สำนักหอสมุดกลาง พระจอมเกล้าลาดกระบัง

การพัฒนา READER/WRITER ของ RFID ในความถี่ 13.56 MHz

DEVELOPMENT OF READER/WRITER FOR 13.56 MHz RFID

T062538

โดย

นาย ธนรัตน์ สมบูรณ์

นาย ธีรพงษ์ แสงงาม

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ว/ท
ว152 ก
2548
ฉ.1

เลขหมู่.....

เลขทะเบียน.....62538

วัน,เดือน,ปี 19 ส.ค. 2549

.b.....11625958.....

.i.....

ปริญญานิพนธ์เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต

สาขาวิชาอิเล็กทรอนิกส์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2548

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การพัฒนา READER/WRITER ของ RFID ในความถี่ 13.56 MHz

DEVELOPMENT OF READER/WRITER FOR 13.56 MHz RFID

โดย

นาย ธนรัตน์ สมบูรณ์

นาย ธีรพงษ์ แสงงาม

อาจารย์ที่ปรึกษา

รศ.ดร.อิทธิชัย อรุณศรีแสงไชย

ปริญญานิพนธ์สำหรับปริญญาวิศวกรรมศาสตร์บัณฑิต

สาขาวิชาอิเล็กทรอนิกส์

คณะวิศวกรรมศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2548

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ปริญญานิพนธ์ ปีการศึกษา 2548

ภาควิชาอิเล็กทรอนิกส์

คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เรื่อง การพัฒนา READER/WRITER ของ RFID ในความถี่ 13.56 MHz

ผู้จัดทำ

1. นาย ธนรัตน์ สมบูรณ์
2. นาย ธีรพงษ์ แสงงาม

.....อาจารย์ที่ปรึกษา

(รศ.ดร.อิทธิชัย อรุณศรีแสงไชย)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การพัฒนา READER/WRITER ของ RFID ในความถี่ 13.56 MHz

DEVELOPMENT OF READER/WRITER FOR 13.56 MHz RFID

1. นาย ธนรัตน์ สมบูรณ์

2. นาย ธีรพงษ์ แสงงาม

โครงงานได้รับการตรวจสอบแล้ว พร้อมที่จะทำการตรวจสอบได้

.....

(รศ.ดร.อิทธิชัย อรุณศรีแสงไชย)

อาจารย์ที่ปรึกษา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การพัฒนา READER/WRITER ของ RFID สำหรับความถี่ 13.56 MHz

DEVELOPMENT OF READER/WRITER FOR 13.56 MHz RFID

นาย ธนรัตน์ สมบูรณ์ รหัส 46015219
นาย ธีรพงษ์ แสงงาม รหัส 46015225
รศ.ดร.อิทธิชัย อรุณศรีแสงไชย อาจารย์ที่ปรึกษา
ปีการศึกษาที่ 2548

บทคัดย่อ

โครงงานนี้เป็นการนำเสนอในส่วนของการพัฒนา และการสร้างเครื่องอ่าน-เขียน RFID (Radio Frequency Identification) ในความถี่ 13.56 MHz การพัฒนามีอยู่ 2 ส่วนใหญ่ คือ วงจรของเครื่องอ่าน-เขียนของ RFID และโปรแกรมที่ใช้ควบคุมการทำงานของเครื่องอ่าน-เขียน RFID การทำงานของเครื่องอ่าน-เขียน RFID นี้ถูกพัฒนา เพื่อใช้งานกับระบบเก็บเงินค่าจอดรถ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Development of Reader / Writer for 13.56 MHz RFID

Thanarat Somboon 46015219

Teerapone Sangngam 46015225

Assoc. Prof.Dr. Itthichai Arungsrisangchai Advisor

Education year 2005

Abstract

This project present the development and construction of a set of Reader / Writer terminal for 13.56 MHz RFID tag. The development consists of 2 main parts ,which are hardware and software part. The software part developes under the application for car park charging system.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ

บทที่1 บทนำ 1.

1.1 ความเป็นมาและความสำคัญ 1.

1.2 การประยุกต์ใช้ RFID ในปัจจุบัน 1.

1.3 วัตถุประสงค์ของโครงงาน 1.

1.4 ขอบเขตของโครงงาน 2.

บทที่ 2 ทฤษฎี 3.

2.1 ทฤษฎีของRFID 3.

2.1.1 การสื่อสารแบบไร้สาย 3.

2.1.1.1 ความถี่ของคลื่นพาหะ 4.

2.1.1.2 อัตราการรับส่งข้อมูลและแบนด์วิดธ์ 4.

2.1.1.3 ระยะการรับส่งข้อมูลและกำลังส่ง 5.

2.1.2 องค์ประกอบของระบบ RFID 5.

2.1.2.1 Tags หรือ Transponders 5.

2.1.2.2 Reader หรือ Interrogator 6.

2.1.3 หลักการทำงานเบื้องต้นของระบบ 7.

2.2 ทฤษฎีไมโครคอนโทรลเลอร์ 7.

2.2.1 คุณสมบัติของไมโครคอนโทรลเลอร์ตระกูล MCS-51 8.

2.2.2 การจัดขาของไมโครคอนโทรลเลอร์ตระกูล MCS – 51 8.

2.2.3 วิธีการเข้าถึงข้อมูล 11.

2.2.4 การเชื่อมต่อพอร์ตอนุกรมของคอมพิวเตอร์ 11.

2.2.5 ความรู้เบื้องต้นเกี่ยวกับ I²C 12.

2.2.5.1 คุณสมบัสโดยทั่วไปของบัส I²C 13.

2.2.5.2 หลักการของบัสI²C 14.

2.2.5.3 สภาวะที่เกิดขึ้นบนบัสI²C 15.

2.2.5.4 การทำงานบนบัสI²C 16.

2.2.5.5 การอ้างถึงแบบ 7 บิต(7-bit addressing) 16.

2.2.5.6 การอ้างถึงแบบ 10 บิต 16.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.3 ความรู้เบื้องต้นเกี่ยวกับ Visual Basic 17.
2.3.1 ชนิดของตัวแปรใน Visual Basic 17.
2.3.2 ตัวดำเนินการ, นิพจน์ และประโยคคำสั่ง 18.
2.3.3ลำดับการประมวลผลของตัวดำเนินการ 21.
2.3.4 คำสั่งต่างๆที่ใช้งาน 22.
2.3.5 การจัดการ String และฟังก์ชันที่เกี่ยวข้อง 25.
2.3.6 Timer Control และ ฟังก์ชันเกี่ยวกับเรา 26.
2.4 วงจรขยายคลาส E 31.
2.5 การMATCHING CIRCUIT 33.
2.6 สายอากาศ และ การหาค่า Inductance 34.
2.7 วงจรภาครับ 35.

บทที่ 3 ขั้นตอนการออกแบบ 40.
3.1 การออกแบบบล็อกคอนโทรล 41.
3.2 การคำนวณหาค่าอุปกรณ์ในวงจร Class E Amplifier 41.
3.3 การคำนวณหาค่าอุปกรณ์ในวงจรสายอากาศ 41.
3.4 การคำนวณหาค่า Inductance ในสายอากาศ 41.
3.5 วงจรทั้งหมดที่ได้ในการออกแบบ 42.
3.6 โปรแกรมควบคุมการทำงาน 43.
3.6.1 ส่วนที่ใช้ติดต่อกับผู้ใช้งาน 44.
3.6.2 ส่วนที่ใช้ควบคุมเครื่อง Reader and Writer 50.

บทที่ 4 ผลการทดลอง 69.
4.1 ผลการทดลองการติดต่อระหว่างคอมพิวเตอร์กับไมโครคอนโทรลเลอร์ 69.
4.2 การทดลองทางภาคส่ง Class E Power Amplifier 72.
4.3 การทดลองทางภาครับ 86.

บทที่ 5 สรุปผลการทดลอง 99.
5.1 สรุปผลการการทดลอง 99.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูป

รูปที่ 2.1 แสดงระบบ RFID 3.

รูปที่ 2.2 แสดงการสื่อสารระหว่างแท็กและตัวรับข้อมูล 4.

รูปที่ 2.3 แสดงตำแหน่งขาของไมโครคอนโทรเลอร์ MCS51 9.

รูปที่ 2.4 วงจรเชื่อมต่อ MCS 51เข้ากับพอร์ตอนุกรมของควมพิวเตอร์และ
ไมโครคอนโทรลเลอร์ 12.

รูปที่ 2.5 ผังแสดงการเชื่อมต่อของอุปกรณ์ต่างๆบนระบบบัส 12.

รูปที่ 2.6 การตัวต้านทาน Rpพื่อลดสัญญารรบกวนขนาดใหญ่ที่อาจเข้ามาในบัส I^2C 13.

รูปที่ 2.7 ไดอะแกรมเวลาแสดงสถานะต่างในบัส I^2C 15.

รูปที่ 2.8 Flow chart IF-THEN-ELSE 23.

รูปที่ 2.9 Flow chart IF-THEN-ELSE-END IF 24.

รูปที่ 2.10 Flow chart IF-THEN-ELSEIF 25.

รูปที่ 2.11 วงจรขยายคลาส E 31.

รูปที่ 2.12 ทรานซิสเทอร์ โวล์ทเตจ และกระแส 31.

รูปที่ 2.13 MATCHING CIRCUIT 33.

รูปที่ 2.14 SERIES TO PARALLAL CIRCUIT 34.

รูปที่ 2.15 Planar Square, Hexagonal and Octagonal Spiral Coils 34.

รูปที่ 2.16 วงจรภาครับ 35.

รูปที่ 2.17 สัญญาณที่รับมาจากสายอากาศที่ถูกตัดซีกลบออก 36.

รูปที่ 2.18 สัญญาณในช่วงฐานของความถี่ 13.56 MHz. 36.

รูปที่ 2.19 ปรับสัญญาณให้อยู่ระดับ Vcc/2 37.

รูปที่ 2.20 สัญญาณจะถูกทำให้อยู่ที่จุด O V 37.

รูปที่ 2.21 สัญญาณจะถูกขายาเพิ่มขึ้นด้วบ OP AMP 38.

รูปที่ 2.22 สัญญาณด้านลบจะถูกตัดออก 38.

รูปที่ 2.23 สัญญาณจะถูกยกไปที่ Vcc/2 39.

รูปที่ 3.1 บล๊อกไดแกรมของระบบ RFID 40.

รูปที่ 3.2 วงจรภาคคอนโทรล 42.

รูปที่ 3.3 วงจร ภาคส่ง ภาครับ และวงจร Matching 43.

รูปที่ 3.4 Flowchart การทำของโปรแกรมที่ที่ติดต่อกับผู้ใช้งาน 44.

รูปที่ 3.5 Flowchart ควบคุมการทำงานเครื่อง Reader and Writer 50.

รูปที่ 3.6 (ต่อ) Flowchart ควบคุมการทำงานเครื่อง Reader and Writer 51.

รูปที่ 4.1 วงจรภาค Control 69.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 4.2 ทดลองวงจรภาค Control 69.
รูปที่ 4.3 ผลการทดลองในการติดต่อระหว่างคอมพิวเตอร์กับไมโครคอนโทรลเลอร์ 70.
รูปที่ 4.4 ผลการทดลองไดอะแกรมเวลาแสดงสถานะการกดคีย์ 0 71.
รูปที่ 4.5 ผลการทดลองไดอะแกรมเวลาแสดงสถานะการกดคีย์ 1 71.
รูปที่ 4.6 วงจรภาคส่ง Class E Power Amplifier 72.
รูปที่ 4.7 แสดงวงจรในส่วนภาคส่ง Class E Power Amplifier 72.
รูปที่ 4.8 แสดงการวัดที่จุดL2และขาเบสของQในภาคส่ง 73.
รูปที่ 4.9 แสดงสัญญาณ ที่ตกคร่อมโหลด, ความถี่ 12.5 MHz 74.
รูปที่ 4.10 แสดงสัญญาณ ที่ตกคร่อมโหลด, ความถี่ 12.7 MHz 74.
รูปที่ 4.11 แสดงสัญญาณ ที่ตกคร่อมโหลด, ความถี่ 12.9 MHz 75.
รูปที่ 4.12 แสดงสัญญาณ ที่ตกคร่อมโหลด, ความถี่ 13.1 MHz 75.
รูปที่ 4.13 แสดงสัญญาณ ที่ตกคร่อมโหลด, ความถี่ 13.3 MHz 76.
รูปที่ 4.14 แสดงสัญญาณ ที่ตกคร่อมโหลด, ความถี่ 13.5 MHz 76.
รูปที่ 4.15 แสดงสัญญาณ ที่ตกคร่อมโหลด, ความถี่ 13.7 MHz 77.
รูปที่ 4.16 แสดงสัญญาณ ที่ตกคร่อมโหลด, ความถี่ 13.9 MHz 77.
รูปที่ 4.17 แสดงสัญญาณ ที่ตกคร่อมโหลด, ความถี่ 14.1 MHz 78.
รูปที่ 4.18 แสดงสัญญาณ ที่ตกคร่อมโหลด, ความถี่ 14.3 MHz 78.
รูปที่ 4.19 แสดงสัญญาณ ที่ตกคร่อมโหลด, ความถี่ 14.5 MHz 79.
รูปที่ 4.20 แสดงจุดวัดกำลังงานที่ใช้ตัวต้านทานเป็นโหลด 81.
รูปที่ 4.21 แสดงการวัดกำลังงานที่ใช้ตัวต้านทานเป็นโหลด 82.
รูปที่ 4.22 แสดงสัญญาณที่วัดตกคร่อมโหลดใช้ตัวต้านทานเป็นโหลด 82.
รูปที่ 4.23 แสดงการวัดกำลังงานที่ใช้เครื่อง Radio Test Set เป็นโหลด 83.
รูปที่ 4.24 แสดงผลลัพธ์ที่ได้จากการวัดหาค่ากำลังงาน 84.
รูปที่ 4.25 แสดงสัญญาณ ASK ที่วัดตกคร่อมโหลดและสัญญาณอินพุท 85.
รูปที่ 4.26 แสดงวงจรภาครับ 86.
รูปที่ 4.27 แสดงการวัดที่จุดC32 ในภาครับ 87.
รูปที่ 4.28 แสดงสัญญาณที่จุด C32 ในภาครับ 88.
รูปที่ 4.29 แสดงการวัดขา K ของ D1 ในภาครับ 88.
รูปที่ 4.30 แสดงสัญญาณที่ขา K ของ D1 ในภาครับ 89.
รูปที่ 4.31 แสดงการวัดขา K ของ D2 ในภาครับ 89.
รูปที่ 4.32 แสดงสัญญาณที่ขา K ของ D2 ในภาครับ 90.
รูปที่ 4.33 แสดงสัญญาณที่ขา A ของ D3 ในภาครับ 90.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 4.34 แสดงสัญญาณที่ขา 3 ของ AD8062 ในภาครับ 91.
รูปที่ 4.35 แสดงการวัดขา L2 ในภาครับ 91.
รูปที่ 4.36 แสดงสัญญาณที่ขา L2 ในภาครับ 92.
รูปที่ 4.37 แสดงสัญญาณที่ขา K ของ D4 ในภาครับ 92.
รูปที่ 4.38 แสดงการวัดสัญญาณที่ RFIN ในภาครับ 93.
รูปที่ 4.39 แสดงสัญญาณที่ RFIN ในภาครับ 93.
รูปที่ 4.40 แสดงสัญญาณที่จุด C32 ในภาครับที่เป็นสัญญาณแบบ ASK 94.
รูปที่ 4.41 แสดงสัญญาณที่ขา K ของ D1 ในภาครับที่เป็นสัญญาณแบบ ASK 95.
รูปที่ 4.42 แสดงสัญญาณที่ขา K ของ D2 ในภาครับที่เป็นสัญญาณแบบ ASK 95.
รูปที่ 4.43 แสดงสัญญาณที่ขา A ของ D3 ในภาครับที่เป็นสัญญาณแบบ ASK 96.
รูปที่ 4.44 แสดงสัญญาณที่ขา 1 ของ AD8062 ในภาครับที่เป็นสัญญาณแบบ ASK 96.
รูปที่ 4.45 แสดงสัญญาณที่ขา L2 ในภาครับที่เป็นสัญญาณแบบ ASK 97.
รูปที่ 4.46แสดงสัญญาณที่ขา K ของ D4 ในภาครับที่เป็นสัญญาณแบบ ASK 97.
รูปที่ 4.47 แสดงสัญญาณที่ RFIN ในภาครับที่เป็นสัญญาณแบบ ASK 98.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญตาราง

ตารางที่ 2.1 ตัวแปรชนิดต่างๆ 18.

ตารางที่ 2.2 ตัวดำเนินการเปรียบเทียบ 19.

ตารางที่ 2.3 Logical Operator “and” 19.

ตารางที่ 2.4 Logical Operator “or” 20.

ตารางที่ 2.5 Logical Operator “xor” 20.

ตารางที่ 2.6 Logical Operator “not” 20.

ตารางที่ 2.7 Logical Operator “eqv” 21.

ตารางที่ 2.8 Logical Operator “imp” 21.

ตารางที่ 2.9 ลำดับการประมวลผลของตัวดำเนินการ 22.

ตารางที่ 2.10 ข้อมูลชนิดวันที่และเวลา 29.

ตารางที่ 2.11 ข้อมูลชนิดตัวเลข 29.

ตารางที่ 2.12 ข้อมูลชนิดตัวเลขโดยใช้ Format เป็นข้อความ 30.

ตารางที่ 4.1 ผลการทดลองวงจรภาคส่ง 79.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญกราฟ

กราฟที่ 4.1 แรงดันที่ขา L7 / ความถี่ 80.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่1
บทนำ

1.1 ความเป็นมาและความสำคัญ

โครงงานนี้เป็นการศึกษากระบวนการทำงานของเครื่องอ่านและเขียน (READER/WRITER) บัตร (Tag) ของ RFID (Radio Frequency Identification) ในย่านความถี่ 13.56 MHz ซึ่งสามารถนำไปใช้งานได้อย่างกว้างขวาง โดยปกติแล้ว ในปัจจุบัน การใช้บัตรอัจฉริยะ (Smart Card) และระบบตรวจสอบรหัสโดยใช้ความถี่วิทยุ (RFID) เป็นที่ยอมรับอย่างสูงว่า เป็นเทคโนโลยีที่เอื้ออำนวยต่อการใช้งานที่ต้องการการบ่งบอกความแตกต่างหรือข้อมูลจำเพาะของแต่ละบุคคล ที่สามารถทำงานได้ถูกต้องแม่นยำ รวดเร็ว และมีความเป็นอัตโนมัติกว่าระบบตรวจสอบรหัสในระบบอื่นๆ เช่น รหัสแบบแท่ง (Barcode) การใช้งานที่ง่ายและยังเพิ่มขีดความสามารถในการให้บริการเสริมในเชิงพาณิชย์ด้านต่างๆ อีกทั้งยังสอดคล้องกับเทคโนโลยีทางการเก็บข้อมูลคอมพิวเตอร์ ยังผลให้การขยายตัวของการใช้งาน RFID/Smart Card สูงขึ้นอย่างก้าวกระโดด

1.2 การประยุกต์ใช้ RFID ในปัจจุบัน

ปัจจุบัน มีการประยุกต์ใช้ RFID Tag กันอย่างแพร่หลาย โดยนำมาใช้แทนระบบบาร์โค้ดแบบเดิม เนื่องจาก RFID มีความสะดวกสบายในการใช้งานมากกว่า เนื่องจาก ไม่จำเป็นต้องนำวัตถุมาอ่านด้วยเครื่องอ่านบาร์โค้ด เพียงแค่นำวัตถุที่ติด Tag ไปผ่านบริเวณที่มีเครื่องอ่านสัญญาณก็จะสามารถอ่านค่าได้ทันที ทำให้สามารถให้บริการได้อย่างรวดเร็วมากขึ้น ตัวอย่างการประยุกต์ใช้ RFID ในปัจจุบัน ได้แก่

- การใช้ระบุเอกลักษณ์บุคคล ในงานด้านการรักษาความปลอดภัยของอาคาร โดยจะฝัง Tag ในรูปบัตรของบุคคลที่มีสิทธิในการเข้าออกพื้นที่ต่างๆ ของอาคาร ซึ่งเครื่องอ่านสัญญาณจาก Tag ที่ฝังไว้จะประมวลผลได้ว่า จะอนุญาตให้บุคคลนั้น ผ่านเข้าไปในบริเวณที่กำหนดไว้ได้หรือไม่
- การใช้งานในด้านการจราจร ขนส่ง มีการประยุกต์ใช้ระบบ RFID กับการคำนวณค่าขึ้นลงทางด่วน โดยรถที่ต้องการใช้ทางด่วน จะติด Tag ไว้บริเวณกระจกหน้า เมื่อรถแล่นผ่านเครื่องอ่านสัญญาณบริเวณทางขึ้นและทางลง เครื่องจะทำการคำนวณค่าใช้จ่ายให้โดยอัตโนมัติ

ในโครงงานนี้เน้นการศึกษาในการทดลองใช้งานและการควบคุมการทำงานโดยอาศัยไมโครคอรโทรลเลอร์ควบคุมการทำงานไปยังชิป CRX 14 ในเครื่องอ่านและเขียนผ่านทางพอร์ต RS 232 เพื่อควบคุมการทำงานของเครื่องอ่านและเขียน

1.3 วัตถุประสงค์ของโครงงาน

- ศึกษากระบวนการทำงานของเครื่องรับ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- สามารถนำระบบ RFID ไปประยุกค์ให้งานได้

1.4 ขอบเขตของโครงงาน

- พัฒนาเครื่องอ่านและเขียนที่ใช้ในย่านความถี่ 13.56 MHz

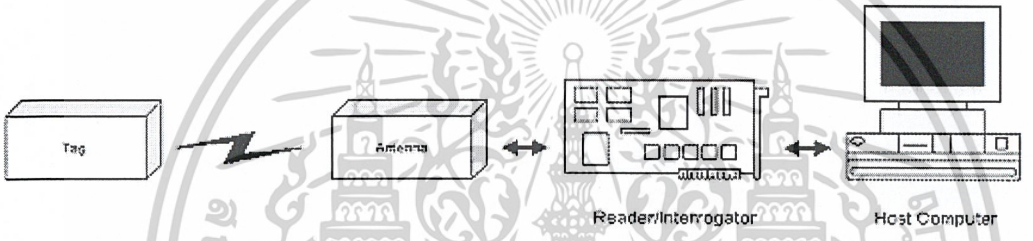
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 2

ทฤษฎี

2.1 ทฤษฎีของRFID

RFID ย่อมาจากคำว่า Radio Frequency Identification เป็นระบบที่นำเอาคลื่นวิทยุมาเป็นคลื่นพาหะเพื่อใช้ในการสื่อสารข้อมูลระหว่างอุปกรณ์สองชนิดที่เรียกว่า แท็ก (Tag) และตัวอ่านข้อมูล (Reader หรือ Interrogator) ซึ่งเป็นการสื่อสารแบบไร้สาย (Wireless) โดยการนำข้อมูลที่ต้องการส่ง มาทำการมอดูเลต (Modulation) กับคลื่นวิทยุแล้วส่งออกผ่านทางสายอากาศที่อยู่ในตัวรับข้อมูล ดังแผนผังการทำงานของระบบ RFID ในรูปที่ 2.1

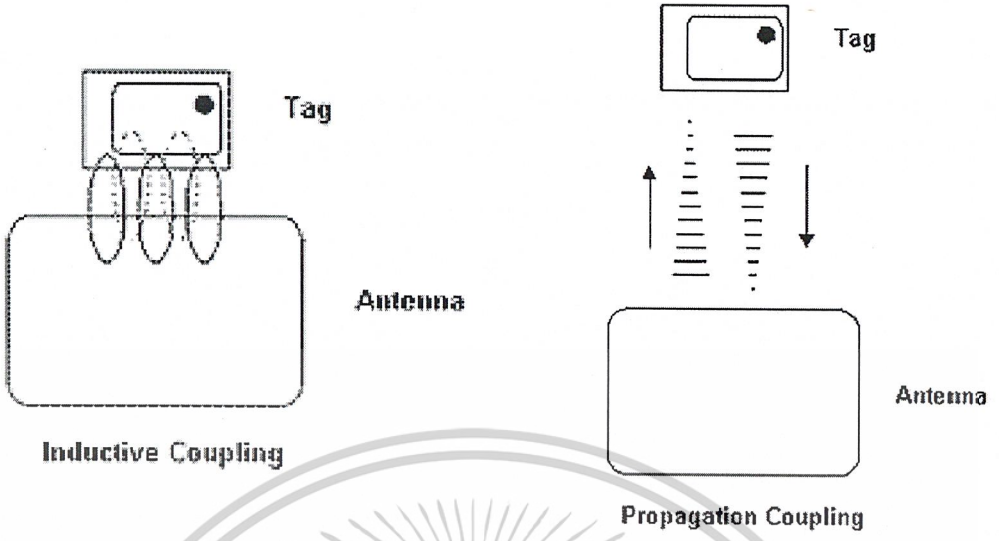


รูปที่ 2.1 แสดงระบบ RFID

2.1.1 การสื่อสารแบบไร้สาย

การสื่อสารข้อมูลของระบบ RFID คือระหว่างแท็กและตัวอ่านข้อมูล (Reader หรือ Interrogator) จะสื่อสารแบบไร้สายผ่านอากาศ โดยจะนำข้อมูลมาทำการมอดูเลต (Modulation) กับคลื่นพาหะที่เป็นคลื่นความถี่วิทยุโดยมีสายอากาศ (Antenna) ที่อยู่ในตัวอ่านข้อมูลเป็นตัวรับและส่งคลื่นซึ่งแบ่งออกเป็น 2 วิธีด้วยกันคือ วิธีเหนี่ยวนำคลื่นแม่เหล็กไฟฟ้า (Inductive Coupling หรือ Proximity Electromagnetic) กับ วิธีการแผ่คลื่นแม่เหล็กไฟฟ้า (Electromagnetic Propagation Coupling) ดังรูปที่ 2 เทคนิคการมอดูเลตข้อมูลเข้ากับคลื่นพาหะก็มีด้วยกันหลายวิธี เช่น ASK (Amplitude Shift Keying), FSK (Frequency Shift Keying) หรือ PSK (Phase Shift Keying) ซึ่งขึ้นอยู่กับผู้ออกแบบจะเลือกให้มีความเหมาะสมกับการใช้งานแต่ละประเภท

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.2 แสดงการสื่อสารระหว่างแท็กและตัวรับข้อมูล

2.1.1.1 ความถี่ของคลื่นพาหะ

ในปัจจุบันได้มีการรวมกลุ่มระหว่างแต่ละประเทศ เพื่อทำการกำหนดมาตรฐานความถี่คลื่นพาหะของระบบ RFID โดยมีสามกลุ่มใหญ่ๆ คือ กลุ่มประเทศในยุโรปและอาฟริกา (Region 1), กลุ่มประเทศอเมริกาเหนือและอเมริกาใต้ (Region 2) และสุดท้ายคือกลุ่มประเทศตะวันออกไกลและออสเตรเลีย (Region 3) ซึ่งแต่ละกลุ่มประเทศจะกำหนดแนวทางในการเลือกใช้ความถี่ต่างๆให้แก่บรรดาประเทศสมาชิก อย่างไรก็ตาม ความถี่ของคลื่นพาหะที่นิยมใช้งานในย่านความถี่ต่ำ ย่านความถี่ปานกลาง และย่านความถี่สูงก็คือ 125 kHz, 13.56 MHz และ 2.45 GHz นอกจากนี้รัฐบาลของแต่ละประเทศ โดยทั่วไปจะมีการออกกฎหมายเกี่ยวกับระเบียบการใช้งานย่านความถี่ต่างๆ รวมถึงกำลังส่งของระบบ RFID ด้วย

2.1.1.2 อัตราการรับส่งข้อมูลและแบนด์วิดธ์

อัตราการรับส่งข้อมูล (Data Transfer Rate) จะขึ้นอยู่กับความถี่ของคลื่นพาหะ โดยปกติถ้าความถี่ของคลื่นพาหะยิ่งสูง อัตราการรับส่งข้อมูลก็จะยิ่งสูงตามไปด้วย ส่วนการเลือกแบนด์วิดธ์ หรือย่านความถี่นั้นก็จะมีผลต่ออัตราการรับส่งข้อมูลเช่นกันโดยมีหลักว่า แบนด์วิดธ์ควรจะมีค่ามากกว่าอัตราการรับส่งข้อมูลที่ต้องการอย่างน้อยสองเท่า ยกตัวอย่างเช่น ถ้าใช้แบนด์วิดธ์ในช่วง 2.4-2.5 GHz ก็จะสามารถรองรับอัตราการรับส่งข้อมูลได้ถึงประมาณ 2 megabits ต่อวินาที เป็นต้น แต่การใช้แบนด์วิดธ์ที่กว้างเกินไปก็อาจทำให้เกิดปัญหาเกี่ยวกับสัญญาณรบกวนมาก หรือทำให้ S/N Ratio ต่ำลงนั่นเอง ดังนั้นการเลือกใช้แบนด์วิดธ์ให้ถูกต้องก็เป็นส่วนสำคัญในการพิจารณา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.1.1.3 ระยะห่างของการรับส่งข้อมูลและกำลังส่ง

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

ความเข้มของคลื่นแม่เหล็กไฟฟ้า โดยทั่วไปจะลดลงตามระยะทางโดยแปรผกผันกับระยะทางยกกำลังสอง แต่ในบางสภาพแวดล้อมซึ่งอาจมีการสะท้อนกลับของคลื่นแม่เหล็กไฟฟ้าจากสิ่งต่างๆ รอบตัว เช่น โลหะ ก็อาจทำให้ความเข้มของคลื่นแม่เหล็กไฟฟ้าลดลงอย่างรวดเร็ว โดยอาจแปรผกผันกับระยะทางยกกำลังสี่ ปรากฏการณ์เช่นนี้เราเรียกว่า "Multi-path Attenuation" ซึ่งจะส่งผลให้ระยะการรับส่งข้อมูลสั้นลง หรือแม้กระทั่งความชื้นในอากาศก็อาจมีผลในกรณีที่ความถี่สูงๆ ดังนั้นการนำระบบ RFID ไปใช้งานก็ควรมีการคำนึงถึงสภาพแวดล้อม เพราะจะมีผลกระทบกับระยะการรับส่งข้อมูล และพยายามติดตั้งระบบให้ห่างไกลจากโลหะ ซึ่งอาจทำให้เกิดการสะท้อนของคลื่นแม่เหล็กไฟฟ้าได้

กำลังส่งของแท็กที่จะส่งกลับมายังตัวอ่านข้อมูลนั้น โดยทั่วไปจะมีกำลังที่ต่ำมากเมื่อเทียบกับกำลังส่งของ ตัวอ่านข้อมูล ดังนั้นความไวในการตรวจจับสัญญาณของตัวอ่านข้อมูล ก็เป็นอีกจุดหนึ่งที่ต้องพิจารณา

ถึงแม้ในทางเทคนิคเราจะสามารถทำให้ตัวอ่านข้อมูลมีกำลังส่งมากแค่ไหนก็ได้ แต่โดยทั่วไปก็จะถูกจำกัดโดยกฎหมายของแต่ละประเทศ เช่นเดียวกับความถี่ ดังนั้นในระบบ RFID โดยทั่วๆไปจะมีกำลังส่งเพียงระหว่าง 100 -500 mW

2.1.2 องค์ประกอบของระบบ RFID

2.1.2.1 Tags หรือ Transponders

แท็ก (Tag) นั้นเรียกอีกชื่อหนึ่งว่าทรานสปอนเดอร์ (Transponder) มาจากคำว่าทรานสมิตเตอร์ (Transmitter) ผสมกับคำว่าเรสปอนเดอร์ (Responder) นั่นเอง ถ้าจะแปลให้ตรงตามศัพท์ แท็กก็จะทำหน้าที่ส่งสัญญาณหรือข้อมูลที่บันทึกอยู่ในแท็กตอบสนองไปที่ตัวอ่านข้อมูล การสื่อสารระหว่างแท็กและตัวอ่านข้อมูลจะเป็นแบบไร้สายผ่านอากาศ ภายในแท็กจะประกอบไปด้วย ชิปสารกึ่งตัวนำ (Semiconductor Chip) ซึ่งเชื่อมต่ออยู่กับสายอากาศ

แท็กอาจมีรูปร่างได้หลายแบบขึ้นอยู่กับการนำไปใช้งาน โดยอาจมีรูปร่างเหมือนบัตรเครดิตในการใช้งานทั่วไป หรือเล็กขนาดไส้ดินสอยาวเพียง 10 มิลลิเมตร เพื่อฝังเข้าไปใต้ผิวหนังสัตว์ในกรณีนำไปใช้ในงานปศุสัตว์ หรืออาจมีขนาดใหญ่มากสำหรับแท็กที่ใช้ติดกับเครื่องจักรขณะทำการขนส่ง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

แท็กอาจนำไปติดไว้กับสินค้าในร้านค้าปลีกทั่วไปเพื่อป้องกันขโมย โดยจะมีการติดตั้งสายอากาศของตัวอ่านข้อมูลขนาดใหญ่ไว้ตรงประตูทางออกเพื่อทำการตรวจจับขโมย

ชิปที่อยู่ในแท็กจะมีหน่วยความจำซึ่งอาจเป็นแบบอ่านได้อย่างเดียว (ROM) หรือทั้งอ่านทั้งเขียน (RAM) ทั้งนี้ขึ้นอยู่กับความต้องการในการใช้งาน โดยปกติหน่วยความจำแบบ ROM จะใช้เก็บข้อมูลเกี่ยวกับการรักษาความปลอดภัย เช่นข้อมูลของบุคคลที่มีสิทธิผ่านเข้าออกในบริเวณที่มีการควบคุมหรือระบบปฏิบัติการ ในขณะที่ RAM จะใช้เก็บข้อมูลชั่วคราวในระหว่างที่แท็กและตัวอ่านข้อมูลทำการติดต่อสื่อสารกัน

นอกจากนี้อาจมีการนำหน่วยความจำแบบ EEPROM มาใช้ในกรณีต้องการเก็บข้อมูลในระหว่างที่แท็กและตัวอ่านข้อมูลทำการสื่อสาร และข้อมูลยังคงอยู่ถึงแม้จะไม่มีพลังงานไฟฟ้าป้อนให้แก่แท็ก เราสามารถแบ่งชนิดของแท็กออกเป็น 2 ชนิดคือ

- แท็กชนิดแอ็กตีฟ (Active Tag) แท็กชนิดนี้จะมีแบตเตอรี่อยู่ภายใน เพื่อป้อนพลังงานไฟฟ้าให้แท็กทำงานโดยปกติ เราจะสามารถทั้งอ่านและเขียนข้อมูลลงในแท็กชนิดนี้ได้ และการที่ต้องใช้แบตเตอรี่จึงทำให้แท็กชนิดแอ็กตีฟมีอายุการใช้งานจำกัดตามอายุของแบตเตอรี่ เมื่อแบตเตอรี่หมดก็ต้องนำแท็กไปทิ้งไม่สามารถนำกลับมาใช้ใหม่ได้ เนื่องจากจะมีการซีล (seal) ที่ตัวแท็กจึงไม่สามารถเปลี่ยนแบตเตอรี่ได้ อย่างไรก็ตามถ้าเราสามารถออกแบบวงจรของแท็กให้กินกระแสไฟน้อยๆ ก็อาจจะมีอายุการใช้งานนานนับสิบปี แท็กชนิดแอ็กทีฟนี้จะมีกำลังส่งสูงและระยะการรับส่งข้อมูลไกลกว่าแท็ก ชนิดพาสซีฟ นอกจากนี้ยังทำงานในบริเวณที่มีสัญญาณรบกวนได้ดี

- แท็กชนิดพาสซีฟ (Passive Tag) จะไม่มีแบตเตอรี่อยู่ภายใน แต่จะทำงานโดยอาศัยพลังงานไฟฟ้าที่เกิดจากการเหนี่ยวนำคลื่นแม่เหล็กไฟฟ้าจากตัวอ่านข้อมูล จึงทำให้แท็กชนิดพาสซีฟมีน้ำหนักเบากว่าแท็กชนิดแอ็กทีฟ ราคาถูกกว่า และมีอายุการใช้งานไม่จำกัด แต่ข้อเสียก็คือระยะการรับส่งข้อมูลใกล้ และตัวอ่านข้อมูลจะต้องมีความไวสูง นอกจากนี้แท็กชนิดพาสซีฟมักจะมีปัญหาเมื่อนำไปใช้งานในสิ่งแวดล้อมที่มีสัญญาณแม่เหล็กไฟฟ้ารบกวนสูงอีกด้วย แต่ข้อได้เปรียบในเรื่องราคาและอายุการใช้งานทำให้แท็กชนิดพาสซีฟนี้เป็นที่นิยมมากกว่า

2.1.2.2 Reader หรือ Interrogator

หน้าที่สำคัญของตัวอ่านข้อมูล (Reader หรือ Interrogator) ก็คือการรับข้อมูลที่ส่งมาจากแท็กแล้วทำการตรวจสอบความผิดพลาดของข้อมูล ถอดรหัสข้อมูล และนำข้อมูลผ่านเข้าสู่กระบวนการต่อไป นอกจากนี้ตัวอ่านข้อมูลที่ดีต้องมีความสามารถในการป้องกันการอ่านข้อมูลซ้ำ เช่นในกรณีที่แท็กถูกวางทิ้งอยู่ในบริเวณสนามแม่เหล็กไฟฟ้าที่ตัวอ่านข้อมูลสร้างขึ้น หรืออยู่ในระยะการรับส่ง ก็อาจทำให้ตัวอ่านข้อมูลทำการรับหรืออ่านข้อมูลจากแท็กซ้ำอยู่เรื่อยๆไม่สิ้นสุด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ดังนั้นตัวอ่านข้อมูลที่ดีต้องมีระบบป้องกันเหตุการณ์เช่นนี้ที่เรียกว่าระบบ "Hands Down Polling" โดยตัวอ่านข้อมูล จะสั่งให้แท็กหยุดการส่งข้อมูลในกรณีเกิดเหตุการณ์ดังกล่าว หรืออาจมีบางกรณีที่มีแท็กหลายแท็กอยู่ในบริเวณสนามแม่เหล็กไฟฟ้าพร้อมกัน หรือที่เรียกว่า "Batch Reading" ตัวอ่านข้อมูลควรมีความสามารถที่จะจัดลำดับการอ่านแท็กทีละตัวได้

2.1.3 หลักการทำงานเบื้องต้นของระบบ

- ตัวอ่านข้อมูลจะปล่อยคลื่นแม่เหล็กไฟฟ้าออกมาตลอดเวลา และคอยตรวจจับว่ามีแท็กเข้ามาอยู่ในบริเวณสนามแม่เหล็กไฟฟ้าหรือไม่ หรืออีกนัยหนึ่งก็คือการคอยตรวจจับว่ามีการมอดูเลตสัญญาณเกิดขึ้นหรือไม่

- เมื่อมีแท็กเข้ามาอยู่ในบริเวณสนามแม่เหล็กไฟฟ้า แท็กจะได้รับพลังงานไฟฟ้าที่เกิดจากการเหนี่ยวนำของคลื่นแม่เหล็กไฟฟ้าเพื่อให้แท็กเริ่มทำงาน และจะส่งข้อมูลในหน่วยความจำที่ผ่านการมอดูเลตกับคลื่นพาหะแล้วออกมาทางสายอากาศที่อยู่ภายในแท็ก

- คลื่นพาหะที่ถูกส่งออกมาจากแท็กจะเกิดการเปลี่ยนแปลงแอมปลิจูด, ความถี่ หรือเฟส ขึ้นอยู่กับวิธีการมอดูเลต

- ตัวอ่านข้อมูลจะตรวจจับความเปลี่ยนแปลงของคลื่นพาหะแปลงออกมาเป็นข้อมูลแล้วทำการถอดรหัสเพื่อนำข้อมูลไปใช้งานต่อไป

2.2 ทฤษฎีไมโครคอนโทรลเลอร์

ไมโครคอนโทรลเลอร์ตระกูล MSC – 51 มีสมาชิกในตระกูลหลายเบอร์ด้วยกัน แต่ละเบอร์จะมีคุณสมบัติพิเศษบางอย่างแตกต่างกัน เช่น มีหน่วยความจำภายในสำหรับเก็บโปรแกรมและข้อมูลภายในชิปเพิ่มขึ้น มีวงจรเปลี่ยนค่าสัญญาณอนาลอกเป็นดิจิตอลในตัว สามารถรับสัญญาณอินเตอร์รัปต์ได้หลายชนิด ทำกระบวนการ DMA (Direct Memory Access) ได้ในตัว มีรีจิสเตอร์สำหรับใช้เป็นไทเมอร์หรือเคาน์เตอร์เพิ่มขึ้น

ไมโครคอนโทรลเลอร์ตระกูล MCS – 51 ทุกเบอร์ใช้แรงดันเพียง 5 โวลต์ในการทำงาน ส่วนกระแสไฟฟ้าที่ใช้จะแตกต่างกันไปตามชนิดของเทคโนโลยีที่ใช้การผลิต เบอร์ของไมโครคอนโทรลเลอร์ตระกูลนี้ที่มีตัวอักษร C อยู่ตรงกลางเบอร์ เช่น 80C31, 80C51 จะเป็นเบอร์ของชิปที่ผลิตโดยอาศัยเทคโนโลยี CMOS ซึ่งใช้พลังงานในการทำงานน้อยกว่าและสามารถควบคุมการใช้พลังงานของตัวชิปได้จากโปรแกรมเพื่อการประหยัดพลังงานในระบบ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.2.1 คุณสมบัติของไมโครคอนโทรลเลอร์ตระกูล MCS-51

