สำนักหอสมุดกลาง พระจอมเกล้าลาดกระบัง

บัตรทางด่วนอัตโนมัติ โดยใช้เทคโนโลยี RFID

Express way's fee system by using RFID

โดย

นายสุธี กิจเกื้อโกศล

นายสุวัฒน์ มหาพรึกษ์พงษ์

เลขหมู่.....
เลขทะเบียน 62534
วัน,เดือน,ปี 19 ส.ค. 2549

b 11625910
i.....

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต

สาขาวิชาวิศวกรรมโทรคมนาคม

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2548

ผ่านการตรวจชิ้นงานแล้ว

(ลงชื่อ).....ผู้ตรวจ

ผ่านการตรวจรูปเล่มแล้ว

(ลงชื่อ).....ผู้ตรวจ

บัตรทางด่วนอัตโนมัติ โดยใช้เทคโนโลยี RFID

Express way's fee system by using RFID

โดย

นายสุธี กิจเกื้อโกศล 45010854

นายสุวัฒน์ มหาพรึกษ์พงษ์ 45010889

เสนอ

ผศ. นภัทร สระเอี่ยม

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต

สาขาวิชาวิศวกรรมโทรคมนาคม

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2548

ปริญญานิพนธ์ปีการศึกษา 2548

ภาควิชาวิศวกรรมโทรคมนาคม

คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เรื่อง **บัตรทางด่วนอัตโนมัติ โดยใช้เทคโนโลยี RFID**

Express way's fee system by using RFID

ผู้จัดทำ

1. **นายสุชี กิจเกื้อโกศล** **45010854**
2. **นายสุวัฒน์ มหาพรึกษ์พงษ์** **45010889**

..... อาจารย์ที่ปรึกษา

(ผศ. นภัทร ธระเอี่ยม)

บัตรทางด่วนอัตโนมัติ โดยใช้เทคโนโลยี RFID

Express way's fee system by using RFID

โดย นายสุธี กิจเกื้อโกศล 45010854
นายสุวัฒน์ มหาพรึกษ์พงษ์ 45010889

อาจารย์ที่ปรึกษา ผศ. นภัทร สระเอี่ยม

บทคัดย่อ

โครงงานนี้เป็นการนำเทคโนโลยี RFID มาใช้ในการเก็บค่าผ่านทาง โดยเทคโนโลยี RFID นี้จะประกอบด้วย Reader และ Tag ซึ่ง Tag จะถูกอ่านข้อมูลโดย Reader และใช้ Reader เป็นตัวตรวจสอบข้อมูล ซึ่งจะช่วยอำนวยความสะดวกแก่ผู้ใช้รถ และจะมีส่วนของแอปพลิเคชั่นเสริมเพื่อเพิ่มประสิทธิภาพในการใช้งานมากขึ้น

Abstract

This project presents RFID (Radio Frequency Identification) technology to apply for express way's fee system. This technology includes Reader and Tag. Tag will be used for read the user's data and Reader will be used for check data. It's will be comfortable for car user. And it has an application for more efficiency.

สารบัญ

	หน้า
บทที่ 1 บทนำ	1
1.1 วัตถุประสงค์	2
1.2 ขอบเขตของโครงงาน	2
1.3 วิธีดำเนินงาน	2
บทที่ 2 ทฤษฎีและหลักการ	3
2.1 หลักการทำงานของ RFID	3
2.1.1 เครื่องอ่านสัญญาณ (Reader)	3
2.1.2 แทค (Tag)	4
2.1.2.1 ชนิดของแทค	5
2.1.2.2.1 แทคแบบอ่านอย่างเดียว	6
2.1.2.2.2 แทคแบบเขียนได้	7
2.1.2.2.3 แทคที่มีฟังก์ชัน cryptological	7
2.1.3 แนวความคิดของ Hierarchy key	8
2.1.3.1 ตัวอย่างการประยุกต์ใช้ Hierarchy key	8
2.1.4 เทคโนโลยีของหน่วยความจำ	10
2.1.5 ความถี่ของคลื่นพาหะ	11
2.1.6 การทำงานของระบบ RFID	11
2.1.7 ประเภทของ ระบบ RFID	12
2.1.7.1 RFID ที่จำแนกโดยขนาดของหน่วยความจำ	12
2.1.7.1.1 RFID ชนิด 1 บิท (2 bit type)	12
2.1.7.1.2 RFID ชนิดหน่วยความจำมากกว่า 1 บิท (Data Carrier Type)	14
2.1.7.2 RFID ที่จำแนกโดยลักษณะการคล้องของสัญญาณ	14
2.1.7.2.1 Close Coupling	14
2.1.7.2.2 Remote Coupling	14
2.1.7.2.3 Long Range	15
2.1.8 เทคโนโลยีการเข้ารหัสและถอดรหัส	16
2.1.8.1 การเข้ารหัสข้อมูล (Signal Coding)	16
2.1.8.1.1 NRZ Coding	16
2.1.8.1.2 Biphase Coding	16

หน้า

2.1.8.2 การผสมรหัสข้อมูลเข้ากับคลื่นพาหะ (Modulation) 17

2.1.8.2.1 Amplitude Shift Keying (ASK) 17

2.1.8.2.2 Frequency Shift Keying (FSK) 18

2.1.8.2.3 Phase Shift Keying (PSK) 18

2.1.8.3 การส่งคลื่นสัญญาณออกไป (Transmission) 19

2.1.8.4 การถอดข้อมูลออกจากคลื่นพาหะ (Demodulation) 19

2.1.9 หลักการและเทคนิคที่ใช้รับและส่งข้อมูลระหว่าง แทคและเครื่องอ่าน 19

2.1.10 การชนกันของข้อมูล 20

2.1.10.1 ขั้นตอนการทำงานระหว่างเครื่องอ่านกับแทค 20

2.2 หลักการของฟิสิคัลของระบบ RFID 21

2.2.1 สนามแม่เหล็ก 21

2.2.2 เส้นทางของกำลังสนาม $H(x)$ ในวงตัวนำ 22

2.2.3 เส้นผ่านศูนย์กลางของสายอากาศที่ดีที่สุด 24

2.2.4 ฟลักซ์แม่เหล็กและความหนาแน่นของฟลักซ์แม่เหล็ก 25

2.2.5 ความเหนี่ยวนำของลูปตัวนำ 26

2.2.6 ความเหนี่ยวนำเสมือน M 26

2.2.7 สัมประสิทธิ์การคัปปลิ้ง k 28

2.2.9 มาตรฐานอาร์เอฟไอดี 29

2.3 โปรแกรมที่ใช้สร้างแอปปลิเคชัน 30

2.3.1 โปรแกรม Delphi 30

2.3.2 โปรแกรม MySQL 31

2.3.3 ภาษา PHP 32

2.3.3.1 รูปแบบภาษา 32

2.3.3.2 สิ่งที่ต้องมีในการใช้ PHP 32

2.3.3.3 PHPMyAdmin 32

2.4 มาตรฐาน RS-485 34

บทที่ 3 การคำนวณและการออกแบบ 36

3.1 การออกแบบส่วนของ RFID 36

3.1.1 EM4095 : Read/Write analog front end for 125 kHz RFID Basestation 36

3.1.1.1 Pin Description 37

3.1.1.2 การกำหนดการทำงาน 37

3.1.2 Tag : Multifunctional 330 bit Read/Write RF-Identification IC 38

หน้า

3.1.2.1 T5557 Building Blocks 38

3.1.2.2 คุณสมบัติของ T5557 39

3.1.2.3 การระบบการสั่งงานของ T5557 39

3.1.2.3.1 Initialization and POR Delay 39

3.1.2.3.2 Tag to Reader Communication 39

3.1.2.1.3 Regular-read Mode 39

3.1.2.1.4 Block-read Mode 40

3.1.2.1.5 e5550 Sequence Terminator (ST) 40

3.1.2.1.6 Reader to Tag Communication 41

3.1.2.2 หลักในการเขียนข้อมูล 42

3.1.2.3 PASSWORD 42

3.2 โปรโตคอลการติดต่อสื่อสาร 43

3.2.1 คำสั่งการอ่านแทค 43

3.2.2 คำสั่งการเขียน 43

3.2.3 โค๊ตคำสั่ง 44

3.2.4 โค๊ตของเฮดเดอร์และสถานะ 44

3.2.5 โค๊ตเซ็กเมนต์ 44

3.2.6 การหา CRC ของแทค 44

3.2.7 คุณสมบัติในการติดต่อสื่อสาร 45

3.2.8 การเขียนข้อมูลลงในแทค 45

3.3 พอร์ตสื่อสารแบบขนาน (Parallel Port) 45

3.4 โฟล์วชาร์ตแสดงการทำงานต่างๆ 46

3.4.1 โฟล์วชาร์ตแสดงการทำงานของระบบ 46

3.4.2 โฟล์วชาร์ตแสดงการทำงานของสถานีต้นทาง 47

3.4.3 โฟล์วชาร์ตแสดงการทำงานของสถานีปลายทาง 48

3.4.4 โฟล์วชาร์ตแสดงการทำงานของการเติมเงิน 49

3.5 วงจรที่ใช้ในการติดต่อกับคอมพิวเตอร์ 50

บทที่ 4 การทดลองและผลการทดลอง 51

4.1 ไอซี EM 4095 51

4.2 ผลการวัดความถี่และขนาดของสัญญาณที่วัดจากสายอากาศ จากวงจรไอซี EM4095 เมื่อไม่มีแทคอยู่ในระยะที่สามารถอ่านได้ 52

หน้า

4.3 ผลการวัดความถี่และขนาดของสัญญาณที่วัดจากสายอากาศขาที่ 1 จากวงจรไอซี EM4095 เมื่อมีแทค 53

4.4 ผลการวัดสัญญาณของสายอากาศขาที่ 1 ที่รับเข้ามาจากแทคและผ่านวงจรของไอซี EM4095 ขณะมีแทคอยู่ในระยะที่อ่านได้ 54

4.5 การวัดสัญญาณส่วนหัวของข้อมูล 54

4.6 ผลการวัดขนาดของแรงดันที่วัดจากวงจรเซ็นเซอร์ 56

4.7 แสดงรูปหน้าต่างเพื่อเลือกว่าเป็นหน้าต่างสถานีต้นทาง สถานีปลายทาง เติมเงิน หรือออกจากระบบ 57

4.8 แสดงรูปหน้าต่างโปรแกรมสถานีต้นทาง 57

4.9 แสดงรูปหน้าต่างโปรแกรมสถานีปลายทาง 61

4.10 แสดงรูปหน้าต่างเติมเงิน 63

4.11 แสดงรูปฐานข้อมูล 66

บทที่ 5 สรุปและวิจารณ์ 72

ภาคผนวก

สารบัญรูปภาพ

		หน้า
รูปที่ 2.1	แสดงภาพรวมของระบบ RFID	3
รูปที่ 2.2	แสดงโครงสร้างภายในเครื่องอ่าน	3
รูปที่ 2.3	แสดง RFID แทคในรูปแบบต่างๆ	4
รูปที่ 2.4	แสดงบล็อกไดอะแกรมของแทคแบบพาสซีฟ	5
รูปที่ 2.5	แสดงรูปตัวอย่างแทคแบบแอคทีฟที่มีแบตเตอรี่ลิเทียม 2 ก้อนอยู่ภายนอก	6
รูปที่ 2.6	แสดงบล็อกไดอะแกรมของแทคแบบอ่านอย่างเดียว เมื่อแทคเข้าสู่บริเวณการตรวจสอบของเครื่องอ่านสัญญาณ การนับจะเริ่มนับทุกแอดเดรสที่อยู่ในหน่วยความจำอย่างต่อเนื่อง เอาท์พุทของข้อมูลของหน่วยความจำจะเชื่อมต่อกับโหลดมอดูเลเตอร์ที่เซตเป็นรหัสเบสแบนด์ของรหัสไบนารี่ ในกระบวนการนี้ข้อมูลทั้งหมดในหน่วยความ (128 บิทซีเรียลนัมเบอร์) สามารถส่งไปได้อย่างต่อเนื่อง	6
รูปที่ 2.7	แสดงการเปรียบเทียบขนาดชิพแทคกับรูของเข็ม	7
รูปที่ 2.8	แสดงบล็อกไดอะแกรมของแทคที่เขียนได้ที่มีฟังก์ชั่น cryptological ระหว่างแทคกับเครื่องอ่านสัญญาณ	8
รูปที่ 2.9	แสดงแทคที่มี 2 หน่วยความจำ	8
รูปที่ 2.10	แสดงการประยุกต์ใช้หลายประเภทใน 1 แทกส์ซึ่งแต่ละอันมีการป้องกันโดย key ของมันเอง	9
รูปที่ 2.11	แสดงความแตกต่างระหว่างเซกเมนต์ที่แน่นอนกับฟรีเซกเมนต์	10
รูปที่ 2.12	แสดงการสื่อสารระหว่างเครื่องอ่านสัญญาณกับแทค	12
รูปที่ 2.13	แสดงวงจรเทียบเคียงของระบบ RFID	12
รูปที่ 2.14	แสดงแทคของระบบ EAS จะมีเฉพาะเสาอากาศอย่างเดียว	13
รูปที่ 2.15	แสดงเสาอากาศของระบบ EAS	13
รูปที่ 2.16	แสดงวงจรเทียบเคียงของ Close – Coupling	14
รูปที่ 2.17	แสดงวงจรเทียบเคียงของ Remote Coupling หรือ Inductive Coupling	15
รูปที่ 2.18	แสดงวงจรเทียบเคียง Long Range	15
รูปที่ 2.19	แสดงการไหลของสัญญาณและข้อมูลในระบบการสื่อสารแบบดิจิตอล	16
รูปที่ 2.20	แสดงรหัสสัญญาณ NRZ	16
รูปที่ 2.21	แสดงรหัสสัญญาณ Biphase	17
รูปที่ 2.22	แสดงความสูงของยอดคลื่นใน ASK จะเปลี่ยนสถานะอยู่ 2 สถานะขึ้นอยู่กับข้อมูลที่ผสมเข้ามา	17
รูปที่ 2.23	แสดง 2FSK จะเปลี่ยนแปลงความถี่ของคลื่นตามข้อมูลที่ผสมเข้ามา	18
รูปที่ 2.24	แสดง 2PSK จะกลับเฟสทุกครั้งเมื่อมีการเปลี่ยนสถานะของข้อมูล	18
รูปที่ 2.25	แสดงรูปคลื่นของสัญญาณระหว่างแทคและเครื่องอ่านแบบ AM	19

หน้า

รูปที่ 2.26 (ซ้าย) แสดงลูปตัวนำที่ถูกเหนี่ยวนำด้วยสนามแม่เหล็ก (ขวา) วงจรที่สมมูลกับรูปทางซ้ายมือ 20

รูปที่ 2.27 แสดงเส้นของฟลักซ์แม่เหล็กถูกสร้างรอบๆ ตัวนำที่มีกระแส 21

รูปที่ 2.28 แสดงเส้นของฟลักซ์แม่เหล็กรอบตัวนำกระแสและกระแสรอบขดลวดทรงกระบอก 22

รูปที่ 2.29 แสดงเส้นของฟลักซ์แม่เหล็กรอบขดลวดทรงกระบอกสั้น หรือลูปตัวนำ ซึ่งคล้ายกับในสายอากาศส่งของระบบ RFID 23

รูปที่ 2.30 แสดงเส้นทางกำลังสนามแม่เหล็ก H ในสนามระยะใกล้ของขดลวดทรงกระบอกสั้น หรือขดลวดตัวนำ ที่ระยะทางในทิศ x เพิ่มขึ้น 24

รูปที่ 2.31 แสดงเส้นทางของกำลังสนาม H ที่ระยะ $x = 20mm$ ซึ่งรัศมีของขดลวด $R = 5 - 55mm$ 25

รูปที่ 2.32 แสดงความสัมพันธ์ระหว่างฟลักซ์แม่เหล็ก Φ และความหนาแน่นฟลักซ์ B 26

รูปที่ 2.33 แสดงการกำหนดความเหนี่ยวนำเสมือน M_{21} โดยการคัปปลิ้งของขดลวดสองอันผ่านการไหลของแม่เหล็กส่วนหนึ่ง 27

รูปที่ 2.34 แสดงเส้นทางของความเหนี่ยวนำเสมือนระหว่างสายอากาศของ Reader และแทค ในระยะทางทิศทาง x ที่เพิ่มขึ้น 27

รูปที่ 2.35 แสดงสัมประสิทธิ์การคัปปลิ้งสำหรับลูปตัวนำที่มีขนาดแตกต่างกัน สายอากาศแทค : $r_{Transp} = 2cm$ สายอากาศของเครื่องอ่าน: $r_1 = 10cm, r_2 = 7.5cm, r_3 = 1cm$ 28

รูปที่ 2.36 แสดงรูปแบบในการจัดเก็บฐานข้อมูล 32

รูปที่ 3.1 แสดงขาของไอซี EM4095 36

รูปที่ 3.2 แสดงวงจรที่ใช้ต่อ EM4095 กับไมโครคอนโทรเลอร์ 36

รูปที่ 3.3 แสดง RFID System Using T5557 Tag 38

รูปที่ 3.4 แสดงบล็อกไดอะแกรมของ T5557 38

รูปที่ 3.5 แสดงตัวอย่างของความแตกต่างของการเซต MAXBLK 40

รูปที่ 3.6 แสดงRead Data Stream with Sequence Terminator 40

รูปที่ 3.7 แสดง e5550-compatible Sequence Terminator Waveforms 41

รูปที่ 3.8 แสดงการเริ่มต้นการสื่อสารของเครื่องอ่านสัญญาณต่อแทค 41

รูปที่ 3.9 แสดงรูปร่างสัญญาณขณะเขียนข้อมูล 42

รูปที่ 3.10 แสดงรูปแบบในการเขียน 42

รูปที่ 3.11 แสดงการเขียนข้อมูลลงในแทค 45

รูปที่ 3.12 แสดงโฟล์วชาร์ดการทำงานของระบบ 46

รูปที่ 3.13 แสดงโฟล์วชาร์ดการทำงานของสถานีต้นทาง 47

รูปที่ 3.14 แสดงโฟล์วชาร์ดการทำงานของสถานีปลายทาง 48

รูปที่ 3.15 แสดงโฟล์วชาร์ดการทำงานของการเติมเงิน 49

หน้า

รูปที่ 3.16 แสดงรูปวงจรที่ใช้ในการติดต่อกับคอมพิวเตอร์ 50
รูปที่ 4.1 แสดงวงจรของไอซี EM4095 51
รูปที่ 4.2 แสดงความถี่และขนาดของสัญญาณที่ออกจากสายอากาศเมื่อไม่มีแทค 52
รูปที่ 4.3 แสดงการเปรียบเทียบความถี่ของสัญญาณที่ออกจากสายอากาศขาที่ 1 และขาที่ 2 52
รูปที่ 4.4 แสดงสัญญาณที่ออกจากสายอากาศที่ 1 เมื่อมีแทคอยู่ในระยะที่อ่านได้ 53
รูปที่ 4.5 แสดงรูปการขยายสัญญาณที่ออกจากสายอากาศ 53
รูปที่ 4.6 แสดงสัญญาณที่เข้ามาที่สายอากาศขาที่ 1 และผ่านไอซี EM 4095 ออกมาทางขา DEMOD_OUT 54
รูปที่ 4.7 แสดงส่วนของเฮดเดอร์ที่วัดจากสายอากาศขาที่ 1 54
รูปที่ 4.8 แสดงการเปรียบเทียบระหว่างเฮดเดอร์ของสัญญาณที่ได้จากสายอากาศขาที่ 1 กับสัญญาณที่ออกมาจากไอซี EM 4095 55
รูปที่ 4.9 แสดงขนาดของแรงดันเมื่อไม่มีวัตถุอยู่ระหว่างตัวส่งและตัวรับของเซ็นเซอร์ 56
รูปที่ 4.10 แสดงขนาดของแรงดันเมื่อมีวัตถุอยู่ระหว่างตัวส่งและตัวรับของเซ็นเซอร์ 56
รูปที่ 4.11 แสดงหน้าต่างเพื่อเลือกว่าเป็นหน้าต่างสถานีต้นทาง สถานีปลายทาง เติมเงิน หรือออกจากระบบ 57
รูปที่ 4.12 แสดงการ Open Port เมื่อเข้าสู่หน้าต่างทางด้านสถานีต้นทาง 57
รูปที่ 4.13 แสดงการเลือกสถานีเพื่อจำลองเป็นสถานีต้นทาง 58
รูปที่ 4.14 แสดงรูปเมื่อทำการ Open Port แล้ว 58
รูปที่ 4.15 แสดงการทำงานของเซ็นเซอร์เมื่อมีรถยนต์ผ่าน 59
รูปที่ 4.16 แสดงรายละเอียดที่หน้าของโปรแกรม เมื่อรถยนต์ที่มีแทค ผ่าน 59
รูปที่ 4.17 แสดงรูปเมื่อรถยนต์ที่ไม่มีแทควิ่งเข้ามา 60
รูปที่ 4.18 แสดงรูปหน้าต่างของโปรแกรมเมื่อมีการบันทึกวีดีโอคลิป 60
รูปที่ 4.19 แสดงการ Open Port เมื่อเข้าสู่หน้าต่างทางด้านสถานีปลายทาง 61
รูปที่ 4.20 แสดงการทำงานของเซ็นเซอร์เมื่อมีรถยนต์ผ่าน 61
รูปที่ 4.21 แสดงรูปการจำลองสถานีปลายทางและการแสดงค่าใช้จ่าย 62
รูปที่ 4.22 แสดงรายละเอียดที่หน้าของโปรแกรม เมื่อรถยนต์ที่มีแทค ผ่าน 62
รูปที่ 4.23 แสดงการ Open Port เมื่อเข้าสู่หน้าต่างการเติมเงิน 63
รูปที่ 4.24 แสดงรายละเอียดที่หน้าของโปรแกรมเมื่อต้องการจะเติมเงิน 63
รูปที่ 4.25 แสดงรายละเอียดที่หน้าของโปรแกรม เมื่อเติมเงินเรียบร้อยแล้ว 64
รูปที่ 4.26 แสดงยอดเงินที่เติมแล้ว 64
รูปที่ 4.27 แสดงรายละเอียดข้อมูลของผู้ใช้ 65
รูปที่ 4.28 แสดงการบันทึกรายละเอียดการเติมเงินลงในฐานข้อมูล 65
รูปที่ 4.29 แสดงหน้าเวปเพจเพื่อที่จะเข้าไปดูไฟล์ที่บันทึกไว้ 66

หน้า

รูปที่ 4.30 แสดงหน้าเวปเพจเมื่อทำการลงชื่อผู้ใช้และใส่รหัสผ่าน 66
รูปที่ 4.31 แสดงหน้าเว็ปเพจเมื่อต้องการเข้าไปดูวีดีโอคลิป 67
รูปที่ 4.32 แสดงหน้าเว็ปเพจเมื่อต้องการดูภาพปัจจุบัน 67
รูปที่ 4.33 แสดงหน้าฐานข้อมูล MySQL 68
รูปที่ 4.34 แสดงตารางฐานข้อมูล 68
รูปที่ 4.35 แสดงชื่อสมาชิกที่อยู่ในตารางฐานข้อมูล 69
รูปที่ 4.36 แสดงรายละเอียดของการบันทึก 69
รูปที่ 4.37 แสดงรายละเอียดที่ผู้ใช้ใช้บริการ 70
รูปที่ 4.38 แสดงฟิลด์ของวีดีโอ 70
รูปที่ 4.39 แสดงรายชื่อคลิปวีดีโอที่สามารถบันทึกไว้ได้ 71

สารบัญตาราง

หน้า

ตารางที่ 2.1 แสดงย่านความถี่ต่างๆ ของระบบ RFID และการใช้งาน 11

ตารางที่ 2.2 แสดงการเปรียบเทียบมาตรฐานระหว่าง ISO/IEC และ EPC 29

ตารางที่ 3.1 แสดง Pin Description 37

ตารางที่ 3.2 แสดงWrite Data Decoding Scheme 41

ตารางที่ 3.3 แสดงคำสั่งการส่งของ GUI ในการอ่านแทค 43

ตารางที่ 3.4 แสดงคำสั่งการตอบสนองของเครื่องอ่านกรณีไม่มีความผิดพลาดในการตรวจสอบ CRC 43

ตารางที่ 3.5 แสดงคำสั่งการตอบสนองของเครื่องอ่านกรณีมีความผิดพลาดในการตรวจสอบ CRC 43

ตารางที่ 3.6 แสดงคำสั่งการส่งของ GUI ในการเขียนแทค 43

ตารางที่ 3.7 แสดงคำสั่งการตอบสนองของเครื่องอ่าน 43

ตารางที่ 3.8 แสดงโค๊ตคำสั่ง 44

ตารางที่ 3.9 แสดงโค๊ตของเฮดเดอร์และสถานะ 44

ตารางที่ 3.10 แสดงโค๊ตเซ็กเมนต์ 44

บทที่ 1
บทนำ

ปัจจุบันระบบบ่งชี้อัตโนมัติ (Automatic Identification) ถูกนำมาใช้งานอย่างกว้างขวาง ซึ่งได้มีการพัฒนามาอย่างต่อเนื่อง ตั้งแต่ระบบบาร์โค๊ต (Barcode System) ซึ่งได้ใช้งานกันอย่างแพร่หลาย สามารถพบได้ง่ายในชีวิตประจำวัน เนื่องจากมีต้นทุนต่อหน่วยต่ำ จึงถูกนำมาใช้งานกันมากที่สุด แต่มีข้อเสียคือ เกิดความเสียหายได้ง่าย จัดเก็บข้อมูลได้จำกัด ระบบต่อมาคือระบบสมาร์ทการ์ด (Smartcard System) เป็นระบบที่มีการพัฒนามาจากระบบบาร์โค๊ต มีข้อดีคือสามารถจัดเก็บข้อมูลได้มากกว่าระบบบาร์โค๊ต แต่เนื่องจากใช้แถบแม่เหล็ก ระบบสมาร์ทการ์ดจึงต้องมีการสัมผัสทำให้เกิดการสึกหรอได้ง่าย จึงมีการพัฒนามาเป็นระบบ RFID (Radio Frequency Identification) ซึ่งเป็นระบบที่ใช้คลื่นแม่เหล็กไฟฟ้าในช่วงคลื่นความถี่วิทยุเป็นพาหะในการสื่อสารข้อมูล ซึ่งระบบ RFID มีข้อดีคือ

- อ่าน/เขียนข้อมูลโดยไม่ต้องสัมผัส
- ทนต่อสภาพแวดล้อมและสิ่งสกปรก
- สามารถอ่าน/เขียนข้อมูลได้สะดวก
- สื่อสารได้ทุกทิศทาง
- แทคสามารถนำกลับมาใช้ใหม่ได้
- แทคมีหลายแบบให้ประยุกต์ใช้งาน
- ความสามารถในการทะลุทะลวงของสัญญาณ
- สื่อสารได้ระยะไกล
- หน่วยความจำขนาดใหญ่
- อ่าน/เขียนข้อมูลได้ครั้งละมากกว่า 1 แทคพร้อมกัน
- สามารถอ่าน/เขียนข้อมูลได้ขณะที่วัตถุกำลังเคลื่อนที่

และสามารถประยุกต์ใช้งานต่างๆ ได้ เช่น

- บัตรทางด่วน
- บัตรเข้าออกสำนักงาน
- บัตรจอดรถ
- บัตรสหกรณ์
- บัตรรับประทานอาหาร
- ระบบเช็คสต๊อกสินค้า
- ระบบห้องสมุดอัตโนมัติ

1.1 วัตถุประสงค์

1. ศึกษาการส่งผ่านข้อมูลระหว่างแทคกับเครื่องอ่านสัญญาณ
2. ศึกษาการส่งผ่านข้อมูลจากเครื่องอ่านสัญญาณไปยังคอมพิวเตอร์โดยผ่านพอร์ตอนุกรม
3. ศึกษาการสร้างระบบฐานข้อมูลโดยใช้ MySQL
4. ศึกษาการการเชื่อมฐานข้อมูลกับเว็ปไซต์โดยการใช้ PHP
5. ศึกษาการทำงานของวงจรเซ็นเซอร์และกล้อง
6. ศึกษาและนำโปรแกรม Delphi มาใช้ในการสร้างแอปพลิเคชั่น

1.2 ขอบเขตของโครงงาน

1.ในส่วนของการเชื่อมต่อระหว่างเครื่องอ่านสัญญาณกับคอมพิวเตอร์จะใช้โปรแกรม Delphi เขียนโค๊ตเพื่อทำการเชื่อมต่อ ซึ่งยังใช้โปรแกรม Delphi ในการสร้างหน้าแอปพลิเคชั่น

2.ในส่วนของฐานข้อมูลจะใช้โปรแกรม MySQL ในการจัดการฐานข้อมูล และใช้ PHP เชื่อมต่อฐานข้อมูลและนำมาแสดงผลบนเว็ปไซต์

3.ในส่วนของการแสดงผลบนเว็ปไซต์ จะใช้โปรแกรม PHP และ Dreamweaver

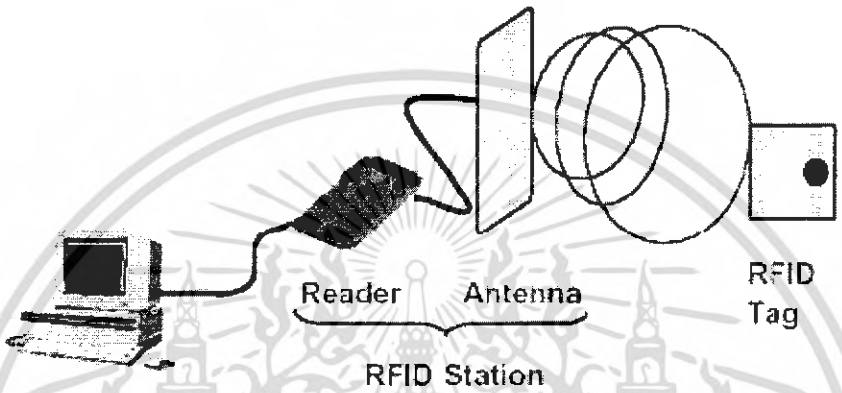
1.3 วิธีดำเนินงาน

ในการทำงานจะเริ่มต้นโดยทำการศึกษาหลักการต่างๆ ของเครื่องอ่านสัญญาณ(Reader), แทค (Tag) ทำการวัดสัญญาณต่างๆ จากเครื่องอ่านสัญญาณ และหลักการต่างๆ ที่เกี่ยวข้องกับเทคโนโลยี RFID ศึกษาการอ่านข้อมูลจากแทคและเขียนข้อมูลลงไปในแทค ต่อจากนั้นจะทำการสร้างแอปพลิเคชันด้วยโปรแกรม Delphi โดยใช้ MySQL ในการจัดการฐานข้อมูล และใช้ PHP ในการเชื่อมระหว่าง MySQL ซึ่งก็คือฐานข้อมูลกับเว็ปไซต์ ซึ่งจะทำให้สามารถเรียกดูข้อมูลผ่านทางเว็ปไซต์ได้

บทที่ 2
ทฤษฎีและหลักการ

2.1 หลักการทำงานของ RFID

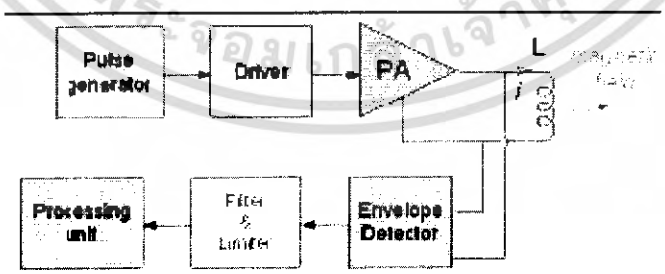
RFID (Radio Frequency Identification) คือ ระบบชี้เฉพาะอัตโนมัติ (Automatic Identification) แบบไร้สัมผัสซึ่งประกอบด้วยส่วนประกอบหลัก 2 ส่วนคือ 1. เครื่องอ่านสัญญาณ (Reader) 2. ตัวแทค (Tag)



รูปที่ 2.1 แสดงภาพรวมของระบบ RFID

2.1.1 เครื่องอ่านสัญญาณ (Reader)

หน้าที่ของเครื่องอ่านคือ การเชื่อมต่อเพื่ออ่านหรือเขียนข้อมูลลงในแทคด้วยสัญญาณความถี่วิทยุ ภายในเครื่องอ่านจะประกอบด้วย เสาอากาศที่ทำจากขดลวดทองแดง เพื่อใช้รับ-ส่งสัญญาณ ภาครับและภาคส่งสัญญาณวิทยุ และวงจรควบคุมการอ่านและเขียนข้อมูล ซึ่งมักจะเป็นวงจรจำพวกไมโครคอนโทรลเลอร์และส่วนของการติดต่อกับคอมพิวเตอร์



รูปที่ 2.2 แสดงโครงสร้างภายในเครื่องอ่าน

จากรูปที่ 2.2 โครงสร้างภายในเครื่องอ่าน ซึ่งมีองค์ประกอบหลักเริ่มจากส่วนกำเนิดสัญญาณรูปเหลี่ยม (pulse generator) ความถี่พาห์เพื่อส่งสัญญาณไปยังภาคขับ (driver) เพื่อเพิ่มสมรรถนะในการขับภาคขยายกำลัง (Power Amplifier, PA) ซึ่งทำหน้าที่ขับกระแสสัญญาณต่อไปยังขดลวดเพื่อเหนี่ยวนำให้

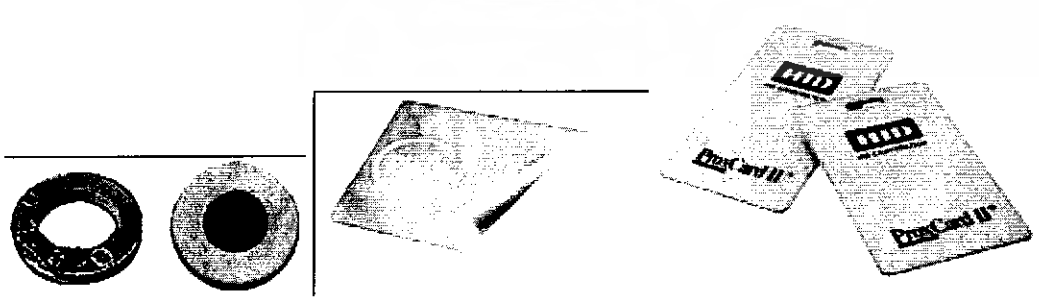
เกิดสนามแม่เหล็กเชื่อมโยงไปยังส่วนแทค ขณะเดียวกันส่วนขดลวดดังกล่าวก็จะทำหน้าที่เสมือนเป็นสายอากาศ (Antenna) รับสัญญาณสนามแม่เหล็กความถี่คลื่นพาห์ที่ถูกมอดูเลตเชิงขนาดจากข้อมูลจำเพาะของส่วนแทค จากนั้นส่วนตรวจจับขอบ (envelop detector) ที่ทำงานร่วมกับส่วนขยายสัญญาณแบบต่ำผ่าน (filter and limiter) ก็จะแยกข้อมูลออกจากสัญญาณคลื่นพาห์และขยายจนกระทั่งได้ระดับศักดาของข้อมูลตามมาตรฐานลอจิก เพื่อส่งต่อเข้าส่วนประมวลผลข้อมูล (processing unit) ต่อไป

โดยทั่วไปหน่วยประมวลข้อมูลที่อยู่ภายในเครื่องอ่านมักใช้เป็นไมโครคอนโทรลเลอร์ ซึ่งอัลกอริทึมที่อยู่ภายในโปรแกรม จะทำหน้าที่ถอดรหัสข้อมูล (decoding) ที่ได้รับ และทำหน้าที่ติดต่อกับคอมพิวเตอร์ โดยลักษณะ ขนาด และ รูปร่างของเครื่องอ่านจะแตกต่างกันไปตามประเภทของการใช้งาน เช่น แบบมือถือ ขนาดเล็ก หรือติดผนัง จนไปถึงขนาดใหญ่เท่าประตู (Gate size) เป็นต้น

2.1.2 แทค (Tag/Transponder)

โครงสร้างภายในของแทคจะประกอบด้วย 2 ส่วนใหญ่ๆ ได้แก่ ส่วนของไมโครชิป (Microchip) ที่ทำหน้าที่เป็นเก็บข้อมูลของวัตถุ และขดลวดขนาดเล็กหรือแบบบางๆ ซึ่งทำหน้าที่เป็นสายอากาศ (Antenna) สำหรับรับ-ส่งสัญญาณคลื่นความถี่วิทยุและสร้างพลังงานป้อนให้ส่วนของไมโครชิป โดยทั่วไปตัวแทคอาจอยู่ในรูปแบบที่เป็นกระดาษ แผ่นฟิลม์ พลาสติก มีขนาดและรูปร่างต่างๆกันไป ทั้งนี้ขึ้นอยู่กับวัสดุที่จะนำเอาไปติด และมีหลายรูปแบบ เช่น บัตรเครดิต เหรียญ กระดุม ฉลากสินค้า แคปซูล หรือป้าย เป็นต้น ทั้งนี้เราสามารถแบ่งแทคที่มีใช้งานกันอยู่ได้เป็น 3 ชนิดใหญ่ๆ ได้แก่ แทคแบบแพสซีฟ แทคแบบกึ่งแพสซีฟและแทคแบบแอคทีฟ โดยแต่ละชนิดก็จะมีความแตกต่างกันตามการใช้งาน ราคาโครงสร้างและหลักการทำงาน

นอกจากการแบ่งจากชนิดที่กล่าวมาแล้ว เราสามารถที่จะแบ่งประเภทของแทคจากรูปแบบการอ่านและหรือบันทึกข้อมูลได้เป็น 3 แบบคือ แทคชนิดที่สามารถถูกอ่านและเขียนข้อมูลได้หลายครั้ง (Read-Write) แทคชนิดที่เขียนได้เพียงครั้งเดียวเท่านั้นแต่อ่านได้อย่างหลายครั้ง (Write-Once Read-Many หรือ WORM) และแทคชนิดอ่านได้เพียงอย่างเดียว (Read-Only) หรือเรายังสามารถแบ่งชนิดของแทคตามความถี่ของการใช้งาน เช่น แทคย่านความถี่ต่ำ (LF) 125-134 กิโลเฮิรตซ์ แทคย่านความถี่สูง (HF) 13.56 เมกะเฮิรตซ์ แทคย่านความถี่สูงยิ่ง (UHF) 433 และ 900 เมกะเฮิรตซ์ และแทคย่านไมโครเวฟ 2.4 กิกะเฮิรตซ์



รูปที่ 2.3 แสดง RFID แทคในรูปแบบต่างๆ

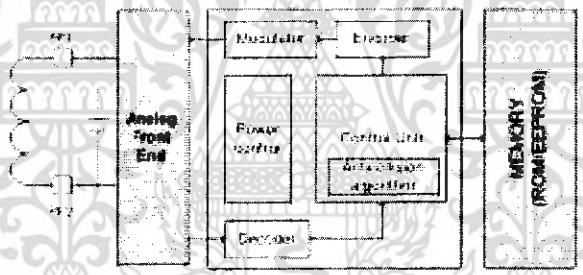
2.1.2.1 ชนิดของแทค

แทคที่มีการใช้งานอยู่นั้นจะมีอยู่ 3 ชนิดใหญ่ๆโดยแต่ละชนิดก็จะมีความแตกต่างกันในแง่ของการใช้งาน ราคา โครงสร้างและหลักการทำงานอยู่ ซึ่งจะขอกล่าวและอธิบายแยกเป็นหัวข้อดังนี้

- แทคของ RFID แบบพาสซีฟ (Passive RFID Tags)

แทคชนิดนี้ทำงานได้ไม่ต้องอาศัยแหล่งจ่ายไฟภายนอกใดๆเพราะภายในแทคจะมีวงจรกำเนิดจะมีวงจรกำเนิดไฟฟ้าเหนี่ยวนำขนาดเล็กเป็นแหล่งจ่ายไฟในตัวอยู่ ทำให้การอ่านข้อมูลทำได้ไม่ไกลมากนักระยะอ่านสูงสุดประมาณ 1 เมตรขึ้นอยู่กับกำลังของเครื่องส่งและคลื่นความถี่วิทยุที่ใช้ โดยปกติแทคชนิดนี้มักมีหน่วยความจำน้อย โดยทั่วไปประมาณ 16-1,024 ไบต์ มีขนาดเล็กและน้ำหนักเบา ราคาต่อหน่วยต่ำ

ไมโครชิปหรือไอซีของแทคชนิดแพสซีฟที่มีการผลิตออกมาจะมีทั้งขนาดและรูปร่างเป็นได้ตั้งแต่แบบแท่ง หรือแผ่นขนาดเล็กจนแทบไม่สามารถมองเห็นได้ไปจนถึงขนาดใหญ่สะดุดตา ซึ่งต่างก็มีความเหมาะสมกับชนิดการใช้งานที่แตกต่างกัน โดยทั่วไปโครงสร้างภายในส่วนที่เป็นไอซีของแทคนั้นก็จะประกอบด้วย 3 ส่วนหลักๆ ได้แก่ ส่วยของควบคุมการทำงานของภาครับส่งสัญญาณวิทยุ (Analog Front-End) ส่วนควบคุมภาคลอจิก (Digital Control Unit) ส่วนของหน่วยความจำ (Memory) ซึ่งอาจจะเป็นแบบ ROM หรือ EEPROM



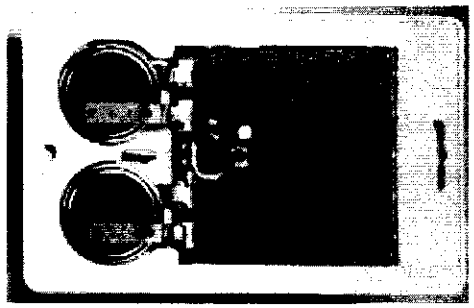
รูปที่ 2.4 แสดงบล็อกไดอะแกรมของแทคแบบพาสซีฟ

- แทคของ RFID แบบกึ่งพาสซีฟ

แทคชนิดนี้จะต้องอาศัยแหล่งจ่ายไฟจากแบตเตอรี่ภายนอก ทำให้สามารถส่งข้อมูลได้ระยะไกลกว่าแทคแบบแพสซีฟเพื่อประหยัดไฟ ตัวแทคจะรอรับสัญญาณกระตุ้นให้ทำงานจากเครื่องอ่านแล้วจึงจะส่งข้อมูลกลับไปยังเครื่องอ่าน

- แทคของ RFID แบบแอกทีฟ (Active RFID Tags)

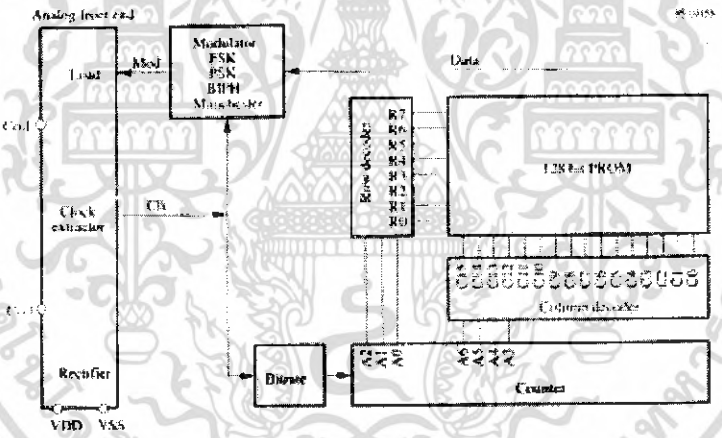
แทคชนิดนี้จะต้องอาศัยแหล่งจ่ายไฟจากแบตเตอรี่ภายนอก เพื่อจ่ายพลังงานให้กับวงจรภายในทำงาน โดยแทคแบบนี้สามารถมีหน่วยความจำภายในขนาดใหญ่ได้ถึง 1 เมกะไบต์และสามารถอ่านได้ในระยะไกลสูงสุดประมาณ 100 เมตร ข้อเสียของแทคแบบนี้คือ มีราคาต่อหน่วยสูง มีขนาดค่อนข้างใหญ่และมีอายุการใช้งานที่จำกัดตามอายุของแบตเตอรี่ซึ่งจะมีอายุการใช้งานประมาณ 3-7 ปี และสามารถส่งสัญญาณออกมาเองได้ตามเงื่อนไขที่กำหนด เช่น การตั้งเวลา หรือเมื่อกรณีที่มีเหตุอันควรตามที่โปรแกรมเอาไว้ หรือเมื่อได้รับสัญญาณสอบถามจากเครื่องอ่าน



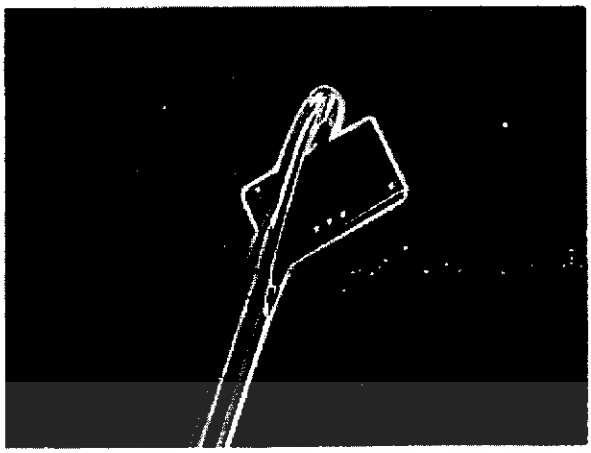
รูปที่ 2.5 แสดงรูปตัวอย่างแทคแบบแอคทีฟที่มีแบตเตอรี่ลิเทียม 2 ก้อนอยู่ภายนอก

2.1.2.2.1 แทคแบบอ่านอย่างเดียว

แทคชนิดนี้จะเข้าสู่บริเวณการตรวจสอบของเครื่องอ่านสัญญาณแล้วมันเริ่มส่งตัวเลขไอดีของมันอย่างต่อเนื่อง โดยปกติแล้วผู้ผลิตชิปจะรับรองว่าแต่ละซีเรียลนัมเบอร์นั้นจะใช้แค่ครั้งเดียว ตัวเลขเฉพาะของแทคจะถูกบรรจุลงไปในแทคในระหว่างการผลิตชิป ผู้ใช้ไม่สามารถเปลี่ยนแปลงซีเรียลนัมเบอร์หรือข้อมูลบนชิปได้



รูปที่ 2.6 แสดงบล็อกไดอะแกรมของแทคแบบอ่านอย่างเดียว เมื่อแทคเข้าสู่บริเวณการตรวจสอบของเครื่องอ่านสัญญาณ การนับจะเริ่มนับทุกแอดเดรสที่อยู่ในหน่วยความจำอย่างต่อเนื่อง เอาท์พุทของข้อมูลของหน่วยความจำจะเชื่อมต่อกับโหลดมอดูเลเตอร์ที่เซตเป็นรหัสเบสแบนด์ของรหัสไบนารี่ ในกระบวนการนี้ข้อมูลทั้งหมดในหน่วยความ (128 บิทซีเรียลนัมเบอร์) สามารถส่งไปได้อย่างต่อเนื่อง



รูปที่ 2.7 แสดงการเปรียบเทียบขนาดชิปแทคกับรูของเข็ม

การติดต่อสื่อสารกับเครื่องอ่านสัญญาณจะเป็นแบบทิศทางเดียวโดยแทคจะส่งตัวเลขไอดีของมันไปยังเครื่องอ่านสัญญาณอย่างต่อเนื่อง เครื่องอ่านสัญญาณและแทคแบบอ่านอย่างเดียวมีขนาดถูก ซึ่งแทคแบบอ่านอย่างเดียวที่มีราคาถูกจะไม่สามารถเก็บข้อมูลลงไปในแทคได้

2.1.2.2.2 แทคแบบเขียนได้

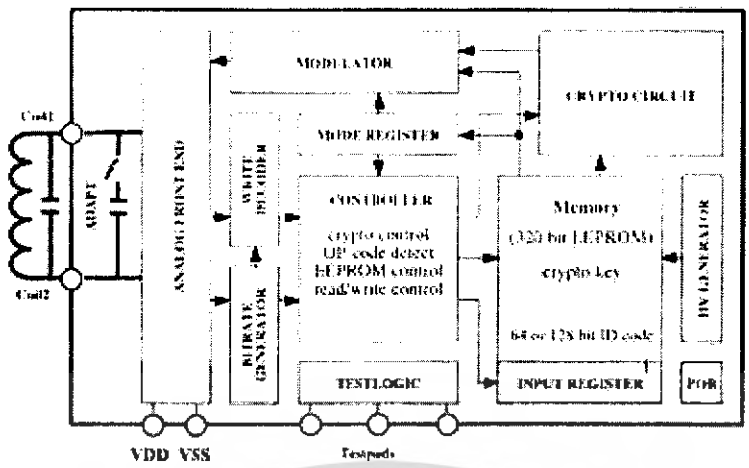
แทคที่สามารถเขียนข้อมูลโดยเครื่องอ่านสัญญาณจะมีขนาดหน่วยความจำ 1 ไบต์ถึง 64 ไบต์ การอ่านและเขียนข้อมูลในแทคจะเป็นแบบบล็อกข้อมูล ในบล็อกจะมีรูปแบบเป็นจำนวนไบต์ที่สามารถอ่านและเขียนได้ การที่จะเปลี่ยนแปลงข้อมูลของแต่ละบล็อกบล็อกทั้งหมดจะถูกอ่านจากแทค หลังจากนั้นในบล็อกเดียวกันรวมกับบล็อกที่แก้ไขแล้วจะถูกเขียนกลับลงไปในแทค

2.1.2.2.3 แทคที่มีฟังก์ชัน cryptological

ถ้าแทคที่สามารถเขียนได้ไม่ได้ป้องกันเอาไว้ เครื่องอ่านสัญญาณที่เป็นระบบ RFID สามารถอ่านและเขียนข้อมูลได้ซึ่งจะไม่ต้องการเพราะอาจจะเกิดความเสียหายได้โดยการอ่านหรือเขียนข้อมูลที่ไม่ได้รับอนุญาตในแทค ตัวอย่างที่เป็นแทคชนิดนี้คือแทคในตั๋วเดินทางและแทคในกุญแจรถ

มีวิธีการแก้ปัญหาในกรณีที่มีการเข้าถึงแทคโดยไม่ได้รับอนุญาต วิธีที่ง่ายที่สุดในการป้องกันการอ่านและเขียนข้อมูลคือการเช็คพาสเวิร์ด วิธีการก็คือการ์ดจะเปรียบเทียบพาสเวิร์ดที่ส่งกับพาสเวิร์ดอ้างอิงและจะสามารถเข้าถึงในหน่วยความจำได้ถ้าพาสเวิร์ดตรงกัน

พื้นที่ในหน่วยความจำที่บรรจุในแอปพลิเคชั่นแทคที่มีฟังก์ชั่นการทำงานแบบ cryptological จะมีพื้นที่ในหน่วยความจำสำหรับเก็บ secret key และ configuration register (access register,Acc) secret key จะถูกเขียนไว้ที่ key memory โดยผู้ผลิตก่อนที่แทคจะส่งไปให้ผู้ใช้ ซึ่ง key memory จะไม่สามารถอ่านได้



รูปที่ 2.8 แสดงบล็อกไดอะแกรมของแทคที่เขียนได้ที่มีฟังก์ชั่น cryptological ระหว่างแทคกับเครื่องอ่านสัญญาณ

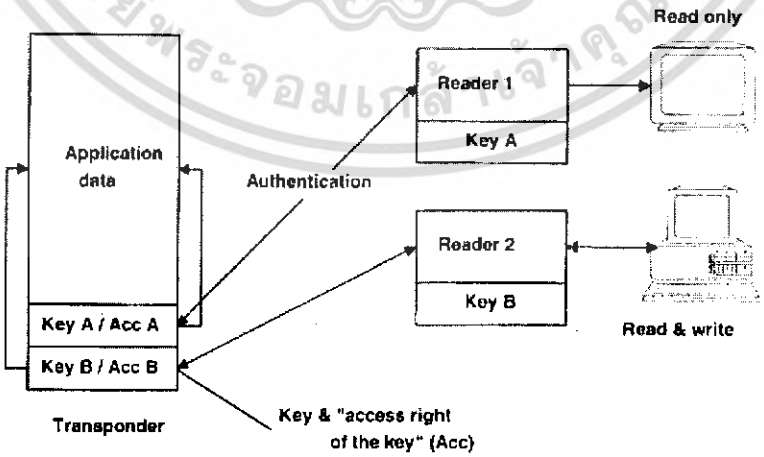
2.1.3 แนวความคิดของ Hierarchy key

ในบางระบบจะมีการแยกการเก็บข้อมูลออกเป็น key A และ key B ซึ่งจะมีวิธีการเข้าถึงที่แตกต่างกัน การตรวจสอบระหว่างแทคกับ reader จะต้องใช้ key A และ key B

Reader 1 จะครอบครอง key A หลังจากการตรวจสอบเสร็จสิ้นจะสามารถอ่านข้อมูลในหน่วยความจำของแทคได้ Reader B จะครอบครอง key B ซึ่งหลังจากตรวจสอบเสร็จสิ้นแล้วมันจะสามารถเขียนลงไปในหน่วยความจำของแทคได้

2.1.3.1 ตัวอย่างการประยุกต์ใช้ Hierarchy key

การประยุกต์ใช้จะใช้ในระบบการเดินทางสาธารณะซึ่งจะแบ่งเครื่องอ่านสัญญาณออกเป็น 2 แบบคือ 'Devaluer' สำหรับการจ่ายค่าเดินทางและ 'Revaluer' เป็นสมาร์ทการ์ดที่สามารถใส่ข้อมูลลงไปได้



รูปที่ 2.9 แสดงแทคที่มี 2 หน่วยความจำ

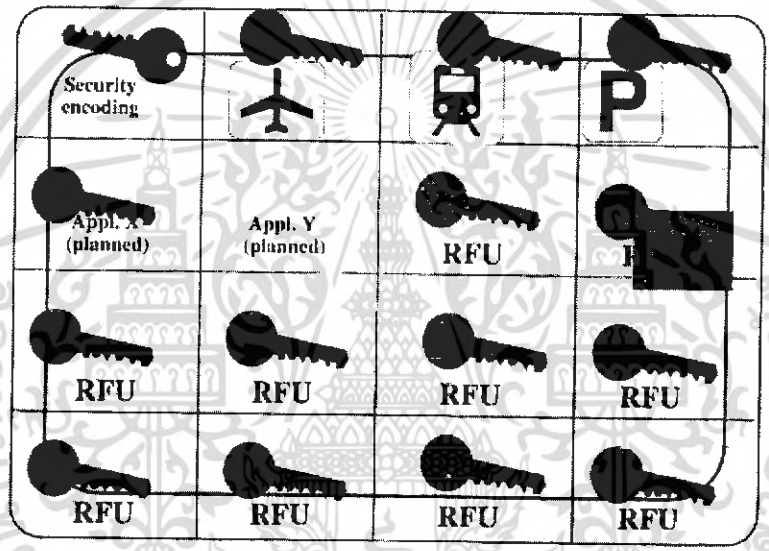
การเข้าถึงไปยังแทคจะมีอยู่ 2 ทางคือรีจิสเตอร์ A และรีจิสเตอร์ B หลังจากตรวจสอบโดยใช้ key A ระบบจะอนุญาตให้ลดจำนวนเงินลง ถ้าตรวจสอบโดยใช้ key B จะสามารถเพิ่มจำนวนเงินได้

ในการป้องกันการโกง แทคจะไม่สามารถเปลี่ยนแปลงข้อมูลได้โดย Devaluer ถ้าซอร์ฟแวร์ถูกขโมย แทคจะไม่สามารถเปลี่ยนแปลงได้ถ้าไม่มี key ที่ถูกต้อง

การรักษาความปลอดภัย key B จะถูกโหลดเข้าไปใน เครื่องอ่านไว้สำหรับการป้องกันขโมย

ก. การแบ่งหน่วยความจำ

แทคสามารถป้องกันจากการเข้าถึงโดยเครื่องอ่านสัญญาณจากแอปพลิเคชั่นอื่นได้โดยวิธีการตรวจสอบดังที่อธิบายไว้ข้างต้น ในแทคที่มีหน่วยความจำขนาดใหญ่จะสามารถแบ่งแยกหน่วยความจำออกเป็นยูนิตเล็กๆได้เรียกว่า เซ็กเมนต์ และป้องกันการเข้าถึงที่ไม่ได้รับอนุญาตได้ key ที่แตกต่างกัน



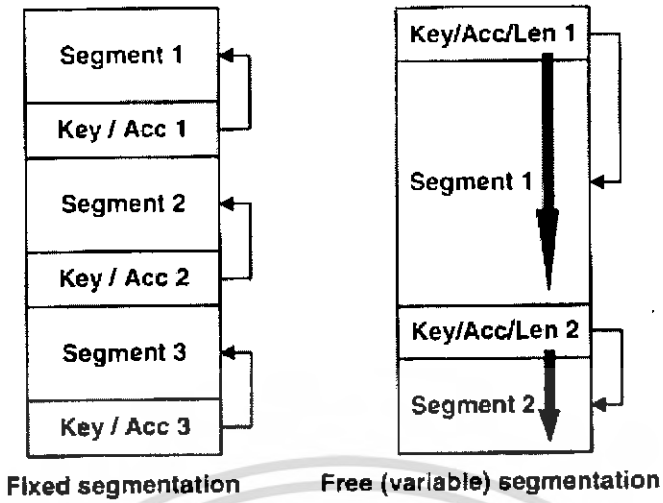
รูปที่ 2.10 แสดงการประยุกต์ใช้หลายประเภทใน 1 แทคซึ่งแต่ละอันมีการป้องกันโดย key ของมันเอง

การเข้าถึงในเซ็กเมนต์แต่ละตัวสามารถตรวจสอบได้ด้วย key ที่เหมาะสม ดังนั้น reader ของแอปพลิเคชั่นสามารถเข้าถึงเฉพาะเซ็กเมนต์ของมันเองได้ถ้ารู้ key ของแอปพลิเคชั่นของตัวมันเอง

หน่วยความจำแต่ละเซ็กเมนต์ส่วนใหญ่จะมีขนาดที่แน่นอนในระบบเหล่านี้พื้นที่การเก็บในเซ็กเมนต์จะไม่สามารถเปลี่ยนแปลงได้จากผู้ใช้ ขนาดเซ็กเมนต์ที่แน่นอนจะมีประโยชน์คือง่ายและถูก ซึ่งจะพิจารณาจากไมโครชิปของแทค

ในแอปพลิเคชั่นเล็กๆค่าของพื้นที่การเก็บจะเสียไปบนแทคเพราะแต่ละเซ็กเมนต์จะเป็นส่วนที่ใช้เท่านั้น ในทางกลับกันแอปพลิเคชั่นขนาดใหญ่มันจะกระจายไปหลายเซ็กเมนต์ ซึ่งหมายความว่าแอปพลิเคชั่นนั้น specific key จะต้องถูกเก็บไว้ในแต่ละเซ็กเมนต์ที่ครอบครองอยู่ ในการเก็บไอดี key หลายๆ อันจะทำให้เสียพื้นที่

ในการที่จะใช้พื้นที่ให้มีประสิทธิภาพกับความยาวของเซ็กเมนต์ หน่วยความจำที่อยู่ในเซ็กเมนต์สามารถตรงกันกับความต้องการของแอปปลิเคชั่นโดยใช้พื้นที่ในหน่วยความจำ



รูปที่ 2.11 แสดงความแตกต่างระหว่างเซ็กเมนต์ที่แน่นอนกับฟรีเซ็กเมนต์

รูปที่ 2.11 แสดงถึงรูปแบบของหน่วยความจำของแทคที่มีเซกเมนต์ที่แน่นอน หน่วยความจำทั้งหมดมี 128 ไบต์จะแบ่งออกเป็น 4 ส่วน แต่ละส่วนสามารถป้องกันการอ่านและเขียนข้อมูลที่ไม่ได้รับอนุญาตได้โดยใช้พาสเวิร์ดของตัวมัน การเข้าถึงรีจิสเตอร์ของแทคนี้จะประกอบด้วยพื้นที่หน่วยความจำ 16 บิตต่อหนึ่งเซ็กเมนต์

2.1.4 เทคโนโลยีของหน่วยความจำ

- RAM (Random Access Memory)

เป็นหน่วยความจำที่ใช้เก็บข้อมูลชั่วคราว ข้อมูลจะสูญหายเมื่อไม่มีไฟเลี้ยง ดังนั้นหน่วยความจำชนิดนี้จึงจำเป็นต้องมีแบตเตอรี่สำรองอยู่ภายในแทคด้วย ข้อดีของหน่วยความจำชนิดนี้คือ สามารถเก็บข้อมูลได้มาก เวลาในการอ่านหรือเขียนข้อมูลนั้นสั้น

- EEPROM (Electric Erasable PROM)

เป็นหน่วยความจำที่ไม่ต้องใช้แบตเตอรี่ในการสำรองไฟเพื่อรักษาข้อมูล เพราะภายในวงจรจะมีคาปาซิเตอร์ชนิดพิเศษทำหน้าที่ชาร์ตประจุไฟฟ้าให้กับวงจรของหน่วยความจำซึ่งมีอายุยาวนานถึง 10 ปี จำนวนครั้งในการเขียนข้อมูลมากกว่า 100,000 ครั้ง แต่เวลาในการอ่านหรือเขียนข้อมูล นานกว่า RAM และความจุก็น้อยกว่า RAM ด้วย

- FRAM (Ferroeletric RAM)

EEPROM มีข้อเด่นคือเป็นหน่วยความจำที่ไม่ต้องมีแบตเตอรี่ แต่ข้อเสียคือใช้เวลาในการเขียนข้อมูลมากและใช้พลังงานในการเขียนข้อมูล FRAM จึงถูกพัฒนาขึ้นมาเพื่อแก้ปัญหาข้อบกพร่องของ EEPROM คือสามารถเขียนข้อมูลได้เร็วกว่า ใช้พลังงานน้อยกว่า การเขียนข้อมูลซ้ำได้มากกว่า 1,000,000,000 ครั้ง และไม่ต้องมีแบตเตอรี่เหมือนกัน แต่เทคโนโลยีของ FRAM นั้นจะมีความซับซ้อนมากกว่า และเนื่องจากการพัฒนา FRAM ยังมีปัญหาอยู่บางประการทำให้หน่วยความจำชนิดนี้ยังไม่ถูกนำมาใช้งานแพร่หลาย

2.1.5 ความถี่ของคลื่นพาหะ

ในปัจจุบันได้มีการรวมกลุ่มระหว่างแต่ละประเทศ เพื่อทำการกำหนดมาตรฐานความถี่คลื่นพาหะของระบบ RFID โดยมี 3 กลุ่มใหญ่ๆ คือ กลุ่มประเทศในยุโรปและแอฟริกา (Region 1), กลุ่มประเทศอเมริกาเหนือและอเมริกาใต้ (Region 2) และสุดท้ายคือกลุ่มประเทศตะวันออกไกลและออสเตรเลีย (Region 3) ซึ่งแต่ละกลุ่มประเทศจะกำหนดแนวทางในการเลือกใช้ความถี่ต่างๆให้แก่บรรดาประเทศสมาชิก อย่างไรก็ตาม ความถี่ของคลื่นพาหะที่นิยมใช้งานในย่านความถี่ต่ำ (125 kHz) ย่านความถี่ปานกลาง (13.56 MHz) และย่านความถี่สูง (2.45 GHz) ดังที่แสดงไว้ในตารางที่ 2.1 นอกจากนี้รัฐบาลของแต่ละประเทศ โดยทั่วไปจะมีการออกกฎหมายเกี่ยวกับระเบียบการใช้งานย่านความถี่ต่างๆ รวมถึงกำลังส่งของระบบ RFID ด้วย

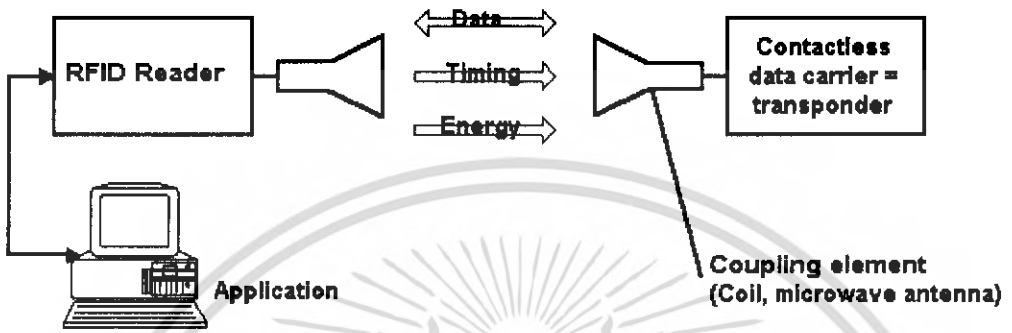
ตารางที่ 2.1 แสดงย่านความถี่ต่างๆ ของระบบ RFID และการใช้งาน

ย่านความถี่	ความถี่มาตรฐาน	คุณสมบัติ	ลักษณะการใช้งาน
ความถี่ต่ำ (100-500 kHz)	125 kHz	- ระยะการรับส่งข้อมูลใกล้ - ต้นทุนไม่สูง - ความเร็วในการอ่านข้อมูลต่ำ	- Access Control - ปศุสัตว์ - ระบบคงคลัง - รถยนต์
ความถี่ปานกลาง (10-15 MHz)	13.56 MHz	- ระยะการรับส่งข้อมูลปานกลาง - ราคามีแนวโน้มถูกลงในอนาคต - ความเร็วในการอ่านข้อมูลปานกลาง	- Access Control - สมาร์ตการ์ด
ความถี่สูง (850-850MHz, 2.4-5.8 GHz)	2.45 GHz	- ระยะการรับส่งข้อมูลไกล (10 เมตร) - ราคาแพง - ความเร็วในการอ่านข้อมูลสูง	- รถไฟ - ระบบเก็บค่าผ่านทาง

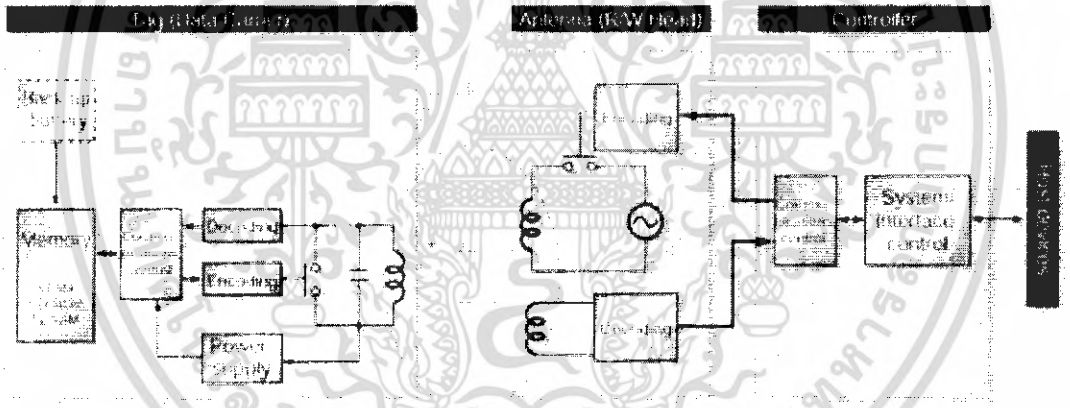
2.1.6 การทำงานของระบบ RFID

จากวงจรเทียบเคียงส่วนควบคุมและติดต่อสื่อสาร (Control and Interface) จะได้รับคำสั่ง (Command) จากส่วนควบคุมที่สูงกว่า (Host) เช่น คอมพิวเตอร์ หรือ PLC (Programmable Logic Controller) จากนั้นตัวควบคุมจะทำการประมวลผลคำสั่งว่า Host ต้องการให้ทำอะไร จากนั้นก็จะสั่งให้ส่วนของภาครับ /ส่งวิทยุ ที่มีส่วนของวงจรเข้ารหัส (Coding) จากนั้นส่วนของวงจรผสมสัญญาณ (Modulation) ทำการผสมข้อมูลเข้ากับคลื่นพาหะแล้วทำการส่งออกไปทางเสาอากาศ ขนาดของพื้นที่ที่มีสัญญาณอยู่นั้นจะขึ้นอยู่กับขนาดของเสาอากาศและพลังงาน (Watt) ของเสาอากาศ เมื่อแทคเข้ามาในพื้นที่ที่มีสัญญาณแล้ว เสาอากาศภายในแทคจะได้รับการคล้องสัญญาณทำให้แทคทำงานได้ตามที่กล่าวมาแล้ว วงจรถอดรหัส (Demodulation) จะทำการแยกสัญญาณข้อมูล ที่ถูกผสมมาจากเครื่องอ่าน ออกจากคลื่น

พาหะแล้วทำการแปลงรหัส (Decoding) จากนั้น CPU ของแทคจะรับคำสั่งไปประมวลผลถ้าเป็นคำสั่งเขียนแทคจะบันทึกข้อมูลที่ส่งมาลงในหน่วยความจำของแทค แต่ถ้าเป็นคำสั่งอ่านแทค จะดึงข้อมูลจากหน่วยความจำที่ระบุไว้จากจากคำสั่ง แล้วทำการผสมข้อมูลที่วงจรผสมข้อมูลภายในแทค กับคลื่นพาหะแล้วส่งออกไปทางเสาอากาศเหมือนกัน เมื่อเครื่องอ่านได้รับสัญญาณจากแทควงจรถอดรหัสของเครื่องอ่านก็จะถอดเอาข้อมูลออกจากคลื่นพาห์และส่งไปที่ Host Unit



รูปที่ 2.12 แสดงการสื่อสารระหว่างเครื่องอ่านสัญญาณกับแทค



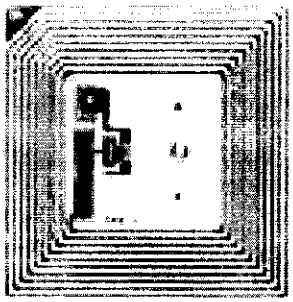
รูปที่ 2.13 แสดงวงจรเทียบเคียงของระบบ RFID

2.1.7 ประเภทของ ระบบ RFID

2.1.7.1 RFID ที่จำแนกโดยขนาดของหน่วยความจำ

2.1.7.1.1 RFID ชนิด 1 บิต (2 Bit Type)

RFID ชนิดนี้หรือเรียกอีกอย่างหนึ่งว่า EAS (Electronic Article Surveillmce) เป็น RFID ที่ใช้แทคที่ไม่มีไมโครชิป RFID ระบบนี้จะตรวจสอบเฉพาะว่ามีแทคอยู่ในพื้นที่สัญญาณหรือไม่ ดังนั้นสถานะจึงแสดงเพียงแค่มีหรือไม่มี ซึ่งเป็นรหัสดิจิตอล 0 หรือ 1 นั่นเอง



รูปที่ 2.14 แสดงแทคของระบบ EAS จะมีเฉพาะเสาอากาศอย่างเดียว

เนื่องจากการทำงานของ RFID ระบบนี้ง่ายไม่ซับซ้อนและแทคมีราคาถูกมาก ระบบจึงถูกนำมาใช้กับการป้องกันสินค้าถูกขโมยในห้างร้านต่างๆ โดยแทคจะติดอยู่กับตัวสินค้าหรือซ่อนไว้ข้างหลังบาร์โค้ดอีกที เครื่องอ่านจะออกแบบเป็นโครงเสาอากาศสูงประมาณ 1-12 เมตร อยู่ที่ทางออกของห้างร้านนั้นๆ เมื่อสินค้าที่มี แทคติดอยู่ผ่านมาในบริเวณเสาอากาศนี้ก็จะถูกตรวจจับได้จากการคล้องสัญญาณจากเสาอากาศและแทค จึงสามารถตรวจจับสินค้าที่ถูกนำออกไปได้ แทคจะถูกดึงออกหรือใช้เครื่องทำลายความเป็นสนามแม่เหล็กไฟฟ้าที่เคาน์เตอร์แคชเชียร์ เมื่อสินค้าถูกนำมาชำระ ซึ่งเราจะพบเห็นระบบ EAS ในซุปเปอร์มาร์เก็ต หรือ ดิสเคาน์สโตร์ต่างๆ เช่น คาร์ฟูร์ โลตัส หรือร้านขายเสื้อผ้า ซีดี ต่างๆ ช่วงความถี่ที่ใช้งานจะเป็นช่วงความถี่ไมโครเวฟ เนื่องจากระยะในการสื่อสารค่อนข้างไกล (ขนาดระหว่างเสาอากาศ)

รูปที่ 2.15 แสดงเสาอากาศของระบบ EAS

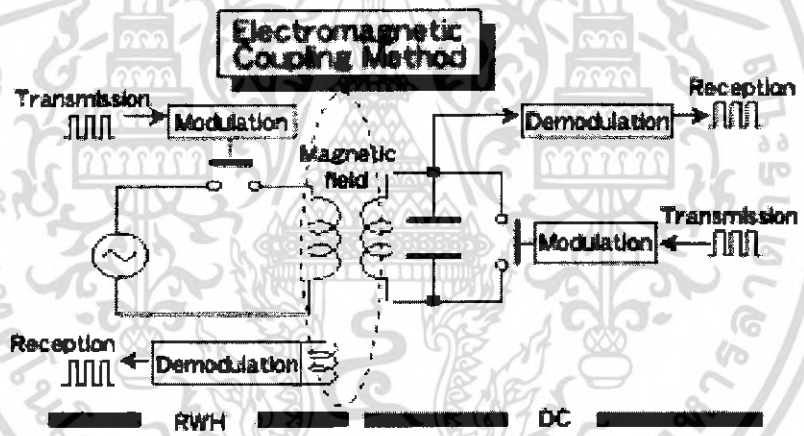
2.1.7.1.2 RFID ชนิดหน่วยความจำมากกว่า 1 บิต (Data Carrier Type)

RFID ชนิดนี้จะใช้แทคที่มีไมโครชิปและหน่วยความจำที่เป็นส่วนประกอบสำคัญ มีราคาสูงกว่าแทคชนิด EAS โดยบางชนิดสามารถเก็บข้อมูลได้สูงสุดถึง 64 Kbytes ใช้ในอุตสาหกรรมหรืองานทั่วไปที่ต้องใช้ แทคในการเก็บข้อมูล

2.1.7.2 RFID ที่จำแนกโดยลักษณะการคล้องของสัญญาณ

2.1.7.2.1 Close Coupling

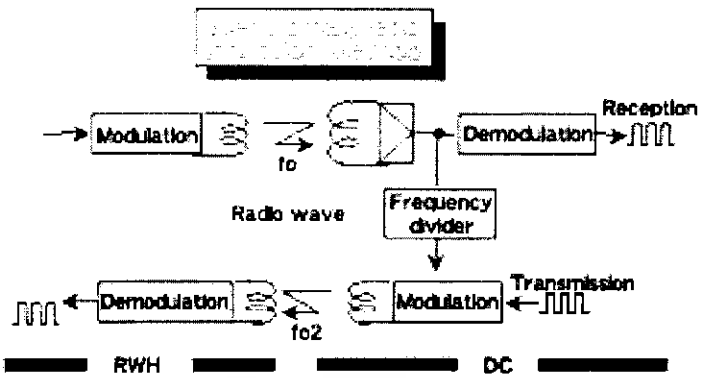
เป็น RFID ที่มีระยะในการอ่าน /เขียนข้อมูลสั้นมากประมาณ 0-1 เซนติเมตร ดังนั้น แทคจะต้องอยู่ใกล้หรือวางอยู่บนเครื่องอ่าน Close Coupling นี้จะสามารถใช้คลื่นความถี่ได้ตั้งแต่ 0 Hz จนถึง 50 MHz เนื่องจากการทำงานของแทคไม่อาศัยการส่งพลังงานจากการแผ่คลื่นแม่เหล็กไฟฟ้าจากเครื่องอ่านแต่อาศัยการเหนี่ยวนำเหมือนหลักการของหม้อแปลงไฟฟ้า ทำให้เกิดพลังงานที่ทำให้วงจรภายในแทคทำงานได้ ระบบ Close-Coupling จะนิยมนำมาใช้งานที่ต้องการความปลอดภัยค่อนข้างสูง แต่ไม่ต้องการติดต่อได้ไกลเช่นประตูอัตโนมัติ หรือ สมาร์ทการ์ดไร้สัมผัส (Contactless Smart Cards)



รูปที่ 2.16 แสดงวงจรเทียบเคียงของ Close – Coupling

2.1.7.2.2 Remote Coupling

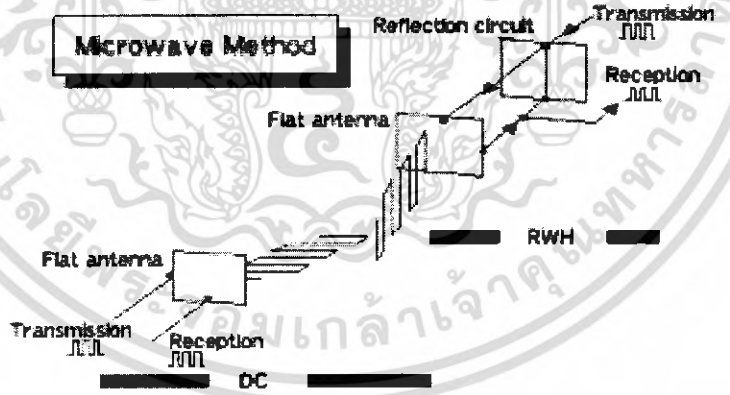
เป็นระบบที่มีระยะการอ่านหรือเขียนสูงถึง 1 เมตร ระบบนี้จะใช้หลักการคล้องสัญญาณแบบ Inductive (Magnetic) Coupling ระหว่างเครื่องอ่านกับแทคประมาณ 90 – 95% ของระบบ RFID ในปัจจุบัน ใช้หลักการ Remote Coupling นี้โดยความถี่ที่ใช้งานมีหลายความถี่ตั้งแต่ต่ำกว่า 135 KHz หรือ 13.56 MHz และ 27.125 MHz พลังงานไฟฟ้าจะถูกส่งโดยหลักการแผ่ของคลื่นแม่เหล็กไฟฟ้าไปให้แทค ทำให้แทคได้รับพลังงานสามารถทำงานได้ ระบบ Remote Coupling นี้จะพบมากในลักษณะงานอุตสาหกรรมเช่น รถยนต์ เครื่องใช้ไฟฟ้าหรืออิเล็กทรอนิกส์



รูปที่2.17 แสดงวงจรเทียบเคียงของ Remote Coupling หรือ Inductive Coupling

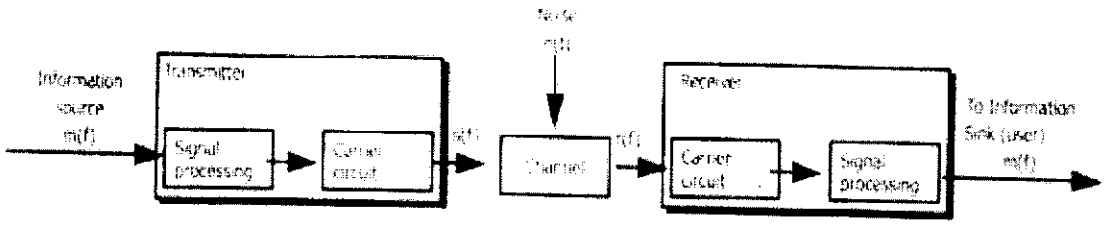
2.1.7.2.3 Long Range

ระบบนี้จะมีระยะการอ่าน / เขียนอยู่ระหว่าง 1 ถึง 10 เมตรหรือบางระบบอาจสูงกว่านี้ ความถี่ที่ใช้ในระบบนี้จะเป็นย่านที่มีความถี่สูงมากหรือ ไมโครเวฟ (Microwave range) ซึ่งปกติที่ความถี่ 2.45 GHz หรือบางครั้งจะพบที่ 915 MHz, 5.8 GHz และ 24.125 GHz แต่การส่งพลังงานจากตัวเครื่องอ่านไปยังแทคทำได้ยาก ดังนั้นแทคที่ใช้งานจะเป็นชนิดที่มีแบตเตอรี่ในตัวซึ่งจะใช้สำหรับเป็นไฟเลี้ยงที่ทำให้ไมโครชิปทำงานและเก็บรักษาข้อมูล ลักษณะงานที่พบเห็นจะเป็นลักษณะงานที่ต้องการการสื่อสารระยะไกล เช่น ในกระบวนการผลิตรถยนต์ ระบบชำระเงินอัตโนมัติของทางด่วน



รูปที่ 2.18 แสดงวงจรเทียบเคียง Long Range

2.1.8 เทคโนโลยีการเข้ารหัสและถอดรหัส



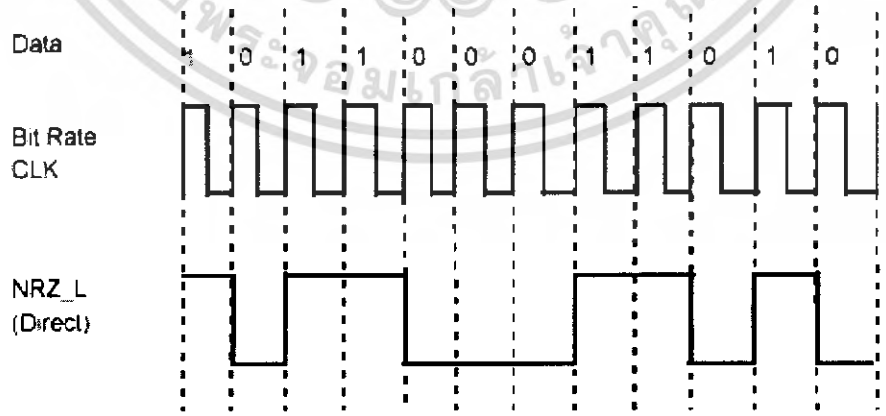
รูปที่ 2.19 แสดงการไหลของสัญญาณและข้อมูลในระบบการสื่อสารแบบดิจิตอล

