สำนักหอสมุดกลาง พระจอมเกล้าลาดกระบัง

ระบบควบคุมอุปกรณ์ไฟฟ้าผ่านเว็บ

CONTROLLING THE ELECTRIC DEVICES

โดย

นางสาว นภาพร เจียมวิจิตรกร

นาย ฤทธิรงค์ ศรีระพันธ์

TK
5105.1993
.N18

เลขหมู่ 73171
เลขทะเบียน
วัน,เดือน,ปี 10 ก.ค. 2550

b 11288112
i

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต

สาขาอิเล็กทรอนิกส์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2548

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ระบบควบคุมอุปกรณ์ไฟฟ้าผ่านเว็บ

CONTROLLING THE ELECTRIC DEVICES

โดย

นางสาว นภาพร เจียมวิจิตรกร รหัส 45010373

นาย ฤทธิรงค์ ศรีระพันธ์ รหัส 45010653

อาจารย์ที่ปรึกษา

รศ.ดร. มนัส สังวรศิลป์

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต

สาขาอิเล็กทรอนิกส์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2548

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ปริญญานิพนธ์ปีการศึกษา 2548

ภาควิชาอิเล็กทรอนิกส์

คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เรื่อง ระบบควบคุมอุปกรณ์ไฟฟ้าผ่านเว็บ (WEB CONTROLLING THE ELECTRIC DEVICES)

ผู้จัดทำ

1.นางสาว นภาพร เจียมวิจิตรกร

2.นาย ฤทธิรงค์ ศรีระพันธ์

รศ.ดร. มนัส สังวรศิลป์ อาจารย์ที่ปรึกษา

(.....)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ระบบควบคุมอุปกรณ์ไฟฟ้าผ่านเว็บ

น.ส. นภาพร เจียมวิจิตรกร รหัส45010373
นาย ฤทธิรงค์ ศรีระพันธ์ รหัส45010653
รศ.ดร. มนัส สังวรศิลป์ อาจารย์ที่ปรึกษา
ปีการศึกษา 2548

บทคัดย่อ

โครงงานนี้เป็นการนำไมโครคอนโทรลเลอร์ และการรับ-ส่งข้อมูลของเครือข่ายคอมพิวเตอร์มาประยุกต์เป็นระบบควบคุมอุปกรณ์ไฟฟ้าภายในบ้านผ่านเครือข่ายคอมพิวเตอร์ โดยแบ่งออกเป็น 3 ส่วน ส่วนแรกเป็นระบบควบคุมอุปกรณ์ไฟฟ้า โดยใช้ แรบบิท (Rabbit) เป็นส่วนประมวลผลกลางติดต่อกับอุปกรณ์ตรวจจับ (Sensor) อินฟราเรด, อุปกรณ์ตรวจวัดอุณหภูมิ และควบคุมอุปกรณ์ไฟฟ้าได้ ส่วนที่สองเป็นส่วนที่ใช้คอมพิวเตอร์ควบคุมเพื่อจะทำการติดต่อกับไมโครคอนโทรลเลอร์(แรบบิท) ผ่านทางพอร์ตอนุกรมและส่วนที่สามเป็นการทำงานผ่านเครือข่ายอินเตอร์เน็ต โดยระบบนี้สามารถที่จะตรวจสอบสถานะของอุปกรณ์ตรวจจับ (Sensor) ตรวจสอบอุณหภูมิหรือการเปิด-ปิดอุปกรณ์ไฟฟ้าผ่านระบบเครือข่ายอินเตอร์เน็ตโดยใช้โปแกรมเว็บบราวเซอร์ (Web Browser) ได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

CONTROLLING THE ELECTRIC DEVICES

Miss. Naphaporn Jeimwijitkorn ID 45010373
Mr. Rittirong Srirapun ID 45010653
Assoc.Prof.Manus Sangworasin Asvisor
Educational Year 2005

Abstract

This project concerns about IC Rabbit and data transmission via computer network to control the electric devices. This project consists of three parts. The first part is controlling the electric device ,showing the infrared sensor's status and temperature's status. All of them we use IC Rabbit to central processing .The second part is computer control which interface with serial port. The third part is controlling the electric devices and show sensor's status via internet by program Web Browser.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กิตติกรรมประกาศ

โครงงานควบคุมอุปกรณ์ไฟฟ้าผ่านเว็บนี้ สำเร็จได้ด้วยความช่วยเหลือของหลายๆฝ่าย ขอบพระคุณ รศ.ดร. มนัส สังวรศิลป์ สำหรับการให้คำปรึกษาต่างๆ และ เข้าร่วมรับฟังการบรรยายโครงงานในวันสอบ ทำให้การบรรยายผ่านไปได้ด้วยดี ขอบคุณพี่ๆทุกๆคนที่ห้องLAB อาคารกรมหลวงนราธิวาสราชนครินทร์ ชั้น 10 ที่ให้ความช่วยเหลือในการทำโครงงานตั้งแต่เริ่มทำจนกระทั่งเสร็จสิ้น ถึงแม้ว่าพี่ๆทุกคนมีงานที่จะต้องทำ แต่ก็ยังสละเวลาอันมีค่ามาช่วยอธิบาย แนะนำและช่วยแก้ปัญหาต่างๆที่เกิดขึ้นของโครงงาน โครงงานนี้มีความจำเป็นที่จะต้องใช้เครื่องมือสำหรับทำอุปกรณ์ฮาร์ดแวร์ ขอบคุณเพื่อนๆ พี่ๆ ห้องPOWER ELECTRONICS ชั้น4 อาคารภาควิชาวิศวกรรมไฟฟ้า ที่ให้หยิบยืมอุปกรณ์ และ แก้ไขวงจรในส่วนที่ได้ทำผิดให้สามารถใช้งานได้ และสุดท้ายขอบคุณคนสำคัญที่ภาควิชาวิศวกรรมไฟฟ้าสำหรับกำลังใจ และความช่วยเหลือในทุกๆอย่างรวมถึงคอยรับส่งเวลาที่ต้องอยู่ทำงานจนถึงเช้า โครงงานนี้อาจจะดูไม่ยิ่งใหญ่สำหรับบางคน แต่สำหรับผู้ทำโครงงานมีความภาคภูมิใจเป็นอย่างยิ่งที่โครงงานนี้สำเร็จได้ด้วยความพยายามที่จะเรียนรู้ในสิ่งใหม่ๆซึ่งไม่เคยได้รู้จักมาก่อน คงต้องขอบคุณตัวเองที่มีความอดทนและไม่ยอมแพ้แม้ว่าจะต้องเจอความกดดันในหลายๆด้าน โครงงานนี้คงจะสำเร็จไม่ได้หากขาดความช่วยเหลือของทุกๆคน ขอขอบคุณด้วยความจริงใจอีกครั้งค่ะ ขอบคุณค่ะ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ

หน้า

กิตติกรรมประกาศ

บทที่ 1 บทนำ

1.1 ส่วน hardware 2

1.2 ส่วน software 2

บทที่ 2 ทฤษฎีและหลักการ

2.1 เว็บเซิร์ฟเวอร์ 3

2.2 ระบบเครือข่ายอินเตอร์เน็ต 4

2.3 เวิลด์ ไวด์ เว็บ คืออะไร 5

2.4 โฮมเพจ 7

2.5 IP แอดเดรส 7

2.6 โปรโตคอล ทีซีพี/ไอพี 8

2.7 ดีเอ็นเอส 13

2.8 HTML 13

2.9 IC MAX 232 14

บทที่ 3 วงจรและการออกแบบ

3.1 วงจรเชื่อมต่อเพื่อควบคุมอุปกรณ์ไฟฟ้า 21

3.2 วงจรภาคส่งสัญญาณอินฟราเรด 22

3.3 วงจรภาครับอินฟราเรด 24

3.4 วงจรตรวจจับอุณหภูมิ 25

บทที่ 4 หลักการและการทำงาน

4.1 วงจรควบคุมอุปกรณ์ไฟฟ้า 27

4.2 วงจรรับ-ส่งอินฟราเรด 28

4.3 วงจรตรวจจับอุณหภูมิ 30

4.4 RCM 3720 30

4.5 Flow chart 35

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 5 ผลการทดลอง

5.1 การทดลองเปิดปิดหลอดไฟ 40

5.2 การทดลองการทำงานของเซ็นเซอร์อินฟาเรด 44

5.3 การทดลองวงจรตรวจจับอุณหภูมิ 47

บทที่ 6 สรุปและวิจารณ์ผลการทดลอง 50

บรรณานุกรม

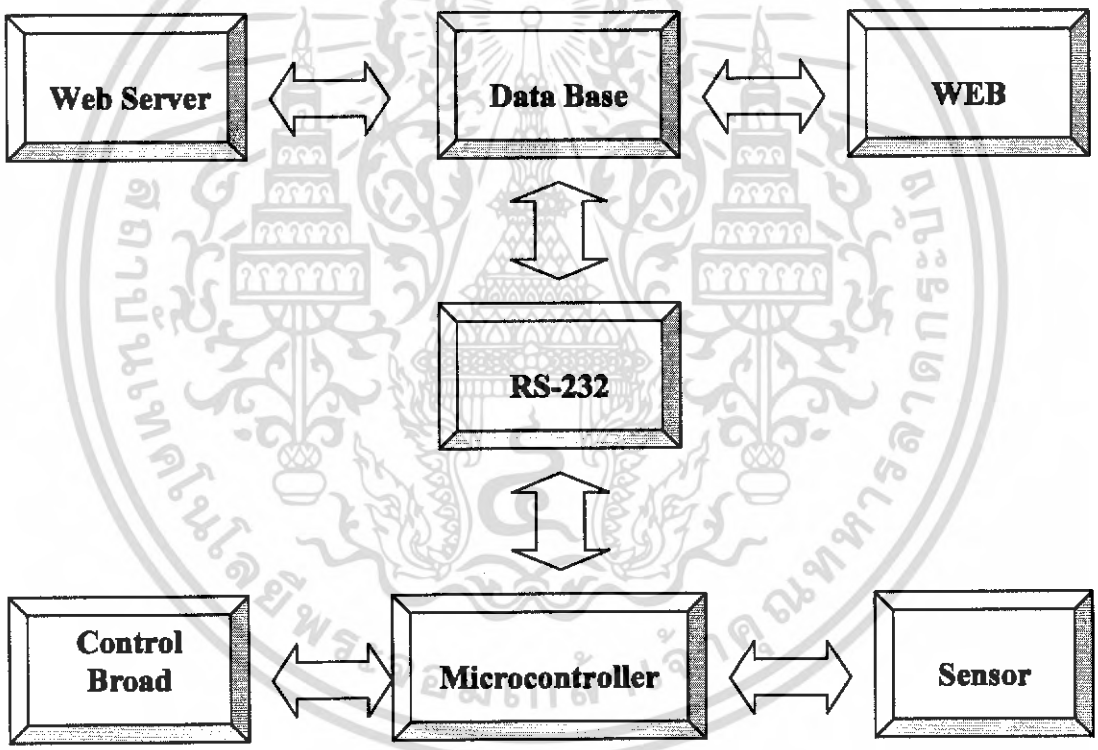
ภาคผนวก

PROGRAM CODE 53

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 1
บทนำ

เทคโนโลยีถูกพัฒนาไปอย่างรวดเร็วโดยเฉพาะทางด้าน Internet ซึ่งปัจจุบัน Internet ถูกใช้อย่างกว้างขวางในชีวิตประจำวัน การนำเอาความรู้ด้าน Internet มาประยุกต์ใช้ในชีวิตประจำวันเพื่อตอบสนองความสะดวกสบายจึงมีความจำเป็นมากขึ้น โดยโครงงานนี้สร้างขึ้นเพื่อการควบคุมอุปกรณ์ไฟฟ้าผ่านโครงข่าย Internet ซึ่งมีหลักการดังนี้



รูปที่ 1.1 แสดงหลักการของการควบคุมอุปกรณ์ไฟฟ้าผ่านเว็บ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หลักการเบื้องต้นของโครงงานนี้ได้ทำ การใช้ ไอซี RABBIT เป็นตัวประมวลผลข้อมูลการสั่งงานควบคุมอุปกรณ์ไฟฟ้า (หลอดไฟ) ซึ่งข้อมูลที่นำมาประมวลผลส่วนหนึ่งมาจาก sensor ต่างๆโดยการสื่อสารผ่านพอร์ตอนุกรมซึ่งมีไมโครคอนโทรลเลอร์ (RABBIT) เป็นตัวควบคุม

นอกจากนี้ยังสามารถใช้งานผ่านเครือข่าย Internet โดยใช้เทคโนโลยี HTML ในการสร้างเว็บเพจ ซึ่งสามารถควบคุมอุปกรณ์ไฟฟ้า และแสดงผลต่างๆจากการประมวลผลข้อมูลผ่านเว็บเพจได้ สามารถแบ่งการทำงานเป็น 2 ส่วน คือ

1.1 ส่วน hardware ประกอบด้วย

1. Sensor ในโครงงานนี้จะใช้ sensor 2 ชนิดคือ sensor ตรวจจับอุณหภูมิ และ sensor อินฟราเรด โดย sensor ทุกตัวเชื่อมต่อกับไมโครคอนโทรลเลอร์เพื่อจัดเรียงข้อมูลและส่งข้อมูลผ่านพอร์ตอนุกรมไปบันทึกในฐานข้อมูลของ computer server เพื่อใช้ในการประมวลผล

2.บอร์ดควบคุมอุปกรณ์ไฟฟ้า เป็นส่วนที่ทำหน้าที่เปิด/ปิด อุปกรณ์ไฟฟ้าโดยรับข้อมูลจาก computer server ผ่านไมโครคอนโทรลเลอร์ ไปยัง วงจรควบคุมอุปกรณ์ไฟฟ้า

1.2 ส่วน software ประกอบด้วย

1. Visual C เป็นโปรแกรมส่วนกลางที่ใช้ในการประมวลผลการสั่งงานอุปกรณ์ไฟฟ้า การควบคุมเปิด/ปิด การแจ้งข้อผลผ่านเว็บเพจ

2. Web server เป็นส่วนที่ใช้เก็บเว็บเพจต่างๆของระบบเพื่อให้ computer client เรียกเว็บเพจที่ต้องการไปใช้ในการควบคุมอุปกรณ์ไฟฟ้าผ่าน Internet บน Windows 2000 server

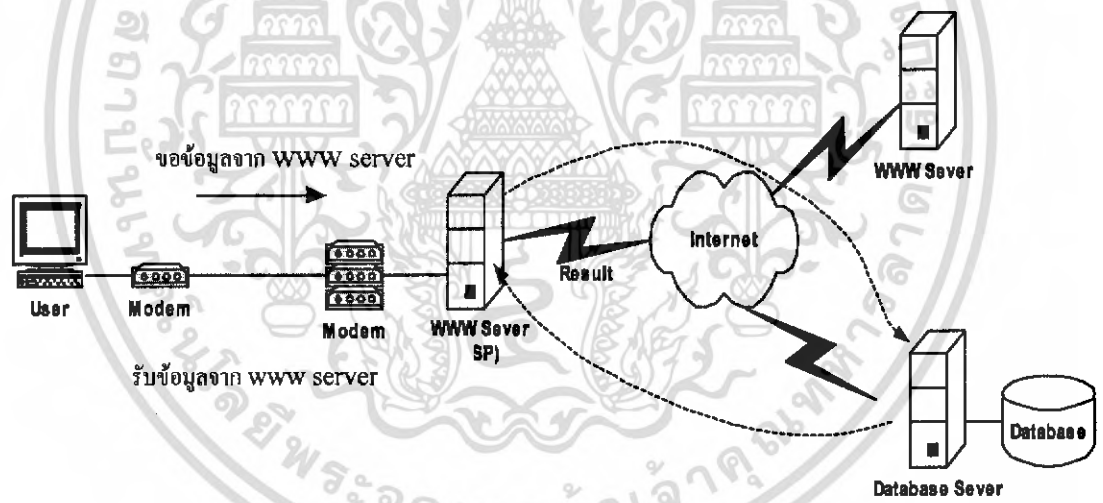
3. HTML เป็นเทคโนโลยีที่ใช้ในการเขียนเว็บเพจ โดยเว็บเพจที่เขียนขึ้นนี้จะมีหลักการทำงานควบคุมเหมือนกับโปรแกรม VB application สำหรับการเขียนเว็บเพจนี้จะใช้โปรแกรม Dream Weaver

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 2
ทฤษฎีและหลักการ

2.1 เว็บเซิร์ฟเวอร์

เว็บเซิร์ฟเวอร์ คือ แอพพลิเคชั่นทำหน้าที่รับ และประมวลผลเอกสาร ที่ถูกร้องขอจากผู้ใช้บริการอินเตอร์เน็ท จากนั้นเว็บเซิร์ฟเวอร์ก็จะส่งเอกสารกลับไปแสดงผลให้ผู้ใช้บริการผ่านบราวเซอร์ นอกจากเว็บเซิร์ฟเวอร์จะถูกนำมาให้บริการในอินเตอร์เน็ตแล้ว แต่อาจมีการประยุกต์ให้นำมาใช้กับเครือข่ายภายในองค์กร หรือ อินเตอร์เน็ตได้เช่นกัน



รูปที่ 2.1 แสดงภาพรวมของ www ซึ่งเป็นบริการที่ได้รับความนิยมสูงสุดของอินเตอร์เน็ต

แต่เดิมนั้นเว็บเซิร์ฟเวอร์มักจะอยู่ในเครื่องคอมพิวเตอร์ ยูนิกซ์ ที่มีกระสิทธิภาพสูง และราคาแพง ต่อมาเมื่ออินเตอร์เน็ตขยายความนิยมมาสู่ผู้ใช้ พีซี ทำให้มีการพัฒนาซอฟแวร์ทำหน้าที่เป็นเว็บเซิร์ฟเวอร์บนพีซี ซึ่งสามารถรันได้ทั้งวินโดวส์ 95/98 และ วินโดวส์เอ็นทีเซิร์ฟเวอร์/Workstation ตัวอย่างเช่น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- NCSA Web Server จาก NCSA เป็นเว็บเซอร์ฟเวอร์ที่สามารถดาวน์โหลดได้ฟรีจากเว็บไซต์ที่ให้บริการดาวน์โหลดฟรีทั่วไป

- Net Server จาก Netscape เป็น เวิลด์ ไวด์ เว็บ (World Wide Web :WWW) เซิร์ฟเวอร์ (Server) ที่มีความสามารถรองรับ จาวา (JAVA) ได้อย่างเต็มรูปแบบ

- ออราเคิลเว็บเซิร์ฟเวอร์ (Oracle Web Server) จากออราเคิล เป็นเว็บเซิร์ฟเวอร์ที่เน้นความสามารถด้านการติดต่อกับเซิร์ฟเวอร์ฐานข้อมูลโดยเฉพาะการใช้งานร่วมกับระบบจัดการฐานข้อมูลของออราเคิล ปัจจุบันเราอาจได้ยินชื่อที่ตั้งขึ้นใหม่เป็น แอพพลิเคชั่นเซิร์ฟเวอร์ (Application Server) คือ ออราเคิล เว็บเซิร์ฟเวอร์ ตัวเดิมที่ถูกพัฒนาขึ้นมา

- Personal Web Server จากไมโครซอฟท์เป็นเว็บเซิร์ฟเวอร์ที่สามารถใช้งานได้กับ วินโดวส์ 95-98 หรือ วินโดวส์เอ็นที เวิร์คสเตชั่น และรองรับการใช้งานร่วมกับ เอเสพี (Active Server Page) โดยเราสามารถดาวน์โหลดมาใช้ได้ฟรีจากเว็บไซต์ของไมโครซอฟท์ แต่ถ้าใช้งาน วินโดวส์ 98 ก็สามารถเลือดติดตั้งได้ ซึ่งมักใช้การทดสอบเว็บเพจ หรือแอพพลิเคชั่นอินเตอร์เน็ต ค่อนจะนำไปใช้งานจริง

- IIS (Internet Information Server) จากไมโครซอฟท์ เป็นอินเตอร์เน็ต เซิร์ฟเวอร์ เวอร์ชั่น 2 ที่แถมมากับวินโดวส์ เอ็นที เซิร์ฟเวอร์ 4.0 มีความสามารถให้บริการได้ทั้ง เวิลด์ ไวด์ เว็บ, เอฟ ที พี (FTP : File Transfer Protocol) และ Gopher ส่วนเวอร์ชั่นที่ใช้งานกับเอเอสพีได้จะเป็นเวอร์ชั่น 3.0 ขึ้นไป

2.2 ระบบเครือข่ายอินเตอร์เน็ต

ในองค์กรหนึ่งๆ อาจมีการติดตั้งระบบเครือข่ายใกล้ตั้งแต่ 2 ระบบขึ้นไปแต่เป็นระบบที่อยู่ห่างไกลกันมาก เช่น อยู่คนละจังหวัด ระบบเครือข่ายแบบใกล้แต่ละระบบก็สามารถถูกเชื่อมโยงเข้าด้วยกันจนกลายเป็นเครือข่ายที่เรียกว่า เครือข่ายระยะไกล (WAN : Wide Area Network) และนอกจากนี้ระบบเครือข่ายระยะไกลที่หนึ่ง ก็สามารถเชื่อมกับเครือข่ายระยะไกลที่อยู่ห่างไกลกันออกไปมากๆ ได้อีก เช่น อยู่คนละประเทศหรือทวีป ทำให้เกิดเครือข่ายขนาดใหญ่ที่เรียกว่า

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

"Internetworking" ซึ่งเป็นระบบเครือข่ายขนาดใหญ่และเป็นหลักการที่กลายมาเป็นระบบเครือข่ายอินเตอร์เน็ตในที่สุด

อินเตอร์เน็ต เป็นเครือข่ายคอมพิวเตอร์ที่มีขนาดใหญ่มาก เกิดจากการเชื่อมต่อเครือข่ายคอมพิวเตอร์จำนวนมากมายในโลกเข้าด้วยกัน ไม่ว่าจะเป็นเครือข่ายขนาดเล็ก เช่น ระบบเครือข่ายแบบใกล้ หรือ ระบบเครือข่ายขนาดใหญ่ เช่น ระบบเครือข่ายของมินิ หรือเมนเฟรมคอมพิวเตอร์ซึ่งแต่ละเครือข่ายก็จะมีเครื่องคอมพิวเตอร์ที่เป็นเครื่องแม่ข่ายหรือ โฮสต์ (Host) ซึ่งมีอยู่หลายชนิดหลายยี่ห้อ โดยจะมีการกำหนดข้อตกลงใจในการสื่อสารที่เรียกว่า โปรโตคอล (Protocol) ขึ้นมาเพื่อให้คอมพิวเตอร์แต่ละชนิดสามารถติดต่อสื่อสารกันได้ โปรโตคอลมาตรฐานที่ใช้ในการสื่อสารบนอินเตอร์เน็ตจะมีชื่อเรียกว่า TCP/IP (Transmission Control Protocal / Internet Protocal)

ถ้าเปรียบให้โปรโตคอลเหมือนภาษาที่ใช้ในการสื่อสารของมนุษย์ซึ่งภาษามนุษย์นั้นมีมากมายหลายภาษาเช่นเดียวกับโปรโตคอล ซึ่งจะมีโปรโตคอลอยู่หลายแบบด้วยกัน TCP/IP นั้นจะเปรียบได้กับภาษาอังกฤษเนื่องจากเป็นโปรโตคอลที่คอมพิวเตอร์ของระบบเข้าใจและมีการใช้โปรโตคอลนี้ร่วมกันเพื่อการเชื่อมต่อเข้าสู่โลกของอินเตอร์เน็ต

2.3 เวิลด์ ไวด์ เว็บ คืออะไร

เมื่อไม่กี่ปีก่อนหน้านี้มีผู้สนใจใช้งานอินเตอร์เน็ตไม่มากนัก เนื่องจากการใช้บริการอินเตอร์เน็ตไม่ว่าจะเป็นการค้นหาข่าวสารข้อมูลการรับส่งจดหมายอิเล็กทรอนิกส์ การทำสำเนาไฟล์ ฯลฯ จะอยู่ในรูปแบบของตัวอักษร (Text Mode) เท่านั้น ไม่มีการเสดงที่เป็น รูปภาพ เสียง และไม่มีตัวอักษรแบบต่างๆ ปรากฏให้เห็นแต่อย่างใด นอกจากนี้ผู้ใช้ต้องเรียนรู้และจดจำคำสั่งคอมพิวเตอร์มากมาย เช่น ต้องเรียนรู้คำสั่งเบื้องต้นของยูนิกส์ (UNIX) เนื่องจากมีการเรียกใช้อินเตอร์เน็ต เครื่องคอมพิวเตอร์ของผู้ใช้จะถูกเปลี่ยนให้เป็นเทอร์มินอล ของโฮสต์ที่ให้บริการอินเตอร์เน็ต และโฮตส์ ส่วนมากจะทำงานอยู่ภายใต้ระบบปฏิบัติการยูนิกซ์ ดังนั้นผู้ใช้จึงต้องเรียนรู้คำสั่งเบื้องต้นของยูนิกซ์ (เช่นเดียวกับการเรียนรู้คำสั่งดอส (Dos) บนเครื่องพีซี) เพื่อทำการป้องกันคำสั่งที่เป็นตัวอักษรด้วยตัวเองให้โฮสต์คอมพิวเตอร์ทำงานตามที่เราต้องการใช้

จนกระทั่งมีบริการที่เรียกว่า เวิลด์ ไวด์ เว็บ หรือเครือข่ายใยแมงมุมเกิดขึ้น ทำให้ความนิยมการใช้อินเตอร์เน็ตสูงขึ้นเป็นทวีคูณ เนื่องจาก เวิลด์ ไวด์ เว็บ เป็นบริการอันหนึ่งที่มีอยู่ในอินเตอร์เน็ต ทำให้การใช้งานอินเตอร์เน็ตเป็นเรื่องที่ง่ายขึ้น ผู้ใช้ไม่ต้องจดจำคำสั่งของยูนิกส์อีกต่อไป การอ่านและค้นหา ข่าวสาร ข้อมูล หรือเอกสารที่เก็บอยู่ในอินเตอร์เน็ตจะอยู่ในรูปแบบของ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สื่อผสม (Multimedia) ที่เรียกว่า "เว็บเพจ" (web page) อันประกอบไปด้วยอักษรข้อความที่มีรูปแบบต่างๆ รูปภาพ เสียง ภาพเคลื่อนไหว วีดีโอ และ ไฮเปอร์เท็กซ์ (Hypertext) ซึ่งเป็นการแสดงข้อความที่มีการเชื่อมโยงถึงกันเหมือนกับการเรียกใช้ Help ในโปรแกรมวินโดวส์ โดยผู้ใช้สามารถเรียกดูเอกสารหนึ่งจากอีกเอกสารหนึ่งได้ ซึ่งเป็นลักษณะการเชื่อมต่อที่คล้ายกับ ใยแมงมุม จึงมีการตั้งชื่อบริการนี้ว่าเป็นเครือข่ายใยแมงมุม

ส่วนประกอบของเวิลด์ ไวด์ เว็บ

การจะใช้บริการ เวิลด์ ไวด์ เว็บ ได้นั้น จำเป็นต้องมีส่วนประกอบเหล่านี้

1. แหล่งข้อมูลหรือเว็บไซต์
2. โปรแกรมเว็บบราวเซอร์

1. แหล่งข้อมูลหรือเว็บไซต์

เว็บไซต์ หรือ เว็บเซิร์ฟเวอร์ คือ ระบบคอมพิวเตอร์ที่เป็นแหล่งเก็บเว็บเพจที่ผู้ใช้บริการสามารถเรียกดูเว็บเพจที่เก็บอยู่ในเว็บไซต์นั้นได้ ซึ่งเครื่องคอมพิวเตอร์ที่ทำหน้าที่เป็นเว็บไซต์อาจจะใช้ระบบปฏิบัติการยูนิกส์ หรือวินโดวส์ เอ็นที ก็ได้ และจะต้องมีการติดตั้งโปรแกรมจัดการจะทำให้เครื่องคอมพิวเตอร์เครื่องนั้นทำหน้าที่เป็นเว็บไซต์ ตัวอย่างเช่น ในกรณีที่เครื่องคอมพิวเตอร์ เครื่องนั้นใช้ระบบปฏิบัติการวินโดวส์ เอ็นที จะมีซอฟท์แวร์เว็บเซิร์ฟเวอร์ให้เลือกใช้ เช่น โปรแกรม Purveyor HTTP Server ของสถาบัน European Microsoft Windows NT Academic Centre (EMWAC) โปรแกรมเว็บไซต์ ของบริษัท O'Reilly and Associate หรือถ้าใช้ระบบปฏิบัติการของยูนิกซ์ก็อาจใช้เว็บเซิร์ฟเวอร์เป็น NSCA http , Apache เป็นต้น

ผู้ที่เป็นเจ้าของเว็บไซต์จะจัดสร้างเว็บเพจของตนเก็บไว้ที่เว็บไซต์นั้นเพื่อให้ผู้ใช้คนอื่นๆ ทั่วโลกสามารถเข้ามาดูข้อมูลที่เก็บไว้ในเว็บไซต์นั้นได้ ตัวอย่างเช่น เว็บเพจของสำนักข่าว CNN จะเก็บอยู่ที่เว็บไซต์ www.cnn.com เป็นต้น

