของส่วนนี้ประกอบด้วยภาคดีมอดูเลตและภาคมอดูเลต (สำหรับรับส่งข้อมูลระหว่างแท็กกับตัวเครื่องอ่าน) และวงจรกำเนิดไฟฟ้าขนาดเล็ก

2. ส่วนของการควบคุมภาคดิจิทัล ซึ่งรับหน้าที่จัดการเกี่ยวกับกระบวนการทางดิจิทัลทั้งหมด โครงสร้างหลักๆ ของส่วนการทำงานนี้ประกอบด้วย ส่วนบันทึกข้อมูล (ประกอบด้วยหน่วยความจำแรม (RAM), รม (ROM), อีอีพรอม (EEPROM) ส่วนของการเข้ารหัส (Crypts Unit) ส่วนตอบรับสัญญาณร้องขอ (Answer to Request) ส่วนควบคุมและประมวลผลทางคณิตศาสตร์ (Control & Arithmetic Unit) อย่างไรก็ตาม โครงสร้างภายในของแท็กที่ต่างผู้ผลิตหรือต่างรุ่นกัน บางครั้งก็อาจมีไม่ครบถ้วนทุกส่วนอย่างที่ได้ออกมา ซึ่งรายละเอียดโครงสร้างตลอดจนรายละเอียดในการทำงานของแท็กเบอร์ใดๆ ก็สามารถดูได้จาก Data Sheet ของบริษัทผู้ผลิตแท็กเบอร์นั้นๆ

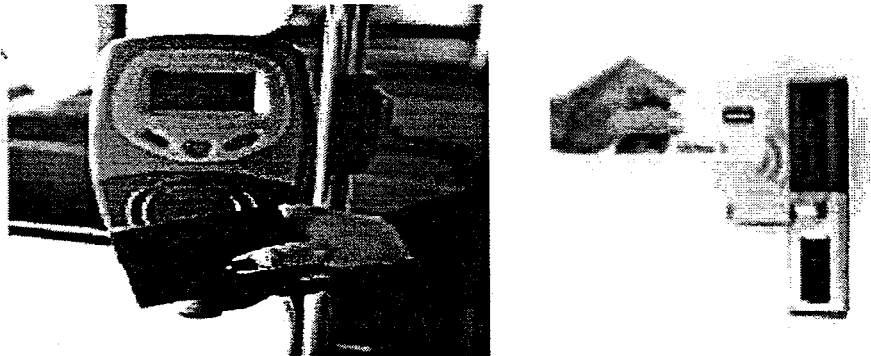
Transponder หรือ Tags มีลักษณะเป็นไมโครชิพ (microchip) ที่ยอมให้ผู้ใช้ติดเข้าระหว่างชั้นของกระดาษหรือพลาสติกที่ใช้ทำป้ายฉลาก ชิพหรือแท็กอาจมีรูปร่างได้หลายแบบขึ้นอยู่กับการนำไปใช้งาน โดยอาจมีรูปร่างเหมือนบัตรเครดิตในการใช้งานทั่วไป หรือเล็กขนาดใส่ดินสอยากเพียง 10 มิลลิเมตร เพื่อฝังเข้าไปใต้ผิวหนังสัตว์ในกรณีนำไปใช้ในงานปศุสัตว์ หรืออาจมีขนาดใหญ่มากสำหรับแท็กที่ใช้ติดกับเครื่องจักรขณะทำการขนส่ง แท็กอาจนำไปติดไว้กับสินค้าในร้านปลีกทั่วไปเพื่อป้องกันขโมย โดยจะมีการติดตั้งสายอากาศของตัวอ่านข้อมูลขนาดใหญ่ไว้ตรงประตูทางออกเพื่อทำการตรวจจับขโมย โดยแท็กจะรับพลังงานจากสัญญาณ RF เพื่อติดต่อสื่อสารกับเครื่องอ่าน หรือใช้พลังงานจากแบตเตอรี่ที่บรรจุภายในป้าย ซึ่งเป็นแบตเตอรี่ Lithium-Ion มีอายุการใช้งานที่ยาวนาน จึงมักนำมาใช้กับแผ่นป้ายนี้



รูปที่ 2.2 ระบบการอ่าน / เขียนข้อมูลอย่างง่ายของ RFID

แท็กประกอบไปด้วยสายอากาศที่มีขนาดเล็กที่จะช่วยให้แท็กตอบสนองกับเครื่องอ่าน โดยสายอากาศจะแผ่สัญญาณวิทยุจำนวนหนึ่งออกมา เพื่อกระตุ้นให้แท็กอ่านหรือเขียนข้อมูลลงไปยังสายอากาศสามารถทำได้ทุกขนาดและรูปร่าง เพื่อที่จะสามารถออกแบบให้ติดตั้งได้ทุกที่ และเพื่อให้เกิดความครอบคลุมได้ดีที่สุดในหลายๆ ระบบสายอากาศจะถูกติดไปโดยตรงกับ Transceiver เหมือนกับเป็น

อุปกรณ์ติดกัน ชิฟที่อยู่ในแท็กจะมีหน่วยความจำซึ่งอาจเป็นแบบอ่านได้อย่างเดียว (ROM) หรือทั้งอ่านทั้งเขียน (RAM) ทั้งนี้ขึ้นอยู่กับความต้องการในการใช้งาน โดยปกติหน่วยความจำแบบ ROM จะใช้เก็บข้อมูลเกี่ยวกับการรักษาความปลอดภัย เช่นข้อมูลของบุคคลที่มีสิทธิผ่านเข้าออกในบริเวณที่มีการควบคุมหรือระบบปฏิบัติการ ในขณะที่ RAM จะใช้เก็บข้อมูลชั่วคราวในระหว่างแท็กและตัวอ่านข้อมูลทำการติดต่อสื่อสารกัน นอกจากนี้อาจมีการนำหน่วยความจำแบบ EEPROM มาใช้ในกรณีที่ต้องการเก็บข้อมูลในระหว่างที่แท็กและตัวอ่านข้อมูลทำการสื่อสาร และข้อมูลยังคงอยู่ถึงแม้จะไม่มีพลังงานไฟฟ้าป้อนให้แก่แท็ก



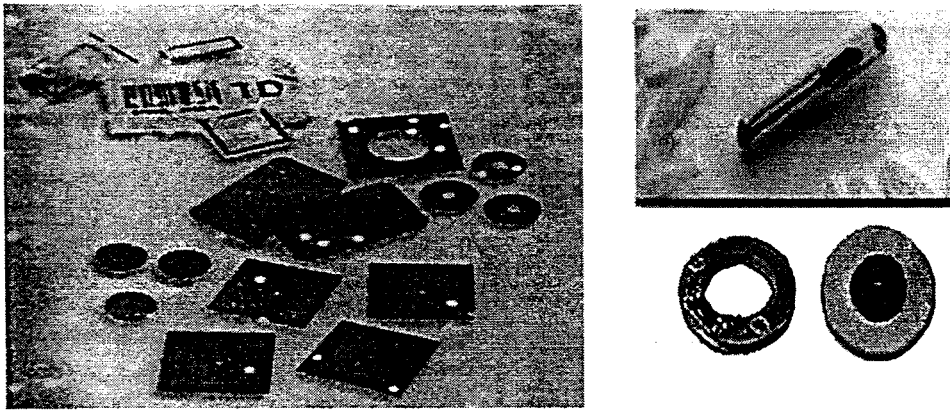
รูปที่ 2.3 แสดงตัวอย่างการใช้งานแท็กและตัวอ่านข้อมูล (Reader)

นอกจากการแท็กแบ่งจากชนิดที่ว่ามาแล้วแท็กก็ยังคงถูกแบ่งประเภทจากรูปแบบในการใช้งานได้เป็น 3 แบบ คือ

1. แบบที่สามารถถูกอ่านและเขียนข้อมูลได้อย่างอิสระ (Read – write)
2. แบบเขียนได้เพียงครั้งเดียวเท่านั้นแต่อ่านได้อย่างอิสระ (Write-One, Read-Many หรือ WORM)
3. แบบอ่านได้เพียงอย่างเดียว (Read-Only)

2.1.3 รูปแบบของ Tag

โครงสร้างภายในของแท็กจะประกอบด้วย 2 ส่วนใหญ่ๆ ได้แก่ ขดลวดขนาดเล็กซึ่งทำหน้าที่เป็นสายอากาศ (Antenna) สำหรับรับส่งสัญญาณคลื่นความถี่วิทยุและสร้างพลังงานป้อนให้ส่วนของไมโครชิพ (Microchip) ที่ทำหน้าที่เก็บข้อมูลของวัตถุ เช่น รหัสสินค้า โดยทั่วไปตัวแท็กอาจอยู่ในชนิดทั้งเป็นกระดาษ แผ่นฟิล์ม พลาสติก มีขนาดและรูปร่างต่างๆ กันไป ทั้งนี้ ขึ้นอยู่กับวัสดุที่จะนำไปติด และมีหลายรูปแบบ เช่น ขนาดเท่ากับบัตรเครดิต เหรียญ กระดุม ฉลากสินค้า แคปซูล เป็นต้น (พิจารณารูปที่ 2.4) แต่โดยหลักการอาจแบ่ง Tag ที่มีใช้งานกันอยู่นั้นจะมีอยู่ 2 ชนิดใหญ่ๆ โดยแต่ละชนิดก็จะแตกต่างกันในแง่ของการใช้งาน ราคาโครงสร้างและหลักการทำงานอยู่ ซึ่งจะกล่าวถึงและอธิบายเป็นหัวข้อดังนี้



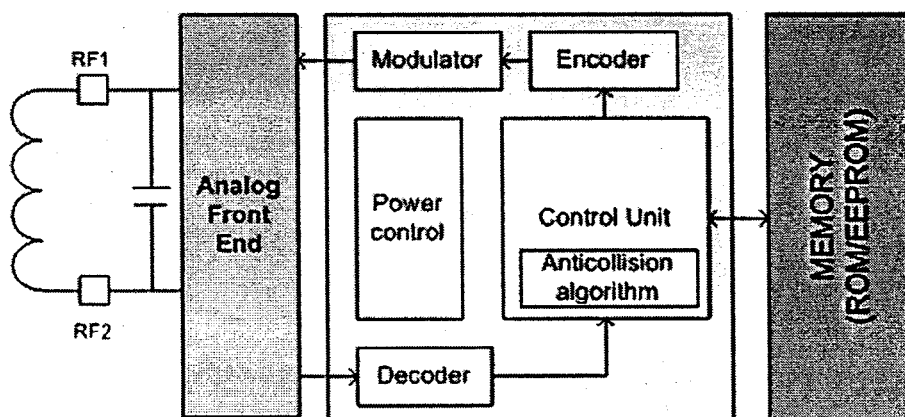
รูปที่ 2.4 แสดง RFID tag ในรูปแบบต่างๆ

2.1.4 ชนิดของ Tag

- Passive tag
- Active tag

Passive RFID Tags

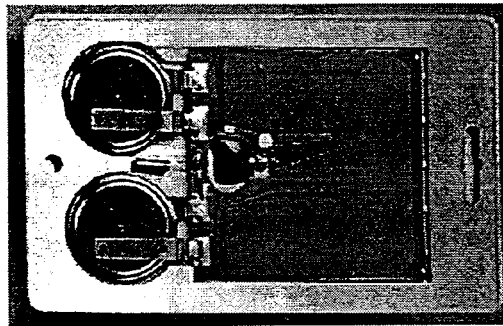
Tag ชนิดนี้ไม่ต้องอาศัยแหล่งจ่ายไฟภายนอกใดๆ เพราะภายใน Tag จะมีวงจรถูกกำเนิดไฟฟ้าเหนี่ยวนำขนาดเล็กเป็นแหล่งจ่ายไฟในตัวทำให้การอ่านข้อมูลทำได้ไม่ไกลมากนัก ระยะอ่านสูงสุดประมาณ 1 เมตรขึ้นอยู่กับความแรงของเครื่องส่งและคลื่นความถี่วิทยุที่ใช้ โดยปกติ tag ชนิดนี้มักมีหน่วยความจำขนาดเล็ก โดยทั่วไปประมาณ 16 ถึง 1,024 ไบต์ มีขนาดเล็กและน้ำหนักเบา ราคาต่อหน่วยต่ำไอซีของ tag ชนิด passive ที่มีการผลิตออกมาจะมีทั้งขนาดและรูปร่างเป็นได้ตั้งแต่แท่งหรือแผ่นขนาดเล็กจนแทบไม่สามารถมองเห็นได้ไปจนถึงขนาดใหญ่สะดุดตา ซึ่งต่างก็มีความเหมาะสมกับชนิดงานที่แตกต่างกันโดยทั่วไปโครงสร้างภายในส่วนที่เป็นไอซีของ tag นั้นก็จะประกอบด้วย 3 ส่วนหลักๆ ได้แก่ ส่วนของควบคุมการทำงานของภาครับส่งสัญญาณวิทยุ (Analog Front-End) ส่วนควบคุมภาคลอจิก (Digital Control Unit) ส่วนของหน่วยความจำ (Memory) ซึ่งอาจจะเป็นแบบ ROM หรือ EEPROM



รูปที่ 2.5 แสดงบล็อกไดอะแกรมของ Passive tag

Active RFID Tags

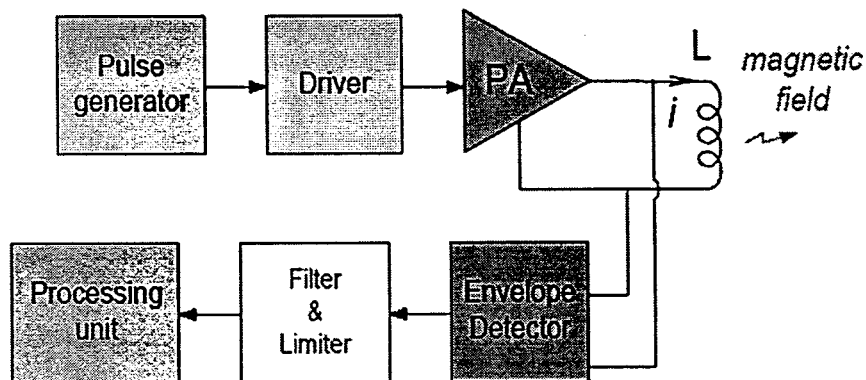
Tag ชนิดนี้จะต้องอาศัยแหล่งจ่ายไฟจากแบตเตอรี่ภายนอก เพื่อจ่ายพลังงานให้กับวงจรภายในทำงานโดย tag แบบนี้สามารถมีหน่วยความจำภายในขนาดใหญ่ๆ ได้ถึง 1 เมกกะไบต์ และสามารถอ่านได้ในระยะไกลสูงสุดประมาณ 10 เมตร แม้ว่า tag ชนิดนี้จะมีข้อดีอยู่หลายข้อแต่ก็มีข้อเสียอยู่ด้วยเหมือนกัน เช่น มีราคาต่อ หน่วยมีขนาดค่อนข้างใหญ่ และมีระยะเวลาในการทำงานที่จำกัดนอกจากการแบ่งจากชนิดที่ว่ามาแล้ว tag ก็ยังถูกแบ่ง ประเภทจากรูปแบบในการใช้งานได้เป็น 3 แบบ คือ แบบที่สามารถถูกอ่านและเขียนข้อมูลได้อย่างอิสระ (Read-Write), แบบเขียนได้เพียงครั้งเดียวเท่านั้นแต่อ่านได้อย่างอิสระ (Write-Once Read-Many หรือ WORM) และแบบอ่านได้เพียงอย่างเดียว (Read-Only) ด้วย อย่างไรก็ตาม เนื่องจากในการใช้งานจะนิยมใช้ Tag ชนิด passive มากกว่าและในที่นี้จะกล่าวถึงเฉพาะ tag ชนิดนี้เป็นหลัก



รูปที่ 2.6 แสดงรูปตัวอย่าง Active tag ที่มีแบตเตอรี่ lithium, 2 ก้อนอยู่ภายนอก

2.1.5 เครื่องอ่าน (Reader)

โดยหน้าที่ของเครื่องอ่านก็คือ การเชื่อมต่อเพื่อเขียนหรือข้อมูลลง tag ด้วยสัญญาณความถี่วิทยุ ภายในเครื่องอ่านจะประกอบด้วย เสาอากาศที่ทำจากขดลวดทองแดง เพื่อใช้รับส่งสัญญาณ ภาครับและภาคส่งสัญญาณวิทยุ และวงจรควบคุมการอ่าน-เขียนข้อมูลจำพวกไมโครคอนโทรลเลอร์ และส่วนของการติดต่อกับคอมพิวเตอร์



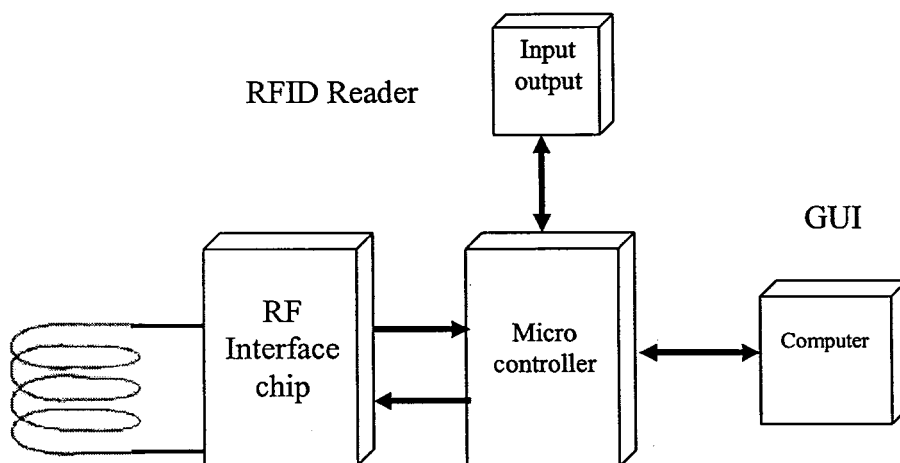
รูปที่ 2.7 แสดงโครงสร้างภายในตัวอ่าน

โดยทั่วไปเครื่องอ่านจะประกอบด้วยส่วนประกอบหลักดังนี้

- ภาครับและส่งสัญญาณวิทยุ
- ภาครับสร้างสัญญาณพาหะ
- ขดลวดที่ทำหน้าที่เป็นสายอากาศ
- วงจรจูนสัญญาณ
- หน่วยประมวลผลข้อมูล และภาคติดต่อกับคอมพิวเตอร์

โดยทั่วไปหน่วยประมวลผลข้อมูลที่อยู่ภายในเครื่องอ่านมักใช้เป็นไมโครคอนโทรลเลอร์ ซึ่งอัลกอริทึมที่อยู่ภายในโปรแกรม จะทำหน้าที่ถอดรหัสข้อมูล (Decoding) ที่ได้รับและทำหน้าที่ติดต่อกับคอมพิวเตอร์โดยลักษณะ ขนาด และรูปร่างของเครื่องอ่านจะแตกต่างกันไปตามประเภทของการใช้งาน เช่น แบบมือถือขนาดเล็กหรือ ติดผนัง จนไปถึงขนาดใหญ่เท่าประตู (Gate size) เป็นต้น

2.1.6 ส่วนประกอบของเครื่องอ่าน RFID



รูปที่ 2.8 แสดงส่วนประกอบของเครื่องอ่าน RFID

RF Interface Chip

- แปลงสัญญาณ Analog เป็น digital ในกรณีที่ส่งข้อมูลจาก Coil ไปยัง MCU
 - แปลงสัญญาณ digital เป็น analog ในกรณีที่ส่งข้อมูลจาก MCU ไปยัง Coil
- MCU (Micro Controller Unit) เป็นหน่วยประมวลผลของ Controller

Microcontroller

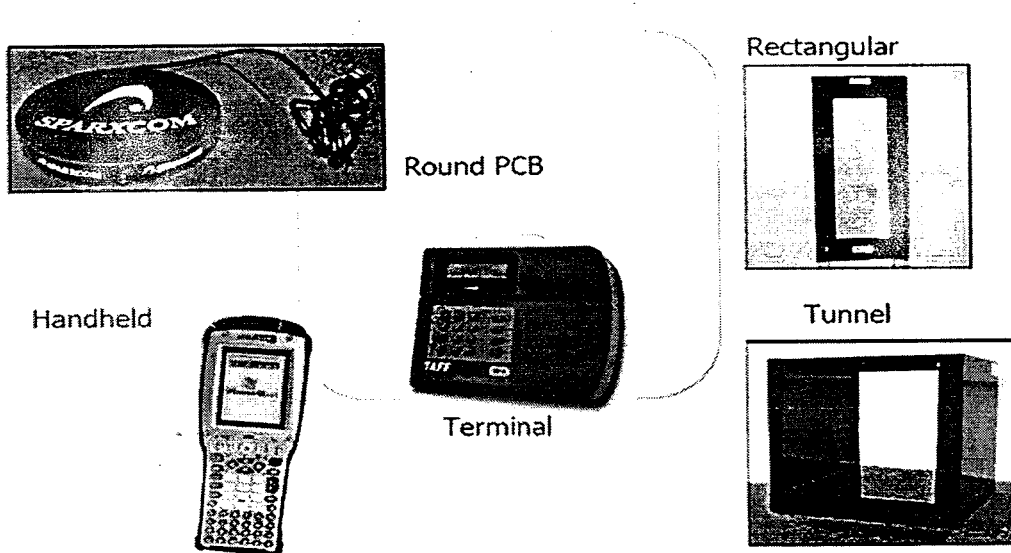
- ควบคุมการทำงานของ RFID Reader
- ถอดรหัสสัญญาณตามรูปแบบที่เข้ารหัสสัญญาณมา

Input/Output

- เป็นตัวรับข้อมูลจากภายนอกเช่น Keypad เป็นต้น
- เป็นอุปกรณ์ที่แสดงหรือควบคุมอุปกรณ์ที่อยู่ภายนอกเช่น 7-Segment, LCD เป็นต้น

Graphic Interface User

- แสดงผลของข้อมูลให้อยู่ในรูปแบบที่ผู้ใช้งานสามารถเข้าใจได้ง่าย
- ควบคุมการทำงานของ RFID Reader

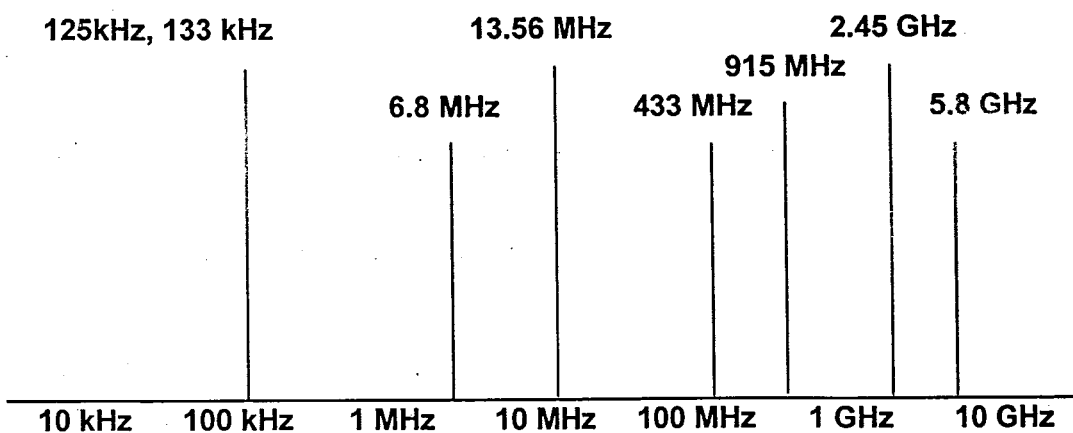


รูปที่ 2.9 ตัวอย่างรูปร่างของ Reader ต่างๆ

2.2 Carrier ในระบบ RFID

ในปี 2000 คลื่นพาหะที่ใช้กันกันในระบบ RFID จะอยู่ในย่านความถี่ ISM (Industrial-Scientific-Medical) ซึ่งเป็นย่านความถี่ที่กำหนดในการใช้งานในเชิงอุตสาหกรรม วิทยาศาสตร์ และการแพทย์ สามารถใช้งานได้โดยไม่ตรงกับย่านความถี่ที่ใช้ในการสื่อสารโดยทั่วไป โดยมี 3 ย่านความถี่ที่ใช้กัน คือสำหรับคลื่นพาหะที่ใช้กันในระบบ RFID อาจ แบ่งออกได้เป็น 3 ย่านหลัก ได้แก่

- ย่านความถี่ต่ำ (Low Frequency: LF) ต่ำกว่า 150 kHz
- ย่านความถี่สูง (High Frequency: HF) 13.56 MHz
- ย่านความถี่สูงยิ่ง (Ultra High Frequency: UHF) 433/868/915 MHz



รูปที่ 2.10 แสดงความถี่ที่ระบบ RFID ถูกใช้งาน

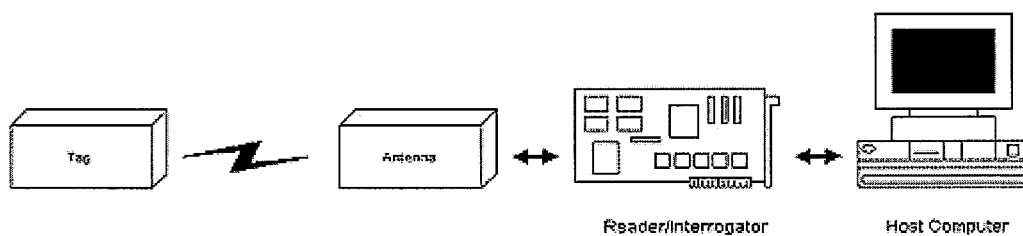
ในแง่การใช้งาน 2 ย่านความถี่แรกจะเหมาะสำหรับใช้ กับงานที่มีระยะการสื่อสารข้อมูลในระยะใกล้ (LH ระยะอ่านประมาณ 10-20 เซนติเมตร และ HF ระยะอ่านประมาณ 1 เมตร) เช่น การตรวจสอบการผ่านเข้าออกพื้นที่ การตรวจหาและเก็บประวัติในสัตว์ ส่วนย่านความถี่สูงยิ่งจะถูกใช้กับงานที่มีระยะการสื่อสารข้อมูลในระยะไกล (UHF ระยะอ่าน ประมาณ 1-10 เมตร) เช่น ระบบเก็บค่าบริการทางด่วน และในปัจจุบันระบบ RFID กำลังถูกวิจัยและพัฒนาในย่านความถี่ไมโครเวฟที่ความถี่ 2.4 GHz และความถี่ 5.8 GHz เพื่อใช้งานที่ต้องการระยะอ่านที่ไกลกว่า 10 เมตร เป็นต้น

ในแง่ของราคาและความเร็วในการสื่อสารข้อมูล เมื่อเทียบกันแล้ว RFID ซึ่งใช้คลื่นพาหะย่านความถี่สูงเป็นระยะที่มีความเร็วในการส่งข้อมูลสูงสุดและมีราคาแพงที่สุดด้วยเช่นกัน ส่วน RFID ที่ใช้คลื่นพาหะในอีก 2 ย่านความถี่จะมีระดับราคาและความเร็วลดหลั่นกันไป

2.2.1 หลักการทำงานและเทคนิคการรับและการส่งข้อมูลของ RFID

ลักษณะการทำงานของระบบ RFID

หัวใจของเทคโนโลยี RFID ได้แก่ “Inlay” ที่บรรจุอุปกรณ์และวงจรรอิเล็กทรอนิกส์กับโลหะที่ยึดหยุ่นได้สำหรับการติดตามหรือทำหน้าที่เป็นเสาอากาศนั่นเอง Inlay มีความหนาสูงสุดอยู่ที่ 0.375 มิลลิเมตร สามารถทำเป็นแผ่นบางอัดเป็นชั้นๆ ระหว่างกระดาษ, แผ่นฟิล์ม หรือพลาสติกก็ได้ ซึ่งเป็นการผลิตเครื่องหมายหรือฉลาก จากวัสดุที่มีราคาไม่แพงมากนัก ซึ่งจะเห็นว่า Inlay มีลักษณะรูปร่างที่บางมาก จึงทำให้ง่ายต่อการติดเป็นป้ายชื่อหรือฉลากของชิ้นงานหรือวัตถุนั้นๆ ได้สะดวก RFID เป็นระบบที่นำเอาคลื่นวิทยุมาเป็นคลื่นพาหะเพื่อใช้ในการสื่อสารข้อมูลระหว่างอุปกรณ์สองชนิดที่เรียกว่า แท็ก (Tag) และตัวอ่านข้อมูล (Reader หรือ Interrogator) ซึ่งเป็นการสื่อสารแบบไร้สาย (Wireless) โดยการนำข้อมูลที่ต้องการส่ง มาทำการมอดูเลต (Modulation) กับคลื่นวิทยุแล้วส่งออกผ่านทางสายอากาศที่อยู่ในตัวรับข้อมูล ดังแผนผังการทำงานของระบบ RFID ในรูปที่ 2.11



รูปที่ 2.11 แสดงแผนผังการทำงานของระบบ RFID

การประยุกต์ใช้งาน RFID จะมีลักษณะการใช้งานที่คล้ายกับบาร์โค้ด (Barcode) และยังสามารถรองรับความต้องการอีกหลายอย่างที่บาร์โค้ดไม่สามารถตอบสนองได้ เนื่องจากบาร์โค้ดจะเป็นระบบที่อ่านได้อย่างเดียว (Reader only) ไม่สามารถทำการเปลี่ยนแปลงข้อมูลที่อยู่บนบาร์โค้ดได้ แต่แท็กของระบบ RFID จะสามารถทั้งอ่านและบันทึกข้อมูลได้ ดังนั้นเราจึงสามารถเปลี่ยนแปลง หรือทำการบันทึกข้อมูลที่อยู่ในแท็กได้ตามความต้องการของผู้ใช้งานนอกจากนี้ระบบ RFID ยังสามารถใช้งานได้แม้ในขณะที่วัตถุ

ที่กำลังเคลื่อนที่ เช่น ในขณะที่สินค้ากำลังเคลื่อนที่อยู่บนสายพานการผลิต (Conveyor) หรือในบางประเทศก็มีการใช้ระบบ RFID ในการเก็บค่าผ่านทางด่วนที่ผู้ใช้บริการทางด่วนไม่ต้องหยุดรถเพื่อจ่ายค่าบริการ ผู้ใช้บริการทางด่วนจะมีแท็กติดอยู่กับรถ และแท็กจะทำการสื่อสารกับตัวอ่านข้อมูล ผ่านสายอากาศขนาดใหญ่ที่ติดตั้งอยู่ตรงบริเวณทางขึ้นทางด่วน ในขณะที่รถแล่นผ่านสายอากาศ ตัวอ่านข้อมูลก็จะคิดค่าบริการและบันทึกจำนวนเงินที่เสียลงในแท็กโดยอัตโนมัติ หรือแม้กระทั่งการใช้งานในปศุสัตว์เพื่อบันทึกประวัติ หรือระบุความแตกต่างของสัตว์แต่ละตัวที่อยู่ในฟาร์มข้อดีของระบบ RFID อีกอย่างก็คือแท็กและตัวอ่านข้อมูลสามารถสื่อสารผ่านตัวกลางได้หลายอย่างเช่น น้ำ, พลาสติก, กระจก, หรือวัสดุทึบแสงอื่นๆ ในขณะที่บาร์โค้ดทำไม่ได้

หลักการทํางานเบื้องต้นของระบบ RFID

1. ตัวอ่านข้อมูลจะปล่อยคลื่นแม่เหล็กไฟฟ้าออกมาตลอดเวลา และคอยตรวจจับว่ามีแท็กเข้ามาอยู่ในบริเวณสนามแม่เหล็กไฟฟ้าหรือไม่ หรืออีกนัยหนึ่งก็คือการคอยตรวจจับว่ามีการมอดูเลตสัญญาณเกิดขึ้นหรือไม่
2. เมื่อมีแท็กเข้ามาอยู่ในบริเวณสนามแม่เหล็กไฟฟ้า แท็กจะได้รับพลังงานไฟฟ้าที่เกิดจากการเหนี่ยวนำของคลื่นแม่เหล็กไฟฟ้าเพื่อให้แท็กเริ่มทํางาน และจะส่งข้อมูลในหน่วยความจำที่ผ่านการมอดูเลตกับคลื่นพาหะแล้วออกมาทางสายอากาศที่อยู่ภายในแท็ก
3. คลื่นพาหะที่ถูกส่งออกมาจากแท็กจะเกิดการเปลี่ยนแปลงแอมพลิจูด, ความถี่ หรือเฟส ขึ้นอยู่กับวิธีการมอดูเลต
4. ตัวอ่านข้อมูลจะตรวจจับความเปลี่ยนแปลงของคลื่นพาหะแปลงออกมาเป็นข้อมูลแล้วทำการถอดรหัสเพื่อนำข้อมูลไปใช้งานต่อไป

2.2.2 วิธีการรับส่งข้อมูลระหว่างแท็กและเครื่องอ่าน

โดยมากมักจะใช้วิธีการมอดูเลตทางแอมพลิจูดหรือใช้การมอดูเลตทางแอมพลิจูดบวกกับการเข้ารหัสแมนเชสเตอร์ (Manchester encoded AM) แต่ในปัจจุบันก็มีแท็กที่ใช้การมอดูเลตแบบอื่นๆ ด้วย เช่น การมอดูเลตแบบเฟสชิฟต์คีย์อิง (Phase Shift Keying: PSK) เฟรควเอนซีชิฟต์คีย์อิง (Frequency Shift Keying: FSK) หรือการใช้การมอดูเลตทางความถี่ (Frequency Modulation : FM)

