วิทยานิพนธ์

การควบคุมและตรวจสอบอุปกรณ์ไฟฟ้าผ่านทางระบบอินเตอร์เน็ต

CONTROLLING AND CHECKING ELECTRICAL EQUIPMENTS VIA INTERNET SYSTEM

นายธนกร ดุจเพ็ญ

บัณฑิตวิทยาลัย มหาวิทยาลัยเกษตรศาสตร์

พ.ศ. 2551

ใบรับรองวิทยานิพนธ์

บัณฑิตวิทยาลัย มหาวิทยาลัยเกษตรศาสตร์

วิศวกรรมศาสตรมหาบัณฑิต (วิศวกรรมไฟฟ้า)

ปริญญา

วิศวกรรมไฟฟ้า	วิศวกรรมไฟฟ้า
สาขา	**ภาควิชา**

เรื่อง การควบคุมและตรวจสอบอุปกรณ์ไฟฟ้าผ่านทางระบบอินเตอร์เน็ต

Controlling and Checking Electrical Equipments via Internet System

นามผู้วิจัย นายธนกร ดุจเพ็ญ

ได้พิจารณาเห็นชอบโดย

อาจารย์ที่ปรึกษาวิทยานิพนธ์หลัก .....

(รองศาสตราจารย์ชัยวัฒน์ ชัยกุล, M.A.)

อาจารย์ที่ปรึกษาวิทยานิพนธ์ร่วม .....

(ผู้ช่วยศาสตราจารย์ดุลย์พิเชษฐ์ ฤกษ์ปรีดาพงษ์, Ph.D.)

อาจารย์ที่ปรึกษาวิทยานิพนธ์ร่วม .....

(อาจารย์ดุสิต ธนเพทาย, Ph.D.)

หัวหน้าภาควิชา .....

(รองศาสตราจารย์มงคล รักษาพัชรวงศ์, M.A.)

บัณฑิตวิทยาลัย มหาวิทยาลัยเกษตรศาสตร์รับรองแล้ว

.....

(รองศาสตราจารย์กัญจนา ธีระกุล, D.Agr.)

คณบดีบัณฑิตวิทยาลัย

วันที่ ..... **เดือน** ..... **พ.ศ.** .....

วิทยานิพนธ์

เรื่อง

การควบคุมและตรวจสอบอุปกรณ์ไฟฟ้าผ่านทางระบบอินเตอร์เน็ต

Controlling and Checking Electrical Equipments via Internet System

โดย

นายธนกร ดุจเพ็ญ

เสนอ

บัณฑิตวิทยาลัย มหาวิทยาลัยเกษตรศาสตร์
เพื่อความสมบูรณ์แห่งปริญญาวิศวกรรมศาสตรมหาบัณฑิต (วิศวกรรมไฟฟ้า)
พ.ศ. 2551

ธนกร ดุจเพ็ญ 2551: การควบคุมและตรวจสอบอุปกรณ์ไฟฟ้าผ่านทางระบบอินเตอร์เน็ต ปริญญาวิศวกรรมศาสตรมหาบัณฑิต (วิศวกรรมไฟฟ้า) สาขาวิศวกรรมไฟฟ้า ภาควิชาวิศวกรรมไฟฟ้า อาจารย์ที่ปรึกษาวิทยานิพนธ์หลัก: รองศาสตราจารย์ชัยวัฒน์ ชัยกุล, M.A. 97 หน้า

วิทยานิพนธ์ฉบับนี้เสนอการวิจัยการควบคุมและตรวจสอบอุปกรณ์ไฟฟ้าผ่านทางระบบอินเตอร์เน็ตนำมาประยุกต์ใช้งานกับชีวิตประจำวัน ในงานวิจัยนี้ได้ทำการควบคุมเปิด/ปิดหลอดไฟฟ้าและควบคุมเปิด/ปิดเครื่องปรับอากาศและตรวจสอบอุณหภูมิห้องโดยใช้ไอซีเบอร์DS1820 เป็นชุดตรวจสอบอุณหภูมิและใช้ไมโครคอนโทรลเลอร์ PIC16F877 เป็นชุดควบคุมและยังได้มีการนำเทคนิคการดึงค่าของกระแสและแรงดัน โดยใช้ Current Transformer (CT) มาใช้ในการตรวจสอบผลของการสั่งงานโดยใช้อินเตอร์เน็ตเป็นตัวแสดงสถานะและสั่งงานไปควบคุมอุปกรณ์ไฟฟ้าโดยใช้โปรแกรมวิชวลเบสิคเป็นตัวสื่อสารกับไมโครคอนโทรลเลอร์

ดังนั้น การสั่งงานอุปกรณ์ไฟฟ้าผ่านทางระบบอินเตอร์เน็ตสามารถตรวจสอบและควบคุมการใช้พลังงานไฟฟ้าซึ่งแสดงสถานะ ON-OFF เป็นอนาล็อกที่เวปไซด์ได้และได้ผลที่เป็นไปตามที่ออกแบบไว้

ลายมือชื่อนิสิต ลายมือชื่ออาจารย์ที่ปรึกษาวิทยานิพนธ์หลัก

__ / __ / __

Thanakorn Dujpen 2008: Controlling and Checking Electrical Equipments via Internet System. Master of Engineering (Electrical Engineering), Major Field: Electrical Engineering, Department of Electrical Engineering. Thesis Advisor: Associate Professor Chaiwat Chaikul, M.A. 97 pages.

This thesis presents a research on controlling and checking electrical appliances via the Internet that can be applied in daily life. Here we control the switches of a fluorescent lamp and an air conditioner and monitor room temperature by making use of IC DS1820 in a temperature monitor kit and PIC16F877 microcontroller in a controller kit. We also utilize voltage and current sinking techniques through current transformer (CT) in order to verify the operation. The internet serves in status indication and controlling of the appliances. An application programmed in Visual Basic communicates with the microcontroller.

Operating electrical appliances via the internet can be monitored and their electrical energy consumption can be controlled as the ON/OFF analog status is displayed on the website. All works as expected.

_____ _____ ___ / ___ / ___

Student's signature Thesis Advisor's signature

กิตติกรรมประกาศ

ข้าพเจ้าขอกราบขอบพระคุณ รองศาสตราจารย์ชัยวัฒน์ ชัยกุล ประธานกรรมการที่ปรึกษา, ผู้ช่วยศาสตราจารย์ ดร.ดุลพิเชษฐ์ ฤกษ์ปรีดาพงษ์ และ ดร.ดุสิต ธนเพทาย ที่ได้ให้ความช่วยเหลือในการวางแผนงานวิจัยในวิทยานิพนธ์ฉบับนี้ตลอดจนการให้คำปรึกษาแนะนำและตรวจสอบแก้ไขข้อบกพร่องในวิทยานิพนธ์ให้สำเร็จลุล่วงไปด้วยดี

ข้าพเจ้าขอกราบขอบพระคุณ คุณพ่อ คุณแม่ ที่คอยให้ความรักและดูแลเอาใจใส่ ขอบคุณเพื่อนที่คอยให้กำลังใจ ขอบคุณพี่น้องในห้อง LAB ทุกคน และขอขอบคุณ คุณอาสุบรรณ บุตรสิงห์ ที่มีส่วนสำคัญในการช่วยแนะนำและแก้ไขในการทำวิทยานิพนธ์ฉบับนี้ให้สำเร็จลุล่วงไปด้วยดี

ธนกร ดุจเพ็ญ

พฤษภาคม 2551

สารบัญ

	หน้า
สารบัญ	(1)
สารบัญตาราง	(2)
สารบัญภาพ	(3)
คำนำ	1
วัตถุประสงค์	2
การตรวจเอกสาร	3
อุปกรณ์และวิธีการ	41
อุปกรณ์	41
วิธีการ	41
ผลและวิจารณ์	60
ผล	60
วิจารณ์	61
สรุปและข้อเสนอแนะ	62
สรุป	62
ข้อเสนอแนะ	62
เอกสารและสิ่งอ้างอิง	63
ภาคผนวก	65
ประวัติการศึกษา และการทำงาน	87

สารบัญตาราง

ตารางที่		หน้า
1	คำสั่ง AT Command สำหรับการ Dial-Up	31
2	แสดงผลการทดลองสั่งงานอุปกรณ์ไฟฟ้า ก	60
3	แสดงผลการทดลองสั่งงานอุปกรณ์ไฟฟ้า ข	61

สารบัญภาพ

ภาพที่		หน้า
1	แสดงหลักการทำงานของระบบควบคุม	3
2	แสดงวิธีการดำเนินงานวิจัย	6
3	แสดงทิศทางการเชื่อมต่อกับ Computer แบบ RS232	14
4	ชุดเชื่อมต่อคอมพิวเตอร์กับวงจรแปลงสัญญาณตามมาตรฐาน RS232	16
5	แสดงทิศทางการเชื่อมต่อกับ Computer แบบ RS485	17
6	ข้อมูลที่เป็นสัญญาณอัตราเปรียบเทียบกับความยาวสายเคเบิลโดยใช้24AWGเป็นสายเคเบิลคู่	19
7	แสดงการเปรียบเทียบความถี่สำหรับข้อมูลในแต่ละสายเคเบิล	20
8	แสดงเบอร์ไอซีของภาคตรวจวัดอุณหภูมิ	21
9	แผนผังของระบบบัสแบบ 1- Wire Bus	24
10	จังหวะเวลาในการทำกระบวนการตรวจสอบว่ามีอุปกรณ์อยู่บนบัส	26
11	จังหวะเวลาที่ไมโครคอนโทรลเลอร์ใช้เขียนข้อมูล ‘0’ หรือ ’1’ไปยังอุปกรณ์บนบัส	
12	จังหวะเวลาที่ไมโครคอนโทรลเลอร์ใช้อ่านบิตข้อมูล ‘0’ หรือ ’1’ จากอุปกรณ์บนบัส	27
13	แสดง Asynchronous serial interface ระหว่าง PC กับ PIC16F877 ส่วน serial interface	29
14	แสดงขั้นตอนการเขียนโปรแกรมเพื่อติดต่อกับโมเด็ม	32
15	แสดงหลักการทำงานของระบบ	42
16	แสดงหน้าที่ของขาควบคุมไมโครคอนโทรเลอร์	44
17	แสดงการกำหนดขาของภาคตรวจวัดอุณหภูมิ	47
18	แสดง Block diagram and in-out ของ DS1621	47
19	กำหนดแนวทางการวิจัย	49
20	แสดงวงจรควบคุมที่ออกแบบ	50
21	แสดงการต่ออุปกรณ์ลงแผ่นวงจรควบคุม(ก,ข)	51
22	แสดงวงจรชุอินพุต	52

สารบัญภาพ (ต่อ)

ภาพที่		หน้า
23	แสดงเมื่อทำการสั่ง RUN ในโปรแกรม Visual Basic	55
24	แสดงหน้าเวปไซด์ขณะอุปกรณ์ไฟฟ้ามีการทำงาน	58
25	แสดงหน้าเวปไซด์ขณะสั่งหยุดการทำงานของอุปกรณ์ไฟฟ้า	58
26	แสดงแบบจำลองโรงงานและอุปกรณ์ไฟฟ้าที่ควบคุม	59

การควบคุมและตรวจสอบอุปกรณ์ไฟฟ้าผ่านทางระบบอินเตอร์เน็ต

Controlling and Checking Electrical Equipments via Internet System

คำนำ

ปัจจุบันพลังงานไฟฟ้าถือว่ามีประโยชน์อย่างมหาศาลทั้งในส่วนของชีวิตประจำวันทั้งในบ้านพักอาศัยหรือโรงงานอุตสาหกรรมต่างก็ต้องการใช้ไฟฟ้า เพื่อให้เพียงพอต่อความต้องการการใช้พลังงานไฟฟ้าและผลจากการใช้พลังงานไฟฟ้าจำนวนมากนี้จึงทำให้เกิดการสิ้นเปลืองพลังงานไฟฟ้าจำนวนมากจากการใช้ไฟฟ้าโดยไม่จำเป็น เช่น การใช้ไฟฟ้าเพื่อปฏิบัติงานเมื่อเสร็จงานแล้วลืมปิด เป็นต้น ซึ่งเหตุการณ์เหล่านี้ทำให้เกิดการสิ้นเปลืองพลังงานไฟฟ้าและสิ้นเปลืองทรัพยากรในการผลิตพลังงานไฟฟ้า

การแสวงหาวิธีในการประหยัดพลังงานไฟฟ้าจึงเข้ามามีบทบาทสำคัญต่อการใช้ไฟฟ้าในปัจจุบันซึ่งตามโรงงานอุตสาหกรรมต่างประสบปัญหาในการใช้งานพลังงานไฟฟ้าที่ไม่จำเป็น ดังนั้นการสั่งงานการควบคุมการใช้ไฟฟ้าผ่านทางอินเตอร์เน็ตจึงเป็นอีกวิธีหนึ่งที่สะดวกและง่ายต่อการสั่งงานปัจจุบันการสื่อสารผ่านระบบเครือข่าย(Network)รวมทั้งระบบอินเตอร์เน็ตเป็นที่แพร่หลายมากในการใช้งาน, ใช้บริการ, อำนวยความสะดวก ในชีวิตประจำวันต่างๆ