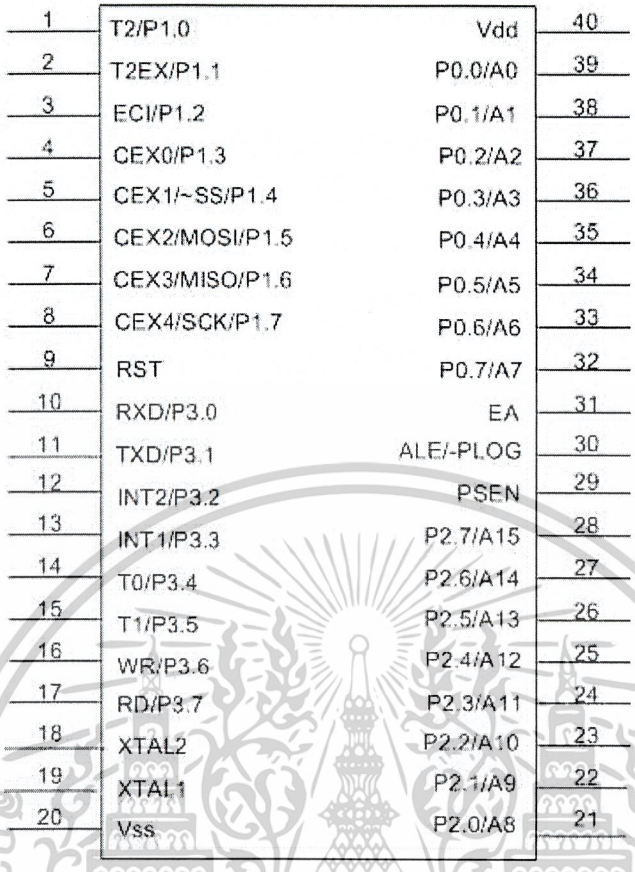
- เป็นไมโครคอนโทรลเลอร์ที่ใช้ซีพียูขนาด 8 บิต
- ภายในมีหน่วยความจำโปรแกรมเป็นแบบแฟลชสามารถลบและเขียนใหม่ได้พันครั้ง
- หน่วยความจำข้อมูลพื้นฐานเป็นหน่วยความจำแบบแรม ในบางเบอร์จะมีหน่วยความจำแบบอีอีพรอมเพิ่มเติม
- ขาพอร์ตเป็นแบบสองทิศทาง สามารถใช้งานเป็นได้ทั้งอินพุต และเอาต์พุต
- มีวงจรสื่อสารอนุกรมแบบฟูลดูเพล็กซ์
- ไทเมอร์/เคาน์เตอร์ขนาด 16 บิตอย่างน้อย 2 ตัว
- สามารถรองรับแหล่งกำเนิดอินเตอร์รัปต์ได้ 6 ประเภท
- สามารถขยายหน่วยความจำภายนอกเพิ่มเติมได้สูงสุด 64 กิโลไบต์
- มีวงจรกำเนิดสัญญาณนาฬิกาอยู่ภายในชิป
- มีวงจรสื่อสารอนุกรมแบบ SPI สำหรับในอนุกรม AT89Sxx
- มีวอตช์ด็อกไทเมอร์ในตัว สำหรับในอนุกรม AT89Sxx

2.2.2 การจัดขาของไมโครคอนโทรลเลอร์ตระกูล MCS – 51

หน้าที่การใช้งานแต่ละขาของชิปไมโครคอนโทรลเลอร์ในตระกูล MCS – 51 มีดังนี้

- ขา GND (ขา 20) สำหรับต่อลงกราวนด์
- ขา Vcc (ขา 40) สำหรับต่อแหล่งจ่ายแรงดันกระแสตรงขนาด 5 โวลต์ (DC 5 Volt)
- ขาพอร์ต 0 (ขา 32 – 39) มี 8 ขา ใช้เป็นขาสำหรับพอร์ต 0 ขนาด 8 บิต (P0.0 – 0.7) แบบ Open Drain Bidirectional พอร์ตนี้สามารถใช้งานเป็นอินพุทเอาท์พุทพอร์ตทั่วไปได้ โดยหากใช้งานเป็นอินพุทพอร์ต ต้องโหลดค่า 1 ไปยังแต่ละบิตของพอร์ตนี้ เพื่อบังคับให้ขาอยู่ในสถานะถูกปล่อยลอย (มีสถานะ high impedance) นอกจากใช้งานเป็นอินพุทพอร์ตแล้ว พอร์ต 0 ยังใช้ในการติดต่อหน่วยความจำสำหรับเก็บโปรแกรมและข้อมูลภายนอกชิปด้วย โดยส่งค่าแอดเดรสบัสไบต์ต่ำ (A0 – A7) และมัลติเพล็กซ์กับการรับส่งข้อมูล (D0 –D7) จากหน่วยความจำภายนอกระหว่างการเขียน หรืออ่านข้อมูล โดยมีวงจรพูลอัพภายใน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.3 แสดงตำแหน่งขาของไมโครคอนโทรเลอร์ MCS51

- ขาพอร์ต 1 (ขา 1 – 8) มี 8 ขา ใช้เป็นขาสำหรับพอร์ต 1 (P1.0 –P1.7) สามารถใช้งานเป็นอินพุทหรือเอาท์พุทพอร์ตทั่วไปได้ หากต้องการใช้งานเป็นอินพุทพอร์ต ต้องโหลดค่า 1 ไปยังแต่ละบิตของพอร์ตนี้ เพื่อให้มีสถานะ high impedance โดยมีวงจรพูลอัพภายใน

- ขาพอร์ต 2 (ขา 21 – 28) มี 8 ขา ใช้เป็นขาสำหรับพอร์ต 2 (P2.0 – P2.7) ขนาด 8 บิต แบบ Open Drain Bidirectional พอร์ตนี้สามารถใช้งานเป็นอินพุทเอาท์พุทพอร์ตทั่วไปได้ โดยหากใช้งานเป็นอินพุทพอร์ต ต้องโหลดค่า 1 ไปยังแต่ละบิตของพอร์ตนี้ เพื่อบังคับให้ขาอยู่ในสถานะ high impedance นอกจากจะใช้งานเป็นอินพุทเอาท์พุทพอร์ตทั่วไปแล้ว พอร์ต 2 ยังใช้งานในการติดต่อหน่วยความจำสำหรับเก็บโปรแกรมและข้อมูลภายนอกชิปด้วย โดยใช้สำหรับส่งค่าแอดเดรสบัสไบต์สูง (A8 – A15) และมีวงจรพูลอัพภายใน

- ขาพอร์ต 3 (ขา 10 – 17) มี 8 ขาใช้เป็นขาสำหรับพอร์ต 3 (P3.0 – 3.7) สามารถใช้งานเป็นอินพุทเอาท์พุทพอร์ตทั่วไปได้ หากต้องการใช้งานเป็นอินพุทพอร์ต ต้องโหลดค่า 1 ไปยังแต่ละบิตของพอร์ตนี้ เพื่อให้มีสถานะ high impedance โดยใช้วงจรพูลอัพภายใน นอกจากนี้ยังใช้งานในหน้าที่พิเศษต่าง ๆ อีกหลายอย่างดังนี้

- ขา P3.0 ใช้รับข้อมูลภายนอกแบบอนุกรม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- ขา P3.1 ใช้ส่งข้อมูลไปภายนอกแบบอนุกรม

- ขาP3.2 ใช้เป็นอินพุทเพื่อรับสัญญาณอินเตอร์รัปต์ชนิดที่ 0

- ขา P3.3 ใช้เป็นอินพุทเพื่อรับสัญญาณอินเตอร์รัปต์ชนิดที่ 1

- ขา P3.4 สัญญาณอินพุทให้เคาน์เตอร์ของไทเมอร์ 0

- ขา P3.5 สัญญาณอินพุทให้เคาน์เตอร์ของไทเมอร์ 1

- ขา P3.6 ใช้เป็นสัญญาณควบคุมการเขียนข้อมูล ไปยังหน่วยความจำสำหรับเก็บข้อมูลภายนอกชิป

- ขา P3.7 ใช้เป็นสัญญาณควบคุมการอ่านข้อมูลจากหน่วยความจำสำหรับเก็บข้อมูลภายนอกชิป

การใช้งานพอร์ต 3 ในหน้าที่พิเศษดังกล่าวนี้จะต้องโหลดค่า 1 แต่ละบิตที่ต้องการใช้ก่อนทุกครั้ง

- ขา RST (ขา 9) ใช้สำหรับการรีเซตวงจรทุกอย่างภายในชิปเพื่อเริ่มต้นการทำงานใหม่ การรีเซตใช้เมื่อเริ่มจ่ายพลังงานหรือเมื่อโปรแกรมเกิดทำงานผิดพลาด เมื่อต้องการรีเซตชิป MCS – 51 ขานี้ต้องมีสถานะ 1 เป็นเวลาอย่างน้อย 2 แมชชีนไซเคิลระหว่างที่ออสซิลเลเตอร์ยังทำงานอยู่ โดยต้องต่อตัวต้านทานค่า 8.2 กิโลโอห์ม เพื่อทำหน้าที่พูลดาวน์ (รักษาค่าแรงดันไฟฟ้าให้มีสถานะเป็นกราวนด์) และเพื่อให้ตัวชิปรีเซทเองเมื่อเริ่มจ่ายพลังงานให้ต่อตัวเก็บประจุขนาด 10 ไมโครฟารัดคร่อมระหว่างขา RST กับ Vcc

- ขา ALE/PROG (ขา 30) เป็นขาใช้สำหรับส่งสัญญาณออกไปภายนอก เพื่อควบคุมการแลตช์ค่าแอดเดรสไบต์ต่ำ (address latch enable) จากพอร์ต 0 ในระหว่างการติดต่อหน่วยความจำสำหรับเก็บโปรแกรม หรือข้อมูลภายนอก ปกติเมื่อไม่มีการติดต่อหน่วยความจำภายนอก ขานี้จะส่งสัญญาณพัลช์ออกมาด้วยความถี่ 1/8 ของความถี่ออสซิลเลเตอร์ ที่ใช้ตลอดเวลา ดังนั้นเราสามารถใช้ความถี่ที่จากขานี้ไปใช้งานอย่างอื่นได้ แต่ความถี่ที่ขานี้จะลดลงครึ่งหนึ่งในระหว่างติดต่อกับหน่วยความจำสำหรับเก็บข้อมูลที่อยู่ภายนอกชิป นอกจากนี้ขา ALE ยังใช้สำหรับควบคุมการเขียนโปรแกรมลงไปใน EPROM สำหรับ MCS – 51 เบอร์ที่มีหน่วยความจำสำหรับเก็บโปรแกรมภายในชิปเป็น EPROM

- ขา PSEN (ขา 29) ใช้ส่งสัญญาณสโตรปเพื่ออ่านสัญญาณจากโปรแกรมที่เก็บไว้ในหน่วยความจำภายนอกชิป (program strobe enable) เมื่อชิปทำงานด้วยโปรแกรมภายนอก ขานี้จะส่งสัญญาณสโตรป 2 ครั้งในแต่ละแมชชีนไซเคิล แต่ในช่วงการเขียนหรืออ่านข้อมูลจากหน่วยความจำภายนอกหรือเมื่อใช้โปรแกรมจากหน่วยความจำสำหรับเก็บโปรแกรมภายในชิปจะไม่มีสัญญาณออกมาจากขานี้

- ขา EA/Vpp (ขา 31) เป็นขาสำหรับใช้เลือกให้ MCS – 51 ทำงานจากโปรแกรมที่อยู่ภายในหรือภายนอกชิป โดยหากขานี้มีสถานะเป็น 0 หมายถึงให้ใช้โปรแกรมจากหน่วยความจำที่เก็บโปรแกรมภายนอก หากขานี้มีสถานะเป็น 1 หมายถึงบังคับให้ MCS – 51 ใช้โปรแกรมจาก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หน่วยความจำสำหรับเก็บโปรแกรมภายในชิป และสำหรับ MCS – 51 ที่มีหน่วยความจำสำหรับเก็บโปรแกรมภายในชิป สามารถเลือกให้ทำงานได้ทั้งจากโปรแกรมที่เก็บในหน่วยความจำภายในชิป หรือจากโปรแกรมที่เก็บไว้ในหน่วยความจำภายนอกชิปด้วยการต่อขาEN กับไฟเลี้ยงหรือกราวน์ตามลำดับ ส่วนใน MCS –51 ที่ไม่มีหน่วยความจำสำหรับเก็บโปรแกรมภายในชิป ให้ต่อขานี้ลงกราวน์เสมอ

- ขา XTAL 1 (ขา 19) ใช้ต่อคริสตอลภายนอก โดยเป็นอินพุทเข้าสู่วงจรออสซิเลเตอร์
- ขา XTAL 2 (ขา 18) ใช้ต่อคริสตอลภายนอก โดยเป็นเอาท์พุทออกจากวงจรออสซิเลตอร์

2.2.3 วิธีการเข้าถึงข้อมูล

คำสั่งที่ใช้ควบคุมการทำงานของ MCS – 51 มีสองประเภทคือ คำสั่งที่ต้องการข้อมูลมาดำเนินการเพื่อให้ได้ผลลัพธ์ และ คำสั่งที่ไม่ต้องการข้อมูลมาดำเนินการคำสั่งที่ต้องการข้อมูลจะมีวิธีในการเข้าถึงข้อมูลได้หลายวิธีดังนี้

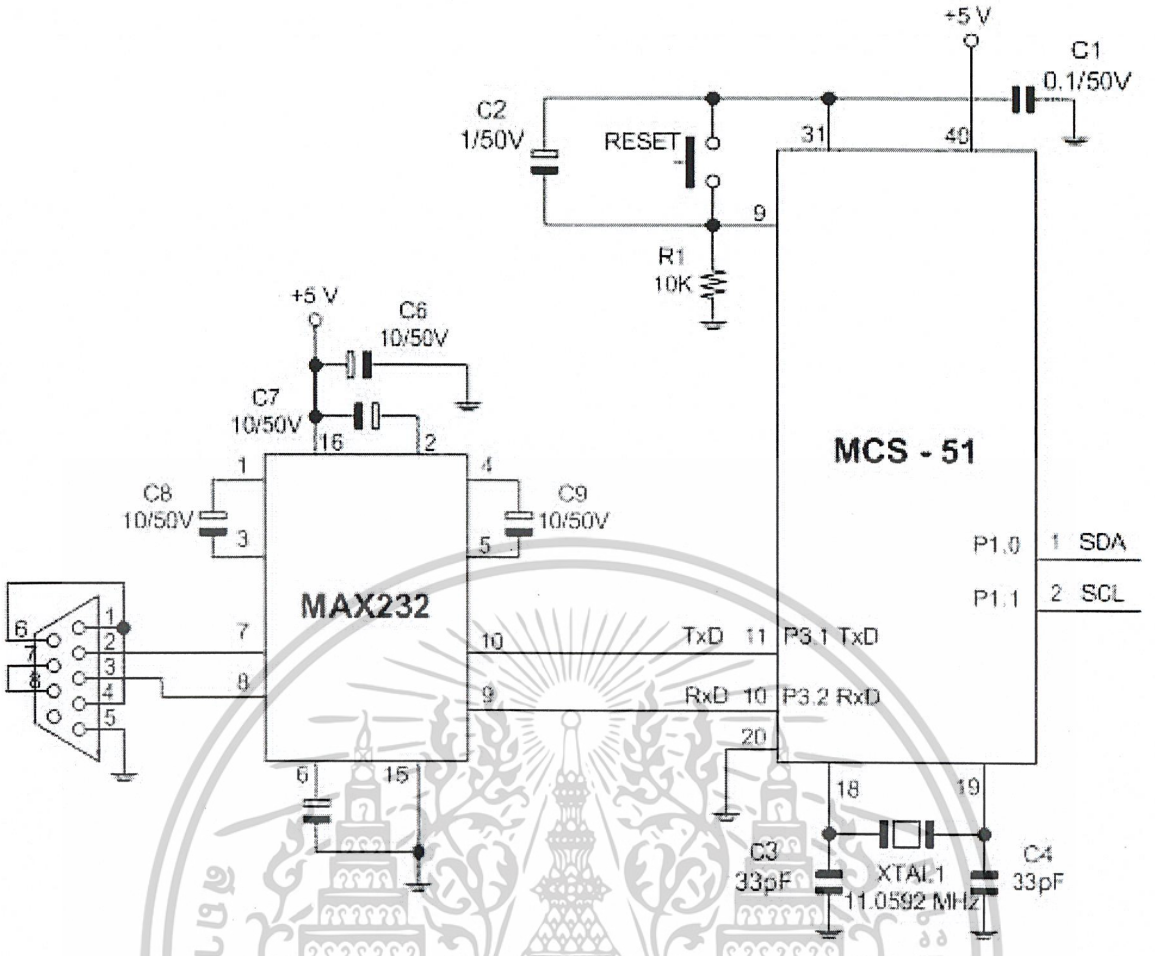
- วิธีการเข้าถึงข้อมูลโดยตรง (Direct addressing)
- วิธีการเข้าถึงข้อมูลโดยทางอ้อม (Indirect addressing)
- วิธีการเข้าถึงข้อมูลในรีจิสเตอร์ใช้งานทั่วไป (Register instructions)
- วิธีการเข้าถึงข้อมูลในรีจิสเตอร์เฉพาะของตัวคำสั่ง (Register – specific instructions)
- วิธีการเข้าถึงข้อมูลที่กำหนดเองโดยตรง (Immediate constants)
- วิธีการเข้าถึงข้อมูลที่มีตัวชี้อ้างอิง (Indexed addressing)

2.2.4 การเชื่อมต่อพอร์ตอนุกรมของคอมพิวเตอร์

การใช้งานพอร์ตอนุกรมของไมโครคอนโทรลเลอร์MCS-51มักนิยมใช้ในการติดต่อติดต่อเพื่อแลกเปลี่ยนกับคอมพิวเตอร์ผ่านทางพอร์ตคอมพิวเตอร์แต่เนื่องจากระดับสัญญาณของ RS-232 มีระดับตั้งแต่±3 ถึง±12V ในขณะที่ระดับสัญญาณของไมโครคอนโทรลเลอร์อยู่ในระดับทีทีแอล ดังนั้นจึงไม่สามารถเชื่อมต่อพอตร์อนุกรมของไมโครคอนโทรลเลอร์MCS-51 ได้โดยตรงจึงต้องอาศัยการเชื่อมต่อผ่านไอซีพิเศษที่ทำหน้าที่ในการแปลงระดับสัญญาณ

ไอซีที่ทำหน้าที่ในการแปลงระดับสัญญาณนี้ต้องทำการแปลงข้อมูลส่งของไมโครคอนโทรลเลอร์MCS-51 จากระดับทีทีแอลไปเป็นระดับของRS-232และทำการแปลงข้อมูลรับจากคอมพิวเตอร์จากระดับ RS-232 เป็นระดับทีทีแอลเพื่อให้สามารถถ่ายทอดไปยังไมโครคอนโทรลเลอร์MCS-51ได้อย่างสมบูรณ์ไอซีดังกล่าวมีด้วยกันหลายเบอร์จากผู้ผลิต อาทิ MAX232 จาก MAXIM หรือ ICL232 จาก HARRIS เป็นต้นไปในรูปที่ 2.4 แสดงการจัดขาของไอซี ICL232 ซึ่งใช้ในการแปลงสัญญาณ RS-232 ส่วนของวงจรของการต่อกับไมโคร คอนโทรลเลอร์ MCS-51 แสดงในรูปที่ 2.4

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



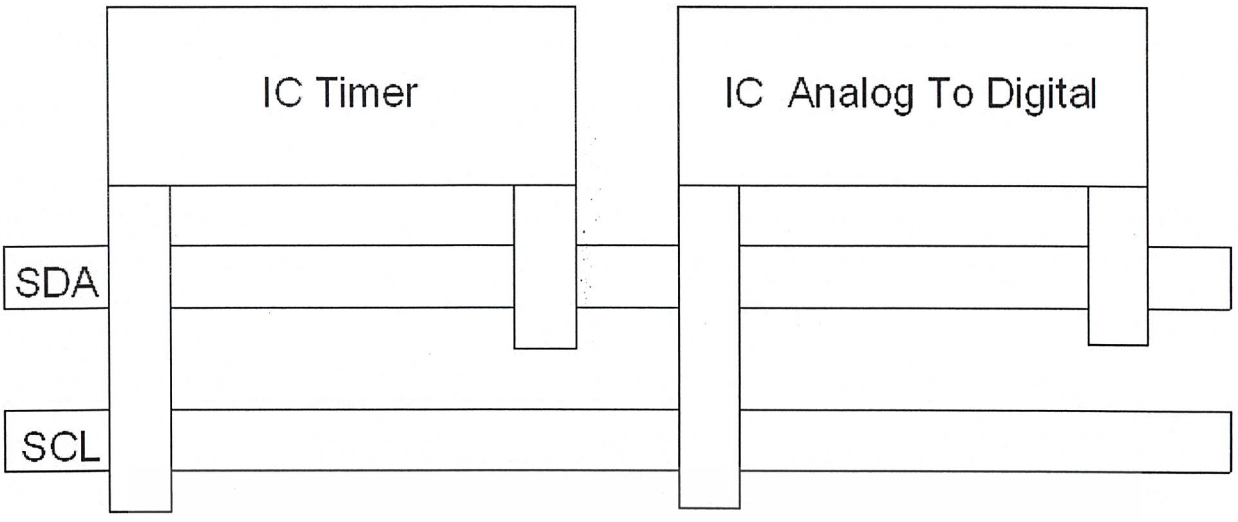
รูปที่ 2.4 วงจรเชื่อมต่อ MCS 51เข้ากับพอร์ตอนุกรมของคอมพิวเตอร์และไมโครคอนโทรลเลอร์

2.2.5 ความรู้เบื้องต้นเกี่ยวกับ I²C

I²C ย่อมาจาก Inter-IC Communication หมายถึง การติดต่อสื่อสารระหว่างไอซีโดยบัส I²C ได้รับการพัฒนาขึ้นโดยฟิลิปส์(Philip)ด้วยจุดมุ่งหมายหลักคือต้องการให้ไอซีหรือโมดูลสามารถติดต่อสั่งงาน และควบคุมภายใต้สัญญาณ 2 เส้น เส้นหนึ่งคือ สายข้อมูล อีกเส้นหนึ่งคือสายสัญญาณนาฬิกาที่ใช้ในการกำหนดจังหวะการทำงาน การต่อร่วมกันของอุปกรณ์บนบัส I²C ทำได้ง่ายมาก เพียงต่อสายสัญญาณข้อมูลและสายสัญญาณนาฬิกาของอุปกรณ์แต่ละตัวขนานหรือพ่วงกันไป ส่วนการกำหนดแอดเดรสหรือตำแหน่งสำหรับติดต่ออุปกรณ์แต่ละตัว จะใช้รหัสข้อมูลและการกำหนดสภาวะลอจิกที่ขาแอดเดรสของอุปกรณ์แต่ละตัว

สายข้อมูลบนบัส I²Cมีชื่อเรียกอย่างเป็นทางการว่า สายข้อมูลอนุกรมหรือ SDA(Serial Data Line)ส่วนสายสัญญาณนาฬิกามีชื่อเรียกว่า สายสัญญาณนาฬิกาอนุกรมหรือ SCL (Serial Clock Line) ในการอธิบายต่อไปนี้จะเรียกสายสัญญาณทั้งสองว่า สาย SDA และ SCL

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.5 ผังแสดงการเชื่อมต่อของอุปกรณ์ต่างๆบนระบบบัส

ในรูปที่ 2.5 แสดงผังของการเชื่อมต่ออุปกรณ์ต่างๆบนบัส I²Cจะเห็นได้ว่าอุปกรณ์ที่ทำการเชื่อมต่อบนบัส I²C มีหลากหลายไม่ว่าจะเป็นไอซีขยายพอร์ตอินพุตเอาต์พุต (I/O Expander),ไอซีแปลงสัญญาณอนาลอกเป็นดิจิตอล(ADC) และแปลงสัญญาณดิจิตอลเป็นอนาลอก(DAC),ไอซีรีลไทม์คล๊อก (RTC),ไอซีขับโมดูลLCD,หน่วยความจำอีอีพรอม และไมโครคอนโทรลเลอร์

2.2.5.1 คุณสมบัติโดยทั่วไปของบัส I²C

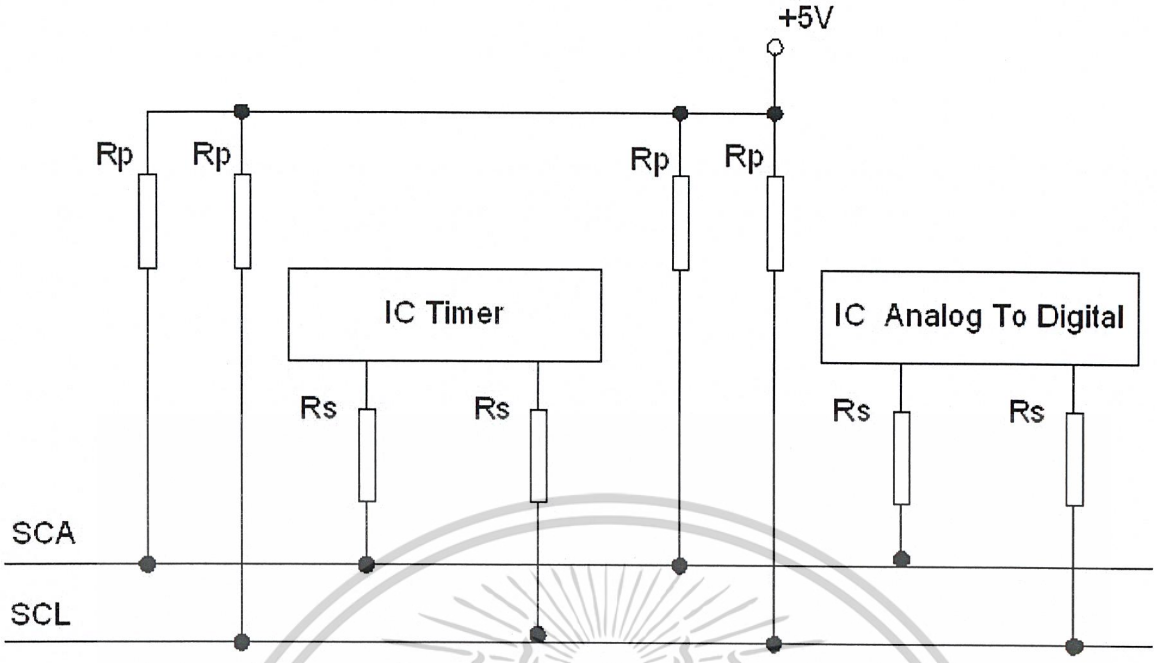
สาย SDA และ SCLเป็นสายสัญญาณ 2 ทิศทาง(bi-directional line)ต้องมีการต่อตัวต้านทานพูลอัปกับแรงดัน+5Vไว้ตลอดเวลา เพื่อให้สายมีสภาวะลอจิกสูงในขณะที่ไม่มีการติดต่อใช้งานทั้งยังช่วยมนการป้องกันสัญญาณรบกวนที่อาจมีเข้ามาในสายสัญญาณทั้งสอง วงจรเอาต์พุต ของอุปกรณ์ที่ต่ออยู่บนบัส I²Cต้องมีลักษณะเป็นวงจรเดรนเปิด(open-drain) หรือคอลเลคเตอร์เปิด(open-collector)ดังรายละเอียดในรูปที่ 2.6

อัตราการถ่ายทอดข้อมูลบนบัส I²C สูงถึง 100 กิโลบิตต่อวินาทีในโหมดปกติ (standard mode) และสูงสุดถึง 400 กิโลบิตต่อวินาทีในโหมดความเร็วสูง (fast mode) อุปกรณ์ที่ต่อร่วมอยู่บนบัสI²C จะต้องมีค่าความจุไฟฟ้ารวมที่เกิดขึ้นระหว่างสาย SDA และ SCL ไม่เกิน 400pF การเข้าถึงอุปกรณ์บนบัสI²C ใช้ข้อมูลสำหรับการเข้าถึง 2 ค่าคือ 7 บิต (7-bit addressing) หรือ 10บิต (10-bit addressing)

ข้อเด่นอีกประการหนึ่งของบัสI²C คือสามารถเชื่อมต่ออุปกรณ์ที่มีไฟเลี้ยงไม่เท่ากันให้สามารถติดต่อสารกันได้ โดยอุปกรณ์บนบัสI²C ตัวหนึ่งอาจใช้ไฟเลี้ยง+5Vในขณะอีกตัวหนึ่งใช้ไฟเลี้ยง+12V การต่อร่วมกันบนบัสI²C สามารถทำได้ในลักษณะเดียวกันกับกรณีที่ทั้ง 2 ใช้ไฟเลี้ยงเท่ากัน กล่าวคือ ให้ต่อสาย SDA และ SCL ของอุปกรณ์แต่ละตัวเข้าด้วยกัน และต้องต่อตัวต้านทานพูลอัป (Rp) เข้ากับแรงดัน+5V ไว้ด้วยเสมอดังแสดงในรูปที่ 2.6

ในกรณีอาจมีแรงดันไฟกระชากขนาดใหญ่ปะปนเข้ามาในบัสI²C ที่ขา SDA และSCLของอุปกรร์แต่ตัวต้องต่อตัวต้านทานอนุกรมกับขา SDA และ SCL เรียกว่า Rs ก่อนต่อเข้าสู่บัสI²Cดังรูปที่ 2.6

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.6 การตัวต้านทาน Rpพื่อลดสัญญารรบกวนขนาดใหญ่ที่อาจเข้ามาในบัส I^2C

2.2.5.2 หลักการของบัสI²C

บัสI²Cประกอบด้วยสายสัญญาณ 2 เส้น ดังที่ได้กล่าวมาแล้วคือSDA และ SCL อุปกรณ์ที่ต่อพ่วงบนบัสสามารถมีได้มากมาย ดังนั้นจึงต้องมีการกำหนดรูปแบบของการกำหนดบนบัส หรือเรียกว่า โปรโตคอล (protocol) เพื่อให้ผู้ใช้งานทราบว่า ขณะนี้อุปกรณ์ใดติดต่ออยู่ และอุปกรณ์ตัวใดเป็นตัวรับหรือตัวส่ง ต่อไปนี้จะอธิบายลักษณะหน้าที่ และนิยามของตัวอุปกรณ์ที่ต่ออยู่บนบัสI²Cเพื่อเป็นข้อตกลงพื้นฐานก่อนที่จะอธิบายการทำงานของบัสI²Cต่อไป

อุปกรณ์ที่ทำหน้าที่เป็นผู้สร้างข้อมูลหรือส่งข้อมูล เรียกว่า ตัวส่ง (transmitter)

อุปกรณ์ที่ทำหน้าที่เป็นผู้รับ เรียกว่า ตัวรับ (receiver) อุปกรณ์บนบัสI²Cสามารถเป็นได้ทั้งตัวรับและตัวส่ง บางอุปกรณ์ทำหน้าที่เป็นตัวรับอย่างเดียว จะไม่มีอุปกรณ์ใดบนบัสทำหน้าที่เป็นตัวส่งเพียงอย่างเดียว

อุปกรณ์ที่ทำหน้าที่ควบคุมจังหวะการติดต่อบนบัสI²C เรียกว่า มาสเตอร์ (master)

อุปกรณ์ที่ถูกควบคุมหรืออุปกรณ์ที่ต่อพ่วงเข้าไปบนบัสI²C เรียกว่า สเลฟ (slave)

ข้อกำหนด 2 ประการสำคัญของการติดต่อบนบัสI²Cคือ

(1) การถ่ายทอดข้อมูลขณะเกิดขึ้นได้เมื่อบัสว่างเท่านั้น

(2) ในการถ่ายทอดข้อมูล เมื่อใดก็ตามที่สาย SCL มีสถานะลอจิกสูง สายข้อมูลต้องรักษาข้อมูลไว้ อย่สให้เกิดการเปลี่ยนแปลงขึ้นเด็ดขาด มิฉะนั้น สัญญาณที่เกิดขึ้นจะได้รับแปลความหมายเป็นสัญญาณควบคุมแทน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.2.5.3 สภาวะที่เกิดขึ้นบนบัสI²C

มีด้วยกัน 5 สภาวะดังนี้

(1) บัสว่าง (Bus not busy) สภาวะนี้เกิดขึ้นเมื่อสภาวะในสาย SDA และ SCL เป็นลอจิกสูงทั้งคู่ นั่นหมายความว่า การถ่ายทอดข้อมูลเริ่มต้นขึ้นได้

(2) เริ่มต้นการถ่ายทอดข้อมูล (Start data transfer) เกิดขึ้นเมื่อสาย SDAมีการเปลี่ยนแปลงระดับลอจิกสูงไปลอจิกต่ำในขณะที่สาย SCLมีสถานะลอจิกสูง เรียกสภาวะนี้ว่า สภาวะเริ่มต้น (start)

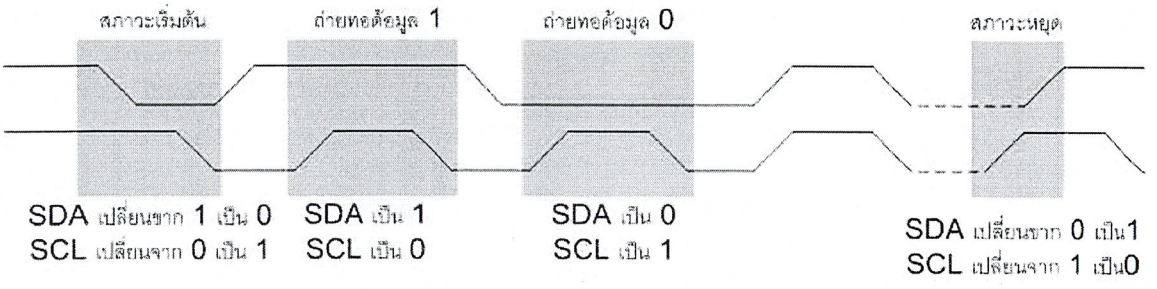
(3) หยุดการถ่ายทอดข้อมูล (Stop data transfer) เกิดขึ้นเมื่อสาย SDA มีการเปลี่ยนแปลงระดับลอจิกจากต่ำไปสูง ในขณะที่สายSCL มีสภาวะลอจิกสูงเรียกสภาวะที่เกิดขึ้นว่าสภาวะหยุด (stop)

(4) ข้อมูลดำรงอยู่บนบัส (data valid) สภาวะนี้เกิดขึ้นจากสภาวะเริ่มต้น โดยสถานะลอจิกที่เกิดขึ้นบนสาย SDA ก็คือข้อมูลที่ทำการถ่ายทอด เมื่อสายSCL เป็นลอจิกสูง สถานะที่สาย SDA ต้องคงที่ เพื่อให้อุปกรณ์รับข้อมูลในจังหวะนั้น เป็น"0"หรือเป็น"1" ข้อมูลอาจเกิดการเป็นเปลี่ยนแปลงได้ในขณะที่สายSCL เป็นลอจิกต่ำ แต่เมื่อใดก็ตามที่ต้องการให้เกิดการถ่ายทอดข้อมูลอย่างสมบูรณ์ สถานะลอจิกที่ขาSDA ต้องคงที่ตลอดช่วงเวลาที่สายSCL มีสถานะลอจิกสูง หากเกิดการเปลี่ยนแปลงสถานะทางลอจิกในขณะที่สายSCL มีลอจิกสูงอยู่นั้น อุปกรณ์มาสเตอร์ที่ทำหน้าที่ควบคุการถ่ายทอดข้อมูลจะแปลความหมายเป็นสภาวะหยุดหรือสภาวะเริ่มต้นก็ได้ทำให้ข้อมูลที่ทำการถ่ายทอดนั้นเกิดความผิดพลาดขึ้น

(5) รับรู้ข้อมูล(acknowledge) เกิดขึ้นหลังจากที่ถ่ายทอดข้อมูลจากตัวส่งมายังตัวรับเกิดขึ้นอย่างสมบูรณ์ โดยตัวส่งจะทำการส่งข้อมูลมา 1 บิตเรียกว่า บิตรับรู้(acknowledge bit) มีสถานะเป็นลอจิกสูง หลังจากส่งข้อมูลมาครบถ้วน ส่วนอุปกรณ์มาสเตอร์จะทำการส่งสัญญาณรับรู้พิเศษซึ่งสัมพันธ์กับสัญญาณนาฬิกาเพื่อตอบสนองบิตรับรู้ที่ส่งมาจากตัวส่งทางด้านตัวรับจะส่งบิตรับรู้ที่มีสถานะลอจิกต่ำลงบนบัส อุปกรณ์สเลฟที่ถูกอ้างถึงในการติดต่อหรือกำลังติดต่อในขณะนั้นก็จะกำเนิดบิดรับรู้เพื่อตอบสนองให้ทราบว่าได้รับชข้อมู,ในแต่ละไบต์เรียบร้อยแล้ว

ในรูปที่ 2.7 เป็นไดอะแกรมเวลาที่แสดงถึงการเกิดสภาวะต่างๆบนบัสI²Cไม่ว่าจะเป็นสภาวะบัสว่าง, เริ่มต้น, ถ่ายทอดข้อมูล, รับรู้, และหยุดการถ่ายทอดข้อมูล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.7 ไดอะแกรมเวลาแสดงสถานะต่างในบัส I^2C

2.2.5.4 การทำงานบนบัสI²C

ก่อนที่จะเริ่มทำการถ่ายทอดข้อมูลระหว่างอุปกรณ์ต่างๆที่ต่ออยู่บนบัสต้องมีการอ้างถึงเสียก่อนโโยการอ้างถึงอุปกรณ์บนบัสI²C นั้นจะใช้การอ้างถึงแบบ 7 บิตหรือ 10 บิต ในกรณีที่มีอุปกรณ์ต่ออยู่บนบัสไม่มากใช้การอ้างถึงแบบ 7 บิตก็เพียงพอ แต่ถ้ามีอุปกรณ์ต่ออยู่บนบัสมากกว่า 127 แอดเดรสจำเป็นต้องใช้การอ้างถึงแบบ 10 บิต หลังจากที่ติดต่ออุปกรณ์แต่ละตัวได้เรียบร้อยแล้ว ก็จะเริ่มต้นการถ่ายทอดข้อมูลกันต่อไป

2.2.5.5 การอ้างถึงแบบ 7 บิต (7-bit addressing)

ข้อมูลไบต์แรกหลังจากที่สภาวะเริ่มต้นคือ ข้อมูลที่ใช้ในการอ้างถึงอุปกรณ์ที่ต้องการติดต่อหรือข้อมูลกำหนดแอดเดรส โดยมีรูปใน7บิตบนรวมทั้งบิตMSB ด้วยจะเป็นข้อมูลแอดเดรสของอุปกรณ์สเลฟที่ต้องการติดต่อ โดยแบ่งเป็น บิตกำหนดแอดเดรสคงที่(fixed address bit) จำนวน4บิต ซึ่งข้อมูลนี้อุปกรณ์แต่ละตัวจะถูกกำหนดมาจากผู้ผลิต ไม่สามรถเปลี่ยนแปลงแก้ไขได้ ถัดมาอีก3บิตเป็นบิตแอดเดรสที่สามารถโปรแกรมได้(programmable address bit) โดยผู้ใช้งานต้องกำหนดสถานะลอจิกให้แก่ขาA0-A2 ของอุปกรณ์ที่มีการเชื่อมต่อแบบบัสI²C ส่วนในบิตLSBเป็นบิตที่ใช้กำหนดการอ่านหรือเขียนข้อมูลกับอุปกรณ์สเลฟตัวนั้นๆหากบิตLSBเป็น"0" หมายถึงต้องการเขียนข้อมูลไปยังอุปกรณ์นั้น ถ้าเป็น"1" จะเป็นการอ่านข้อมูลจากอุปกรณ์สเลฟ

ข้อมูลในไบต์ต่อมาคือ ข้อมูลควบคุม(Control byte) ในอุปกรณ์แต่ละตัวมีการกำหนดอุปกรณ์ควบคุมที่แตกต่างกันไป

ข้อมูลในไบต์ต่อมาคือ ข้อมูที่ทำการถ่ายทอดจริง(Data)

หลังจากที่มีการถ่ายทอดข้อมูลในแต่ละไบต์ อุปกรณ์สเลฟที่มีการติดต่อส่งสัญญาณรับรู้ตอบกลับมาด้วยทุกครั้ง เพื่อให้ขบวบการถ่ายทอดข้อมูลสามารถดำเนินต่อไปได้ รูปแบบข้อมูลอนุกรมที่เกิดขึ้นมาในการติดต่อบนบัส I²Cของการอ้างถึงแบบ7บิต