จากรูปแสดงกระบวนการสื่อสารแบบดิจิตอล ซึ่งคือการส่งข้อมูลระหว่างเครื่องอ่านกับแทคในระบบ RFID นั่นเอง กระบวนการสื่อสารจะอาศัยหลักการดังนี้

2.1.8.1 การเข้ารหัสข้อมูล (Signal Coding)

การรับส่งข้อมูลแบบตรงไปตรงมาจะทำให้ข้อมูลอาจถูกรบกวนและทำให้การซิงโครไนซ์ของข้อมูลเกิดการคลาดเคลื่อน (โดยปกติวงจรดิจิตอลจะปรับการซิงโครไนซ์ของข้อมูลได้เฉพาะในช่วงที่มีการเปลี่ยนระดับของข้อมูลจาก 1 เป็น 0 หรือจาก 0 เป็น 1) ทำให้รับข้อมูลผิดพลาด เพื่อป้องกันปัญหาดังกล่าวจึงจะต้องมีการนำสัญญาณดิจิตอลปกติไปผ่านเข้ารหัสเสียก่อน ก่อนที่รหัสจะถูกส่งไปมอดูเลต แต่การเข้ารหัสบางอย่างก็มีข้อเสียคือช่วงความถี่ที่ใช้ในการส่งข้อมูลอาจต้องเพิ่มขึ้นเป็น 2 เท่า การเข้ารหัสมักเป็นที่นิยมใช้กันอย่างกว้างขวางในการสื่อสารแบบดิจิตอล ในระบบสื่อสารแบบดิจิตอลนั้นจะใช้สัญลักษณ์ 0 กับ 1 แทนข้อมูลโดยความแตกต่างของข้อมูลจะเป็นตัวกำหนดค่า 0 กับ 1 ของแต่ละบิตข้อมูล ข้อมูลจะถูกจัดเรียงเป็นแนวนอนหรือเส้นตรง (Line Code) ซึ่งมาตรฐานของ Line Code จะมีหลายมาตรฐาน และมาตรฐานที่นำมาใช้ในระบบ RFID มีดังนี้คือ

2.1.8.1.1 NRZ coding

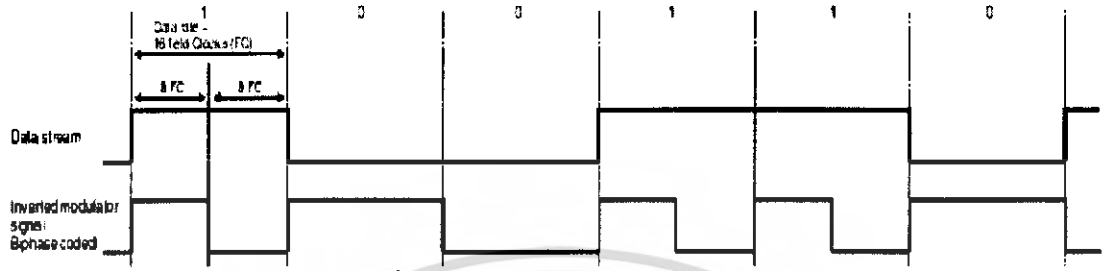


รูปที่ 2.20 แสดงรหัสสัญญาณ NRZ

สำนักหอสมุดกลาง พระจอมเกล้าลาดกระบัง

สัญลักษณ์ " 1 " จะแทนด้วยช่วงสัญญาณที่เป็น High ,สัญลักษณ์ " 0 " จะแทนด้วยช่วงสัญญาณที่เป็น Low

2.1.8.1.2 Biphase coding



รูปที่ 2.21 แสดงรหัสสัญญาณ Biphase

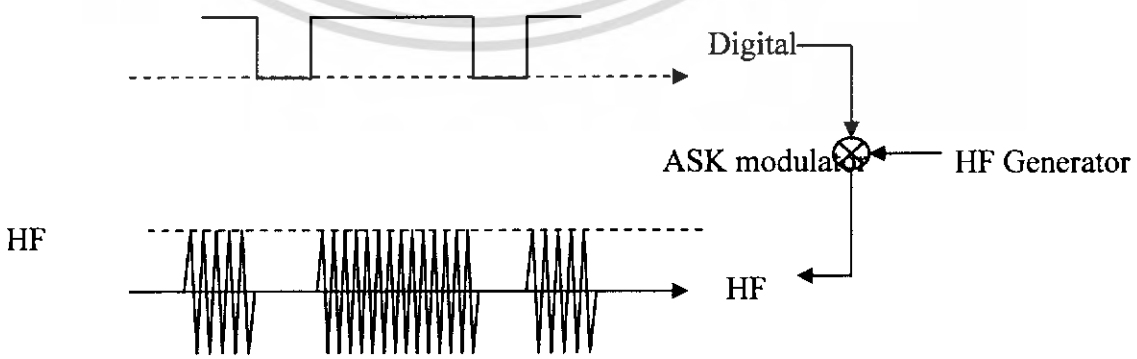
ข้อกำหนดของลักษณะการเข้าโค๊ดแต่ละแบบนี้อาจจะแตกต่างกันออกไปจากที่ได้กล่าวมาข้างต้น ตามชนิดของแทคของแต่ละบริษัทผู้ผลิตแทค

2.1.8.2 การผสมรหัสข้อมูลเข้ากับคลื่นพาหะ (Modulation)

เนื่องจากหลักการพื้นฐานของ RFID คือใช้คลื่นสนามแม่เหล็กไฟฟ้าในการสื่อสารรับส่งข้อมูล ดังนั้นก่อนการสื่อสารจะต้องมีกระบวนการผสมข้อมูลที่เข้ารหัสไว้แล้วไปกับคลื่นพาห์ (Modulation) แต่เนื่องจากลักษณะข้อมูลเป็นแบบดิจิตอลจึงจำเป็นต้องใช้วิธีการผสมแบบดิจิตอลซึ่งจะแตกต่างจากการผสมแบบอนาล็อกที่ใช้กันอยู่หลายระบบในปัจจุบันเช่น FM และ AM การผสมข้อมูลแบบดิจิตอลมี 3 วิธี ดังนี้

2.1.8.2.1 Amplitude Shift Keying (ASK)

เป็นการผสมข้อมูลโดยอาศัยการเปลี่ยนแปลงของยอดคลื่นพาหะ (Amplitude) เป็นตัวแสดงลักษณะข้อมูลโดยความถี่ของคลื่นพาหะไม่เปลี่ยนแปลงดังรูป

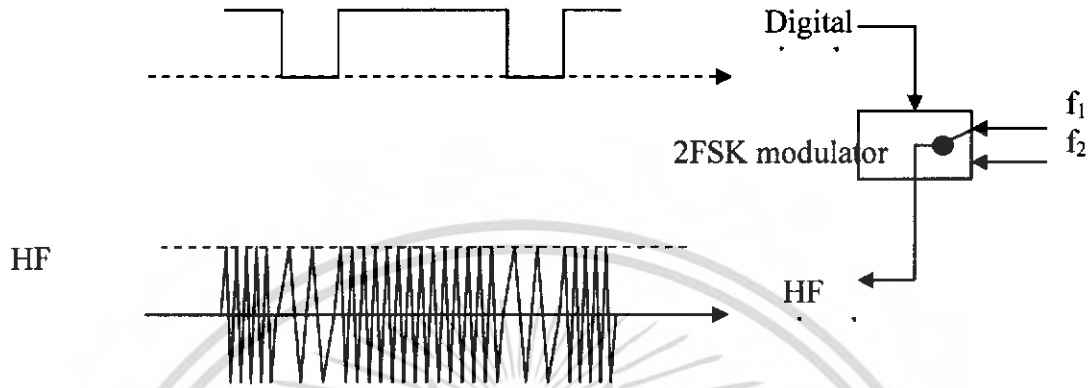


รูปที่ 2.22 แสดงความสูงของยอดคลื่นใน ASK จะเปลี่ยนสถานะอยู่ 2 สถานะขึ้นอยู่กับข้อมูลที่ผสมเข้ามา

62534

2.1.8.2.2 Frequency Shift Keying (FSK)

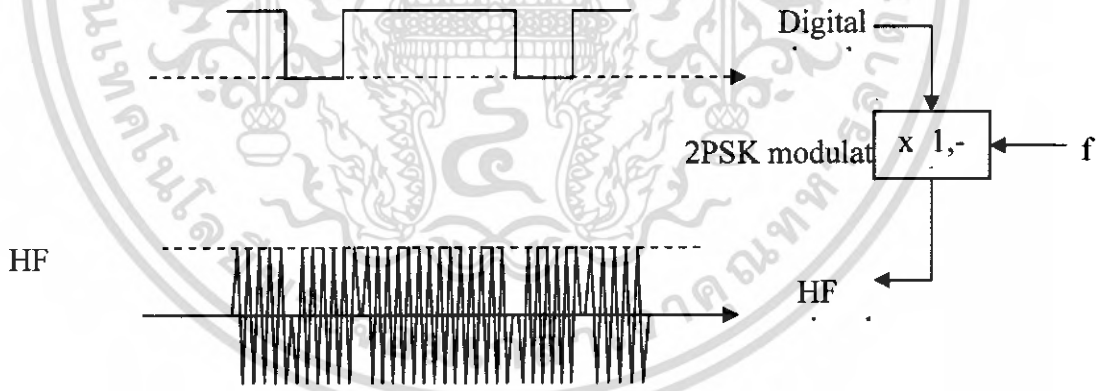
วิธีนี้จะอาศัยการเปลี่ยนแปลงความถี่ของคลื่นพาหะระหว่าง 2 ความถี่ขึ้นอยู่กับสถานะของข้อมูล 0 กับ 1 โดยความสูงของยอดคลื่น (Amplitude) ไม่เปลี่ยนแปลง



รูปที่ 2.23 แสดง 2FSK จะเปลี่ยนแปลงความถี่ของคลื่นตามข้อมูลที่ผสมเข้ามา

2.1.8.2.3 Phase Shift Keying (2PSK)

วิธีนี้จะใช้หลักการเปลี่ยนเฟสของลูกคลื่นเป็นตรงกันข้าม (0 องศา กับ 180 องศา) เมื่อมีการเปลี่ยนสถานะของข้อมูล ดังแสดงในรูป



รูปที่ 2.24 แสดง PSK จะกลับเฟสทุกครั้งเมื่อมีการเปลี่ยนสถานะของข้อมูล

แต่ในระบบ RFID จะใช้เพียงการมอดูเลตสัญญาณที่เข้ารหัสมาแล้วกับสัญญาณ คลื่นพาห์แบบ ASK เท่านั้น ในการรับส่งข้อมูลระหว่างเครื่องอ่านกับแทค

2.1.8.3 การส่งคลื่นสัญญาณออกไป (Transmission)

เครื่องอ่านจะเป็นตัวส่งคลื่นสนามแม่เหล็กไฟฟ้าออกไปด้วยความถี่ที่เราจะใช้งาน เมื่อแทคอยู่ในระยะที่เหมาะสมจะสามารถเหนี่ยวนำคลื่นสนามแม่เหล็กไฟฟ้านั้นมาแปลงเป็นพลังงานที่ใช้ในการทำงานภายในได้และแทคจะนำสัญญาณข้อมูล ดิจิตอลที่เข้ารหัสแล้วมาทำการ มอดูเลตกับคลื่นสนามแม่เหล็กไฟฟ้าที่รับได้แบบ ASK ส่งกลับไปยังเครื่องอ่านซึ่งเรียกสัญญาณนี้ว่า "สัญญาณ backscattering"

2.1.8.4 การถอดข้อมูลออกจากคลื่นพาหะ (Demodulation)

สัญญาณที่ทางเครื่องอ่านรับได้นั้นจะเป็นสัญญาณข้อมูลที่ถูกเข้ารหัสมาแล้วมอดูเลตกับคลื่นพาห์แบบ ASK เพราะฉะนั้นเราสามารถนำสัญญาณเดิมกลับมาได้โดยการดีมอดูเลตด้วยวิธีการตรวจจับกรอบสัญญาณ สัญญาณที่ได้จากการตรวจจับกรอบสัญญาณจะเป็นสัญญาณข้อมูลที่ถูกเข้ารหัสโดยแทค

2.1.9 หลักการและเทคนิคที่ใช้รับและส่งข้อมูลระหว่าง แทคและเครื่องอ่าน

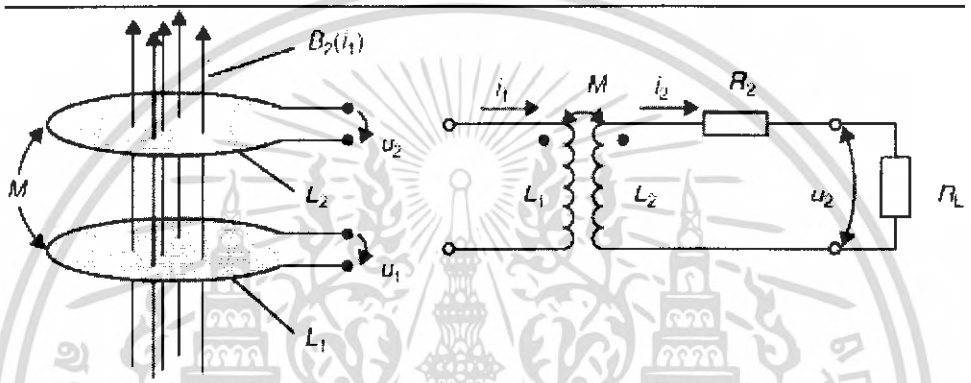
โดยมากเทคนิคในการรับส่งข้อมูลระหว่างเครื่องอ่านและแทค จะใช้หลักการมอดูเลตทางแอมปลิจูด (Amplitude Modulation: AM) หรือใช้การมอดูเลตทางแอมปลิจูดบวกกับการเข้ารหัสแมนเชสเตอร์ (Manchester encoded AM) แต่ทว่าในปัจจุบันก็มีแทคที่ใช้การมอดูเลตแบบอื่นๆ ด้วย เช่น การมอดูเลชั่นแบบเฟสชิปคีย์อิง (Phase Shift Keying: PSK), ฟรีเควนซี่ชิปคีย์อิง (Frequency Shift Keying: FSK) หรือการใช้การมอดูเลตทางความถี่ (Frequency Modulation: FM) ดังรูปที่

รูปที่ 2.25 แสดงรูปคลื่นของสัญญาณระหว่างแทคและเครื่องอ่านแบบ AM

ในการรับส่งข้อมูลหรือสัญญาณวิทยุระหว่างแทคกับเครื่องอ่านจะทำได้อย่างมีประสิทธิภาพต่อเมื่อสายอากาศมีความยาวที่เหมาะสมกับความถี่พาหะที่ใช้งาน เช่น เมื่อความถี่ใช้งานเป็น 13.56 เมกะเฮิรตซ์ ความยาวของเสาอากาศ (เป็นเส้นตรง) ที่เหมาะสมก็คือ 22.12 เมตร แน่นอนว่าในทางปฏิบัติเราคงไม่สามารถนำเสาอากาศที่ใหญ่ขนาดนั้นมาใช้งานกับแทคขนาดเล็กของเราได้ สายอากาศที่ดูเหมาะจะใช้ร่วมกับแทคมากที่สุดก็คือสายอากาศที่เป็นขดลวดขนาดเล็ก หรือที่มีชื่ออย่างเป็นทางการว่าสายอากาศแบบแมกเนติกไดโพล (magnetic dipole antenna) รูปแบบของสายอากาศแบบนี้ก็จะมีอยู่หลากหลาย ทั้งแบบที่เป็นขดลวดพันบนแกน อากาศหรือแกนเฟอร์ไรต์ แบบที่เป็นวงลูปที่ทำขึ้นจาก

ลายทองแดง บนแผ่นวงจรพิมพ์ ทั้งที่เป็นลูปแบบวงกลมและสี่เหลี่ยม ทั้งนี้ความเหมาะสมในการใช้งานก็แตกต่างกันไปตามความถี่พาหะ และประเภทของงานด้วยเช่นกัน

นอกจากการรับส่งข้อมูลแล้วสายอากาศก็ยังทำหน้าที่เป็นแหล่งจ่ายพลังงานให้กับแทคด้วย โดยอาศัยหลักการทำงานตามแนวคิดของไมเคิล ฟาราเดย์ เรื่องแรงดันเหนี่ยวนำในขดลวดที่เกิดขึ้นจากเส้นแรงแม่เหล็ก (จากเครื่องอ่าน) ที่มีค่าเปลี่ยนแปลงไปตามเวลา (Time-varying magnetic field) พุ่งผ่านสายอากาศของแทค เมื่อแทคและเครื่องอ่านตั้งอยู่ห่างกันในระยะ 0.16 เท่าของความยาวคลื่นพาหะที่ใช้ เรียกปรากฏการณ์ที่เกิดขึ้นนี้ว่า transformer-type coupling ซึ่งเป็นปรากฏการณ์แบบเดียวกับการเกิดแรงดันไฟฟ้าเหนี่ยวนำขึ้นระหว่างขดลวดปฐมภูมิ (primary) และขดลวดทุติยภูมิ (secondary) ในหม้อแปลงไฟฟ้า (transformer) จะเป็นวงจรพื้นฐานสำหรับอธิบายกลไกที่เกิดขึ้นในการส่งข้อมูลของแทค



รูปที่ 2.26 (ซ้าย) แสดงลูปตัวนำที่ถูกเหนี่ยวนำด้วยสนามแม่เหล็ก (ขวา) วงจรที่สมมูลกับรูปทางซ้ายมือ

2.1.10 การชนกันของข้อมูล

เมื่อมีแทคหลายๆ อันเข้ามาอยู่ใกล้เครื่องอ่าน เมื่อแทคมีพลังงานเพียงพอ แทคแต่ละอันจะพยายามส่งข้อมูลของตัวเองมาที่เครื่องอ่านพร้อมๆกัน ทำให้เครื่องอ่านไม่สามารถแยกแยะข้อมูลที่ส่งมาได้ ซึ่งเราเรียกปรากฏการณ์นี้ว่า การชนกันของข้อมูล (Collision) วิธีการแก้ไขโดยการทำการเพิ่มฟังก์ชั่นป้องกันการชนกันบนแทคและเครื่องอ่าน (Anti-collision) ซึ่งจะมีหลายเทคนิค เช่น จัดคิวการอ่านแทคโดยทำเป็นช่วงเวลาสั้นๆ เมื่อแทคโดนอ่านแล้วจะไม่มีการอ่านซ้ำอีก เช่น เทคนิค SDMA: Space Division Multiple Access TDMA, FDMA, CDMA หรือเทคนิคขั้นสูงจะใช้ FTDMA และการกระโดดความถี่ (frequency hopping) เข้าช่วย

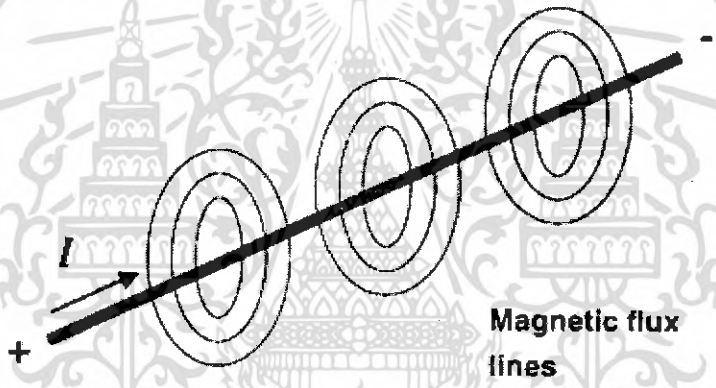
2.1.10.1 ขั้นตอนการทำงานระหว่างเครื่องอ่านกับแทค

- ตัวเครื่องอ่านจะทำการส่งสัญญาณวิทยุอย่างต่อเนื่องหรือเป็นจังหวะ และรอคอยสัญญาณตอบจากตัวแทค

- เมื่อแทคได้รับสัญญาณคลื่นวิทยุที่ส่งมาจากเครื่องอ่านในระดับที่เพียงพอ ก็จะทำเหนี่ยวนำเพื่อสร้างพลังงานป้อนให้แทคทำงาน โดยแทคจะสร้างสัญญาณนาฬิกาเพื่อกระตุ้นให้วงจรภาคดิจิตอลในแทคทำงาน
- วงจรภาคดิจิตอลจะไปอ่านข้อมูลจากหน่วยความจำภายในและเข้ารหัสข้อมูลแล้วส่งไปยังภาคอนาล็อกที่ทำหน้าที่มอดูเลตข้อมูล
- ข้อมูลที่ถูกมอดูเลตจะถูกส่งไปส่งขดลวดที่ทำหน้าที่เป็นสายอากาศ เพื่อส่งไปยังเครื่องอ่าน
- เครื่องอ่านจะสามารถตรวจจับสัญญาณการเปลี่ยนแปลงของแอมปลิจูด (Envelope Detector) และใช้พีก ดีเทกเตอร์ (Peak Detector) ในการแปลงสัญญาณข้อมูลที่มอดูเลตแล้วจากแทค
- เครื่องอ่านจะถอดรหัสข้อมูลและส่งไปยังคอมพิวเตอร์ผ่านทางพอร์ตอนุกรมต่อไป

2.2 หลักการของฟิสิกัลของระบบ RFID

2.2.1 สนามแม่เหล็ก

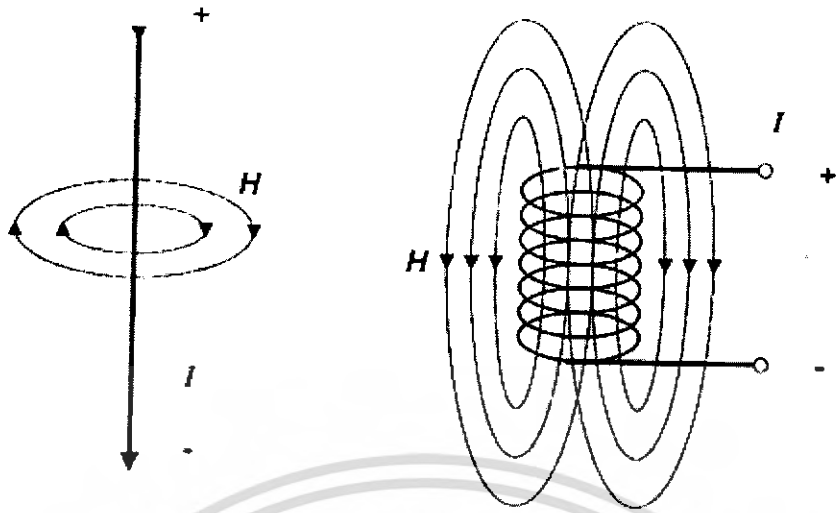


รูปที่ 2.27 แสดงเส้นของฟลักซ์แม่เหล็กถูกสร้างรอบๆ ตัวนำที่มีกระแส

ทุกๆ การเคลื่อนที่ของอิเล็กตรอนในสายหรือในสุญญากาศ เช่น การไหลของกระแสจะเกี่ยวกับสนามแม่เหล็ก ความเข้มของสนามแม่เหล็กสามารถแสดงด้วยการทดลองโดยดูจากแรงที่กระทำบนเข็มทิศ ขนาดของสนามแม่เหล็กอธิบายด้วย ความเข้มสนามแม่เหล็ก H

การอินทิเกรตความเข้มสนามแม่เหล็กในวงปิดเท่ากับผลรวมของกระแสในนั้น

$$\sum I = \oint \vec{H} \cdot d\vec{s} \qquad (2.1)$$



รูปที่ 2.28 แสดงเส้นของฟลักซ์แม่เหล็กรอบตัวนำกระแสและกระแสรอบขดลวดทรงกระบอก

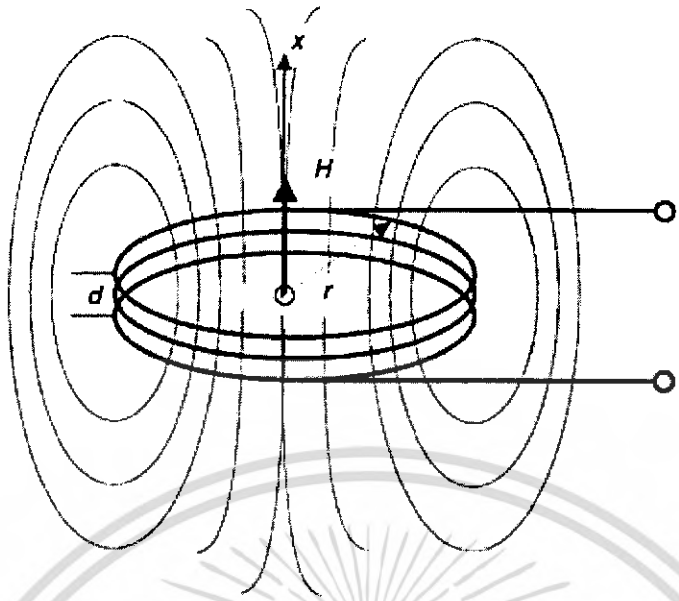
เราสามารถใช้สมการนี้คำนวณหาความเข้มสนามแม่เหล็ก H สำหรับตัวนำหลายชนิด ในตัวนำตรง ความเข้มสนาม H ระหว่างเส้นฟลักซ์วงกลม ที่ระยะ r เป็นค่าคงที่ เป็นดังนี้

$$H = \frac{1}{2\pi r} \qquad (2.2)$$

2.2.2 เส้นทางของกำลังสนาม $H(x)$ ในวงตัวนำ

ขดลวดทรงกระบอกสั้นหรือวงตัวนำใช้เป็นสายอากาศแม่เหล็ก เพื่อสร้างสนามแม่เหล็กสลับในอุปกรณ์เขียนหรืออ่านของการเหนี่ยวนำระบบ RFID

ถ้าจุดที่วัดเคลื่อนจากจุดศูนย์กลางของขดลวดตามแนวแกนขดลวด (แกน x) ดังนั้น H จะลดลงเมื่อระยะทาง x เพิ่มขึ้น กำลังสนามในความสัมพันธ์กับรัศมี(หรือพื้นที่) ของขดลวดยังคงคงที่ และต่อจากนั้นจะลดลงอย่างรวดเร็ว ในอวกาศ กำลังสนามจะสลายประมาณ $60dB$ ต่อสิบปี ในสนามระยะใกล้ของขดลวด และทำให้แบนเป็น $20dB$ ต่อสิบปี ในสนามระยะไกลของคลื่นแม่เหล็กไฟฟ้าซึ่งถูกสร้าง



รูปที่ 2.29 แสดงเส้นของฟลักซ์แม่เหล็กรอบขดลวดทรงกระบอกสั้น หรือลูปตัวนำ ซึ่งคล้ายกับในสายอากาศส่งของระบบ RFID

สมการข้างล่างสามารถใช้คำนวณเส้นทางของกำลังสนามบนแกน x ของขดลวด ซึ่งคล้ายกับสายอากาศส่งของการคัปเปิ้ลเหนี่ยวนำระบบ RFID

$$H = \frac{I \cdot N \cdot R^2}{2\sqrt{\left(R^2 + x^2\right)^3}} \qquad (2.3)$$

ซึ่ง N จำนวนของขดลวด

R รัศมีขดลวด r

x ระยะจากจุดศูนย์กลางของขดลวดในทิศ x

ขอบเขตที่ใช้ในสมการนี้คือ $d << R$ และ $x < \lambda / 2\pi$

ที่ระยะศูนย์ หรือที่จุดศูนย์กลางของสายอากาศ สมการสามารถเปลี่ยนได้เป็น

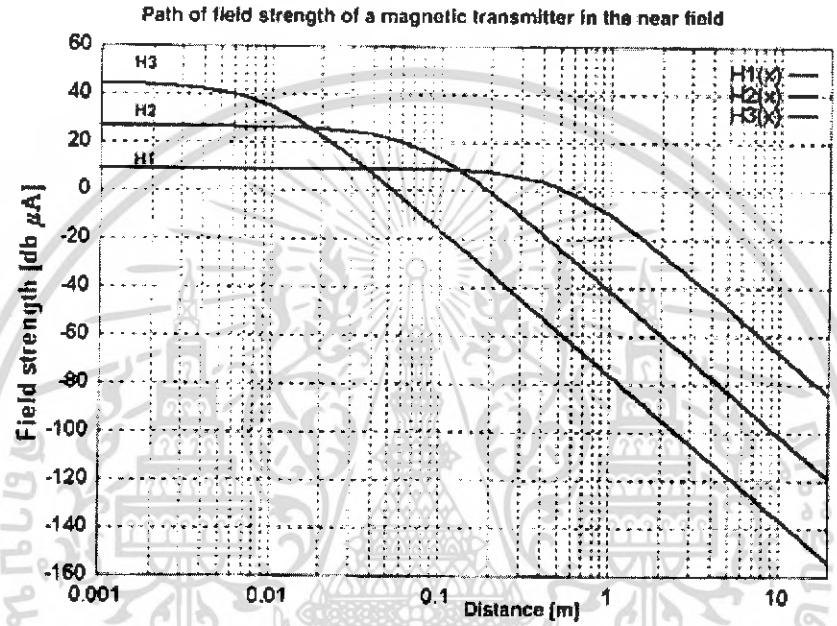
$$H = \frac{I \cdot N}{2R} \qquad (2.4)$$

เราสามารถคำนวณเส้นทางกำลังสนามของวงตัวนำสี่เหลี่ยมที่มีความยาวของขอบ $a \cdot b$ ใช้สมการดังนี้

$$H = \frac{N \cdot I \cdot ab}{4\pi\sqrt{\left(\frac{a}{2}\right)^2 + \left(\frac{b}{2}\right)^2 + x^2}} \cdot \left(\frac{1}{\left(\frac{a}{2}\right)^2 + x^2} + \frac{1}{\left(\frac{b}{2}\right)^2 + x^2} \right) \qquad (2.5)$$

รูปด้านล่างแสดงการคำนวณเส้นทางกำลังสนาม $H(x)$ สำหรับสายอากาศที่ต่างกัน 3 ชนิด ที่ระยะ 0-20 เมตร จำนวนของขดลวดและกระแสของสายอากาศคงที่ในแต่ละกรณี สายอากาศต่างกันแค่รัศมี R การคำนวณขึ้นกับค่าดังนี้ $H1: R = 55cm, H2: R = 7.5cm, H3: R = 1cm$

ผลจากการคำนวณยืนยันการเพิ่มของกำลังสนามที่ระบบใกล้ $(x < R)$ จากขดลวดสายอากาศ สายอากาศที่เล็กที่สุดแสดงกำลังสนามที่สูงกว่าที่จุดศูนย์กลางของสายอากาศ (ระยะทาง = 0) แต่ที่ระยะทางไกลกว่า $(x > R)$ สายอากาศที่ใหญ่ที่สุดสร้างกำลังสนามสูงขึ้น ผลกระทบนี้มีผลต่อการออกแบบสายอากาศคัปเปิ้ลเหนี่ยวนำระบบ RFID

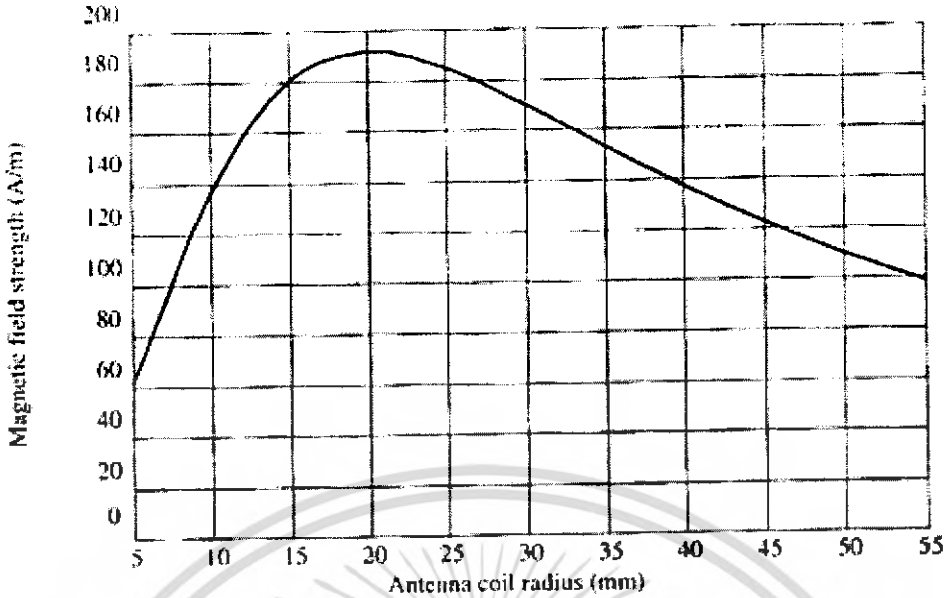


รูปที่ 2.30 แสดงเส้นทางกำลังสนามแม่เหล็ก H ในสนามระยะใกล้ของขดลวดทรงกระบอกสั้นหรือขดลวดตัวนำ ที่ระยะทางในทิศ x เพิ่มขึ้น

2.2.3 เส้นผ่านศูนย์กลางของสายอากาศที่ดีที่สุด

ถ้ารัศมี R ของสายอากาศส่งเปลี่ยนแปลงที่ระยะทางคงที่ x จากสายอากาศส่งภายใต้การใช้ค่าคงที่กระแสขดลวด I ในสายอากาศเครื่องส่ง กำลังสนาม H ถูกพบที่ค่าสูงสุด $R = x$ การเพิ่มรัศมี R ทำให้สัดส่วนกำลังสนามลดลง การลดรัศมี R ของสายอากาศส่ง ซึ่ง $x > a$ ทำให้เกิดการลดเป็นสัดส่วนในกำลังสนาม $H(x) \sim x^{-3}$ ในกรณีซึ่งระยะทาง x จะมากกว่ารัศมี R การเปลี่ยนแปลงรัศมีของขดลวด R ที่ระยะ x คงที่ ทำให้เส้นทางกำลังสนาม H เพิ่มขึ้น แสดงดังรูปที่ 2.24 กำลังสนาม H เพิ่มอย่างรวดเร็วที่ $x > R$ กำลังสนามจะถึงค่าสูงสุดที่ $x = R$

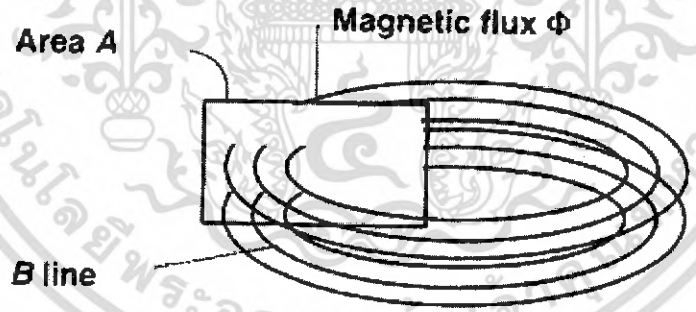
กฎนิ้วโป้ง เราสามารถแสดงได้ว่า ระยะการอ่านของเครื่องอ่านสัญญาณเช่น ช่วงที่ตัวแทคสามารถอ่านได้ ขึ้นกับรัศมีของสายอากาศเครื่องส่ง จากสิ่งนี้เราพบว่า รัศมีของสายอากาศที่เหมาะสม R สำหรับช่วงของระบบ x_{max} สามารถแสดงได้ว่า $R \approx x_{max}$ ถ้ารัศมีของสายอากาศ R มากเกินไป กำลังสนาม H อาจจะต่ำเกินไป ถึงแม้ว่าระยะทาง $x = 0$



รูปที่ 2.31 แสดงเส้นทางของกำลังสนาม H ที่ระยะ $x = 20mm$ ซึ่งรัศมีของขดลวด $R = 5 - 55mm$

2.2.4 ฟลักซ์แม่เหล็กและความหนาแน่นของฟลักซ์แม่เหล็ก

สนามแม่เหล็กของขดลวดจะทำให้เกิดแรงบนเข็มทิศ ถ้าใส่แกนเหล็กอ่อนในขดลวด แรงกระทำบนเข็มทิศจะเพิ่มขึ้น $I \times N$ และความเข้มสนามจะคงที่ อย่างไรก็ตามความหนาแน่นของฟลักซ์ หรือจำนวนฟลักซ์ทั้งหมดเพิ่มขึ้น



รูปที่ 2.32 แสดงความสัมพันธ์ระหว่างฟลักซ์แม่เหล็ก Φ และความหนาแน่นฟลักซ์ B

จำนวนเส้นทั้งหมดของฟลักซ์แม่เหล็กซึ่งผ่านขดลวดทรงกระบอก แสดงโดย ฟลักซ์แม่เหล็ก Φ ความหนาแน่นฟลักซ์แม่เหล็ก B จะสัมพันธ์กับพื้นที่ A ฟลักซ์แม่เหล็กแสดงได้ดังนี้

$$\Phi = B \cdot A \tag{2.6}$$

ความสัมพันธ์ระหว่างความหนาแน่นฟลักซ์ B และกำลังสนาม H ถูกแสดงโดยสมการนี้

$$B = \mu_0 \mu_r H = \mu H \tag{2.7}$$

ค่าคงที่ μ_0 คือค่าคงที่ของสนามแม่เหล็กและเพอร์มีอบิลิตี้ของสุญญากาศ $(\mu_0 = 4\pi \times 10^{-6} Vs/Am)$ ค่า μ_r เรียกว่าค่าเพอร์มีอบิลิตี้สัมพันธ์ และบ่งบอกถึงว่ามีค่ามากกว่าหรือน้อยกว่าค่าเพอร์มีอบิลิตี้ μ_0

2.2.5 ความเหนี่ยวนำของลูปตัวนำ

ถ้าเราให้เส้นผ่าศูนย์กลางของสาย d เล็มมากเมื่อเปรียบเทียบกับเส้นผ่าศูนย์กลางของขดลวดตัวนำ D $(d/D < 0.0001)$ ประมาณได้ดังนี้

$$L = N^2 \mu_0 R \cdot \ln\left(\frac{2R}{d}\right) \tag{2.8}$$

ซึ่ง R คือรัศมีของลูปตัวนำและ d คือเส้นผ่าศูนย์กลางของสายที่ใช้

2.2.6 ความเหนี่ยวนำเสมือน M

ถ้าลูปตัวนำตัวที่สองถูกตั้งไว้ในบริเวณใกล้เคียงกับลูปตัวนำตัวที่หนึ่ง ผ่านกระแสที่กำลังไหลดังนั้นนี่จะเป็นสัดส่วนของฟลักซ์แม่เหล็กทั้งหมดที่ผ่าน A_i วงจรสองวงจรที่เชื่อมต่อกันโดยฟลักซ์การคัปปลิ้ง แมกนิจูดของฟลักซ์ ψ_{21} ขึ้นกับมิติทางเรขาคณิตของลูปตัวนำทั้งสอง ตำแหน่งของลูปตัวนำจะสัมพันธ์กับอีกตัวหนึ่ง และคุณสมบัติทางแม่เหล็กของตัวกลาง

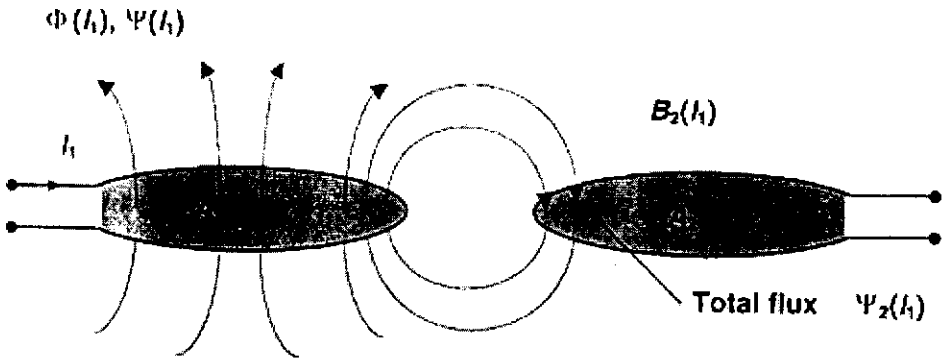
คล้ายกับคำจำกัดความของความเหนี่ยวนำ L ของลูปตัวนำ ความเหนี่ยวนำเสมือน M_{21} ของลูปตัวนำตัวที่สองในความสัมพันธ์กับลูปตัวนำตัวที่หนึ่ง จะกำหนดเป็นอัตราส่วนของฟลักซ์ ψ_{21} ของลูปตัวนำตัวที่สอง ต่อกระแส I_1 ในลูปตัวนำตัวที่หนึ่ง

$$M_{21} = \frac{\psi_{21}(I_1)}{I_1} = \oint_{A2} \frac{B_2(I_1)}{I_1} \cdot dA_2 \tag{2.9}$$

ความเหนี่ยวนำเสมือน M_{12} กระแส I_2 ผ่านลูปตัวนำตัวที่สอง ดังนั้นฟลักซ์การคัปปลิ้ง ψ_{12} ในลูปหนึ่ง จะได้ความสัมพันธ์ดังนี้

$$M = M_{12} = M_{21} \tag{2.10}$$

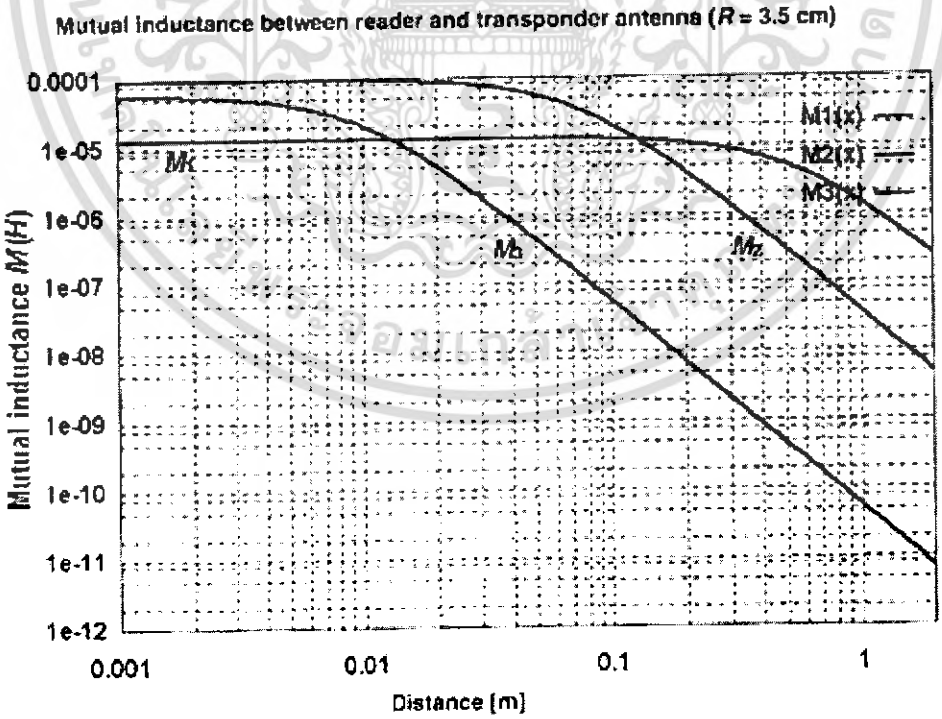
ความเหนี่ยวนำเสมือนอธิบายการคัปปลิ้งของวงจรสองวงจรผ่านตัวกลางสนามแม่เหล็ก ความเหนี่ยวนำเสมือนถูกนำเสนอระหว่างวงจรไฟฟ้าสองวงจร มิติและหน่วยคล้ายกันสำหรับความเหนี่ยวนำ



รูปที่ 2.33 แสดงการกำหนดความเหนี่ยวนำเสมือน M_{21} โดยการคัปปลิ้งของขดลวดสองอันผ่านการไหลของแม่เหล็กส่วนหนึ่ง

การคัปปลิ้งของวงจรไฟฟ้าสองวงจรผ่านสนามแม่เหล็กเป็นหลักการทางฟิสิคัลซึ่งการคัปเปิ้ลเหนี่ยวนำของระบบ RFID เป็นพื้นฐาน รูปที่ 2.27 แสดงการคำนวณของความเหนี่ยวนำเสมือนระหว่างสายอากาศแทคและสายอากาศของเครื่องอ่านสัญญาณที่ต่างกันสามชนิด ซึ่งต่างกันที่เส้นผ่านศูนย์กลางการคำนวณขึ้นกับค่าดังนี้ $M1: R = 55cm, M2: R = 7.5cm, M3: R = 1cm$, แทค : $R = 3.5cm.N = 1$ สำหรับสายอากาศของเครื่องอ่านสัญญาณทุกตัว

M_3 น้อยที่สุดเพราะว่าสายอากาศของ Reader เล็ก (1 เซนติเมตร) หมายความว่าสนามของสายอากาศของเครื่องอ่านสัญญาณสามารถผ่านสัดส่วนพื้นที่ A_2 ของสายอากาศของแทคได้น้อย



รูปที่ 2.34 แสดงเส้นทางของความเหนี่ยวนำเสมือนระหว่างสายอากาศของเครื่องอ่านสัญญาณและแทคในระยะทางทิศทาง x ที่เพิ่มขึ้น

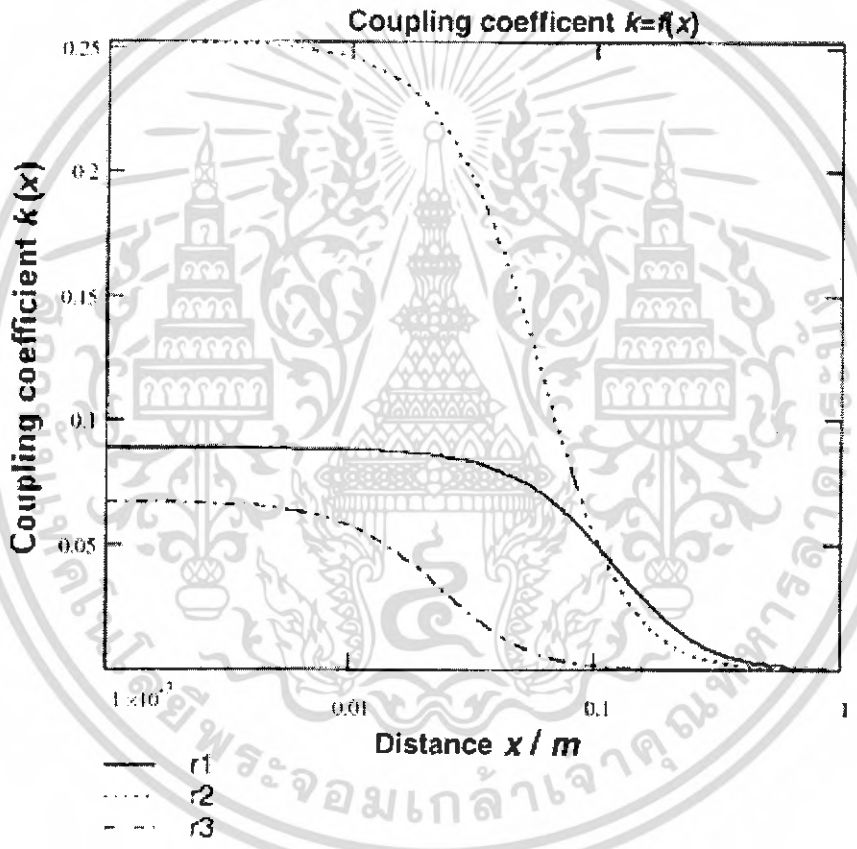
2.2.7 สัมประสิทธิ์การคัปปลิ้ง k

ความเหนี่ยวนำเสมือน เป็นการอธิบายเกี่ยวกับคุณภาพของการคัปปลิ้งฟลักซ์ ของลูปตัวนำสองลูป เราสามารถทำนายคุณภาพของการคัปปลิ้งของลูปตัวนำดังสมการนี้

$$k = \frac{M}{\sqrt{L_1 \cdot L_2}} \tag{2.11}$$

สัมประสิทธิ์การคัปปลิ้งจะเป็นค่าระหว่าง $0 \le k \le 1$

- $k = 0$ ไม่เกิดการคัปปลิ้งกันเลยเนื่องจากระยะทางที่ไกลหรือการป้องกันเกี่ยวสนามแม่เหล็ก
- $k = 1$ เกิดการคัปปลิ้งทั้งหมดขดลวดทั้งสองมีฟลักซ์แม่เหล็กเหมือนกัน ทรานสฟรอมเมอร์เป็นเทคนิคสำหรับการคัปปลิ้งทั้งหมด



รูปที่ 2.35 แสดงสัมประสิทธิ์การคัปปลิ้งสำหรับลูปตัวนำที่มีขนาดแตกต่างกัน สายอากาศแท็กส์: $r_{Transp} = 2cm$ สายอากาศของเครื่องอ่าน: $r_1 = 10cm, r_2 = 7.5cm, r_3 = 1cm$

การวิเคราะห์การคำนวณเป็นไปได้สำหรับรูปร่างสายอากาศที่ง่ายๆ สำหรับลูปตัวนำขนานกันสองอันที่มีจุดศูนย์กลางบนแกน x สัมประสิทธิ์การคัปปลิ้งจะสามารถประมาณได้จากสมการข้างล่าง อย่างไรก็ตามสมการนี้จะใช้ได้กรณีที่รัศมีของลูปตัวนำมีสภาพเป็น $r_{Transp} \le r_{reader}$ ระยะทางระหว่างลูปตัวนำบนแกน x ถูกแสดงโดย x

$$k(x) = \frac{r_{Transp}^2 \cdot r_{\mathrm{Reader}}^2}{\sqrt{r_{Transp} \cdot r_{\mathrm{Reader}}} \cdot \left(\sqrt{x^2 + r_{\mathrm{Reader}}^2}\right)^3} \qquad (2.12)$$

เนื่องจากความสัมพันธ์ระหว่างสัมประสิทธิ์การคัปปลิ้งและความเหนี่ยวนำเสมือน M $M = M_{12} = M_{21}$ สูตรนี้ใช้สำหรับสายอากาศของเครื่องส่งเล็กกว่าสายอากาศของแทค ซึ่ง $r_{Transp} \geq r_{reader}$ สามารถเขียนได้ดังนี้

$$k(x) = \frac{r_{Transp}^2 \cdot r_{\mathrm{Reader}}^2}{\sqrt{r_{Transp} \cdot r_{\mathrm{Reader}}} \cdot \left(\sqrt{x^2 + r_{Transp}^2}\right)^3} \qquad (2.13)$$

สัมประสิทธิ์การคัปปลิ้ง $k(x) = 1$ หรือ เท่ากับ 100 % จะทำได้เมื่อระยะทางระหว่างลูปตัวนำเป็นศูนย์ $(x = 0)$ และรัศมีของสายอากาศเท่ากัน $r_{Transp} = r_{reader}$ เพราะในกรณีนี้ลูปตัวนำอยู่ที่เดียวกัน และมีฟลักซ์แม่เหล็กตรงกัน ในทางปฏิบัติสัมประสิทธิ์การคัปปลิ้งอาจจะน้อยเท่ากัน 0.01(<1%)

2.2.8 มาตรฐานอาร์เอฟไอดี

โดยมาตรฐานระหว่างประเทศสำหรับการใช้งานอาร์เอฟไอดีมีอยู่ 2 หน่วยงานหลัก ได้แก่ International Organization of Standard หรือ ISO และ EPC Global โดยที่มาตรฐานของอาร์เอฟไอดีมีการกำหนดไว้ 4 ด้านดังนี้

- มาตรฐานด้านเทคโนโลยี (Technology)
- มาตรฐานรูปแบบของข้อมูล (Data format)
- มาตรฐานวิธีการทดสอบ (Conformance)
- มาตรฐานการใช้งาน (Application)

ตารางที่ 2.2 แสดงการเปรียบเทียบมาตรฐานระหว่าง ISO/IEC และ EPC

	ISO/IEC	**EPC**
เทคโนโลยี	ISO/IEC 18000 – RF-ID for Item Management Part 2 - < 135 kHz Part 3 – 13.56 MHz Part 4 – 2450 MHz Part 6 – 860 – 960 MHz Part 7 – 433.92 MHz (active)	Class I-V (13.56 and UHF only) Class 0/Class I: read-only passive tags Class II tags: passive tags with additional functionality Class III tags: semi-passive RF-ID tags Class IV tags: active tags with broad-band peer-to-peer communication Class V tags: Readers Can power other Class I, II and III tags; Communicate with Classes IV and V

รูปแบบของข้อมูล	ISO/IEC 15418 – Application Identifiers & Data Identifiers ISO/IEC 15434 – Syntax ISO/IEC 15459 – Transport License Plate ISO/IEC 15961 – Data Protocol: Application Interface ISO/IEC 15962 – Data Protocol: Data Encoding Rules and Logical Memory Functions	EPC Class 0 – 64 bits Class 1 – 96 bits Class 1 – G2 – 128/256 bits Class2 – Class 1 with larger memory and read/write Class 3 – Class 2 with sensors (semi-passive) Class 4 – passive tags
วิธีการทดสอบ	ISO/IEC 18047 – RF-ID device conformance test methods	-
การใช้งาน	Vary by Industries e.g. ISO 10374 - Freight containers – Automatic identification ISO 18185 – Freight Containers Radio-frequency communication protocol for electronic seal ISO 11785 – Radio – frequency identification of animals	-

หมายเหตุ EPC: Electronic Product Code คือการกำหนดรหัสสินค้าโดยใช้ระบบอิเล็กทรอนิกส์ และนอกเหนือจาก ISO และ EPC Glocal แล้วยังมีหน่วยงานอื่นอีก เช่น Ubiquitous ID หรือมาตรฐาน UID ที่ทางประเทศญี่ปุ่นให้การสนับสนุนและกำหนดมาตรฐานเพื่อใช้งานในประเทศโดยมีความแตกต่างกับ ISO และ EPC Global ในเชิงรายละเอียดทางเทคนิค หรือจะเป็นมาตรฐาน AIM (Automatic Identification Manufacturers) ที่กำหนดโดย AIDC (Automatic Identification and Data Collection) ซึ่งเป็นผู้เริ่มต้นทำรหัสแท่ง เป็นต้น

2.3 โปรแกรมที่ใช้สร้างแอปพลิเคชัน

2.3.1 โปรแกรม Delphi

Delphi เป็นซอฟต์แวร์ตัวแปลภาษาปาสกาล (Pascal) ตัวหนึ่ง ที่พัฒนาโดยบริษัท Borland ซึ่งรูปแบบของภาษานั้น ได้ถูกออกแบบระบบภาษาใหม่ ใช้ในการเขียนโปรแกรมเพื่อสร้างแอปพลิเคชันหรือซอร์ฟแวร์ โดยจะประกอบด้วยเครื่องมือชนิดต่างๆ ที่ใช้ให้การเขียนโปรแกรมทำได้อย่างสะดวกและเพื่อสนับสนุนการเขียนโปรแกรมแบบเชิงวัตถุ (Object oriented programming หรือเรียกแบบย่อว่า

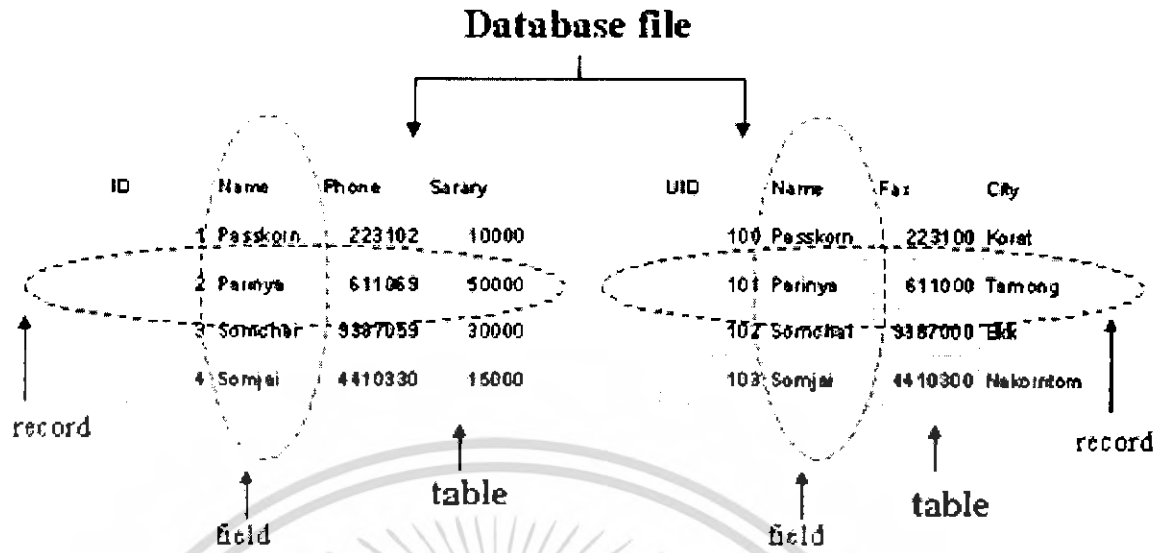
OOP) การเขียนโปรแกรมแบบเชิงวัตถุนั้น มีความแตกต่างกับการเขียนโปรแกรมแบบโครงสร้างอยู่ด้วยกันหลายจุด ซึ่งส่วนใหญ่แล้ว ความสามารถของโปรแกรมเชิงวัตถุ ยังคงต้องการความรู้ของการเขียนโปรแกรมแบบโครงสร้างเป็นพื้นฐาน ทั้งนี้เนื่องจากภาษาโปรแกรมเชิงวัตถุเป็นภาษาที่ออกแบบมาเพื่อเพิ่มประสิทธิภาพของการเขียนโปรแกรมในรูปแบบเดิม คือ ให้ง่ายต่อการพัฒนาต่อ และสามารถนำไปใช้ได้

2.3.2 โปรแกรม MySQL

MySQL เป็นโปรแกรมฐานข้อมูลที่ใช้จัดเก็บข้อมูลโปรแกรมหนึ่ง ทำงานในลักษณะ Client Server ทำงานบนระบบ Telnet บน Linux Redhad หรือ Unix System ซึ่งไม่เสียค่าใช้จ่าย และบน Win32 ซึ่งต้องเสียค่าใช้จ่าย ทั่วไปบนระบบเครือข่ายอินเตอร์เน็ตและอินทราเน็ตนั้นหมายความว่าเราสามารถเรียกใช้ MySQL ได้ทั่วโลกกรณีเป็นอินเตอร์เน็ต และทั่วบริเวณที่เป็นอินทราเน็ตและยังสามารถเรียกใช้บนเว็ปบราวเซอร์ ได้กรณีใช้ภาษาเป็นอินเตอร์เฟส ในการเชื่อมภาษาที่ใช้เป็นอินเตอร์เฟส เช่น Delphi, PHP, Perl , C, C++ ฯลฯ

MySQL ถือเป็นระบบจัดการฐานข้อมูล มีลักษณะเป็นโครงสร้างของการเก็บรวบรวมข้อมูล การที่จะเพิ่มเติม เข้าถึงหรือประมวลผลข้อมูลที่เก็บในฐานข้อมูลจำเป็นจะต้องอาศัยระบบจัดการฐานข้อมูล ซึ่งจะทำหน้าที่เป็นตัวกลางในการจัดการกับข้อมูลในฐานข้อมูลทั้งสำหรับการใช้งานเฉพาะ และรองรับการทำงานของแอปปลิเคชันอื่นๆ ที่ต้องการใช้งานข้อมูลในฐานข้อมูล เพื่อให้ได้รับความสะดวกในการจัดการกับข้อมูลจำนวนมาก MySQL ทำหน้าที่เป็นทั้งตัวฐานข้อมูลและระบบจัดการฐานข้อมูล MySQL เป็นระบบจัดการฐานข้อมูลแบบ relational ฐานข้อมูลแบบ relational จะทำการเก็บข้อมูลทั้งหมดในรูปแบบของตารางแทนการเก็บข้อมูลทั้งหมดลงในไฟล์เพียงไฟล์เดียว ทำให้ทำงานได้รวดเร็วและมีความยืดหยุ่น นอกจากนั้น แต่ละตารางที่เก็บข้อมูลสามารถเชื่อมโยงเข้าหากันทำให้สามารถรวมหรือจัดกลุ่มข้อมูลได้ตามต้องการ โดยอาศัยภาษา SQL ที่เป็นส่วนหนึ่งของโปรแกรม MySQL ซึ่งเป็นภาษามาตรฐานในการเข้าถึงฐานข้อมูล

ไม่ว่าจะจัดเก็บข้อมูลโดยตู้บัตรรายการหรือจัดเก็บข้อมูลโดยคอมพิวเตอร์นั้น จำเป็นจะต้องมีรูปแบบในการจัดเก็บข้อมูลที่เหมาะสม ซึ่งรูปแบบการจัดเก็บข้อมูลทั้งสองอย่างรวมทั้ง MySQL ด้วยนั้นไม่แตกต่างกันนัก โดยสามารถแบ่งได้ตามภาพข้างล่างดังนี้



รูปที่ 2.36 รูปแบบในการจัดเก็บฐานข้อมูล

ซึ่งการจัดเก็บฐานข้อมูลนั้นอธิบายคร่าวๆก็คือจะทำโดยรวมข้อมูลทั้งหมดที่จัดเก็บเข้าพวกเดียวกันเรียกว่า ไฟล์ฐานข้อมูล จากนั้นภายในไฟล์ฐานข้อมูล ก็จะประกอบไปด้วยตารางต่างๆที่จัดเก็บข้อมูล แล้วภายในตารางก็จะประกอบไปด้วยคอลัมน์ ซึ่งจะบ่งบอกลักษณะของข้อมูลจากนั้นก็เป็นแถว ซึ่งเป็นส่วนของข้อมูลที่ถูกจัดเข้าจำพวกด้วยกันตามลักษณะการแบ่งพวกของคอลัมน์

2.3.3 ภาษา PHP

PHP หรือ Professional Home Pages เป็นภาษาสคริปต์แบบเซิร์ฟเวอร์ไซด์ (server-side scripting language) หมายถึง การประมวลผลจะเกิดขึ้นบนเครื่องแม่ข่าย หรือเซิร์ฟเวอร์ (server) แล้วจึงสร้างผลลัพธ์เป็นภาษา HTML ส่งให้กับเครื่องลูกข่ายหรือไคลเอนต์ (client) เพื่อแสดงผล ซึ่งลดภาระการส่งถ่ายข้อมูลจำนวนมากเพื่อมาประมวลผลบนเครื่องลูกข่าย PHP เป็น Complier หรือ Interpreter PHP เป็น Interpreter ประมวลผลการทำงานโดยแปลความหมายทีละบรรทัด

ข้อดีของ Interpreter คือ เป็นการ Open source โปรแกรมที่ Open source จะพัฒนาอย่างรวดเร็ว เนื่องจาก เกิดการ Copy แก้ไข ตลอดจนพัฒนาขึ้นมาใหม่ตามแนวทางตัวอย่าง การที่มีต้นแบบหรือต้นฉบับ จะทำให้ไม่เสียเวลาเขียนใหม่ สามารถนำไปแก้ไขนิดหน่อยก็ใช้ได้ ตัวอย่าง Open source เช่น Linux Java Script Perl PHP ASP เป็นต้น

2.3.3.1 รูปแบบภาษา

PHP แตกต่างจากภาษาสคริปต์แบบอื่นๆคือ PHP ได้รับการพัฒนาและออกแบบมาเพื่อใช้งานในการสร้างเอกสารแบบ HTML โดยสามารถสอดแทรกหรือแก้ไขเนื้อหาได้โดยอัตโนมัติ สามารถนำมาใช้ทำเว็บเพจที่จำเป็นต้องมีการตอบนองกับผู้ใช้โดยเฉพาะอย่างยิ่ง PHP มีความสามารถในการนำข้อมูลจาก

Database Server มาแสดงในเว็บเพจจึงเหมาะแก่การนำมาใช้ทำเว็บบอร์ด, เว็บเมล์, ไดนามิกเว็บเพจเพื่อประโยชน์ในทางพาณิชย์อิเล็กทรอนิกส์ (e-commerce) ตลอดจนการสร้างเว็บแอปปลิเคชันเพื่อใช้ในองค์กรที่ต้องการคุณสมบัติในการเรียกใช้งานได้จากทุกที่ เช่นการเรียกใช้แอปปลิเคชันจากสาขาต่างๆ

2.3.3.2 สิ่งที่ต้องมีในการใช้ PHP

เมื่อตัดสินใจที่จะนำ PHP มาใช้แล้วสิ่งที่ต้องมีคือ

- เครื่องเซิร์ฟเวอร์ ซึ่งใช้ PC ธรรมดาก็ได้
- เครื่องไคลเอนต์ จะเป็นเครื่องเดียวกับเครื่องเซิร์ฟเวอร์ก็ได้
- โปรแกรมเว็บเซิร์ฟเวอร์ (Web Server) เช่น Apache, Microsoft Internet Information Server (IIS), Microsoft Personal Web Server (PWS)
- โปรแกรม PHP
- โปรแกรมดาต้าเบสเซิร์ฟเวอร์ (Database Server) เช่น MySQL, PostgreSQL, Microsoft SQL Server

2.3.3.3 PHPMyAdmin

การจัดการฐานข้อมูล MySQL ผ่านทาง command promt นั้นเป็นวิธีที่ผู้ใช้งาน (หรือผู้บริหารระบบ) จำเป็นต้องจดจำคำสั่งให้ได้ และอาจไม่ได้ผลดังที่ต้องการหากพิมพ์คำสั่งผิดไปเพียงเล็กน้อย ส่วนการนำฟังก์ชันใน PHP มาใช้จัดการกับฐานข้อมูลและตารางจะช่วยให้เราสามารถสร้างโปรแกรมสำหรับจัดการฐานข้อมูล MySQL เพื่อดำเนินการกับฐานข้อมูลและตารางต่างๆ โดยไม่ต้องพิมพ์คำสั่งผ่านทาง command prompt ของ MySQL แต่การพิมพ์คำสั่งผ่านทาง command prompt นั้นไม่นิยมใช้กัน จึงได้ใช้โปรแกรม phpMyAdmin ในการเข้าจัดการฐานข้อมูลและตารางใน MySQL โดยแทนที่จะต้องพิมพ์คำสั่งผ่านทาง command prompt ก็เปลี่ยนมาใช้อินเทอร์เฟสแบบกราฟฟิกแทน ทำให้การสร้างฐานข้อมูล ตาราง รวมถึงการเพิ่มข้อมูล แก้ไขข้อมูล และลบข้อมูลในตารางทำได้ดีขึ้น

ความสามารถของ phpMyAdmin เช่น

- สร้างและลบฐานข้อมูล
- สร้าง ทำสำเนา ลบ เปลี่ยนชื่อ และแก้ไขโครงสร้างของตาราง
- เพิ่ม แก้ไข และลบฟิลด์
- สั่งดำเนินการด้วยคำสั่งในภาษา SQL ผ่าน phpMyAdmin
- อ่านข้อมูลจากเท็กซ์ไฟล์เข้าสู่ตาราง
- จัดการได้หลายเซิร์ฟเวอร์
- จัดการรายชื่อผู้ใช้และกำหนดสิทธิการใช้งานของผู้ใช้
- สามารถใช้ Query-by-example (QBE) ได้

- สามารถส่งออกโครงสร้างและคำสั่งการสร้างข้อมูลในตารางเป็นคำสั่ง SQL หรือรูปแบบ CSV (Comma-Separated Values), LaTeX และสามารถเลือกให้บีบอัดเป็นไฟล์ zip หรือ gzipped ได้
- แสดงผลเป็นภาษาต่างๆได้ถึง 47 ภาษา ซึ่งรวมทั้งภาษาไทยด้วย

2.4 มาตรฐาน RS-485

RS-485 ได้รับการออกแบบให้รองรับมาตรฐาน TIA/EIA-485-A จุดเด่นที่สำคัญคือ สามารถเชื่อมต่ออุปกรณ์ได้มากกว่า 2 ตัว (โดยทั่วไปกำหนดอยู่ที่ 32 ตัว) เพื่อทำการสื่อสารข้อมูลร่วมกัน และระยะห่างไกลสุดระหว่างอุปกรณ์ภายในเครือข่ายอยู่ที่ประมาณ 1.2 กิโลเมตร (ขึ้นอยู่กับความสามารถของอุปกรณ์ไอซีที่ทำหน้าที่เป็นตัวขับสัญญาณด้วย) ด้วยคุณสมบัติ 2 ข้อนี้ ทำให้มีความเหมาะสมกับการนำไปใช้งานในระบบสื่อสารข้อมูลได้เป็นอย่างดี

ลักษณะโดยทั่วไปของการเชื่อมต่อข้อมูลแบบอนุกรมตามมาตรฐาน RS-232 คือเป็นการสื่อสารข้อมูลแบบจุดต่อจุด ซึ่งเดิมทีเป็นการสื่อสารข้อมูลระหว่างคอมพิวเตอร์กับโมเด็ม ซึ่งจริงๆแล้วทั้งสองฝั่งจะเป็นอะไรก็ได้ การสื่อสารเป็นแบบสองทางพร้อมกัน (Full-duplex) โดยอาจใช้สายสัญญาณอื่นร่วมเพื่อทำแฮนด์เชค (Hand-shake) หรือไม่ก็ได้ มาตรฐาน RS-232 จำกัดความยาวสายไว้ที่ 50 ฟุต (หรือประมาณ 15 เมตร) สำหรับการส่งสัญญาณที่ความเร็ว 19,200 บิทต่อวินาที โดยที่ความยาวสายจะต้องสั้นลงถ้าต้องการสื่อสารที่ความเร็วสูงขึ้น และถ้ามีสัญญาณรบกวนมากๆ เช่นในโรงงาน หรือบริเวณใกล้เครื่องจักรที่เป็นแบบมีการสวิทช์สัญญาณไฟฟ้าที่กระแสสูงๆ ก็จะทำให้ต้องมีการลดความเร็วในการส่งสัญญาณลงหรือใช้สายที่สั้นลง

มาตรฐาน RS-422 หรือ RS-422-A ถูกกำหนดขึ้นโดยสมาคมผู้ประกอบการอุตสาหกรรมอิเล็กทรอนิกส์หรือ EIA เช่นเดียวกันกับมาตรฐาน RS-232 โดยมีจุดมุ่งหมายที่จะแก้ปัญหาเรื่องความยาวของสายสื่อสารโดยใช้การส่งสัญญาณแบบผลต่าง (Differential) แทนที่จะใช้การส่งสัญญาณแบบอ้างอิงกับจุดกราวนด์ (หรือสายดิน) เช่นเดียวกันกับ RS-232 การส่งสัญญาณแบบ Differential นี้ช่วยลดปัญหาสัญญาณรบกวนจาก 2 ปัจจัยด้วยกัน ได้แก่ ปัญหาแรงดันกราวนด์ 2 ฝั่งสายไม่เท่ากัน อันเกิดจากกระแสไฟฟ้าที่ไหลในสายกราวนด์ที่ยาวมากๆ ก่อให้เกิดความต่างศักดิ์ และปัญหาสัญญาณรบกวนที่เกิดจากแม่เหล็กไฟฟ้าเหนี่ยวนำในสาย โดยหากสายไฟที่ใช้ถูกตีเกลียวและวางไว้ใกล้กัน เมื่อมีแรงดันเหนี่ยวนำจะปรากฏแรงดันรบกวนบนสายทั้งสองเท่าๆ กันเป็นผลให้ ตัวรับที่อ่านความต่างศักดิ์ระหว่างสายอ่านข้อมูลได้เช่นเดิม ทั้งสองปัจจัยนี้เองเป็นสาเหตุที่ทำให้ความต้านทานต่อสัญญาณรบกวนของการสื่อสารแบบ RS-232 ด้อยกว่า RS-422) ตามมาตรฐาน RS-422 นี้จะใช้สายสัญญาณทั้งหมด 4 เส้น (2 เส้นสำหรับการส่งสัญญาณ และอีก 2 เส้นสำหรับรับสัญญาณ) และสามารถใช้ความยาวสายสัญญาณได้ถึง 4,000 ฟุต (หรือ 1.2 กม.) ที่ความเร็ว 100,000 บิทต่อวินาที และการสื่อสารเป็นแบบ 2 ทางพร้อมกัน (Full Duplex)

มาตรฐาน RS-485 กำหนดโดยสมาคมผู้ประกอบการอุตสาหกรรมอิเล็กทรอนิกส์หรือ EIA เป็นมาตรฐานการเชื่อมต่อสัญญาณแบบอนุกรม (Serial Communication) มีลักษณะการเชื่อมต่อเป็นแบบหลายจุด (Multi-point) หรือ Multi-drop สายสัญญาณที่ใช้มีทั้งแบบที่เป็น 2 สายและแบบที่เป็น 4 สาย การต่อแบบหลายจุดนี้ทำให้สามารถมองสายสัญญาณเป็นบัสนำสัญญาณได้ (Signal Bus) จำนวนคอมพิวเตอร์หรืออุปกรณ์ที่สามารถอยู่บน RS-485 บัสหนึ่งถูกกำหนดไว้ที่ 32 ตัว ในกรณีที่ต้องการเพิ่มจะต้องมีตัวทวนสัญญาณ (Signal Repeater) หรือใช้ตัวส่ง-รับสัญญาณที่มีอิมพิแดนซ์ (ความต้านทานเสมือน) สูงขึ้น ซึ่งเราอาจเพิ่มจำนวนจุดเชื่อมต่อขึ้นได้ถึง 128 จุด ความยาวของสายสัญญาณตามมาตรฐาน RS-485 นี้สามารถยาวได้ถึง 1.2 กม เช่นเดียวกับมาตรฐาน RS-422 แต่การสื่อสารจะเป็นแบบสองทางไม่พร้อมกัน (Half Duplex) มีเพียงคอมพิวเตอร์หรืออุปกรณ์ตัวเดียวเท่านั้นที่สามารถส่งสัญญาณออกได้ ณ เวลาหนึ่งๆ ส่วนที่เหลือจะเป็นผู้รับสัญญาณ หรือผู้ฟัง

บทที่ 3
การคำนวณและการสร้าง

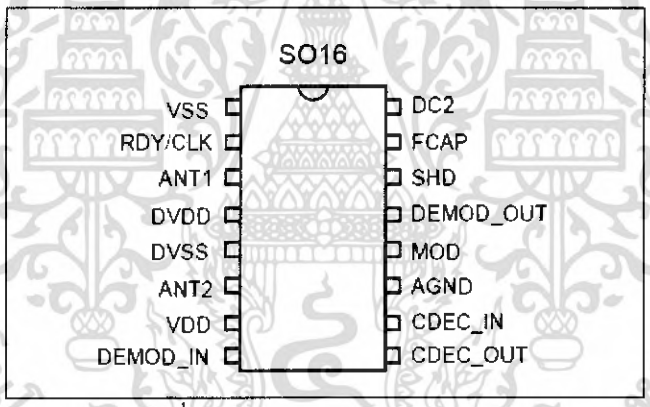
3.1 การออกแบบส่วนของ RFID

ระบบ RFID จะมีในส่วนของเครื่องอ่านสัญญาณและส่วนของแผ่นป้ายระบุข้อมูล ในแต่ละส่วนจะมีไอซีที่ใช้งานแตกต่างกันไป โดยในส่วนของเครื่องอ่านสัญญาณเราจะใช้ไอซี EM4095 และในส่วนของแผ่นป้ายข้อมูล เราจะใช้เป็นแบบ T5557

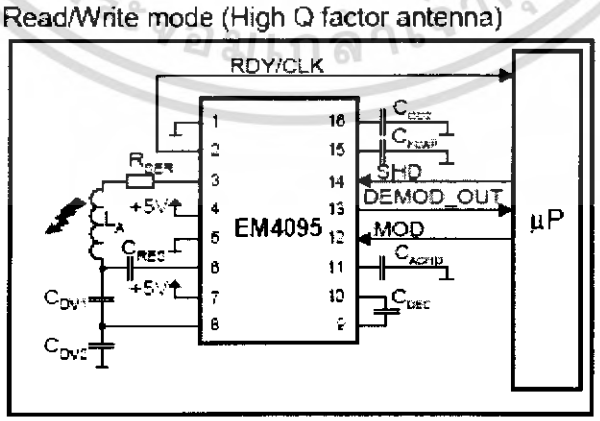
3.1.1 EM4095 : Read/Write analog front end for 125 kHz RFID Basestation

ไอซี EM4095 (หรือ P4095) เป็นไอซีแบบวงจรรวม CMOS สร้างขึ้นมาเพื่อใช้ในเครื่องอ่านสัญญาณซึ่งมีคุณสมบัติดังนี้

- มีสายอากาศขับเคลื่อนด้วยความถี่คลื่นพาห์
- การมอดดูเลตสัญญาณแบบ AM สำหรับเขียนข้อมูล
- การดีมอดดูเลตสัญญาณแบบ AM จากเสาอากาศที่ตรวจจับได้



รูปที่ 3.1 แสดงขาของไอซี EM4095



รูปที่ 3.2 แสดงวงจรที่ใช้ต่อ EM4095 กับไมโครคอนโทรเลอร์

เหตุผลที่ใช้วงจรนี้ก็เพราะว่าวงจรนี้มีประสิทธิภาพสูง เนื่องจากวงจรนี้มีการต่อค่าความต้านทานอนุกรมกับขดลวดทำให้กระแสที่ไหลผ่านขดลวดมีค่าลดลง เมื่อกระแสลดลงก็จะมีผลให้ค่าการกระจายตัวของกำลังมีค่าลดลงด้วย

3.1.1.1 Pin Description

ตารางที่ 3.1 แสดง Pin Description

Pin	Name	Description	Type
1	V_{SS}	Negative power supply (substrate)	GND
2	RDY/CLK	Ready flag and clock output, driver for AM modulation	O
3	ANT1	Antenna driver	O
4	D_{VDD}	Positive power supply for antenna drivers	PWR
5	D_{VSS}	Negative power supply for antenna drivers	GND
6	ANT2	Antenna driver	O
7	V_{DD}	Positive power supply	PWR
8	DEMOD_IN	Antenna sensing voltage	ANA
9	CDEC_OUT	DC blocking capacitor connection « out »	ANA
10	CDEC_IN	DC blocking capacitor connection « in »	ANA
11	A_{GND}	Analog ground	ANA
12	MOD	A High level voltage modulates the antenna	IPD
13	DEMOD_OUT	Digital signal representing the AM seen on the antenna	O
14	SHD	A High level voltage forces the circuit into sleep mode	IPU
15	FCAP	PLL Loop filter capacitor	ANA
16	DC2	DC decoupling capacitor	ANA

GND: reference ground PWR: power supply ANA: analog signal
IPD: input with internal pull down IPU: input with internal pull up O: output

3.1.1.2 การกำหนดการทำงาน

ในการกำหนดการทำงานของไอซี EM4095 สามารถกำหนดการทำงานได้จากไมโครคอนโทรลเลอร์ (AT89S8252) ไปควบคุมการทำงานจากขา SHD (ขา 14) และ MOD (ขา 12) ของ EM4095 ซึ่งเราสามารถกำหนดได้ดังนี้

ก. ขา SHD

1.1 ถ้า SHD เป็น High ไอซี EM4095 จะทำงานใน Sleep mode

1.2 ถ้า SHD เป็น Low ไอซี EM4095 จะพร้อมในการรับส่งข้อมูลตลอดเวลา

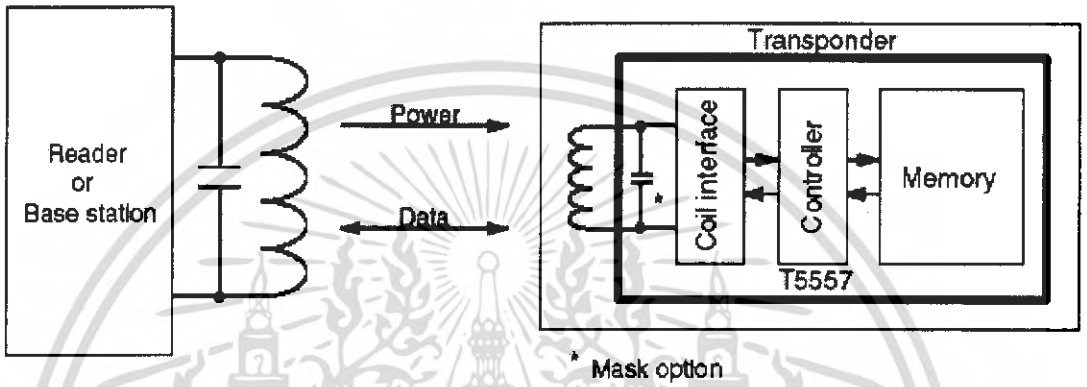
ข. ขา MOD

2.1 ถ้า MOD เป็น High ไอซี EM4095 จะทำงานใน mode write

2.2 ถ้า MOD เป็น Low ไอซี EM4095 จะทำงานใน mode read

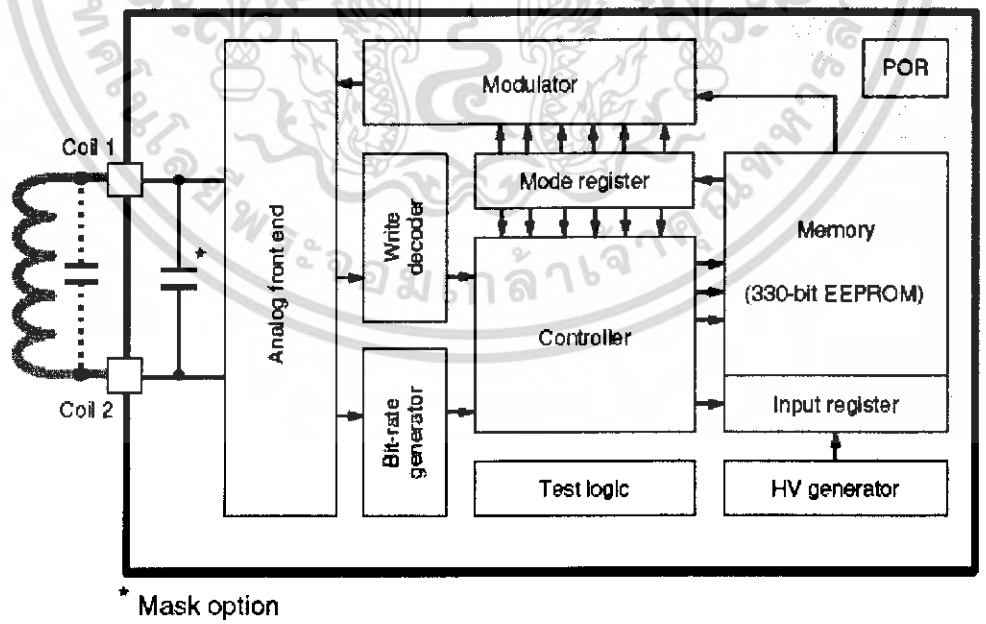
3.1.2 Tag : Multifunctional 330 bit Read/Write RF-Identification IC