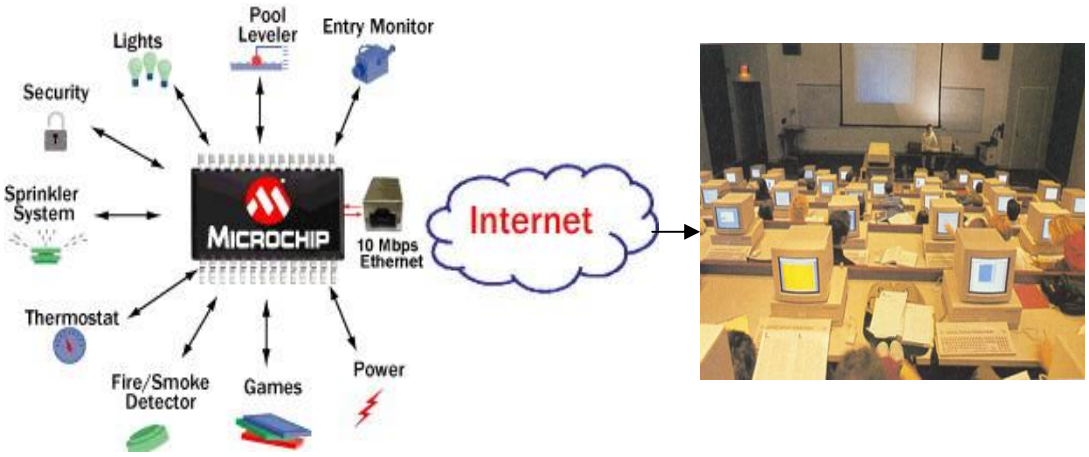
งานวิจัยนี้ได้ศึกษาถึงวิธีการควบคุมการใช้พลังงานไฟฟ้าโดยใช้ระบบอินเตอร์เน็ตและออกแบบอุปกรณ์เชื่อมต่อและพัฒนาโปรแกรมต้นแบบเพื่อควบคุมการสั่งงานและตรวจสอบอุปกรณ์ไฟฟ้าในโรงงานอุตสาหกรรม

วัตถุประสงค์

1. เพื่อควบคุมการใช้งานอุปกรณ์ไฟฟ้าภายในโรงงานด้วยระบบอินเตอร์เน็ต
2. เพื่อศึกษาการสร้างแบบจำลองวงจรควบคุมระบบจัดการอุปกรณ์ไฟฟ้าของโรงงาน
3. ออกแบบวิธีการควบคุมวงจรอิเล็กทรอนิกส์ที่ใช้ควบคุมระบบจัดการอุปกรณ์ไฟฟ้า
4. ออกแบบและสร้างวงจรควบคุมต้นแบบเพื่อยืนยันการใช้งานได้จริง
5. ออกแบบเวปไซด์เพื่อสั่งการและควบคุมอุปกรณ์ไฟฟ้า

การตรวจเอกสาร

การควบคุมอุปกรณ์ไฟฟ้าโดยการสั่งงานผ่านเวปไซต์โดยเราจะควบคุมการทำงานอุปกรณ์ไฟฟ้าผ่านทางอินเตอร์เน็ตและสามารถตรวจสอบผลการสั่งงาน



ภาพที่ 1 แสดงหลักการทำงานของระบบควบคุม

จากภาพที่ 1 แสดงความสามารถในการควบคุมอุปกรณ์ไฟฟ้าผ่านอินเตอร์เน็ตซึ่งสามารถควบคุมอุปกรไฟฟ้าได้หลายชนิด ในงานวิจัยนี้ได้ทำการควบคุมหลอดไฟฟ้า, เครื่องปรับอากาศ และสามารถตรวจสอบอุณหภูมิห้อง ณ ขณะนั้นได้

ความเป็นมาและความสำคัญของปัญหา

1. การเปิดอุปกรณ์ไฟฟ้าที่ไม่ได้ใช้งานแล้วทิ้งไว้ทำให้เกิดการสิ้นเปลืองพลังงานไฟฟ้า
2. พนักงานลืมปิดการใช้งานอุปกรณ์ไฟฟ้า
3. เจ้าของกิจการไม่สามารถตรวจสอบการใช้งานอุปกรณ์ไฟฟ้าได้ทั่วถึง
4. ความไม่สะดวกในการตรวจสอบอุปกรณ์ไฟฟ้าทำให้สิ้นเปลืองเวลาในการตรวจสอบ
5. เมื่อสั่งงานไปแล้วไม่สามารถรู้ได้ว่ามีการทำงานของอุปกรณ์ไฟฟ้าจริงหรือไม่
6. ควบคุมอุปกรณ์ไฟฟ้าได้จำนวนน้อย

ดังนั้นจึงได้ทำการศึกษาและหาวิธีควบคุมและสามารถตรวจสอบอุปกรณ์ไฟฟ้าให้มีประสิทธิภาพมากยิ่งขึ้น

งานวิจัยที่เกี่ยวข้อง

ในปัจจุบันคอมพิวเตอร์ได้เข้ามามีบทบาทเป็นอย่างมาก ไม่ว่าจะเป็นงานทางด้านวิศวกรรม การสื่อสาร ภาคธุรกิจ หรือแม้กระทั่งในระบบรักษาความปลอดภัย คณะผู้จัดทำจึงได้นำเอาความสามารถของคอมพิวเตอร์มาควบคุมอุปกรณ์ไฟฟ้าภายในบ้าน ในขณะที่ไม่มีคนอยู่ในบ้าน แต่เจ้าของบ้านยังสามารถตรวจสอบความเรียบร้อยของประตูหน้าต่าง และเห็นภาพภายในบ้านได้ โดยการติดต่อสื่อสารผ่านโมเด็ม และโทรศัพท์บ้านควบคู่กัน ดังนั้น ในการจัดทำโครงงานปริญญานิพนธ์ในครั้งนี้ จึงเป็นการศึกษาการใช้คอมพิวเตอร์ควบคุมอุปกรณ์ภายนอก โดยใช้โปรแกรม Visual Basic 6 สั่งงานผ่านทางพอร์ตขนานของคอมพิวเตอร์และการติดต่อสื่อสารโดยโมเด็ม การส่งแฟ้มข้อมูล, แฟ้มภาพ ผ่านทางโมเด็ม การควบคุมสเต็ปมอเตอร์ และการ Capture ภาพแล้วบันทึกลงในแฟ้มภาพ ผลการดำเนินงานโครงงานปริญญานิพนธ์ในครั้งนี้ สามารถใช้คอมพิวเตอร์ควบคุมอุปกรณ์ไฟฟ้าภายในบ้าน ตรวจสอบสถานะประตูหน้าต่าง และดูภาพในบ้านผ่านทางโมเด็มและได้สร้างโมเด็ลบ้าน เพื่อจำลองการทำงานของคอมพิวเตอร์ได้ระบบการควบคุมและแสดงสถานะอุปกรณ์ไฟฟ้าพร้อมแสดงภาพด้วยกล้อง CCDโดยผ่านทางโมเด็ม (ภาณุวิชญ์ กิจเจริญ และวศิน ศิริวาลย์, 2546)

ระบบควบคุมเครื่องใช้ไฟฟ้าผ่านทางโครงข่ายอินเตอร์เน็ต โดยผู้ใช้สามารถป้อนคำสั่งผ่านทางหน้าเว็บเซิร์ฟเวอร์ที่เชื่อมกับกับอินเตอร์เน็ตความเร็วสูง (ADSL) ไปยัง ADSL Router ซึ่งทำหน้าที่เชื่อมต่อสัญญาณอินเตอร์เน็ต และจะส่งข้อมูลไปยังระบบสมองกลฝังตัวโดยมีไมโครคอนโทรเลอร์ เป็นตัวควบคุมการทำงานของเครื่องใช้ไฟฟ้า จะทำให้เราสามารถควบคุมเครื่องใช้ไฟฟ้าจากระยะไกลได้ (นายธีรสักก์ และ คณะ, 2548)

โครงการนี้เป็นการศึกษาและพัฒนาเครื่องควบคุมอุปกรณ์ไฟฟ้าผ่านเครือข่ายอินเตอร์เน็ต โดยการประยุกต์ใช้งานไมโครคอนโทรเลอร์ที่มีชื่อว่า แรบบิท ซึ่งเป็นไมโครคอนโทรเลอร์ ที่สามารถเป็นเซิร์ฟเวอร์ได้ด้วยตัวเอง และมีการแจ้งสถานะ การทำงานให้ผู้ใช้ได้รับทราบจึงทำให้สามารถตรวจสอบสถานะของอุปกรณ์ได้ (นายอภิรักษ์ และ นางสาวอัญชลี, 2548)อุปกรณ์สื่อสาร อย่างเช่น โทรศัพท์มือถือ หรือ พีดีเอโฟนได้เข้ามามีบทบาทในชีวิตประจำวันมากขึ้น เนื่องจากตัวอุปกรณ์นั้นมีความสามารถในการใช้งานได้หลากหลาย อย่างเช่น ส่ง เอสเอ็มเอสจีพีอาเอส ส่งบลู

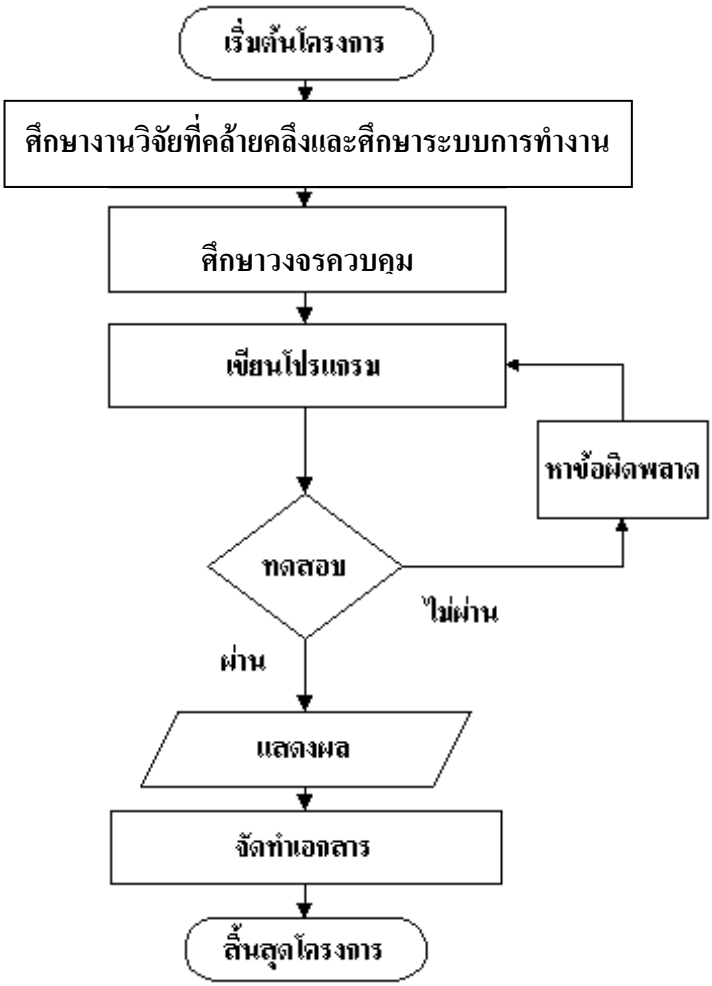
ทูธ ฯ และหนึ่งในความสามารถที่เราสนใจจะนำมาประยุกต์ใช้งาน ให้อุปกรณ์เหล่านี้กลายเป็น รีโมทคอนโทรล เป็นการส่งผ่านข้อมูลแบบไร้สาย สามารถนำไปควบคุมเครื่องใช้ไฟฟ้า เช่น ควบคุมประตูควบคุมโคมไฟ รวมไปถึงอุปกรณ์ไฟฟ้าทุกชนิด โดยการเขียนโปรแกรมควบคุมอุปกรณ์ไฟฟ้า ไว้บนตัวอุปกรณ์สื่อสาร แล้วใช้อุปกรณ์สื่อสารนั้นส่งสัญญาณควบคุมผ่านช่องทางของบลูทูธ จากนั้นใช้ไมโครคอนโทรลเลอร์ ซึ่งได้มีการติดตั้งอุปกรณ์รับสัญญาณบลูทูธเอา มารับสัญญาณควบคุม แล้วนำไปสั่งงานอุปกรณ์ไฟฟ้าให้ทำงานตามคำสั่งได้โครงงานนี้สามารถนำไปใช้งานได้อย่างกว้างขวาง เนื่องจากโทรศัพท์มือถือ และ พีดีเอโฟน จะมีการติดตั้งบลูทูธเอาไว้แทบทุกรุ่น นับว่าเป็นความสะดวกสบายอีกรูปแบบหนึ่ง (นายยมนา และ คณะ, 2548)

ขอบเขตของงานวิจัย

1. สามารถสร้างแบบจำลองการควบคุมอุปกรณ์ไฟฟ้าโดยเราควบคุมการสั่งเปิดปิดหลอดไฟ 4 ชุด, เครื่องปรับอากาศ 1 ชุด และสามารถตรวจสอบอุณหภูมิห้อง 1 ชุดขณะนั้นได้ (สามารถสร้างชุดควบคุมได้ถึง 3825 ชุด ด้วยการเชื่อมต่อไมโครคอนโทรลเลอร์ได้มากถึง255 ตัว ในการเชื่อมตัวแบบเป็นเครือข่าย)
2. เขียนซอร์ฟแวร์ Visual Basic เพื่อใช้ในการติดต่อสื่อสารและควบคุมไมโครคอลโทรลเลอร์
3. สร้างเวปไซด์ที่ใช้ในการสั่งงานอุปกรณ์