2.2.5.6 การอ้างถึงแบบ 10 บิต

ในการอ้างถึงแบบนี้ ยังคงใช้รูปแบบข้อมูลอนุกรมที่เหมือนกับแบบ7 บิต หากจะมีข้อมูลเพิ่มเติมขึ้นมาเล็กน้อยโดยข้อมูลไบต์แรกหลังจากเกิดสภาวะเริ่มต้น ต้องกำหนดให้5 บิตบนมีข้อมูลเป็น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สำนักหอสมุดกลาง พระจอมเกล้าลาดกระบัง

11110 ส่วนอีก2 บิตต่อมาเป็นบิตแอดเดรสของอุปกรณ์ที่ต้องการติดต่อด้วย ในบิตLSB ของข้อมูลไบต์แรกยังคงเป็นการกำหนดว่าต้องการอ่านหรือเขียนข้อมูลกับอุปกรณ์สเลฟตัวที่ต้องการติดต่อด้วย ข้อมูลไบต์ต่อมาเป็นข้อมูลแอดเดรสในไบต์ที่2อุปกรณ์ที่ต้องการติดต่อด้วย ข้อมูลไบต์ถัดไปจึงเป็นข้อมูลควบคุม ข้อมูลหลังจากนั้นจะเป็นข้อมูลจริงที่ใช้ในการติดต่อ

เช่นเดียวกับการอ้างถึงแบบ7 บิต หลังจากการถ่ายทอดข้อมูลครบทุกไบต์ต้องมีสภาวะรับรู้เกิดขึ้นเพื่อให้ขบวบการถ่ายทอดข้อมูลสามารถดำเนินต่อไปได้รูปแบบข้อมูลการอ้างถึงแบบ10 บิต

2.3 ความรู้เบื้องต้นเกี่ยวกับ Visual Basic

Visual Basic จะเรียกโปรแกรมที่กำลังพัฒนาอยู่ว่าโปรเจ็ค (Project) ซึ่งประกอบด้วยส่วนต่าง ๆ เช่น ฟอร์ม ปุ่มคำสั่ง (Command button) ช่องใส่ข้อความ (Text box) และ โปรแกรมที่เราต้องการเขียนปะไว้กัเหตุการณ์ต่าง

2.3.1 ชนิดของตัวแปรใน Visual Basic

ตัวแปร (Variable) หมายถึงชื่อของหน่วยเก็บข้อมูลหรือตำแหน่งในหน่วยความจำของคอมพิวเตอร์ที่ใช้เก็บค่าต่าง ๆ ที่เปลี่ยนแปลงได้ระหว่างการประมวลผลในเครื่อง คอมพิวเตอร์ตัวแปรในภาษาทางคอมพิวเตอร์นั้นมีอยู่มากมายหลายชนิดตามลักษณะการใช้งาน โดยตัวแปรแต่ละชนิดก็จะมีช่วงการเก็บค่า หรือประเภทของค่าที่เก็บ รวมทั้งเนื้อที่ที่ใช้ในหน่วยความจำในการเก็บค่าแตกต่างกันในตามตาราง 2.1

ชนิด	รายละเอียด	หน่วยความจำ
Byte	มีค่าตั้งแต่ 0 – 255	1 ไบต์
Integer	ใช้เก็บเลขจำนวนเต็ม มีค่าตั้งแต่ -32,768 ถึง 32,767	2 ไบต์
Long	ใช้เก็บเลขจำนวนเต็ม มีค่าตั้งแต่ -2,147,483,648 ถึง 2,147,483,647	4 ไบต์
Single	ใช้เก็บเลขจำนวนจริงมีจุดทศนิยม โดยมีค่าลบตั้งแต่ -3.402823×10^{38} ถึง $-1.401298 \times 10^{-45}$ และ ค่าบวกตั้งแต่ 1.401298×10^{-45} ถึง 3.402823×10^{38}	4 ไบต์

ตารางที่2.1 ตัวแปรชนิดต่างๆ

62538
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ชนิด	รายละเอียด	หน่วยความจำ
Double	ใช้เก็บเลขจำนวนจริงมีจุดทศนิยม โดยมีค่าลบตั้งแต่ $-1.79769313486232 \times 10^{308}$ ถึง $-4.94065645841247 \times 10^{-324}$ และค่าบวกตั้งแต่ $4.94065645841247 \times 10^{-324}$ ถึง $1.79769313486232 \times 10^{308}$	8 ไบต์
Currency	ใช้เก็บเลขทศนิยมตำแหน่งจำกัด (Fixed Decimal Point) มีค่าตั้งแต่ -922,337,203,685,477.5808 ถึง 922,337,203,685,477.5808	8 ไบต์
Boolean	ใช้เก็บค่าทางตรรกะ (logic) คือ True และ False	2 ไบต์
String	ใช้เก็บกลุ่มของตัวอักษรหรือชุดของข้อความ (Character string)	ขึ้นกับความยาวของข้อความ
Date	ใช้เก็บตัวแปรที่เป็นเวลา หรือ วันที่	8 ไบต์
Object	ใช้เก็บตัวแปรที่อ้างถึงออบเจ็คใด ๆ	4 ไบต์
Variant	เป็นตัวแปรพิเศษของ Visual Basic ไมารถใช้เก็บตัวแปรได้ทุกแบบตามข้างต้น	22 ไบต์

ตารางที่ 2.1 (ต่อ) ตัวแปรชนิดต่างๆ

2.3.2ตัวดำเนินการ, นิพจน์ และประโยคคำสั่ง

ตัวดำเนินการ (Operator) และนิพจน์ (Expression) ต่าง ๆ ซึ่งเป็นพื้นฐานสำคัญอันหนึ่งในการเขียนโปรแกรมเพื่อให้ทำงานได้ตามที่เราต้องการ

ตัวดำเนินการ (Operator) หมายถึงการคำนวณหรือการทำฟังก์ชันทางคณิตศาสตร์ เช่น บวก ลบ คูณ หาร โดยเรียกประโยคที่มีการนำตัวดำเนินการมาใช้กับตัวแปรหรือตัวเลขว่าเป็น *"นิพจน์"* หรือ *Expression* เช่น เครื่องหมายบวก (+) เป็นตัวดำเนินการ ส่วน 1+2 จะเป็นนิพจน์ สำหรับทางคอมพิวเตอร์จะแบ่งตัวดำเนินการออกเป็น 3 กลุ่ม คือตัวดำเนินการทางคณิตศาวตร์ (Arithmetic), ตัวดำเนินการเปรียบเทียบ (Relational) และ ตัวดำเนินการทางตรรกะ (Logical) ซึ่งมีรายละเอียด ดังนี้

- **ตัวดำเนินการทางคณิตศาสตร์** (Arithmetic Operator) เป็นตัวดำเนินการใช้คำนวณทางคณิตศาสตร์ เช่น +,-,*,/, และ ^ (ยกกำลัง) เป็นต้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- ตัวดำเนินการเปรียบเทียบ (Relational Operator) หรือเรียกอีกอย่างว่า Comparison Operator เป็นตัวดำเนินการที่ใช้ประเมินค่าข้อมูล ผลลัพธ์ที่ได้จะมีค่าเป็นจริง (True) หรือเท็จ (False) เท่านั้น ตัวอย่าง ได้แก่

ตัวดำเนินการ	ความหมาย	ตัวอย่างนิพจน์
=	เท่ากับ	A = B
>	มากกว่า	A > B
<	น้อยกว่า	A < B
<>	ไม่เท่ากับ	A <> B
>=	มากกว่าหรือเท่ากับ	A >= B
<=	น้อยกว่าหรือเท่ากับ	A <= B

ตารางที่ 2.2 ตัวดำเนินการเปรียบเทียบ

ตัวดำเนินการทางตรรกะ (Logical Operator) จะเป็นตัวดำเนินการที่ให้ค่าเป็นจริง (True) หรือเท็จ (False) โดยจะมีตารางค่าการกระทำ ดังนี้

and เทียบได้กับคำว่า "และ" ยกตัวอย่างประโยคที่ว่า "น้องป๊อปสวยและเก่ง" ประโยคนี้จะเป็นจริงก็ต่อเมื่อ "น้องป๊อปสวย" เป็นจริง และ "น้อง ป๊อปเก่ง" เป็นจริงด้วย ถ้าประโยคใดประโยคหนึ่งเป็นเท็จก็จะทำให้ประโยคและเป็นเท็จ

นิพจน์1	นิพจน์2	นิพจน์1 และ นิพจน์ 2
0	0	0
0	1	0
1	0	0
1	1	1

1 = จริง (True), 0 = เท็จ (False)

ตารางที่ 2.3 Logical Operator **"and"**

or เทียบได้กับคำว่า "หรือ" ยกตัวอย่างประโยคที่ว่า "น้องป๊อปสวยและเก่ง" ประโยคนี้จะเป็นจริงก็ต่อเมื่อ "น้องป๊อปสวย" หรือ "น้อง ป๊อปเก่ง" เป็นจริง เพียงแต่น้องป๊อปสวยหรือเก่งก็เพียงพอให้ประโยคเป็นจริงแล้ว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

นิพจน์1	นิพจน์2	นิพจน์1 และ นิพจน์ 2
0	0	0
0	1	1
1	0	1
1	1	1

1 = จริง (True), 0 = เท็จ (False)

ตารางที่ 2.4 Logical Operator **"or"**

xor หรือ Exclusive OR จะมีค่าเป็นจริงก็ต่อเมื่อประโยคทั้งสองมีค่าตรรกะต่างกัน ถ้ามีค่าตรรกะเหมือนกัน xor จะเป็นเท็จ

นิพจน์1	นิพจน์2	นิพจน์1 และ นิพจน์ 2
0	0	0
0	1	1
1	0	1
1	1	0

1 = จริง (True), 0 = เท็จ (False)

ตารางที่ 2.5 Logical Operator **"xor"**

not เป็นการสลับค่าตรรกะของตัวแปรหรือนิพจน์ จากจริงเป็นเท็จหรือจากเท็จเป็นจริง

นิพจน์1	นิพจน์2
0	1
1	0

1 = จริง (True), 0 = เท็จ (False)

ตารางที่ 2.6 Logical Operator **"not"**

eqv มาจากคำว่า Equivalence หรือเทียบเท่า ค่าผลลัพธ์จะเป็นจริงก็ต่อเมื่อค่าตัวแปรหรือนิพจน์ที่นำมา Eqv กันมีค่าตรรกะเหมือนกัน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

นิพจน์1	นิพจน์2	นิพจน์1 และ นิพจน์ 2
0	0	1
0	1	0
1	0	0
1	1	1

1 = จริง (True), 0 = เท็จ (False)

ตารางที่ 2.7 Logical Operator **"eqv"**

imp มาจากคำว่า Implication หรือ ถ้า – แล้ว ในวิชาตรรกะศาสตร์ ค่าผลลัพธ์จะเป็นเท็จก็ต่อเมื่อ นิพจน์1 จริง และ นิพจน์2 เป็นเท็จเท่านั้น ในกรณีอื่นจะเป็นจริงหมด

นิพจน์1	นิพจน์2	นิพจน์1 และ นิพจน์ 2
0	0	1
0	1	1
1	0	0
1	1	1

* 1 = จริง (True), 0 = เท็จ (False)

ตารางที่ 2.8 Logical Operator **"imp"**

2.3.3ลำดับการประมวลผลของตัวดำเนินการ

สมมติว่ามีนิพจน์หนึ่งเขียนว่า 3 + 4 * 12 ถามว่าคำตอบของนิพจน์นี้คือเท่าไร ? จะพบว่าสามารถตอบได้สองคำตอบขึ้นอยู่กับว่าเราจะทำตัวดำเนินการใดก่อน เช่น ถ้าทำคูณก่อนก็จะได้คำตอบคือ 3 + 48 = 51 แต่ถ้าทำบอกก่อนจะได้คำตอบเป็น 7 * 12 = 84 คราวนี้ลองนึกดูว่าจะเกิดอะไรขึ้น ถ้าคอมพิวเตอร์ไม่กำหนดเป็นมาตรฐานที่แน่นอนไว้ว่าจะประมวลผลตัวดำเนินการใดก่อนหลัง ซึ่งนั่นก็คือเราจะไม่ทราบได้เลยว่าผลลัพธ์ของโปรแกรมที่เขียนไปจะเป็นอย่างไร และ ถูกต้องตรงตามความต้องการของเราหรือไม่

จากปัญหาดังกล่าวจึงได้มีการกำหนดว่า ถ้าในนิพจน์หนึ่ง ๆ มีตัวดำนเนินการหลายตัว การคำนวณจะมีลำดับการประมวลผล (Precedence) ตามตารางต่อไปนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ลำดับ			
ก่อน	**Arithmetic**	**Relational**	**Logical**
	ยกกำลัง (^)	เท่ากับ (=)	not
	ทำเป็นลบ (-)	ไม่เท่ากัน (<>)	and
	คูณและหาร (* , /)	น้อยกว่า (<)	or
	หารจำนวนเต็ม (\ )	มากกว่า (>)	xor
	หาเศษการหาร (Mod)	น้อยกว่าหรือเท่ากับ (<=)	eqv
	บวกและลบ (+ , -)	มากกว่าหรือเท่ากับ (>=)	Imp
หลัง	เชื่อมตัวข้อความ (&)	Like, Is	

ตารางที่ 2.9 ลำดับการประมวลผลของตัวดำเนินการ

2.3.4 คำสั่งต่างๆที่ใช้งาน

- การประกาศตัวแปรและกฎการตั้งชื่อตัวแปรใน Visual Basic

การประกาศตัวแปรคือ การที่เราบอกเครื่องคอมพิวเตอร์ให้เตรียมเนื้อที่ในหน่วยความจำสำหรับเก็บตัวแปรที่เราจะใช้ในการประมวลผลในโปรแกรม โดยการประกาศตัวแปรจะใช้คำสั่งว่า Dim (Dimension) ซึ่งมีไวยากรณ์ ดังนี้

รูปแบบ Dim ชื่อตัวแปร [as ชนิดของตัวแปร]

ค่าที่อยู่ภายใต้เครื่องหมาย [ ] จะเป็น Optional คือใส่หรือไม่ก็ได้ ถ้าไม่ได้ใส่จะเป็นค่า ดีฟอลต์ (Default) ที่กำหนดไว้แล้ว สำหรับชนิดของตัวแปรก็คือชนิด Variantส่วนการตั้งชื่อตัวแปรใน Visual Basic นั้นมีกฎเกณฑ์ ดังต่อไปนี้

1. ชื่อตัวแปรที่ใช้ต้องขึ้นต้นด้วยตัวอักษร
2. ชื่อตัวแปรห้ามยาวเกิน 255 ตัวอักษร
3. สามารถใช้ตัวอักษรตัวเลขหรือเครื่องหมายขีดล่าง (_) ผสมเป็นชื่อ แต่ห้ามมีจุด, ขีดหรือเครื่องหมายอื่นภายในชื่อ
4. ห้ามซ้ำกับคำที่สงวนไว้ในโปรแกรม เช่น Dim เป็นต้น
5. ชื่อตัวแปรห้ามซ้ำกันถ้าอยู่ในขอบเขต (Scope) เดียวกัน

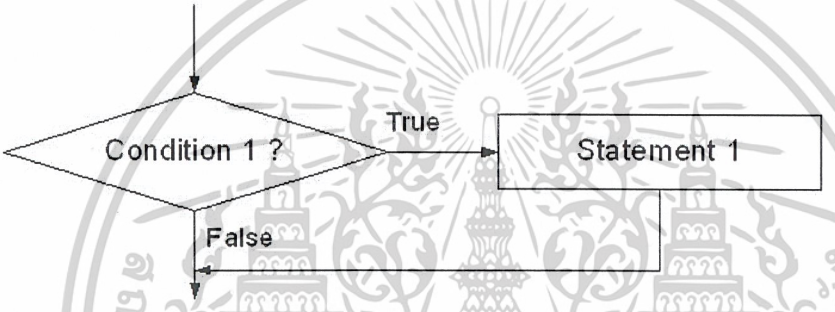
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

-ประโยคแบบเงื่อนไข (Condition Statement)

จะเป็นประโยคคำสั่งที่ใช้การสร้างเงื่อนไขในโปรแกรมเพื่อให้โปรแกรมไปดำเนินการที่ส่วนของโปรแกรมที่เราต้องการเมื่อเกิดเงื่อนไขที่เรากำหนดไว้ คำสั่งประเภทนี้ได้แก่ IF-THEN-ELSE, IF-THEN-ELSE-END IF, IF-THEN-ELSEIF โดยมีรายละเอียด ดังนี้

IF-THEN-ELSE

คำสั่ง IF-THEN เป็นคำสั่งที่ใช้เมื่อต้องการให้คอมพิวเตอร์ตัดสินใจว่าจะไปทำงานในโปรแกรมส่วนใดจากทางเลือกและข้อกำหนดที่เราได้เตรียมไว้ให้ในโปรแกรม รูปแบบของคำสั่ง IF-THEN จะมีลักษณะ ดังนี้



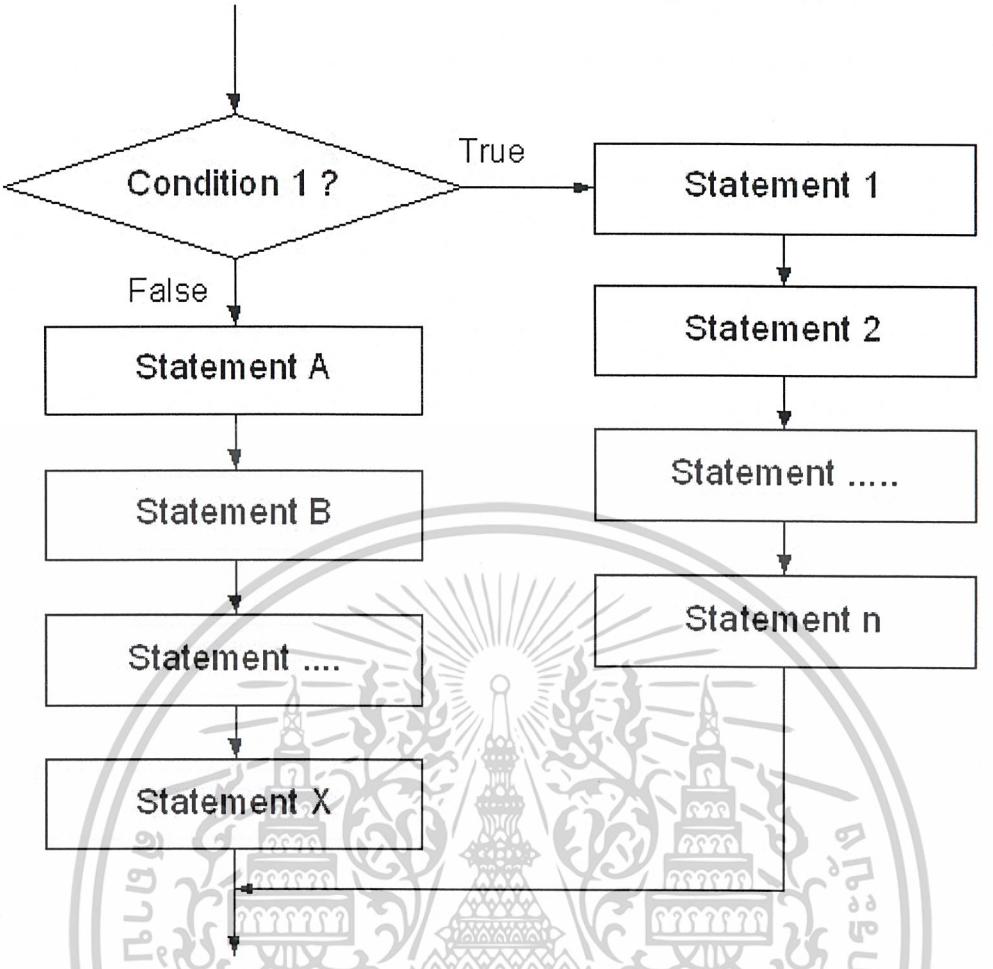
รูปที่ 2.8 Flow chart IF-THEN-ELSE

การทำงานของโปรแกรมจะตรวจสอบ Condition1 ว่าเป็นจริงหรือเท็จ ถ้าเป็นจริงโปรแกรมจะวิ่งไปทำงานที่ Statement1 แต่ถ้า Condition1 เป็นเท็จ โปรแกรมจะข้าม Statement1 และวิ่งไปทำงานบรรทัดถัดไปแทน หรือ If (condition1) THEN Statement1 ELSE Statement2เพื่อให้ทำงาน Statement2 ในกรณีที่ Condition1 เป็นเท็จด้วยก็ได้

IF-THEN-ELSE-END IF

จะเหมือนกับ IF-THEN-ELSE เพียงแต่สามารถมี Statement ที่จะทำงานมากกว่าหนึ่งบรรทัดได้ (IF-THEN มี Statement ได้เพียงหนึ่งคำสั่งหลัง THEN และหน่งคำสั่งหลัง ELSE และ จะต้องอยู่ในบรรทัดเดียวกันเท่านั้น) รูปแบบของ IF-THEN-ELSE-END IF จะเป็น ดังนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



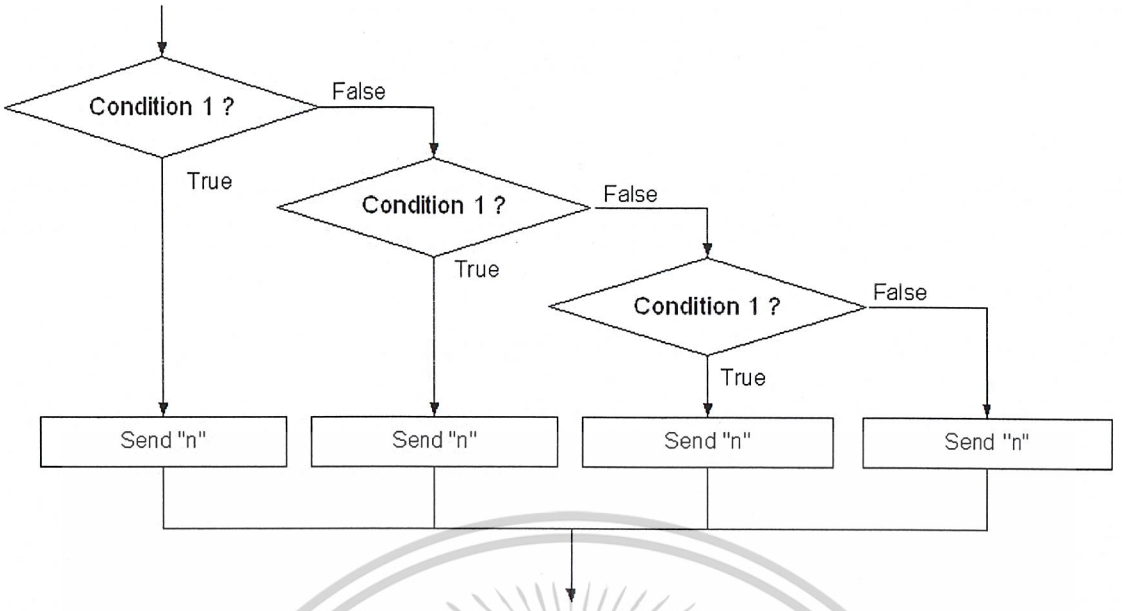
รูปที่ 2.9 Flow chart IF-THEN-ELSE-END IF

โปรแกรมจะตรวจสอบ Condition1 ว่าเป็นจริงหรือเท็จ ถ้า Condition1 เป็นจริงจะวิ่งไปทำงานส่วนของโปรแกรมในบล็อก A จาก Statement1 จนถึงบรรทัด StatementN แล้วจึงทำงานต่อที่คำสั่งหลังบรรทัด END IF ถ้า Condition1 เป็นเท็จจะวิ่งข้ามบล็อก A ไปทำงานส่วนของโปรแกรมในบล็อก B จาก Statement A จนถึงบรรทัด Statement X แล้วจึงทำงานต่อหลังบรรทัด END IF คำสั่งนี้จะใช้โดยไม่มีส่วนของ ELSE เพื่อให้ IF-THEN สามารถทำงานหลาย ๆ คำสั่งได้โดยจะต้องยังคงมี END IF อยู่

IF-THEN-ELSEIF

จะมีรูปแบบการใช้งานเหมือนกับ IF-THEN-ELSE-END IF แต่จะสามารถใส่ condition ซ้อนเข้าไปได้อีก โดยคำสั่งจะมีลักษณะและแผนภาพการทำงาน ดังนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.10 Flow chart IF-THEN-ELSEIF

จะเห็นว่า ถ้า Condition1 เป็นจริงโปรแกรมจะทำงานที่ Block1 หลังจากทำ Block1 เสร็จเรียบร้อยแล้วจะไปทำงานบรรทัดหลัง END IF เลย ไม่มีการตรวจสอบ Condition2 ต่ออีก ถ้า Condition 1 เป็นเท็จ โปรแกรมจึงจะไปตรวจสอบ Condition2 ต่อไป

2.3.5 การจัดการ String และฟังก์ชันที่เกี่ยวข้อง

- การเชื่อมต่อ String 2 ชุดเข้าด้วยกัน

การเชื่อมต่อค่าชุดตัวอักษรหรือตัวแปรแบบ String (Concatenation) จะทำได้สองวิธีคือการใช้เครื่องหมาย & หรือการใช้เครื่องหมาย + ตัวอย่าง เช่น

strC = strA + strB

- ฟังก์ชันสำหรับตัดคำใน String

ฟังก์ชัน *Right ( ), Left ( ) และ Mid ( )*

รูปแบบ Right (*string, length*)

จะเป็นการดึงตัวอักษรทางด้านขวาของ *String* ออกมาเป็นตัวอักษรจำนวนเท่ากับที่กำหนดใน *length*

รูปแบบ Left (*string, length*)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จะเป็นการดึงตัวอักษรทางด้านซ้ายของ *String* ออกมาเป็นตัวอักษรจำนวนเท่ากับที่กำหนดใน *length*

รูปแบบ Mid s*tring,start[, length*])

จะดึงชุดตัวอักษรจาก *String* โดยเริ่มจากตัวอักษรที่ start เป็นจำนวนทั้งหมด *length* ตัวอักษร โดยจะกำหนดค่า *length* หรือไม่ก็ได้ ถ้าไม่กำหนดจะดึงตัวอักษรที่เหลือทั้งหมด

- ฟังก์ชัน *Len ( )*

รูปแบบ Len (*string : varname*)

ใช้ในการหาความยาวของตัวอักษรใน String หรือเนื้อที่ที่ใช้ในหน่วยความจำของตัวแปร

2.3.6 Timer Control และ ฟังก์ชันเกี่ยวกับเรา

Timer Control ที่ทำหน้าที่จับเวลาในโปรแกรม การตั้งตัวแปรแบบวันที่และเวลา ฟังก์ชันต่าง ๆ เกี่ยวกับเวลาที่มีอยู่ใน Visual Basic รวมทั้งคำสั่ง Format ที่ใช้ในการแสดงผลลัพธ์ให้อยู่ในรูปแบบที่เราต้องการ โดยการเรียนรู้เกี่ยวกับ Timer Control จะทำให้เราสามารถนำมาสร้างโปรแกรมที่น่าสนใจ เช่น โปรแกรมแข่งม้าง่าย ๆ

รู้จักกับ Timer Control

ไทม์เมอร์ (Timer) เป็นคอนโทรลที่แสดงเป็นปุ่มรูปนาฬิกาจับเวลาอยู่ใน Toolbox ทำหน้าที่เหมือนตัวจับเวลาที่เราสามารถสั่งให้ทำงานเมื่อครบช่วงเวลาที่กำหนด อย่างเช่นถ้าต้องการให้แสดงเวลาในฟอร์ม ก็ต้องใช้ไทม์เมอร์เพื่อคอยอัพเดทเวลาที่แสดงให้ตรงตามที่นาฬิกาเดิน คอนโทรลไทม์เมอร์นี้จะมองไม่เห็นขณะโปรแกรมทำงาน และ จะปรากฏเฉพาะไอคอนเท่านั้น ในขณะออกแบบ ดังนั้นตำแหน่งของไทม์เมอร์ในฟอร์มจึงไม่ใช่สิ่งสำคัญ แม้ว่าจะมีคุณสมบัติ *Left* และ *Top* ให้กำหนดได้ก็ตาม

คุณสมบัติของไทม์เมอร์

Enabled : จะเป็นพร็อพเพอร์ตี้ที่ใช้ควบคุมการทำงานของไทม์เมอร์ โดยถ้า Enabled มีค่าเป็นจริง จะทำให้ไทม์เมอร์ทำงาน และจะหยุดไทม์เมอร์เมื่อมีค่าเป็นเท็จ

Interval : จะใช้ระบุช่วงเวลาในการทำงานแต่ละครั้ง ค่าของ Interval อยู่ในช่วง 1 ถึง 65,535 มิลลิวินาที (65.535 วินาที) นั่นคือถ้ากำหนดเป็น 1,000 จะมีการทำงานทุก 1 วินาที และ หากกำหนดให้ Interval เป็น 0 ก็จะเหมือนกับสั่งไม่ให้ไทม์เมอร์ทำงานนั่นเอง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Enabled : จะเป็นพร็อพเพอร์ตี้ที่ใช้ควบคุมการทำงานของไทม์เมอร์ โดยถ้า Enabled มีค่าเป็นจริง จะทำให้ไทม์เมอร์ทำงาน และจะหยุดไทม์เมอร์เมื่อมีค่าเป็นเท็จ

เหตุการณ์ Timer

สำหรับเหตุการณ์ Timer นั้นจะมี Event Procedure ชื่อไทม์เมอร์_Timer ทีเป็นหัวใจของการทำงาน โดยในเหตุการณ์นี้จะเป็นโค้ดที่จะถูกเรียกใช้งานเมื่อพร็อพเพอร์ตี้ Enabled เป็นจริงและไทม์เมอร์จับเวลาครบตามที่ตั้งไว้ในพร็อพเพอร์ตี้ Interval เพื่อให้เห็นภาพลองพิจารณาโปรแกรมสำหรับแสดงไทม์เมอร์ โดยสามารถสร้างโปรแกรม ดังต่อไปนี้

ฟังก์ชันเกี่ยวกับเวลา Visual Basic ได้เตรียมฟักชันเกี่ยวกับเวลามาให้หลายฟังก์ชันด้วยกัน โดยรายละเอียด ดังนี้

ฟังก์ชัน Now ใช้ในการดึงค่าเวลาและวันที่จากเวลาในระบบของเครื่อง

ฟังก์ชัน Date ใช้ในการตั้งวันที่ในเครื่องคอมพิวเตอร์

ฟังก์ชัน Year (DateVariable) ใช้ในการหาค่าปีจากตัวแปรชนิดวันที่ DateVariable

ฟังก์ชัน Month (DateVariable) ใช้ในการหาค่าเดือนเป็นตัวเลข 1 – 12 จากตัวแปรชนิดวันที่ DateVariable

ฟังก์ชัน Day (DateVariable) ใช้ในการหาค่าวันที่เป็นตัวเลข 1 – 31 จากตัวแปรชนิดวันที่ DateVariable

ฟังก์ชัน WeekDay (DateVariable [, firstday of Week]) ใช้ในการหาวันในสัปดาห์จากตัวแปรชนิดวันที่ DateVariable ค่าที่ส่งกลับมาจะเป็นตัวเลข 1 – 7 โดยสามารถกำหนดพารามิเตอร์ firstdayofweek สำหรับค่าที่เป็นวันแรกของสัปดาห์ได้ ถ้าไม่กำหนดจะได้ค่า 1 – อาทิตย์, 2 – จันทร์, .. , 7 – เสาร์

ฟังก์ชัน Time ใช้ในการดึงค่าเวลาจากเครื่องคอมพิวเตอร์

ฟังก์ชัน Hour (time) ใช้ในการหาค่าชั่วโมงเป็นตัวเลข 0 – 23 จากตัวแปรชนิดเวลา (time)

ฟังก์ชัน Minute (time) ใช้ในการหาค่านาทีเป็นตัวเลข จากตัวแปรชนิดเวลา (time)

ฟังก์ชัน Second (time) ใช้ในการหาค่าวินาทีเป็นตัวเลข จากตัวแปรชนิดเวลา (time)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ฟังก์ชัน TimeValue (String) ทำการแปลงค่าตัวแปรแบบ string ให้เป็นตัวแปรแบบเวลา

ฟังก์ชัน TimeSerial (hour, minute, second)

hour เป็นค่าชั่วโมง มีค่าระหว่าง 0-23

minute เป็นค่านาที มีค่าระหว่าง 0-59

second เป็นค่าวินาที มีค่าระหว่าง 0-59

จะได้ค่าคืนกลับมาเป็นตัวแปรแบบเวลาจากค่าชั่วโมง, นาที, วินาที (hour, minute, second)

ฟังก์ชัน DateSerial (year, month, day)

Year เป็นค่าปี มีค่าตั้งแต่ 100 ถึง 9999

month เป็นค่าเดือน

day เป็นค่าวันที่

จะได้ค่าคืนกลับมาเป็นตัวแปรแบบวันที่จากตัวแปรค่าปี, เดือน, วัน (year, month, day)

คำสั่ง Format

คำสั่ง Format จะมีประโยชน์อย่างยิ่งในการจัดค่าตัวเลข วันที่ หรือ เวลาให้อยู่ในรูปแบบที่สวยงามตามที่ผู้ใช้ต้องการในการแสดงผล โดยรูปแบบของคำสั่งจะมีดังนี้

รูปแบบ Format *(expression[, format, firstdayofweek[, firstweekofyear]]])*

expression : ข้อมูลที่ต้องการใช้แสดงผล

format : คือรูปแบบที่ต้องการใช้แสดงผล ใส่หรือไม่ก็ได้

firstdayofweek : คือค่าคงที่ที่ใช้กำหนดการแสดงผลเป็นวันแรกของสัปดาห์ ใส่หรือไม่ก็ได้

firstweekofyear : คือค่าคงที่ที่ใช้กำหนดการแสดงผลเป็นสัปดาห์ของปี ใส่หรือไม่ก็ได้

พารามิเตอร์ format มีรูปแบบมากมายให้ใช้โดยขึ้นอยู่กับชนิดของข้อมูลที่เป็นผลมาจาก expression ด้วยดังนี้

การแสดงผลข้อมูลชนิดวันที่และเวลา

พารามิเตอร์ format	**ผลที่ได้**
General Date	ใช้แสดงข้อมูลในรูปวันที่และเวลาตามที่กำหนดไว้ในระบบของเครื่องคอมพิวเตอร์ (Control Panel ▶ Regional Setting ▶ Dater/Time)

ตารางที่ 2.10 ข้อมูลชนิดวันที่และเวลา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

พารามิเตอร์ format	ผลที่ได้
Long Date	ใช้แสดงข้อมูลในรูปวันที่แบบ Long Date ตามที่กำหนดไว้ในระบบของเครื่อง
Medium Date	ใช้แสดงข้อมูลในรูปวันที่แบบ Medium Date ตามที่กำหนดไว้ในระบบของเครื่อง
Short Date	ใช้แสดงข้อมูลในรูปวันที่แบบ Short Date ตามที่กำหนดไว้ในระบบของเครื่อง
Long Time	ใช้แสดงข้อมูลในรูปเวลาแบบ Long Time (HH:MM:SS) เช่น 22:41:00
Medium Time	ใช้แสดงข้อมูลในรูปเวลาแบบ Medium Time (HH:MM AM/PM) เช่น 10:41PM
Short Time	ใช้แสดงข้อมูลในรูปเวลาแบบ Short Time (HH:MM แบบ 24 ชั่วโมง) เช่น 22:41
H หรือ HH	ใช้แทนค่าชั่วโมงของเวลา (ถ้า HH จะมีเลข 0 นำหน้ากรณีที่มีค่าเพียงหลักเดียว)
M หรือ MM	ใช้แสดงนาทีของเวลา (ถ้า MM จะมีเลข 0 นำหน้ากรณีที่มีค่าเพียงหลักเดียว)
S หรือ SS	ใช้แสดงวินาทีของเวลา (ถ้า SS จะมีเลข 0 นำหน้ากรณีที่มีค่าเพียงหลักเดียว)
D	ใช้แสดงเลขวันที่ของวันที่
M	ใช้แสดงเลขเดือนของวันที่
Y	ใช้แสดงเลขปีของวันที่

ตารางที่ 2.10 (ต่อ) ข้อมูลชนิดวันที่และเวลา

การแสดงผลข้อมูลชนิดตัวเลข

พารามิเตอร์ format	ผลที่ได้
0	ใช้แทนตัวเลข ในกรณีที่ไม่มีตัวเลข ณ ตำแหน่งนั้นจะแสดงผลเป็นตัวเลข 0 แทน

ตารางที่ 2.11 ข้อมูลชนิดตัวเลข

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

พารามิเตอร์ format	ผลที่ได้
#	ใช้แทนตัวเลข แต่ในกรณีที่ไม่มีค่าตัวเลขที่ตำแหน่งนั้นจะไม่มีการแสดงผล
.	ใช้แทนจุดทศนิยม
%	ใช้แสดงข้อมูลที่เป็นเปอร์เว็นต์พร้อมเครื่องหมาย % (จะคูณ 100 ให้)
,	ใช้แทนเครื่องหมายจุลภาคในการแบ่งตัวเลข
:	ใช้แทนเครื่องหมาย : มักใช้ในการแบ่งเวลา
/	ใช้แทนเครื่องหมาย / มักใช้ในการแบ่งวันที่
+	ใช้แทนเครื่องหมาย +
-	ใช้แทนเครื่องหมาย -
$	ใช้แทนเครื่องหมาย $
(และ)	ใช้แทนเครื่องหมาย (และ)

ตารางที่ 2.11 (ต่อ) ข้อมูลชนิดตัวเลข

การแสดงผลข้อมูลชนิดตัวเลขโดยใช้ Format เป็นข้อความ

พารามิเตอร์ format	ผลที่ได้
General Number	ใช้แทนตัวเลขทั่วไป ไม่มีเครื่องหมายจุลภาคคั่นระหว่างตัวเลข
Currency	ใช้แทนตัวเลขที่เป็นจำนวนเงิน โดยมีเครื่องหมาย, และ ทศนิยมตามที่กำหนดในคอมพิวเตอร์เครื่องนั้น ๆ
Fixed	ใช้แทนตัวเลขโดยไม่มีเครื่องหมายจุลภาคแต่มีจุดทศนิยม 2 ตำแหน่ง
Standard	ใช้แทนตัวเลขโดยมีเครื่องหมายจุลภาคแต่มีจุดทศนิยม 2 ตำแหน่ง
Percent	ใช้แสดงข้อมูลที่เป็นเปอร์เซ็นต์พร้อมเครื่องหมาย % จะคูณ 100 ให้
Scientific	ใช้แสดงตัวเลขในรูปตัวเลขทางวิทยาศาสตร์ เช่น 1E-10
Yes/No	แสดงคำว่า No ถ้าข้อมูลเป็น 0 และ แสดงคำว่า Yes เมื่อข้อมูลเป็น 1
True/False	แสดงคำว่า False ถ้าข้อมูลเป็น 0 และ แสดงคำว่า True เมื่อข้อมูลเป็น 1
On/Off	แสดงคำว่า Off ถ้าข้อมูลเป็น 0 และ แสดงคำว่า On เมื่อข้อมูลเป็น 1

ตารางที่ 2.12 ข้อมูลชนิดตัวเลขโดยใช้ Format เป็นข้อความ

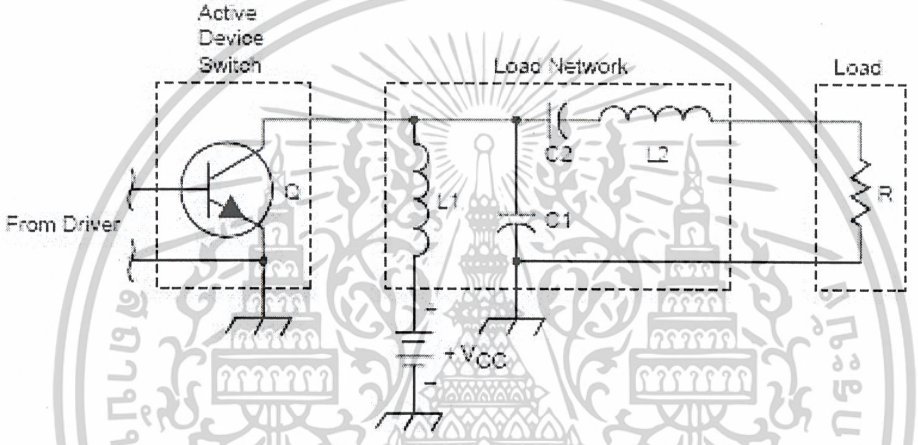
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.4 วงจรขยายคลาส E