T5557 เป็น Identification IC แบบไร้สัมผัส สำหรับการประยุกต์ใช้ในย่านความถี่ 125 kHz มีขดลวด (coil) ต่ออยู่กับชิพ (chip) ใช้เป็นแหล่งจ่ายพลังงาน (Power supply) และรับส่งข้อมูลระหว่างตัวเองกับเครื่องอ่านสัญญาณ (Reader or Base station)ใน T5557 นี้ มีหน่วยความจำเป็นแบบ EEPROM 330-bit (แบ่งเป็น 10 บล็อก บล็อกละ 33 บิต) สามารถอ่านและเขียนบล็อกได้จาก Reader โดยบล็อก 0 จะถูกสงวนไว้ในการตั้งค่าโหมดการทำงานของ T5557 และ บล็อก 7 อาจใช้สำหรับเก็บรหัส (Password) เพื่อป้องกันผู้อื่นมาเขียน



รูปที่ 3.3 แสดง RFID System Using T5557 Tag

3.1.2.1 T5557 Building Blocks



รูปที่ 3.4 แสดงบล็อกไดอะแกรมของ T5557

3.1.2.2 คุณสมบัติของ T5557

- ส่งผ่านข้อมูลแบบไร้สัมผัส
- ความถี่คลื่นวิทยุ FRF ตั้งแต่ 100 kHz ถึง 150 kHz
- e5550 เข้ากันได้กับ T5557 extended mode
- ขนาดเล็ก,เหมะสมกับ ISO/IEC 11784/785
- มีตัวเก็บประจุเรโซแนนซ์ 75 *pF* บนชิพ
- มีหน่วยความจำข้อมูลแบบ EEPROM ขนาด 7×32 บิต รวมทั้งพาสเวิร์ดขนาด 32 บิต
- แบ่งความจำ 64 บิตไว้สำหรับข้อมูล
- อีก 32 บิตของรีจิสเตอร์ใน EEPROM นั้นกำหนด

  อัตราข้อมูล

  - RF/2 ถึง RF/8, Binary Selectable หรือ
  - Fixed e5550 Data Rates

  การมอดูเลต/การเข้ารหัส

  - FSK, PSK, Manchester, Biphase, NRZ

3.1.2.3 การระบบการสั่งงานของ T5557

3.1.2.3.1 Initialization and POR Delay

วงจร POR จะยังทำงานจะกระทั่งถึงจุดที่มีแรงดันเท่ากับแรงดัน threshold จึงทำให้ค่าเริ่มต้นของ trigger เริ่มหน่วงเป็นลำดับ ระหว่างการทำงานนี้จะมีคาบเวลาประมาณ 192 สัญญาณนาฬิกา ไอซี T5557 ถูกกำหนดค่าเริ่มต้นให้เก็บข้อมูลไว้ใน EEPROM บล็อก 0 ระหว่างการกำหนดค่าเริ่มต้นที่บล็อก 0 ตัวแปรทั้งหมดของ T55570x ใน load damping จะทำงานอย่างถาวร และแบบ T55571x จะมีระยะการอ่านที่มากกว่าแต่ขึ้นอยู่กับสนามของคลื่น

3.1.2.3.2 Tag to Reader Communication

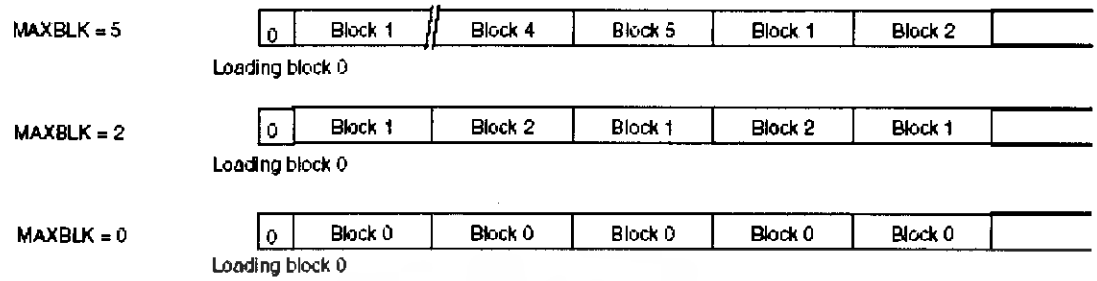
ในระหว่างการสั่งงานทั่วไป ข้อมูลที่เก็บไว้ใน EEPROM จะถูกวนและที่จุด Coil 1, Coil 2 จะถูกมอลดูเลท ซึ่งสามารถตรวจจับได้โดยเครื่องอ่านสัญญาณ

3.1.2.1.3 Regular-read Mode

ใน regular-read mode ข้อมูลจากหน่วยความจำจะถูกส่งอย่างต่อเนื่อง โดยเริ่มจาก block 1 bit 1 จนถึง บล็อกสุดท้าย โดยเราสามารถกำหนดบล็อกสุดท้ายได้ในพารามิเตอร์ MAXBLK ใน EEPROM block 0

เราสามารถกำหนดวงรอบของข้อมูลได้จากการตั้งค่า MAXBLK (ตั้งแต่ 0 ถึง 7) ถ้าเราตั้งค่า MAXBLK เท่ากับ 7 บล็อก 1 ถึงบล็อก 7 ก็จะถูกอ่าน ถ้า MAXBLK เท่ากับ 1 บล็อก 1 เพียงบล็อกเดียว

เท่านั้นที่ถูกส่งออกไปอย่างต่อเนื่อง ถ้า MAXBLK เท่ากับ 0 บล็อก 0 เท่านั้นที่จะถูกส่งออกไป (ซึ่งโดยปกติแล้วมันจะไม่ส่งบล็อก 0 ออกไป) ดังรูป



รูปที่ 3.5 แสดงตัวอย่างของความแตกต่างของการเซต MAXBLK

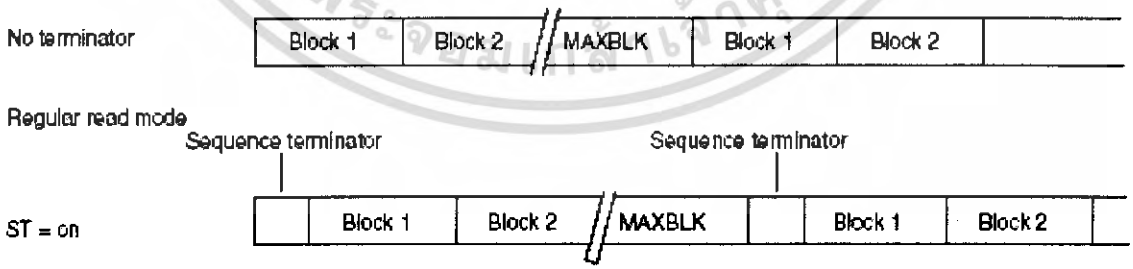
ทุกๆครั้งที่ T5557 เข้าสู่ regular-read mode หรือ block-read mode บิตแรกที่จะถูกส่งก็คือลอจิก 0 หลังจากนั้นก็จะตามด้วยข้อมูลเริ่มจาก block 1 bit 1 ไปเรื่อยๆ จนถึง MAXBLK bit 32 แล้วจะวนอย่างนี้ไปเรื่อยๆ

3.1.2.1.4 Block-read Mode

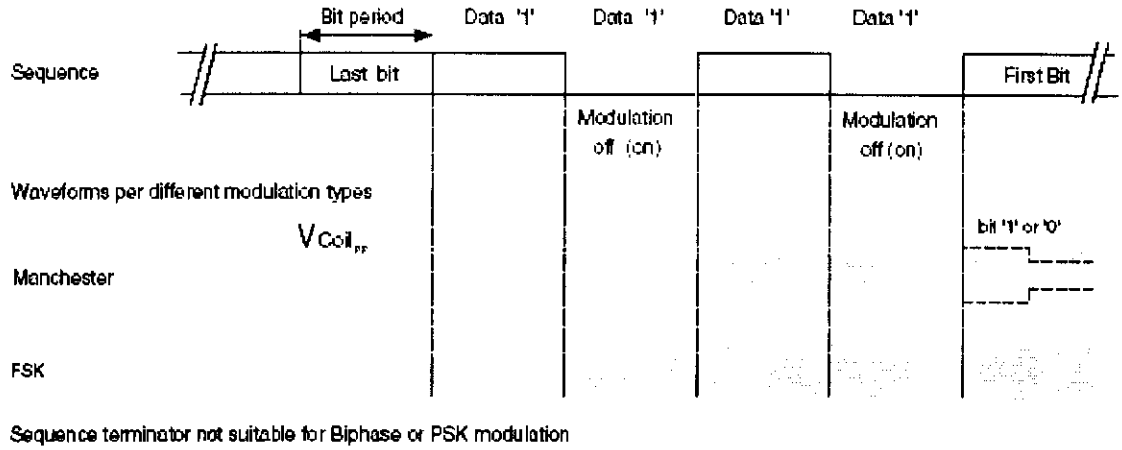
คือการสั่งให้เข้าถึงข้อมูลโดยตรง (direct access) ทำได้โดยการส่ง page access opcode แล้วตามด้วยเลขบล็อกที่เราต้องการอ่าน (3-bit block address)

3.1.2.1.5 e5550 Sequence Terminator (ST)

ใน regular-read mode ST จะมีการแทรกเข้าไปก่อนที่บล็อกแรกจะถูกส่งไป เราสามารถสั่งให้ ST ทำงานได้โดยการตั้งค่าที่บิต 29 ของบล็อก 0 ใน block-read mode (ในกรณีที่ MAXBLK เท่ากับ 0 หรือ 1) ST จะแทรกเข้าไปก่อนบล็อกที่เลือกให้ส่งแบบ direct access



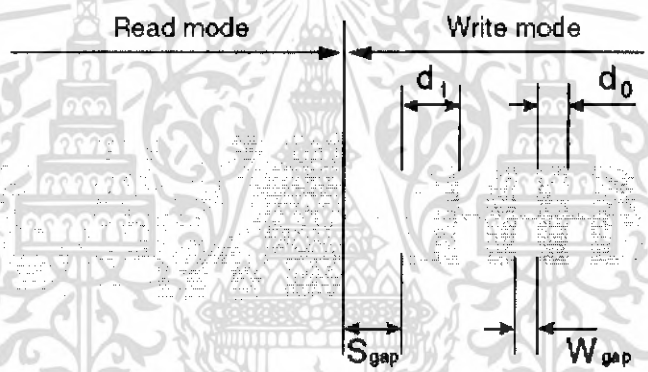
รูปที่ 3.6 แสดงRead Data Stream with Sequence Terminator



รูปที่ 3.7 แสดง e5550-compatible Sequence Terminator Waveforms

3.1.2.1.6 Reader to Tag Communication

เราสามารถเขียนข้อมูลไปที่ Tag ได้โดยการรบกวน RF field ด้วย field gaps สั้นๆ ดังรูปและตารางด้านล่าง



รูปที่ 3.8 แสดงการเริ่มต้นการสื่อสารของเครื่องอ่านสัญญาณต่อแทค

ตารางที่ 3.2 แสดงWrite Data Decoding Scheme

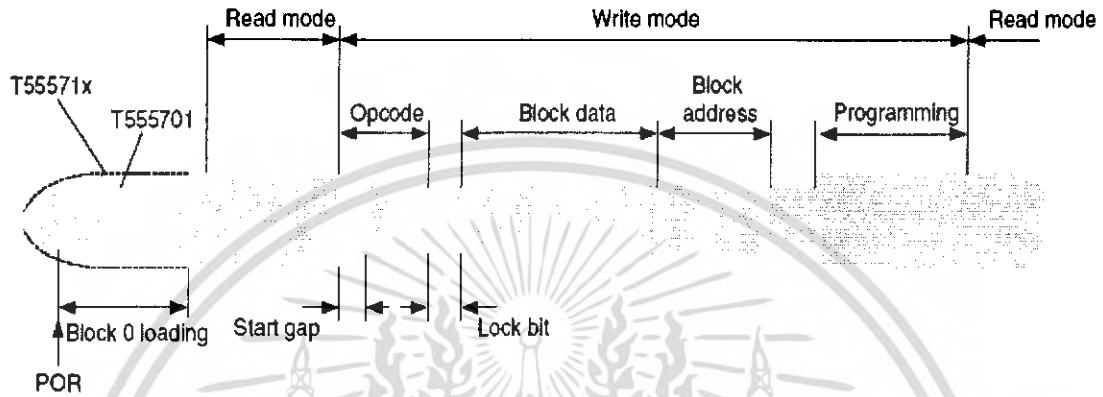
Parameters	Remark	Symbol	Min.	Max.	Unit
Start gap		S_{gap}	10	50	FC
Write gap	Normal write mode	W_{gap}	8	30	FC
Write data in normal mode	'0' data	d_0	16	31	FC
	'1' data	d_1	48	63	FC

จากรูปและตารางจะเห็นว่า เมื่อเราจะเขียนข้อมูลจะต้องเริ่มจาก Start gap เสมอซึ่งมีค่าประมาณ 30 FC ((10+50)/2) หลังจากนั้นก็จะตามด้วยข้อมูลที่ต้องการเขียน (ถ้าต้องการเขียน 1 ก็จะต้องใช้ประมาณ 54 FC ถ้าต้องการเขียน 0 ใช้ประมาณ 24 FC) หลังจากนั้นจะตามด้วย Write gap ซึ่งมีค่าประมาณ 18 FC อย่างนี้ไปเรื่อยๆ จนกระทั่งไม่มี gap มากกว่า 64 FC ก็จะออกจาก Write mode

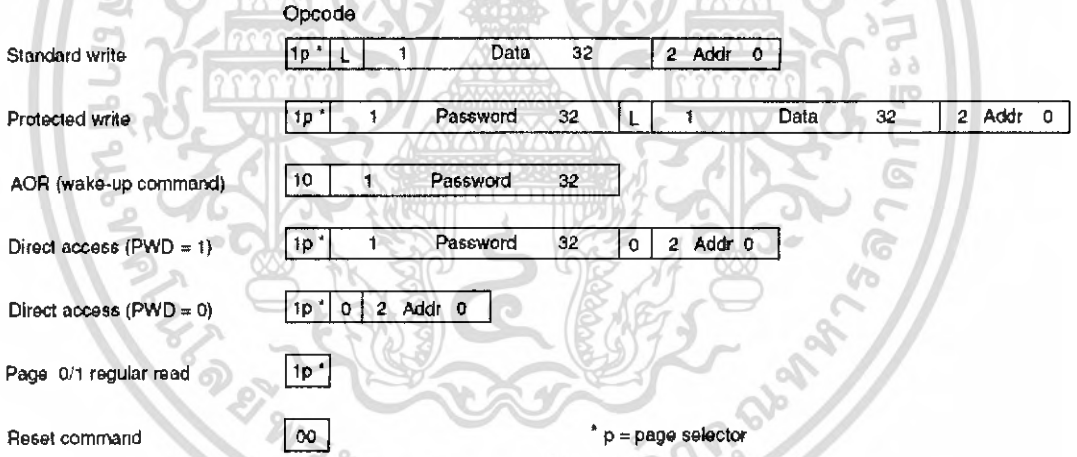
3.1.2.2 หลักในการเขียนข้อมูล

ในการเขียนทั่วไปต้องการ

- Opcode
- Lock Bit
- บิตข้อมูลอีก 32 บิต
- 3 Bit Address ใช้เลือกบล็อกที่จะเขียน



รูปที่ 3.9 แสดงรูปร่างสัญญาณขณะเขียนข้อมูล



รูปที่ 3.10 แสดงรูปแบบในการเขียน

3.1.2.3 PASSWORD

เมื่ออยู่ในโหมด password (PWD = 1) 32 บิตหลัง opcode จะถูกพิจารณาให้เป็น รหัส โดยที่รหัสจะถูกเปรียบเทียบแบบบิตต่อบิตโดยเริ่มตั้งแต่บิตที่ 1 ของ block 7 ถ้ารหัสไม่ถูกต้อง T5557 ก็จะไม่ถูกโปรแกรมลงไป แล้วจะเริ่มเข้าสู่ภาวะ regular-read mode หลังจากที่คำสั่งถูกส่งไปทันที

ในโหมด password ค่าของ MAXBLK เราควรจะตั้งค่าให้มันมีค่าน้อยกว่า 7 เพื่อป้องกัน T5557 ส่ง block ที่ 7 ออกไป

3.2 โปรโตคอลการติดต่อสื่อสาร

โปรโตคอลการติดต่อสื่อสารระหว่างแทคกับเครื่องอ่านสัญญาณ ซึ่งจะใช้สำหรับอ่านข้อมูลจากแทคหรือเขียนข้อมูลลงในแทค

3.2.1 คำสั่งการอ่านแทค

การอ่านข้อมูลจากแทคจะอ่านโดยระบุเซ็กเมนต์โดยเครื่องอ่านจะตัดข้อมูลที่อ่านจากแทคตามที่ระบุส่งข้อมูลกลับมาให้ตามตำแหน่งที่ระบุไว้

การส่งของ GUI

ตารางที่ 3.3 แสดงคำสั่งการส่งของ GUI ในการอ่านแทค

1 byte	1 byte	1 byte	1 byte	1 byte
STR	Command Read	Segment	CRC	END

ซึ่งในการตอบสนองของเครื่องอ่านสัญญาณกรณีไม่มีความผิดพลาดในการตรวจสอบ CRC หรือความผิดพลาดในการอ่านแทค

ตารางที่ 3.4 แสดงคำสั่งการตอบสนองของเครื่องอ่านกรณีไม่มีความผิดพลาดในการตรวจสอบ CRC

1 byte	1 byte	1 byte	28/4 bytes	1 byte	1 byte
STR	Command Read	Segment	DATA	CRC	END

แต่ในกรณีที่มีความผิดพลาดเกิดขึ้น เครื่องอ่านจะตอบสนองด้วยการส่ง

ตารางที่ 3.5 แสดงคำสั่งการตอบสนองของเครื่องอ่านกรณีมีความผิดพลาดในการตรวจสอบ CRC

1 byte	1 byte	1 byte	1 byte	1 byte
STR	Command Read	Status	CRC	END

หลังจากส่งเฟรมเพื่อบอกความผิดพลาดแล้วจะกลับไปทำงานในสถานะเริ่มต้น เพื่อรอคำสั่งใหม่

3.2.2 คำสั่งการเขียน

การส่งของ GUI

ตารางที่ 3.6 แสดงคำสั่งการส่งของ GUI ในการเขียนแทค

1 byte	1 byte	1 byte	4 bytes	1 byte	1 byte
STR	Command Write	Segment	DATA	CRC	END

ซึ่งในการตอบสนองของเครื่องอ่าน

ตารางที่ 3.7 แสดงคำสั่งการตอบสนองของเครื่องอ่าน

1 byte	1 byte	1 byte	1 byte	1 byte
STR	Command Write	Status	CRC	END

3.2.3 โค๊ดคำสั่ง

ตารางที่ 3.8 แสดงโค๊ตคำสั่ง

คำสั่ง	โค๊ต	ทิศทาง
อ่าน	0x02	แทค < -- > เครื่องอ่านสัญญาณ
เขียน	0x04	แทค < -- > เครื่องอ่านสัญญาณ

3.2.4 โค๊ตของเฮดเดอร์และสถานะ

ตารางที่ 3.9 แสดงโค๊ตของเฮดเดอร์และสถานะ

คำอธิบาย	โค๊ต
STR	0xFB
END	0xFC
CRC BAD	0x88
OK	0x99
Write not complete	0xA1

3.2.5 โค๊ดเซ็กเมนต์

ตารางที่ 3.10 แสดงโค๊ตเซ็กเมนต์

คำอธิบาย	โค๊ต
ตำแหน่งของข้อมูลบล็อค 0 หน้า 0	0x00
ตำแหน่งของข้อมูลบล็อค 1 หน้า 0	0x01
ตำแหน่งของข้อมูลบล็อค 2 หน้า 0	0x02
ตำแหน่งของข้อมูลบล็อค 3 หน้า 0	0x03
ตำแหน่งของข้อมูลบล็อค 4 หน้า 0	0x04
ตำแหน่งของข้อมูลบล็อค 5 หน้า 0	0x05
ตำแหน่งของข้อมูลบล็อค 6 หน้า 0	0x06
ตำแหน่งของข้อมูลบล็อค 7 หน้า 0	0x07
ตำแหน่งของข้อมูลบล็อค 1 หน้า 1	0x08
ตำแหน่งของข้อมูลบล็อค 2 หน้า 1	0x09

3.2.6 การหา CRC ของแทค

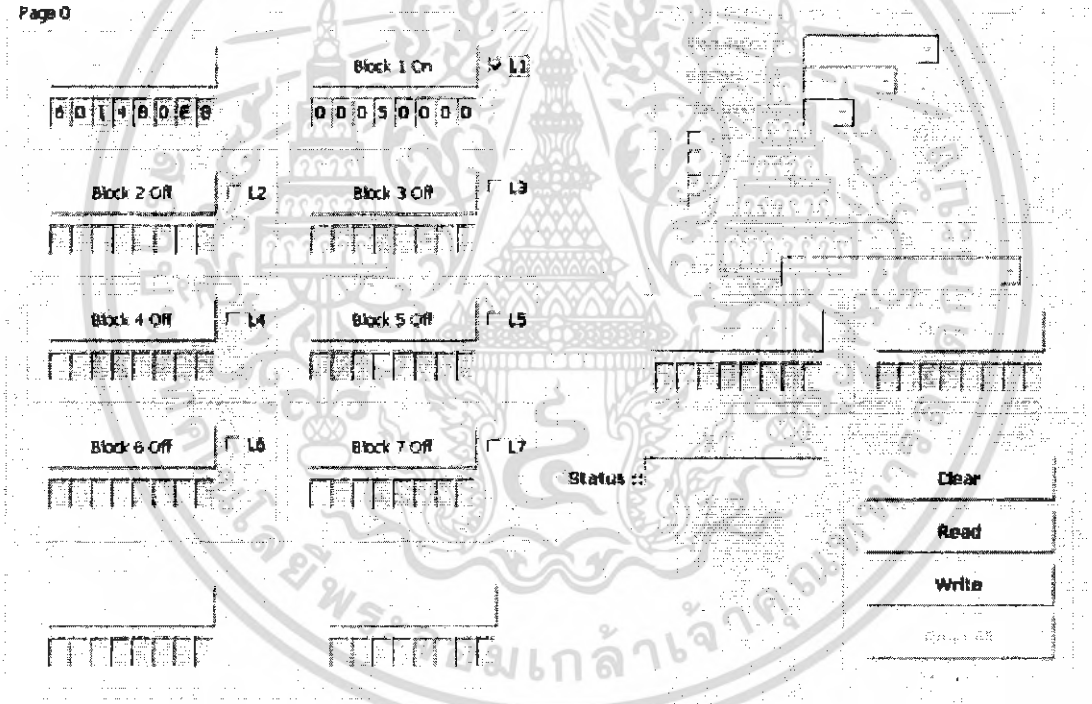
การหา CRC ของแทคสามารถหาได้โดย XOR ข้อมูล เริ่มตั้งแต่ STR จนถึงข้อมูลก่อนหน้า CRC

3.2.7 คุณสมบัติในการติดต่อสื่อสาร

- บอร์ดเรตในการติดต่อ 19.2 Kbps
- ช่วงเวลาในการติดต่อสื่อสาร 5 วินาที หลังจากช่วงเวลาในการติดต่อสื่อสารหมดแล้ว ให้กลับไปอยู่ในสถานะรอคำสั่งใหม่เสมอ
- การติดต่อกับแทคเป็นแบบต่อเนื่อง คือสามารถสั่งคำสั่งต่างกันได้ โดยไม่ต้องยกแทคออกระหว่างแต่ละคำสั่ง

3.2.8 การเขียนข้อมูลลงในแทค

การเขียนข้อมูลลงในแทคสามารถทำได้โดยดูจากโปรโตคอลด้านบน ซึ่งในโครงงานนี้ในการเขียนข้อมูลจะใช้บล็อค 1 ในการเขียนข้อมูลลงไป ซึ่งจะไม่ใช้บล็อค 0 เนื่องจากเป็นบล็อกที่ใช้เซตค่าต่างๆ เช่น Modulation, Bit rate, Max block โดยในการเขียนข้อมูลลงไปนั้น บล็อกแต่ละบล็อกจะแบ่งออกเป็น 8 ไบต์ แล้วแบ่งเป็นทีละ 2 ไบต์ เช่นดังในรูปต้องการเขียนข้อมูลเป็นจำนวนเงิน 500 เก็บเอาไว้ ก็จะแบ่งเป็น 00 05 00 00 ก็จะสามารถเขียนลงไปในแทคได้



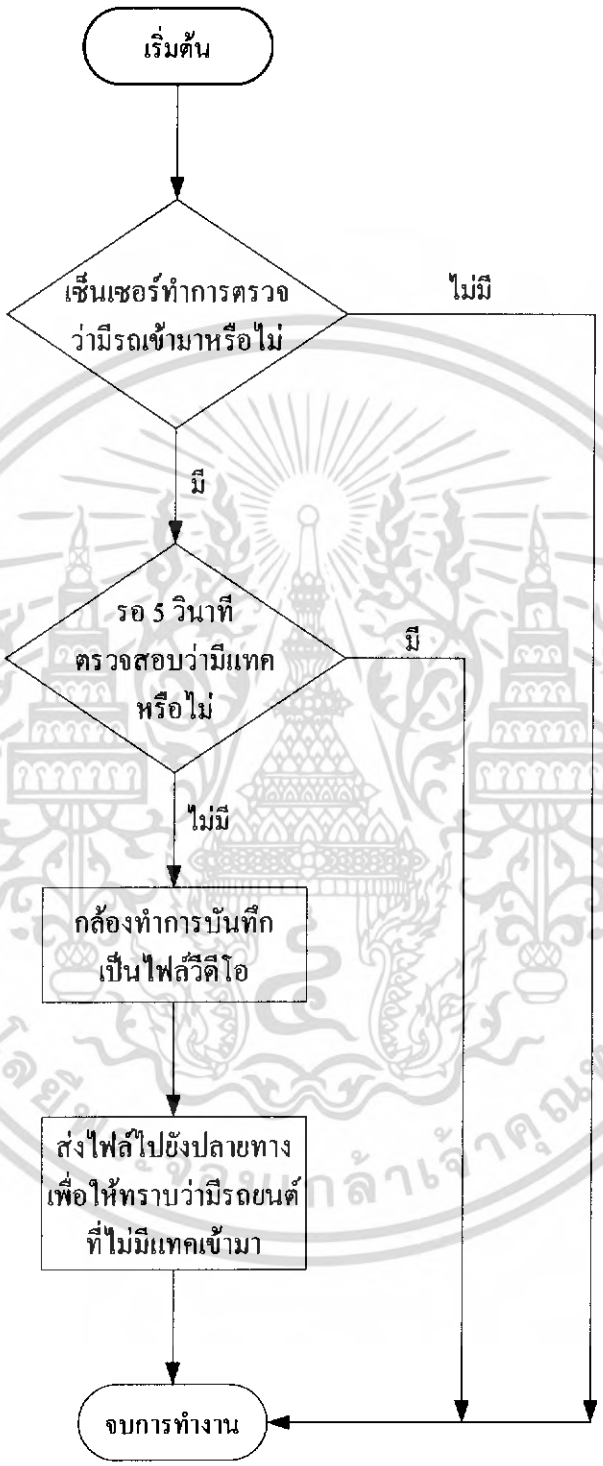
รูปที่ 3.11 แสดงการเขียนข้อมูลลงในแทค

3.3 พอร์ตสื่อสารแบบขนาน (Parallel Port)

จะนำไปต่อกับวงจรเซ็นเซอร์เพื่อรับสัญญาณที่ออกมาว่ามีวัตถุผ่านวงจรเซ็นเซอร์หรือไม่ ซึ่งในการใช้พอร์ตสื่อสารแบบขนานนี้ จะใช้ขาที่ 10 (ACK) ในการรับสัญญาณเมื่อมีการเปลี่ยนแปลงที่วงจรเซ็นเซอร์ โดยเมื่อไม่มีวัตถุผ่านจะสามารถวัดระดับแรงดันได้ 0 โวลต์ แต่เมื่อมีวัตถุผ่านจะสามารถวัดได้ประมาณ 5 โวลต์ แล้วจะใช้โปรแกรมเดลไฟล์ เพื่อนำสัญญาณไปใช้ในการควบคุมการบันทึกของกล้องต่อไป

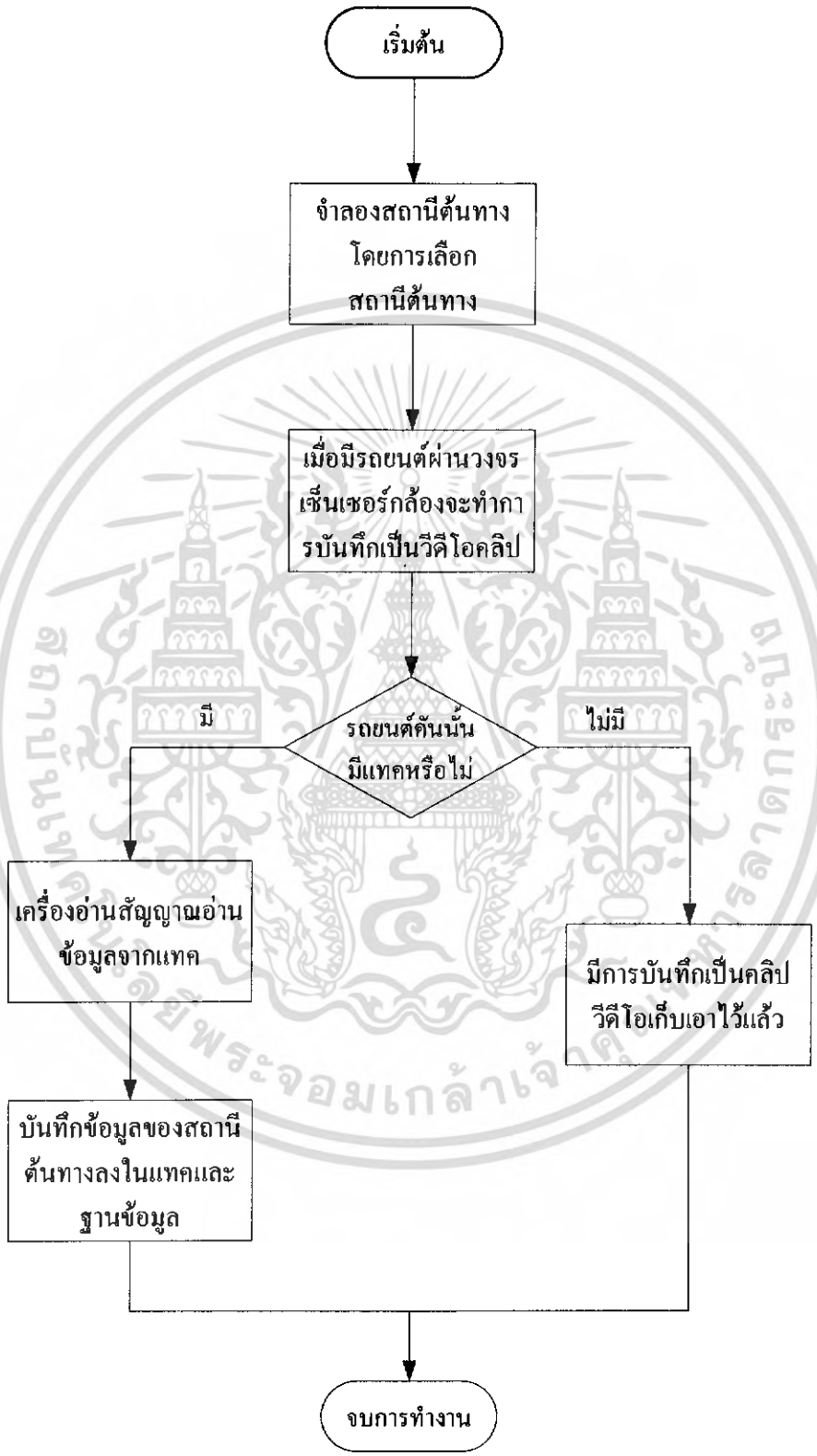
3.4 โฟล์วชาร์ดแสดงการทำงานต่างๆ

3.4.1 โฟล์วชาร์ดแสดงการทำงานของระบบ



รูปที่ 3.12 แสดงโฟล์วชาร์ดการทำงานของระบบ

3.4.2 โฟล์วชาร์ตแสดงการทำงานของสถานีต้นทาง

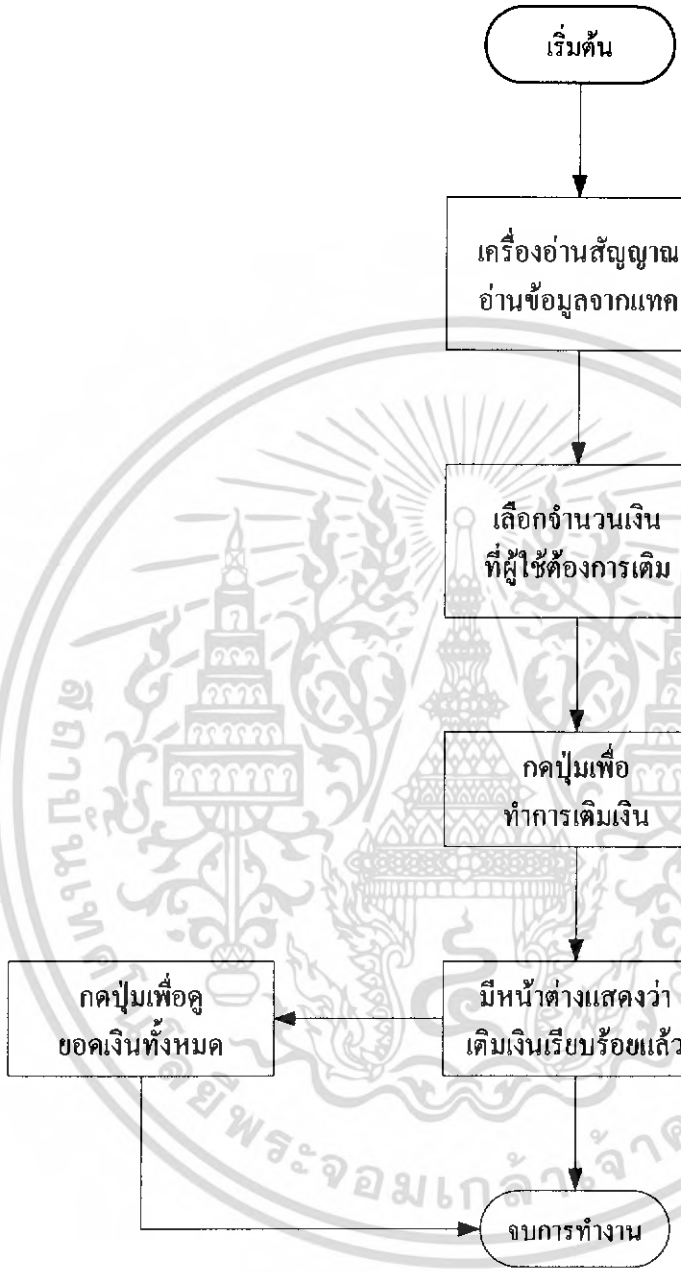


รูปที่ 3.13 แสดงโฟล์วชาร์ตการทำงานของสถานีต้นทาง

3.4.3 โฟล์วชาร์ดแสดงการทำงานของสถานีปลายทาง

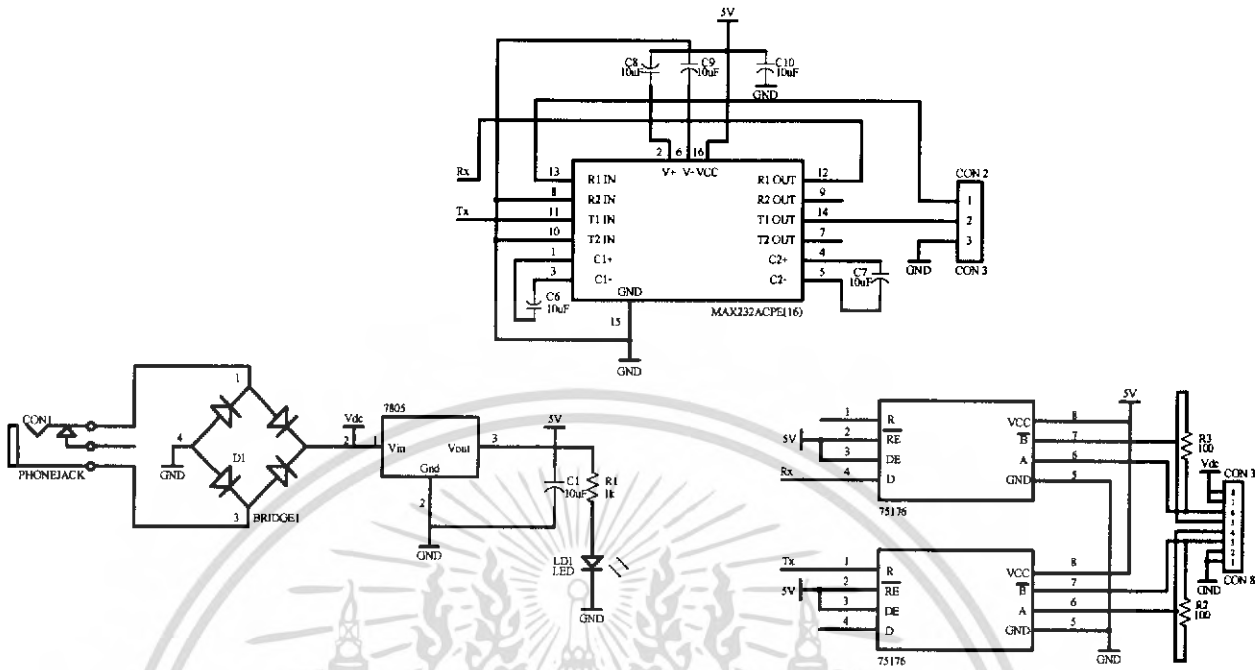
รูปที่ 3.14 แสดงโฟล์วชาร์ดการทำงานของสถานีปลายทาง

3.4.4 โฟล์วชาร์ตแสดงการทำงานของการเติมเงิน



รูปที่ 3.15 แสดงโฟล์วชาร์ตการทำงานของการเติมเงิน

3.5 วงจรที่ใช้ในการติดต่อกับคอมพิวเตอร์



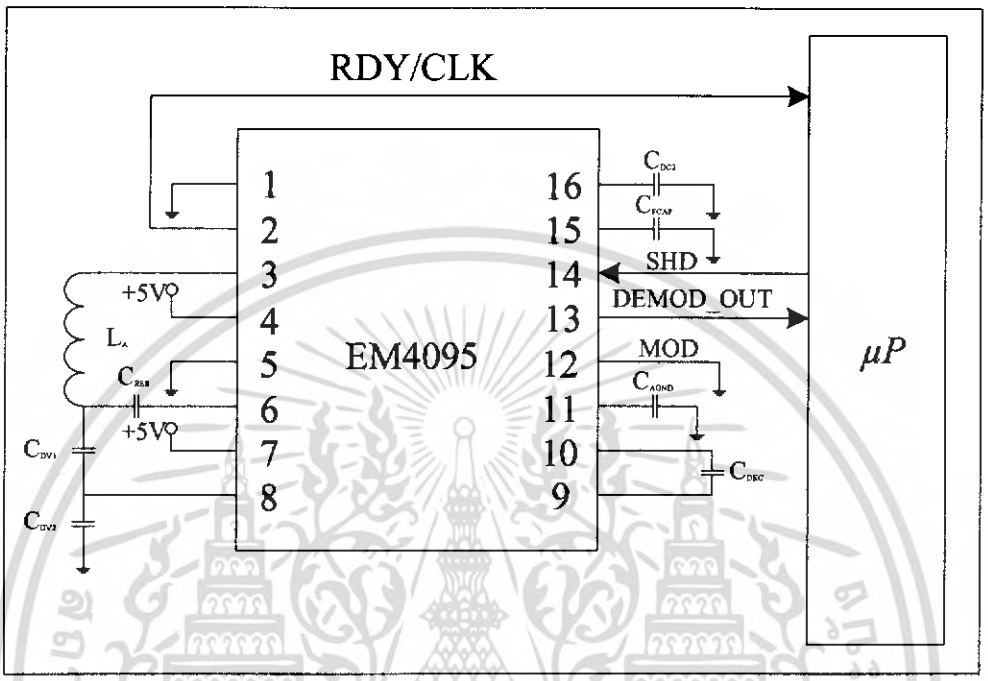
รูปที่ 3.16 แสดงรูปวงจรที่ใช้ในการติดต่อกับคอมพิวเตอร์

ขั้นตอนการทำงาน เริ่มจากเครื่องอ่านสัญญาณเมื่อจะติดต่อกับแทค สัญญาณที่ส่งจะถูกกำหนดให้ส่งสัญญาณเป็นผ่านมาตรฐาน RS-485 เนื่องจากต้องการสื่อสารให้มีระยะไกลขึ้นโดยไม่มีสัญญาณรบกวน เมื่อเข้าสู่บอร์ดจะเข้าสู่ไอซี SN65176BP เพื่อแปลงสัญญาณเป็นสัญญาณที่แท้จริงแล้วจะเข้าสู่ไอซี MAX232ACPE เพื่อจะแปลงสัญญาณเป็นมาตรฐาน RS-232 เพื่อจะติดต่อกับเครื่องคอมพิวเตอร์ ส่วนบิทไดโอดจะทำหน้าที่ป้องกันการจ่ายไฟผิดพลาด เราจะใช้สายแลนในการสื่อสารเพราะสะดวกต่อการเชื่อมต่อโดยภายในจะประกอบด้วยสาย 8 เส้น โดยข้อมูลจะใช้ 2 เส้น (แต่ถ้าเป็นมาตรฐาน RS-422 จะใช้ 4 เส้นเพราะเป็นแบบ Full Duplex ส่วนมาตรฐาน RS-485 จะเป็นแบบ Half Duplex) การต่อโวลต์เตจกับการต่อกราวด์จะใช้อย่างละ 2 เส้น

บทที่ 4

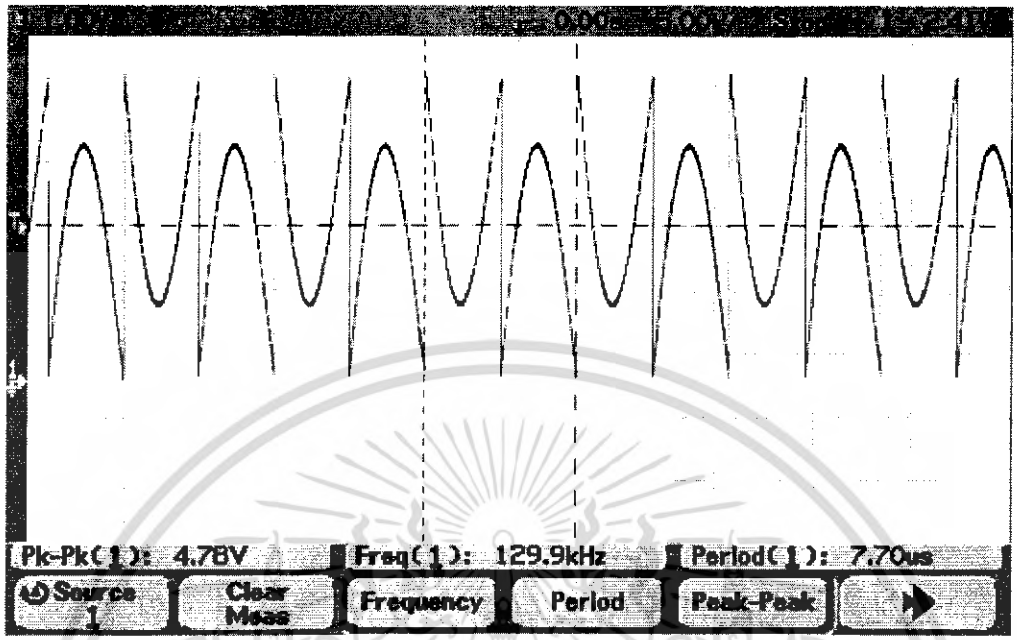
การทดลองและผลการทดลอง

4.1 ไอซี EM 4095

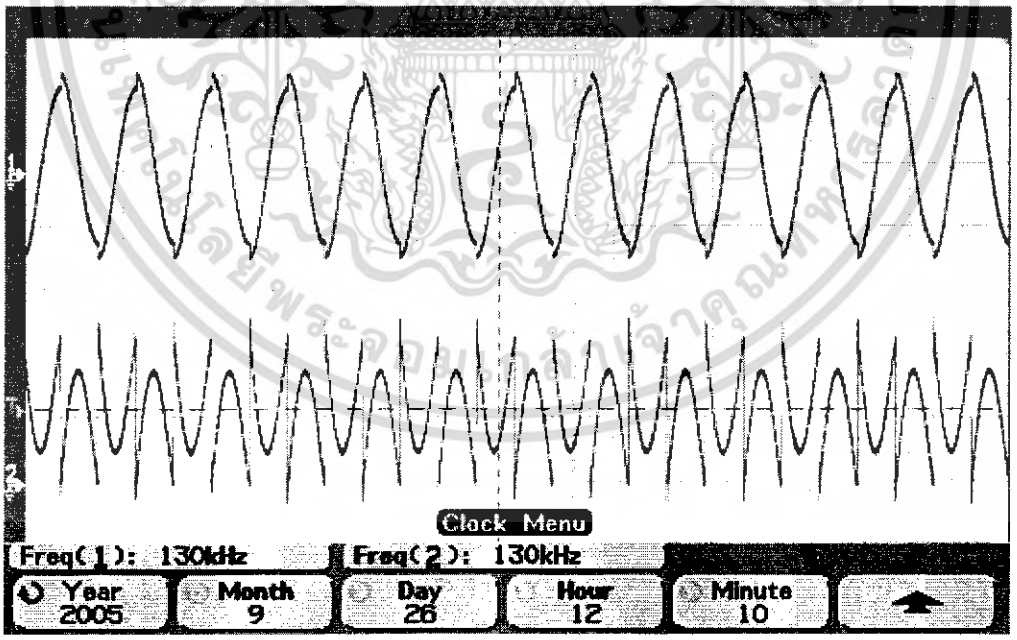


รูปที่ 4.1 แสดงวงจรของไอซี EM4095

4.2 ผลการวัดความถี่และขนาดของสัญญาณที่วัดจากสายอากาศ จากวงจรไอซี EM4095 เมื่อไม่มีแทกอยู่ในระยะที่สามารถอ่านได้



รูปที่ 4.2 แสดงความถี่และขนาดของสัญญาณที่ออกจากสายอากาศเมื่อไม่มีแทค
ซึ่งจะเห็นว่าความถี่ของสัญญาณที่ออกมาจะมีความถี่ 129.9 *kHz* และมีแอมปลิจูด 4.78 *V*

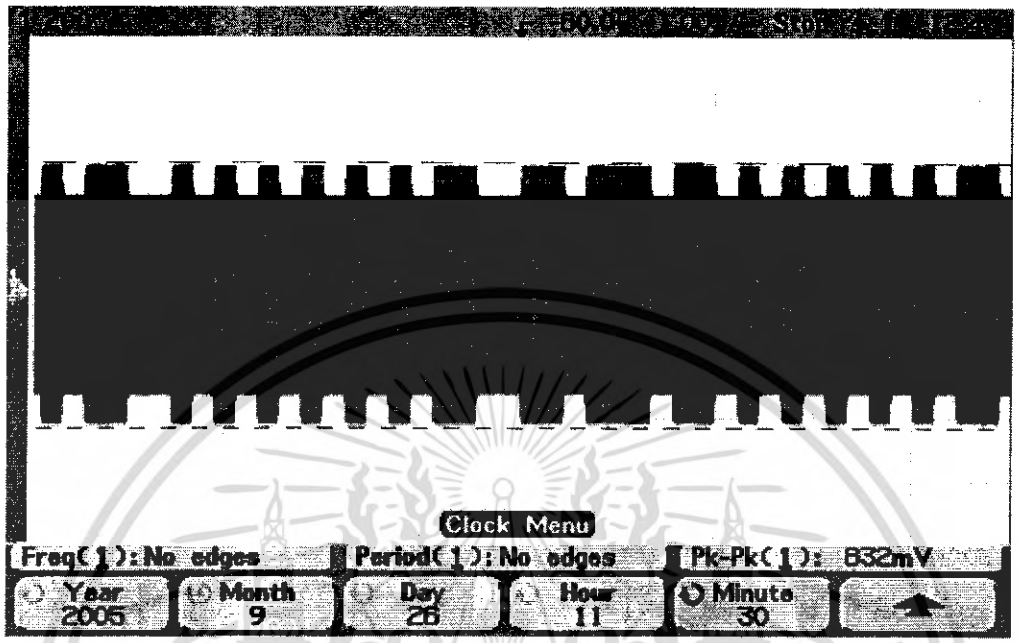


รูปที่ 4.3 แสดงการเปรียบเทียบความถี่ของสัญญาณที่ออกจากสายอากาศขาที่ 1 และขาที่ 2

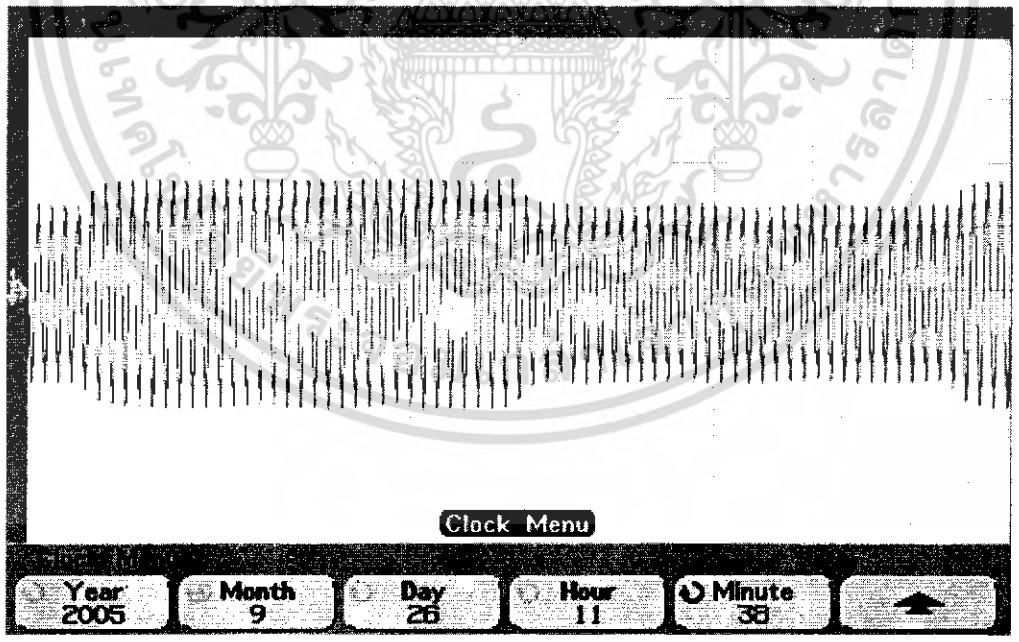
ซึ่งในการวัดสายอากาศขาที่ 1 จะวัดได้จากขาที่ 3 ของไอซี EM 4095 และในการวัดสายอากาศขาที่ 2 วัดได้จากขาที่ 6 ของไอซี EM 4095

4.3 ผลการวัดความถี่และขนาดของสัญญาณที่วัดจากสายอากาศขาที่ 1 จากวงจรไอซี EM4095 เมื่อมีแทค

ซึ่งในการวัดสายอากาศขาที่ 1 จะวัดได้จากขาที่ 3 ของไอซี EM 4095

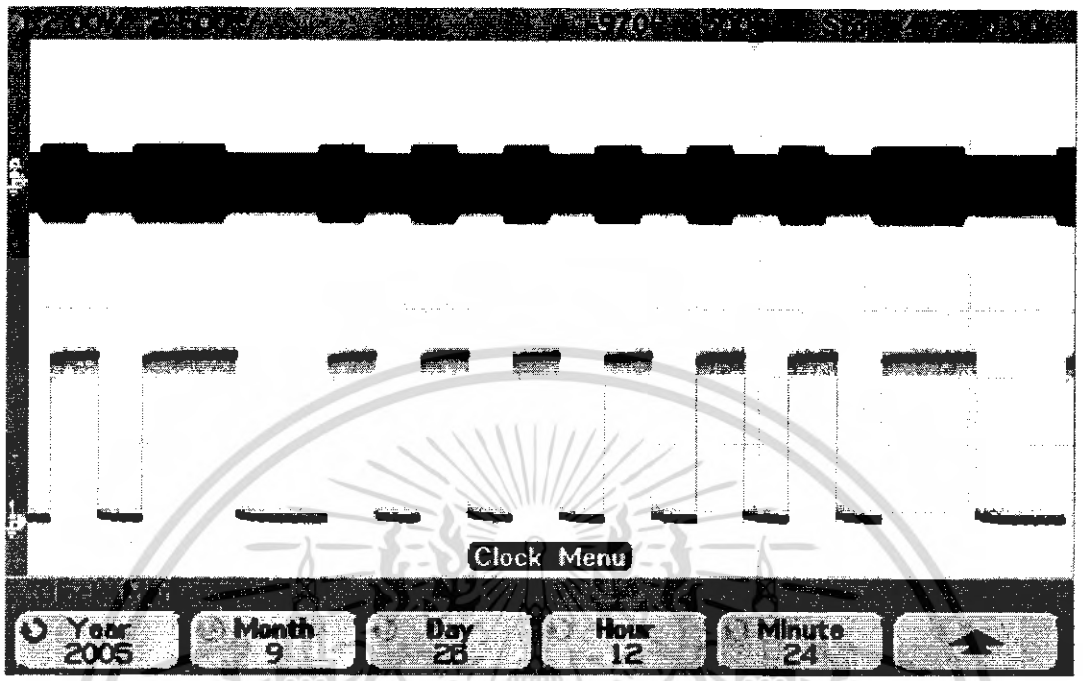


รูปที่ 4.4 แสดงสัญญาณที่ออกจากสายอากาศที่ 1 เมื่อมีแทคอยู่ในระยะที่อ่านได้



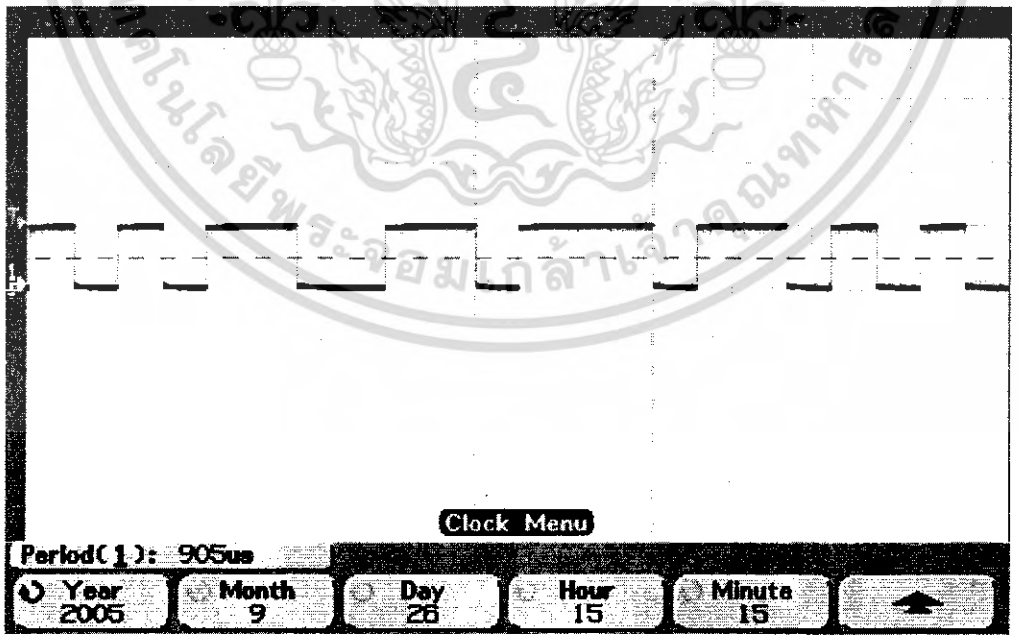
รูปที่ 4.5 แสดงรูปการขยายสัญญาณที่ออกจากสายอากาศ

4.4 ผลการวัดสัญญาณของสายอากาศขาที่ 1 ที่รับเข้ามาจากแทคและผ่านวงจรของไอซี EM4095 ขณะมีแทคอยู่ในระยะที่อ่านได้

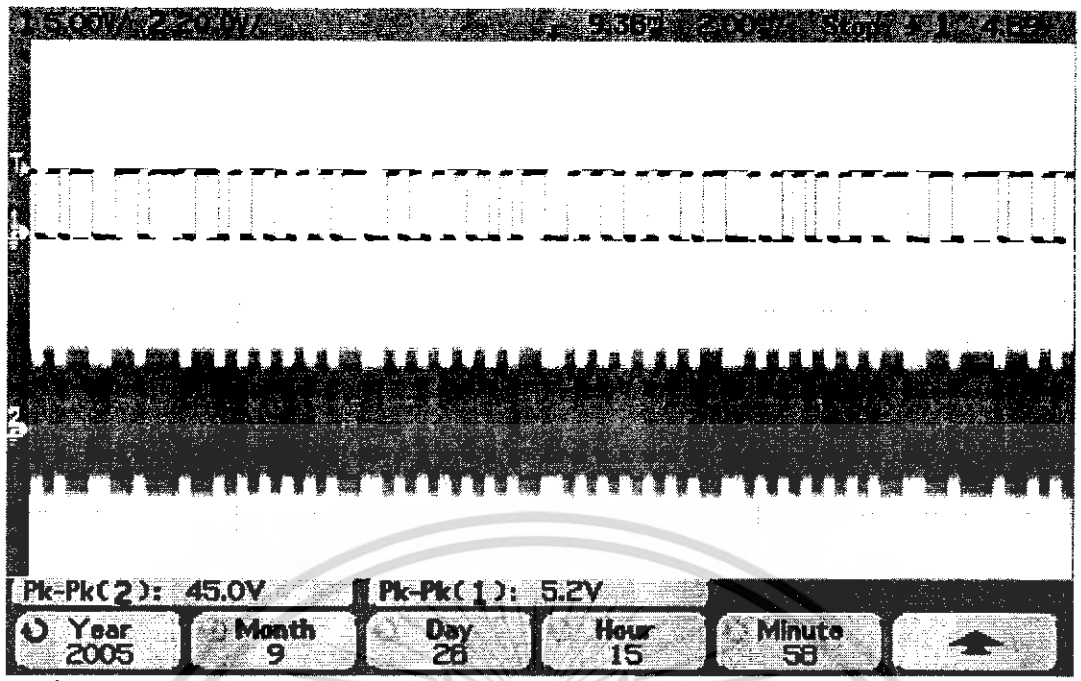


รูปที่ 4.6 แสดงสัญญาณที่เข้ามาที่สายอากาศขาที่ 1 และผ่านไอซี EM 4095 ออกมาทางขา DEMOD_OUT

4.5 การวัดสัญญาณส่วนหัวของข้อมูล

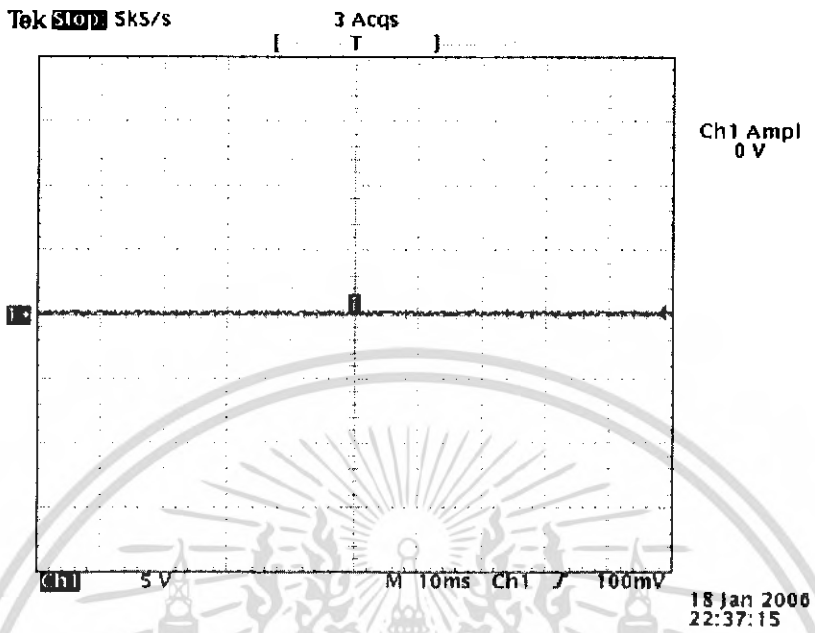


รูปที่ 4.7 แสดงส่วนของเฮดเดอร์ที่วัดจากสายอากาศขาที่ 1

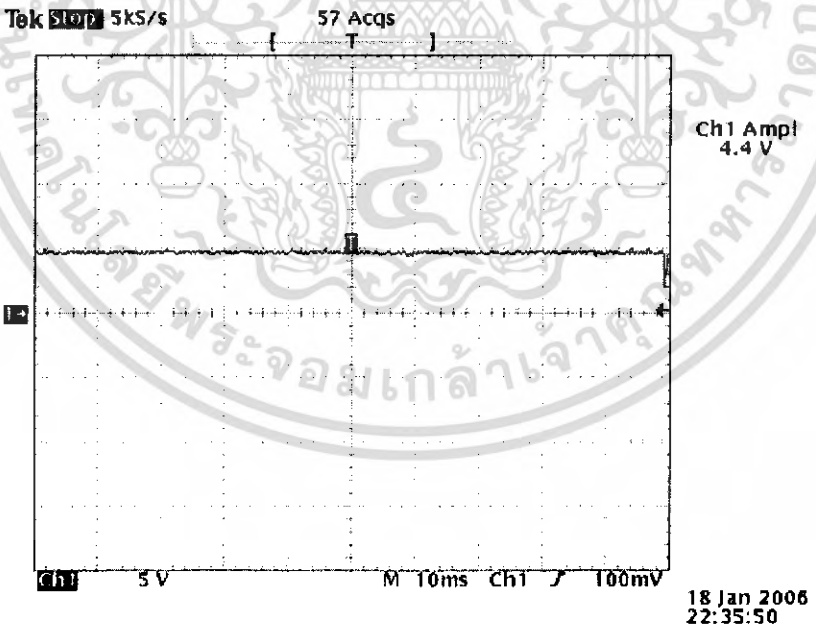


รูปที่ 4.8 แสดงการเปรียบเทียบระหว่างเฮดเดอร์ของสัญญาณที่ได้จากสายอากาศขาที่ 1 กับสัญญาณที่ออกมาจากไอซี EM 4095

4.6 ผลการวัดขนาดของแรงดันที่วัดจากวงจรเซ็นเซอร์

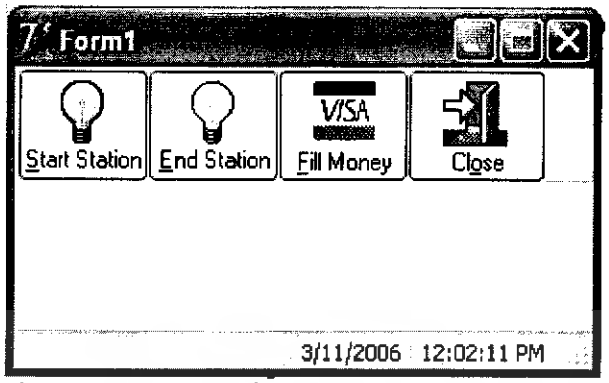


รูปที่ 4.9 แสดงขนาดของแรงดันเมื่อไม่มีวัตถุอยู่ระหว่างตัวส่งและตัวรับของเซ็นเซอร์



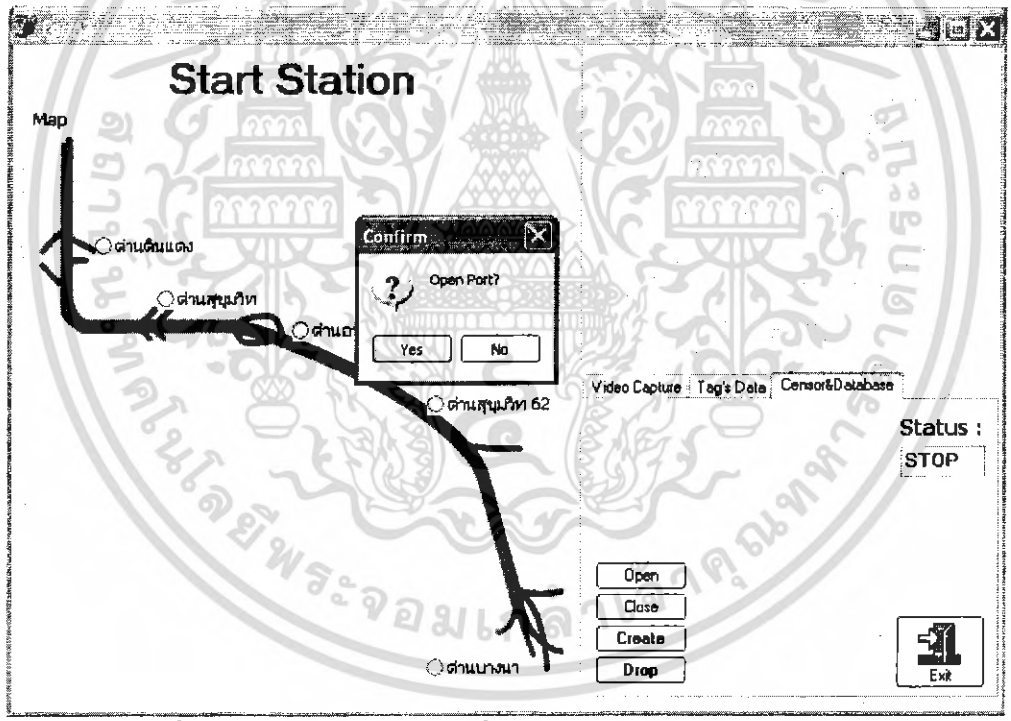
รูปที่ 4.10 แสดงขนาดของแรงดันเมื่อมีวัตถุอยู่ระหว่างตัวส่งและตัวรับของเซ็นเซอร์

4.7 แสดงรูปหน้าต่างเพื่อเลือกว่าเป็นหน้าต่างสถานีต้นทาง สถานีปลายทาง เติมเงิน หรือออกจากระบบ



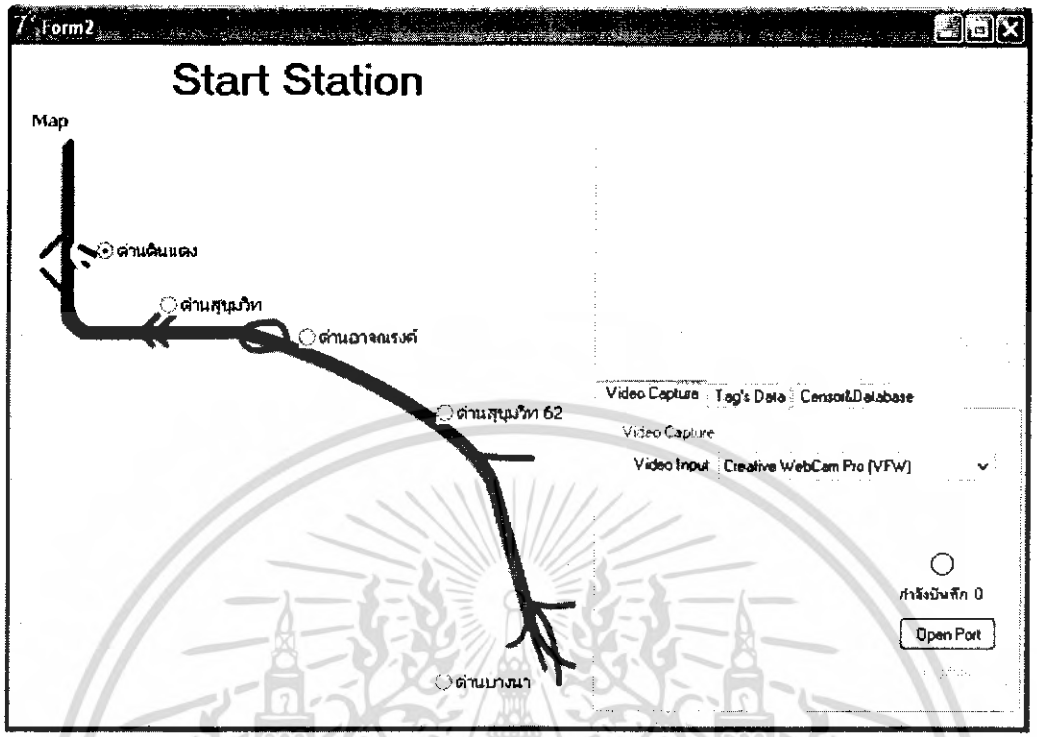
รูปที่ 4.11 แสดงหน้าต่างเพื่อเลือกว่าเป็นหน้าต่างสถานีต้นทาง
สถานีปลายทาง เติมเงิน หรือออกจากระบบ

4.8 แสดงรูปหน้าต่างโปรแกรมสถานีต้นทาง



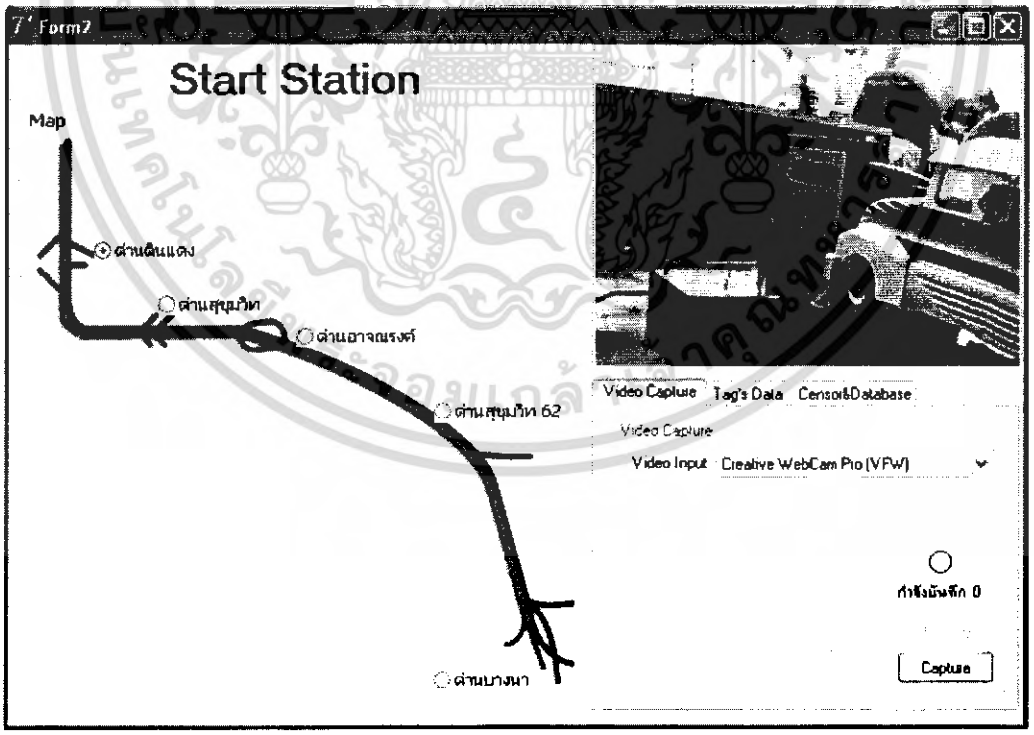
รูปที่ 4.12 แสดงการ Open Port เมื่อเข้าสู่หน้าต่างทางด้านสถานีต้นทาง

เมื่อเริ่มการทำงานต้องทำการเลือกสถานีต้นทาง เพื่อจำลองสถานีว่าสถานีต้นทางคือสถานีไหน โดยเลือกสถานีจากแผนที่



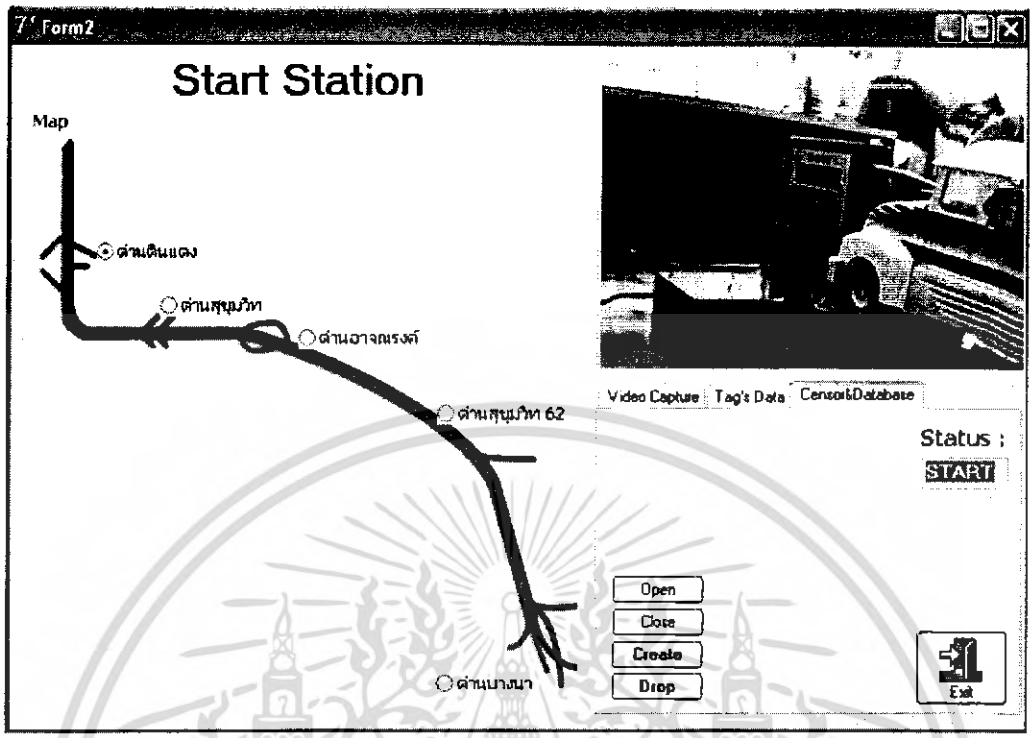
รูปที่ 4.13 แสดงการเลือกสถานีเพื่อจำลองเป็นสถานีต้นทาง

ทำการ Open Port หลังจากที่เลือกสถานีต้นทางแล้ว เพื่อบันทึกวีดีโอคลิปรถยนต์ที่วิ่งเข้ามา



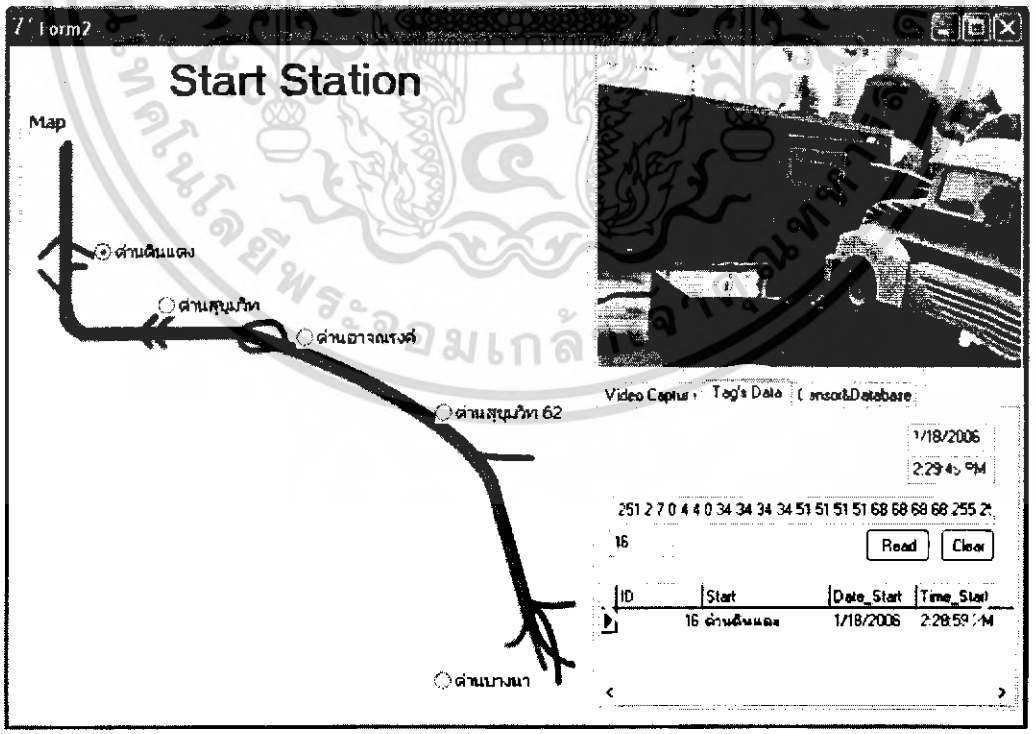
รูปที่ 4.14 แสดงรูปเมื่อทำการ Open Port แล้ว

เมื่อรถยนต์ที่เข้ามาผ่านเซ็นเซอร์จะทำให้สถานะของเซ็นเซอร์เปลี่ยนจาก STOP เป็น START และทำการบันทึกเป็นวีดีโอคลิปเก็บเอาไว้



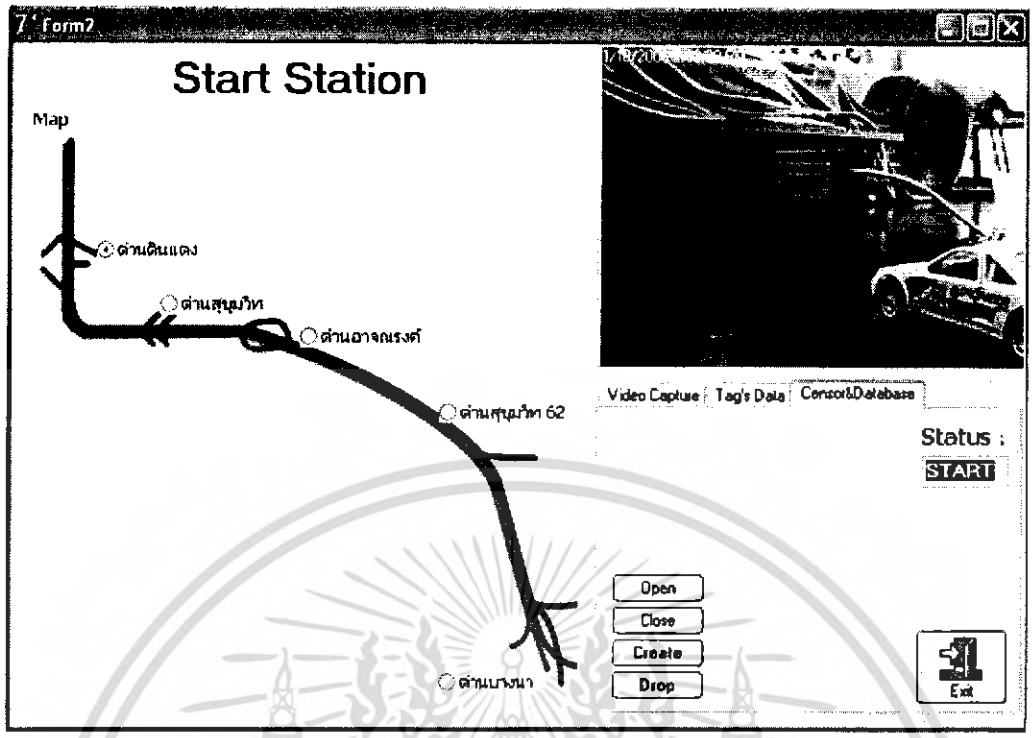
รูปที่ 4.15 แสดงการทำงานของเซ็นเซอร์เมื่อมีรถยนต์ผ่าน

เมื่อรถยนต์ที่มีแทคผ่านไปถึงเครื่องอ่านสัญญาณ เครื่องจะทำการอ่าน แสดงรายละเอียดที่หน้าของโปรแกรมและบันทึกรายละเอียดลงในฐานข้อมูล