2. โปรแกรมเว็บบราวเซอร์

เป็นโปรแกรมหรือซอฟท์แวร์ที่ใช้ในการเข้าสู่ เวิลด์ ไวด์ เว็บ จะเปิดดูเว็บเพจที่เก็บอยู่ภายในเว็บไซต์ใดๆ ตัวอย่างของ โปรแกรมเว็บบราวเซอร์ ที่มีให้ดาวน์โหลดใช้กันฟรีได้แก่ โปรแกรม Netscape Navigator จากบริษัท Netscape Communications และโปรแกรม Microsoft Internet Explorer จากบริษัทไมโครซอฟท์ ผู้ผลิตโปรแกรมวินโดวส์ที่ผู้คนรู้จักกันดี เป็นต้น ซึ่งโปรแกรมเว็บบราวเซอร์ส่วนใหญ่นั้นจะทำงานร่วมกันกับโปรแกรมวินโดวส์ และนอกจากจะใช้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เพื่อดูเว็บไซต์ใดๆแล้ว หลายโปรแกรมยังมีความสามารถอื่นๆ เช่น บริการส่งจดหมายอิเล็กทรอนิกส์ การค้นหาข้อมูล การโอนถ่ายโปรแกรมด้วย FTP เป็นต้น

2.4 โฮมเพจ

โฮมเพจ (Home page) จะหมายถึงหน้าแรกของเว็บเพจทั้งหมดที่ผู้ใช้บริการอินเตอร์เน็ตจะพบเมื่อมีการเข้าไปยังเว็บไซต์ใดๆ โฮมเพจเปรียบเสมือนกับสารบัญ และคำนำที่เจ้าของเว็บไซต์สร้างขึ้นเพื่อจะใช้ประชาสัมพันธ์องค์กรของตนว่าให้บริการในสิ่งใดบ้าง นอกจากนี้แล้วภายในโอมเพจก็อาจมีเอกสารข้อความอื่นๆที่เชื่อมโยงต่อจากโฮมเพจนั้นๆได้อีก ที่เรียกว่า เว็บเพจ ซึ่งโฮมเพจหนึ่งก็อาจมีการเชื่อมกับเว็บเพจอื่นเป็นจำนวนมากก็ได้

2.5 IP Address

เนื่องจากในระบบเครือข่ายอินเตอร์เน็ตจะใช้โปรโตคอล TCP/IP เป็นมาตรฐานในการสื่อสารข้อมูลซึ่งจะมีการกำหนดหมายเลขประจำตัวที่ไม่ซ้ำกันให้กับเครื่องคอมพิวเตอร์ทุกเครื่องที่มีการเชื่อมต่อกันอยู่ในระบบเครือข่าย หมายเลขประจำตัวนี้จะถูกเรียกว่า IP Address หรือหมายเลข IP โดยมีรูปแบบเป็นชุดของตัวเลข 4 ชุดที่คั่นด้วยเครื่องหมายจุด เช่น 202.44.192.43 ตัวเลขในแต่ละชุดจะมีขนาด 8 บิท ดังนั้นแต่ละชุดจะมีค่าได้ตั้งแต่ 0 ถึง 255 เท่านั้น ซึ่งจะทำให้สามารถกำหนดหมายเลข IP ให้กับเครื่องคอมพิวเตอร์ได้ทั้งหมดถึง 4พันล้านเลขหมายที่ไม่ซ้ำกันเลย

ดังนั้นเมื่อมีการติดตั้งเครื่องคอมพิวเตอร์ที่เป็นโฮสต์คอมพิวเตอร์เพื่อเชื่อมต่อเข้ากับอินเตอร์เน็ตและให้บริการต่างๆจำเป็นต้องขอเลขหมาย IP ประจำหน่วยงาน Internet Network Information Center (InterNIC) ขององค์การ Network Solution Incorporated (NSO) ที่รัฐเวอร์จิเนียร์ สหรัฐอเมริกา แต่ถ้าผู้ใช้สมัครเป็นสมาชิกกับหน่วยงานที่ทำหน้าที่เป็นผู้ให้บริการอินเตอร์เน็ต (Internet Service Provider : ISP) (ในประเทศไทยมีอยู่หลายหน่วยงาน) ก็ไม่ต้องขอหมายเลข IP เนื่องจาก ISP จะเป็นผู้ส่งหมายเลข IP ให้แก่ผู้ใช้หมายเลขเอง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.6 โปรโตคอลทีซีพี/ไอพี (TCP/IP Protocol)

ในหัวข้อนี้จะกล่าวถึงลักษณะการทำงานของอินเตอร์เน็ต ซึ่งประกอบไปด้วยเรื่องเกี่ยวกับ TCP/IP, การกำหนดชื่อ และ เลข IP อินเตอร์เน็ตนับได้ว่าเป็นเครือข่ายคอมพิวเตอร์อื่นๆ เชื่อมโยงเข้ามาใช้งาน หรือเป็นศูนย์กลางเชื่อมโยงเครือข่ายคอมพิวเตอร์อื่นๆ อีก แต่ปัญหาที่เกิดขึ้นในการเชื่อมโยงเครือข่ายคอมพิวเตอร์เข้าด้วยกันคือ แต่ละเครือข่ายใช้คอมพิวเตอร์ต่างชนิด ต่างยี่ห้อ และต่างระบบปฏิบัติการ มาตรฐาน TCP/IP จึงถูกใช้เป็นกุญแจสำคัญในการแก้ปัญหาเหล่านี้โดยจะกลายเป็นระบบเปิดที่สมบูรณ์และที่มีการเชื่อมโยงคอมพิวเตอร์ได้ตั้งแต่พีซีจนถึงเมนเฟรม และไม่จำกัดระบบปฏิบัติการที่ใช้ TCP/IP จึงเป็นมาตรฐานทั่วโลกยอมรับ มีอุปกรณ์และซอฟต์แวร์ผลิตออกมาสนับสนุน TCP/IP มากมาย ดังนั้นจึงนับได้ว่า TCP/IP เป็นหัวใจของอินเตอร์เน็ตเลยทีเดียว

TCP/IP คืออะไร

TCP/IP เป็นข้อกำหนดเกี่ยวกับรูปแบบการเชื่อมโยงในเครือข่าย (Networking Protocol) จัดทำขึ้นเพื่อใช้เป็นเกณฑ์ให้เครื่องคอมพิวเตอร์ใช้งานร่วมกัน ในลักษณะของระบบเปิด (Open System) คือไม่ว่าจะเป็นคอมพิวเตอร์ชนิดใดหรือระบบใดก็ตาม จะสามารถ

ติดต่อสื่อสารและแลกเปลี่ยนข้อมูลกันได้ เช่น เครื่องคอมพิวเตอร์ของ Digital Equipment ซึ่งเป็นระบบ Mini ติดต่อสื่อสารกับ Compaq ซึ่งเป็นเครื่อง PC ได้เมื่อกำเนินการด้วย TCP/IP

TCP/IP เป็นการกำหนดรูปแบบการสื่อสารระหว่างซอฟต์แวร์ การจัดการโอนย้ายข้อมูล การแสดงสถานะของเครื่องคอมพิวเตอร์ที่อยู่บนเครือข่าย ตลอดจนกฎระเบียบต่างๆที่กำหนดให้ทำเมื่อเกิดความผิดพลาดหรือต้องทำเพื่อป้องกันไม่ให้เกิดความผิดพลาด

TCP/IP เกิดจากการนำข้อกำหนดของรูปแบบต่างๆ กันมาใช้ร่วมกัน TCP และ IP ต่างก็เป็นรูปแบบหนึ่งของชุดข้อกำหนดนี้ แต่เรียกชุดข้อกำหนดนี้ว่า TCP/IP ถูกออกแบบมาเพื่อใช้รับส่งหรือโอนย้ายข้อมูลระหว่างคอมพิวเตอร์ที่อยู่บนระบบเครือข่ายเดียวกัน หรือต่างเครือข่ายกันก็ได้และมีการจัดเตรียมข้อมูลสถานะของเครือข่ายขึ้นได้ภายในดังข้อกำหนดรูปแบบเอง ในการสร้างซอฟต์แวร์ของระบบเครือข่ายจะใช้ TCP/IP เป็นส่วนสนับสนุนได้ทั้งระบบเครือข่ายเฉพาะบริเวณ Local Area Network) และเครือข่ายบริเวณกว้าง (Wide Are Network) ไม่ได้ใช้งานเฉพาะกับอินเตอร์เน็ตเท่านั้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.6.1 ส่วนประกอบของ TCP/IP

จากที่กล่าวมาแล้วข้างต้นว่า TCP/IP ประกอบไปด้วยชุดข้อกำหนดรูปแบบต่างๆซึ่งแบ่งเป็นกลุ่มได้ดังนี้

2.6.1.1 กลุ่มข้อกำหนดรูปแบบการขนส่ง (Transport Protocols)

ทำหน้าที่ควบคุมการเคลื่อนย้ายข้อมูลระหว่างคอมพิวเตอร์สองเครื่อง แบ่งย่อยออกได้เป็นสองชนิดคือ

ก. TCP (Transmission Control Protocols) เป็นการบริการแบบ Connection Based Service ซึ่งคอมพิวเตอร์ด้านผู้รับและผู้ส่งต้องต่อถึงกันอยู่ตลอดเวลาในระหว่างการสื่อสาร ถ้าเปรียบเทียบก็คล้ายกับระบบโทรศัพท์ที่ต้องติดต่อกันให้ได้ก่อนจะพูดคุยกันได้

ข. UDP (User Datagram Protocol) เป็นการบริการแบบ Connection Service คอมพิวเตอร์ด้านผู้ส่งไม่จำเป็นต้องติดต่อกับด้านผู้รับก่อน เพียงรู้ที่อยู่ของด้านผู้รับแล้วใส่ที่อยู่นั้นไปกับข้อมูลที่ส่งออก ข้อมูลจะเดินทางตามเส้นทางต่างๆ เพื่อไปถึงปลายทางตามที่อยู่ คล้ายกับการส่งจดหมายที่ไปรษณีย์จะส่งให้ตามที่อยู่ที่จ่าหน้าซองจดหมาย โดยผู้ส่งและผู้รับไม่ต้องติดต่อกัน

2.6.1.2 กลุ่มข้อกำหนดเกี่ยวกับรูปแบบเส้นทาง (Routing Protocol)

ทำหน้าที่พิจารณาเส้นทางที่ดีที่สุดที่ใช้ส่งข้อมูลและถ้ามีข้อมูลเป็นจำนวนมากหรือมีขนาดใหญ่กลุ่มข้อกำหนดรูปแบบนี้ก็จะทำการแบ่งย่อยข้อมูลให้มีขนาดเหมาะสม แล้วส่งออกไป เมื่อถึงผู้รับปลายทาง กลุ่มข้อกำหนดนี้ก็จะทำหน้าที่ตรงข้าม คือรวบรวมข้อมูลย่อยให้ถูกต้องก่อนการแสดงผลกลุ่ม ข้อกำหนดรูปแบบกลุ่มนี้ประกอบด้วย

ก. IP (Internet Protocol)

ข. ICMP (Internet Control Message Protocol) เป็นข้อกำหนดรูปแบบของข้อมูลข่างสารเกี่ยวกับสถานะของ IP เช่นข่าวสารความผิดพลาดและผลกระทบต่อเส้นทางเมื่อมีการเปลี่ยนแปลงฮาร์ดแวร์ในเครือข่าย

ค. RIT (Routing Information Protocol) ข้อกำหนดรูปแบบหนึ่งที่ใช้สำหรับการพิจารณาวิธีการเสือกกเส้นทางเพื่อให้ได้เส้นทางที่ดีที่สุดเหมาะสมกับข้อมูลมากที่สุด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ง. PE (Open Shortest Path First) ข้อกำหนดรูปแบบอีกประเภทหนึ่งที่ใช้ตัดสินเลือกเส้นทางโดยพิจารณาจากเส้นทางที่สั้นที่สุดก่อน

2.6.1.3 กลุ่มข้อกำหนดรูปแบบเกี่ยวกับที่อยู่เครือข่าย (Network Address)

ทำหน้าที่พิจารณาที่อยู่ของเครือข่ายและเครื่องคอมพิวเตอร์ไม่ว่าจะเป็นลักษณะตัวเลขหรือชื่อก็ตาม เพื่อความถูกต้องของข้อมูลที่จะไปยังผู้รับปลายทาง โดยที่ไม่ว่าเครือข่ายจะใหญ่โตสักเพียงใดหรือมีเครื่องคอมพิวเตอร์จำนวนมากก็ตามที่อยู่จะต้องไม่ซ้ำกัน กลุ่มข้อกำหนดรูปแบบกลุ่มนี้มีดังนี้

ก. ARP (Address Resolution Protocol) ข้อกำหนดรูปแบบที่พิจารณาตัวเลขที่อยู่เพื่อไม่ให้เกิดที่อยู่ซ้ำกัน

ข. DNS (Domain Name System) ข้อกำหนดรูปแบบที่พิจารณาตัวเลขที่อยู่เมื่อรู้ชื่อของเครือข่าย หรือเครื่องคอมพิวเตอร์ เพราะในการใช้งานจริงนั้นใช้เพียงที่อยู่ที่เป็นตัวเลข แต่ระบบชื่อจัดทำขึ้นเพื่อให้สะดวกต่อการใช้งานของผู้ใช้

ค. RARP (Reverse Address Resolution Protocol) ข้อกำหนดรูปแบบที่พิจารณาตัวเลขที่อยู่เช่นเดียวกับ ARP แต่จะทำตรงกันข้ามกับ ARP

2.6.1.4 กลุ่มข้อกำหนดรูปแบบเกี่ยวกับเส้นทางการสื่อสารระหว่างเครือข่าย (Gateway Protocols)

สนับสนุนข้อมูลสถานะเพื่อนำไปใช้เลือกเส้นทางที่เหมาะสม ข้อกำหนดรูปแบบเหล่านี้ประกอบด้วย

ก. EGP (Exterior Gateway Protocol) ข้อกำหนดรูปแบบนี้จะทำการถ่ายโอน

ข. GGKP (Gateway – to – Gateway Protocol) เป็นข้อกำหนดรูปแบบที่ทำงานถ่ายโอนข้อมูลเส้นทางระหว่าง Gateway Protocol

ค. IGP (Interior Gateway Protocol) ข้อกำหนดรูปแบบที่ถ่ายโอนข้อมูลเส้นทางกันภายในเครือข่ายเดียวกัน

Gateway เป็นอุปกรณ์หรือเครื่องคอมพิวเตอร์ที่ทำหน้าที่เสมือนประตูสื่อสารเป็นช่องสัญญาณ เข้า หรือออกไปยังระบบสื่อสารอื่น หรือภายในเครือข่ายเดียวกันติดต่อกัน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.6.1.5 กลุ่มข้อกำหนดรูปแบบเกี่ยวกับการบริการผู้ใช้ (User Services)

ผู้ใช้สามารถใช้ข้อกำหนดรูปแบบได้โดยตรง ข้อกำหนดรูปแบบนี้ประกอบด้วย

ก. BOOTP (BOOT Protocol) เมื่อผู้ใช้เปิดเครื่องคอมพิวเตอร์บนเครือข่ายให้เริ่มทำงานข้อกำหนดรูปแบบนี้จะอ่านโปรแกรมควบคุมการทำงานจากคอมพิวเตอร์ให้บริการ (Server Computer)มาให้

ข. FRP (File Transfer Protocol) ข้อกำหนรูปแบบที่ให้บริการถ่ายโอนไฟล์ข้อมูลระหว่างเครื่องคอมพิวเตอร์ ซึ่งอาจจะอยู่บนเครือข่ายเดียวกันก็ได้

ค. TELNET เป็นข้อกำหนดรูปแบบที่ให้บริการเกี่ยวกับการติดต่อระยะทางไกล (Remote Login)

กลุ่มข้อกำหนดรูปแบบอื่น ที่นอกเหนือจากกลุ่มที่จัดไว้และบริการที่สำคัญๆ จัดทำไว้ให้บนระบบเครือข่ายที่สนใจมีดังนี้

- NFS (Network file System) เป็นข้อกำหนดรูปแบบที่ทำให้ผู้ใช้คอมพิวเตอร์เครื่องอื่นได้
- NIS (Network Information Services) เป็นข้อกำหนดรูปแบบที่ให้บริการกับ User Accounts ข้ามเครือจ่าย เช่น Logins และ Password
- RPC (Remote Procedure Call) ข้อกำหนดรูปแบบที่อำนวยความสะดวกให้กับโปรแกรมประยุกต์ที่ใช้งานกับการควบคุมระยะทางไกล
- SMTP (Simple Mail Transfer Protocol) ข้อกำหนดรูปแบบที่ให้บริการถ่ายโอนจดหมายอิเล็กทรอนิกส์ (Electronic Mail) ระหว่างคอมพิวเตอร์

2.6.2 สถาปัตยกรรม TCP/IP

TCP/IP ออกแบบเป็นชุดกำหนดรูปแบบ แบ่งหารบริหารออกเป็นกลุ่มๆ ภายในชุดข้อกำหนดรูปแบบเช่น กลุ่มผู้ใช้บริการ กลุ่มการขนส่ง และกลุ่มเกี่ยวกับเครือข่าย ดังกล่าวมาแล้วจากกลุ่มต่างๆ ที่แบ่งไว้นำไปพัฒนาเป็นสถาปัตยกรรมซึ่งมีลักษณะเป็นระดับชั้น (Layer) ตามกลุ่มการบริการ

สถาปัตยกรรมแบบระดับชั้นมีข้อดีอยู่หลายประการ แต่ละชั้นมีอิสระไม่ขึ้นต่อกันทำให้การเปลี่ยนแปลงการบริการของชั้นใดๆ ไม่ก่อปัญหากับบริการชั้นอื่น การเพิ่มเติมการบริการใหม่ทำได้โดยไม่ต้องเปลี่ยนแปลงโปรแกรมระบบเดิมและสิ่งสำคัญก็คือการทำระดับชั้นนั้นทำให้ตัวโปรแกรมมีขนาดเล็ก สามารถระบุส่วนที่จะต้องปรับปรุงได้แน่นอนไม่ต้องวิตกกังวลถึงโปรแกรมส่วนอื่นทำให้การพัฒนาประสิทธิภาพของระบบทำได้ง่าย และดียิ่งขึ้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ชุดข้อกำหนดรูปแบบ TCP/IP เมื่อนำไปใช้งานจะอยู่ระหว่างระดับชั้นที่เป็นฮาร์ดแวร์กับระดับชันที่เป็นซอฟท์แวร์อื่น เมื่อติดตั้งกับบางระบบ

TCP/IP เมื่อนำไปใช้งานกับระบบ Windows ต้องเพิ่ม Drivers พิเศษเข้าไปด้วยเพื่อให้ Windows รู้จักข่าวสารที่เป็นแพ็กเกจ (Packet) และส่งแพ็กเกจเข้าสู่ระบบเครือข่ายได้ แต่ถ้าเป็นซอฟต์แวร์ที่ใช้ชุดข้อกำหนดรูปแบบ TCP/IP อยู่แล้วข่าวสารจากโปรแกรมประยุกต์จะผ่านมายังระดับชั้น TCP/IP ทำการประมวลส่งเข้าเครือข่ายโดยให้ระบบปฏิบัติการทำการส่ง

กำหนดชื่อและเลข IP

เครือข่ายอินเตอร์เน็ตใช้รหัสหมายเลขมากำหนดให้แต่ละเครือข่ายและเครื่องคอมพิวเตอร์ภายในเครือข่ายที่เชื่อมโยง เรียกรหัสหมายเลขนี้ว่า อินเตอร์เน็ตแอดเดรส (Internet Address) หรือนิยมเรียกกันทั่วไปว่า IP แอดเดรส

IP แอดเดรส ประกอบด้วย เลขฐานสองจำนวน 32 บิต แบ่งออกเป็น 4 ส่วน แต่ละส่วนมี 8 บิต เมื่อดูเฉพาะแต่ละส่วนเป็นฐานสิบจะได้เลขจำนวน 256 ค่าไม่ซ้ำกัน (0-256) IP แอดเดรส จะนำเอาหมายเลขทั้ง 4 ส่วนมารวมกัน โดยแยกแต่ละส่วนด้วยจุด ดังนั้นหมายเลขทั้งหมดที่เป็นไปได้โดยไม่ซ้ำกันคือ 256 กำลัง 4 หรือ 4294937296 จำนวนมีค่าหมายเลขจาก 000.000.000.000 จนถึง 255.255.255.255 หมายเลขเหล่านี้เองที่อินเตอร์เน็ต ใช้กำหนดให้กับเครือข่าย และเครื่องคอมพิวเตอร์เพื่อใช้อ้างอิงถึง IP แอดเดรส บางหมายเลขสงวนไว้ใช้ด้วยจุดหมายกรณีพิเศษ ทำให้ IP แอดเดรส ที่ใช้งานทั่วไป ลดลงจาก จำนวนที่เป็นไปได้ ความหมายของ IP แอดเดรส จะแบ่งได้เป็น 2 กลุ่มดังนี้

- กลุ่มที่ใช้เป็นรหัสประจำเครือข่าย
- กลุ่มที่ใช้เป็นรหัสประจำเครื่องคอมพิวเตอร์ที่อยู่ภายในเครือข่าย (Host Computers)

IP แอดเดรส ในกลุ่มรหัสประจำเครื่องคอมพิวเตอร์ สามารถซ้ำกันได้ แต่ละกลุ่มรหัสประจำเครือข่ายจะซ้ำกัน ไม่ได้ ดังนั้นรหัสเครื่องที่ซ้ำกันจึงไม่มีผลต่อการอ้างอิงถึง นอกจากนี้เพื่อความเหมาะสมในการกำหนด IP แอดเดรส ให้กับผู้ขอ ทางผู้บริการอินเตอร์เน็ตแบ่งคลาสของผู้ขอ IP แอดเดรสตามขนาดของเครือข่าย เพื่อให้ทรัพยากรในส่วนนี้ถูกใช้อย่างคุ้มค่าที่สุดองค์กรขนาดใหญ่ก็จะจัดให้อยู่ในคลาสที่สามารถกำหนด IP แอดเดรส ให้กับเครื่องคอมพิวเตอร์ในเครือข่ายได้มาก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การกำหนดหมายเลขของเครือข่ายและเครื่องคอมพิวเตอร์ นอกจากจะแบ่ง IP แอดเดรส เป็นคลาสทั้ง 5 ประเภทแล้ว ยังมีข้อกำหนดปลีกย่อยอีกหลายประการที่ผู้วางระบบต้องรู้ไว้เพื่อที่จะกำหนดแอดเดรสใช้งานได้อย่างถูกต้องไม่ผิดพลาด

2.7 ดีเอนเอส (Domain Name System : DNS)

การใช้หมายเลข IP ซึ่งเป็นตัวเลขล้วนๆในการอ้างอิงถึงเครื่องคอมพิวเตอร์แต่ละเครื่องนั้น จะมีข้อเสียคือ จำยากและก่อให้เกิดความสับสนได้ง่าย จึงมีการพัฒนาวิธีการอ้างอิงถึงหมายเลข IP แบบใหม่ที่เรียกว่า DNS ขึ้นมา

DNS เป็นเทคนิคการเปลี่ยนหมายเลข IP ที่เป็นตัวเลขให้เป็นตัวอักษรแทนหมายเลข IP เป็น 202.12.97.1 ผู้ใช้บริการสามารถเขียนเป็นชื่อโดเมนคือ kkul.kku.ac.th แทนเครื่องคอมพิวเตอร์ที่เรียกว่าดีเอนเอสจะทำหน้าที่ในการแปลงชื่อจากโดเมนเนมให้เป็นหมายเลข IP อีกทีหนึ่ง

2.8 HTML

โฮมเพจและเว็บเพจใดๆ โดยส่วนใหญ่จะถูกสร้างขึ้นมาด้วยภาษาที่เรียกว่า HTML (Hyper Text Markup Language) หรือ " ภาษาสำหรับทำเครื่องหมาย " ประกอบด้วยคำสั่ง (Tags) ที่ใช้ในการกำหนดว่าเว็บเพจจะมีข้อความอะไร มีการแสดงรูปภาพ เสียง และภาพวีดีโอที่ตำแหน่งใด นอกจากนี้ ยังมีคำสั่งสำหรับการเชื่อมโยงเว็บเพจหนึ่งไปยังอีกเว็บเพจหนึ่งและไปยังบริการอื่นๆ ในอินเตอร์เน็ตอีกด้วย

การสร้างเว็บเพจนอกจากจะสร้างโดยการเขียน HTML บนโปรแกรม Editer ตัวใดตัวหนึ่ง เช่น โปรแกรม Notepad บนวินโดวส์แล้ว ยังสามารถใช้โปรแกรมช่วยสร้างเว็บเพจ กำหนดรูปร่างหน้าตาของเว็บเพจได้โดยตรง โดยที่ผู้สร้างไม่จำเป็นต้องเขียนหรือเรียนรู้ HTML แต่อย่างใด ซึ่งในปัจจุบันจะมีโปรแกรมช่วยสร้างเว็บเพจหลายตัว เช่น โปรแกรม Microsoft Frontpage โปรแกรม Netscape Navigator Gold เป็นต้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.9 IC MAX 232

เป็น IC ที่ทำหน้าที่เปลี่ยนแรงดันที่เข้ามาจาก serial port ไปเป็นแรงดันตามมาตรฐานของ RS-232 โดยเปลี่ยนระดับแรงดัน TTL เพื่อทำให้ใช้กับไมโครคอนโทรลเลอร์

2.9.1 การสื่อสารพอร์ตนุกรม RS-232

ลักษณะของการส่งข้อมูลแบบอนุกรมนั้น ข้อมูลที่ส่งออกมาทีละบิตจากตัวส่งไปตัวรับข้อมูลช่องสัญญาณในการส่งข้อมูลอาจใช้เพียง 1 หรือ 2 ช่องสัญญาณเท่านั้น ทำให้ค่าใช้จ่ายในการสื่อสารจะถูกกว่าแบบขนาน ในการส่งข้อมูลแบบอนุกรม ข้อมูลที่ต้องการส่งจะอยู่ในลักษณะเป็นไบต์จะทยอยส่งทีละบิต และทางตัวรับจะต้องรับข้อมูลเข้ามาทีละบิด แล้วมารวมกันเป็นไบต์ซึ่งทางตัวรับต้องคอยตรวจสอบว่า บิตใดเป็นบิตเริ่มต้นหรือบิตสุดท้ายของข้อมูล การตรวจสอบนั้นจะขึ้นอยู่กับรูปแบบของรหัสของบิตข้อมูลที่ใช้ ซึ่งในการรับส่งข้อมูลแบบอนุกรมระหว่างไมโครคอมพิวเตอร์กับอุปกรณ์ภายนอกนั้นจำเป็นต้องมีมาตรฐานในการรับส่งข้อมูล ซึ่งมาตรฐานที่นิยมมากที่สุดก็คือ มาตรฐาน RS-232

2.9.2 มาตรฐาน RS-232

มาตรฐานพอร์ตอนุกรมแบบ RS-232 เป็นมาตรฐานอุตสาหกรรมที่ออกแบบมาเพื่อใช้ในการส่งข้อมูลอนุกรมแบบอะซิงโครนัส 2 ทิศทางโดยมาตรฐานแบบ RS-232 ในอดีตนั้นถูกออกแบบมาเพื่อการส่งผ่านข้อมูลจากคอมพิวเตอร์ไปยังโมเด็มเพียงอย่างเดียว เพื่อที่จะนำข้อมูลจากโมเด็มนี้สื่อสารผ่านสายโทรศัพท์ไปยังคอมพิวเตอร์อีกชุดซึ่งอยู่ห่างไกลกัน โดยคณะกรรมการที่เรียกว่า สมาคมอุตสาหกรรมอิเล็กทรอนิกส์ (Electronic Industries Association : EIA) ได้วางมาตรฐานที่มีชื่อเรียกกันว่า EIA RS-232 มาตรฐานนี้ในช่วงแรกจะใช้คอนเน็กเตอร์ปีนแบบ DB-25 โดยกำหนดความยาวสูงสุดของสายสัญญาณไว้ที่ 50 ฟุตมีระดับสัญญาณตั้งแต่ -3V ถึง -12V แสดงว่ามีข้อมูล (Mark) และ +3 ถึง +12V แสดงว่าเป็นช่องว่าง

เพื่อที่จะทำให้อุปกรณ์จากผู้ผลิตต่างกันทำงานร่วมกันได้ มาตรฐานหลายชนิดจึงได้รับการออกแบบขึ้น มาตรฐานที่ใช้กันอย่างกว้างขวางมากที่สุดคือ RS-232 ซึ่งโดยปกติไมโครคอมพิวเตอร์ จะมีพอร์ตที่เป็นแบบอนุกรมอยู่ในตัวแล้ว

มาตรฐาน RS 232 ได้กำหนดรูปแบบของอุปกรณ์เชื่อมต่อข้อมูล (Data Terminal Equipment : DTE) (Data Circuit Terminating : DCE) ไว้ว่าอุปกรณ์ DTE จะต้องเป็นอุปกรณ์ที่มีการประมวลผลในตัว เช่น ไมโครคอนโทรลเลอร์หรือ ไมโครคอมพิวเตอร์ ซึ่งมีความสามารถใน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การสร้างบิตข้อมูลแบบอนุกรมได้ ส่วนอุปกรณ์ DCE จะทำหน้าที่เพียงตัวรับข้อมูลที่ส่งมาจาก DTE เท่านั้น โดยการรับส่งข้อมูลระหว่างอุปกรณ์ทั้งสอง จะกระทำผ่าน มาตรฐานแบบ RS-232