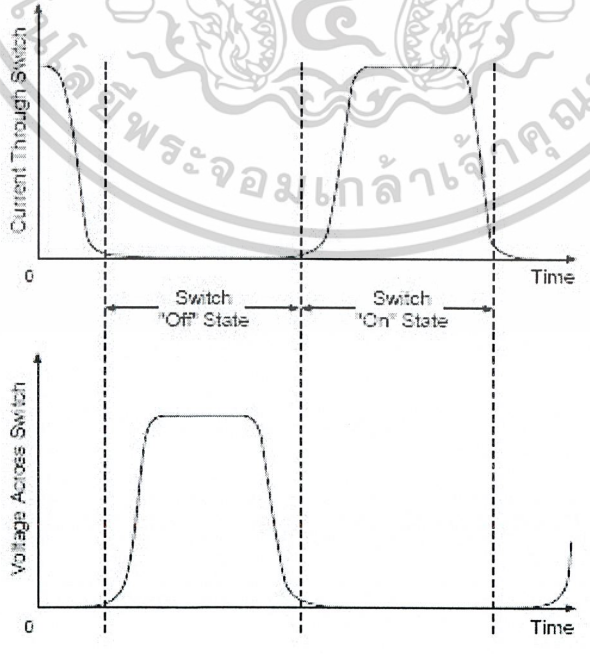
วงจรขยายคลาส E เป็นวงจรที่มีประสิทธิภาพมาก ใช้ขยายในความถี่ใดความถี่หนึ่งหรือกำหนดให้ขยายเฉพาะความถี่ที่กำหนดไว้ ทรานซิสเตอร์จะทำงานในลักษณะสวิตช์

ในสภาวะที่ทรานซิสเตอร์ทำงานที่ขา collector ของทรานซิสเตอร์จะมีค่าแรงดันเข้าใกล้ศูนย์ จึงไม่มีกระแสไหลไปยังโหลด

ในสภาวะที่ทรานซิสเตอร์ไม่ทำงานที่ขา collector ของทรานซิสเตอร์จะมีค่าแรงดันสูงเข้าใกล้ VCC จึงเกิดกระแสไหลไปยังโหลด แต่ในการทำงานจริงค่าแรงดันและกระแสจะไม่ใช่ค่าสูงสุดของแหล่งจ่าย เนื่องจากจะเกิดการสูญเสียภายในทรานซิสเตอร์เองซึ่งในรูปแสดงกราฟในอุดมคติของทรานซิสเตอร์โดยแรงดันกับกระแสใช้ในการเปรียบเทียบ แสดงในรูปที่ 2.11



รูปที่ 2.11 วงจรขยายคลาส E



รูปที่ 2.12 ทรานซิสเทอร์ โวล์ทเตจ และกระแส

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การทำงานของวงจร

ในรูปที่2.10 L1ใช้สำหรับให้กระแสไฟ D.C. ไหลผ่านในวงจร C1 ทำหน้าที่หน่วงแรงดันที่ขา collector ในขณะที่ทรานซิสเตอร์อยู่ในสภาวะจาก ON-OFF และที่ C2,L2,R ออกแบบให้ขาcollector มีแรงดันลดลงถึงศูนย์ เมื่อทรานซิสเตอร์อยู่ในสภาวะจาก OFF-ON

โดยเราสามารถคำนวณค่าอุปกรณ์ต่างๆได้จากสมการ

Load Resistor R.

$$R_L = 0.577 \cdot \frac{(U_{cc} - U_{cesat})^2}{P_S} \quad (2.1)$$

Capacitor C1.

$$C_1 = \frac{0.2}{\omega \cdot R_L} \quad (2.2)$$

Inductor L1.

$$L_1 \geq \frac{10}{\omega^2 \cdot C_1} \quad (2.3)$$

Inductor L2.

$$L_2 = \frac{Q_L \cdot R_L}{\omega} \quad (2.4)$$

Capacitor C2.

$$\omega \cdot L_2 - \frac{1}{\omega \cdot C_2} = 1.1525 \cdot R_L \quad (2.5)$$

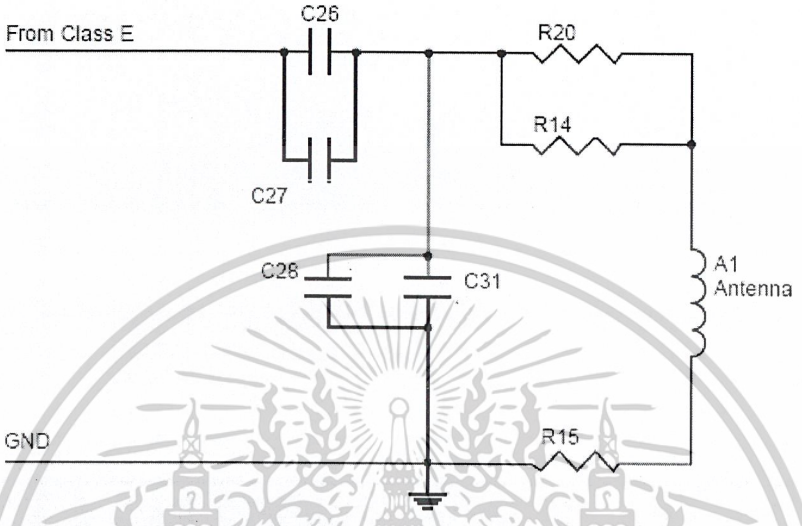
Where:

P_S is the output power
U_{cc} is the supply voltage
U_{cesat} is the saturation voltage of the transistor
$\omega$is the signal frequency

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.5 การMATCHING CIRCUIT

การทำสายอากาศให้เข้ากับวงจร แทนที่โหลดความต้านทานที่ได้จาก วงจรClass E Amplifier โดยจะต้องทำให้สายอากาศมีค่าตรงกับโหลด สามารถปรับได้โดยการใช้ตัวเก็บประจุ C26,C27 และ C28,C31 และ R14,R15,R20 ดูจากรูปที่ 2.12



รูปที่ 2.13 MATCHING CIRCUIT

ดังนั้นเราสามารถหาค่า Capacitor ที่ต่ออนุกรมในวงจร เราสามารถคำนวณได้จากสมการที่ 2. 6

Serial Capacitor C_s

$$C_S = \frac{1}{\sqrt{Z \cdot R_P} \times \omega} \qquad (2.6)$$

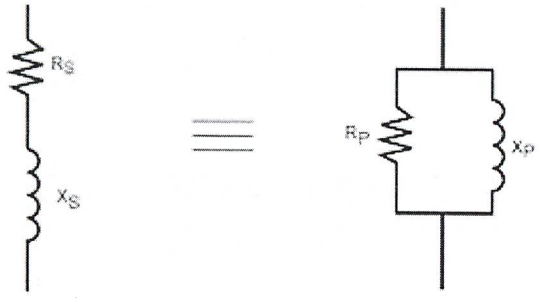
Parallel Capacitor C_P

$$C_P = \frac{1}{L\omega^2} - C_S \qquad (2.7)$$

where:

ω is 13.56MHz
L is the inductance of the antenna loop
Z equals R_L of the Class E Amplifier

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



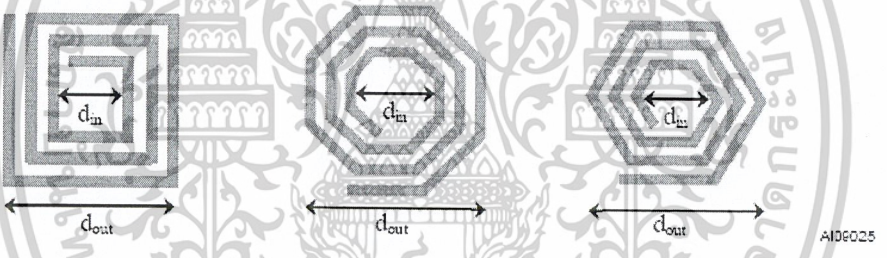
รูปที่ 2.14 SERIES TO PARALLAL CIRCUIT

สามารถคำนวณโดยใช้สมการดังต่อไปนี้

Parallel Resistor Rp.

$$R_P = \frac{X_S^2}{R_S} \qquad (2.8)$$

2.6 สายอากาศ และ การหาค่า Inductance



รูปที่ 2.15 Planar Square, Hexagonal and Octagonal Spiral Coils

การหาค่า Inductance และ สายอากาศ สามารถคำนวณได้จากสมการนี้

Inductance of Planar Square, Hexagonal and Octagonal Coil

$$L = K_1 \times \mu_0 \times N^2 \times \frac{d^2}{1 + K_2 \cdot \rho} \qquad (2.9)$$

Where:

d = (dout+din)/2

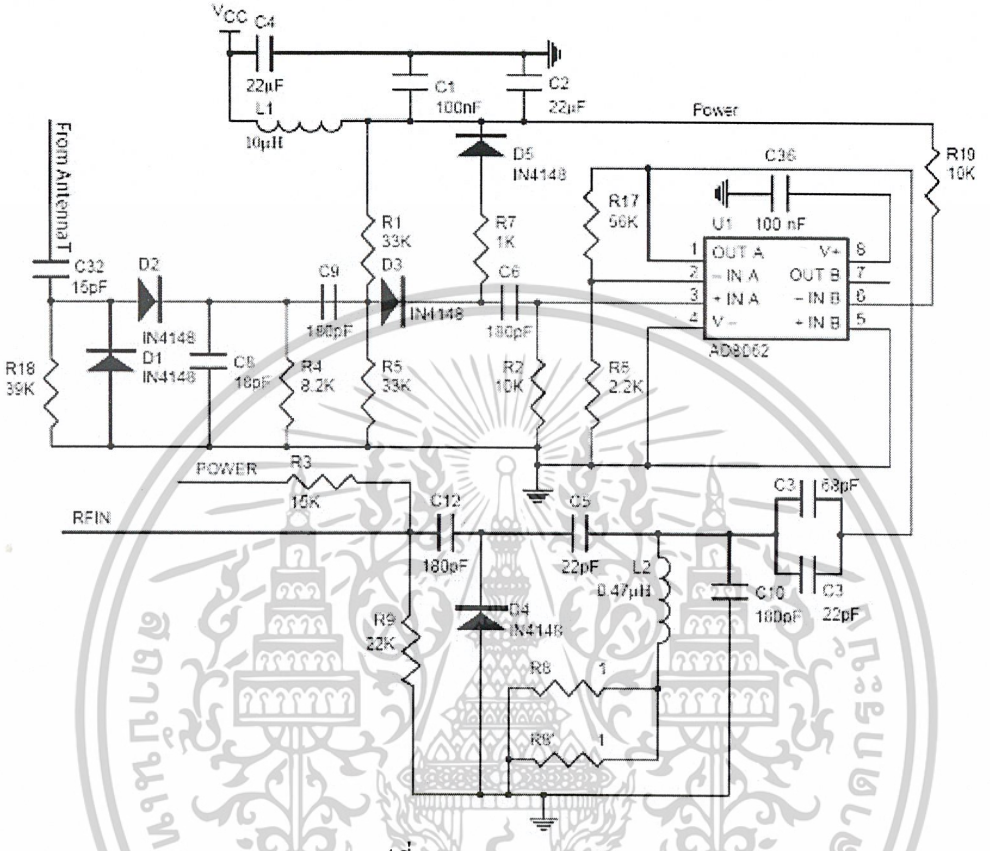
p = (dout–din)/(dout+din)

K1 and K2 depend on the layout (K1 K2; Square 2.34 2.75 Hexagonal 2.33 3.82 Octagonal 2.25 3.55)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.7 วงจรภาครับ

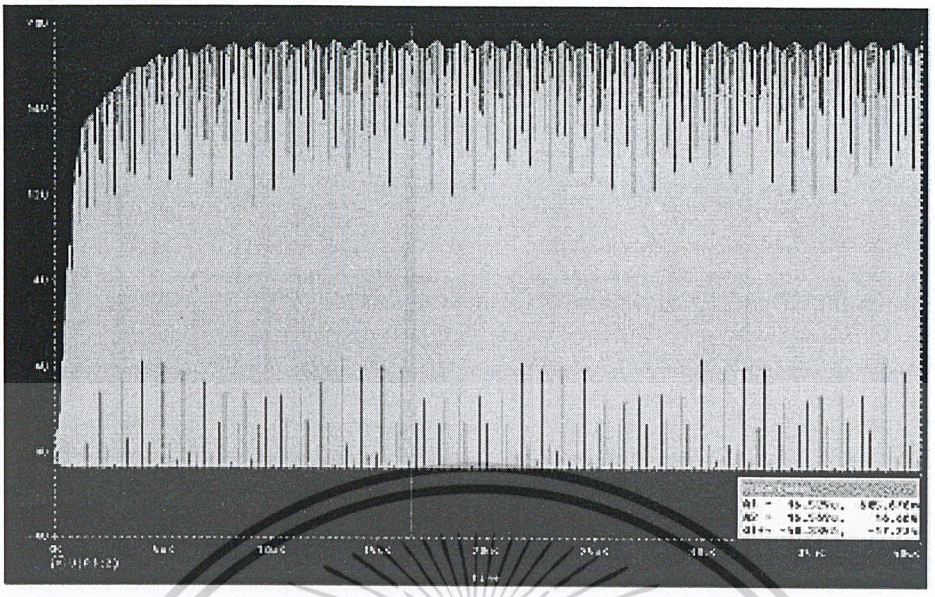
สัญญาณจากสายอากาศ จะส่งไปถึง CRX14 หลังจากภาคกรองความถี่และขยายสัญญาณตามวงจร ในรูปที่ 2.12



รูปที่ 2.16 วงจรภาครับ

C32 , R18 , D1 ทำหน้าที่ปรับสัญญาณที่มาจากสายอากาศ โดยตัดสัญญาณในส่วนของ ซีกลบออก ดังในรูปที่ 2.13

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



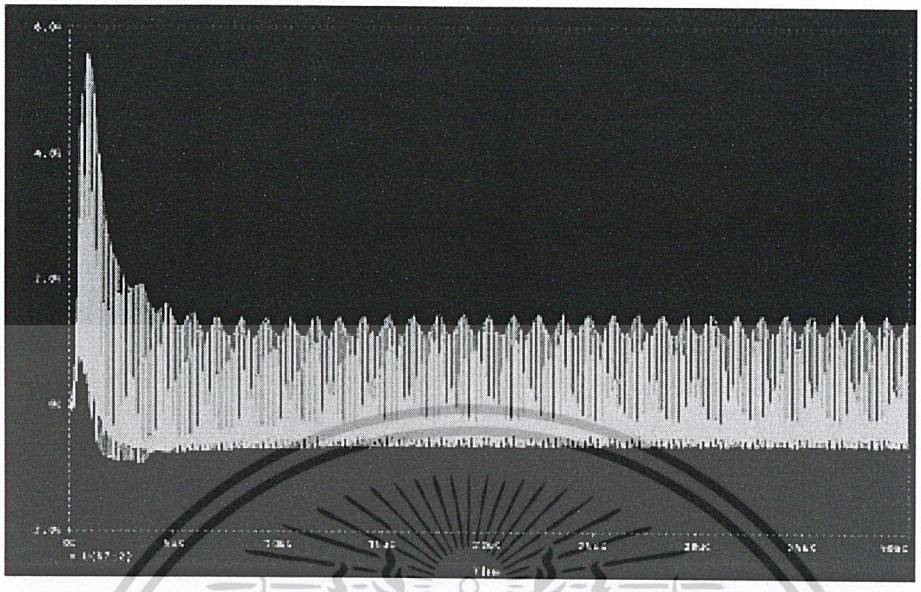
รูปที่ 2.17 สัญญาณที่รับมาจากสายอากาศที่ถูกตัดซีกลบออก

D2, C8, R4 ทำหน้าที่ตัดสัญญาณในช่วงฐานของความถี่ 13.56 MHz. ทำให้สัญญาณลดลงดังในรูปที่ 2.14

รูปที่ 2.18 สัญญาณในช่วงฐานของความถี่ 13.56 MHz.

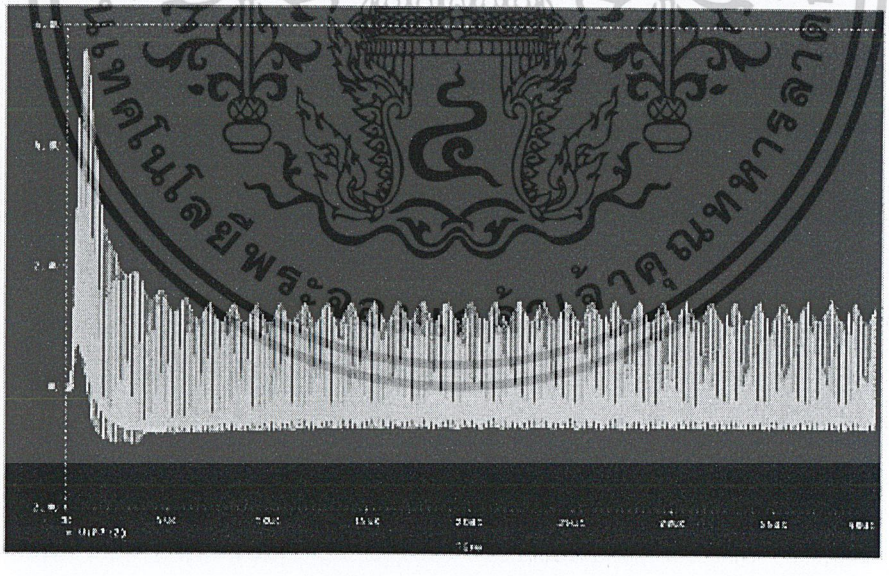
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

C9, R1 และ R5 เป็นตัวทำให้สัญญาณอยู่ในระดับ Vcc/2 ดังในรูปที่ 2.15



รูปที่ 2.19 ปรับสัญญาณให้อยู่ระดับ Vcc/2

หลังจากนั้นสัญญาณจะถูกทำให้อยู่ที่จุด OV โดยใช้ C6 และ R2 ดังในรูปที่ 2.16



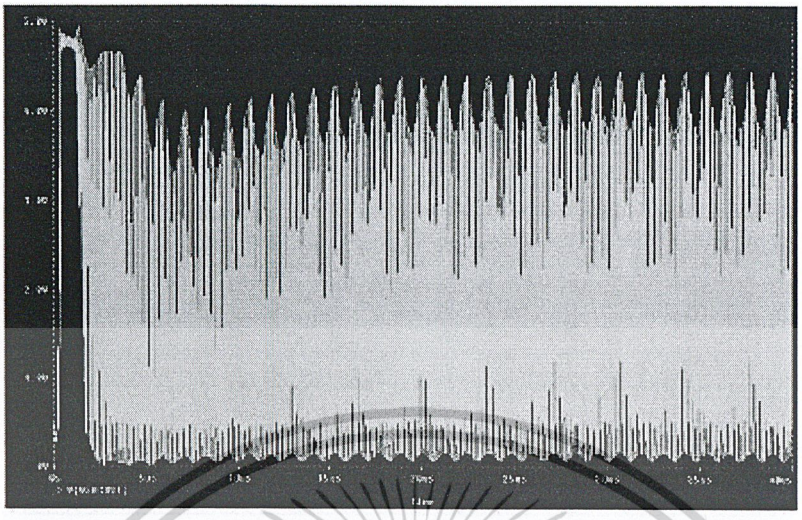
รูปที่ 2.20 สัญญาณจะถูกทำให้อยู่ที่จุด O V

AD8062 เป็น OP - Amp ที่ใช้ในการขยายแบบ non - nivreting

$$\text{Gain} = 1 + \frac{R17}{R16} \qquad (2.10)$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ดังนั้นสัญญาณจะถูกขยายเพิ่มขึ้นที่ความถี่ 13.56 MHz. ดังในรูปที่ 2.17



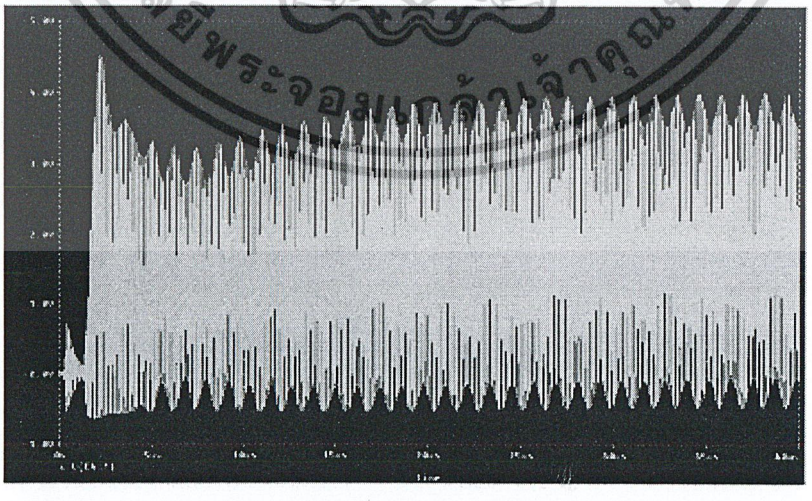
รูปที่ 2.21 สัญญาณจะถูกขยายเพิ่มขึ้นด้วย OP AMP

หลังจากที่ขยายสัญญาณให้ใหญ่ขึ้น สัญญาณจะถูกกรองโดยวงจรกรองความถี่ ที่ 14.4MHz กรองโดยใช้ C3, C10, L2, R8 ตอบสนองความถี่สามารถคำนวณได้ดังนี้

Resonant Frequency ω.

$$\omega = \frac{1}{\sqrt{L_2(C_3 + C_3^{'} + C_{10})}} \qquad (2.11)$$

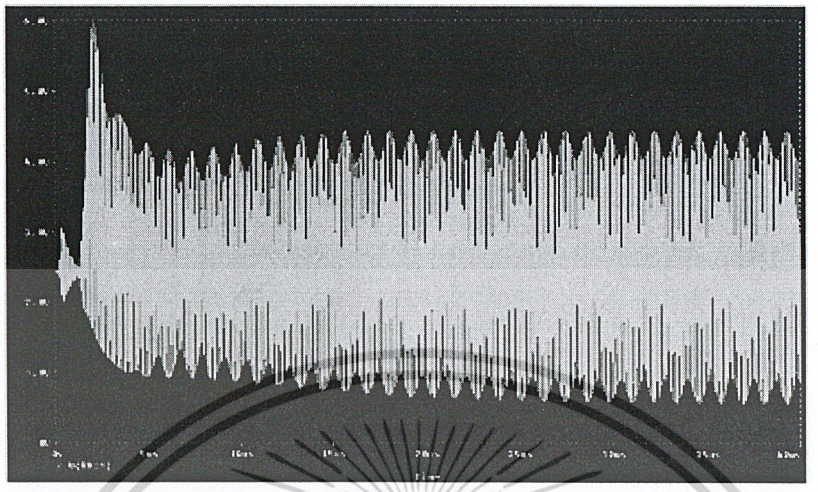
สัญญาณด้านลบถูกตัดโดย C5 และ D4 ดังในรูปที่ 2.18



รูปที่ 2.22 สัญญาณด้านลบจะถูกตัดออก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ท้ายที่สุดสัญญาณจะถูกยกไปที่จุดศูนย์กลาง Vcc/2 โดย C12, R3, R9 และจะส่งไปที่ CRX14 แล้วทำการถอดข้อมูลออกมากจาก Tag

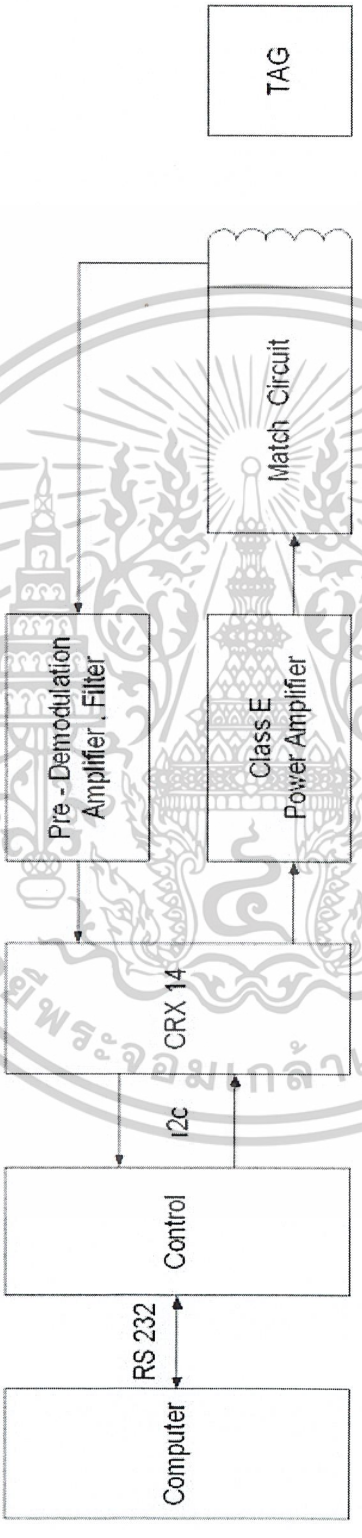


รูปที่ 2.23 สัญญาณจะถูกยกไปที่ Vcc/2

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 3
ขั้นตอนการออกแบบ

ในระบบ RFID สามารถแยกภาคการทำงานได้เป็นบล๊อกไดแกรมดังรูปที่ 3.1 ในการออกแบบจะทำการออกแบบเป็นบล๊อกๆ



รูปที่ 3.1 บล๊อกไดแกรมของระบบ RFID

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.1 การออกแบบบล๊อกคอนโทรล

ในการออกแบบนั้นจะใช้ไมโครคอนโทนเลอร์ตระกูล MCS 51ในการติดต่อกับคอมพิวเตอร์ผ่านพอร์ต RS – 232 และใช้ไมโครคอนโทนเลอร์ ติดต่อกับชิป CRX 14 โดยติดต่อผ่านระบบบัส I^2C

การออกแบบไมโครคอนโทรลเลอร์ ติดต่อกับคอมพิวเตอร์นั้น จำเป็นต้องมี IC แปลงระดับสัญญาณ ซึ่งในวงจรนี้ใช้ MAX232 ทำหน้าที่แปลงระดับสัญญาณ วงจรเชื่อมต่อไมโครคอนโทนเลอร์ติดต่อกับคอมพิวเตอร์ แสดงดังรูปที่ 2.4

3.2 การคำนวณหาค่าอุปกรณ์ในวงจร Class E Amplifier

ในการหาค่าอุปกรณ์ กำหนดให้ P = 1 W , VCC = 9V , Vcesat = 4V และ ความถี่เท่ากับ 13.56MHz สามารถคำนวณได้ดังนี้

สมาการที่ 2.1 RL = 24.5 Ω

สมาการที่ 2.2 C1 = 96.16 pF

สมาการที่ 2.3 L1 = 4.32 uH

สมาการที่ 2.4 L2 = 1.146 uH

สมาการที่ 2.5 C2 = 169pF

3.3 การคำนวณหาค่าอุปกรณ์ในวงจรสายอากาศ

ในการหาค่าอุปกรณ์ กำหนดให้ Z มีค่า 24.5 Ω, Rs มีค่า 0.5 Ω และ inductance ของสายอากาศมีค่า 0.9 uH ดังนั้น

สมาการที่ 2.6 RP = 11760 Ω

สมาการที่ 2.7 CS = 22 pF

สมาการที่ 2.8 CP = 131pF

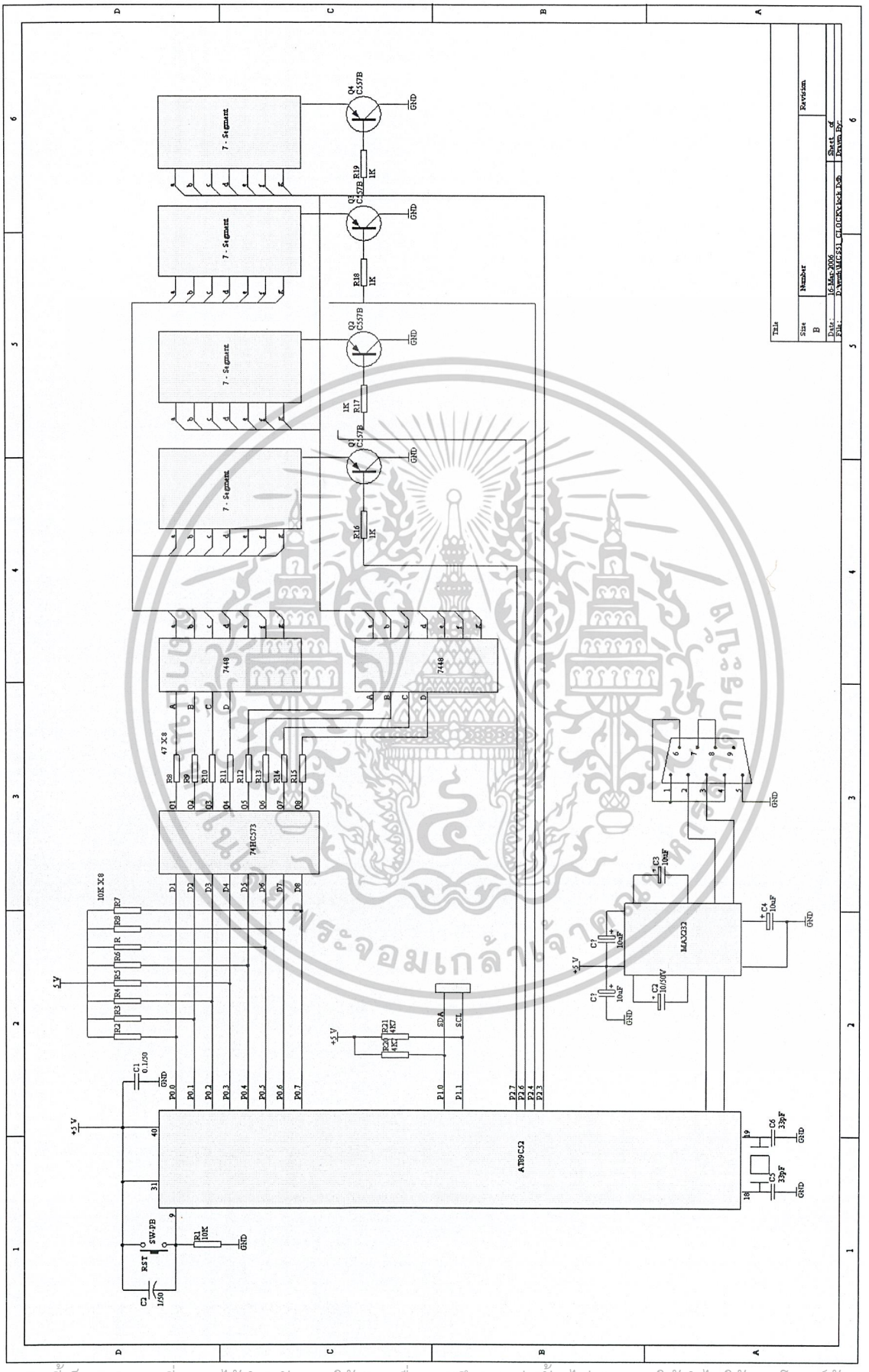
3.4 การคำนวณหาค่า Inductance ในสายอากาศ

ในการหาค่า Inductance กำหนดให้ d = 66.93mm, p = 0.074, K1= 2.34, K2 = 2.75 ดังนั้น

สมาการที่ 2.9 L = 0.9 uH

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

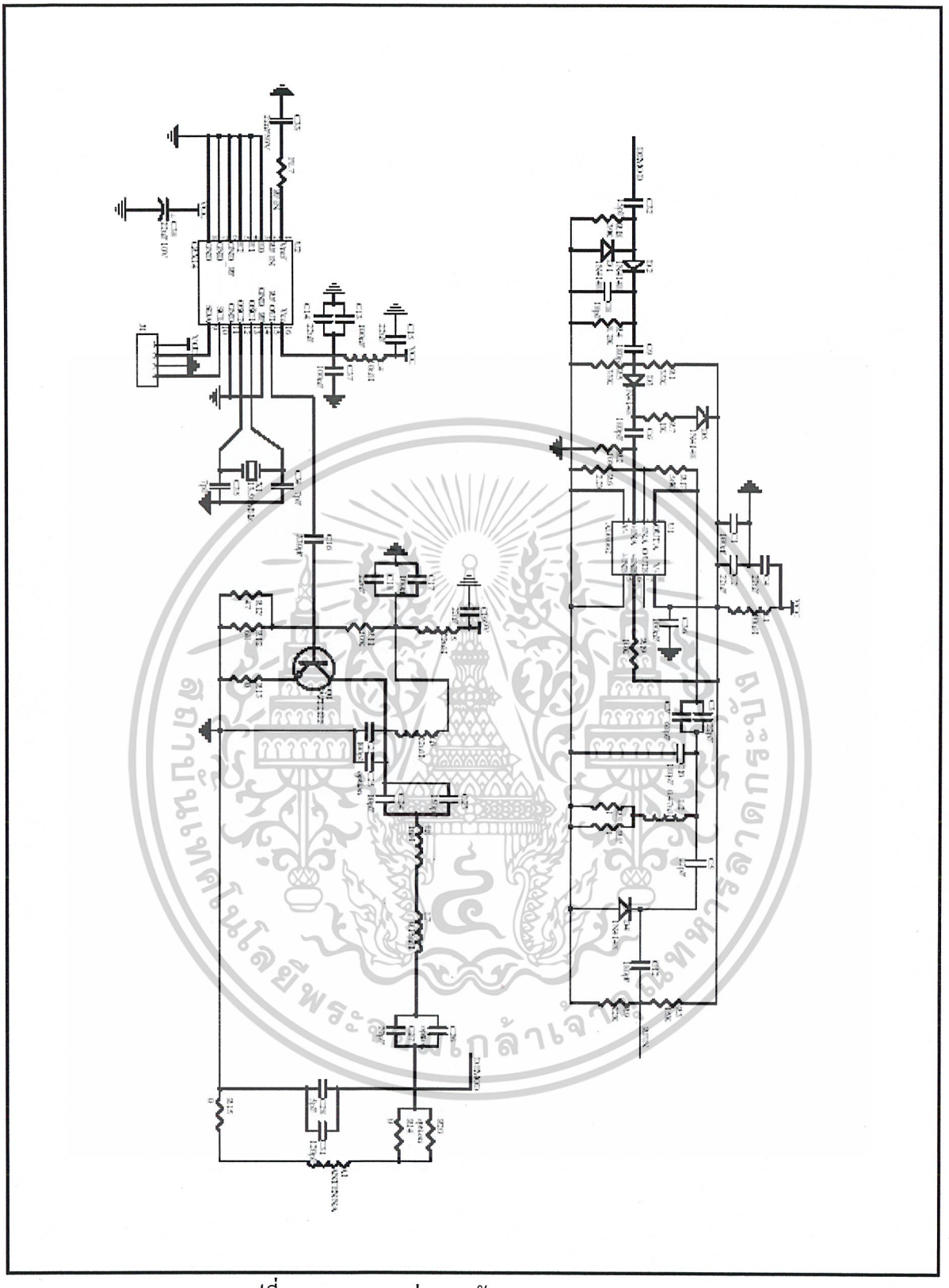
3.5 วงจรทั้งหมดที่ได้ในการออกแบบ



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

รูปที่ 3.2 วงจรภาคคอนโทรล

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.3 วงจร ภาคส่ง ภาครับ และวงจร Matching

3.6 โปรแกรมควมคุมการทำงาน

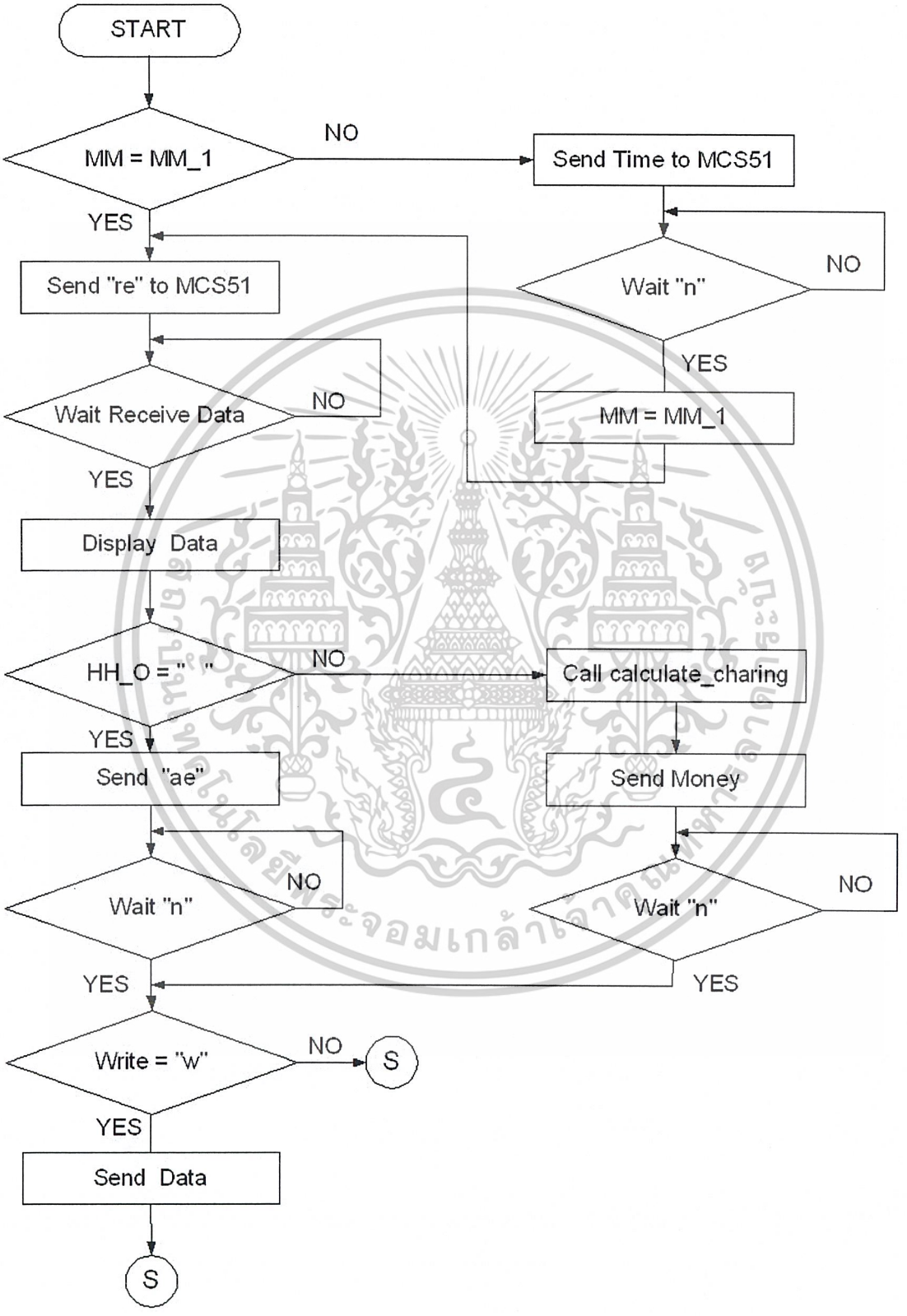
การออกแบบโปรแกรมเพื่อควบคุมการทำงานแบ่งได้เป็นสองส่วนด้วยกันคือ

- ส่วนที่ใช้ติดต่อกับผู้ใช้งาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- ส่วนที่ใช้ควบคุมเครื่อง Reader and Writer