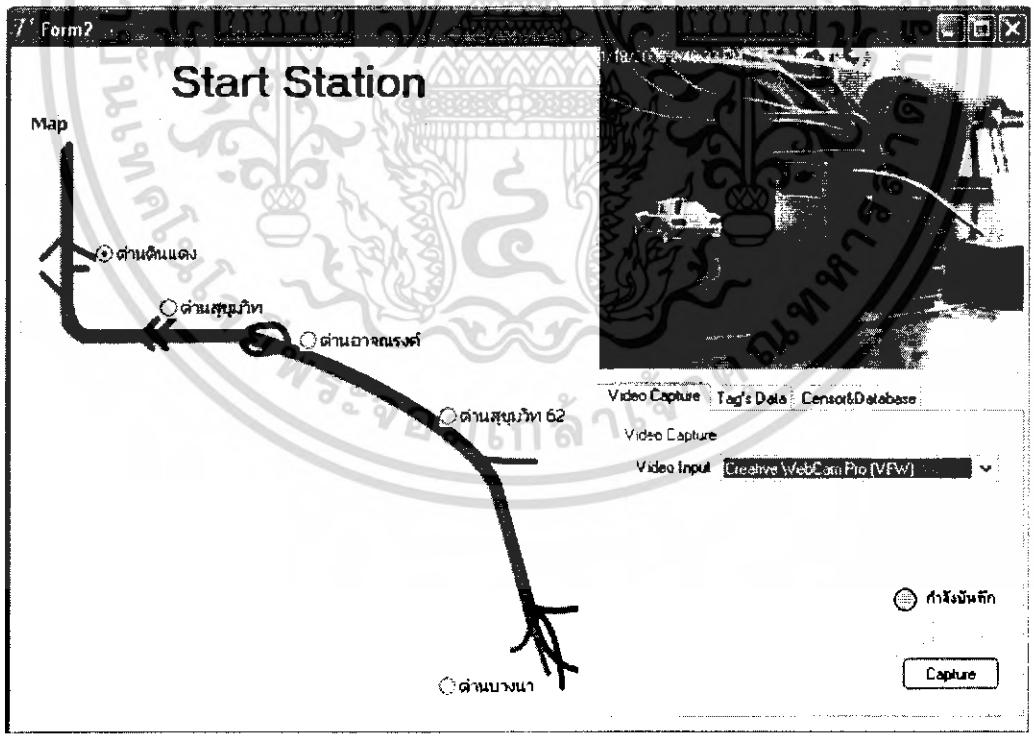
รูปที่ 4.16 แสดงรายละเอียดที่หน้าของโปรแกรม เมื่อรถยนต์ที่มีแทค ผ่าน

เมื่อรถยนต์ที่ไม่มีแทควิ่งเข้ามา



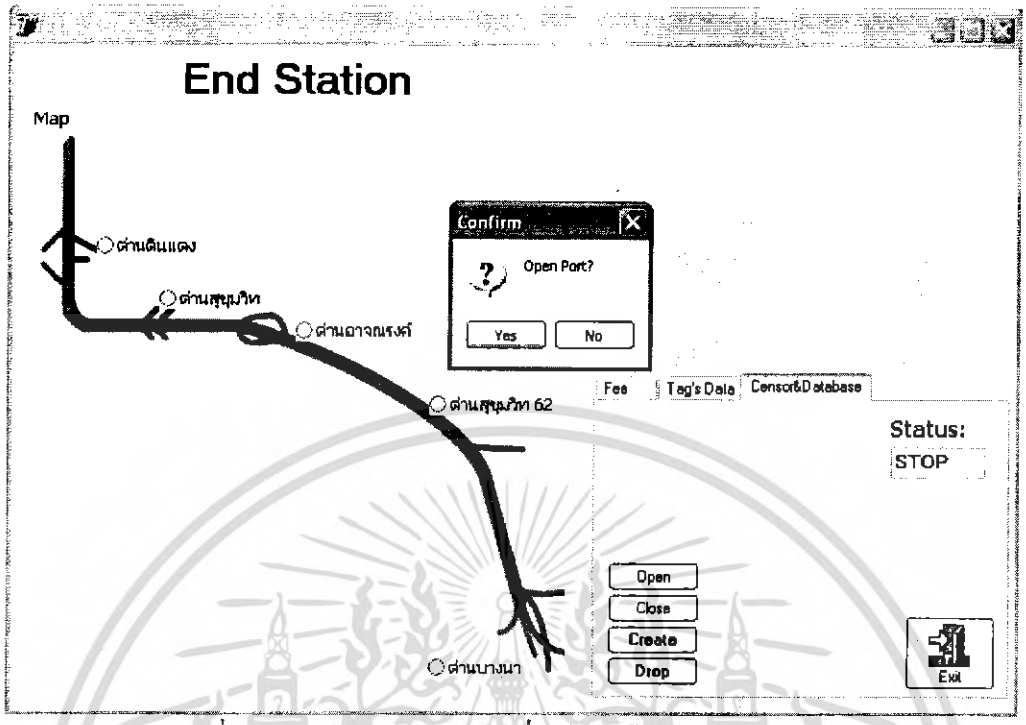
รูปที่ 4.17 แสดงรูปเมื่อรถยนต์ที่ไม่มีแทควิ่งเข้ามา

เมื่อรถยนต์ที่ไม่มีแทควิ่งเข้ามา เมื่อผ่านเซนเซอร์ก็จะทำการบันทึกวีดีโอคลิปไว้



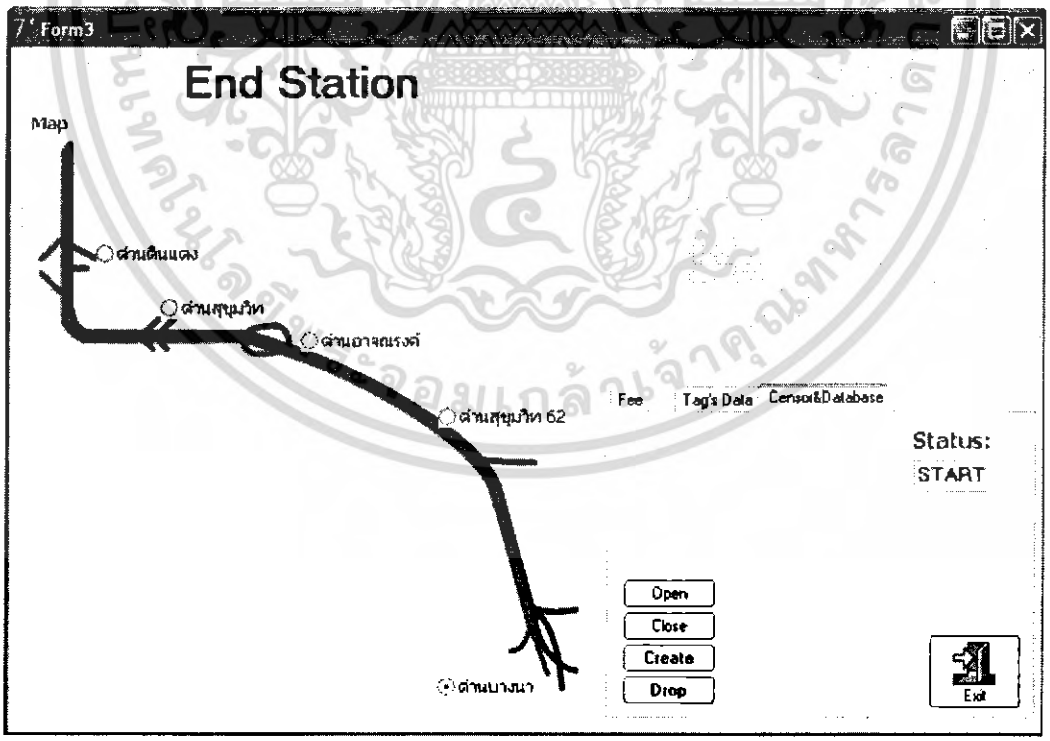
รูปที่ 4.18 แสดงรูปหน้าต่างของโปรแกรมเมื่อมีการบันทึกวีดีโอคลิป

4.9 แสดงรูปหน้าต่างโปรแกรมสถานีปลายทาง



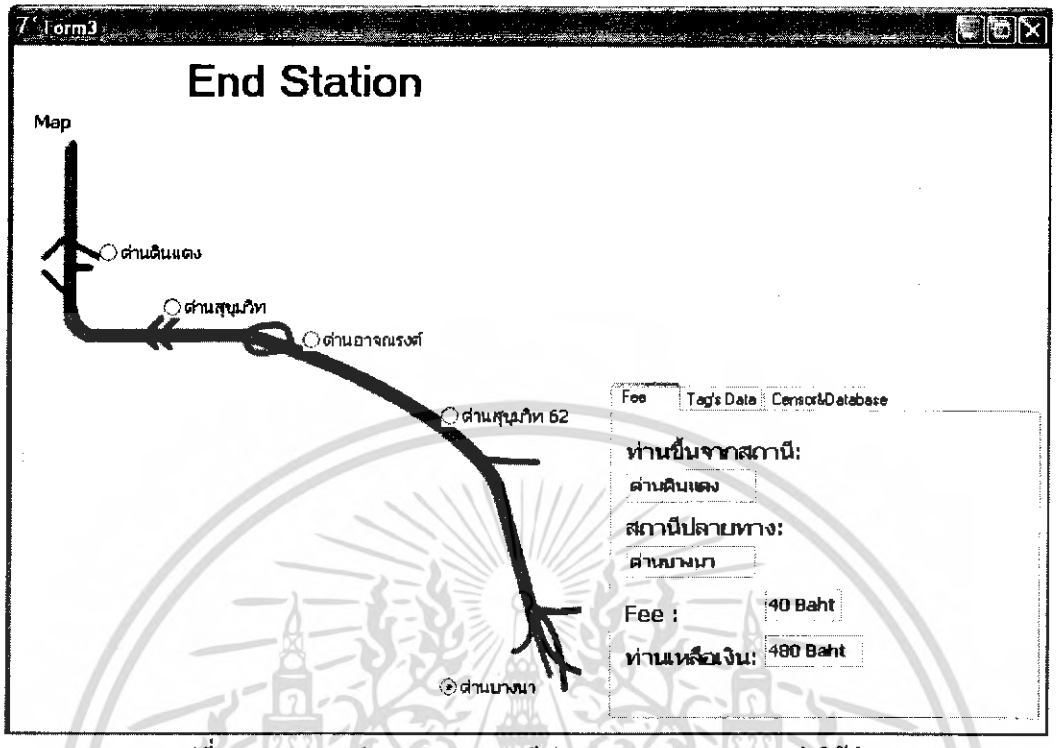
รูปที่ 4.19 แสดงการ Open Port เมื่อเข้าสู่หน้าต่างทางด้านสถานีปลายทาง

เมื่อรถยนต์ที่เข้ามาผ่านเซ็นเซอร์จะทำให้สถานะของเซ็นเซอร์เปลี่ยนจาก STOP เป็น START



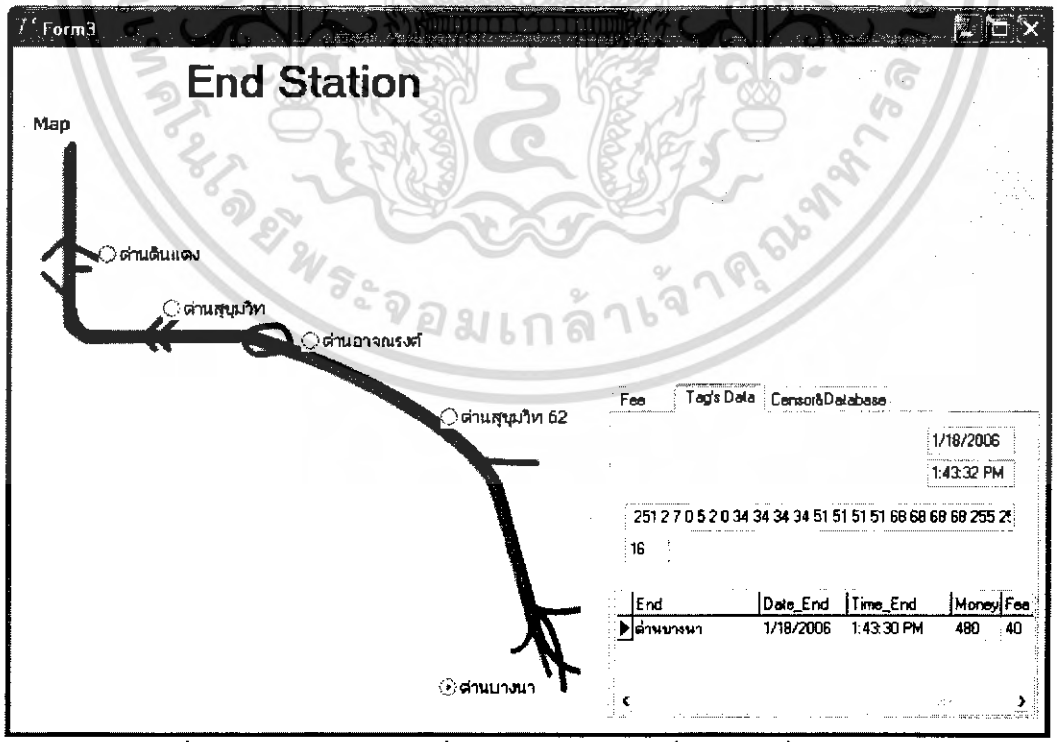
รูปที่ 4.20 แสดงการทำงานของเซ็นเซอร์เมื่อมีรถยนต์ผ่าน

เมื่อรถยนต์มาถึงสถานีปลายทาง เราก็จะทำการจำลองว่าเป็นสถานีปลายทางอะไร โดยการเลือกสถานีปลายทางในแผนที่ เมื่อแทคอยู่ในระยะที่สามารถอ่านได้ โปรแกรมจะคำนวณค่าใช้จ่ายออกมา



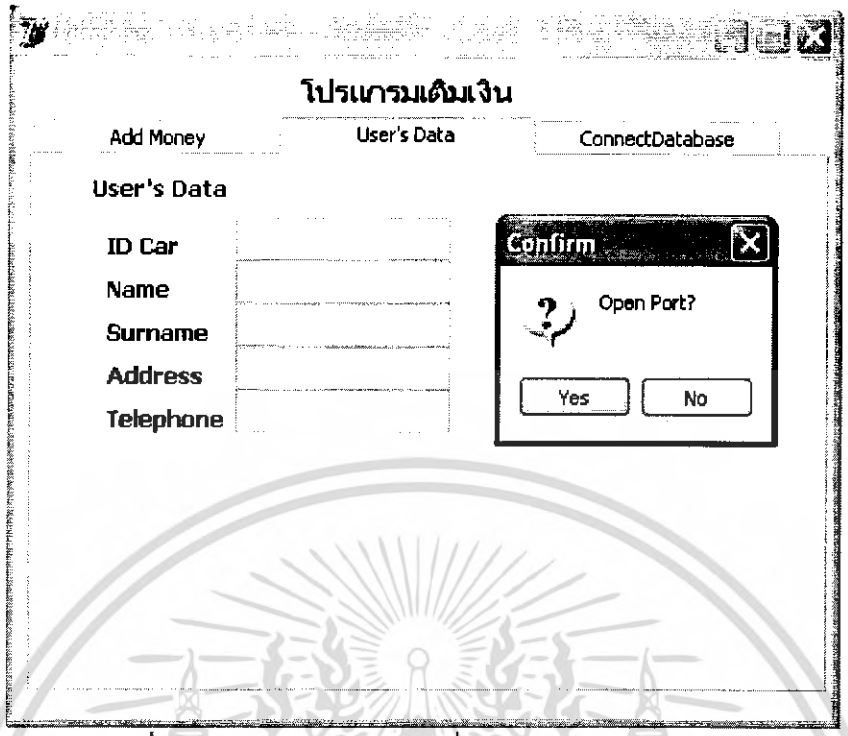
รูปที่ 4.21 แสดงรูปการจำลองสถานีปลายทางและการแสดงค่าใช้จ่าย

เมื่อรถยนต์ที่มีแทคผ่านไปถึงเครื่องอ่านสัญญาณ เครื่องจะทำการอ่าน แสดงรายละเอียดที่หน้าของโปรแกรมและบันทึกรายละเอียดลงในฐานข้อมูล



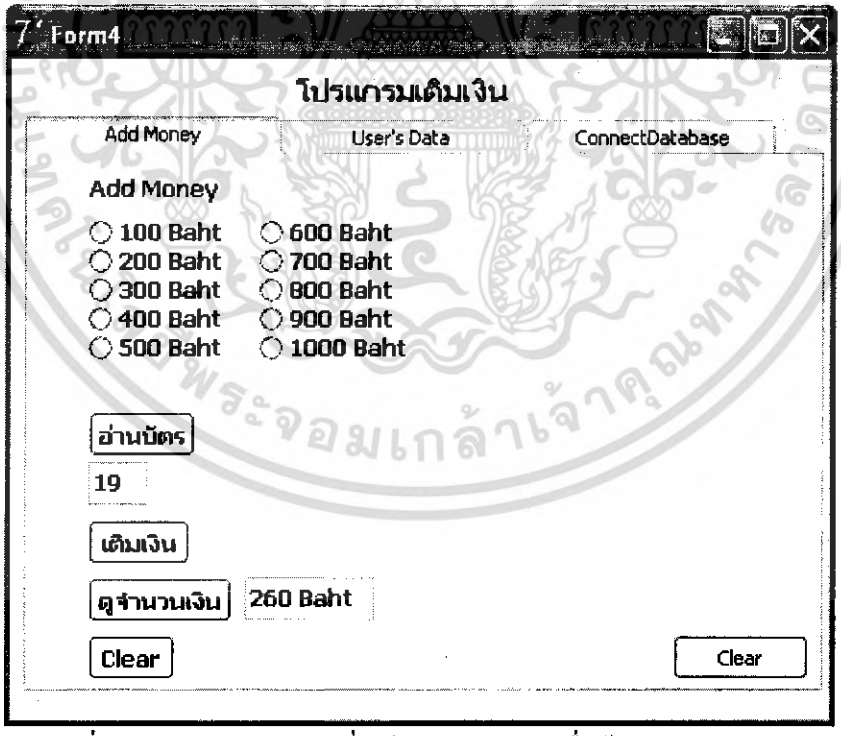
รูปที่ 4.22 แสดงรายละเอียดที่หน้าของโปรแกรม เมื่อรถยนต์ที่มีแทค ผ่าน

4.10 แสดงรูปหน้าต่างเติมเงิน



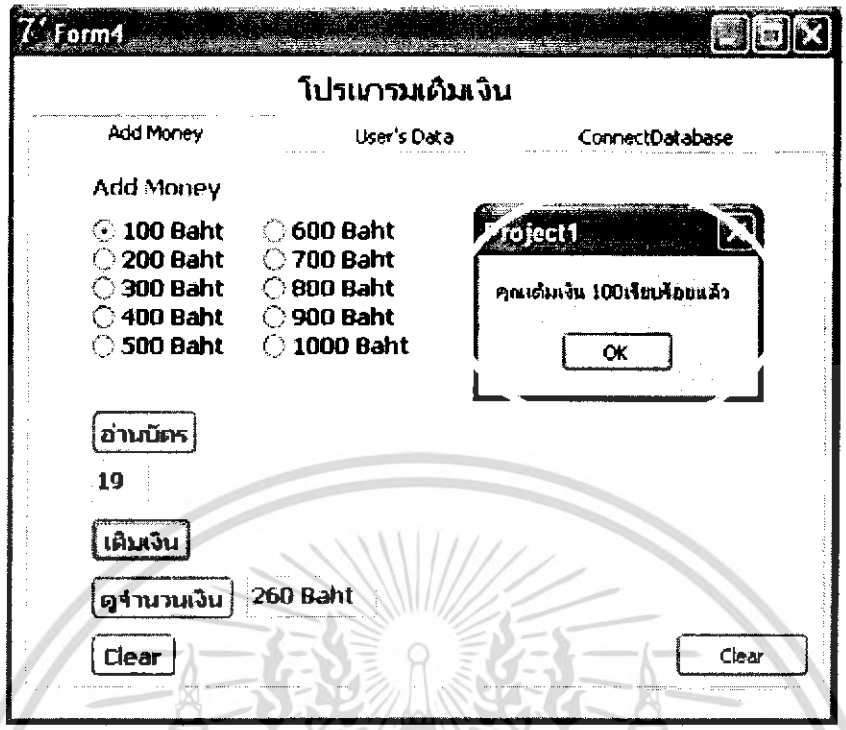
รูปที่ 4.23 แสดงการ Open Port เมื่อเข้าสู่หน้าต่างการเติมเงิน

การเติมเงินในขั้นแรกจะทำการอ่านข้อมูลจากแทค



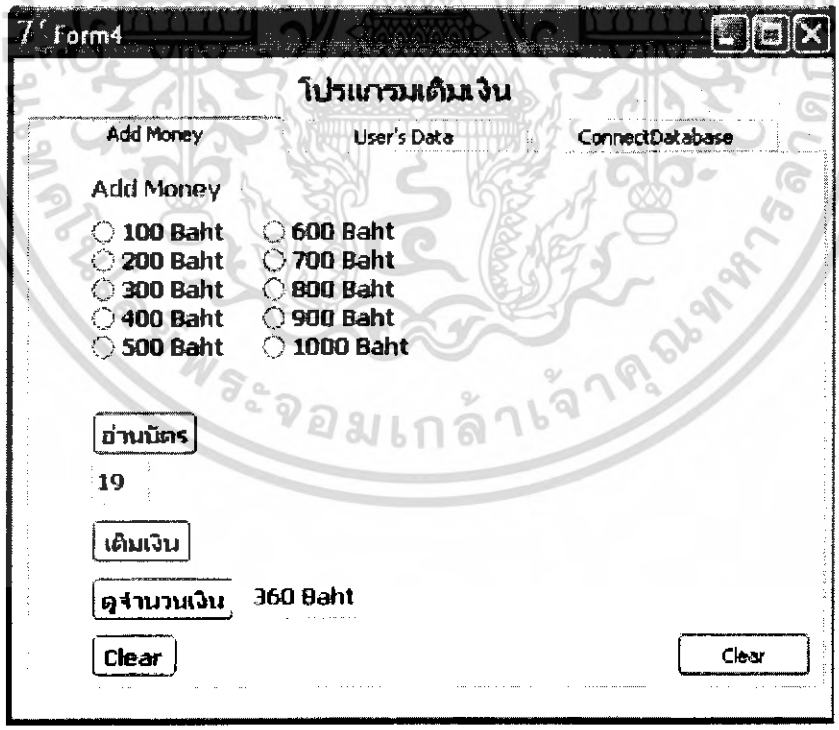
รูปที่ 4.24 แสดงรายละเอียดที่หน้าของโปรแกรมเมื่อต้องการจะเติมเงิน

เมื่อเติมเงินเรียบร้อยแล้ว จะมีหน้าต่างแสดงขึ้นมา



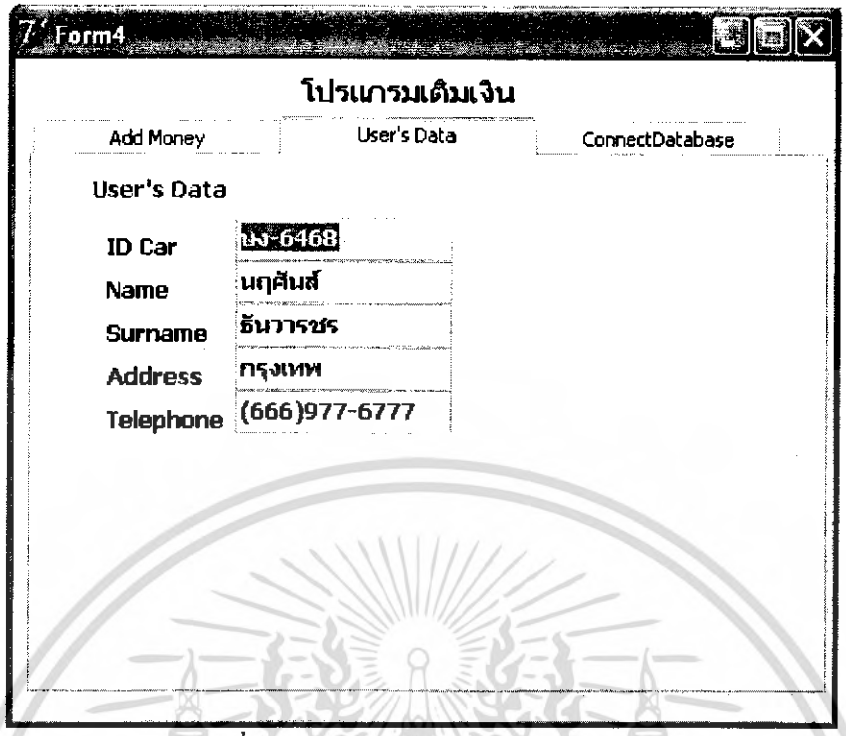
รูปที่ 4.25 แสดงรายละเอียดที่หน้าของโปรแกรม เมื่อเติมเงินเรียบร้อยแล้ว

เมื่อเติมเงินเรียบร้อยแล้ว จะแสดงจำนวนเงินใหม่ขึ้นมา



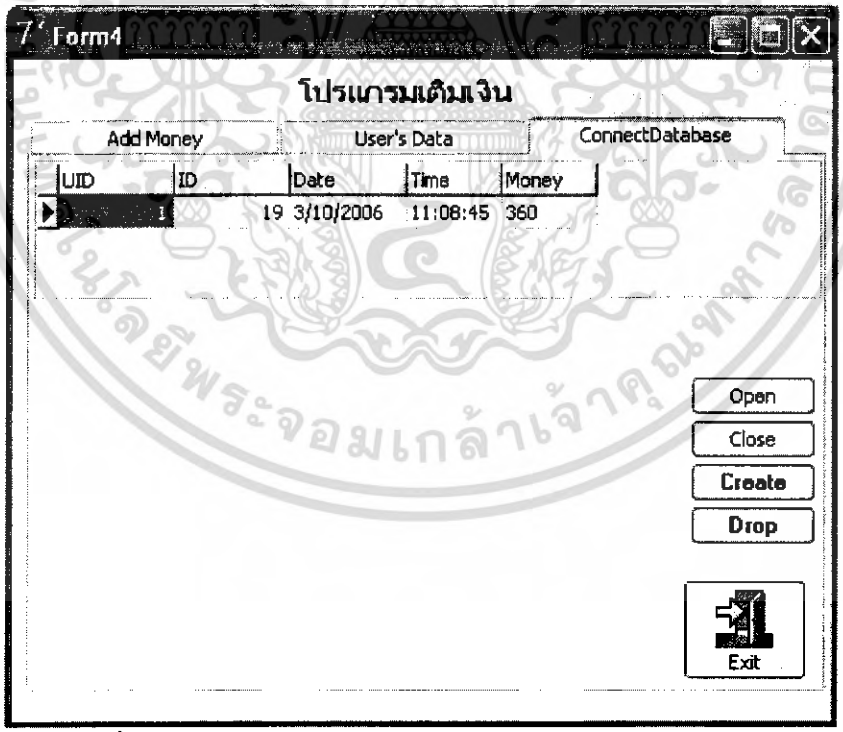
รูปที่ 4.26 แสดงยอดเงินที่เติมแล้ว

ซึ่งจะสามารถดูข้อมูลของผู้ใช้ได้



รูปที่ 4.27 แสดงรายละเอียดข้อมูลของผู้ใช้

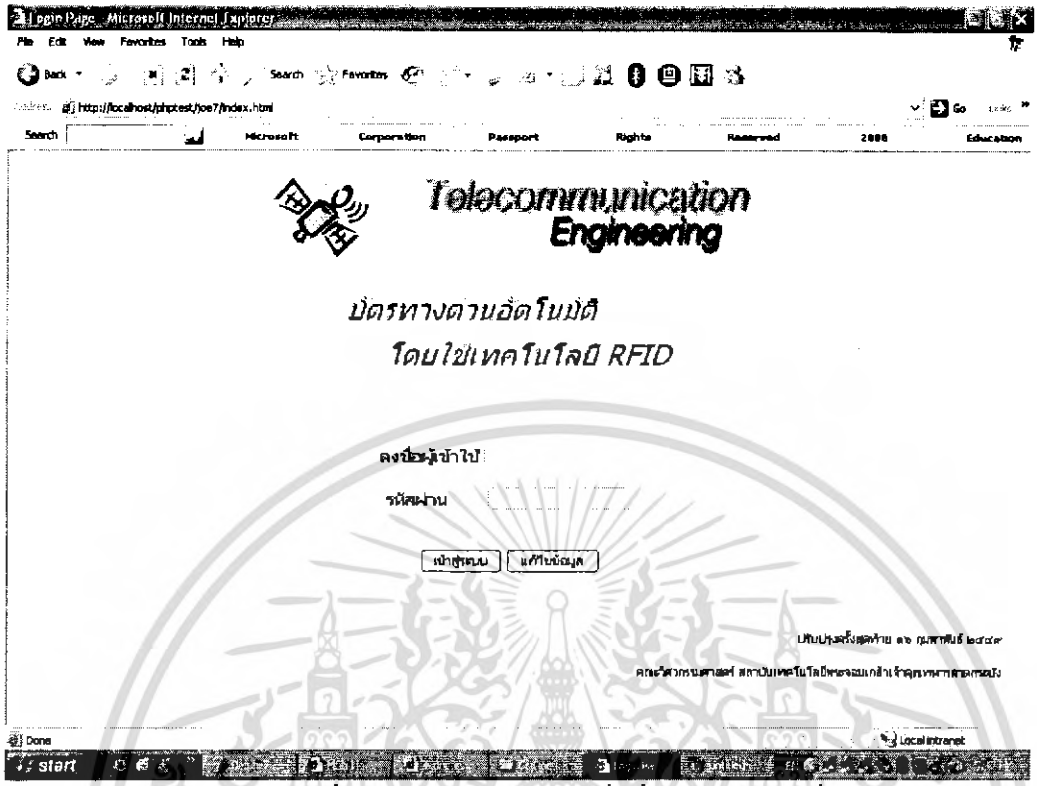
แสดงการบันทึกรายละเอียดการเติมเงินลงในฐานข้อมูล



รูปที่ 4.28 แสดงการบันทึกรายละเอียดการเติมเงินลงในฐานข้อมูล

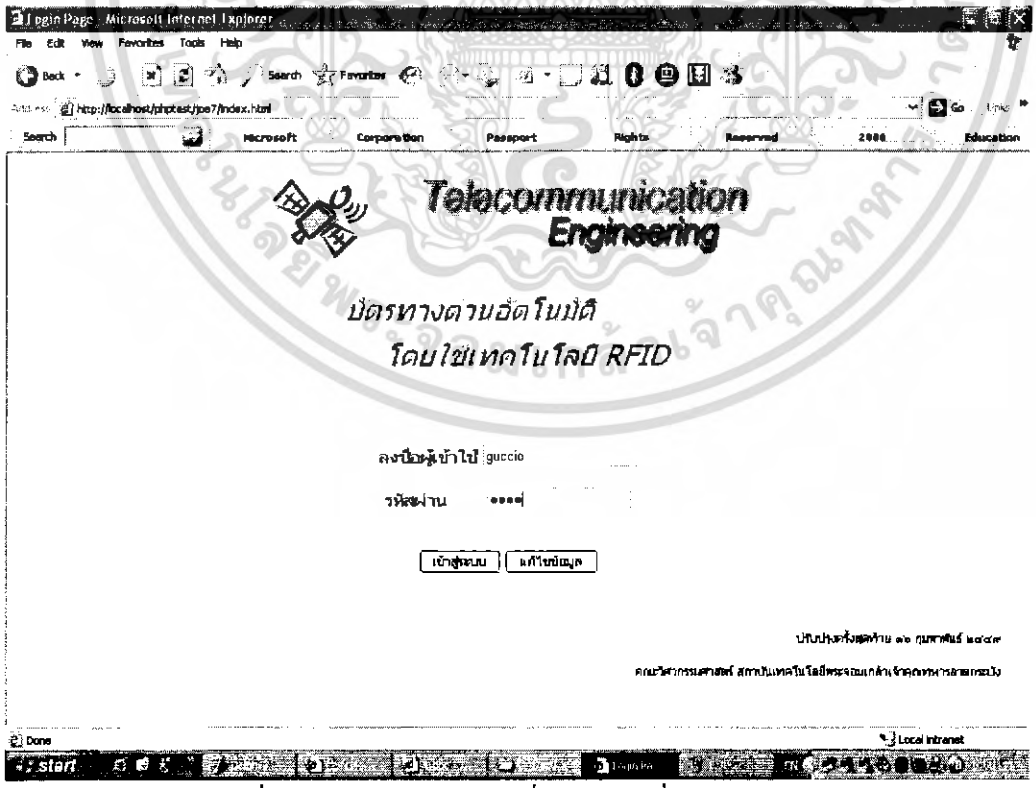
4.10 แสดงรูปฐานข้อมูล

การเข้าไปสู่ฐานข้อมูลจะต้องทำการล็อกอิน และใส่พาสเวิร์ดเพื่อเข้าสู่ระบบ

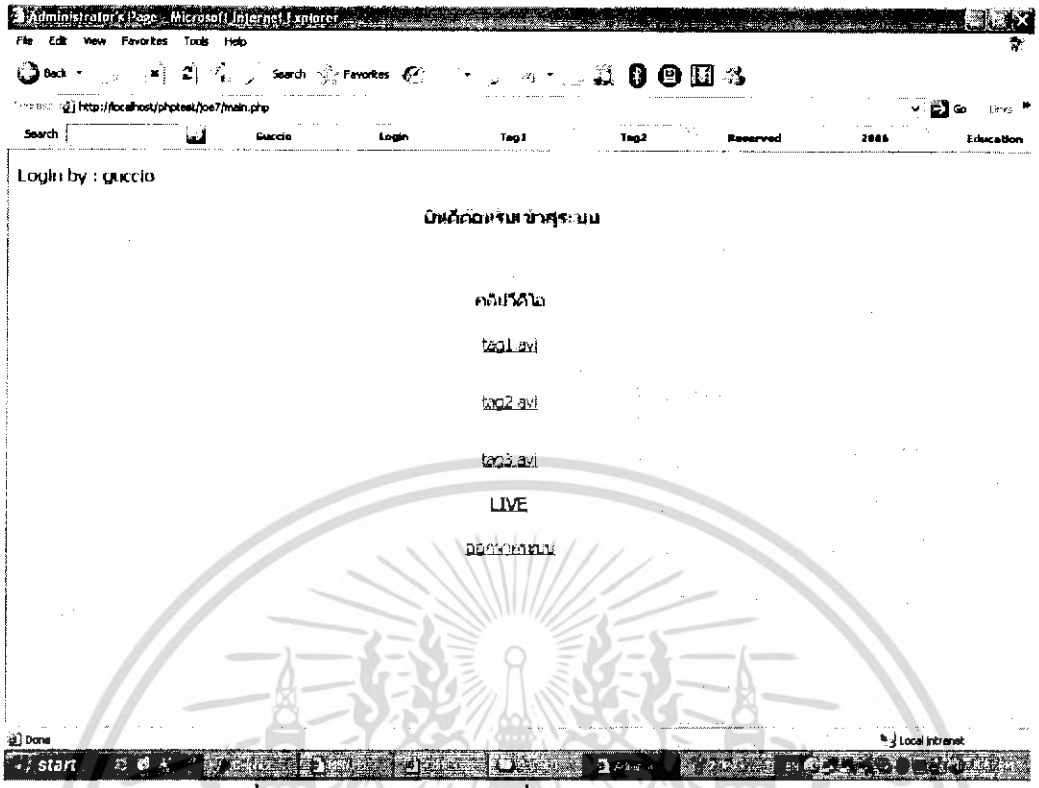


รูปที่ 4.29 แสดงหน้าเวปเพจเพื่อที่จะเข้าไปดูไฟล์ที่บันทึกไว้

ทำการลงชื่อผู้เข้าใช้และใส่รหัสผ่าน

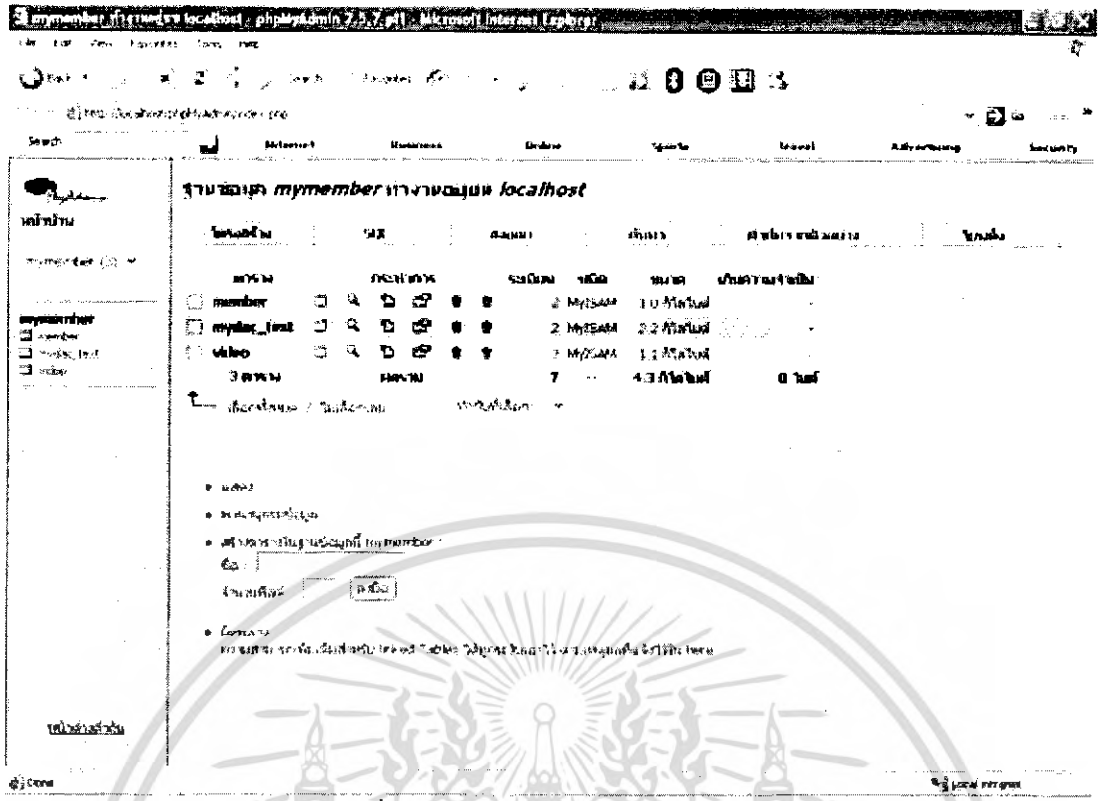


รูปที่ 4.30 แสดงหน้าเวปเพจเมื่อทำการลงชื่อผู้ใช้และใส่รหัสผ่าน

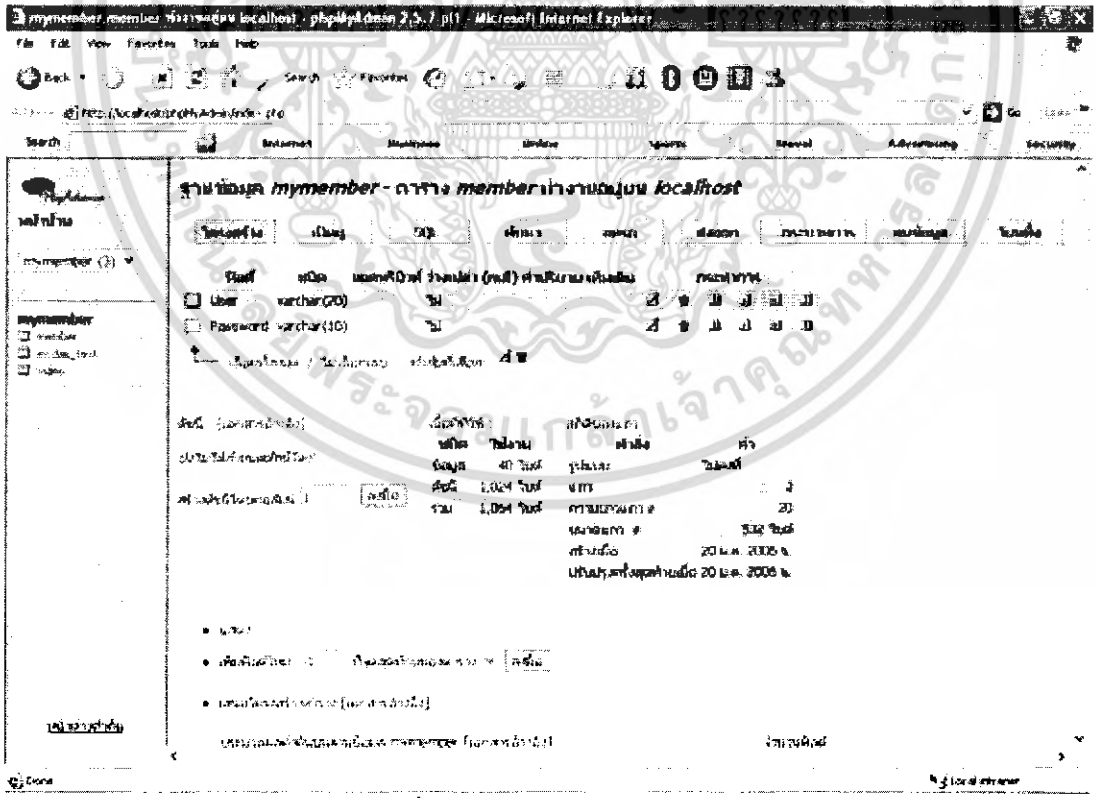


รูปที่ 4.31 แสดงหน้าเว็ปเพจเมื่อต้องการเข้าไปดูวีดีโอคลิป

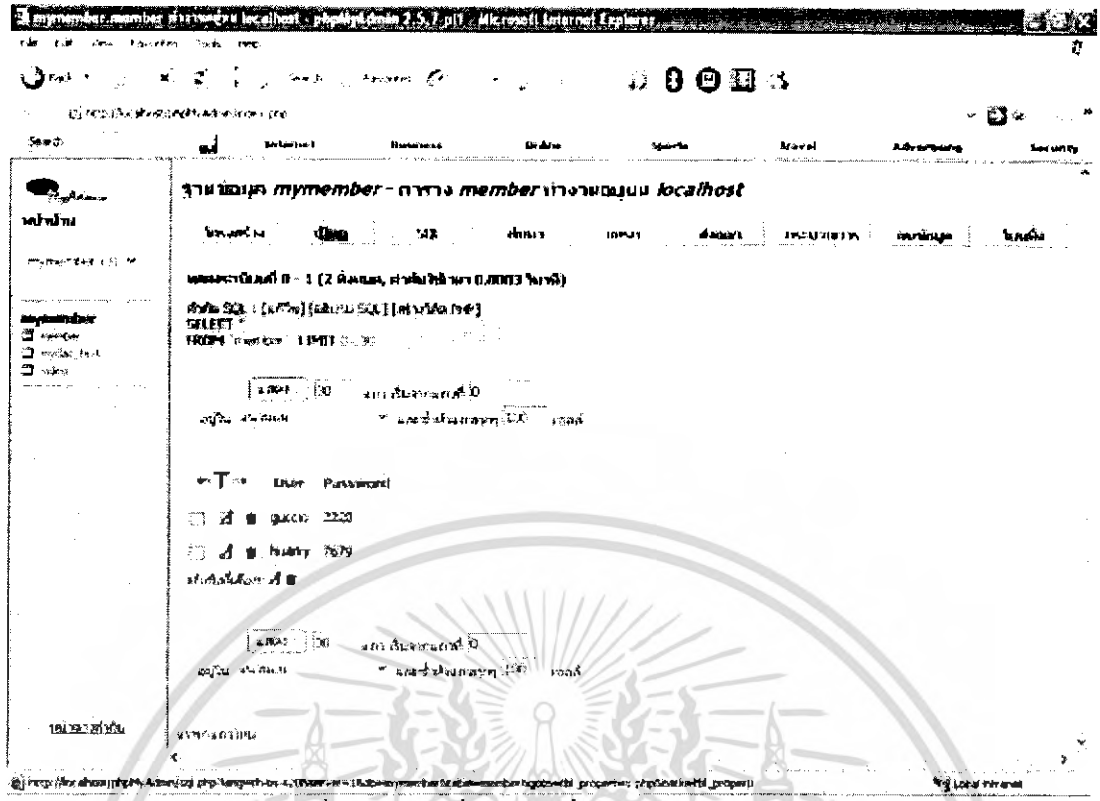
รูปที่ 4.32 แสดงหน้าเว็ปเพจเมื่อต้องการดูภาพเคลื่อนไหว



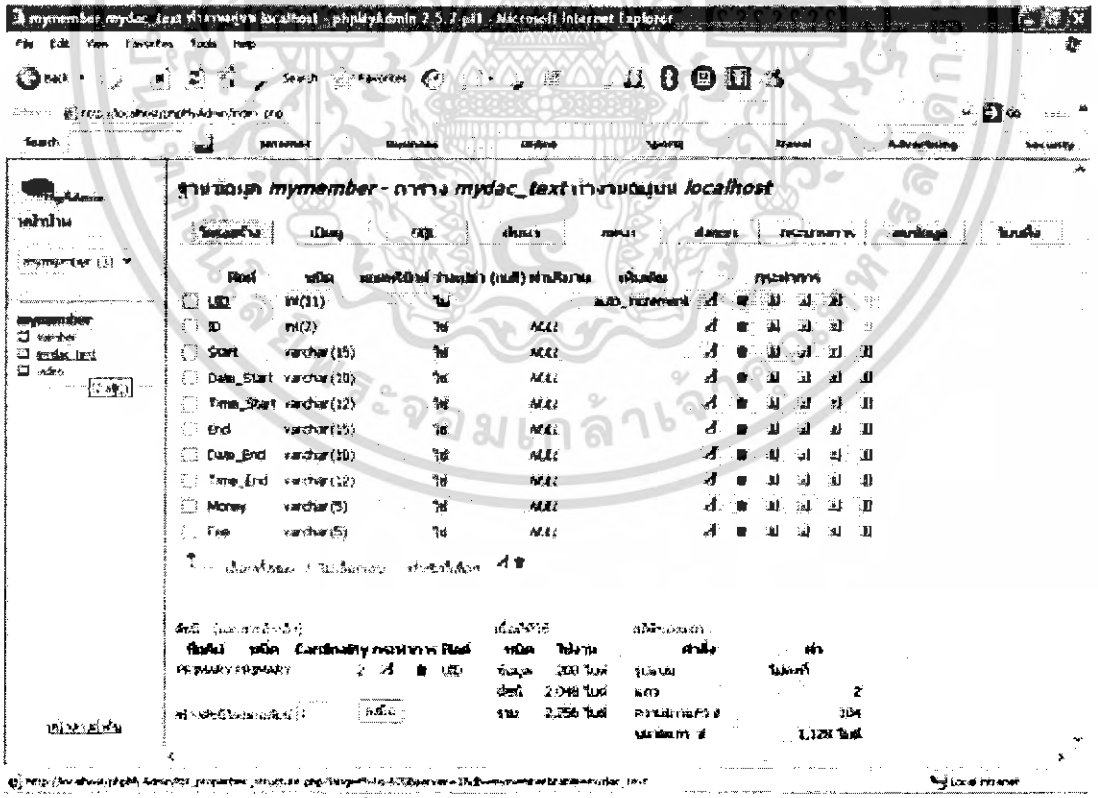
รูปที่ 4.33 แสดงหน้าฐานข้อมูล MySQL



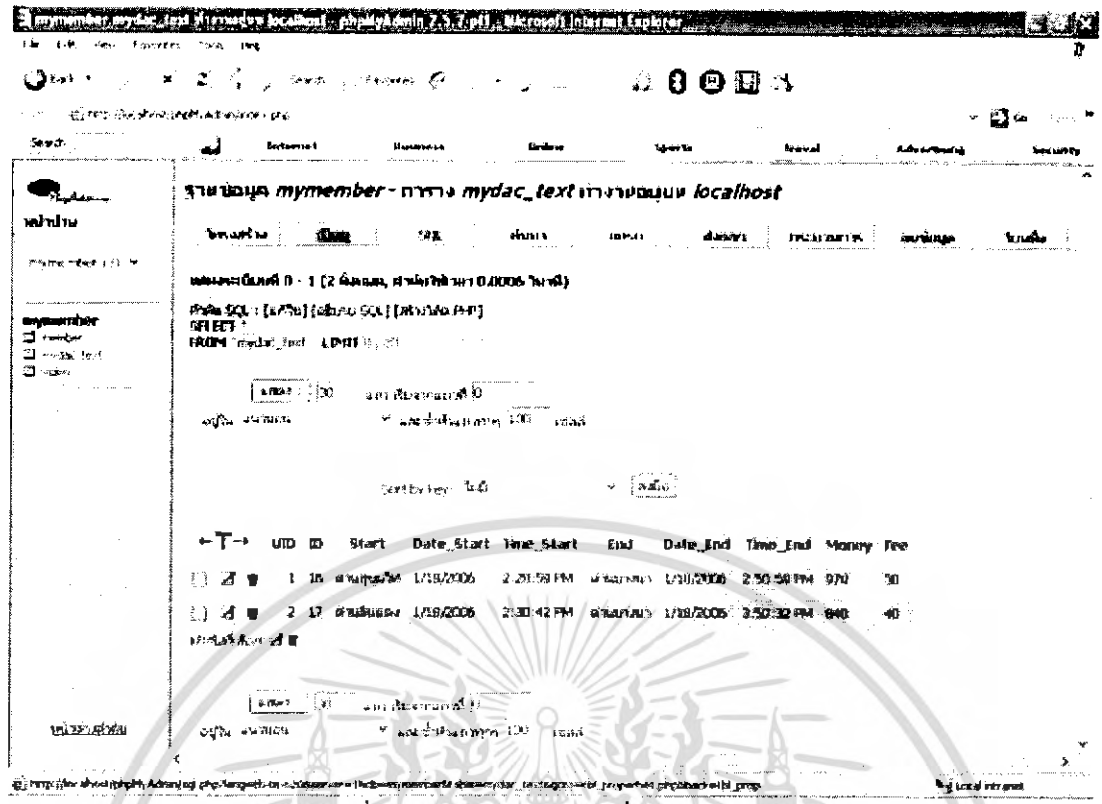
รูปที่ 4.34 แสดงตารางฐานข้อมูล



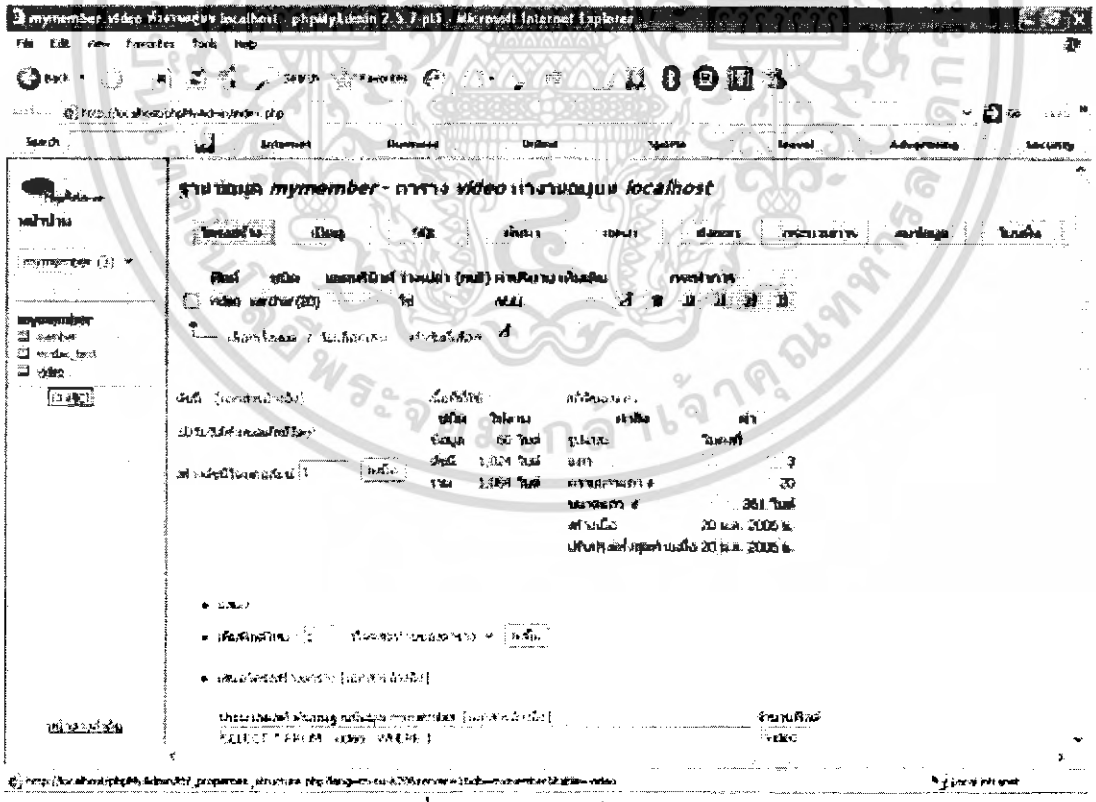
รูปที่ 4.35 แสดงชื่อสมาชิกที่อยู่ในตารางฐานข้อมูล



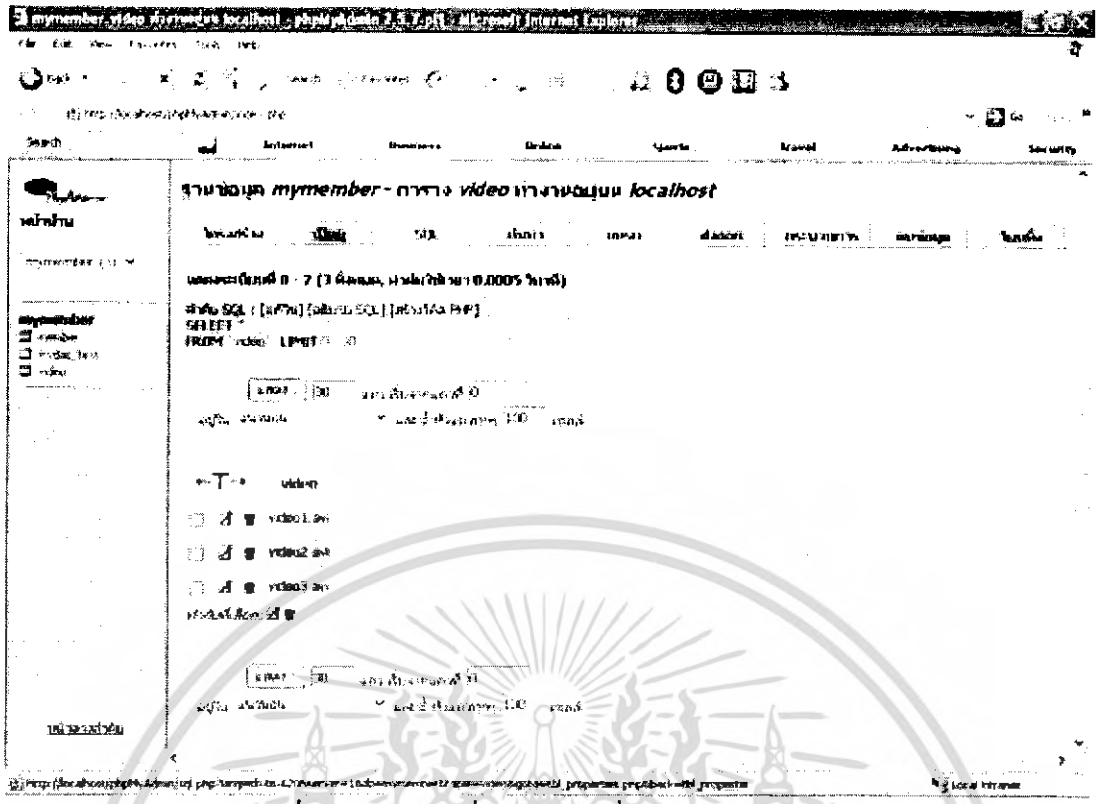
รูปที่ 4.36 แสดงรายละเอียดของการบันทึก



รูปที่ 4.37 แสดงรายละเอียดที่ผู้ใช้ใช้บริการ



รูปที่ 4.38 แสดงฟิลด์ของวีดีโอ



รูปที่ 4.39 แสดงรายชื่อคลิปวีดีโอที่สามารถบันทึกไว้ได้

บทที่ 5
บทสรุปและวิจารณ์

จากการศึกษาเทคโนโลยี RFID ซึ่งจะมีหลักการทำงานคร่าวๆ คือ เมื่อมีแอปพลิเคชั่นที่ต้องการติดต่อไปยังแทคก็จะผ่านวงจรการดีโค๊ดซึ่งจะใช้การเข้ารหัสแบบแมนเชสเตอร์ เข้าสู่วงจรการมอดูเลตซึ่งจะใช้การมอดูเลตแบบแอมปลิจูด แล้วส่งผ่านเสาอากาศออกไป ทางด้านรับเมื่อมีแทคอยู่ในบริเวณที่สามารถอ่านได้ ขดลวดในแทคก็จะคัปปลิ้งสัญญาณ แล้วส่งผ่านไปยังวงจรดีมอดูเลต เข้าวงจรการดีโค๊ต แล้วเข้าไปในไมโครโปรซิพเพื่อทำการประมวลผลว่าผู้ใช้มีคำสั่งอะไร ถ้าเป็นคำสั่งเขียนข้อมูล ก็จะนำข้อมูลใส่เข้าไปในหน่วยความจำของแทค ถ้าเป็นคำสั่งอ่านข้อมูลก็จะผ่านวงจรดีโค้ด วงจรการดีมอดูเลต แล้วส่งผ่านออกทางเสาอากาศของแทค วงจรทางด้านรับก็จะทำการดีมอดูเลต วงจรดีโค๊ตและเข้าสู่ไมโครคอนโทรลเลอร์เพื่อทำการประมวลผลต่อไป

จากการวัดความถี่และขนาดของสัญญาณ จะเห็นว่าความถี่ของสัญญาณที่วัดได้จากสายอากาศขาที่ 1 และสายอากาศขาที่ 2 จะมีความถี่เท่ากันคือ 130 *kHz* สัญญาณที่ออกจากขา DEMOD_OUT ของไอซี EM 4095 จะถูกแปลงจากสัญญาณอนาล็อกเป็นสัญญาณดิจิตอล และจะนำสัญญาณที่ได้เข้าสู่ไมโครคอนโทรเลอร์เพื่อทำหน้าที่ต่อไป แต่ในการวัดสัญญาณจากไมโครคอลโทรเลอร์นั้นไม่สามารถทำการวัดได้ เนื่องจากสัญญาณมีความเร็วมาก

ในส่วนของแอปพลิเคชันได้ทำการสร้างแบบจำลองขึ้นมา ซึ่งจะแบ่งโปรแกรมออกเป็น 3 หน้าต่างคือ หน้าต่างทางด้านสถานีต้นทาง หน้าต่างทางด้านสถานีปลายทาง และหน้าต่างการเติมเงิน เริ่มต้นเมื่อมีรถเข้ามาที่สถานีต้นทางจะผ่านเซนเซอร์เพื่อตรวจสอบว่ามีรถเข้ามา กล้องก็จะเริ่มทำการบันทึกรถที่วิ่งผ่าน ถ้ารถคันไหนมีแทคก็จะทำการบันทึกสถานีต้นทางลงในแทค เมื่อถึงสถานีปลายทางตัวเครื่องอ่านจะทำการอ่านข้อมูลจากแทคแล้วทำการคิดจำนวนเงินตามระยะทาง และหักจำนวนเงินออกแล้วทำการบันทึกจำนวนเงินที่เหลือลงในแทค จะมีส่วนของโปรแกรมเติมเงินเพื่อที่สามารถเติมจำนวนเงินได้ สามารถนำไปประยุกต์ใช้งานจริงได้

หนังสืออ้างอิง

[1] สัจจะ จรัสรุ่งรวีวร จักรพงษ์ สุขประเสริฐ, "คู่มือการสร้างแอพพลิเคชันด้วย Delphi 5.0 ฉบับสมบูรณ์", อินโฟเพรส, พิมพ์ครั้งที่ 2, 2543

[2] นิรุธ อำนวยศิลป์, "การสร้างเวปเพจอย่างไร้ขีดจำกัด PHP เพื่อการประยุกต์ใช้งาน", ซัคเซส มีเดีย, พิมพ์ครั้งที่ 1

[3] วงศ์ประชา จันทร์สมวงศ์ ดวงพร เกี๋ยงคำ, "อินไซต์ Dreamweaver MX 2004", กรุงเทพฯ ไปรวิชั่น, พิมพ์ครั้งที่ 1, 2547

ภาคผนวก

Main Pages

```
unit First;
interface
uses
  Windows, Messages, SysUtils, Variants, Classes, Graphics, Controls,
Forms,
  Dialogs, ExtCtrls, ComCtrls, Buttons;
type
  TForm1 = class(TForm)
    MainPanel: TPanel;
    StartStation: TSpeedButton;
    EndStation: TSpeedButton;
    FillBtn: TSpeedButton;
    StatusBar1: TStatusBar;
    Timer1: TTimer;
    CloseBtn: TSpeedButton;
    procedure Timer1Timer(Sender: TObject);
    procedure CloseBtnClick(Sender: TObject);
    procedure StartStationClick(Sender: TObject);
    procedure EndStationClick(Sender: TObject);
    procedure FillBtnClick(Sender: TObject);
  private
    { Private declarations }
  public
    { Public declarations }
  end;
var
  Form1: TForm1;

implementation
uses Unit2,Unit3,Unit4;

{$R *.dfm}

procedure TForm1.Timer1Timer(Sender: TObject);
begin
    Statusbar1.Panels[1].Text:=DateToStr(Date);
    Statusbar1.Panels[2].Text:=TimeToStr(Time);
end;
procedure TForm1.CloseBtnClick(Sender: TObject);
begin
    Showmessage('<<<Thank You>>>');
    Close;
end;
procedure TForm1.StartStationClick(Sender: TObject);
begin
    Form2.ShowModal;
end;
procedure TForm1.EndStationClick(Sender: TObject);
begin
    Form3.ShowModal;
end;
procedure TForm1.FillBtnClick(Sender: TObject);
begin
    Form4.ShowModal;
end;
end.
```