ในการรับส่งข้อมูลหรือสัญญาณวิทยุระหว่างแท็กกับเครื่องอ่าน จะได้อย่างมีประสิทธิภาพต่อเมื่อสายอากาศมีความยาวที่เหมาะสมกับความถี่พาหะที่ใช้งาน เช่น เมื่อความถี่ใช้งานเป็น 13.56 เมกะเฮิร์ตซ์ ความยาวของเสาอากาศ (เป็นเส้นตรง) ที่เหมาะสมก็คือ 22.12 แน่นอนว่าในทางปฏิบัติคงไม่สามารถนำเสาอากาศที่ใหญ่ขนาดนั้นมาใช้งานกับแท็กขนาดเล็กได้ สายอากาศที่ดีจะเหมาะจะใช้ร่วมกับแท็กมากที่สุดก็คือ สายอากาศที่เป็นขดลวดขนาดเล็กหรือที่มีชื่ออย่างเป็นทางการว่าสายอากาศแบบแมกเนติกไดโพล (Magnetic dipole Antenna) รูปแบบของสายอากาศแบบนี้ก็จะมียูหลากหลายทั้งแบบที่เป็นขดลวดพันแกนอากาศหรือแกนเฟอร์ไรต์ แบบที่เป็นวงลูปที่ทำขึ้นจากลายทองแดงบนแผ่นวงจรพิมพ์ ทั้งที่เป็น

รูปแบบวงกลมและสี่เหลี่ยม ทั้งนี้ความเหมาะสมในการใช้งานก็แตกต่างกันไปตามถี่พาหะและประเภทของงานด้วยเช่นกัน

นอกจากการรับส่งข้อมูลแล้วสายอากาศก็ยังทำหน้าที่เป็นแหล่งจ่ายไฟให้กับแท็กด้วย โดยอาศัยหลักการทำงานตามแนวคิดของไมเคิล ฟาราเดย์ เรื่องแรงดันเหนี่ยวนำในขดลวดที่เกิดขึ้นจากเส้นแรงแม่เหล็ก (จากเครื่องอ่าน) ที่มีค่าเปลี่ยนแปลงไปตามเวลา (Time-varying magnetic field) พุ่งผ่านสายอากาศของแท็ก เมื่อแท็กและเครื่องอ่านตั้งอยู่ห่างกันในระยะ 0.16 เท่าของความยาวของคลื่นพาหะที่ใช้ เรียกปรากฏการณ์ที่เกิดขึ้นว่า Transformer-type Coupling ซึ่งเป็นปรากฏการณ์แบบเดียวกับการเกิดแรงดันไฟฟ้าเหนี่ยวนำขึ้นระหว่างขดลวดปฐมภูมิ (Primary) และขดลวดทุติยภูมิ (Secondary) ในทรานส์ฟอร์มเมอร์ (Transformer) จะเป็นวงจรพื้นฐานสำหรับอธิบายกลไกที่เกิดขึ้นในการส่งข้อมูลของแท็ก

การป้องกันการชนกันของสัญญาณข้อมูล (Anti-Collision)

ในการที่จะรับข้อมูลจากแท็กหลายๆ อัน ทั้งแท็กและตัวเครื่องอ่านต้องได้รับการออกแบบให้รองรับสถานะที่มีแท็กมากกว่า 1 อันทำงาน (ส่งสัญญาณ) มิเช่นนั้นแล้วสัญญาณพาหะก็จะมีการส่งออกในเวลาเดียวกันทำให้เกิดการชนกันของสัญญาณ (Collision) จะทำให้ไม่มีข้อมูลใดๆ ส่งถึงตัวเครื่องอ่านเลย การติดต่อระหว่างแท็กกับตัวเครื่องอ่านเปรียบเสมือน บัสแบบอนุกรม แต่บัสชนิดนี้จะใช้อากาศเป็นตัวกลางในการส่งสัญญาณ ในระบบบัสที่ใช้เคเบิลเป็นตัวกลางก็ต้องมีการควบคุมไม่ให้เกิดการชนกันของสัญญาณ RFID ก็จำเป็นต้องมีการป้องกันให้มีการส่งสัญญาณจากแท็กอันเดียวต่อช่วงเวลาหนึ่งเช่นกัน

อัตราการรับส่งข้อมูลและแบนด์วิดท์

อัตราการรับส่งข้อมูล (Data Transfer Rate) จะขึ้นอยู่กับความถี่ของคลื่นพาหะ โดยปกติถ้าความถี่ของคลื่นพาหะยิ่งสูง อัตราการรับส่งข้อมูลก็จะยิ่งสูงตามไปด้วย ส่วนการเลือกแบนด์วิดท์ หรือย่านความถี่นั้นก็จะมีผลต่ออัตราการรับส่งข้อมูลเช่นกัน โดยมีหลักว่า แบนด์วิดท์ควรจะมีค่ามากกว่าอัตราการรับส่งข้อมูลที่ต้องการอย่างน้อยสองเท่า ยกตัวอย่างเช่น ถ้าใช้แบนด์วิดท์ในช่วง 2.4-2.5 GHz ก็จะสามารถรองรับอัตราการรับส่งข้อมูลได้ถึงประมาณ 2 Megabits ต่อวินาที เป็นต้น แต่การใช้แบนด์วิดท์ที่กว้างเกินไปก็อาจทำให้เกิดปัญหาเกี่ยวกับสัญญาณรบกวนมาก หรือทำให้ S/N Ratio ต่ำลงนั่นเอง ดังนั้นการเลือกใช้แบนด์วิดท์ให้ถูกต้องก็เป็นส่วนสำคัญในการพิจารณา

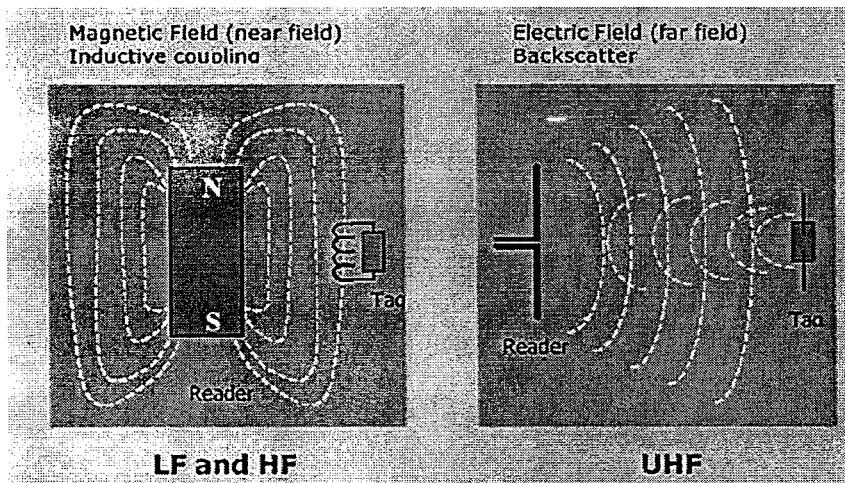
ระยะการรับส่งข้อมูลและกำลังส่ง

ระยะการรับส่งข้อมูลในระบบ RFID ขึ้นอยู่กับปัจจัยสำคัญต่างๆ คือ กำลังส่งของตัวอ่านข้อมูล (Reader/Interrogator Power) กำลังส่งของแท็ก (Tag Power) และสภาพแวดล้อม ส่วนการออกแบบสายอากาศของตัวอ่านข้อมูล จะเป็นตัวกำหนดลักษณะรูปร่างของคลื่นแม่เหล็กไฟฟ้าที่แผ่กระจายออกมาจากสายอากาศ ดังนั้นระยะการรับส่งข้อมูล บางทีอาจขึ้นอยู่กับมุมของการรับส่งระหว่างแท็กและตัวอ่านข้อมูลด้วยเช่นกัน ทั้งนี้ขึ้นอยู่กับรูปร่างของคลื่นแม่เหล็กไฟฟ้าเป็นสำคัญความเข้มของคลื่นแม่เหล็กไฟฟ้า

โดยทั่วไปจะลดลงตามระยะทางโดยแปรผกผันกับระยะทางยกกำลังสอง แต่ในบางสภาพแวดล้อมซึ่งอาจมีการสะท้อนกลับของคลื่นแม่เหล็กไฟฟ้าจากสิ่งต่างๆรอบตัว เช่น โลหะ ก็อาจทำให้ความเข้มของคลื่นแม่เหล็กไฟฟ้าลดลงอย่างรวดเร็ว โดยอาจแปรผกผันกับระยะทางยกกำลังสี่ ปรากฏการณ์เช่นนี้เราเรียกว่า “Multi-path Attenuation” ซึ่งจะส่งผลให้ระยะการรับส่งข้อมูลสั้นลง หรือแม้กระทั่งความชื้นในอากาศก็อาจมีผลในกรณีที่ความถี่สูงๆ ดังนั้นการนำระบบ RFID ไปใช้งานก็ควรมีการคำนึงถึงสภาพแวดล้อม เพราะจะมีผลกระทบต่อระยะการรับส่งข้อมูล และพยายามติดตั้งระบบให้ห่างไกลจากโลหะ ซึ่งอาจทำให้เกิดการสะท้อนของคลื่นแม่เหล็กไฟฟ้าได้ กำลังส่งของแท็กที่จะส่งกลับมายังตัวอ่านข้อมูลนั้น โดยทั่วไปจะมีกำลังที่ต่ำมากเมื่อเทียบกับกำลังส่งของ ตัวอ่านข้อมูล ดังนั้นความไวในการตรวจจับสัญญาณของตัวอ่านข้อมูล ก็เป็นอีกจุดหนึ่งที่ต้องพิจารณา ถึงแม้ในทางเทคนิคเราจะสามารถทำให้ตัวอ่านความถี่ ดังนั้นในระบบ RFID โดยทั่วไป จะมีกำลังส่งเพียงระหว่าง 100-500 mW

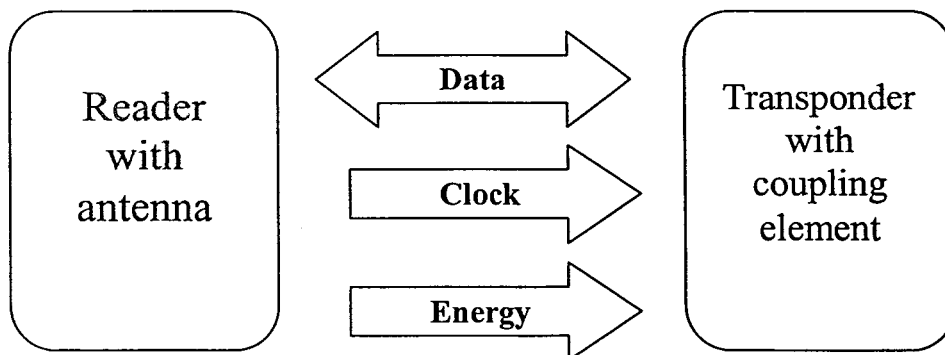
การส่งข้อมูลระหว่างตัวอ่านและ Tag มี 2 วิธีคือ

1. Inductive Coupling (LF and HF)
2. Backscatter Coupling (UHF and Microwave)

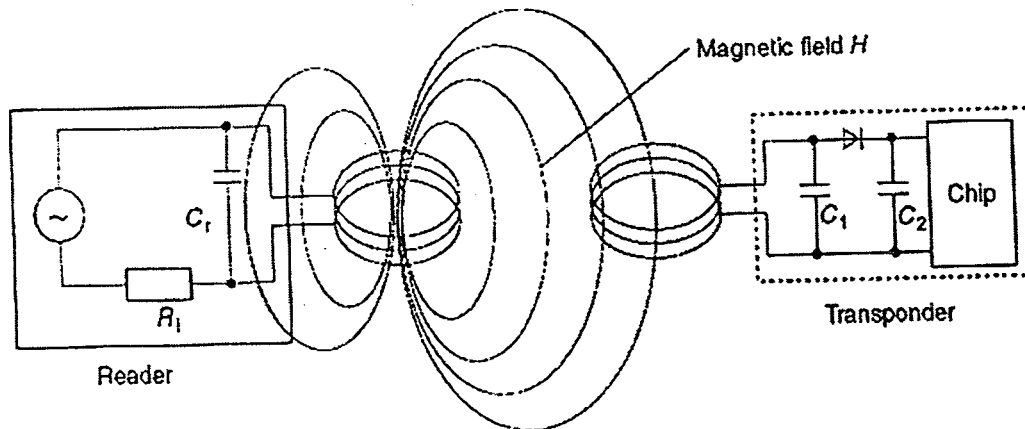


รูปที่ 2.12 แสดงการการส่งข้อมูลระหว่างตัวอ่านและ Tag รูปแบบต่างๆ

ที่นี้เราจะขอกกล่าวถึงการส่งข้อมูลระหว่างตัวอ่านและ Tag แบบ inductive coupling



รูปที่ 2.13 การสื่อสารระหว่าง Reader กับ Tag



รูปที่ 2.14 แสดงการสร้าง Power supply ให้กับ tag โดย reader

จากรูปเราจะเห็นว่ารูปแบบของ Inductive Coupling จะอาศัยการเชื่อมโยงของสนามแม่เหล็ก โดยจะถูก generate ด้วยเครื่องอ่านเพื่อเป็นแหล่งจ่ายให้กับ Tag

2.2.3 Power supply to passive transponder

การเหนี่ยวนำของสนามแม่เหล็กที่ถูก เชื่อมโยงจากเครื่องอ่านไปยัง Tag จะประกอบด้วย ข้อมูลของสัญญาณ ไมโครชิพและส่วนของพื้นที่ของขดลวดซึ่งอยู่ในส่วนของสายอากาศ หมายถึงว่ากำลังงานทั้งหมดที่ต้องการสำหรับการดำเนินการของไมโครชิพที่อยู่ส่วนของ tag จะถูกจัดการหรือจ่ายให้โดยเครื่องอ่าน (Reader) ตามรูปที่ 2.14 สำหรับในส่วนนี้สายอากาศของเครื่องอ่านจะทำการสร้างสนามแม่เหล็กไฟฟ้าความถี่สูงที่ทะลุผ่านและตัดตามขวางของขดลวดและรอบๆขดลวด เพราะว่าความยาวคลื่นของช่วงความถี่ที่ถูกใช้ (<135 kHz : 2400 m, 13.56 MHz : 22.1 m) และระยะทางมันไกลกว่าระหว่างระยะทางสายอากาศของเครื่องอ่านและสายอากาศของ Tag ดังนั้นสนามแม่เหล็กไฟฟ้าอาจจะถูกพิจารณาหรือจัดการการเปลี่ยนแปลงของสนามที่ทะลุผ่านเข้าไปในสายอากาศของ Tag ที่ระยะทางจากเครื่องอ่านและแรงดันจะถูกสร้างที่ขดลวดสายอากาศของ Tag โดยการเหนี่ยวนำและแรงดันนี้จะถูก rectifier และก็เป็นแหล่งจ่ายของ ไมโครชิพที่อยู่ใน Tag

2.2.4 การ Interface ข้อมูลของ RFID

การ Interface ข้อมูลของ RFID เราจะแยกเป็น 2 แบบคือ

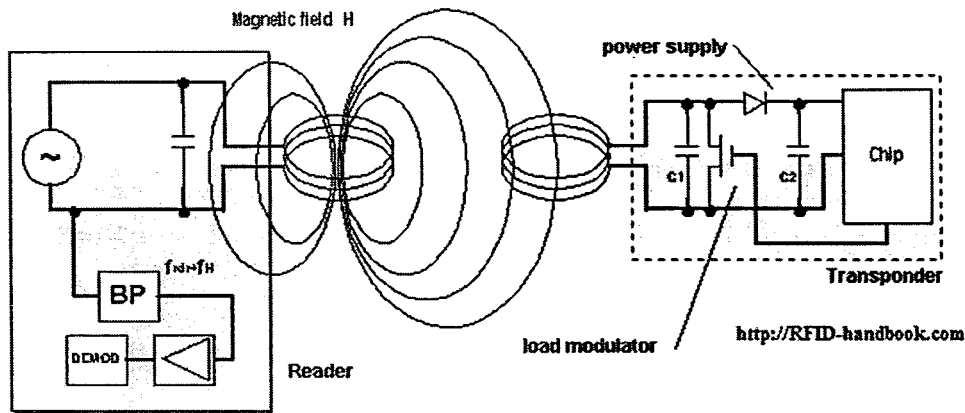
- Data transfer reader – tag
- Data transfer tag – reader

Data transfer reader-tag

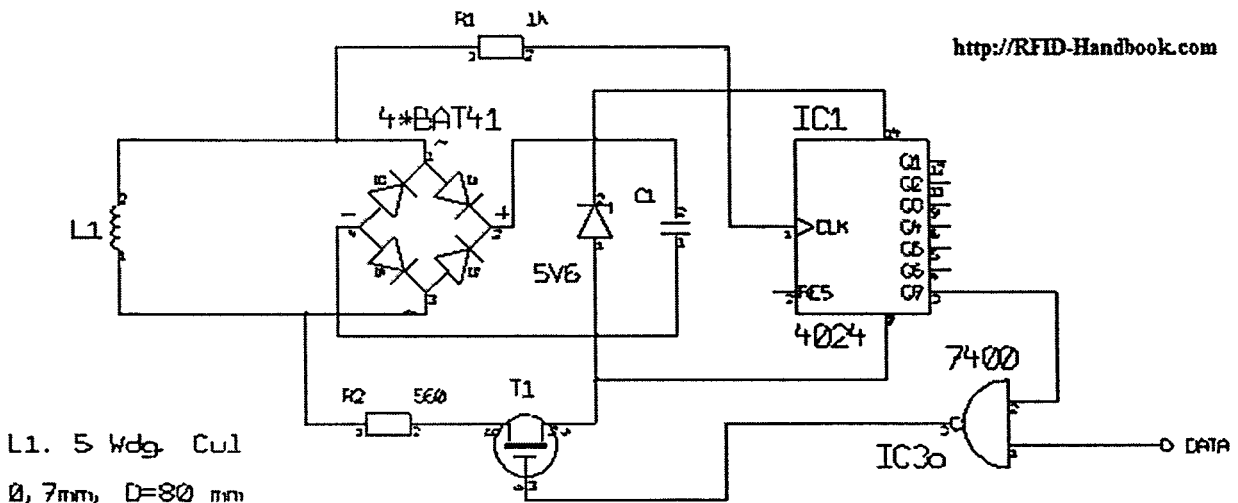
การส่งจาก Reader – Tag จะใช้การมอดูเลชั่นแบบดิจิตอลคือ Reader จะแปลงสัญญาณดิจิตอลเป็นอนาล็อกเพื่อส่งไปให้ Tag การมอดูเลตแบบ ASK จะใช้ในการส่งผ่านข้อมูลจาก Reader ไปยัง Tag

Data transfer tag – reader

การส่งข้อมูลจาก Tag ไปยัง Reader คือ Tag มันจะใช้พลังงานของการทำงานของสนามแม่เหล็กเพื่อที่จะส่งกลับ (Return) หรือตอบสนอง (Response) ไปยังเครื่องอ่าน (Reader) วิธีการนี้ถูกเรียกว่า Load Modulation และ Load Modulation นี้จะอาศัยวิธีการของความต้านทานโหลด (Load Resistance) ที่รวมอยู่ในวงจรที่ถูก On – Off “1” หรือ “0” ขึ้นอยู่กับข้อมูลและถูกควบคุมโดยข้อมูลและจาก รูปที่ 2.15 จะเป็นกระบวนการการสร้าง Load Modulation ใน Tag ด้วยสวิตซ์ Drain – Source Resistance ของ FET บนตัว Chip



รูปที่ 2.15 การ Generation of Load Modulation



รูปที่ 2.16 วงจร Generation of Load Modulation with Sub Carrier

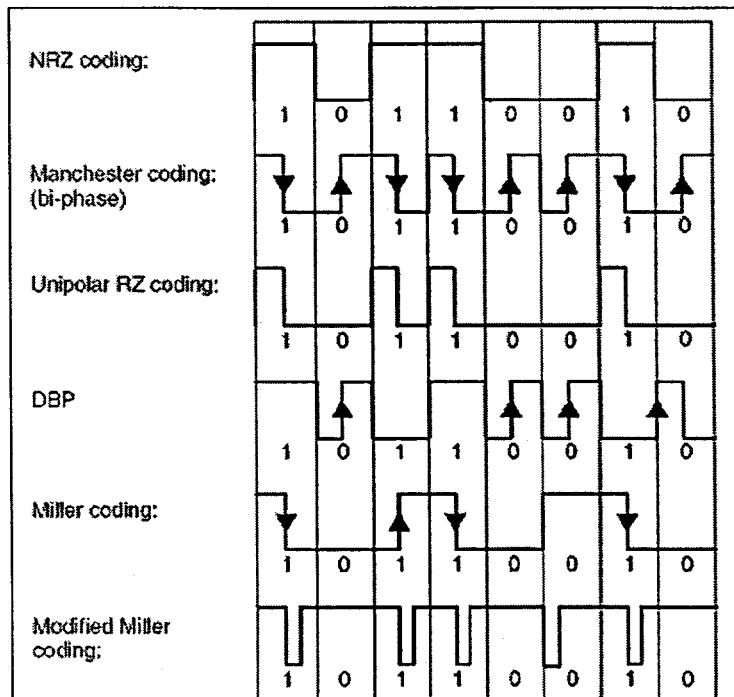
จากรูปที่ 2.16 เป็นตัวอย่างวงจร Generation of Load Modulation with Sub Carrier สำหรับ Tag ในการใช้ Load Modulation With Sub Carrier วงจรนี้ถูกออกแบบสำหรับ การดำเนินการของความถี่

13.56 MHz และสร้าง Sub Carrier 212 kHz จากวงจรจะเห็นว่าแรงดันที่ถูกเหนี่ยวนำที่ขดลวดสายอากาศ L1 โดยการเปลี่ยนแปลงของสนามแม่เหล็กของเครื่องอ่าน (Reader) และถูก Rectified โดยใช้บริดจ์ไดโอด (D1-D4) และทำให้เรียบด้วย C1 และมีวงจรที่เป็น Supply Voltage (ZD 5V6) ป้อนกันแรงดันที่ไม่สามารถควบคุมได้เมื่อแรงดันมันเพิ่มขึ้นเมื่อ Tag เข้ามาใกล้เสาอากาศของเครื่องอ่านและในส่วนของแรงดันความถี่สูงของสายอากาศ (13.56 MHz) เดินทางไปที่ความถี่อินพุท (CLK) และถูกป้อนโดย R1 และใช้ Tag เป็นส่วนสำคัญสำหรับการสร้างสัญญาณคล็อกภายในหลังจากที่หารด้วย 64 สัญญาณคล็อกของ Sub Carrier คือ 212 kHz และออกมาที่เอาต์พุทของ Q7 สัญญาณคล็อกของ Sub Carrier ถูกควบคุมด้วยการไหลของ Serial Data ที่ Data Input (DATA) ถูกผ่านด้วยสวิตช์ (T1) ถ้ามีสัญญาณลอจิก High ที่ Input (DATA) แล้วสัญญาณคล็อกของ Sub Carrier จะถูกส่งผ่านเข้าไปในสวิตช์ (T1) และ Load – Resistor (R2) แล้วถูกสวิตช์ให้ ON และ OFF ในเวลาเดียวกันกับ Sub Carrier Frequency

2.2.5 การเข้ารหัสข้อมูล (Data Coding)

การรับส่งข้อมูลแบบตรงไปตรงมาจะทำให้ข้อมูลที่ส่งและรับนั้นมีความยาวเกินไปจึงมีการคิดค้นวิธีเข้ารหัสซึ่งการใช้รหัสจะช่วยให้การรับส่งข้อมูลสั้นสุดและไม่ถูกรบกวนจากภายนอก การเข้ารหัสมักเป็นที่นิยมอย่างกว้างขวางในการสื่อสารแบบดิจิทัล

ในการสื่อสารแบบดิจิทัลนั้นใช้สัญลักษณ์ 0 กับ 1 แทนข้อมูลโดยความแตกต่างของข้อมูลจะเป็นจะเป็นตัวกำหนดค่า 0 กับ 1 ของมาตรฐานของแต่ละบิตของข้อมูล ข้อมูลจะถูกจัดเรียงแบบแนวนอนหรือแบบเส้นตรง (Line Code) ซึ่งมาตรฐานของ Line Code จะมีหลายมาตรฐานดังนี้



รูปที่ 2.17 การเข้ารหัสข้อมูลในแบบต่างๆ

จากรูปที่ 2.17 เราจะพบว่ามาตรฐานของ Line Code มีอยู่หลายชนิด ความแตกต่างของมันก็คือ การแสดงค่า 0 กับ 1 ที่ให้สัญลักษณ์แตกต่างกัน Line Code ที่พบบ่อยและที่นิยมใช้คือ NRZ และ Manchester Coding

NRZ : สัญลักษณ์แทนค่า "1" คือช่วงสัญญาณเป็น High สัญลักษณ์แทนค่า "0" คือช่วงสัญญาณเป็น Low

Manchester Coding : สัญลักษณ์ "1" จะแทนด้วยช่วงขอบขาลงของสัญญาณ สัญลักษณ์ "0" จะแทนด้วยช่วงขอบขาขึ้นของสัญญาณ

2.2.6 Digital modulation

เนื่องจากหลักการพื้นฐานของ RFID คือการใช้หลักคลื่นแม่เหล็กไฟฟ้าในการสื่อสารรับส่งข้อมูล ดังนั้นก่อนการสื่อสารจะต้องมีกระบวนการผสมข้อมูลที่เข้ารหัสไว้แล้วคลื่นพาหะ (Modulation) แต่เนื่องจากลักษณะข้อมูลเป็นแบบดิจิทัลจึงต้องใช้วิธีผสมแบบดิจิทัลจะแบ่งเป็น 3 แบบ

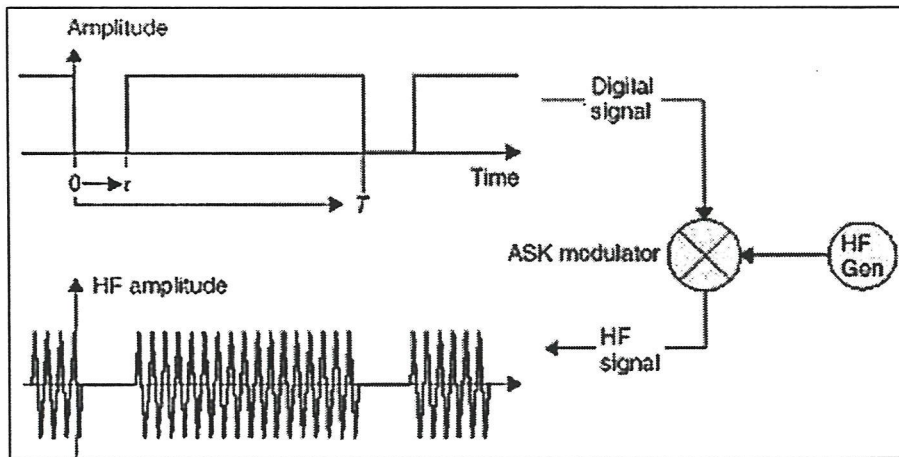
ASK : amplitude shift keying

PSK : phase shift keying

FSK : frequency shift keying

- Amplitude Shift Key

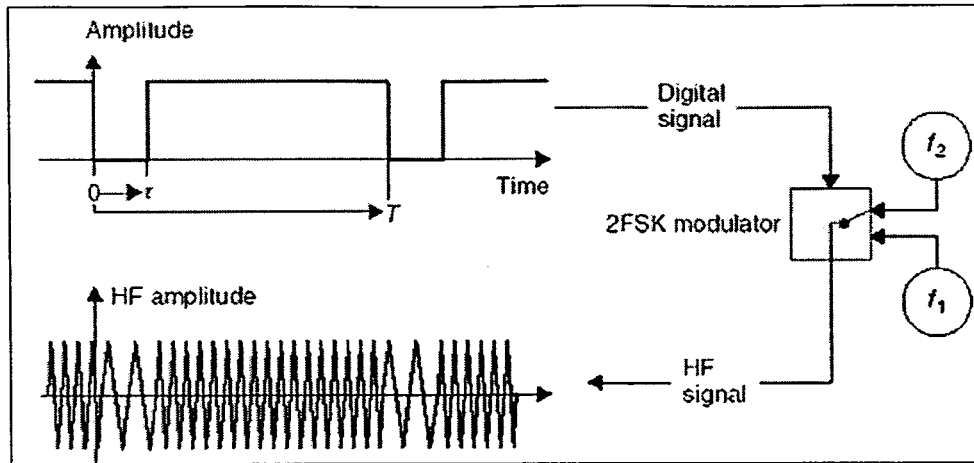
เป็นการผสมข้อมูลโดยอาศัยการเปลี่ยนแปลงของยอดคลื่นพาหะ (Amplitude) เป็นตัวแสดงลักษณะข้อมูลโดยความถี่ของคลื่นพาหะไม่เปลี่ยนแปลงรูป



รูปที่ 2.18 การผสมข้อมูลโดยอาศัยการเปลี่ยนแปลงของยอดคลื่นพาหะ

- Frequency Shift Keying (FSK)

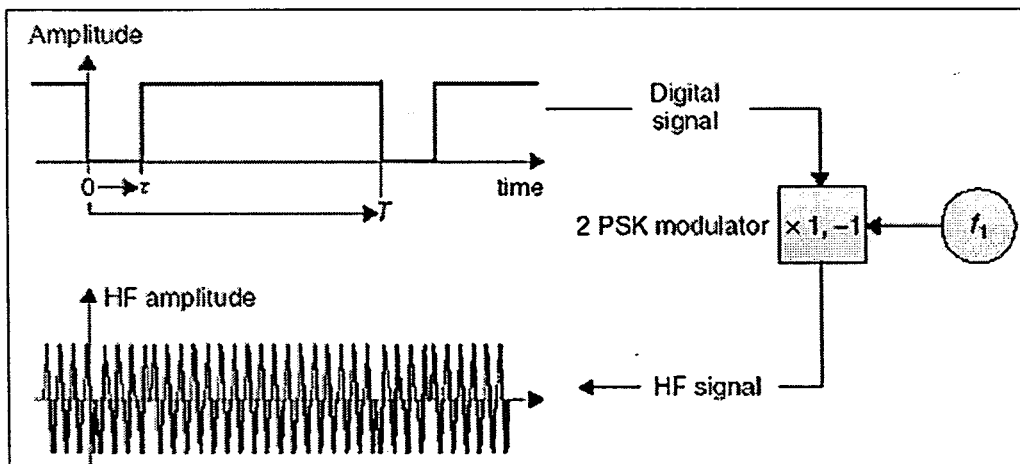
วิธีนี้อาศัยการเปลี่ยนแปลงความถี่ของคลื่นพาหะระหว่าง 2 ความถี่ขึ้นอยู่กับสถานะของข้อมูล 0 กับ 1 โดยความสูงของยอดคลื่น (Amplitude) ไม่เปลี่ยนแปลง



รูปที่ 2.19 การผสมข้อมูลโดยอาศัยการเปลี่ยนแปลงความถี่

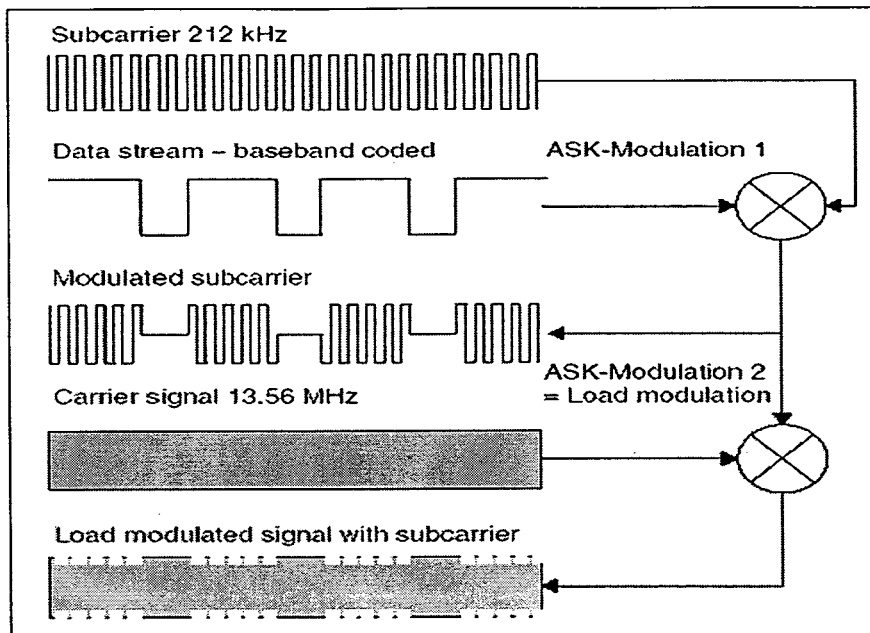
- Phase Shift Keying (PSK)

วิธีนี้จะใช้หลักการเปลี่ยนแปลงของเฟสของรูปคลื่นเป็นตรงกันข้าม (0 องศา กับ 180 องศา) เมื่อมีการเปลี่ยนแปลงสถานะข้อมูลดังรูปที่ 2.20