3.6.1 ส่วนที่ใช้ติดต่อกับผู้ใช้งานจะมี Flowchart การทำดังนี้



รูปที่ 3.3 Flowchart การทำของโปรแกรมที่ที่ติดต่อกับผู้ใช้งาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Source Code โปรแกรมที่ที่ติดต่อกับผู้ใช้งาน

```
Dim next_1 As Boolean
Dim MM_1, Dinput, Sinput, Car_license_Text_old, HH_O, MM_O, Read_D, Write_D As String
Dim HOUSE_BASE, MINUTES_BASE, Input_Len, Money As Integer

Private Sub Form_Load()
MSComm1.CommPort = 1
MSComm1.Settings = "9600,n,8,1"
MSComm1.PortOpen = True

End Sub

Private Sub Timer1_Timer()
   Timer1.Enabled = False
   Call Send_time_And_Read_Data
End Sub

Private Sub Send_time_And_Read_Data()
   HOUSE_BASE = Hour(Time)
   MINUTES_BASE = Minute(Time)
   Time_Base.Caption = "Time In      " + Format(Time, "HH:MM:SS")
   Date_1.Caption = Format(Date, "Long Date")
   HH.Caption = HOUSE_BASE
   MM.Caption = MINUTES_BASE
   If MM = MM_1 Then
      Else
      MM_1 = MM
      MSComm1.Output = "s"
      Call send_Time
      MSComm1.Output = "e"
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        Do While MSComm1.Input <> "n"
        Loop
    End If
If Read_D = "r" Then
    Read_D = ""
    MSComm1.Output = "re"
        Do While Input_Len < 10
            Sinput = MSComm1.Input
            Dinput = Dinput + Sinput
            Input_Len = Len(Dinput)
        Loop
        Input_Len = 0
        Car_license_Text_out.Caption = "Car license out   " + Mid(Dinput, 2, 6)
        Time_Out = Mid(Dinput, 9, 4)
        HH_O = Left(Time_Out, 2)
        MM_O = Right(Time_Out, 2)
        Time_Out.Caption = "Time Out        " + HH_O + ":" + MM_O
        Dinput = ""

        If HH_O <> "" Then
           Call calculate_charing
           If Car_license_Text_out.Caption = Car_license_Text_old Then
                MSComm1.Output = "ae"
              Else
                 If Money = 0 Then
                    MSComm1.Output = "m"
                    MSComm1.Output = Money_D.Caption
                    MSComm1.Output = "0"
                    MSComm1.Output = "e"
                   Else
                    MSComm1.Output = "m" + Money_D.Caption + "e"
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
            End If
            Car_license_Text_old = Car_license_Text_out.Caption
        End If
    Else
      MSComm1.Output = "ae"
    End If

    Do While MSComm1.Input <> "n"
    Loop
Else
End If

If Write_D = "w" Then
        MSComm1.Output = "w"
        Call send_Time
        MSComm1.Output = Car_license_Text.Text
        MSComm1.Output = "e"
        Write_D = ""
  Do While MSComm1.Input <> "n"
  Loop

Else
End If

  MSComm1.Output = "de"
  Do While MSComm1.Input <> "n"
  Loop

  Timer1.Enabled = True

End Sub
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
Sub send_Time()

If HOUSE_BASE = 0 Then
   MSComm1.Output = "00"
  Else
   If HOUSE_BASE < 10 Then
      MSComm1.Output = "0"
      MSComm1.Output = HH.Caption
   Else
      MSComm1.Output = HH.Caption
   End If
End If

If MINUTES_BASE = 0 Then
    MSComm1.Output = "00"
  Else
   If MINUTES_BASE < 10 Then
      MSComm1.Output = "0"
      MSComm1.Output = MM.Caption
   Else
      MSComm1.Output = MM.Caption
   End If
End If

End Sub
Sub calculate_charing()
House_S = HOUSE_BASE - HH_O
Minutes_S = MINUTES_BASE - MM_O
Money = 0
If House_S > 1 Then
     Money = House_S * 10
      If Minutes_S > 30 Then
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

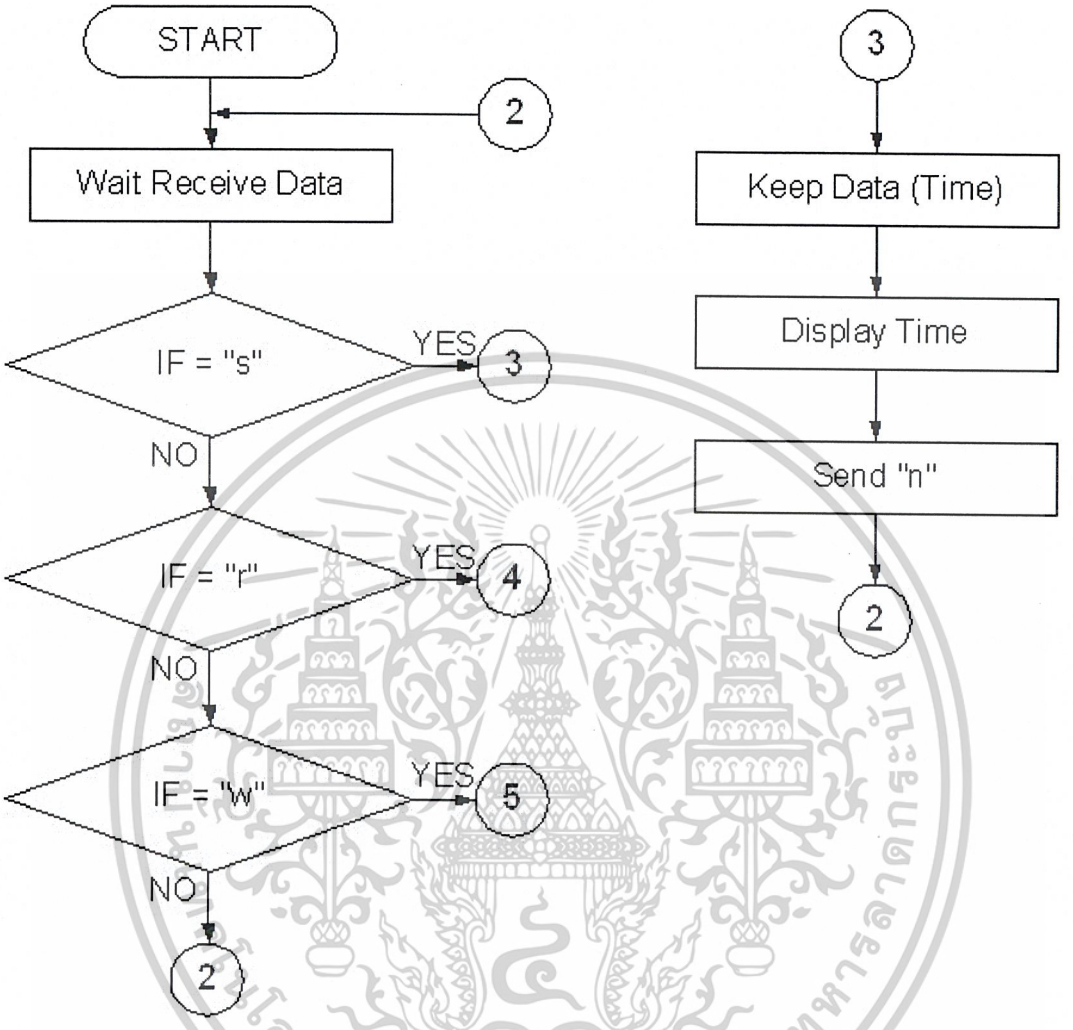
```
            Money = Money + 10
        End If
    Else
        If Minutes_S > 30 Then
            Money = Money + 10
        Else
            Money = Money + 0
        End If
End If
    Money_D.Caption = Money
    Money_s.Caption = "Money     " + Money_D.Caption
End Sub

Private Sub WR_BT_Click()
Write_D = "w"
End Sub

Private Sub RD_BT_Click()
Read_D = "r"
End Sub
```

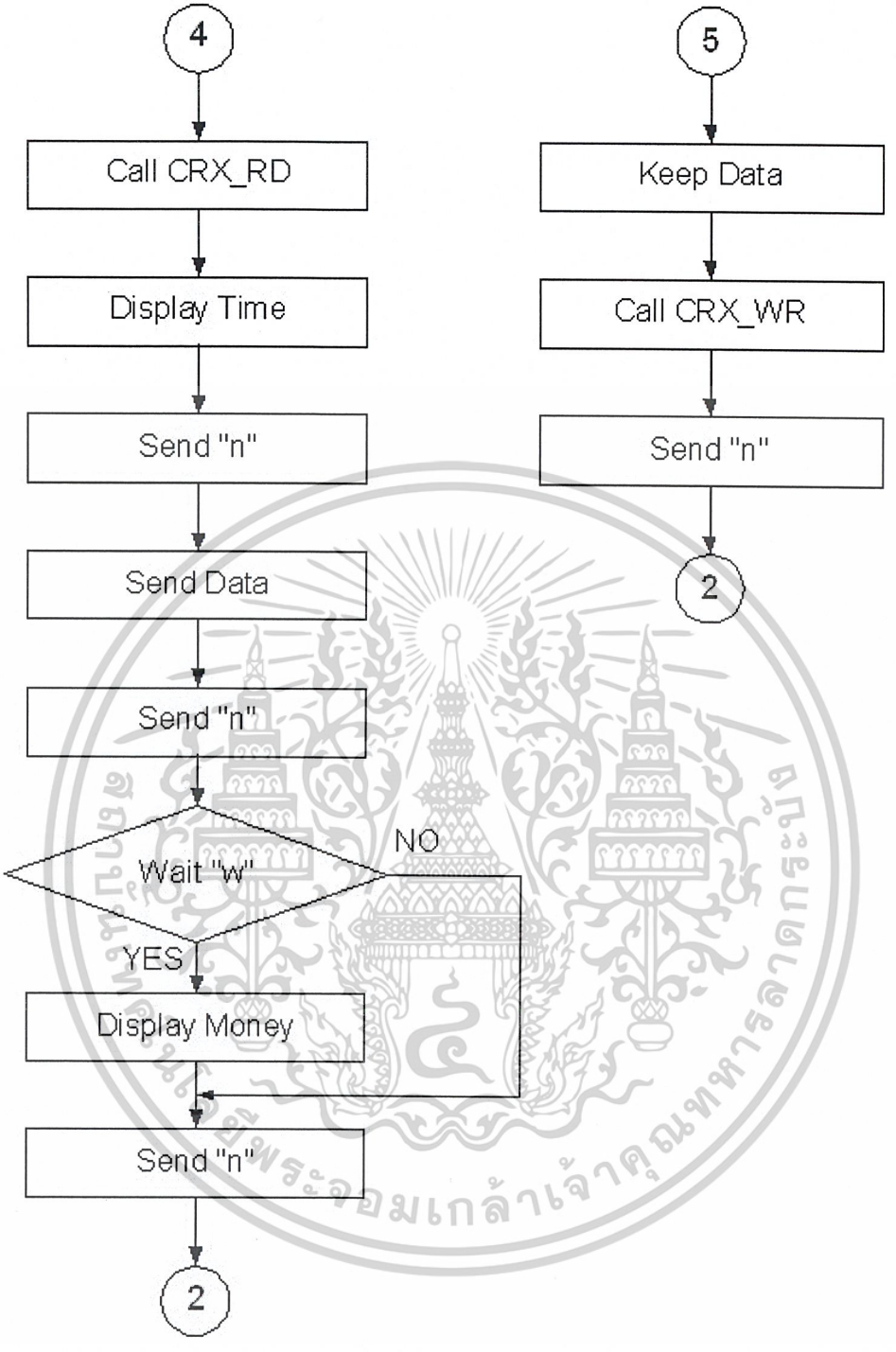
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.5.2 ส่วนที่ใช้ควบคุมเครื่อง Reader and Writer จะมี Flowchart การทำดังนี้



รูปที่ 3.5 Flowchart ควบคุมการทำงานเครื่อง Reader and Writer

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.6(ต่อ) Flowchart ควบคุมการทำงานเครื่อง Reader and Writer

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โปรแกรมควบคุมการทำงานเครื่อง Reader and Writer

```
;-------------------------------------------------------------------------------------------------------
; Define Port&Pin Name
;-------------------------------------------------------------------------------------------------------
SDA             BIT     P1.0            ; SDA I2C Bus
SCL             BIT     P1.1            ; SCL I2C Bus
DISP_H          BIT     P1.2            ; Display 1 and 2
DISP_M          BIT     P1.3            ; Display 3 and 4
Dot_Time        BIT     P1.4
TX              BIT     P3.0            ; Control Delay 1
RX              BIT     P3.1            ; Control Delay 2

;-------------------------------------------------------------------------------------------------------
; Define User Register
;-------------------------------------------------------------------------------------------------------
FLAG            EQU     02FH
I2C_ACK         BIT     FLAG.0
I2C_ADDR        EQU     033H
I2C_DATA        EQU     034H
BUFFER          EQU     030H
Keep_Data       EQU     050H

Car_license_Text1   EQU     051H
Car_license_Text2   EQU     052H
Number_1        EQU     053H
Number_2        EQU     054H
Number_3        EQU     055H
Number_4        EQU     056H
Car_license_END EQU     057H
HOURS_1         EQU     058H
HOURS_2         EQU     059H
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
MINUTES_1       EQU     05AH
MINUTES_2       EQU     05BH
Time_END_R      EQU     05CH
MONEY           EQU     05DH

HOURS           EQU     060H
MINUTES         EQU     061H
TIME_END        EQU     062H
Ti_Data         EQU     063H
End_Ti          EQU     064H


;-------------------------------------------------------------------------------
; Define I2C Slave Address
;-------------------------------------------------------------------------------
CRX_ID          EQU     10100000B
;-------------------------------------------------------------------------------
; Main Program.
;-------------------------------------------------------------------------------
                ORG     0000H
                MOV     P0,#00000000B
                MOV     P1,#00000000B
                MOV     TMOD,#021H
                MOV     TH1,#0FDH
                MOV     TL1,#0FDH
                SETB    TR1
                MOV     SCON,#040H

                SETB    SDA
                SETB    SCL

                SETB    DISP_H
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                SETB        DISP_M

Start:          ACALL       START_LOOP_RX
                MOV         A,30H

;-------------------------------------------------------------------------------------------------------------------
;  Display and go to start
;-------------------------------------------------------------------------------------------------------------------
                CJNE        A,#073H,Read_CRX
                MOV         HOURS_1,031H
                MOV         HOURS_2,032H
                MOV         MINUTES_1,033H
                MOV         MINUTES_2,034H

                MOV         R0,#HOURS
                MOV         R1,#HOURS_1
                MOV         Time_END_R,#000H
                ACALL       Convert_ANSI_To_BCD

                ACALL       DISPLAY
                ACALL       SEND_n

                AJMP        START

; -------------------------------------------------------------------------------------------------------------------
;  Read Data from Tag
; -------------------------------------------------------------------------------------------------------------------
Read_CRX:       CJNE        A,#072H,Write_CRX
                ;ACALL      CRX_RD

                MOV         Car_license_Text1,#0A1H
                MOV         Car_license_Text2,#0A1H
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                MOV     Number_1,#039H
                MOV     Number_2,#039H
                MOV     Number_3,#039H
                MOV     Number_4,#039H
                MOV     Car_license_END,#000H

                MOV     HOURS_1,#032H
                MOV     HOURS_2,#032H
                MOV     MINUTES_1,#032H
                MOV     MINUTES_2,#032H

                MOV     R4,#040H
DISPLAY_TIME:   ACAll   DISPLAY
                DJNZ    R4,DISPLAY_TIME

                ACALL   SEND_n
                ACALL   TX_TEXT
                ACALL   SEND_n

                ACALL   START_LOOP_RX
                MOV     A,30H
                CJNE    A,#06DH,N

                MOV     A,031H
                SUBB    A,#030H
                RL      A
                RL      A
                RL      A
                RL      A
                MOV     MONEY,A
                MOV     A,032H
                SUBB    A,#030H
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                ORL     A,MONEY
                MOV     R4,#000H
                RR      A
                RR      A
                RR      A
                RR      A
                MOV     P0,A
DISPLAY_MONEY:  CPL     DISP_M
                ACALL   DELAY_5ms
                CPL     DISP_M
                DJNZ    R4,DISPLAY_MONEY

N:              ACALL   SEND_n

                AJMP    Start

; ---------------------------------------------------------------------------
;  Write Data from Tag
; ---------------------------------------------------------------------------

Write_CRX:      CJNE    A,#077H,Display1
                MOV     HOURS_1,031H
                MOV     HOURS_2,032H
                MOV     MINUTES_1,033H
                MOV     MINUTES_2,034H

                MOV     Car_license_Text1,035H
                MOV     Car_license_Text2,036H
                MOV     Number_1,037H
                MOV     Number_2,038H
                MOV     Number_3,039H
                MOV     Number_4,03AH
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                    ACALL       CRX_WR
                    ACALL       DISPLAY
                    ACALL       SEND_n
                    AJMP        Start

Display1:           CJNE        A,#064H,end_1
                    ACALL       DISPLAY
end_1:              ACALL       SEND_n
                    AJMP        Start

;-------------------------------------------------------------------------------------------------
; RX Serial Text from ROM Pointer
;-------------------------------------------------------------------------------------------------

START_LOOP_RX:      MOV         R0,#BUFFER
                    SETB        REN
LOOP:               JNB         RI,$
                    CLR         RI
                    MOV         A,SBUF
                    MOV         @R0,A
                    MOV         @R1,A
                    CJNE        A,#065H,RX_NEXT
                    RET

RX_NEXT:            INC         R0
                    AJMP        LOOP

;-------------------------------------------------------------------------------------------------
; TX Serial Text from ROM Pointer
;-------------------------------------------------------------------------------------------------

TX_TEXT:            MOV         R1,#Car_license_Text1
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                    MOV     Time_END_R,#0FFH
                    CLR     TI
TX_LOOP:            MOV     A,@R1
                    INC     R1
                    CJNE    A,#0FFH,TX_CHAR
                    RET
TX_CHAR:            MOV     SBUF,A
                    JNB     TI,$
                    CLR     TI
                    AJMP    TX_LOOP

;-------------------------------------------------------------------------------
; Convert ANSI To BCD
;-------------------------------------------------------------------------------
Convert_ANSI_To_BCD:MOV     A,@R1
                    SUBB    A,#030H
                    RL      A
                    RL      A
                    RL      A
                    RL      A
                    MOV     @R0,A
                    INC     R1

                    MOV     A,@R1
                    SUBB    A,#030H
                    ORL     A,@R0
                    MOV     @R0,A
                    INC     R0
                    INC     R1
                    CJNE    @R1,#000H,Convert_ANSI_To_BCD
                    RET
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
;-------------------------------------------------------------------------------------------------
; Send 'n'
;-------------------------------------------------------------------------------------------------
SEND_n:         CLR     TI
                MOV     A,#06EH
                MOV     SBUF,A
                JNB     TI,$
                CLR     TI
                RET
;-------------------------------------------------------------------------------------------------
; Display Time to 7 Segment
;-------------------------------------------------------------------------------------------------
DISPLAY:        ;MOV    R4,#030H
                SETB    Dot_time        ; ON Dot Time
                MOV     A,HOURS
                RR      A
                RR      A
                RR      A
                RR      A
                MOV     P0,A            ; Send Hours to P0
                CPL     DISP_H          ; ON Display Hours
                ACALL   DELAY_5ms
                CPL     DISP_H

                MOV     A,MINUTES
                RR      A
                RR      A
                RR      A
                RR      A
                MOV     P0,A            ; Send Minutes to P0
                CPL     DISP_M          ; ON Display Minutes
                ACALL   DELAY_5ms
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                CPL     DISP_M
                CPL     Dot_time        ; OFF Dot Time
                ;DJNZ   R4,DISPLAY_TIME
                RET                     ; Return

;-------------------------------------------------------------------------------------------------------------
; I2C CRX 14 Read
;-------------------------------------------------------------------------------------------------------------
CRX_RD:         MOV     I2C_ADDR,#CRX_ID
                LCALL   I2C_SLAVE
                MOV     I2C_DATA,#01H
                LCALL   I2C_DATA_WR

                MOV     I2C_ADDR,#CRX_ID+1
                LCALL   I2C_SLAVE

                MOV     I2C_DATA,#00AH
                LCALL   I2C_DATA_WR

                LCALL   I2C_DATA_RD
                MOV     Car_license_Text1,I2C_DATA
                LCALL   I2C_ACK_BIT

                LCALL   I2C_DATA_RD
                MOV     Car_license_Text2,I2C_DATA
                LCALL   I2C_ACK_BIT
                LCALL   I2C_DATA_RD
                MOV     Number_1,I2C_DATA
                LCALL   I2C_ACK_BIT

                LCALL   I2C_DATA_RD
                MOV     Number_2,I2C_DATA
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        LCALL   I2C_ACK_BIT

        LCALL   I2C_DATA_RD
        MOV     Number_3,I2C_DATA
        LCALL   I2C_ACK_BIT

        LCALL   I2C_DATA_RD
        MOV     Number_4,I2C_DATA
        LCALL   I2C_ACK_BIT
        LCALL   I2C_DATA_RD
        MOV     HOURS_1,I2C_DATA
        LCALL   I2C_ACK_BIT

        LCALL   I2C_DATA_RD
        MOV     HOURS_2,I2C_DATA
        LCALL   I2C_ACK_BIT

        LCALL   I2C_DATA_RD
        MOV     MINUTES_1,I2C_DATA
        LCALL   I2C_ACK_BIT

        LCALL   I2C_DATA_RD
        MOV     MINUTES_2,I2C_DATA
        LCALL   I2C_ACK_BIT

        LCALL   I2C_STOP
        RET

;-------------------------------------------------------------------------------------------------------------
; I2C RTC Write
;-------------------------------------------------------------------------------------------------------------
CRX_WR:         MOV     I2C_ADDR,#CRX_ID
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
LCALL     I2C_SLAVE

MOV       I2C_DATA,#001H
LCALL     I2C_DATA_WR

MOV       I2C_DATA,#00AH
LCALL     I2C_DATA_WR

MOV       I2C_DATA,Car_license_Text1
LCALL     I2C_DATA_WR

MOV       I2C_DATA,Car_license_Text2
LCALL     I2C_DATA_WR

MOV       I2C_DATA,Number_1
LCALL     I2C_DATA_WR

MOV       I2C_DATA,Number_2
LCALL     I2C_DATA_WR

MOV       I2C_DATA,Number_3
LCALL     I2C_DATA_WR

MOV       I2C_DATA,Number_4
LCALL     I2C_DATA_WR

MOV       I2C_DATA,HOURS_1
LCALL     I2C_DATA_WR

MOV       I2C_DATA,HOURS_2
LCALL     I2C_DATA_WR
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                    MOV     I2C_DATA,MINUTES_1
                    LCALL   I2C_DATA_WR

                    MOV     I2C_DATA,MINUTES_2
                    LCALL   I2C_DATA_WR

                    LCALL   I2C_STOP
                    RET

;-------------------------------------------------------------------------------
; I2C Data Write
; I/P:          I2C_DATA
; Reserve:      R5
;-------------------------------------------------------------------------------
I2C_DATA_WR:        PUSH    ACC
                    SETB    I2C_ACK
                    MOV     A,I2C_DATA
                    MOV     R5,#008
I2C_DATA_WR_1:      RLC     A
                    MOV     SDA,C
                    ACALL   I2C_CLK
                    DJNZ    R5,I2C_DATA_WR_1
                    SETB    SDA
                    ACALL   I2C_DELAY
                    SETB    SCL
                    ACALL   I2C_DELAY
                    JB      SDA,I2C_DATA_WR_2
                    CLR     I2C_ACK
I2C_DATA_WR_2:      CLR     SCL
                    POP     ACC
                    RET
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
;-------------------------------------------------------------------------------------------------------
; I2C Data Read
; O/P:          I2C_DATA
; Reserve:      R5
;-------------------------------------------------------------------------------------------------------
I2C_DATA_RD:            PUSH        ACC
                        CLR         A
                        MOV         R5,#008
I2C_DATA_RD_1:          ACALL       I2C_DELAY
                        SETB        SCL
                        ACALL       I2C_DELAY
                        MOV         C,SDA
                        RLC         A
                        CLR         SCL
                        DJNZ        R5,I2C_DATA_RD_1
                        MOV         I2C_DATA,A
                        POP         ACC
                        RET
;-------------------------------------------------------------------------------------------------------
; I2C Slave Connect
; I/P:          I2C_ADDR
; O/P Flag:     I2C_ACK
; Reserve:      R5
;-------------------------------------------------------------------------------------------------------
I2C_SLAVE:              PUSH        ACC
                        SETB        I2C_ACK
                        MOV         A,I2C_ADDR
                        ACALL       I2C_START
                        MOV         R5,#008
I2C_SLAVE_1:            RLC         A
                        MOV         SDA,C
                        ACALL       I2C_CLK
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                    DJNZ    R5,I2C_SLAVE_1
                    SETB    SDA
                    ACALL   I2C_DELAY
                    SETB    SCL
                    ACALL   I2C_DELAY
                    JB      SDA,I2C_SLAVE_2
                    CLR     I2C_ACK
I2C_SLAVE_2:        CLR     SCL
                    POP     ACC
                    RET

;-------------------------------------------------------------------------------
; I2C Start Condition
;-------------------------------------------------------------------------------
I2C_START:          JNB     SCL,I2C_START_1
                    CLR     SCL
I2C_START_1:        SETB    SDA
                    SETB    SCL
                    ACALL   I2C_DELAY
                    CLR     SDA
                    ACALL   I2C_DELAY
                    CLR     SCL
                    RET
;-------------------------------------------------------------------------------
; I2C Stop Condition
;-------------------------------------------------------------------------------
I2C_STOP:           JNB     SCL,I2C_STOP_1
                    CLR     SCL
I2C_STOP_1:         CLR     SDA
                    ACALL   I2C_DELAY
                    SETB    SCL
                    ACALL   I2C_DELAY
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                SETB    SDA
                RET

;-------------------------------------------------------------------------------------------
; I2C Clock
;-------------------------------------------------------------------------------------------
I2C_CLK:        ACALL   I2C_DELAY
                SETB    SCL
                ACALL   I2C_DELAY
                CLR     SCL
                RET

;-------------------------------------------------------------------------------------------
; I2C Acknowledge
;-------------------------------------------------------------------------------------------
I2C_ACK_BIT:    CLR     SDA
                ACALL   I2C_DELAY
                ACALL   I2C_CLK
                SETB    SDA
                RET
;-------------------------------------------------------------------------------------------
; I2C Not Acknowledge
;-------------------------------------------------------------------------------------------
I2C_NACK_BIT:   SETB    SDA
                ACALL   I2C_DELAY
                ACALL   I2C_CLK
                SETB    SCL
                RET

;-------------------------------------------------------------------------------------------
; Dummy Delay time I2C_DELAY, 10m, 100m, 1s
;-------------------------------------------------------------------------------------------
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
I2C_DELAY:      MOV     R6,#00CH
I2C_DELAY_1:    NOP
                NOP
                DJNZ    R6,I2C_DELAY_1

DELAY_5ms:      MOV     R7,#05
DELAY_5ms_1:    MOV     R6,#0E6H
DELAY_5ms_2:    NOP
                NOP
                DJNZ    R6,DELAY_5ms_2
                DJNZ    R7,DELAY_5ms_1
                RET

DELAY_10ms:     MOV     R7,#010
DELAY_10ms_1:   MOV     R6,#0E6H
DELAY_10ms_2:   NOP
                NOP
                DJNZ    R6,DELAY_10ms_2
                DJNZ    R7,DELAY_10ms_1
                RET

DELAY_20ms:     MOV     R7,#20
DELAY_20ms_1:   MOV     R6,#0E6H
DELAY_20ms_2:   NOP
                NOP
                DJNZ    R6,DELAY_20ms_2
                DJNZ    R7,DELAY_20ms_1
                RET

DELAY_100ms:    MOV     R7,#100
DELAY_100ms_1:  MOV     R6,#0E6H
DELAY_100ms_2:  NOP
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                NOP
                DJNZ        R6,DELAY_100ms_2
                DJNZ        R7,DELAY_100ms_1
                RET

DELAY_1s:       MOV         R5,#100
DELAY_1s_1:     ACALL       DELAY_10ms
                DJNZ        R5,DELAY_1s_1
                RET

                END
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

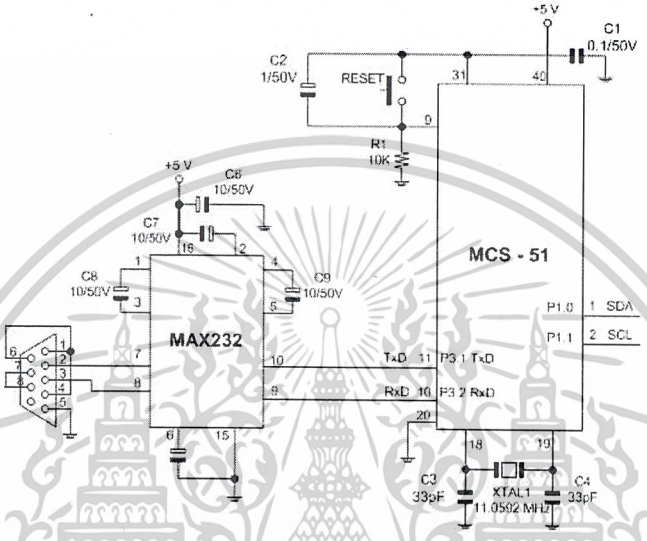
บทที่ 4

ผลการการทดลอง

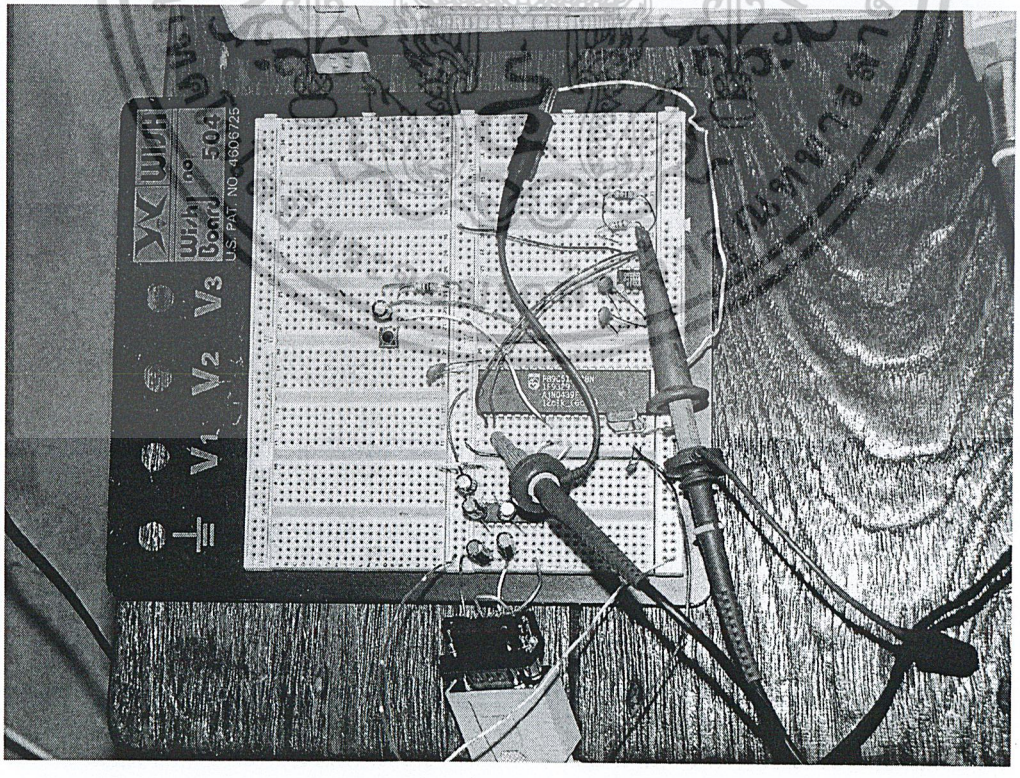
4.1 การทดลองการติดต่อระหว่างคอมพิวเตอร์กับไมโครคอนโทรลเลอร์

วัตถุประสงค์

- เพื่อทดสอบการติดต่อกันระหว่างคอมพิวเตอร์กับไมโครคอนโทรลเลอร์



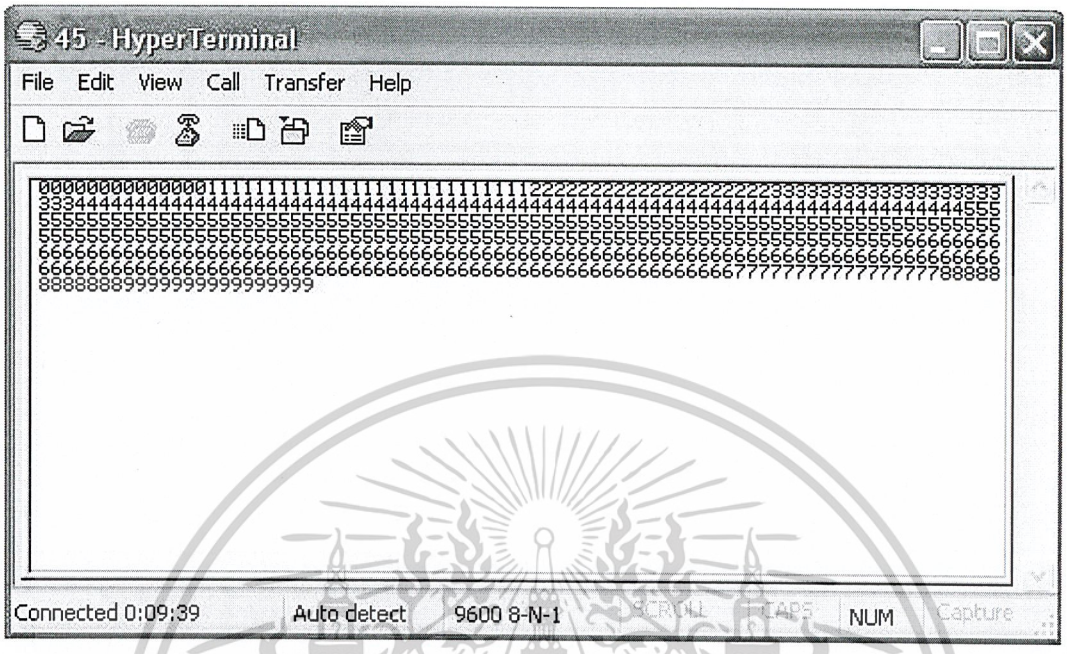
รูปที่ 4.1 วงจรภาค Control



รูปที่ 4.2 ทดลองวงจรภาค Control

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

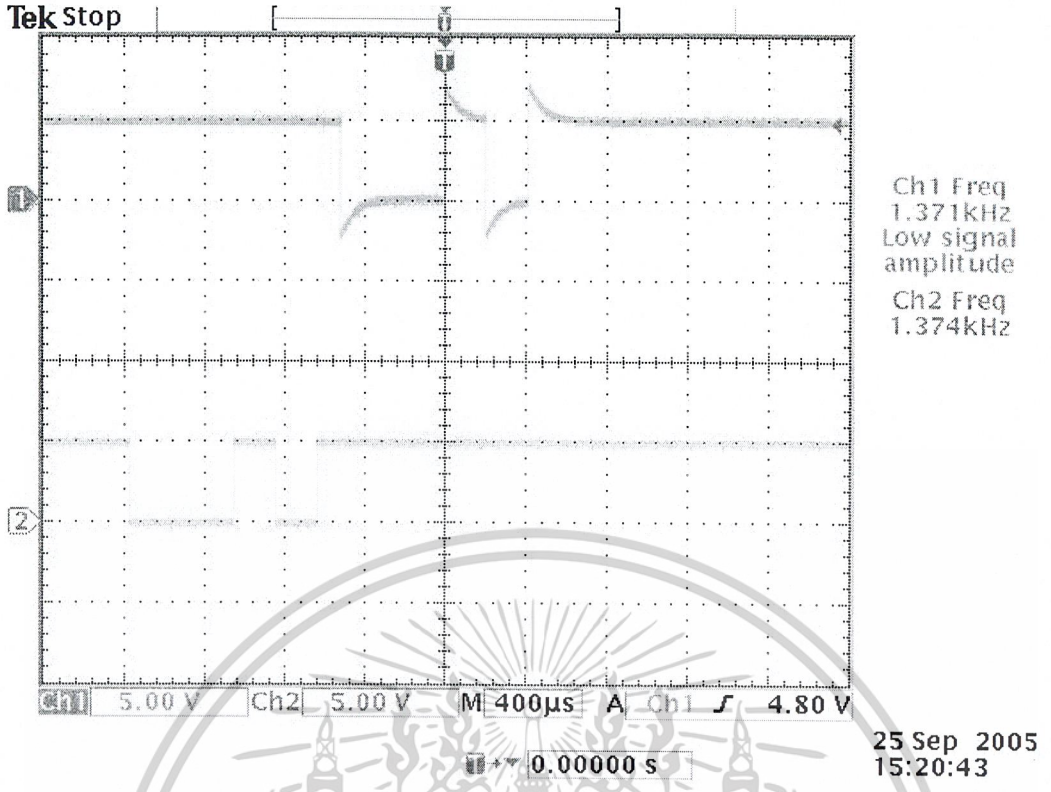
การทดลองทำการติดต่อคอมพิวเตอร์กับไมโครคอนโทรลเลอร์
โดยใช้โปรแกรม Hyper terminal ในการติดต่อ ผลที่ได้เป็นดังนี้



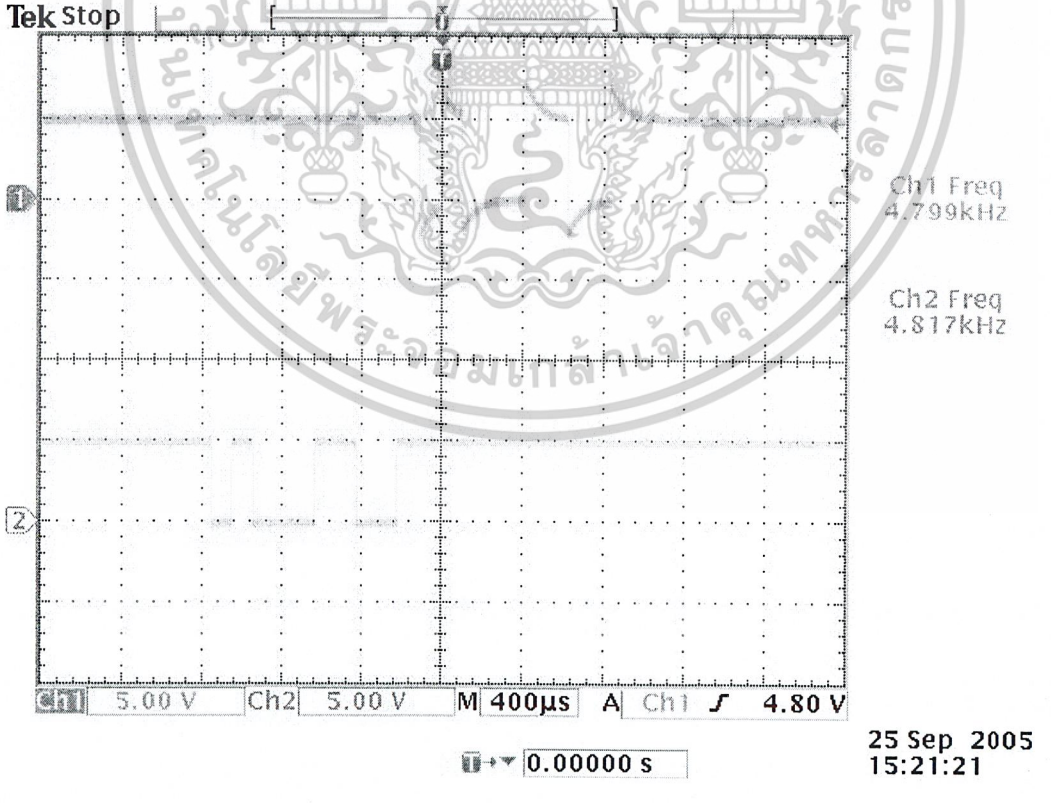
รูปที่ 4.3 ผลการทดลองในการติดต่อระหว่างคอมพิวเตอร์กับไมโครคอนโทรลเลอร์

ผลที่ได้คือ เมื่อทำการกดคีย์ คอมพิวเตอร์จะทำการส่งค่าคีย์ไปยังไมโครคอนโทรลเลอร์ แล้วไมโครคอนโทรลเลอร์ จะส่งค่ากลับมายังโปรแกรม Hyper terminal โดยแสดงผลดังรูปที่ 4.3 และไดอะแกรมเวลาแสดงสถานะของข้อมูลที่คอมพิวเตอร์ส่งไปยังไมโครคอนโทรลเลอร์แสดงโดยสัญญาณที่2 และข้อมูลที่ไมโครคอนโทรลเลอร์ส่งกลับไปยังคอมพิวเตอร์โดยไมโครคอนโทรลเลอร์แสดงโดยสัญญาณที่1 และสัญญาณที่ 1 ทำการวัดวัดจากขาที่ 11 ของไมโครคอนโทรลเลอร์ และสัญญาณที่ 2 ทำการวัดวัดจากขาที่ 10 ของไมโครคอนโทรลเลอร์ ดังรูปที่ 4.4 ถึงรูปที่ 4.5

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.4 ผลการทดลองไดอะแกรมเวลาแสดงสถานะการกดคีย์ 0



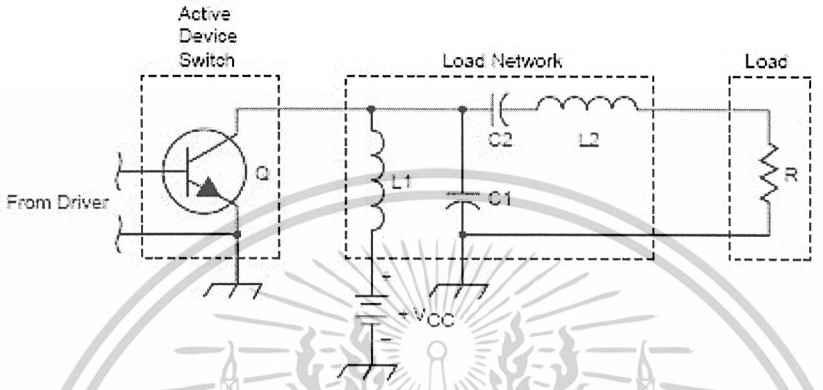
รูปที่ 4.5 ผลการทดลองไดอะแกรมเวลาแสดงสถานะการกดคีย์ 1