วิธีการดำเนินงาน

วิธีการดำเนินงาน สามารถแสดงเป็นโฟล์วชาร์ทการทำงานดังภาพที่ 2



ภาพที่ 2 แสดงวิธีการดำเนินงานวิจัย

ระบบเครือข่าย

วิธีการควบคุมและตรวจสอบอุปกรณ์ไฟฟ้าผ่านทางระบบอินเตอร์เน็ตโดยการเอา TCP/IP Stack ใส่ลงใน Microcontroller แล้วเลือก H/W Interface โดยให้ผ่านทาง RS232/RS485 (SLIP or PPP)

TCP/IP, SLIP และ PPP

ระบบเครือข่ายหรือ Internet ที่ใช้โปรโตคอลมาตรฐานชื่อ TCP/IP ในการสื่อสารผ่านระบบเพื่อติดต่อกับเครื่องคอมพิวเตอร์อื่นๆ โปรโตคอล TCP/IP นั้นประกอบด้วยส่วนที่สำคัญ 2 ส่วนก็คือ

Transmission Control Protocol (TCP)

- Internet Protocol (IP)

ในการติดต่อสื่อสารกันจริงๆเราจะไม่สามารถเห็นขั้นตอนการทำงานของระบบ ได้เพราะเป็นการทำงานของ Software Hardware แต่เราจะอธิบายเพื่อความเข้าใจของ โปรโตคอล TCP/IP ให้ดูกันดังมีส่วนประกอบดังนี้

- IP Address:

สำหรับการรับส่งข้อมูลในระบบ Internet จะถูกกำหนดและอ้างอิงด้วยหมายเลขประจำเครื่องนั้นก็คือเครื่อง IP Address ซึ่งในระบบ Internet จะมีเครื่องคอมพิวเตอร์เป็นจำนวนมากที่อยู่ในระบบ ในการที่จะใช้ IP Address อาจจะไม่สะดวก จึงได้มีการเปลี่ยนมาใช้เป็นชื่อในความหมายที่เข้ากันก็คือ Domain name โดยทั้งหมดนี้อยู่ในระบบ Name Services ซึ่งเป็นการอ้างอิงชื่อแทนหมายเลขนั้นเอง

- Routing Configuration:

ข้อดีของโปโตคอล TCP/IP ก็คือในการกำหนดเส้นทางสำหรับการรับส่ง ที่สามารถเลือกเส้นทางในการรับส่งข้อมูลได้อย่างอัตโนมัติหากถ้าเกิดเส้นทาง บางเส้นทางเสียหาย ระบบกลไกใน

การกำหนดเส้นทางสำหรับการรับส่งข้อมูลของโปรโตคอล TCP/IP ก็จะเลือกเส้นทางใหม่ให้เหมาะสมถูกต้องให้สามารถรับส่งข้อมูลได้

- Protocol, Ports, Sockets:

เป็นช่องทางสำหรับกำหนดทิศทางของการรับส่งข้อมูลนอกเหนือจากที่จะต้องกำหนดหลังจาก IP Address

SLIP (Serial Line Internet Protocol) และ PPP (Point-to-Point Protocol) เป็นวิธีหนึ่งของ IP ในการเข้ารหัสข้อมูลในการส่งข้อมูลผ่านโมเด็ม สำหรับ SLIP เป็นอีกหนึ่งวิธีในการเข้ารหัสของ IP ในการส่งข้อมูลให้กับโมเด็ม สำหรับ CSLIP (Compressed Serial Line Internet Protocol) เป็นอีกหนึ่งเวอร์ชั่นของ SLIP ซึ่งมีการบีบอัดข้อมูล ซึ่งจะมีขนาดเล็ก

PPP เป็นอีกหนึ่งวิธีในการเข้ารหัสข้อมูล ซึ่งจะมีความปลอดภัยของข้อมูลดีกว่า

แม่ข่าย (Server) และ ลูกข่าย (Client)

สิ่งที่เราจะพูดถึงในการเขียนโปรแกรมคือเพื่อใช้ในระบบเครือข่ายจุดหลักๆของระบบ จะแบ่งฝ่ายที่ต้องติดต่อรับส่งข้อมูลระหว่างกันออกเป็น 2 ส่วน คือ แม่ข่าย และลูกข่าย ซึ่งในการใช้งานจริงอาจมีส่วนประกอบอื่นๆอีก แต่จะขอไม่กล่าวถึงเนื่องจากต้องการให้เห็นภาพและเข้าใจง่ายขึ้นจึงยก แค่ 2 ส่วนนี้มากล่าว

แม่ข่ายจะเป็นส่วนทำหน้าที่เสมือนกองอำนวยการ, ประชาสัมพันธ์, เมสเซทเจอร์ ร่วมถึงให้กับระบบ, ลูกข่าย ที่จะเป็นส่วนร้องขอข้อมูลจากแม่ข่าย โดยแม่ข่ายในที่นี้จะเป็นส่วนที่เก็บข้อมูล, จัดการ, บริหารข้อมูลหรือทรัพยากรระบบ เพื่อให้ฝ่ายลูกข่ายสามารถใช้บริการ

แม่ข่ายและลูกข่าย ต่างก็จะต้องมีตำแหน่งที่อยู่ (IP Address), ช่องทางการติดต่อ (Port) โดยทั้งสองฝ่าย จะสามารถติดต่อถึงกันได้จะต้องอยู่ในช่องทางเดียวกัน ซึ่งเราสามารถกำหนดหมายเลขของ Port ได้ ทั้งนี้โปรแกรมที่ติดต่อนั้นจะต้องอ้างอิงหมายเลขของ Port ทุกครั้งเนื่องจากในระบบมีโปรแกรมมากมายที่กำลังติดต่อกันอยู่

การสร้างระบบควบคุมและการวัดค่าต่างๆ

การสร้างระบบควบคุมและการวัดค่าต่างๆเรามักจะวัดออกมาเป็นปริมาณทางฟิสิกส์โดยใช้เซ็นเซอร์ชนิดต่างๆเปลี่ยนจากปริมาณที่เราต้องการวัด เช่นความเข้มแสง ไปเป็นความต่างศักย์ที่เปลี่ยนแปลงตามค่าของความเข้มแสงใน PIC 16F877 สามารถวัดความต่างศักย์ได้โดยใช้ A/D (Analog to Digital Convertor)

เหตุที่เลือก CPU เป็นเบอร์ PIC 16F877 ก็เพราะเป็นซีพียูที่มีคุณสมบัติและฟังก์ชั่นการทำงานต่างๆที่ค่อนข้างครบถ้วนและมีโครงสร้างการทำงานที่ไม่ยุ่งยาก

ไมโครคอนโทรลเลอร์

ไมโครคอนโทรลเลอร์เป็นไอซีที่รวมวงจรที่จำเป็น เช่น หน่วยความจำ อินพุต เอาท์พุต บางส่วนเข้าไปในไอซีตัวเดียวกัน และเพิ่มวงจรบางอย่างเข้าไปเพื่อให้มีความสามารถเหมาะกับการใช้ในงานควบคุม เช่น วงจรตั้งเวลา วงจรการสื่อสารอนุกรมและสามารถโปรแกรมการทำงานได้หลายครั้ง ไมโครคอนโทรลเลอร์ของแต่ละบริษัท จะมีโครงสร้างพื้นฐานภายในเหมือนกัน แต่ต่างกันในเรื่องของฟังก์ชั่นพิเศษที่เพิ่มเข้ามา เช่น จำนวนอินพุต เอาท์พุตที่แตกต่างกัน

ไมโครคอนโทรลเลอร์ตระกูลPIC

เป็นไมโครคอนโทรลเลอร์ขนาดเล็ก ที่ออกแบบสำหรับงานควบคุมขนาดเล็ก เนื่องจาก PIC มีพอร์ตที่มีศักยภาพสูง สามารถเป็นได้ทั้งอินพุตและเอาท์พุต โดยเมื่อใช้งานเป็นเอาต์พุตก็สามารถจ่ายกระแสได้มากถึง 20-25 มิลลิแอมป์ ทำให้สามารถขับไดโอดเปล่งแสงหรือ LED และเซอร์โวมอเตอร์แรงดันต่ำได้ โดยไม่ต้องใช้อุปกรณ์ภายนอกเพิ่มเติมเนื่องจากพอร์ตของ PIC สามารถเป็นได้ทั้งอินพุตและเอาต์พุต ทำให้สามารถใช้รับสัญญาณอะนาลอกได้โดยตรง โดยไม่ต้องพึ่งวงจร แปลงสัญญาณอะนาลอกเป็นดิจิตอล วงจรที่ว่าคือ วงจร RC โดย R จะแทนที่ด้วยตัวต้านทานแปรค่าตามแสง หรือที่เรียกว่า LDR (Light Dependent Resistor), เทอร์มิสเตอร์ หรือตัวต้านทานแปรค่าตามอุณหภูมิ เป็นต้น ซึ่งอุปกรณ์ทั้งสองแบบนั้นก็คือ ตัวตรวจจับหรือเซนเซอร์แบบหนึ่งนั่นเองด้านการโปรแกรม PIC สามารถทำการโปรแกรมแบบในวงจร ได้ทุกเบอร์ (เว้นอนุกรม PIC16Cx ซึ่งล้าสมัยไปแล้ว) ทำให้การพัฒนาสามารถทำได้อย่างสะดวก และใช้อุปกรณ์ไม่

มากนอกจากนั้นใน PIC เบอร์ใหม่ๆจะบรรจุโมดูล แปลงสัญญาณอะนาลอกเป็นดิจิตอล ความละเอียด 10 บิตไว้ในตัว ยิ่งทำให้ PIC มีศักยภาพสูงขึ้น และเหมาะอย่างยิ่ง ในการนำมาสร้าง ระบบควบคุมหุ่นยนต์อัตโนมัติขนาดเล็ก เบอร์ที่แนะนำ คือ PIC16F873, PIC16F874, PIC16F876 และ PIC16F877 ซึ่งนอกจากจะมีโมดูล แปลงสัญญาณอะนาลอกเป็นดิจิตอลแล้ว ยังมีโมดูลสร้างสัญญา PWM, โมดูลสื่อสารกับพอร์ตอนุกรม และโมดูลเชื่อมต่อกับไอซีฟังก์ชั่นพิเศษ ในระบบบัส I2C และ SPI ทำให้การขยายความสามารถทำได้ง่ายขึ้นแต่ข้อจำกัดของ PIC คือ การต่อขยายหน่วยความจำข้อมูล ขนาดใหญ่ PIC สามารถทำได้แต่ไม่สะดวก

การรับและส่งข้อมูลแบบต่างๆ

การวิจัยนี้จะใช้งานโปรแกรม Visual BASIC เวอร์ชั่น 6 เป็นภาษาสำหรับติดต่อระหว่างอุปกรณ์ภายนอกกับคอมพิวเตอร์ ซึ่งโปรแกรมดังกล่าวจะมีคัสตอมคอนโทรล (MSComm) ซึ่งเป็นคอนโทรลตัวหนึ่งที่ช่วยในการติดต่อกับพอร์ตทอนุกรม (serial port) ซึ่งสามารถทำการส่งข้อมูลผ่านทางพอร์ตทอนุกรมได้ด้วยคอนโทรลนี้ สำหรับการสื่อสารอนุกรมผ่านทางพอร์ตอนุกรมของคอมพิวเตอร์

มาตรฐานการสื่อสารข้อมูลแบบอนุกรม

ระบบบัสแบบ I2C

ระบบบัสแบบ I2C ย่อมาจาก Inter-IC Communication ซึ่งพัฒนาโดยห้องวิจัยของ Phillips ในประเทศฮอลแลนด์ เมื่อปี ค.ศ.1980 สำหรับเชื่อมต่อชิปต่าง ๆ กับระบบไมโครคอนโทรลเลอร์แบบอนุกรมโดยใช้สัญญาณเพียงสองเส้น เส้นที่หนึ่งเป็นสายข้อมูล อีกเส้นหนึ่งเป็นสายสัญญาณนาฬิกาสำหรับกำหนดจังหวะการสื่อสารข้อมูล ปัจจุบันได้มีชิปสนับสนุนการทำงานของไมโครคอนโทรลเลอร์หลายตัวที่ใช้การเชื่อมต่อระบบบัสแบบนี้สายข้อมูลที่ใช้รับส่งข้อมูลแบบอนุกรมมีชื่อว่า Serial Data Line (SDA) ส่วนสายสัญญาณนาฬิกาควบคุมมีชื่อว่า Serial Clock Line (SCL) สายทั้งสองนี้จะรับส่งข้อมูลได้สองทิศทาง วงจรทางเอาท์พุตของอุปกรณ์ที่ใช้บัสแบบนี้จะเป็นวงจรเดรนเปิด (Open-drain) หรือคอลเล็กเตอร์เปิด (Open collector) ดังนั้นการต่อบัสแบบนี้จะต้องมีตัวต้านทานต่อพูลอัพกับขั้วบวกของแรงดันไฟฟ้า +5 โวลต์ด้วยการเชื่อมต่อด้วยระบบบัสแบบ I2Cระบบบัสแบบนี้สามารถรับส่งข้อมูลได้สองทิศทางด้วยความเร็วสูงถึง 100 กิโลบิตต่อ