รูปที่ 2.20 การผสมข้อมูลโดยอาศัยการเปลี่ยนแปลงเฟส

การ Modulation แบบ ASK Sub carrier



รูปที่ 2.21 การ Modulation แบบ ASK Sub carrier

2.2.7 การสื่อสารระหว่างตัวอ่านและ Tag

- FDX (Full duplex)
- HDX (Half duplex)
- SEQ (Sequential)

ตารางที่ 2.1 กระบวนการส่งข้อมูลของ RFID

Procedure:							
FDX:							
Energy transfer:	[Solid black bar]						
downlink:	[Bar]	[Bar]	[Bar]	[Bar]	[Bar]	[Bar]	[Bar]
uplink:	[Bar]	[Bar]	[Bar]	[Bar]	[Bar]	[Bar]	[Bar]
HDX:							
Energy transfer:	[Solid black bar]						
downlink:	[Bar]	[Bar]	[Bar]	[Bar]	[Bar]	[Bar]	[Bar]
uplink:	[Bar]	[Bar]	[Bar]	[Bar]	[Bar]	[Bar]	[Bar]
SEQ:							
Energy transfer:	[Bar]	[Bar]	[Bar]	[Bar]	[Bar]	[Bar]	[Bar]
downlink:	[Bar]	[Bar]	[Bar]	[Bar]	[Bar]	[Bar]	[Bar]
uplink:	[Bar]	[Bar]	[Bar]	[Bar]	[Bar]	[Bar]	[Bar]

→ t

ในส่วนของ Tag จะถูกพิจารณาในส่วนที่ใช้อิเล็กทรอนิกส์ไมโครชิพแสดงเป็น Data carrying และมันจะมีความจุของข้อมูลและสะสมไว้ประมาณ 2-3 กิโลไบต์ขึ้นไป และการอ่านและเขียน Data Carrying ต้องมีความเป็นไปได้ที่จะส่งข้อมูล ระหว่าง Tag กับเครื่องอ่าน การส่งผ่านข้อมูลนี้จะนำเอากระบวนการอย่างใดอย่างหนึ่งมาพิจารณาคือกระบวนการ Full – Half Duplex ที่จะถูกพิจารณาในส่วนนี้ และจะพิจารณาในส่วนของ Sequential ด้วย

ในส่วนของกระบวนการของ Half Duplex (HDX) ข้อมูลที่จะส่งจาก Tag ไปยังเครื่องอ่านหรือเครื่องอ่านไปยังตัว Tag ที่ความถี่ที่ต่ำกว่า 30 MHz ในส่วนนี้มักจะถูกใช้กับกระบวนการ Load Modulation ทั้งมีและไม่มี Sub Carrier รวมถึงวงจรที่ค่อนข้างง่าย รวมทั้งเทคโนโลยีของ เรดาร์ ที่ใช้ความถี่มากกว่า 100 MHz กระบวนการ Load Modulation และสัญญาณที่ถูก Modulate กระบวนการ การสะท้อน ที่มีอิทธิพลโดยตรงของสนามแม่เหล็กหรือสนามแม่เหล็กไฟฟ้าจะถูกสร้างโดยเครื่องอ่านและนั่นคือเป็นกระบวนการ ฮาโมนิก

ในส่วนของ Full Duplex (FDX) ข้อมูลที่จะส่งจาก Tag ไปยังเครื่องอ่านและในเวลาเดียวกันก็เป็นการส่งจากเครื่องอ่านไปยัง Tag และนี่เป็นการรวมกระบวนการที่ข้อมูลจะถูกส่งผ่านจาก Tag ที่เป็นความถี่ส่วนน้อยของเครื่องอ่านเป็นซับฮาโมนิก ความถี่ไม่ขึ้นกับใคร อย่างไรก็ตามทั้งสองกระบวนการจะมีสิ่งที่เหมือนกันตรงที่ว่า การส่งพลังงานจากเครื่องอ่านไปยัง Tag จะส่งไปแบบต่อเนื่องทิศทาง การไหลของข้อมูลเป็นอิสระและในระบบ Sequential (SEQ) การส่งพลังงานจาก Tag ไปยังเครื่องอ่านมันจะถูกจำกัดด้วยเวลาเท่านั้น (การดำเนินการของ พัลส์และระบบของ พัลส์) ข้อมูลที่จะส่งจาก Tag ไปยังเครื่องอ่าน เกิดขึ้นระหว่างแหล่งจ่ายกำลังงานกับ Tag

2.3 BARCODE กับ RFID

BARCODE หรือ รหัสแท่ง

คือ การแทนข้อมูลที่เป็นรหัสเลขฐานสอง (Binary codes) ในรูปแบบของแถบสีดำและขาวที่มีความกว้างของแถบที่ต่างกัน แถบที่มีสีและความกว้างที่แตกต่างกันนี้จะมีค่าเป็นตัวเลขที่แตกต่างกันและมาตรฐานสากลได้กำหนดค่าไว้ เทคโนโลยีบาร์โค้ดถูกนำมาใช้ทดแทนในส่วนการบันทึกข้อมูล (Data Entry) การบันทึกด้วยคีย์บอร์ดมีอัตราความผิดพลาดอยู่ประมาณ 1 ใน 100 หรือบันทึกข้อมูลผิดพลาด 1 ตัวอักษรในทุกๆ 100 ตัวอักษรและเมื่อเปลี่ยนมาใช้ระบบบาร์โค้ดแทนในขั้นตอนการบันทึกข้อมูล อัตราการเกิดความผิดพลาดจะลดลงเหลือเพียง 1 ใน 10,000,000 ตัวอักษร

2.3.1 ส่วนประกอบของบาร์โค้ด

สัญลักษณ์ของบาร์โค้ดที่ใช้กันมีการกำหนดขึ้นมาหลายรูปแบบ ตามมาตรฐานของแต่ละองค์กร และตามจุดประสงค์ของการใช้งาน แต่โดยทั่วไปแล้วบาร์โค้ดจะมีส่วนประกอบต่างๆ ดังต่อไปนี้

1. Quiet Zone เป็นบริเวณที่ว่างเปล่าไม่มีการพิมพ์ข้อความใดๆ โดยจะอยู่ก่อนและหลังบาร์โค้ด
2. Start/Stop Character เป็นบริเวณแถบแท่งหรือช่องว่าง

3. Data เป็นบริเวณแถบแท่งหรือช่องว่างที่แทนข้อมูลต่างๆ ที่เราต้องการ
4. Check Digit เป็นบริเวณแถบแท่งที่ไว้สำหรับเก็บค่าตัวเลข เพื่อตรวจสอบในข้อมูลส่วน Data เพื่อเตรียมสั่งให้เซนเซอร์เริ่มต้นหรือหยุดบาร์โค้ด เพื่อให้มั่นใจว่าถูกต้องแม่นยำ

หลักการทํางาน ของ Barcode

1. ส่วนลายเส้น ซึ่งเป็นลายเส้นสีขาว (โปร่งแสง) และสีดำ มีขนาดความกว้างของลายเส้นตามมาตรฐานแต่ละชนิดของบาร์โค้ด
2. ส่วนข้อมูลตัวอักษร เป็นส่วนที่แสดงความหมายของชุดข้อมูลลายเส้นสำหรับให้อ่านเข้าใจ
3. ส่วนแถบสว่าง เป็นส่วนที่เครื่องอ่านบาร์โค้ดใช้กำหนดขอบเขตบาร์โค้ดและกำหนดค่าให้กับสีขาว (ความเข้มของการสะท้อนแสงในสีของพื้นผิวแต่ละชนิดที่ใช้แทนสีขาว) โดยทุกเส้นจะมีความยาวเท่ากันเรียงตามลำดับในแนวนอนจากซ้ายไปขวา

แถบสีทั้งสีขาวและสีดำที่มีความกว้างจะแทนค่าเป็น 1 และแถบสีที่มีความแคบ (หรือมองด้วยตาเหมือนเป็นเส้นตรงเล็กๆ) ทั้งขาวและดำจะมีค่าเป็น 0 แถบขาวและดำที่มีลักษณะและชื่อที่ใช้คือ

- แถบสีดำที่มีความกว้างมากกว่าเรียกว่า Wide Bar ถ้ามีความกว้างน้อยกว่าเรียกว่า Narrow Bar
- ช่องว่างหรือแถบสีขาวที่มีความกว้างมากกว่าเรียกว่า Wide Space ถ้ามีความกว้างน้อยกว่าเรียกว่า Narrow Space



รูปที่ 2.22 รูปแบบของ Barcode ทั่วไป

2.3.3 หลักการของ RFID

เป็นเทคโนโลยีไร้สายที่ใช้คลื่นความถี่วิทยุในการระบุลักษณะเฉพาะของวัตถุแต่ละชิ้น โดยการติดป้ายฉลากด้วยแผ่นอิเล็กทรอนิกส์ (Tag) ที่มีการลงโปรแกรมควบคุมที่ระบุอย่างเฉพาะเจาะจง โดยติดไปกับวัตถุที่ต้องการตรวจสอบ และระบุถึงข้อมูลของวัตถุนั้น ๆ กล่าวคือแนวคิดในการทำงานนั้นเช่นเดียวกับบาร์โค้ด

2.3.4 จุดเด่นของ RFID

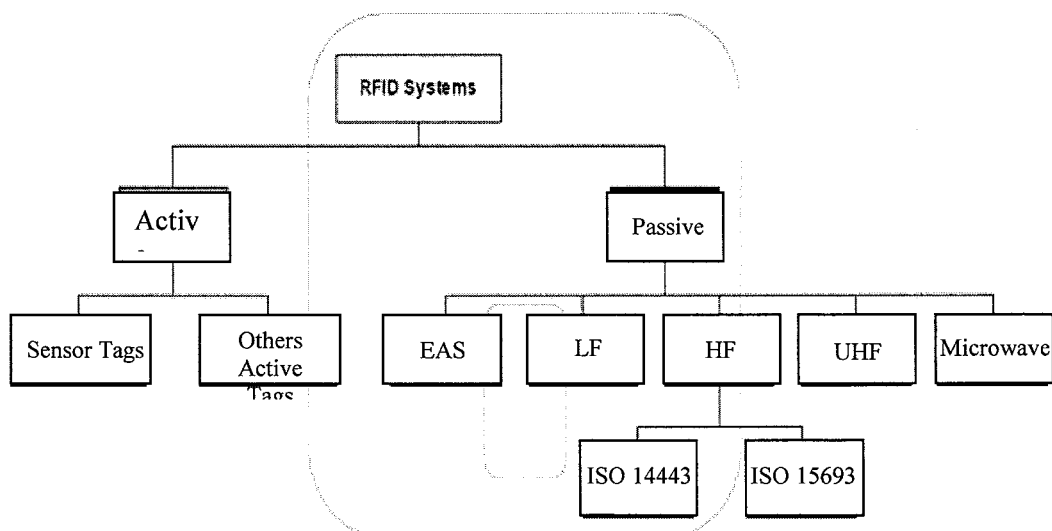
- สามารถอ่านค่าข้อมูลจากป้ายหรือ Tag (Transponder/Tag) ได้หลายๆ tag พร้อมๆ กันแบบไร้สัมผัส
- สามารถอ่านค่าได้แม้ในสภาพที่ทัศนวิสัยไม่ดี (มองไม่เห็น)
- สามารถอ่านค่าได้แม้ไม่ต้องอยู่ในแนวเส้นตรง (Non-Line of Sight) เดียวกับเครื่องอ่าน (RFID Reader)

- ทนต่อความเปียกชื้น แรงสั่นสะเทือน การกระทบกระแทก
- สามารถอ่านค่าข้อมูลได้ระยะไกล
- สามารถอ่านค่าข้อมูลได้ด้วยความเร็วสูง

2.3.5 ข้อแตกต่างระหว่าง Barcode กับ RFID

ตาราง 2.2 เปรียบเทียบระบบ Barcode กับระบบ RFID

Barcode	RFID
ใช้แสงเลเซอร์ในการอ่าน	ใช้สัญญาณความถี่วิทยุในการถอดรหัส
ต้องติดแผ่นบาร์โค้ดในตำแหน่งที่อ่านง่าย	ติดไว้ตำแหน่งใดก็ได้
ขณะอ่านแสงเลเซอร์ต้องตกกระทบพอดีและต้องสัมผัสโดยตรงกับวัตถุ	สามารถอ่านผ่านวัตถุได้
อ่านได้อย่างเดียว (Read Only)	อ่านและลบเพื่อเขียนข้อมูลใหม่ได้ (Read / Write)
ไม่สามารถเป็นระบบรักษาความปลอดภัยในตัวได้ ต้องใช้ร่วมกับระบบรักษาความปลอดภัย	เป็นระบบรักษาความปลอดภัยด้วย
ความถูกต้องแม่นยำอยู่ที่อัตราเพียง 1 ใน 10^7 หรือ 10,000,000 ตัวอักษร	ความถูกต้องแม่นยำมากกว่าบาร์โค้ด
เวลาในการอ่านข้อมูลประมาณ 2 วินาที โดยอ่านได้ทีละชิ้น	เวลาในการอ่านข้อมูลประมาณ 800 ms สามารถอ่านได้ที่ละหลายๆ ชิ้น
เสื่อมคุณภาพได้ง่าย	คงทนต่อสภาพแวดล้อม ทั้งอุณหภูมิและสนามแม่เหล็ก



รูปที่ 2.23 ระบบ RFID

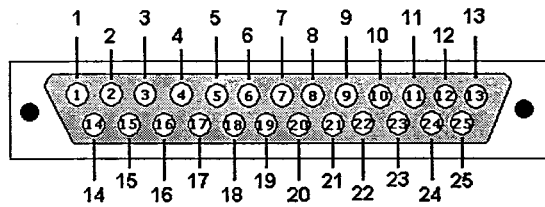
2.4 มาตรฐานพอร์ตอนุกรมแบบ RS-232

มาตรฐานการเชื่อมต่อแบบอนุกรม RS-232 เป็นมาตรฐานอุตสาหกรรมที่ออกแบบมาเพื่อใช้ในการส่งข้อมูลอนุกรมแบบอะซิงโครนัส 2 ทิศทางด้วยมาตรฐาน RS-232 ในอดีตนั้นถูกออกแบบมาเพื่อการส่งผ่านข้อมูลจากคอมพิวเตอร์ไปยังโมเด็มเพียงอย่างเดียวเพื่อที่จะนำข้อมูลจากโมเด็มนี้สื่อสารผ่านสายโทรศัพท์ไปยังคอมพิวเตอร์อีกชุดหนึ่งซึ่งอยู่ห่างไกลกันโดยคณะกรรมการที่เรียกว่า “สมาคมอุตสาหกรรมอิเล็กทรอนิกส์” (Electronic Industries Association : EIA) ได้วางมาตรฐานที่มีชื่อเรียกว่า EIA RS-232 มาตรฐานนี้ในช่วงแรกจะใช้คอนเน็คเตอร์เป็นแบบ DB-25 โดยกำหนดความยาวสูงสุดของสายสัญญาณไว้ที่ 50 ฟุต มีระดับสัญญาณตั้งแต่ -3 ถึง -12V แสดงว่ามีข้อมูล (MARK) และ +3 ถึง +12V แสดงว่าเป็นช่องว่าง (SPACE) มาตรฐาน RS-232 ได้กำหนดรูปแบบของอุปกรณ์เชื่อมต่อข้อมูล (Data Terminal Equipment : DTE) กับวงจรข้อมูลปลายทาง (Data Circuit Terminating : DCE) ไว้ว่าอุปกรณ์ DTE จะต้องเป็นอุปกรณ์ที่มีการประมวลผลในตัวเช่น ไมโครคอลโทรลเลอร์ หรือไมโครคอมพิวเตอร์ ซึ่งมีความสามารถในการสร้างบิตข้อมูลแบบอนุกรมได้ ส่วนอุปกรณ์ DCE จะทำหน้าที่เป็นเพียงตัวรับข้อมูลที่ส่งมาจาก DTE เท่านั้น โดยการรับส่งข้อมูลระหว่างอุปกรณ์ ทั้งสองจะทำผ่านมาตรฐาน RS-232 ข้อแตกต่างของอุปกรณ์ DTE และอุปกรณ์ DCE อย่างหนึ่งให้เห็นได้ชัดคือ คอนเน็คเตอร์ของ DTE เป็นตัวผู้ ส่วนคอนเน็คเตอร์ของ DCE จะเป็นตัวเมียซึ่งพอร์ตอนุกรมของคอมพิวเตอร์ที่ใช้กันอยู่ทั่วไปจะเป็นแบบ DTE ส่วนคอนเน็คเตอร์ที่อยู่โมเด็มจะเป็นแบบ DCE สำหรับการใช้งานบนคอมพิวเตอร์พอร์ตอนุกรม RS-232 มักถูกใช้เชื่อมต่อกับโมเด็มหรือเมาส์โดยสามารถรับส่งข้อมูลได้ที่ความยาวของสายสัญญาณสูงสุดถึง 20 เมตร

2.4.1 คอนเน็คเตอร์สำหรับพอร์ต RS-232 และการเชื่อมต่อ

มาตรฐานการเชื่อมต่อแบบ RS-232 จะใช้คอนเน็คเตอร์แบบ DB-25 ตัวผู้หรือ DB-9 ตัวผู้คอนเน็คเตอร์ DB-25 จะมีขาต่อใช้งานเพียง 9 เส้นเช่นเดียวกับคอนเน็คเตอร์แบบ DB-9 เนื่องจากขาอื่นๆ ที่เคยใช้งานในอดีตนั้นปัจจุบันมีการใช้งานไม่มากนัก จึงถูกยกเลิกไปโดยแสดงรูปร่างและตำแหน่งขาในรูปที่ 2.24 สำหรับการเชื่อมต่อคอมพิวเตอร์กับอุปกรณ์ ภายนอกที่แสดงในรูปที่ 2.25 ลูกศรในรูปแสดงถึงทิศทางในข้อมูลในรูปที่ 2.25 (ก) เป็นการเชื่อมต่อแบบ Null Modem หรือการเชื่อมต่อแบบโดยตรง โดยไม่ต้องผ่านโมเด็ม ส่วนในรูป 2.25 (ข) เป็นการเชื่อมต่อแบบ Null Modem ในลักษณะที่ใช้สายสัญญาณเพียง 3 เส้นโดยเส้นหนึ่ง สำหรับส่งข้อมูล อีกเส้นสำหรับรับข้อมูลและเส้นสุดท้ายเป็นกราวด์สำหรับรายละเอียดหน้าที่การทำงานในแต่ละขาของพอร์ตอนุกรม RS-232 มีดังนี้

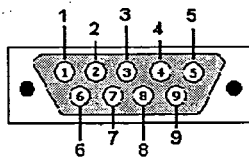
DB25	Signal	DB25	Signal
3	RD	2	TD
2	TD	3	RD
20	DTR	6,8	DSR, DCD
6,8	DSR, DCD	20	DTR
4	RTS	5	CTS
5	CTS	4	RTS
7	SGND	7	SGND
22	RI	22	RI



(ก) แบบ DB-25

DB9	Signal	DB9	Signal
2	RD	3	TD
3	TD	2	RD
4	DTR	6,1	DSR, DCD
6,1	DSR, DCD	4	DTR
7	RTS	8	CTS
8	CTS	7	RTS
5	SGND	5	SGND
9	RI	9	RI

(ข) แบบ DB-9

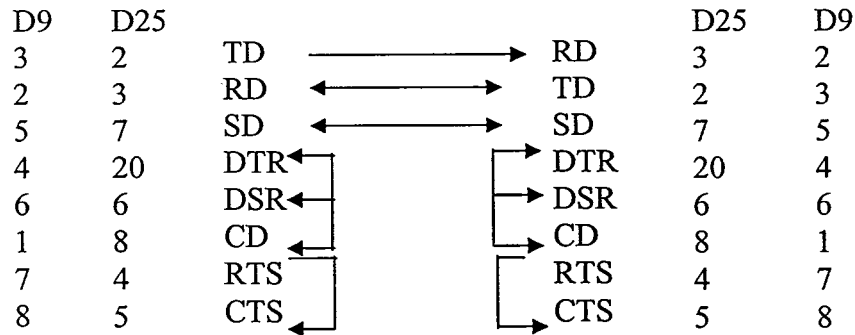


รูปที่ 2.24 แสดงการจัดขาของคอนเน็คเตอร์พอร์ตต่ออนุกรมตามมาตรฐาน RS-232

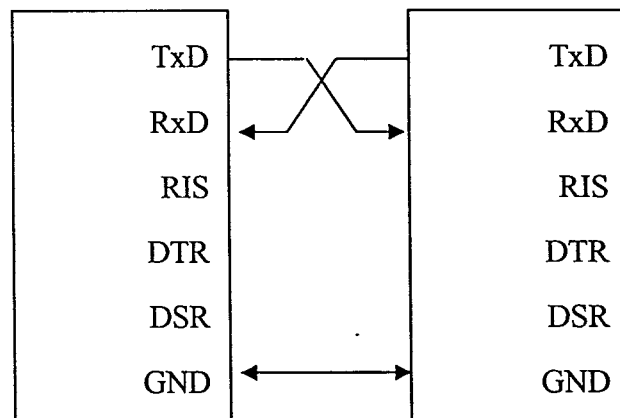
Data Carrier Detect : DCD หรืออาจเรียกว่า Carrier Detect : CD ขานี้จะแอกทีฟเมื่อมีการส่งสัญญาณพาห้จากอุปกรณ์สื่อสารข้อมูลเช่น โมเด็ม สำหรับการใช้งานปกติขานี้จะไม่ได้ถูกใช้งานมากนัก

Receive Data : RD หรือ RxD ขานี้ใช้เพื่อรับสัญญาณอนุกรมเข้ามายังคอมพิวเตอร์โดยนำข้อมูลที่อ่านได้เก็บไว้ในรีจิสเตอร์บัฟเฟอร์

Transmitted Data : TD หรือ TxD ขานี้ใช้เพื่อส่งข้อมูลออกจากคอมพิวเตอร์โดยนำข้อมูลที่เก็บอยู่ในบัฟเฟอร์สำหรับส่งข้อมูลออกไป



(ก) การเชื่อมต่อแบบ Null Modem



(ข) การต่ออุปกรณ์ภายนอกเข้ากับคอมพิวเตอร์แบบ RS-232 โดยใช้สายสัญญาณเพียง 3 เส้น รูปที่ 2.25 แสดงการต่ออุปกรณ์ภายนอกกับพอร์ตอนุกรมของคอมพิวเตอร์ในลักษณะต่างๆ

Data Terminal Ready : DTR เป็นขาสัญญาณที่ส่งออกจากคอมพิวเตอร์ให้อุปกรณ์ปลายทางรับรู้ว่าการติดต่อด้วย โดยขา DTR จะต้องเชื่อมกับขา DSR ของอุปกรณ์ปลายทางและขา DTR ของอุปกรณ์ปลายทางจะต้องเชื่อมกับขา DSR ของคอมพิวเตอร์ถ้าใช้การเชื่อมต่อแบบ Null Modem ซึ่งใช้สายในการเชื่อมต่อเพียง 3 เส้น จะต้องต่อขา DTR และ DSR ของตัวมันเองเข้าด้วยกันและต้องต่อกับขา DCD ด้วยในกรณีที่โปรแกรมสื่อสารที่ใช้ในการตรวจสอบจับสัญญาณพาห้ Signal Ground : GND ขากราวด์ของระบบ

Data Set Ready : DSR ขานี้จะใช้คู่กับขา DTR เพื่อตรวจสอบการเชื่อมต่อกันระหว่างคอมพิวเตอร์กับอุปกรณ์ปลายทางซึ่งขา DSR นี้จะเป็นขาสำหรับรับข้อมูลจากภายนอกซึ่งถูกส่งมาจากขา DTR

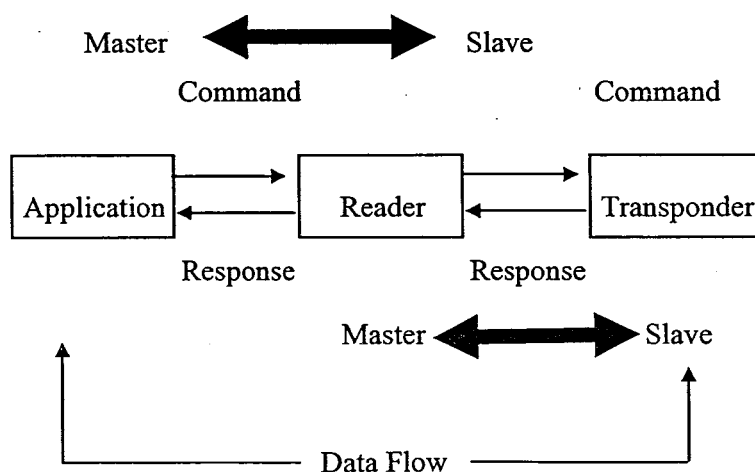
Request to Send : RTS เป็นขาสำหรับส่งสัญญาณร้องขอให้ทางอุปกรณ์ปลายทางส่งข้อมูลกลับมายังคอมพิวเตอร์ โดยขาที่รับสัญญาณ RTS ก็คือขา CTS ในกรณีที่ใช้การเชื่อมต่อแบบ Null Modem ที่ใช้สาย 3 เส้น จะต้องเชื่อมต่อขา RTS และ CTS ของตัวมันเองเข้าด้วยกันเพื่อที่จะให้การรับและส่งข้อมูลสามารถเกิดขึ้นได้ตลอดเวลา

Ring Indicator : RI ใช้แสดงสถานะสัญญาณเรียกจากสายโทรศัพท์ปกติในการสื่อสารโดยทั่วไปสายนี้จะไม่ถูกใช้งานก็ต่อเมื่อมีการเชื่อมต่อกับโมเด็ม

2.5 รูปแบบการทำงาน (หลักการ) ของ Reader

2.5.1 รูปแบบของ Reader

การประยุกต์ใช้งานของซอฟต์แวร์ที่ถูกกำหนดกับการอ่านข้อมูลจาก Tag หรือเขียนข้อมูลลงใน Tag ซึ่งกระบวนการนี้จะเรียกเครื่องอ่าน (Reader) เป็นตัวเชื่อมต่อ (Interface) จากจุดของการ Application Software และกระบวนการข้อมูลควรจะเข้าใจง่ายและเป็นไปได้การเขียนและอ่านจะรวมกับข้อมูล Carrier ที่ถูกกระทำบนพื้นฐานของ Master-Slave ตามรูป 2.26 หมายถึงว่าเครื่องอ่านทั้งหมดและ Tag จะถูกกระตุ้นและถูกรวมเข้าไปใน Application Software และ Application Software จะแสดงเป็น Master ในขณะที่เครื่องอ่าน (Reader) จะเป็น Slave และถูกกระตุ้นเมื่อมีคำสั่งอ่านและเขียนข้อมูลที่รับจาก Application Software เท่านั้น การดำเนินคำสั่งจาก Application Software เริ่มแรกเครื่องอ่านเข้าไปสื่อสารกับ Tag ก่อนและตอนนี้เครื่องอ่านก็จะมีสถานะเป็น Master ที่มีความสัมพันธ์กับ Tag และ Tag ก็จะตอบสนอง (Response) กับคำสั่งจากเครื่องอ่านเท่านั้นตัวอย่างของคำสั่งอ่านจาก Application Software กับเครื่องสามารถรวมเข้ากับการติดต่อสื่อสารกันระหว่างเครื่องอ่านและ Tag ในตัวอย่างตารางที่ 2.3 ตอนแรกคำสั่งอ่านก็จะคำสั่งนี้ไปกระตุ้นกับ Tag และตามด้วยการดำเนินการของขั้นตอนการส่งข้อมูลที่ถูกร้องขอ (Request)

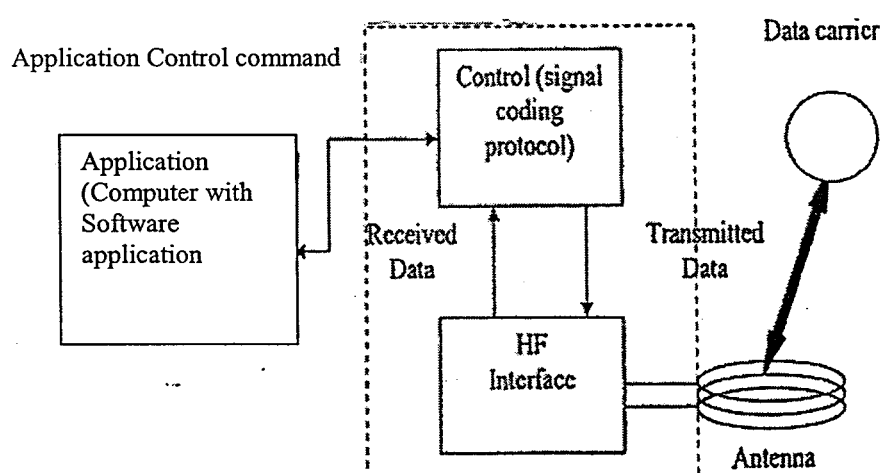


รูปที่ 2.26 Master – Slave ระหว่าง Application Software, เครื่องอ่านและ Tag

ตารางที่ 2.3 ตัวอย่างการดำเนินการอ่านข้อมูลด้วย Application Software เครื่องอ่านและ Tag

Application Reader	Reader Tag	Comment
Block read_Address[00]		Read Tag memory [address[
	Request	Tag operate with serial number Initiate authentication
	ART_SNR[4712] GET Random	Tag operate with serial number Initiate authentication
	Random[081514]	
	SEND_Token1	
	GET_Token2	Authentication successfully completed
	Read_@[00]	Read command [address]
Data[987654321]	Data[987654321]	Data from Tag Data to application

และการเชื่อมโยง (Inductive-electromagnetic) ในขั้นตอนการสื่อสาร (FDX, HDX, SEQ) ขั้นตอนการส่งข้อมูลจาก Tag ไปยังเครื่องอ่าน Reader (Load modulation, backscatter, sub harmonic) ช่วงความถี่และเครื่องอ่านทุกๆ แบบก็มีการกระทำแบบนี้เป็นพื้นฐานในกาออกแบบเครื่องอ่านในระบบทุกระบบสามารถยุบลงหรือลดลงเหลือ 2 ส่วนที่เป็นส่วนสำคัญมีสอง Block คือ ส่วนควบคุมระบบและ HF Interface จะประกอบด้วยในส่วนของ Transmitter และ Receiver ดังแสดงในรูปที่ 2.27



รูปที่ 2.27 Block Diagram ของเครื่องอ่านที่ประกอบด้วยส่วนควบคุมและ HF Interface

2.5.2 HF Interface of Reader

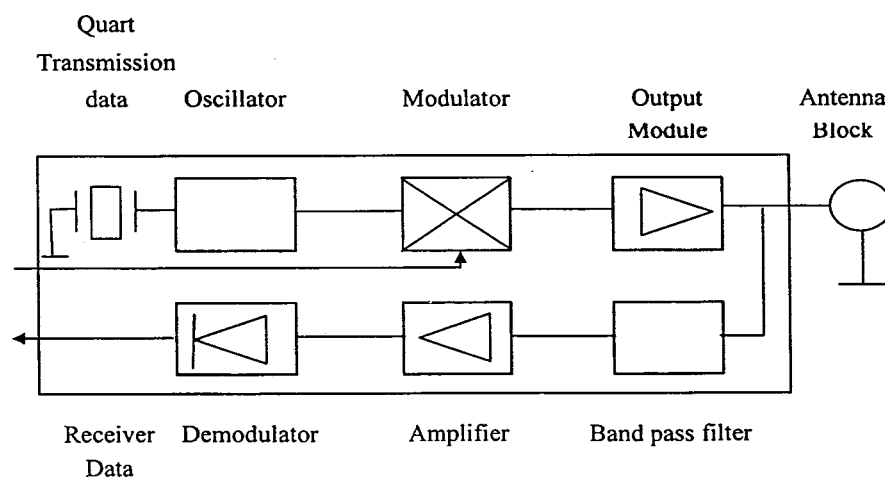
HF Interface ของเครื่องอ่านจะกระทำตามฟังก์ชันตามนี้

- ทำการสร้างกำลังงานความถี่สูงที่จะส่งไปและไปกระตุ้น Tag เพื่อเป็นกำลังงาน และเป็นแหล่งจ่ายให้ Tag
- ทำการ Modulation ในส่วนของสัญญาณข้อมูลที่จะส่งข้อมูลไปให้ Tag
- ทำการรับและ Demodulation ของสัญญาณ HF ที่จะถูกส่งมาโดย Tag

HF Interface จะแยกสัญญาณเป็น 2 ส่วนโดยการพิจารณาของสองทิศทางการส่งและรับข้อมูล จาก Tag ตามรูปที่ 2.28 ข้อมูลจะถูกส่งผ่านไปยัง Tag และเดินทางผ่านในส่วน Transmission Data ในทางกลับกันข้อมูลที่ถูกรับจาก Tag จะเป็นขบวนการใน Receiver Data