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

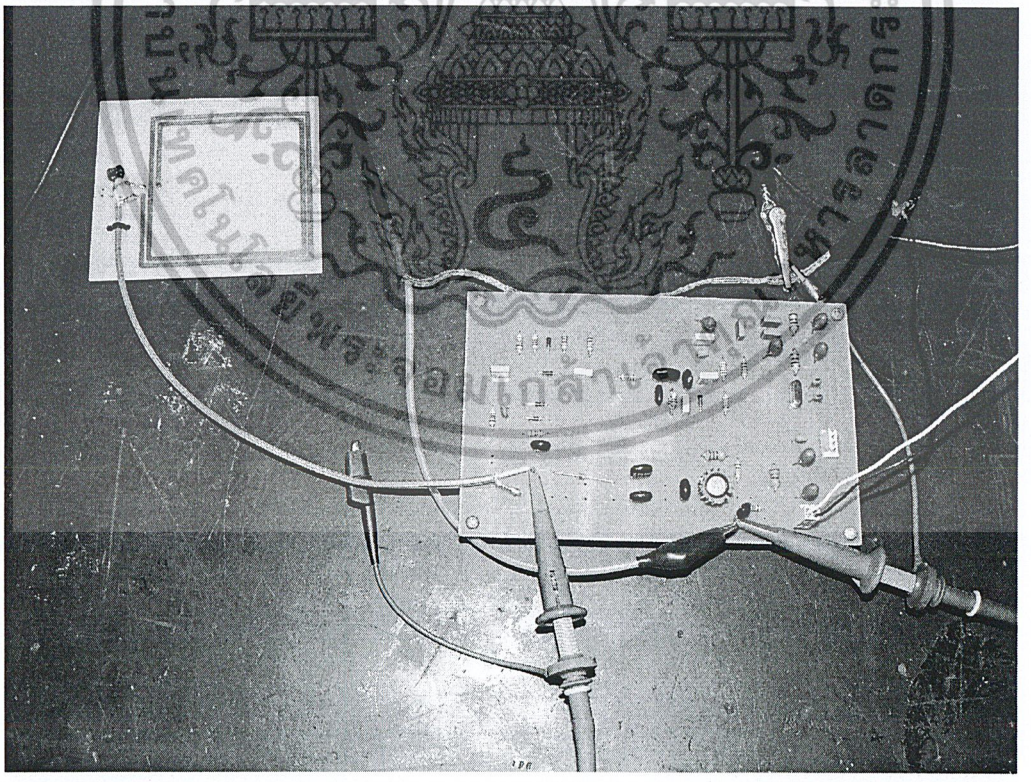
4.2 การทดลองทางภาคส่ง Class E Power Amplifier

วัตถุประสงค์

- เพื่อหาค่ากำลังงานที่ได้ของวงจร Class E Power Amplifier
- เพื่อหาผลตอบสนองต่อความถี่ของวงจร
- เพื่อทดสอบการขยายสัญญาณแบบ ASK



รูปที่ 4.6 วงจรภาคส่ง **Class E Power Amplifier**



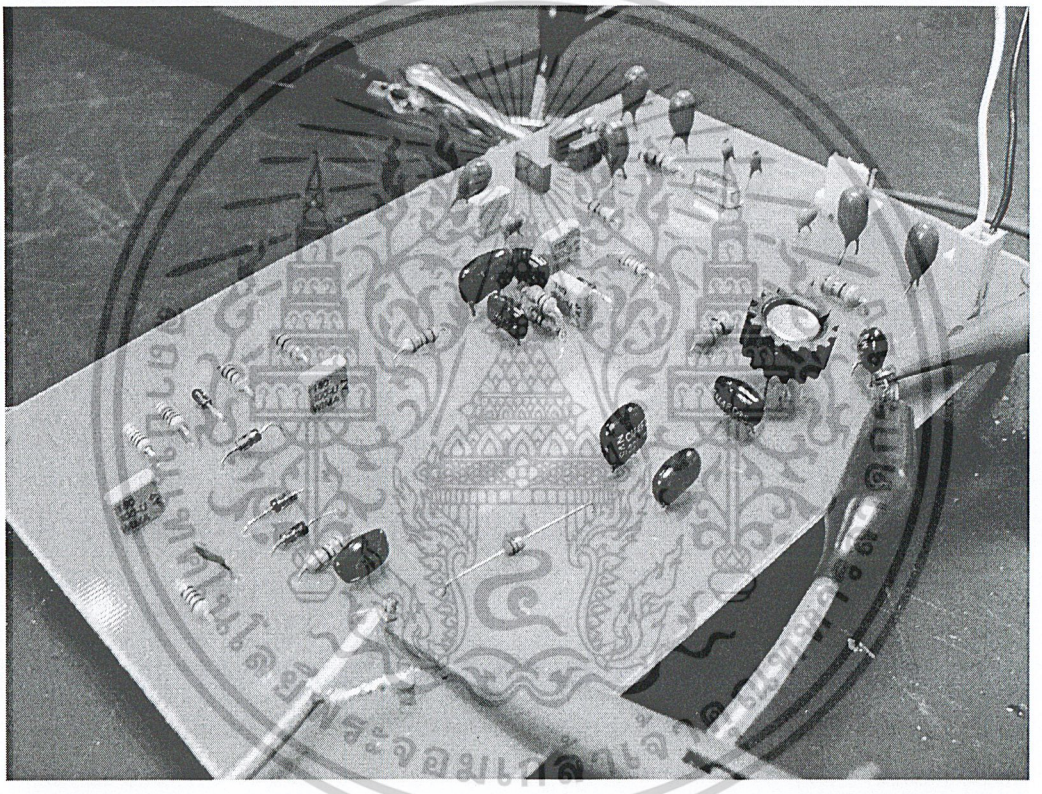
รูปที่ 4.7 แสดงวงจรในส่วนภาคส่ง **Class E Power Amplifier**

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การทดลองหาผลตอบสนองต่อความถี่

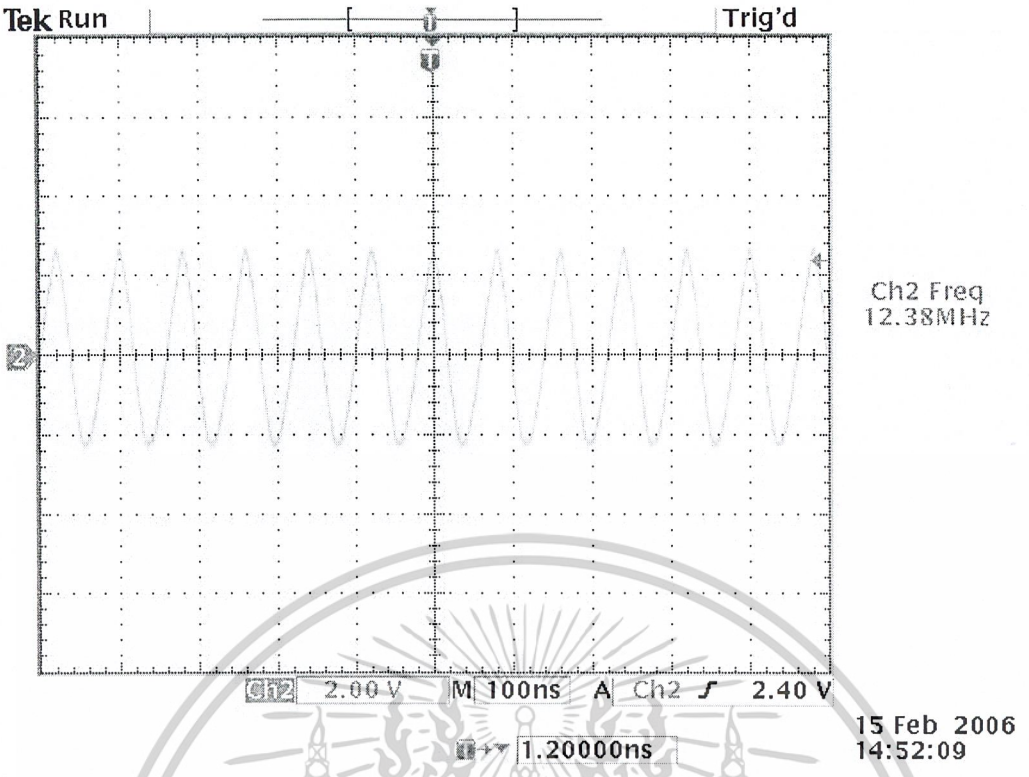
โดยใช้วงจรดังรูปที่ 4.6 โดยใช้ค่าที่ได้จากการคำนวณ มีขั้นตอนการทดลองดังนี้

- ทำการต่อวงจร ได้วงจรที่สำเร็จแสดงรูปที่ 4.7
- ใช้ VCC = 9V, C1 = 55.01 pF, L1 = 33 uH, L2 = 2.50 uH, C2 = 238 pF, Q=2N3866 , ใช้วงจรสายอากาศเป็น Load
- ป้อนสัญญาณ SINEWAVE ที่ความถี่ 12.5 MHz , 5V ที่ ขาเบสของQ
- บันทึกค่าตกคร่อมโหลด
- ทำซ้ำอีกครั้งตามความถี่ที่กำหนด 12.7 MHz, 12.9 MHz, 13.1 MHz, 13.3 MHz, 13.5 MHz, 13.7 MHz, 13.9 MHz, 14.1 MHz, 14.3 MHz,14.5 MHz

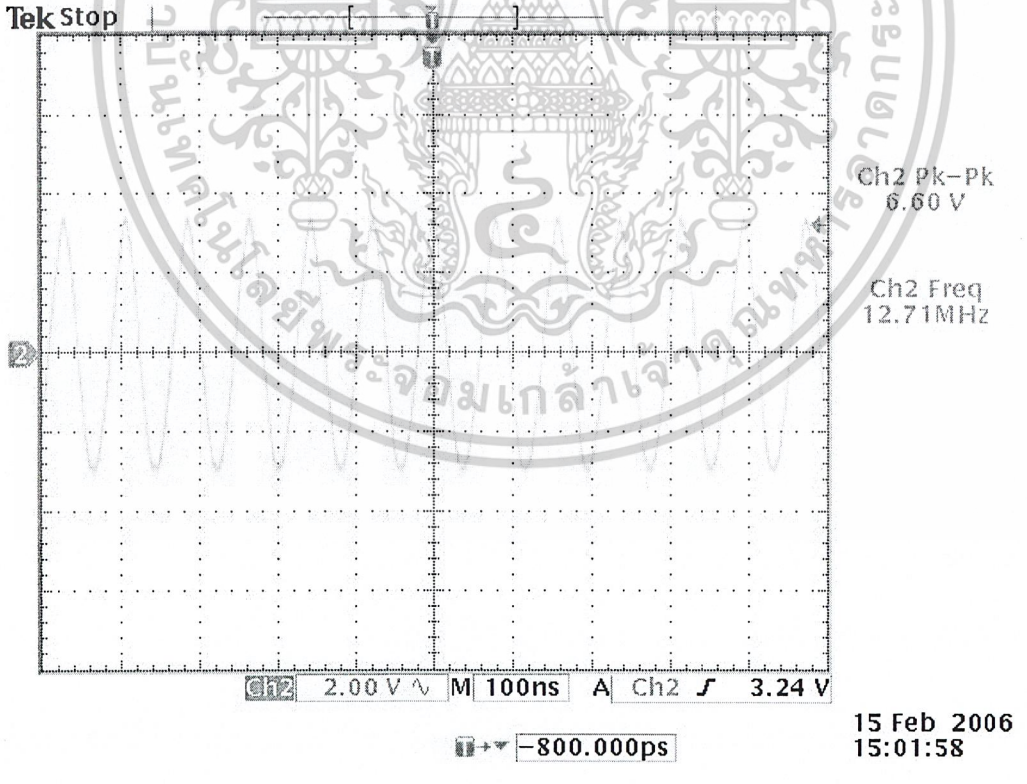


รูปที่ 4.8 แสดงการวัดที่จุดL2และขาเบสของQในภาคส่ง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

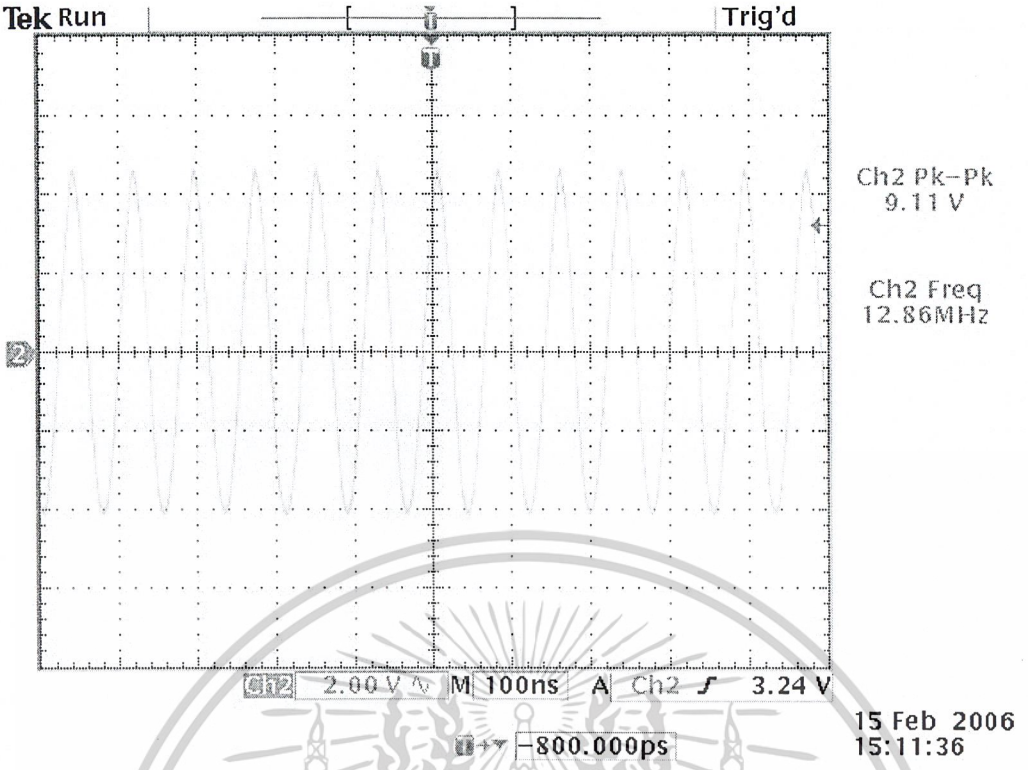


รูปที่ 4.9 แสดงสัญญาณ ที่ตกคร่อมโหลด, ความถี่ 12.5 MHz

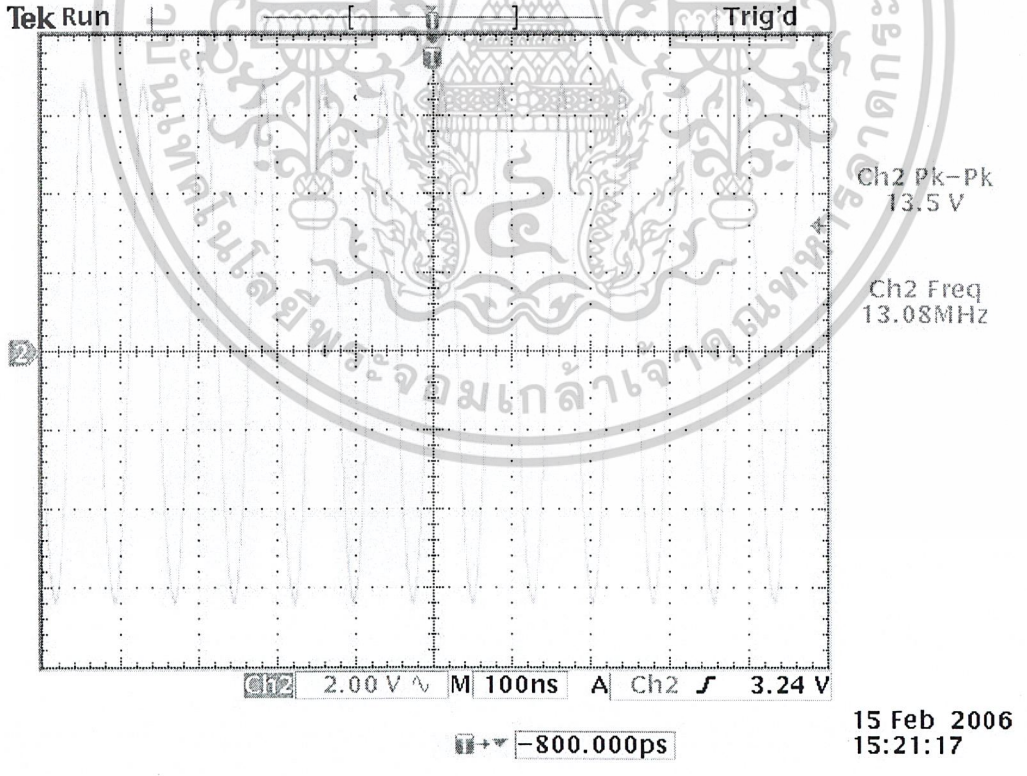


รูปที่ 4.10 แสดงสัญญาณ ที่ตกคร่อมโหลด, ความถี่ 12.7 MHz

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

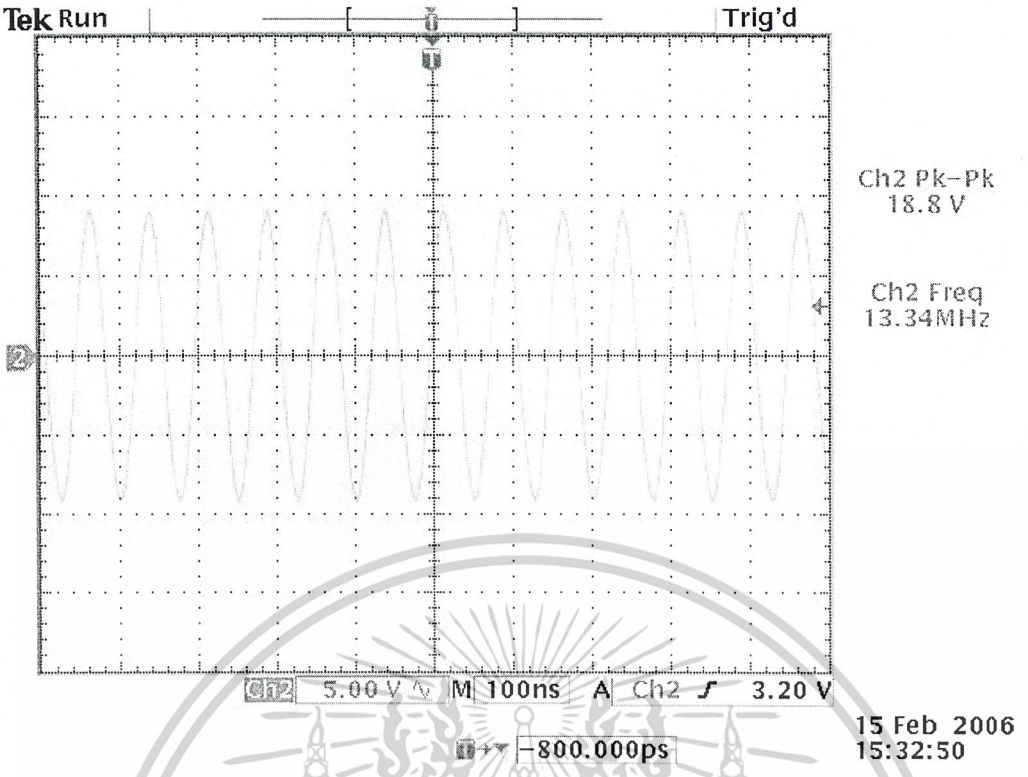


รูปที่ 4.11 แสดงสัญญาณ ที่ตกคร่อมโหลด, ความถี่ 12.9 MHz

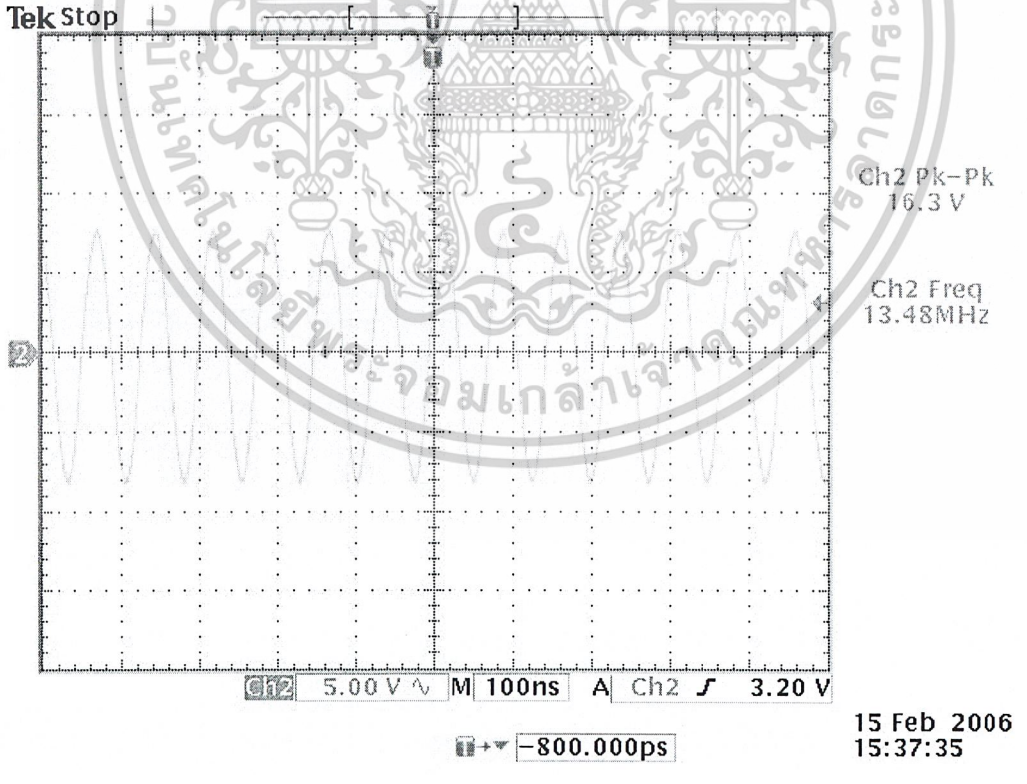


รูปที่ 4.12 แสดงสัญญาณ ที่ตกคร่อมโหลด, ความถี่ 13.1 MHz

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

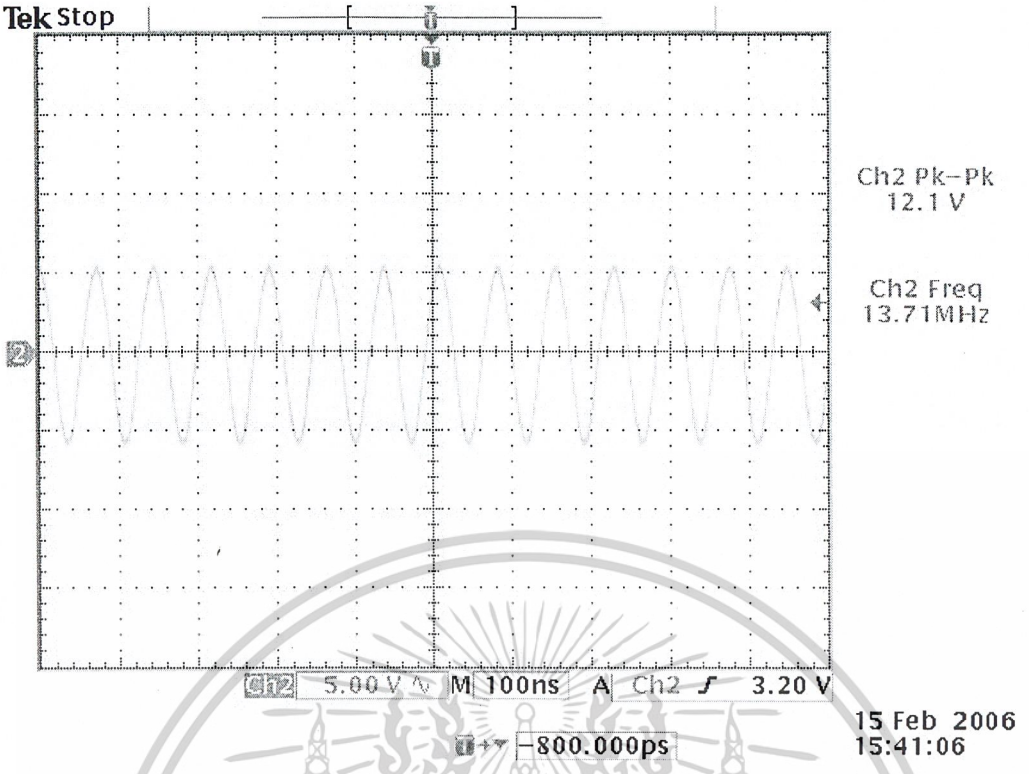


รูปที่ 4.13 แสดงสัญญาณ ที่ตกคร่อมโหลด, ความถี่ 13.3 MHz

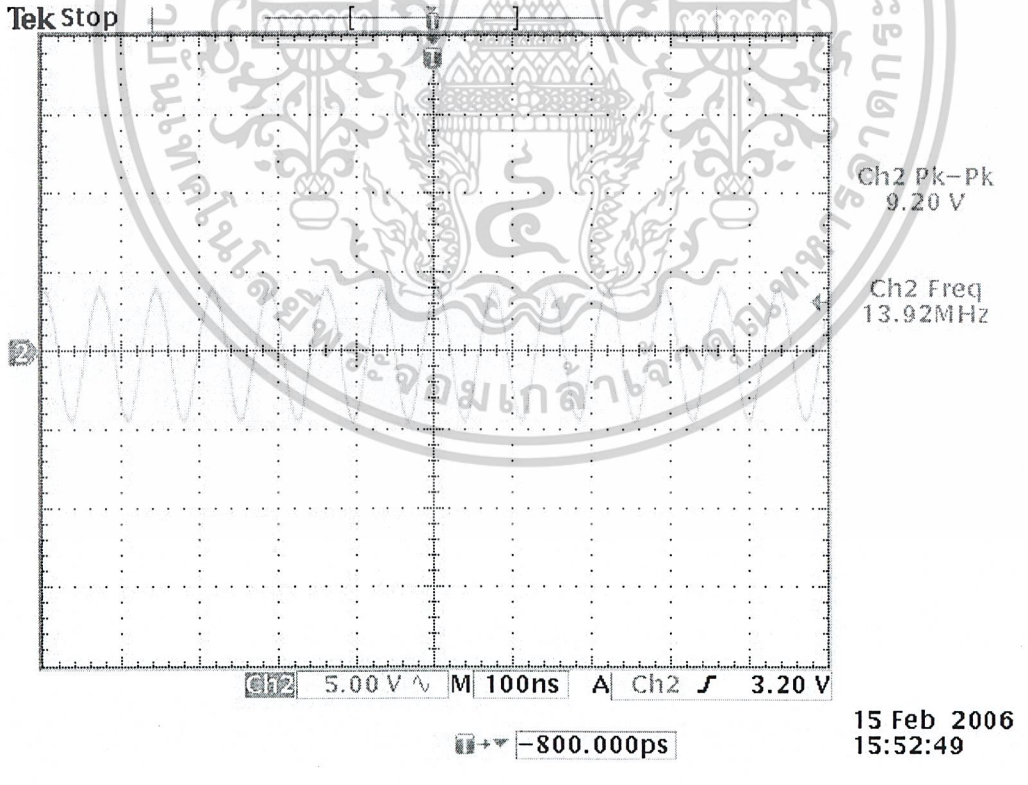


รูปที่ 4.14 แสดงสัญญาณ ที่ตกคร่อมโหลด, ความถี่ 13.5 MHz

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

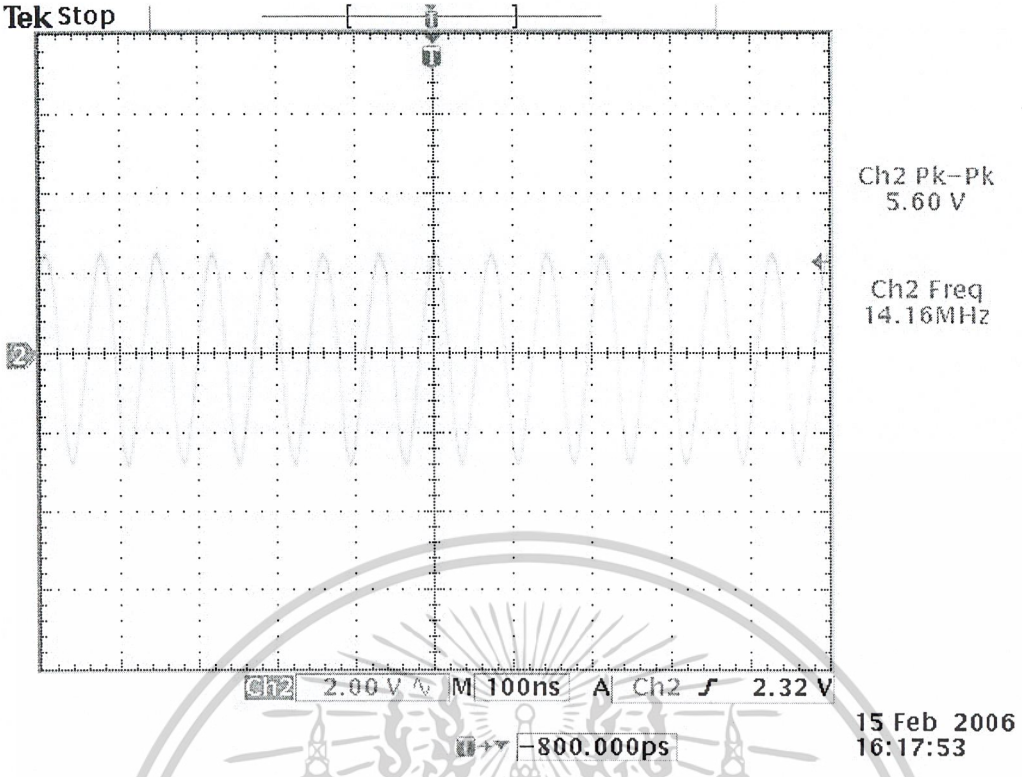


รูปที่ 4.15 แสดงสัญญาณ ที่ตกคร่อมโหลด, ความถี่ 13.7 MHz

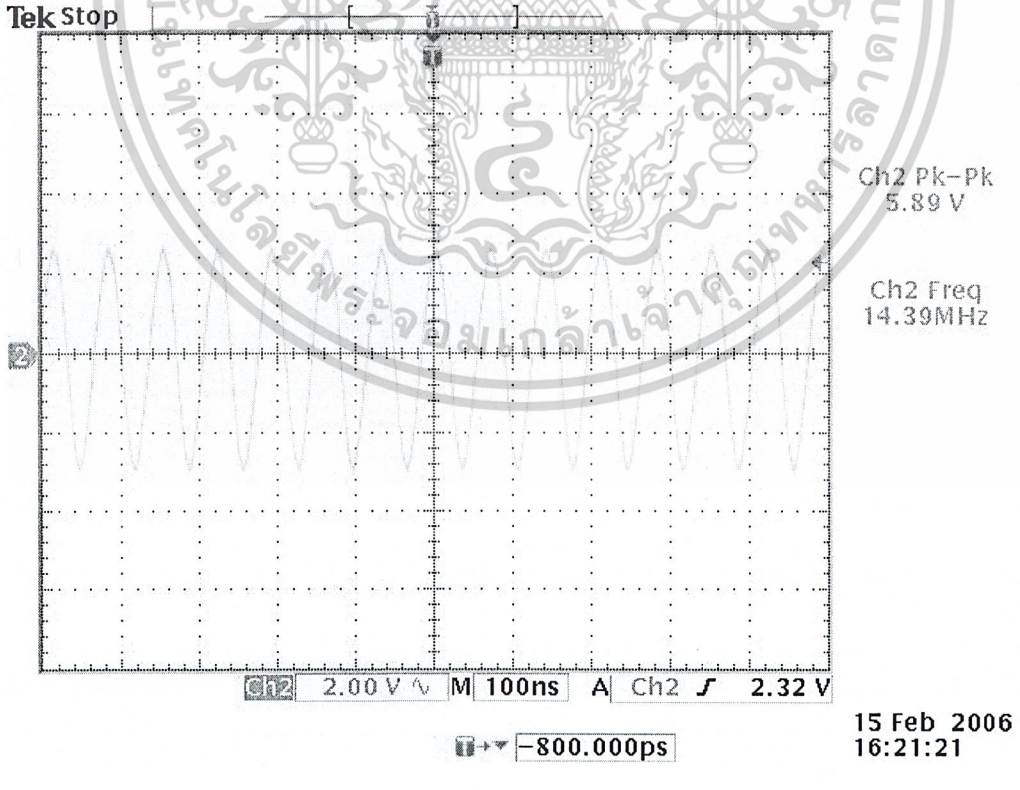


รูปที่ 4.16 แสดงสัญญาณ ที่ตกคร่อมโหลด, ความถี่ 13.9 MHz

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

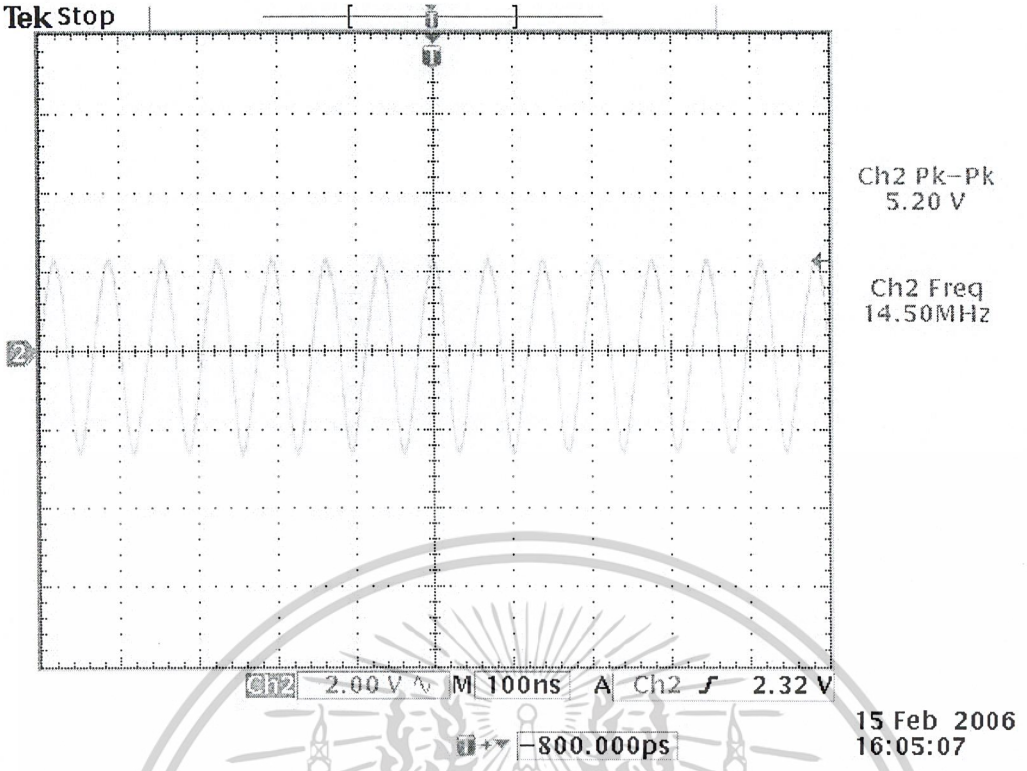


รูปที่ 4.17 แสดงสัญญาณ ที่ตกคร่อมโหลด, ความถี่ 14.1 MHz



รูปที่ 4.18 แสดงสัญญาณ ที่ตกคร่อมโหลด, ความถี่ 14.3 MHz

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.19 แสดงสัญญาณ ที่ตกคร่อม โหลด, ความถี่ 14.5 MHz

นำค่าแรงดันกับความถี่ที่ได้ทำการทดลองไปแทนในตารางที่ 4.1

ความถี่ INPUT	แรงดันที่ตกคร่อมโหลด
12.5 MHz	4.63 Vp-p
12.7 MHz	6.62 Vp-p
12.9 MHz	9.13 Vp-p
13.1 MHz	13.51 Vp-p
13.3 MHz	18.82 Vp-p
13.5 MHz	16.31 Vp-p
13.7 MHz	12.12 Vp-p
13.9 MHz	9.22 Vp-p
14.1 MHz	5.61 Vp-p
14.3 MHz	5.89 Vp-p
14.5 MHz	5.20 Vp-p

ตารางที่ 4.1 ผลการทดลองวงจรภาคส่ง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

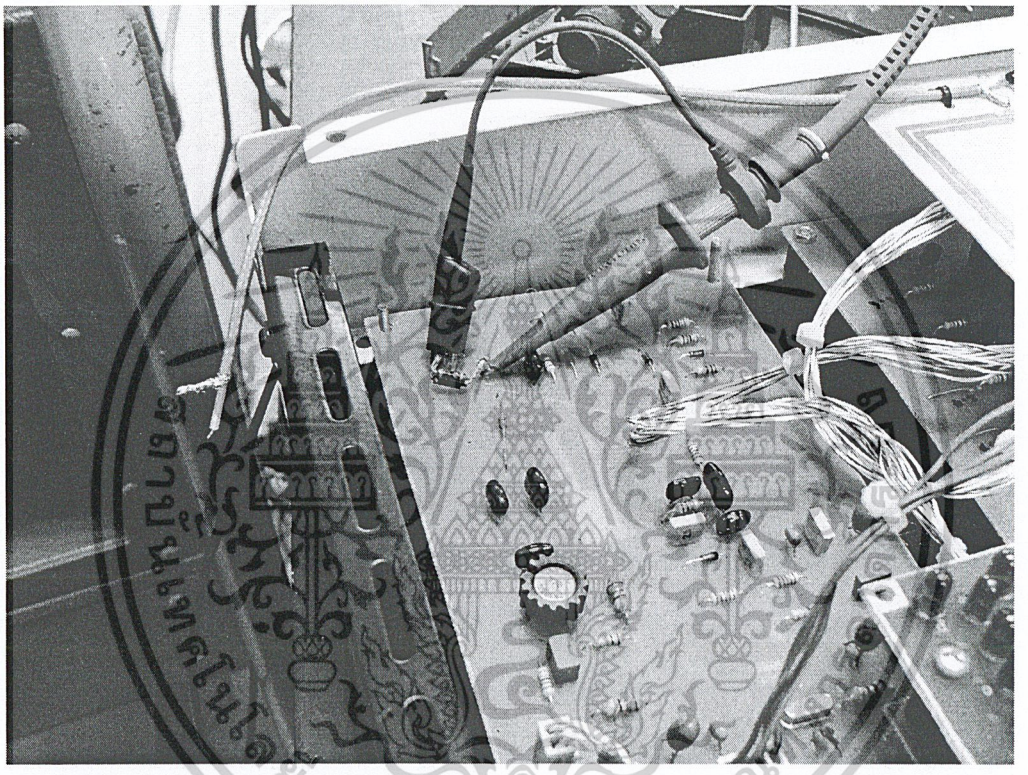
นำค่าแรงดันที่ L7 ไปพล๊อตเทียบกับความถี่ จะได้กราฟที่ 4.1

กราฟที่ 4.1 แรงดันที่ขา L7 / ความถี่

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

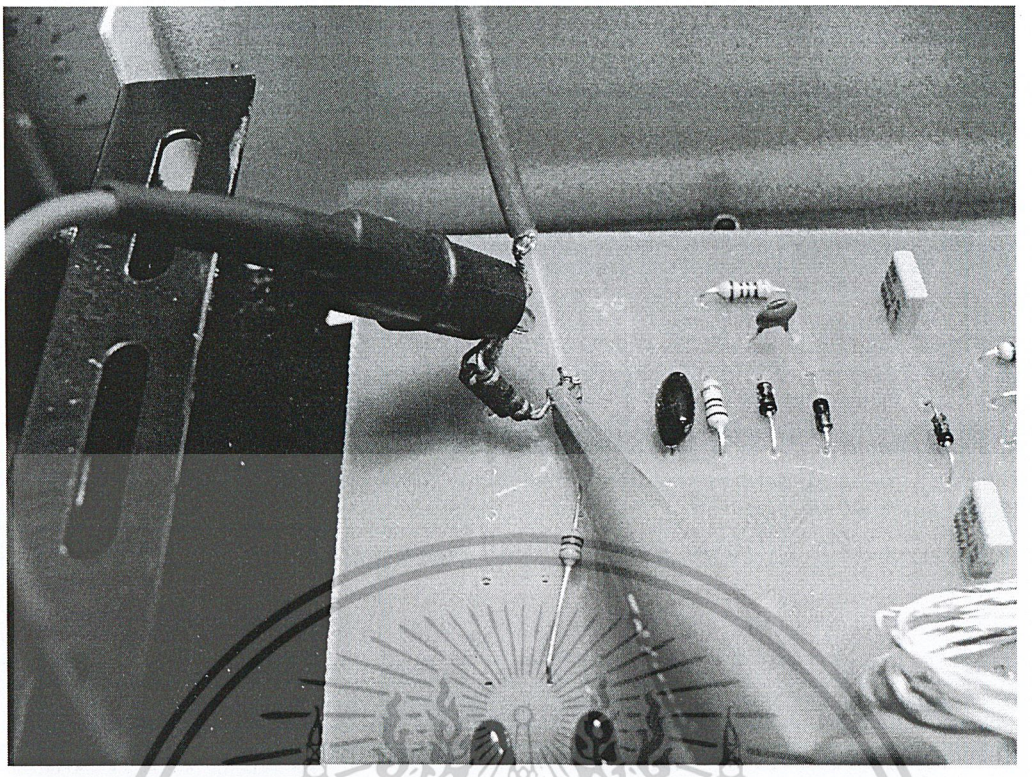
การทดลองหาค่ากำลังงานที่ได้ของวงจร Class E Power Amplifier ใช้ตัวต้านทานเป็นโหลด โดยใช้วงจรดังรูปที่ 4.6 โดยใช้ค่าที่ได้จากการคำนวณ มีขั้นตอนการทดลองดังนี้