Start Station

```
unit Unit2;

interface

uses
  Windows, Messages, SysUtils, Variants, Classes, Graphics, Controls,
Forms,
  Dialogs, XPMan, VaClasses, VaComm, ExtCtrls, DBAccess, MyAccess, DB,
  MemDS, MyDacVcl, StdCtrls, Buttons, Mask, DBCtrls, Grids, DBGrids,
  ComCtrls, ieview, imageenview, imageen;

type
  TForm2 = class(TForm)
    Panel1: TPanel;
    Label2: TLabel;
    GroupBox1: TGroupBox;
    Image1: TImage;
    RadioButton1: TRadioButton;
    RadioButton2: TRadioButton;
    RadioButton3: TRadioButton;
    RadioButton4: TRadioButton;
    RadioButton5: TRadioButton;
    ImageEn1: TImageEn;
    Panel2: TPanel;
    PageControl1: TPageControl;
    TabSheet1: TTabSheet;
    GroupBox2: TGroupBox;
    Label1: TLabel;
    SpeedButton1: TSpeedButton;
    Label4: TLabel;
    Shape1: TShape;
    Label3: TLabel;
    SpeedButton2: TSpeedButton;
    Label5: TLabel;
    ComboBox1: TComboBox;
    TabSheet2: TTabSheet;
    DBGrid: TDBGrid;
    DBEdit1: TDBEdit;
    DBEdit2: TDBEdit;
    BitBtn1: TBitBtn;
    BitBtn2: TBitBtn;
    Edit1: TEdit;
    Edit5: TEdit;
    TabSheet4: TTabSheet;
    Label6: TLabel;
    Edit7: TEdit;
    MyConnection: TMyConnection;
    DataSource: TDataSource;
    ConnectDialog: TMyConnectDialog;
    Query: TMyQuery;
    sqDropTable: TMyCommand;
    Timer1: TTimer;
    VaComm1: TVaComm;
    XPManifest1: TXPManifest;
    Timer3: TTimer;
    Timer4: TTimer;
```

```
    Timer5: TTimer;
    Timer6: TTimer;
    Timer7: TTimer;
    sqCreateTable: TMyCommand;
    Timer8: TTimer;
    btOpen: TButton;
    btCreate: TButton;
    btClose: TButton;
    btDrop: TButton;
    ExitBtn: TSpeedButton;
    procedure btOpenClick(Sender: TObject);
    procedure btCloseClick(Sender: TObject);
    procedure btCreateClick(Sender: TObject);
    procedure btDropClick(Sender: TObject);
    procedure VaComm1RxChar(Sender: TObject; Count: Integer);
    procedure Timer1Timer(Sender: TObject);
    procedure BitBtn1Click(Sender: TObject);
    procedure BitBtn2Click(Sender: TObject);
    procedure SpeedButton2Click(Sender: TObject);
    procedure SpeedButton1Click(Sender: TObject);
    procedure Timer3Timer(Sender: TObject);
    procedure Timer4Timer(Sender: TObject);
    procedure FormActivate(Sender: TObject);
    procedure ComboBox1Change(Sender: TObject);
    procedure Timer8Timer(Sender: TObject);
    procedure Timer5Timer(Sender: TObject);
    procedure Edit7Change(Sender: TObject);
    procedure Timer6Timer(Sender: TObject);
    procedure Timer7Timer(Sender: TObject);
    procedure ImageEn1DShowNewFrame(Sender: TObject);
    procedure ExitBtnClick(Sender: TObject);
  private
    { Private declarations }
  public
    { Public declarations }
    procedure Connect;
    procedure Disconnect;
    procedure ShowVideoFormats;
  end;

procedure PortOut(Port : Word; Data : Byte); StdCall; External
'io.dll';
function PortIn(Port : Word) : byte; StdCall; External 'io.dll'

var
  Form2: TForm2;
  RxBuff : array[1..33] of byte;
  count : integer;
  port,byte31,byte4,byte5,byte6,byte7,p : string;

implementation

{$R *.dfm}

procedure TForm2.btOpenClick(Sender: TObject);
begin
   Query.Open;
```

```
end;
procedure TForm2.btCloseClick(Sender: TObject);
begin
   Query.Close;
end;
procedure TForm2.btCreateClick(Sender: TObject);
begin
   sqCreateTable.Execute;
end;
procedure TForm2.btDropClick(Sender: TObject);
begin
   sqDropTable.Execute;
end;
procedure TForm2.VaComm1RxChar(Sender: TObject; Count: Integer);
var  St,start:string;
     AA:integer;
begin
  if count >= 33 then
  begin
    St := Vacomm1.ReadText;
    Edit1.Text := '';
    for AA := 1 to 33 do
    begin
    Edit1.text := Edit1.Text + ' ' + inttostr((ord(St[AA])));
    end;
    byte31:= inttostr((ord(st[31])));
    byte4:= inttostr((ord(st[4])));
    byte5:= inttostr((ord(st[5])));
    byte6:= inttostr((ord(st[6])));
    byte7:= inttostr((ord(st[7])));
    if (byte31='16') or (byte31='17') or (byte31='18') or
       (byte31='19') or (byte31='20') then
begin
    with query do
    begin
    append;
    Fieldbyname('Date_Start').AsString := datetostr(date);
    Fieldbyname('Time_Start').AsString := timetostr(time);
    Fieldbyname('ID').AsString := byte31;
begin
        if byte5='0' then
          begin
            if byte6='0'  then Fieldbyname('Money').AsString :='0';
            if byte6='1'  then Fieldbyname('Money').AsString :='10';
            if byte6='2'  then Fieldbyname('Money').AsString :='20';
            if byte6='3'  then Fieldbyname('Money').AsString :='30';
            if byte6='4'  then Fieldbyname('Money').AsString :='40';
            if byte6='5'  then Fieldbyname('Money').AsString :='50';
            if byte6='6'  then Fieldbyname('Money').AsString :='60';
            if byte6='7'  then Fieldbyname('Money').AsString :='70';
            if byte6='8'  then Fieldbyname('Money').AsString :='80';
            if byte6='9'  then Fieldbyname('Money').AsString :='90';
          end;
       if byte5='1' then
          begin
            if byte6='0'  then Fieldbyname('Money').AsString :='100';
            if byte6='1'  then Fieldbyname('Money').AsString :='110';
```

```
      if byte6='2'  then Fieldbyname('Money').AsString :='120';
      if byte6='3'  then Fieldbyname('Money').AsString :='130';
      if byte6='4'  then Fieldbyname('Money').AsString :='140';
      if byte6='5'  then Fieldbyname('Money').AsString :='150';
      if byte6='6'  then Fieldbyname('Money').AsString :='160';
      if byte6='7'  then Fieldbyname('Money').AsString :='170';
      if byte6='8'  then Fieldbyname('Money').AsString :='180';
      if byte6='9'  then Fieldbyname('Money').AsString :='190';
    end;
if byte5='2' then
    begin
      if byte6='0'  then Fieldbyname('Money').AsString :='200';
      if byte6='1'  then Fieldbyname('Money').AsString :='210';
      if byte6='2'  then Fieldbyname('Money').AsString :='220';
      if byte6='3'  then Fieldbyname('Money').AsString :='230';
      if byte6='4'  then Fieldbyname('Money').AsString :='240';
      if byte6='5'  then Fieldbyname('Money').AsString :='250';
      if byte6='6'  then Fieldbyname('Money').AsString :='260';
      if byte6='7'  then Fieldbyname('Money').AsString :='270';
      if byte6='8'  then Fieldbyname('Money').AsString :='280';
      if byte6='9'  then Fieldbyname('Money').AsString :='290';
    end;
if byte5='3' then
    begin
      if byte6='0'  then Fieldbyname('Money').AsString :='300';
      if byte6='1'  then Fieldbyname('Money').AsString :='310';
      if byte6='2'  then Fieldbyname('Money').AsString :='320';
      if byte6='3'  then Fieldbyname('Money').AsString :='330';
      if byte6='4'  then Fieldbyname('Money').AsString :='340';
      if byte6='5'  then Fieldbyname('Money').AsString :='350';
      if byte6='6'  then Fieldbyname('Money').AsString :='360';
      if byte6='7'  then Fieldbyname('Money').AsString :='370';
      if byte6='8'  then Fieldbyname('Money').AsString :='380';
      if byte6='9'  then Fieldbyname('Money').AsString :='390';
    end;
 if byte5='4' then
    begin
      if byte6='0'  then Fieldbyname('Money').AsString :='400';
      if byte6='1'  then Fieldbyname('Money').AsString :='410';
      if byte6='2'  then Fieldbyname('Money').AsString :='420';
      if byte6='3'  then Fieldbyname('Money').AsString :='430';
      if byte6='4'  then Fieldbyname('Money').AsString :='440';
      if byte6='5'  then Fieldbyname('Money').AsString :='450';
      if byte6='6'  then Fieldbyname('Money').AsString :='460';
      if byte6='7'  then Fieldbyname('Money').AsString :='470';
      if byte6='8'  then Fieldbyname('Money').AsString :='480';
      if byte6='9'  then Fieldbyname('Money').AsString :='490';
    end;
 if byte5='5' then
    begin
      if byte6='0'  then Fieldbyname('Money').AsString :='500';
      if byte6='1'  then Fieldbyname('Money').AsString :='510';
      if byte6='2'  then Fieldbyname('Money').AsString :='520';
      if byte6='3'  then Fieldbyname('Money').AsString :='530';
      if byte6='4'  then Fieldbyname('Money').AsString :='540';
      if byte6='5'  then Fieldbyname('Money').AsString :='550';
      if byte6='6'  then Fieldbyname('Money').AsString :='560';
```

```
    if byte6='7'  then Fieldbyname('Money').AsString :='570';
    if byte6='8'  then Fieldbyname('Money').AsString :='580';
    if byte6='9'  then Fieldbyname('Money').AsString :='590';
  end;
if byte5='6' then
  begin
    if byte6='0'  then Fieldbyname('Money').AsString :='600';
    if byte6='1'  then Fieldbyname('Money').AsString :='610';
    if byte6='2'  then Fieldbyname('Money').AsString :='620';
    if byte6='3'  then Fieldbyname('Money').AsString :='630';
    if byte6='4'  then Fieldbyname('Money').AsString :='640';
    if byte6='5'  then Fieldbyname('Money').AsString :='650';
    if byte6='6'  then Fieldbyname('Money').AsString :='660';
    if byte6='7'  then Fieldbyname('Money').AsString :='670';
    if byte6='8'  then Fieldbyname('Money').AsString :='680';
    if byte6='9'  then Fieldbyname('Money').AsString :='690';
  end;
if byte5='7' then
  begin
    if byte6='0'  then Fieldbyname('Money').AsString :='700';
    if byte6='1'  then Fieldbyname('Money').AsString :='710';
    if byte6='2'  then Fieldbyname('Money').AsString :='720';
    if byte6='3'  then Fieldbyname('Money').AsString :='730';
    if byte6='4'  then Fieldbyname('Money').AsString :='740';
    if byte6='5'  then Fieldbyname('Money').AsString :='750';
    if byte6='6'  then Fieldbyname('Money').AsString :='760';
    if byte6='7'  then Fieldbyname('Money').AsString :='770';
    if byte6='8'  then Fieldbyname('Money').AsString :='780';
    if byte6='9'  then Fieldbyname('Money').AsString :='790';
  end;
if byte5='8' then
 begin
    if byte6='0'  then Fieldbyname('Money').AsString :='800';
    if byte6='1'  then Fieldbyname('Money').AsString :='810';
    if byte6='2'  then Fieldbyname('Money').AsString :='820';
    if byte6='3'  then Fieldbyname('Money').AsString :='830';
    if byte6='4'  then Fieldbyname('Money').AsString :='840';
    if byte6='5'  then Fieldbyname('Money').AsString :='850';
    if byte6='6'  then Fieldbyname('Money').AsString :='860';
    if byte6='7'  then Fieldbyname('Money').AsString :='870';
    if byte6='8'  then Fieldbyname('Money').AsString :='880';
    if byte6='9'  then Fieldbyname('Money').AsString :='890';
  end;
if byte5='9' then
  begin
    if byte6='0'  then Fieldbyname('Money').AsString :='900';
    if byte6='1'  then Fieldbyname('Money').AsString :='910';
    if byte6='2'  then Fieldbyname('Money').AsString :='920';
    if byte6='3'  then Fieldbyname('Money').AsString :='930';
    if byte6='4'  then Fieldbyname('Money').AsString :='940';
    if byte6='5'  then Fieldbyname('Money').AsString :='950';
    if byte6='6'  then Fieldbyname('Money').AsString :='960';
    if byte6='7'  then Fieldbyname('Money').AsString :='970';
    if byte6='8'  then Fieldbyname('Money').AsString :='980';
    if byte6='9'  then Fieldbyname('Money').AsString :='990';
  end;
if byte5 = '16' then Fieldbyname('Money').AsString:='1000';
```

```
end;

begin
if radiobutton1.Checked then
    begin
    Fieldbyname('Start').AsString := '´èÒ¹´Ô¹á´§';
    if Fieldbyname('Money').AsString='0' then
start:=chr($FB)+chr($04)+chr($01)+chr($10)+chr($00)+chr($00)+chr($00)+
           chr($EE)+chr($FC);
    if Fieldbyname('Money').AsString='10' then
start:=chr($FB)+chr($04)+chr($01)+chr($10)+chr($00)+chr($01)+chr($00)+
           chr($EF)+chr($FC);
    if Fieldbyname('Money').AsString='20' then
start:=chr($FB)+chr($04)+chr($01)+chr($10)+chr($00)+chr($02)+chr($00)+
           chr($EC)+chr($FC);
    if Fieldbyname('Money').AsString='30' then
start:=chr($FB)+chr($04)+chr($01)+chr($10)+chr($00)+chr($03)+chr($00)+
           chr($ED)+chr($FC);
    if Fieldbyname('Money').AsString='40' then
start:=chr($FB)+chr($04)+chr($01)+chr($10)+chr($00)+chr($04)+chr($00)+
           chr($EA)+chr($FC);
    if Fieldbyname('Money').AsString='50' then
start:=chr($FB)+chr($04)+chr($01)+chr($10)+chr($00)+chr($05)+chr($00)+
           chr($EB)+chr($FC);
    if Fieldbyname('Money').AsString='60' then
start:=chr($FB)+chr($04)+chr($01)+chr($10)+chr($00)+chr($06)+chr($00)+
           chr($E8)+chr($FC);
    if Fieldbyname('Money').AsString='70' then
start:=chr($FB)+chr($04)+chr($01)+chr($10)+chr($00)+chr($07)+chr($00)+
           chr($E9)+chr($FC);
    if Fieldbyname('Money').AsString='80' then
start:=chr($FB)+chr($04)+chr($01)+chr($10)+chr($00)+chr($08)+chr($00)+
           chr($E6)+chr($FC);
    if Fieldbyname('Money').AsString='90' then
start:=chr($FB)+chr($04)+chr($01)+chr($10)+chr($00)+chr($09)+chr($00)+
           chr($E7)+chr($FC);
    if Fieldbyname('Money').AsString='100' then
start:=chr($FB)+chr($04)+chr($01)+chr($10)+chr($01)+chr($00)+chr($00)+
           chr($EF)+chr($FC);
    if Fieldbyname('Money').AsString='110' then
start:=chr($FB)+chr($04)+chr($01)+chr($10)+chr($01)+chr($01)+chr($00)+
           chr($EE)+chr($FC);
    if Fieldbyname('Money').AsString='120' then
start:=chr($FB)+chr($04)+chr($01)+chr($10)+chr($01)+chr($02)+chr($00)+
           chr($ED)+chr($FC);
    if Fieldbyname('Money').AsString='130' then
start:=chr($FB)+chr($04)+chr($01)+chr($10)+chr($01)+chr($03)+chr($00)+
           chr($EC)+chr($FC);
    if Fieldbyname('Money').AsString='140' then
start:=chr($FB)+chr($04)+chr($01)+chr($10)+chr($01)+chr($04)+chr($00)+
           chr($EB)+chr($FC);
    if Fieldbyname('Money').AsString='150' then
start:=chr($FB)+chr($04)+chr($01)+chr($10)+chr($01)+chr($05)+chr($00)+
           chr($EA)+chr($FC);
    if Fieldbyname('Money').AsString='160' then
start:=chr($FB)+chr($04)+chr($01)+chr($10)+chr($01)+chr($06)+chr($00)+
           chr($E9)+chr($FC);
```

```
    if Fieldbyname('Money').AsString='170' then
start:=chr($FB)+chr($04)+chr($01)+chr($10)+chr($01)+chr($07)+chr($00)+
           chr($E8)+chr($FC);
    if Fieldbyname('Money').AsString='180' then
start:=chr($FB)+chr($04)+chr($01)+chr($10)+chr($01)+chr($08)+chr($00)+
           chr($E7)+chr($FC);
    if Fieldbyname('Money').AsString='190' then
start:=chr($FB)+chr($04)+chr($01)+chr($10)+chr($01)+chr($09)+chr($00)+
           chr($E6)+chr($FC);
    if Fieldbyname('Money').AsString='200' then
start:=chr($FB)+chr($04)+chr($01)+chr($10)+chr($02)+chr($00)+chr($00)+
           chr($EC)+chr($FC);
    if Fieldbyname('Money').AsString='210' then
start:=chr($FB)+chr($04)+chr($01)+chr($10)+chr($02)+chr($01)+chr($00)+
           chr($ED)+chr($FC);
    if Fieldbyname('Money').AsString='220' then
start:=chr($FB)+chr($04)+chr($01)+chr($10)+chr($02)+chr($02)+chr($00)+
           chr($EE)+chr($FC);
    if Fieldbyname('Money').AsString='230' then
start:=chr($FB)+chr($04)+chr($01)+chr($10)+chr($02)+chr($03)+chr($00)+
           chr($EF)+chr($FC);
    if Fieldbyname('Money').AsString='240' then
start:=chr($FB)+chr($04)+chr($01)+chr($10)+chr($02)+chr($04)+chr($00)+
           chr($E8)+chr($FC);
    if Fieldbyname('Money').AsString='250' then
start:=chr($FB)+chr($04)+chr($01)+chr($10)+chr($02)+chr($05)+chr($00)+
           chr($E9)+chr($FC);
    if Fieldbyname('Money').AsString='260' then
start:=chr($FB)+chr($04)+chr($01)+chr($10)+chr($02)+chr($06)+chr($00)+
           chr($EA)+chr($FC);
    if Fieldbyname('Money').AsString='270' then
start:=chr($FB)+chr($04)+chr($01)+chr($10)+chr($02)+chr($07)+chr($00)+
           chr($EB)+chr($FC);
    if Fieldbyname('Money').AsString='280' then
start:=chr($FB)+chr($04)+chr($01)+chr($10)+chr($02)+chr($08)+chr($00)+
           chr($E4)+chr($FC);
    if Fieldbyname('Money').AsString='290' then
start:=chr($FB)+chr($04)+chr($01)+chr($10)+chr($02)+chr($09)+chr($00)+
           chr($E5)+chr($FC);
    if Fieldbyname('Money').AsString='300' then
start:=chr($FB)+chr($04)+chr($01)+chr($10)+chr($03)+chr($00)+chr($00)+
           chr($ED)+chr($FC);
    if Fieldbyname('Money').AsString='310' then
start:=chr($FB)+chr($04)+chr($01)+chr($10)+chr($03)+chr($01)+chr($00)+
           chr($EC)+chr($FC);
    if Fieldbyname('Money').AsString='320' then
start:=chr($FB)+chr($04)+chr($01)+chr($10)+chr($03)+chr($02)+chr($00)+
           chr($EF)+chr($FC);
    if Fieldbyname('Money').AsString='330' then
start:=chr($FB)+chr($04)+chr($01)+chr($10)+chr($03)+chr($03)+chr($00)+
           chr($EE)+chr($FC);
    if Fieldbyname('Money').AsString='340' then
start:=chr($FB)+chr($04)+chr($01)+chr($10)+chr($03)+chr($04)+chr($00)+
           chr($E9)+chr($FC);
    if Fieldbyname('Money').AsString='350' then
start:=chr($FB)+chr($04)+chr($01)+chr($10)+chr($03)+chr($05)+chr($00)+
           chr($E8)+chr($FC);
```

```
    if Fieldbyname('Money').AsString='360' then
start:=chr($FB)+chr($04)+chr($01)+chr($10)+chr($03)+chr($06)+chr($00)+
           chr($EB)+chr($FC);
    if Fieldbyname('Money').AsString='370' then
start:=chr($FB)+chr($04)+chr($01)+chr($10)+chr($03)+chr($07)+chr($00)+
           chr($EA)+chr($FC);
    if Fieldbyname('Money').AsString='380' then
start:=chr($FB)+chr($04)+chr($01)+chr($10)+chr($03)+chr($08)+chr($00)+
           chr($E5)+chr($FC);
    if Fieldbyname('Money').AsString='390' then
start:=chr($FB)+chr($04)+chr($01)+chr($10)+chr($03)+chr($09)+chr($00)+
           chr($E4)+chr($FC);
    if Fieldbyname('Money').AsString='400' then
start:=chr($FB)+chr($04)+chr($01)+chr($10)+chr($04)+chr($00)+chr($00)+
           chr($EA)+chr($FC);
    if Fieldbyname('Money').AsString='410' then
start:=chr($FB)+chr($04)+chr($01)+chr($10)+chr($04)+chr($01)+chr($00)+
           chr($EB)+chr($FC);
    if Fieldbyname('Money').AsString='420' then
start:=chr($FB)+chr($04)+chr($01)+chr($10)+chr($04)+chr($02)+chr($00)+
           chr($E8)+chr($FC);
    if Fieldbyname('Money').AsString='430' then
start:=chr($FB)+chr($04)+chr($01)+chr($10)+chr($04)+chr($03)+chr($00)+
           chr($E9)+chr($FC);
    if Fieldbyname('Money').AsString='440' then
start:=chr($FB)+chr($04)+chr($01)+chr($10)+chr($04)+chr($04)+chr($00)+
           chr($EE)+chr($FC);
    if Fieldbyname('Money').AsString='450' then
start:=chr($FB)+chr($04)+chr($01)+chr($10)+chr($04)+chr($05)+chr($00)+
           chr($EF)+chr($FC);
    if Fieldbyname('Money').AsString='460' then
start:=chr($FB)+chr($04)+chr($01)+chr($10)+chr($04)+chr($06)+chr($00)+
           chr($EC)+chr($FC);
    if Fieldbyname('Money').AsString='470' then
start:=chr($FB)+chr($04)+chr($01)+chr($10)+chr($04)+chr($07)+chr($00)+
           chr($ED)+chr($FC);
    if Fieldbyname('Money').AsString='480' then
start:=chr($FB)+chr($04)+chr($01)+chr($10)+chr($04)+chr($08)+chr($00)+
           chr($E2)+chr($FC);
    if Fieldbyname('Money').AsString='490' then
start:=chr($FB)+chr($04)+chr($01)+chr($10)+chr($04)+chr($09)+chr($00)+
           chr($E3)+chr($FC);
    if Fieldbyname('Money').AsString='500' then
start:=chr($FB)+chr($04)+chr($01)+chr($10)+chr($05)+chr($00)+chr($00)+
           chr($EB)+chr($FC);
    if Fieldbyname('Money').AsString='510' then
start:=chr($FB)+chr($04)+chr($01)+chr($10)+chr($05)+chr($01)+chr($00)+
           chr($EA)+chr($FC);
    if Fieldbyname('Money').AsString='520' then
start:=chr($FB)+chr($04)+chr($01)+chr($10)+chr($05)+chr($02)+chr($00)+
           chr($E9)+chr($FC);
    if Fieldbyname('Money').AsString='530' then
start:=chr($FB)+chr($04)+chr($01)+chr($10)+chr($05)+chr($03)+chr($00)+
           chr($E8)+chr($FC);
    if Fieldbyname('Money').AsString='540' then
start:=chr($FB)+chr($04)+chr($01)+chr($10)+chr($05)+chr($04)+chr($00)+
           chr($EF)+chr($FC);
```

```
    if Fieldbyname('Money').AsString='550' then
start:=chr($FB)+chr($04)+chr($01)+chr($10)+chr($05)+chr($05)+chr($00)+
           chr($EE)+chr($FC);
    if Fieldbyname('Money').AsString='560' then
start:=chr($FB)+chr($04)+chr($01)+chr($10)+chr($05)+chr($06)+chr($00)+
           chr($ED)+chr($FC);
    if Fieldbyname('Money').AsString='570' then
start:=chr($FB)+chr($04)+chr($01)+chr($10)+chr($05)+chr($07)+chr($00)+
           chr($EC)+chr($FC);
    if Fieldbyname('Money').AsString='580' then
start:=chr($FB)+chr($04)+chr($01)+chr($10)+chr($05)+chr($08)+chr($00)+
           chr($E3)+chr($FC);
    if Fieldbyname('Money').AsString='590' then
start:=chr($FB)+chr($04)+chr($01)+chr($10)+chr($05)+chr($09)+chr($00)+
           chr($E2)+chr($FC);
    if Fieldbyname('Money').AsString='600' then
start:=chr($FB)+chr($04)+chr($01)+chr($10)+chr($06)+chr($00)+chr($00)+
           chr($E8)+chr($FC);
    if Fieldbyname('Money').AsString='610' then
start:=chr($FB)+chr($04)+chr($01)+chr($10)+chr($06)+chr($01)+chr($00)+
           chr($E9)+chr($FC);
    if Fieldbyname('Money').AsString='620' then
start:=chr($FB)+chr($04)+chr($01)+chr($10)+chr($06)+chr($02)+chr($00)+
           chr($EA)+chr($FC);
    if Fieldbyname('Money').AsString='630' then
start:=chr($FB)+chr($04)+chr($01)+chr($10)+chr($06)+chr($03)+chr($00)+
           chr($EB)+chr($FC);
    if Fieldbyname('Money').AsString='640' then
start:=chr($FB)+chr($04)+chr($01)+chr($10)+chr($06)+chr($04)+chr($00)+
           chr($EC)+chr($FC);
    if Fieldbyname('Money').AsString='650' then
start:=chr($FB)+chr($04)+chr($01)+chr($10)+chr($06)+chr($05)+chr($00)+
           chr($ED)+chr($FC);
    if Fieldbyname('Money').AsString='660' then
start:=chr($FB)+chr($04)+chr($01)+chr($10)+chr($06)+chr($06)+chr($00)+
           chr($EE)+chr($FC);
    if Fieldbyname('Money').AsString='670' then
start:=chr($FB)+chr($04)+chr($01)+chr($10)+chr($06)+chr($07)+chr($00)+
           chr($EF)+chr($FC);
    if Fieldbyname('Money').AsString='680' then
start:=chr($FB)+chr($04)+chr($01)+chr($10)+chr($06)+chr($08)+chr($00)+
           chr($E0)+chr($FC);
    if Fieldbyname('Money').AsString='690' then
start:=chr($FB)+chr($04)+chr($01)+chr($10)+chr($06)+chr($09)+chr($00)+
           chr($E1)+chr($FC);
    if Fieldbyname('Money').AsString='700' then
start:=chr($FB)+chr($04)+chr($01)+chr($10)+chr($07)+chr($00)+chr($00)+
           chr($E9)+chr($FC);
    if Fieldbyname('Money').AsString='710' then
start:=chr($FB)+chr($04)+chr($01)+chr($10)+chr($07)+chr($01)+chr($00)+
           chr($E8)+chr($FC);
    if Fieldbyname('Money').AsString='720' then
start:=chr($FB)+chr($04)+chr($01)+chr($10)+chr($07)+chr($02)+chr($00)+
           chr($EB)+chr($FC);
    if Fieldbyname('Money').AsString='730' then
start:=chr($FB)+chr($04)+chr($01)+chr($10)+chr($07)+chr($03)+chr($00)+
           chr($EA)+chr($FC);
```

```
    if Fieldbyname('Money').AsString='740' then
start:=chr($FB)+chr($04)+chr($01)+chr($10)+chr($07)+chr($04)+chr($00)+
           chr($ED)+chr($FC);
    if Fieldbyname('Money').AsString='750' then

start:=chr($FB)+chr($04)+chr($01)+chr($10)+chr($07)+chr($05)+chr($00)+
           chr($EC)+chr($FC);
    if Fieldbyname('Money').AsString='760' then
start:=chr($FB)+chr($04)+chr($01)+chr($10)+chr($07)+chr($06)+chr($00)+
           chr($EF)+chr($FC);
    if Fieldbyname('Money').AsString='770' then
start:=chr($FB)+chr($04)+chr($01)+chr($10)+chr($07)+chr($07)+chr($00)+
           chr($EE)+chr($FC);
    if Fieldbyname('Money').AsString='780' then
start:=chr($FB)+chr($04)+chr($01)+chr($10)+chr($07)+chr($08)+chr($00)+
           chr($E1)+chr($FC);
    if Fieldbyname('Money').AsString='790' then
start:=chr($FB)+chr($04)+chr($01)+chr($10)+chr($07)+chr($09)+chr($00)+
           chr($E0)+chr($FC);
    if Fieldbyname('Money').AsString='800' then
start:=chr($FB)+chr($04)+chr($01)+chr($10)+chr($08)+chr($00)+chr($00)+
           chr($E6)+chr($FC);
    if Fieldbyname('Money').AsString='810' then
start:=chr($FB)+chr($04)+chr($01)+chr($10)+chr($08)+chr($01)+chr($00)+
           chr($E7)+chr($FC);
    if Fieldbyname('Money').AsString='820' then
start:=chr($FB)+chr($04)+chr($01)+chr($10)+chr($08)+chr($02)+chr($00)+
           chr($E4)+chr($FC);
    if Fieldbyname('Money').AsString='830' then
start:=chr($FB)+chr($04)+chr($01)+chr($10)+chr($08)+chr($03)+chr($00)+
           chr($E5)+chr($FC);
    if Fieldbyname('Money').AsString='840' then
start:=chr($FB)+chr($04)+chr($01)+chr($10)+chr($08)+chr($04)+chr($00)+
           chr($E2)+chr($FC);
    if Fieldbyname('Money').AsString='850' then
start:=chr($FB)+chr($04)+chr($01)+chr($10)+chr($08)+chr($05)+chr($00)+
           chr($E3)+chr($FC);
    if Fieldbyname('Money').AsString='860' then
start:=chr($FB)+chr($04)+chr($01)+chr($10)+chr($08)+chr($06)+chr($00)+
           chr($E0)+chr($FC);
    if Fieldbyname('Money').AsString='870' then
start:=chr($FB)+chr($04)+chr($01)+chr($10)+chr($08)+chr($07)+chr($00)+
           chr($E1)+chr($FC);
    if Fieldbyname('Money').AsString='880' then
start:=chr($FB)+chr($04)+chr($01)+chr($10)+chr($08)+chr($08)+chr($00)+
           chr($EE)+chr($FC);
    if Fieldbyname('Money').AsString='890' then
start:=chr($FB)+chr($04)+chr($01)+chr($10)+chr($08)+chr($09)+chr($00)+
           chr($EF)+chr($FC);
    if Fieldbyname('Money').AsString='900' then
start:=chr($FB)+chr($04)+chr($01)+chr($10)+chr($09)+chr($00)+chr($00)+
           chr($E7)+chr($FC);
    if Fieldbyname('Money').AsString='910' then
start:=chr($FB)+chr($04)+chr($01)+chr($10)+chr($09)+chr($01)+chr($00)+
           chr($E6)+chr($FC);
    if Fieldbyname('Money').AsString='920' then
start:=chr($FB)+chr($04)+chr($01)+chr($10)+chr($09)+chr($02)+chr($00)+
```

```
            chr($E5)+chr($FC);
    if Fieldbyname('Money').AsString='930' then
start:=chr($FB)+chr($04)+chr($01)+chr($10)+chr($09)+chr($03)+chr($00)+
            chr($E4)+chr($FC);
    if Fieldbyname('Money').AsString='940' then
start:=chr($FB)+chr($04)+chr($01)+chr($10)+chr($09)+chr($04)+chr($00)+
            chr($E3)+chr($FC);
    if Fieldbyname('Money').AsString='950' then
start:=chr($FB)+chr($04)+chr($01)+chr($10)+chr($09)+chr($05)+chr($00)+
            chr($E2)+chr($FC);
    if Fieldbyname('Money').AsString='960' then
start:=chr($FB)+chr($04)+chr($01)+chr($10)+chr($09)+chr($06)+chr($00)+
            chr($E1)+chr($FC);
    if Fieldbyname('Money').AsString='970' then
start:=chr($FB)+chr($04)+chr($01)+chr($10)+chr($09)+chr($07)+chr($00)+
            chr($E0)+chr($FC);
    if Fieldbyname('Money').AsString='980' then
start:=chr($FB)+chr($04)+chr($01)+chr($10)+chr($09)+chr($08)+chr($00)+
            chr($EF)+chr($FC);
    if Fieldbyname('Money').AsString='990' then
start:=chr($FB)+chr($04)+chr($01)+chr($10)+chr($09)+chr($09)+chr($00)+
            chr($EE)+chr($FC);
    if Fieldbyname('Money').AsString='1000' then
start:=chr($FB)+chr($04)+chr($01)+chr($10)+chr($10)+chr($00)+chr($00)+
            chr($FE)+chr($FC);
    Vacomm1.WriteText(start);
    Timer8.Enabled:=true;
    end;
if radiobutton2.Checked then
    begin
    Fieldbyname('Start').AsString := '´èÒ¹ÊØ¢ØÁÇÔ·';
    if Fieldbyname('Money').AsString='0' then
start:=chr($FB)+chr($04)+chr($01)+chr($20)+chr($00)+chr($00)+chr($00)+
            chr($DE)+chr($FC);
    if Fieldbyname('Money').AsString='10' then
start:=chr($FB)+chr($04)+chr($01)+chr($20)+chr($00)+chr($01)+chr($00)+
            chr($DF)+chr($FC);
    if Fieldbyname('Money').AsString='20' then
start:=chr($FB)+chr($04)+chr($01)+chr($20)+chr($00)+chr($02)+chr($00)+
            chr($DC)+chr($FC);
    if Fieldbyname('Money').AsString='30' then
start:=chr($FB)+chr($04)+chr($01)+chr($20)+chr($00)+chr($03)+chr($00)+
            chr($DD)+chr($FC);
    if Fieldbyname('Money').AsString='40' then
start:=chr($FB)+chr($04)+chr($01)+chr($20)+chr($00)+chr($04)+chr($00)+
            chr($DA)+chr($FC);
    if Fieldbyname('Money').AsString='50' then
start:=chr($FB)+chr($04)+chr($01)+chr($20)+chr($00)+chr($05)+chr($00)+
            chr($DB)+chr($FC);
    if Fieldbyname('Money').AsString='60' then
start:=chr($FB)+chr($04)+chr($01)+chr($20)+chr($00)+chr($06)+chr($00)+
            chr($D8)+chr($FC);
    if Fieldbyname('Money').AsString='70' then
start:=chr($FB)+chr($04)+chr($01)+chr($20)+chr($00)+chr($07)+chr($00)+
            chr($D9)+chr($FC);
    if Fieldbyname('Money').AsString='80' then
start:=chr($FB)+chr($04)+chr($01)+chr($20)+chr($00)+chr($08)+chr($00)+
```

```
            chr($D6)+chr($FC);
    if Fieldbyname('Money').AsString='90' then
start:=chr($FB)+chr($04)+chr($01)+chr($20)+chr($00)+chr($09)+chr($00)+
            chr($D7)+chr($FC);
    if Fieldbyname('Money').AsString='100' then
start:=chr($FB)+chr($04)+chr($01)+chr($20)+chr($01)+chr($00)+chr($00)+
            chr($DF)+chr($FC);
    if Fieldbyname('Money').AsString='110' then
start:=chr($FB)+chr($04)+chr($01)+chr($20)+chr($01)+chr($01)+chr($00)+
            chr($DE)+chr($FC);
    if Fieldbyname('Money').AsString='120' then
start:=chr($FB)+chr($04)+chr($01)+chr($20)+chr($01)+chr($02)+chr($00)+
            chr($DD)+chr($FC);
    if Fieldbyname('Money').AsString='130' then
start:=chr($FB)+chr($04)+chr($01)+chr($20)+chr($01)+chr($03)+chr($00)+
            chr($DC)+chr($FC);
    if Fieldbyname('Money').AsString='140' then
start:=chr($FB)+chr($04)+chr($01)+chr($20)+chr($01)+chr($04)+chr($00)+
            chr($DB)+chr($FC);
    if Fieldbyname('Money').AsString='150' then
start:=chr($FB)+chr($04)+chr($01)+chr($20)+chr($01)+chr($05)+chr($00)+
            chr($DA)+chr($FC);
    if Fieldbyname('Money').AsString='160' then
start:=chr($FB)+chr($04)+chr($01)+chr($20)+chr($01)+chr($06)+chr($00)+
            chr($D9)+chr($FC);
    if Fieldbyname('Money').AsString='170' then
start:=chr($FB)+chr($04)+chr($01)+chr($20)+chr($01)+chr($07)+chr($00)+
            chr($D8)+chr($FC);
    if Fieldbyname('Money').AsString='180' then
start:=chr($FB)+chr($04)+chr($01)+chr($20)+chr($01)+chr($08)+chr($00)+
            chr($D7)+chr($FC);
    if Fieldbyname('Money').AsString='190' then
start:=chr($FB)+chr($04)+chr($01)+chr($20)+chr($01)+chr($09)+chr($00)+
            chr($D6)+chr($FC);
    if Fieldbyname('Money').AsString='200' then
start:=chr($FB)+chr($04)+chr($01)+chr($20)+chr($02)+chr($00)+chr($00)+
            chr($DC)+chr($FC);
    if Fieldbyname('Money').AsString='210' then
start:=chr($FB)+chr($04)+chr($01)+chr($20)+chr($02)+chr($01)+chr($00)+
            chr($DD)+chr($FC);
    if Fieldbyname('Money').AsString='220' then
start:=chr($FB)+chr($04)+chr($01)+chr($20)+chr($02)+chr($02)+chr($00)+
            chr($DE)+chr($FC);
    if Fieldbyname('Money').AsString='230' then
start:=chr($FB)+chr($04)+chr($01)+chr($20)+chr($02)+chr($03)+chr($00)+
            chr($DF)+chr($FC);
    if Fieldbyname('Money').AsString='240' then
start:=chr($FB)+chr($04)+chr($01)+chr($20)+chr($02)+chr($04)+chr($00)+
            chr($D8)+chr($FC);
    if Fieldbyname('Money').AsString='250' then
start:=chr($FB)+chr($04)+chr($01)+chr($20)+chr($02)+chr($05)+chr($00)+
            chr($D9)+chr($FC);
    if Fieldbyname('Money').AsString='260' then
start:=chr($FB)+chr($04)+chr($01)+chr($20)+chr($02)+chr($06)+chr($00)+
            chr($DA)+chr($FC);
    if Fieldbyname('Money').AsString='270' then
start:=chr($FB)+chr($04)+chr($01)+chr($20)+chr($02)+chr($07)+chr($00)+
```

```
            chr($DB)+chr($FC);
    if Fieldbyname('Money').AsString='280' then
start:=chr($FB)+chr($04)+chr($01)+chr($20)+chr($02)+chr($08)+chr($00)+
            chr($D4)+chr($FC);
    if Fieldbyname('Money').AsString='290' then
start:=chr($FB)+chr($04)+chr($01)+chr($20)+chr($02)+chr($09)+chr($00)+
            chr($D5)+chr($FC);
    if Fieldbyname('Money').AsString='300' then
start:=chr($FB)+chr($04)+chr($01)+chr($20)+chr($03)+chr($00)+chr($00)+
            chr($DD)+chr($FC);
    if Fieldbyname('Money').AsString='310' then
start:=chr($FB)+chr($04)+chr($01)+chr($20)+chr($03)+chr($01)+chr($00)+
            chr($DC)+chr($FC);
    if Fieldbyname('Money').AsString='320' then
start:=chr($FB)+chr($04)+chr($01)+chr($20)+chr($03)+chr($02)+chr($00)+
            chr($DF)+chr($FC);
    if Fieldbyname('Money').AsString='330' then
start:=chr($FB)+chr($04)+chr($01)+chr($20)+chr($03)+chr($03)+chr($00)+
            chr($DE)+chr($FC);
    if Fieldbyname('Money').AsString='340' then
start:=chr($FB)+chr($04)+chr($01)+chr($20)+chr($03)+chr($04)+chr($00)+
            chr($D9)+chr($FC);
    if Fieldbyname('Money').AsString='350' then
start:=chr($FB)+chr($04)+chr($01)+chr($20)+chr($03)+chr($05)+chr($00)+
            chr($D8)+chr($FC);
    if Fieldbyname('Money').AsString='360' then
start:=chr($FB)+chr($04)+chr($01)+chr($20)+chr($03)+chr($06)+chr($00)+
            chr($DB)+chr($FC);
    if Fieldbyname('Money').AsString='370' then
start:=chr($FB)+chr($04)+chr($01)+chr($20)+chr($03)+chr($07)+chr($00)+
            chr($DA)+chr($FC);
    if Fieldbyname('Money').AsString='380' then
start:=chr($FB)+chr($04)+chr($01)+chr($20)+chr($03)+chr($08)+chr($00)+
            chr($D5)+chr($FC);
    if Fieldbyname('Money').AsString='390' then
start:=chr($FB)+chr($04)+chr($01)+chr($20)+chr($03)+chr($09)+chr($00)+
            chr($D4)+chr($FC);
    if Fieldbyname('Money').AsString='400' then
start:=chr($FB)+chr($04)+chr($01)+chr($20)+chr($04)+chr($00)+chr($00)+
            chr($DA)+chr($FC);
    if Fieldbyname('Money').AsString='410' then
start:=chr($FB)+chr($04)+chr($01)+chr($20)+chr($04)+chr($01)+chr($00)+
            chr($DB)+chr($FC);
    if Fieldbyname('Money').AsString='420' then
start:=chr($FB)+chr($04)+chr($01)+chr($20)+chr($04)+chr($02)+chr($00)+
            chr($D8)+chr($FC);
    if Fieldbyname('Money').AsString='430' then
start:=chr($FB)+chr($04)+chr($01)+chr($20)+chr($04)+chr($03)+chr($00)+
            chr($D9)+chr($FC);
    if Fieldbyname('Money').AsString='440' then
start:=chr($FB)+chr($04)+chr($01)+chr($20)+chr($04)+chr($04)+chr($00)+
            chr($DE)+chr($FC);
    if Fieldbyname('Money').AsString='450' then
start:=chr($FB)+chr($04)+chr($01)+chr($20)+chr($04)+chr($05)+chr($00)+
            chr($DF)+chr($FC);
    if Fieldbyname('Money').AsString='460' then
start:=chr($FB)+chr($04)+chr($01)+chr($20)+chr($04)+chr($06)+chr($00)+
```

```
            chr($DC)+chr($FC);
    if Fieldbyname('Money').AsString='470' then
start:=chr($FB)+chr($04)+chr($01)+chr($20)+chr($04)+chr($07)+chr($00)+
            chr($DD)+chr($FC);
    if Fieldbyname('Money').AsString='480' then
start:=chr($FB)+chr($04)+chr($01)+chr($20)+chr($04)+chr($08)+chr($00)+
            chr($D2)+chr($FC);
    if Fieldbyname('Money').AsString='490' then
start:=chr($FB)+chr($04)+chr($01)+chr($20)+chr($04)+chr($09)+chr($00)+
            chr($D3)+chr($FC);
    if Fieldbyname('Money').AsString='500' then
start:=chr($FB)+chr($04)+chr($01)+chr($20)+chr($05)+chr($00)+chr($00)+
            chr($DB)+chr($FC);
    if Fieldbyname('Money').AsString='510' then
start:=chr($FB)+chr($04)+chr($01)+chr($20)+chr($05)+chr($01)+chr($00)+
            chr($DA)+chr($FC);
    if Fieldbyname('Money').AsString='520' then
start:=chr($FB)+chr($04)+chr($01)+chr($20)+chr($05)+chr($02)+chr($00)+
            chr($D9)+chr($FC);
    if Fieldbyname('Money').AsString='530' then
start:=chr($FB)+chr($04)+chr($01)+chr($20)+chr($05)+chr($03)+chr($00)+
            chr($D8)+chr($FC);
    if Fieldbyname('Money').AsString='540' then
start:=chr($FB)+chr($04)+chr($01)+chr($20)+chr($05)+chr($04)+chr($00)+
            chr($DF)+chr($FC);
    if Fieldbyname('Money').AsString='550' then
start:=chr($FB)+chr($04)+chr($01)+chr($20)+chr($05)+chr($05)+chr($00)+
            chr($DE)+chr($FC);
    if Fieldbyname('Money').AsString='560' then
start:=chr($FB)+chr($04)+chr($01)+chr($20)+chr($05)+chr($06)+chr($00)+
            chr($DD)+chr($FC);
    if Fieldbyname('Money').AsString='570' then
start:=chr($FB)+chr($04)+chr($01)+chr($20)+chr($05)+chr($07)+chr($00)+
            chr($DC)+chr($FC);
    if Fieldbyname('Money').AsString='580' then
start:=chr($FB)+chr($04)+chr($01)+chr($20)+chr($05)+chr($08)+chr($00)+
            chr($D3)+chr($FC);
    if Fieldbyname('Money').AsString='590' then
start:=chr($FB)+chr($04)+chr($01)+chr($20)+chr($05)+chr($09)+chr($00)+
            chr($D2)+chr($FC);
    if Fieldbyname('Money').AsString='600' then
start:=chr($FB)+chr($04)+chr($01)+chr($20)+chr($06)+chr($00)+chr($00)+
            chr($D8)+chr($FC);
    if Fieldbyname('Money').AsString='610' then
start:=chr($FB)+chr($04)+chr($01)+chr($20)+chr($06)+chr($01)+chr($00)+
            chr($D9)+chr($FC);
    if Fieldbyname('Money').AsString='620' then
start:=chr($FB)+chr($04)+chr($01)+chr($20)+chr($06)+chr($02)+chr($00)+
            chr($DA)+chr($FC);
    if Fieldbyname('Money').AsString='630' then
start:=chr($FB)+chr($04)+chr($01)+chr($20)+chr($06)+chr($03)+chr($00)+
            chr($DB)+chr($FC);
    if Fieldbyname('Money').AsString='640' then
start:=chr($FB)+chr($04)+chr($01)+chr($20)+chr($06)+chr($04)+chr($00)+
            chr($DC)+chr($FC);
    if Fieldbyname('Money').AsString='650' then
start:=chr($FB)+chr($04)+chr($01)+chr($20)+chr($06)+chr($05)+chr($00)+
```

```
            chr($DD)+chr($FC);
    if Fieldbyname('Money').AsString='660' then
start:=chr($FB)+chr($04)+chr($01)+chr($20)+chr($06)+chr($06)+chr($00)+
            chr($DE)+chr($FC);
    if Fieldbyname('Money').AsString='670' then
start:=chr($FB)+chr($04)+chr($01)+chr($20)+chr($06)+chr($07)+chr($00)+
            chr($DF)+chr($FC);
    if Fieldbyname('Money').AsString='680' then
start:=chr($FB)+chr($04)+chr($01)+chr($20)+chr($06)+chr($08)+chr($00)+
            chr($D0)+chr($FC);
    if Fieldbyname('Money').AsString='690' then
start:=chr($FB)+chr($04)+chr($01)+chr($20)+chr($06)+chr($09)+chr($00)+
            chr($D1)+chr($FC);
    if Fieldbyname('Money').AsString='700' then
start:=chr($FB)+chr($04)+chr($01)+chr($20)+chr($07)+chr($00)+chr($00)+
            chr($D9)+chr($FC);
    if Fieldbyname('Money').AsString='710' then
start:=chr($FB)+chr($04)+chr($01)+chr($20)+chr($07)+chr($01)+chr($00)+
            chr($D8)+chr($FC);
    if Fieldbyname('Money').AsString='720' then
start:=chr($FB)+chr($04)+chr($01)+chr($20)+chr($07)+chr($02)+chr($00)+
            chr($DB)+chr($FC);
    if Fieldbyname('Money').AsString='730' then
start:=chr($FB)+chr($04)+chr($01)+chr($20)+chr($07)+chr($03)+chr($00)+
            chr($DA)+chr($FC);
    if Fieldbyname('Money').AsString='740' then
start:=chr($FB)+chr($04)+chr($01)+chr($20)+chr($07)+chr($04)+chr($00)+
            chr($DD)+chr($FC);
    if Fieldbyname('Money').AsString='750' then
start:=chr($FB)+chr($04)+chr($01)+chr($20)+chr($07)+chr($05)+chr($00)+
            chr($DC)+chr($FC);
    if Fieldbyname('Money').AsString='760' then
start:=chr($FB)+chr($04)+chr($01)+chr($20)+chr($07)+chr($06)+chr($00)+
            chr($DF)+chr($FC);
    if Fieldbyname('Money').AsString='770' then
start:=chr($FB)+chr($04)+chr($01)+chr($20)+chr($07)+chr($07)+chr($00)+
            chr($DE)+chr($FC);
    if Fieldbyname('Money').AsString='780' then
start:=chr($FB)+chr($04)+chr($01)+chr($20)+chr($07)+chr($08)+chr($00)+
            chr($D1)+chr($FC);
    if Fieldbyname('Money').AsString='790' then
start:=chr($FB)+chr($04)+chr($01)+chr($20)+chr($07)+chr($09)+chr($00)+
            chr($D0)+chr($FC);
    if Fieldbyname('Money').AsString='800' then
start:=chr($FB)+chr($04)+chr($01)+chr($20)+chr($08)+chr($00)+chr($00)+
            chr($D6)+chr($FC);
    if Fieldbyname('Money').AsString='810' then
start:=chr($FB)+chr($04)+chr($01)+chr($20)+chr($08)+chr($01)+chr($00)+
            chr($D7)+chr($FC);
    if Fieldbyname('Money').AsString='820' then
start:=chr($FB)+chr($04)+chr($01)+chr($20)+chr($08)+chr($02)+chr($00)+
            chr($D4)+chr($FC);
    if Fieldbyname('Money').AsString='830' then
start:=chr($FB)+chr($04)+chr($01)+chr($20)+chr($08)+chr($03)+chr($00)+
            chr($D5)+chr($FC);
    if Fieldbyname('Money').AsString='840' then
start:=chr($FB)+chr($04)+chr($01)+chr($20)+chr($08)+chr($04)+chr($00)+
```

```
            chr($D2)+chr($FC);
    if Fieldbyname('Money').AsString='850' then
start:=chr($FB)+chr($04)+chr($01)+chr($20)+chr($08)+chr($05)+chr($00)+
            chr($D3)+chr($FC);
    if Fieldbyname('Money').AsString='860' then
start:=chr($FB)+chr($04)+chr($01)+chr($20)+chr($08)+chr($06)+chr($00)+
            chr($D0)+chr($FC);
    if Fieldbyname('Money').AsString='870' then
start:=chr($FB)+chr($04)+chr($01)+chr($20)+chr($08)+chr($07)+chr($00)+
            chr($D1)+chr($FC);
    if Fieldbyname('Money').AsString='880' then
start:=chr($FB)+chr($04)+chr($01)+chr($20)+chr($08)+chr($08)+chr($00)+
            chr($DE)+chr($FC);
    if Fieldbyname('Money').AsString='890' then
start:=chr($FB)+chr($04)+chr($01)+chr($20)+chr($08)+chr($09)+chr($00)+
            chr($DF)+chr($FC);
    if Fieldbyname('Money').AsString='900' then
start:=chr($FB)+chr($04)+chr($01)+chr($20)+chr($09)+chr($00)+chr($00)+
            chr($D7)+chr($FC);
    if Fieldbyname('Money').AsString='910' then
start:=chr($FB)+chr($04)+chr($01)+chr($20)+chr($09)+chr($01)+chr($00)+
            chr($D6)+chr($FC);
    if Fieldbyname('Money').AsString='920' then
start:=chr($FB)+chr($04)+chr($01)+chr($20)+chr($09)+chr($02)+chr($00)+
            chr($D5)+chr($FC);
    if Fieldbyname('Money').AsString='930' then
start:=chr($FB)+chr($04)+chr($01)+chr($20)+chr($09)+chr($03)+chr($00)+
            chr($D4)+chr($FC);
    if Fieldbyname('Money').AsString='940' then
start:=chr($FB)+chr($04)+chr($01)+chr($20)+chr($09)+chr($04)+chr($00)+
            chr($D3)+chr($FC);
    if Fieldbyname('Money').AsString='950' then
start:=chr($FB)+chr($04)+chr($01)+chr($20)+chr($09)+chr($05)+chr($00)+
            chr($D2)+chr($FC);
    if Fieldbyname('Money').AsString='960' then
start:=chr($FB)+chr($04)+chr($01)+chr($20)+chr($09)+chr($06)+chr($00)+
            chr($D1)+chr($FC);
    if Fieldbyname('Money').AsString='970' then
start:=chr($FB)+chr($04)+chr($01)+chr($20)+chr($09)+chr($07)+chr($00)+
            chr($D0)+chr($FC);
    if Fieldbyname('Money').AsString='980' then
start:=chr($FB)+chr($04)+chr($01)+chr($20)+chr($09)+chr($08)+chr($00)+
            chr($DF)+chr($FC);
    if Fieldbyname('Money').AsString='990' then
start:=chr($FB)+chr($04)+chr($01)+chr($20)+chr($09)+chr($09)+chr($00)+
            chr($DE)+chr($FC);
    if Fieldbyname('Money').AsString='1000' then
start:=chr($FB)+chr($04)+chr($01)+chr($20)+chr($10)+chr($00)+chr($00)+
            chr($CE)+chr($FC);
    Vacomm1.WriteText(start);
    Timer8.Enabled:=true;
    end;
if radiobutton3.Checked then
    begin
    Fieldbyname('Start').AsString := '´èÒ¹ÍÒ¨³Ã§¤ì';
    if Fieldbyname('Money').AsString='0' then
start:=chr($FB)+chr($04)+chr($01)+chr($30)+chr($00)+chr($00)+chr($00)+
```

```
            chr($CE)+chr($FC);
    if Fieldbyname('Money').AsString='10' then
start:=chr($FB)+chr($04)+chr($01)+chr($30)+chr($00)+chr($01)+chr($00)+
            chr($CF)+chr($FC);
    if Fieldbyname('Money').AsString='20' then
start:=chr($FB)+chr($04)+chr($01)+chr($30)+chr($00)+chr($02)+chr($00)+
            chr($CC)+chr($FC);
    if Fieldbyname('Money').AsString='30' then
start:=chr($FB)+chr($04)+chr($01)+chr($30)+chr($00)+chr($03)+chr($00)+
            chr($CD)+chr($FC);
    if Fieldbyname('Money').AsString='40' then
start:=chr($FB)+chr($04)+chr($01)+chr($30)+chr($00)+chr($04)+chr($00)+
            chr($CA)+chr($FC);
    if Fieldbyname('Money').AsString='50' then
start:=chr($FB)+chr($04)+chr($01)+chr($30)+chr($00)+chr($05)+chr($00)+
            chr($CB)+chr($FC);
    if Fieldbyname('Money').AsString='60' then
start:=chr($FB)+chr($04)+chr($01)+chr($30)+chr($00)+chr($06)+chr($00)+
            chr($C8)+chr($FC);
    if Fieldbyname('Money').AsString='70' then
start:=chr($FB)+chr($04)+chr($01)+chr($30)+chr($00)+chr($07)+chr($00)+
            chr($C9)+chr($FC);
    if Fieldbyname('Money').AsString='80' then
start:=chr($FB)+chr($04)+chr($01)+chr($30)+chr($00)+chr($08)+chr($00)+
            chr($C6)+chr($FC);
    if Fieldbyname('Money').AsString='90' then
start:=chr($FB)+chr($04)+chr($01)+chr($30)+chr($00)+chr($09)+chr($00)+
            chr($C7)+chr($FC);
    if Fieldbyname('Money').AsString='100' then
start:=chr($FB)+chr($04)+chr($01)+chr($30)+chr($01)+chr($00)+chr($00)+
            chr($CF)+chr($FC);
    if Fieldbyname('Money').AsString='110' then
start:=chr($FB)+chr($04)+chr($01)+chr($30)+chr($01)+chr($01)+chr($00)+
            chr($CE)+chr($FC);
    if Fieldbyname('Money').AsString='120' then
start:=chr($FB)+chr($04)+chr($01)+chr($30)+chr($01)+chr($02)+chr($00)+
            chr($CD)+chr($FC);
    if Fieldbyname('Money').AsString='130' then
start:=chr($FB)+chr($04)+chr($01)+chr($30)+chr($01)+chr($03)+chr($00)+
            chr($CC)+chr($FC);
    if Fieldbyname('Money').AsString='140' then
start:=chr($FB)+chr($04)+chr($01)+chr($30)+chr($01)+chr($04)+chr($00)+
            chr($CB)+chr($FC);
    if Fieldbyname('Money').AsString='150' then
start:=chr($FB)+chr($04)+chr($01)+chr($30)+chr($01)+chr($05)+chr($00)+
            chr($CA)+chr($FC);
    if Fieldbyname('Money').AsString='160' then
start:=chr($FB)+chr($04)+chr($01)+chr($30)+chr($01)+chr($06)+chr($00)+
            chr($C9)+chr($FC);
    if Fieldbyname('Money').AsString='170' then
start:=chr($FB)+chr($04)+chr($01)+chr($30)+chr($01)+chr($07)+chr($00)+
            chr($C8)+chr($FC);
    if Fieldbyname('Money').AsString='180' then
start:=chr($FB)+chr($04)+chr($01)+chr($30)+chr($01)+chr($08)+chr($00)+
            chr($C7)+chr($FC);
    if Fieldbyname('Money').AsString='190' then
start:=chr($FB)+chr($04)+chr($01)+chr($30)+chr($01)+chr($09)+chr($00)+
```

```
            chr($C6)+chr($FC);
    if Fieldbyname('Money').AsString='200' then
start:=chr($FB)+chr($04)+chr($01)+chr($30)+chr($02)+chr($00)+chr($00)+
            chr($CC)+chr($FC);
    if Fieldbyname('Money').AsString='210' then
start:=chr($FB)+chr($04)+chr($01)+chr($30)+chr($02)+chr($01)+chr($00)+
            chr($CD)+chr($FC);
    if Fieldbyname('Money').AsString='220' then
start:=chr($FB)+chr($04)+chr($01)+chr($30)+chr($02)+chr($02)+chr($00)+
            chr($CE)+chr($FC);
    if Fieldbyname('Money').AsString='230' then
start:=chr($FB)+chr($04)+chr($01)+chr($30)+chr($02)+chr($03)+chr($00)+
            chr($CF)+chr($FC);
    if Fieldbyname('Money').AsString='240' then
start:=chr($FB)+chr($04)+chr($01)+chr($30)+chr($02)+chr($04)+chr($00)+
            chr($C8)+chr($FC);
    if Fieldbyname('Money').AsString='250' then
start:=chr($FB)+chr($04)+chr($01)+chr($30)+chr($02)+chr($05)+chr($00)+
            chr($C9)+chr($FC);
    if Fieldbyname('Money').AsString='260' then
start:=chr($FB)+chr($04)+chr($01)+chr($30)+chr($02)+chr($06)+chr($00)+
            chr($CA)+chr($FC);
    if Fieldbyname('Money').AsString='270' then
start:=chr($FB)+chr($04)+chr($01)+chr($30)+chr($02)+chr($07)+chr($00)+
            chr($CB)+chr($FC);
    if Fieldbyname('Money').AsString='280' then
start:=chr($FB)+chr($04)+chr($01)+chr($30)+chr($02)+chr($08)+chr($00)+
            chr($C4)+chr($FC);
    if Fieldbyname('Money').AsString='290' then
start:=chr($FB)+chr($04)+chr($01)+chr($30)+chr($02)+chr($09)+chr($00)+
            chr($C5)+chr($FC);
    if Fieldbyname('Money').AsString='300' then
start:=chr($FB)+chr($04)+chr($01)+chr($30)+chr($03)+chr($00)+chr($00)+
            chr($CD)+chr($FC);
    if Fieldbyname('Money').AsString='310' then
start:=chr($FB)+chr($04)+chr($01)+chr($30)+chr($03)+chr($01)+chr($00)+
            chr($CC)+chr($FC);
    if Fieldbyname('Money').AsString='320' then
start:=chr($FB)+chr($04)+chr($01)+chr($30)+chr($03)+chr($02)+chr($00)+
            chr($CF)+chr($FC);
    if Fieldbyname('Money').AsString='330' then
start:=chr($FB)+chr($04)+chr($01)+chr($30)+chr($03)+chr($03)+chr($00)+
            chr($CE)+chr($FC);
    if Fieldbyname('Money').AsString='340' then
start:=chr($FB)+chr($04)+chr($01)+chr($30)+chr($03)+chr($04)+chr($00)+
            chr($C9)+chr($FC);
    if Fieldbyname('Money').AsString='350' then
start:=chr($FB)+chr($04)+chr($01)+chr($30)+chr($03)+chr($05)+chr($00)+
            chr($C8)+chr($FC);
    if Fieldbyname('Money').AsString='360' then
start:=chr($FB)+chr($04)+chr($01)+chr($30)+chr($03)+chr($06)+chr($00)+
            chr($CB)+chr($FC);
    if Fieldbyname('Money').AsString='370' then
start:=chr($FB)+chr($04)+chr($01)+chr($30)+chr($03)+chr($07)+chr($00)+
            chr($CA)+chr($FC);
    if Fieldbyname('Money').AsString='380' then
start:=chr($FB)+chr($04)+chr($01)+chr($30)+chr($03)+chr($08)+chr($00)+
```

```
            chr($C5)+chr($FC);
    if Fieldbyname('Money').AsString='390' then
start:=chr($FB)+chr($04)+chr($01)+chr($30)+chr($03)+chr($09)+chr($00)+
            chr($C4)+chr($FC);
    if Fieldbyname('Money').AsString='400' then
start:=chr($FB)+chr($04)+chr($01)+chr($30)+chr($04)+chr($00)+chr($00)+
            chr($CA)+chr($FC);
    if Fieldbyname('Money').AsString='410' then
start:=chr($FB)+chr($04)+chr($01)+chr($30)+chr($04)+chr($01)+chr($00)+
            chr($CB)+chr($FC);
    if Fieldbyname('Money').AsString='420' then
start:=chr($FB)+chr($04)+chr($01)+chr($30)+chr($04)+chr($02)+chr($00)+
            chr($C8)+chr($FC);
    if Fieldbyname('Money').AsString='430' then
start:=chr($FB)+chr($04)+chr($01)+chr($30)+chr($04)+chr($03)+chr($00)+
            chr($C9)+chr($FC);
    if Fieldbyname('Money').AsString='440' then
start:=chr($FB)+chr($04)+chr($01)+chr($30)+chr($04)+chr($04)+chr($00)+
            chr($CE)+chr($FC);
    if Fieldbyname('Money').AsString='450' then
start:=chr($FB)+chr($04)+chr($01)+chr($30)+chr($04)+chr($05)+chr($00)+
            chr($CF)+chr($FC);
    if Fieldbyname('Money').AsString='460' then
start:=chr($FB)+chr($04)+chr($01)+chr($30)+chr($04)+chr($06)+chr($00)+
            chr($CC)+chr($FC);
    if Fieldbyname('Money').AsString='470' then
start:=chr($FB)+chr($04)+chr($01)+chr($30)+chr($04)+chr($07)+chr($00)+
            chr($CD)+chr($FC);
    if Fieldbyname('Money').AsString='480' then
start:=chr($FB)+chr($04)+chr($01)+chr($30)+chr($04)+chr($08)+chr($00)+
            chr($C2)+chr($FC);
    if Fieldbyname('Money').AsString='490' then
start:=chr($FB)+chr($04)+chr($01)+chr($30)+chr($04)+chr($09)+chr($00)+
            chr($C3)+chr($FC);
    if Fieldbyname('Money').AsString='500' then
start:=chr($FB)+chr($04)+chr($01)+chr($30)+chr($05)+chr($00)+chr($00)+
            chr($CB)+chr($FC);
    if Fieldbyname('Money').AsString='510' then
start:=chr($FB)+chr($04)+chr($01)+chr($30)+chr($05)+chr($01)+chr($00)+
            chr($CA)+chr($FC);
    if Fieldbyname('Money').AsString='520' then
start:=chr($FB)+chr($04)+chr($01)+chr($30)+chr($05)+chr($02)+chr($00)+
            chr($C9)+chr($FC);
    if Fieldbyname('Money').AsString='530' then
start:=chr($FB)+chr($04)+chr($01)+chr($30)+chr($05)+chr($03)+chr($00)+
            chr($C8)+chr($FC);
    if Fieldbyname('Money').AsString='540' then
start:=chr($FB)+chr($04)+chr($01)+chr($30)+chr($05)+chr($04)+chr($00)+
            chr($CF)+chr($FC);
    if Fieldbyname('Money').AsString='550' then
start:=chr($FB)+chr($04)+chr($01)+chr($30)+chr($05)+chr($05)+chr($00)+
            chr($CE)+chr($FC);
    if Fieldbyname('Money').AsString='560' then
start:=chr($FB)+chr($04)+chr($01)+chr($30)+chr($05)+chr($06)+chr($00)+
            chr($CD)+chr($FC);
    if Fieldbyname('Money').AsString='570' then
start:=chr($FB)+chr($04)+chr($01)+chr($30)+chr($05)+chr($07)+chr($00)+
```

```
            chr($CC)+chr($FC);
    if Fieldbyname('Money').AsString='580' then
start:=chr($FB)+chr($04)+chr($01)+chr($30)+chr($05)+chr($08)+chr($00)+
            chr($C3)+chr($FC);
    if Fieldbyname('Money').AsString='590' then
start:=chr($FB)+chr($04)+chr($01)+chr($30)+chr($05)+chr($09)+chr($00)+
            chr($C2)+chr($FC);
    if Fieldbyname('Money').AsString='600' then
start:=chr($FB)+chr($04)+chr($01)+chr($30)+chr($06)+chr($00)+chr($00)+
            chr($C8)+chr($FC);
    if Fieldbyname('Money').AsString='610' then
start:=chr($FB)+chr($04)+chr($01)+chr($30)+chr($06)+chr($01)+chr($00)+
            chr($C9)+chr($FC);
    if Fieldbyname('Money').AsString='620' then
start:=chr($FB)+chr($04)+chr($01)+chr($30)+chr($06)+chr($02)+chr($00)+
            chr($CA)+chr($FC);
    if Fieldbyname('Money').AsString='630' then
start:=chr($FB)+chr($04)+chr($01)+chr($30)+chr($06)+chr($03)+chr($00)+
            chr($CB)+chr($FC);
    if Fieldbyname('Money').AsString='640' then
start:=chr($FB)+chr($04)+chr($01)+chr($30)+chr($06)+chr($04)+chr($00)+
            chr($CC)+chr($FC);
    if Fieldbyname('Money').AsString='650' then
start:=chr($FB)+chr($04)+chr($01)+chr($30)+chr($06)+chr($05)+chr($00)+
            chr($CD)+chr($FC);
    if Fieldbyname('Money').AsString='660' then
start:=chr($FB)+chr($04)+chr($01)+chr($30)+chr($06)+chr($06)+chr($00)+
            chr($CE)+chr($FC);
    if Fieldbyname('Money').AsString='670' then
start:=chr($FB)+chr($04)+chr($01)+chr($30)+chr($06)+chr($07)+chr($00)+
            chr($CF)+chr($FC);
    if Fieldbyname('Money').AsString='680' then
start:=chr($FB)+chr($04)+chr($01)+chr($30)+chr($06)+chr($08)+chr($00)+
            chr($C0)+chr($FC);
    if Fieldbyname('Money').AsString='690' then
start:=chr($FB)+chr($04)+chr($01)+chr($30)+chr($06)+chr($09)+chr($00)+
            chr($C1)+chr($FC);
    if Fieldbyname('Money').AsString='700' then
start:=chr($FB)+chr($04)+chr($01)+chr($30)+chr($07)+chr($00)+chr($00)+
            chr($C9)+chr($FC);
    if Fieldbyname('Money').AsString='710' then
start:=chr($FB)+chr($04)+chr($01)+chr($30)+chr($07)+chr($01)+chr($00)+
            chr($C8)+chr($FC);
    if Fieldbyname('Money').AsString='720' then
start:=chr($FB)+chr($04)+chr($01)+chr($30)+chr($07)+chr($02)+chr($00)+
            chr($CB)+chr($FC);
    if Fieldbyname('Money').AsString='730' then
start:=chr($FB)+chr($04)+chr($01)+chr($30)+chr($07)+chr($03)+chr($00)+
            chr($CA)+chr($FC);
    if Fieldbyname('Money').AsString='740' then
start:=chr($FB)+chr($04)+chr($01)+chr($30)+chr($07)+chr($04)+chr($00)+
            chr($CD)+chr($FC);
    if Fieldbyname('Money').AsString='750' then
start:=chr($FB)+chr($04)+chr($01)+chr($30)+chr($07)+chr($05)+chr($00)+
            chr($CC)+chr($FC);
    if Fieldbyname('Money').AsString='760' then
start:=chr($FB)+chr($04)+chr($01)+chr($30)+chr($07)+chr($06)+chr($00)+
```

```
            chr($CF)+chr($FC);
    .if Fieldbyname('Money').AsString='770' then
start:=chr($FB)+chr($04)+chr($01)+chr($30)+chr($07)+chr($07)+chr($00)+
            chr($CE)+chr($FC);
     if Fieldbyname('Money').AsString='780' then
start:=chr($FB)+chr($04)+chr($01)+chr($30)+chr($07)+chr($08)+chr($00)+
            chr($C1)+chr($FC);
     if Fieldbyname('Money').AsString='790' then
start:=chr($FB)+chr($04)+chr($01)+chr($30)+chr($07)+chr($09)+chr($00)+
            chr($C0)+chr($FC);
     if Fieldbyname('Money').AsString='800' then
start:=chr($FB)+chr($04)+chr($01)+chr($30)+chr($08)+chr($00)+chr($00)+
            chr($C6)+chr($FC);
     if Fieldbyname('Money').AsString='810' then
start:=chr($FB)+chr($04)+chr($01)+chr($30)+chr($08)+chr($01)+chr($00)+
            chr($C7)+chr($FC);
     if Fieldbyname('Money').AsString='820' then
start:=chr($FB)+chr($04)+chr($01)+chr($30)+chr($08)+chr($02)+chr($00)+
            chr($C4)+chr($FC);
     if Fieldbyname('Money').AsString='830' then
start:=chr($FB)+chr($04)+chr($01)+chr($30)+chr($08)+chr($03)+chr($00)+
            chr($C5)+chr($FC);
     if Fieldbyname('Money').AsString='840' then
start:=chr($FB)+chr($04)+chr($01)+chr($30)+chr($08)+chr($04)+chr($00)+
            chr($C2)+chr($FC);
     if Fieldbyname('Money').AsString='850' then
start:=chr($FB)+chr($04)+chr($01)+chr($30)+chr($08)+chr($05)+chr($00)+
            chr($C3)+chr($FC);
     if Fieldbyname('Money').AsString='860' then
start:=chr($FB)+chr($04)+chr($01)+chr($30)+chr($08)+chr($06)+chr($00)+
            chr($C0)+chr($FC);
     if Fieldbyname('Money').AsString='870' then
start:=chr($FB)+chr($04)+chr($01)+chr($30)+chr($08)+chr($07)+chr($00)+
            chr($C1)+chr($FC);
     if Fieldbyname('Money').AsString='880' then
start:=chr($FB)+chr($04)+chr($01)+chr($30)+chr($08)+chr($08)+chr($00)+
            chr($CE)+chr($FC);
     if Fieldbyname('Money').AsString='890' then
start:=chr($FB)+chr($04)+chr($01)+chr($30)+chr($08)+chr($09)+chr($00)+
            chr($CF)+chr($FC);
     if Fieldbyname('Money').AsString='900' then
start:=chr($FB)+chr($04)+chr($01)+chr($30)+chr($09)+chr($00)+chr($00)+
            chr($C7)+chr($FC);
     if Fieldbyname('Money').AsString='910' then
start:=chr($FB)+chr($04)+chr($01)+chr($30)+chr($09)+chr($01)+chr($00)+
            chr($C6)+chr($FC);
     if Fieldbyname('Money').AsString='920' then
start:=chr($FB)+chr($04)+chr($01)+chr($30)+chr($09)+chr($02)+chr($00)+
            chr($C5)+chr($FC);
     if Fieldbyname('Money').AsString='930' then
start:=chr($FB)+chr($04)+chr($01)+chr($30)+chr($09)+chr($03)+chr($00)+
            chr($C4)+chr($FC);
     if Fieldbyname('Money').AsString='940' then
start:=chr($FB)+chr($04)+chr($01)+chr($30)+chr($09)+chr($04)+chr($00)+
            chr($C3)+chr($FC);
     if Fieldbyname('Money').AsString='950' then
start:=chr($FB)+chr($04)+chr($01)+chr($30)+chr($09)+chr($05)+chr($00)+
```

```
            chr($C2)+chr($FC);
    if Fieldbyname('Money').AsString='960' then
start:=chr($FB)+chr($04)+chr($01)+chr($30)+chr($09)+chr($06)+chr($00)+
            chr($C1)+chr($FC);
    if Fieldbyname('Money').AsString='970' then
start:=chr($FB)+chr($04)+chr($01)+chr($30)+chr($09)+chr($07)+chr($00)+
            chr($C0)+chr($FC);
    if Fieldbyname('Money').AsString='980' then
start:=chr($FB)+chr($04)+chr($01)+chr($30)+chr($09)+chr($08)+chr($00)+
            chr($CF)+chr($FC);
    if Fieldbyname('Money').AsString='990' then
start:=chr($FB)+chr($04)+chr($01)+chr($30)+chr($09)+chr($09)+chr($00)+
            chr($CE)+chr($FC);
    if Fieldbyname('Money').AsString='1000' then
start:=chr($FB)+chr($04)+chr($01)+chr($30)+chr($10)+chr($00)+chr($00)+
            chr($DE)+chr($FC);
    Vacomm1.WriteText(start);
    Timer8.Enabled:=true;
    end;

if radiobutton4.Checked then
    begin
    Fieldbyname('Start').AsString := '´èÒ¹ÊØ¢ØÁÇÔ· 62';
    if Fieldbyname('Money').AsString='0' then
start:=chr($FB)+chr($04)+chr($01)+chr($40)+chr($00)+chr($00)+chr($00)+
            chr($BE)+chr($FC);
    if Fieldbyname('Money').AsString='10' then
start:=chr($FB)+chr($04)+chr($01)+chr($40)+chr($00)+chr($01)+chr($00)+
            chr($BF)+chr($FC);
    if Fieldbyname('Money').AsString='20' then
start:=chr($FB)+chr($04)+chr($01)+chr($40)+chr($00)+chr($02)+chr($00)+
            chr($BC)+chr($FC);
    if Fieldbyname('Money').AsString='30' then
start:=chr($FB)+chr($04)+chr($01)+chr($40)+chr($00)+chr($03)+chr($00)+
            chr($BD)+chr($FC);
    if Fieldbyname('Money').AsString='40' then
start:=chr($FB)+chr($04)+chr($01)+chr($40)+chr($00)+chr($04)+chr($00)+
            chr($BA)+chr($FC);
    if Fieldbyname('Money').AsString='50' then
start:=chr($FB)+chr($04)+chr($01)+chr($40)+chr($00)+chr($05)+chr($00)+
            chr($BB)+chr($FC);
    if Fieldbyname('Money').AsString='60' then
start:=chr($FB)+chr($04)+chr($01)+chr($40)+chr($00)+chr($06)+chr($00)+
            chr($B8)+chr($FC);
    if Fieldbyname('Money').AsString='70' then
start:=chr($FB)+chr($04)+chr($01)+chr($40)+chr($00)+chr($07)+chr($00)+
            chr($B9)+chr($FC);
    if Fieldbyname('Money').AsString='80' then
start:=chr($FB)+chr($04)+chr($01)+chr($40)+chr($00)+chr($08)+chr($00)+
            chr($B6)+chr($FC);
    if Fieldbyname('Money').AsString='90' then
start:=chr($FB)+chr($04)+chr($01)+chr($40)+chr($00)+chr($09)+chr($00)+
            chr($B7)+chr($FC);
    if Fieldbyname('Money').AsString='100' then
start:=chr($FB)+chr($04)+chr($01)+chr($40)+chr($01)+chr($00)+chr($00)+
            chr($BF)+chr($FC);
```

```
    if Fieldbyname('Money').AsString='110' then
start:=chr($FB)+chr($04)+chr($01)+chr($40)+chr($01)+chr($01)+chr($00)+
           chr($BE)+chr($FC);
    if Fieldbyname('Money').AsString='120' then
start:=chr($FB)+chr($04)+chr($01)+chr($40)+chr($01)+chr($02)+chr($00)+
           chr($BD)+chr($FC);
    if Fieldbyname('Money').AsString='130' then
start:=chr($FB)+chr($04)+chr($01)+chr($40)+chr($01)+chr($03)+chr($00)+
           chr($BC)+chr($FC);
    if Fieldbyname('Money').AsString='140' then
start:=chr($FB)+chr($04)+chr($01)+chr($40)+chr($01)+chr($04)+chr($00)+
           chr($BB)+chr($FC);
    if Fieldbyname('Money').AsString='150' then
start:=chr($FB)+chr($04)+chr($01)+chr($40)+chr($01)+chr($05)+chr($00)+
           chr($BA)+chr($FC);
    if Fieldbyname('Money').AsString='160' then
start:=chr($FB)+chr($04)+chr($01)+chr($40)+chr($01)+chr($06)+chr($00)+
           chr($B9)+chr($FC);
    if Fieldbyname('Money').AsString='170' then
start:=chr($FB)+chr($04)+chr($01)+chr($40)+chr($01)+chr($07)+chr($00)+
           chr($B8)+chr($FC);
    if Fieldbyname('Money').AsString='180' then
start:=chr($FB)+chr($04)+chr($01)+chr($40)+chr($01)+chr($08)+chr($00)+
           chr($B7)+chr($FC);
    if Fieldbyname('Money').AsString='190' then
start:=chr($FB)+chr($04)+chr($01)+chr($40)+chr($01)+chr($09)+chr($00)+
           chr($B6)+chr($FC);
    if Fieldbyname('Money').AsString='200' then
start:=chr($FB)+chr($04)+chr($01)+chr($40)+chr($02)+chr($00)+chr($00)+
           chr($BC)+chr($FC);
    if Fieldbyname('Money').AsString='210' then
start:=chr($FB)+chr($04)+chr($01)+chr($40)+chr($02)+chr($01)+chr($00)+
           chr($BD)+chr($FC);
    if Fieldbyname('Money').AsString='220' then
start:=chr($FB)+chr($04)+chr($01)+chr($40)+chr($02)+chr($02)+chr($00)+
           chr($BE)+chr($FC);
    if Fieldbyname('Money').AsString='230' then
start:=chr($FB)+chr($04)+chr($01)+chr($40)+chr($02)+chr($03)+chr($00)+
           chr($BF)+chr($FC);
    if Fieldbyname('Money').AsString='240' then
start:=chr($FB)+chr($04)+chr($01)+chr($40)+chr($02)+chr($04)+chr($00)+
           chr($B8)+chr($FC);
    if Fieldbyname('Money').AsString='250' then
start:=chr($FB)+chr($04)+chr($01)+chr($40)+chr($02)+chr($05)+chr($00)+
           chr($B9)+chr($FC);
    if Fieldbyname('Money').AsString='260' then
start:=chr($FB)+chr($04)+chr($01)+chr($40)+chr($02)+chr($06)+chr($00)+
           chr($BA)+chr($FC);
    if Fieldbyname('Money').AsString='270' then
start:=chr($FB)+chr($04)+chr($01)+chr($40)+chr($02)+chr($07)+chr($00)+
           chr($BB)+chr($FC);
    if Fieldbyname('Money').AsString='280' then
start:=chr($FB)+chr($04)+chr($01)+chr($40)+chr($02)+chr($08)+chr($00)+
           chr($B4)+chr($FC);
    if Fieldbyname('Money').AsString='290' then
start:=chr($FB)+chr($04)+chr($01)+chr($40)+chr($02)+chr($09)+chr($00)+
           chr($B5)+chr($FC);
```

```
    if Fieldbyname('Money').AsString='300' then
start:=chr($FB)+chr($04)+chr($01)+chr($40)+chr($03)+chr($00)+chr($00)+
           chr($BD)+chr($FC);
    if Fieldbyname('Money').AsString='310' then
start:=chr($FB)+chr($04)+chr($01)+chr($40)+chr($03)+chr($01)+chr($00)+
           chr($BC)+chr($FC);
    if Fieldbyname('Money').AsString='320' then
start:=chr($FB)+chr($04)+chr($01)+chr($40)+chr($03)+chr($02)+chr($00)+
           chr($BF)+chr($FC);
    if Fieldbyname('Money').AsString='330' then
start:=chr($FB)+chr($04)+chr($01)+chr($40)+chr($03)+chr($03)+chr($00)+
           chr($BE)+chr($FC);
    if Fieldbyname('Money').AsString='340' then
start:=chr($FB)+chr($04)+chr($01)+chr($40)+chr($03)+chr($04)+chr($00)+
           chr($B9)+chr($FC);
    if Fieldbyname('Money').AsString='350' then
start:=chr($FB)+chr($04)+chr($01)+chr($40)+chr($03)+chr($05)+chr($00)+
           chr($B8)+chr($FC);
    if Fieldbyname('Money').AsString='360' then
start:=chr($FB)+chr($04)+chr($01)+chr($40)+chr($03)+chr($06)+chr($00)+
           chr($BB)+chr($FC);
    if Fieldbyname('Money').AsString='370' then
start:=chr($FB)+chr($04)+chr($01)+chr($40)+chr($03)+chr($07)+chr($00)+
           chr($BA)+chr($FC);
    if Fieldbyname('Money').AsString='380' then
start:=chr($FB)+chr($04)+chr($01)+chr($40)+chr($03)+chr($08)+chr($00)+
           chr($B5)+chr($FC);
    if Fieldbyname('Money').AsString='390' then
start:=chr($FB)+chr($04)+chr($01)+chr($40)+chr($03)+chr($09)+chr($00)+
           chr($B4)+chr($FC);
    if Fieldbyname('Money').AsString='400' then
start:=chr($FB)+chr($04)+chr($01)+chr($40)+chr($04)+chr($00)+chr($00)+
           chr($BA)+chr($FC);
    if Fieldbyname('Money').AsString='410' then
start:=chr($FB)+chr($04)+chr($01)+chr($40)+chr($04)+chr($01)+chr($00)+
           chr($BB)+chr($FC);
    if Fieldbyname('Money').AsString='420' then
start:=chr($FB)+chr($04)+chr($01)+chr($40)+chr($04)+chr($02)+chr($00)+
           chr($B8)+chr($FC);
    if Fieldbyname('Money').AsString='430' then
start:=chr($FB)+chr($04)+chr($01)+chr($40)+chr($04)+chr($03)+chr($00)+
           chr($B9)+chr($FC);
    if Fieldbyname('Money').AsString='440' then
start:=chr($FB)+chr($04)+chr($01)+chr($40)+chr($04)+chr($04)+chr($00)+
           chr($BE)+chr($FC);
    if Fieldbyname('Money').AsString='450' then
start:=chr($FB)+chr($04)+chr($01)+chr($40)+chr($04)+chr($05)+chr($00)+
           chr($BF)+chr($FC);
    if Fieldbyname('Money').AsString='460' then
start:=chr($FB)+chr($04)+chr($01)+chr($40)+chr($04)+chr($06)+chr($00)+
           chr($BC)+chr($FC);
    if Fieldbyname('Money').AsString='470' then
start:=chr($FB)+chr($04)+chr($01)+chr($40)+chr($04)+chr($07)+chr($00)+
           chr($BD)+chr($FC);
    if Fieldbyname('Money').AsString='480' then
start:=chr($FB)+chr($04)+chr($01)+chr($40)+chr($04)+chr($08)+chr($00)+
           chr($B2)+chr($FC);
```

```
    if Fieldbyname('Money').AsString='490' then
start:=chr($FB)+chr($04)+chr($01)+chr($40)+chr($04)+chr($09)+chr($00)+
           chr($B3)+chr($FC);
    if Fieldbyname('Money').AsString='500' then
start:=chr($FB)+chr($04)+chr($01)+chr($40)+chr($05)+chr($00)+chr($00)+
           chr($BB)+chr($FC);
    if Fieldbyname('Money').AsString='510' then
start:=chr($FB)+chr($04)+chr($01)+chr($40)+chr($05)+chr($01)+chr($00)+
           chr($BA)+chr($FC);
    if Fieldbyname('Money').AsString='520' then
start:=chr($FB)+chr($04)+chr($01)+chr($40)+chr($05)+chr($02)+chr($00)+
           chr($B9)+chr($FC);
    if Fieldbyname('Money').AsString='530' then
start:=chr($FB)+chr($04)+chr($01)+chr($40)+chr($05)+chr($03)+chr($00)+
           chr($B8)+chr($FC);
    if Fieldbyname('Money').AsString='540' then
start:=chr($FB)+chr($04)+chr($01)+chr($40)+chr($05)+chr($04)+chr($00)+
           chr($BF)+chr($FC);
    if Fieldbyname('Money').AsString='550' then
start:=chr($FB)+chr($04)+chr($01)+chr($40)+chr($05)+chr($05)+chr($00)+
           chr($BE)+chr($FC);
    if Fieldbyname('Money').AsString='560' then
start:=chr($FB)+chr($04)+chr($01)+chr($40)+chr($05)+chr($06)+chr($00)+
           chr($BD)+chr($FC);
    if Fieldbyname('Money').AsString='570' then
start:=chr($FB)+chr($04)+chr($01)+chr($40)+chr($05)+chr($07)+chr($00)+
           chr($BC)+chr($FC);
    if Fieldbyname('Money').AsString='580' then
start:=chr($FB)+chr($04)+chr($01)+chr($40)+chr($05)+chr($08)+chr($00)+
           chr($B3)+chr($FC);
    if Fieldbyname('Money').AsString='590' then
start:=chr($FB)+chr($04)+chr($01)+chr($40)+chr($05)+chr($09)+chr($00)+
           chr($B2)+chr($FC);
    if Fieldbyname('Money').AsString='600' then
start:=chr($FB)+chr($04)+chr($01)+chr($40)+chr($06)+chr($00)+chr($00)+
           chr($B8)+chr($FC);
    if Fieldbyname('Money').AsString='610' then
start:=chr($FB)+chr($04)+chr($01)+chr($40)+chr($06)+chr($01)+chr($00)+
           chr($B9)+chr($FC);
    if Fieldbyname('Money').AsString='620' then
start:=chr($FB)+chr($04)+chr($01)+chr($40)+chr($06)+chr($02)+chr($00)+
           chr($BA)+chr($FC);
    if Fieldbyname('Money').AsString='630' then
start:=chr($FB)+chr($04)+chr($01)+chr($40)+chr($06)+chr($03)+chr($00)+
           chr($BB)+chr($FC);
    if Fieldbyname('Money').AsString='640' then
start:=chr($FB)+chr($04)+chr($01)+chr($40)+chr($06)+chr($04)+chr($00)+
           chr($BC)+chr($FC);
    if Fieldbyname('Money').AsString='650' then
start:=chr($FB)+chr($04)+chr($01)+chr($40)+chr($06)+chr($05)+chr($00)+
           chr($BD)+chr($FC);
    if Fieldbyname('Money').AsString='660' then
start:=chr($FB)+chr($04)+chr($01)+chr($40)+chr($06)+chr($06)+chr($00)+
           chr($BE)+chr($FC);
    if Fieldbyname('Money').AsString='670' then
start:=chr($FB)+chr($04)+chr($01)+chr($40)+chr($06)+chr($07)+chr($00)+
           chr($BF)+chr($FC);
```

```
    if Fieldbyname('Money').AsString='680' then
start:=chr($FB)+chr($04)+chr($01)+chr($40)+chr($06)+chr($08)+chr($00)+
           chr($B0)+chr($FC);
    if Fieldbyname('Money').AsString='690' then
start:=chr($FB)+chr($04)+chr($01)+chr($40)+chr($06)+chr($09)+chr($00)+
           chr($B1)+chr($FC);
    if Fieldbyname('Money').AsString='700' then
start:=chr($FB)+chr($04)+chr($01)+chr($40)+chr($07)+chr($00)+chr($00)+
           chr($B9)+chr($FC);
    if Fieldbyname('Money').AsString='710' then
start:=chr($FB)+chr($04)+chr($01)+chr($40)+chr($07)+chr($01)+chr($00)+
           chr($B8)+chr($FC);
    if Fieldbyname('Money').AsString='720' then
start:=chr($FB)+chr($04)+chr($01)+chr($40)+chr($07)+chr($02)+chr($00)+
           chr($BB)+chr($FC);
    if Fieldbyname('Money').AsString='730' then
start:=chr($FB)+chr($04)+chr($01)+chr($40)+chr($07)+chr($03)+chr($00)+
           chr($BA)+chr($FC);
    if Fieldbyname('Money').AsString='740' then
start:=chr($FB)+chr($04)+chr($01)+chr($40)+chr($07)+chr($04)+chr($00)+
           chr($BD)+chr($FC);
    if Fieldbyname('Money').AsString='750' then
start:=chr($FB)+chr($04)+chr($01)+chr($40)+chr($07)+chr($05)+chr($00)+
           chr($BC)+chr($FC);
    if Fieldbyname('Money').AsString='760' then
start:=chr($FB)+chr($04)+chr($01)+chr($40)+chr($07)+chr($06)+chr($00)+
           chr($BF)+chr($FC);
    if Fieldbyname('Money').AsString='770' then
start:=chr($FB)+chr($04)+chr($01)+chr($40)+chr($07)+chr($07)+chr($00)+
           chr($BE)+chr($FC);
    if Fieldbyname('Money').AsString='780' then
start:=chr($FB)+chr($04)+chr($01)+chr($40)+chr($07)+chr($08)+chr($00)+
           chr($B1)+chr($FC);
    if Fieldbyname('Money').AsString='790' then
start:=chr($FB)+chr($04)+chr($01)+chr($40)+chr($07)+chr($09)+chr($00)+
           chr($B0)+chr($FC);
    if Fieldbyname('Money').AsString='800' then
start:=chr($FB)+chr($04)+chr($01)+chr($40)+chr($08)+chr($00)+chr($00)+
           chr($B6)+chr($FC);
    if Fieldbyname('Money').AsString='810' then
start:=chr($FB)+chr($04)+chr($01)+chr($40)+chr($08)+chr($01)+chr($00)+
           chr($B7)+chr($FC);
    if Fieldbyname('Money').AsString='820' then
start:=chr($FB)+chr($04)+chr($01)+chr($40)+chr($08)+chr($02)+chr($00)+
           chr($B4)+chr($FC);
    if Fieldbyname('Money').AsString='830' then
start:=chr($FB)+chr($04)+chr($01)+chr($40)+chr($08)+chr($03)+chr($00)+
           chr($B5)+chr($FC);
    if Fieldbyname('Money').AsString='840' then
start:=chr($FB)+chr($04)+chr($01)+chr($40)+chr($08)+chr($04)+chr($00)+
           chr($B2)+chr($FC);
    if Fieldbyname('Money').AsString='850' then
start:=chr($FB)+chr($04)+chr($01)+chr($40)+chr($08)+chr($05)+chr($00)+
           chr($B3)+chr($FC);
    if Fieldbyname('Money').AsString='860' then
start:=chr($FB)+chr($04)+chr($01)+chr($40)+chr($08)+chr($06)+chr($00)+
           chr($B0)+chr($FC);
```

```
    if Fieldbyname('Money').AsString='870' then
start:=chr($FB)+chr($04)+chr($01)+chr($40)+chr($08)+chr($07)+chr($00)+
           chr($B1)+chr($FC);
    if Fieldbyname('Money').AsString='880' then
start:=chr($FB)+chr($04)+chr($01)+chr($40)+chr($08)+chr($08)+chr($00)+
           chr($BE)+chr($FC);
    if Fieldbyname('Money').AsString='890' then
start:=chr($FB)+chr($04)+chr($01)+chr($40)+chr($08)+chr($09)+chr($00)+
           chr($BF)+chr($FC);
    if Fieldbyname('Money').AsString='900' then
start:=chr($FB)+chr($04)+chr($01)+chr($40)+chr($09)+chr($00)+chr($00)+
           chr($B7)+chr($FC);
    if Fieldbyname('Money').AsString='910' then
start:=chr($FB)+chr($04)+chr($01)+chr($40)+chr($09)+chr($01)+chr($00)+
           chr($B6)+chr($FC);
    if Fieldbyname('Money').AsString='920' then
start:=chr($FB)+chr($04)+chr($01)+chr($40)+chr($09)+chr($02)+chr($00)+
           chr($B5)+chr($FC);
    if Fieldbyname('Money').AsString='930' then
start:=chr($FB)+chr($04)+chr($01)+chr($40)+chr($09)+chr($03)+chr($00)+
           chr($B4)+chr($FC);
    if Fieldbyname('Money').AsString='940' then
start:=chr($FB)+chr($04)+chr($01)+chr($40)+chr($09)+chr($04)+chr($00)+
           chr($B3)+chr($FC);
    if Fieldbyname('Money').AsString='950' then
start:=chr($FB)+chr($04)+chr($01)+chr($40)+chr($09)+chr($05)+chr($00)+
           chr($B2)+chr($FC);
    if Fieldbyname('Money').AsString='960' then
start:=chr($FB)+chr($04)+chr($01)+chr($40)+chr($09)+chr($06)+chr($00)+
           chr($B1)+chr($FC);
    if Fieldbyname('Money').AsString='970' then
start:=chr($FB)+chr($04)+chr($01)+chr($40)+chr($09)+chr($07)+chr($00)+
           chr($B0)+chr($FC);
    if Fieldbyname('Money').AsString='980' then
start:=chr($FB)+chr($04)+chr($01)+chr($40)+chr($09)+chr($08)+chr($00)+
           chr($BF)+chr($FC);
    if Fieldbyname('Money').AsString='990' then
start:=chr($FB)+chr($04)+chr($01)+chr($40)+chr($09)+chr($09)+chr($00)+
           chr($BE)+chr($FC);
    if Fieldbyname('Money').AsString='1000' then
start:=chr($FB)+chr($04)+chr($01)+chr($40)+chr($10)+chr($00)+chr($00)+
           chr($AE)+chr($FC);
    Vacomm1.WriteText(start);
    Timer8.Enabled:=true;
    end;
if radiobutton5.Checked then
    begin
    Fieldbyname('Start').AsString := ''èÒ¹ºÒ§¹Ò';
   if Fieldbyname('Money').AsString='0' then
start:=chr($FB)+chr($04)+chr($01)+chr($50)+chr($00)+chr($00)+chr($00)+
           chr($AE)+chr($FC);
    if Fieldbyname('Money').AsString='10' then
start:=chr($FB)+chr($04)+chr($01)+chr($50)+chr($00)+chr($01)+chr($00)+
           chr($AF)+chr($FC);
    if Fieldbyname('Money').AsString='20' then
start:=chr($FB)+chr($04)+chr($01)+chr($50)+chr($00)+chr($02)+chr($00)+
           chr($AC)+chr($FC);
```

```
    if Fieldbyname('Money').AsString='30' then
start:=chr($FB)+chr($04)+chr($01)+chr($50)+chr($00)+chr($03)+chr($00)+
          chr($AD)+chr($FC);
    if Fieldbyname('Money').AsString='40' then
start:=chr($FB)+chr($04)+chr($01)+chr($50)+chr($00)+chr($04)+chr($00)+
          chr($AA)+chr($FC);
    if Fieldbyname('Money').AsString='50' then
start:=chr($FB)+chr($04)+chr($01)+chr($50)+chr($00)+chr($05)+chr($00)+
          chr($AB)+chr($FC);
    if Fieldbyname('Money').AsString='60' then
start:=chr($FB)+chr($04)+chr($01)+chr($50)+chr($00)+chr($06)+chr($00)+
          chr($A8)+chr($FC);
    if Fieldbyname('Money').AsString='70' then
start:=chr($FB)+chr($04)+chr($01)+chr($50)+chr($00)+chr($07)+chr($00)+
          chr($A9)+chr($FC);
    if Fieldbyname('Money').AsString='80' then
start:=chr($FB)+chr($04)+chr($01)+chr($50)+chr($00)+chr($08)+chr($00)+
          chr($A6)+chr($FC);
    if Fieldbyname('Money').AsString='90' then
start:=chr($FB)+chr($04)+chr($01)+chr($50)+chr($00)+chr($09)+chr($00)+
          chr($A7)+chr($FC);
    if Fieldbyname('Money').AsString='100' then
start:=chr($FB)+chr($04)+chr($01)+chr($50)+chr($01)+chr($00)+chr($00)+
          chr($AF)+chr($FC);
    if Fieldbyname('Money').AsString='110' then
start:=chr($FB)+chr($04)+chr($01)+chr($50)+chr($01)+chr($01)+chr($00)+
          chr($AE)+chr($FC);
    if Fieldbyname('Money').AsString='120' then
start:=chr($FB)+chr($04)+chr($01)+chr($50)+chr($01)+chr($02)+chr($00)+
          chr($AD)+chr($FC);
    if Fieldbyname('Money').AsString='130' then
start:=chr($FB)+chr($04)+chr($01)+chr($50)+chr($01)+chr($03)+chr($00)+
          chr($AC)+chr($FC);
    if Fieldbyname('Money').AsString='140' then
start:=chr($FB)+chr($04)+chr($01)+chr($50)+chr($01)+chr($04)+chr($00)+
          chr($AB)+chr($FC);
    if Fieldbyname('Money').AsString='150' then
start:=chr($FB)+chr($04)+chr($01)+chr($50)+chr($01)+chr($05)+chr($00)+
          chr($AA)+chr($FC);
    if Fieldbyname('Money').AsString='160' then
start:=chr($FB)+chr($04)+chr($01)+chr($50)+chr($01)+chr($06)+chr($00)+
          chr($A9)+chr($FC);
    if Fieldbyname('Money').AsString='170' then
start:=chr($FB)+chr($04)+chr($01)+chr($50)+chr($01)+chr($07)+chr($00)+
          chr($A8)+chr($FC);
    if Fieldbyname('Money').AsString='180' then
start:=chr($FB)+chr($04)+chr($01)+chr($50)+chr($01)+chr($08)+chr($00)+
          chr($A7)+chr($FC);
    if Fieldbyname('Money').AsString='190' then
start:=chr($FB)+chr($04)+chr($01)+chr($50)+chr($01)+chr($09)+chr($00)+
          chr($A6)+chr($FC);
    if Fieldbyname('Money').AsString='200' then
start:=chr($FB)+chr($04)+chr($01)+chr($50)+chr($02)+chr($00)+chr($00)+
          chr($AC)+chr($FC);
    if Fieldbyname('Money').AsString='210' then
start:=chr($FB)+chr($04)+chr($01)+chr($50)+chr($02)+chr($01)+chr($00)+
          chr($AD)+chr($FC);
```

```
    if Fieldbyname('Money').AsString='220' then
start:=chr($FB)+chr($04)+chr($01)+chr($50)+chr($02)+chr($02)+chr($00)+
           chr($AE)+chr($FC);
    if Fieldbyname('Money').AsString='230' then
start:=chr($FB)+chr($04)+chr($01)+chr($50)+chr($02)+chr($03)+chr($00)+
           chr($AF)+chr($FC);
    if Fieldbyname('Money').AsString='240' then
start:=chr($FB)+chr($04)+chr($01)+chr($50)+chr($02)+chr($04)+chr($00)+
           chr($A8)+chr($FC);
    if Fieldbyname('Money').AsString='250' then
start:=chr($FB)+chr($04)+chr($01)+chr($50)+chr($02)+chr($05)+chr($00)+
           chr($A9)+chr($FC);
    if Fieldbyname('Money').AsString='260' then
start:=chr($FB)+chr($04)+chr($01)+chr($50)+chr($02)+chr($06)+chr($00)+
           chr($AA)+chr($FC);
    if Fieldbyname('Money').AsString='270' then
start:=chr($FB)+chr($04)+chr($01)+chr($50)+chr($02)+chr($07)+chr($00)+
           chr($AB)+chr($FC);
    if Fieldbyname('Money').AsString='280' then
start:=chr($FB)+chr($04)+chr($01)+chr($50)+chr($02)+chr($08)+chr($00)+
           chr($A4)+chr($FC);
    if Fieldbyname('Money').AsString='290' then
start:=chr($FB)+chr($04)+chr($01)+chr($50)+chr($02)+chr($09)+chr($00)+
           chr($A5)+chr($FC);
    if Fieldbyname('Money').AsString='300' then
start:=chr($FB)+chr($04)+chr($01)+chr($50)+chr($03)+chr($00)+chr($00)+
           chr($AD)+chr($FC);
    if Fieldbyname('Money').AsString='310' then
start:=chr($FB)+chr($04)+chr($01)+chr($50)+chr($03)+chr($01)+chr($00)+
           chr($AC)+chr($FC);
    if Fieldbyname('Money').AsString='320' then
start:=chr($FB)+chr($04)+chr($01)+chr($50)+chr($03)+chr($02)+chr($00)+
           chr($AF)+chr($FC);
    if Fieldbyname('Money').AsString='330' then
start:=chr($FB)+chr($04)+chr($01)+chr($50)+chr($03)+chr($03)+chr($00)+
           chr($AE)+chr($FC);
    if Fieldbyname('Money').AsString='340' then
start:=chr($FB)+chr($04)+chr($01)+chr($50)+chr($03)+chr($04)+chr($00)+
           chr($A9)+chr($FC);
    if Fieldbyname('Money').AsString='350' then
start:=chr($FB)+chr($04)+chr($01)+chr($50)+chr($03)+chr($05)+chr($00)+
           chr($A8)+chr($FC);
    if Fieldbyname('Money').AsString='360' then
start:=chr($FB)+chr($04)+chr($01)+chr($50)+chr($03)+chr($06)+chr($00)+
           chr($AB)+chr($FC);
    if Fieldbyname('Money').AsString='370' then
start:=chr($FB)+chr($04)+chr($01)+chr($50)+chr($03)+chr($07)+chr($00)+
           chr($AA)+chr($FC);
    if Fieldbyname('Money').AsString='380' then
start:=chr($FB)+chr($04)+chr($01)+chr($50)+chr($03)+chr($08)+chr($00)+
           chr($A5)+chr($FC);
    if Fieldbyname('Money').AsString='390' then
start:=chr($FB)+chr($04)+chr($01)+chr($50)+chr($03)+chr($09)+chr($00)+
           chr($A4)+chr($FC);
    if Fieldbyname('Money').AsString='400' then
start:=chr($FB)+chr($04)+chr($01)+chr($50)+chr($04)+chr($00)+chr($00)+
           chr($AA)+chr($FC);
```

```
    if Fieldbyname('Money').AsString='410' then
start:=chr($FB)+chr($04)+chr($01)+chr($50)+chr($04)+chr($01)+chr($00)+
           chr($AB)+chr($FC);
    if Fieldbyname('Money').AsString='420' then
start:=chr($FB)+chr($04)+chr($01)+chr($50)+chr($04)+chr($02)+chr($00)+
           chr($A8)+chr($FC);
    if Fieldbyname('Money').AsString='430' then
start:=chr($FB)+chr($04)+chr($01)+chr($50)+chr($04)+chr($03)+chr($00)+
           chr($A9)+chr($FC);
    if Fieldbyname('Money').AsString='440' then
start:=chr($FB)+chr($04)+chr($01)+chr($50)+chr($04)+chr($04)+chr($00)+
           chr($AE)+chr($FC);
    if Fieldbyname('Money').AsString='450' then
start:=chr($FB)+chr($04)+chr($01)+chr($50)+chr($04)+chr($05)+chr($00)+
           chr($AF)+chr($FC);
    if Fieldbyname('Money').AsString='460' then
start:=chr($AC)+chr($04)+chr($01)+chr($50)+chr($04)+chr($06)+chr($00)+
           chr($F9)+chr($FC);
    if Fieldbyname('Money').AsString='470' then
start:=chr($FB)+chr($04)+chr($01)+chr($50)+chr($04)+chr($07)+chr($00)+
           chr($AD)+chr($FC);
    if Fieldbyname('Money').AsString='480' then
start:=chr($FB)+chr($04)+chr($01)+chr($50)+chr($04)+chr($08)+chr($00)+
           chr($A2)+chr($FC);
    if Fieldbyname('Money').AsString='490' then
start:=chr($FB)+chr($04)+chr($01)+chr($50)+chr($04)+chr($09)+chr($00)+
           chr($A3)+chr($FC);
    if Fieldbyname('Money').AsString='500' then
start:=chr($FB)+chr($04)+chr($01)+chr($50)+chr($05)+chr($00)+chr($00)+
           chr($AB)+chr($FC);
    if Fieldbyname('Money').AsString='510' then
start:=chr($FB)+chr($04)+chr($01)+chr($50)+chr($05)+chr($01)+chr($00)+
           chr($AA)+chr($FC);
    if Fieldbyname('Money').AsString='520' then
start:=chr($FB)+chr($04)+chr($01)+chr($50)+chr($05)+chr($02)+chr($00)+
           chr($A9)+chr($FC);
    if Fieldbyname('Money').AsString='530' then
start:=chr($FB)+chr($04)+chr($01)+chr($50)+chr($05)+chr($03)+chr($00)+
           chr($A8)+chr($FC);
    if Fieldbyname('Money').AsString='540' then
start:=chr($FB)+chr($04)+chr($01)+chr($50)+chr($05)+chr($04)+chr($00)+
           chr($AF)+chr($FC);
    if Fieldbyname('Money').AsString='550' then
start:=chr($FB)+chr($04)+chr($01)+chr($50)+chr($05)+chr($05)+chr($00)+
           chr($AE)+chr($FC);
    if Fieldbyname('Money').AsString='560' then
start:=chr($FB)+chr($04)+chr($01)+chr($50)+chr($05)+chr($06)+chr($00)+
           chr($AD)+chr($FC);
    if Fieldbyname('Money').AsString='570' then
start:=chr($FB)+chr($04)+chr($01)+chr($50)+chr($05)+chr($07)+chr($00)+
           chr($AC)+chr($FC);
    if Fieldbyname('Money').AsString='580' then
start:=chr($FB)+chr($04)+chr($01)+chr($50)+chr($05)+chr($08)+chr($00)+
           chr($A3)+chr($FC);
    if Fieldbyname('Money').AsString='590' then
start:=chr($FB)+chr($04)+chr($01)+chr($50)+chr($05)+chr($09)+chr($00)+
           chr($A2)+chr($FC);
```

```
    if Fieldbyname('Money').AsString='600' then
start:=chr($FB)+chr($04)+chr($01)+chr($50)+chr($06)+chr($00)+chr($00)+
           chr($A8)+chr($FC);
    if Fieldbyname('Money').AsString='610' then
start:=chr($FB)+chr($04)+chr($01)+chr($50)+chr($06)+chr($01)+chr($00)+
           chr($A9)+chr($FC);
    if Fieldbyname('Money').AsString='620' then
start:=chr($FB)+chr($04)+chr($01)+chr($50)+chr($06)+chr($02)+chr($00)+
           chr($AA)+chr($FC);
    if Fieldbyname('Money').AsString='630' then
start:=chr($FB)+chr($04)+chr($01)+chr($50)+chr($06)+chr($03)+chr($00)+
           chr($AB)+chr($FC);
    if Fieldbyname('Money').AsString='640' then
start:=chr($FB)+chr($04)+chr($01)+chr($50)+chr($06)+chr($04)+chr($00)+
           chr($AC)+chr($FC);
    if Fieldbyname('Money').AsString='650' then
start:=chr($FB)+chr($04)+chr($01)+chr($50)+chr($06)+chr($05)+chr($00)+
           chr($AD)+chr($FC);
    if Fieldbyname('Money').AsString='660' then
start:=chr($FB)+chr($04)+chr($01)+chr($50)+chr($06)+chr($06)+chr($00)+
           chr($AE)+chr($FC);
    if Fieldbyname('Money').AsString='670' then
start:=chr($FB)+chr($04)+chr($01)+chr($50)+chr($06)+chr($07)+chr($00)+
           chr($AF)+chr($FC);
    if Fieldbyname('Money').AsString='680' then
start:=chr($FB)+chr($04)+chr($01)+chr($50)+chr($06)+chr($08)+chr($00)+
           chr($A0)+chr($FC);
    if Fieldbyname('Money').AsString='690' then
start:=chr($FB)+chr($04)+chr($01)+chr($50)+chr($06)+chr($09)+chr($00)+
           chr($A1)+chr($FC);
    if Fieldbyname('Money').AsString='700' then
start:=chr($FB)+chr($04)+chr($01)+chr($50)+chr($07)+chr($00)+chr($00)+
           chr($A9)+chr($FC);
    if Fieldbyname('Money').AsString='710' then
start:=chr($FB)+chr($04)+chr($01)+chr($50)+chr($07)+chr($01)+chr($00)+
           chr($A8)+chr($FC);
    if Fieldbyname('Money').AsString='720' then
start:=chr($FB)+chr($04)+chr($01)+chr($50)+chr($07)+chr($02)+chr($00)+
           chr($AB)+chr($FC);
    if Fieldbyname('Money').AsString='730' then
start:=chr($FB)+chr($04)+chr($01)+chr($50)+chr($07)+chr($03)+chr($00)+
           chr($AA)+chr($FC);
    if Fieldbyname('Money').AsString='740' then
start:=chr($FB)+chr($04)+chr($01)+chr($50)+chr($07)+chr($04)+chr($00)+
           chr($AD)+chr($FC);
    if Fieldbyname('Money').AsString='750' then
start:=chr($FB)+chr($04)+chr($01)+chr($50)+chr($07)+chr($05)+chr($00)+
           chr($AC)+chr($FC);
    if Fieldbyname('Money').AsString='760' then
start:=chr($FB)+chr($04)+chr($01)+chr($50)+chr($07)+chr($06)+chr($00)+
           chr($AF)+chr($FC);
    if Fieldbyname('Money').AsString='770' then
start:=chr($FB)+chr($04)+chr($01)+chr($50)+chr($07)+chr($07)+chr($00)+
           chr($AE)+chr($FC);
    if Fieldbyname('Money').AsString='780' then
start:=chr($FB)+chr($04)+chr($01)+chr($50)+chr($07)+chr($08)+chr($00)+
           chr($A1)+chr($FC);
```

```
    if Fieldbyname('Money').AsString='790' then
start:=chr($FB)+chr($04)+chr($01)+chr($50)+chr($07)+chr($09)+chr($00)+
           chr($A0)+chr($FC);
    if Fieldbyname('Money').AsString='800' then
start:=chr($FB)+chr($04)+chr($01)+chr($50)+chr($08)+chr($00)+chr($00)+
           chr($A6)+chr($FC);
    if Fieldbyname('Money').AsString='810' then
start:=chr($FB)+chr($04)+chr($01)+chr($50)+chr($08)+chr($01)+chr($00)+
           chr($A7)+chr($FC);
    if Fieldbyname('Money').AsString='820' then
start:=chr($FB)+chr($04)+chr($01)+chr($50)+chr($08)+chr($02)+chr($00)+
           chr($A4)+chr($FC);
    if Fieldbyname('Money').AsString='830' then
start:=chr($FB)+chr($04)+chr($01)+chr($50)+chr($08)+chr($03)+chr($00)+
           chr($A5)+chr($FC);
    if Fieldbyname('Money').AsString='840' then
start:=chr($FB)+chr($04)+chr($01)+chr($50)+chr($08)+chr($04)+chr($00)+
           chr($A2)+chr($FC);
    if Fieldbyname('Money').AsString='850' then
start:=chr($FB)+chr($04)+chr($01)+chr($50)+chr($08)+chr($05)+chr($00)+
           chr($A3)+chr($FC);
    if Fieldbyname('Money').AsString='860' then
start:=chr($FB)+chr($04)+chr($01)+chr($50)+chr($08)+chr($06)+chr($00)+
           chr($A0)+chr($FC);
    if Fieldbyname('Money').AsString='870' then
start:=chr($FB)+chr($04)+chr($01)+chr($50)+chr($08)+chr($07)+chr($00)+
           chr($A1)+chr($FC);
    if Fieldbyname('Money').AsString='880' then
start:=chr($FB)+chr($04)+chr($01)+chr($50)+chr($08)+chr($08)+chr($00)+
           chr($AE)+chr($FC);
    if Fieldbyname('Money').AsString='890' then
start:=chr($FB)+chr($04)+chr($01)+chr($50)+chr($08)+chr($09)+chr($00)+
           chr($AF)+chr($FC);
    if Fieldbyname('Money').AsString='900' then
start:=chr($FB)+chr($04)+chr($01)+chr($50)+chr($09)+chr($00)+chr($00)+
           chr($A7)+chr($FC);
    if Fieldbyname('Money').AsString='910' then
start:=chr($FB)+chr($04)+chr($01)+chr($50)+chr($09)+chr($01)+chr($00)+
           chr($A6)+chr($FC);
    if Fieldbyname('Money').AsString='920' then
start:=chr($FB)+chr($04)+chr($01)+chr($50)+chr($09)+chr($02)+chr($00)+
           chr($A5)+chr($FC);
    if Fieldbyname('Money').AsString='930' then
start:=chr($FB)+chr($04)+chr($01)+chr($50)+chr($09)+chr($03)+chr($00)+
           chr($A4)+chr($FC);
    if Fieldbyname('Money').AsString='940' then
start:=chr($FB)+chr($04)+chr($01)+chr($50)+chr($09)+chr($04)+chr($00)+
           chr($A3)+chr($FC);
    if Fieldbyname('Money').AsString='950' then
start:=chr($FB)+chr($04)+chr($01)+chr($50)+chr($09)+chr($05)+chr($00)+
           chr($A2)+chr($FC);
    if Fieldbyname('Money').AsString='960' then
start:=chr($FB)+chr($04)+chr($01)+chr($50)+chr($09)+chr($06)+chr($00)+
           chr($A1)+chr($FC);
    if Fieldbyname('Money').AsString='970' then
start:=chr($FB)+chr($04)+chr($01)+chr($50)+chr($09)+chr($07)+chr($00)+
           chr($A0)+chr($FC);
```

```
    if Fieldbyname('Money').AsString='980' then
start:=chr($FB)+chr($04)+chr($01)+chr($50)+chr($09)+chr($08)+chr($00)+
          chr($AF)+chr($FC);
    if Fieldbyname('Money').AsString='990' then
start:=chr($FB)+chr($04)+chr($01)+chr($50)+chr($09)+chr($09)+chr($00)+
          chr($AE)+chr($FC);
    if Fieldbyname('Money').AsString='1000' then
start:=chr($FB)+chr($04)+chr($01)+chr($50)+chr($10)+chr($00)+chr($00)+
          chr($BE)+chr($FC);
    Vacomm1.WriteText(start);
    Timer8.Enabled:=true;
    end;
end;
    post;
    Timer7.Enabled:=true;
    end;
end
else
exit;
end;
end;
procedure TForm2.Timer1Timer(Sender: TObject);
begin
   DBEdit1.Text:=Datetostr(date);
   DBEdit2.Text:=Timetostr(Time);
end;
procedure TForm2.BitBtn1Click(Sender: TObject);
var st:string;
begin
    st:=chr($FB)+chr($02)+chr($FF)+chr($06)+chr($FC);
    Vacomm1.WriteText(st);
end;
procedure TForm2.BitBtn2Click(Sender: TObject);
begin
  Edit1.Clear;
  Edit5.Clear;
  radiobutton1.Checked := false;
  radiobutton2.Checked := false;
  radiobutton3.Checked := false;
  radiobutton4.Checked := false;
  radiobutton5.Checked := false;
end;
procedure TForm2.SpeedButton2Click(Sender: TObject);
begin
     if SpeedButton2.Down then
  begin
      SpeedButton1.Enabled:=True;
      SpeedButton2.Caption:='Close Port';
      SpeedButton2.Enabled:=False;
  end
  else
  begin
      SpeedButton1.Enabled:=False;
      SpeedButton1.Down:= False;
      SpeedButton2.Caption:='Open Port';
  end;
end;
```

```
procedure TForm2.SpeedButton1Click(Sender: TObject);
var
  w, h: integer;
  f: string;
begin
   if SpeedButton1.Down then
  begin
    Connect;
    w := 300;
    h := 300;
    ImageEn1.IO.DShowParams.SetCurrentVideoFormat(w, h, '');
    ImageEn1.IO.DShowParams.GetCurrentVideoFormat(w, h, f);
    Label4.Caption := 'Capturing at ' + inttostr(w) + 'x' + inttostr(h)
+ ' ' + f;
    ImageEn1.IO.DShowParams.Run;
    Timer3.Enabled:=True;
    SpeedButton1.Caption:='Stop';
  end
  else
  begin
    Disconnect;
    SpeedButton2.Enabled:=True;
    SpeedButton1.Enabled:=False;
    Timer3.Enabled:=False;
    SpeedButton1.Caption:='Capture';
  end;
end;
procedure TForm2.Timer3Timer(Sender: TObject);
begin
ImageEn1.IO.SaveToFileJpeg('c:\AppServ\www\phptest\joe7\photo\joe.jpg')
;
end;
procedure TForm2.Timer4Timer(Sender: TObject);
begin
   ImageEn1.IO.CloseAVIFile;
   Timer4.Enabled:=False;
   SpeedButton1.Down:=False;
   Shape1.Brush.Color:=clwhite;
end;
procedure TForm2.FormActivate(Sender: TObject);
var resp,resp1:word;
begin
  resp:=MessageDlg('Open Port?',mtConfirmation,[mbYes,mbNo],0);
  if resp = mrYes then
  Vacomm1.Open;
  if resp = mrNo then
  Showmessage('Cant Connect To The Port');
  Close;
  ImageEn1.LayersSync:=false;
  ComboBox1.Items.Assign(ImageEn1.IO.DShowParams.VideoInputs);
  ComboBox1.ItemIndex := 0;
  ShowVideoFormats;end;
procedure TForm2.ComboBox1Change(Sender: TObject);
begin
  ShowVideoFormats;
end;
```

```
procedure TForm2.Disconnect;
begin
  ImageEn1.IO.DShowParams.Disconnect;
end;
procedure TForm2.ShowVideoFormats;
var
  i: integer;
  s: string;
begin
  Connect;
  Disconnect;
end;
procedure TForm2.Connect;
begin
  if (not ImageEn1.IO.DShowParams.Connected) then
  begin
    ImageEn1.IO.DShowParams.SetVideoInput(ComboBox1.ItemIndex);
    ImageEn1.IO.DShowParams.EnableSampleGrabber := true;
    ImageEn1.IO.DShowParams.Connect;
  end;
end;
procedure TForm2.Timer8Timer(Sender: TObject);
var st:string;
begin
  st:=chr($FB)+chr($02)+chr($FF)+chr($06)+chr($FC);
  Vacomm1.WriteText(st);
  radiobutton1.Checked:=false;
  radiobutton2.Checked:=false;
  radiobutton3.Checked:=false;
  radiobutton4.Checked:=false;
  radiobutton5.Checked:=false;
  Edit1.Clear;
  Edit5.Clear;
  timer8.Enabled:=false;
end;
procedure TForm2.Timer5Timer(Sender: TObject);
begin
  port := IntToStr(PortIn($379));
  if port = '127' then
  Edit7.Text:='START'
  else
  if port = '63' then
  Edit7.Text:='STOP';
end;
procedure TForm2.Edit7Change(Sender: TObject);
Var
  a,b,st: String;
  i:Integer;
begin
    if Edit7.Text='START' then
    begin
    if (radiobutton1.Checked = true) or
       (radiobutton2.Checked = true) or
       (radiobutton3.Checked = true) or
       (radiobutton4.Checked = true) or
       (radiobutton5.Checked = true) then
      begin
```

```
        st:=chr($FB)+chr($02)+chr($FF)+chr($06)+chr($FC);
        Vacomm1.WriteText(st);
        Timer6.Enabled:=true;
        end
        else
        Showmessage('Please Select Station First!!!');
    end;
end;
procedure TForm2.Timer6Timer(Sender: TObject);
var  a,b,x: String;
     i:Integer;
begin
    Shape1.Brush.Color:=clyellow;
    i:=StrToIntDef(Label5.Caption,0);
    i:=i+1;
    b:=inttostr(i);
    Label5.Caption:=b;
    Timer4.Enabled:=True;
ImageEn1.IO.CreateAVIFile('c:\AppServ\www\phptest\joe7\photo\tag'+b+'.a
vi', 10, 'msvc');
    timer6.Enabled:=false;
end;
procedure TForm2.Timer7Timer(Sender: TObject);
begin
    Edit1.Clear;
    Timer7.Enabled:=false;
end;
procedure TForm2.ImageEn1DShowNewFrame(Sender: TObject);
var
  ss: string;
begin
  ImageEn1.LayersCurrent:=0;
  ImageEn1.IO.DShowParams.GetSample(ImageEn1.IEBitmap);
  ss := FormatDateTime('c', date + time);
  with ImageEn1.IEBitmap.Canvas do
  begin
    Brush.Style := bsClear;
    Font.Color := clWhite;
    TextOut(0, 0, ss);
  end;
  if Shape1.Brush.Color=clyellow then
  ImageEn1.IO.SaveToAVI;
  ImageEn1.Update;
  ImageEn1.Paint;
end;
procedure TForm2.ExitBtnClick(Sender: TObject);
begin
    Vacomm1.Close;
    Close;
end;
end.
```