2.5.3 Inductively Coupled System, FDX / HDX

สัญญาณที่ต้องการจากคำเนิการของความถี่ 135 KHz หรือ 13.56 MHz จะถูกสร้างใน Transmitter arm ด้วยความถี่ Quartz Oscillator และเพื่อป้องกันสัญญาณ Noise Radio กับสัญญาณที่รับจาก Tag Oscillator จะเป็นส่วนที่ร้องขอการพิจารณาเฟสที่แน่นอนสัญญาณ Oscillator ที่เข้าไปในส่วน ของ Modulation จะถูกควบคุมด้วยสัญญาณ Base Band ของระบบสัญญาณการ Coding และ Base Band Signal นี้จะเป็นตัวถอดรหัสสัญญาณแรงดันใน Binary Data ที่ถูกแสดงเป็น Serial Code (Manchester, Miller, NRZ) ขึ้นอยู่กับชนิดของการ Modulator และ ASK หรือ PSK Modulator จะถูกกระทำบนสัญญาณ Oscillator และสัญญาณที่ถูก Modulated แล้วก็ถูกส่งไปโดย Power output Module และจะถูก Decoupled ไปยัง Antenna Block และในส่วนของ Receiver arm เริ่มต้นที่ Antenna Block และองค์ประกอบแรก Band Pass Filter หรือ Notch Filter ในระบบ FDX/HDX Filter นี้มันก็จะทำการบล็อกสัญญาณที่ส่งมาจาก Out put Module และตอบสนองสัญญาณจาก Tag จะเห็นว่าการส่งและการรับความถี่จะเป็นส่วนที่ แยกต่างหากและในระบบ Load Modulation ในการใช้ Sub Carrier ตัว Filter ควรจะพิจารณาให้ดีเพราะว่า ในส่วนนี้สัญญาณที่ถูกส่งและถูกรับจะถูกแยกคนละส่วนด้วย Sub Carrier Frequency เท่านั้น ชนิด Sub Carrier ในระบบ 13.56 MHz, 847 kHz หรือ 212 kHz



รูปที่ 2.28 Block Diagram ของ HF Interface สำหรับ Inductive Couple

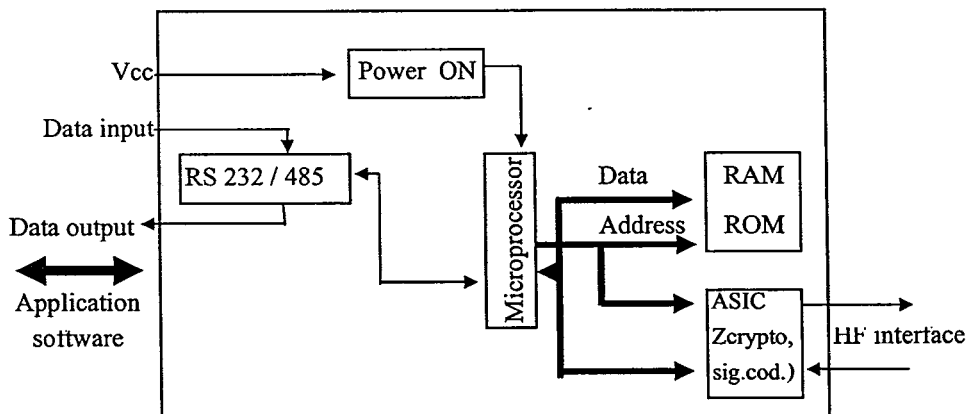
2.5.4 Control Unit

หน่วยควบคุมเครื่องอ่าน (Control Unit) ตามรูปที่ 2.29 จะถูกกระทำตามฟังก์ชันดังต่อไปนี้

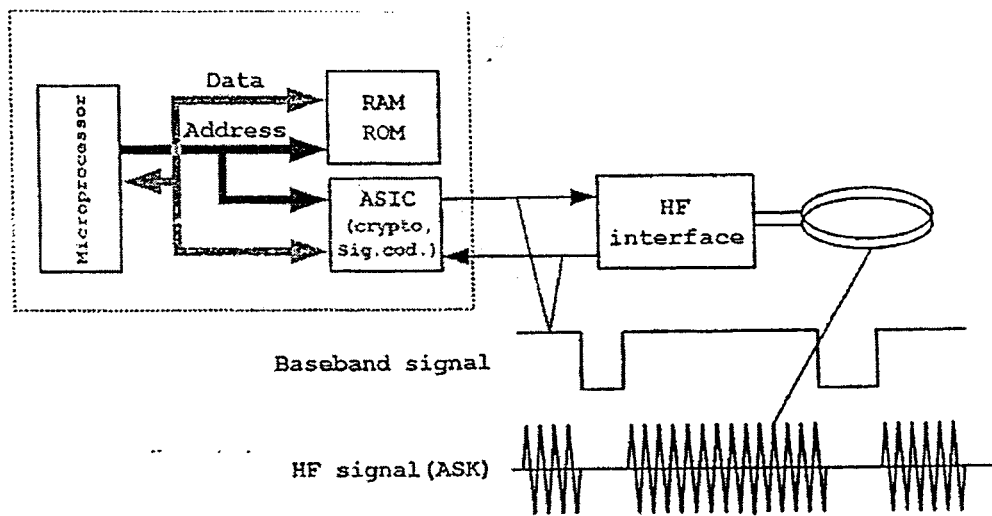
- การสื่อสารกับ Application Software และดำเนินการกับคำสั่งจาก Application Software
- ควบคุมการติดต่อสื่อสารกับ Tag (Master-Slave)
- สัญญา Coding และ Decoding
- ดำเนินการตามกระบวนการ Anti-Collision
- การเข้ารหัสและถอดรหัสของข้อมูลที่ถูกส่งผ่านระหว่างเครื่องอ่าน (Reader) และ Tag
- ตรวจสอบความถูกต้องของข้อมูลที่สื่อสารกันระหว่าง Tag และเครื่องอ่าน (Reader)

Control Unit ปกติมักจะขึ้นอยู่กับการทำงานของฟังก์ชันของไมโครโปรเซสเซอร์และกระบวนการนี้เป็นกระบวนการเข้ารหัสที่ส่งผ่านระหว่าง Tag และเครื่องอ่าน (Reader) ดังนั้นสัญญา Coding มักจะถูกกระทำเข้าไปใน ASI Module กับขบวนการคำนวณของไมโครโปรเซสเซอร์สำหรับเหตุผลของ ASI ที่ถูกส่งเข้าไปใน ไมโครโปรเซสเซอร์ข้อมูลที่ถูกแลกเปลี่ยนระหว่าง Application Software และส่วนหน่วยประมวลผลของเครื่องอ่าน (Reader) ที่ถูกกระทำโดย RS232 หรือ RS485 Interface ซึ่งเป็นพอร์ตที่ใช้กันอยู่ทั่วไป

การ Interface กันระหว่าง HF Interface และ Control Unit จะแสดงสถานะ HF Interface เป็นเลข Binary ในระบบ ASK Modulated ลอจิก "1" ที่มีอคูเลตอินพุทของ HF Interface จะแสดงสถานะของ HF Interface เป็น 'HF Signal On' และลอจิก '0' จะแสดงสถานะเป็น 'HF Signal Off'



รูปที่ 2.29 Block Diagram ของ Control Unit ของ Reader และสื่อสารกับ Application Software



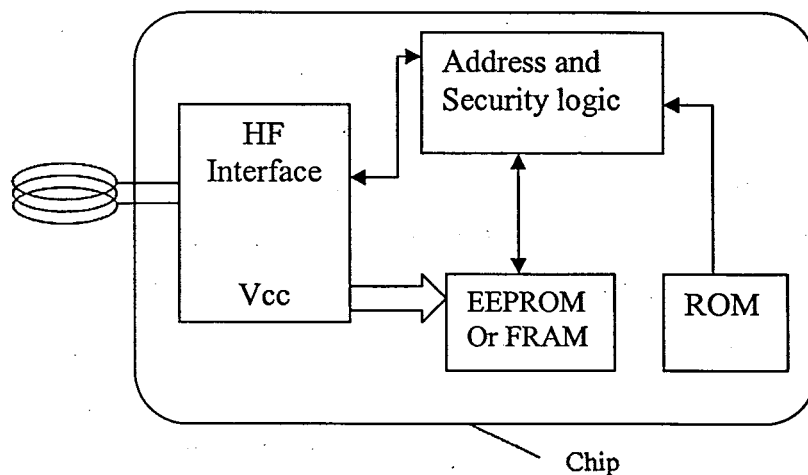
รูปที่ 2.30 สัญญาณ Coding และ Decoding ถูกกระทำโดย Control Unit ของ Reader

2.5.5 สถาปัตยกรรมของ Electronic Data Carrier

ก่อนที่เราจะพิจารณาฟังก์ชันของ Data Carrier ที่ถูกใช้ในระบบ RFID อันดับแรกเราต้องรู้ความแตกต่างระหว่างสองฟังก์ชัน คือ มี Electronic Data Carrier ขึ้นอยู่กับวงจรที่ถูกอินทิเกรต (Microchip) และ Data Carrier คือข้อมูลที่ใช้สำหรับบันทึกข้อมูล

Tag with Function Memory

Tag กับฟังก์ชันหน่วยความจำจาก Tag ที่อ่านอย่างเดียวจนถึง Tag ที่สามารถอ่านและเขียนได้จากฟังก์ชันต่างๆ ตามรูปที่ 2.31 ที่ Tag กับฟังก์ชันหน่วยความจำจะบรรจุ RAM, ROM, EEPROM หรือ FRAM และ HF Interface ที่ใช้เป็นแหล่งจ่ายและยอมให้สื่อสารกับเครื่องอ่าน (Reader)



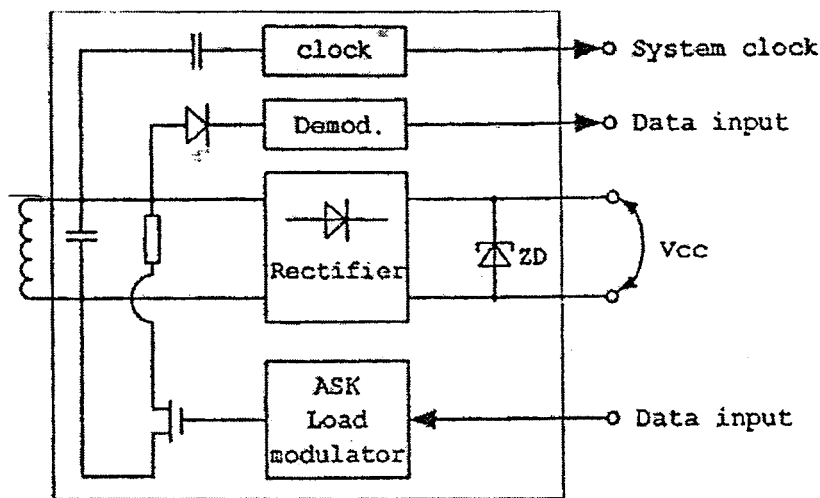
รูปที่ 2.31 Block Diagram ของ RFID Data Carrier ของ Memory Function

2.5.6 HF Interface of Tag

HF Interface เป็นรูปแบบของการ Interface ระหว่าง สัญญาณ Analog ที่ส่งในรูปแบบความถี่สูง จากเครื่องอ่าน (Reader) ไปยัง Tag และวงจร Digital ของ Tag HF Interface ก็กระทำตามฟังก์ชันทั่วไปของ โมเด็มเหมือนโมเด็ม (Modulator – Demodulator) ที่ถูกใช้ในการส่งข้อมูล Analog ในสายโทรศัพท์

สัญญาณ HF ที่ถูกมอดูเลตจากเครื่องอ่าน (Reader) จะถูกสร้างใหม่ใน HF Interface โดยการ Demodulation ก้กับการสร้างสัญญาณข้อมูล Digital ข้อมูลจะผ่านไปสำหรับกระทำการใหม่อีกครั้งใน Address Security Logic การสร้าง Clock – Pulse ระบบวงจรจะสร้าง Clock สำหรับ Data Carrier จากความถี่ Carrier ของ HF field

HF Interface จะถูกรวมเข้าไปใน Load Modulator และถูกควบคุมด้วยข้อมูล Digital ที่ถูกส่งออกไปกับข้อมูลที่ส่งกลับไปให้เครื่องอ่าน (Reader) ตามรูปที่ 2.32

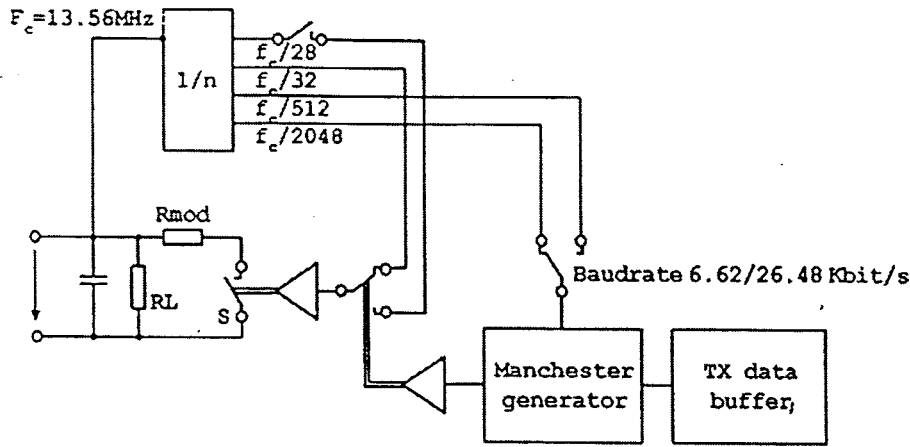


รูปที่ 2.32 Block Diagram ของ HF Interface ของ Inductive Coupled ของ Tag with Load Modulation

Example Circuit – load Modulation with Sub Carrier

วงจรพื้นฐานของ Load Modulation แสดงในรูปที่ 2.33 ส่วนนี้จะเป็นการสร้าง Load Modulation โดยการใช้ ASK หรือ FSK Modulated Sub Carrier ความถี่ของ Sub Carrier และ baud rate จะถูกพิจารณาตามมาตรฐานของ ISO 15693 (Vicinity Coupling Smart Cards) คุณสมบัติความถี่ในมาตรฐานสำหรับ Sub Carrier และ baud rate ที่ได้มาจาก Binary ในส่วนของสัญญาณอินพุต 13.56 MHz ตามตารางที่ 2.4

ข้อมูลจะถูกส่งไปและส่งผ่านไปยัง Manchester Generator การพิจารณานี้ baud rate ของสัญญาณ Base Band Signal ที่จะถูกเปลี่ยนระหว่างค่าสองค่า Manchester Coded ของ Base Band Signal ตอนนี้จะถูกสวิตช์ระหว่าง ความถี่ Carrier สองค่าคือ f_1 และ f_2 ในการใช้ '1' และ '0' ตามระดับของสัญญาณในส่วนของการสร้าง FSK ที่ถูกมอดูเลตของสัญญาณ Sub Carrier ถ้า Clock ของสัญญาณ f_2 ถูก Interface ผลของในส่วนนี้สัญญาณ Sub Carrier ที่ถูกมอดูเลตใน ASK หมายถึงว่าจะมีการสวิตช์ระหว่าง ASK และ FSK Modulation สัญญาณ Sub Carrier ที่ถูกมอดูเลตและตอนนี้ถูกส่งผ่านไปที่สวิตช์ S ดังนั้น Modulation Resistor ของ Load Modulator สามารถที่จะสวิตช์ 'On' และ 'Off' ในเวลาเดียวกันกับความถี่ Sub Carrier



รูปที่ 2.33 การสร้าง Load Modulation กับ Sub carrier Modulated

ตารางที่ 2.4 ความถี่ Clock ที่ต้องการใน HF Interface ถูกสร้างโดย Binary ของสัญญาณ Carrier

13.56 MHz

Splitter N	Frequency	Use
1/28	485 kHz	Q2 of the FSK Sub carrier
1/32	423 kHz	Q1 of the FSK Sub carrier, plus ASK Sub carrier
1/512	26.48 kHz	Bit clock Signal For high baud rate
1/2048	6.62 kHz	Bit clock Signal For high baud rate

2.6 ภาษาคอมพิวเตอร์

ภาษาคอมพิวเตอร์ (Computer Language) เป็นสื่อกลางในการติดต่อสื่อสารระหว่างผู้ใช้กับเครื่องคอมพิวเตอร์ เพื่อให้สามารถทำงานร่วมกันได้อย่างสะดวกราบรื่น โดยชุดคำสั่งสำหรับใช้งานจะควบคุมระบบคอมพิวเตอร์โดยจะถูกเขียนขึ้นจากภาษาคอมพิวเตอร์ในระยะเริ่มต้นของการพัฒนาและการนำคอมพิวเตอร์มาใช้งาน

2.6.1 ภาษาเอชทีเอ็มแอล (HTML)

ไพศาล (2544:25) ได้อธิบายว่าเอชทีเอ็มแอล (HTML) ย่อมาจากคำว่า Hypertext Markup Language เป็นภาษาหลักที่ใช้ในการแสดงผลบนเว็บเบราว์เซอร์ในอินเทอร์เน็ต โดยเราสามารถนำเสนอข้อมูลที่มีตัวอักษร ภาพ เสียง ภาพยนตร์ และสามารถเชื่อมโยงกับเอกสารอื่นๆ ได้ง่าย

เอชทีเอ็มแอล (HTML : Hyper Text Language) เป็นภาษาที่ใช้ในการเขียนเว็บเพจในเวอร์ชันแรกๆ นั้นมีวัตถุประสงค์เพื่อใช้ในการนำเสนอข้อมูลที่เป็นข้อความเสียเป็นส่วนใหญ่ และต่อมาเมื่ออุปกรณ์ต่างๆ เครื่องคอมพิวเตอร์มีราคาถูกลงทำให้มีการใช้มัลติมีเดียมีมากขึ้น ดังนั้นจึงได้มีการพัฒนาเพิ่มขึ้น ในส่วนที่สามารถให้ทำงานกับงานกับรูปภาพและมีลูกเล่นต่างๆ เพิ่มขึ้นมากมายข้อความภายในไฟล์เอชทีเอ็มแอล (HTML) นั้นสามารถแบ่งออกเป็น 2 แบบ คือ

2.6.1.1 ข้อความเนื้อหาที่ผู้เขียนต้องการให้ปรากฏทางหน้าจอ

2.6.1.2 แท็ก (TAG) ใช้เป็นส่วนหนึ่งที่น่าเสนอข้อมูลออกทางหน้าจอ ข้อความส่วนนี้จะไม่ปรากฏออกทางหน้าจอ ข้อความประเภทนี้จะมีเครื่องหมายน้อยกว่า (<) และเครื่องหมายมากกว่า (>) ครอบอยู่ตัวอย่าง <BODY> ข้อมูลเนื้อหา </BODY>

2.6.1.3 ภาษา HTML มีองค์ประกอบอยู่ 2 ส่วนคือ ส่วนที่เป็นเนื้อหาและส่วนที่เป็นคำสั่งก็คือแท็กและในภาษา HTML ก็มีแท็กพื้นฐานในไฟล์ HTML แบ่งออกเป็น 4 ส่วนหลักดังนี้ <HTML>...</HTML> เป็นคำสั่งที่บอกจุดเริ่มต้นและจุดสิ้นสุดของไฟล์ HTML โดยแท็ก<HTML> จำเป็นต้องอยู่บรรทัดแรกและบรรทัดสุดท้ายของไฟล์ HTML

<HEAD>...</HEAD> เป็นคำสั่งที่ทำหน้าที่กำหนดส่วนหัวเรื่องของเว็บเพจ

<TITLE>...</TITLE> จะแสดงชื่อหัวเรื่องของเว็บเพจที่ต้องการตรงไตเติลบาร์ของบราวเซอร์

<BODY>...</BODY> เป็นแท็กที่ใช้แสดงส่วนของเนื้อหาที่จะปรากฏบนเว็บเพจไม่ว่าจะเป็นข้อความ รูปภาพ ตาราง เสียง และการเชื่อมต่อไปยังเอกสารอื่นๆ ที่จะไปแสดงผลที่เว็บเพจบราวเซอร์

2.6.2 เอเอสพีคอทเน็ต (ASP.NET)

ASP.NET เป็นเทคโนโลยีที่สืบเนื่องมาจากเทคโนโลยี ASP ดังนั้นแนวคิดและองค์ประกอบโดยทั่วไปจึงยังคงมีลักษณะเช่นเดียวกับ ASP เว้นเพียงแต่ว่า ASP.NET นี้ ได้นำเอาเทคโนโลยี “.NET Framework” เข้ามาใช้ Web Page ที่พัฒนาขึ้นด้วยเทคโนโลยี ASP.NET จะจัดเก็บอยู่ในไฟล์ .aspx แต่ก็ยังคงมีขั้นตอนการทำงานเมื่อถูกเรียกใช้งานเช่นเดียวกับ Web Page ที่พัฒนาขึ้นด้วยเทคโนโลยี ASP เพียงแต่ใน ASP.NET ได้มีการทำให้ Web Page มีการทำงานที่แบ่งออกเป็นสถานะ ซึ่งจะทำได้ว่า การเรียกใช้ Page ในครั้งนั้น เป็นการเรียกใช้ Page ในครั้งแรก หรือเป็นการเรียกใช้ที่สืบเนื่องมาจากการที่ถูก Submit มา รวมทั้งนำเอา ASP.NET Server Control ที่สามารถจดจำค่าที่กำหนดให้กับ Control ที่เกิดขึ้นในคราวก่อนหน้าไว้ และสามารถตอบสนองต่อเหตุการณ์ต่างๆ ที่เกิดขึ้น ณ Web Service เข้ามาใช้แทนที่ Element ที่สร้างขึ้นจาก HTML Tag ด้วยเหตุนี้ Web Page ที่พัฒนาขึ้นด้วยเทคโนโลยี ASP.NET จึงสามารถแก้ไขปัญหา Stateless ที่เกิดขึ้นกับ Web Page ที่พัฒนาขึ้นด้วยเทคโนโลยี ASP เนื่องจากเมื่อถูกเรียกใช้ จะมีขั้นตอนการทำงานที่แตกต่างกันไปตามสภาพแวดล้อมที่เกิดขึ้น ไม่ใช้มีการทำงานที่เหมือนกันทุกครั้งเช่นเดียวกับ Web Page ที่พัฒนาขึ้นด้วยเทคโนโลยี ASP

นอกเหนือจากคุณสมบัติดังที่ได้กล่าวมาข้างต้นแล้ว ASP.NET ยังได้มีการแก้ไขปรับปรุงและเพิ่มเติมความสามารถด้านต่างๆ เข้าไป เพื่อให้ Web Page ที่ได้มีการทำงานที่มีประสิทธิภาพมากขึ้น เช่น การทำให้ Web Page ที่พัฒนาขึ้นด้วยเทคโนโลยี ASP.NET อยู่ในรูปของ Web Form ที่ถูกแบ่งออกเป็น 2 ส่วน คือ ส่วนของ Tag ต่างๆ ที่ใช้กำหนดจอภาพขึ้นใช้งาน และส่วนของโปรแกรมที่ใช้ประมวลผลและควบคุมการทำงานส่วนประกอบต่างๆ ของจอภาพ ซึ่งมีลักษณะคล้ายกับเครื่องมือที่ใช้ในการพัฒนาโปรแกรมควบคุมการทำงานส่วนประกอบต่างๆ ของจอภาพ ซึ่งมีลักษณะคล้ายกับเครื่องมือที่ใช้ในการพัฒนาโปรแกรมอย่างเช่น Visual Basic C++

สามารถใช้งานร่วมกับโปรแกรม Browser ได้ทุกประเภท เนื่องจากคำสั่งต่างๆ ที่กำหนดขึ้นใน Web Form จะถูกแปลงไปเป็น HTML Tag ที่เหมาะสมกับโปรแกรม Browser ที่ใช้โดยอัตโนมัติ ซึ่งต่างจาก ASP ในรูปแบบเดิมที่บางคำสั่งไม่สามารถนำไปใช้งานในบาง Browser ได้

สามารถใช้ส่วนต่างๆ ของโปรแกรม Internet Explorer Version 6 ได้อย่างมีประสิทธิภาพมากกว่า Form ในรูปแบบเดิม เนื่องจาก Web Form ได้ถูกออกแบบมาสำหรับโปรแกรม Internet Explorer Version 6 โดยเฉพาะรองรับการทำงานร่วมกับโปรแกรม Script ที่พัฒนาขึ้นจากภาษาที่ใช้เทคโนโลยี .NET เช่น ภาษา VB.NET, C#, Jscript เป็นต้น

มีการแยกส่วนของโปรแกรมและคำสั่งที่ใช้สร้างจอภาพออกจากกัน จึงช่วยทำให้โครงสร้างแบบ “spaghetti code” ที่พบใน ASP หดไปสามารถนำไปใช้ร่วมกับ Third-party Control ต่างๆ สิ่งที่ต้องใช้ในการสร้างและ Run ASP.NET มีดังนี้

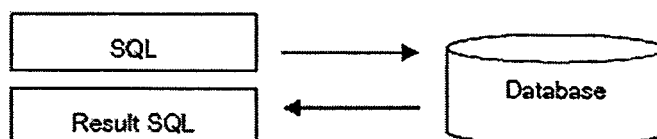
2.6.2.1 Web Server ลำดับแรกสุดที่จะต้องเตรียมในการนำ ASP.NET มาใช้งาน ได้แก่ การติดตั้ง Web Server ให้กับเครื่องของเรา สำหรับโปรแกรมที่ใช้ติดตั้ง Web Server นี้มีด้วยกันหลายโปรแกรมเช่น IIS

2.6.2.2 NET Framework ได้แก่ Component ต่างๆ ที่จะต้องนำมาใช้ประกอบกับ Web Server เพื่อใช้ในการ Run ASP.NET ซึ่งจะปรากฏอยู่ในเครื่องโดยอัตโนมัติหลังจากที่ติดตั้งโปรแกรม Microsoft Visual Studio.NET 7.0 โปรแกรม Browser ได้แก่ โปรแกรมที่ใช้สำหรับนำ ASP.NET Page ไป Run และแสดงผลลัพธ์ที่ได้สำหรับ โปรแกรม Browser ที่ใช้ได้แก่ Internet Explore 6

2.6.3 ซีเควล (SQL : Structured Query Language)

ศิริลักษณ์ (2538) ได้อธิบายเรื่องเทคโนโลยีซีเควล (SQL) ไว้คือซีเควล (SQL : Structured Query Language) เป็นภาษาคอมพิวเตอร์อีกภาษาหนึ่งที่ใช้จัดการฐานข้อมูล(Query Language) ซึ่งเป็นที่นิยมมากที่สุดในปัจจุบันนี้ เนื่องจากโปรแกรมทางด้านการจัดการฐานข้อมูลมีอยู่หลายโปรแกรมด้วยกัน ซึ่งแต่ละโปรแกรมจะมีชุดคำสั่งในการติดต่อฐานข้อมูลของแต่ละโปรแกรมเอง ซึ่งถ้าผู้พัฒนาไม่ทราบถึงชุดคำสั่งในการติดต่อฐานข้อมูล นั่นก็หมายความว่า ไม่สามารถติดต่อกับฐานข้อมูลได้เลย ต้องมาศึกษาค้นคว้าเพิ่มเติมกันใหม่ภาษา SQL เป็นชุดคำสั่งที่ทุกโปรแกรมทาง ด้านการจัดการฐานข้อมูลสามารถที่จะทำงานได้ และยังเป็นภาษาที่มีความใกล้เคียงกับภาษาที่มนุษย์ใช้อยู่ทุกวัน

หลักของการใช้ภาษาซีเควล (SQL) คือเป็นภาษาที่ไม่เป็นกระบวนการ (Nonprocedural Language) เป็นภาษาที่มีโครงสร้างง่ายๆ เหมาะสำหรับการวิเคราะห์ข้อมูล การทำรายงานอย่างง่าย ผู้ใช้โปรแกรมจะใช้คำสั่ง เพื่อถามว่าจะทำอะไร และไม่จำเป็นต้องอธิบายว่าทำอะไร นอกจากนั้นผู้ใช้ระบบและโปรแกรมเมอร์ทั้งหลาย ไม่จำเป็นต้องทราบถึงกระบวนการจัดเก็บและรูปแบบของข้อมูลที่เก็บก็สามารถเขียนคิวรี (Query) ได้ดังภาพ 2.34



ภาพที่ 2.34 การติดต่อกับฐานข้อมูลโดยใช้ซีเควล (SQL)

ประเภทของภาษา SQL สามารถแบ่งออกเป็นประเภทต่างๆ ได้ดังนี้

2.6.3.1 Data Manipulation Language (DML) เป็นกลุ่มคำสั่งที่ใช้สำหรับจัดการเก็บข้อมูลใน Relation ต่างๆ เช่น การเพิ่ม การลบ หรือเปลี่ยนแปลงข้อมูลในฐานข้อมูล คำสั่งในกลุ่มนี้ประกอบด้วยคำสั่งที่ใช้เพิ่มข้อมูลให้กับ Relation คำสั่งที่ใช้เพิ่มข้อมูลให้กับ Relation จาก Relation อื่น คำสั่งที่ใช้เปลี่ยนค่าของข้อมูลใน Relation คำสั่งที่ใช้ในการลบข้อมูลออกจาก Relation

2.6.3.2 Data Definition Language (DDL) เป็นกลุ่มคำสั่งที่ใช้สำหรับสร้าง กำหนดเพิ่ม เปลี่ยนแปลง และลบ โครงสร้างฐานข้อมูล และ Table คำสั่งในกลุ่มนี้ประกอบด้วย คำสั่งที่ใช้ลบ Domain คำสั่งที่ใช้กำหนด Domain ให้กับข้อมูล คำสั่งที่ใช้กำหนด PK คำสั่งที่ใช้ลบ Domain คำสั่งที่ใช้กำหนด FK คำสั่งที่ใช้ลบ Relation(table) คำสั่งที่ใช้สร้าง Index คำสั่งที่ใช้แก้ไขโครงสร้าง ของ Attribute คำสั่งที่ใช้ลบ Index

2.6.3.3 Data Query Language (DQL) เป็นกลุ่มคำสั่งที่ใช้ในการเลือกข้อมูลจาก Relation (table) ต่างๆ ออกมาแสดงผลในรูปแบบที่ตามต้องการ ตามโครงสร้างจะแบ่งออกเป็น 5 ส่วนหลักๆ ดังนี้

ก) Select เป็นคำสั่งใช้สำหรับกำหนดเลือก Attribute(Field) ที่ต้องการให้แสดงผล

ข) From เป็นคำสั่งใช้สำหรับกำหนดเลือก Relation(Table) ที่เป็นแหล่งของข้อมูลที่ต้องการให้แสดงผล

ค) Where เป็นคำสั่งใช้สำหรับกำหนดเงื่อนไขในการเลือกข้อมูลจาก Relation(Table) จะกำหนดหรือไม่ก็ได้

ง) Group By เป็นคำสั่งใช้จัดกลุ่มข้อมูลจาก Relation(Table) จะกำหนดหรือไม่ก็ได้

จ) Order By เป็นคำสั่งใช้สำหรับเรียงลำดับข้อมูลจาก Relation(Table) จะกำหนดหรือไม่ก็ได้

2.7 เว็บแอปพลิเคชัน (Web Application)

เว็บแอปพลิเคชัน (Web Application) คือ โปรแกรมที่อยู่ในเว็บ เซิร์ฟเวอร์ (Web Server) ที่คอยให้บริการ สิ่งที่ต้องการ (Request) จากทางไคลเอนต์ ผ่าน Protocol HTTP และแสดงผลการร้องขอ ในรูปแบบ HTML Page ผ่านทางบราวเซอร์ ซึ่งก็คือเว็บไซต์ต่างๆที่เราใช้บริการนั่นเอง

เว็บแอปพลิเคชัน สามารถตอบสนองแนวคิด Distributed Processing ได้ในระดับหนึ่ง ซึ่งก็คือ การแบ่งการประมวลผลไว้ที่ฝั่งไคลเอนต์และเซิร์ฟเวอร์ และมักจะมีการใช้ฐานข้อมูลควบคู่กับการ ทำเว็บแอปพลิเคชันไปด้วย

รูปแบบและสถาปัตยกรรมของเว็บแอปพลิเคชันจะไม่เหมือนกับการพัฒนาแอปพลิเคชันบน Desktop หรือ Client-Server Application ทั้งนี้เว็บแอปพลิเคชันจะประกอบไปด้วยองค์ประกอบ หลายๆ ส่วนที่จะทำงานร่วมกัน

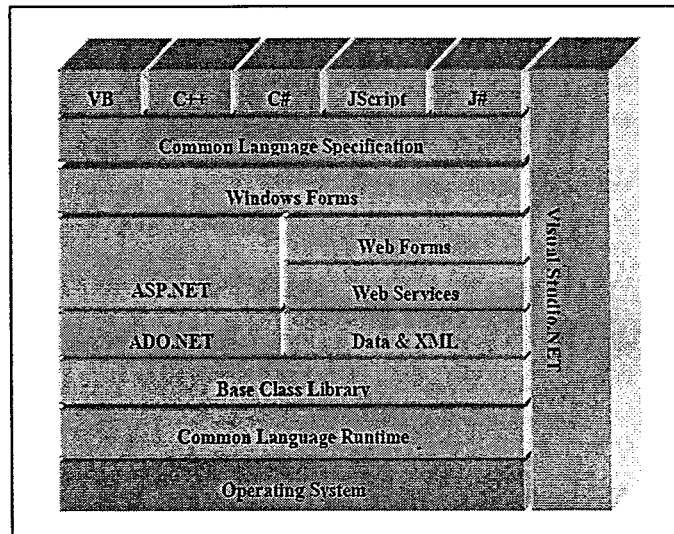
สถาปัตยกรรมขั้นพื้นฐานในการพัฒนาเว็บแอปพลิเคชันทั่วไป และเป็นสถาปัตยกรรมเดียวกันกับการพัฒนาระบบในรูปแบบ Distributed Application หรือบางครั้งเราเรียก แอปพลิเคชันรูปแบบ นี้ว่า