วินาที และสามารถใช้กับไอซีที่ใช้แรงดันไฟฟ้าต่างกันได้ ระบบบัสแบบนี้สามารถเชื่อมต่อกับอุปกรณ์ได้หลายตัวโดยที่อุปกรณ์แต่ละตัวที่ต่ออยู่กับระบบบัสแบบนี้สามารถส่งข้อมูลถึงกันได้ โดยใช้รูปแบบการรับส่งข้อมูลหรือโปรโตคอล (Protocol) ที่อุปกรณ์ทุกตัวรู้จัก อุปกรณ์แต่ละชนิดจะมีค่าแอดเดรสประจำตัวมัน ถ้าหากต้องการให้อุปกรณ์ตัวใดรับข้อมูล ตัวส่งจะส่งแอดเดรสของอุปกรณ์ตัวนั้นออกไปก่อน ถ้าหากอุปกรณ์ตัวใดมีแอดเดรสตรงกันก็จะรับข้อมูลนั้นไป สำหรับอุปกรณ์ที่ต้องการส่งข้อมูลอุปกรณ์ตัวนั้นจะเรียกว่ามาสเตอร์ (Master) โดยจะเป็นตัวสร้างจังหวะสัญญาณต่าง ๆ บนระบบบัส ส่วนอุปกรณ์ที่ถูกควบคุมหรือเป็นตัวรับข้อมูลจะเรียกว่าสเลฟ (Slave) ถ้าหากทำการเชื่อมต่อไมโครคอนโทรลเลอร์กับอุปกรณ์ที่ใช้บัสแบบ I2C หลายตัว ถ้าไมโครคอนโทรลเลอร์ต้องการส่งข้อมูล ตัวมันจะทำหน้าที่เป็นมาสเตอร์ และจะส่งสัญญาณไปบนบัส โดยส่งค่าแอดเดรสออกไปก่อน ถ้าค่าแอดเดรสนี้ตรงกับแอดเดรสของชิปตัวใด ชิปตัวนั้นจะส่งสัญญาณตอบรับ(ACK) ออกมา และชิปตัวนั้นก็จะถูกไมโครคอนโทรลเลอร์ควบคุม แต่ถ้าแอดเดรสไม่ตรงกับชิปตัวใดก็จะไม่เกิดอะไรขึ้นการส่งข้อมูลต่าง ๆ จะต้องเกิดขึ้นเมื่อระบบบัสว่างเท่านั้น โดยทั่วไปแล้วสภาวะที่มีบนระบบบัสแบบนี้จะมี 5 สภาวะด้วยกันดังนี้

การถ่ายโอนข้อมูล

1. สภาวะบัสว่าง (Bus not busy) สภาวะนี้ค่าลอจิกบนสาย SDA และ SCL จะเป็นลอจิกสูงทั้งคู่

2. ภาวะเริ่มส่งข้อมูล (Start data transfer) สภาวะนี้สาย SCL จะเป็นลอจิกสูง แต่สาย SDA จะเปลี่ยนจากลอจิกสูงไปเป็นลอจิกต่ำ เรียกว่าสภาวะเริ่มต้น (Start)

3. สภาวะหยุด (Stop) สภาวะนี้สาย SCL จะเป็นลอจิกสูง แต่สาย SDA จะเปลี่ยนจากลอจิกต่ำไปเป็นลอจิกสูง

4. สภาวะมีข้อมูล (Data valid) สภาวะนี้จะอยู่ระหว่างสภาวะเริ่มต้นและสภาวะหยุด โดยการรับส่งข้อมูลต่าง ๆ จะเกิดขึ้นในสภาวะนี้ การรับส่งข้อมูลแต่ละบิตจะใช้สัญญาณนาฬิกาหนึ่งลูก โดยข้อมูลบนSDA จะต้องคงที่ในขณะที่ SCL เป็นลอจิกสูง และบิตข้อมูลใน SDA จะเปลี่ยนแปลงได้ขณะที่เงื่อนไขเริ่มต้น SDAเปลี่ยนจาก 1 เป็น 0ขณะที่ SCL เป็น 1เงื่อนไขสุดท้าย SDAเปลี่ยนจาก 0 เป็น 1 ขณะที่ SCL เป็น SCL ที่เป็นลอจิกต่ำ ถ้าหากบน SDA มีการเปลี่ยนแปลงในขณะที่ SCL เป็นลอจิกสูง ระบบจะตีความว่าเป็นสภาวะเริ่มต้นการส่งข้อมูล หรือมีสภาวะหยุดแทน

5. สภาวะตอบรับ (Acknowledge) เมื่ออุปกรณ์มาสเตอร์ส่งข้อมูลออกมาจนครบหนึ่งไบต์แล้วในช่วงสัญญาณ SCL ลูปที่มาสเตอร์จะส่งข้อมูลลอจิกสูงออกมา และถ้าตัวรับได้รับข้อมูล-

ครบแล้วมันจะส่งสัญญาณตอบรับ ACK โดยทำให้ระดับลอจิกสูงบนสัญญาณ SDA ให้กลับเป็นระดับลอจิกต่ำแต่ถ้าตัวรับได้รับข้อมูลไม่ถูกต้องตัวรับจะบังคับให้ตัวส่งหยุดในสภาวะรอในการถ่ายโอนข้อมูลระหว่างสภาวะเริ่มต้นและสภาวะหยุดสามารถโอนถ่ายข้อมูลไม่จำกัด แต่เมื่อ slave ได้รับข้อมูลแต่ละไบต์แล้ว มันจะส่งสัญญาณกลับมาทุกครั้งการเชื่อมต่ออุปกรณ์ต่าง ๆ ที่ใช้ระบบบัสแบบ I2C นั้นเราสามารถใช้อุปกรณ์หลาย ๆ ตัวมาต่อกับไมโครคอนโทรลเลอร์ได้ แต่สายสัญญาณทั้งหมดมีเพียงสองเส้นเท่านั้น ดังนั้นการติดต่อกับอุปกรณ์แต่ละตัวจะต้องระบุด้วยว่าจะต้องการติดต่อกับอุปกรณ์ตัวใด ขั้นตอนการเขียนข้อมูลที่ใช้กับระบบบัสแบบ I2C มีสามขั้นตอนดังนี้

5.1 เขียนข้อมูลอ้างแอดเดรส (Addressing) ข้อมูลไบต์แรกที่ master ส่งออกไปคือข้อมูลที่ใช้อ้างแอดเดรสของอุปกรณ์ที่ต้องการติดต่อ โดยทั่วไปแล้วข้อมูลที่กำหนดแอดเดรสจะมีจำนวน 7 บิต บิตแอดเดรสคงที่บิตแอดเดรสโปรแกรมได้จะเห็นว่าบิตต่ำสุดจะเป็นตัวระบุว่าอุปกรณ์ที่ต้องการติดต่อนั้นเราจะอ่านหรือเขียนข้อมูลลงไป ถ้าบิตนี้เป็นลอจิก “0” หมายความว่าจะใช้เขียนข้อมูลลงไป ถ้าเป็นลอจิก “1” หมายความว่าจะอ่านข้อมูล ส่วน 7 บิตบนจะเป็นบิตแอดเดรส ซึ่งแบ่งออกเป็นบิตแอดเดรสคงที่ (Fixed address bit) จำนวน 4 บิตบน ซึ่งจะถูกโปรแกรมมาจากโรงงานที่ผลิตชิปแต่ละประเภท ส่วนอีก 3 บิตต่อมาจะเป็นบิตแอดเดรสที่ผู้ใช้สามารถโปรแกรมได้ (Programmable address bit)

5.2 การเขียนไบต์ควบคุม (Control byte) ข้อมูลไบต์นี้จะขึ้นกับอุปกรณ์แต่ละประเภท เพื่อกำหนดการทำงานต่าง ๆ ของตัวมัน อุปกรณ์บางประเภทอาจจะไม่ต้องมีการเขียนไบต์นี้ก็ได้

5.3 การเขียนไบต์ข้อมูล (Data byte) เป็นข้อมูลที่จะถ่ายโอนในระบบ ระหว่างสภาวะเริ่มต้นและสภาวะหยุดจะมีการอ่านเขียนข้อมูลจำนวนกี่ไบต์ก็ได้

การสื่อสารข้อมูลแบบอนุกรม RS-232

มาตรฐานการสื่อสารข้อมูลแบบอนุกรม RS-232 หรือ EIA-232 จะมีการกำหนดทั้งลักษณะของอุปกรณ์ (Mechanical), คุณสมบัติทางไฟฟ้า (Electrical) และหน้าที่การทำงาน (Function) ของการเชื่อมต่อระหว่างอุปกรณ์ DTE (Data Terminal Equipment) และ DCE (Data Communication
Equipment) ดังต่อไปนี้

Mechanical Specification

กำหนดใช้สายเคเบิลนำสัญญาณแบบ 25 หรือ 9 เส้น หัวต่อตัวผู้และตัวเมียเป็นแบบ DB-25 หรือ DB-9 ความยาวของสายเคเบิลไม่เกิน 15 เมตร อนุญาตให้ใช้ Bit Rate ได้สูงสุดคือ 20 Kbps

Electrical Specification

กำหนดระดับแรงดันของสัญญาณและชนิดของสัญญาณ RS-232 จะต้องส่งสัญญาณข้อมูล (Data Signal) เป็นลอจิก "0" หรือ "1" ใช้การเข้ารหัสแบบ NRZ-L (Non-return to zero, level) โดย

- ลอจิก "0" กำหนดเป็นแรงดันไฟฟ้าบวกซึ่งอยู่ในช่วง +3 ถึง +15 V
- ลอจิก "1" กำหนดเป็นแรงดันไฟฟ้าลบซึ่งอยู่ในช่วง -3 ถึง -15 V

Function Specification

กำหนดหน้าที่ให้แต่ละขาของ DB-25 และ DB-9

ความรู้เบื้องต้นเกี่ยวกับมาตรฐาน RS-232

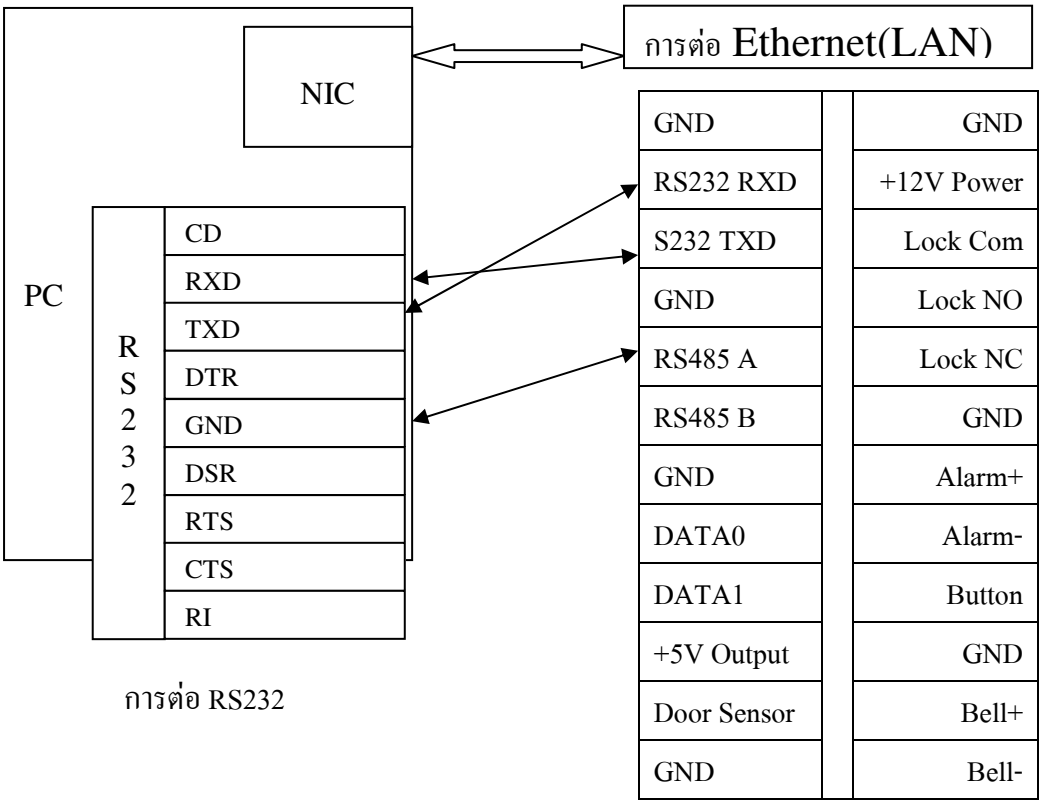
โดยปกติคอมพิวเตอร์จะมีพอร์ทที่เป็นอนุกรมชื่อว่า RS-232 อยู่ในตัว ซึ่งพอร์ท RS-232 นี้ทำหน้าที่รับและส่งข้อมูลในแบบอนุกรมเรียกว่า Universal Asynchronous Adapter เหตุที่มีชื่อเรียกว่า RS-232 เนื่องมาจากสมาคมผู้ผลิตอุปกรณ์อิเล็กทรอนิกส์ของอเมริกา หรือ EIA (RS-232: Recommended Standard Number 232,EIA: Electronic Industry Association) ได้กำหนดมาตรฐานของอุปกรณ์สื่อสารแบบอนุกรมเอาไว้ภายใต้ชื่อว่า RS-232