ตามจุดประสงค์ของมาตรฐาน RS-232 นั้นเพื่อจะสามารถเชื่อมต่อกันระหว่างอุปกรณ์รับส่งปลายทาง (DATA TERMINAL EQUIPMENT : DTE) เช่นพอร์ตของคอมพิวเตอร์หลักหรืออุปกรณ์ปลายทางกับอุปกรณ์ RS-232 เป็นข้อกำหนดของการอินเตอร์เฟสมาตรฐาน และสามารถใช้เพื่อจุดประสงค์อื่นๆต่างๆกันไป เช่นการสื่อสารแบบซิงโคนัส (Synchronous Communication) และรูแบบการสื่อสารที่ต้องการสัญญาณนาฬิกา และสัญญาณกำหนดจังหวะเพิ่มเติมขึ้นมาในความเป็นจริงแล้วเราสามารถทำให้มีการสนทนาระหว่าง DTE และ DCE โดยการใช้สายสัญญาณเพียง 3 เส้นเท่านั้น คือ ใช้สาย TD สาย RD และสายกราวด์เท่านั้น

ข้อแตกต่างของอุปกรณ์ DTE และ DCE อย่างหนึ่งที่เห็นได้ชัดคือ คอนเน็กเตอร์ของ DTE จะเป็นตัวผู้ ส่วนคอนเน็กเตอร์ของ DCE จะเป็นตัวเมีย ซึ่งพอร์ตอนุกรมของคอมพิวเตอร์ที่ใช้กันอยู่ทั่วไปจะเป็นแบบ DTE ส่วนคอนเน็กเตอร์ที่อยู่ที่โมเด็มจะเป็นแบบ DCE

สำหรับการใช้งานบนคอมพิวเตอร์ พอร์ตอนุกรมแบบ RS-232 มักถูกใช้เชื่อมต่อกับโมเด็มหรือเมาส์โดยสามารถรับส่งข้อมูลได้ที่ความยาวของสายสัญญาณสูงสุดถึง 20 เมตร

หมายเลขสัญญาณ	ชื่อของสายเลขสัญญาณ
1	Data Carrier
2	Received Data
3	Transmitted Data
4	Data Terminal Ready
5	Signal Common
6	Data Set Ready
7	Request To Send
8	Clear To Send
9	Ring Indicator

ตารางที่2.1 แสดงรายละเอียดการต่อคอนเน็กเตอร์แบบ DB9 มาตรฐาน RS-232

- ขา 1 (Data Carrier Detect Circuit CF, DCD) ขานี่รู้จักกันในนามจอง Receive Line Signal Detect (RLSD) หรือขา Carrier Detect(CD) สัญญาณนี้จะเกิด Active เมื่อเกิดสัญญาณนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

พาหะที่เหมาะสมระหว่างอุปกรณ์ DCE ที่สถานีกับที่อยู่ระยะไกล เมื่อสัญญาณนี้อยู่ในสภาวะ OFF สัญญาณที่ขา RD ควรทำให้อยู่ในสภาวะ “MARK”

- ขา 2 (Receive Data Circuit BB,RD) สัญญาณที่ขานี้จุถูกสร้างจากอุปกรณ์ DCE กระแสบิตอนุกรมนี้จะกำเนิดที่อุปกรณ์ DTE ปลายทางและเป็นผลผลิตของวงจรรับข้อมูล DCE สัญญาณนี้มักเป็นข้อมูลที่ถูกสร้างขึ้นโดยอุปกรณ์ DCE

ขา 3 (Transmitted Data circuit BA,TD) เป็นขาสัญญาณข้อมูลที่ออกจากอุปกรณ์ DTE หระแสบิตอนุกรมจากขานี้ คือข้อมูลที่ถูกถ่ายทอดไปโดยโมเด็ม หรือถูกถอดรหัสโดยอุปกรณ์ DCE ที่มี

- ขา 4 (Rata Terminal Ready Circuit CD,DTR) สัญญาณ DTR ถูกใช้ในการควบคุมการสวิตช์อุปกรณ์ DCR เข้ากับตัวกลางในการสื่อสารสัญญาณ DTE ON บ่งชี้ว่าอุปกรณ์ DCE ที่กำลังเชื่อมต่อกันอยู่ก็บังต่อร่วมกัน และถ้ายังไม่มีการเชื่อมต่อกันก็สามารถทำให้มีการเชื่อมต่อกันครั้งใหม่ได้ ปกติแล้วสัญญาณ DTR จะอยู่ในสภาวะ OFF เพื่อการกระตุ้นให้อยู่ในสภาวะ “ON HOOK” (วางสาย) อุปกรณ์ DCE โดยปกติแล้วจะตอบสนองต่อการกระตุ้นจากสัญญาณ DTR โดยการทำให้อุปกรณ์ DSR แอกทีฟ

ขา 5 (Signal Common Circuit AB) สายนี้จะให้สัญญาณอ้างอิงของกราวด์ร่วมกันสำหรับวงจรการแลกเปลี่ยนกันทั้งหมด ยกเว้นวงจร AA หรือ Protective Ground ข้อกำหนด RS-232 จะอนุญาตให้วงจรนี้ถูกตัดต่อเพิ่มเติมกับ Protective Ground ภายในอุปกรณ์ DCE ได้ถ้าจำเป็น

- ขา 6 (Data Set Ready Circuit CC,DSR) สัญญาณ DSR จะบอกต่ออุปกรณ์ DTE ว่าอุปกรณ์ DCE ได้ต่อกับตัวกลางการสื่อสารที่ถูกต้องแล้ว และในบางกรณีจะบ่งชี้ว่าสายโทรศัพท์อยู่ในสภาวะ “OFF HOOK” สภาวะ “OFF HOOK” จะเป็นตัวบ่งชี้ว่าอุปกรณ์ DCE กำลังอยู่ในโหมด dialing หรือกำลังติดต่อกับอุปกรณ์ DCE อีกตัวหนึ่งอยู่ เมื่อสัญญาณ DSR นี้อยู่ในสภาวะ “OFF” อุปกรณ์ DTE ก็ควรจะถูกกำหนดให้ไม่สัญญาณอื่นทั้งจากอุปกรณ์ DCE ถ้าสัญญาณนี้ถูกกำหนดให้อยู่ในสภาวะ “OFF” ก่อนอุปกรณ์ DTR แล้วอุปกรณ์ DCE ก็จะสรุปว่าการสื่อสารนี้สิ้นสุดลง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สำนักหอสมุดกลาง พระจอมเกล้าลาดกระบัง

- ขา 7 (Request to Send Circuit CA, RST) สัญญาณนี้จะเตรียมพร้อมอุปกรณ์ DCE สำหรับการทำการส่งข้อมูลเมื่อสัญญาณ RST นี้อยู่ในสภาวะ "ON" จะทำให้อุปกรณ์ DCE อยู่ในโหมดส่งข้อมูล (Transmit mode) ในขณะที่สัญญาณนี้อยู่ในสภาวะ "OFF" ทำให้อุปกรณ์ DCE อยู่ในโหมดรับข้อมูล (Receive mode) อุปกรณ์ DCE ควรจะตอบสนองต่อสัญญาณ RST ON โดยการทให้สัญญาณ Clear to Send (CTS) อยู่ในสภาวะ "ON" ด้วยเมื่อ RST อยู่ในสภาวะ "OFF" เสียก่อน สัญญาณนี้ไม่ควรจะ "ON" ขึ้นอีก จนกว่าสัญญาณ CTS อยู่ในสภาวะ "OFF" เสียก่อน สัญญาณนี้จะถูกใช้ร่วมกับสัญญาณ DTR DSR และ DCD ขาสัญญาณ RTS จะถูกใช้อย่างมากในการควบคุมการไหลของข้อมูล

- ขา 8 (Clear to Send Circuit CB, CTS) สัญญาณนี้จะตอบรับกลับไปยังอุปกรณ์ DTE เมื่อได้รับสัญญาณ RST และข้อมูลสามารถถูกส่งออกไปได้ ข้อมูลจะถูกส่งออกไปตามตัวกลางที่ใช้สื่อสารได้ ก็ต่อเมื่อสัญญาณ CTS นี้ อยู่ในสภาวะ "ON" เท่านั้น สัญญาณนี้จะใช้ร่วมกับสัญญาณ DTR DSR และ DCD ขาสัญญาณนี้จะใช้ร่วมกับขา RTS สำหรับควบคุมการไหลข้อมูล

- ขา 9 (Ring Indicator Circuit CE, RI) สภาวะ "ON" ของขานี้จะบ่งชี้ว่าได้รับสัญญาณโทรศัพท์ตัวกลางในการสื่อสาร โดยปกติแล้วจะขึ้นอยู่กับโปรแกรมควบคุม ในการที่จะทำให้เกิดสัญญาณนี้ขึ้นหรือไม่

ขั้นตอนการติดต่อระหว่างอุปกรณ์ DTE และ DCE

1. เมื่อจ่ายกำลังงานให้กับ DTE และอุปกรณ์ก็จะส่งสัญญาณ DTR ออกมา
2. อุปกรณ์ DCE ถูกเปิดขึ้นและรับรู้สัญญาณ DTR ที่ส่งมาจากอุปกรณ์ DTE
3. อุปกรณ์ DCE ส่งสัญญาณ DSR ออกมา และ โมเด็มก็กระทำหระบวนการ OFF HOOK
4. ถ้าสายสัญญาณอยู่ในสภาพดีและอุปกรณ์อีกข้างหนึ่งก็พร้อมจะรับข้อมูลแล้ว โดยจะตรวจจับสัญญาณพาหะแล้วอุปกรณ์ DCE จะส่งสัญญาณ DCD ออกมา
5. อุปกรณ์ DCE จะตอบสนองด้วยการส่งสัญญาณ DTS ออกมา
6. การติดต่อสื่อสารก็เริ่มขึ้น โปรแกรมควบคุมจะทำการส่งหรือรับข้อมูล

73171

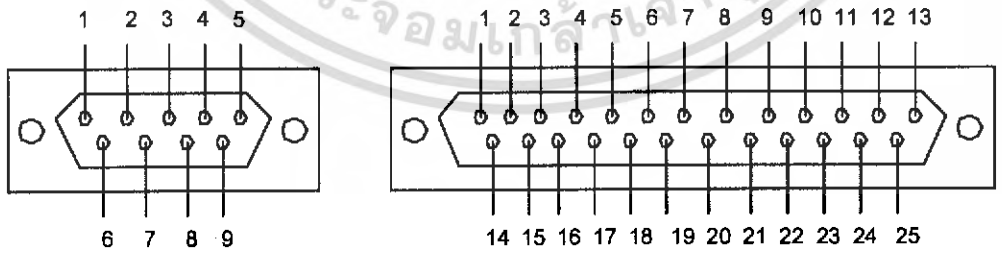
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ส่วนลำดับขั้นตอนในการตอบรับก็เป็นดังนี้

1. อุปกรณ์ DTE จะส่งสัญญาณ DTR ออกมา
2. อุปกรณ์ DCE จะอยู่ในโหมดจอบรับอัตโนมัติ (auto answer mode) โดยมีสัญญาณ DSR ออกมา
3. สถานีปลายทางส่งสัญญาณเรียกอุปกรณ์ DCE และอุปกรณ์ DCE ส่งสัญญาณ RI ออกมา
4. อุปกรณ์ DTE รับรู้ถึง RI ที่ส่งมาจากเครื่องปลายทาง และอุปกรณ์ DCE ก็เข้าสู่สภาวะ OFF HOOK
5. อุปกรณ์ DCE ทำการแลกเปลี่ยนข้อมูลกับอุปกรณ์ DCE ที่อีกปลายทางหนึ่งสัญญาณ DCD ออกมา
6. อุปกรณ์ DTE จะส่งสัญญาณ RTS ออกมาหรือจะรอข้อมูลก็ได้ ขึ้นอยู่กับโปรแกรมควบคุม
7. อุปกรณ์ DCE จะตอบสนองด้วยการส่งสัญญาณ DTS ออกมาการติดต่อสื่อสารก็จะเริ่มขึ้น

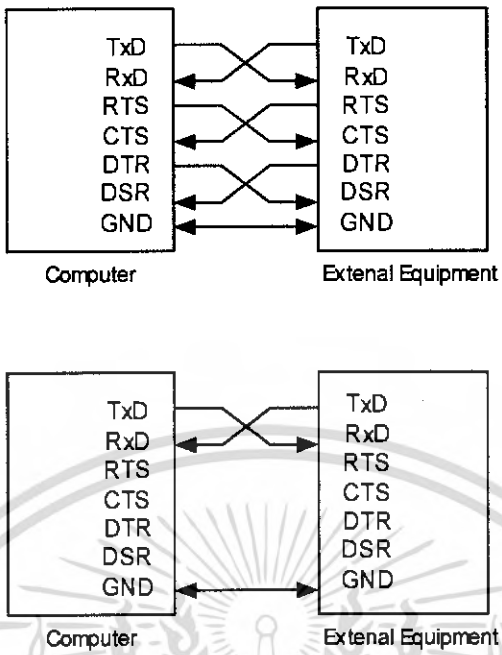
คอนเน็กเตอร์สำหรับพอร์ตแบบ RS-232 และการเชื่อมต่อ

มาตรฐานการเชื่อมต่อแบบ RS-232 จะใช้คอนเน็กเตอร์แบบ DB-25 ตัวผู้หรือ DB-9 เนื่องจากขาอื่นๆ ที่ใช้งานในอดีต ปัจจุบันมีการใช้งานไม่มากนัก จึงถูกยกเลิกไป โดยแสดงรูปร่างตำแหน่งขาในรูป



รูปที่ 2.8 แสดงการจัดขาของคอนเน็กเตอร์พอร์ตอนุกรมมาตรฐานแบบ RS-232 ทั้งแบบ DB-9 และ DB-25

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.9 การต่ออุปกรณ์ภายนอกกับพอร์ตอนุกรมของคอมพิวเตอร์ในลักษณะต่างๆ

ลักษณะสัญญาณอินพุทแลเอาท์พุทของพอร์ต RS-232

สัญญาณเอาท์พุทที่ใช้ควบคุม (RTS และ DTR) และสัญญาณสถานะอินพุท (CTS,DSR และ DCD) ของพอร์ตอนุกรม RS-232 จะถูกกลับสถานะภายในตัว UART ส่วนสัญญาณข้อมูลทั้งภาคส่งและรับจะไม่ถูกกลับสถานะ UART จะให้ระดับสัญญาณเอาท์พุทออกมาเป็นแบบทีทีแอลเท่านั้น ดังนั้นเมื่อสัญญาณถูกส่งออกมาจาก UART จึงต้องส่งเข้าสู่วงจรขับเพื่อปรับระดับแรงดันให้ได้ระดับสัญญาณเป็นไปตามมาตรฐาน RS-232 ก่อนส่งออกจากคอมพิวเตอร์ สำหรับอุปกรณ์ต่อเชื่อมปลายทางก็จะต้องมีวงจรขับในลักษณะนี้เช่นเดียวกัน เพื่อให้ได้ระดับสัญญาณเดียวกัน แต่วงจรที่ใช้ขับภายในคอมพิวเตอร์และอุปกรณ์เชื่อมต่อปลายทางนั้นจะถูกกลับสถานะดังรูป

การสื่อสารข้อมูลผ่านพอร์ตอนุกรม

การสื่อสารข้อมูลผ่านพอร์ตอนุกรมในไมโครคอนโทรลเลอร์นั้นทำได้ 2 วิธี คือ

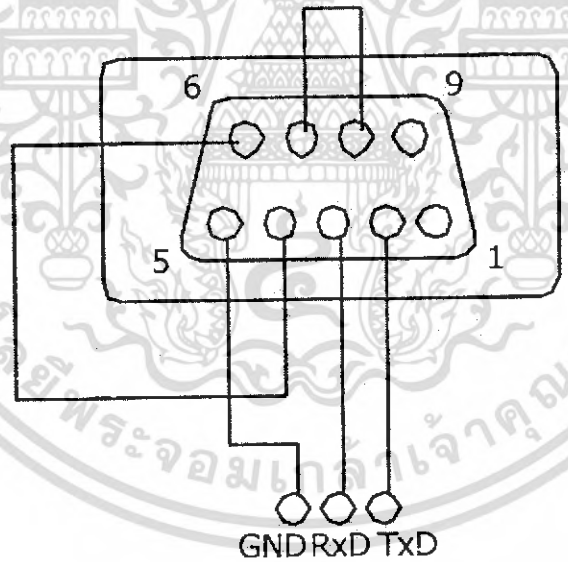
1. ใช้อินเตอร์รัปต์ เป็นวิธีที่ให้ผลการทำงานเร็วที่สุด แต่มีความยุ่งยากในการทำงานมาก เนื่องจากตำแหน่งของอินเตอร์รัปต์ทั้งการรับและการส่งข้อมูลนั้นอยู่ที่ตำแหน่งเดียวกัน จะต้องพิจารณาที่แฟลก TI หรือ RI ก่อนว่าเกิดการอินเตอร์รัปต์จากสาเหตุใดแลต้องพิจารณาการใช้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รีจิสเตอร์ในช่วงเวลานั้นๆว่ามีโอกาสซ้อนทับกันหรือไม่ ทำโปรแกรมการทำงานของในส่วนนี้มีความซับซ้อนมาก

2. การวนโปรแกรมตรวจสอบแฟลก เป็นวิธีที่มีความซับซ้อนน้อยกว่า โดยเขียนโปรแกรมให้วนตรวจสอบแฟลกอยู่ตลอดเวลา จนกว่าจะเกิดการเปลี่ยนแปลง ยกตัวอย่างเมื่อต้องการตรวจสอบการส่งข้อมูลให้ทำการวนตรวจสอบแฟลก TI ว่าถูกเซตหรือไม่ เมื่อถูกเซต แสดงว่ามีการส่งข้อมูลเกิดขึ้นเรียบร้อยแล้ว จากนั้นให้ทำการเคลียร์ TI แล้วทำการส่งข้อมูลตัวต่อไป หรือทำงานในคำสั่งต่อไปได้ ในการที่ต้องการตรวจสอบการรับข้อมูลให้ทำการตรวจสอบ RI ว่าถูกเซตหรือไม่ เมื่อตรวจสอบได้ว่าถูกเซตแสดงว่าเกิดการรับข้อมูลขึ้นให้ทำการเคลียร์ RI แล้วนำค่าจาก SBUF มาใช้ได้เลย ข้อเสียของวิธีนี้ก็คือ การทำงานเป็นแบบเรียงลำดับทำให้ขั้นตอนการทำงานล่าช้ากว่าการทำงานแบบอินเตอร์รัปต์

หัวใจสำคัญของการสื่อสารข้อมูลผ่านพอร์ตอนุกรมคือ การกำหนดอัตราบอดและรูปแบบของข้อมูลจำนวนบิตเริ่มต้น,บิตข้อมูล,บิตหยุดหรือว่ามีการตรวจสอบบิตพาริตี้หรือไม่ ถ้าหากข้อกำหนดเหล่านี้ในตัวส่งและตัวรับไม่ตรงกันจะทำให้การถ่ายทอดข้อมูลเกิดความผิดพลาดได้อย่างง่าย และคอนเน็คเตอร์ที่ใช้ต้องเป็นแบบ DB-9 ซึ่งมีลักษณะดังรูป



รูปที่ 2.10 DB-9 Connector

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 3

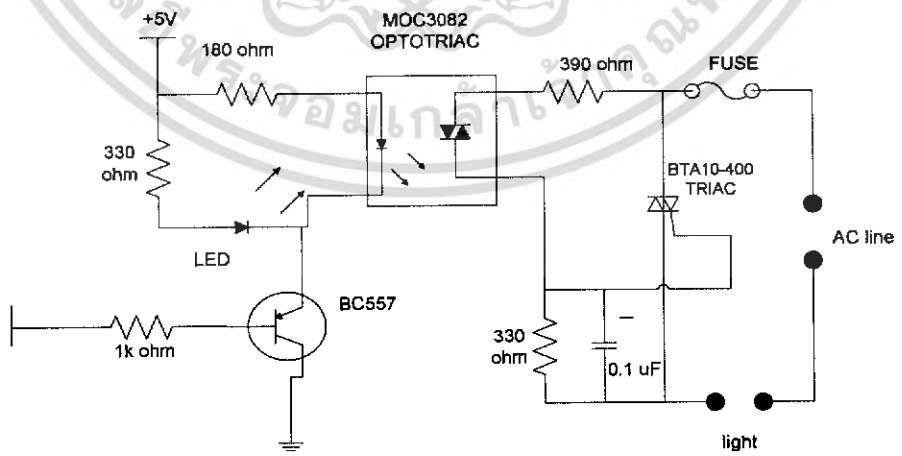
วงจรและการออกแบบ

ในโครงงานนี้จะจำลองระบบควบคุมอุปกรณ์ไฟฟ้าผ่านเว็บ ภายในบ้าน โดยใช้ ไอซี Rabbit เป็นตัวควบคุม ซึ่งจะถูกแบ่งออกเป็น 2 ส่วนดังนี้

1. วงจรเชื่อมต่อเพื่อควบคุมอุปกรณ์ไฟฟ้า (หลอดไฟ)
2. วงจรตัวส่งและตัวรับสัญญาณอินฟราเรด (วงจร sensor)
3. วงจรตรวจจับอุณหภูมิ

3.1 วงจรเชื่อมต่อเพื่อควบคุมอุปกรณ์ไฟฟ้า (หลอดไฟ)

ในการควบคุมอุปกรณ์ไฟฟ้าซึ่งใช้แรงดันไฟฟ้ากระแสสลับ 220 โวลท์ ไมโครคอนโทรลเลอร์ ไม่สามารถควบคุมได้โดยตรง จึงต้องใช้อุปกรณ์ โฟโต้ไตรแอค (Triac) เพื่อทำการควบคุมอุปกรณ์ไฟฟ้า โดยจะส่งสัญญาณควบคุมมาทางพอร์ตของไมโครคอนโทรลเลอร์ ซึ่งในการควบคุมอุปกรณ์ไฟฟ้าได้กำหนดไว้จำนวน 2 ชุดด้วยกัน ส่วนควบคุมอุปกรณ์ไฟฟ้า หรือที่เรียกว่า โซลิดสเตทรีเลย์ (Solid State Relay) ซึ่งมีโครงสร้างประกอบด้วยส่วนต่างๆ คือ ส่วนควบคุม ตัวเชื่อมโยงทางแสง ส่วนป้องกันทรานเซี้ยน และ ไตรแอค ดังรูปที่ 3.1



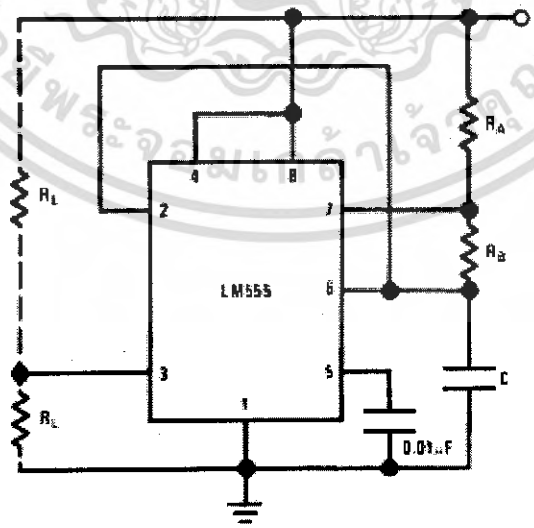
รูปที่ 3.1 แสดงวงจรควบคุมไฟ 220 โวลต์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.2 วงจรภาคส่งสัญญาณอินฟราเรด

วงจรอินฟราเรดนี้ใช้เป็นอุปกรณ์ตรวจสอบสถานะบริเวณประตูหรือหน้าต่าง โดยจะติดตั้งตัวส่งสัญญาณอินฟราเรดและตัวรับสัญญาณระหว่างประตูหรือหน้าต่าง แล้วติดตั้งตัวรับสัญญาณอินฟราเรดอีกด้านของประตูหรือหน้าต่าง เพื่อคอยตรวจสอบสัญญาณอินฟราเรดที่ส่งมาจากตัวส่ง ซึ่งในสภาวะปกติไม่มีวัตถุอะไรมาบังระหว่างตัวส่งกับตัวรับ แสดงว่าไม่มีผู้ผ่านเข้าออก แต่ถ้าตัวรับไม่สามารถรับสัญญาณที่ส่งมาจากตัวส่งได้ แสดงว่ามีผู้ผ่านเข้าออก เราสามารถใช้ไมโครคอนโทรเลอร์ตรวจสอบสถานะเอาท์พุทของตัวรับ

ในส่วนการออกแบบวงจรนี้จะใช้ IC555 สร้างสัญญาณพัลส์โดยการเก็บประจุเข้าไปในตัวเก็บประจุ Ca ผ่าน Ra และ Rb เมื่อแรงดันคร่อมมีค่า 2/3 ของ Vcc จะทำให้วงจรเปรียบเทียบแรงดันชุดบนทำงานซึ่งจะไปกระตุ้นให้วงจรควบคุมฟลิปฟลอปทำงานอีกต่อเป็นผลให้ Ca คายประจุผ่าน Rb และขา 7 ลงกราวด์ ช่วงนี้เอาท์พุทขา 3 จะมีแรงดันต่ำและเมื่อแรงดันที่ Ca ลดลงมาถึง 1/3 ของ Vcc ก็จะไปกระตุ้นให้วงจรเปรียบเทียบชุดล่างทำงานกระตุ้นวงจรฟลิปฟลอป ทำให้แรงดันเอาท์พุทสูงขึ้นและ Ca ไม่สามารถคายประจุได้อีกเพราะทรานซิสเตอร์ภายในซึ่งทำหน้าที่เป็นสวิตช์จะอยู่ในสภาวะไม่ทำงาน ดังนั้น Ca จะเริ่มการเก็บประจุอีกแรงดันเท่ากับ 2/3 ของ Vcc ก็จะเริ่มทำงานแบบเดิมอีกครั้ง ซึ่งจะมีลักษณะดังรูปที่ 3.2



รูปที่ 3.2 แสดงวงจรที่ใช้คำนวณหาค่าความถี่ของ IC555

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

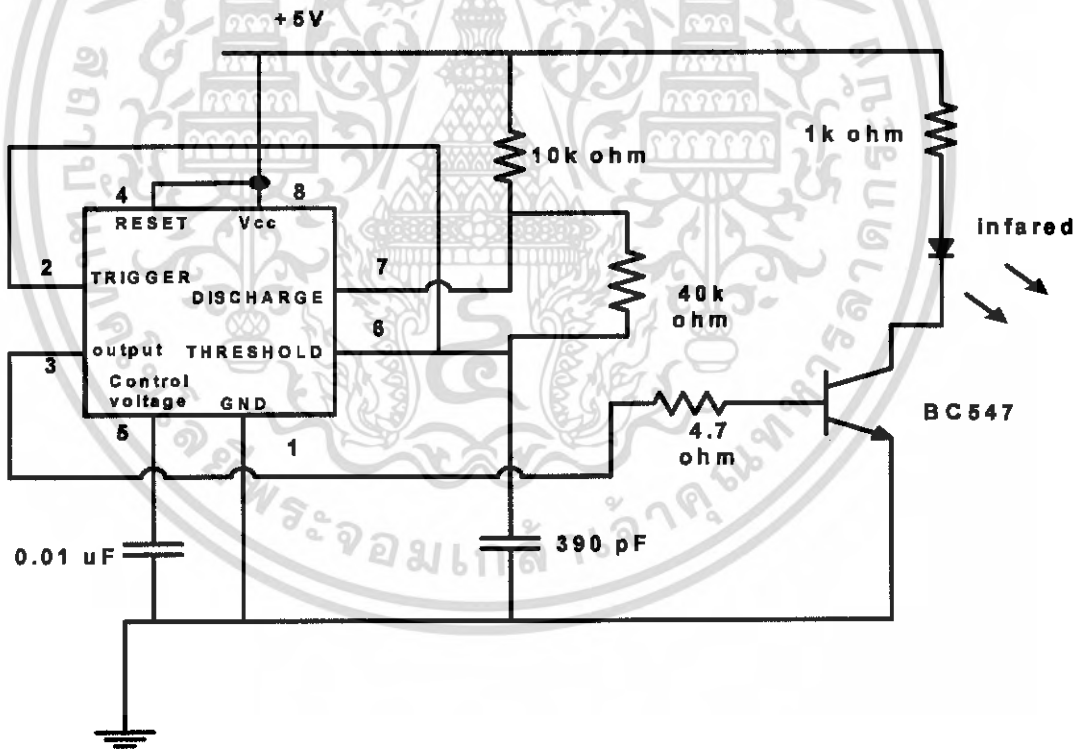
จากลักษณะการทำงานของวงจร จะเห็นว่าช่วงความกว้างของพัลส์ที่ได้ทางเอาท์พุทจะขึ้นอยู่กับการเก็บประจุ และการคายประจุของ Ca ซึ่งจาก Data sheet จะได้ว่า

ช่างการเก็บประจุ : $t1 = 0.693 * (Ra+Rb) * Ca$

ช่วงการคายประจุ : $t2 = 0.693 * Ra * Ca$

ความถี่ของสัญญาณเอาท์พุท : $f = 1/(t1+t2)$

วงจรอินฟราเรดทางด้านส่งนี้จะสร้างพัลส์โดยใช้ความถี่ที่สร้างขึ้นสามารถกำหนดได้จาก Ra, Rb และ Ca จากสูตรที่กล่าวมาแล้ว ในวงจรนี้เราจะสร้างสัญญาณพัลส์ความถี่ประมาณ 38 KHz เพื่อให้ตรงกับความถี่ของวงจรทางด้านตัวรับอินฟราเรด ซึ่งลักษณะของวงจรทางด้านตัวส่งจะเป็นดังรูปที่ 3.3