Multi-Tiers, n-Tiers หรือ 3-Tiers แอปพลิเคชัน สำหรับ เว็บแอปพลิเคชันขนาดใหญ่ องค์ประกอบอาจมีมากกว่าหรือซับซ้อนกว่า สถาปัตยกรรมนี้จะช่วยในการออกแบบและพัฒนา เว็บแอปพลิเคชันที่ต้องรองรับในหลายด้าน เช่น ความยืดหยุ่น (Scalability), ประสิทธิภาพ (Performance/Efficiency), ความพร้อมในการใช้งาน (Availability), ความเชื่อถือได้ (Reliability) ความสามารถที่จะให้บริการ (Serviceability), การทำงานร่วมกับ แอปพลิเคชันอื่นๆ (Interoperability) และ การดูแลจัดการ (Manageability) เป็นต้น ซึ่งจะทำได้สามารถจัดการกับ แอปพลิเคชัน ได้อย่างถูกต้อง

2.8 ASP.NET

2.8.1 .NET Framework

เกิดจากการที่ไมโครซอฟท์ต้องการที่จะสร้างมาตรฐานขึ้นมา เพื่อให้ทุกสิ่งทุกอย่างสามารถทำงาน และติดต่อสื่อสารกันได้หมด โดยคิดค้นระบบที่ต้องการให้ได้มาตรฐาน ระบบนี้คือ .NET Framework ซึ่งระบบนี้ไม่ใช่ระบบปฏิบัติการ (OS) แต่เปรียบเสมือนโปรแกรมหนึ่งที่จะสามารถ สร้างสถานะแวดล้อมที่ซึ่งสามารถทำงานในระบบ .NET นี้ได้ แสดงดังภาพที่ 2.35



รูปที่ 2.35 โครงสร้างสถาปัตยกรรมของ .NET

จากภาพ เป็นการแสดงถึงสถาปัตยกรรมของแอปพลิเคชัน .NET ที่พัฒนาด้วย Visual Studio .NET ถือว่าเป็นการเปลี่ยนแปลงครั้งสำคัญ ทั้งแพลตฟอร์มที่ใช้ในการพัฒนาแอปพลิเคชัน และสถาปัตยกรรมที่ใช้ โดยมีเลเยอร์ล่างสุดคือ .NET Framework SDK เปรียบเสมือนกับ Runtime Library ที่รันอยู่และคอยสนับสนุนการทำงานของแอปพลิเคชัน จากนั้นเป็นเลเยอร์ของ Common Language Runtime เป็นผลลัพธ์ของการคอมไพล์แอปพลิเคชัน .NET เลเยอร์ถัดขึ้นมา เป็นเครื่องมือ (Tools) และเทคนิคต่าง ๆ ที่สามารถใช้พัฒนาแอปพลิเคชันได้ทั้งในเรื่องของ เว็บเซอร์วิส ADO.NET และ ASP.NET จนกระทั่งถึงเลเยอร์บนสุด คือภาษาที่ใช้ในการพัฒนา แอปพลิเคชันด้วย Visual Studio.NET

ในอนาคตไมโครซอฟท์ต้องการที่จะนำเอาระบบนี้ไปติดตั้งลงบนอุปกรณ์ทุกชนิด เพื่อให้
อุปกรณ์ทุกอย่างมีระบบๆ หนึ่งๆ ที่เหมือนกันหมดโดย .NET Framework มีส่วนประกอบภายใน แบ่ง
ออกเป็น 3 ชั้นใหญ่ๆ คือ

2.8.1.1 Programming Language เป็นรูปแบบของภาษาที่ออกแบบมา เพื่อให้สามารถ
ทำงานในสถานะที่เป็น .NET ได้ โดยที่ทางไมโครซอฟท์ได้เปิดตัวภาษาหลัก ที่จะใช้พัฒนาบน .NET นี้ 3
ภาษา ได้แก่

2.8.1.1.1 C# เป็นภาษาใหม่ที่ไม่โครซอฟท์พัฒนามาจาก C++ กับ JAVA

2.8.1.1.2 VB.NET เป็นภาษาที่พัฒนามาจาก Visual Basic ในเวอร์ชัน 6.0

2.8.1.1.3 JScript.net เป็นภาษาที่พัฒนามาจาก JScript ซึ่งเป็น JavaScript ในเวอร์ชัน
ของไมโครซอฟท์

2.8.1.2 Base Classes Library โดยที่ไลบรารี (Library) เปรียบเสมือนชุดคำสั่งสำเร็จรูป ย่อยๆ
ที่เพิ่มเข้ามา ซึ่งส่วนใหญ่จะเป็นชุดคำสั่งที่ต้องใช้งานอยู่เป็นประจำ ดังนั้น จึงมีผู้คิดค้น เพื่ออำนวยความสะดวก
สะดวกในการเขียนโปรแกรม ซึ่งไลบรารีในภาษาต่างๆ ส่วนใหญ่จะอยู่ในรูปแบบ ไฟล์ Incould แต่ถ้า
เป็น ASP สิ่งที่เป็นไลบรารี คือคอมโพเนนต์ต่างๆ นั่นเอง ซึ่งภายในระบบ .NET จะสร้างสิ่งที่เรียกว่าเป็น
ไลบรารีพื้นฐานขึ้น ทำให้ไม่ว่าจะใช้ภาษาใดในการ พัฒนาโปรแกรม ก็สามารถที่จะเรียกใช้ไลบรารีที่เป็น
ตัวเดียวกันได้หมด

2.8.1.3 Common Language Runtime (CLR) เป็นสิ่งสำคัญแทบจะที่สุด ของระบบ .NET
เพราะ CLR มีหน้าที่ทำให้โปรแกรมที่เขียนขึ้นมาด้วยภาษาต่างๆ กัน กลายเป็นภาษารูปแบบ มาตรฐาน
เดียวกันทั้งหมด เราเรียกภาษานี้ว่า Intermediate language (IL) ซึ่งเมื่อต้องการที่จะรัน โปรแกรมใด CLR
ที่ว่าจะตรวจสอบเครื่องที่รันว่ามีสถานะแวดล้อมการทำงานเช่นใด จากนั้น จะคอมไพล์เป็น โปรแกรมที่
เหมาะสมต่อการทำงานของเครื่องนั้น ทำให้สามารถใช้งานโปรแกรม ต่างๆ ได้อย่างมีประสิทธิภาพสูงสุด
ในแต่ละเครื่อง

2.8.2 ประโยชน์และข้อดีของ .NET Framework

2.8.2.1 เป็นระบบที่มีไลบรารีที่เป็นมาตรฐานเดียวกัน เนื่องจากมีไลบรารีที่เป็นมาตรฐาน
เดียวกันทั้งหมดทำให้เราไม่ต้องกังวลว่าภาษาที่ใช้เขียนนั้น มีไลบรารีตัวที่ต้องการหรือไม่ รวมทั้งไม่ต้อง
กังวลว่าถ้าใช้ไลบรารีของภาษาหนึ่งแล้วอีกภาษาหนึ่งจะไม่มีไลบรารีตัวนั้น

2.8.2.2 ไม่ขึ้นกับระบบปฏิบัติการ (Operating System) เนื่องจากระบบปฏิบัติการ ที่แต่ละ
บุคคล หรือองค์กรใช้นั้นย่อมไม่เหมือนกัน แต่ภายใน .NET Framework จะไม่มีปัญหานี้ เพียงแค่มีระบบ
.NET Framework จะทำให้สามารถใช้งานโปรแกรมต่างๆ ได้ ซึ่งเป็นข้อดีตรงที่ เราจะสามารถใช้
โปรแกรมต่างๆ ได้ทุกระบบปฏิบัติการ

2.8.2.3 ใช้ในการพัฒนาได้ทุกภาษา ทำให้เราไม่ต้องคอยมาศึกษาภาษาใหม่ๆ เมื่อต้องการ
สร้างโปรแกรมในแต่ละครั้ง นอกจากนั้น เรายังสามารถเลือก ใช้ภาษาที่เราถนัดที่สุด ในการพัฒนา
โปรแกรมต่างๆ ได้อีกด้วย

2.8.2.4 มีการควบคุมสิ่งแวดล้อมในการทำงานเป็นอย่างดี เนื่องจากเป็นระบบที่เป็นมาตรฐานทำให้การควบคุมจัดสรรระบบต่างๆ ทำได้ง่ายขึ้น ไม่ว่าจะเป็นการจัดสรรหน่วยความจำ ด้านการใช้งานเครื่องมีความรวดเร็วมากขึ้น ทำให้ลดโอกาสที่เครื่องจะเสียหายได้เป็นอย่างดี

2.8.2.5 ความปลอดภัยที่มีมากขึ้น .NET Framework สามารถกำหนดสิทธิ์การใช้งานหรือ Permission ของผู้ใช้งานได้มากขึ้น ทำให้สามารถกำหนดว่าจะให้โปรแกรมในส่วนใด ใช้งานได้หรือไม่ได้แล้วแต่เฉพาะบุคคล

2.8.3 ภาษา ASP.NET

ภาษา ASP.NET พัฒนามาจากภาษา ASP เพราะรูปแบบ ไวยากรณ์ต่างๆ และภาษาที่นำมาใช้งานนั้นต่างจากเดิมแทบทั้งสิ้น สามารถพูดได้ว่า ASP.NET เป็นอีก Generation หนึ่งของ ASP มากกว่า ASP.NET มีความแตกต่างจาก ASP รุ่นก่อน อย่างไร

2.8.3.1 ใช้ภาษาใดๆ ในการเขียนสคริปต์ก็ได้ จากเดิมที่สามารถใช้ได้เฉพาะภาษาที่เป็นสคริปต์ของ VBScript และ JScript แต่ใน ASP.NET สามารถที่จะใช้ภาษาที่มีรูปแบบของภาษา เต็มๆ ซึ่งในเบื้องต้นมี 3 ภาษาคือ C#, VB.NET และ JScript.Net ที่ออกมาเป็นมาตรฐาน แต่ในอนาคตไมโครซอฟท์มีแผนที่จะเพิ่มตัวแปลภาษาให้ครบทุกภาษา

2.8.3.2 มีความยืดหยุ่นในการเขียนโปรแกรมมากขึ้น โดยที่เราสามารถใช้ภาษาในการเขียน ASP.NET ได้มากกว่า 1 ภาษาภายในไฟล์เดียวกัน ทำให้สามารถเลือกรูปแบบ ของภาษาที่ง่ายที่สุดต่อการเขียนในแต่ละส่วนได้

2.8.3.3 ลักษณะการแปลภาษา และนามสกุลไฟล์เปลี่ยนไป ในภาษา ASP เวอร์ชันก่อนๆ มีลักษณะการแปลภาษาเป็นแบบอินเตอร์พรีเตอร์ (Interpreter) คือ การจะทำคำสั่งใดค้อยแปลคำสั่ง นั้น แต่ในเวอร์ชัน .NET นี้จะมี ลักษณะเป็นคอมไพเลอร์ (Compiler) คือการแปลคำสั่งรวม ทั้งโปรแกรม นอกจากนี้นามสกุลของไฟล์ก็มีการเปลี่ยนแปลงจากเดิมที่ใช้นามสกุลไฟล์เป็น "*.asp" เป็น "*.aspx"

2.8.3.4 รูปแบบ และการใช้งานคอมโพเนนต์ที่ง่ายขึ้น รูปแบบของคอมโพเนนต์ จะเน้นที่ XML มากที่สุด และที่สำคัญคือการใช้งานคอมโพเนนต์ใน ASP.NET นั้นเราสามารถ อัปโหลดไฟล์ไปไว้ในไคลเอนต์ที่ผู้ดูแลเซิร์ฟเวอร์ (Admin) กำหนด หลังจากนั้นคอมโพเนนต์ จะติดตั้งตัวเองโดยอัตโนมัติ ปัญหาที่เกิดจาก ASP เวอร์ชันก่อนๆ ได้เป็นอย่างดี เนื่องจากใน ASP เวอร์ชันก่อนนั้น การติดตั้งคอมโพเนนต์กระทำได้เพียงผู้ดูแลเซิร์ฟเวอร์ เพียงคนเดียวเท่านั้น ทำให้เวลาต้องการใช้คอมโพเนนต์อื่นที่ในเซิร์ฟเวอร์ไม่มีจึงเป็นเรื่องที่ลำบาก

2.8.3.5 มีไลบรารีให้เลือกใช้ได้มากขึ้น ใน ASP เวอร์ชันก่อนๆ นั้น แอปพลิเคชัน บางอย่าง สร้างได้ไม่สะดวกนัก ต้องอาศัยคอมโพเนนต์ต่างๆ มากมาย แต่ใน ASP.NET นั้นได้เพิ่ม ไลบรารีในส่วนเหล่านี้ให้กลายเป็นพื้นฐานของการใช้งาน

2.8.3.6 มีคอนโทรลทำให้การใช้งานในบางสิ่งง่ายขึ้น เป็นส่วนพิเศษที่เพิ่มเติมมาจาก ASP รุ่นก่อนๆ ที่ไม่มีส่วนที่เรียกว่า คอนโทรล ซึ่งคอนโทรลนี้จะช่วยให้เราสามารถสร้างเว็บ ไซด์ ได้อย่างง่าย และมีประสิทธิภาพมากขึ้น จึงไม่ต้องกังวลว่าบราวเซอร์รุ่นใดๆ ก็ตามจะรองรับกับ ภาษาที่เราเขียนหรือไม่

2.8.3.7 สามารถเรียกขอข้อมูลจากเซิร์ฟเวอร์ได้ ใน ASP เวอร์ชันก่อนๆ เซิร์ฟเวอร์ สามารถเรียกขอข้อมูลได้จากเครื่องผู้ใช้นั้น แต่ใน ASP.NET เครื่องเซิร์ฟเวอร์สามารถเรียกขอ ข้อมูลจากเครื่องเซิร์ฟเวอร์ด้วยกันได้

2.8.3.8 ไม่ต้องต่อ Hardware เนื่องจากเป็นระบบใน .NET Framework ดังนั้นจึงมี คุณสมบัติของ Common Language Runtime (CLR) ทำให้มีการคอมไพล์โปรแกรมเป็นภาษา มาตรฐานที่เรียกว่า IL ก่อน ดังนั้น ไม่ว่าจะเล่นเครื่องปาล์มหรือโน้ตบุ๊ก PDA ก็ไม่เกิดปัญหา

2.8.3.9 ง่ายต่อการหาจุดผิดพลาดในการเขียนโปรแกรม หากเป็น ASP รุ่นก่อนเมื่อเกิด ความผิดพลาด (Error) เครื่องจะบอกแต่ว่าเป็นความผิดพลาดชนิดใดบรรทัดไหน แต่ใน ASP.NET นี้เครื่องจะแสดงรายละเอียดที่มากขึ้นพร้อมกับแนวทางแก้ไข

2.8.3.10 มีการตรวจสอบเหตุการณ์ต่างๆ ได้ภายในเว็บเพจ มีการตรวจสอบเหตุการณ์ ต่างๆ ตั้งแต่โหลดหน้าเว็บเพจไปจนถึงปิดหน้าเว็บเพจลง ทำให้สามารถเขียน โปรแกรมกำหนด เหตุการณ์ได้ง่ายขึ้น

2.8.3.11 แยกส่วนที่เป็น HTML กับ ASP ออกมาอย่างชัดเจน ในเวอร์ชันก่อนๆ ส่วนที่เป็น HTML กับ ASP จะเขียนปนกันไปมา แต่ในเวอร์ชันนี้จะแยกส่วนกันอย่างชัดเจน ว่าส่วนไหนเป็น HTML และส่วนไหนเป็น ASP

2.9 Object Oriented Programming (OOP) หรือ การออกแบบและพัฒนาโปรแกรมเชิงวัตถุ

โดยหลักการคือ มองสิ่งต่างๆ ในโปรแกรมเป็นวัตถุ และยังสามารถนำวัตถุดังกล่าว ไปใช้ได้ ที่โปรแกรมอื่น ที่การทำงานเหมือนกัน ทำให้ไม่จำเป็นต้องเขียนโค้ดขึ้นมาใหม่ทั้งหมด

โดยการสร้างวัตถุนั้น เราจะต้องมีต้นแบบของวัตถุนั้นเสียก่อน โดยที่ ต้นแบบเราจะเรียกว่า คลาส (class)

โดยคลาสนั้น เราไม่สามารถนำมาใช้งานได้โดยตรง แต่ต้อง ใช้คลาส สร้าง object ออกมาก่อน ยกตัวอย่าง ถ้าเปรียบเทียบ class เป็นแม่พิมพ์ลูกก็แล้ว object คือ ลูกก็ ที่แม่พิมพ์ พิมพ์ออกมา ดังนั้น คงไม่มีใคร ไปกินแม่พิมพ์โซ่ใหม่ แต่ลูกก็กินได้ อร่อยมากด้วย OOP ก็เช่นกัน เราต้องใช้ class แล้ว ป้อนออกมา เป็น object ก่อน แล้วจึงนำ object ไปใช้งาน .โดยออบเจกต์ต่างๆ จะมี property และ method เป็นส่วนประกอบ property คือ คุณสมบัติของมัน และ method คือสิ่งที่มันทำได้

ยกตัวอย่างเช่น Property คลาสของ มนุษย์ มี property คือ มี แขน ขา ลำตัว หัว นี่ก็คุณสมบัติ object ทุกอันที่เกิดจากคลาส มนุษย์ ต้องมี Method คลาสของ มนุษย์ สามารถทำกิริยาต่อไปนี้ เดิน วิ่ง กิน พูด มองจะเห็นว่าถ้าหาก class ที่ไม่มี method ก็แทบจะไม่มีประโยชน์เลย เช่น มนุษย์ ที่ ทำอะไรไม่ได้ เลย ก็ไม่มีประโยชน์ (คือไม่มีเมธอดทำอะไรเลย แม้แต่ เมธอด นอน)

ลำดับการเขียนโปรแกรม แบบ OOP สามารถเรียงได้ดังนี้

- 1.ประกาศสร้าง คลาส
- 2.นำคลาส มาป้อน เป็น object
- 3.นำออบเจกต์ไปใช้งาน

ข้อดีอีกข้อของ OOP คือ คลาสต่างๆ เราไม่จำเป็นต้องรู้เลยว่า คลาสนั้น ประกอบไปด้วยอะไรบ้าง เพียงแค่รู้ว่า เมื่อนำคลาสนั้นมาสร้าง object แล้วใช้งาน method ของมัน จะเกิดอะไรขึ้น เช่น เราไม่จำเป็นต้องรู้ว่า คลาส มนุษย์ มีลำไส้ยาวกี่เมตร มีสมองหยักมาก หยักน้อยแค่ไหน แต่เรารู้กันอยู่ถ้าเราใช้เมธอด "กิน" ก็จะเกิดการอ้วม เป็นต้น

ข้อดีนี้มีตัวอย่างน้อยก็ 2 ข้อ

1. เมื่อเขียนโปรแกรมต่อจากคนอื่น มันเขียนคลาสมา เราก็ไม่ต้องไปดูโค้ดว่า มันทำงานอย่างไร แค่รู้ว่า มันมีเมธอด ทำแล้ว เกิด output อะไรออกมา

2. เราไม่ต้องเขียนคลาสมันเอง เราอาจจะไปดาวน์โหลดคลาสดูฟรี จากเน็ต แล้วไม่ต้องเข้าใจเลยว่า คลาสมันทำงานยังไง เพียงแค่รู้ว่า ใช้เมธอดแล้ว เกิด output อะไรออกมา ก็สามารถนำคลาสนั้นมาใช้งานได้ทันที เป็นต้น

ในการสร้าง มนุษย์ แต่ละคนจากตัวต้นแบบ เราเรียกว่ากระบวนการ constructor ซึ่งสามารถกำหนดคุณสมบัติต่างๆ ให้กับมนุษย์ที่จะสร้างขึ้นได้ เช่น ลักษณะสีผิว แขน ขา ต่างๆ เป็นต้น และเมื่อมนุษย์ถูกสร้างขึ้นใหม่ (object, instane) คุณสมบัติต่างๆ สามารถที่จะถูกเปลี่ยนแปลงไปได้อย่างอิสระ โดยที่ไม่ต้องกระทบกับต้นแบบ ในการถ่ายทอดคุณสมบัติ (inheritance) เราสามารถรับการถ่ายทอดคุณสมบัติต่างๆ ด้วยการ extend

เราอาจกล่าวคิดได้ว่า ต้นแบบมนุษย์นั้น สามารถแยกออกไปเป็น เชื้อชาติต่างๆ ซึ่งมีคุณสมบัติที่เฉพาะตัวมากขึ้นได้ แน่นนอน มนุษย์ในแต่ละเชื้อชาติจะได้รับการถ่ายทอดคุณสมบัติจาก ต้นแบบมนุษย์มาทั้งหมด มีแขน ขา ตา จมูก เพียงแต่ว่าเราสามารถกำหนดคุณสมบัติเฉพาะเจาะจงลงไปให้กับเชื้อชาตินั้นๆ ได้ และในเรื่องของความสามารถที่สามารถทำได้ (method) เช่น มนุษย์ต้นแบบ ต้องมีการกินเหมือนกัน แต่ในแต่ละเชื้อชาติที่ได้รับการถ่ายทอดคุณสมบัติการกินมานั้น สามารถมีการกินที่มีการกินที่เป็นของตัวเอง เราสามารถกำหนดเงื่อนไขเพิ่มเติมลงไปได้ ซึ่งอาจกล่าวได้ว่าเป็นการกินเหมือนกัน แต่รูปแบบการกินไม่เหมือนกัน เช่น ใช้ตะเกียบ ใช้มือ ใช้มีด ต่างๆ นานา ซึ่งวิธีการนี้เราเรียกว่าการทำ override นั่นคือ method จะมีชื่อเหมือนกับของ class แม้ที่เรารับการถ่ายทอดคุณสมบัติมา แต่ทำงานไม่เหมือนกัน แต่เรายังสามารถใช้ รูปแบบการกินของ class แม่ได้โดยการใช้คำสั่ง super

2.9.1 คุณสมบัติของ OOP

สำหรับคุณสมบัติที่สำคัญ ที่ต้องรู้ เพื่อเราจะเขียนโปรแกรมแบบ OOP มี 3 ประการดังนี้

2.9.1.1 Encapsulation เป็นคุณสมบัติที่ว่า เราไม่จำเป็นต้องสนใจรายละเอียด ที่สมควรจะสนใจ เช่น เราเขียนโปรแกรมที่มีการใช้ Button เราก็ไม่ต้องไปสนใจ ถึงขั้น ตอนการสร้าง Button ขึ้นมา หรือแม้กระทั่งขั้นตอนการใส่ชื่อ ให้กับ Button ว่ามีขั้นตอนการใส่อย่างไร พุดง่าย อย่าไปสนใจมันเลย เราเพียงแค่กำหนดคุณสมบัติให้มันก็เพียงพอแล้ว และใช้งานตามความสามารถที่มีเท่านั้น

2.9.1.2. Inheritance เป็นคุณสมบัติที่ว่า Class ต้องสามารถสืบทอด ได้เช่นเดียวกับภาษาโปรแกรม ที่กำหนด เป็น Component ทั้งที่ มองเห็นและมองไม่เห็น ก็ต้องสืบทอดได้ โดย ดีไรฟ์คลาส ก็คือ Class ที่ถูกสืบทอดมา นั้น สามารถเพิ่มเติม Property หรือ Method เดิมได้ตามความเหมาะสม

2.9.1.3 Polymorphism เป็นคุณสมบัติที่ว่า สามารถเปลี่ยนแปลงความสามารถของ Class ให้เข้ากับสภาพแวดล้อมได้ ตัวอย่างง่ายๆ เช่น เราสร้าง Class ที่ชื่อว่า Shape ซึ่งจะใช้สร้าง Object เป็นรูปทรงต่างๆ เช่น วงกลม, สามเหลี่ยม และ สี่เหลี่ยม เป็นต้น แล้วเราก็มี Method Area เพื่อหาพื้นที่ของรูปทรงต่างๆ แน่แน่นอนว่า Method Area ของการเรียกใช้งานแต่ละครั้ง ต้องคำนึงด้วยว่า เราระบุ Property ของรูปทรงว่าเป็นรูปทรงอะไร ซึ่งจะทำให้เรามีวิธีการคำนวณหา ที่แตกต่างกันนั่นเอง

2.9.2 Class และ Object (เพิ่มเติม)

แนวคิดของ OOP นั้นจะกำหนดให้สิ่งต่างๆ เป็นวัตถุ หรือเรียกว่า Object ซึ่ง Object แต่ละตัว จะถูกสร้างขึ้นมาจาก Class ดังนั้น Object ที่สร้างจาก Class จะถือว่าเป็นคนละตัวกันแต่มีชนิดเดียวกัน เพราะสร้างจาก class เดียวกัน

ตัวอย่างเช่น Class ของโทรศัพท์มือถือ ที่สร้างออกมาจากโทรศัพท์มือถือคนละรุ่นกัน เมื่อมองถึง Object ตัวหนึ่งๆ จะเห็นว่ามันมีข้อมูลที่ใช้กำหนดลักษณะเฉพาะ ของ Object ซึ่งเรียกว่า Property

ตัวอย่างเช่น Object ของโทรศัพท์มือถือ จะมี Property ที่แสดง ลักษณะเฉพาะเช่น ยี่ห้อ, สี, รุ่น, จอภาพ และ ภาษา เป็นต้น เมื่อมี Property แล้ว สิ่งหนึ่งที่เราขาดไม่ได้ของ Object คือ ความสามารถในการทำงาน ของ Object ซึ่งเรียกว่า Method ตัวอย่างเช่น Object ของโทรศัพท์มือถือ จะมี Switch ปิดเปิด, ปุ่มกดตัวเลข, ระบบการรับส่ง และ เสียงโทนต่างๆ เป็นต้น

เมื่อเราสร้าง Object หนึ่งๆ ขึ้นมาจาก Class ในภาษาการเขียนโปรแกรม จะกล่าวได้ว่าเป็น Object หรือ Instance ของ Class

ตัวอย่างเช่น ในการสร้าง Application กับ Delphi เมื่อเรานำ Component มาวางลงบน Form เช่น นำ Edit Box ลงมาวางใน Form จะเห็นว่าใน Code Editor นั้น Delphi ได้เขียน Code เพื่อแสดงว่า ได้สร้าง Object Edit ขึ้นมาจาก Class Tedit

2.10 Unified Modeling Language (UML)

UML ย่อมาจาก Unified Modeling Language เป็นภาษาที่ใช้ในการอธิบายโมเดลเชิงวัตถุในรูปแบบของกราฟฟิกเป็นสัญลักษณ์ ซึ่งโมเดลที่สร้างขึ้นจะสามารถช่วยให้เข้าใจปัญหาได้ง่ายขึ้นและยังสามารถนำโมเดลมาเป็นเครื่องมือในการสื่อสารถ่ายทอดความคิดกับบุคคลอื่น ๆ ได้ง่ายขึ้น เช่น นักวิเคราะห์ นักออกแบบ ลูกค้า เป็นต้น นอกจากนี้การสร้างโมเดลยังช่วยลดค่าใช้จ่ายในการพัฒนา กล่าวคือ โมเดลจะถูกทดลองสร้างขึ้นก่อนการสร้างระบบจริงเพื่อศึกษาหรือค้นหาข้อบกพร่องรวมถึงการแก้ไขเปลี่ยนแปลงส่วนต่าง ๆ ซึ่งจะเป็นไปได้ไม่ง่ายและสิ้นเปลืองเป็นอย่างมากหากต้องทำกับระบบจริงที่พัฒนาขึ้นมา

2.10.1 ข้อดีของยูเอ็มแอล

2.10.1.1 UML เป็นภาษาสากลที่ใช้ในการพัฒนาซอฟต์แวร์เชิงวัตถุและสามารถใช้ในการแลกเปลี่ยนโมเดลอย่างสื่อความหมายรวมถึงการจัดสร้างเอกสารการวิเคราะห์ออกแบบระบบ โดยเฉพาะในการสร้างระบบขนาดใหญ่ซึ่งต้องอาศัยการทำงานเป็นทีม ช่วยการทำงานมีการทำงานในทิศทางเดียวกัน

2.10.1.2 สนับสนุนหลักการเชิงวัตถุได้อย่างครบถ้วนชัดเจน ทำให้เข้าใจกับปัญหาและวิธีแก้ไข ได้อย่างรวดเร็วและง่ายยิ่งขึ้น

2.10.1.3 เป็นภาษาง่ายต่อการทำความเข้าใจ ใช้เพียงแนวคิดเชิงวัตถุมาช่วยในการสร้างระบบจริง

2.10.1.4 ไม่ผูกติดกับภาษาโปรแกรมใดภาษาหนึ่ง สามารถแปลไปเป็นระบบจริงที่ถูกต้องสร้างขึ้นด้วยภาษาโปรแกรมเชิงวัตถุใด ๆ ก็ได้

2.10.1.5 สามารถถูกแปลงเป็นภาษาที่ใช้ในการสร้างระบบขึ้นจริงได้อย่างอัตโนมัติ จึงช่วยลดเวลา ภาระและค่าใช้จ่ายในการพัฒนาระบบได้เป็นอย่างมาก โดยในโครงการนี้จะใช้ Microsoft Visio เป็นตัวช่วยในการแปลงภาษาขึ้นมาสร้างระบบที่ทำการวิเคราะห์

2.10.2 ไคอะแกรมที่ใช้งานใน UML

สำหรับไคอะแกรมที่ใช้งานเราจะทำการแบ่งออกเป็น 8 ไคอะแกรม ซึ่งมีรายละเอียด คือ

2.10.2.1 ยูสเคสไคอะแกรม (Use Case Diagram) ใช้ในการโมเดลฟังก์ชันการทำงานของระบบ ในการพัฒนาระบบงานใดๆ นั้น การเก็บรวบรวมความต้องการของผู้ใช้มีความสำคัญมาก และจะทำในระยะแรกๆ ของการพัฒนาระบบงานเสมอ Use Case diagram เป็นเทคนิคในการสร้างแบบจำลองเพื่อใช้อธิบายหน้าที่ของระบบใหม่ หรือระบบปัจจุบัน องค์ประกอบของยูสเคสไคอะแกรมจะมี Use Case , Actor , Use case Relation และ System

2.10.2.2 คลาสไคอะแกรม (Class Diagram) ใช้ในการโมเดลคลาสต่าง ๆ ที่จำเป็นในระบบ และความสัมพันธ์ต่างๆ ระหว่างคลาส

2.10.2.3 แอ็กทิวิตีไคอะแกรม (Activity Diagram) ใช้ในลำดับการทำงาน เหมือนกับโฟลว์ชาร์ต จะแสดงขั้นตอนการทำงานในการปฏิบัติการ โดยประกอบด้วยสถานะต่าง ๆ ที่เกิดขึ้นระหว่างการทำงาน และผลจากการทำงานในขั้นตอนต่างๆ

2.10.2.4 สเตตชาร์ตไคอะแกรม (State chart Diagram) ใช้สำหรับแสดงถึงสถานะของออบเจกต์ในระหว่างการทำงาน

2.10.2.5 คอลแลบอเรชันไคอะแกรม (Collaboration Diagram) ใช้แสดงการทำงานร่วมกันของออบเจกต์ในระบบ

2.10.2.6 ซีควเอนซ์ไคอะแกรม (Sequence Diagram) ใช้ในการโมเดลกิจกรรมต่าง ๆ ที่เกิดขึ้นกับออบเจกต์ในระบบ ตามลำดับของเหตุการณ์ที่เกิดขึ้น ณ เวลาที่กำหนด

2.10.2.7 คอมโพเนนต์ไคอะแกรม (Component Diagram) ใช้สำหรับสร้างโมเดลของคอมโพเนนต์ในระบบ แสดงโครงสร้างทางกายภาพของ Software โดยจะประกอบด้วยองค์ประกอบซึ่งอยู่ในรูปต่างๆ เช่น Binary , text และ executable ภายใน Component Diagram ก็จะมีความสัมพันธ์แสดงอยู่เช่นเดียวกับ Class diagram