มาตรฐาน RS-232C

มาตรฐาน RS-232C ได้จัดพิมพ์ขึ้นเมื่อ ค.ศ. 1969 RS ย่อมาจาก Recommended Standard ส่วน 232 คือหมายเลขบ่งบอกมาตรฐานตัวนี้ และ C เป็นหมายเลขฉบับสุดท้ายของมาตรฐานตัวนี้ จุดประสงค์ของมาตรฐาน RS-232 ก็เพื่อบรรยายคุณลักษณะของการเชื่อมต่ออุปกรณ์รับส่งข้อมูล (DCE: Data Communication Equipment) กับอุปกรณ์สื่อสารข้อมูล (DTE: Data Terminal Equipment)

การเชื่อมต่อกับ Computer

เชื่อมต่อ Computer แบ่งได้ 3 แบบคือ RS232, RS485 และ Ethernet (TCP/IP) ดังภาพที่ 3



การต่อ RS232

ภาพที่ 3 แสดงทิศทางการเชื่อมต่อกับ Computer แบบ RS232

ในการสื่อสารแบบ RS485 นี้จะมีคุณสมบัติของสัญญาณทางไฟฟ้าเหมือนกับ RS422 ทุกประการเพียงแต่ว่าในการสื่อสารแบบ RS485 นี้จะใช้สายสัญญาณในการรับส่งข้อมูลกันเพียง 2 เส้น เท่านั้น แต่จะมีความพิเศษกว่าแบบ RS422 ตรงที่ ทิศทางของสัญญาณจะสามารถปรับเปลี่ยนได้จากโปรแกรม กล่าวคือสัญญาณทั้ง 2 เส้นนี้สามารถจะสลับหน้าที่เป็นด้านส่ง และ เป็นด้านรับได้ ตามต้องการ โดยการควบคุมจาก MCU โดยจะกำหนดให้สัญญาณ PTC3 ทำหน้าที่สำหรับควบคุมทิศทางของข้อมูลว่าจะให้เป็นรับหรือส่ง โดยถ้าควบคุมให้ PTC3 มีสภาวะเป็น “1” จะเป็นการกำหนดทิศทางให้เป็นฝ่ายส่งข้อมูล แต่ถ้าสภาวะของ PTC3 เป็น “0” จะเป็นการกำหนดทิศทางให้เป็นฝ่ายรับข้อมูล ซึ่งจากคุณสมบัติข้อนี้จะทำให้การสื่อสารแบบ RS485 สามารถทำการต่อขนานอุปกรณ์ร่วมกันในสายส่งเดียวกันได้จำนวนหลายๆจุด โดยถ้าใช้ไอซี Line Driver เบอร์ 75176 จะสามารถต่อขนานอุปกรณ์กันได้ จำนวน32 จุด แต่ถ้าเลือกใช้ไอซี Line Driver เบอร์

MAX3088 แล้วจะสามารถต่อขนานอุปกรณ์ในสายคู่เดียวกันได้มากถึง 256 จุด เลยทีเดียว แต่มีข้อแม้ว่า เมื่อมีการต่ออุปกรณ์ขนานกันในสายสัญญาณคู่เดียวกันมากกว่า 2 จุดแล้ว จะต้องเขียนโปรแกรมควบคุมให้มีการส่งข้อมูลออกมาในสายครั้งละ 1 จุดเท่านั้น เพราะถ้ามีการกำหนดทิศทางของข้อมูลให้เป็นส่งในเวลาเดียวกันมากกว่า 1 จุดแล้วจะทำให้เกิดการชนกันของข้อมูลและไม่สามารถสื่อสารกันได้อย่างถูกต้องโดยเมื่อต้องการใช้วิธีการสื่อสารแบบ RS485 นี้ จะต้องทำการติดตั้งไอซี Line Driver เบอร์ 75176หรือ MAX3088 ในตำแหน่งของ “TXD/485” เพียงตัวเดียว พร้อมกับเลือกกำหนดเป็นแบบ RS485 ดังนี้

- ทำการเลือก Jumper สำหรับเลือก “422/485” ไว้ทางด้าน 485 (RS485)
- ทำการเลือก Jumper “F/H” ไว้ทางด้าน H (Half Duplex)
- ทำการ Short Jumper สำหรับต่อตัวต้านทาน Fail Safe Resister คือ “TL” ไว้
- ทำการ Short Jumper สำหรับต่อตัวต้านทาน Fail Safe Resister คือ “TH” ไว้
- สายสัญญาณที่ใช้จะต่อจาก TXB (TX-) และ TXA (TX+) เพียง 2 เส้น ออกไปใช้งาน

ซึ่งในการสื่อสารข้อมูลแบบ RS485 นี้ จะต้องเขียนโปรแกรมขึ้นมารองรับการสื่อสารโดยเฉพาะ เนื่องจากทิศทางของข้อมูลสามารถจะกำหนดจากโปรแกรมได้โดยตรง ซึ่งการสื่อสารวิธีนี้จะมีข้อดี คือ ใช้สายสัญญาณในการรับส่งน้อยเส้น แต่จะเสียเวลาในการสื่อสารมากกว่าวิธีอื่นๆ เนื่องจากว่าการสื่อสารแบบนี้จะไม่สามารถทำการรับและส่งข้อมูลในเวลาเดียวกันได้ แต่จะต้องใช้วิธีการ ผลัดกันรับ ผลัดกันส่งแทน ซึ่งในความเป็นจริงแล้วในปัจจุบันนี้ ราคาของสายสัญญาณแบบ 2 เส้น และ 4 เส้น แทบจะไม่มีความแตกต่างกันเลย

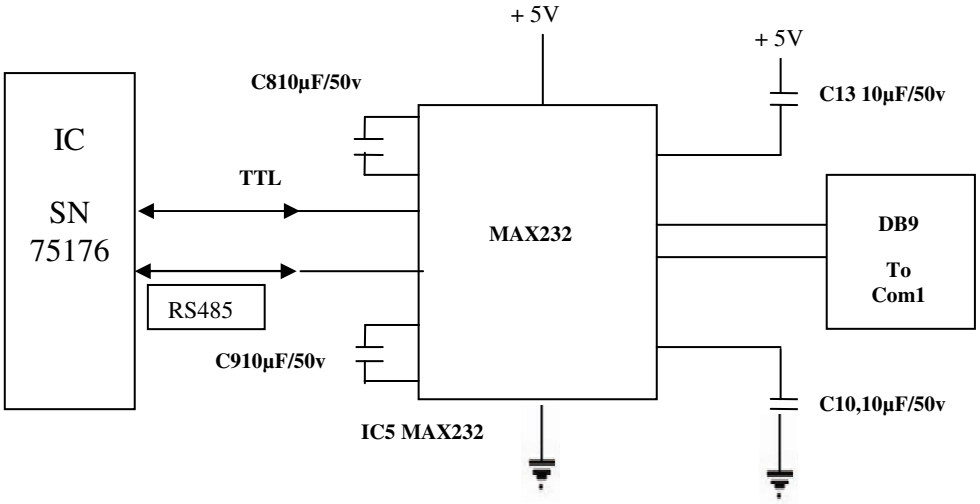
มาตรฐาน RS-422 หรือ RS-422-A

ถูกกำหนดขึ้นโดยสมาคมผู้ประกอบการอุตสาหกรรมอิเล็กทรอนิกส์หรือ EIA เช่นเดียวกันกับมาตรฐาน RS-232 โดยมีจุดมุ่งหมายที่จะแก้ปัญหาเรื่องความยาวของสายสื่อสารโดยใช้การส่งสัญญาณแบบผลต่าง (Differential) แทนที่จะใช้การส่งสัญญาณแบบอ้างอิงกับจุดกราวนด์ (หรือสายดิน) เช่นเดียวกันกับ RS-232 การส่งสัญญาณแบบ Differential นี้ช่วยลดปัญหาสัญญาณรบกวนจาก 2 ปัจจัยด้วยกัน ได้แก่ ปัญหาแรงดันกราวนด์ 2 ฝั่งสายไม่เท่ากัน อันเกิดจากกระแสไฟฟ้าที่ไหลในสายกราวนด์ที่ยาวมากๆ ก่อให้เกิดความต่างศักดิ์ และปัญหาสัญญาณรบกวนที่เกิดจากแม่เหล็กไฟฟ้าเหนี่ยวนำในสาย โดยหากสายไฟที่ใช้ถูกตีเกลียวและวางไว้ใกล้กัน เมื่อมีแรงดันเหนี่ยวนำจะปรากฏแรงดันรบกวนบนสายทั้งสองเท่าๆ กันเป็นผลให้ ตัวรับที่อ่านความต่างศักดิ์ระหว่างสายอ่านข้อมูลได้เช่นเดิม ทั้งสองปัจจัยนี้เองเป็นสาเหตุที่ทำให้ความต้านทานต่อสัญญาณรบกวนของการสื่อสารแบบ RS-232 ด้อยกว่า RS-422) ตามมาตรฐาน RS-422 นี้จะใช้สายสัญญาณ

ทั้งหมด 4 เส้น (2 เส้นสำหรับการส่งสัญญาณ และอีก 2 เส้นสำหรับรับสัญญาณ) และสามารถใช้ความยาวสายสัญญาณได้ถึง 4,000 ฟุต (หรือ 1.2 กม.) ที่ความเร็ว 100,000 บิทต่อวินาที และการสื่อสารเป็นแบบ 2 ทางพร้อมกัน (Full Duplex)

วงจรรับ-ส่ง ข้อมูลตามมาตรฐาน RS 485

ชุดแปลงสัญญาณตามมาตรฐาน RS232จะทำหน้าที่เชื่อมต่อการติดต่อสื่อสารระหว่างเครื่องคอมพิวเตอร์กับชุดควบคุมด้วยไมโครคอนโทรเลอร์เนื่องจากระดับแรงดันไฟในการติดต่อสื่อสาร ของเครื่องคอมพิวเตอร์นั้นมีระดับไฟที่ใช้ในการติดต่อสื่อสารที่ -15V กับ +15V แต่ในการติดต่อสื่อสารของชุดควบคุมด้วยไมโครคอนโทรเลอร์ใช้ระดับแรงดันไฟในการติดต่อสื่อสารที่ 0V กับ 5V เพื่อให้เกิดการติดต่อสื่อสารกันได้ จะต้องมีชุดแปลงสัญญาณตามมาตรฐาน RS232 ซึ่งใช้ไอซีเบอร์ MAX232 ในการปรับระดับแรงดันในการติดต่อกันระหว่างเครื่องคอมพิวเตอร์ และชุดวงจรควบคุมดังนั้นการติดต่อสื่อสารต้องใช้สายนำสัญญาณมาใช้ในการติดต่อผ่านพอร์ตของเครื่องคอมพิวเตอร์โดยใช้ พอร์ตอนุกรมในการควบคุม

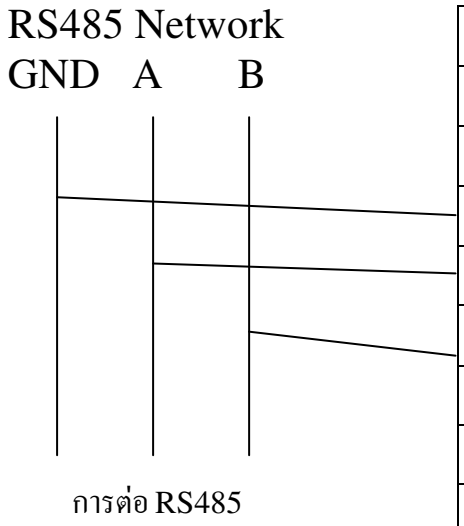


ภาพที่ 4 ชุดเชื่อมต่อคอมพิวเตอร์กับวงจรแปลงสัญญาณตามมาตรฐาน RS232

จากภาพที่4 ขาที่ 7, 8 จะเป็นขาที่ใช้งานในการส่งสัญญาณผ่านพอร์ตแบบ DB9 เข้าสู่คอมพิวเตอร์ที่พอร์ต COM1 และขาใช้งานที่ 9, 10 จะเป็นขาที่ต่อเชื่อมกับ วงจรรับ-ส่งวงจรรับ-ส่งข้อมูลตามมาตรฐาน RS 485 เพื่อส่งข้อมูลได้ระยะไกล

มาตรฐานRS485 เป็นไอซีที่ใช้เปรียบเทียบผลต่างของสัญญาณสามารถเป็นทั้งตัวส่งและตัวรับได้ภายในตัวเดียวกัน ทำหน้าที่ รับหรือส่งข้อมูล ออกทางParallel port (RS-485) โดยไอซี

เบอร์ SN75176 จะรับสัญญาณ TTL มาจากไอซีเบอร์ MAX232 ที่ขา 1 และ ขา 4 และ ส่งข้อมูลที่ขา 6 (RX, TX) ทางสายส่งซึ่งมีแรงดันไฟฟ้า ที่ระหว่าง -7 กับ +12 โวลต์สามารถรับส่งได้ไกลประมาณ 1,300 เมตร เมื่อส่งข้อมูลแล้วที่ภาครับจำเป็นต้องแปลงข้อมูลเป็นมาตรฐาน RS232 อีกครั้ง เพื่อให้แรงดันของสัญญาณสามารถติดต่อกับ ชุดไมโครคอนโทรเลอร์ เพื่อควบคุมและติดต่อกันในระบบต่อไป