End Station

```
unit Unit3;

interface

uses
  Windows, Messages, SysUtils, Variants, Classes, Graphics, Controls,
Forms,
  Dialogs, ExtCtrls, VaClasses, VaComm, XPMan, DBAccess, MyAccess, DB,
  MemDS, MyDacVcl, StdCtrls, Grids, DBGrids, Mask, DBCtrls, Buttons,
  ComCtrls;

type
  TForm3 = class(TForm)
    Panel2: TPanel;
    Panel1: TPanel;
    Label2: TLabel;
    GroupBox1: TGroupBox;
    Image1: TImage;
    RadioButton1: TRadioButton;
    RadioButton2: TRadioButton;
    RadioButton3: TRadioButton;
    RadioButton4: TRadioButton;
    RadioButton5: TRadioButton;
    MyConnection: TMyConnection;
    DataSource: TDataSource;
    ConnectDialog: TMyConnectDialog;
    Query: TMyQuery;
    sqDropTable: TMyCommand;
    XPManifest1: TXPManifest;
    sqCreateTable: TMyCommand;
    VaComm1: TVaComm;
    Timer1: TTimer;
    Timer3: TTimer;
    Timer4: TTimer;
    Timer5: TTimer;
    PageControl1: TPageControl;
    TabSheet1: TTabSheet;
    Label5: TLabel;
    Edit2: TEdit;
    Edit6: TEdit;
    TabSheet2: TTabSheet;
    DBEdit1: TDBEdit;
    DBEdit2: TDBEdit;
    Edit1: TEdit;
    Edit5: TEdit;
    DBGrid1: TDBGrid;
    BitBtn1: TBitBtn;
    BitBtn2: TBitBtn;
    TabSheet4: TTabSheet;
    Label1: TLabel;
    Edit4: TEdit;
    Edit7: TEdit;
    Label3: TLabel;
    Label4: TLabel;
    Edit8: TEdit;
    Label6: TLabel;
```

```
    btOpen: TButton;
    btClose: TButton;
    btCreate: TButton;
    btDrop: TButton;
    ExitBtn: TSpeedButton;
    procedure btOpenClick(Sender: TObject);
    procedure btCloseClick(Sender: TObject);
    procedure btCreateClick(Sender: TObject);
    procedure btDropClick(Sender: TObject);
    procedure VaComm1RxChar(Sender: TObject; Count: Integer);
    procedure Timer1Timer(Sender: TObject);
    procedure Timer3Timer(Sender: TObject);
    procedure Timer4Timer(Sender: TObject);
    procedure BitBtn2Click(Sender: TObject);
    procedure BitBtn1Click(Sender: TObject);
    procedure Edit4Change(Sender: TObject);
    procedure Timer5Timer(Sender: TObject);
    procedure addmoney;
    procedure ExitBtnClick(Sender: TObject);
    procedure FormActivate(Sender: TObject);
  private
    { Private declarations }
  public
    { Public declarations }
  end;

procedure PortOut(Port : Word; Data : Byte); StdCall; External
'io.dll';
function PortIn(Port : Word) : byte; StdCall; External 'io.dll'

var
  Form3: TForm3;
  RxBuff : array[1..33] of byte;
  count : integer;
  byte4,byte5,byte6,byte7,byte8,byte31,port,p:string;

implementation

{$R *.dfm}

procedure TForm3.btOpenClick(Sender: TObject);
begin
   Query.Open;
end;
procedure TForm3.btCloseClick(Sender: TObject);
begin
   Query.Close;
end;
procedure TForm3.btCreateClick(Sender: TObject);
begin
   sqCreateTable.Execute;
end;
procedure TForm3.btDropClick(Sender: TObject);
begin
   sqDropTable.Execute;
end;
procedure TForm3.VaComm1RxChar(Sender: TObject; Count: Integer);
```

```
var  St,a,b,c:string;
     AA:integer;
begin
  if count >= 33 then
  begin
    St := Vacomm1.ReadText;
    Edit1.Text := '';
    for AA := 1 to 33 do
    begin
    Edit1.text := Edit1.Text + ' ' + inttostr((ord(St[AA])));
    end;
    Edit5.text:= inttostr((ord(st[31])));
    byte31:= inttostr((ord(st[31])));
    byte4:=inttostr((ord(st[4])));
    byte5:=inttostr((ord(st[5])));
    byte6:=inttostr((ord(st[6])));
    byte7:=inttostr((ord(st[7])));
    byte8:=inttostr((ord(st[8])));
if (byte31='16') or (byte31='17') or (byte31='18') or
   (byte31='19') or (byte31='20') then
begin
    with query do
    begin
    append;
    Fieldbyname('Date_End').AsString := datetostr(date);
    Fieldbyname('Time_End').AsString := timetostr(time);
    Fieldbyname('ID').AsString := Edit5.Text;
    if radiobutton1.Checked then
    Edit8.Text:='´èÒ¹´Ô¹á´§';
    Fieldbyname('End').AsString := '´èÒ¹´Ô¹á´§';
    if radiobutton2.Checked then
    Edit8.Text:='´èÒ¹ÊØ¢ØÁÇÔ·';
    Fieldbyname('End').AsString := '´èÒ¹ÊØ¢ØÁÇÔ·';
    if radiobutton3.Checked then
    Edit8.Text:='´èÒ¹ÍÒ¨³Ã§¤ì';
    Fieldbyname('End').AsString := '´èÒ¹ÍÒ¨³Ã§¤ì';
    if radiobutton4.Checked then
    Edit8.Text:='´èÒ¹ÊØ¢ØÁÇÔ· 62';
    Fieldbyname('End').AsString := '´èÒ¹ÊØ¢ØÁÇÔ· 62';
    if radiobutton5.Checked then
    Edit8.Text:='´èÒ¹°Ò§¹Ò';
    Fieldbyname('End').AsString := '´èÒ¹°Ò§¹Ò';
if byte4='16' then
Edit7.Text:='´èÒ¹´Ô¹á´§';
if byte4='32' then
Edit7.Text:='´èÒ¹ÊØ¢ØÁÇÔ·';
if byte4='48' then
Edit7.Text:='´èÒ¹ÍÒ¨³Ã§¤ì';
if byte4='64' then
Edit7.Text:='´èÒ¹ÊØ¢ØÁÇÔ· 62';
if byte4='80' then
Edit7.Text:='´èÒ¹°Ò§¹Ò';
begin
        if byte5='0' then
          begin
            if byte6='0'  then Fieldbyname('Money').AsString :='0';
            if byte6='1'  then Fieldbyname('Money').AsString :='10';
```

```
      if byte6='2'  then Fieldbyname('Money').AsString :='20';
      if byte6='3'  then Fieldbyname('Money').AsString :='30';
      if byte6='4'  then Fieldbyname('Money').AsString :='40';
      if byte6='5'  then Fieldbyname('Money').AsString :='50';
      if byte6='6'  then Fieldbyname('Money').AsString :='60';
      if byte6='7'  then Fieldbyname('Money').AsString :='70';
      if byte6='8'  then Fieldbyname('Money').AsString :='80';
      if byte6='9'  then Fieldbyname('Money').AsString :='90';
    end;
  if byte5='1' then
    begin
      if byte6='0'  then Fieldbyname('Money').AsString :='100';
      if byte6='1'  then Fieldbyname('Money').AsString :='110';
      if byte6='2'  then Fieldbyname('Money').AsString :='120';
      if byte6='3'  then Fieldbyname('Money').AsString :='130';
      if byte6='4'  then Fieldbyname('Money').AsString :='140';
      if byte6='5'  then Fieldbyname('Money').AsString :='150';
      if byte6='6'  then Fieldbyname('Money').AsString :='160';
      if byte6='7'  then Fieldbyname('Money').AsString :='170';
      if byte6='8'  then Fieldbyname('Money').AsString :='180';
      if byte6='9'  then Fieldbyname('Money').AsString :='190';
    end;
  if byte5='2' then
    begin
      if byte6='0'  then Fieldbyname('Money').AsString :='200';
      if byte6='1'  then Fieldbyname('Money').AsString :='210';
      if byte6='2'  then Fieldbyname('Money').AsString :='220';
      if byte6='3'  then Fieldbyname('Money').AsString :='230';
      if byte6='4'  then Fieldbyname('Money').AsString :='240';
      if byte6='5'  then Fieldbyname('Money').AsString :='250';
      if byte6='6'  then Fieldbyname('Money').AsString :='260';
      if byte6='7'  then Fieldbyname('Money').AsString :='270';
      if byte6='8'  then Fieldbyname('Money').AsString :='280';
      if byte6='9'  then Fieldbyname('Money').AsString :='290';
    end;
  if byte5='3' then
    begin
      if byte6='0'  then Fieldbyname('Money').AsString :='300';
      if byte6='1'  then Fieldbyname('Money').AsString :='310';
      if byte6='2'  then Fieldbyname('Money').AsString :='320';
      if byte6='3'  then Fieldbyname('Money').AsString :='330';
      if byte6='4'  then Fieldbyname('Money').AsString :='340';
      if byte6='5'  then Fieldbyname('Money').AsString :='350';
      if byte6='6'  then Fieldbyname('Money').AsString :='360';
      if byte6='7'  then Fieldbyname('Money').AsString :='370';
      if byte6='8'  then Fieldbyname('Money').AsString :='380';
      if byte6='9'  then Fieldbyname('Money').AsString :='390';
      end;
   if byte5='4' then
     begin
      if byte6='0'  then Fieldbyname('Money').AsString :='400';
      if byte6='1'  then Fieldbyname('Money').AsString :='410';
      if byte6='2'  then Fieldbyname('Money').AsString :='420';
      if byte6='3'  then Fieldbyname('Money').AsString :='430';
      if byte6='4'  then Fieldbyname('Money').AsString :='440';
      if byte6='5'  then Fieldbyname('Money').AsString :='450';
      if byte6='6'  then Fieldbyname('Money').AsString :='460';
```

```
      if byte6='7'  then Fieldbyname('Money').AsString :='470';
      if byte6='8'  then Fieldbyname('Money').AsString :='480';
      if byte6='9'  then Fieldbyname('Money').AsString :='490';
    end;
  if byte5='5' then
    begin
      if byte6='0'  then Fieldbyname('Money').AsString :='500';
      if byte6='1'  then Fieldbyname('Money').AsString :='510';
      if byte6='2'  then Fieldbyname('Money').AsString :='520';
      if byte6='3'  then Fieldbyname('Money').AsString :='530';
      if byte6='4'  then Fieldbyname('Money').AsString :='540';
      if byte6='5'  then Fieldbyname('Money').AsString :='550';
      if byte6='6'  then Fieldbyname('Money').AsString :='560';
      if byte6='7'  then Fieldbyname('Money').AsString :='570';
      if byte6='8'  then Fieldbyname('Money').AsString :='580';
      if byte6='9'  then Fieldbyname('Money').AsString :='590';
    end;
  if byte5='6' then
    begin
      if byte6='0'  then Fieldbyname('Money').AsString :='600';
      if byte6='1'  then Fieldbyname('Money').AsString :='610';
      if byte6='2'  then Fieldbyname('Money').AsString :='620';
      if byte6='3'  then Fieldbyname('Money').AsString :='630';
      if byte6='4'  then Fieldbyname('Money').AsString :='640';
      if byte6='5'  then Fieldbyname('Money').AsString :='650';
      if byte6='6'  then Fieldbyname('Money').AsString :='660';
      if byte6='7'  then Fieldbyname('Money').AsString :='670';
      if byte6='8'  then Fieldbyname('Money').AsString :='680';
      if byte6='9'  then Fieldbyname('Money').AsString :='690';
    end;
  if byte5='7' then
    begin
      if byte6='0'  then Fieldbyname('Money').AsString :='700';
      if byte6='1'  then Fieldbyname('Money').AsString :='710';
      if byte6='2'  then Fieldbyname('Money').AsString :='720';
      if byte6='3'  then Fieldbyname('Money').AsString :='730';
      if byte6='4'  then Fieldbyname('Money').AsString :='740';
      if byte6='5'  then Fieldbyname('Money').AsString :='750';
      if byte6='6'  then Fieldbyname('Money').AsString :='760';
      if byte6='7'  then Fieldbyname('Money').AsString :='770';
      if byte6='8'  then Fieldbyname('Money').AsString :='780';
      if byte6='9'  then Fieldbyname('Money').AsString :='790';
    end;
  if byte5='8' then
   begin
      if byte6='0'  then Fieldbyname('Money').AsString :='800';
      if byte6='1'  then Fieldbyname('Money').AsString :='810';
      if byte6='2'  then Fieldbyname('Money').AsString :='820';
      if byte6='3'  then Fieldbyname('Money').AsString :='830';
      if byte6='4'  then Fieldbyname('Money').AsString :='840';
      if byte6='5'  then Fieldbyname('Money').AsString :='850';
      if byte6='6'  then Fieldbyname('Money').AsString :='860';
      if byte6='7'  then Fieldbyname('Money').AsString :='870';
      if byte6='8'  then Fieldbyname('Money').AsString :='880';
      if byte6='9'  then Fieldbyname('Money').AsString :='890';
    end;
  if byte5='9' then
```

```
          begin
            if byte6='0'  then Fieldbyname('Money').AsString :='900';
            if byte6='1'  then Fieldbyname('Money').AsString :='910';
            if byte6='2'  then Fieldbyname('Money').AsString :='920';
            if byte6='3'  then Fieldbyname('Money').AsString :='930';
            if byte6='4'  then Fieldbyname('Money').AsString :='940';
            if byte6='5'  then Fieldbyname('Money').AsString :='950';
            if byte6='6'  then Fieldbyname('Money').AsString :='960';
            if byte6='7'  then Fieldbyname('Money').AsString :='970';
            if byte6='8'  then Fieldbyname('Money').AsString :='980';
            if byte6='9'  then Fieldbyname('Money').AsString :='990';
          end;
        if byte5 = '16' then Fieldbyname('Money').AsString:='1000';
end;
if byte4='16' then
      begin
      if radiobutton1.Checked then  //dindang
      begin
      Fieldbyname('Fee').AsString:='0';
      a:=Fieldbyname('Fee').AsString;
      Edit2.Text:=a+ 'Baht';
      b:=fieldbyname('Money').AsString;
        if strtoint(b) < strtoint(a) then
        begin
        showmessage('Please Fill Your Money');
        exit;
        end
        else
        begin
        c:=inttostr(strtoint(b)-strtoint(a));
        fieldbyname('Money').AsString:=c;
        Edit6.Text:=Fieldbyname('Money').AsString + ' ' + 'Baht';
        addmoney;
        Timer3.Enabled:=true;
        end;
      end;
      if radiobutton2.Checked then //sukumvit
      begin
      fieldbyname('Fee').AsString:='10';
      a:=fieldbyname('Fee').AsString;
      Edit2.Text:=a+ ' Baht';
      b:=fieldbyname('Money').AsString;
        if strtoint(b) < strtoint(a) then
        begin
        showmessage('Please fill your money');
        exit;
        end
        else
        begin
        c:=inttostr(strtoint(b)-strtoint(a));
        fieldbyname('Money').AsString:=c;
        Edit6.Text:=Fieldbyname('Money').AsString + ' ' + 'Baht';
        addmoney;
        Timer3.Enabled:=true;
        end;
      end;
```

```
if radiobutton3.Checked then //argnarong
begin
fieldbyname('Fee').AsString:='20';
a:=fieldbyname('Fee').AsString;
Edit2.Text:=a+ ' Baht';
b:=fieldbyname('Money').AsString;
  if strtoint(b) < strtoint(a) then
  begin
  showmessage('Please fill your money');
  exit;
  end
  else
  begin
  c:=inttostr(strtoint(b)-strtoint(a));
  fieldbyname('Money').AsString:=c;
  Edit6.Text:=Fieldbyname('Money').AsString + ' ' + 'Baht';
  addmoney;
  Timer3.Enabled:=true;
  end;
end;
if radiobutton4.Checked then //sukumvit 62
begin
fieldbyname('Fee').AsString:='30';
a:=fieldbyname('Fee').AsString;
Edit2.Text:=a+ ' Baht';
b:=fieldbyname('Money').AsString;
  if strtoint(b) < strtoint(a) then
  begin
  showmessage('Please fill your money');
  exit;
  end
  else
  begin
  c:=inttostr(strtoint(b)-strtoint(a));
  fieldbyname('Money').AsString:=c;
  Edit6.Text:=Fieldbyname('Money').AsString + ' ' + 'Baht';
  addmoney;
  Timer3.Enabled:=true;
  end;
end;
if radiobutton5.Checked then   //bangna
begin
fieldbyname('Fee').AsString:='40';
a:=fieldbyname('Fee').AsString;
Edit2.Text:=a+ ' Baht';
b:=fieldbyname('Money').AsString;
  if strtoint(b) < strtoint(a) then
  begin
  showmessage('Please fill your money');
  exit;
  end
  else
  begin
  c:=inttostr(strtoint(b)-strtoint(a));
  fieldbyname('Money').AsString:=c;
  Edit6.Text:=Fieldbyname('Money').AsString + ' ' + 'Baht';
  addmoney;
```

```
        Timer3.Enabled:=true;
        end;
      end;
      end;
if byte4='32' then
      begin
      if radiobutton1.Checked then  //dindang
      begin
      Fieldbyname('Fee').AsString:='10';
      a:=Fieldbyname('Fee').AsString;
      Edit2.Text:=a+ 'Baht';
      b:=fieldbyname('Money').AsString;
        if strtoint(b) < strtoint(a) then
        begin
        showmessage('Please Fill Your Money');
        exit;
        end
        else
        begin
        c:=inttostr(strtoint(b)-strtoint(a));
        fieldbyname('Money').AsString:=c;
        Edit6.Text:=Fieldbyname('Money').AsString + ' ' + 'Baht';
        addmoney;
        Timer3.Enabled:=true;
        end;
      end;
      if radiobutton2.Checked then //sukumvit
      begin
      fieldbyname('Fee').AsString:='0';
      a:=fieldbyname('Fee').AsString;
      Edit2.Text:=a+ ' Baht';
      b:=fieldbyname('Money').AsString;
        if strtoint(b) < strtoint(a) then
        begin
        showmessage('Please fill your money');
        exit;
        end
        else
        begin
        c:=inttostr(strtoint(b)-strtoint(a));
        fieldbyname('Money').AsString:=c;
        Edit6.Text:=Fieldbyname('Money').AsString + ' ' + 'Baht';
        addmoney;
        Timer3.Enabled:=true;
        end;
      end;
      if radiobutton3.Checked then //argnarong
      begin
      fieldbyname('Fee').AsString:='10';
      a:=fieldbyname('Fee').AsString;
      Edit2.Text:=a+ ' Baht';
      b:=fieldbyname('Money').AsString;
        if strtoint(b) < strtoint(a) then
        begin
        showmessage('Please fill your money');
        exit;
        end
```

```
        else
        begin
        c:=inttostr(strtoint(b)-strtoint(a));
        fieldbyname('Money').AsString:=c;
        Edit6.Text:=Fieldbyname('Money').AsString + ' ' + 'Baht';
        addmoney;
        Timer3.Enabled:=true;
        end;
      end;
      if radiobutton4.Checked then //sukumvit 62
      begin
      fieldbyname('Fee').AsString:='20';
      a:=fieldbyname('Fee').AsString;
      Edit2.Text:=a+ ' Baht';
      b:=fieldbyname('Money').AsString;
        if strtoint(b) < strtoint(a) then
        begin
        showmessage('Please fill your money');
        exit;
        end
        else
        begin
        c:=inttostr(strtoint(b)-strtoint(a));
        fieldbyname('Money').AsString:=c;
        Edit6.Text:=Fieldbyname('Money').AsString + ' ' + 'Baht';
        addmoney;
        Timer3.Enabled:=true;
        end;
      end;
      if radiobutton5.Checked then    //bangna
      begin
      fieldbyname('Fee').AsString:='30';
      a:=fieldbyname('Fee').AsString;
      Edit2.Text:=a+ ' Baht';
      b:=fieldbyname('Money').AsString;
        if strtoint(b) < strtoint(a) then
        begin
        showmessage('Please fill your money');
        exit;
        end
        else
        begin
        c:=inttostr(strtoint(b)-strtoint(a));
        fieldbyname('Money').AsString:=c;
        Edit6.Text:=Fieldbyname('Money').AsString + ' ' + 'Baht';
        addmoney;
        Timer3.Enabled:=true;
        end;
      end;
      end;
if byte4='48' then
      begin
      if radiobutton1.Checked then  //dindang
      begin
      Fieldbyname('Fee').AsString:='20';
      a:=Fieldbyname('Fee').AsString;
      Edit2.Text:=a+ 'Baht';
```

```
b:=fieldbyname('Money').AsString;
  if strtoint(b) < strtoint(a) then
  begin
  showmessage('Please Fill Your Money');
  exit;
  end
  else
  begin
  c:=inttostr(strtoint(b)-strtoint(a));
  fieldbyname('Money').AsString:=c;
  Edit6.Text:=Fieldbyname('Money').AsString + ' ' + 'Baht';
  addmoney;
  Timer3.Enabled:=true;
  end;
end;
if radiobutton2.Checked then //sukumvit
begin
fieldbyname('Fee').AsString:='10';
a:=fieldbyname('Fee').AsString;
Edit2.Text:=a+ ' Baht';
b:=fieldbyname('Money').AsString;
  if strtoint(b) < strtoint(a) then
  begin
  showmessage('Please fill your money');
  exit;
  end
  else
  begin
  c:=inttostr(strtoint(b)-strtoint(a));
  fieldbyname('Money').AsString:=c;
  Edit6.Text:=Fieldbyname('Money').AsString + ' ' + 'Baht';
  addmoney;
  Timer3.Enabled:=true;
  end;
end;
if radiobutton3.Checked then //argnarong
begin
fieldbyname('Fee').AsString:='0';
a:=fieldbyname('Fee').AsString;
Edit2.Text:=a+ ' Baht';
b:=fieldbyname('Money').AsString;
  if strtoint(b) < strtoint(a) then
  begin
  showmessage('Please fill your money');
  exit;
  end
  else
  begin
  c:=inttostr(strtoint(b)-strtoint(a));
  fieldbyname('Money').AsString:=c;
  Edit6.Text:=Fieldbyname('Money').AsString + ' ' + 'Baht';
  addmoney;
  Timer3.Enabled:=true;
  end;
end;
if radiobutton4.Checked then //sukumvit 62
begin
```

```
        fieldbyname('Fee').AsString:='10';
        a:=fieldbyname('Fee').AsString;
        Edit2.Text:=a+ ' Baht';
        b:=fieldbyname('Money').AsString;
          if strtoint(b) < strtoint(a) then
          begin
          showmessage('Please fill your money');
          exit;
          end
          else
          begin
          c:=inttostr(strtoint(b)-strtoint(a));
          fieldbyname('Money').AsString:=c;
          Edit6.Text:=Fieldbyname('Money').AsString + ' ' + 'Baht';
          addmoney;
          Timer3.Enabled:=true;
          end;
        end;
        if radiobutton5.Checked then   //bangna
        begin
        fieldbyname('Fee').AsString:='20';
        a:=fieldbyname('Fee').AsString;
        Edit2.Text:=a+ ' Baht';
        b:=fieldbyname('Money').AsString;
          if strtoint(b) < strtoint(a) then
          begin
          showmessage('Please fill your money');
          exit;
          end
          else
          begin
          c:=inttostr(strtoint(b)-strtoint(a));
          fieldbyname('Money').AsString:=c;
          Edit6.Text:=Fieldbyname('Money').AsString + ' ' + 'Baht';
          addmoney;
          Timer3.Enabled:=true;
          end;
        end;
        end;
if byte4='64' then
        begin
        if radiobutton1.Checked then  //dindang
        begin
        Fieldbyname('Fee').AsString:='30';
        a:=Fieldbyname('Fee').AsString;
        Edit2.Text:=a+ 'Baht';
        b:=fieldbyname('Money').AsString;
          if strtoint(b) < strtoint(a) then
          begin
          showmessage('Please Fill Your Money');
          exit;
          end
          else
          begin
          c:=inttostr(strtoint(b)-strtoint(a));
          fieldbyname('Money').AsString:=c;
          Edit6.Text:=Fieldbyname('Money').AsString + ' ' + 'Baht';
```

```
  addmoney;
  Timer3.Enabled:=true;
  end;
end;
if radiobutton2.Checked then //sukumvit
begin
fieldbyname('Fee').AsString:='20';
a:=fieldbyname('Fee').AsString;
Edit2.Text:=a+ ' Baht';
b:=fieldbyname('Money').AsString;
  if strtoint(b) < strtoint(a) then
  begin
  showmessage('Please fill your money');
  exit;
  end
  else
  begin
  c:=inttostr(strtoint(b)-strtoint(a));
  fieldbyname('Money').AsString:=c;
  Edit6.Text:=Fieldbyname('Money').AsString + ' ' + 'Baht';
  addmoney;
  Timer3.Enabled:=true;
  end;
end;
if radiobutton3.Checked then //argnarong
begin
fieldbyname('Fee').AsString:='10';
a:=fieldbyname('Fee').AsString;
Edit2.Text:=a+ ' Baht';
b:=fieldbyname('Money').AsString;
  if strtoint(b) < strtoint(a) then
  begin
  showmessage('Please fill your money');
  exit;
  end
  else
  begin
  c:=inttostr(strtoint(b)-strtoint(a));
  fieldbyname('Money').AsString:=c;
  Edit6.Text:=Fieldbyname('Money').AsString + ' ' + 'Baht';
  addmoney;
  Timer3.Enabled:=true;
  end;
end;
if radiobutton4.Checked then //sukumvit 62
begin
fieldbyname('Fee').AsString:='0';
a:=fieldbyname('Fee').AsString;
Edit2.Text:=a+ ' Baht';
b:=fieldbyname('Money').AsString;
  if strtoint(b) < strtoint(a) then
  begin
  showmessage('Please fill your money');
  exit;
  end
  else
  begin
```

```
        c:=inttostr(strtoint(b)-strtoint(a));
        fieldbyname('Money').AsString:=c;
        Edit6.Text:=Fieldbyname('Money').AsString + ' ' + 'Baht';
        addmoney;
        Timer3.Enabled:=true;
        end;
      end;
      if radiobutton5.Checked then   //bangna
      begin
      fieldbyname('Fee').AsString:='10';
      a:=fieldbyname('Fee').AsString;
      Edit2.Text:=a+ ' Baht';
      b:=fieldbyname('Money').AsString;
        if strtoint(b) < strtoint(a) then
        begin
        showmessage('Please fill your money');
        exit;
        end
        else
        begin
        c:=inttostr(strtoint(b)-strtoint(a));
        fieldbyname('Money').AsString:=c;
        Edit6.Text:=Fieldbyname('Money').AsString + ' ' + 'Baht';
        addmoney;
        Timer3.Enabled:=true;
        end;
      end;
      end;
if byte4='80' then
      begin
      if radiobutton1.Checked then  //dindang
      begin
      Fieldbyname('Fee').AsString:='40';
      a:=Fieldbyname('Fee').AsString;
      Edit2.Text:=a+ 'Baht';
      b:=fieldbyname('Money').AsString;
        if strtoint(b) < strtoint(a) then
        begin
        showmessage('Please Fill Your Money');
        exit;
        end
        else
        begin
        c:=inttostr(strtoint(b)-strtoint(a));
        fieldbyname('Money').AsString:=c;
        Edit6.Text:=Fieldbyname('Money').AsString + ' ' + 'Baht';
        addmoney;
        Timer3.Enabled:=true;
        end;
      end;
      if radiobutton2.Checked then //sukumvit
      begin
      fieldbyname('Fee').AsString:='30';
      a:=fieldbyname('Fee').AsString;
      Edit2.Text:=a+ ' Baht';
      b:=fieldbyname('Money').AsString;
        if strtoint(b) < strtoint(a) then
```

```
  begin
  showmessage('Please fill your money');
  exit;
  end
  else
  begin
  c:=inttostr(strtoint(b)-strtoint(a));
  fieldbyname('Money').AsString:=c;
  Edit6.Text:=Fieldbyname('Money').AsString + ' ' + 'Baht';
  addmoney;
  Timer3.Enabled:=true;
  end;
end;
if radiobutton3.Checked then //argnarong
begin
fieldbyname('Fee').AsString:='20';
a:=fieldbyname('Fee').AsString;
Edit2.Text:=a+ ' Baht';
b:=fieldbyname('Money').AsString;
  if strtoint(b) < strtoint(a) then
  begin
  showmessage('Please fill your money');
  exit;
  end
  else
  begin
  c:=inttostr(strtoint(b)-strtoint(a));
  fieldbyname('Money').AsString:=c;
  Edit6.Text:=Fieldbyname('Money').AsString + ' ' + 'Baht';
  addmoney;
  Timer3.Enabled:=true;
  end;
end;
if radiobutton4.Checked then //sukumvit 62
begin
fieldbyname('Fee').AsString:='10';
a:=fieldbyname('Fee').AsString;
Edit2.Text:=a+ ' Baht';
b:=fieldbyname('Money').AsString;
  if strtoint(b) < strtoint(a) then
  begin
  showmessage('Please fill your money');
  exit;
  end
  else
  begin
  c:=inttostr(strtoint(b)-strtoint(a));
  fieldbyname('Money').AsString:=c;
  Edit6.Text:=Fieldbyname('Money').AsString + ' ' + 'Baht';
  addmoney;
  Timer3.Enabled:=true;
  end;
end;
if radiobutton5.Checked then   //bangna
begin
fieldbyname('Fee').AsString:='0';
a:=fieldbyname('Fee').AsString;
```

```
        Edit2.Text:=a+ ' Baht';
        b:=fieldbyname('Money').AsString;
          if strtoint(b) < strtoint(a) then
          begin
          showmessage('Please fill your money');
          exit;
          end
          else
          begin
          c:=inttostr(strtoint(b)-strtoint(a));
          fieldbyname('Money').AsString:=c;
          Edit6.Text:=Fieldbyname('Money').AsString + ' ' + 'Baht';
          addmoney;
          Timer3.Enabled:=true;
          end;
        end;
        end;
        post;
        Timer4.Enabled:=true;
end;
end
else
exit;
end;
end;
procedure TForm3.Timer1Timer(Sender: TObject);
begin
   DBEdit1.Text:=datetostr(date);
   DBEdit2.Text:=timetostr(time);
end;
procedure TForm3.addmoney;
var st,find:string;
    aa:integer;
begin
    with query do
    find:=fieldbyname('Money').asstring;
    aa:=strtoint(find);
      if aa=0 then
st:=chr($FB)+chr($04)+chr($01)+chr($90)+chr($00)+chr($00)+chr($00)
      +chr($6E)+chr($FC);
      if aa=10 then
st:=chr($FB)+chr($04)+chr($01)+chr($90)+chr($00)+chr($01)+chr($00)
      +chr($6F)+chr($FC);
      if aa=20 then
st:=chr($FB)+chr($04)+chr($01)+chr($90)+chr($00)+chr($02)+chr($00)
      +chr($6C)+chr($FC);
      if aa=30 then
st:=chr($FB)+chr($04)+chr($01)+chr($90)+chr($00)+chr($03)+chr($00)
      +chr($6D)+chr($FC);
      if aa=40 then
st:=chr($FB)+chr($04)+chr($01)+chr($90)+chr($00)+chr($04)+chr($00)
      +chr($6A)+chr($FC);
      if aa=50 then
st:=chr($FB)+chr($04)+chr($01)+chr($90)+chr($00)+chr($05)+chr($00)
      +chr($6B)+chr($FC);
      if aa=60 then
st:=chr($FB)+chr($04)+chr($01)+chr($90)+chr($00)+chr($06)+chr($00)
```

```
      +chr($68)+chr($FC);
      if aa=70 then
st:=chr($FB)+chr($04)+chr($01)+chr($90)+chr($00)+chr($07)+chr($00)
      +chr($69)+chr($FC);
      if aa=80 then
st:=chr($FB)+chr($04)+chr($01)+chr($90)+chr($00)+chr($08)+chr($00)
      +chr($66)+chr($FC);
      if aa=90 then
st:=chr($FB)+chr($04)+chr($01)+chr($90)+chr($00)+chr($09)+chr($00)
      +chr($67)+chr($FC);
      if aa=100 then
st:=chr($FB)+chr($04)+chr($01)+chr($90)+chr($01)+chr($00)+chr($00)
      +chr($6F)+chr($FC);
      if aa=110 then
st:=chr($FB)+chr($04)+chr($01)+chr($90)+chr($01)+chr($01)+chr($00)
      +chr($6E)+chr($FC);
      if aa=120 then
st:=chr($FB)+chr($04)+chr($01)+chr($90)+chr($01)+chr($02)+chr($00)
      +chr($6D)+chr($FC);
      if aa=130 then
st:=chr($FB)+chr($04)+chr($01)+chr($90)+chr($01)+chr($03)+chr($00)
      +chr($6C)+chr($FC);
      if aa=140 then
st:=chr($FB)+chr($04)+chr($01)+chr($90)+chr($01)+chr($04)+chr($00)
      +chr($6B)+chr($FC);
      if aa=150 then
st:=chr($FB)+chr($04)+chr($01)+chr($90)+chr($01)+chr($05)+chr($00)
      +chr($6A)+chr($FC);
      if aa=160 then
st:=chr($FB)+chr($04)+chr($01)+chr($90)+chr($01)+chr($06)+chr($00)
      +chr($69)+chr($FC);
      if aa=170 then
st:=chr($FB)+chr($04)+chr($01)+chr($90)+chr($01)+chr($07)+chr($00)
      +chr($68)+chr($FC);
      if aa=180 then
st:=chr($FB)+chr($04)+chr($01)+chr($90)+chr($01)+chr($08)+chr($00)
      +chr($67)+chr($FC);
      if aa=190 then
st:=chr($FB)+chr($04)+chr($01)+chr($90)+chr($01)+chr($09)+chr($00)
      +chr($66)+chr($FC);
      if aa=200 then
st:=chr($FB)+chr($04)+chr($01)+chr($90)+chr($02)+chr($00)+chr($00)
      +chr($6C)+chr($FC);
      if aa=210 then
st:=chr($FB)+chr($04)+chr($01)+chr($90)+chr($02)+chr($01)+chr($00)
      +chr($6D)+chr($FC);
      if aa=220 then
st:=chr($FB)+chr($04)+chr($01)+chr($90)+chr($02)+chr($02)+chr($00)
      +chr($6E)+chr($FC);
      if aa=230 then
st:=chr($FB)+chr($04)+chr($01)+chr($90)+chr($02)+chr($03)+chr($00)
      +chr($6F)+chr($FC);
      if aa=240 then
st:=chr($FB)+chr($04)+chr($01)+chr($90)+chr($02)+chr($04)+chr($00)
      +chr($68)+chr($FC);
      if aa=250 then
st:=chr($FB)+chr($04)+chr($01)+chr($90)+chr($02)+chr($05)+chr($00)
```

```
      +chr($69)+chr($FC);
      if aa=260 then
st:=chr($FB)+chr($04)+chr($01)+chr($90)+chr($02)+chr($06)+chr($00)
      +chr($6A)+chr($FC);
      if aa=270 then
st:=chr($FB)+chr($04)+chr($01)+chr($90)+chr($02)+chr($07)+chr($00)
      +chr($6B)+chr($FC);
      if aa=280 then
st:=chr($FB)+chr($04)+chr($01)+chr($90)+chr($02)+chr($08)+chr($00)
      +chr($64)+chr($FC);
      if aa=290 then
st:=chr($FB)+chr($04)+chr($01)+chr($90)+chr($02)+chr($09)+chr($00)
      +chr($65)+chr($FC);
      if aa=300 then
st:=chr($FB)+chr($04)+chr($01)+chr($90)+chr($03)+chr($00)+chr($00)
      +chr($6D)+chr($FC);
      if aa=310 then
st:=chr($FB)+chr($04)+chr($01)+chr($90)+chr($03)+chr($01)+chr($00)
      +chr($6C)+chr($FC);
      if aa=320 then
st:=chr($FB)+chr($04)+chr($01)+chr($90)+chr($03)+chr($02)+chr($00)
      +chr($6F)+chr($FC);
      if aa=330 then
st:=chr($FB)+chr($04)+chr($01)+chr($90)+chr($03)+chr($03)+chr($00)
      +chr($6E)+chr($FC);
      if aa=340 then
st:=chr($FB)+chr($04)+chr($01)+chr($90)+chr($03)+chr($04)+chr($00)
      +chr($69)+chr($FC);
      if aa=350 then
st:=chr($FB)+chr($04)+chr($01)+chr($90)+chr($03)+chr($05)+chr($00)
      +chr($68)+chr($FC);
      if aa=360 then
st:=chr($FB)+chr($04)+chr($01)+chr($90)+chr($03)+chr($06)+chr($00)
      +chr($6B)+chr($FC);
      if aa=370 then
st:=chr($FB)+chr($04)+chr($01)+chr($90)+chr($03)+chr($07)+chr($00)
      +chr($6A)+chr($FC);
      if aa=380 then
st:=chr($FB)+chr($04)+chr($01)+chr($90)+chr($03)+chr($08)+chr($00)
      +chr($65)+chr($FC);
      if aa=390 then
st:=chr($FB)+chr($04)+chr($01)+chr($90)+chr($03)+chr($09)+chr($00)
      +chr($64)+chr($FC);
      if aa=400 then
st:=chr($FB)+chr($04)+chr($01)+chr($90)+chr($04)+chr($00)+chr($00)
      +chr($6A)+chr($FC);
      if aa=410 then
st:=chr($FB)+chr($04)+chr($01)+chr($90)+chr($04)+chr($01)+chr($00)
      +chr($6B)+chr($FC);
      if aa=420 then
st:=chr($FB)+chr($04)+chr($01)+chr($90)+chr($04)+chr($02)+chr($00)
      +chr($68)+chr($FC);
      if aa=430 then
st:=chr($FB)+chr($04)+chr($01)+chr($90)+chr($04)+chr($03)+chr($00)
      +chr($69)+chr($FC);
      if aa=440 then
st:=chr($FB)+chr($04)+chr($01)+chr($90)+chr($04)+chr($04)+chr($00)
```

```
      +chr($6E)+chr($FC);
      if aa=450 then
st:=chr($FB)+chr($04)+chr($01)+chr($90)+chr($04)+chr($05)+chr($00)
      +chr($6F)+chr($FC);
      if aa=460 then
st:=chr($FB)+chr($04)+chr($01)+chr($90)+chr($04)+chr($06)+chr($00)
      +chr($6C)+chr($FC);
      if aa=470 then
st:=chr($FB)+chr($04)+chr($01)+chr($90)+chr($04)+chr($07)+chr($00)
      +chr($6D)+chr($FC);
      if aa=480 then
st:=chr($FB)+chr($04)+chr($01)+chr($90)+chr($04)+chr($08)+chr($00)
      +chr($62)+chr($FC);
      if aa=490 then
st:=chr($FB)+chr($04)+chr($01)+chr($90)+chr($04)+chr($09)+chr($00)
      +chr($63)+chr($FC);
      if aa=500 then
st:=chr($FB)+chr($04)+chr($01)+chr($90)+chr($05)+chr($00)+chr($00)
      +chr($6B)+chr($FC);
      if aa=510 then
st:=chr($FB)+chr($04)+chr($01)+chr($90)+chr($05)+chr($01)+chr($00)
      +chr($6A)+chr($FC);
      if aa=520 then
st:=chr($FB)+chr($04)+chr($01)+chr($90)+chr($05)+chr($02)+chr($00)
      +chr($69)+chr($FC);
      if aa=530 then
st:=chr($FB)+chr($04)+chr($01)+chr($90)+chr($05)+chr($03)+chr($00)
      +chr($68)+chr($FC);
      if aa=540 then
st:=chr($FB)+chr($04)+chr($01)+chr($90)+chr($05)+chr($04)+chr($00)
      +chr($6F)+chr($FC);
      if aa=550 then
st:=chr($FB)+chr($04)+chr($01)+chr($90)+chr($05)+chr($05)+chr($00)
      +chr($6E)+chr($FC);
      if aa=560 then
st:=chr($FB)+chr($04)+chr($01)+chr($90)+chr($05)+chr($06)+chr($00)
      +chr($6D)+chr($FC);
      if aa=570 then
st:=chr($FB)+chr($04)+chr($01)+chr($90)+chr($05)+chr($07)+chr($00)
      +chr($6C)+chr($FC);
      if aa=580 then
st:=chr($FB)+chr($04)+chr($01)+chr($90)+chr($05)+chr($08)+chr($00)
      +chr($63)+chr($FC);
      if aa=590 then
st:=chr($FB)+chr($04)+chr($01)+chr($90)+chr($05)+chr($09)+chr($00)
      +chr($62)+chr($FC);
      if aa=600 then
st:=chr($FB)+chr($04)+chr($01)+chr($90)+chr($06)+chr($00)+chr($00)
      +chr($68)+chr($FC);
      if aa=610 then
st:=chr($FB)+chr($04)+chr($01)+chr($90)+chr($06)+chr($01)+chr($00)
      +chr($69)+chr($FC);
      if aa=620 then
st:=chr($FB)+chr($04)+chr($01)+chr($90)+chr($06)+chr($02)+chr($00)
      +chr($6A)+chr($FC);
      if aa=630 then
st:=chr($FB)+chr($04)+chr($01)+chr($90)+chr($06)+chr($03)+chr($00)
```

```
      +chr($6B)+chr($FC);
      if aa=640 then
st:=chr($FB)+chr($04)+chr($01)+chr($90)+chr($06)+chr($04)+chr($00)
      +chr($6C)+chr($FC);
      if aa=650 then
st:=chr($FB)+chr($04)+chr($01)+chr($90)+chr($06)+chr($05)+chr($00)
      +chr($6D)+chr($FC);
      if aa=660 then
st:=chr($FB)+chr($04)+chr($01)+chr($90)+chr($06)+chr($06)+chr($00)
      +chr($6E)+chr($FC);
      if aa=670 then
st:=chr($FB)+chr($04)+chr($01)+chr($90)+chr($06)+chr($07)+chr($00)
      +chr($6F)+chr($FC);
      if aa=680 then
st:=chr($FB)+chr($04)+chr($01)+chr($90)+chr($06)+chr($08)+chr($00)
      +chr($60)+chr($FC);
      if aa=690 then
st:=chr($FB)+chr($04)+chr($01)+chr($90)+chr($06)+chr($09)+chr($00)
      +chr($61)+chr($FC);
      if aa=700 then
st:=chr($FB)+chr($04)+chr($01)+chr($90)+chr($07)+chr($00)+chr($00)
      +chr($69)+chr($FC);
      if aa=710 then
st:=chr($FB)+chr($04)+chr($01)+chr($90)+chr($07)+chr($01)+chr($00)
      +chr($68)+chr($FC);
      if aa=720 then
st:=chr($FB)+chr($04)+chr($01)+chr($90)+chr($07)+chr($02)+chr($00)
      +chr($6B)+chr($FC);
      if aa=730 then
st:=chr($FB)+chr($04)+chr($01)+chr($90)+chr($07)+chr($03)+chr($00)
      +chr($6A)+chr($FC);
      if aa=740 then
st:=chr($FB)+chr($04)+chr($01)+chr($90)+chr($07)+chr($04)+chr($00)
      +chr($6D)+chr($FC);
      if aa=750 then
st:=chr($FB)+chr($04)+chr($01)+chr($90)+chr($07)+chr($05)+chr($00)
      +chr($6C)+chr($FC);
      if aa=760 then
st:=chr($FB)+chr($04)+chr($01)+chr($90)+chr($07)+chr($06)+chr($00)
      +chr($6F)+chr($FC);
      if aa=770 then
st:=chr($FB)+chr($04)+chr($01)+chr($90)+chr($07)+chr($07)+chr($00)
      +chr($6E)+chr($FC);
      if aa=780 then
st:=chr($FB)+chr($04)+chr($01)+chr($90)+chr($07)+chr($08)+chr($00)
      +chr($61)+chr($FC);
      if aa=790 then
st:=chr($FB)+chr($04)+chr($01)+chr($90)+chr($07)+chr($09)+chr($00)
      +chr($60)+chr($FC);
      if aa=800 then
st:=chr($FB)+chr($04)+chr($01)+chr($90)+chr($08)+chr($00)+chr($00)
      +chr($66)+chr($FC);
      if aa=810 then
st:=chr($FB)+chr($04)+chr($01)+chr($90)+chr($08)+chr($01)+chr($00)
      +chr($67)+chr($FC);
      if aa=820 then
st:=chr($FB)+chr($04)+chr($01)+chr($90)+chr($08)+chr($02)+chr($00)
```

```
      +chr($64)+chr($FC);
      if aa=830 then
st:=chr($FB)+chr($04)+chr($01)+chr($90)+chr($08)+chr($03)+chr($00)
      +chr($65)+chr($FC);
      if aa=840 then
st:=chr($FB)+chr($04)+chr($01)+chr($90)+chr($08)+chr($04)+chr($00)
      +chr($62)+chr($FC);
      if aa=850 then
st:=chr($FB)+chr($04)+chr($01)+chr($90)+chr($08)+chr($05)+chr($00)
      +chr($63)+chr($FC);
      if aa=860 then
st:=chr($FB)+chr($04)+chr($01)+chr($90)+chr($08)+chr($06)+chr($00)
      +chr($60)+chr($FC);
      if aa=870 then
st:=chr($FB)+chr($04)+chr($01)+chr($90)+chr($08)+chr($07)+chr($00)
      +chr($61)+chr($FC);
      if aa=880 then
st:=chr($FB)+chr($04)+chr($01)+chr($90)+chr($08)+chr($08)+chr($00)
      +chr($6E)+chr($FC);
      if aa=890 then
st:=chr($FB)+chr($04)+chr($01)+chr($90)+chr($08)+chr($09)+chr($00)
      +chr($6F)+chr($FC);
      if aa=900 then
st:=chr($FB)+chr($04)+chr($01)+chr($90)+chr($09)+chr($00)+chr($00)
      +chr($67)+chr($FC);
      if aa=910 then
st:=chr($FB)+chr($04)+chr($01)+chr($90)+chr($09)+chr($01)+chr($00)
      +chr($66)+chr($FC);
      if aa=920 then
st:=chr($FB)+chr($04)+chr($01)+chr($90)+chr($09)+chr($02)+chr($00)
      +chr($65)+chr($FC);
      if aa=930 then
st:=chr($FB)+chr($04)+chr($01)+chr($90)+chr($09)+chr($03)+chr($00)
      +chr($64)+chr($FC);
      if aa=940 then
st:=chr($FB)+chr($04)+chr($01)+chr($90)+chr($09)+chr($04)+chr($00)
      +chr($63)+chr($FC);
      if aa=950 then
st:=chr($FB)+chr($04)+chr($01)+chr($90)+chr($09)+chr($05)+chr($00)
      +chr($62)+chr($FC);
      if aa=960 then
st:=chr($FB)+chr($04)+chr($01)+chr($90)+chr($09)+chr($06)+chr($00)
      +chr($61)+chr($FC);
      if aa=970 then
st:=chr($FB)+chr($04)+chr($01)+chr($90)+chr($09)+chr($07)+chr($00)
      +chr($60)+chr($FC);
      if aa=980 then
st:=chr($FB)+chr($04)+chr($01)+chr($90)+chr($09)+chr($08)+chr($00)
      +chr($6F)+chr($FC);
      if aa=990 then
st:=chr($FB)+chr($04)+chr($01)+chr($90)+chr($09)+chr($09)+chr($00)
      +chr($6E)+chr($FC);
      if aa=1000 then
st:=chr($FB)+chr($04)+chr($01)+chr($90)+chr($10)+chr($00)+chr($00)
      +chr($7E)+chr($FC);
    Vacomm1.writetext(st);
  end;
```

```
{procedure TForm3.BitBtn3Click(Sender: TObject);
begin
        if byte5='0' then
          begin
            if byte6='0'  then Edit6.Text:='0' + ' ' + 'Baht';
            if byte6='1'  then Edit6.Text:='10' + ' ' + 'Baht';
            if byte6='2'  then Edit6.Text:='20' + ' ' + 'Baht';
            if byte6='3'  then Edit6.Text:='30' + ' ' + ' Baht';
            if byte6='4'  then Edit6.Text:='40' + ' ' + ' Baht';
            if byte6='5'  then Edit6.Text:='50' + ' ' + ' Baht';
            if byte6='6'  then Edit6.Text:='60' + ' ' + ' Baht';
            if byte6='7'  then Edit6.Text:='70' + ' ' + ' Baht';
            if byte6='8'  then Edit6.Text:='80' + ' ' + ' Baht';
            if byte6='9'  then Edit6.Text:='90' + ' ' + ' Baht';
          end;
       if byte5='1' then
          begin
            if byte6='0'  then Edit6.Text:='100' + ' ' + ' Baht';
            if byte6='1'  then Edit6.Text:='110' + ' ' + ' Baht';
            if byte6='2'  then Edit6.Text:='120' + ' ' + ' Baht';
            if byte6='3'  then Edit6.Text:='130' + ' ' + ' Baht';
            if byte6='4'  then Edit6.Text:='140' + ' ' + ' Baht';
            if byte6='5'  then Edit6.Text:='150' + ' ' + ' Baht';
            if byte6='6'  then Edit6.Text:='160' + ' ' + ' Baht';
            if byte6='7'  then Edit6.Text:='170' + ' ' + ' Baht';
            if byte6='8'  then Edit6.Text:='180' + ' ' + ' Baht';
            if byte6='9'  then Edit6.Text:='190' + ' ' + ' Baht';
          end;
       if byte5='2' then
          begin
            if byte6='0'  then Edit6.Text:='200' + ' ' + ' Baht';
            if byte6='1'  then Edit6.Text:='210' + ' ' + ' Baht';
            if byte6='2'  then Edit6.Text:='220' + ' ' + ' Baht';
            if byte6='3'  then Edit6.Text:='230' + ' ' + ' Baht';
            if byte6='4'  then Edit6.Text:='240' + ' ' + ' Baht';
            if byte6='5'  then Edit6.Text:='250' + ' ' + ' Baht';
            if byte6='6'  then Edit6.Text:='260' + ' ' + ' Baht';
            if byte6='7'  then Edit6.Text:='270' + ' ' + ' Baht';
            if byte6='8'  then Edit6.Text:='280' + ' ' + ' Baht';
            if byte6='9'  then Edit6.Text:='290' + ' ' + ' Baht';
          end;
       if byte5='3' then
          begin
            if byte6='0'  then Edit6.Text:='300' + ' ' + ' Baht';
            if byte6='1'  then Edit6.Text:='310' + ' ' + ' Baht';
            if byte6='2'  then Edit6.Text:='320' + ' ' + ' Baht';
            if byte6='3'  then Edit6.Text:='330' + ' ' + ' Baht';
            if byte6='4'  then Edit6.Text:='340' + ' ' + ' Baht';
            if byte6='5'  then Edit6.Text:='350' + ' ' + ' Baht';
            if byte6='6'  then Edit6.Text:='360' + ' ' + ' Baht';
            if byte6='7'  then Edit6.Text:='370' + ' ' + ' Baht';
            if byte6='8'  then Edit6.Text:='380' + ' ' + ' Baht';
            if byte6='9'  then Edit6.Text:='390' + ' ' + ' Baht';
          end;
        if byte5='4' then
          begin
            if byte6='0'  then Edit6.Text:='400' + ' ' + ' Baht';
```

```
    if byte6='1'  then Edit6.Text:='410' + ' ' + ' Baht';
    if byte6='2'  then Edit6.Text:='420' + ' ' + ' Baht';
    if byte6='3'  then Edit6.Text:='430' + ' ' + ' Baht';
    if byte6='4'  then Edit6.Text:='440' + ' ' + ' Baht';
    if byte6='5'  then Edit6.Text:='450' + ' ' + ' Baht';
    if byte6='6'  then Edit6.Text:='460' + ' ' + ' Baht';
    if byte6='7'  then Edit6.Text:='470' + ' ' + ' Baht';
    if byte6='8'  then Edit6.Text:='480' + ' ' + ' Baht';
    if byte6='9'  then Edit6.Text:='490' + ' ' + ' Baht';
  end;
if byte5='5' then
  begin
    if byte6='0'  then Edit6.Text:='500' + ' ' + ' Baht';
    if byte6='1'  then Edit6.Text:='510' + ' ' + ' Baht';
    if byte6='2'  then Edit6.Text:='520' + ' ' + ' Baht';
    if byte6='3'  then Edit6.Text:='530' + ' ' + ' Baht';
    if byte6='4'  then Edit6.Text:='540' + ' ' + ' Baht';
    if byte6='5'  then Edit6.Text:='550' + ' ' + ' Baht';
    if byte6='6'  then Edit6.Text:='560' + ' ' + ' Baht';
    if byte6='7'  then Edit6.Text:='570' + ' ' + ' Baht';
    if byte6='8'  then Edit6.Text:='580' + ' ' + ' Baht';
    if byte6='9'  then Edit6.Text:='590' + ' ' + ' Baht';
  end;
if byte5='6' then
  begin
    if byte6='0'  then Edit6.Text:='600' + ' ' + ' Baht';
    if byte6='1'  then Edit6.Text:='610' + ' ' + ' Baht';
    if byte6='2'  then Edit6.Text:='620' + ' ' + ' Baht';
    if byte6='3'  then Edit6.Text:='630' + ' ' + ' Baht';
    if byte6='4'  then Edit6.Text:='640' + ' ' + ' Baht';
    if byte6='5'  then Edit6.Text:='650' + ' ' + ' Baht';
    if byte6='6'  then Edit6.Text:='660' + ' ' + ' Baht';
    if byte6='7'  then Edit6.Text:='670' + ' ' + ' Baht';
    if byte6='8'  then Edit6.Text:='680' + ' ' + ' Baht';
    if byte6='9'  then Edit6.Text:='690' + ' ' + ' Baht';
  end;
if byte5='7' then
  begin
    if byte6='0'  then Edit6.Text:='700' + ' ' + ' Baht';
    if byte6='1'  then Edit6.Text:='710' + ' ' + ' Baht';
    if byte6='2'  then Edit6.Text:='720' + ' ' + ' Baht';
    if byte6='3'  then Edit6.Text:='730' + ' ' + ' Baht';
    if byte6='4'  then Edit6.Text:='740' + ' ' + ' Baht';
    if byte6='5'  then Edit6.Text:='750' + ' ' + ' Baht';
    if byte6='6'  then Edit6.Text:='760' + ' ' + ' Baht';
    if byte6='7'  then Edit6.Text:='770' + ' ' + ' Baht';
    if byte6='8'  then Edit6.Text:='780' + ' ' + ' Baht';
    if byte6='9'  then Edit6.Text:='790' + ' ' + ' Baht';
  end;
if byte5='8' then
 begin
    if byte6='0'  then Edit6.Text:='800' + ' ' + ' Baht';
    if byte6='1'  then Edit6.Text:='810' + ' ' + ' Baht';
    if byte6='2'  then Edit6.Text:='820' + ' ' + ' Baht';
    if byte6='3'  then Edit6.Text:='830' + ' ' + ' Baht';
    if byte6='4'  then Edit6.Text:='840' + ' ' + ' Baht';
    if byte6='5'  then Edit6.Text:='850' + ' ' + ' Baht';
```

```
            if byte6='6'  then Edit6.Text:='860' + ' ' + ' Baht';
            if byte6='7'  then Edit6.Text:='870' + ' ' + ' Baht';
            if byte6='8'  then Edit6.Text:='880' + ' ' + ' Baht';
            if byte6='9'  then Edit6.Text:='890' + ' ' + ' Baht';
          end;
        if byte5='9' then
          begin
            if byte6='0'  then Edit6.Text:='900' + ' ' + ' Baht';
            if byte6='1'  then Edit6.Text:='910' + ' ' + ' Baht';
            if byte6='2'  then Edit6.Text:='920' + ' ' + ' Baht';
            if byte6='3'  then Edit6.Text:='930' + ' ' + ' Baht';
            if byte6='4'  then Edit6.Text:='940' + ' ' + ' Baht';
            if byte6='5'  then Edit6.Text:='950' + ' ' + ' Baht';
            if byte6='6'  then Edit6.Text:='960' + ' ' + ' Baht';
            if byte6='7'  then Edit6.Text:='970' + ' ' + ' Baht';
            if byte6='8'  then Edit6.Text:='980' + ' ' + ' Baht';
            if byte6='9'  then Edit6.Text:='990' + ' ' + ' Baht';
          end;
        if byte5 = '16' then Edit6.Text:='1000' + ' ' + ' Baht';
end;}
procedure TForm3.Timer3Timer(Sender: TObject);
var st:string;
begin
    st:=chr($FB)+chr($02)+chr($FF)+chr($06)+chr($FC);
    Vacomm1.WriteText(st);
    Timer3.Enabled:=false;
    Timer4.Enabled:=true;
end;
procedure TForm3.Timer4Timer(Sender: TObject);
begin
    Edit1.Text:='';
    Edit5.Text:='';
    radiobutton1.Checked:=false;
    radiobutton2.Checked:=false;
    radiobutton3.Checked:=false;
    radiobutton4.Checked:=false;
    radiobutton5.Checked:=false;
    Timer4.Enabled:=false;
end;
procedure TForm3.BitBtn2Click(Sender: TObject);
begin
  Edit1.Text:='';
  radiobutton1.Checked := false;
  radiobutton2.Checked := false;
  radiobutton3.Checked := false;
  radiobutton4.Checked := false;
  radiobutton5.Checked := false;
  Edit2.Text:='';
  Edit5.Text:='';
  Edit6.Text:='';
end;
procedure TForm3.BitBtn1Click(Sender: TObject);
var st:string;
begin
    st:=chr($FB)+chr($02)+chr($FF)+chr($06)+chr($FC);
    Vacomm1.WriteText(st);
end;
```

```
procedure TForm3.Edit4Change(Sender: TObject);
Var
  a,b,st: String;
  i:Integer;
begin
    if Edit4.Text='START' then
    begin
    if (radiobutton1.Checked = true) or
       (radiobutton2.Checked = true) or
       (radiobutton3.Checked = true) or
       (radiobutton4.Checked = true) or
       (radiobutton5.Checked = true) then
       begin
       st:=chr($FB)+chr($02)+chr($FF)+chr($06)+chr($FC);
       Vacomm1.WriteText(st);
       end
       else
       Showmessage('Please Select Station First!!!');
    end;
end;
procedure TForm3.Timer5Timer(Sender: TObject);
begin
  port := IntToStr(PortIn($379));
  if port = '127' then
  Edit4.Text:='START'
  else
  if port = '63' then
  Edit4.Text:='STOP';
end;
procedure TForm3.ExitBtnClick(Sender: TObject);
begin
    Vacomm1.Close;
    Close;
    Edit7.Clear;
    Edit8.Clear;
    Edit2.Clear;
    Edit6.Clear;
end;
procedure TForm3.FormActivate(Sender: TObject);
var resp,resp1:word;
begin
  resp:=MessageDlg('Open Port?',mtConfirmation,[mbYes,mbNo],0);
  if resp = mrYes then
  Vacomm1.Open;
  if resp = mrNo then
  Showmessage('Cannot Connect To The Port');
  Close;
end;
end.
```