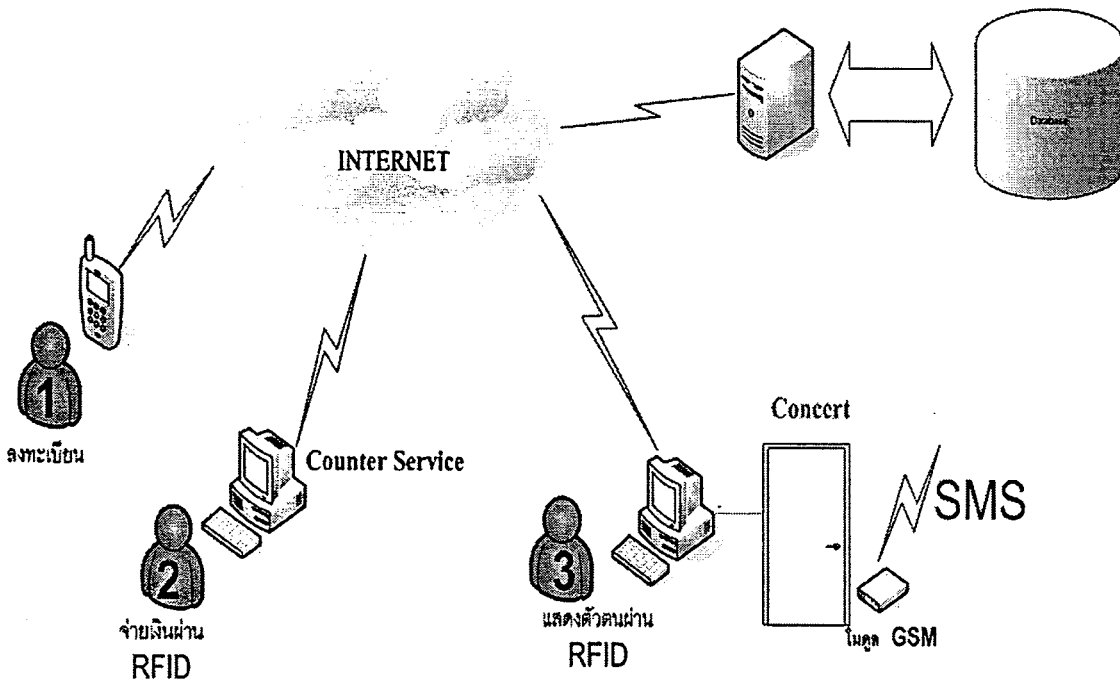
2.10.2.8 ดีพลอยเมนต์ไคอะแกรม (Deployment Diagram) ใช้แสดงการติดตั้งใช้งานส่วนประกอบต่างๆ ของระบบ เป็นสิ่งที่สามารถทำการแสดงระบบสถาปัตยกรรมและความสัมพันธ์ของ Hardware / Software

บทที่ 3

การออกแบบและการสร้าง

การออกแบบและการสร้างระบบการจองตั๋วคอนเสิร์ตผ่านโทรศัพท์เคลื่อนที่และพิสูจน์ตัวตนเพื่อการออกตั๋ว จะทำการแบ่งออกเป็น 3 ส่วนประกอบหลักด้วยกัน

1. ส่วนของการจองตั๋วผ่านเว็บบนโทรศัพท์เคลื่อนที่โดยใช้ ASP.NET ในการพัฒนา
2. ส่วนของการจ่ายเงินผ่านระบบรับชำระผ่านเว็บเซอร์วิส (Web service) โดยใช้ RFID Tag ในการชำระแทนเงินสด โดยใช้ภาษา C# ในการพัฒนา
3. ส่วนของระบบจำลองการแสดงตัวตนเพื่อรับสิทธิ์ ในการเข้าชมคอนเสิร์ตที่ได้ทำการชำระค่าเรียบร้อยแล้ว โดยใช้ระบบเว็บเซอร์วิส (Web service) ร่วมกับการออกแบบจำลองประตูโดยใช้ไมโครคอนโทรลเลอร์ PIC 16F877A ควบคุมการเปิด-ปิด LED และ โมดูล GSM/GPRS รุ่น SIM300CZ เพื่อส่ง SMS ให้กับผู้ใช้บริการ



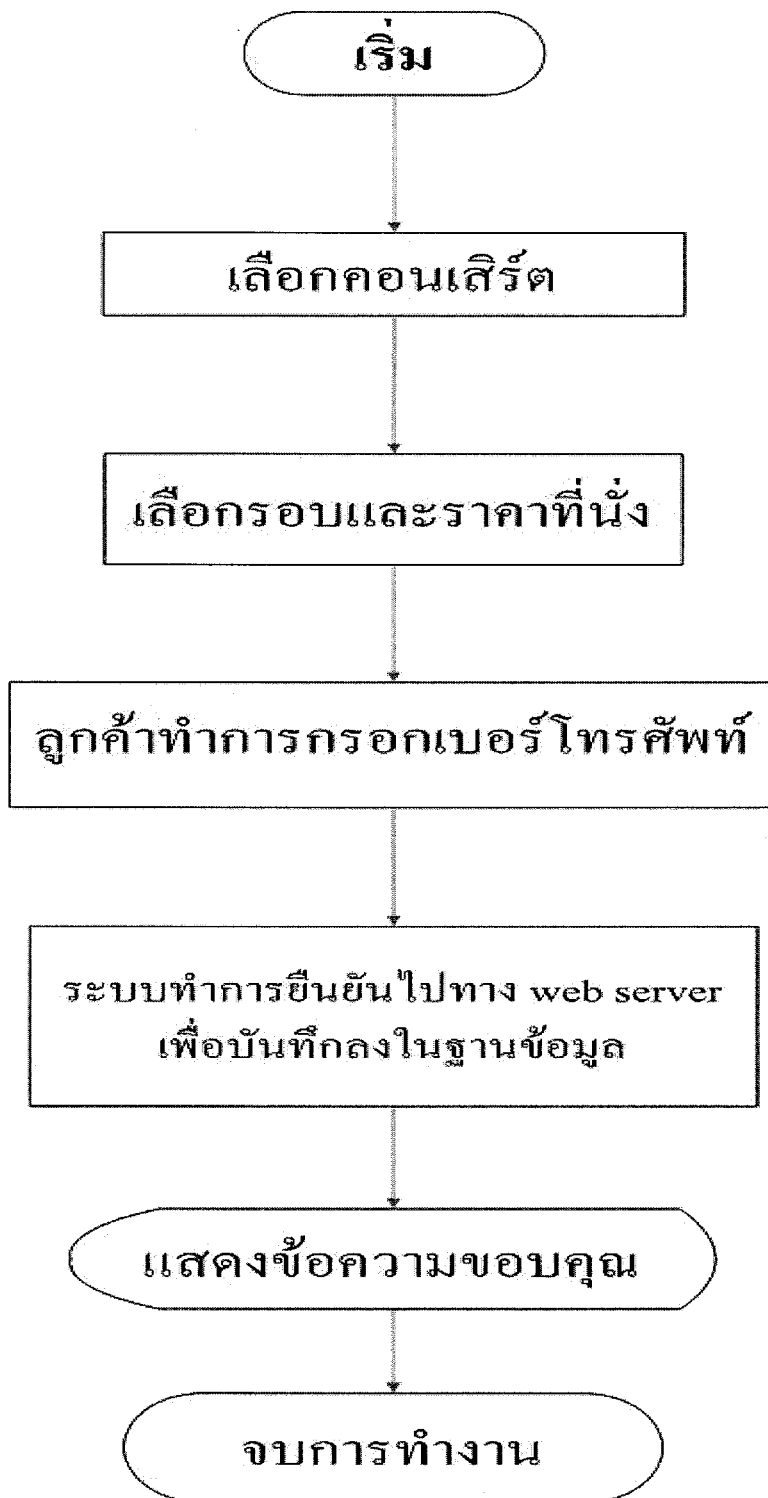
รูปที่ 3.1 แสดงมุมมองทั้งหมดของผู้ใช้บริการ

จากรูปแสดงมุมมองทั้งหมดของผู้ใช้บริการ แบ่งเป็น 3 ขั้นตอนคือ

1. ผู้ใช้บริการต้องทำการจองคอนเสิร์ตผ่านเว็บบนโทรศัพท์เคลื่อนที่
2. ผู้ใช้บริการสามารถชำระค่าเข้าชมคอนเสิร์ตผ่านจุดชำระเงินออนไลน์ตามเคาน์เตอร์เซอร์วิส โดยใช้ระบบ RFID
3. ผู้ใช้บริการแสดงตัวตนผ่านระบบ RFID เพื่อเข้าชมคอนเสิร์ตต่อไป

3.1 การออกแบบและการสร้างระบบการจองตั๋วผ่านเว็บบนโทรศัพท์เคลื่อนที่

3.1.1 บล็อกการทำงาน

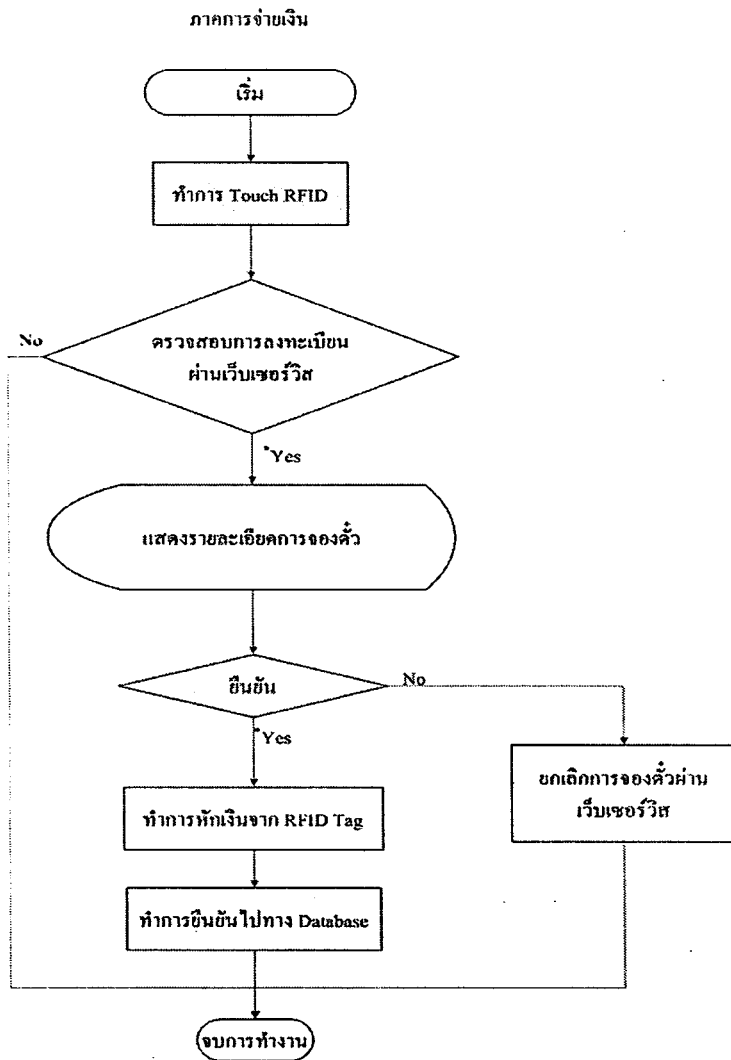


รูปที่ 3.2 บล็อกไดอะแกรมการทำงานของโปรแกรมเพื่อการจองตั๋ว

จากรูปบล็อกไดอะแกรมการทำงานของโปรแกรมเพื่อการจองตั๋วผ่านโทรศัพท์เคลื่อนที่ โดยผู้ให้บริการจะต้องทำการเลือกคอนเสิร์ต รอบและราคาที่นั่ง โดยผู้ให้บริการทำการกรอกเบอร์โทรศัพท์ เพื่อให้ระบบทำการบันทึกข้อมูล โดยเราใช้ ASP.NET ด้วยภาษา C# เพื่อใช้พัฒนาระบบในส่วนภาคจองตั๋ว

3.2 การออกแบบและการสร้างระบบการจ่ายเงินผ่านระบบ RFID ซึ่งทำการติดต่อไปยังเว็บเซิร์ฟเวอร์

3.2.1 บล็อกการทำงาน

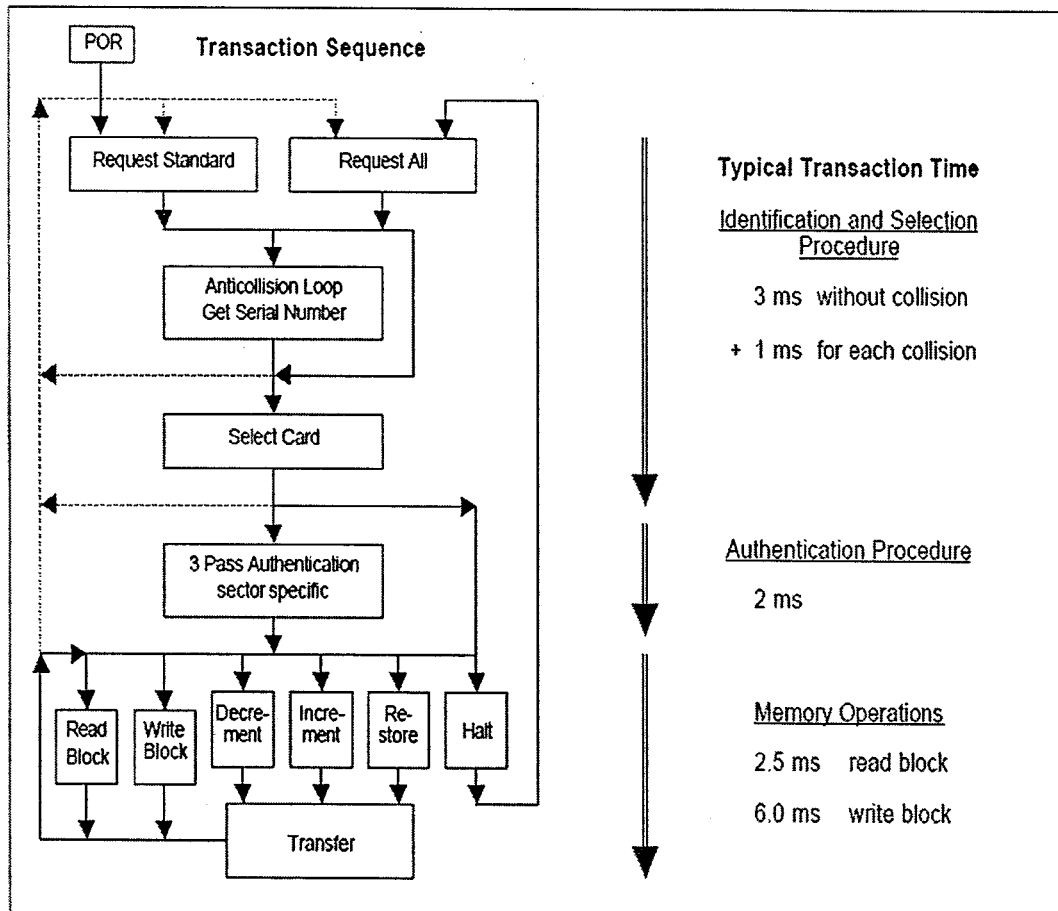


รูปที่ 3.3 บล็อกไดอะแกรมการทำงานของโปรแกรมเพื่อการจ่ายเงินผ่านระบบ RFID

ส่วนนี้เป็นภาคการชำระเงินผ่านระบบเว็บเซอร์วิสโดยใช้ RFID มาตรฐาน Mifare ของ PHILIPS โดยมีขั้นตอนดังนี้

1. ผู้ให้บริการทำการสัมผัส RFID Tag ที่เคาน์เตอร์เซอร์วิสเพื่อเป็นการตรวจสอบการจอง
2. ระบบชำระเงินออนไลน์จะทำการตรวจสอบและทำการหักเงินตามค่าบริการของผู้ใช้งานที่ได้ทำการลงทะเบียนไว้ ผ่าน Web service ที่ใช้ ASP.NET ในการพัฒนา

3. ระบบจะทำการยืนยันว่าได้ชำระเงินเรียบร้อยแล้ว และให้สิทธิเข้าชมคอนเสิร์ตตามรอบและที่นั่งที่ได้จองไว้แล้ว ซึ่งจะบันทึกไว้ในฐานข้อมูล



รูปที่ 3.4 บล็อกไคอะแกรมการทำงานของระบบ RFID มาตรฐาน Mifare

จากรูปที่ 3.4 บล็อกไคอะแกรมการทำงานของระบบ RFID มาตรฐาน Mifare เราจะต้องทำการเขียนโปรแกรมขั้นตอนต่าง ๆ ตามบล็อกนี้เพื่อที่จะเข้าถึง RFID Tag ได้ โดยมีขั้นตอนดังนี้

1. ทำการรอเพื่อที่จะมีการทัก RFID Tag ผ่าน RFID reader โดยจะใช้เวลาทำงาน 3 ms
2. จะมีการป้องกันการชนของลูป โดยรับค่าจาก Serial Number ของ RFID Tag โดยจะใช้เวลาทำงาน 3 ms
3. RFID Reader จะทำการเลือก RFID Tag มาเพียง 1 ใบ เพื่อนำไปทำกระบวนการต่อไป โดยจะใช้เวลาทำงาน 3 ms
4. จะมีการตรวจสอบรหัสเพื่อจะเข้าถึง Memory ของ RFID Tag ตามรูปที่ 3.4 Tag โดยจะใช้เวลาทำงาน 2 ms
5. เป็นขั้นตอนกระบวนการต่าง ๆ ที่จะทำบน RFID Tag

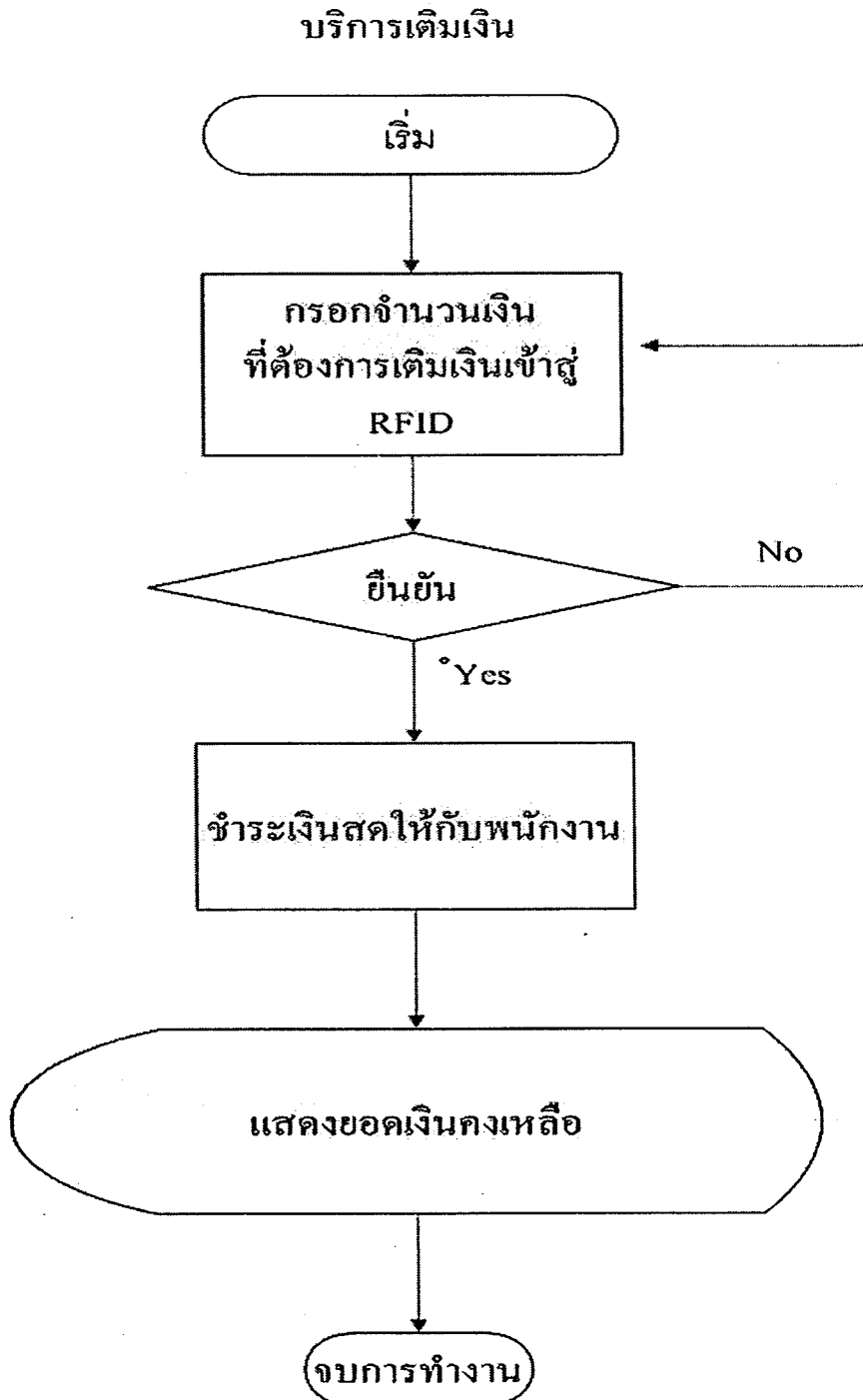
Sector	Block	Byte Number within a Block														Description	
		0	1	2	3	4	5	6	7	8	9	10	11	12	13		14
15	3	Key A				Access Bits				Key B						Sector Trailer 15	
	2																Data
	1																Data
	0																Data
14	3	Key A				Access Bits				Key B						Sector Trailer 14	
	2																Data
	1																Data
	0																Data
:	:																
:	:																
:	:																
1	3	Key A				Access Bits				Key B						Sector Trailer 1	
	2																Data
	1																Data
	0																Data
0	3	Key A				Access Bits				Key B						Sector Trailer 0	
	2																Data
	1																Data
	0																Manufacturer Block

รูปที่ 3.5 Memory ของ RFID Tag มาตรฐาน Mifare ขนาด 1024 X 8 บิต

จากรูปที่ 3.5 หน่วยความจำของ RFID Tag มาตรฐาน Mifare ขนาด 1024 X 8 บิต โดยจะแบ่งเป็น 16 Sector โดยแต่ละ Sector แบ่งออกเป็น 4 Block ซึ่งแต่ละ Block จะสามารถเก็บข้อมูลได้ 16 Byte มีข้อควรระวังว่า Block สุดท้ายของแต่ละ Sector จะนำไปใช้ในการเก็บรหัสเพื่อเข้าถึงในแต่ละ Sector และ RFID Tag จะเก็บ Serial card ไว้ที่ Sector 0 Block 0 ของ Tag แต่ละใบ จะมีหมายเลขที่ต่างกัน

3.2.2 บริการเติมเงินให้กับระบบ RFID

การบริการทางเคาน์เตอร์เซอร์วิส จะมีบริการให้ผู้ให้บริการสามารถทำการเติมเงินได้ เพื่อเป็นการเพิ่มความสะดวกรวดสบายให้กับผู้ใช้บริการ ได้อีกทางเลือกหนึ่ง



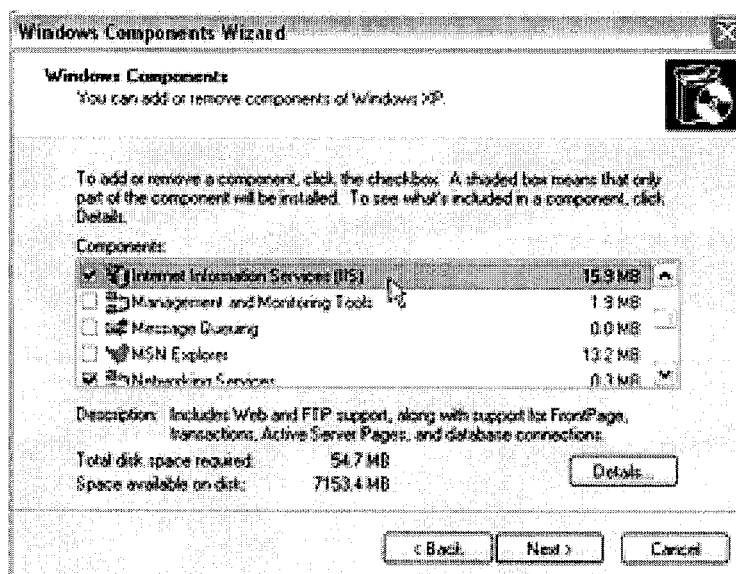
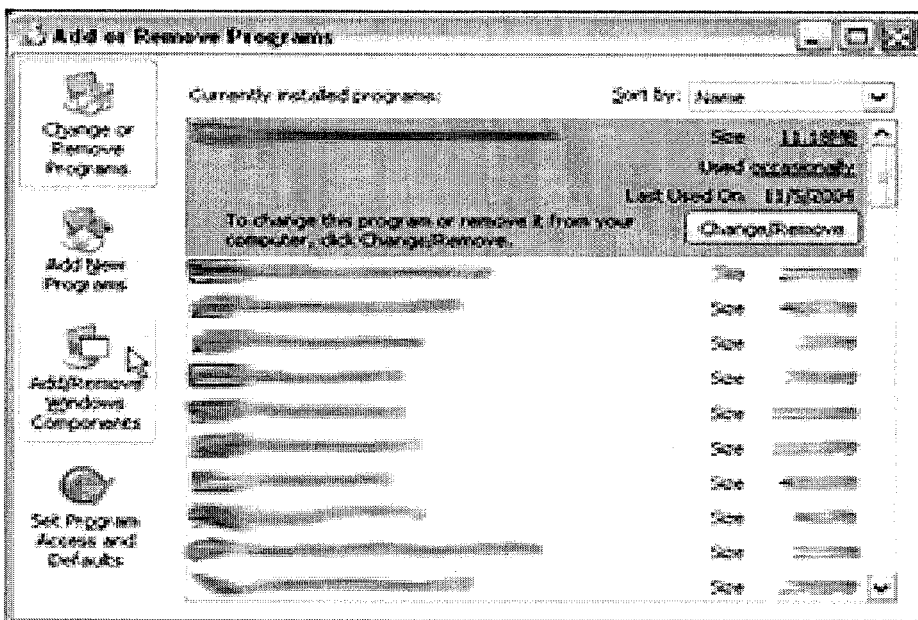
รูปที่ 3.6 บล็อกไดอะแกรมบริการเติมเงินให้กับผู้ใช้บริการ

จากรูปที่ 3.6 บล็อกไดอะแกรมบริการเติมเงินให้กับผู้ใช้บริการ โดยมีกระบวนการทำดังนี้
ผู้ใช้บริการทำการกรอกเงินที่ต้องการเติมเงิน และทำการจ่ายเงินให้กับเจ้าหน้าที่ประจำเคาน์เตอร์เซอร์วิส

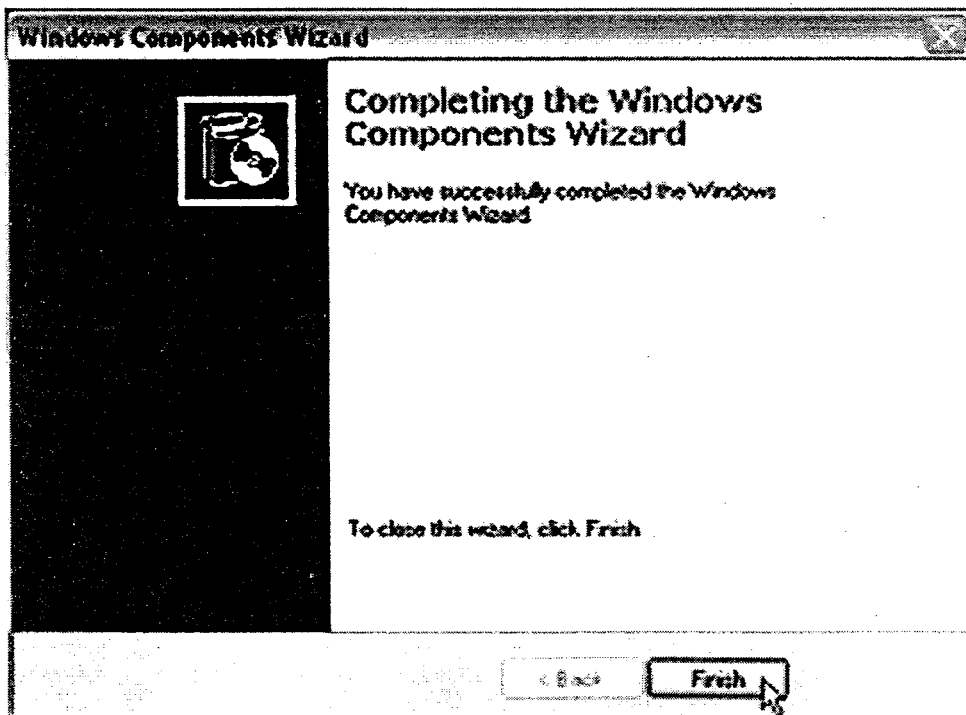
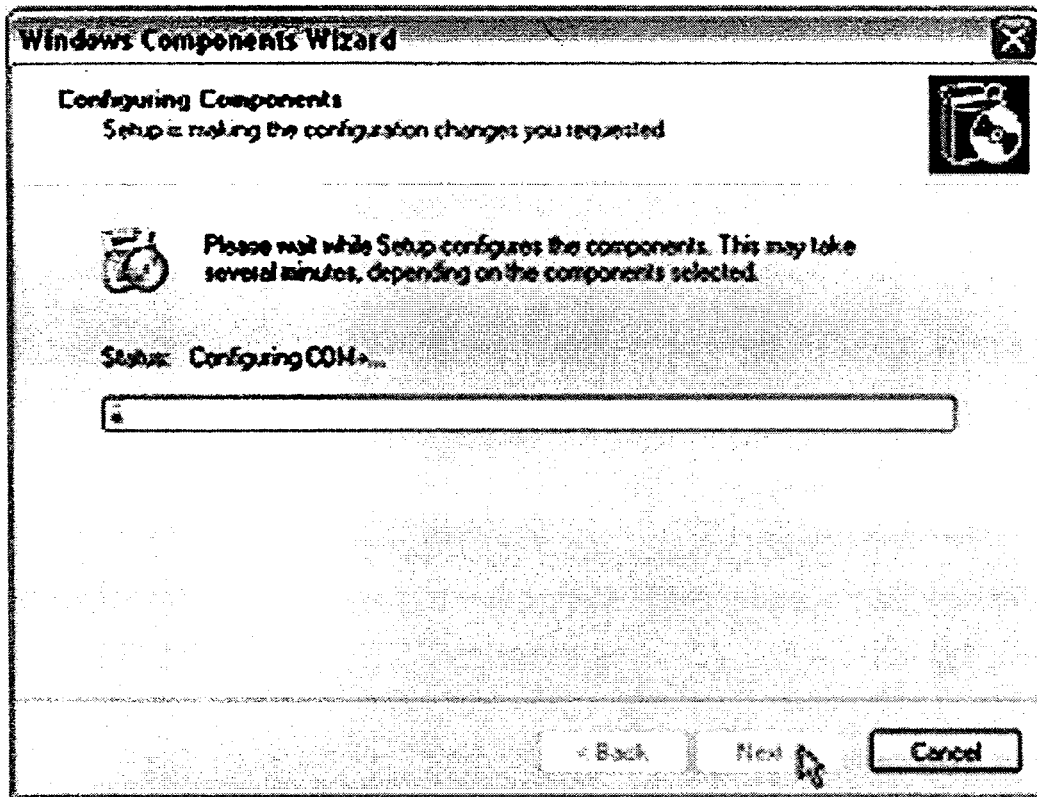
3.3 การออกแบบและการสร้างเว็บเซิร์ฟเวอร์

ในเว็บเซิร์ฟเวอร์เราจะมีฐานข้อมูลของผู้ใช้บริการไว้ทั้งหมด โดยเราใช้ **IIS Web Server** เป็น Web Server ที่มีติดมากับตัวโปรแกรม Windows 2000 Windows XP เราสามารถทำการติดตั้ง IIS เพิ่มเติมได้จากแผ่น CD-ROM Install Windows ซึ่ง IIS เมื่อได้ทำการติดตั้งแล้ว Web Server จะมีความสามารถในการให้บริการต่าง ๆ เกี่ยวกับ Web Site และ IIS ยังมีความสามารถใช้งาน ASP ได้ในทันทีหลังจากที่เราได้ทำการติดตั้งตัวโปรแกรม IIS เสร็จสิ้นแล้ว ขั้นตอนการติดตั้งมีดังนี้

1. ให้คุณทำการใส่แผ่น CD-ROM ของโปรแกรม Windows กับเครื่องอ่าน CD
2. จากนั้นให้ไปที่ Control Panel -> Add or Remove Programs จากนั้นให้ทำตามขั้นตอนดังนี้

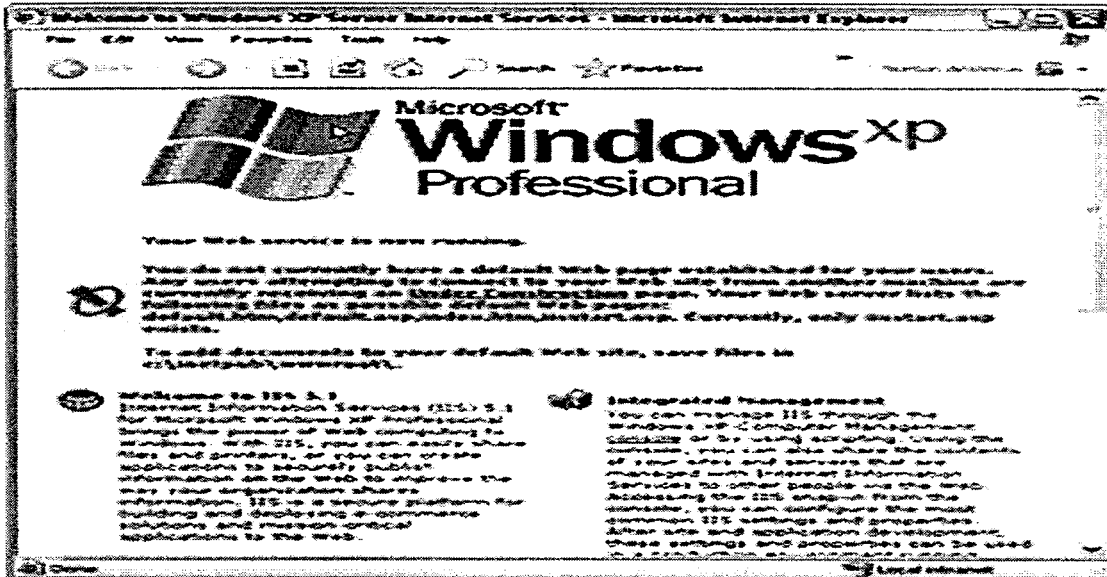


เลือกไปที่ Add/Remove Windows Components จากนั้นที่หน้าต่าง Windows Components ให้เราทำการคลิกเครื่องหมายถูกที่หน้าเมนู Internet Information Services (IIS) แล้วทำการ Click ที่ปุ่ม Next เพื่อเริ่มการติดตั้ง