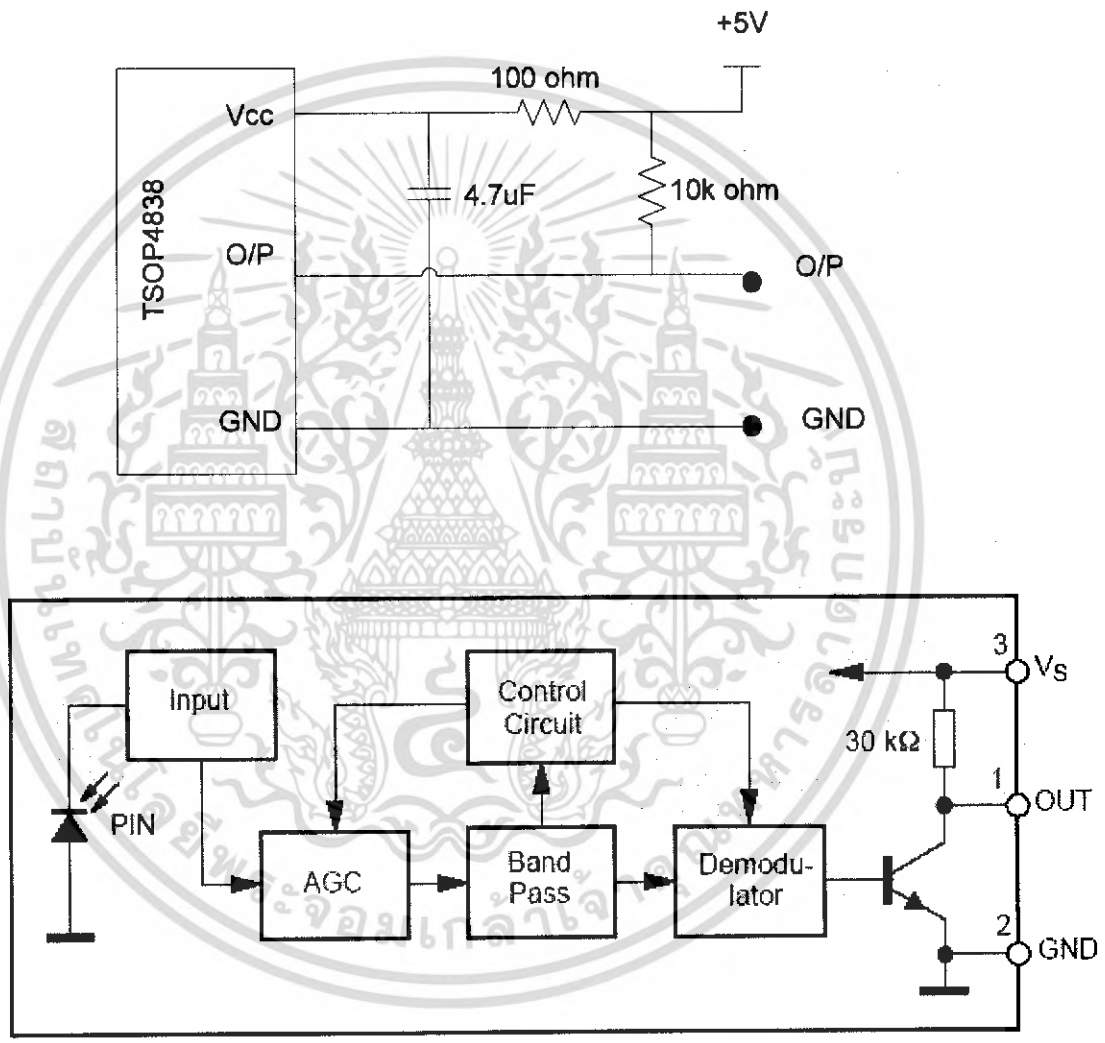


รูปที่ 3.3 แสดงวงจรอินฟราเรดทางด้านส่ง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.3 วงจรภาครับอินฟราเรด

ตัวรับอินฟราเรดจะใช้เป็น Photo Modules เบอร์ TSOP4838 ที่สามารถรับสัญญาณอินฟราเรดที่ความถี่ 38 KHz ซึ่งภายใน Photo Modules TSOP4838 นี้จะประกอบด้วยส่วนรับแสงอินฟราเรด เฟสล็อคลูปที่ความถี่ 38 KHz และภาคขยายสัญญาณเอาท์พุท

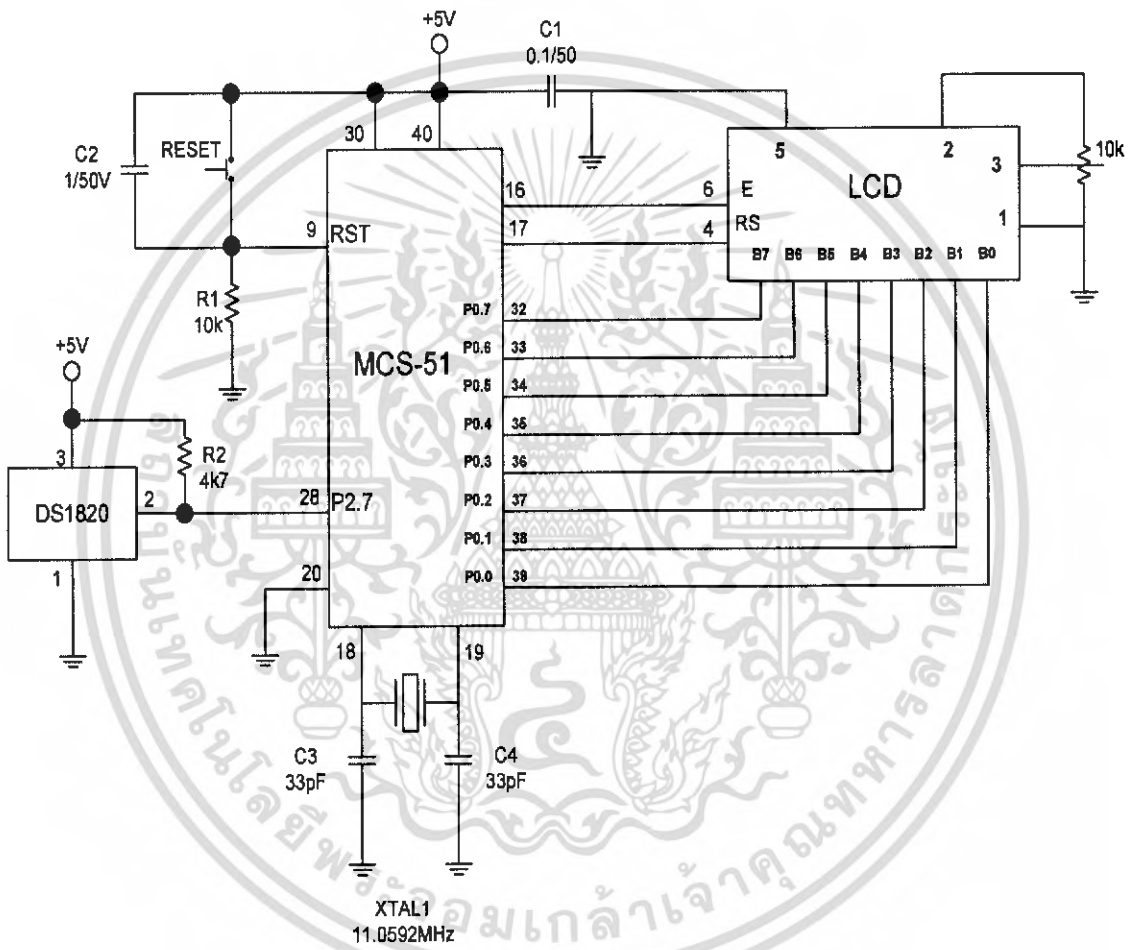


รูปที่ 3.4 วงจรภายใน TSOP4838

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.4 วงจรตรวจจับอุณหภูมิ

จะใช้เซนเซอร์ DS1820 เป็นตัวตรวจจับ เชื่อมต่อกับ MCS-51 และแสดงผลออกทั้งทาง LCD และ หน้าเว็บซึ่งส่งออกทางserial port ดังแสดงดังรูป

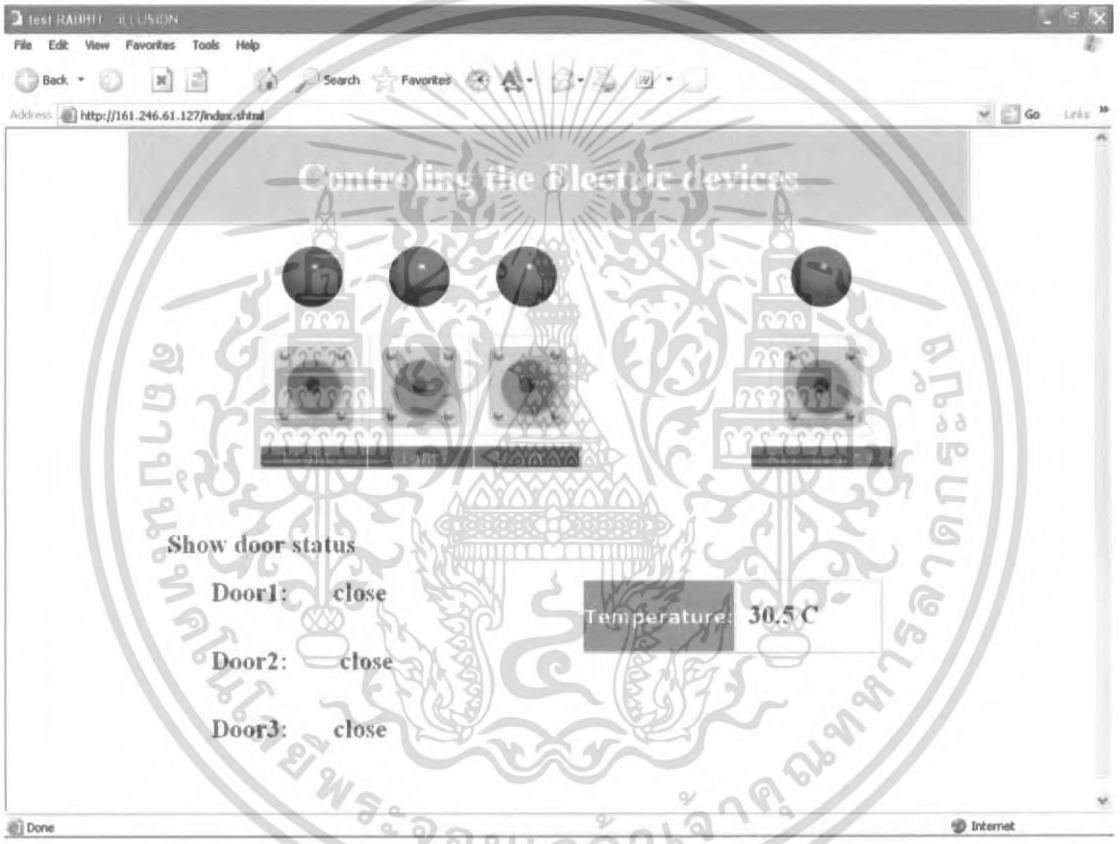


รูปที่ 3.5 แสดงการเชื่อมต่อ DS 1820 และ LCD

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 4
หลักการและการทำงาน

ในโครงงานนี้จะมีการทำห้องจำลองขนาดเล็กซึ่งมีประตูสามารถ เปิด-ปิดได้ ในส่วนของฮาร์ดแวร์วงจร จะแบ่งเป็น 3 ส่วน



รูปที่4.1 แสดงหน้าเว็บเพจของโครงงาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 4.2 แสดงฮาร์ดแวร์ของโครงงาน

4.1 วงจรควบคุมอุปกรณ์ไฟฟ้า

เราจะให้มีการเปิด-ปิดหลอดไฟโดยผ่านการควบคุมจากหน้าเว็บเพจที่เราสร้างขึ้นซึ่งจะใช้เป็นบอร์ดควบคุมอุปกรณ์ไฟฟ้า 1 บอร์ด(มีหลอดไฟขนาด 3W จำนวน 1 หลอด) และใช้ LED จำนวน 2 ตัว เป็นการจำลองสถานการณ์เสมือนเป็นหลอดไฟ อีก 2 หลอด
โดยมีหลักในการทำงานดังนี้

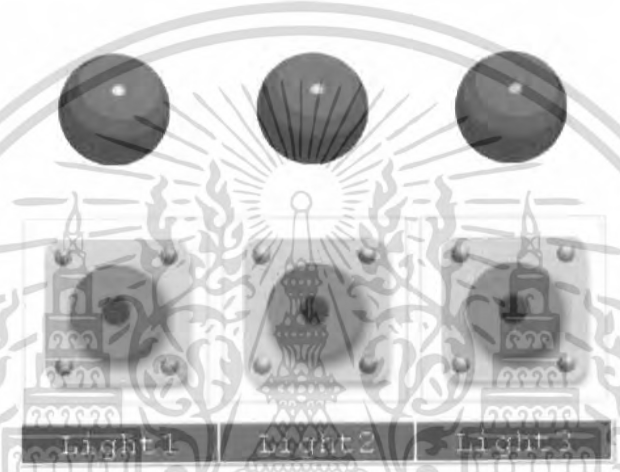
เมื่อเรากดสวิตช์ ทางหน้าเว็บเพื่อควบคุมการเปิด หลอดไฟ จะมีการส่งค่า high เป็นเอาท์พุทออกจากไมโครคอนโทรลเลอร์ เข้าเป็นอินพุทของวงจรควบคุมอุปกรณ์ไฟฟ้า (หลอดไฟ) หลอดไฟจะดับ (ปิด)

เมื่อเรากดสวิตช์ ทางหน้าเว็บเพื่อควบคุมการปิด หลอดไฟจะมีการส่งค่า low เป็นเอาท์พุทออกจากไมโครคอนโทรลเลอร์เข้าเป็นอินพุทของวงจรควบคุมอุปกรณ์ไฟฟ้า(หลอดไฟ) หลอดไฟจะสว่าง (เปิด)

และยังมีการแสดงสถานะของหลอดไฟ เป็นรูปภาพ LED แสดงผลการเปิด-ปิด บนหน้าเว็บด้วย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 4.3 แสดงภาพของหลอดไฟ และหลอดLED อยู่ในสถานะปิด ทางฮาร์ดแวร์



รูปที่ 4.4 แสดงรูปของปุ่มควบคุมหลอดไฟบนหน้าเว็บ

4.2 วงจรรับ-ส่งอินฟราเรด

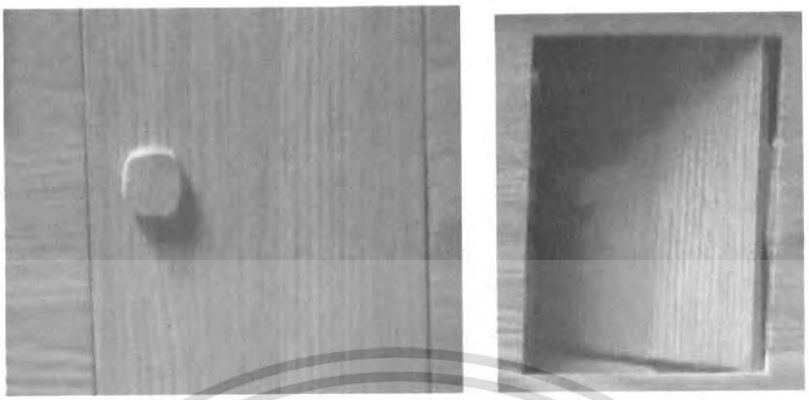
เราจะมีการนำวงจรรับ-ส่งจำนวน 1 ชุดไปติดไว้ที่ประตูจำลองที่เราได้สร้างขึ้น และมีการใช้สวิตช์จำนวน 2 ตัวจำลองสถานการณ์ ให้เสมือนเป็นตัวอินฟราเรดรับ-ส่งจำนวน2 ชุด ซึ่งสามารถส่งค่าเป็น low หรือ high ได้เมื่อมีคนเปิด-ปิดประตู

โดยมีหลักการในการทำงานดังนี้

เมื่อประตูเปิดจะมีการกั้นสัญญาณระหว่างตัวรับและตัวส่ง ตัวรับไม่สามารถรับสัญญาณได้ เอาท์พุทที่ออกมาจากตัวรับจะเป็น high เข้าไมโครคอนโทรลเลอร์และแสดงผลทางหน้าเว็บถึงสถานะว่ามีการเปิดประตู

เมื่อประตูปิดจะมีการรับ-ส่งสัญญาณระหว่างตัวส่งและตัวรับ ตัวรับสามารถรับสัญญาณจากตัวส่งได้ ทำให้เอาท์พุทที่ออกมาจากตัวรับเป็น low เข้าไมโครคอนโทรลเลอร์และแสดงผลทางหน้าเว็บถึงสถานะว่าประตูปิด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.5 แสดงประตูจำลองที่อยู่ในสถานะปิด-เปิด

รูปที่ 4.6 แสดงสวิทช์ที่เปรียบเสมือนประตูอีก 2 บาน

รูปที่ 4.7 แสดงหน้าเว็บในส่วนที่แสดงผลจากฮาร์ดแวร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.3 วงจรตรวจจับอุณหภูมิ

จะใช้เซนเซอร์ DS1820 เป็นตัวตรวจจับอุณหภูมิ เชื่อมต่อกับ MCS-51 แสดงผลทางLCD และ ทางหน้าเว็บโดยผ่านทาง serial port ซึ่งแสดงผลเป็นองศาเซลเซียส

รูปที่ 4.8 แสดงผลอุณหภูมิทางหน้าจอ LCD

รูปที่ 4.9 แสดงหน้าเว็บในส่วนของอุณหภูมิ

4.4 รูปแบบของบอร์ด RCM 3720

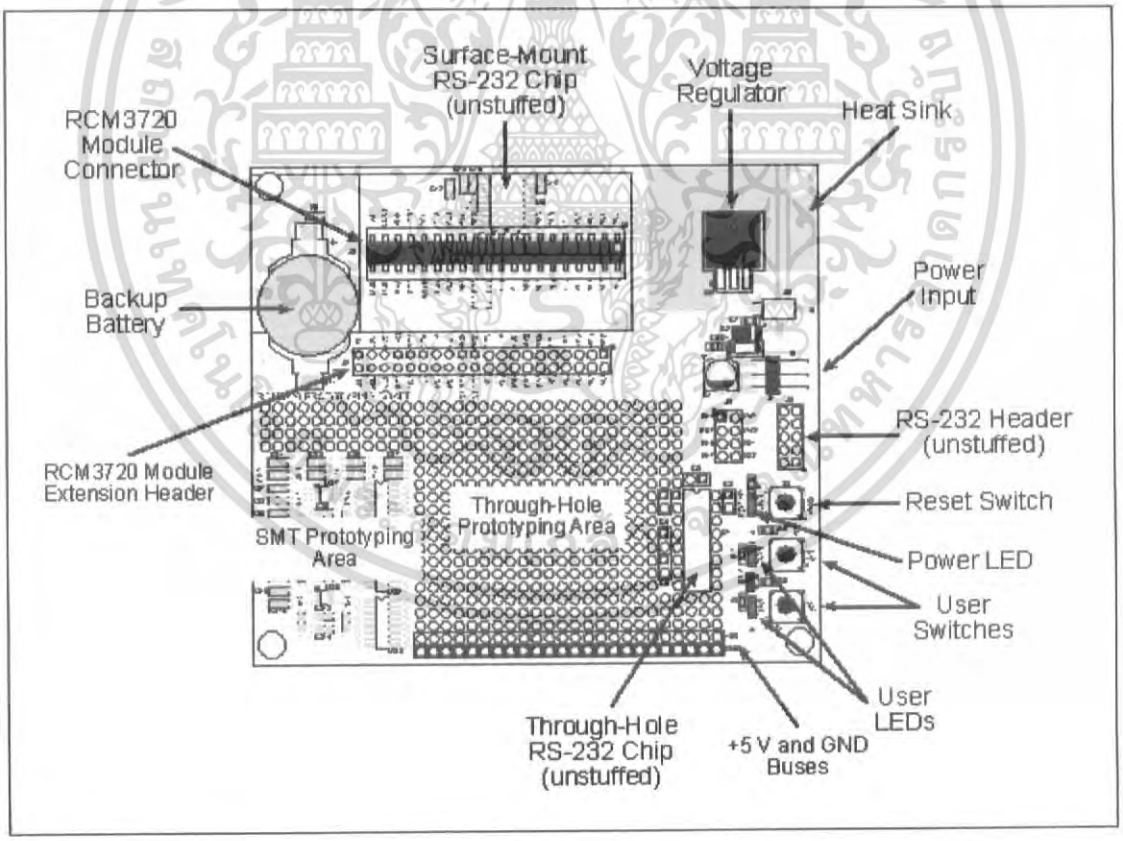
รูปแบบของบอร์ด RCM 3720 นั้นถูกรวมเข้ากับอุปกรณ์Ethernet เพื่อทำให้ง่ายต่อการต่อเข้ากับแหล่งจ่ายพลังงาน (Power Supply) และ เครื่อง PC ในการทำงาน ทั้งนี้บอร์ดยังมีอุปกรณ์พื้นฐานภายนอก I/O อยู่ด้วย อุปกรณ์นี้คือ หลอด LED และ Switch ในส่วนของ Prototyping area จะเป็นส่วนที่สามารถนำอุปกรณ์ฮาร์ดแวร์อื่นๆที่ต้องการมาต่อเพิ่มเพื่อการพัฒนาในส่วนของการทำงานที่เราต้องการได้

ในระดับพื้นฐานของการพัฒนาเพื่อการใช้งานของบอร์ด RCM 3720 นั้นสามารถใช้งานได้ทันทีโดยที่ไม่ต้องต่ออุปกรณ์ฮาร์ดแวร์อื่นๆเพิ่มเติม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ในการที่ต้องการจะพัฒนาการทำงานตามที่เราต้องการให้มีมากขึ้น หรือ การต่ออุปกรณ์ฮาร์ดแวร์อื่นๆเพิ่ม ก็สามารถนำมาต่อเพิ่มได้เลยโดยที่ไม่ต้องมีการปรับแต่งเพิ่มเติมในบอร์ดของ RCM 3720 บอร์ดก็สามารถทำงานได้

รูปที่ 4.10 แสดงรูปของบอร์ด RCM3720



รูปที่ 4.11 แสดงรูปแบบของ RCM 3720 และส่วนประกอบหลัก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.4.1 ส่วนประกอบหลักของ RCM 3720

- **Power Connection**

  ตัว Connector 3 ขา ที่ถูกจัดทำขึ้นมาเพื่อต่อเข้ากับ แหล่งจ่ายพลังงาน (Power Supply) ขาทั้งสามนั้นเหมือนกัน โดยขาที่อยู่ริมทั้งสองขาจะต่อเข้ากับกราวด์ และขากลางจะต่อกับDC input ที่ยังไม่ได้แปลงโดยที่ สายหม้อแปลง (AC) จะมีheader ต่ออยู่เพื่อสามารถที่จะนำไปต่อกับ connectorได้

  ผู้ใช้ควรที่จะแน่ใจว่าหม้อแปลงที่ใช้นั้นควรอยู่ที่ 7.5-15 VDC ที่ 200mA เพราะอาจจะทำให้วงจรกรองแรงดันเกิดความร้อนขึ้นได้

- **Linear Power Supply**

  โวลต์ DC ที่ยังไม่มีการแปลงค่าจะต่อเข้ากับแจ็กของ POWER IN ซึ่งจะเป็นตัวเข้าไปกรองแรงดันแปลงเป็น 5 โวลต์

- **Power LED**

  หลอด power LED จะสว่างเมื่อมีการต่อกับแหล่งจ่ายไฟบน Prototyping Board

- **Reset Switch**

  เป็นตัวติดต่อชั่วคราว ปกติแล้วสวิตช์เปิดจะถูกต่อเข้าโดยตรงกับขา RESET_IN ของ RCM3720 การกดสวิตช์จะเป็นการทำให้ hardware รีเซตระบบ

- **I/O Switches and LEDs**

  เป็นตัวติดต่อชั่วคราวสองตัว ปกติแล้วสวิตช์เปิดจะถูกต่อเข้ากับขา PF4 และ PB7 ของ โมดูลRCM3720 และอาจจะอ่านได้เป็นอินพุท โดยการใช้งานตามตัวอย่างที่ให้มา

  LED 2 ตัวถูกต่อเข้ากับขา PF6 และ PF7 ของโมดูล RCM3720 และอาจจะขับออกมาแสดงถึงเอาท์พุทโดยการใช้งานของตัวอย่าง (sample)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

● **Prototyping Area**

พื้นที่ของบอร์ดโดยทั่วไปจะมีรูของบอร์ดสำหรับการติดตั้งวงจรต่างๆ ซึ่งแหล่งจ่ายไฟ +5 V และ กราวด์จะเป็นแนวยาวตามขอบริมของบอร์ด มีพื้นที่สำหรับ อุปกรณ์ surface-mount (ในบอร์ดนี้จะมี SMT อยู่บนบอร์ดทั้งด้านบนและขอบริมของบอร์ด) แต่ละ SMT จะถูกติดต่อกับรูที่ถูกออกแบบให้ใช้กับ 30 AWG solid wire หรือ wire-wrap wire

● **Module Extension Headers**

เซตของขา (pin) ของ โมดูลRCM3720 จะถูกสร้างขึ้นอีก1ชุดที่ header J2 ผู้ใช้สามารถวายสายเข้าได้โดยตรงดับรูที่แยกๆกันอยู่บนบอร์ต สำหรับ 2 × 20 header 0.1 pitch สามารถที่จะต่อเข้าได้

● **RS-232**

ชิพ MAX232 RS-232 สามารถเลือกได้ว่าจะต่อที่ U1หรือU3 (ตัวเก็บประจุ 0.1uF 5ตัว จำเป็นที่จะต้องต่อเข้าไปด้วยเพื่อการทำงานของวงจร RS-232) ชิพ RS-232 มีการเปิดให้เห็นถึง Serial port C และ D ที่ header J3 บนบอร์ดของ RCM3720 ตัว 2 × 5 header 0.1 pitch สามารถนำไปดิดตั้งที่ J3 เพื่อให้ติตต่อกับสายเคเบิล ที่เป็นมาตรฐาน คอนเน็กเตอร์อนุกรม DB-9 3-wire serial ports จำนวน 2 ตัว และ 5-wire RS-232 serial port จะอยู่ที่ header J3

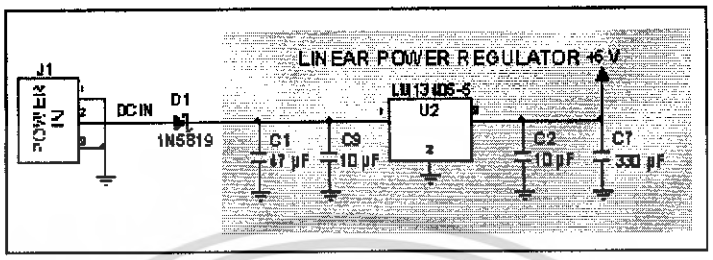
● **Backup Battery**

2032 lithium-ion แบตเตอร์รี่ 3.0 V, 220 mA·h เป็นแบตเตอร์รี่สำรองสำหรับ RCM3720 SRAM และ รีล ไทม์ คล็อก

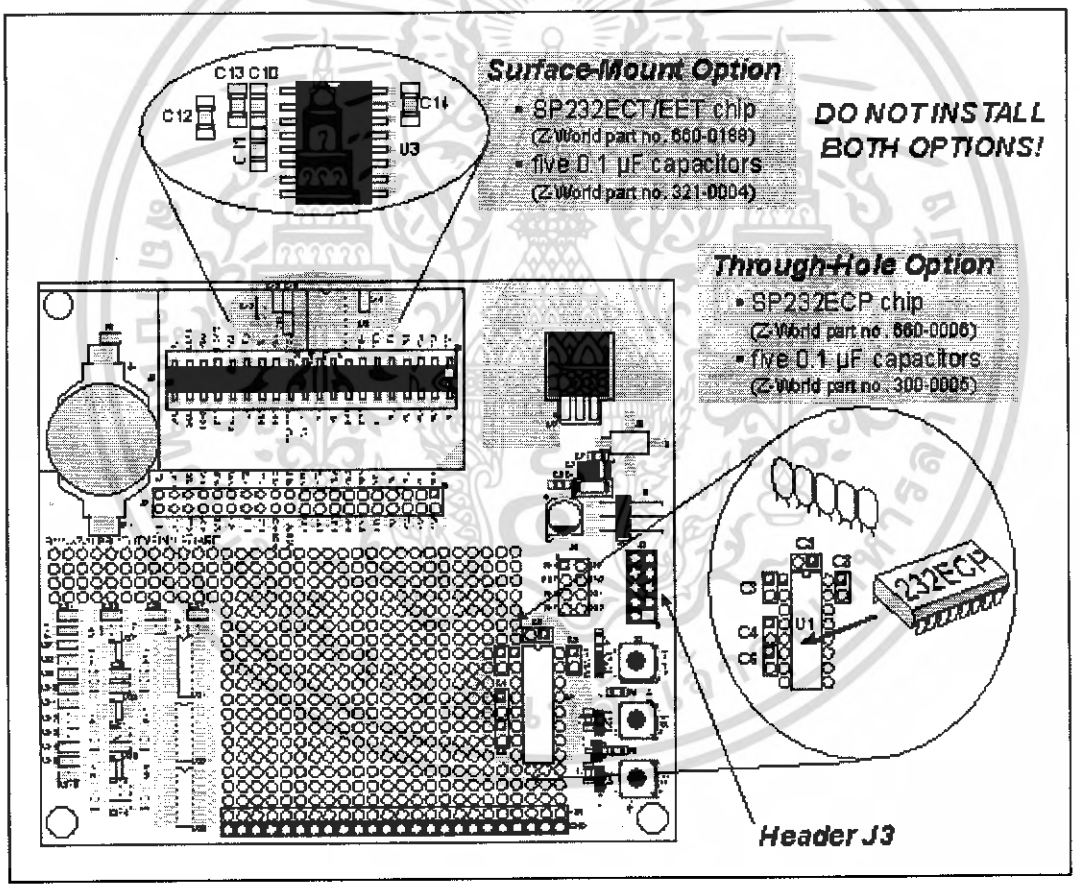
4.4.2 Power Supply

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บอร์ด RCM3720 จะต้องจ่ายไฟให้ 4.75V - 5.25V DC ในการทำงาน ซึ่งบนบอร์ด RCM3720จะมีตัวกรองแรงดัน +5 V ด้วย ในตัวบอร์ดจะมีการป้องการกระแสไหลย้อนกลับโดยการใช้ ช็อตกี้ไดโอด ที่ D1 ดังรูป