- ใช้วงจรที่สำเร็จในรูปที่ 4.7
- ใช้ VCC = 9V, C1 = 55.01 pF, L1 = 33 uH, L2 = 2.50 uH, C2 = 238 pF, Q=2N3866 , ใช้ตัวต้านทานเป็นโหลดเท่ากับ 42 Ω
- ใช้สัญญาณจาก CRX14 ที่มีความถี่ 13.56 MHz, 5V จ่ายให้ที่
- บันทึกค่าตกคร่อมโหลด

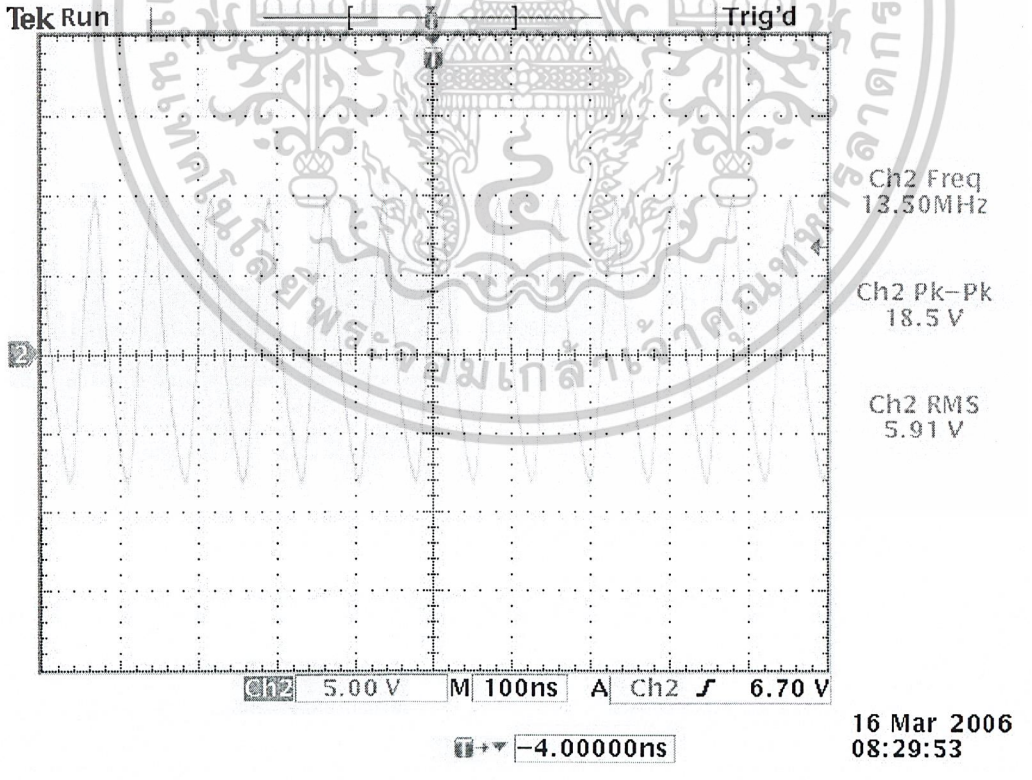


รูปที่ 4.20 แสดงจุดวัดกำลังงานที่ใช้ตัวต้านทานเป็นโหลด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.21 แสดงการวัดกำลังงานที่ใช้ตัวต้านทานเป็นโหลด



รูปที่ 4.22 แสดงสัญญาณที่วัดตกคร่อมโหลดใช้ตัวต้านทานเป็นโหลด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

นำค่าแรงดันที่อ่านได้จากสโคปไปคำนวณหากำลังงาน

$V_{RMS} = 5.91V$

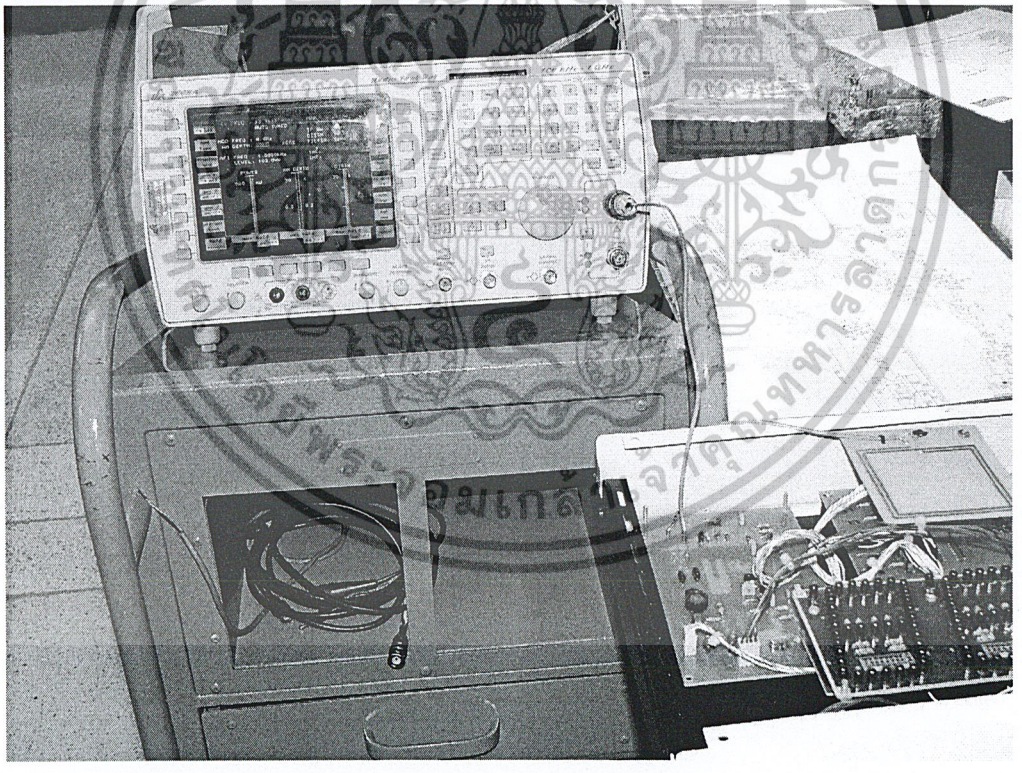
$R_L = 42\ \Omega$

$P_{RMS} = V^2_{RMS} / R_L$

ดังนั้น $P_{RMS} = 34.93 / 42 = 0.831W$

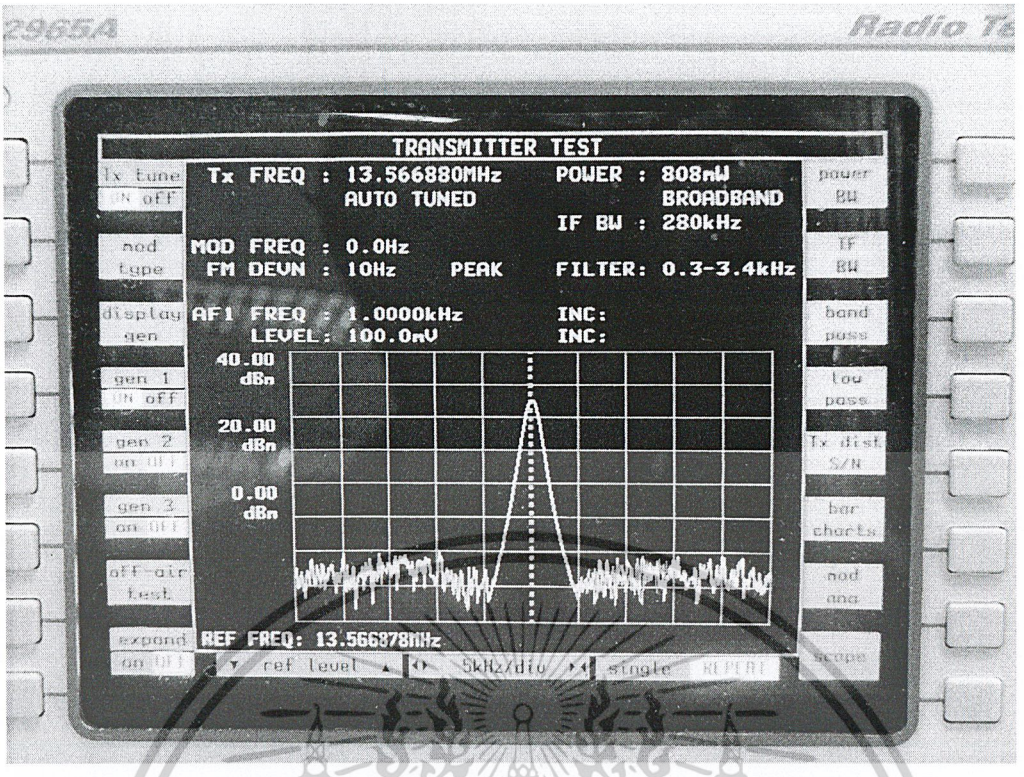
การทดลองหาค่ากำลังงานที่ได้ของวงจร Class E Power Amplifier ใช้เครื่อง Radio Test Set โดยใช้วงจรดังรูปที่ 4.6 โดยใช้ค่าที่ได้จากการคำนวณ มีขั้นตอนการทดลองดังนี้

- ใช้วงจรที่สำเร็จในรูปที่ 4.7
- ใช้ VCC = 9V, C1 = 55.01 pF, L1 = 33 uH, L2 = 2.50 uH, C2 = 238 pF, Q=2N3866
- ใช้สัญญาณจาก CRX14 ที่มีความถี่ 13.56 MHz, 5V จ่ายให้ที่ขาเบสของQ
- ใช้เครื่อง Radio Test Set เป็นโหลด(มีค่าความต้านทานภายใน 50Ω)



รูปที่ 4.23 แสดงการวัดกำลังงานที่ใช้เครื่อง Radio Test Set เป็นโหลด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.24 แสดงผลลัพธ์ที่ได้จากการวัดหาค่ากำลังงาน

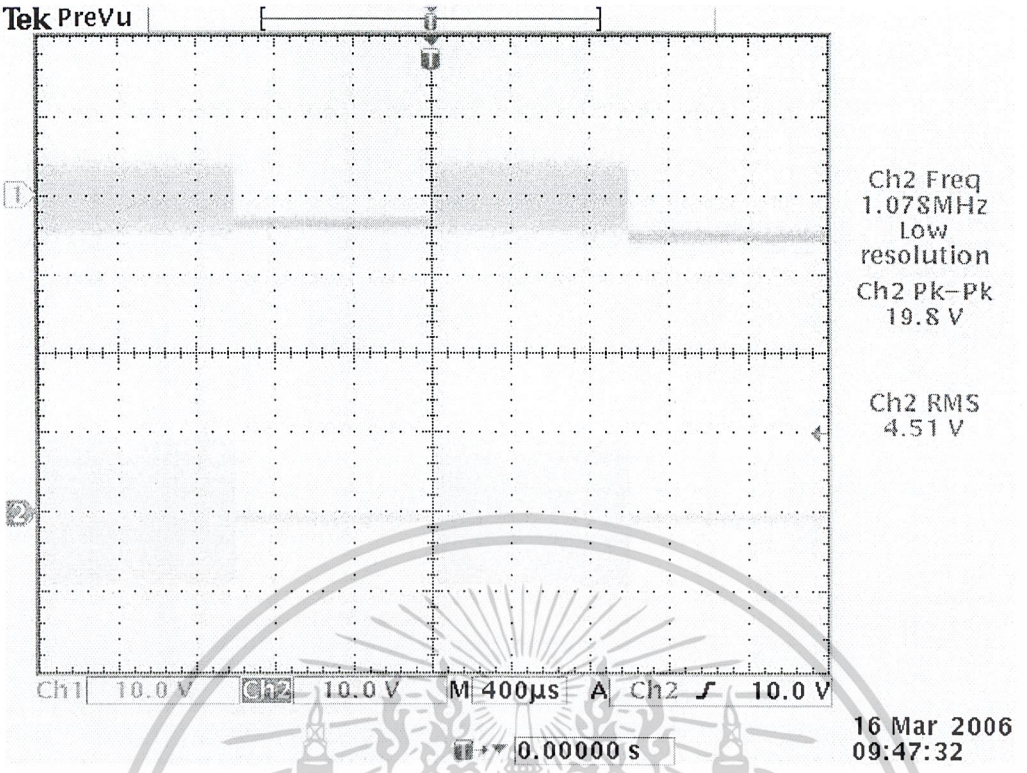
จากการใช้เครื่อง Radio Test Set หาค่ากำลังงานมีค่าเท่ากับ 808 mW

การทดลองเมื่อสัญญาณอินพุทเป็นแบบ ASK

โดยใช้วงจรดังรูปที่ 4.6 โดยใช้ค่าที่ได้จากการคำนวณ มีขั้นตอนการทดลองดังนี้

- ทำการต่อวงจร ได้วงจรที่สำเร็จแสดงรูปที่ 4.7
- ใช้ VCC = 9V, C1 = 55.01 pF, L1 = 33 uH, L2 = 2.50 uH, C2 = 238 pF, Q=2N3866 , ใช้วงจรสายอากาศเป็น Load
- ป้อนสัญญาณแบบ ASK ที่ความถี่ 13.56 MHz , 5V ที่ ขาเบสของQ
- บันทึกค่าตกคร่อมโหลด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.25 แสดงสัญญาณ ASK ที่วัดตกคร่อมโหลดและสัญญาณอินพุท

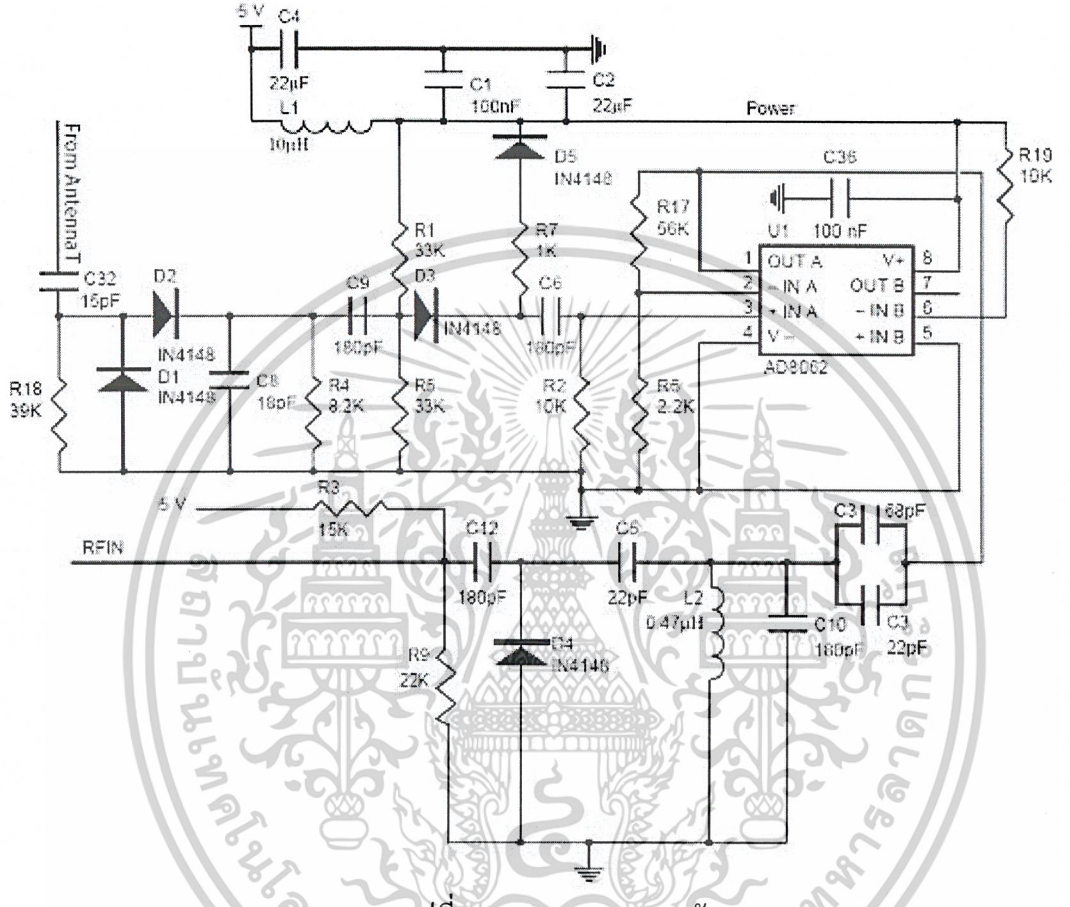
- สัญญาณสีเหลืองเป็นสัญญาณอินพุท
- สัญญาณสีฟ้าเป็นสัญญาณตกคร่อมโหลด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.3 การทดลองทางภาครับ

วัตถุประสงค์

- เพื่อทดสอบว่าที่ภาครับสามารถรับสัญญาณแบบ SINE WAVEได้
- เพื่อทดสอบว่าที่ภาครับสามารถรับสัญญาณแบบ ASK ได้



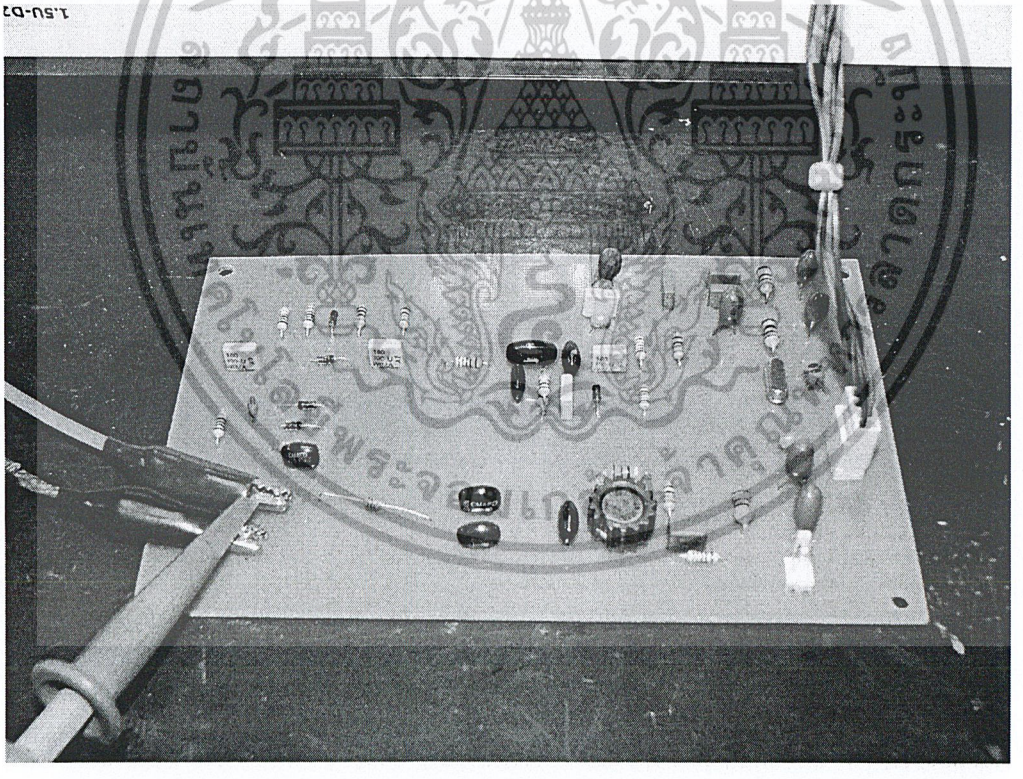
รูปที่ 4.26 แสดงวงจรภาครับ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ทำการทดลองภาครับสัญญาณแบบ SINE WAVE

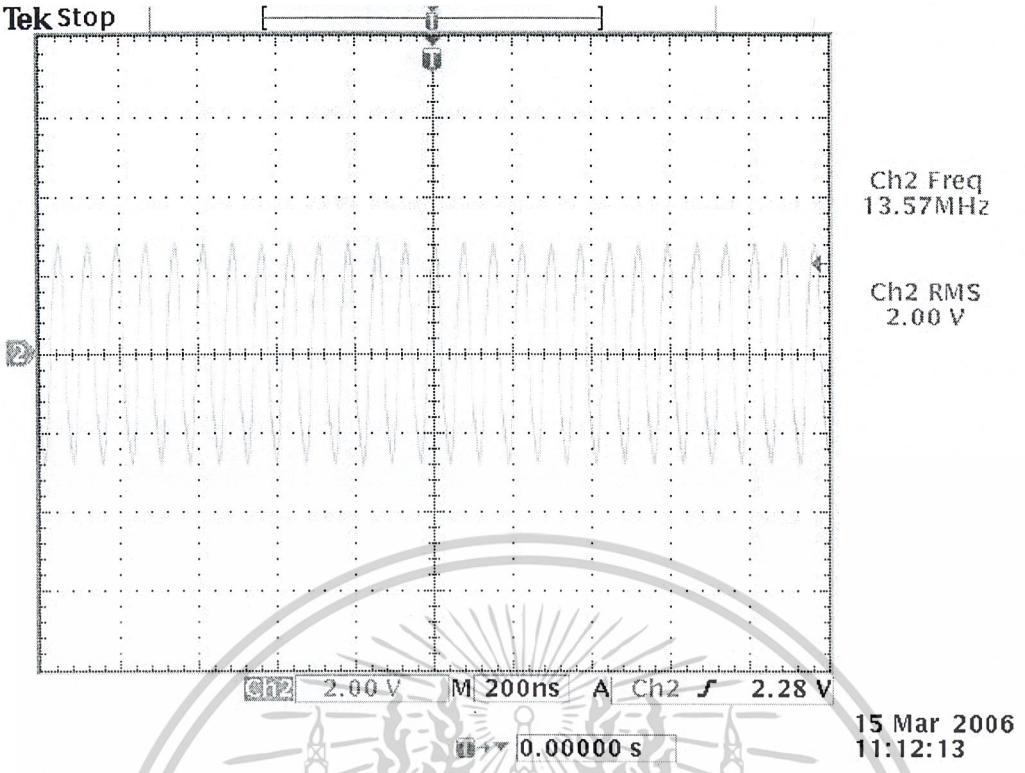
โดยใช้วงจรดังรูปที่ 4.26 ในการทดลองทางภาครับมีขั้นตอนการทดลองดังนี้

- ใช้ VCC = 5 V
- ป้อนสัญญาณ sine waveที่ความถี่ 13.5 MHz , 2 Vp-p ที่C32 (From Antenna)
- วัดสัญญาณที่ขา C32 แล้วทำการบันทึกค่า
- วัดสัญญาณที่ขา K ของ D1 แล้วทำการบันทึกค่า
- วัดสัญญาณที่ขา K ของ D2 แล้วทำการบันทึกค่า
- วัดสัญญาณที่ขา A ของ D3 แล้วทำการบันทึกค่า
- วัดสัญญาณที่ขา 3 ของ AD8062 แล้วทำการบันทึกค่า
- วัดสัญญาณที่ขา L2 แล้วทำการบันทึกค่า
- วัดสัญญาณที่ขา K ของ D4 แล้วทำการบันทึกค่า
- วัดสัญญาณที่ RFIN แล้วทำการบันทึกค่า

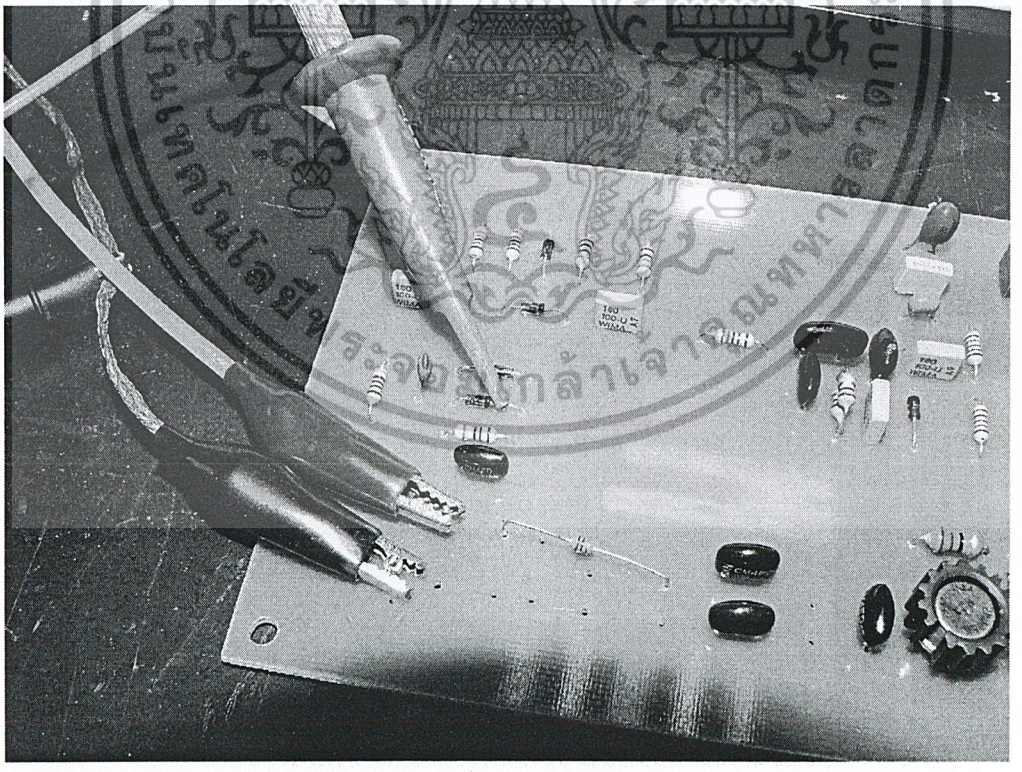


รูปที่ 4.27 แสดงการวัดที่จุดC32 ในภาครับ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

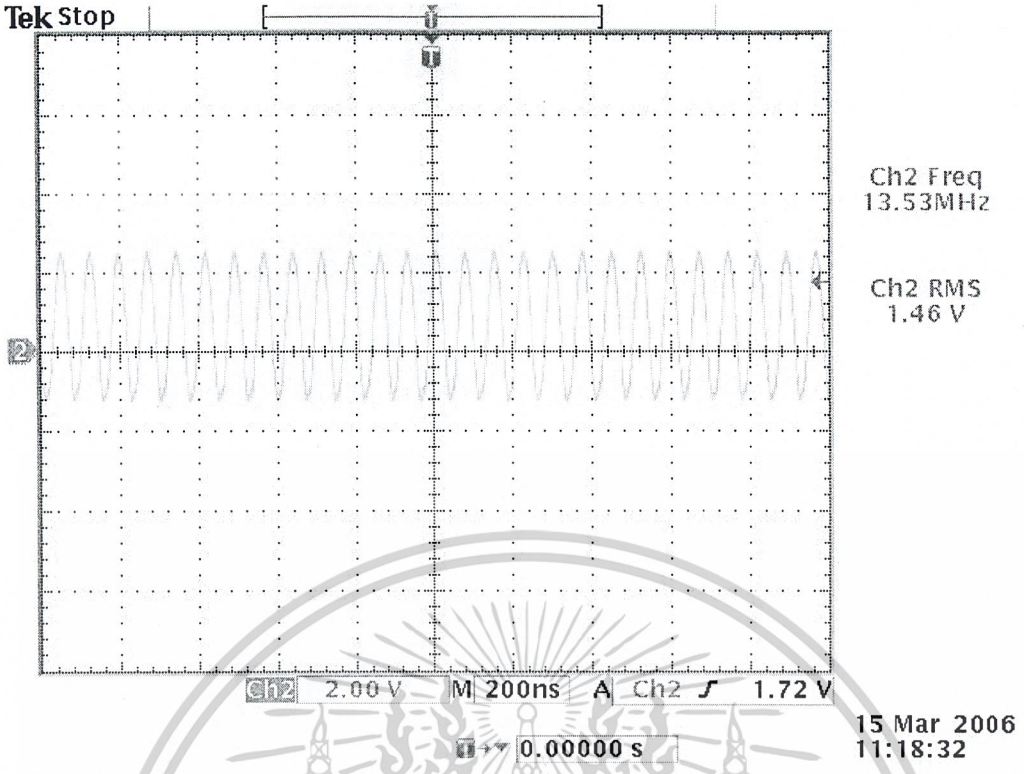


รูปที่ 4.28 แสดงสัญญาณที่จุด C32 ในภาครับ

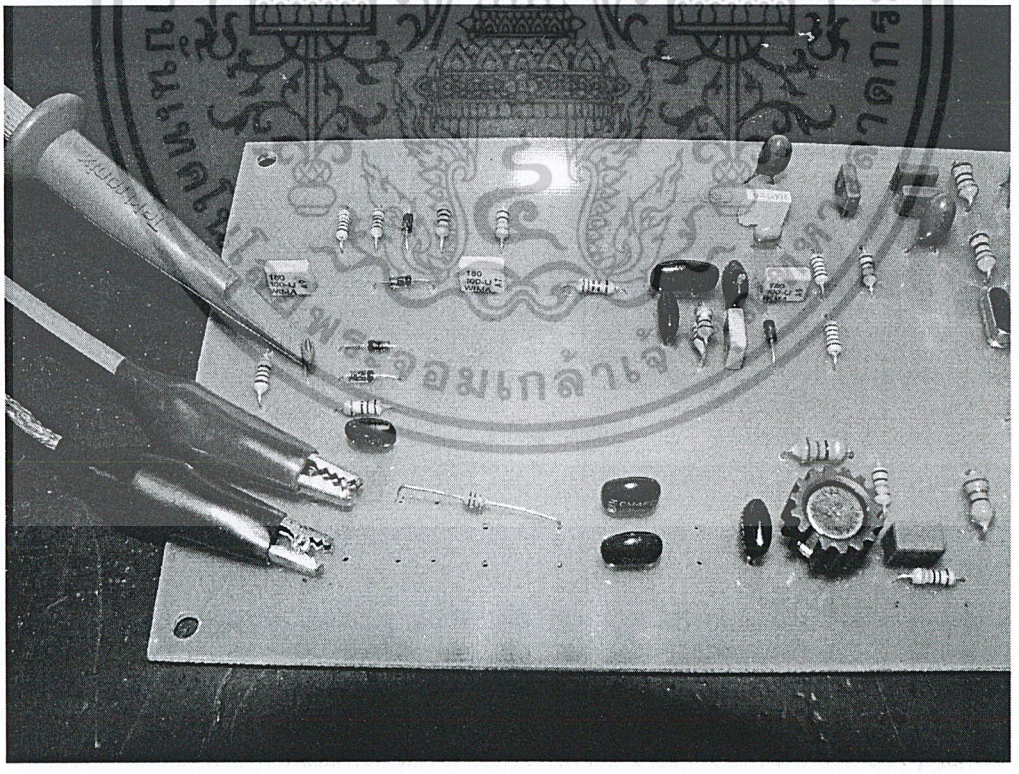


รูปที่ 4.29 แสดงการวัดขา K ของ D1 ในภาครับ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

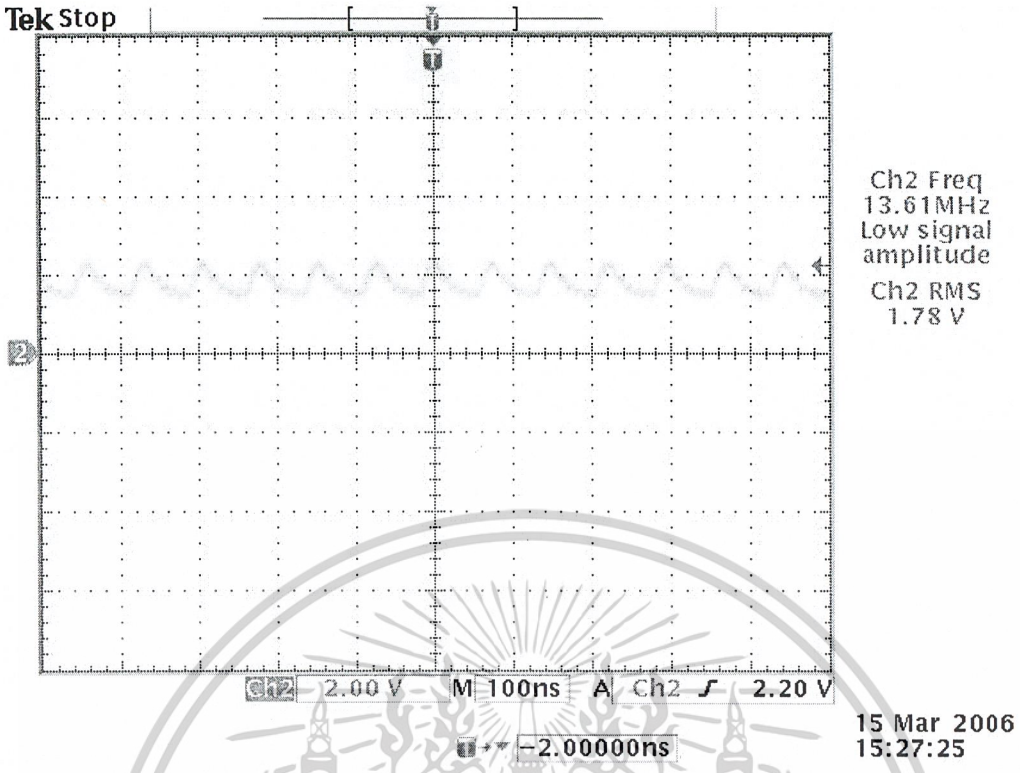


รูปที่ 4.30 แสดงสัญญาณที่ขา K ของ D1 ในภาครับ

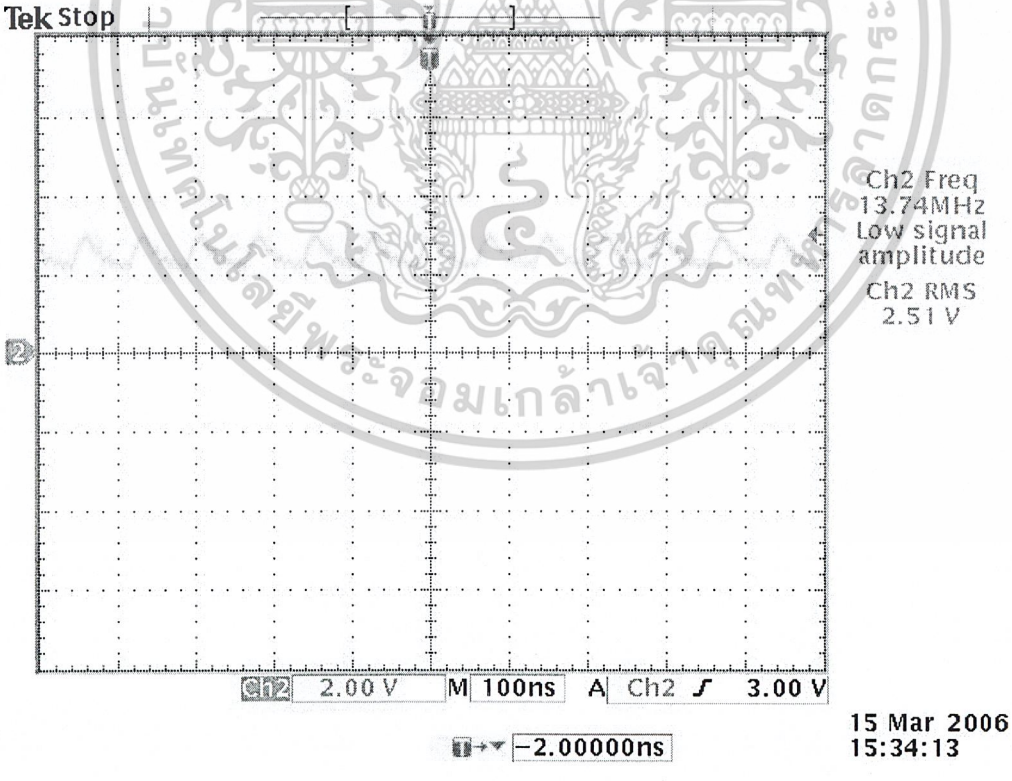


รูปที่ 4.31 แสดงการวัดขา K ของ D2 ในภาครับ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

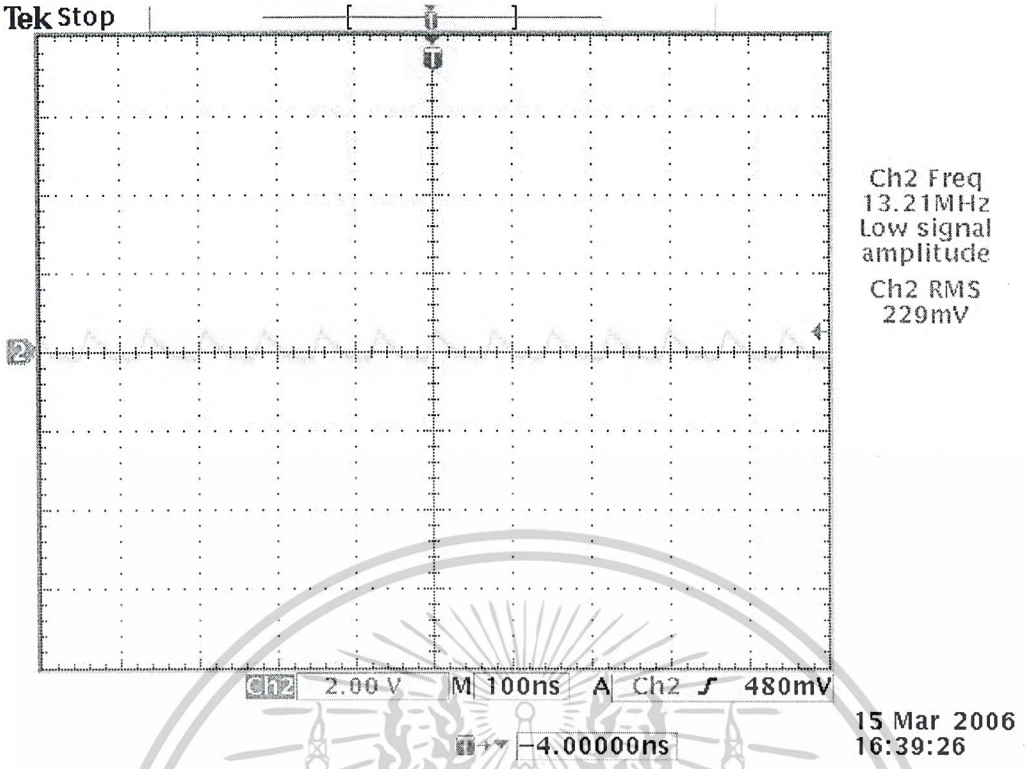


รูปที่ 4.32 แสดงสัญญาณที่ขา K ของ D2 ในภาครับ

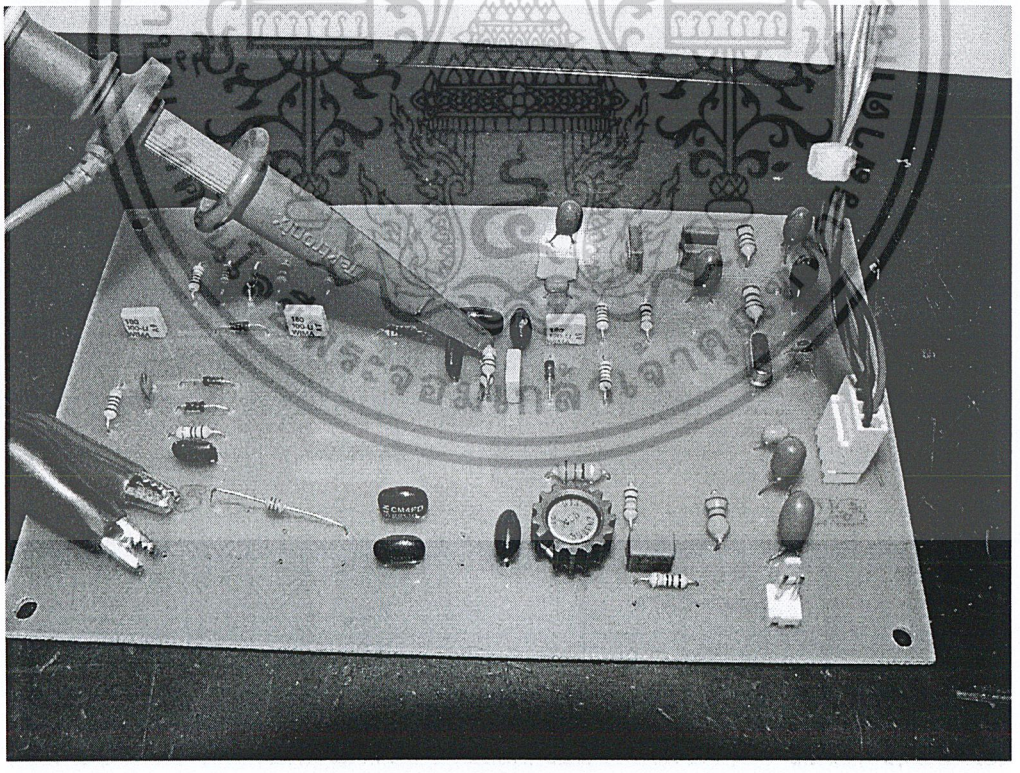


รูปที่ 4.33 แสดงสัญญาณที่ขา A ของ D3 ในภาครับ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