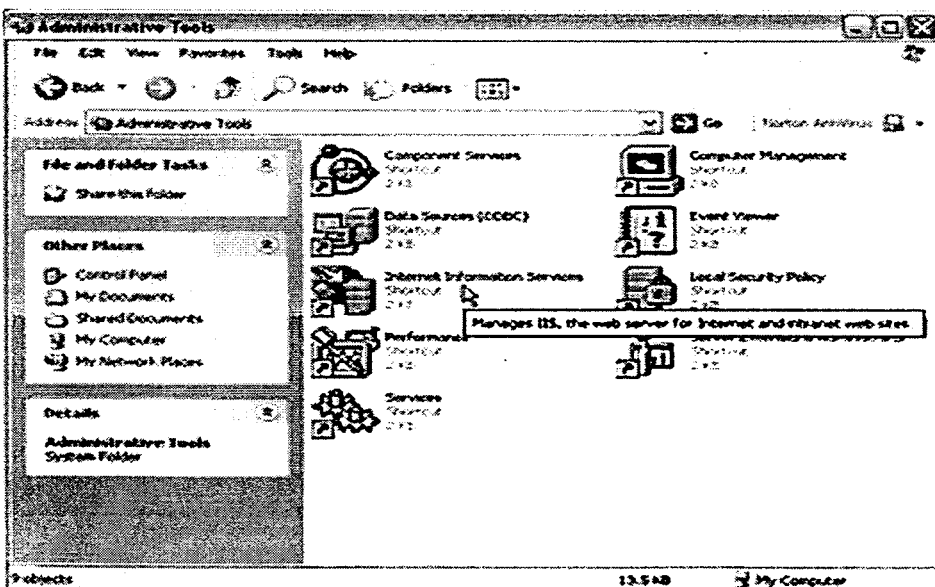


โปรแกรมจบทำการติดตั้ง IIS ให้เรา เมื่อทำการติดตั้งเสร็จสิ้นแล้ว ให้เรา Click ที่ปุ่ม Finish ดังรูป

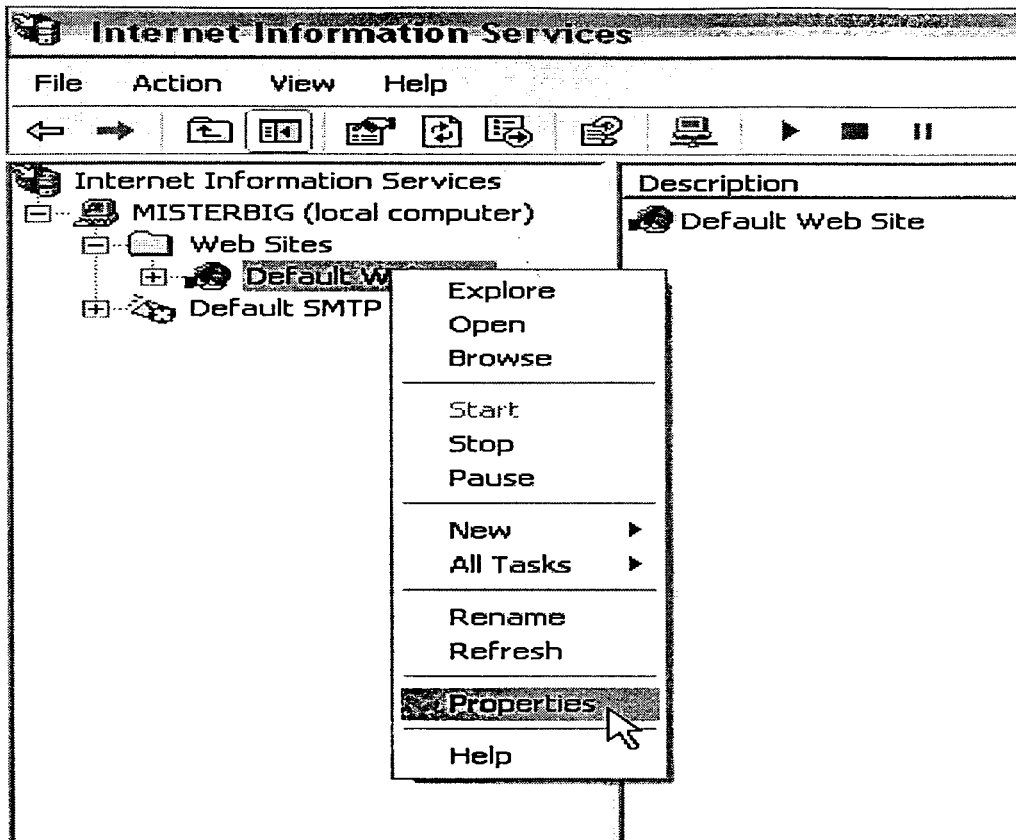
3. จากนั้นให้เราทำการเปิด Web Browser ของเราขึ้นมา ที่ Address Bar ให้เราทำการพิมพ์ <http://localhost/> เพื่อทำการทดสอบการใช้งาน IIS หากปรากฏมีหน้าจอต้อนรับ ดังรูปด้านล่างนี้ แสดงว่าเราสามารถใช้งาน IIS Web Server ได้แล้ว



4. สำหรับการ Set ค่าอื่น ๆ เพิ่มเติมให้กับ IIS คุณสามารถทำการกำหนดค่าเพิ่มเติมได้จาก Control Panel -> Administrative Tools -> Internet Information Services ดังรูป



5. ที่ IIS Control นี้คุณสามารถ Click ขวาแล้วกำหนดคุณสมบัติต่าง ๆ ได้จากเมนู Properties ดังรูป



เป็นอันเสร็จสิ้นการติดตั้ง IIS Web Server ให้กับ Windows

3.4 การออกแบบและการสร้างฐานข้อมูลโดยใช้ Microsoft SQL Server 2005

เราจะทำการจำลองและออกแบบฐานข้อมูลที่รันบนเว็บเซิร์ฟเวอร์ โดยเราเลือก Microsoft SQL Server 2005

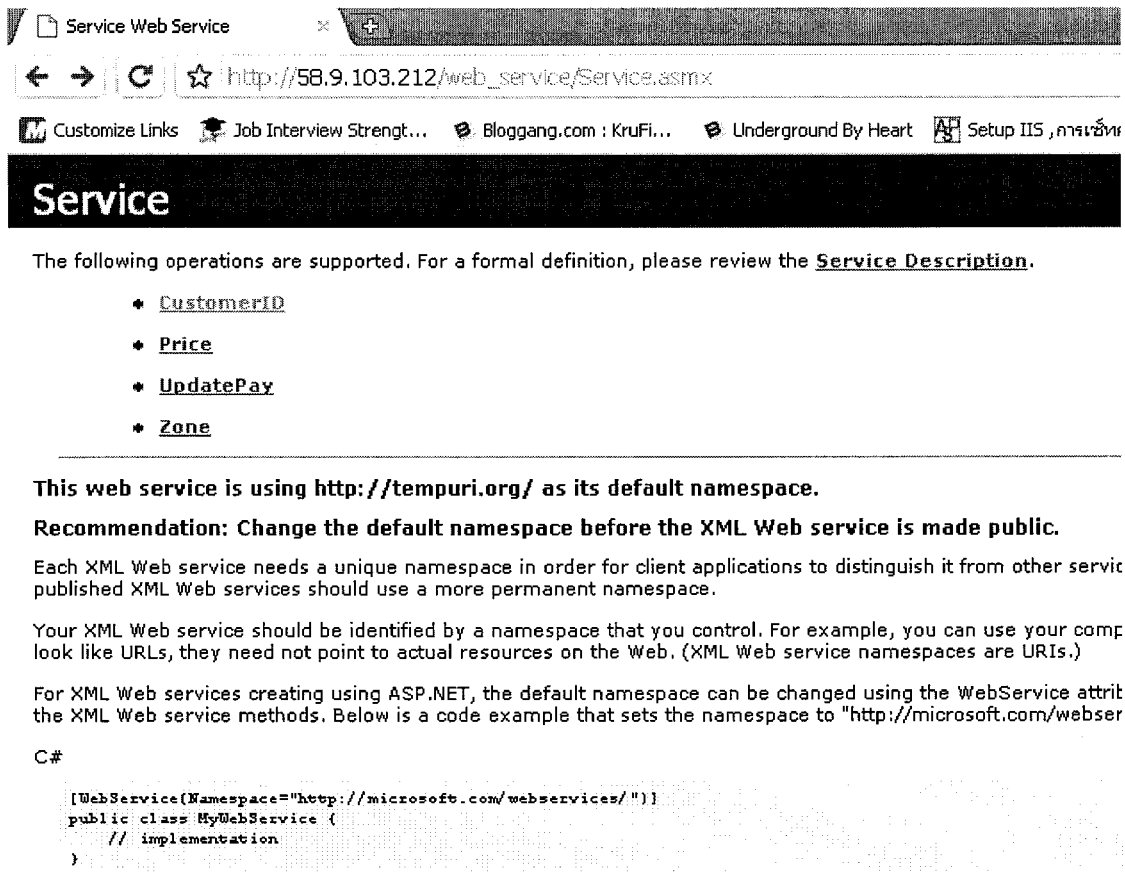
Column Name	Data Type	Allow Nulls
Number	int	<input type="checkbox"/>
CustomerID	int	<input type="checkbox"/>
CustomerName	text	<input checked="" type="checkbox"/>
CustomerLastName	text	<input checked="" type="checkbox"/>
ProductID	nvarchar(50)	<input checked="" type="checkbox"/>
ZoneID	nvarchar(50)	<input checked="" type="checkbox"/>
Price	int	<input checked="" type="checkbox"/>
Description	nvarchar(50)	<input checked="" type="checkbox"/>

รูปที่ 3.7 การออกแบบฐานข้อมูลบนเว็บเซิร์ฟเวอร์

จากรูปที่ 3.7 การออกแบบฐานข้อมูลบนเว็บเซิร์ฟเวอร์เพื่อเก็บข้อมูลผู้ใช้บริการ เพื่อเป็นตัวกลางติดต่อหลักของทั้ง 3 ส่วนประกอบใหญ่

3.5 การออกแบบและการสร้างเว็บเซอร์วิสโดยใช้ ASP.NET

เราจะออกแบบโดยให้สามารถมีการลงทะเบียนผ่านเว็บเซอร์วิส นำฐานข้อมูลมาใช้ในการหักเงิน และนำมายืนยันตัวตนเพื่อการอนุญาตให้เข้าชมงานคอนเสิร์ตได้



รูปที่ 3.8 เว็บเซอร์วิสเพื่อให้บริการในขั้นตอนต่าง ๆ

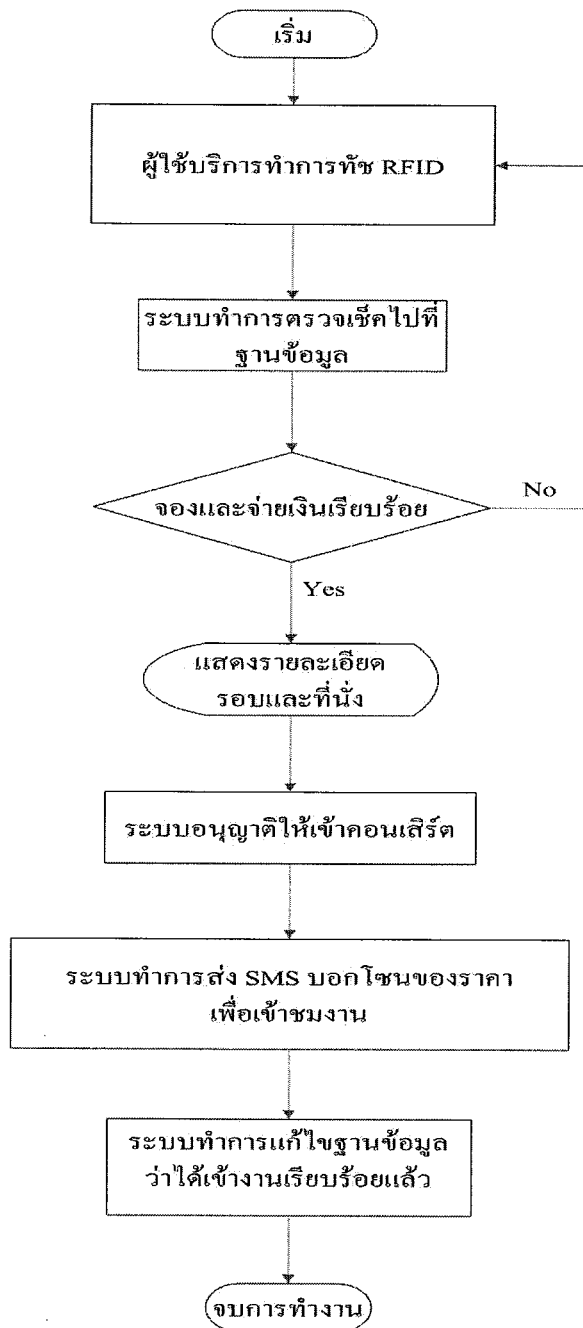
เว็บเซอร์วิสมีกระบวนการอยู่ 4 ขบวนการ คือ

1. CustomerID ใช้สำหรับเรียกดูข้อมูลของผู้ใช้บริการ โดยใช้ตัวแปรเบอร์โทรศัพท์ในการเรียกดู
2. Price ใช้สำหรับบันทึกราคาของตั๋วคอนเสิร์ตที่ผู้ใช้บริการได้ทำการจองไว้
3. UpdatePay ใช้สำหรับบันทึกการกระทำของผู้ใช้บริการ เช่น การจอง การจ่าย และ เข้างาน
4. Zone ใช้สำหรับบันทึกรหัสคอนเสิร์ตที่ลูกค้าทำการจองไว้

3.6 การออกแบบและการสร้างระบบการแสดงตัวตนเพื่ออนุญาตเข้าชมคอนเสิร์ต

ในส่วนประกอบนี้เราจะทำการแบ่งออกเป็น 3 ประกอบย่อย ๆ คือ

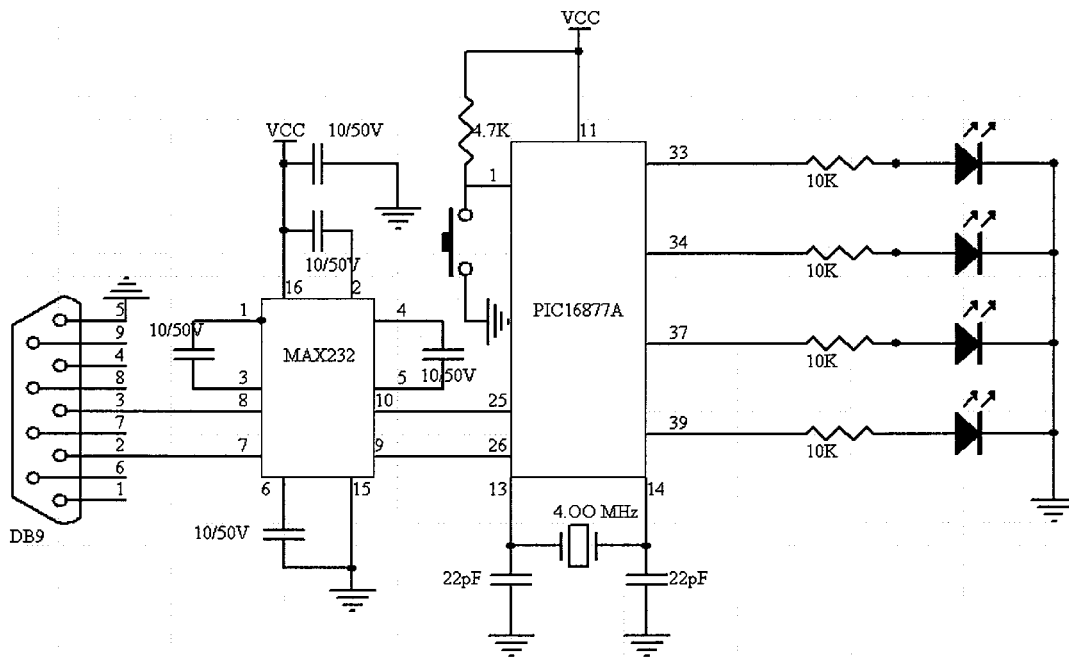
1. ส่วนของโปรแกรมเพื่อติดต่อกับ RFID และติดต่อฐานข้อมูลผ่านเว็บเซอร์วิส
2. ส่วนของการจำลองประตู โดยใช้ไมโครคอนโทรลเลอร์ในการควบคุมการ ติด-ดับ ของ LED
3. ส่วนของการส่ง SMS บอกให้ลูกค้ารู้ว่าได้จองที่นั่งอยู่ในโซนอะไร



รูปที่ 3.9 บล็อกไดอะแกรมในส่วนของการแสดงตัวตนเพื่อเข้าชมงาน

3.7 การออกแบบวงจรควบคุมไฟ LED ด้วยไมโครคอนโทรลเลอร์

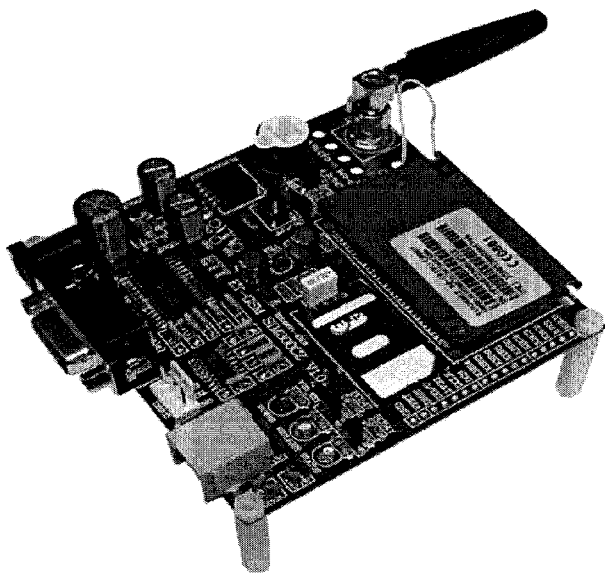
3.7.1 วงจรรวม



รูปที่ 3.10 วงจรรวมจำลองเปิด-ปิดประตูดโดยใช้ LED

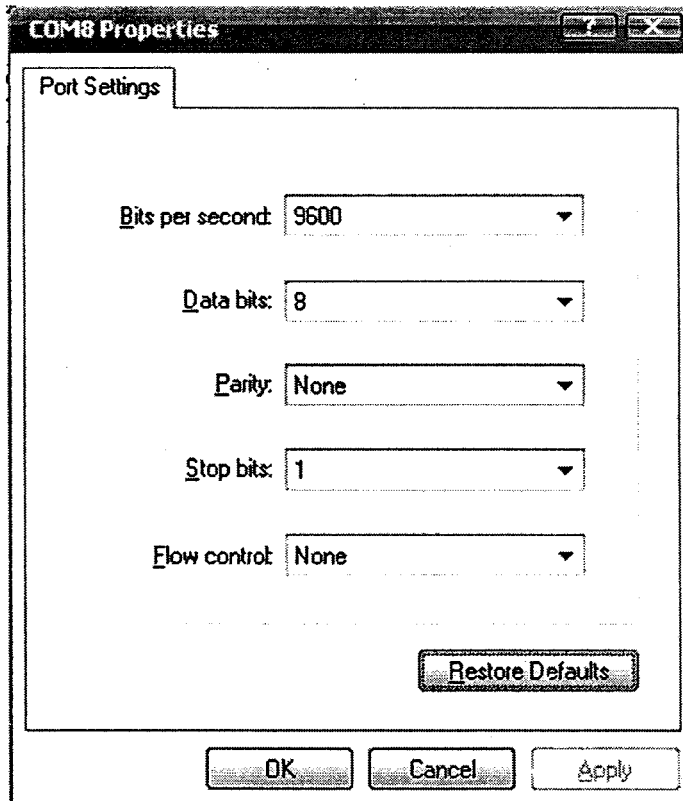
ทำการต่อ RS-232 จากคอมพิวเตอร์เข้ากับวงจร เราจะใช้ PIC16F877A ในการควบคุมการเปิด-ปิดของ LED โดยเราได้เขียนโปรแกรมเพื่อรับค่าจาก C# เข้ามา โดยเราจะเอาไปจำลองประตูต่อไป

3.8 การออกแบบระบบการส่ง SMS เพื่อบอกโซนให้กับลูกค้าทราบ

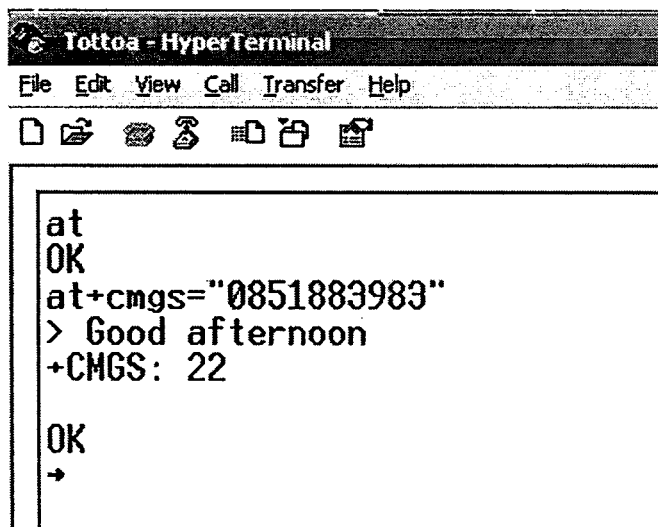


รูปที่ 3.11 โมดูล GSM/GPRS รุ่น 300SIMCZ

ในส่วนนี้เราจะนำโมดูล GSM/GPRS มาใช้ร่วมกับระบบพีซีส่วนตัวคุณ เพื่อให้ผู้ใช้บริการทราบถึงโซนที่มันั่งของตนเอง โดยเราต้องใช้ AT Command ในการเขียนติดต่อกับ โมดูล



รูปที่ 3.12 เชื่อมบิตเรทติดต่อกับ โมดูลผ่าน HyperTerminal



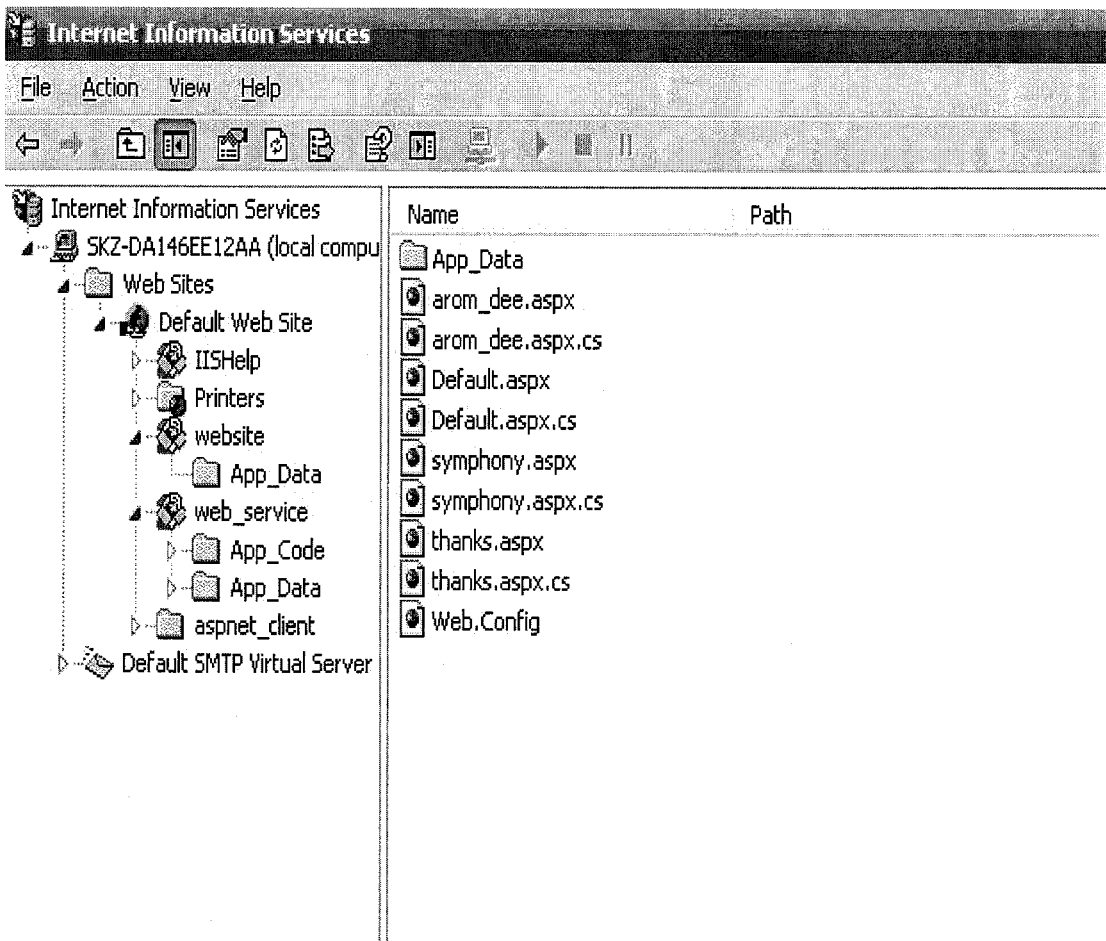
รูปที่ 3.13 แสดงคำสั่ง AT command ใช้ติดต่อกับ โมดูล เพื่อส่ง SMS

บทที่ 4

ผลการทดลอง และวิเคราะห์ผลการทดลอง

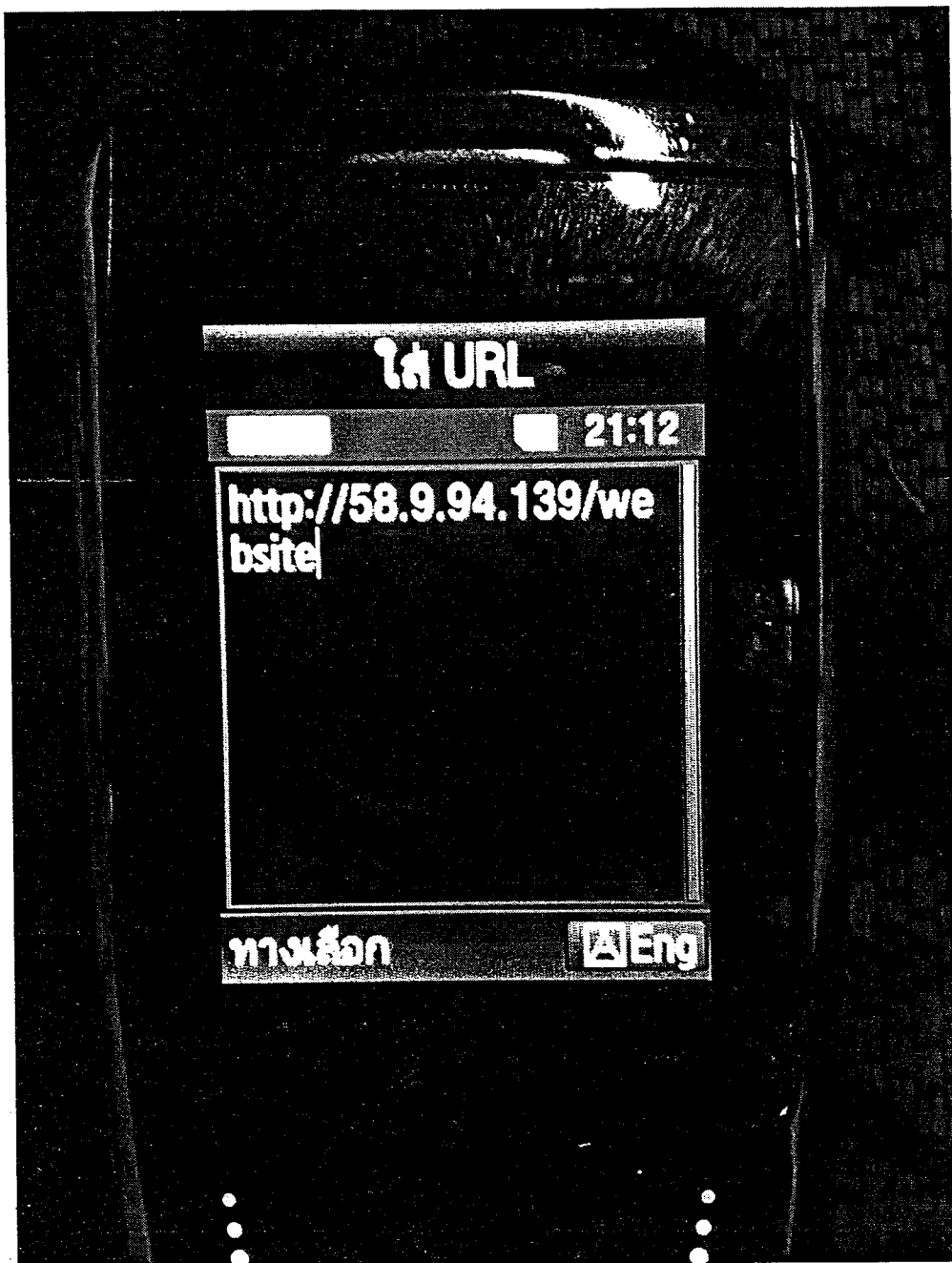
4.1 ผลการทดลองระบบการจองตั๋วผ่านเว็บเบราว์เซอร์บนโทรศัพท์เคลื่อนที่

1. ทำการนำโปรแกรมที่ได้ทำการเขียนไว้ย้ายไป Path ของ Web server IIS (Internet Information Services)

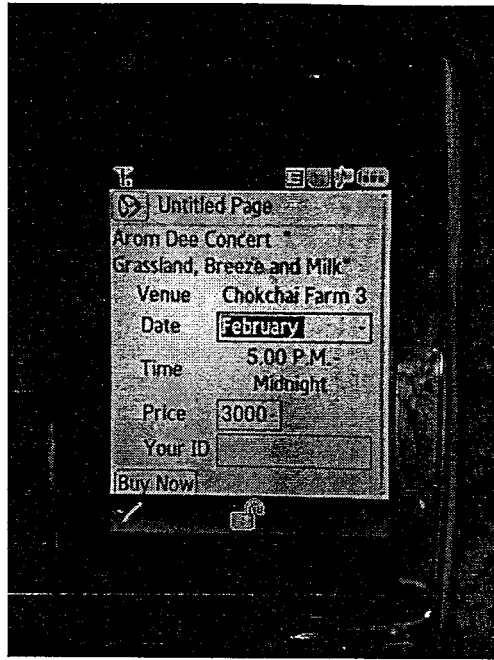


รูปที่ 4.1 แสดงเว็บเซอวิส IIS

2. ทำการกรอก URL ตาม Public IP Address ที่เว็บเซิร์ฟเวอร์ที่ทำการเชื่อมต่อกับ Internet