การต่อ RS485

GND		GND
RS232 RXD		+12V Power
S232 TXD		Lock Com
GND		Lock NO
RS485 A		Lock NC
RS485 B		GND
GND		Alarm+
DATA0		Alarm-
DATA1		Button
+5V Output		GND
Door Sensor		Bell+
GND		Bell-

ภาพที่ 5 แสดงทิศทางการเชื่อมต่อกับ Computer แบบ RS485

จากภาพที่ 5 เป็นการเชื่อมต่อกับคอมพิวเตอร์แบบ RS485 การเลือกสายเคเบิลเป็นสิ่งที่ควรให้ความสำคัญอย่างหนึ่งในการออกแบบระบบแม้จะเป็นรายละเอียดเล็กๆน้อยๆ เพื่อให้ได้สายเคเบิ้ลที่มีคุณภาพเหมาะสมและราคาไม่แพงเกินความจำเป็น

จำนวนของตัวนำในสายเคเบิ้ล (Number of Conductor)

สายกราวด์เป็นสายที่ไม่ควรลืมในการสั่งซื้อสายเคเบิล ควรเลือกสายเคเบิลให้มีตัวนำเพียงพอในการส่งสัญญาณทั้งสัญญาณข้อมูลและสัญญาณกราวด์ ยกตัวอย่างเช่น ถ้าเราเลือกใช้สาย

คู่ตีเกลียวในระบบแบบ 2 สาย ต้องใช้สายคู่ตีเกลียว 2 คู่ และสำหรับระบบแบบ 4 สาย ต้องใช้สายคู่ตีเกลียว 3 คู่ เป็นต้น

การชิลด์ (Shielding)

เป็นการยากในการคำนวณว่าควรใช้สายที่มีการชิลด์หรือไม่ แต่อย่างไรก็ตามควรเลือกสายเคเบิ้ลที่มีการชิลด์เนื่องจากมีความปลอดภัยมากกว่าเมื่อเทียบกับราคาที่แพงกว่าสายเคเบิลที่ไม่มีการชิลด์เพียงเล็กน้อย

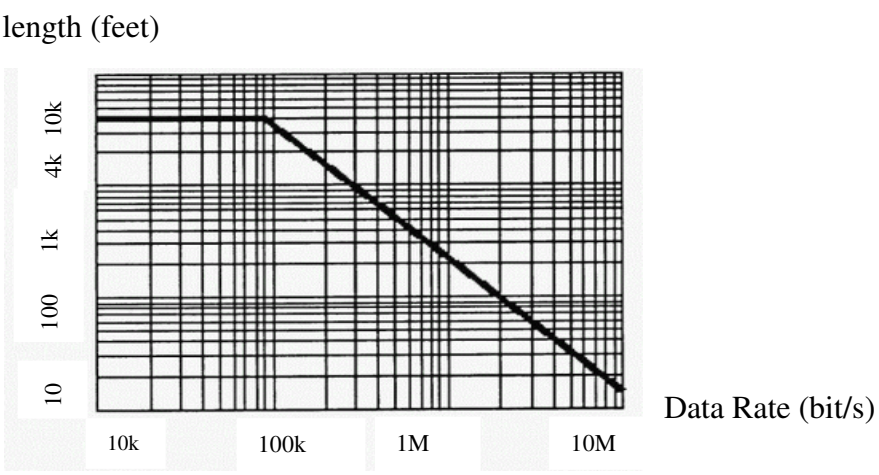
คุณสมบัติของสายเคเบิ้ล (Cable Characteristic)

การเลือกใช้สายเคเบิลสำหรับระบบ RS-422 หรือ RS-485 ขึ้นอยู่กับระยะทางและอัตราการส่งข้อมูลของระบบในสายเคเบิลมีค่าอิมพีแดนซ์หลายชนิด เช่น ความนำไฟฟ้า, ความต้านทาน, คามจุไฟฟ้าและความเหนี่ยวนำ เป็นต้น ซึ่งค่าอิมพีแดนซ์เหล่านี้มีผลต่อระยะทางในการส่งโดยเฉพาะค่าความจุไฟฟ้าขนานยิ่งมีค่ามาก ระยะทางการส่งข้อมูลจะลดลงค่าความเร็วของสัญญาณที่วิ่งในสายเคเบิล หรือเรียกว่า Propagation velocity เป็นสิ่งที่ควรพิจารณาถึงอีกประการหนึ่ง ผู้ผลิตสายเคเบิลจะบอกค่านี้เป็นเปอร์เซ็นต์ และเราสามารถหาค่า Propagation velocity ได้จากผลคูณของค่าที่ผู้ผลิตให้กับความเร็วแสง (3x10E8) เช่น สายเคเบิลมีค่า Propagation velocity เท่ากับ 78% ของความเร็วแสง ดังนั้น

Propagation velocity = 0.78x3x10e8 = 234x10e6 m/s

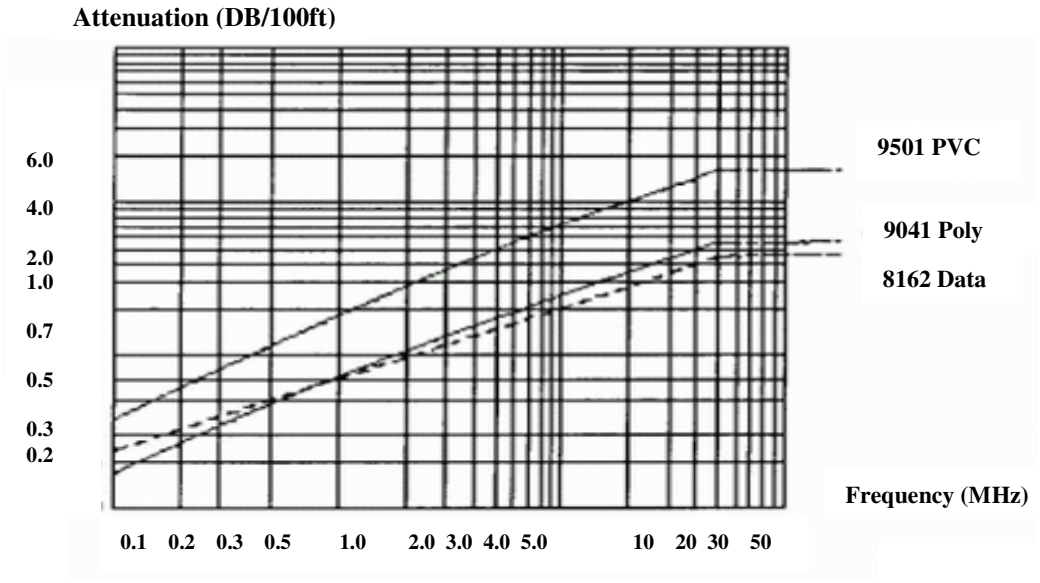
มาตรฐาน EIA RS-422-A แนะนำให้มีการใช้สายคู่ตีเกลียว 24 AWG ที่มีค่าความจุไฟฟ้าขนานขนาด 16 pF/ft. และค่าความต้านทาน 100 โอห์มสำหรับระบบ RS-422 และ RS-485 รูปที่5 แสดงรูปกราฟที่ได้จากการทดสอบสายคู่ตีเกลียว 24 AWG ดังกล่าว ซึ่งแสดงเป็นความสัมพันธ์ระหว่างความยาวของสายเคเบิลและอัตราข้อมูลของสายโทรศัพท์ จากการทดสอบที่ได้กำหนดเงื่อนไขอีก 2 ข้อ ดังนี้

1. ระยะเวลาของสัญญาณในขาขึ้นและขาลงต้องน้อยกว่าหรือเท่ากับ 1/2 ของระยะระหว่างบิทที่อัตราการ modulate นั้น
2. ค่าความสูญเสียของระดับแรงดันระหว่างตัวส่งและโหลดต้องมีค่าไม่เกิน 6 dB.



ภาพที่ 6 ข้อมูลที่เป็นสัญญาณอัตราเปรียบเทียบกับความยาวสายเคเบิลโดยใช้24AWGเป็นสายเคเบิลคู่

จากภาพที่ 6 เป็นสัญญาณอัตราเปรียบเทียบกับความยาวสายเคเบิลโดยใช้24AWGเป็นสายเคเบิลคู่ นอกจากนี้ยังต้องพิจารณาถึงความสูญเสียในสายส่งซึ่งประกอบไปด้วย ความสูญเสียด้าน AC (skin effect) , ความสูญเสียในตัวนำ ในรูป DC , การรั่วไหล (leakage) และความสูญเสีย AC ที่เกิดจาก dielectric ในสายเคเบิลที่มีคุณภาพสูงมักมีค่าความสูญเสียในตัวนำเท่ากับค่าความสูญเสียจาก dielectric ชี้ให้เห็นถึงความแตกต่างด้านคุณภาพของสายเคเบิ้ลต่างชนิดกัน โดยแสดงความสัมพันธ์ระหว่างการลดทอนสัญญาณกับความถี่ของสายเคเบิล 3 ชนิด จากกราฟสังเกตได้ว่าสายเคเบิลที่ทำจาก Polyethylene มีค่าการลดทอนสัญญาณต่ำกว่าสายเคเบิลที่ทำจาก PVC สายเคเบิลที่เลือกใช้กันมากอีกชนิดหนึ่งคือ สายเคเบิลที่ใช้ใน twisted pair Ethernet Cabling ซึ่งอ้างอิงจากหมวด 5 ของข้อกำหนด EIA/TIA/ANSI 568 เป็นเคเบิลที่นิยมใช้มาก อาจเพราะมีราคาถูกกว่าครึ่งของสายเคเบิลที่กำหนดตามมาตรฐาน RS-422/485 ในสายเคเบิลชนิดนี้มีค่าความจุไฟฟ้าประมาณ 17 pF/ft. และมีความต้านทานเท่ากับ 100 Ω



Above attenuation data is supplied courtesy Belden Wire and Cable Company

Note2 Datalene is a registered trademark of Belden Wire and Cable Company. Datalene is a type of foamed cellular polyethylene insulation.

ภาพที่ 7 แสดงการเปรียบเทียบความถี่สำหรับข้อมูลในแต่ละสายเคเบิล

จากภาพที่ 7 เป็นการเปรียบเทียบความถี่สำหรับข้อมูลในแต่ละสายเคเบิลการเลือกสายเคเบิลอีกวิธีหนึ่งอาจใช้หลักการ E-Grade Program ซึ่งเป็นโปรแกรมการแบ่งหมวดของสายเคเบิลตามระยะทางของระบบ กำหนดโดยบริษัท Anixter Bros. ซึ่งเป็นบริษัทขายผลิตภัณฑ์ที่เกี่ยวข้องกับการ wiring ที่มีเครือข่ายทั่วโลก บริษัท Anixter Bros. ได้แบ่งสายเคเบิลออกเป็น 4 หมวด ดังนี้

E-Grade 1 Limited Distance

E-Grade 2 Standard Distance

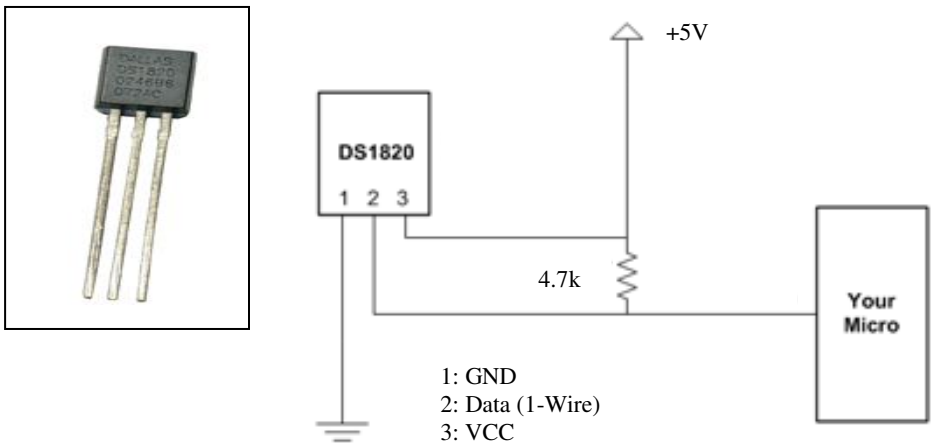
E-Grade 3 Extended Distance

E-Grade 4 Maximum Distance

นอกจากนี้ยังมีข้อมูลง่ายๆที่สามารถใช้พิจารณาในการเลือกสายเคเบิล เช่น มาตรฐาน EIA-232-D , EIA-422-A และ EIA-423-A ซึ่งขึ้นอยู่กับระบบว่าเป็นแบบใด เช่น ถ้าเป็นระบบ RS-485 ก็พิจารณาตามมาตรฐาน EIA-422-A เป็นต้น