Add Money

```
unit Unit4;

interface

uses
  Windows, Messages, SysUtils, Variants, Classes, Graphics, Controls,
Forms,
  Dialogs, DBTables, XPMan, ExtCtrls, DBAccess, MyAccess, DB, MemDS,
  MyDacVcl, VaClasses, VaComm, Grids, DBGrids, StdCtrls, Buttons,
ComCtrls;

type
  TForm4 = class(TForm)
    Panel1: TPanel;
    Label1: TLabel;
    PageControl1: TPageControl;
    TabSheet1: TTabSheet;
    BitBtn4: TBitBtn;
    Edit5: TEdit;
    GroupBox1: TGroupBox;
    RadioButton1: TRadioButton;
    RadioButton2: TRadioButton;
    RadioButton3: TRadioButton;
    RadioButton4: TRadioButton;
    RadioButton5: TRadioButton;
    RadioButton6: TRadioButton;
    RadioButton7: TRadioButton;
    RadioButton8: TRadioButton;
    RadioButton9: TRadioButton;
    RadioButton10: TRadioButton;
    BitBtn1: TBitBtn;
    BitBtn2: TBitBtn;
    Edit3: TEdit;
    BitBtn3: TBitBtn;
    TabSheet2: TTabSheet;
    Label5: TLabel;
    GroupBox2: TGroupBox;
    Label4: TLabel;
    Label2: TLabel;
    Label3: TLabel;
    Label6: TLabel;
    Label7: TLabel;
    Edit6: TEdit;
    Edit7: TEdit;
    Edit8: TEdit;
    Edit9: TEdit;
    Edit10: TEdit;
    VaComm1: TVaComm;
    MyConnection: TMyConnection;
    DataSource: TDataSource;
    ConnectDialog: TMyConnectDialog;
    Query: TMyQuery;
    sqCreateTable: TMyCommand;
    sqDropTable: TMyCommand;
    XPManifest1: TXPManifest;
    Timer5: TTimer;
```

```
    Timer6: TTimer;
    DataSource1: TDataSource;
    Table1: TTable;
    Timer7: TTimer;
    Edit1: TEdit;
    TabSheet3: TTabSheet;
    DBGrid1: TDBGrid;
    btOpen: TButton;
    btClose: TButton;
    btCreate: TButton;
    btDrop: TButton;
    ExitBtn: TSpeedButton;
    BitBtn5: TBitBtn;
    procedure BitBtn2Click(Sender: TObject);
    procedure VaComm1RxChar(Sender: TObject; Count: Integer);
    procedure BitBtn1Click(Sender: TObject);
    procedure BitBtn3Click(Sender: TObject);
    procedure BitBtn4Click(Sender: TObject);
    procedure TabSheet2Show(Sender: TObject);
    procedure Timer5Timer(Sender: TObject);
    procedure Timer6Timer(Sender: TObject);
    procedure Timer7Timer(Sender: TObject);
    procedure addmoney;
    procedure ExitBtnClick(Sender: TObject);
    procedure FormActivate(Sender: TObject);
    procedure btOpenClick(Sender: TObject);
    procedure btCloseClick(Sender: TObject);
    procedure btCreateClick(Sender: TObject);
    procedure btDropClick(Sender: TObject);
    procedure database;
    procedure BitBtn5Click(Sender: TObject);
  private
    { Private declarations }
  public
    { Public declarations }
  end;

var
  Form4: TForm4;
  a,b,c,d,e,z,AA:integer;
  count:integer;
  byte5,byte6,byte7,p:string;
implementation

{$R *.dfm}

procedure TForm4.BitBtn2Click(Sender: TObject);
var st:string;
begin
    st:=chr($FB)+chr($02)+chr($FF)+chr($06)+chr($FC);
    Vacomm1.WriteText(st);
    Edit3.Text:=inttostr(z);
    Timer7.Enabled:=true;
end;
procedure TForm4.VaComm1RxChar(Sender: TObject; Count: Integer);
var  St:string;
     AA:integer;
```

```
Begin
  if count >= 33 then
  begin
    St := Vacomm1.ReadText;
    Edit1.Text := '';
    for AA := 1 to 33 do
    begin
    Edit1.text := Edit1.Text + ' ' + inttostr((ord(St[AA])));
    end;
    a:=ord(St[4]);
    b:=ord(St[5]);
    c:=ord(St[6]);
    d:=ord(St[7]);
    e:=ord(St[2]);
    Edit3.Text:=inttostr(ord(St[31]));
    byte5:=inttostr(ord(St[5]));
    byte6:=inttostr(ord(St[6]));
    byte7:=inttostr(ord(St[7]));
end;
end;
procedure TForm4.BitBtn1Click(Sender: TObject);
var st,a,b,c:string;
    aa,bb,cc,dd:integer;
begin
  with query do
  if radiobutton1.Checked then
      database;
      addmoney;
      Timer5.Enabled:=true;
      Timer6.Enabled:=true;
  if radiobutton2.Checked then
      database;
      addmoney;
      Timer5.Enabled:=true;
      Timer6.Enabled:=true;
  if radiobutton3.Checked then
      database;
      addmoney;
      Timer5.Enabled:=true;
      Timer6.Enabled:=true;
  if radiobutton4.Checked then
      database;
      addmoney;
      Timer5.Enabled:=true;
      Timer6.Enabled:=true;
  if radiobutton5.Checked then
      database;
      addmoney;
      Timer5.Enabled:=true;
      Timer6.Enabled:=true;
  if radiobutton6.Checked then
      database;
      addmoney;
      Timer5.Enabled:=true;
      Timer6.Enabled:=true;
  if radiobutton7.Checked then
      database;
```

```
        addmoney;
        Timer5.Enabled:=true;
        Timer6.Enabled:=true;
   if radiobutton8.Checked then
        database;
        addmoney;
        Timer5.Enabled:=true;
        Timer6.Enabled:=true;
   if radiobutton9.Checked then
        database;
        addmoney;
        Timer5.Enabled:=true;
        Timer6.Enabled:=true;
   if radiobutton10.Checked then
        database;
        addmoney;
        Timer5.Enabled:=true;
        Timer6.Enabled:=true;
end;
procedure TForm4.BitBtn3Click(Sender: TObject);
begin
  Edit1.Text:='';
  Edit3.Text:='';
  Edit5.Text:='';
end;
procedure TForm4.database;
var st,aaa,s,old,money:string;
    sum:integer;
begin
  with query do
  begin
  append;
  Fieldbyname('ID').AsString:=Edit3.Text;
  Fieldbyname('Date').AsString:=DatetoStr(Date);
  Fieldbyname('Time').AsString:=TimetoStr(Time);
  begin
  if byte5='0' then
          begin
            if byte6='0'  then Money:='0';
            if byte6='1'  then Money:='10';
            if byte6='2'  then Money:='20';
            if byte6='3'  then Money:='30';
            if byte6='4'  then Money:='40';
            if byte6='5'  then Money:='50';
            if byte6='6'  then Money:='60';
            if byte6='7'  then Money:='70';
            if byte6='8'  then Money:='80';
            if byte6='9'  then Money:='90';
          end;
  if byte5='1' then
          begin
            if byte6='0'  then Money:='100';
            if byte6='1'  then Money:='110';
            if byte6='2'  then Money:='120';
            if byte6='3'  then Money:='130';
            if byte6='4'  then Money:='140';
            if byte6='5'  then Money:='150';
```

```
          if byte6='6'  then Money:='160';
          if byte6='7'  then Money:='170';
          if byte6='8'  then Money:='180';
          if byte6='9'  then Money:='190';
        end;
if byte5='2' then
        begin
          if byte6='0'  then Money:='200';
          if byte6='1'  then Money:='210';
          if byte6='2'  then Money:='220';
          if byte6='3'  then Money:='230';
          if byte6='4'  then Money:='240';
          if byte6='5'  then Money:='250';
          if byte6='6'  then Money:='260';
          if byte6='7'  then Money:='270';
          if byte6='8'  then Money:='280';
          if byte6='9'  then Money:='290';
        end;
if byte5='3' then
        begin
          if byte6='0'  then Money:='300';
          if byte6='1'  then Money:='310';
          if byte6='2'  then Money:='320';
          if byte6='3'  then Money:='330';
          if byte6='4'  then Money:='340';
          if byte6='5'  then Money:='350';
          if byte6='6'  then Money:='360';
          if byte6='7'  then Money:='370';
          if byte6='8'  then Money:='380';
          if byte6='9'  then Money:='390';
        end;
if byte5='4' then
        begin
          if byte6='0'  then Money:='400';
          if byte6='1'  then Money:='410';
          if byte6='2'  then Money:='420';
          if byte6='3'  then Money:='430';
          if byte6='4'  then Money:='440';
          if byte6='5'  then Money:='450';
          if byte6='6'  then Money:='460';
          if byte6='7'  then Money:='470';
          if byte6='8'  then Money:='480';
          if byte6='9'  then Money:='490';
        end;
if byte5='5' then
        begin
          if byte6='0'  then Money:='500';
          if byte6='1'  then Money:='510';
          if byte6='2'  then Money:='520';
          if byte6='3'  then Money:='530';
          if byte6='4'  then Money:='540';
          if byte6='5'  then Money:='550';
          if byte6='6'  then Money:='560';
          if byte6='7'  then Money:='570';
          if byte6='8'  then Money:='580';
          if byte6='9'  then Money:='590';
        end;
```

```
if byte5='6' then
        begin
          if byte6='0'  then Money:='600';
          if byte6='1'  then Money:='610';
          if byte6='2'  then Money:='620';
          if byte6='3'  then Money:='630';
          if byte6='4'  then Money:='640';
          if byte6='5'  then Money:='650';
          if byte6='6'  then Money:='660';
          if byte6='7'  then Money:='670';
          if byte6='8'  then Money:='680';
          if byte6='9'  then Money:='690';
        end;
if byte5='7' then
        begin
          if byte6='0'  then Money:='700';
          if byte6='1'  then Money:='710';
          if byte6='2'  then Money:='720';
          if byte6='3'  then Money:='730';
          if byte6='4'  then Money:='740';
          if byte6='5'  then Money:='750';
          if byte6='6'  then Money:='760';
          if byte6='7'  then Money:='770';
          if byte6='8'  then Money:='780';
          if byte6='9'  then Money:='790';
        end;
if byte5='8' then
       begin
          if byte6='0'  then Money:='800';
          if byte6='1'  then Money:='810';
          if byte6='2'  then Money:='820';
          if byte6='3'  then Money:='830';
          if byte6='4'  then Money:='840';
          if byte6='5'  then Money:='850';
          if byte6='6'  then Money:='860';
          if byte6='7'  then Money:='870';
          if byte6='8'  then Money:='880';
          if byte6='9'  then Money:='890';
        end;
if byte5='9' then
        begin
          if byte6='0'  then Money:='900';
          if byte6='1'  then Money:='910';
          if byte6='2'  then Money:='920';
          if byte6='3'  then Money:='930';
          if byte6='4'  then Money:='940';
          if byte6='5'  then Money:='950';
          if byte6='6'  then Money:='960';
          if byte6='7'  then Money:='970';
          if byte6='8'  then Money:='980';
          if byte6='9'  then Money:='990';
        end;
if byte5 = '16' then money :='1000';

if radiobutton1.Checked then
  aaa:='100';
  if radiobutton2.Checked then
```

```
    aaa:='200';
    if radiobutton3.Checked then
    aaa:='300';
    if radiobutton4.Checked then
    aaa:='400';
    if radiobutton5.Checked then
    aaa:='500';
    if radiobutton6.Checked then
    aaa:='600';
    if radiobutton7.Checked then
    aaa:='700';
    if radiobutton8.Checked then
    aaa:='800';
    if radiobutton9.Checked then
    aaa:='900';
    if radiobutton10.Checked then
    aaa:='1000';
    sum:=strtoint(money)+strtoint(aaa);
    begin
    Fieldbyname('Money').asstring:=inttostr(sum);
    s:=inttostr(e);
    showmessage('¤Ø³àµÔÁà§Ô¹ '+ aaa + 'àÃÕº°ÃéíÃáÅéÇ');
  end;
  post;
end;
end;
end;
procedure TForm4.BitBtn4Click(Sender: TObject);
begin
       if byte5='0' then
         begin
           if byte6='0'  then Edit5.Text :='0 Baht';
           if byte6='1'  then Edit5.Text :='10 Baht';
           if byte6='2'  then Edit5.Text :='20 Baht';
           if byte6='3'  then Edit5.Text :='30 Baht';
           if byte6='4'  then Edit5.Text :='40 Baht';
           if byte6='5'  then Edit5.Text :='50 Baht';
           if byte6='6'  then Edit5.Text :='60 Baht';
           if byte6='7'  then Edit5.Text :='70 Baht';
           if byte6='8'  then Edit5.Text :='80 Baht';
           if byte6='9'  then Edit5.Text :='90 Baht';
         end;
       if byte5='1' then
         begin
           if byte6='0'  then Edit5.Text :='100 Baht';
           if byte6='1'  then Edit5.Text :='110 Baht';
           if byte6='2'  then Edit5.Text :='120 Baht';
           if byte6='3'  then Edit5.Text :='130 Baht';
           if byte6='4'  then Edit5.Text :='140 Baht';
           if byte6='5'  then Edit5.Text :='150 Baht';
           if byte6='6'  then Edit5.Text :='160 Baht';
           if byte6='7'  then Edit5.Text :='170 Baht';
           if byte6='8'  then Edit5.Text :='180 Baht';
           if byte6='9'  then Edit5.Text :='190 Baht';
         end;
       if byte5='2' then
         begin
```

```
      if byte6='0'  then Edit5.Text :='200 Baht';
      if byte6='1'  then Edit5.Text :='210 Baht';
      if byte6='2'  then Edit5.Text :='220 Baht';
      if byte6='3'  then Edit5.Text :='230 Baht';
      if byte6='4'  then Edit5.Text :='240 Baht';
      if byte6='5'  then Edit5.Text :='250 Baht';
      if byte6='6'  then Edit5.Text :='260 Baht';
      if byte6='7'  then Edit5.Text :='270 Baht';
      if byte6='8'  then Edit5.Text :='280 Baht';
      if byte6='9'  then Edit5.Text :='290 Baht';
    end;
if byte5='3' then
    begin
      if byte6='0'  then Edit5.Text :='300 Baht';
      if byte6='1'  then Edit5.Text :='310 Baht';
      if byte6='2'  then Edit5.Text :='320 Baht';
      if byte6='3'  then Edit5.Text :='330 Baht';
      if byte6='4'  then Edit5.Text :='340 Baht';
      if byte6='5'  then Edit5.Text :='350 Baht';
      if byte6='6'  then Edit5.Text :='360 Baht';
      if byte6='7'  then Edit5.Text :='370 Baht';
      if byte6='8'  then Edit5.Text :='380 Baht';
      if byte6='9'  then Edit5.Text :='390 Baht';
    end;
 if byte5='4' then
    begin
      if byte6='0'  then Edit5.Text :='400 Baht';
      if byte6='1'  then Edit5.Text :='410 Baht';
      if byte6='2'  then Edit5.Text :='420 Baht';
      if byte6='3'  then Edit5.Text :='430 Baht';
      if byte6='4'  then Edit5.Text :='440 Baht';
      if byte6='5'  then Edit5.Text :='450 Baht';
      if byte6='6'  then Edit5.Text :='460 Baht';
      if byte6='7'  then Edit5.Text :='470 Baht';
      if byte6='8'  then Edit5.Text :='480 Baht';
      if byte6='9'  then Edit5.Text :='490 Baht';
    end;
 if byte5='5' then
    begin
      if byte6='0'  then Edit5.Text :='500 Baht';
      if byte6='1'  then Edit5.Text :='510 Baht';
      if byte6='2'  then Edit5.Text :='520 Baht';
      if byte6='3'  then Edit5.Text :='530 Baht';
      if byte6='4'  then Edit5.Text :='540 Baht';
      if byte6='5'  then Edit5.Text :='550 Baht';
      if byte6='6'  then Edit5.Text :='560 Baht';
      if byte6='7'  then Edit5.Text :='570 Baht';
      if byte6='8'  then Edit5.Text :='580 Baht';
      if byte6='9'  then Edit5.Text :='590 Baht';
    end;
 if byte5='6' then
    begin
      if byte6='0'  then Edit5.Text :='600 Baht';
      if byte6='1'  then Edit5.Text :='610 Baht';
      if byte6='2'  then Edit5.Text :='620 Baht';
      if byte6='3'  then Edit5.Text :='630 Baht';
      if byte6='4'  then Edit5.Text :='640 Baht';
```

```
            if byte6='5'  then Edit5.Text :='650 Baht';
            if byte6='6'  then Edit5.Text :='660 Baht';
            if byte6='7'  then Edit5.Text :='670 Baht';
            if byte6='8'  then Edit5.Text :='680 Baht';
            if byte6='9'  then Edit5.Text :='690 Baht';
          end;
        if byte5='7' then
          begin
            if byte6='0'  then Edit5.Text :='700 Baht';
            if byte6='1'  then Edit5.Text :='710 Baht';
            if byte6='2'  then Edit5.Text :='720 Baht';
            if byte6='3'  then Edit5.Text :='730 Baht';
            if byte6='4'  then Edit5.Text :='740 Baht';
            if byte6='5'  then Edit5.Text :='750 Baht';
            if byte6='6'  then Edit5.Text :='760 Baht';
            if byte6='7'  then Edit5.Text :='770 Baht';
            if byte6='8'  then Edit5.Text :='780 Baht';
            if byte6='9'  then Edit5.Text :='790 Baht';
          end;
        if byte5='8' then
         begin
            if byte6='0'  then Edit5.Text :='800 Baht';
            if byte6='1'  then Edit5.Text :='810 Baht';
            if byte6='2'  then Edit5.Text :='820 Baht';
            if byte6='3'  then Edit5.Text :='830 Baht';
            if byte6='4'  then Edit5.Text :='840 Baht';
            if byte6='5'  then Edit5.Text :='850 Baht';
            if byte6='6'  then Edit5.Text :='860 Baht';
            if byte6='7'  then Edit5.Text :='870 Baht';
            if byte6='8'  then Edit5.Text :='880 Baht';
            if byte6='9'  then Edit5.Text :='890 Baht';
          end;
        if byte5='9' then
          begin
            if byte6='0'  then Edit5.Text :='900 Baht';
            if byte6='1'  then Edit5.Text :='910 Baht';
            if byte6='2'  then Edit5.Text :='920 Baht';
            if byte6='3'  then Edit5.Text :='930 Baht';
            if byte6='4'  then Edit5.Text :='940 Baht';
            if byte6='5'  then Edit5.Text :='950 Baht';
            if byte6='6'  then Edit5.Text :='960 Baht';
            if byte6='7'  then Edit5.Text :='970 Baht';
            if byte6='8'  then Edit5.Text :='980 Baht';
            if byte6='9'  then Edit5.Text :='990 Baht';
          end;
        if byte5 = '16' then Edit5.Text :='1000 Baht';
end;
procedure TForm4.addmoney;
var st,find:string;
    aa:integer;
begin
    begin
    with query do
    find:=fieldbyname('Money').asstring;
    aa:=strtoint(find);
      if aa=0 then
st:=chr($FB)+chr($04)+chr($01)+chr($90)+chr($00)+chr($00)+chr($00)
```

```
      +chr($6E)+chr($FC);
      if aa=10 then
st:=chr($FB)+chr($04)+chr($01)+chr($90)+chr($00)+chr($01)+chr($00)
      +chr($6F)+chr($FC);
      if aa=20 then
st:=chr($FB)+chr($04)+chr($01)+chr($90)+chr($00)+chr($02)+chr($00)
      +chr($6C)+chr($FC);
      if aa=30 then
st:=chr($FB)+chr($04)+chr($01)+chr($90)+chr($00)+chr($03)+chr($00)
      +chr($6D)+chr($FC);
      if aa=40 then
st:=chr($FB)+chr($04)+chr($01)+chr($90)+chr($00)+chr($04)+chr($00)
      +chr($6A)+chr($FC);
      if aa=50 then
st:=chr($FB)+chr($04)+chr($01)+chr($90)+chr($00)+chr($05)+chr($00)
      +chr($6B)+chr($FC);
      if aa=60 then
st:=chr($FB)+chr($04)+chr($01)+chr($90)+chr($00)+chr($06)+chr($00)
      +chr($68)+chr($FC);
      if aa=70 then
st:=chr($FB)+chr($04)+chr($01)+chr($90)+chr($00)+chr($07)+chr($00)
      +chr($69)+chr($FC);
      if aa=80 then
st:=chr($FB)+chr($04)+chr($01)+chr($90)+chr($00)+chr($08)+chr($00)
      +chr($66)+chr($FC);
      if aa=90 then
st:=chr($FB)+chr($04)+chr($01)+chr($90)+chr($00)+chr($09)+chr($00)
      +chr($67)+chr($FC);
      if aa=100 then
st:=chr($FB)+chr($04)+chr($01)+chr($90)+chr($01)+chr($00)+chr($00)
      +chr($6F)+chr($FC);
      if aa=110 then
st:=chr($FB)+chr($04)+chr($01)+chr($90)+chr($01)+chr($01)+chr($00)
      +chr($6E)+chr($FC);
      if aa=120 then
st:=chr($FB)+chr($04)+chr($01)+chr($90)+chr($01)+chr($02)+chr($00)
      +chr($6D)+chr($FC);
      if aa=130 then
st:=chr($FB)+chr($04)+chr($01)+chr($90)+chr($01)+chr($03)+chr($00)
      +chr($6C)+chr($FC);
      if aa=140 then
st:=chr($FB)+chr($04)+chr($01)+chr($90)+chr($01)+chr($04)+chr($00)
      +chr($6B)+chr($FC);
      if aa=150 then
st:=chr($FB)+chr($04)+chr($01)+chr($90)+chr($01)+chr($05)+chr($00)
      +chr($6A)+chr($FC);
      if aa=160 then
st:=chr($FB)+chr($04)+chr($01)+chr($90)+chr($01)+chr($06)+chr($00)
      +chr($69)+chr($FC);
      if aa=170 then
st:=chr($FB)+chr($04)+chr($01)+chr($90)+chr($01)+chr($07)+chr($00)
      +chr($68)+chr($FC);
      if aa=180 then
st:=chr($FB)+chr($04)+chr($01)+chr($90)+chr($01)+chr($08)+chr($00)
      +chr($67)+chr($FC);
      if aa=190 then
st:=chr($FB)+chr($04)+chr($01)+chr($90)+chr($01)+chr($09)+chr($00)
```

```
      +chr($66)+chr($FC);
      if aa=200 then
st:=chr($FB)+chr($04)+chr($01)+chr($90)+chr($02)+chr($00)+chr($00)
      +chr($6C)+chr($FC);
      if aa=210 then
st:=chr($FB)+chr($04)+chr($01)+chr($90)+chr($02)+chr($01)+chr($00)
      +chr($6D)+chr($FC);
      if aa=220 then
st:=chr($FB)+chr($04)+chr($01)+chr($90)+chr($02)+chr($02)+chr($00)
      +chr($6E)+chr($FC);
      if aa=230 then
st:=chr($FB)+chr($04)+chr($01)+chr($90)+chr($02)+chr($03)+chr($00)
      +chr($6F)+chr($FC);
      if aa=240 then
st:=chr($FB)+chr($04)+chr($01)+chr($90)+chr($02)+chr($04)+chr($00)
      +chr($68)+chr($FC);
      if aa=250 then
st:=chr($FB)+chr($04)+chr($01)+chr($90)+chr($02)+chr($05)+chr($00)
      +chr($69)+chr($FC);
      if aa=260 then
st:=chr($FB)+chr($04)+chr($01)+chr($90)+chr($02)+chr($06)+chr($00)
      +chr($6A)+chr($FC);
      if aa=270 then
st:=chr($FB)+chr($04)+chr($01)+chr($90)+chr($02)+chr($07)+chr($00)
      +chr($6B)+chr($FC);
      if aa=280 then
st:=chr($FB)+chr($04)+chr($01)+chr($90)+chr($02)+chr($08)+chr($00)
      +chr($64)+chr($FC);
      if aa=290 then
st:=chr($FB)+chr($04)+chr($01)+chr($90)+chr($02)+chr($09)+chr($00)
      +chr($65)+chr($FC);
      if aa=300 then
st:=chr($FB)+chr($04)+chr($01)+chr($90)+chr($03)+chr($00)+chr($00)
      +chr($6D)+chr($FC);
      if aa=310 then
st:=chr($FB)+chr($04)+chr($01)+chr($90)+chr($03)+chr($01)+chr($00)
      +chr($6C)+chr($FC);
      if aa=320 then
st:=chr($FB)+chr($04)+chr($01)+chr($90)+chr($03)+chr($02)+chr($00)
      +chr($6F)+chr($FC);
      if aa=330 then
st:=chr($FB)+chr($04)+chr($01)+chr($90)+chr($03)+chr($03)+chr($00)
      +chr($6E)+chr($FC);
      if aa=340 then
st:=chr($FB)+chr($04)+chr($01)+chr($90)+chr($03)+chr($04)+chr($00)
      +chr($69)+chr($FC);
      if aa=350 then
st:=chr($FB)+chr($04)+chr($01)+chr($90)+chr($03)+chr($05)+chr($00)
      +chr($68)+chr($FC);
      if aa=360 then
st:=chr($FB)+chr($04)+chr($01)+chr($90)+chr($03)+chr($06)+chr($00)
      +chr($6B)+chr($FC);
      if aa=370 then
st:=chr($FB)+chr($04)+chr($01)+chr($90)+chr($03)+chr($07)+chr($00)
      +chr($6A)+chr($FC);
      if aa=380 then
st:=chr($FB)+chr($04)+chr($01)+chr($90)+chr($03)+chr($08)+chr($00)
```

```
      +chr($65)+chr($FC);
      if aa=390 then
st:=chr($FB)+chr($04)+chr($01)+chr($90)+chr($03)+chr($09)+chr($00)
      +chr($64)+chr($FC);
      if aa=400 then
st:=chr($FB)+chr($04)+chr($01)+chr($90)+chr($04)+chr($00)+chr($00)
      +chr($6A)+chr($FC);
      if aa=410 then
st:=chr($FB)+chr($04)+chr($01)+chr($90)+chr($04)+chr($01)+chr($00)
      +chr($6B)+chr($FC);
      if aa=420 then
st:=chr($FB)+chr($04)+chr($01)+chr($90)+chr($04)+chr($02)+chr($00)
      +chr($68)+chr($FC);
      if aa=430 then
st:=chr($FB)+chr($04)+chr($01)+chr($90)+chr($04)+chr($03)+chr($00)
      +chr($69)+chr($FC);
      if aa=440 then
st:=chr($FB)+chr($04)+chr($01)+chr($90)+chr($04)+chr($04)+chr($00)
      +chr($6E)+chr($FC);
      if aa=450 then
st:=chr($FB)+chr($04)+chr($01)+chr($90)+chr($04)+chr($05)+chr($00)
      +chr($6F)+chr($FC);
      if aa=460 then
st:=chr($FB)+chr($04)+chr($01)+chr($90)+chr($04)+chr($06)+chr($00)
      +chr($6C)+chr($FC);
      if aa=470 then
st:=chr($FB)+chr($04)+chr($01)+chr($90)+chr($04)+chr($07)+chr($00)
      +chr($6D)+chr($FC);
      if aa=480 then
st:=chr($FB)+chr($04)+chr($01)+chr($90)+chr($04)+chr($08)+chr($00)
      +chr($62)+chr($FC);
      if aa=490 then
st:=chr($FB)+chr($04)+chr($01)+chr($90)+chr($04)+chr($09)+chr($00)
      +chr($63)+chr($FC);
      if aa=500 then
st:=chr($FB)+chr($04)+chr($01)+chr($90)+chr($05)+chr($00)+chr($00)
      +chr($6B)+chr($FC);
      if aa=510 then
st:=chr($FB)+chr($04)+chr($01)+chr($90)+chr($05)+chr($01)+chr($00)
      +chr($6A)+chr($FC);
      if aa=520 then
st:=chr($FB)+chr($04)+chr($01)+chr($90)+chr($05)+chr($02)+chr($00)
      +chr($69)+chr($FC);
      if aa=530 then
st:=chr($FB)+chr($04)+chr($01)+chr($90)+chr($05)+chr($03)+chr($00)
      +chr($68)+chr($FC);
      if aa=540 then
st:=chr($FB)+chr($04)+chr($01)+chr($90)+chr($05)+chr($04)+chr($00)
      +chr($6F)+chr($FC);
      if aa=550 then
st:=chr($FB)+chr($04)+chr($01)+chr($90)+chr($05)+chr($05)+chr($00)
      +chr($6E)+chr($FC);
      if aa=560 then
st:=chr($FB)+chr($04)+chr($01)+chr($90)+chr($05)+chr($06)+chr($00)
      +chr($6D)+chr($FC);
      if aa=570 then
st:=chr($FB)+chr($04)+chr($01)+chr($90)+chr($05)+chr($07)+chr($00)
```

```
      +chr($6C)+chr($FC);
      if aa=580 then
st:=chr($FB)+chr($04)+chr($01)+chr($90)+chr($05)+chr($08)+chr($00)
      +chr($63)+chr($FC);
      if aa=590 then
st:=chr($FB)+chr($04)+chr($01)+chr($90)+chr($05)+chr($09)+chr($00)
      +chr($62)+chr($FC);
      if aa=600 then
st:=chr($FB)+chr($04)+chr($01)+chr($90)+chr($06)+chr($00)+chr($00)
      +chr($68)+chr($FC);
      if aa=610 then
st:=chr($FB)+chr($04)+chr($01)+chr($90)+chr($06)+chr($01)+chr($00)
      +chr($69)+chr($FC);
      if aa=620 then
st:=chr($FB)+chr($04)+chr($01)+chr($90)+chr($06)+chr($02)+chr($00)
      +chr($6A)+chr($FC);
      if aa=630 then
st:=chr($FB)+chr($04)+chr($01)+chr($90)+chr($06)+chr($03)+chr($00)
      +chr($6B)+chr($FC);
      if aa=640 then
st:=chr($FB)+chr($04)+chr($01)+chr($90)+chr($06)+chr($04)+chr($00)
      +chr($6C)+chr($FC);
      if aa=650 then
st:=chr($FB)+chr($04)+chr($01)+chr($90)+chr($06)+chr($05)+chr($00)
      +chr($6D)+chr($FC);
      if aa=660 then
st:=chr($FB)+chr($04)+chr($01)+chr($90)+chr($06)+chr($06)+chr($00)
      +chr($6E)+chr($FC);
      if aa=670 then
st:=chr($FB)+chr($04)+chr($01)+chr($90)+chr($06)+chr($07)+chr($00)
      +chr($6F)+chr($FC);
      if aa=680 then
st:=chr($FB)+chr($04)+chr($01)+chr($90)+chr($06)+chr($08)+chr($00)
      +chr($60)+chr($FC);
      if aa=690 then
st:=chr($FB)+chr($04)+chr($01)+chr($90)+chr($06)+chr($09)+chr($00)
      +chr($61)+chr($FC);
      if aa=700 then
st:=chr($FB)+chr($04)+chr($01)+chr($90)+chr($07)+chr($00)+chr($00)
      +chr($69)+chr($FC);
      if aa=710 then
st:=chr($FB)+chr($04)+chr($01)+chr($90)+chr($07)+chr($01)+chr($00)
      +chr($68)+chr($FC);
      if aa=720 then
st:=chr($FB)+chr($04)+chr($01)+chr($90)+chr($07)+chr($02)+chr($00)
      +chr($6B)+chr($FC);
      if aa=730 then
st:=chr($FB)+chr($04)+chr($01)+chr($90)+chr($07)+chr($03)+chr($00)
      +chr($6A)+chr($FC);
      if aa=740 then
st:=chr($FB)+chr($04)+chr($01)+chr($90)+chr($07)+chr($04)+chr($00)
      +chr($6D)+chr($FC);
      if aa=750 then
st:=chr($FB)+chr($04)+chr($01)+chr($90)+chr($07)+chr($05)+chr($00)
      +chr($6C)+chr($FC);
      if aa=760 then
st:=chr($FB)+chr($04)+chr($01)+chr($90)+chr($07)+chr($06)+chr($00)
```

```
      +chr($6F)+chr($FC);
      if aa=770 then
st:=chr($FB)+chr($04)+chr($01)+chr($90)+chr($07)+chr($07)+chr($00)
      +chr($6E)+chr($FC);
      if aa=780 then
st:=chr($FB)+chr($04)+chr($01)+chr($90)+chr($07)+chr($08)+chr($00)
      +chr($61)+chr($FC);
      if aa=790 then
st:=chr($FB)+chr($04)+chr($01)+chr($90)+chr($07)+chr($09)+chr($00)
      +chr($60)+chr($FC);
      if aa=800 then
st:=chr($FB)+chr($04)+chr($01)+chr($90)+chr($08)+chr($00)+chr($00)
      +chr($66)+chr($FC);
      if aa=810 then
st:=chr($FB)+chr($04)+chr($01)+chr($90)+chr($08)+chr($01)+chr($00)
      +chr($67)+chr($FC);
      if aa=820 then
st:=chr($FB)+chr($04)+chr($01)+chr($90)+chr($08)+chr($02)+chr($00)
      +chr($64)+chr($FC);
      if aa=830 then
st:=chr($FB)+chr($04)+chr($01)+chr($90)+chr($08)+chr($03)+chr($00)
      +chr($65)+chr($FC);
      if aa=840 then
st:=chr($FB)+chr($04)+chr($01)+chr($90)+chr($08)+chr($04)+chr($00)
      +chr($62)+chr($FC);
      if aa=850 then
st:=chr($FB)+chr($04)+chr($01)+chr($90)+chr($08)+chr($05)+chr($00)
      +chr($63)+chr($FC);
      if aa=860 then
st:=chr($FB)+chr($04)+chr($01)+chr($90)+chr($08)+chr($06)+chr($00)
      +chr($60)+chr($FC);
      if aa=870 then
st:=chr($FB)+chr($04)+chr($01)+chr($90)+chr($08)+chr($07)+chr($00)
      +chr($61)+chr($FC);
      if aa=880 then
st:=chr($FB)+chr($04)+chr($01)+chr($90)+chr($08)+chr($08)+chr($00)
      +chr($6E)+chr($FC);
      if aa=890 then
st:=chr($FB)+chr($04)+chr($01)+chr($90)+chr($08)+chr($09)+chr($00)
      +chr($6F)+chr($FC);
      if aa=900 then
st:=chr($FB)+chr($04)+chr($01)+chr($90)+chr($09)+chr($00)+chr($00)
      +chr($67)+chr($FC);
      if aa=910 then
st:=chr($FB)+chr($04)+chr($01)+chr($90)+chr($09)+chr($01)+chr($00)
      +chr($66)+chr($FC);
      if aa=920 then
st:=chr($FB)+chr($04)+chr($01)+chr($90)+chr($09)+chr($02)+chr($00)
      +chr($65)+chr($FC);
      if aa=930 then
st:=chr($FB)+chr($04)+chr($01)+chr($90)+chr($09)+chr($03)+chr($00)
      +chr($64)+chr($FC);
      if aa=940 then
st:=chr($FB)+chr($04)+chr($01)+chr($90)+chr($09)+chr($04)+chr($00)
      +chr($63)+chr($FC);
      if aa=950 then
st:=chr($FB)+chr($04)+chr($01)+chr($90)+chr($09)+chr($05)+chr($00)
```

```
      +chr($62)+chr($FC);
      if aa=960 then
st:=chr($FB)+chr($04)+chr($01)+chr($90)+chr($09)+chr($06)+chr($00)
      +chr($61)+chr($FC);
      if aa=970 then
st:=chr($FB)+chr($04)+chr($01)+chr($90)+chr($09)+chr($07)+chr($00)
      +chr($60)+chr($FC);
      if aa=980 then
st:=chr($FB)+chr($04)+chr($01)+chr($90)+chr($09)+chr($08)+chr($00)
      +chr($6F)+chr($FC);
      if aa=990 then
st:=chr($FB)+chr($04)+chr($01)+chr($90)+chr($09)+chr($09)+chr($00)
      +chr($6E)+chr($FC);
      if aa=1000 then
st:=chr($FB)+chr($04)+chr($01)+chr($90)+chr($10)+chr($00)+chr($00)
      +chr($7E)+chr($FC);
    Vacomm1.writetext(st);
    end;
    end;
procedure TForm4.TabSheet2Show(Sender: TObject);
begin
  with table1 do
  if findkey([Edit3.text]) then
  begin
  Edit6.Text:=Fieldbyname('ID Car').AsString;
  Edit7.Text:=Fieldbyname('Name').AsString;
  Edit8.Text:=Fieldbyname('Surname').AsString;
  Edit9.Text:=Fieldbyname('Address').AsString;
  Edit10.Text:=Fieldbyname('Telephone').AsString;
  end;
end;
procedure TForm4.Timer5Timer(Sender: TObject);
var st:string;
begin
    st:=chr($FB)+chr($02)+chr($FF)+chr($06)+chr($FC);
    Vacomm1.WriteText(st);
    Timer5.Enabled:=false;
end;
procedure TForm4.Timer6Timer(Sender: TObject);
var st:string;
begin
    st:=chr($FB)+chr($02)+chr($FF)+chr($06)+chr($FC);
    Vacomm1.WriteText(st);
    Timer5.Enabled:=false;
end;
procedure TForm4.Timer7Timer(Sender: TObject);
begin
    Edit1.Text:='';
    Timer7.Enabled:=false;
end;
procedure TForm4.ExitBtnClick(Sender: TObject);
begin
    Vacomm1.Close;
    Close;
    Edit3.Clear;
    Edit5.Clear;
    Edit6.Clear;
```

```
    Edit7.Clear;
    Edit8.Clear;
    Edit9.Clear;
    Edit10.Clear;
end;
procedure TForm4.FormActivate(Sender: TObject);
var resp,resp1:word;
begin
  resp:=MessageDlg('Open Port?',mtConfirmation,[mbYes,mbNo],0);
  if resp = mrYes then
  Vacomm1.Open;
  if resp = mrNo then
  Showmessage('Cant Connect To The Port');
  Close;
end;
procedure TForm4.btOpenClick(Sender: TObject);
begin
    Query.Open;
end;
procedure TForm4.btCloseClick(Sender: TObject);
begin
    Query.Close;
end;
procedure TForm4.btCreateClick(Sender: TObject);
begin
    sqCreateTable.Execute;
end;
procedure TForm4.btDropClick(Sender: TObject);
begin
    sqDropTable.Execute;
end;
procedure TForm4.BitBtn5Click(Sender: TObject);
var st:string;
begin
st:=chr($FB)+chr($04)+chr($01)+chr($00)+chr($00)+chr($00)+chr($00)+
        chr($FE)+chr($FC);
      Vacomm1.WriteText(st);
      Timer5.Enabled:=true;
      Timer6.Enabled:=true;
end;
end.
```

Features

- Contactless Read/Write Data Transmission
- Radio Frequency f_{RF} from 100 kHz to 150 kHz
- e5550 Binary Compatible or T5557 Extended Mode
- Small Size, Configurable for ISO/IEC 11784/785 Compatibility
- 75 pF On-chip Resonant Capacitor (Mask Option)
- 7 × 32-bit EEPROM Data Memory Including 32-bit Password
- Separate 64-bit memory for Traceability Data
- 32-bit Configuration Register in EEPROM to Setup:
  - Data Rate
    - RF/2 to RF/128, Binary Selectable or
    - Fixed e5550 Data Rates
  - Modulation/Coding
    - FSK, PSK, Manchester, Biphase, NRZ
  - Other Options
    - Password Mode
    - Max Block Feature
    - Answer-On-Request (AOR) Mode
    - Inverse Data Output
    - Direct Access Mode
    - Sequence Terminator(s)
    - Write Protection (Through Lock-bit per Block)
    - Fast Write Method (5 kbps versus 2 kbps)
    - OTP Functionality
    - POR Delay up to 67 ms

Multifunctional 330-bit Read/Write RF-Identification IC

T5557

Description

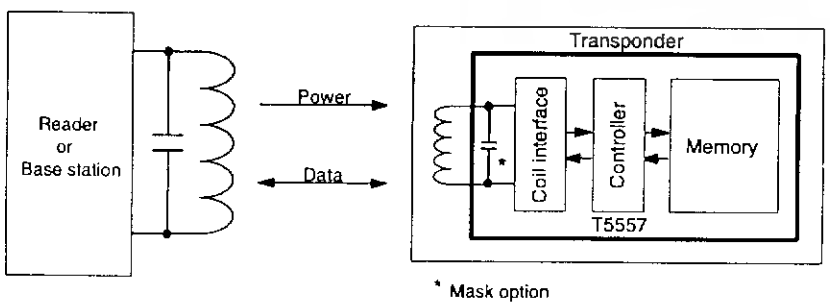
The T5557 is a contactless R/W IDentification IC (IDIC®) for applications in the 125 kHz frequency range. A single coil, connected to the chip, serves as the IC's power supply and bi-directional communication interface. The antenna and chip together form a transponder or tag.

The on-chip 330-bit EEPROM (10 blocks, 33 bits each) can be read and written block-wise from a reader. Block 0 is reserved for setting the operation modes of the T5557 tag. Block 7 may contain a password to prevent unauthorized writing.

Data is transmitted from the IDIC using load modulation. This is achieved by damping the RF field with a resistive load between the two terminals Coil 1 and Coil 2. The IC receives and decodes 100% amplitude modulated (OOK) pulse interval encoded bit streams from the base station or reader.

System Block Diagram

Figure 1. RFID System Using T5557 Tag

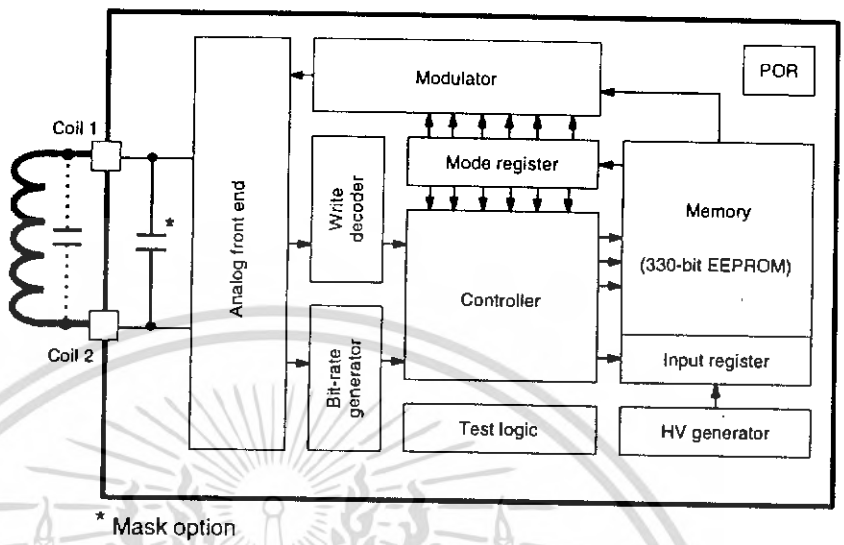


* Mask option

Rev. 4517F–RFID–11/03

T5557 – Building Blocks

Figure 2. Block Diagram



Analog Front End (AFE)

The AFE includes all circuits which are directly connected to the coil. It generates the IC's power supply and handles the bi-directional data communication with the reader. It consists of the following blocks:

- Rectifier to generate a DC supply voltage from the AC coil voltage
- Clock extractor
- Switchable load between Coil 1/Coil 2 for data transmission from tag to the reader
- Field gap detector for data transmission from the base station to the tag
- ESD protection circuitry

Data-rate Generator

The data rate is binary programmable to operate at any data rate between RF/2 and RF/128 or equal to any of the fixed e5550/e5551 and T5554 bitrates (RF/8, RF/16, RF/32, RF/40, RF/50, RF/64, RF/100 and RF/128).

Write Decoder

This function decodes the write gaps and verifies the validity of the data stream according to the Atmel e555x write method (pulse interval encoding).

HV Generator

This on-chip charge pump circuit generates the high voltage required for programming of the EEPROM.

DC Supply

Power is externally supplied to the IDIC via the two coil connections. The IC rectifies and regulates this RF source and uses it to generate its supply voltage.

Power-On Reset (POR)

This circuit delays the IDIC functionality until an acceptable voltage threshold has been reached.

Clock Extraction

The clock extraction circuit uses the external RF signal as its internal clock source.

Controller

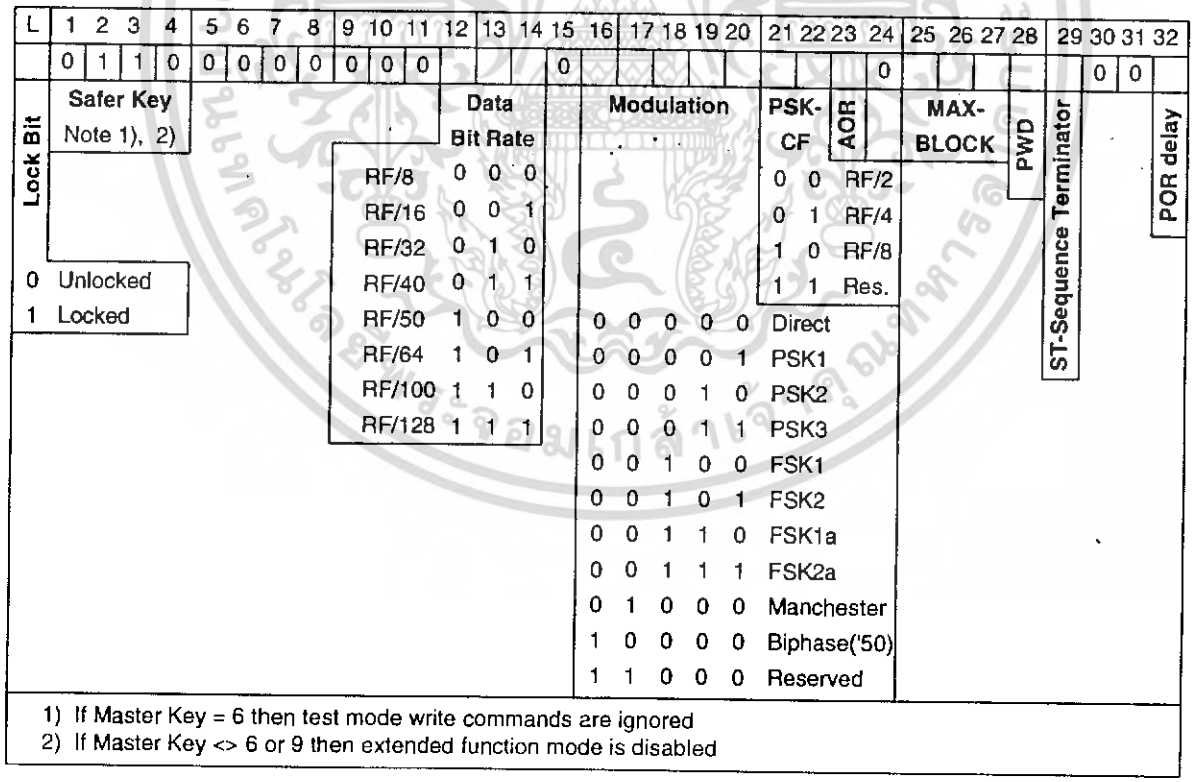
The control-logic module executes the following functions:

- Load-mode register with configuration data from EEPROM block 0 after power-on and also during reading
- Control memory access (read, write)
- Handle write data transmission and write error modes
- The first two bits of the reader to tag data stream are the opcode, e.g., write, direct access or reset
- In password mode, the 32 bits received after the opcode are compared with the password stored in memory block 7

Mode Register

The mode register stores the configuration data from the EEPROM block 0. It is continually refreshed at the start of every block read and (re-)loaded after any POR event or reset command. On delivery the mode register is preprogrammed with the value '0014 8000'h which corresponds to continuous read of block 0, Manchester coded, RF/64.

Figure 3. Block 0 Configuration Mapping – e5550 Compatibility Mode



Modulator

The modulator consists of data encoders for the following basic types of modulation:

Table 1. Types of e5550-compatible Modulation Modes

Mode	Direct Data Output		
FSK1a[1]	FSK/8-/5	'0' = rf/8;	'1' = rf/5
FSK2a[1]	FSK/8-/10	'0' = rf/8;	'1' = rf/10
FSK1[1]	FSK/5-/8	'0' = rf/5;	'1' = rf/8
FSK2[1]	FSK/10-/8	'0' = rf/10;	1' = rf/8
PSK1[2]	Phase change when input changes		
PSK2[2]	Phase change on bit clock if input high		
PSK3[2]	Phase change on rising edge of input		
Manchester	'0' = falling edge, '1' = rising edge		
Biphase	'1' creates an additional mid-bit change		
NRZ	'1' = damping on, '0' = damping off		

Notes: 1. A common multiple of bitrate and FSK frequencies is recommended.
2. In PSK mode the selected data rate has to be an integer multiple of the PSK sub-carrier frequency.

Memory

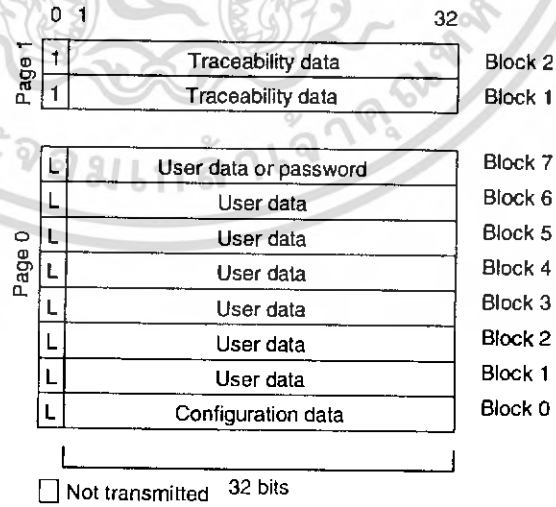
The memory is a 330-bit EEPROM, which is arranged in 10 blocks of 33 bits each. All 33 bits of a block, including the lock bit, are programmed simultaneously.

Block 0 of page 0 contains the mode/configuration data, which is not transmitted during regular-read operations. Block 7 of page 0 may be used as a write protection password.

Bit 0 of every block is the lock bit for that block. Once locked, the block (including the lock bit itself) is not re-programmable through the RF field again.

Blocks 1 and 2 of page 1 contain traceability data and are transmitted with the modulation parameters defined in the configuration register after the opcode '11' is issued by the reader (see Figure 11 on page 9). These tracebility data blocks are programmed and locked by Atmel.

Figure 4. Memory Map

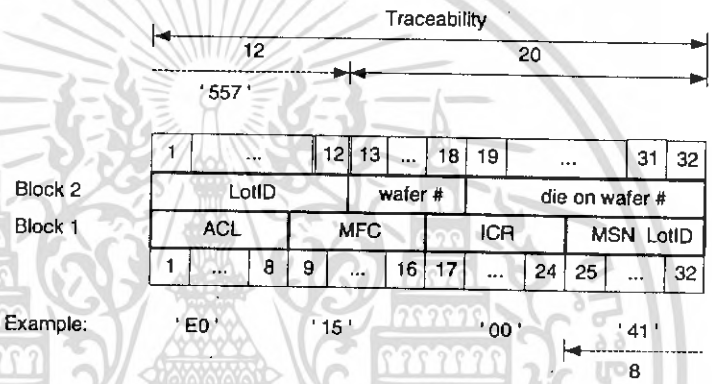


Traceability Data Structure

Blocks 1 and 2 of page 1 contain the traceability data and are programmed and locked by Atmel during production testing. The most significant byte of block 1 is fixed to 'E0'hex, the allocation class (ACL) as defined in ISO/IEC 15963-1. The second byte is therefore defined as the manufacturer's ID of Atmel (= '15'hex). The following 8 bits are used as IC reference byte (ICR - Bits 47 to 40). The 3 most significant bits define the IC and/or foundry version of the T5557. The lower 5 bits are by default reset (=00) as the Atmel standard value. Other values may be assigned on request to high volume customers as tag issuer identification.

The lower 40 bits of the data encode the traceability information of Atmel and conform to a unique numbering system. These 40 data bits are divided in two sub-groups, a 5-digit lot ID number, the binary wafer number (5 bit) concatenated with the sequential die number per wafer.

Figure 5. T5557 Traceability Data Structure



ACL Allocation class as defined in ISO/IEC 15963-1 = E0h
MFC Manufacturer code of Atmel Corporation as defined in ISO/IEC 7816-6 = 15h
ICR IC reference of silicon and/or tag manufacturer.
Top 3 bits define IC revision
Lower 5 bits may contain a customer ID code on request
MSN Manufacturer serial number consists of:
LotID 5-digit lot number, e.g., '38765'
DPW 20 bits encoded as sequential die per wafer number (with top 5 bits = wafer#)

Operating the T5557

Initialization and POR Delay

The Power-On-Reset (POR) circuit remains active until an adequate voltage threshold has been reached. This in turn triggers the default start-up delay sequence. During this configuration period of about 192 field clocks, the T5557 is initialized with the configuration data stored in EEPROM block 0. During initialization of the configuration block 0, all T55570x variants the load damping is active permanently (see Figure 10 on page 9). The T55571x types (without damping option) achieve a longer read range based on the lower activation field strength.

If the POR-delay bit is reset, no additional delay is observed after the configuration period. Tag modulation in regular-read mode will be observed about 3 ms after entering the RF field. If the POR delay bit is set, the T5557 remains in a permanent damping state until 8190 internal field clocks have elapsed.

$$T_{INIT} = (192 + 8190 \times \text{POR delay}) \times T_C \approx 67 \text{ ms}; \quad T_C = 8 \text{ µs at } 125 \text{ kHz}$$

Any field gap occurring during this initialization phase will restart the complete sequence. After this initialization time the T5557 enters regular-read mode and modulation starts automatically using the parameters defined in the configuration register.

Tag to Reader Communication

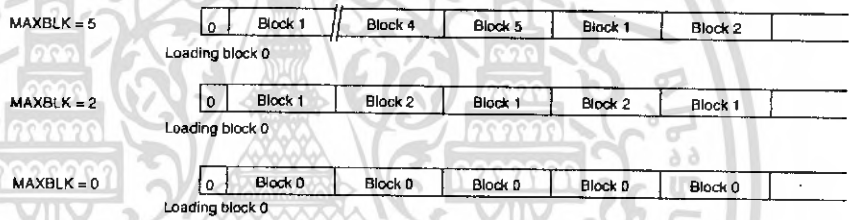
During normal operation, the data stored within the EEPROM is cycled and the Coil 1, Coil 2 terminals are load modulated. This resistive load modulation can be detected at the reader module.

Regular-read Mode

In regular-read mode data from the memory is transmitted serially, starting with block 1, bit 1, up to the last block (e.g., 7), bit 32. The last block which will be read is defined by the mode parameter field MAXBLK in EEPROM block 0. When the data block addressed by MAXBLK has been read, data transmission restarts with block 1, bit 1.

The user may limit the cyclic datastream in regular-read mode by setting the MAXBLK between 0 and 7 (representing each of the 8 data blocks). If set to 7, blocks 1 through 7 can be read. If set to 1, only block 1 is transmitted continously. If set to 0, the contents of the configuration block (normally not transmitted) can be read. In the case of MAXBLK = 0 or 1, regular-read mode can not be distinguished from block-read mode.

Figure 6. Examples for Different MAXBLK Settings



Every time the T5557 enters regular- or block-read mode, the first bit transmitted is a logical '0'. The data stream starts with block 1, bit 1, continues through MAXBLK, bit 32, and cycles continuously if in regular-read mode.

Note: This behavior is different from the original e555x and helps to decode PSK-modulated data.

Block-read Mode

With the direct access command, the addressed block is repetitively read only. This mode is called block-read mode. Direct access is entered by transmitting the page access opcode ('10' or '11'), a single '0' bit and the requested 3-bit block address when the tag is in normal mode.

In password mode (PWD bit set), the direct access to a single block needs the valid 32-bit password to be transmitted after the page access opcode whereas a '0' bit and the 3-bit block address follow afterwards. In case the transmitted password does not match with the contents of block 7, the T5557 tag returns to the regular-read mode.

Note: A direct access to block 0 of page 1 will read the configuration data of block 0, page 0.
A direct access to bock 3 .. 7 of page 1 reads all data bits as zero.

e5550 Sequence Terminator

The sequence terminator ST is a special damping pattern which is inserted before the first block and may be used to synchronize the reader. This e5550-compatible sequence terminator consists of 4 bit periods with underlaying data values of '1'. During the second and the fourth bit period, modulation is switched off (Manchester encoding – switched on). Biphase modulated data blocks need fixed leading and trailing bits in combination with the sequence terminator to be identified reliable.

The sequence terminator may be individually enabled by setting of mode bit 29 (ST = '1') in the e5550-compatibility mode (X-mode = '0').

In the regular-read mode, the sequence terminator is inserted at the start of each MAXBLK-limited read data stream.

In block-read mode – after any block-write or direct access command – or if MAXBLK was set to 0 or 1, the sequence terminator is inserted before the transmission of the selected block.

Especially this behavior is different to former e5550 – compatible ICs (T5551, T5554).

Figure 7. Read Data Stream with Sequence Terminator

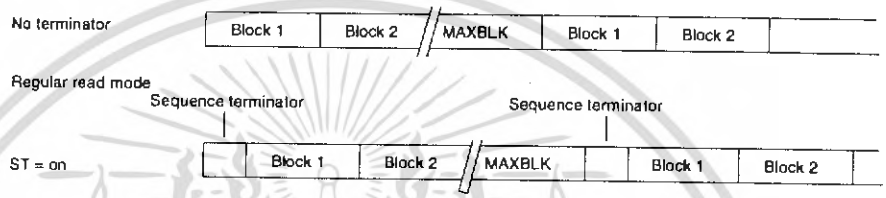
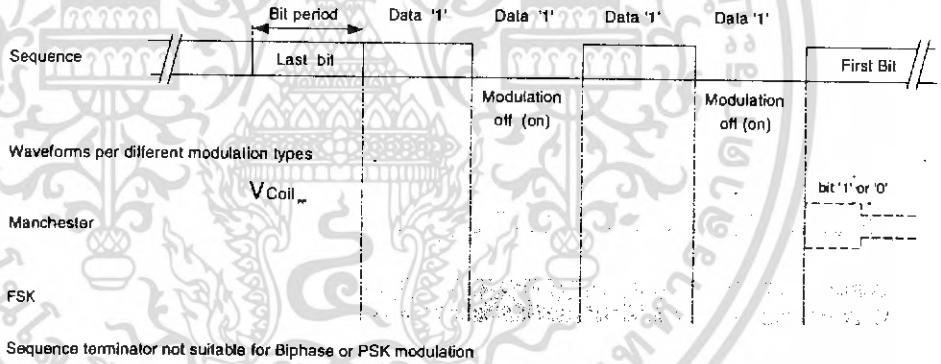


Figure 8. e5550-compatible Sequence Terminator Waveforms



Reader to Tag Communication

Data is written to the tag by interrupting the RF field with short field gaps (on-off keying) in accordance with the e5550 write method. The time between two gaps encodes the '0/1' information to be transmitted (pulse interval encoding). The duration of the gaps is usually 50 µs to 150 µs. The time between two gaps is nominally 24 field clocks for a '0' and 54 field clocks for a '1'. When there is no gap for more than 64 field clocks after a previous gap, the T5557 exits the write mode. The tag starts with the command execution if the correct number of bits were received. If there is a failure detected the T5557 does not continue and will enter regular-read mode.

Start Gap

The initial gap is referred to as the start gap. This triggers the reader to tag communication. During this mode of operation, the receive damping is permanently enabled to ease gap detection. The start gap may need to be longer than subsequent gaps in order to be detected reliably.

A start gap will be accepted at any time after the mode register has been loaded (≥ 3 ms). A single gap will not change the previously selected page (by former opcode '10' or '11').

Figure 9. Start of Reader to Tag Communication

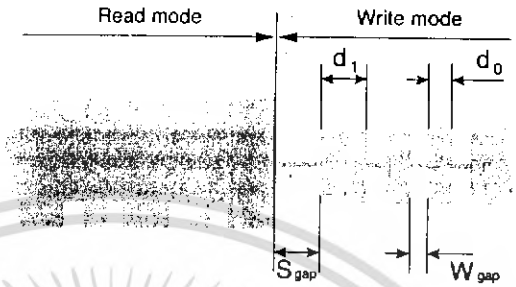


Table 2. Write Data Decoding Scheme

Parameters	Remark	Symbol	Min.	Max.	Unit
Start gap		S_{gap}	10	50	FC
Write gap	Normal write mode	W_{gap}	8	30	FC
Write data in normal mode	'0' data	d_0	16	31	FC
	'1' data	d_1	48	63	FC

Write Data Protocol

The T5557 expects to receive a dual bit opcode as the first two bits of a reader command sequence. There are three valid opcodes:

- The opcodes '10' and '11' precede all block write and direct access operations for page 0 and page 1
- The RESET opcode '00' initiates a POR cycle
- The opcode '01' precedes all test mode write operations. Any test mode access is ignored after master key (bits 1..4) in block 0 has been set to '6'. Any further modifications of the master key are prohibited by setting the lock bit of block 0 or the OTP bit.

Writing has to follow these rules:

- Standard write needs the opcode, the lock bit, 32 data bits and the 3-bit address (38 bits total)
- Protected write (PWD bit set) requires a valid 32-bit password between opcode and data, address bits
- For the AOR wake-up command an opcode and a valid password are necessary to select and activate a specific tag

Note: The data bits are read in the same order as written.

If the transmitted command sequence is invalid, the T5557 enters regular-read mode with the previously selected page (by former opcode '10' or '11').

Figure 10. Complete Writing Sequence

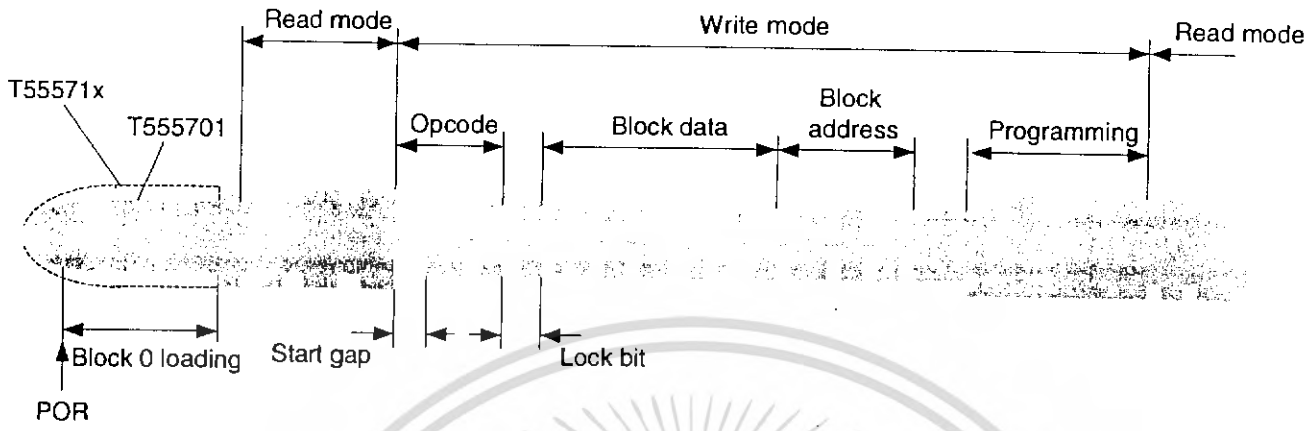
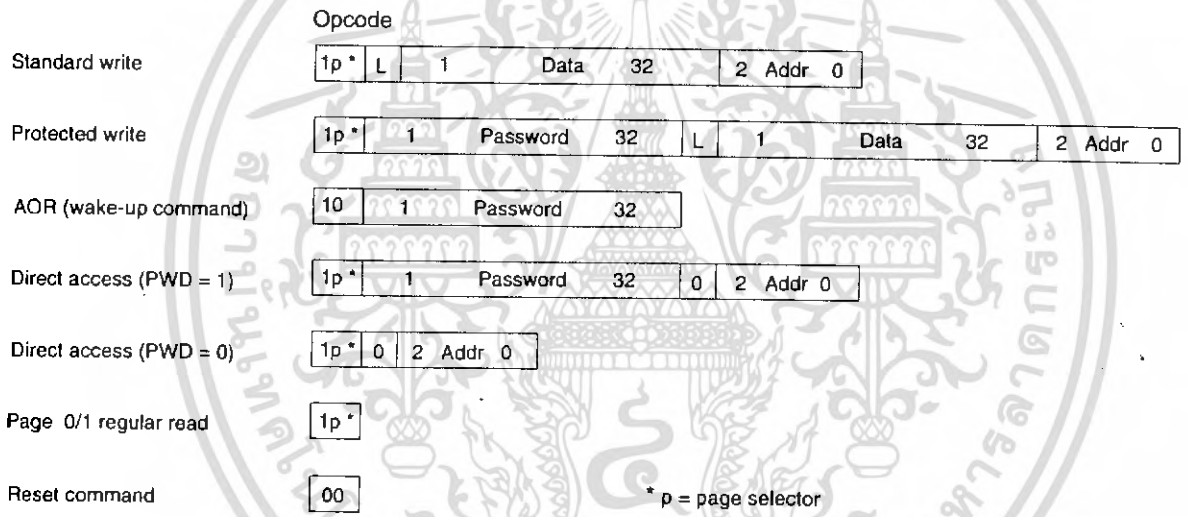


Figure 11. T5557 Command Formats



Password

When password mode is active (PWD = 1), the first 32 bits after the opcode are regarded as the password. They are compared bit by bit with the contents of block 7, starting at bit 1. If the comparison fails, the T5557 will not program the memory, instead it will restart in regular-read mode once the command transmission is finished.

Note: In password mode, MAXBLK should be set to a value below 7 to prevent the password from being transmitted by the T5557.

Each transmission of the direct access command (two opcode bits, 32 bits password, '0' bit plus 3 address bits = 38 bits) needs about 18 ms. Testing all possible combinations (about 4.3 billion) takes about two years.

Answer-On-Request (AOR) Mode

When the AOR bit is set, the T5557 does not start modulation in the regular-read mode after loading configuration block 0. The tag waits for a valid AOR data stream ("wake-up command") from the reader before modulation is enabled. The wake-up command consists of the opcode ('10') followed by a valid password. The selected tag will remain active until the RF field is turned off or a new command with a different password is transmitted which may address another tag in the RF field.

Table 3. T5557 — Modes of Operation

PWD	AOR	Behavior of Tag after Reset Command or POR	De-activate Function
1	1	Answer-On-Request (AOR) mode: • Modulation starts after wake-up with a matching password • Programming needs valid password	Command with non-matching password deactivates the selected tag
1	0	Password mode: • Modulation in regular-read mode starts after reset • Programming and direct access needs valid password	
0	--	Normal mode: • Modulation in regular-read mode starts after reset • Programming and direct access without password	

Figure 12. Answer-On-Request (AOR) Mode

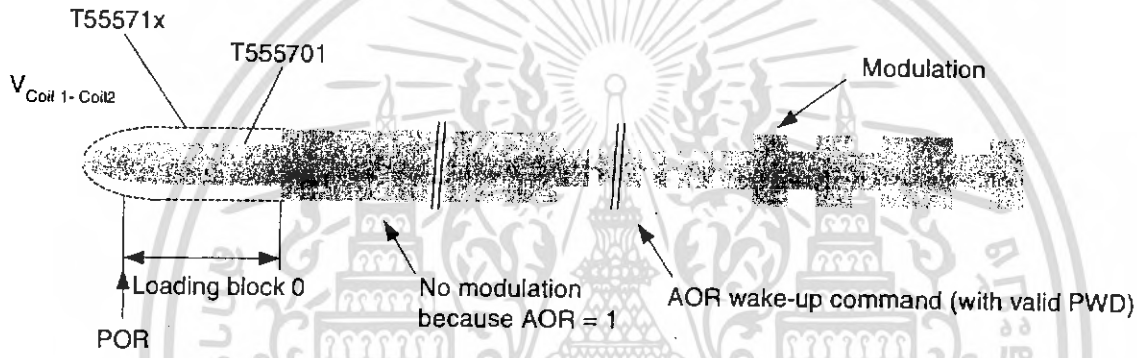


Figure 13. Coil Voltage after Programming of a Memory Block

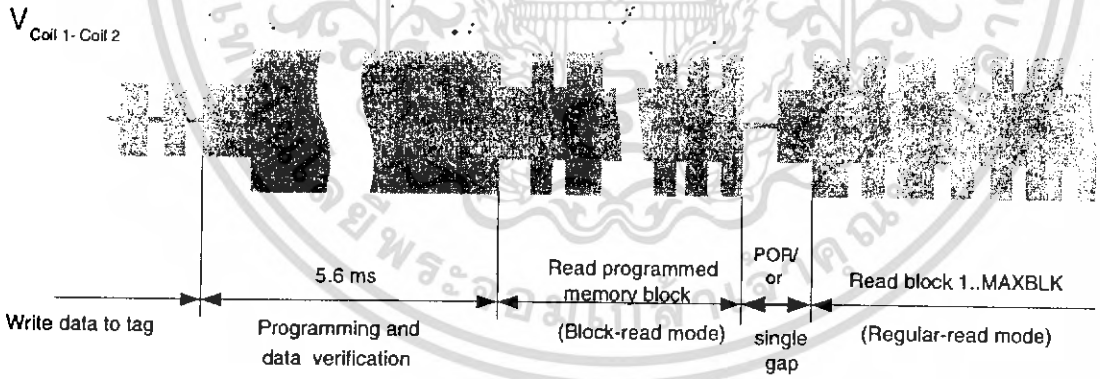
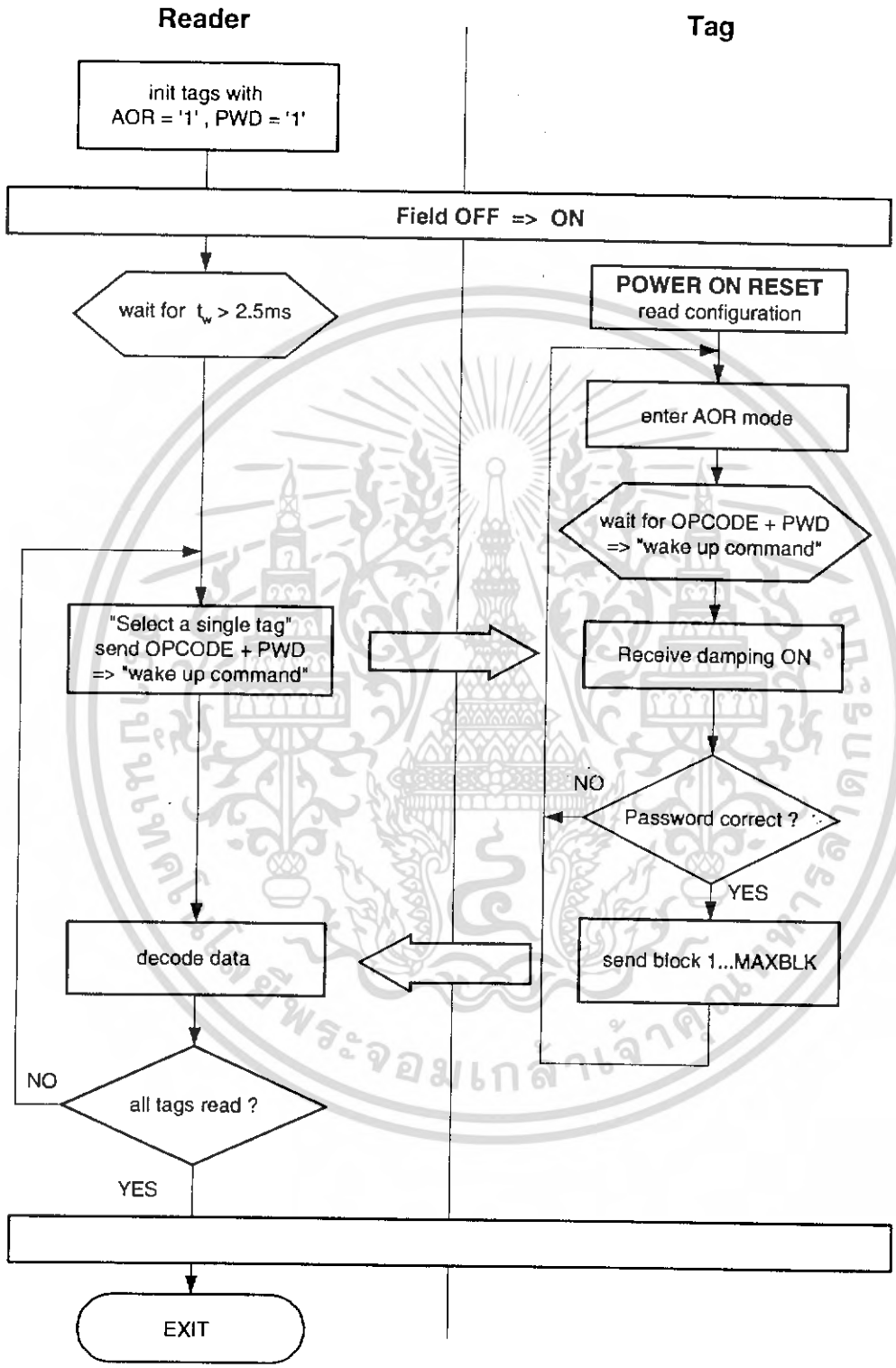


Figure 14. Anticollision Procedure Using AOR Mode



Programming

When all necessary information has been received by the T5557, programming may proceed. There is a clock delay between the end of the writing sequence and the start of programming.

Typical programming time is 5.6 ms. This cycle includes a data verification read to grant secure and correct programming. After programming was executed successfully, the T5557 enters block-read mode transmitting the block just programmed (see Figure 13 on page 10).

Note: This timing and behavior is different from the e555x-family predecessors.

Error Handling

Several error conditions can be detected to ensure that only valid bits are programmed into the EEPROM. There are two error types, which lead to two different actions.

Errors During Writing

The following detectable errors could occur during writing data into the T5557:

- Wrong number of field clocks between two gaps (i.e., not a valid '1' or '0' pulse stream)
- Password mode is activated and the password does not match the contents of block 7
- The number of bits received in the command sequence is incorrect

Valid bit counts accepted by the T5557 are:

Password write	70 bits	(PWD = 1)
Standard write	38 bits	(PWD = 0)
AOR wake up	34 bits	(PWD = 1)
Direct access with PWD	38 bits	(PWD = 1)
Direct access	6 bits	(PWD = 0)
Reset command	2 bits	
Page 0/1 regular-read	2 bits	

If any of these erroneous conditions were detected, the T5557 enters regular-read mode, starting with block 1 of the page defined in the command sequence.

Errors Before/During Programming

If the command sequence was received successfully, the following error could still prevent programming:

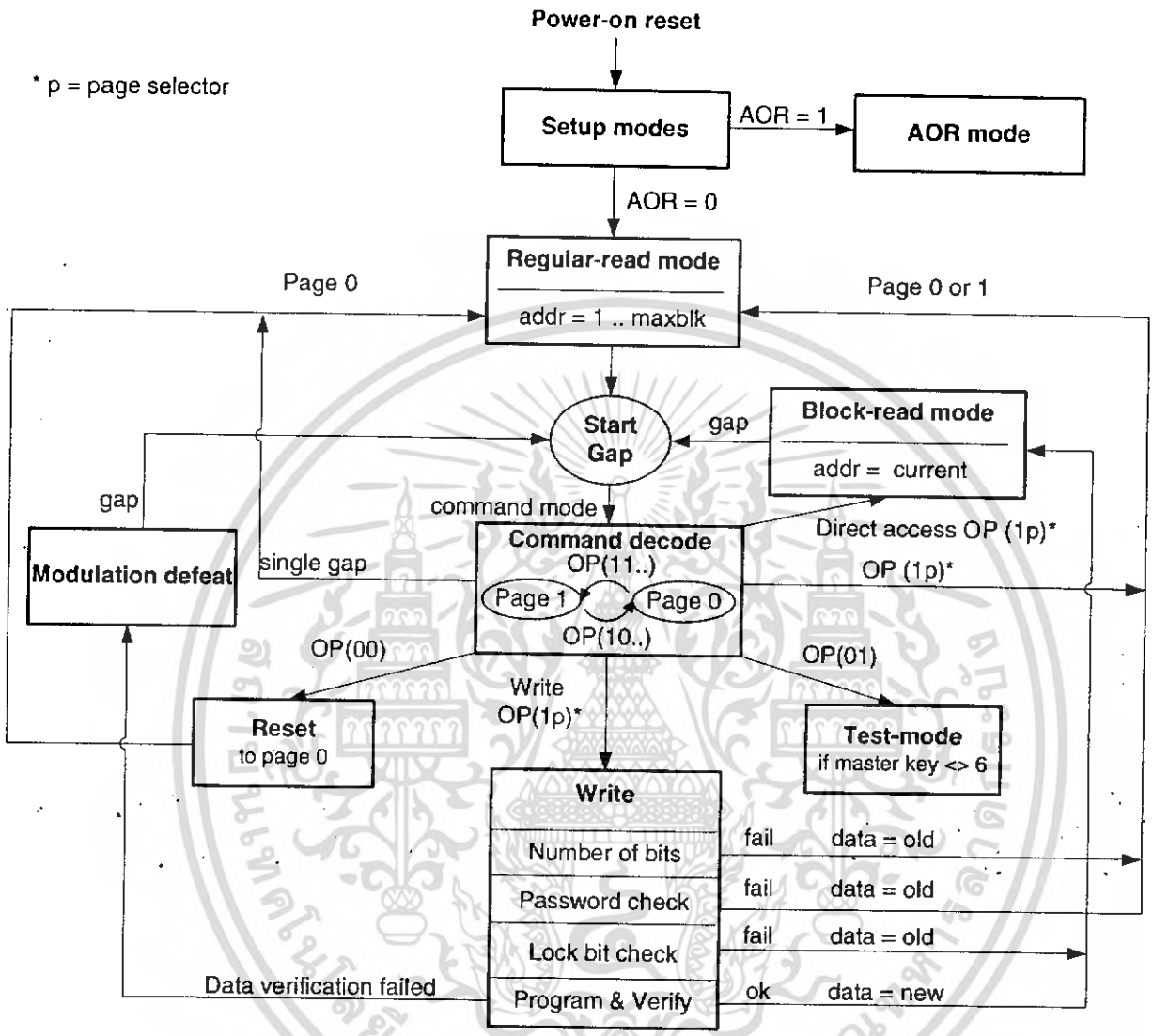
- The lock bit of the addressed block is set already
- In case of a locked block, programming mode will not be entered. The T5557 reverts to block-read mode continuously transmitting the currently addressed block.

If the command sequence is validated and the addressed block is not write protected, the new data will be programmed into the EEPROM memory. The new state of the block write protection bit (lock bit) will be programmed at the same time accordingly.

Each programming cycle consists of 4 consecutive steps: erase block, erase verification (data = '0'), programming, write verification (corresponding data bits = '1').

- If a data verification error is detected after an executed data block programming, the tag will stop modulation (modulation defeat) until a new command is transmitted.

Figure 15. T5557 Functional Diagram



T5557 in Extended Mode (X-mode)

In general, the block 0 setting of the master key (bits 1 to 4) to the value '6' or '9' together with the X-mode bit will enable the extended mode functions.

- Master key = '9': Test mode access and extended mode are both enabled.
- Master key = '6': Any test mode access will be denied but the extended mode is still enabled.

Any other master key setting will prevent the activation of the T5557 extended mode options, even when the X-mode bit is set.

Binary Bit-rate Generator

In extended mode the data rate is binary programmable to operate at any data rate between RF/2 and RF/128 as given in the formula below.

Data rate = RF/(2n+2)

OTP Functionality

If the OTP bit is set to '1', all memory blocks are write protected and behave as if all lock bits are set to 1. If the master key is set to '6' additionally, the T5557 mode of operation is locked forever (= OTP functionality).

If the master key is set to '9', the test-mode access allows the re-configuration of the tag again.

Figure 16. Block 0 — Configuration Map in Extended Mode (X-mode)

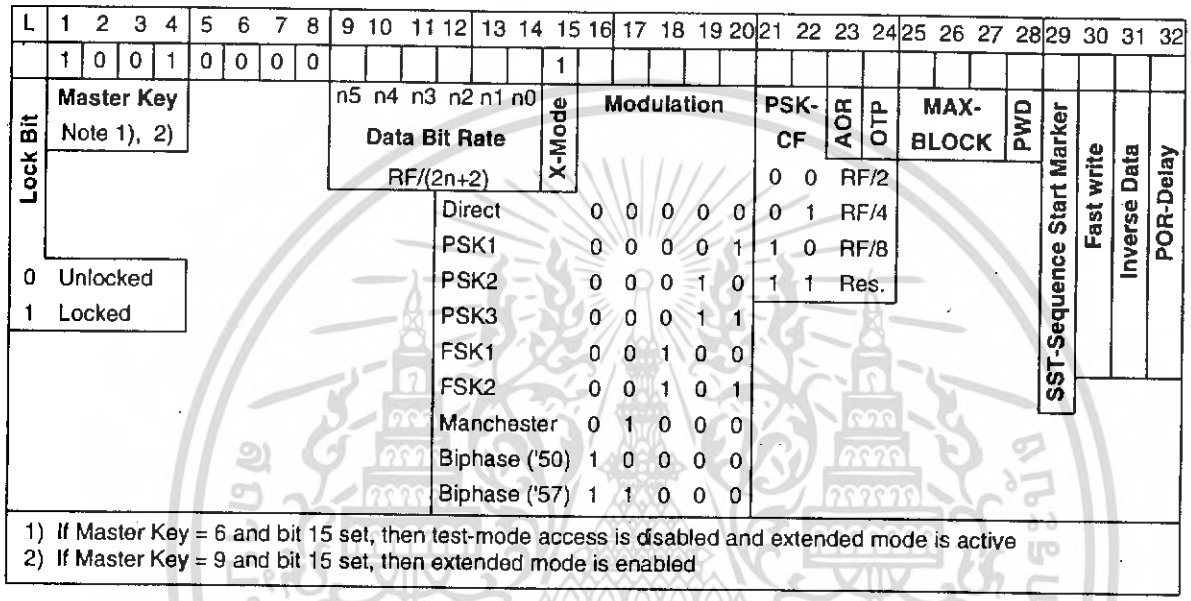


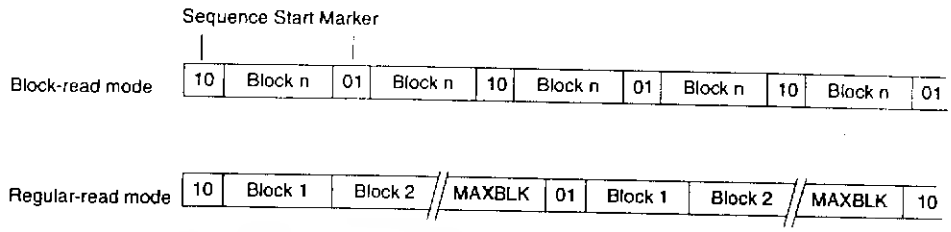
Table 4. T5557 Types of Modulation in Extended Mode

Mode	Direct Data Output Encoding	Inverse Data Output Encoding
FSK1[1]	FSK/5-/8 '0' = RF/5; '1' = RF/8	FSK/8-/5 '0' = RF/8; '1' = RF/5 (= FSK1a)
FSK2[1]	FSK/10-/8 '0' = RF/10; '1' = RF/8	FSK/8-/10 '0' = RF/8; '1' = RF/10 (= FSK2a)
PSK1[2]	Phase change when input changes	Phase change when input changes
PSK2[2]	Phase change on bit clock if input high	Phase change on bit clock if input low
PSK3[2]	Phase change on rising edge of input	Phase change on falling edge of input
Manchester	'0' = falling edge, '1'= rising edge on mid-bit	'1' = falling edge, '1'= rising edge on mid-bit
Biphase 1 ('50)	'1' creates an additional mid-bit change	'0' creates an additional mid-bit change
Biphase 2 ('57)	'0' creates an additional mid-bit change	'1' creates an additional mid-bit change
NRZ	'1'= damping on, '0'= damping off	'0'= damping on, '1'= damping off

Notes: 1. A common multiple of bitrate and FSK frequencies is recommended.
2. In PSK mode the selected data rate has to be an integer multiple of the PSK sub-carrier frequency.

Sequence Start Marker

Figure 17. T5557 Sequence Start Marker in Extended Mode

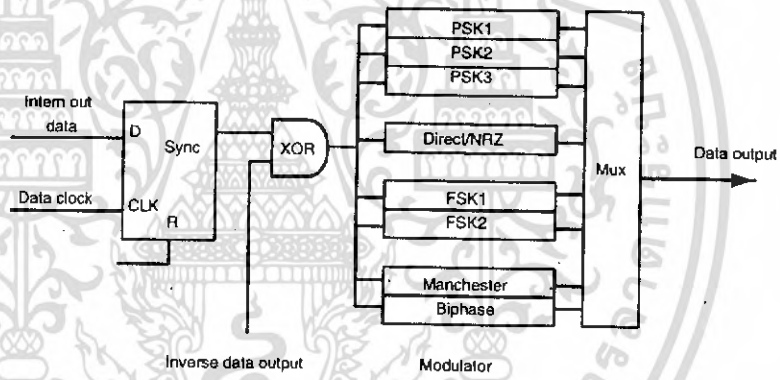


The T5557 sequence start marker is a special damping pattern, which may be used to synchronize the reader. The sequence start marker consists of two bits ('01' or '10') which are inserted as header before the first block to be transmitted if the bit 29 in extended mode ist set. At the start of a new block sequence, the value of the two bits is inverted.

Inverse Data Output

The T5557 supports in its extended mode (X-mode) an inverse data output option. If inverse data is enabled, the modulator as shown in Figure 18 works on inverted data (see Table 4 on page 14). This function is supported for all basic types of encoding.