รูปที่ 4.2 แสดงผลการกรอก URL เพื่อเข้าเว็บการจ้องตัวคอนเสิร์ต



รูปที่ 4.3 แสดงหน้าเว็บบนโทรศัพท์เคลื่อนที่เพื่อการจองตั๋ว

3. ผู้ใช้บริการทำการเลือกรอบคอนเสิร์ต ราคาที่ต้องการจอง และกรอก ID เพื่อยืนยัน

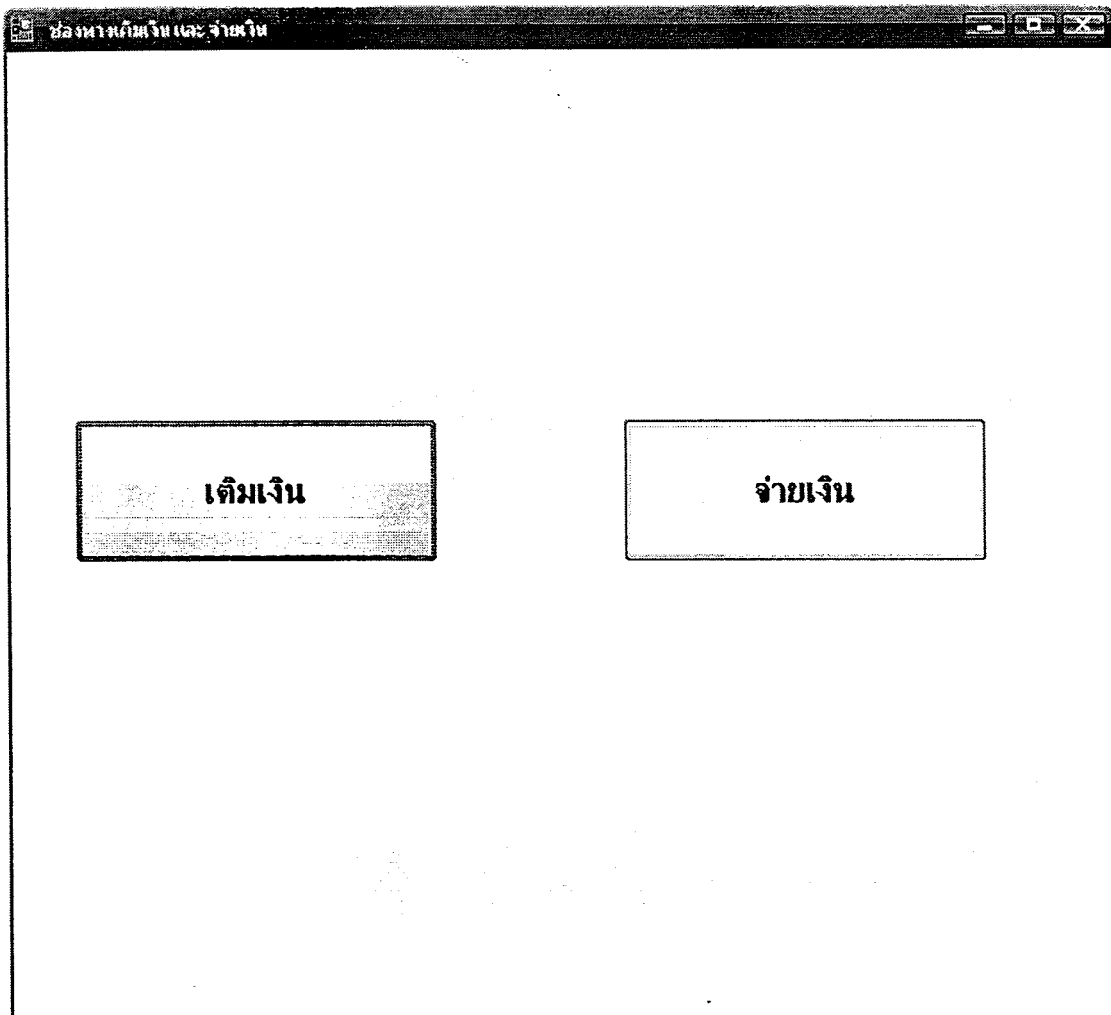
SKZ-DA146EE12...o.BookConcert Summary								
Number	CustomerID	CustomerName	CustomerLastN...	ProductID	ZoneID	Price	Description	
1	832474163	PHATTARAPONG	SAWAENGSIK	NULL	NULL	NULL	NULL	
2	891200093	SUTHINAN	SAWAENGSIK	NULL	NULL	NULL	NULL	
* NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	

รูปที่ 4.4 แสดงฐานข้อมูลเมื่อยังไม่ได้ทำการจองตั๋ว

SKZ-DA146EE12...o.BookConcert Summary								
Number	CustomerID	CustomerName	CustomerLastN...	ProductID	ZoneID	Price	Description	
1	832474163	PHATTARAPONG	SAWAENGSIK	001	February 7th, 2009	3000	Book	
2	891200093	SUTHINAN	SAWAENGSIK	NULL	NULL	NULL	NULL	
* NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	

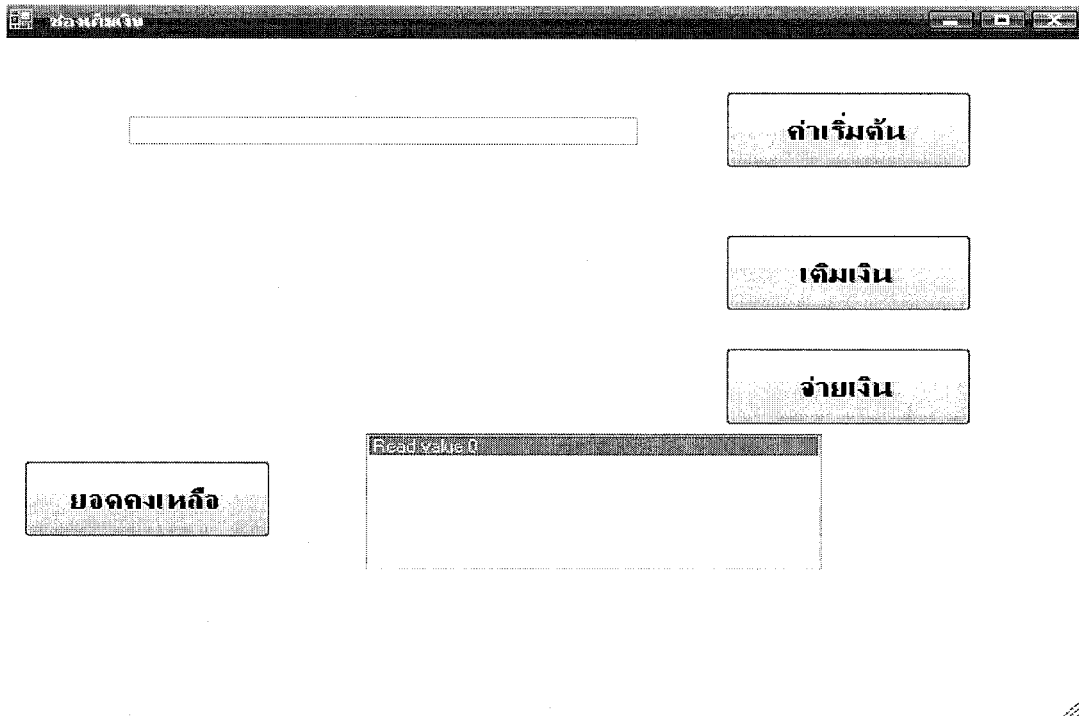
รูปที่ 4.5 แสดงฐานข้อมูลเมื่อทำการจองตั๋วคอนเสิร์ต

4.2 ผลการทดลองเคาน์เตอร์เซอร์วิส

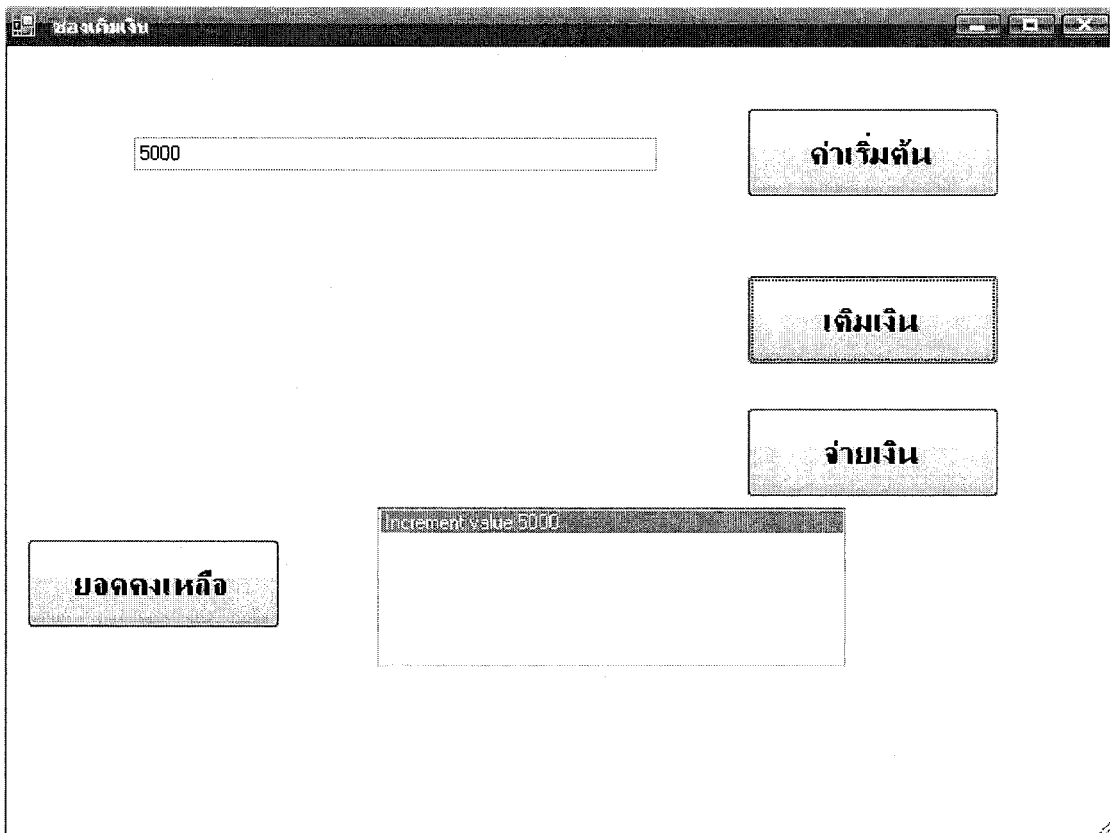


รูปที่ 4.6 แสดงการออกแบบ Window Applications

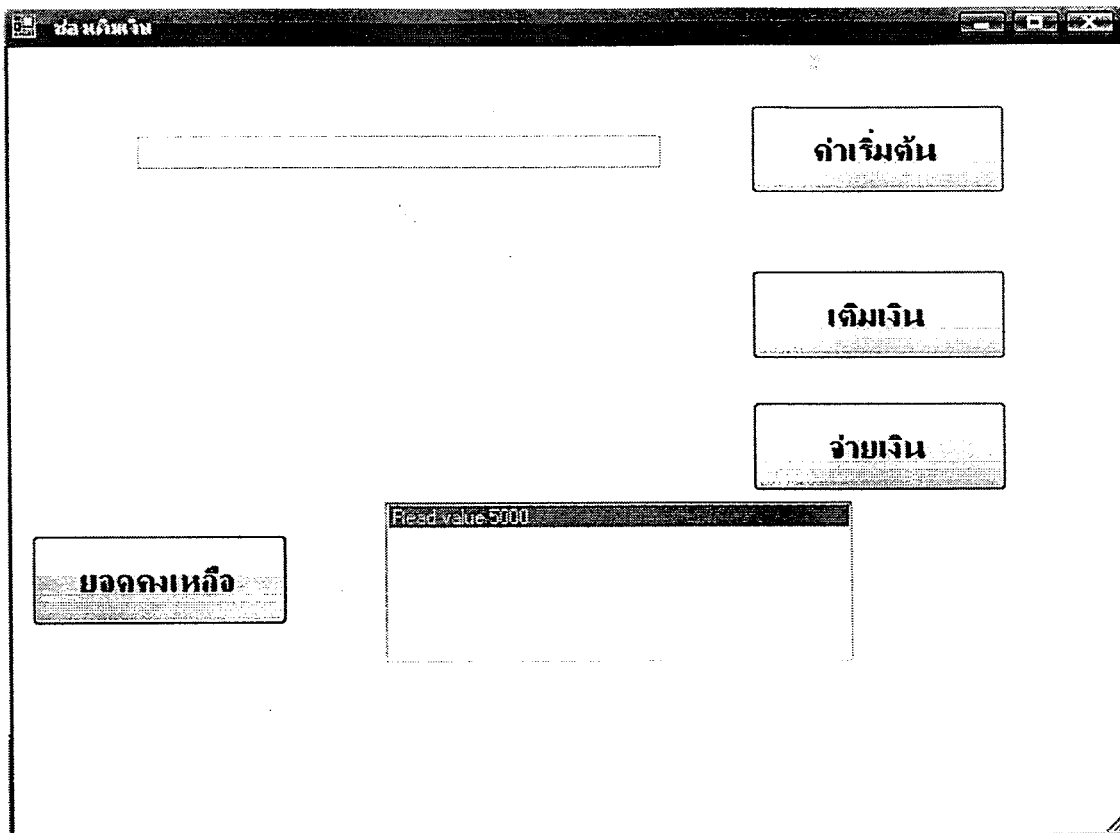
1. ทำการออกแบบหน้าหลักของ Window Applications ด้วย C#
2. เขียนโปรแกรมติดต่อกับ RFID reader และ ฐานข้อมูลผ่าน Web service
3. ทำการเติมเงินให้กับ Touch SIM เป็นจำนวนเงิน 5,000 บาท



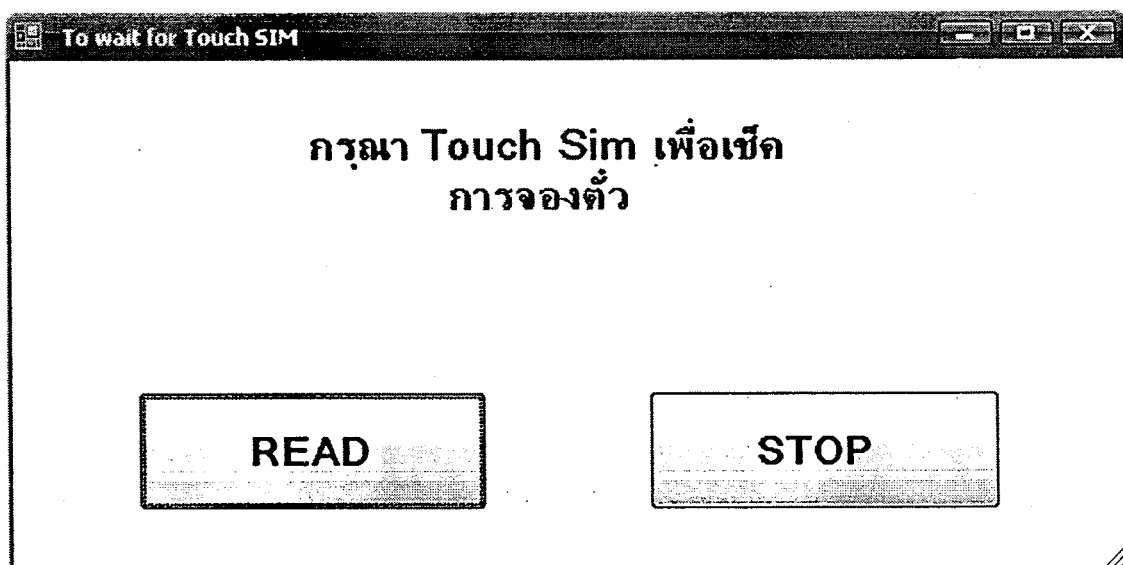
รูปที่ 4.7 แสดงหน้าต่างจำนวนเงินใน Touch SIM เป็นเงิน 0 บาท



รูปที่ 4.8 แสดงหน้าต่างเมื่อทำการเติมเงินในราคา 5,000 บาท



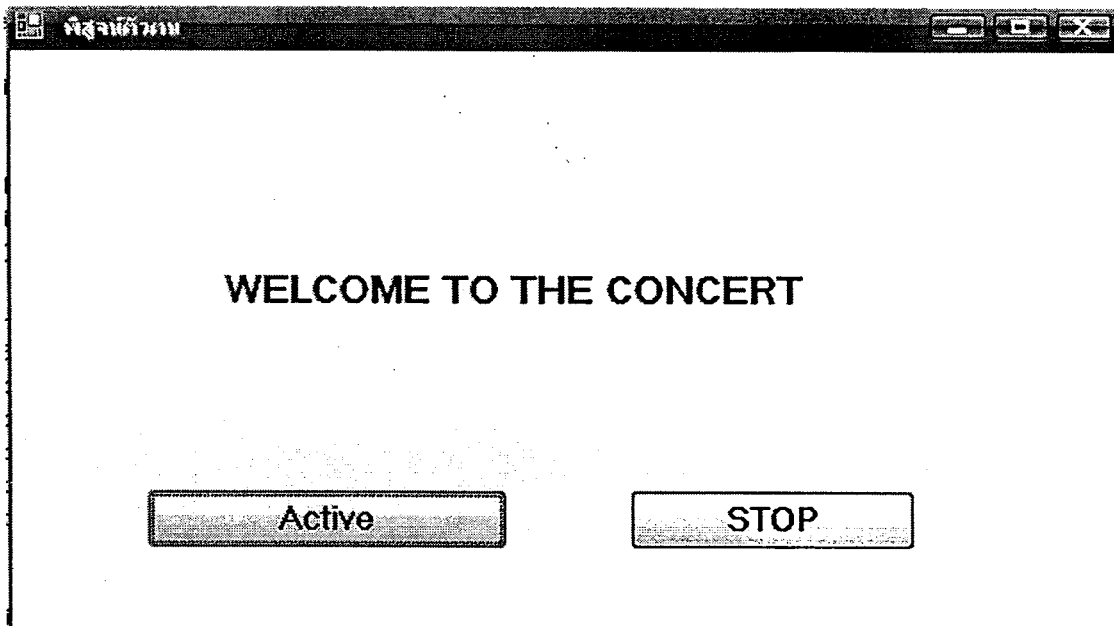
รูปที่ 4.9 แสดงหน้าต่างเมื่อทำการอ่านค่าเงินจาก Touch SIM หลังจากเติมเงินเรียบร้อยแล้ว



รูปที่ 4.10 แสดงหน้าต่างต้อนรับผู้ให้บริการเพื่อให้ผู้ให้บริการ Touch SIM

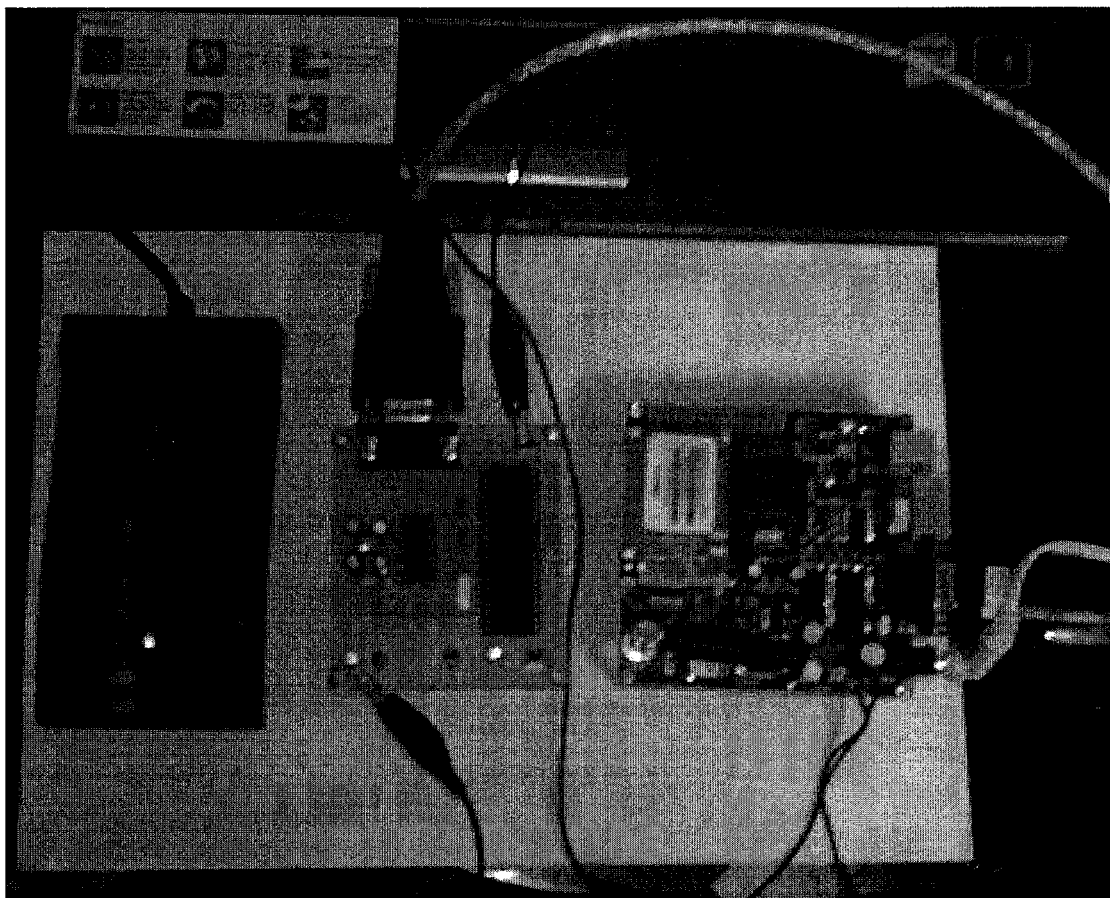
4. ทำการทดสอบจ่ายเงินตามที่ได้จอตัวคอนเสิร์ตไว้แล้ว โดยทำการอ่าน ID จาก Touch SIM
5. โดยโปรแกรมจะไปเช็ครายละเอียดต่าง ๆ ผ่านเว็บเซอร์วิส

4.3 ผลการทดลองทางเข้าคอนเสิร์ต

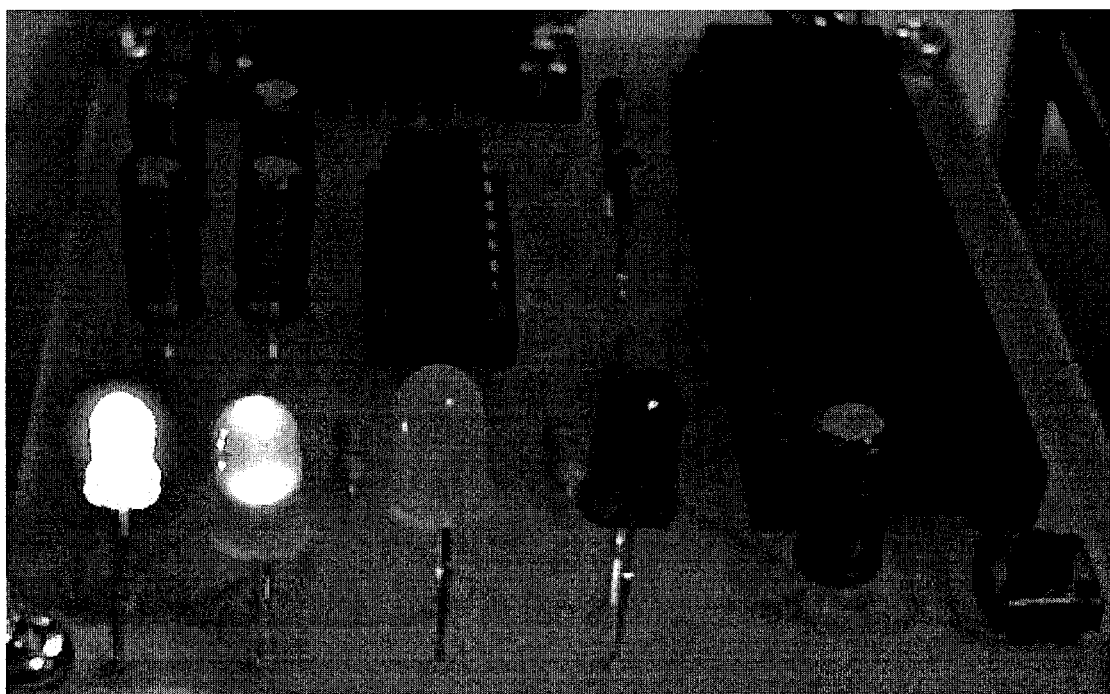


รูปที่ 4.13 แสดงการออกแบบ Window Applications ในส่วนหน้าคอนเสิร์ต

1. ทำการเขียนโปรแกรมหลัก Window Applications
2. ทำการเขียนโปรแกรมติดต่อกับ RFID Reader
3. ทำการเขียนโปรแกรมติดต่อกับฐานข้อมูลผ่านเว็บเซอร์วิส
4. ทำการออกแบบวงจรดังรูปที่ 3.9



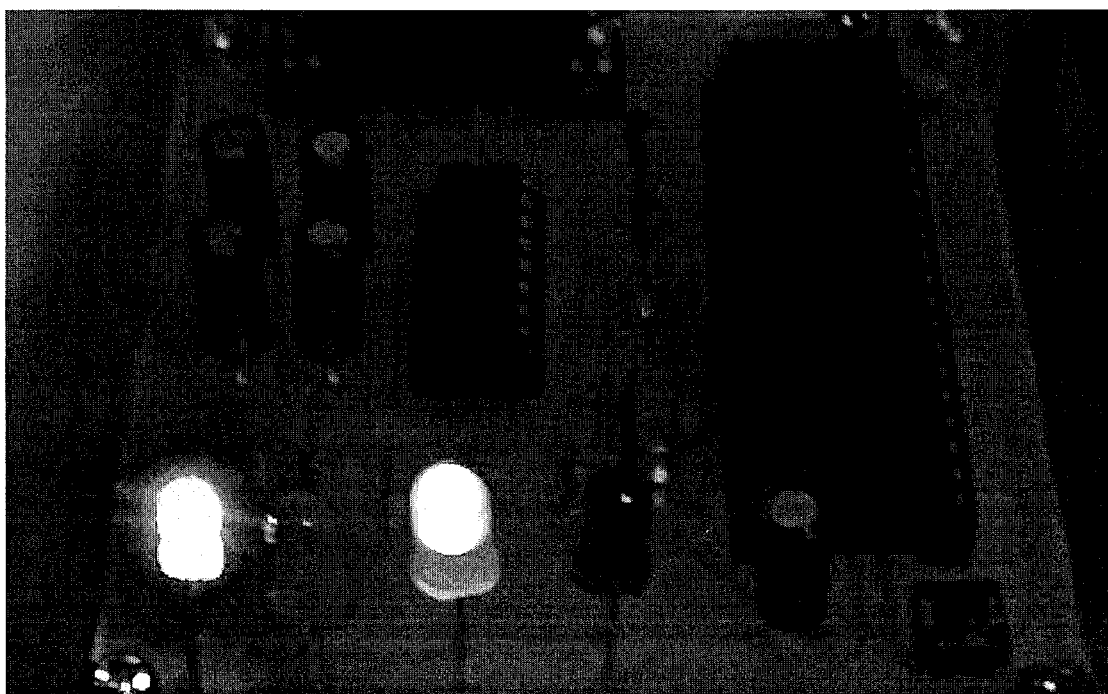
รูปที่ 4.14 ส่วนประกอบรวมในภาคน้ำคอนกรีต



รูปที่ 4.15 แสดง LED เมื่อเริ่มทำงาน

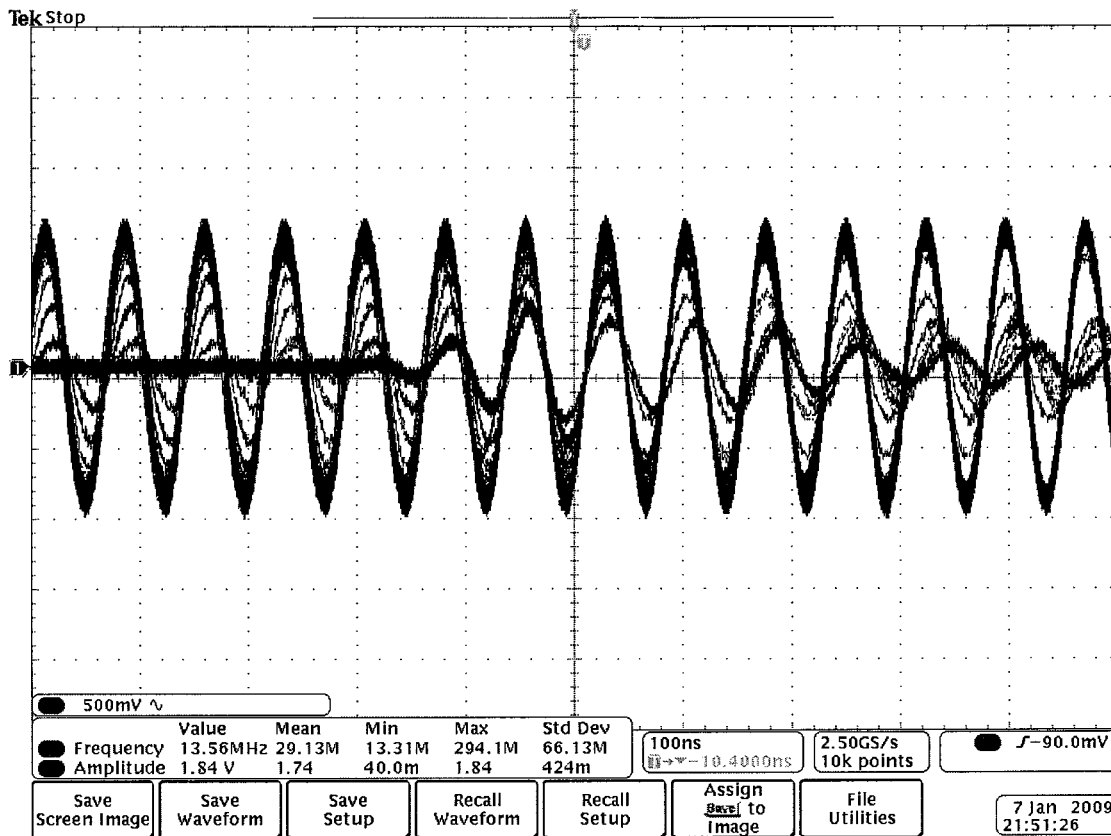


รูปที่ 4.16 แสดง LED เมื่อสมมุติให้ประตูเปิด



รูปที่ 4.17 แสดงหลอด LED เมื่อไม่ได้อนุญาตให้เข้าประตูได้

จากรูปที่ 4.19 แสดงคลื่นความถี่ของเครื่อง RFID reader/writer เมื่อยังไม่ได้ทำการส่ง-รับ ข้อมูล จึงวัดความถี่ของคลื่นแม่เหล็กไฟฟ้าได้ 13.52 MHz และ Amplitude 2.14 V



รูปที่ 4.20 แสดงสเปกตรัมเมื่อทำการส่ง-รับข้อมูล ของ RFID reader/writer

จากรูปที่ 4.20 แสดงสเปกตรัมเมื่อทำการส่ง-รับข้อมูล ของ RFID reader/writer ทำให้วัดความถี่ได้ 13.56 MHz และ Amplitude 1.84 V โดยมีการ Modulate ASK จากผลการทดลองแสดงว่าเมื่อมีการรับ-ส่งข้อมูล ความถี่และขนาดจะเปลี่ยน

บทที่ 5

สรุปผลและวิจารณ์การทดลอง

ระบบการจองตั๋วคอนเสิร์ตผ่านโทรศัพท์เคลื่อนที่และพิสูจน์ตัวตนเพื่อการออกตั๋ว เป็นแนวคิดอีกทางเลือกหนึ่งที่ทำให้ ผู้ใช้บริการได้มีโอกาสใช้เทคโนโลยีเพื่อเพิ่มความสะดวกสบายมากยิ่งขึ้น โดยได้นำเทคโนโลยีต่าง ๆ มาใช้งานร่วมกัน ที่ซึ่งมีเทคโนโลยี RFID ร่วมกับ Web service ในปัจจุบัน เทคโนโลยีบนโทรศัพท์เคลื่อนที่ได้พัฒนาความเร็วในการรับ-ส่ง ข้อมูลมากขึ้น ทำให้บริการต่างๆ ใช้เวลาเพียงน้อยนิดก็สำเร็จแล้ว จึงเกิดการพัฒนา Applications ต่าง ๆ มากมาย ในโครงการนี้เราได้เสนออีก Applications หนึ่ง ขึ้นมา โดยในโครงการนี้เราสามารถทำเสร็จสมบูรณ์ตามเป้าหมายที่วางไว้

โครงการนี้ยังสามารถนำไปพัฒนาต่อได้อีก ไม่ว่าจะเป็นส่วนของการจองตั๋วโดยการใช้ SIM Tool Kit และในส่วนของด้านคาน์เตอร์เซอร์วิสเราสามารถติดตั้งเครื่องปรี้นตั๋วคอนเสิร์ตเมื่อลูกค้าซื้อตั๋วมากกว่า 1 บัตร เพื่อนำไปแสดงตัวตนเพื่อเข้าชมงานต่อไป

กิตติกรรมประกาศ

โครงการนี้สำเร็จลุล่วงไปได้ด้วยความช่วยเหลืออย่างยิ่งของ รศ.ดร. ไกรสิน ส่องวัฒนา อาจารย์ที่ปรึกษาโครงการ ที่ได้ให้คำแนะนำและข้อคิดต่าง ๆ ในการทำงานมาโดยตลอด ขอขอบคุณ คุณพ่อ เฉลิมชัย แสงสุข คุณแม่ ประเสริฐ แสงสุข ที่ให้การเลี้ยงดูระผมมาตลอด 22 ปี

ขอขอบใจเพื่อน วนากร ไกยสิทธิ์ ที่เสียสละเวลาอันมีค่าที่ช่วยแนะนำในโครงการนี้ ขอขอบคุณ บริษัท ทูม่นนี่ จำกัด ที่ให้การสนับสนุนทั้งอุปกรณ์และค่าใช้จ่ายของโครงการนี้

บรรณานุกรม

- [1] ประภาพร ช่างไม้, คู่มือการเขียนโปรแกรม ภาษา C ฉบับผู้เริ่มต้น, บริษัท เอช เอ็น กรุป จำกัด: จังหวัด นนทบุรี, สิงหาคม 2545
- [2] สัจจะ จรัสรุ่งรวิวรร, คู่มือ Visual C# ฉบับสมบูรณ์, บริษัท ไอดีซีฯ จำกัด: จังหวัด นนทบุรี, มิถุนายน 2550
- [3] สัจจะ จรัสรุ่งรวิวรร, พัฒนาระบบฐานข้อมูลด้วย VB2005&VC#2005 ฉบับมืออาชีพ, บริษัท ไอดีซีฯ จำกัด: จังหวัด นนทบุรี, มิถุนายน 2550