ภาคตรวจวัดอุณหภูมิไอซี DS1820

ใช้ IC DS1820 เป็นวงจรชุดตรวจวัดอุณหภูมิสามารถทำงานโดย ไอซี DS1820 ใช้แหล่งจ่ายไฟฟ้าขนาด +5 โวลต์ ไอซี DS1820 จะมีการตรวจวัดอุณหภูมิเกิดขึ้นแล้วค่าของอุณหภูมิก็จะทำการเปรียบเทียบเพื่อทำสัญญาณกระตุ้น การเปรียบเทียบค่าอุณหภูมิจะเปรียบเทียบกับค่าที่ถูกบันทึกหรือกำหนดไว้ของค่าอุณหภูมิสูงสุด (TH) และค่าอุณหภูมิต่ำสุด (TL) ตลอดค่าที่อุณหภูมิที่วัดได้โดยจะใช้พื้นที่รีจิสเตอร์ 8 บิต สำหรับการทำงานนี้ใน เมื่อ MSB ของ TH หรือ TL ที่ตรงกันจะถูกส่งตรงไปยัง SB ของรีจิสเตอร์อุณหภูมิขนาด 16 บิต เพื่อส่งออก ที่ขา D เพื่อติดต่อกับชุดไมโครคอนโทรเลอร์ ที่ขา 16 (P3.6) โดยชุดไมโครคอนโทรเลอร์ จะนำสัญญาณที่ได้รับมาเข้ารหัสเพื่อส่งค่าข้อมูลเป็นแพ็คเก็จโดยวงจรชุดตรวจวัดอุณหภูมิจะเข้ารหัสเก็บข้อมูล 24 บิตโดยข้อมูล 8 บิตแรก จะเป็นเฮดเดอร์ (Header)เป็นตัวอักษร ”B” และข้อมูลอุณหภูมิ(Data) เป็นเลขฐาน 16 จำนวน 16 บิต ให้กับชุดแปลงสัญญาณตามมาตรฐาน RS232 เพื่อส่งข้อมูลเข้าเครื่องคอมพิวเตอร์ต่อไป



ภาพที่ 8 แสดงเบอร์ไอซีของภาคตรวจวัดอุณหภูมิ

จากภาพที่ 8 เป็นลักษณะของไอซีและวิธีการเชื่อมต่อไอซีเบอร์DS1820เหตุผลที่เลือกเซ็นเซอร์ตรวจวัดอุณหภูมิเบอร์ DS1820 เพราะว่ามีค่าความละเอียดอยู่ที่ประมาณ + - 0.5 ซึ่งจะได้ค่าที่ถูกต้องกับอุณหภูมิห้องขณะนั้น

หลักการทำงานของวงจรดิจิตอลเทอร์โมมิเตอร์

วงจรดิจิตอลเทอร์โมมิเตอร์ การต่อใช้งานไอซีตรวจจับอุณหภูมิเบอร์ DS1820 มีเพียง 3 ขาคือ ขา DQ ซึ่งเป็นขาเชื่อมต่อกับระบบบัส ขาต่อไฟเลี้ยงภายนอกและขา GND เมื่อ ไอซี DS1621 ตรวจจับอุณหภูมิได้ จะนำค่าที่ได้นี้มาเก็บไว้ในหน่วยความจำ ไอซี DS1820 สามารถให้ข้อมูลของอุณหภูมิได้ 16 bit เพื่อส่งค่านี้ให้กับไมโครคอนโทรลเลอร์ PIC16F877 คำนวณค่าของอุณหภูมิ แล้วแปลงออกมาเป็นข้อมูลเลขฐาน 10 จึงสามารถแสดงความละเอียดของค่าอุณหภูมิได้ถึง 0.5 องศาเซลเซียส โดยมีย่านวัดอุณหภูมิตั้งแต่ -55 ถึง +125 องศาเซลเซียส

การเชื่องต่อไมโครคอนโทลเลอร์กับไอซี DS1820 จะใช้ขาพอร์ต P3.4 เพียง 1 ขาเท่านั้น โดยมีตัวต้านทาน 4.7 kΩ ต่อพูลอัพกับไฟเลี้ยง +5V. จากนั้นก็ทำการเขียนโปรแกรมลงในไมโครคอนโทรลเลอร์เบอร์ PIC16F877 เพื่อให้วงจรทั้งหมดสามารถทำงานได้ การใช้งานไอซีไมโครคอนโทรลเลอร์ PIC16F877 จะต้องมีการเขียนโปรแกรมควบคุมบันทึกลงในหน่วยความจำโปรแกรมที่มีขนาด 2 Kbyte แล้วต่ออุปกรณ์ภายนอก เช่น Crystal ซึ่งเป็นตัวผลิตความถี่นาฬิกา 11.0592 MHz โดยมีตัวเก็บประจุค่า 22 pF 2 ตัวต่อไว้ร่วมกับผลึกเพื่อสร้างความถี่สัญญาณนาฬิกา และที่ขา RST จะเป็นขาไว้สำหรับสั่งให้ไมโครคอนโทรลเลอร์รีเซตตัวเองขณะที่เริ่มเปิดเครื่องครั้งแรก

ไอซี ไมโครคอนโทรลเลอร์ PIC16F877 จะมีพอร์ตสื่อสารอยู่ 2 พอร์ตคือพอร์ต P3 และ P1 ข้อมูลขนาด 8 bit ที่ขาพอร์ต P1.0-P1.7 จะต่อกับอินพุตของ ไอซีเบอร์ PIC16F877 เพื่อช่วยขยายกระแสให้เพียงพอสำหรับที่จะส่งข้อมูลทั้ง 8 bit หรือข้อมูลตัวอักษรไปเข้าที่ขา P0-P7 ที่ตัวแสดงผล LCD ซึ่งใช้แบบ 16 ตัวอักษร 1 บรรทัด มีขาใช้งานทั้งหมด 14 ขา

การควบคุมให้ LCD แสดงตัวอักษร จะใช้ขา RS, R/W และ E ทำงานร่วมกันโดยขา E เป็นขา Enable LCD ให้ทำงานซึ่งจะต่อกับขาพอร์ต P3.5 ส่วนขา RS เป็นขา INPUT ใช้ในการแยกชนิดของข้อมูลที่ทำการประมวลผลในขณะนั้น ว่าเป็นคำสั่งสำหรับ Register หรือเป็นข้อมูลสำหรับการแสดงผล ซึ่งขานี้จะต่อกับพอร์ต P3.7 ส่วนขา R/W เป็นขาที่ใช้เลือกการอ่านหรือเขียนข้อมูลกับ LCD ในที่นี้จะต่อลง GND ซึ่งหมายความว่า ให้มีการเขียนข้อมูลได้อย่างเดียว

นอกจากนี้ LCD ตัวนี้ยังสามารถปรับความเข้มของการแสดงผลได้ด้วย ซึ่งสามารถปรับได้ที่ VR1 โดยที่ขา V0 เป็น INPUT รับแรงดันเพื่อปรับความเข้มของการแสดงผลว่าจะเลือกสว่างขนาดไหน

ภาคจ่ายไฟในวงจรจะใช้แบตเตอรี่ 9V. จากนั้นทำการลดและควบคุมแรงดันให้เหลือ 5V. โดยไอซี Regulator เบอร์ 7805 สำหรับจ่ายให้วงจรทั้งหมด

หลักการเบื้องต้นของไอซี DS1820

ไอซี DS1820 เป็นไอซีที่มีระบบการสื่อสารข้อมูลอนุกรมซึ่งถือได้ว่าเป็นระบบที่มีความชาญฉลาด และใช้จำนวนสายสัญญาณเพียง 1 เส้นเท่านั้น โดยไม่ต้องมีสายสัญญาณนาฬิกามาควบคุมจังหวะการถ่ายทอดข้อมูลเหมือนกับระบบสื่อสารข้อมูลอนุกรมในแบบอื่น สายข้อมูลจะทำหน้าที่เสมือนเป็นสายสัญญาณนาฬิกาในตัว ส่วนค่าของข้อมูลจะพิจารณาจากลักษณะของรูปสัญญาณที่ปรากฏบนสายสัญญาณในแต่ละช่องของเวลาซึ่งเรียกว่า ไทม์สล็อต (Time Slot) โดยคาบเวลาต่ำสุดและสูงสุดของสถานะต่าง ๆ ในการสื่อสารข้อมูลในแต่ละไทม์สล็อตมีการกำหนดขอบเขตไว้อย่างชัดเจนการถ่ายทอดข้อมูลจะเกิดขึ้นในแต่ละไทม์สล็อตนั้น รูปแบบการถ่ายทอดข้อมูลจะเป็นแบบอะซิงโครนัสในระดับบิต ไม่มีการกำหนดความยาวของข้อมูลเป็นระดับไบต์ ระบบสื่อสารแบบนี้เหมาะที่จะใช้ในการสื่อสารข้อมูลระหว่างไอซีแผงวงจรเดียวกัน

การอินเตอร์เฟสผ่านสายเส้นเดียว การเชื่อมต่อหรือการอินเตอร์เฟส (Interface) ระหว่างไมโครคอนโทรลเลอร์กับอุปกรณ์ภายนอก โดยใช้จำนวนสายสัญญาณให้น้อยที่สุดได้มีการพัฒนามาอย่างต่อเนื่องจากหลายบริษัท ผู้ผลิต เช่น การเชื่อมต่ออุปกรณ์ต่อพ่วงแบบอนุกรม (Serial Peripheral Interface, SPI) ในไมโครคอนโทรลเลอร์ 68HC1 ของโมโตโรลา การเชื่อมต่อแบบ SPI นี้ช่วยให้ ไมโครคอนโทรลเลอร์แลกเปลี่ยนข้อมูลกับอุปกรณ์ต่อพ่วงได้ด้วยความเร็วถึง 1 ล้านบิตต่อวินาที โดยใช้สายรับส่งสัญญาณเพียง 4 เส้น รวมกับสายกราวด์อีกหนึ่งเส้น

คุณสมบัติ

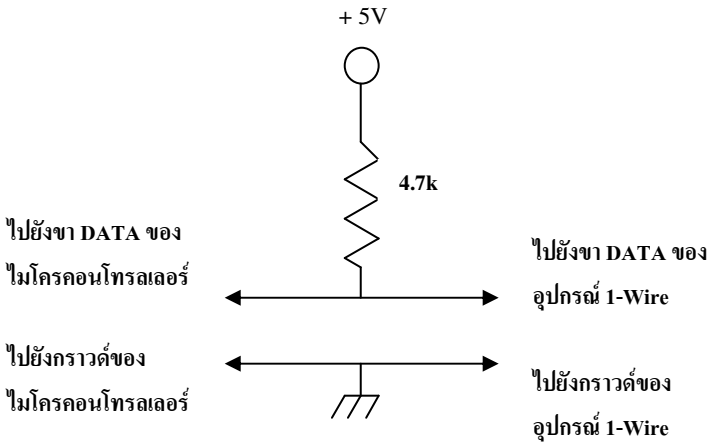
DS1820 สามารถ Interface โดยใช้สายสัญญาณเพียงเส้นเดียว

DS1820 เพียงตัวเดียว สามารถวัดอุณหภูมิได้โดยไม่ต้องต่ออุปกรณ์ร่วม

DS1820 มีย่านวัดอยู่ที่ -55 ถึง +125˚C

DS1820 มีความละเอียดในการวัดได้ 0.5˚C

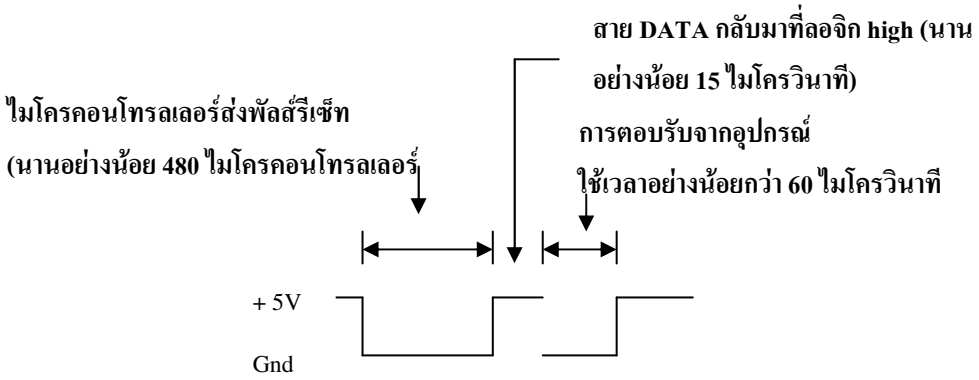
อุปกรณ์ที่สนับสนุนระบบบัสเพียงเส้นเดียวจะมีสายสัญญาณเพียง 2 เส้นเท่านั้นคือสายกราวด์และสายสัญญาณ ซึ่งเรียกอีกอย่างว่า สาย DATA สายนี้จะจัดการเกี่ยวกับทั้งสัญญาณข้อมูลและสัญญาณนาฬิกาที่ใช้ในการแลกเปลี่ยนข้อมูล สาย DATA นี้จะเป็นชนิด Open Drain ดังนั้นในการออกแบบวงจร จะต้องออกแบบให้มีตัวต้านทานมาพูลอัพสาย DATA นี้ด้วยให้ดูรูปแผนผังแสดงการต่อระบบบัสของ 1 – Wire Bus ดังภาพที่ 8