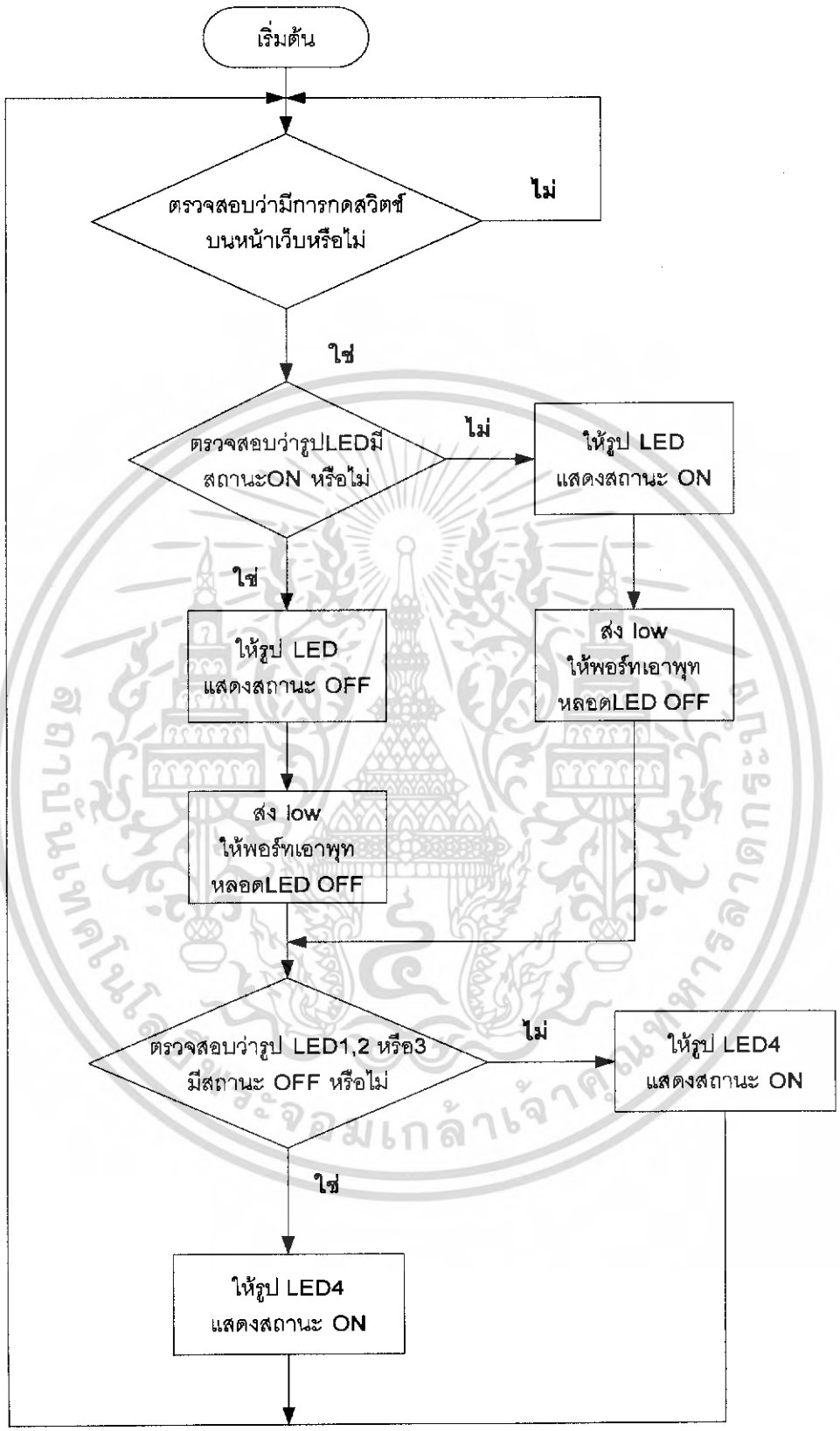


รูปที่ 4.12 แสดงถึงแหล่งจ่ายไฟ ของบอร์ด RCM3720



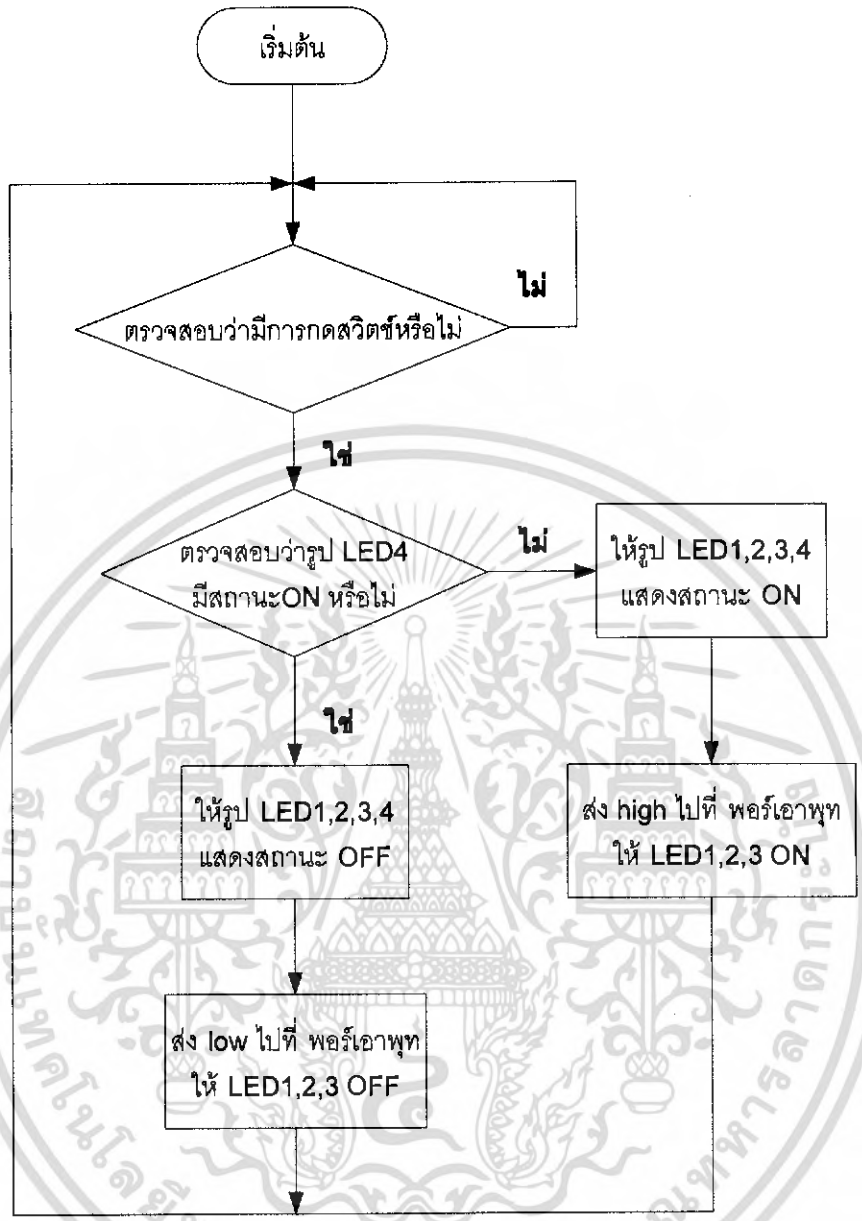
รูปที่ 4.13 แสดงรูปบริเวณที่ต่อกับ RS-23

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



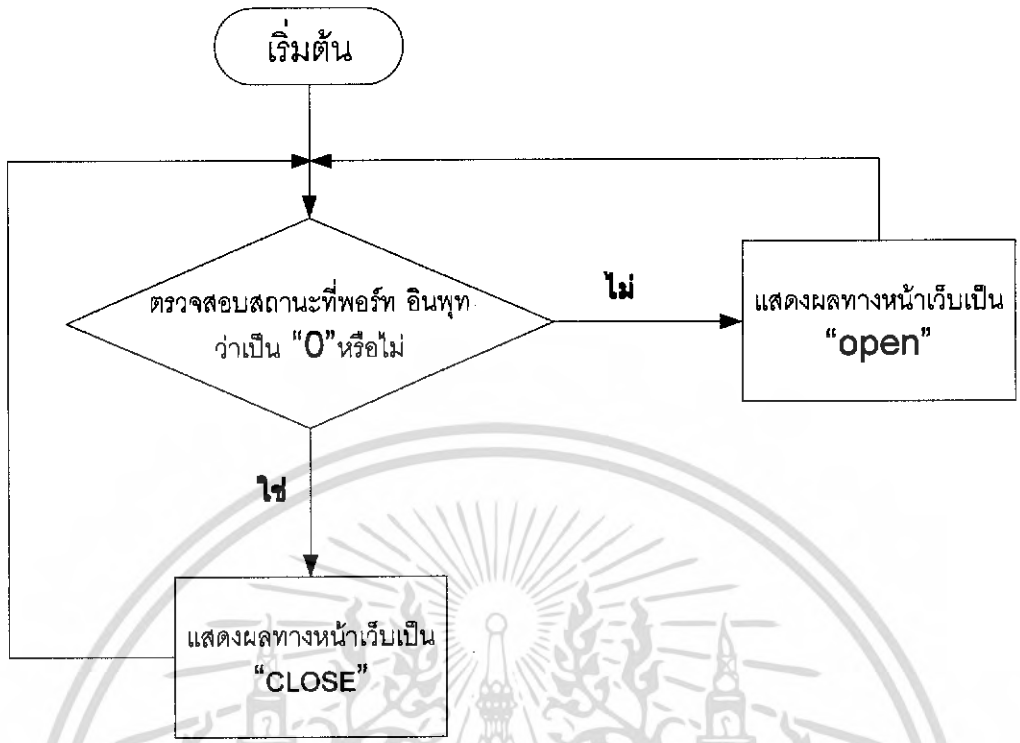
รูปที่ 4.14 แสดง FLOW CHART ของการกดสวิตช์จากหน้าเว็บเพื่อควบคุมหลอดไฟแต่ละหลอด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



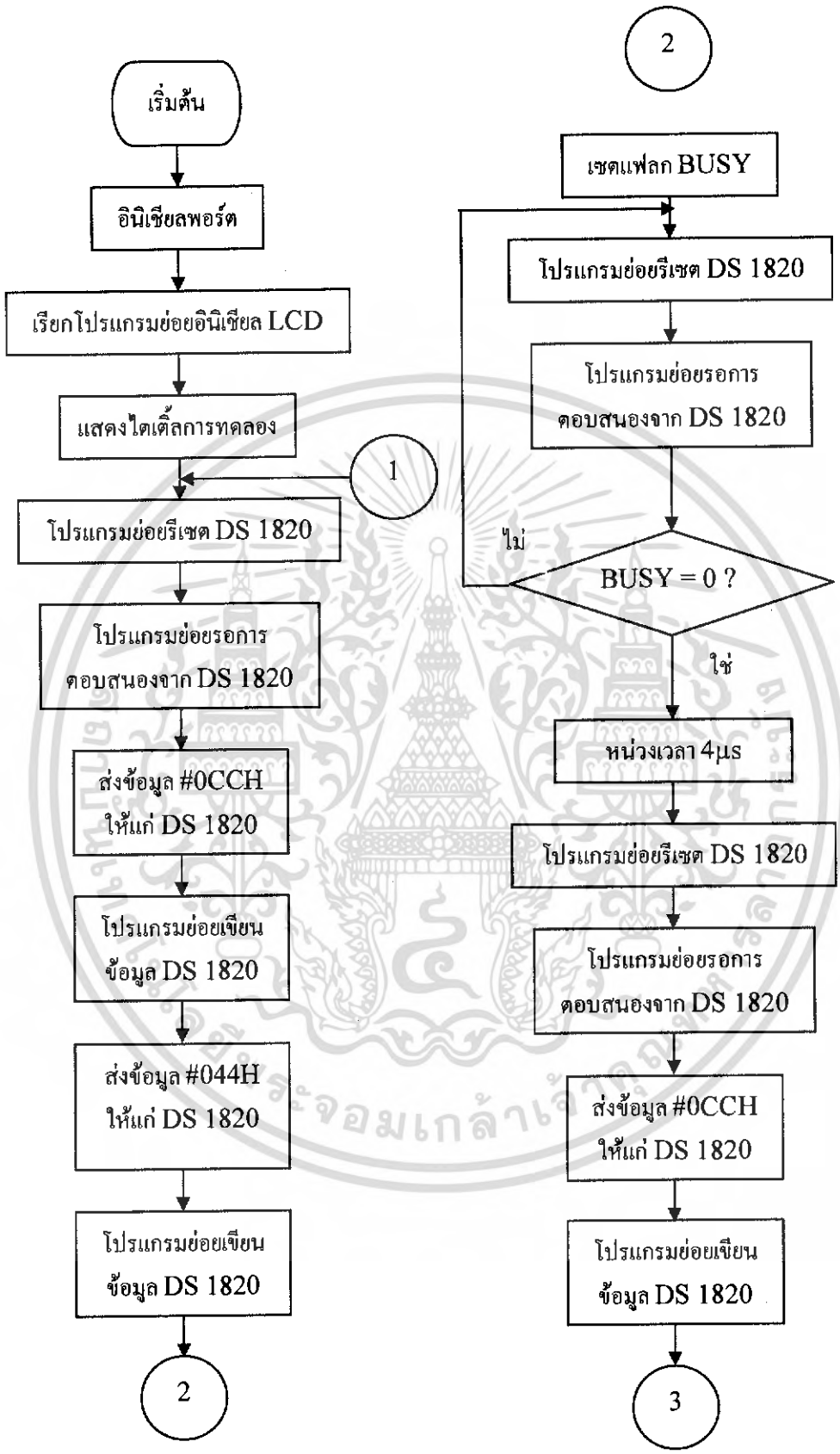
รูปที่ 4.15 แสดง FLOW CHART ของการกดสวิตช์จากหน้าเว็บเพื่อควบคุมหลอดไฟทั้ง 3 หลอด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



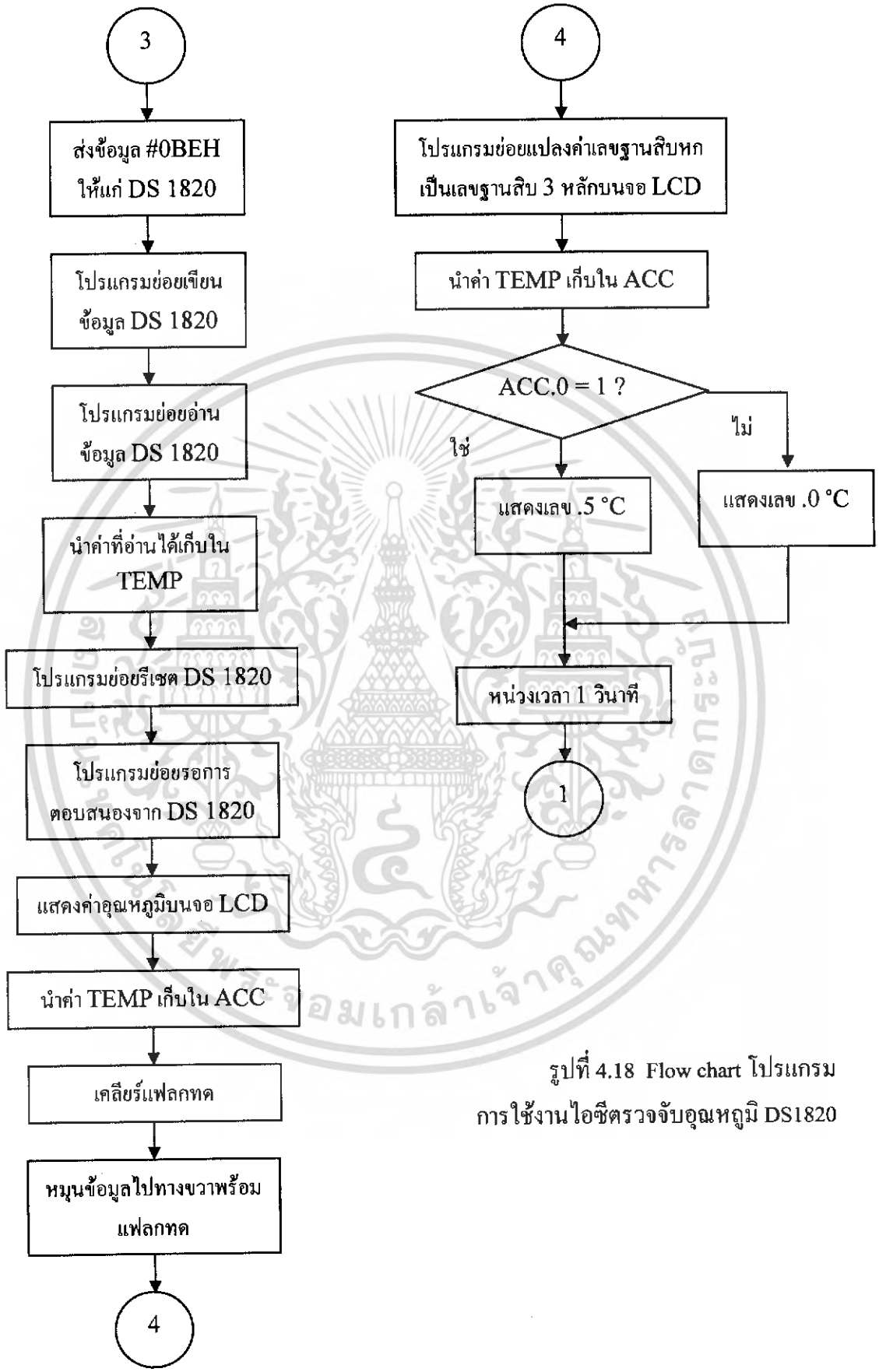
รูปที่ 4.16 แสดง FLOW CHART ของการแสดงผลสถานะของอินฟราเรดเซนเซอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.17 Flow chart โปรแกรมการใช้งานไอซีตรวจจับอุณหภูมิ DS1820

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.18 Flow chart โปรแกรมการใช้งานไอซีตรวจจับอุณหภูมิ DS1820

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 5

ผลการทดลอง

การทดลองจะถูกแบ่งออกเป็น 2 ส่วน คือ

1. การทดลองแสดงผลทางฮาร์ดแวร์
2. การทดลองแสดงผลผ่านทางหน้าเว็บ

เราจะทำการทดลองจากฮาร์ดแวร์ที่เราได้สร้างไว้แล้วผ่านทางหน้าเว็บคือวงจรควบคุมเปิด-ปิดหลอดไฟ ,วงจรรับส่งอินฟราเรด และวงจรตรวจจับอุณหภูมิ ซึ่งสามารถเปิด-ปิด และแสดงผลได้ตามที่ต้องการ

5.1 การทดลองการเปิด-ปิดหลอดไฟ

จะทำการเปิด-ปิดหลอดไฟโดย โดยการกดสวิตช์ที่หน้าเว็บ แล้วดูที่หลอดไฟที่ฮาร์ดแวร์และรูปหลอดไฟ LEDที่แสดงทางหน้าเว็บว่าดับ หรือสว่าง (มีการ on-off)

5.1.1 การทดลอง แสดงผลทางฮาร์ดแวร์

การทดลอง	การทดลองครั้งที่ 1		การทดลองครั้งที่ 2		การทดลองครั้งที่ 3	
	สถานะเมื่อให้อินพุทเป็น		สถานะเมื่อให้อินพุทเป็น		สถานะเมื่อให้อินพุทเป็น	
	HIGH	LOW	HIGH	LOW	HIGH	LOW
หลอดไฟ(LED1)	ดับ	สว่าง	ดับ	สว่าง	ดับ	สว่าง
LED 2	ดับ	สว่าง	ดับ	สว่าง	ดับ	สว่าง
LED 3	ดับ	สว่าง	ดับ	สว่าง	ดับ	สว่าง

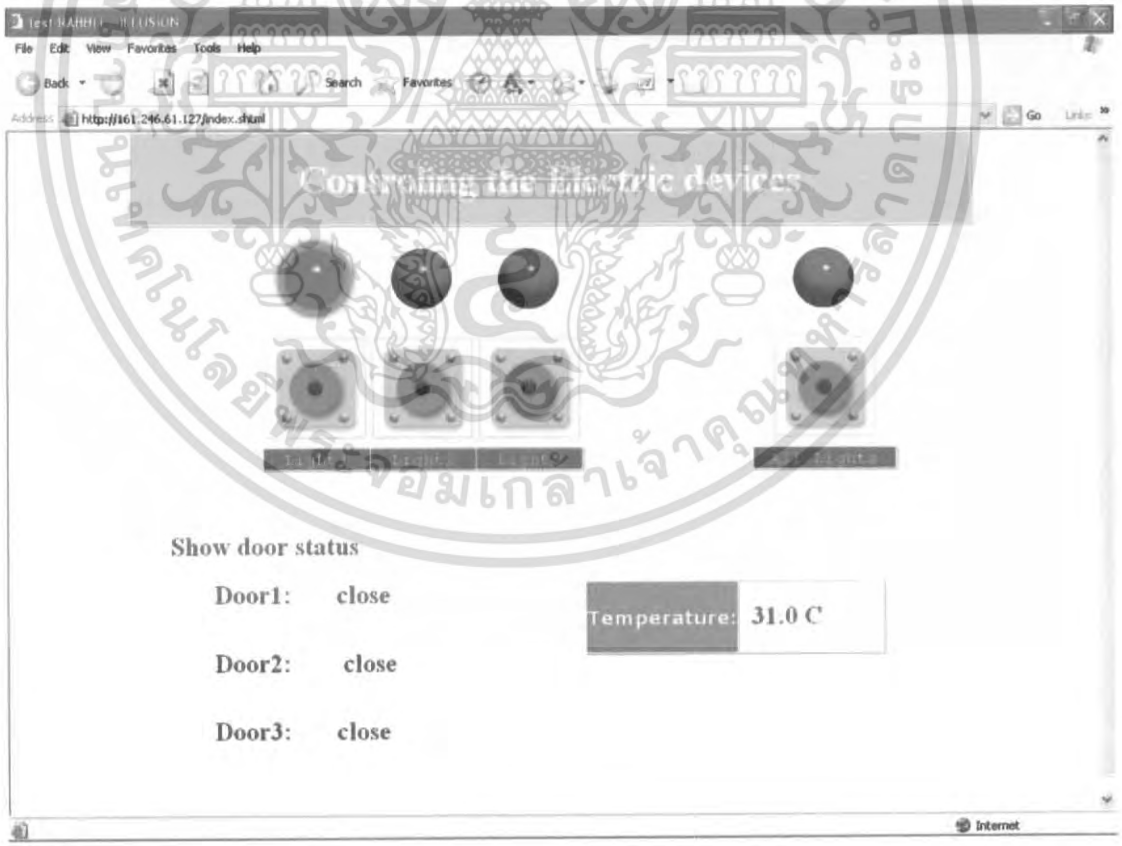
ตารางที่ 5.1 แสดงผลการควบคุมอุปกรณ์ไฟฟ้าแสดงผลทางฮาร์ดแวร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5.1.2 การทดลองแสดงผลทางเว็บ

การทดลอง	การทดลองครั้งที่ 1		การทดลองครั้งที่ 2		การทดลองครั้งที่ 3	
	สถานะเมื่อกดสวิตช์		สถานะเมื่อกดสวิตช์		สถานะเมื่อกดสวิตช์	
	OFF	ON	OFF	ON	OFF	ON
รูป LED1	ดับ	สว่าง	ดับ	สว่าง	ดับ	สว่าง
รูปLED 2	ดับ	สว่าง	ดับ	สว่าง	ดับ	สว่าง
รูปLED 3	ดับ	สว่าง	ดับ	สว่าง	ดับ	สว่าง

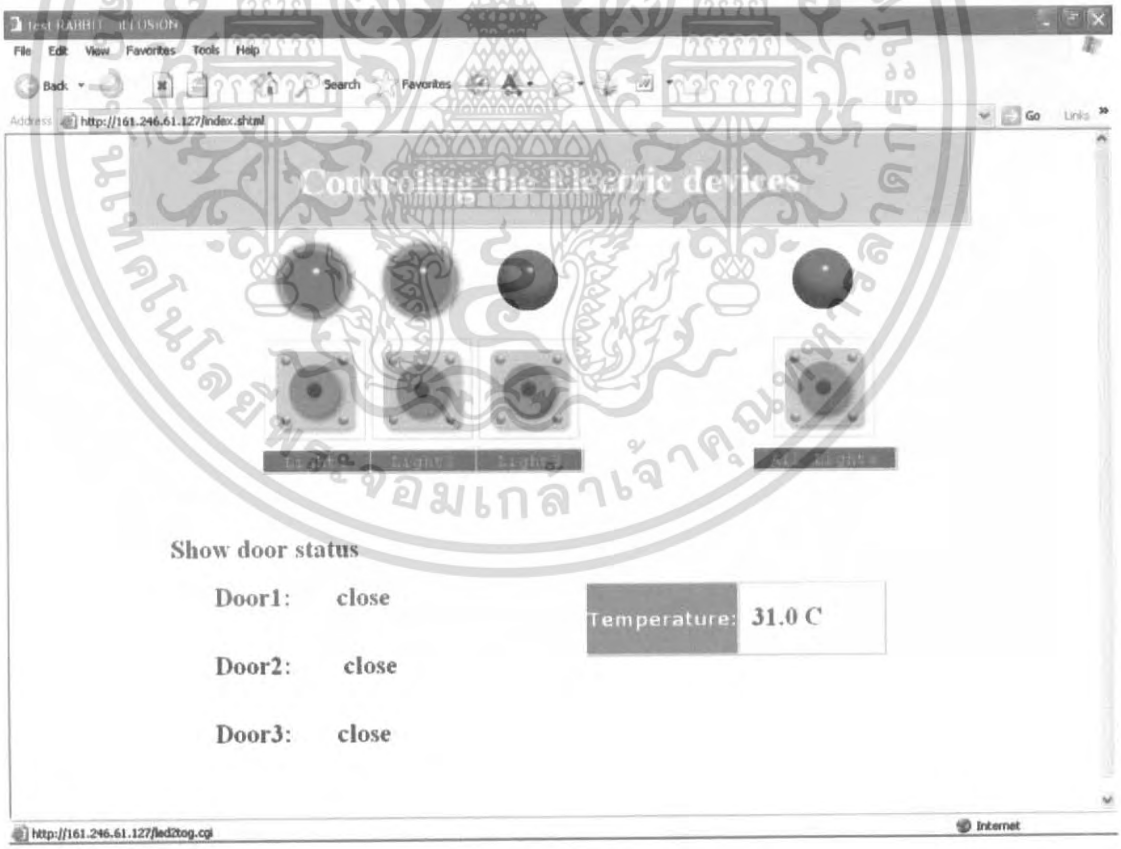
ตารางที่ 5.2 การทดลองแสดงผลการเปิด-ปิดหลอดไฟผ่านทางหน้าเว็บ



รูปที่ 5.1 แสดงสถานะของรูปหลอดไฟหลอดที่ 1 อยู่ในสถานะเปิดบนหน้าเว็บ

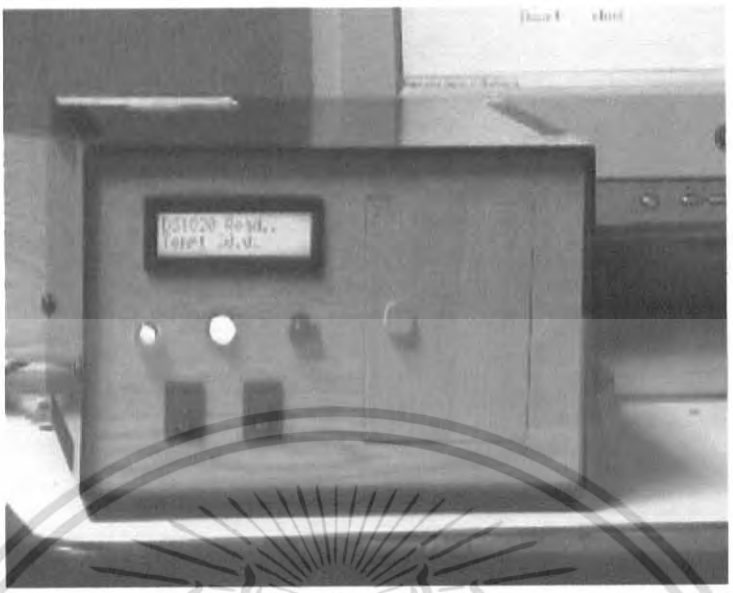
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 5.2 แสดงผลหลอดไฟหลอดที่1 สว่าง เมื่อมีการควบคุมจากทางหน้าเว็บ