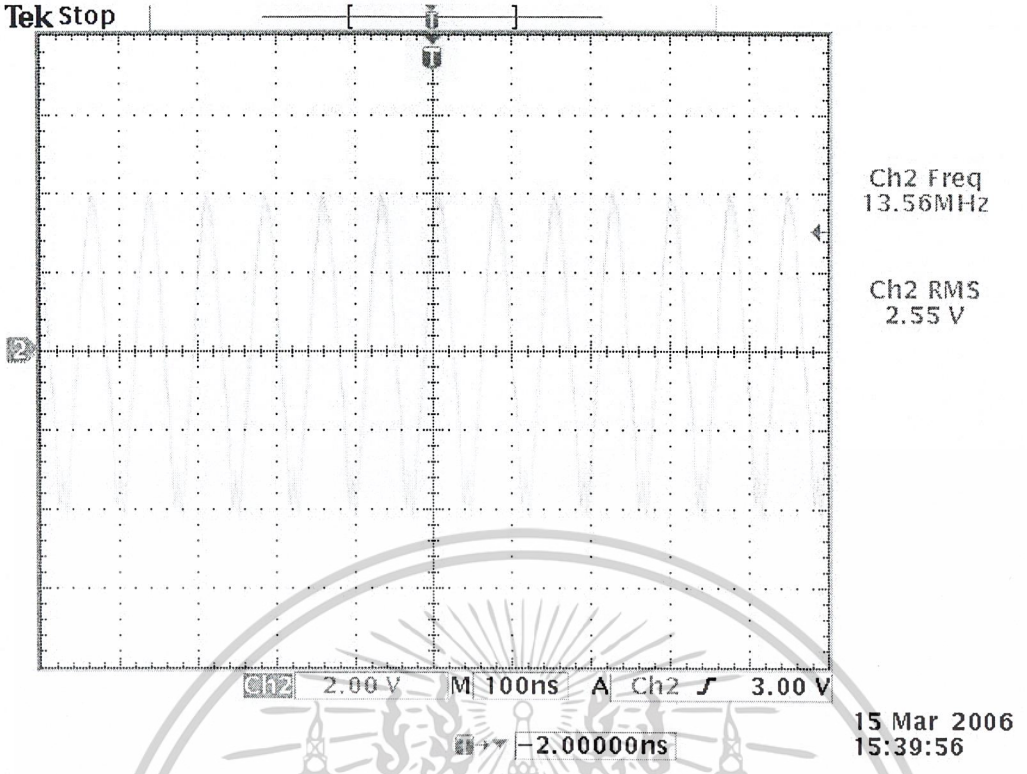


รูปที่ 4.34 แสดงสัญญาณที่ขา 3 ของ AD8062 ในภาครับ

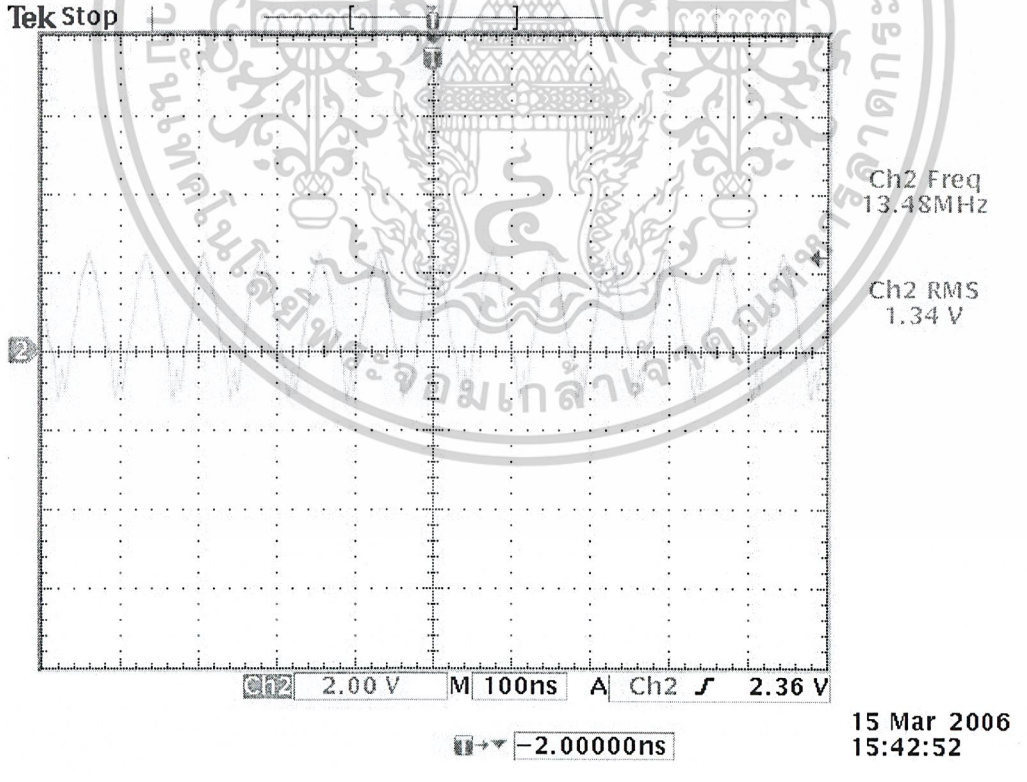


รูปที่ 4.35 แสดงการวัดขา L2 ในภาครับ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

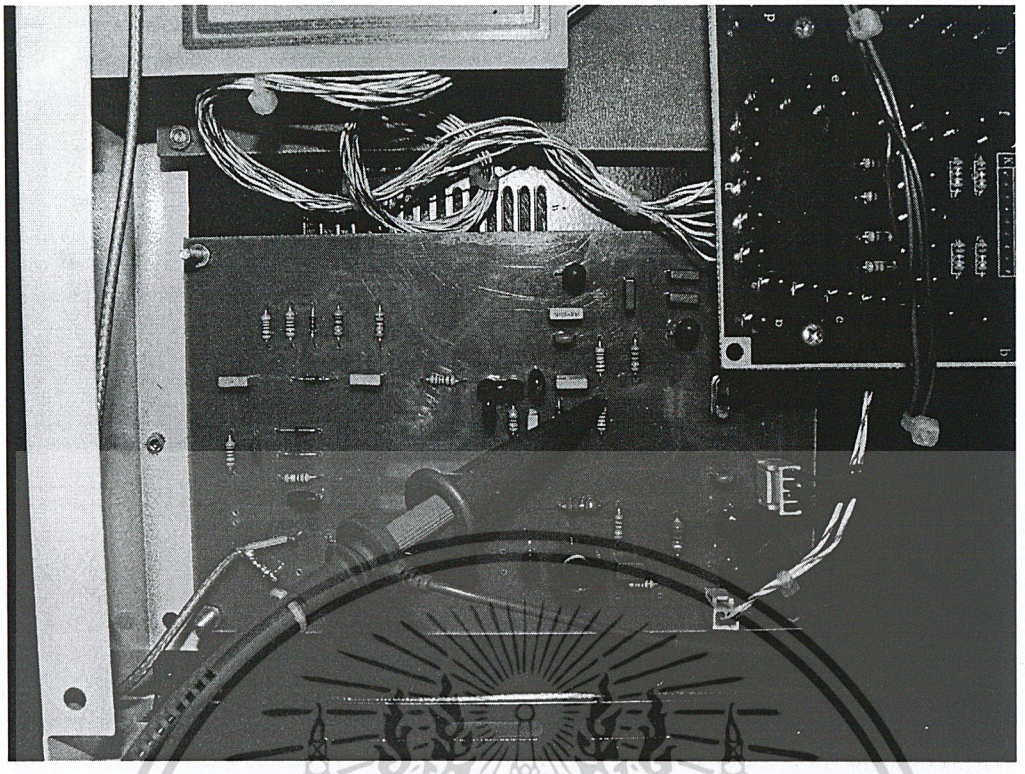


รูปที่ 4.36 แสดงสัญญาณที่ขา L2 ในภาครับ

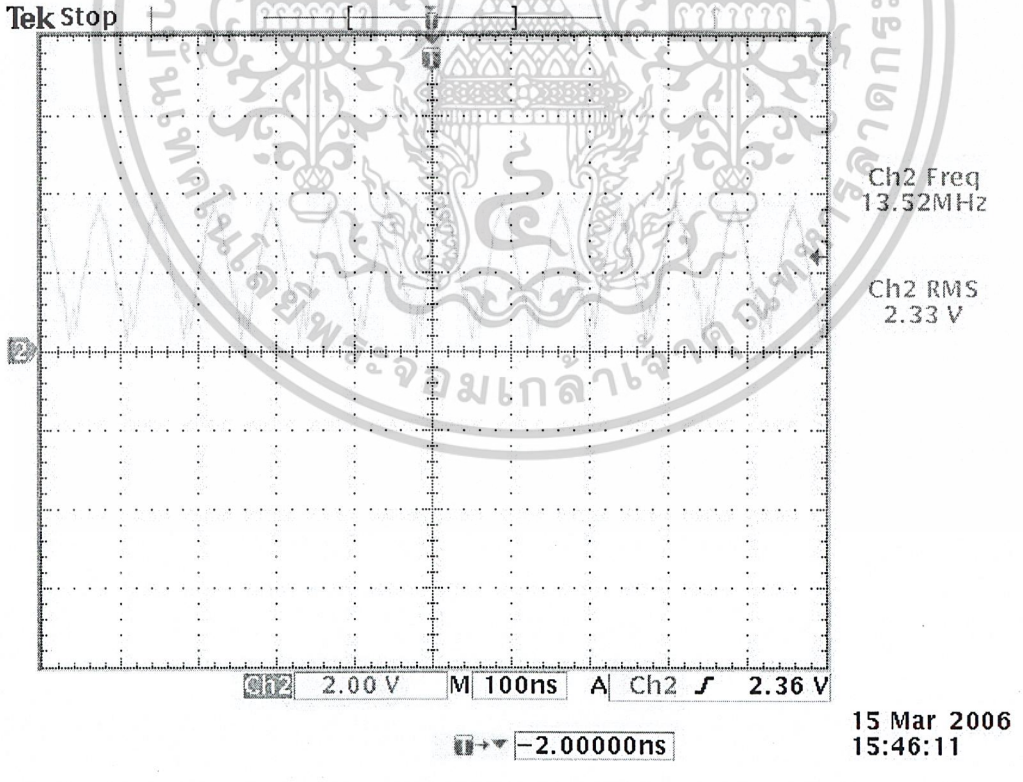


รูปที่ 4.37 แสดงสัญญาณที่ขา K ของ D4 ในภาครับ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.38 แสดงการวัดสัญญาณที่ RFIN ในภาครับ



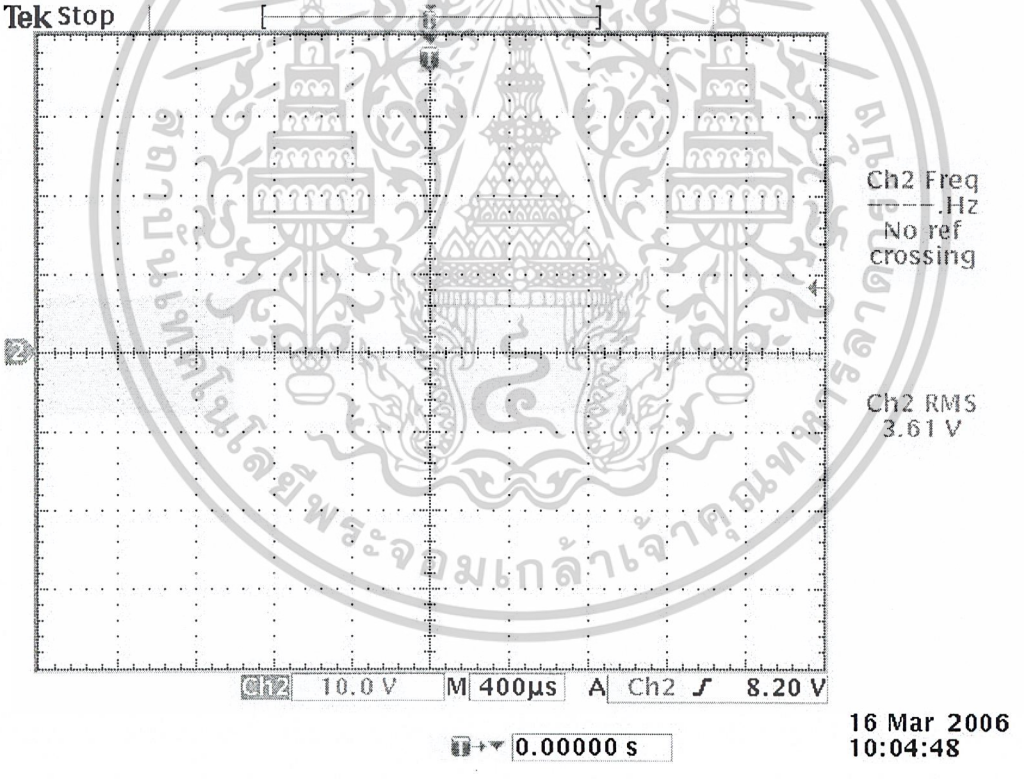
รูปที่ 4.39 แสดงสัญญาณที่ RFIN ในภาครับ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ทำการทดลองภาครับสัญญาณแบบ ASK

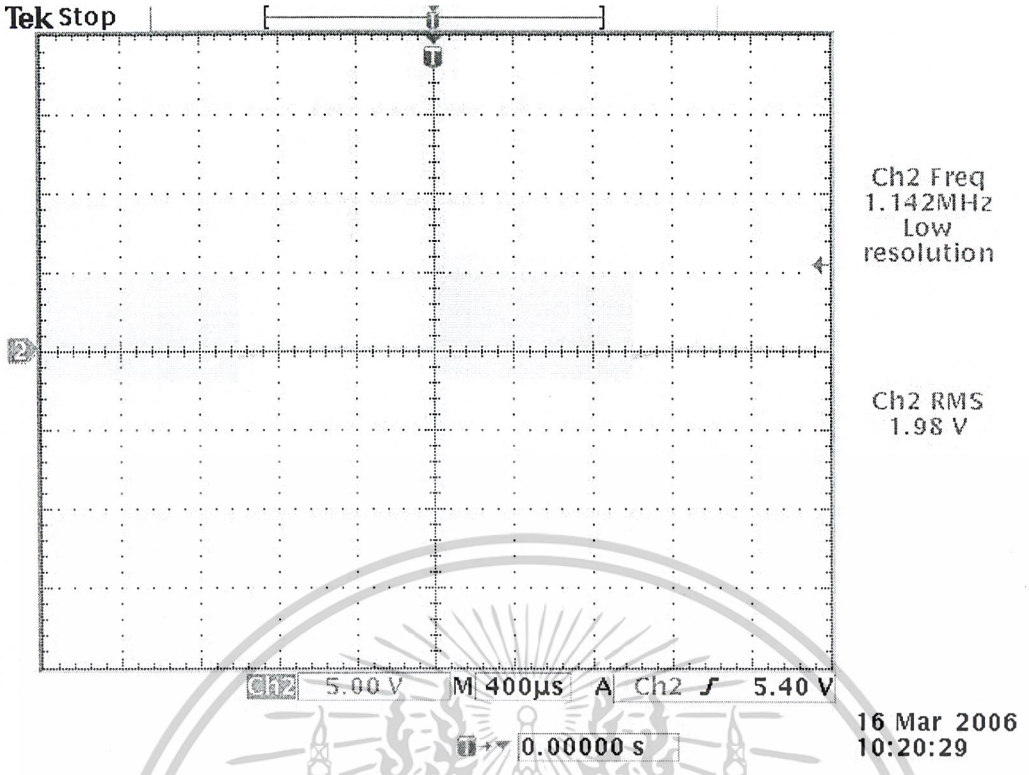
โดยใช้วงจรดังรูปที่ 4.26 ในการทดลองทางภาครับมีขั้นตอนการทดลองดังนี้

- ใช้ VCC = 5 V
- ป้อนสัญญาณแบบ ASK ที่ความถี่ 13.5 MHz, 2 Vp-p ที่C32 (From Antenna)
- วัดสัญญาณที่ขา C32 แล้วทำการบันทึกค่า
- วัดสัญญาณที่ขา K ของ D1 แล้วทำการบันทึกค่า
- วัดสัญญาณที่ขา K ของ D2 แล้วทำการบันทึกค่า
- วัดสัญญาณที่ขา A ของ D3 แล้วทำการบันทึกค่า
- วัดสัญญาณที่ขา 1 ของ AD8062 แล้วทำการบันทึกค่า
- วัดสัญญาณที่ขา L2 แล้วทำการบันทึกค่า
- วัดสัญญาณที่ขา K ของ D4 แล้วทำการบันทึกค่า
- วัดสัญญาณที่ RFIN แล้วทำการบันทึกค่า

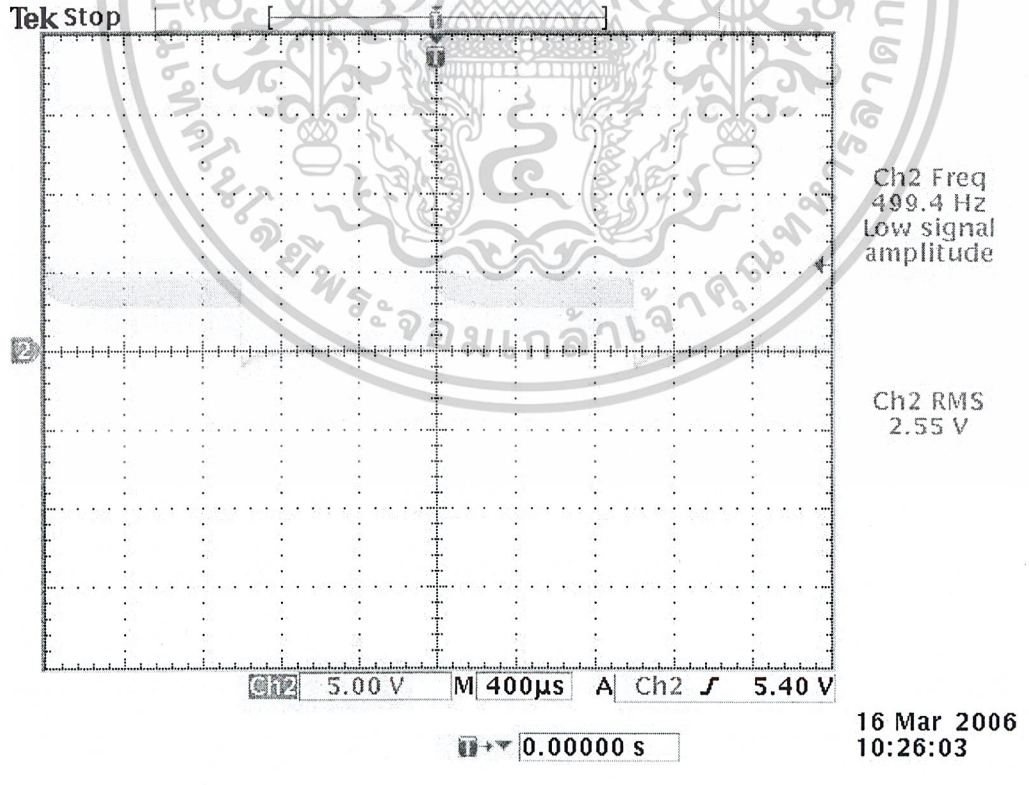


รูปที่ 4.40 แสดงสัญญาณที่จุด C32 ในภาครับที่เป็นสัญญาณแบบ ASK

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

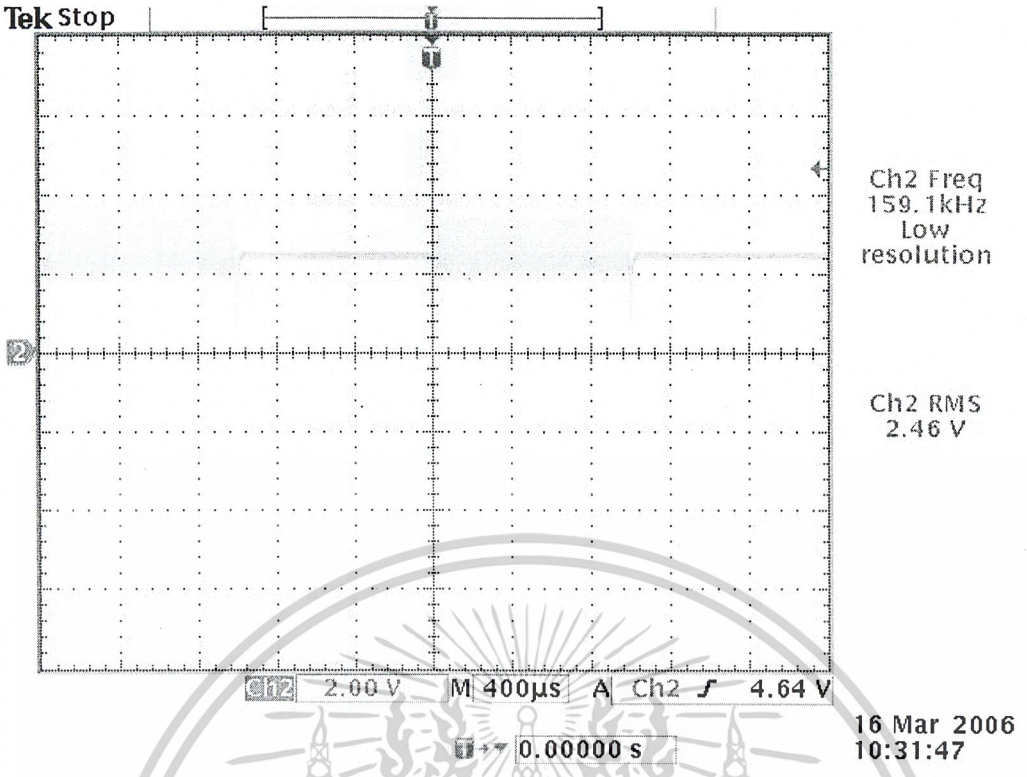


รูปที่ 4.41 แสดงสัญญาณที่ขา K ของ D1 ในภาครับที่เป็นสัญญาณแบบ ASK

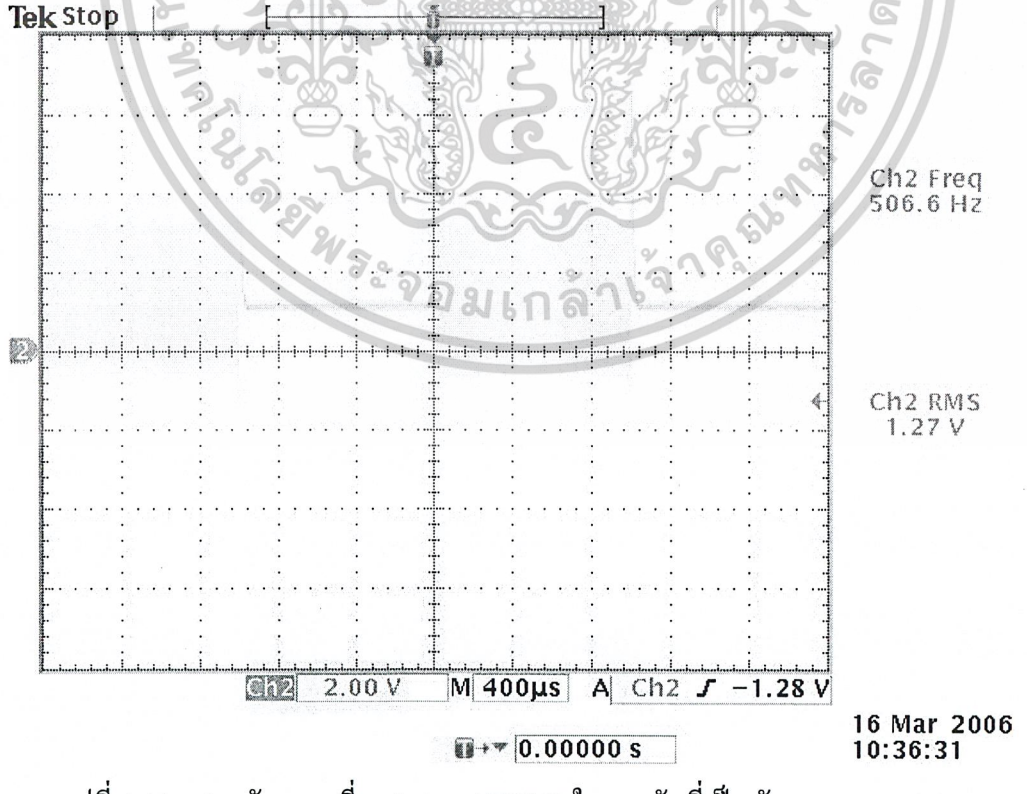


รูปที่ 4.42 แสดงสัญญาณที่ขา K ของ D2 ในภาครับที่เป็นสัญญาณแบบ ASK

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

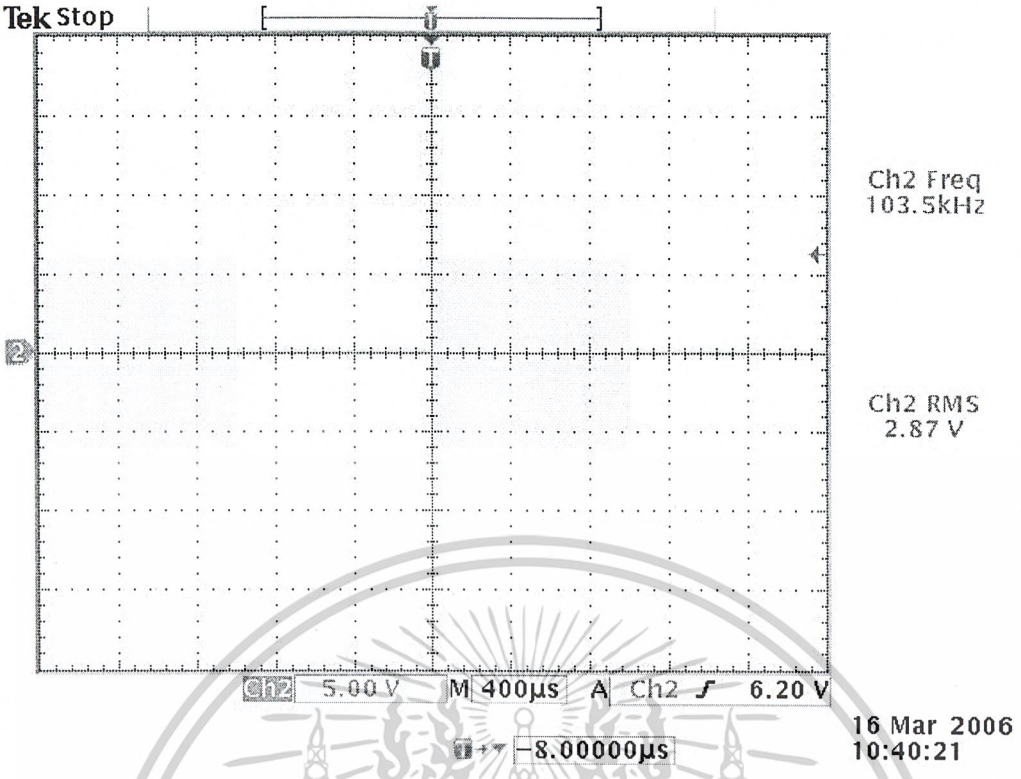


รูปที่ 4.43 แสดงสัญญาณที่ขา A ของ D3 ในภาครับที่เป็นสัญญาณแบบ ASK

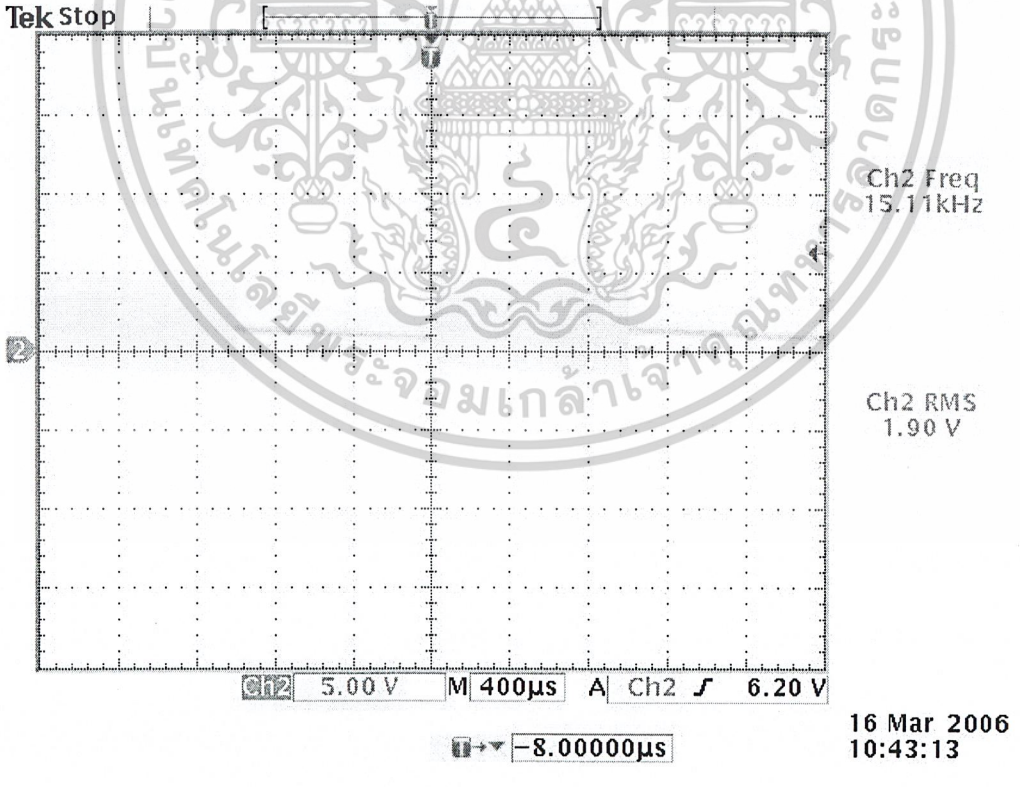


รูปที่ 4.44 แสดงสัญญาณที่ขา 1 ของ AD8062 ในภาครับที่เป็นสัญญาณแบบ ASK

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

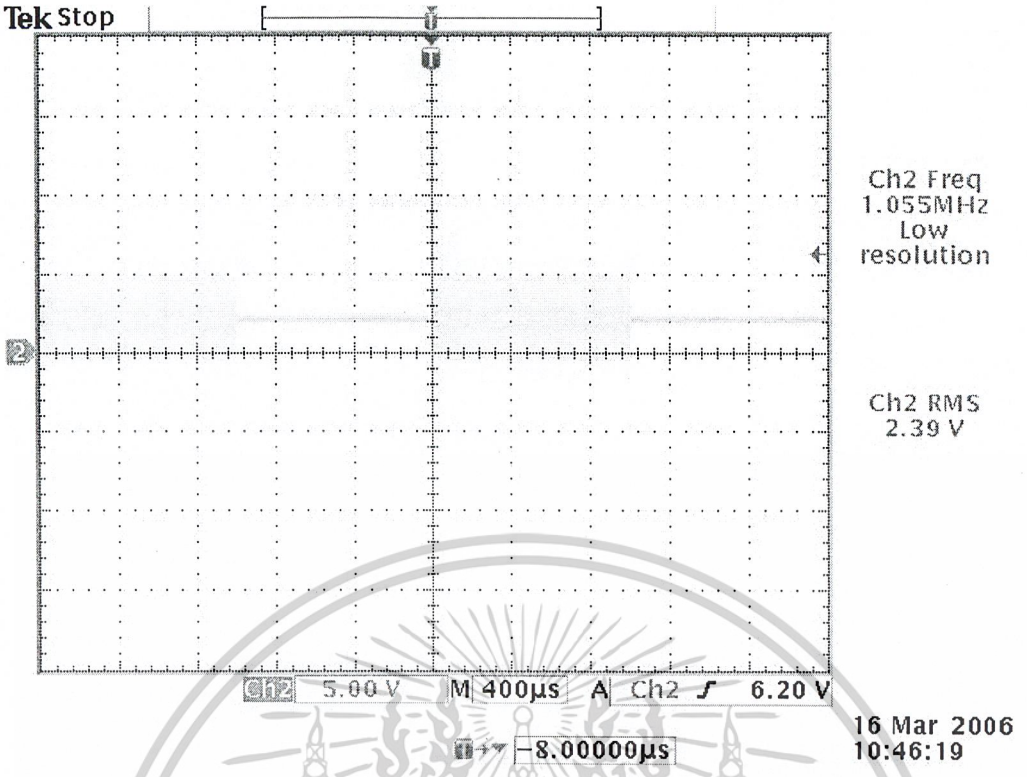


รูปที่ 4.45 แสดงสัญญาณที่ขา L2 ในภาครับที่เป็นสัญญาณแบบ ASK



รูปที่ 4.46แสดงสัญญาณที่ขา K ของ D4 ในภาครับที่เป็นสัญญาณแบบ ASK

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.47 แสดงสัญญาณที่ RFIN ในภาครับที่เป็นสัญญาณแบบ ASK

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 5
บทสรุป

5.1 สรุปผลการการทดลอง

จากการทดลองเครื่องอ่านและเขียนดังกล่าว สามารถแบ่งออกเป็น 2 ส่วนที่สำคัญคือ ส่วนทางด้านอนาลอก(Analog) กับส่วนทางด้านดิจิตอล(Digital) และแต่ละส่วนยังแบ่งออกเป็นภาคต่างๆ เมื่อพิจารณาทางด้านอนาลอก อันประกอบด้วย วงจรภาคส่ง วงจรภาครับ และวงจรสายอากาศ โดยทำการทดลองทีละวงจร ในการทดลองภาคส่งเมื่อจ่ายสัญญาณอินพุทเข้าที่ขา RFout ของไอซี CRX14 เพื่อทดสอบการทำงานของวงจรขยายคลาสอี ผลที่ได้คือเมื่อจับสัญญาณที่ขาคอเลคเตอร์ของทรานซิสเตอร์จะได้สัญญาณที่มีระดับแรงดันสูงขึ้น และเมื่อสัญญาณผ่านวงจรกรองความถี่แล้วปรากฏว่าที่ความถี่ 13.3MHz จะมีระดับแรงดันสูงที่สุดโดยสังเกตได้จากกราฟของวงจรภาคส่ง ส่วนการทดลองทางภาครับโดยป้อนสัญญาณเข้าที่จุด DEMOD ผลที่ได้เมื่อจับสัญญาณที่เอาท์พุทของออปแอมป์ จะได้สัญญาณตามความถี่ที่ได้จ่ายไว้ โดยก่อนที่สัญญาณจะผ่านวงจรขยายนั้นมีวงจรที่ใช้ในการตัดสัญญาณในซีกลบและในช่วงที่สัญญาณสูงกว่าแหล่งจ่ายแล้วจึงทำการขยายสัญญาณ และเมื่อได้สัญญาณที่ขยายแล้วจะมีวงจรกรองความถี่ ซึ่งผลที่ได้ ปรากฏว่าที่ความถี่ 14.4 MHz จะมีระดับแรงดันสูงที่สุดโดยสังเกตได้จากกราฟของวงจรภาครับ และในส่วนของการทำสายอากาศ ออกแบบสายอากาศโดยใช้ลายวงจร(PCB) เป็นสายอากาศจากการออกแบบแล้วทำการคำนวณหาค่าความนำเพื่อหาค่าความเก็บประจุมาใช้ในการ MATCHING กันเพื่อให้ได้โหลดตามที่ได้คำนวณไว้ ผลที่ได้จากการทดลองคือเมื่อนำไปทดสอบโดยใช้เครื่อง Spectrum Analyzer ทำให้ทราบว่าสายอากาศตอบสนองในความถี่ประมาณ14 MHz ซึ่งไม่ตรงกับการคำนวณ

เมื่อพิจารณาทางด้านดิจิตอล ประกอบด้วยส่วนวงจรคอนโทรลเลอร์ในด้านซอฟต์แวร์ ที่ใช้ติดต่อกับคอมพิวเตอร์ และซอฟต์แวร์ ที่ใช้ติดต่อกับไอซีCRX14 และวงจรแสดงค่าตัวเลข หลังจากที่ได้ทำการออกแบบวงจรและสร้างวงจร แล้วทำการทดลองทีละส่วนโดยทำการทดลองในส่วนของการติดต่อกับคอมพิวเตอร์ โดยใช้ Hyper Terminal ซึ่งในการทดลองรับค่าและส่งค่ากลับไปให้คอมพิวเตอร์ ซึ่งผลลัพธ์ที่ได้เป็นไปตามค่าที่ได้ส่งไป และส่วนของการทดสอบในวงจรแสดงค่าตัวเลข ซึ่งได้แสดงค่าถูกต้องตามข้อมูลที่ได้กำหนดให้

ในการพัฒนาปริญญานิพนธ์ฉบับนี้ ในด้านการออกแบบวงจรภาคส่งในการคำนวณค่าความเก็บประจุ และค่าความนำ ตามสูตรนั้นค่าที่ได้ออกมานั้นมีค่าไม่ตรงกับการใช้งานจริง เนื่องจากเป็นค่าที่ไม่มีขายตามท้องตลาด การพัฒนาจึงต้องทำการทดลองเปลี่ยนค่าอุปกรณ์เพื่อหาค่าที่เหมาะสมที่สุดโดยการวัดสัญญาณที่ผ่านวงจรกรองความถี่ในความถี่ที่เราต้องการ ให้มีค่าแรงดันมากที่สุด และในส่วนของสายอากาศสาเหตุที่ทำให้สายอากาศตอบสนองไม่ตรงกับความถี่ที่

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ต้องการ อาจมาจากการสร้างสายอากาศที่มีค่าผิดไปจากการคำนวณ สามารถทำการแก้ไขโดยการเปลี่ยนค่าตัวเก็บประจุในค่าที่ใกล้กับที่ได้คำนวณ

ในส่วนชองโปรแกรม ที่ใช้ Microsoft Visual Basic 6.0 เป็นโปรแกรมเพื่อพัฒนาการติดต่อระหว่างคอมพิวเตอร์กับคอนโทนเลอร์ MCS 51 ในการทดสอบโปรแกรมพบว่าเมื่อการส่งข้อมูลระหว่างคอมพิวเตอร์กับคอนโทนเลอร์นั้น เมื่อการส่งข้อมูลแล้วไม่มีการตรวจสอบการรับการส่งซึ่งกันและกันแล้วจะทำให้โปรแกรมไม่สามารถทำงานได้เพราะเวลาในการประมวลผลของ Microsoft Visual Basic 6.0 กับ เวลาในการประมวลผลของ คอนโทนเลอร์ MCS 51 มีค่าไม่เท่ากัน ซึ่งทำให้โปรแกรมไม่สามารถทำงานได้ สามารถแก้ไขได้โดยทำการ เพิ่มข้อมูลเติมท้ายให้กับ ข้อมูลที่ต้องการจะรับหรือส่งเพื่อใช้ในการตรวจสอบไปในตัว

ในส่วนของ คอนโทนเลอร์ เมื่อได้รับคำสั่งมาจากคอมพิวเตอร์แล้วก็จะนำข้อมูลมาตรวจสอบว่าต้องการให้ทำอะไร เมื่อตั้งให้ให้แสดงเวลาค่าของเวลาที่ คอมพิวเตอร์ส่งมาให้นั้นจะอยู่ในรูปแบบรหัส ANSI Code ซึ่งไม่สามารถแสดงผลได้โดยตรงจะต้องทำการแปล ANSI Code ให้อยู่ในรูปแบบ BCD Code แล้วจึงนำไปแสดงผล และ เมื่อต้องการให้ เขียนข้มูลไป Tag หรือต้องการอ่านข้อมูล จาก Tag ก็จะมีคำส่งที่ทำให้ตัว CRX 14 ทำงาน โดยการเขียนค่าไปที่ รีจีสเตอร์ Parameter และ เขียนข้อมูลไปที่ รีจีสเตอร์ Authenticate โดยรีจีสเตอร์ทั้งสองจำเป็นต้องเขียนข้อมูลให้กับตัวชิป และข้อมูลเขียนข้อมูลไป Tag หรืออ่าน ข้อมูลจาก Tag สามรถทำได้โดยผ่าน รีจีสเตอร์ Input/Output ซึ่งในการทดลองโปรแกรมสามารถทำได้ถึงการเขียน Parameter ให้กับตัวชิป ซึ่งสามารถส่งให้ ชิป CRX ผลิตสัญญาณความถี่ 13.56 MHz โดยไม่สามารถเขียนค่า Authenticate ได้เนื่องจากเป็นความรับของบริษัทผู้ผลิต ในการทดลองได้ทำการทดลองโดยไม่ได้มีการเขียนค่าไปที่ Authenticate ซึ่งผลที่ได้คือ ไม่มาสามารถ ติดต่อกับ Tag ได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บรรณานุกรม

1. อภิชาติ ภู่พลับ , “เขียนโปรแกรม Hardware Interface ด้วย VB 6”
2. ชัยวัฒน์ ลิ้มพรจิตรวิไล วรพจน์ กรแก้ววัฒนากุล , “เรียนรู้และปฏิบัติการไมโครคอลโทรล เลอร์ MCS – 51 แบบแฟลซ”
3. ฉัททวุฒิ พืชพล พิชิต สันติกุลานนท์ พร้อมเลิศ หล่อวิจิตร “คู่มือเรียน Visual Basic 6”
4. Nathan O. Sokal , “Class-E RF Power Amplifiers”

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้