Figure 18. Data Encoder for Inverse Data Output



Fast Write

In the optional fast write mode the time between two gaps is nominally 12 field clocks for a '0' and 27 field clocks for a '1'. When there is no gap for more than 32 field clocks after a previous gap, the T5557 will exit the write mode. Please refer to Table 5 and Figure 8 on page 7.

Table 5. Fast Write Data Decoding Schemes

Parameters	Remark	Symbol	Min.	Max.	Unit
Start gap	–	S_{gap}	10	50	FC
Write gap	Normal write mode	Wn_{gap}	8	30	FC
	Fast write mode	Wf_{gap}	8	20	FC
Write data in normal mode	'0' data	d_0	16	31	FC
	'1' data	d_1	48	63	FC
Write data in fast mode	'0' data	d_0	8	15	FC
	'1' data	d_1	24	31	FC

Figure 19. Example of Manchester Coding with Data Rate RF/16

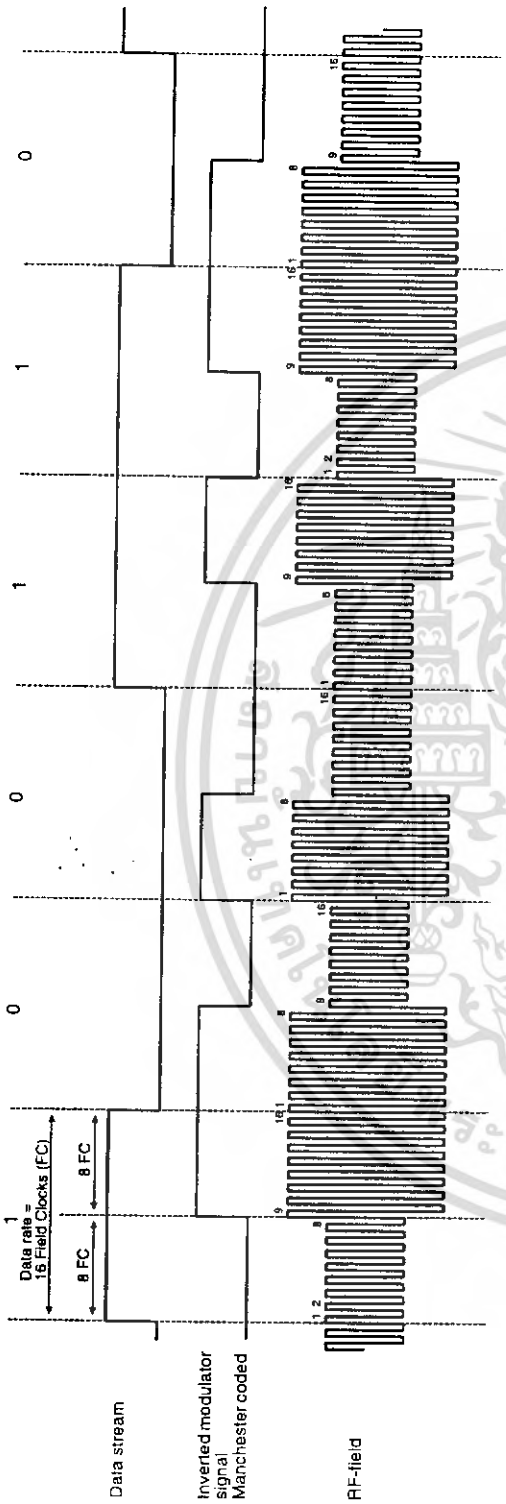


Figure 20. Example of Biphase Coding with Data Rate RF/16

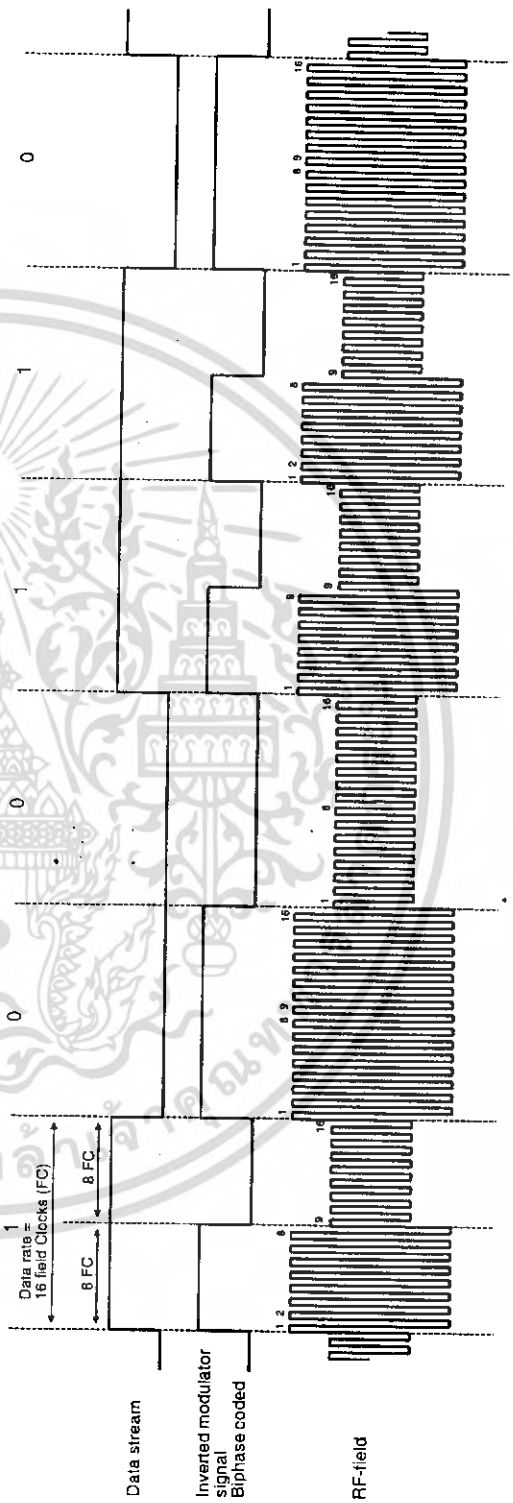


Figure 21. Example: FSK1a Coding with Data Rate RF/40, Subcarrier f_0 = RF/8, f_1 = RF/5

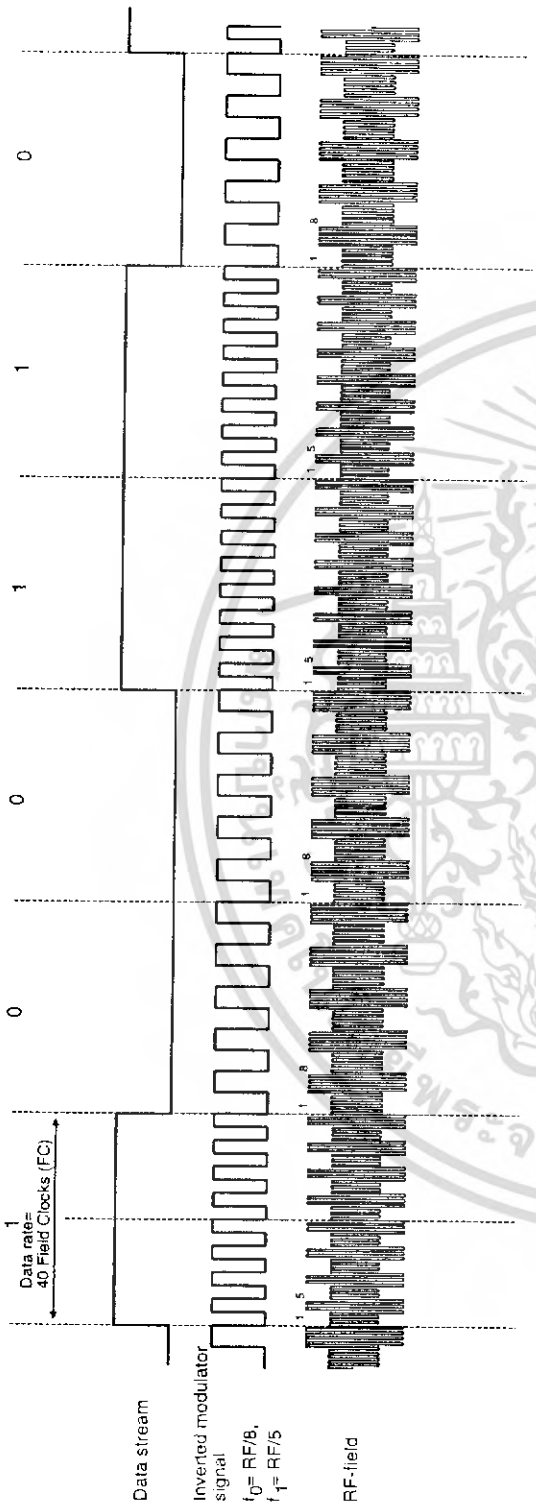


Figure 22. Example of PSK1 Coding with Data Rate RF/16

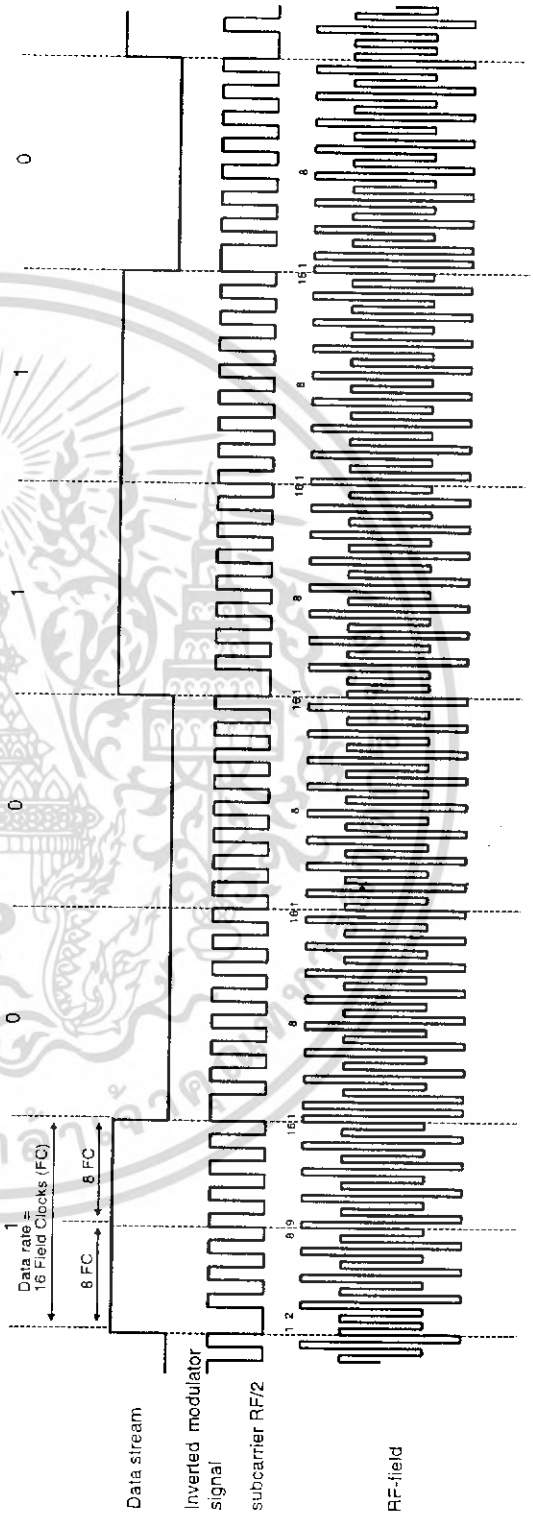


Figure 23. Example of PSK2 Coding with Data Rate RF/16

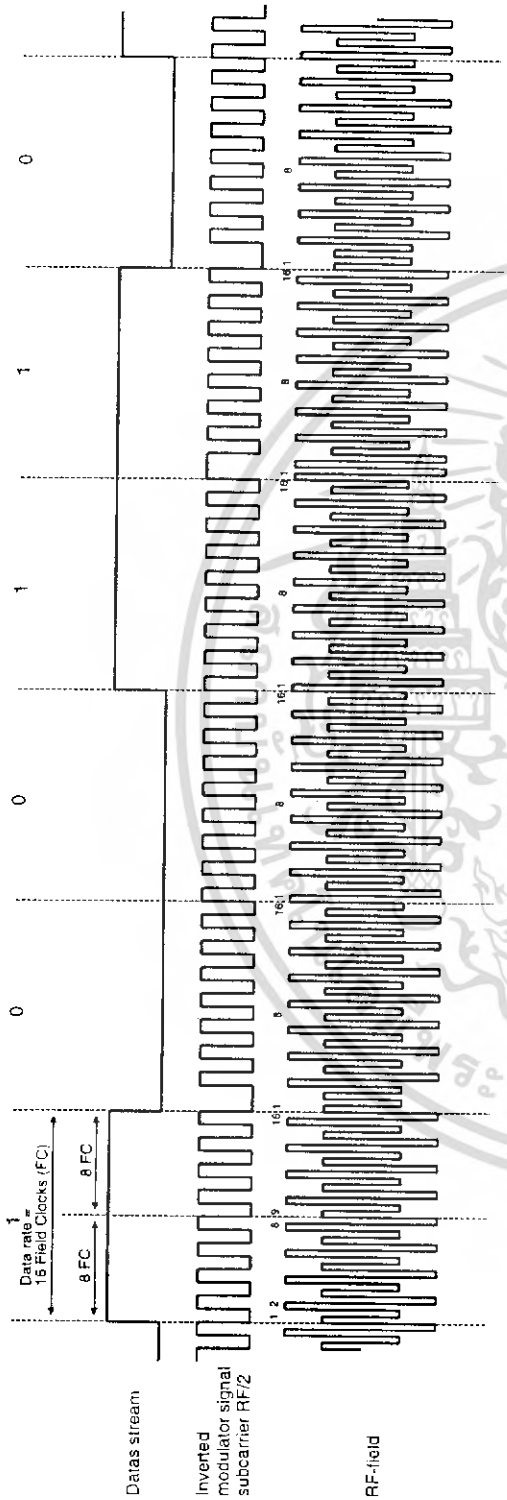
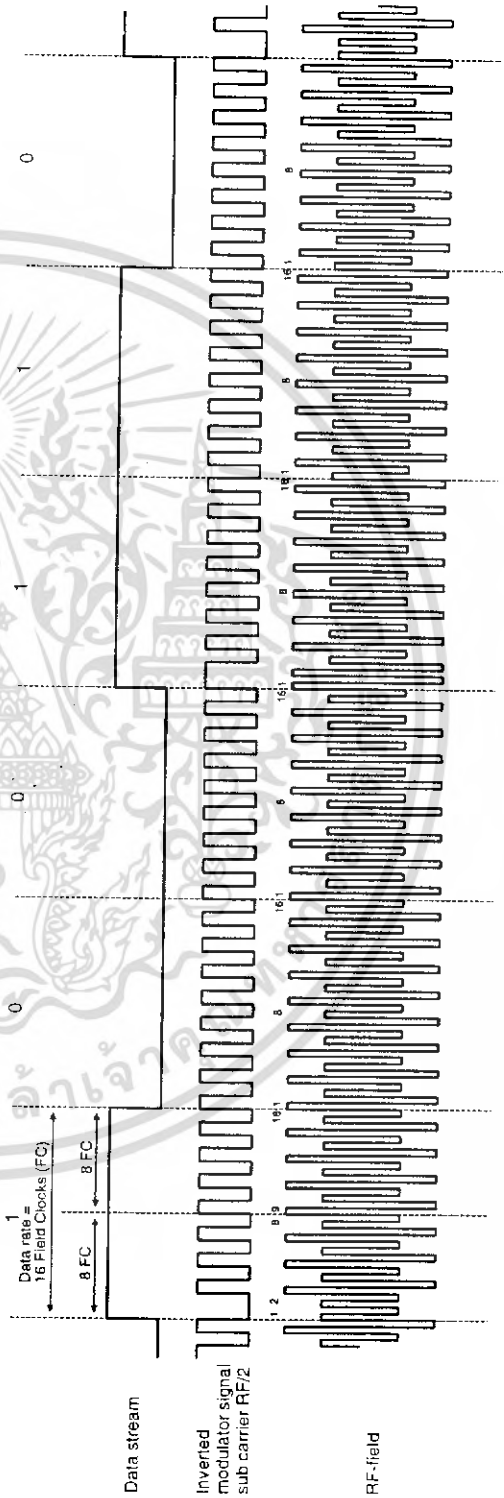


Figure 24. Example of PSK3 Coding with Data Rate RF/16



Absolute Maximum Ratings

Stresses beyond those listed under "Absolute Maximum Ratings" may cause permanent damage to the device. This is a stress rating only and functional operation of the device at these or any other conditions beyond those indicated in the operational sections of this specification is not implied. Exposure to absolute maximum rating conditions for extended periods may affect device reliability.

Parameters	Symbol	Value	Unit
Maximum DC current into Coil 1/Coil 2	I_{coil}	20	mA
Maximum AC current into Coil 1/Coil 2 f = 125 kHz	$I_{coil\ p}$	20	mA
Power dissipation (dice) (free-air condition, time of application: 1 s)	P_{tot}	100	mW
Electrostatic discharge maximum to MIL-Standard 883 C method 3015	V_{max}	4000	V
Operating ambient temperature range	T_{amb}	-40 to +85	°C
Storage temperature range (data retention reduced)	T_{stg}	-40 to +150	°C

Electrical Characteristics

T_{amb} = +25° C; f_{coil} = 125 kHz; unless otherwise specified

No.	Parameters	Test Conditions	Symbol	Min.	Typ.	Max.	Unit	Type*
1	RF frequency range		f_{RF}	100	125	150	kHz	
2.1	Supply current (without current consumed by the external LC tank circuit)	T_{amb} = 25° C[1] (see Figure 24 on page 18)	I_{DD}		1.5	3	µA	T
2.2		Read – full temperature range			2	4	µA	Q
2.3		Programming full temperature range			25	40	µA	Q
3.1	Coil voltage (AC supply)	POR threshold (50 mV hysteresis)	$V_{coil\ pp}$	3.2	3.6	4.0	V	Q
3.2		Read mode and write command[2]		6		V_{clamp}	V	Q
3.3		Program EEPROM [2]		8		V_{clamp}	V	Q
4	Start-up time	$V_{coil\ pp}$ = 6 V	$t_{startup}$		2.5	3	ms	Q
5	Clamp voltage	10 mA current into Coil 1/2	V_{clamp}	17		23	V	T

*) Type means: T: directly or indirectly tested during production; Q: guaranteed based on initial product qualification data

Notes:
1. I_{DD} measurement setup R = 100 k; V_{CLK} = V_{coil} = 5 V; EEPROM programmed to 00 ... 000 (erase all); chip in modulation defeat. I_{DD} = (V_{OUTmax} - V_{CLK})/R
2. Current into Coil 1/Coil 2 is limited to 10 mA. The damping circuitry has the same structure as the e5550. The damping characteristics are defined by the internally limited supply voltage (= minimum AC coil voltage)
3. V_{mod} measurement setup: R = 2.3 k; V_{CLK} = 3 V; setup with modulation enabled (see Figure 25 on page 20).
4. Since EEPROM performance is influenced by assembly processes, Atmel confirms the parameters for DOW (tested dice on uncutted wafer) delivery.
5. The tolerance of the on-chip resonance capacitor C_r is ±10% at 3σ over whole production. The capacitor tolerance is ±3% at 3σ on a wafer basis.
6. The tolerance of the microcodule resonance capacitor C_r is ±5% at 3σ over whole production.

Electrical Characteristics

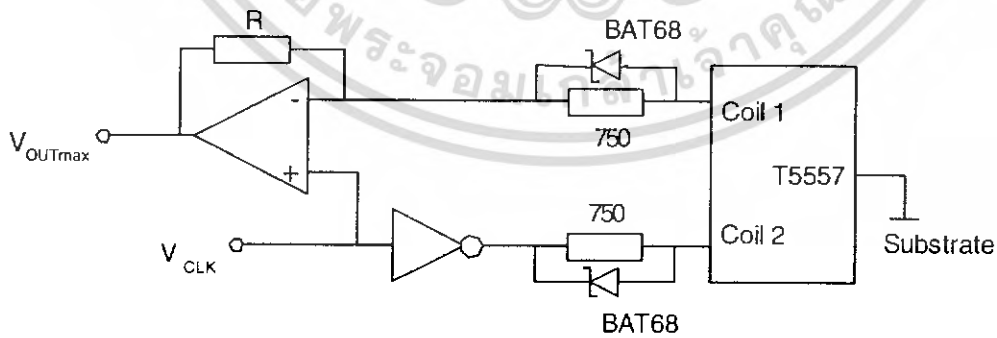
T_{amb} = +25° C; f_{coil} = 125 kHz; unless otherwise specified

No.	Parameters	Test Conditions	Symbol	Min.	Typ.	Max.	Unit	Type*
6.1	Modulation parameters	V_{coilpp} = 6 V on test circuit generator and modulation ON[3]	$V_{mod\,pp}$		4.2	4.8	V	T
6.2			$I_{mod\,pp}$	400	600		µA	T
6.3		Thermal stability	V_{mod}/T_{amb}		-6		mV/° C	Q
7	Programming time	From last command gap to re-enter read mode (64 + 648 internal clocks)	T_{prog}	5	5.7	6	ms	T
8	Endurance	Erase all / Write all[4]	n_{cycle}	100000			Cycles	Q
9.1	Data retention	Top = 55° C[4]	$t_{retention}$	10	20	50	Years	
9.2		Top = 150° C[4]	$t_{retention}$	96			hrs	T
9.3		Top = 250° C[4]	$t_{retention}$	24			hrs	Q
10	Resonance capacitor	Mask option[5]	C_r	70	78	86	pF	T
11.1	Microdule capacitor parameters	Capacitance tolerance T_{amb}	C_r	313.5	330	346.5	pF	T
11.2		Temperature coefficient	TBD	TBD	TBD	TBD	TBD	TBD
11.3			TBD	TBD	TBD	TBD	TBD	TBD

*) Type means: T: directly or indirectly tested during production; Q: guaranteed based on initial product qualification data

Notes:
1. I_{DO} measurement setup R = 100 k; V_{CLK} = V_{coil} = 5 V: EEPROM programmed to 00 ... 000 (erase all); chip in modulation defeat. I_{DO} = (V_{OUTmax} - V_{CLK})/R
2. Current into Coil 1/Coil 2 is limited to 10 mA. The damping circuitry has the same structure as the e5550. The damping characteristics are defined by the internally limited supply voltage (= minimum AC coil voltage)
3. V_{mod} measurement setup: R = 2.3 k; V_{CLK} = 3 V; setup with modulation enabled (see Figure 25 on page 20).
4. Since EEPROM performance is influenced by assembly processes, Atmel confirms the parameters for DOW (tested dice on uncutted wafer) delivery.
5. The tolerance of the on-chip resonance capacitor C_r is ±10% at 3σ over whole production. The capacitor tolerance is ±3% at 3σ on a wafer basis.
6. The tolerance of the microcodule resonance capacitor C_r is ±5% at 3σ over whole production.

Figure 25. Measurement Setup for I_{DD} and V_{mod}



Ordering Information[2]

T5557	ab	Mcc	- xxx	Package		Drawing
				- DDW	- Dice on wafer, 6" un-sawn wafer, thickness 300 µm	
				- DDT	- Dice in Tray (waffle pack), thickness 300 µm	
				- DBW	- Dice on solder bumped wafer, thickness 390 µm	see Figure 27 on page 23
					Sn63Pb37 on 5 µm Ni/Au, height 70 µm	see Figure 28 on page 23
				- TAS	- SO8 Package	see Figure 31 on page 26
				- PAE	- MOA2 Micro-Module	see Figure 29 on page 24
				Customer ID[1]		
					- Atmel standard (corresponds to "00")	
				M01	- Customer 'X' unique ID code[1]	
				11	- 2 Pads without on-chip C	see Figure 26 on page 22
				14	- 4 Pads with on-chip 75 pF	see Figure 27 on page 23
				15	- Micro - Module with 330 pF	see Figure 29 on page 24
				01	- 2 Pads without C; Damping during initialisation	see Figure 26 on page 22

Notes: 1. Unique customer ID code programming according to Figure 5 is linked to a minimum order quantity of 1 Mio parts per year.
2. For available order codes refer to Atmel Sales/Marketing.

Ordering Examples (Recommended)

T555711-DDW Tested dice on unsawn 6" wafer, thickness 300 µm, no on-chip capacitor, no damping during POR initialisation; especially for ISO 11784/785 and access control applications

Available Order Codes

T555711-DDW, DDT, TAS
T555714-DDW, DBW, TAS
T555715-PAE

Package Information

Figure 26. 2 Pad Layout for Wire Bonding

Dimensions in µm

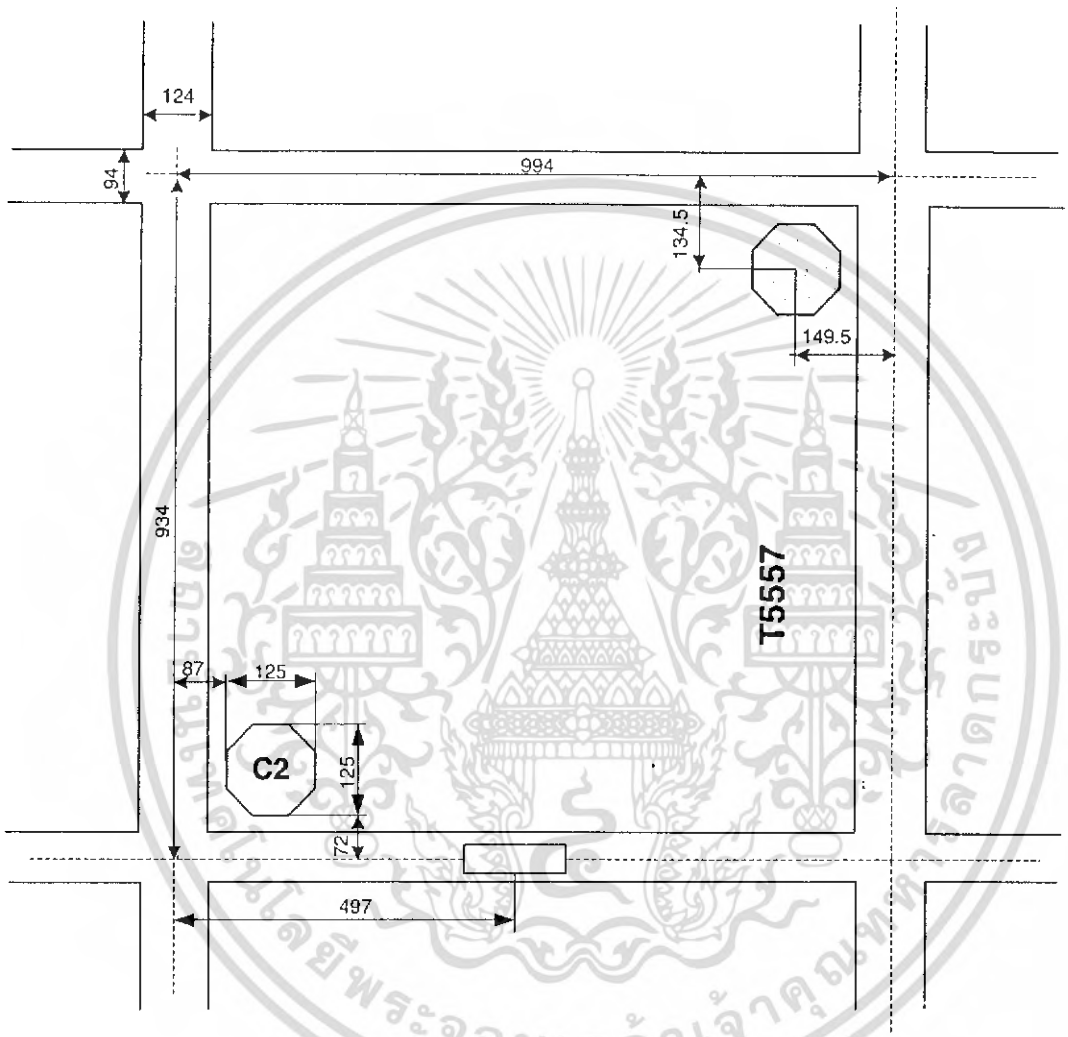


Figure 27. 4 Pad Flip-chip Version with 70 µm Solder Bumps

Dimensions in µm

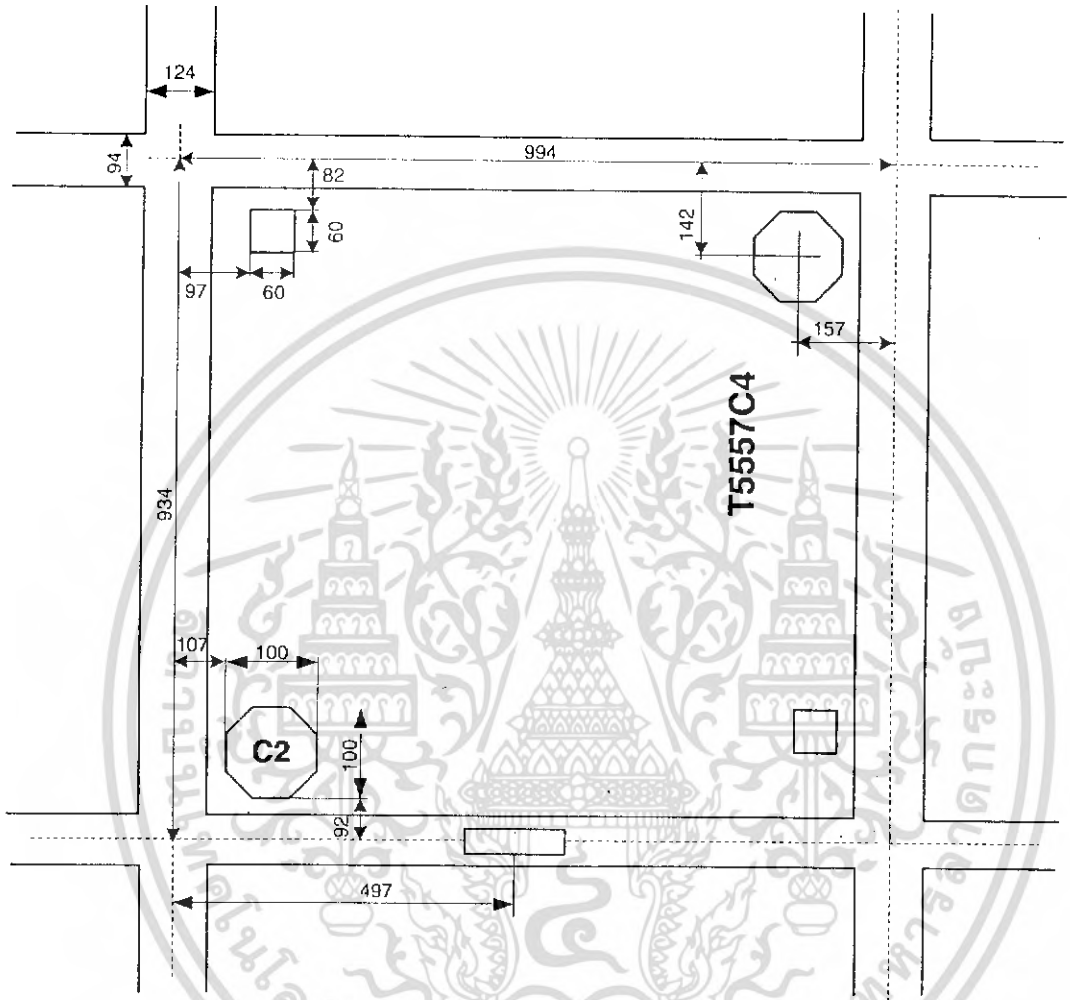


Figure 28. Solder bump on NiAu

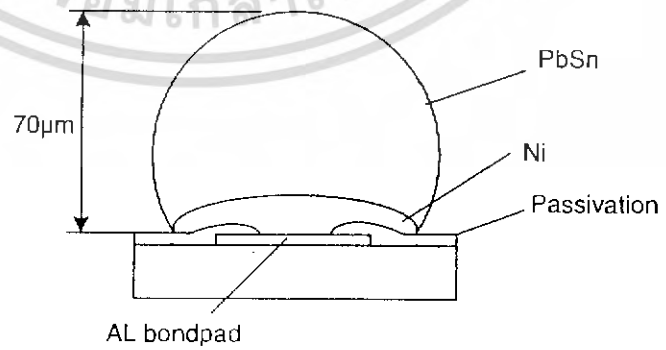
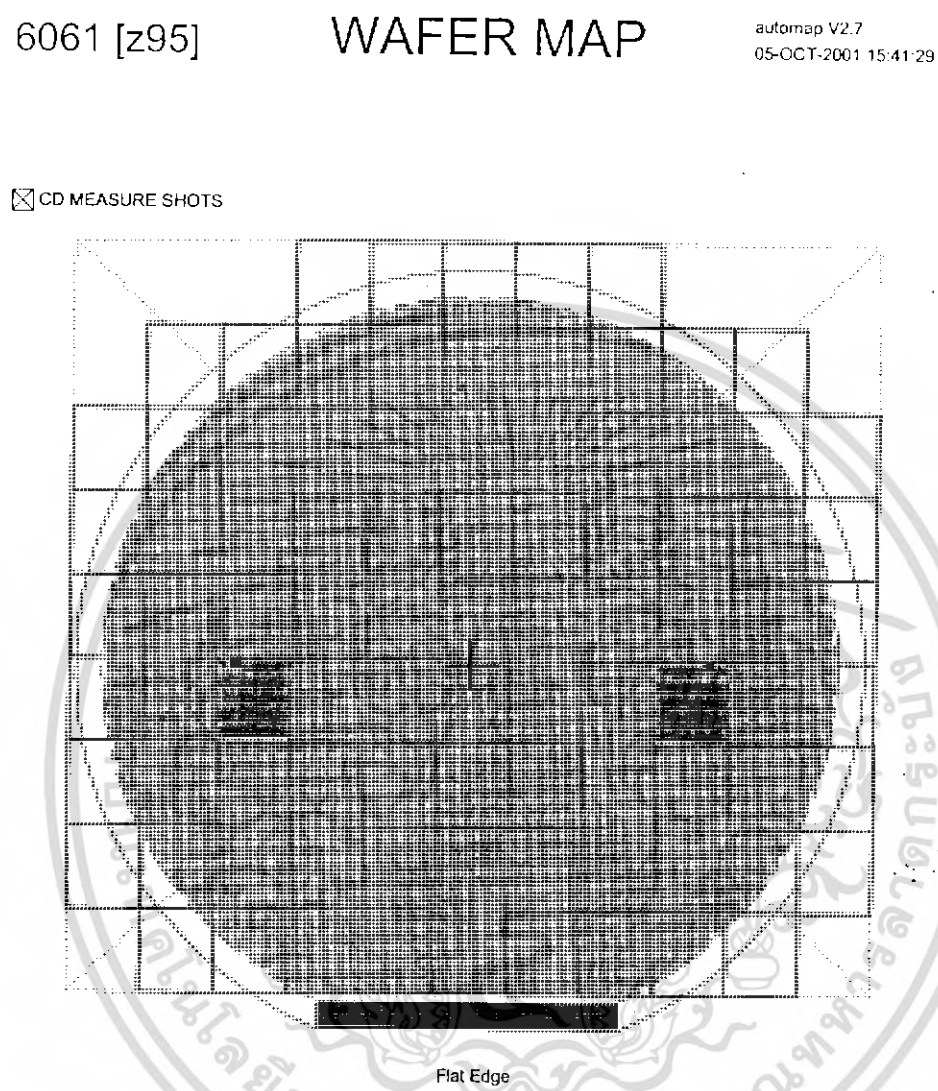


Figure 29. Wafer Map



Die: 0.894x0.864, Step: 0.994x0.934, N: 14x17, Frame Step: 13.916x15.878

> Shift-ASML=[0.3;-6.9] : 15539 dice, 87 shots (11cols x 9rows)

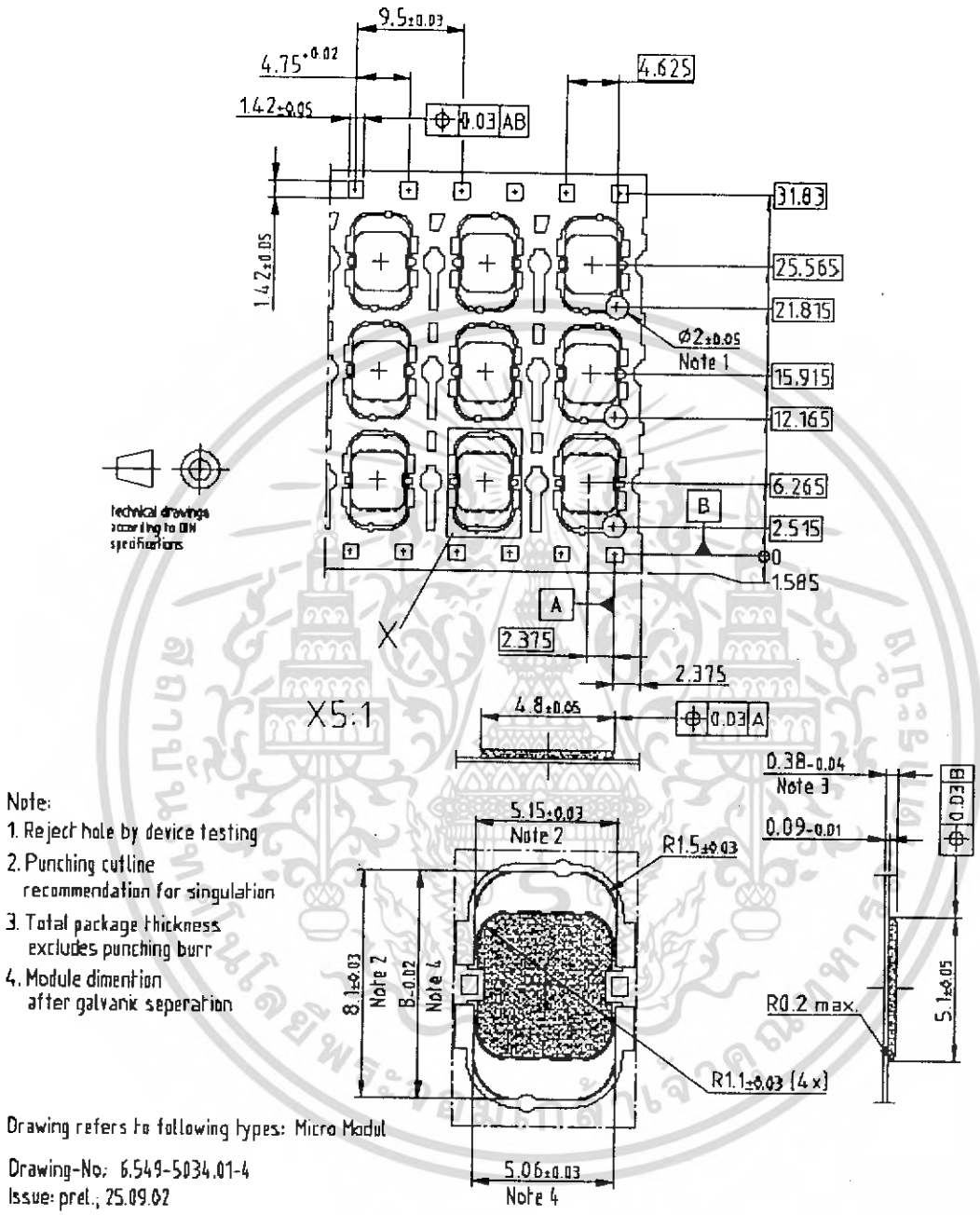
> Shift-CANON/ALARM/SEM=[0.3;-6.9] - W2=[-13.152;6.9] - W1=[-6.648;6.9]

Failed Die Identification

Every die on the wafer not passing Atmel test sequence is marked with inch.
The inch dot specification:

- Dot size: 200 µm
- Position: center of die
- Color: black

Figure 30. NOA2 Micromodule



Note:
1. Reject hole by device testing
2. Punching cutline recommendation for singulation
3. Total package thickness excludes punching burr
4. Module dimention after galvanik seperation

Drawing refers to following types: Micro Modul

Drawing-No: 6.549-5034.01-4
Issue: prel.; 25.09.02

Figure 31. Shipping Reel

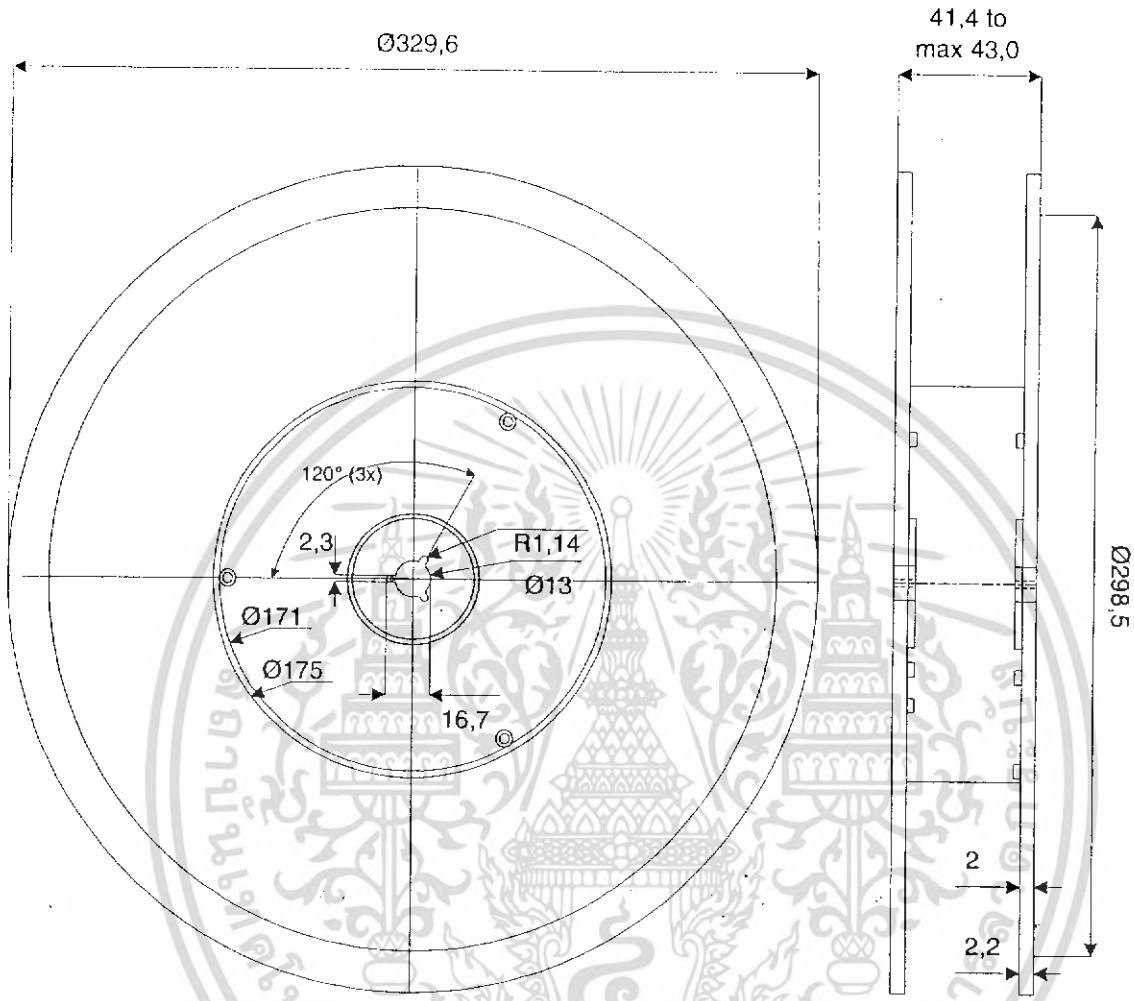


Figure 32. SO8 Package

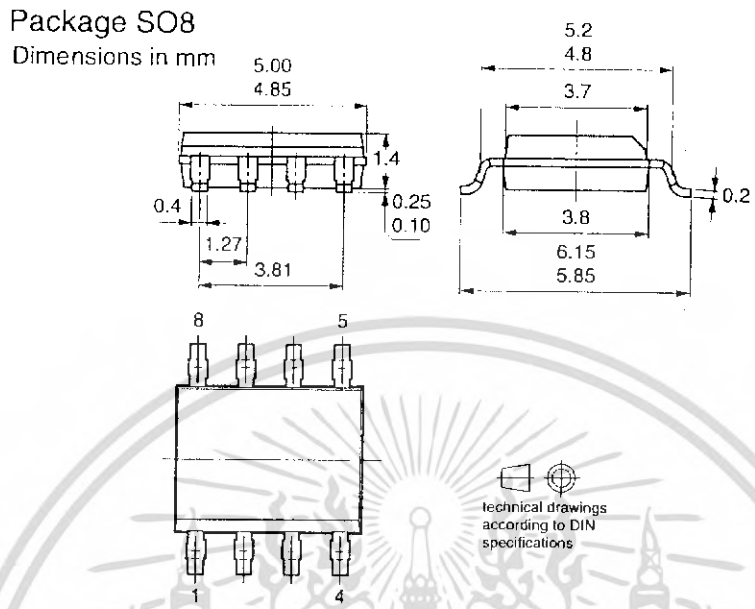
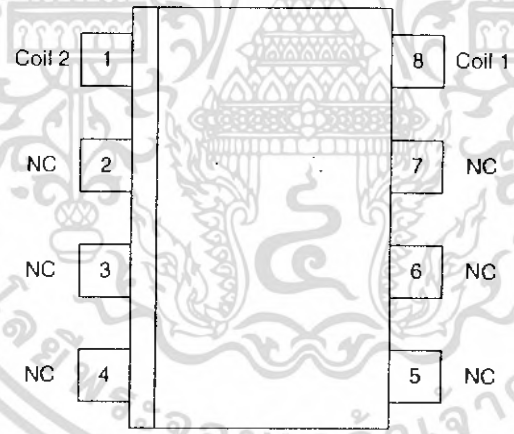


Figure 33. Pinning SO8



Atmel Corporation

2325 Orchard Parkway
San Jose, CA 95131, USA
Tel: 1(408) 441-0311
Fax: 1(408) 487-2600

Regional Headquarters

Europe
Atmel Sarl
Route des Arsenaux 41
Case Postale 80
CH-1705 Fribourg
Switzerland
Tel: (41) 26-426-5555
Fax: (41) 26-426-5500

Asia
Room 1219
Chinachem Golden Plaza
77 Mody Road Tsimshatsui
East Kowloon
Hong Kong
Tel: (852) 2721-9778
Fax: (852) 2722-1369

Japan
9F, Tonetsu Shinkawa Bldg.
1-24-8 Shinkawa
Chuo-ku, Tokyo 104-0033
Japan
Tel: (81) 3-3523-3551
Fax: (81) 3-3523-7581

Atmel Operations

Memory
2325 Orchard Parkway
San Jose, CA 95131, USA
Tel: 1(408) 441-0311
Fax: 1(408) 436-4314

Microcontrollers
2325 Orchard Parkway
San Jose, CA 95131, USA
Tel: 1(408) 441-0311
Fax: 1(408) 436-4314

La Chantrerie
BP 70602
44306 Nantes Cedex 3, France
Tel: (33) 2-40-18-18-18
Fax: (33) 2-40-18-19-60

ASIC/ASSP/Smart Cards
Zone Industrielle
13106 Rousset Cedex, France
Tel: (33) 4-42-53-60-00
Fax: (33) 4-42-53-60-01

1150 East Cheyenne Mtn. Blvd.
Colorado Springs, CO 80906, USA
Tel: 1(719) 576-3300
Fax: 1(719) 540-1759

Scottish Enterprise Technology Park
Maxwell Building
East Kilbride G75 0QR, Scotland
Tel: (44) 1355-803-000
Fax: (44) 1355-242-743

RF/Automotive
Theresienstrasse 2
Postfach 3535
74025 Heilbronn, Germany
Tel: (49) 71-31-67-0
Fax: (49) 71-31-67-2340

1150 East Cheyenne Mtn. Blvd.
Colorado Springs, CO 80906, USA
Tel: 1(719) 576-3300
Fax: 1(719) 540-1759

Biometrics/Imaging/Hi-Rel MPU/
High Speed Converters/RF Datacom
Avenue de Rochepleine
BP 123
38521 Saint-Egreve Cedex, France
Tel: (33) 4-76-58-30-00
Fax: (33) 4-76-58-34-80

Literature Requests
www.atmel.com/literature

Disclaimer: Atmel Corporation makes no warranty for the use of its products, other than those expressly contained in the Company's standard warranty which is detailed in Atmel's Terms and Conditions located on the Company's web site. The Company assumes no responsibility for any errors which may appear in this document, reserves the right to change devices or specifications detailed herein at any time without notice, and does not make any commitment to update the information contained herein. No licenses to patents or other intellectual property of Atmel are granted by the Company in connection with the sale of Atmel products, expressly or by implication. Atmel's products are not authorized for use as critical components in life support devices or systems.

© Atmel Corporation 2003. All rights reserved.

Atmel® and combinations thereof are the registered trademarks of Atmel Corporation or its subsidiaries.

Other terms and product names may be the trademarks of others.

Printed on recycled paper.

4517F–RFID–11/03

EM MICROELECTRONIC - MARIN SA

A COMPANY OF THE SWATCH GROUP

EM4095

Read/Write analog front end for 125kHz RFID Basestation

Description

The EM4095 (previously named P4095) chip is a CMOS integrated transceiver circuit intended for use in an RFID basestation to perform the following functions:

- antenna driving with carrier frequency
- AM modulation of the field for writable transponder
- AM demodulation of the antenna signal modulation induced by the transponder

communicate with a microprocessor via simple interface.

Features

- Integrated PLL system to achieve self adaptive carrier frequency to antenna resonant frequency
- No external quartz required
- 100 to 150 kHz carrier frequency range
- Direct antenna driving using bridge drivers
- Data transmission by OOK (100% Amplitude Modulation) using bridge driver
- Data transmission by Amplitude Modulation with externally adjustable modulation index using single ended driver
- Multiple transponder protocol compatibility (Ex: EM400X, EM4050 EM4150, EM4070, EM4170, EM4069 ...)
- Sleep mode 1µA
- USB compatible power supply range
- 40 to +85°C temperature range
- Small outline plastic package SO16

Applications

- Car immobiliser
- Hand held reader
- Low cost reader

Typical Operating Configuration

Read Only Mode

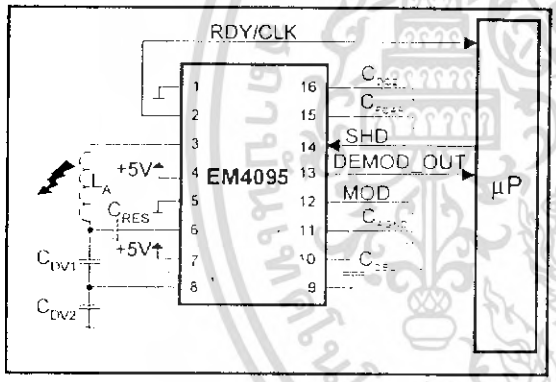


Fig. 1

Pin Assignment

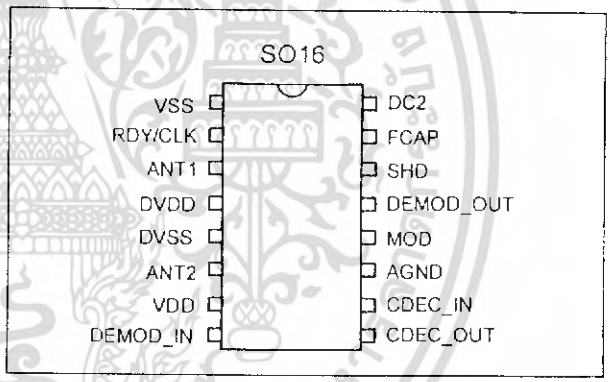


Fig. 3

Read/Write Mode

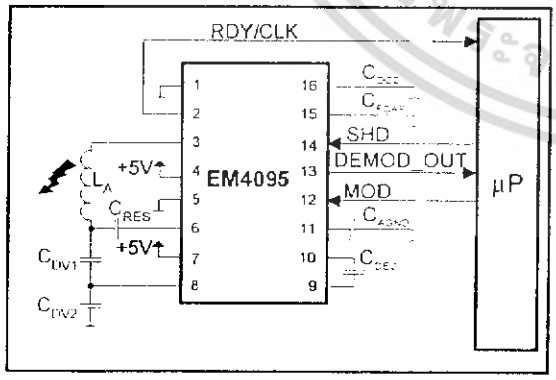


Fig. 2

Copyright © 2002, EM Microelectronic-Marin SA

System principle

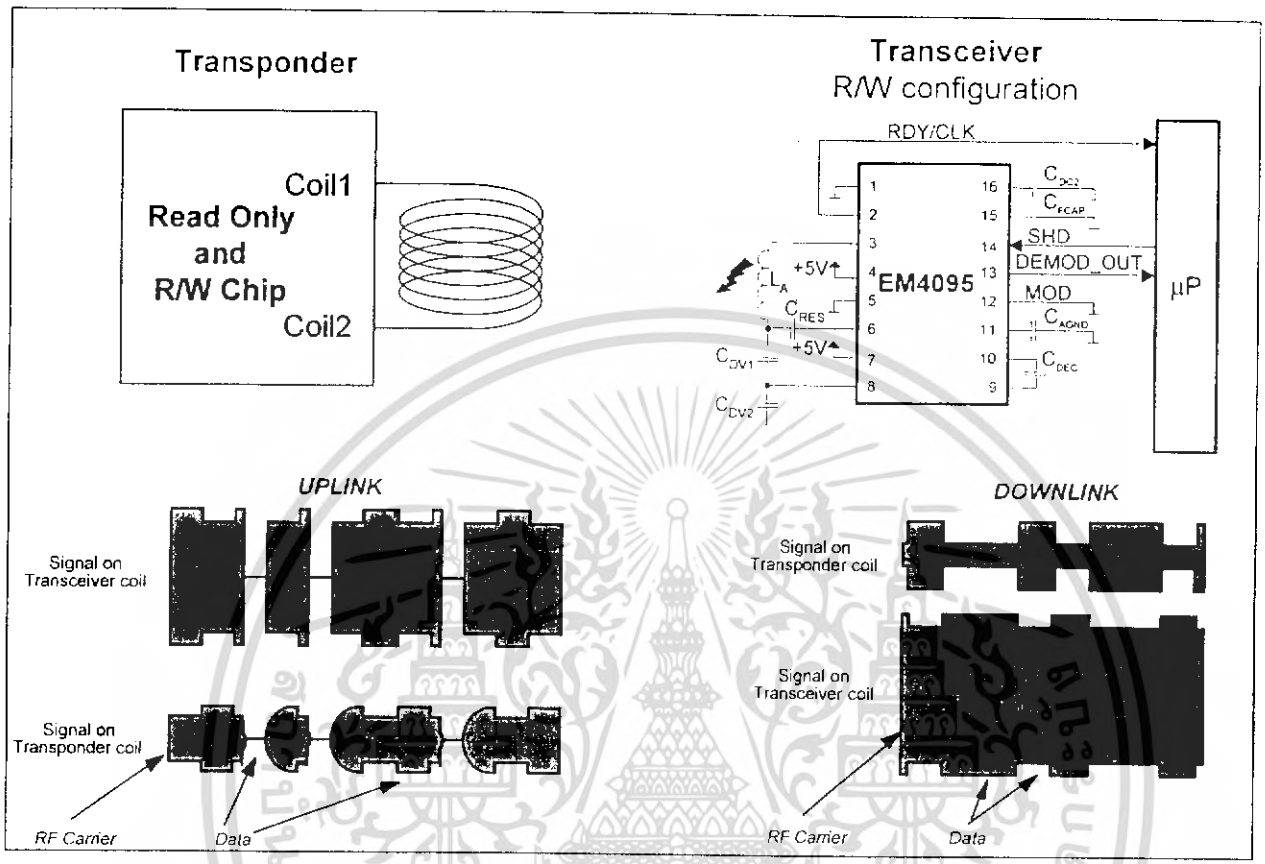


Fig. 4

Copyright © 2002, EM Microelectronic-Marin SA

Absolute Maximum Ratings

Parameter	Symbol	Conditions
Storage temperature	$T_{S...}$	-55 to +150°C
Maximum voltage at V_{DD}	V_{DDmax}	V_{SS}+6V
Minimum voltage at V_{DD}	V_{DDmin}	V_{SS} -0.3V
Max. voltage other pads	V_{MAX}	V_{DD} +0.3V
Min. voltage other pads	V_{MIN}	V_{SS} -0.3V
Max. junction temperature	T_{JMAX}	+125°C
Electrostatic discharge max. to MIL-STD-883C method 3015 against V_{SS}	V_{ESD}	4000V
Electrostatic discharge max. to MIL-STD-883C method 3015 (only for pins ANT1 and ANT2) against V_{SS}	V_{ESD_ANT}	10000V
Maximum Input/Output current on all pads except V_{DD}, V_{SS}, DV_{DD}, DV_{SS}, ANT1, ANT2, RDY/CLK	I_{IMAX} I_{OMAX}	10mA
Maximum AC peak current on ANT1 and ANT2 pads 100 kHz duty cycle 50%	I_{ANTmax}	300mA

Stresses above these listed maximum ratings may cause permanent damages to the device. Exposure beyond specified operating conditions may affect device reliability or cause malfunction.

Handling Procedures

This device has built-in protection against high static voltages or electric fields; however, anti-static precautions must be taken as for any other CMOS component. Unless otherwise specified, proper operation can only occur when all terminal voltages are kept within the voltage range. Unused inputs must always be tied to a defined logic voltage level.

Operating Conditions

Parameter	Symb	Min	Typ	Max	Units
Operating junction temperature	T_J	-40		+110	°C
Supply voltage	V_{DD}	4.1	5	5.5	V
Antenna circuit resonant frequency	F_{RES}	100	125	150	kHz
AC peak current on ANT1 & ANT2 pads	I_{ANT}			250	mA
C_{FCAP}		*	10	*	nF
C_{DEC}		*	100	*	nF
C_{DC2}		*	6.80	*	nF
C_{AGND}		100		220	nF
Package thermal resistor SO16	$R_{th\,j\text{-}a}$ **	69	70	71	°C/W

* ±10% tolerance capacitors should be used

** According to 1S2P JEDEC test board

Due to antenna driver current the internal junction temperature is higher than ambient temperature. Please calculate ambient temperature range from max. antenna current and package Thermal Resistor. It is the user's responsibility to guarantee that T_J remains below 110°C.

Supply voltage (V_{DD} and DV_{DC} pads) must be blocked by a 100nF capacitor (to V_{SS}) as close as possible to the chip

Copyright © 2002, EM Microelectronic-Marin SA

Electrical and Switching Characteristics:

Parameters specified below are valid only in case the device is used according to Operating Conditions defined on previous page.

V_{SS}=D_{VSS}=0V, V_{DD} =D_{VDD} = 5V, T_j = -40 to 110°C, unless otherwise specified

Parameter	Symbol	Test Conditions	Min	Typ	Max	Units
Supply current in sleep mode	$I_{DDstecp}$			1	2	µA
Supply current excluding drivers current	I_{DDon}			5	7	mA
AGND level	V_{AGND}	Note 1	2.35	2.5	2.65	V
Logic signals SHD, MOD, DEMOD_OUT						
Input logic high	V_{IH}		$0.8V_{DD}$			V
Input logic low	V_{IL}				$0.2V_{DD}$	V
Output logic high	V_{OH}	I_{SOURCE}=1mA	$0.9V_{DD}$			V
Output logic low	V_{OL}	I_{SINK}=1mA			$0.1V_{DD}$	V
MOD pull down resistor	R_{PD}	$0.2V_{DD}$	20	50	90	kΩ
SHD pull up resistor	R_{PU}	$0.8V_{DD}$	20	50	90	kΩ
PLL						
Antenna capture frequency range	F_{ANT_C}		100		150	kHz
Antenna locking frequency range	F_{ANT_L}		100		150	kHz
Drivers						
ANT drivers output resistance	R_{AD}	I_{ANT}=100mA		3	9	Ω
RDY/CLK driver output resistance	R_{CL}	$I_{RDY/CLK}$=10mA		12	36	Ω
AM demodulation						
DEMOD_IN common mode range	V_{CM}		V_{SS} + 0.5		V_{DD} - 0.5	V
DEMOD_IN input sensitivity	V_{sense}	Note 2		0.85	2	mVpp

Note 1: A_{GND} is a EM4095 internal reference point. Any external connection except specified capacitor to V_{SS} may lead to device malfunction.

Note 2: Modulating signal 2Khz square wave on 125 kHz carrier, total signal inside V_{CM}

Copyright © 2002, EM Microelectronic-Marin SA

Timing Characteristics:

Parameters specified below are valid only in case the device is used according to Operating Conditions defined on previous page

$V_{SS}=D_{VSS}=0V$, $V_{DD}=D_{VDD}=5V$

Parameter	Symbol	Test Conditions	Typ	Max	Units
Set-up time after a sleep period	T_{set}		25	35	ms
Time from full power to modulation state	T_{mdon}	antenna circuit specifications: Q=15.F_{RES}=125Khz modulation index: 100%		50	µs
AM demodulation: Delay time from input to output	T_{pd}	Modulating signal 2Khz square wave 10mVpp	40	100	µs
Recovery time of reception after antenna modulation	T_{rec}	Note 1	400	500	µs

Note 1: RF period is time of one period transmitted on ANT outputs (at 125 kHz 8µs). T_{rec} after antenna modulation receiver chain is ready to demodulate. The condition is of course that the amplitude on antenna has already reached its steady state by that time (this depends on Q of antenna). See also Application Notes.

Block Diagram

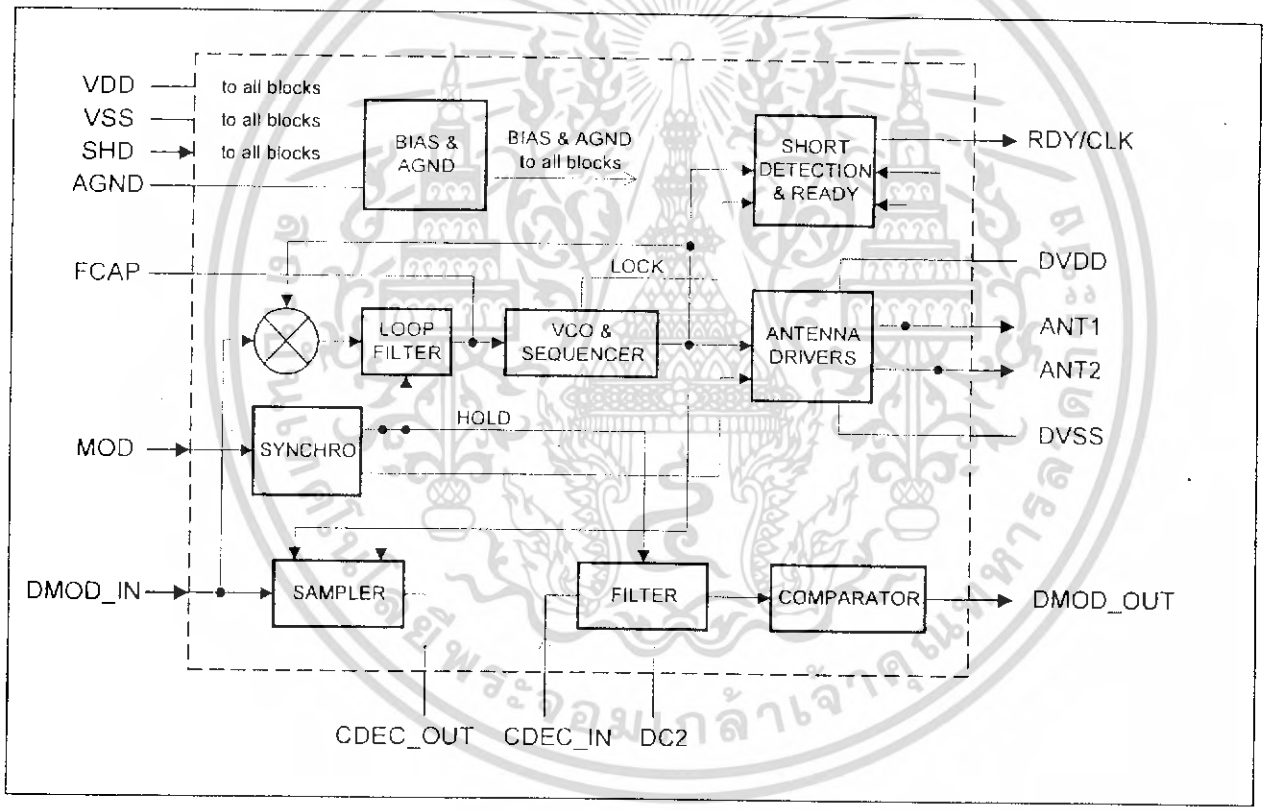


Fig. 5

Copyright © 2002, EM Microelectronic-Marin SA

Functional Description

General

The EM4095 is intended to be used with an attached antenna circuit and a microcontroller. Few external components are needed to achieve DC and RF filtering, current sensing and power supply decoupling.

A stabilised power supply has to be provided. Please refer to EM4095 Application Notes for advice.

Device operation is controlled by logic inputs SHD and MOD. When SHD is high EM4095 is in sleep mode, current consumption is minimised. At power up the input SHD has to be high to enable correct initialisation. When SHD is low the circuit is enabled to emit RF field, it starts to demodulate any amplitude modulation (AM) signal seen on the antenna. This digital signal coming from the AM demodulation block is provided through DEMOD_OUT pin to the microcontroller for decoding and processing.

High level on MOD pin forces in tri-state the main antenna drivers synchronously with the RF carrier. While MOD is high the VCO and AM demodulation chain are kept in state before the MOD went high. This ensures fast recovery after MOD is released. The switching ON of VCO and AM demodulation is delayed by 41 RF clocks after falling edge on MOD. In this way the VCO and AM demodulation operating points are not perturbed by start-up of antenna resonant circuit.

Analog Blocks

The circuit performs the two functions of an RFID basestation, namely: transmission and reception. Transmission involves antenna driving and AM modulation of the RF field. The antenna drivers deliver a current into the external antenna to generate the magnetic field.

Reception involves the AM demodulation of the antenna signal modulation induced by the transponder. This is achieved by sensing the absorption modulation applied by the tag (transponder).

Transmission

Referring to the block diagram, transmission is achieved by a Phase Locked Loop (PLL) and the antenna drivers.

Drivers

The antenna drivers supply the reader basestation antenna with the appropriate energy. They deliver current at the resonant frequency which is typically 125 kHz. Current delivered by drivers depends on Q of external resonant circuit.

It is strongly recommended that design of antenna circuit is done in a way that maximum peak current of 250 mA is never exceeded (see Typical Operating Configuration for antenna current calculation). Another limiting factor for antenna current is Thermal Convection of package. Maximum peak current should be designed in a way that internal junction temperature does not exceed maximum junction temperature at maximum application ambient temperature. 100% modulation (field stop) is done by switching OFF the drivers. The ANT drivers are protected against antenna DC short circuit to the power supplies. When a short circuit has been detected the RDY/CLK pin is pulled low while the main driver is forced in tri-state. The circuit can be restarted by activating the SHD pin.

Phase locked loop

The PLL is composed of the loop filter, the Voltage Controlled Oscillator (VCO), and the phase comparator blocks. By using an external capacitive divider, pin DEMOD_IN gets information about the actual high voltage signal on antenna.

Phase of this signal is compared with the signal driving antenna drivers. Therefore the PLL is able to lock the carrier frequency to the resonant frequency of the antenna. Depending on the antenna type the resonant frequency of the system can be anywhere in the range from 100 kHz to 150 kHz. Wherever the resonant frequency is in this range it will be maintained by the Phase Lock Loop.

Reception

The demodulation input signal for the reception block is the voltage sensed on the antenna. DEMOD_IN pin is also used as input to Reception chain. The signal level on the DEMOD_IN input must be lower than VDD-0.5V and higher than VSS+0.5V. The input level is adjusted by the use of an external capacitive divider. Additional capacitance of divider must be compensated by accordingly smaller resonant capacitor. The AM demodulation scheme is based on the "AM Synchronous Demodulation" technique.

The reception chain is composed of sample and hold, DC offset cancellation, bandpass filter and comparator. DC voltage of signal on DEMOD_IN is set to AGND by internal resistor. The AM signal is sampled, the sampling is synchronised by a clock from VCO. Any DC component is removed from this signal by the CDEC capacitor. Further filtering to remove the remaining carrier signal, high and low frequency noise is made by second order highpass filter and CDC2. The amplified and filtered receive signal is fed to asynchronous comparator. Comparator output is buffered on output pin DEMOD_OUT.

Copyright © 2002, EM Microelectronic-Marin SA

Signal RDY/CLK

This signal provides the external microprocessor with clock signal which is synchronous with the signal on ANT1 and with information about EM4095 internal state. Clock signal synchronous with ANT1 indicates that PLL is in lock and that Reception chain operation point is set. When SHD is high RDY/CLK pin is forced low. After high to low transition on SHD the PLL starts-up, and the reception chain is switched on. After time T_{SET} the PLL is locked and reception chain operation point has been established. At this moment the same signal which is being transmitted to ANT1 is also put to RDY/CLK pin indicating to microprocessor that it can start observing signal on DEMOD_OUT and giving at the same time reference clock signal. Clock on RDY/CLK pin is continuous, it is also present during time the ANT drivers are OFF due to high level on MOD pin. During the time T_{SET} from high to low transition on SHD pin RDY/CLK pin is pulled down by 100 kΩ pull down resistor. The reason for this is in additional functionality of RDY/CLK pin in case of AM modulation with index which is lower then 100%. In that case it is used as auxiliary driver which maintains lower amplitude on coil during modulation. (see also Typical Operating Configuration)

Remark: Please refer to AN4095 for external components calculation and limits.

Typical Operating Configuration

Read Only Mode

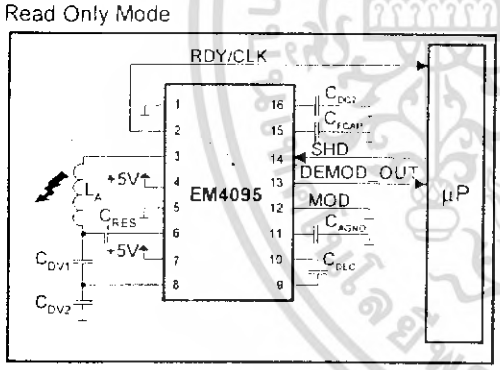


Fig. 6

Read/Write mode (Low Q factor antenna)

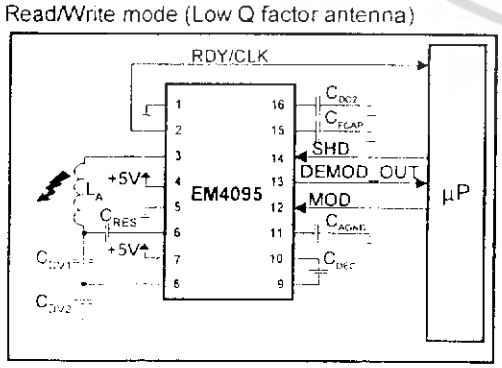


Fig. 7

Read/Write mode (High Q factor antenna)

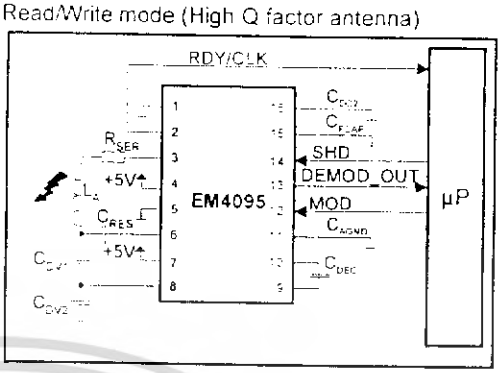


Fig. 8

Read/Write mode (AM modulation)

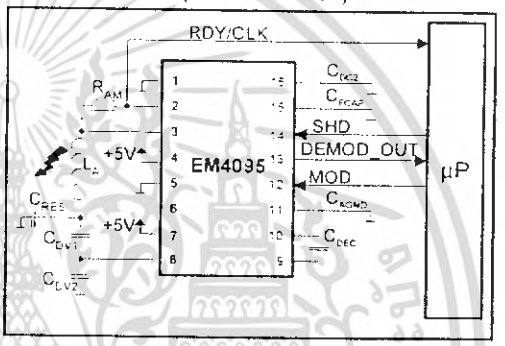


Fig. 9

Figure 6 presents EM4095 used in Read Only mode. Pin MOD is not used. It is recommended to connect it to VSS.

Figure 7 presents typical R/W configuration for OOK communication protocol reader to transponder (eg. EM4150). It is recommended to be used with low Q factor antennas (up to 15).

When the antenna quality is high using configuration of figure 6 or 7 the voltage on antenna can arrive in the range of few hundred volts and antenna peak current may exceed its maximum value. In such a case the capacitive divider ratio has to be high thus limiting the sensitivity. For such case it is better to reduce antenna circuit quality by adding serial resistor. In this way the antenna current is lower and thus power dissipation of IC is reduced with practically the same performance (Fig. 8).

In the case AM modulation communication protocol reader to transponder (eg. EM4069) is needed a single ended configuration has to be used (figure 9). When pin MOD is pulled high driver on ANT1 is put in three state, driver RDY/CLK continues driving thus maintaining lower antenna current. Modulation index is adjusted by resitor R_{AM}. As mentioned above RDY/CLK signal becomes active only after the demodulation chain operating point is set.
Before it is pulled down by high impedance pull down resistor (100 kΩ) in order not to load ANT1 output. In the case of AM modulation configuration the total antenna current change at the moment RDY/CLK pin becomes active, so external microprocessor has to wait another T_{SET} before it can start observing DEMOD_OUT.

Copyright © 2002, EM Microelectronic-Marin SA

Read Only mode with external peak detector

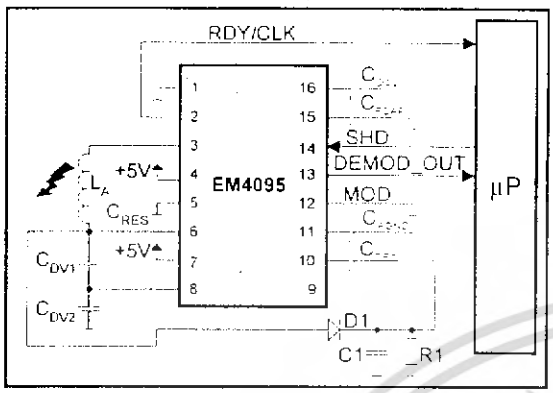


Fig. 10

Read/Write mode with external peak detector

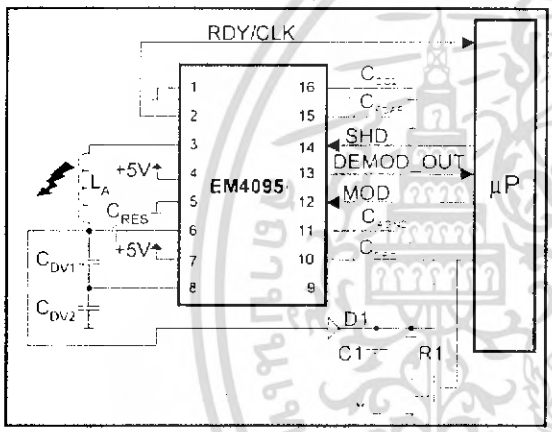


Fig. 11

As mentioned above for high Q antennas the voltage on antenna is high and read sensitivity is limited by demodulator sensitivity due to capacitive divider. Read sensitivity (and thus reading range) can be increased by using external envelope detector circuit. Input is taken on antenna high voltage side output is directly fed to CDEC_IN pin. However, the capacitor divider is still needed for PLL locking. Such configuration is shown in figure 5, the envelope detector is formed by three components: D1, R1 and C1.

The configuration presented in figure 9 may also be used for read write applications but it has a drawback in the case fast recovery of reading is needed after communication reader to transponder is finished. The reason is in fact that DC voltage after diode D1 is lost during modulation and it takes very long time before it is established again.

Figure 10 presents a solution to that problem. A high voltage NMOS transistor blocks the discharge path during modulation, so operating point is preserved. The signal controlling NMOS gate has to be put low synchronously with signal MOD, but it can be put high only after the amplitude on antenna has recovered after modulation.

PCB Layout

Refer to "EM4095 Application Note" (App. Note 404)

Copyright © 2002, EM Microelectronic-Marin SA

Pin Description

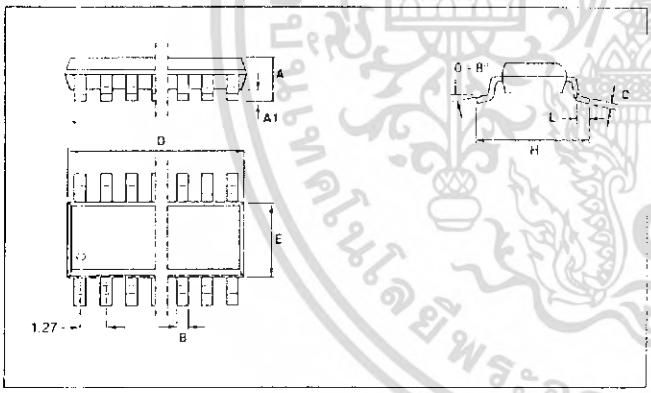
SOIC 16 package

Pin	Name	Description	Type
1	V_{SS}	Negative power supply (substrate)	GND
2	RDY/CLK	Ready flag and clock output, driver for AM modulation	O
3	ANT1	Antenna driver	O
4	D_{VDD}	Positive power supply for antenna drivers	PWR
5	D_{VSS}	Negative power supply for antenna drivers	GND
6	ANT2	Antenna driver	O
7	V_{DD}	Positive power supply	PWR
8	DEMOD_IN	Antenna sensing voltage	ANA
9	CDEC_OUT	DC blocking capacitor connection « out »	ANA
10	CDEC_IN	DC blocking capacitor connection « in »	ANA
11	A_{GND}	Analog ground	ANA
12	MOD	A High level voltage modulates the antenna	IPD
13	DEMOD_OUT	Digital signal representing the AM seen on the antenna	O
14	SHD	A High level voltage forces the circuit into sleep mode	IPU
15	FCAP	PLL Loop filter capacitor	ANA
16	DC2	DC decoupling capacitor	ANA

GND: reference ground PWR: power supply ANA: analog signal
IPD: input with internal pull down IPU: input with internal pull up O: output

Package and Ordering Information

Dimensions of SOIC 16 Package (table in millimeters)



	Common Dimensions (mm)		
Symbol	Min	Nom	Max
A	1.55	1.63	1.73
A1	0.127	0.15	0.25
B	0.35	0.41	0.49
C	0.19	0.20	0.25
D	9.80	9.93	9.98
E	3.81	3.94	3.99
H	5.84	5.99	6.20
L	0.41	0.64	0.89

Fig. 12

Ordering Information

Please make sure to give the complete part number when ordering.
The EM4095 is available in the following package:

Part Number	Package	Delivery Form
EM4095HMSO16A	SOIC 16 package	stick

Product Support

Check our Web Site under Products/RF Identification section.
Questions can be sent to cid@emmicroelectronic.com

Copyright © 2002, EM Microelectronic-Marin SA

Appendix

Equations

Antenna resonant frequency f_0:

$$f_0 = \frac{1}{2\pi\sqrt{L_A C_0}} \quad (1)$$

Where C_0 is resonant capacitor composed of C_{RES}, C_{DV1} and C_{DV2}:

$$C_0 = C_{RES} + \frac{C_{DV1} * C_{DV2}}{C_{DV1} + C_{DV2}} \quad (2)$$

Usually antenna coil is specified by its inductance (L_A) and Q factor (Q_A). Serial resistance of antenna is defined by following equation:

$$R_{ANT} = \frac{2\pi f_0 L_A}{Q_A} \quad (3)$$

The equations which follow are valid for bridge configuration as defined on Figures 1, 2 and 3. For figures 1 and 2 R_{SER} has to be considered 0.

The AC current amplitude at resonant frequency is defined as follows:

$$I_{ANT} = \frac{4}{\pi} \frac{V_{dd} - V_{ss}}{R_{ANT} + R_{SER} + 2R_{AD}} \quad (4)$$

RMS antenna current (important for power dissipation calculation):

$$I_{RMS} = \frac{I_{ANT}}{\sqrt{2}} \quad (5)$$

Peak to peak voltage on antenna is defined by following equation

$$V_{ANTpp} = \frac{I_{ANT}}{\pi f_0 C_0} \quad (6)$$

To ensure correct operation of the AM demodulation chain, the AC peak to peak voltage on DEMOD_IN pin (V_{DMOD_INpp}) has to be inside common mode range. Once peak to peak voltage on antenna is known the capacitor divider division factor can be calculated:

$$V_{DMOD_INpp} = V_{ANTpp} \frac{C_{DV1}}{C_{DV1} + C_{DV2}} \quad (7)$$

Power dissipation is composed of power dissipated on ANT drivers and internal power consumption:

$$P = 2 \cdot I_{RMS}^{\ 2} \cdot R_{AD} + I_{DDon}\left(V_{DD} - V_{SS}\right) \quad (7)$$

Temperature increase of die due to power dissipation is:

$$\Delta T = P \cdot R_{Th} \quad (8)$$

Where R_{Th} is Package thermal resistor.

EM Microelectronic-Marin SA cannot assume responsibility for use of any circuitry described other than circuitry entirely embodied in an EM Microelectronic-Marin SA product. EM Microelectronic-Marin SA reserves the right to change the circuitry and specifications without notice at any time. You are strongly urged to ensure that the information given has not been superseded by a more up-to-date version.

© EM Microelectronic-Marin SA, 07/02, Rev. E/490