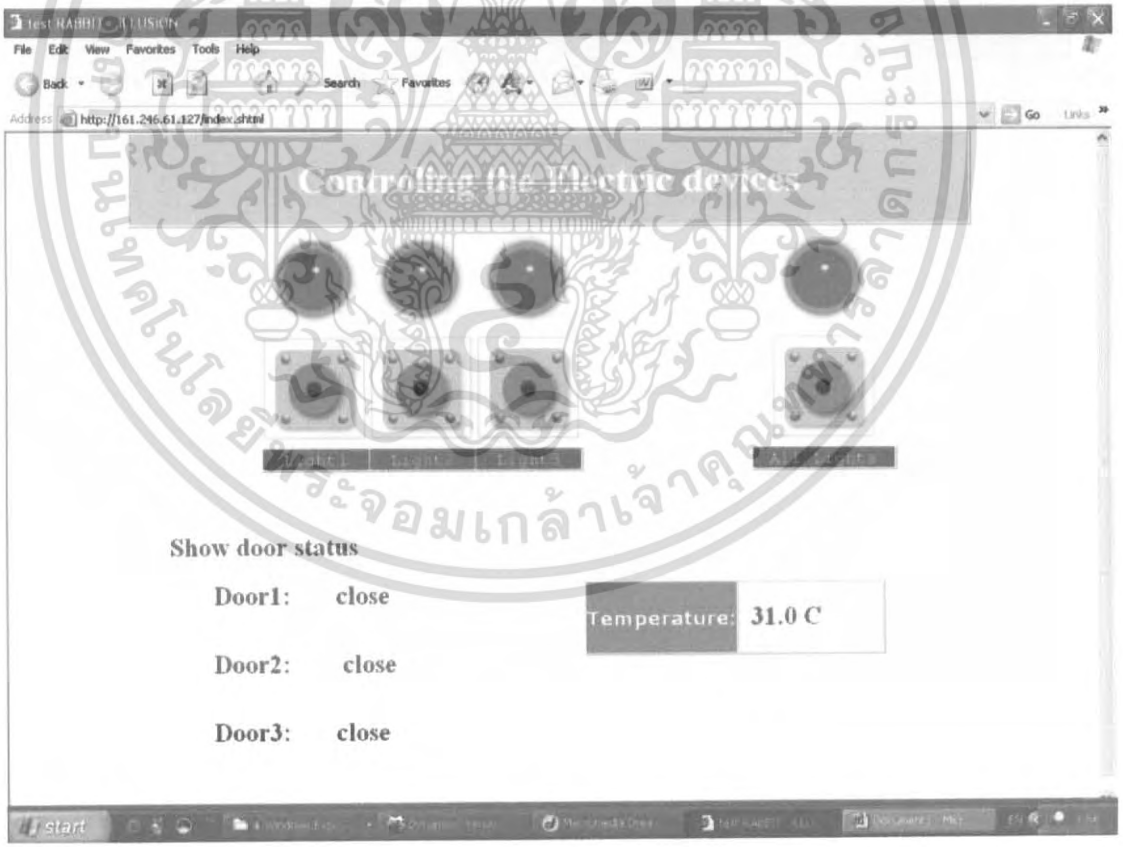


รูปที่ 5.3 แสดงสถานะของรูปหลอดไฟหลอดที่1และ 2 อยู่ในสถานะเปิดบนหน้าเว็บ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

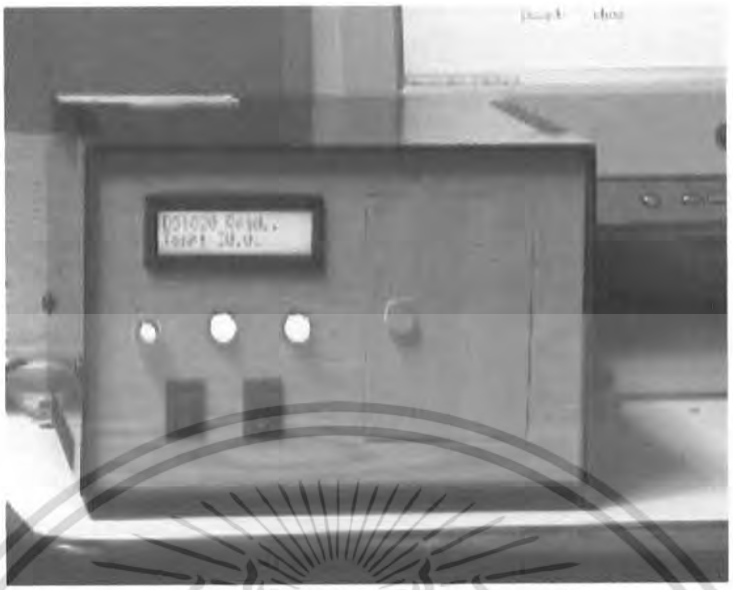


รูปที่ 5.4 แสดงผลหลอดไฟหลอดที่1และ 2 สว่าง เมื่อมีการควบคุมจากทางหน้าเว็บ



รูปที่ 5.5 แสดงสถานะของรูปหลอดไฟหลอดที่ 1,2 และ 3 อยู่ในสถานะเปิดบนหน้าเว็บ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 5.6 แสดงผลหลอดไฟหลอดที่1, 2 และ 3 สว่างเมื่อมีการควบคุมจากทางหน้าเว็บ

5.2 การทดลองการทำงานของ เซนเซอร์ อินฟราเรด

โดยเซนเซอร์ที่นำไปติดตั้งที่ประตูจำลองได้ทำการต่อวงจร เซนเซอร์ อินฟราเรดรับ-ส่ง จริง และจะใช้ switch อีก 2 ตัวจำลองการทำงานแทน เสมือนเป็นตัวรับ-ส่งอินฟราเรด

การทดลองจะเปลี่ยนสถานะจาก 0 เป็น 1 เมื่อประตูเปิด ในการทดลองจะทำการทดลอง 3 ครั้ง

5.2.1การทดลอง แสดงผลทางฮาร์ดแวร์

ตำแหน่งของ sensor	การทดลองครั้งที่ 1		การทดลองครั้งที่ 2		การทดลองครั้งที่ 3	
	เอาท์พุทเมื่อ		เอาท์พุทเมื่อ		เอาท์พุทเมื่อ	
	ประตูเปิด	ประตูปิด	ประตูเปิด	ประตูปิด	ประตูเปิด	ประตูปิด
sensor ประตู	HIGH	LOW	HIGH	LOW	HIGH	LOW
switch 1	LOW	HIGH	LOW	HIGH	LOW	HIGH
switch 2	LOW	HIGH	LOW	HIGH	LOW	HIGH

ตารางที่ 5.3 แสดงผลการทดลองการทำงานของ เซนเซอร์ และสวิตช์แสดงผลทางฮาร์ดแวร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

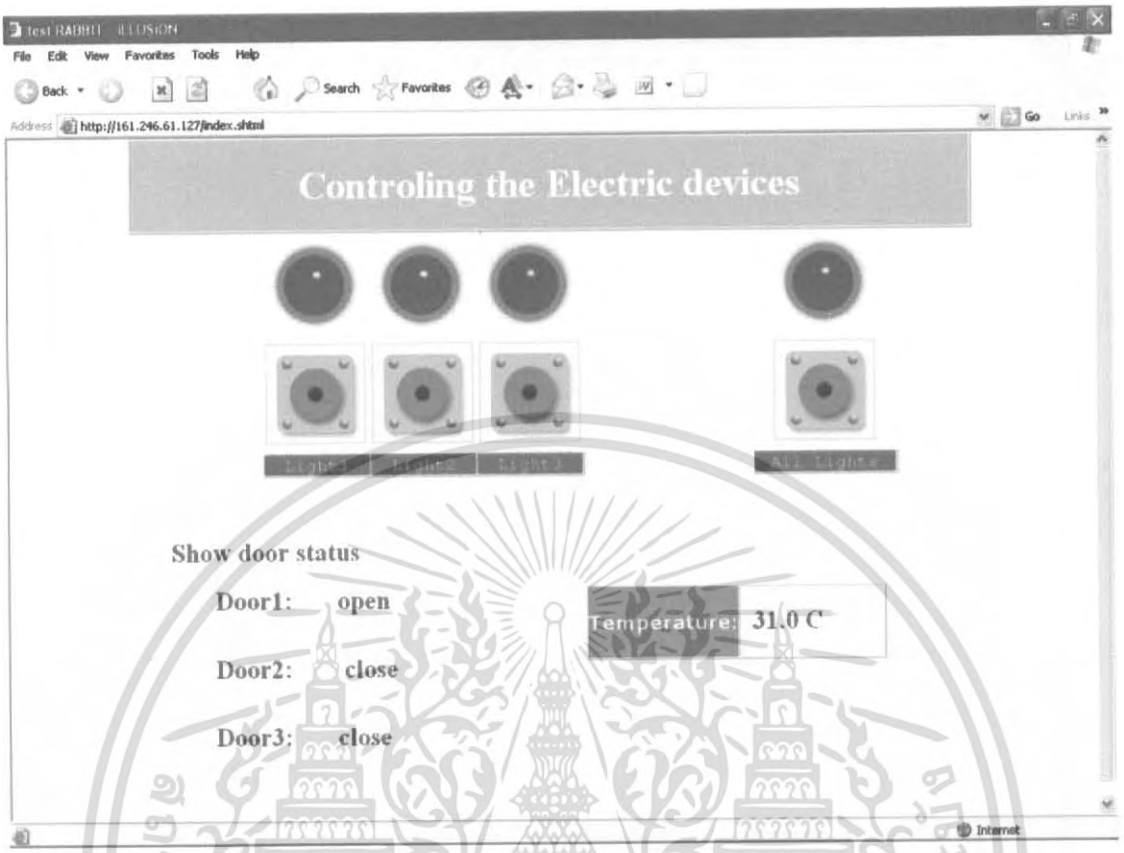
5.2.2 การทดลองแสดงผลทางเว็บ

ตำแหน่งของ sensor	การทดลองครั้งที่ 1		การทดลองครั้งที่ 2		การทดลองครั้งที่ 3	
	สถานะเมื่อ		สถานะเมื่อ		สถานะเมื่อ	
	ประตูเปิด	ประตูปิด	ประตูเปิด	ประตูปิด	ประตูเปิด	ประตูปิด
Door1	CLOSE	OPEN	CLOSE	OPEN	CLOSE	OPEN
Door2	CLOSE	OPEN	CLOSE	OPEN	CLOSE	OPEN
Door3	CLOSE	OPEN	CLOSE	OPEN	CLOSE	OPEN

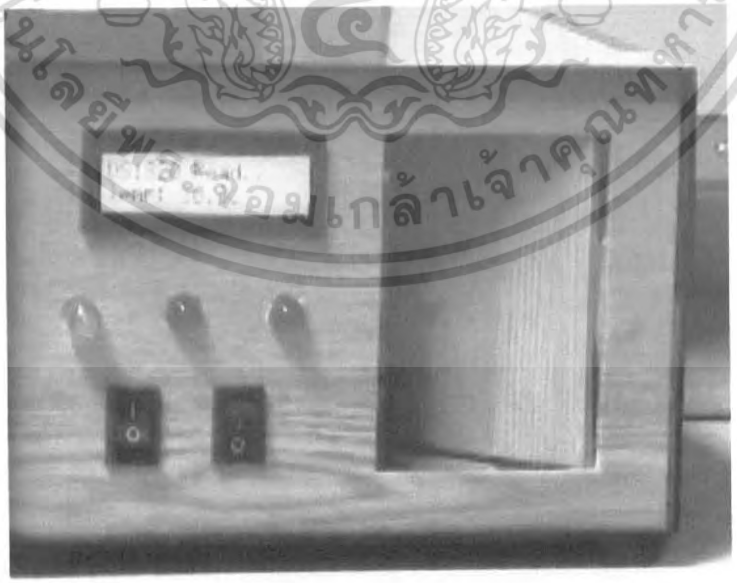
ตารางที่ 5.4 แสดงผลการทดลองการทำงานของเซนเซอร์ แสดงผลทางเว็บ

รูปที่ 5.7 แสดงสถานะประตูที่1 เปิดเพื่อส่งผลไปยังหน้าเว็บ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

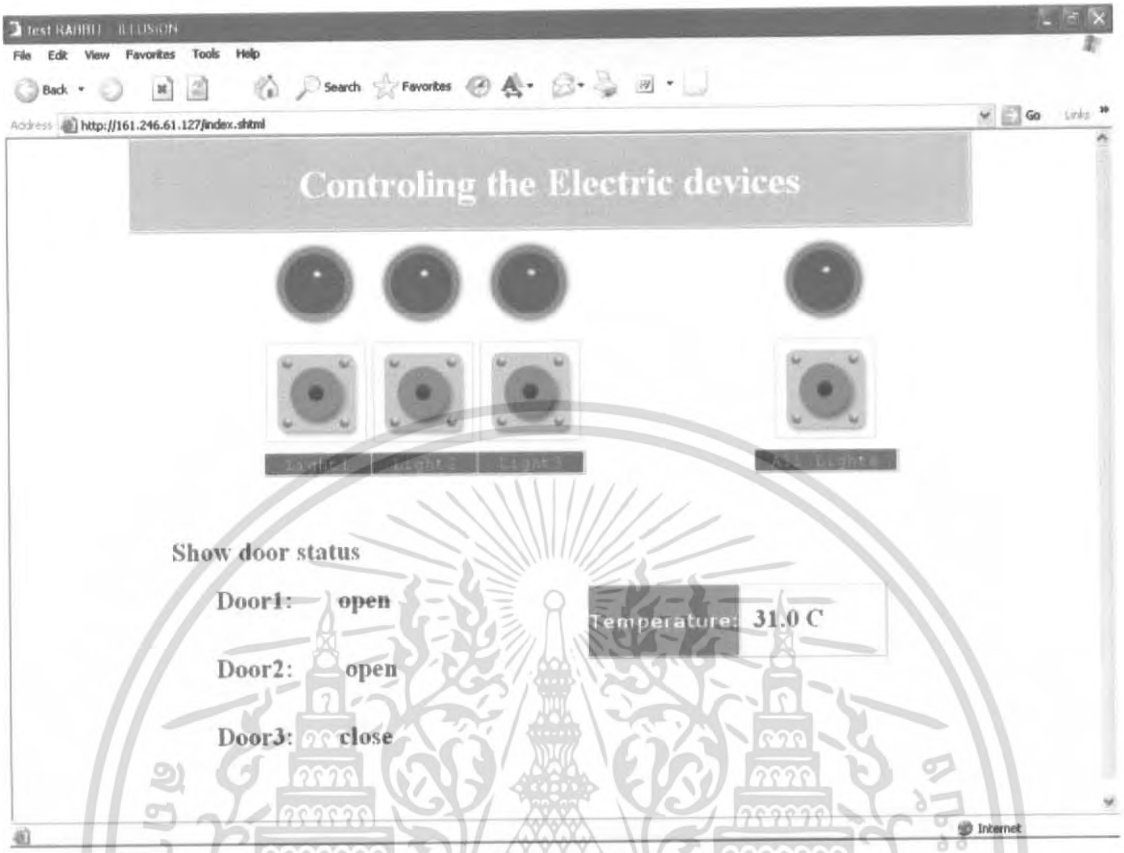


รูปที่ 5.8 แสดงสถานะประตูที่ 1 ปิด บนหน้าเว็บ



รูปที่ 5.9 แสดงสถานะประตูที่1 และ สวิทช์(เสมือนประตู 2) เปิด เพื่อส่งผลไปยังหน้าเว็บ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 5.10 แสดงสถานะประตูที่ 1 และ 2 เปิด บนหน้าเว็บ

5.3 การทดลองวงจรตรวจจับอุณหภูมิ

จะทำการวัดอุณหภูมิโดยใช้หัวแร้งเป็นตัวเพิ่มความร้อนแล้วดูค่าอุณหภูมิที่วัดได้จากทาง LCD และทางหน้าเว็บ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5.3.1 การทดลอง แสดงผลทาง ฮาร์ดแวร์

การทดลอง	อุณหภูมิก่อนการทดลอง (องศาเซลเซียส)	อุณหภูมิหลังการทดลอง (องศาเซลเซียส)
นำหัวแร้งไปเข้าใกล้เป็นเวลา 20 วินาที	26	34
นำหัวแร้งไปเข้าใกล้เป็นเวลา 60 วินาที	26.5	45

ตารางที่ 5.5 แสดงผลการทดลองการตรวจวัดค่าอุณหภูมิแสดงผลทางLCD

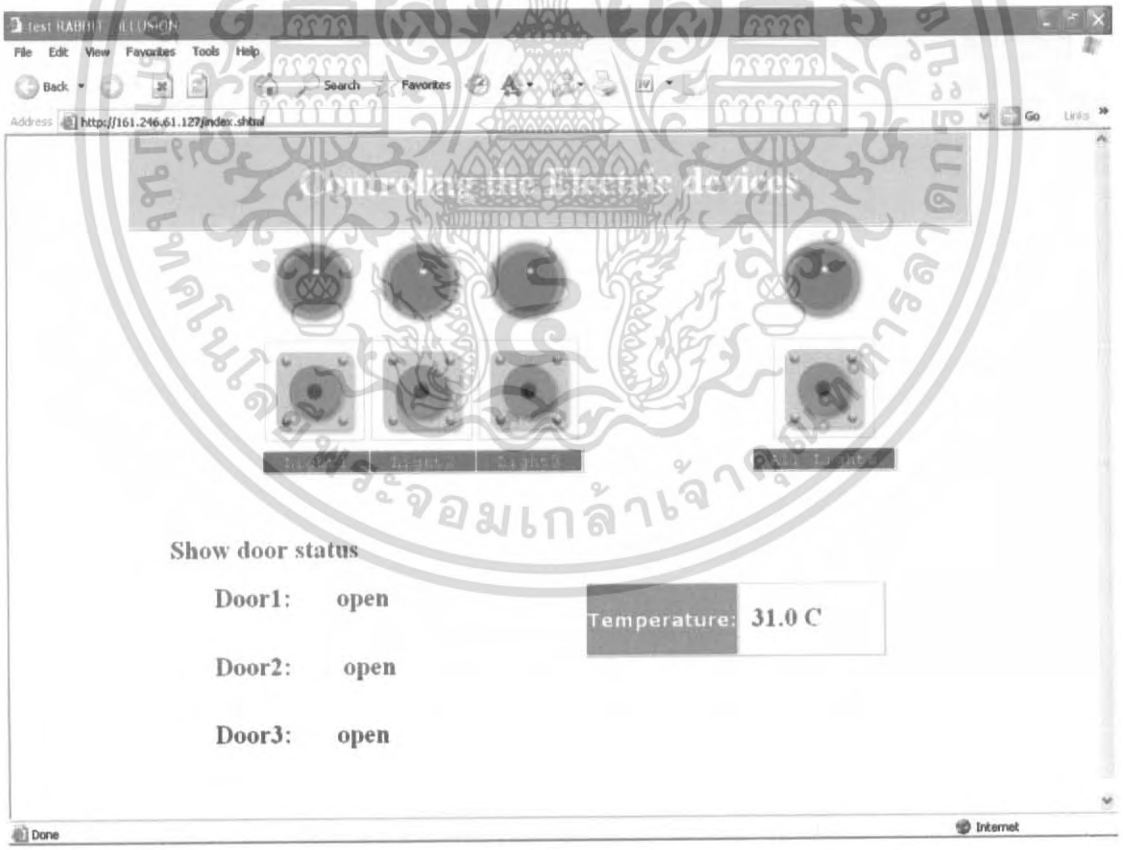
5.3.2 การทดลองแสดงผลทางหน้าเว็บ

การทดลอง	อุณหภูมิก่อนการทดลอง (องศาเซลเซียส)	อุณหภูมิหลังการทดลอง (องศาเซลเซียส)
นำหัวแร้งไปเข้าใกล้เป็นเวลา 20 วินาที	26	34
นำหัวแร้งไปเข้าใกล้เป็นเวลา 60 วินาที	26.5	45

ดารางที่ 5.6 แสดงผลการทดลองการตรวจวัดค่าอุณหภูมิแสดงผลทางหน้าWEB

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 5.11 แสดงผลอุณหภูมิที่ได้ทางจอ LCD



รูปที่ 5.12 แสดงผลอุณหภูมิที่ได้บนหน้าเว็บ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 6
สรุปและวิจารณ์ผลการทดลอง

โครงงานนี้ได้ทำการควบคุมการเปิด-ปิดอุปกรณ์ไฟฟ้า และแสดงสถานะต่างๆ ผ่านทางหน้าเว็บแบ่งเป็น

1. การควบคุมการเปิด-ปิดของหลอดไฟ สามารถที่จะทำการเปิด-ปิดได้โดยสั่งจากทางหน้าเว็บ
2. ตัวรับส่งอินฟราเรดและสวิทช์ สามารถที่จะบอกสถานะของการเปิด-ปิดประตู และแสดงผลออกทางหน้าเว็บได้
3. วงจรตรวจวัดอุณหภูมินั้นสามารถที่จะส่งค่าอุณหภูมิที่ตรวจจับได้ แสดงผลทางLCDและทางหน้าเว็บ

จากการทดลองโครงงาน

ได้ทำการทดลองการรับส่งข้อมูลของสถานะเซนเซอร์ ของการเปิด-ปิดประตู และการสับสวิตช์ที่เสมือนมีการเปิด-ปิดประตู สามารถส่งสถานะแสดงผลผ่านทางหน้าเว็บได้ตามที่ต้องการ

ในส่วนของการเปิด-ปิดหลอดไฟจากการกดสวิตช์ทางหน้าเว็บนั้นให้ผลการทดลองเป็นที่พอใจ ซึ่งสามารถสั่งงานควบคุมการ ปิด-เปิดไฟได้อย่างแม่นยำและรวดเร็ว

วงจรตรวจจับอุณหภูมิสามารถวัดค่าอุณหภูมิได้ และสามารถแสดงผลได้ทั้งทาง LCD และทางหน้าเว็บได้อย่างใกล้เคียงกับสภาพอากาศ ซึ่งค่าอุณหภูมิตามสภาพอากาศมีการเปลี่ยนแปลงตลอดเวลา การตรวจวัดค่าอุณหภูมิจะมีการเปลี่ยนแปลงค่าทุกๆ 5 วินาที หรือมีการ กดคำสั่งต่างๆ บนหน้าเว็บ

โครงงานนี้ได้ทำการทดลองโดยทดลองจากการนำอุปกรณ์ฮาร์ดแวร์ไว้ที่ตึกอธิการ และควบคุมผ่านทางหน้าเว็บจากตึกต่างๆในสถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง เริ่มจาก ตึกภาควิศวกรรมไฟฟ้า ,ตึก ภาคอิเล็กทรอนิกส์ และตึก ศูนย์วิจัยคอมพิวเตอร์ และทั้งนี้ยังสามารถเรียกใช้งานหน้าเว็บและควบคุมผ่านอินเตอร์เน็ทได้อีกด้วยโดยทดสอบจากการเรียกใช้งานหน้าเว็บจากนอกสถาบันได้ ซึ่งโครงงานนี้ให้ผลเป็นที่น่าพอใจ และสามารถใช้งานได้จริงตามที่ต้องการ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บรรณานุกรม

1. ชื่อหนังสือ การพัฒนาไมโครคอนโทรลเลอร์ด้วยภาษาซี

   ผู้แต่ง : ผู้ช่วยศาสตราจารย์ธีรวัฒน์ ประกอบผล

   พิมพ์ปีที่3 ธันวาคม 2545

   จัดพิมพ์โดย สำนักพิมพ์ ส.ส.ท. (สมาคมส่งเสริมเทคโนโลยี(ไทย-ญี่ปุ่น))

   ISBN : 974-8329-60-7

2. ข้อมูล RABBIT ETHANET KIT จาก WWW.ZWORLD.COM

3. ข้อมูลจากWWW.ALLDATASHEET.COM

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

PROGRAM CODE

```
#class auto


#define CINBUFSIZE  15
#define COUTBUFSIZE 15
#define maxs 4

#define TCPCONFIG 1

#define TCP_BUF_SIZE 2048

#define HTTP_MAXSERVERS 2
#define MAX_TCP_SOCKET_BUFFERS 2

#define REDIRECTTO      "http://" REDIRECTHOST "/index.shtml"


#define REDIRECTHOST                    _PRIMARY_STATIC_IP

#memmap xmem
#use "dcrtcp.lib"
#use "http.lib"


#ximport "samples/rcm3700/tcpip/pages/yee1+TEMP.htm" index_html
#ximport "samples/rcm3700/tcpip/pages/rabbit1.gif"        rabbit1_gif
#ximport "samples/rcm3700/tcpip/pages/ledon.gif"          ledon_gif
#ximport "samples/rcm3700/tcpip/pages/ledoff.gif"         ledoff_gif
#ximport "samples/rcm3700/tcpip/pages/button.gif"         button_gif
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
#ximport "samples/rcm3700/tcpip/pages/showsrc.shtml"    showsrc_shtml
#ximport "samples/rcm3700/tcpip/browseled.c"            browseled_c

SSPEC_MIMETABLE_START
        SSPEC_MIME_FUNC(".shtml", "text/html", shtml_handler),
        SSPEC_MIME(".html", "text/html"),
        SSPEC_MIME(".gif", "image/gif"),
        SSPEC_MIME(".cgi", "")
SSPEC_MIMETABLE_END

char led1[15];
char led2[15];
char led3[15];
char led4[15];
char show1[10];
char show2[10];
char show3[10];
char ser[10];


/*******************OPEN/CLOSE LIGHT 1 FROM WEB*****************/

int led1toggle(HttpState* state)
{
  int a,b;
  a=0;
  b=1;
  if (strcmp(led1,"ledon.gif")==0)
        {
        strcpy(led1,"ledoff.gif");
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        BitWrPortI(PBDR, &PBDRShadow, a, 2);
        }
  else
        {
        strcpy(led1,"ledon.gif");
        BitWrPortI(PBDR, &PBDRShadow, b, 2);
        }

        if (strcmp(led1,"ledon.gif")==0)
                if (strcmp(led3,"ledon.gif")==0)
                if (strcmp(led2,"ledon.gif")==0)
                        strcpy(led4,"ledon.gif");
                    else
                        strcpy(led4,"ledoff.gif");
                else
                    strcpy(led4,"ledoff.gif");
        else
            strcpy(led4,"ledoff.gif");

  cgi_redirectto(state,REDIRECTTO);

  return 0;
}


/******************OPEN/CLOSE LIGHT 2 FROM WEB****************/

int led2toggle(HttpState* state)
{
  int a,b;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
a=0;
b=1;
if (strcmp(led2,"ledon.gif")==0)
        {
        strcpy(led2,"ledoff.gif");
        BitWrPortI(PBDR, &PBDRShadow, a, 3);
        }


else
   {
        strcpy(led2,"ledon.gif");
        BitWrPortI(PBDR, &PBDRShadow, b, 3);
   }
        if (strcmp(led1,"ledon.gif")==0)
                if (strcmp(led3,"ledon.gif")==0)
                if (strcmp(led2,"ledon.gif")==0)
                        strcpy(led4,"ledon.gif");
                    else
                        strcpy(led4,"ledoff.gif");
                else
                   strcpy(led4,"ledoff.gif");
        else
           strcpy(led4,"ledoff.gif");


  cgi_redirectto(state,REDIRECTTO);
  return 0;
}

/*******************OPEN/CLOSE LIGHT 3 FROM WEB****************/
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
int led3toggle(HttpState* state)
{
  int a,b;
  a=0;
  b=1;
  if (strcmp(led3,"ledon.gif")==0)
        {
        strcpy(led3,"ledoff.gif");
        BitWrPortI(PBDR, &PBDRShadow, a, 4);
        }
  else
        {
        strcpy(led3,"ledon.gif");
        BitWrPortI(PBDR, &PBDRShadow, b, 4);
        }

        if (strcmp(led1,"ledon.gif")==0)
                if (strcmp(led3,"ledon.gif")==0)
                if (strcmp(led2,"ledon.gif")==0)
                        strcpy(led4,"ledon.gif");
                      else
                        strcpy(led4,"ledoff.gif");
                else
                   strcpy(led4,"ledoff.gif");
        else
           strcpy(led4,"ledoff.gif");


  cgi_redirectto(state,REDIRECTTO);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
  return 0;
}

/*****************OPEN/CLOSE ALL LIGHT FROM WEB*****************/

int led4toggle(HttpState* state)
{
  int a,b;
  a=0;
  b=1;
  if (strcmp(led4,"ledon.gif")==0)
        {
        strcpy(led4,"ledoff.gif");
        BitWrPortI(PBDR, &PBDRShadow, a, 2);
        BitWrPortI(PBDR, &PBDRShadow, a, 3);
        BitWrPortI(PBDR, &PBDRShadow, a, 4);
        strcpy(led1,"ledoff.gif");
        strcpy(led2,"ledoff.gif");
        strcpy(led3,"ledoff.gif");
        }
  else
        {
        strcpy(led4,"ledon.gif");
        BitWrPortI(PBDR, &PBDRShadow, b, 2);
        BitWrPortI(PBDR, &PBDRShadow, b, 3);
        BitWrPortI(PBDR, &PBDRShadow, b, 4);
        strcpy(led1,"ledon.gif");
         strcpy(led2,"ledon.gif");
        strcpy(led3,"ledon.gif");
        }
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
  cgi_redirectto(state,REDIRECTTO);
  return 0;
}

/********************* CHANGE CODE C TO HTML*******************/

SSPEC_RESOURCETABLE_START
        SSPEC_RESOURCE_XMEMFILE("/", index_html),
        SSPEC_RESOURCE_XMEMFILE("/index.shtml", index_html),
        SSPEC_RESOURCE_XMEMFILE("/showsrc.shtml", showsrc_shtml),
        SSPEC_RESOURCE_XMEMFILE("/rabbit1.gif", rabbit1_gif),
        SSPEC_RESOURCE_XMEMFILE("/ledon.gif", ledon_gif),
        SSPEC_RESOURCE_XMEMFILE("/ledoff.gif", ledoff_gif),
        SSPEC_RESOURCE_XMEMFILE("/button.gif", button_gif),
        SSPEC_RESOURCE_XMEMFILE("browseled.c", browseled_c),
        SSPEC_RESOURCE_ROOTVAR("led1", led1, PTR16, "%s"),
        SSPEC_RESOURCE_ROOTVAR("led2", led2, PTR16, "%s"),
        SSPEC_RESOURCE_ROOTVAR("led3", led3, PTR16, "%s"),
        SSPEC_RESOURCE_ROOTVAR("led4", led4, PTR16, "%s"),
        SSPEC_RESOURCE_FUNCTION("/led1tog.cgi", led1toggle),
        SSPEC_RESOURCE_FUNCTION("/led2tog.cgi", led2toggle),
        SSPEC_RESOURCE_FUNCTION("/led3tog.cgi", led3toggle),
        SSPEC_RESOURCE_FUNCTION("/led4tog.cgi", led4toggle),
         SSPEC_RESOURCE_ROOTVAR("show1", &show1, PTR16, "%s"),
        SSPEC_RESOURCE_ROOTVAR("show2", &show2, PTR16, "%s"),
        SSPEC_RESOURCE_ROOTVAR("show3", &show3, PTR16, "%s"),
        SSPEC_RESOURCE_ROOTVAR("ser", &ser, PTR16, "%s"),
SSPEC_RESOURCETABLE_END
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
/*****************SHOW DOORS STATUS*****************************/

void input()
{
  if (BitRdPortI(PADR, 0))
        //show1 = "open" ;
    strcpy(show1,"open");
  else
        //show1 = "close" ;
    strcpy(show1,"close");

  if (BitRdPortI(PADR, 1))
        //show2 = "close" ;
    strcpy(show2," close ");
  else
        //show2 = "open" ;
    strcpy(show2," open ");

  if (BitRdPortI(PADR, 2))
        //show3 = " close " ;
    strcpy(show3," close ");
  else
        //show3 = " open " ;
    strcpy(show3," open ");
}
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
/*************************MAIN PROGRAM*******************/

main()
{
  int getOk,bb ;

  char tempc[maxs + 1];
  brdInit();
  strcpy(ser,"  ");
  strcpy(led1,"ledoff.gif");
  strcpy(led2,"ledon.gif");
  strcpy(led3,"ledon.gif");
  strcpy(led4,"ledoff.gif");
  input();
  sock_init();
  http_init();
  tcp_reserveport(80);
  serCopen(9600);
  getOk=0;
  bb=0;
  while (1)
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
{
    costate
        {
        wfd getOk = cof_serCgets(tempc, maxs, 3000);
        if (getOk)
                {
                strcat(tempc," C") ;
                strcpy(ser,tempc);
                }
        }

    costate
        {
        input();
        http_handler();
        }
  }
  serCclose();
}


#nodebug
```

**********HTML CODE**********

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD W3 HTML//EN">
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
<HTML>
<HEAD>
        <TITLE>test RABBIT</TITLE>
<style type="text/css">
<!--
body,td,th {
        font-family: Courier New, Courier, mono;
}
body {
        background-image: url();
        background-color: #FFFFFF;
}
.style9 {color: #FFFFFF}
.style10 {
        color: #003366;
        font-weight: bold;
        font-size: 24px;
        font-family: "Times New Roman", Times, serif;
}
.style11 {color: #993300}
.style13 {
        color: #FFFFFF;
        font-weight: bold;
        font-family: Verdana, Arial, Helvetica, sans-serif;
}
.style16 {color: #FF0000}
.style21 {color: #660099; font-weight: bold; font-size: 24px; font-family: "Times New Roman",
Times, serif; }
.style23 {color: #CC0066; font-weight: bold; font-size: 24px; font-family: "Times New Roman",
Times, serif; }
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
.style25 {color: #FFFFFF; font-weight: bold; font-family: Verdana, Arial, Helvetica, sans-serif;
font-size: 18px; }
.style27 {
        font-size: 36px;
        font-weight: bold;
        font-family: "Times New Roman", Times, serif;
        color: #FFFFFF;
}
-->
</style><meta http-equiv="Content-Type" content="text/html; charset=iso-8859-1"></HEAD>

<BODY topmargin="0" leftmargin="0" marginwidth="0" marginheight="0" link="#009966"
vlink="#FFCC00" alink="#006666"
onLoad=window.setTimeout("location.href='index.shtml'",10000)>

<!--img SRC="rabbit1.gif"-->
<div align="center">
 <table width="780" border="2" cellspacing="0" cellpadding="0">
  <tr>
   <td height="85" bgcolor="#F9AA06"><div align="center">
    <p class="style27">Controling the Electric devices </p>
    </div></td>
  </tr>
 </table>
 <table width="780" border="0" cellpadding="0" cellspacing="0" background="bg.gif">
  <tr>
   <td>    <div align="center">
     </div>
    <table width="100%" border="0" cellspacing="0" cellpadding="0">
     <tr>
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
<td> </td>
<td><TABLE width="299" BORDER="0" CELLPADDING="1" CELLSPACING="2">
 <TR>
  <TD>
   <div align="center"><img SRC="<!--#echo var="led1"-->"> </div></TD>
  <TD>
   <div align="center"><img SRC="<!--#echo var="led2"-->"> </div></TD>
  <TD>
   <div align="center"><img SRC="<!--#echo var="led3"-->"> </div></TD>
 </TR>
 <TR>
  <TD>
   <div align="center"><A HREF="/led1tog.cgi"> <img SRC="button.gif"> </A>
</div></TD>
  <TD>
   <div align="center"><A HREF="/led2tog.cgi"> <img SRC="button.gif"> </A>
</div></TD>
  <TD>
   <div align="center"><A HREF="/led3tog.cgi"> <img SRC="button.gif"> </A>
</div></TD>
 </TR>
</TABLE></td>
<td> </td>
<td><TABLE width="135" BORDER="0" CELLPADDING="1" CELLSPACING="2">
 <TR>
  <TD width="144">
   <div align="center"><img src="<!--#echo var="led4"-->"> </div></TD>
 </TR>
 <TR>
  <TD>
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
            <div  align="center"><A  HREF="/led4tog.cgi">  <img  src="button.gif">  </A>
</div></TD>
          </TR>
         </TABLE></td>
         <td> </td>
        </tr>
        <tr>
         <td width="17%"> </td>
         <td width="38%"><table width="100%"  border="2" cellspacing="0" cellpadding="0">
          <tr>
           <td bgcolor="#003366"><div align="center" class="style9">Light1</div></td>
           <td bgcolor="#003366"><div align="center" class="style9">Light2</div></td>
           <td bgcolor="#003366"><div align="center" class="style9">Light3</div></td>
          </tr>
         </table></td>
         <td width="21%"> </td>
         <td width="14%" bgcolor="#FFFFF7"><div align="center" class="style9">
         <table width="100%"  border="2" cellspacing="0" cellpadding="0">
          <tr>
           <td  bgcolor="#003366"><div  align="center"><span  class="style9">All  Lights
</span></div></td>
          </tr>
         </table>
        </div></td>
         <td width="10%"><p> </p></td>
        </tr>
        <tr>
         <td height="18" colspan="5"> </td>
        </tr>
       </table>
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
<table width="780"  border="0" cellspacing="0" cellpadding="0">
 <tr>
  <td width="5%" height="234"> </td>
  <td width="49%"><p> </p>
   <p><span class="style23">Show door status</span></p>
   <table width="91%"  border="0" cellspacing="0" cellpadding="0">
   <tr>
    <td width="44%" height="63"><div align="center">
      <p class="style11"><span class="style10">Door1:</span></p>
      <p class="style11"> </p>
    </div></td>
    <td width="56%"><p class="style21"><!--#echo var="show1"--> </p>
     <p> </p></td>
   </tr>
   <tr>
    <td><div align="center">
      <p class="style10">Door2:</p>
      <p> </p>
    </div></td>
    <td><p  class="style16"><span  class="style21"> <!--#echo  var="show2"--
></span></p>
     <p> </p></td>
   </tr>
   <tr>
    <td height="63"><div align="center">
      <p class="style11"><span class="style10">Door3:</span></p>
      <p> </p>
    </div></td>
    <td><p  class="style16"><span  class="style21"><!--#echo  var="show3"--
> </span></p>
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
            <p> </p></td>
          </tr>
        </table></td>
        <td width="40%"><table width="89%"  border="2" cellpadding="0" cellspacing="0">
          <tr>
            <td      width="33%"      height="66"      bgcolor="#A65300"><div      align="center"
class="style13"><span class="style25">Temperature:</span></div></td>
            <td width="67%"><table width="95%"  border="0" cellspacing="0" cellpadding="0">
              <tr>
                <td width="84%"><span class="style10"> 
                    <!--#echo var="ser"-->
                </span></td>
                <td                                                    width="8%"><span
class="style10">&#3661;&#3661;&#3661;&#3661;</span></td>
                <td width="8%"> </td>
              </tr>
            </table></td>
          </tr>
        </table>          <p> </p></td>
        <td width="6%"> </td>
      </tr>
    </table>
    <p align="center"> </p>
  </td>
 </tr>
</table>
</div>
</BODY>
</HTML>
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
**********Program temperature**********

#include<reg51.h>
sbit e = P3^6;
sbit rs = P3^7;
#include<intrins.h>
sbit latch = P1^4;
sbit onewire = P2^7;
bit ans;
code unsigned char *ptr1 = "DS1820 Read..";
code unsigned char *ptr2 = "Temp:   . c";
unsigned char temp = 0,temp_1 = 0,temp_h = 0,half = 0;
memory
unsigned int i;    // Counter for delay time (for 1-wire bus)
unsigned char j;
/*************************************************************************/

void delay(int tick)
{
        unsigned int i,j;
        for(i=0;i<tick;i++)
        for(j=0;j<250;j++);
}
void lcd_command(unsigned char com)
{
        rs = 0;
        e = 1;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        P0 = com;
        delay(10);
}
void lcd_text(unsigned char text)
{
        rs = 1;
        e = 1;
        P0 = text;
        delay(10);
        e = 0;
        delay(10);
}
void lcd_clear()
{
        lcd_command(0x01);
}
void lcd_jumporigin()
{
        lcd_command(0x02);
}
void lcd_init()
{
        delay(250);
        delay(250);
        lcd_command(0x38);
        lcd_command(0x0C);
        lcd_command(0x01);
}
/* **********Function 1-wire Start condition **********/
void ds1820_reset()
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
{
        onewire = 0;
        for(i=0;i<100;i++) _nop_();
        {while(i>0) i--; ==> 8 machine cycle}
        onewire = 1;
        for(i=0;i<2;i++) _nop_();
}
/********** Function Check 1-wire acknowledge ******************/
void ds1820_ans(void)
{
        ans = 0;
        while(onewire);
        while(~onewire);
        for(i=0;i<2;i++) _nop_();
}
/************ Function Read data 1 time bit ********************/
bit readbit (void)
{
        bit dat;
        onewire = 0;
        _nop_();
        _nop_();
        onewire = 1;
        _nop_();
        _nop_();
        _nop_();
        _nop_();
        dat = onewire;
   for(i=0;i<8;i++) _nop_();
        return(dat);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
}
/******** Function Read data 1 byte(Temperature) ******************/
unsigned char ds1820_read()
{
        unsigned char i,j,dat;
        dat = 0;
        for(i=0;i<8;i++)
        {
                j = readbit();
                dat = (j<<7)|(dat>>1);
        }
        return(dat);
}

/****** Function Write data to 1-wire bus ************************/
void ds1820_write(unsigned char com)
{
        unsigned char j;
        bit send;
        for(j=0;j<8;j++)
        {
                send = com & 0x01;
                com = com>>1;
                if(send)
                {
                        onewire = 0;
                        _nop_();
                        _nop_();
                        _nop_();
                        _nop_();
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                onewire = 1;
                for(i=0;i<8;i++) _nop_();
            }
            else
{
                onewire = 0;
                for(i=0;i<8;i++) _nop_();
                onewire = 1;
                _nop_();
                _nop_();
                _nop_();
                _nop_();
            }
        }
}

/****** Function Delay during convert Temperature ******************/
void delay_convert(unsigned int count)
{
        do
        {
            count--;
        }while(count>0);
}
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
/************************ Main loop ************************/
void main()
{
        unsigned char j;

                TMOD = 0x21;
                SCON = 0x50;
                TH1 = 0xFD;
                TL1 = 0xFD;
                TF1 = 0;
                TI = 0;
                TR1 = 1;

        P0 = 0x00;
        latch = 0;
        lcd_init();

        lcd_jumporigin();
        lcd_command(0x80);
        for(j=0;j<13;j++)
        lcd_text(*(ptr1+j));

        lcd_jumporigin();
        lcd_command(0xC0);
        for(j=0;j<11;j++)
        lcd_text(*(ptr2+j));
        while(1)
        {
                lcd_command(0xC6);
                ds1820_reset();
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
ds1820_ans();
ds1820_write(0xCC);
ds1820_write(0x44);
delay_convert(50000);

ds1820_reset();
ds1820_ans();
ds1820_write(0xCC);
ds1820_write(0xBE);
temp = ds1820_read();
half = temp & 0x01;
if(half)
half = 0x35;
else half = 0x30;
temp = temp>>1;
temp_h = (temp/10)|0x30;
temp_l = (temp%10)|0x30;

SBUF = temp_h;
while(~TI);
TI = 0;

SBUF = temp_l;
while(~TI);
TI = 0;

SBUF = 0x2e;
while(~TI);
TI = 0;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        SBUF = half;
        while(~TI);
        TI = 0;

        /*-------------------------Send Thermal to LCD display-----------------*/
        lcd_text(temp_h);
        lcd_text(temp_l);

        lcd_jumporigin();
        lcd_command(0xC9);
        lcd_text(half);
        /*--------------------------------------------------------------------*/
    }
}
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Rabbit 3000 Microprocessor

Low EMI, High Performance

The new Rabbit 3000®. is a high-performance, low-EMI microprocessor designed specifically for embedded control, communications, and Ethernet connectivity. The 8-bit Rabbit 3000 outperforms most 16–bit processors without losing the efficiency of an 8–bit architecture. Extensive integrated features and glueless architecture facilitate rapid hardware design, while a C-friendly instruction set promotes efficient development of even the most complex applications.

The Rabbit 3000 is fast, running at up to 55.5 MHz, with compact code and direct software support for 1 MB of code/data space. Typically operating at 3.3 V (with 5 V tolerant I/O), the Rabbit 3000 boasts 6 serial ports with IrDA, 56+ digital I/O, quadrature decoder, PWM outputs, and pulse capture and measurement capabilities. It also features a battery-backable real-time clock, glueless memory and I/O interfacing, and ultra-low power modes. 4 levels of interrupt priority allow fast response to real-time events. Its compact instruction set and high clock speeds give the Rabbit 3000 exceptionally fast math, logic, and I/O performance.

Rabbit 3000 Features

The Rabbit 3000 has several powerful design features that practically eliminate EMI problems (typically <10 dB µV/m @ 3 m), which is essential for OEMs that need to pass CE and regulatory RF emissions tests. The amplitude of EM radiation is reduced by up to 25 dB µV by the internal spectrum spreader, gated clocks to prevent unnecessary clocking of unused registers, and separate power pins for the processor core and I/O. An auxiliary I/O bus can be used by designers to enable separate buses for I/O and memory or to limit memory bus loading to reduce EMI and ground bounce problems when interfacing external peripherals to the processor. The auxiliary I/O bus accomplishes this by mirroring the Rabbit's data bus on Port A and uses Port B to provide the processor's 6 least significant address lines for interfacing with external peripherals.

The high-performance instruction set offers both greater efficiency and execution speed of compiler-generated C code. Instructions include numerous single-byte opcodes that execute in two clock cycles, 16-bit loads and stores, 16-bit logical and arithmetic operations, 16 x 16 multiply (executes in 12 clocks), long jumps and returns for accessing a full megabyte of memory, and one byte prefixes to turn memory access instructions into internal and external I/O instructions.

The Rabbit 3000 requires no external memory driver or interface-logic. Its 20-bit address bus, 8-bit data bus, 3 chip select lines, 2 output-enable lines, and 2 write-enable lines can be directly interfaced with up to 6 Flash/SRAM devices. Up to 1 MB of memory can be accessed directly via the Dynamic C development software, and up to 6 MB can be interfaced with additional software development. A built-in slave port allows the Rabbit 3000 to be used as master or slave in multi-processor systems, permitting separate tasks to be assigned to dedicated processors. An 8-line data port and 5 control signals simplify the exchange of data between devices. A remote cold boot enables startup and programming via a serial or the slave port.

The Rabbit 3000 features seven 8-bit parallel ports, yielding a total of 56 digital I/O. Six CMOS-compatible serial ports are available. All 6 are configurable as asynchronous (also as IrDA), while 4 are configurable as clocked serial (SPI) and 2 as SDLC/HDLC. The Rabbit 3000 also offers alternate I/O functions. Pulse capture and measurement-2 input capture channels each have a 16-bit counter, clocked by the output of an internal timer. These channels can be used for a variety of functions, such as pulse width measurement or serial baud-rate detection. 2 quadrature decoder channels each have 2 inputs, as well as an 8-bit up-down counter. Each channel provides a direct interface to optical encoder units. 4 independent PWM outputs, each based on a 1024 pulse frame, are driven by the output of a programmable internal timer. The PWM outputs can be filtered to create a 10-bit D/A converter or used to directly drive devices such as motors or solenoids.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Programming the Rabbit 3000

The Rabbit 3000 is programmed using the industry-proven Dynamic C® software development system-an integrated C compiler, editor, loader, and debugger created specifically for Rabbit-based systems. Developing software with Dynamic C is easy. Users can write, compile, and test both C and Assembly code without leaving the Dynamic C development environment, and no costly in-circuit emulators are required. Full TCP/IP stack with source code is provided royalty free in Dynamic C and with our Development Kits. TCP/IP support includes PPP and SNMP, socket-level TCP and UDP, FTP, TFTP, HTTP (w/ SSI and CGI), DHCP, SMTP, POP3, and PING.

Rabbit 3000 Specifications and Features		
Packaging	**128-pin LGFP**	**128-ball TFBGA**
Package Size	16 x 16 x 1.5 mm	10 x 10 x 1.2 mm
Operating Voltage	1.8 – 3.6 V DC (5 V tolerant)	
Operating Current	2 mA/MHz @ 3.3 V	
Operating Temp.	-55°C to +85°C	
Maximum Clock Speed	55.5 MHz	
Digital I/O	56+ (arranged in seven 8-bit ports)	
Serial Ports	6 CMOS-compatible	
Baud Rate	Clock speed/8 max asynchronous	
Address Bus	20-bit	
Data Bus	8-bit	
Timers	Ten 8-bit and one 10-bit with 2 match registers	
Real-Time Clock	Yes, battery backable	
RTC Oscillator Circuitry	External	
Watchdog Timer/Supervisor	Yes	
Clock Modes	1x, 2x, /2, /3, /4, /6, /8	
Power Down Modes	Sleepy (32 kHz) Ultra-Sleepy (16, 8, 2 kHz)	
Auxiliary I/O Bus	8 data, 6 address lines	

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1.1 Features and Specifications Rabbit 3000

- 128-pin LQFP package. Operating voltage 1.8 V to 3.6 V. Clock speed to 54+ MHz. All specifications are given for both industrial and commercial temperature and voltage ranges. Rabbit microprocessors are low-cost.
- Industrial specifications are for 3.3 V ±10% and a temperature range from -40°C to +85°C. Modified commercial specifications are for a voltage variation of 5% and a temperature range from -40°C to 70°C.
- 1-megabyte code-data space allows C programs with 50,000+ lines of code. The extended Z80-style instruction set is C-friendly, with short and fast opcodes for the most important C operations.
- Four levels of interrupt priority make a fast interrupt response practical for critical applications. The maximum time to the first instruction of an interrupt routine is about 0.5 μs at a clock speed of 50 MHz.
- Access to I/O devices is accomplished by using memory access instructions with an I/O prefix. Access to I/O devices is thus faster and easier compared to processors with a distinct and narrow I/O instruction set. As an option the auxiliary I/O bus can be enabled to use separate pins for address and data, allowing the I/O bus to have a greater physical extent with less EMI and less conflict with the requirements of the fast memory bus.(Further described below.)
- Hardware design is simple. Up to six static memory chips (such as RAM and flash memory) connect directly to the microprocessor with no glue logic. A memory-access time of 55 ns suffices to support up to a 30 MHz clock with no wait states; with a 30 ns memory-access time, a clock speed of up to 50 MHz is possible with no wait states. Most I/O devices may be connected without glue logic.
- The memory read cycle is two clocks long. The write cycle is 3 clocks long. A clean memory and I/O cycle completely avoid the possibility of bus fights. Peripheral I/O devices can usually be interfaced in a glueless fashion using the common /IORD and /IOWR strobes in addition to the user-configurable IO strobes on Parallel Port E. The Parallel Port E pins can be configured as I/O read, write, read/write, or chip select when they are used as I/O strobes.
- EMI reduction features reduce EMI levels by as much as 25 dB compared to other similar microprocessors. Separate power pins for the on-chip I/O buffers prevent high-frequency noise generated in the processor core from propagating to the signal output pins. A built-in clock spectrum spreader reduces electromagnetic interference and facilitates passing EMI tests to prove compliance with government regulatory requirements. As a consequence, the designer of a Rabbit-3000-based system can be assured of passing FCC or CE EMI tests as long as minimal design precautions are followed.
- The Rabbit may be cold-booted via a serial port or the parallel access slave port. This means that flash program memory may be soldered in unprogrammed, and can be reprogrammed at any time without any assumption of an existing program or BIOS. A Rabbit that is slaved to a master processor can operate entirely with volatile RAM, depending on the master for a cold program boot.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- There are 56 parallel I/O lines (shared with serial ports). Some I/O lines are timer synchronized, which permits precisely timed edges and pulses to be generated under combined hardware and software control. Pulse-width modulation outputs are implemented in addition to the timer-synchronization feature (see below).
- Four pulse width modulated (PWM) outputs are implemented by special hardware. The repetition frequency and the duty cycle can be varied over a wide range. The resolution of the duty cycle is 1 part in 1024.
- There are six serial ports. All six serial ports can operate asynchronously in a variety of commonly used operating modes. Four of the six ports (designated A, B, C, D) support clocked serial communications suitable for interfacing with "SPI" devices and various similar devices such as A/D converters and memories that use a clocked serial protocol. Two of the ports, E and F, support HDLC/SDLC synchronous communication. These ports have a 4-byte FIFO and can operate at a high data rate. Ports E and F also have a digital phase-locked loop for clock recovery, and support popular data-encoding methods. High data rates are supported by all six serial ports. The asynchronous ports also support the 9th bit network scheme as well as infrared transmission using the IRDA protocol. The IRDA protocol is also supported in SDLC format by the two ports that support SDLC.
- A slave port allows the Rabbit to be used as an intelligent peripheral device slaved to a master processor. The 8-bit slave port has six 8-bit registers, 3 for each direction of communication. Independent strobes and interrupts are used to control the slave port in both directions. Only a Rabbit and a RAM chip are needed to construct a complete slave system, if the clock and reset control are shared with the master processor
- There is an option to enable an auxiliary I/O bus that is separate from the memory bus. The auxiliary I/O bus toggles only on I/O instructions. It reduces EMI and speeds the operation of the memory bus, which only has to connect to memory chips when the auxiliary I/O bus is used to connect I/O devices. This important feature makes memory design easy and allows a more relaxed approach to interfacing I/O devices.
- The built-in battery-backable time/date clock uses an external 32.768 kHz crystal oscillator. The suggested model circuit for the external oscillator utilizes a single "tiny logic" active component. The time/date clock can be used to provide periodic interrupts every 488 μs. Typical battery current consumption is about 3 μA.
- Numerous timers and counters can be used to generate interrupts, baud rate clocks, and timing for pulse generation.
- Two input-capture channels can be used to measure the width of pulses or to record the times at which a series of events take place. Each capture channel has a 16-bit counter and can take input from one or two pins selected from any of 16 pins.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- Two quadrature decoder units accept input from incremental optical shaft encoders. These units can be used to track the motion of a rotating shaft or similar device.
- A built-in clock doubler allows ½-frequency crystals to be used.
- The built-in main clock oscillator uses an external crystal or a ceramic resonator. Typical crystal or resonator frequencies are in the range of 1.8 MHz to 30 MHz. Since precision timing is available from the separate 32.768 kHz oscillator, a low-cost ceramic resonator with ½ percent error is generally satisfactory. The clock can be doubled or divided down to modify speed and power dynamically. The I/O clock, which clocks the serial ports, is divided separately so as not to affect baud rates and timers when the processor clock is divided or multiplied. For ultra low power operation, the processor clock can be driven from the separate 32.768 kHz oscillator and the main oscillator can be powered down. This allows the processor to operate at approximately between 20 and 100 μA and still execute instructions at the rate of up to 10,000 instructions per second. The 32.768 kHz clock can also be divided by 2, 4, 8 or 16 to reduce power. This "sleepy mode" is a powerful alternative to sleep modes of operation used by other processors.
- Processor current requirement is approximately 65 mA at 30 MHz and 3.3 V. The current is proportional to voltage and clock speed--at 1.8 V and 3.84 MHz the current would be about 5 mA, and at 1 MHz the current is reduced to about 1 mA.
- To allow extreme low power operation there are options to reduce the duty cycle of memories when running at low clock speeds by only enabling the chip select for a brief period, long enough to complete a read. This greatly reduces the power used by flash memory when operating at low clock speeds.
- The excellent floating-point performance is due to a tightly coded library and powerful processing capability. For example, a 50 MHz clock takes 7 μs for a floating add, 7 μs for a multiply, and 20 μs for a square root. In comparison, a 386EX processor running with an 8-bit bus at 25 MHz and using Borland C is about 20 times slower.
- There is a built-in watchdog timer.
- The standard 10-pin programming port eliminates the need for in-circuit emulators. A very simple 10-pin connector can be used to download and debug software using Z-World's Dynamic C and a simple connection to a PC serial port. The incremental cost of the programming port is extremely small.

2.1 The Rabbit 8-bit Processor vs. Other Processors

The Rabbit 3000 processor has been designed with the objective of creating practical systems to solve real world problems in an economical fashion. A cursory comparison of the Rabbit 3000 compared to other processors with similar capabilities may miss certain Rabbit strong points.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- The Rabbit is a processor that can be used to build a system in which EMI is nearly absent, even at clock frequencies in excess of 40 MHz. This is due to the split power supply, the clock doubler, the clock spectrum spreader and the PC board layout advice (or processor core modules) that we provide. Low EMI is a huge timesaver for the designer pressed to meet schedules and pass government EMI tests of the final product.
- Execution speed with the Rabbit is usually a pleasant surprise compared to other processors. This is due to the well-chosen and compact instruction set partnered with and excellent compiler and library. We have many benchmarks, comparing the Rabbit to 186, 386, 8051, Z180 and ez80 families of processors that prove the point.
- The Rabbit memory bus is an exceptionally efficient and very clean design. No external logic is required to support static memory chips. Battery-backed external memory is supported by built-in functionality. During reduced-power slow-clock operation the memory duty cycle can be correspondingly reduced using built-in hardware, resulting in low power consumption by the memories.
- The Rabbit external bus uses 2 clocks for read cycles and 3 clocks for write cycles. This has many advantages compared to a single-clock design, and on closer examination the advantages of the single-clock system turn out to be mostly chimerical. The advantages include: easy design to avoid bus fights, clean write cycles with solid data and address hold times, flexibility to have memory output enable access times greater than ½ of the bus cycle, and the ability to use an asymmetric clock generated by a clock doubler. The supposed advantage that single-clock systems have of double-speed bus operation is not possible with real-world memories unless the memory is backed with fast-cache RAM.
- The Rabbit 3000 operates at 3.6 V or less, but it has 5 V tolerant inputs and has a second complete bus for I/O operations that is separate from the memory bus. This second auxiliary bus can be enabled by the application as a designer option. These features make it easy to design systems that mix 3 V and 5 V components, and avoid the loading problems and the EMI problems that result if the memory bus is extended to connect with many I/O devices.
- The Rabbit may be remotely programmed, including complete cold-boot, via a serial link, Ethernet, or even via a network or the Internet using built in capabilities and/or the RabbitLink ethernet network accessory device. These capabilities proven and inexpensive to implement.
- The Rabbit 3000 on-chip peripheral complement is huge compared to competitive processors.

The Rabbit is an 8-bit processor with an 8-bit external data bus and an 8-bit internal data bus. Because the Rabbit makes the most of its external 8-bit bus and because it has a compact instruction set, its performance is as good as many 16-bit processors.

We hesitate to compare the Rabbit to 32-bit processors, but there are undoubtedly occasions where the user can use a Rabbit instead of a 32-bit processor and save a

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

vast amount of money. Many Rabbit instructions are 1 byte long. In contrast, the minimum instruction length on most 32-bit RISC processors is 32 bits.

2.2 Overview of On-Chip Peripherals and Features

The on-chip peripherals were chosen based on our experience as to what types of peripheral devices are most useful in small embedded systems. The major on-chip peripherals are the serial ports, system clock, time/date oscillator, parallel I/O, slave port, motion encoders, pulse width modulators, pulse measurement, and timers. These and other features are described below.

1.2 Summary of Rabbit 3000 Advantages

- The glueless architecture makes it is easy to design the hardware system.
- There are a lot of serial ports and they can communicate very fast.
- Precision pulse and edge generation is a standard feature.
- EMI is at extremely low levels.
- Interrupts can have multiple priorities.
- Processor speed and power consumption are under program control.
- The ultra low power mode can perform computations and execute logical tests since the processor continues to execute, albeit at 32 kHz or even as slow as 2 kHz.
- The Rabbit may be used to create an intelligent peripheral or a slave processor. For example, protocol stacks can be off loaded to a Rabbit slave. The master can be any processor.
- The Rabbit can be cold-booted so unprogrammed flash memory can be soldered in place.
- You can write serious software, be it 1,000 or 50,000 lines of C code. The tools are there and they are low in cost.
- If you know the Z80 or Z180, you know most of the Rabbit.
- A simple 10-pin programming interface replaces in-circuit emulators and PROM programmers.
- The battery-backable time/date clock is included.
- The standard Rabbit chip is made to industrial temperature and voltage specifications.
- The Rabbit 3000 is backed by extensive software development tools and libraries, especially in the area of networking and embedded Internet.

1.3 Differences Rabbit 3000 vs. Rabbit 2000

For the benefit of readers who are familiar with the Rabbit 2000 microprocessor the Rabbit 3000 is contrasted with the Rabbit 2000 in the table below.

Feature	Rabbit 3000	Rabbit 2000

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Maximum clock speed	54 MHz	30 MHz
Maximum crystal frequency main oscillator (may be doubled internally)	30 MHz	32 MHz
32.768 kHz crystal oscillator	External	Internal
Maximum operating voltage	3.6 V	5.5 V
Maximum I/O input voltage	5.5 V	5.5 V
Current consumption	2 mA/MHz @ 3.3 V	4 mA/MHz @5 V
Number of package pins	128	100
Size of package	16 × 16 × 1.5 mm LQFP 10 × 10 × 1.2 mm TFBGA	24 × 18 × 3 mm PQFP
Spacing between package pins	0.4 mm (16 mils) LQFP 0.8 mm TFBGA	0.65 mm (26 mils) PQFP
Separate power and ground for I/O buffers (EMI reduction)	Yes	No
Clock Spectrum Spreader (EMI reduction)	Yes	Rabbit 2000B and Rabbit 2000C versions.
Clock Modes	1x, 2x, /2, /3, /4, /6, /8	1x, 2x, /4, /8
Power Down Modes	Sleepy (32 kHz) Ultra-Sleepy (16, 8, 2 kHz)	Sleepy (32 kHz)
Low Power Memory Control (Chip Select)	Short CS (CLK /4 /6 /8) Self Timed (32,16,8,2 kHz)	None
Extended memory timing for high freq. operation	Yes	No
Number of 8-bit I/O ports	7	5
Auxiliary I/O Data/Address bus	Yes	None
Number of serial ports	6	4
Serial ports capable of SPI/clocked serial	4 (A, B, C, D)	2 (A, B)
Serial ports capable of SDLC/HDLC	2 (E, F)	None
Asynch serial ports with support for IrDA communications	6	None
Serial ports with support for SDLC/HDLC IrDA communications	2	None
Maximum asynchronous baud rate	clock speed/8	clock speed/32
Input capture unit	2	None

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Table B-10. RCM3720 Prototyping Board Use of Rabbit 3000 Parallel Ports

Port	I/O	Use		Initial State
PA0–PA7	Input	Not used		Pulled up (core module)
PB0	Output	CLKB, Serial Flash SCLK		High
PB1	Output	CLKA Programming Port		High (when not driven by CLKA)
PB2–PB6	Output	Not used		High
PB7	Input	External IA5, Switch S2		Pulled up (Proto Board)
PC0	Output	TXD RS-232	Serial Port D	High (set by drivers)
PC1	Input	RXD RS-232		Pulled up (core module)
PC2	Output	TXC RS-232	Serial Port C	High (set by drivers)
PC3	Input	RXC RS-232		Pulled up (core module)
PC4	Output	TXB Serial Flash	Serial Port B	High (set by drivers)
PC5	Input	RXB Serial Flash		Pulled up (core module)
PC6	Output	TXA Programming Port	Serial Port A	High (when not driven)
PC7	Input	RXA Programming Port		Pulled up (core module)
PD0	Output	Ethernet RSTDRV		Pulled up (core module)
PD1	Input	Ethernet BD5 (EESK)		Set by Ethernet
PD2	Input	Ethernet BD6 (EEDI)		Set by Ethernet
PD3	Input	Ethernet BD6 (EEDO)		Set by Ethernet
PD4–PD5	Output	Not used		High
PD6–PD7	Input	Not used		Pulled up (core module)
PE0–PE1	Output	Not used		High
PE2	Output	Ethernet AEN		High (driven by Ethernet)
PE3	Input	Not used		Pulled up (core module)
PE4–PE5	Output	Not used		High

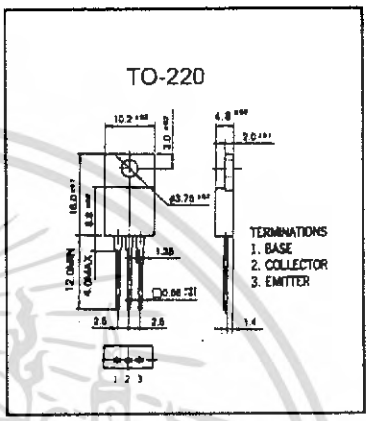
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

PE6	Input	Serial Flash Select	Pulled up (core module)
PE7	Output	Not used	High
PF0	Output	Not used	High
PF1	Input	Not used	Low
PF2–PF3	Input	Not used	Pulled up (core module)
PF4	Input	Switch S1	Pulled up (Proto Board)
PF5	Output	Not used	High
PF6	Output	LED DS1	High
PF7	Output	LED DS2	High
PG0–PG1	Output	Not used	High
PG2–PG3	Input	Not used	Pulled up (core module)
PG4–PG7	Output	Not used	High

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2SC1061 NPN EPITAXIAL SILICON TRANSISTOR

LOW FREQUENCY POWER AMPLIFIER



- Complement to 2SA671

ABSOLUTE MAXIMUM RATINGS (T_A=25℃)

Characteristic	Symbol	Rating	Unit
Collector-Base Voltage	V_{CBO}	50	V
Collector-Emitter Voltage	V_{CEO}	50	V
Emitter-Base voltage	V_{EBO}	7	V
Collector Current (DC)	I_C	3	A
Collector Dissipation (Tc=25℃)	P_C	25	W
Junction Temperature	Tj	150	℃
Storage Temperature	Tstg	-50~150	℃

ELECTRICAL CHARACTERISTICS (T_A=25℃)

Characteristic	Symbol	Test Condition	Min	Typ	Max	Unit
Collector Cutoff Current	I_{CBO}	V_{CB}= 50V , I_E=0			100	μA
Emitter Cutoff Current	I_{EBO}	V_{EB}= 7V , I_C=0			100	μA
DC Current Gain	h_{FE1}	V_{CE}= 4V , I_C=1A	35		200	
Collector- Emitter Saturation Voltage	$V_{CE(sat)}$	I_C=3A , I_B=0.3A			1.0	V
Current Gain Bandwidth Product	f_T	V_{CE}= 5V , I_C=0.5A		8		MHz

Wing Shing Computer Components Co., (H.K.)Ltd.
Homepage: http://www.wingshing.com
Tel:(852)2341 9276 Fax:(852)2797 8153
E-mail: wsccltd@hkstar.com

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

PNP general purpose transistors BC556; BC557

FEATURES

- Low current (max. 100 mA)
- Low voltage (max. 65 V).

APPLICATIONS

- General purpose switching and amplification.

DESCRIPTION

PNP transistor in a TO-92; SOT54 plastic package.
NPN complements: BC546 and BC547.

PINNING

PIN	DESCRIPTION
1	emitter
2	base
3	collector

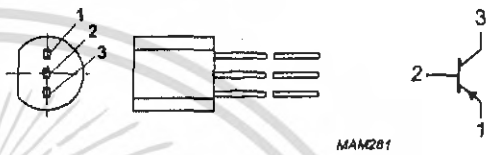


Fig.1 Simplified outline (TO-92; SOT54) and symbol.

LIMITING VALUES

In accordance with the Absolute Maximum Rating System (IEC 134).

SYMBOL	PARAMETER	CONDITIONS	MIN.	MAX.	UNIT
V_{CBO}	collector-base voltage	open emitter			
	BC556		–	–80	V
	BC557		–	–50	V
V_{CEO}	collector-emitter voltage	open base			
	BC556		–	–65	V
	BC557		–	–45	V
V_{EBO}	emitter-base voltage	open collector	–	–5	V
I_C	collector current (DC)		–	–100	mA
I_{CM}	peak collector current		–	–200	mA
I_{BM}	peak base current		–	–200	mA
P_{tot}	total power dissipation	$T_{amb} \leq 25$ °C	–	500	mW
T_{stg}	storage temperature		–65	+150	°C
T_j	junction temperature		–	150	°C
T_{amb}	operating ambient temperature		–65	+150	°C

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

THERMAL CHARACTERISTICS

SYMBOL	PARAMETER	CONDITIONS	VALUE	UNIT
$R_{th\ j\text{-}a}$	thermal resistance from junction to ambient	note 1	250	K/W

Note

1. Transistor mounted on an FR4 printed-circuit board.

CHARACTERISTICS

T_j = 25 °C unless otherwise specified.

SYMBOL	PARAMETER	CONDITIONS	MIN.	TYP.	MAX.	UNIT
I_{CBO}	collector cut-off current	I_E = 0; V_{CB} = –30 V	–	–1	–15	nA
		I_E = 0; V_{CB} = –30 V; T_j = 150 °C	–	–	–4	µA
I_{EBO}	emitter cut-off current	I_C = 0; V_{EB} = –5 V	–	–	–100	nA
h_{FE}	DC current gain	I_C = –2 mA; V_{CE} = –5 V; see Figs 2, 3 and 4				
	BC556		125	–	475	
	BC557		125	–	800	
	BC556A		125	–	250	
	BC556B; BC557B		220	–	475	
	BC557C		420	–	800	
V_{CEsat}	collector-emitter saturation voltage	I_C = –10 mA; I_B = –0.5 mA	–	–60	–300	mV
		I_C = –100 mA; I_B = –5 mA	–	–180	–650	mV
V_{BEsat}	base-emitter saturation voltage	I_C = –10 mA; I_B = –0.5 mA; note 1	–	–750	–	mV
		I_C = –100 mA; I_B = –5 mA; note 1	–	–930	–	mV
V_{BE}	base-emitter voltage	I_C = –2 mA; V_{CE} = –5 V; note 2	–600	–650	–750	mV
		I_C = –10 mA; V_{CE} = –5 V; note 2	–	–	–820	mV
C_c	collector capacitance	$I_E = i_e$ = 0; V_{CB} = –10 V; f = 1 MHz	–	3	–	pF
C_e	emitter capacitance	$I_C = i_c$ = 0; V_{EB} = –0.5 V; f = 1 MHz	–	10	–	pF
f_T	transition frequency	I_C = –10 mA; V_{CE} = –5 V; f = 100 MHz	100	–	–	MHz
F	noise figure	I_C = –200 µA; V_{CE} = –5 V; R_S = 2 kΩ; f = 1 kHz; B = 200 Hz	–	2	10	dB

Notes

1. V_{BEsat} decreases by about –1.7 mV/K with increasing temperature.
2. V_{BE} decreases by about –2 mV/K with increasing temperature.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

May 1997

National Semiconductor

LMC555
CMOS Timer

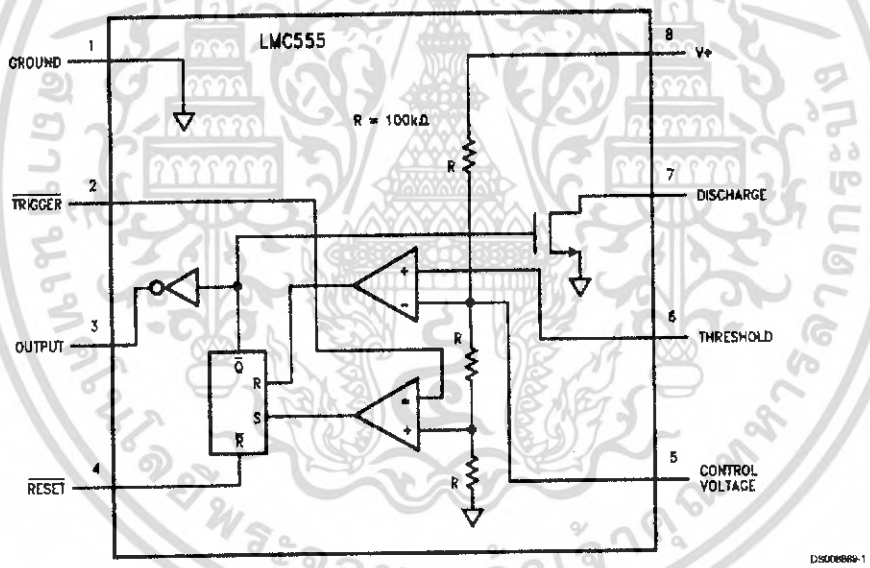
General Description

The LMC555 is a CMOS version of the industry standard 555 series general purpose timers. It offers the same capability of generating accurate time delays and frequencies but with much lower power dissipation and supply current spikes. When operated as a one-shot, the time delay is precisely controlled by a single external resistor and capacitor. In the astable mode the oscillation frequency and duty cycle are accurately set by two external resistors and one capacitor. The use of National Semiconductor's LMCMOS™ process extends both the frequency range and low supply capability.

Features

- Less than 1 mW typical power dissipation at 5V supply
- 3 MHz astable frequency capability
- 1.5V supply operating voltage guaranteed
- Output fully compatible with TTL and CMOS logic at 5V supply
- Tested to −10 mA, +50 mA output current levels
- Reduced supply current spikes during output transitions
- Extremely low reset, trigger, and threshold currents
- Excellent temperature stability
- Pin-for-pin compatible with 555 series of timers
- Available in 8 pin MSOP Package

Block and Connection Diagrams



(Pinouts for Molded and Metal Can Packages are identical)

Order Number LMC555CH, LMC555CM, LMC555CMM or LMC555CN
See NS Package Number H08C, M08A, MUA08A, or N08E

LMCMOS™ is a trademark of National Semiconductor Corp.

© 1997 National Semiconductor Corporation DS008669

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Absolute Maximum Ratings (Note 2)

If Military/Aerospace specified devices are required, please contact the National Semiconductor Sales Office/ Distributors for availability and specifications.

Supply Voltage, V8	15V
Input Voltages, V2, V4, V5, V6	−0.3V to V_S + 0.3V
Output Voltages, V3, V7	15V
Output Current I3, I7	100 mA
Operating Temperature Range (Note 4)	−40˚C to +85˚C
Storage Temperature Range	−65˚C to +150˚C

Soldering Information

Dual-In-Line Package	
Soldering (10 seconds)	260˚C
Small Outline Package	
(SOIC and MSOP)	
Vapor Phese (60 seconds)	215˚C
Infrared (15 seconds)	220˚C

See AN-450 "Surface Mounting Methods and Their Effect on Product Reliability" for other methods of soldering surface mount devices.

Electrical Characteristics (Notes 1, 2)

Test Circuit, T = 25˚C, all switches open, $\overline{\text{RESET}}$ to V_S unless otherwise noted

Symbol	Parameter	Conditions	Min	Typ	Max	Units (Limits)
I8	Supply Current	V_S = 1.5V		50	150	µA
		V_S = 5V		100	250	
		V_S = 12V		150	400	
V5	Control Voltage	V_S = 1.5V	0.8	1.0	1.2	V
		V_S = 5V	2.8	3.3	3.8	
		V_S = 12V	7.4	8.0	8.6	
V7	Discharge Saturation Voltage	V_S = 1.5V, I_7 = 1 mA		75	150	mV
		V_S = 5V, I_7 = 10 mA		150	300	
$V3_L$	Output Voltage (Low)	V_S = 1.5V, I_3 =1 mA		0.2	0.4	V
		V_S = 5V, I_3 = 8 mA		0.3	0.6	
		V_S = 12V, I_3 = 50 mA		1.0	2.0	
$V3_H$	Output Voltage (High)	V_S = 1.5V, I_3 =−0.25 mA	1.0	1.25		V
		V_S = 5V, I_3 = −2 mA	4.4	4.7		
		V_S = 12V, I_3 = −10 mA	10.5	11.3		
V2	Trigger Voltage	V_S = 1.5V	0.4	0.5	0.6	V
		V_S = 12V	3.7	4.0	4.3	
I2	Trigger Current	V_S = 5V		10		pA
V4	Reset Voltage	V_S = 1.5V (Note 5)	0.4	0.7	1.0	V
		V_S = 12V	0.4	0.75	1.1	
I4	Reset Current	V_S = 5V		10		pA
I6	Threshold Current	V_S = 5V		10		pA
I7	Discharge Leakage	V_S = 12V		1.0	100	nA
t	Timing Accuracy	SW 2, 4 Closed				ms
		V_S = 1.5V	0.9	1.1	1.25	
		V_S = 5V	1.0	1.1	1.20	
		V_S = 12V	1.0	1.1	1.25	
$\Delta t/\Delta V_S$	Timing Shift with Supply	V_S = 5V ±1V		0.3		%/V
$\Delta t/\Delta T$	Timing Shift with Temperature	V_S = 5V		75		ppm/˚C
		−40˚C ≤ T ≤ +85˚C				
f_A	Astable Frequency	SW 1, 3 Closed	4.0	4.8	5.6	kHz
		V_S = 12V				
f_{MAX}	Maximum Frequency	Max. Freq. Test		3.0		MHz
		Circuit, V_S = 5V				
t_R, t_F	Output Rise and Fall Times	Max. Freq. Test Circuit		15		ns
		V_S = 5V, C_L = 10 pF				

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

GlobalOptoisolator™

VDE | UL | CSA | SETI | SEMKO | DEMKO | NEMKO | BABT

6-Pin DIP Zero-Cross Optoisolators Triac Driver Output
(800 Volts Peak)

MOC3081
[IFT = 15 mA Max]
MOC3082
[IFT = 10 mA Max]
MOC3083*
[IFT = 5 mA Max]

*Motorola Preferred Device

The MOC3081, MOC3082 and MOC3083 devices consist of gallium arsenide infrared emitting diodes optically coupled to monolithic silicon detectors performing the function of Zero Voltage Crossing bilateral triac drivers.

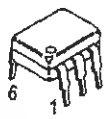
They are designed for use with a triac in the interface of logic systems to equipment powered from 240 Vac lines, such as solid–state relays, industrial controls, motors, solenoids and consumer appliances, etc.

- Simplifies Logic Control of 240 Vac Power
- Zero Voltage Crossing
- dv/dt of 1500 V/µs Typical, 600 V/µs Guaranteed
- ***To order devices that are tested and marked per VDE 0884 requirements, the suffix "V" must be included at end of part number. VDE 0884 is a test option.***

Recommended for 240 Vac(rms) Applications:

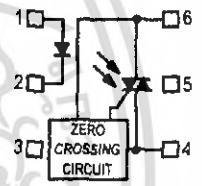
- Solenoid/Valve Controls
- Lighting Controls
- Static Power Switches
- AC Motor Drives
- Temperature Controls
- E.M. Contactors
- AC Motor Starters
- Solid State Relays

STYLE 6 PLASTIC



STANDARD THRU HOLE
CASE 730A–04

COUPLER SCHEMATIC



1. ANODE
2. CATHODE
3. NC
4. MAIN TERMINAL
5. SUBSTRATE
   DO NOT CONNECT
6. MAIN TERMINAL

MAXIMUM RATINGS

Rating	Symbol	Value	Unit
INPUT LED			
Reverse Voltage	V_R	6	Volts
Forward Current — Continuous	I_F	60	mA
Total Power Dissipation @ T_A = 25°C Negligible Power in Output Driver Derate above 25°C	P_D	120 1.41	mW mW/°C
OUTPUT DRIVER			
Off–State Output Terminal Voltage	V_{DRM}	800	Volts
Peak Repetitive Surge Current (PW = 100 µs, 120 pps)	I_{TSM}	1	A
Total Power Dissipation @ T_A = 25°C Derate above 25°C	P_D	150 1.76	mW mW/°C
TOTAL DEVICE			
Isolation Surge Voltage(1) (Peak ac Voltage, 60 Hz, 1 Second Duration)	V_{ISO}	7500	Vac(pk)
Total Power Dissipation @ T_A = 25°C Derate above 25°C	P_D	250 2.94	mW mW/°C
Junction Temperature Range	T_J	–40 to +100	°C
Ambient Operating Temperature Range(2)	T_A	–40 to +85	°C
Storage Temperature Range(2)	T_{stg}	–40 to +150	°C
Soldering Temperature (10 s)	T_L	260	°C

1. Isolation surge voltage, V_{ISO}, is an internal device dielectric breakdown rating.
   For this test, Pins 1 and 2 are common, and Pins 4, 5 and 6 are common.
2. Refer to Quality and Reliability Section in Opto Data Book for information on test conditions.

Preferred devices are Motorola recommended choices for future use and best overall value.

GlobalOptoisolator is a trademark of Motorola, Inc.

© Motorola, Inc. 1995

MOTOROLA

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

NPN general purpose transistors — BC546; BC547

FEATURES

- Low current (max. 100 mA)
- Low voltage (max. 65 V).

APPLICATIONS

- General purpose switching and amplification.

DESCRIPTION

NPN transistor in a TO-92; SOT54 plastic package.
PNP complements: BC556 and BC557.

PINNING

PIN	DESCRIPTION
1	emitter
2	base
3	collector

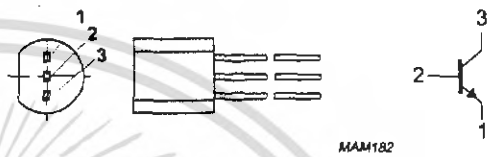


Fig.1 Simplified outline (TO-92; SOT54) and symbol.

LIMITING VALUES

In accordance with the Absolute Maximum Rating System (IEC 134).

SYMBOL	PARAMETER	CONDITIONS	MIN.	MAX.	UNIT
V_{CBO}	collector-base voltage	open emitter			
	BC546		–	80	V
	BC547		–	50	V
V_{CEO}	collector-emitter voltage	open base			
	BC546		–	65	V
	BC547		–	45	V
V_{EBO}	emitter-base voltage	open collector			
	BC546		–	6	V
	BC547		–	6	V
I_C	collector current (DC)		–	100	mA
I_{CM}	peak collector current		–	200	mA
I_{BM}	peak base current		–	200	mA
P_{tot}	total power dissipation	$T_{amb} \leq 25$ °C; note 1	–	500	mW
T_{stg}	storage temperature		–65	+150	°C
T_j	junction temperature		–	150	°C
T_{amb}	operating ambient temperature		–65	+150	°C

Note

1. Transistor mounted on an FR4 printed-circuit board.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

THERMAL CHARACTERISTICS

SYMBOL	PARAMETER	CONDITIONS	VALUE	UNIT
$R_{th\ j-a}$	thermal resistance from junction to ambient	note 1	0.25	K/mW

Note

1. Transistor mounted on an FR4 printed-circuit board.

CHARACTERISTICS

T_j = 25 °C unless otherwise specified.

SYMBOL	PARAMETER	CONDITIONS	MIN.	TYP.	MAX.	UNIT
I_{CBO}	collector cut-off current	I_E = 0; V_{CB} = 30 V	–	–	15	nA
		I_E = 0; V_{CB} = 30 V; T_j = 150 °C	–	–	5	µA
I_{EBO}	emitter cut-off current	I_C = 0; V_{EB} = 5 V	–	–	100	nA
h_{FE}	DC current gain BC546A BC546B; BC547B BC547C	I_C = 10 µA; V_{CE} = 5 V; see Figs 2, 3 and 4	 – – –	 90 150 270	 – – –	
	DC current gain BC546A BC546B; BC547B BC547C BC547 BC546	I_C = 2 mA; V_{CE} = 5 V; see Figs 2, 3 and 4	 110 200 420 110 110	 180 290 520 – –	 220 450 800 800 450	
V_{CEsat}	collector-emitter saturation voltage	I_C = 10 mA; I_B = 0.5 mA	–	90	250	mV
		I_C = 100 mA; I_B = 5 mA	–	200	600	mV
V_{BEsat}	base-emitter saturation voltage	I_C = 10 mA; I_B = 0.5 mA; note 1	–	700	–	mV
		I_C = 100 mA; I_B = 5 mA; note 1	–	900	–	mV
V_{BE}	base-emitter voltage	I_C = 2 mA; V_{CE} = 5 V; note 2	580	660	700	mV
		I_C = 10 mA; V_{CE} = 5 V	–	–	770	mV
C_c	collector capacitance	I_E = i_e = 0; V_{CB} = 10 V; f = 1 MHz	–	1.5	–	pF
C_e	emitter capacitance	I_C = i_c = 0; V_{EB} = 0.5 V; f = 1 MHz	–	11	–	pF
f_T	transition frequency	I_C = 10mA; V_{CE} = 5 V; f = 100 MHz	100	–	–	MHz
F	noise figure	I_C = 200 µA; V_{CE} = 5 V; R_S = 2 kΩ; f = 1 kHz; B = 200 Hz	–	2	10	dB

Notes

1. V_{BEsat} decreases by about 1.7 mV/K with increasing temperature.
2. V_{BE} decreases by about 2 mV/K with increasing temperature.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

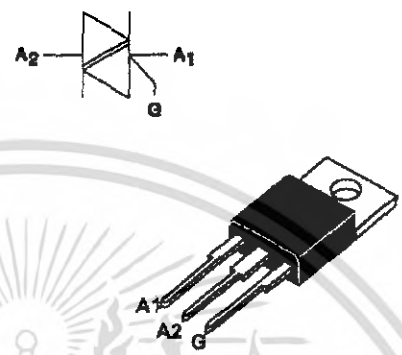
SGS-THOMSON MICROELECTRONICS

BTA10 B/C
BTB10 B/C

STANDARD TRIACS

FEATURES

- HIGH SURGE CURRENT CAPABILITY
- COMMUTATION : (dV/dt)c > 5 V/µs
- BTA Family :
  INSULATING VOLTAGE = 2500$V_{(RMS)}$
  (UL RECOGNIZED : E81734)



TO220AB
(Plastic)

DESCRIPTION

The BTA/BTB10 B/C triac family are high performance glass passivated PNPN devices.
These parts are suitables for general purpose applications where high surge current capability is required. Application such as phase control and static switching on inductive or resistive load.

ABSOLUTE RATINGS (limiting values)

Symbol	Parameter			Value	Unit
$I_{T(RMS)}$	RMS on-state current (360° conduction angle)	BTA	Tc = 90 °C	10	A
		BTB	Tc = 95 °C		
I_{TSM}	Non repetitive surge peak on-state current (Tj initial = 25°C)		tp = 8.3 ms	105	A
			tp = 10 ms	100	
I^2t	I^2t value		tp = 10 ms	50	A^2s
dI/dt	Critical rate of rise of on-state current Gate supply : I_G = 500mA di_G/dt = 1A/µs		Repetitive F = 50 Hz	10	A/µs
			Non Repetitive	50	
Tstg Tj	Storage and operating junction temperature range			- 40 to + 150 - 40 to + 125	°C °C
Tl	Maximum lead temperature for soldering during 10 s at 4.5 mm from case			260	°C

Symbol	Parameter	BTA / BTB10-... B/C				Unit
		400	600	700	800	
V_{DRM} V_{RRM}	Repetitive peak off-state voltage Tj = 125 °C	400	600	700	800	V

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

THERMAL RESISTANCES

Symbol	Parameter		Value	Unit
Rth (j-a)	Junction to ambient		60	°C/W
Rth (j-c) DC	Junction to case for DC	BTA	3.9	°C/W
		BTB	3.1	
Rth (j-c) AC	Junction to case for 360° conduction angle (F= 50 Hz)	BTA	2.9	°C/W
		BTB	2.3	

GATE CHARACTERISTICS (maximum values)

$P_{G\ (AV)}$ = 1W P_{GM} = 10W (tp = 20 µs) I_{GM} = 4A (tp = 20 µs) V_{GM} = 16V (tp = 20 µs).

ELECTRICAL CHARACTERISTICS

Symbol	Test Conditions		Quadrant		Suffix		Unit
					B	C	
I_{GT}	V_D=12V (DC) R_L=33Ω	Tj=25°C	I-II-III	MAX	50	25	mA
			IV	MAX	100	50	
V_{GT}	V_D=12V (DC) R_L=33Ω	Tj=25°C	I-II-III-IV	MAX	1.5		V
V_{GD}	V_D=V_{DRM} R_L=3.3kΩ	Tj=110°C	I-II-III-IV	MIN	0.2		V
tgt	V_D=V_{DRM} I_G = 500mA dI_G/dt = 3A/µs	Tj=25°C	I-II-III-IV	TYP	2		µs
I_L	I_G=1.2 I_{GT}	Tj=25°C	I-III-IV	TYP	40	20	mA
			II		70	35	
I_H *	I_T= 500mA gate open	Tj=25°C		MAX	50	25	mA
V_{TM} *	I_{TM}= 14A tp= 380µs	Tj=25°C		MAX	1.5		V
I_{DRM} I_{RRM}	V_{DRM} Rated V_{RRM} Rated	Tj=25°C		MAX	0.01		mA
		Tj=110°C		MAX	0.5		
dV/dt *	Linear slope up to V_D=67%V_{DRM} gate open	Tj=110°C		MIN	250	100	V/µs
(dV/dt)c *	(dI/dt)c = 4.4A/ms	Tj=110°C		MIN	10	5	V/µs

* For either polarity of electrode A2 voltage with reference to electrode A_1.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้