ภาพที่ 9 แผนผังของระบบบัสแบบ 1- Wire Bus

จากภาพที่ 9 เป็นการแสดงแผนผังของระบบบัสแบบ1-Wire จากการแลกเปลี่ยนข้อมูลระหว่างไมโครคอนโทรลเลอร์กับอุปกรณ์ที่ใช้ 1 – Wire Bus นี้ไม่ได้ง่าย ๆ เหมือนกับการส่งข้อมูลผ่านทางบัสแบบ SPI เพราะในระบบ 1 – Wire Bus นั้นสาย DATA จะต้องจัดการเกี่ยวกับจังหวะเวลา (Timing) ระดับสัญญาณ (Level) และทิศทาง (Direction)ของข้อมูลทั้งหมด ทำให้การเขียนโปรแกรมของไมโครคอนโทรลเลอร์ที่ติดต่อกับอุปกรณ์พวกนี้ต้องมีความซับซ้อนมากขึ้น ในสภาวะหยุดนิ่ง (Quiescent State) อุปกรณ์ที่ใช้บัสแบบ 1 – Wire Bus จะทำให้สาย DATA อยู่ในสภาวะลอย (float) ทำให้ขานี้มีแรงดันเท่ากับแรงดันพูลอัพ (Vcc) ซึ่งปกติก็คือ 5 โวลต์นั้นเอง ส่วนไมโครคอนโทรลเลอร์ก็จะปล่อยให้ขาเอาต์พุตที่ใช้ติดต่อกับขา DATA นี้อยู่ในสภาวะ ความต้านทานสูง (High-Impedance State) เช่นกัน เมื่ออุปกรณ์ทั้งสองชนิดนี้ปล่อยให้ขา DATA อยู่ในสภาวะลอยแล้วนี้ความต้านทานพูลอัพก็จะช่วยรักษาระดับแรงดันไฟเลี้ยงที่จ่ายให้กับอุปกรณ์ที่ใช้บัสแบบ 1 – Wire Bus นี้ได้อย่างสม่ำเสมอเพราะอุปกรณ์ที่ใช้บัสแบบ 1 – Wire Bus นี้จะใช้พลังงานในการทำงานน้อยมาก ในการแลกเปลี่ยนข้อมูลกันไมโครคอนโทรลเลอร์และอุปกรณ์ที่

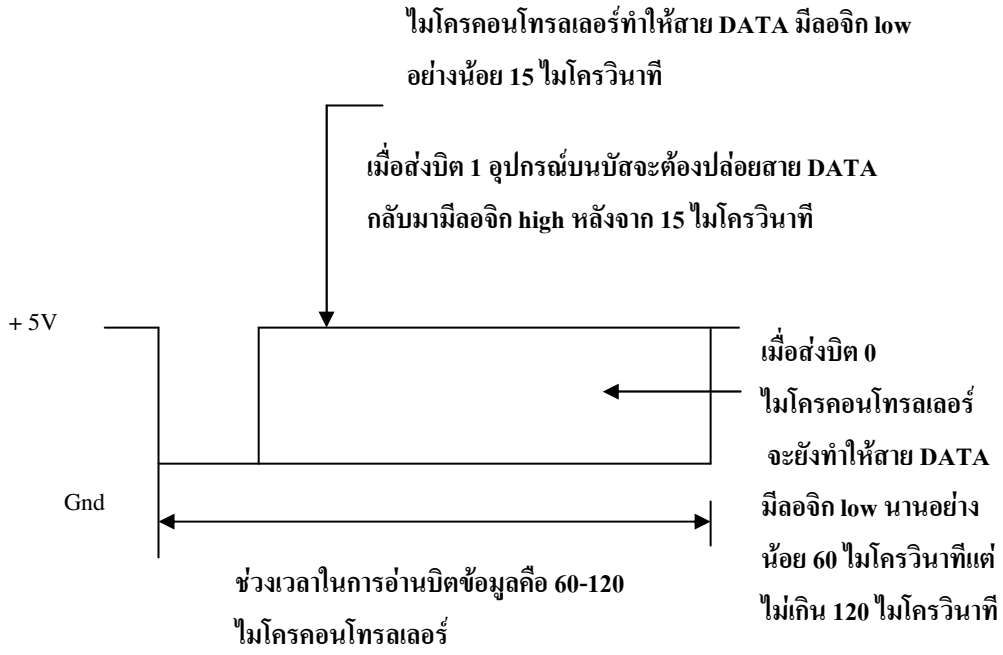
ใช้บัสแบบ 1 – Wire Bus จะต้องดำเนินการอย่างระมัดระวังตามลำดับขั้นตอนในการทำให้สาย DATA มีลอจิกเป็น 0 ปล่อยให้สาย DATA กลับมามีลอจิกเป็น1 และตรวจจับการตอบรับกับอุปกรณ์อีกด้านหนึ่ง ช่วงจังหวะเวลาที่ใช้ในกระบวนการนี้จะถูกกำหนดโดยข้อกำหนดเฉพาะของระบบ 1 – Wire Bus นี้ซึ่งต้องใช้ไมโครคอนโทรลเลอร์ที่มีการตอบสนองได้อย่างรวดเร็วด้วย ดังนั้นเราจะต้องตรวจสอบความสามารถของระบบให้ดีเสียก่อนที่จะใช้ระบบบัสแบบ 1 – Wire Bus นี้ จากสภาวะพักข้างต้นการแลกเปลี่ยนข้อมูลจะเริ่มขึ้นด้วยการที่ไมโครคอนโทรลเลอร์กระทำกระบวนการรีเซ็ต (Reset Sequence) ดังภาพที่ 10



ภาพที่ 10 จังหวะเวลาในการทำกระบวนการตรวจสอบว่ามีอุปกรณ์อยู่บนบัส

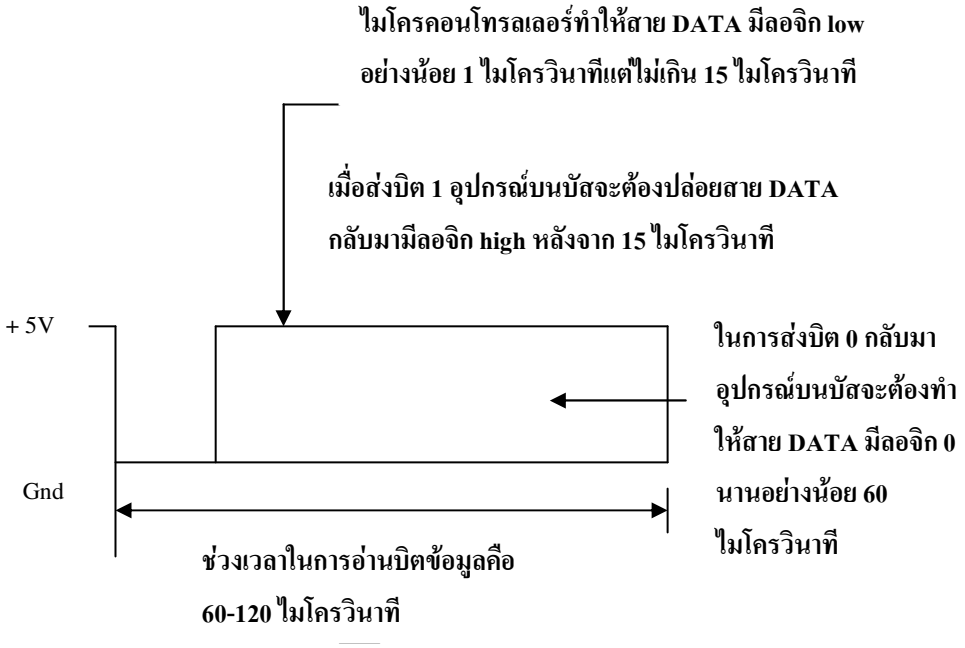
เมื่ออุปกรณ์ที่ใช้บัสแบบ 1 – Wire Bus นี้ตรวจพบสภาวะ RESET นี้ มันก็จะตอบสนองด้วยการส่งพัลส์กลับไปเพื่อบอกให้ไมโครคอนโทรลเลอร์รู้ว่าบนสายบัสนี้มีอุปกรณ์แบบ 1 – Wire Device กำลังทำงานอยู่ โดย อุปกรณ์แบบ 1 – Wire Device จะปล่อยให้สาย DATA อยู่ในลอจิก 1 อย่างน้อย 15 ไมโครวินาที แต่ไม่เกิน 60 ไมโครวินาที จากนั้นมันก็จะให้สาย DATA ลงมามีลอจิกเป็น 0 ในช่วงเวลาประมาณ 60-240 ไมโครวินาที แล้วจึงปล่อยให้กลับไปมีลอจิก ‘high’เช่นเดิม ช่วงเวลานี้มีชื่อเรียกกันหลายชื่อเช่น ช่วงเวลาเริ่มติดต่อ (Initialization) หลังจากอุปกรณ์ 1 – Wire Device ปล่อยให้สาย DATA กลับมาอยู่ในลอจิก 1 แล้ว ไมโครคอนโทรลเลอร์จะต้องปล่อยให้สาย DATA อยู่ในลอจิกนี้นานอย่างน้อย 240 ไมโครวินาที ต่อไปจากนั้นไมโครคอนโทรลเลอร์ก็จะส่งคำสั่งเริ่มต้น (Initial Command) ขนาด 1 ไบต์ไปยังอุปกรณ์ 1 – Wire Bus ซึ่งคำสั่งนี้อาจจะเป็นคำสั่งอะไรก็ได้ ไมโครคอนโทรลเลอร์จะส่งบิตของ คำสั่งนั้นออกไป โดยการเปลี่ยนสถานะของสาย DATA กลับไปกลับมาโดยตอนแรกจะให้เป็น ลอจิก 0 แล้วจึงกลับมาเป็น 1 ตามช่วงจังหวะเวลาที่เหมาะสม ช่วงเวลานานที่ ไมโครคอนโทรลเลอร์ทำให้

สาย DATA มีลอจิก 0 จะเป็นตัวแยกแยะว่าบิตไหนที่มีลอจิกเป็น ‘1’ บิตไหนมีลอจิกเป็น ‘0’ ให้ดูแผนภูมิเวลาในการเขียนบิต ‘1’ หรือ ‘0’ ดังภาพที่ 11



ภาพที่ 11 จังหวะเวลาที่ไมโครคอนโทรลเลอร์ใช้เขียนข้อมูล ‘0’ หรือ ’1’ไปยังอุปกรณ์บนบัส

จากภาพที่ 11 เป็นจังหวะเวลาที่ไมโครคอนโทรลเลอร์ใช้เขียนข้อมูล ‘0’ หรือ ’1’ไปยังอุปกรณ์บนบัสช่วงเวลาระหว่าง 15-120 ไมโครวินาที หลังไมโครคอนโทรลเลอร์ทำให้สาย DATA มี ลอจิก 0ในตอนนี้จะเรียกว่าช่วงการอ่านสถานะบิตข้อมูล (Sampling Window)



ภาพที่ 12 จังหวะเวลาที่ไมโครคอนโทรลเลอร์ใช้อ่านบิตข้อมูล ‘0’ หรือ ’1’ จากอุปกรณ์บนบัส

จากภาพที่ 12 แสดงจังหวะเวลาที่ไมโครคอนโทรลเลอร์ใช้อ่านบิตข้อมูล ‘0’ หรือ ’1’ จากอุปกรณ์บนบัสในการอ่านบิตข้อมูลจากบัสแบบ 1-Wire Bus ตอนแรกไมโครคอนโทรลเลอร์จะต้องทำให้สาย DATA มีลอจิก 0 เป็นเวลานานไม่เกิน 15 ไมโครวินาทีแล้วจึงปล่อยให้สาย DATA กลับมามีลอจิก 1 เช่นเดิมจากนั้นอุปกรณ์ 1 – Wire Device ก็จะเข้าควบคุมสาย DATA แทนโดยจะส่งบิต ‘0’ โดยการทำให้สาย DATA มีลอจิกเป็น 0 และส่งบิต ‘1’ โดยการปล่อยให้สาย DATA กลับมามีลอจิก 1 ตามเดิม เมื่อส่งข้อมูลออกไปยังไมโครคอนโทรลเลอร์เรียบร้อยแล้ว จะมีการพักอยู่ชั่วขณะหนึ่ง จากนั้นอุปกรณ์ 1 – Wire Device จะปล่อยการควบคุมจากสาย DATA ให้เป็นอิสระ และรอรับ คำสั่งการอ่านของข้อมูลครั้งต่อไป ถ้าอุปกรณ์ 1 – Wire Device ส่งบิต ‘0’ ออกไป ไมโครคอนโทรลเลอร์ก็จะสามารถตรวจสอบจุดสิ้นสุดของลอจิก ‘0’ นี้ง่าย เพราะสาย DATA จะกลับมาอยู่ที่ลอจิก 1 ตามเดิม แต่ถ้าตรวจสอบจุดสิ้นสุดของการส่งบิต ‘1’ ของอุปกรณ์

1 – Wire Device จะต้องใช้เทคนิคมากกว่านี้ เพราะสาย DATA จะอยู่ที่ลอจิก 1 อยู่แล้วนี่ก็เป็นเหตุผลว่าทำไมจังหวะเวลาในการอ่านเขียนข้อมูลจึงเป็นเรื่องที่สำคัญมาก เมื่อจะอ่านบิต ‘1’ จากอุปกรณ์ 1 – Wire Device ไมโครคอนโทรลเลอร์จะต้องทำตามช่วงเวลาที่แสดงในแผนภูมิเวลาอย่างเคร่งครัด และต้องไม่ทำการอ่านลอจิกของบิตถัดมา

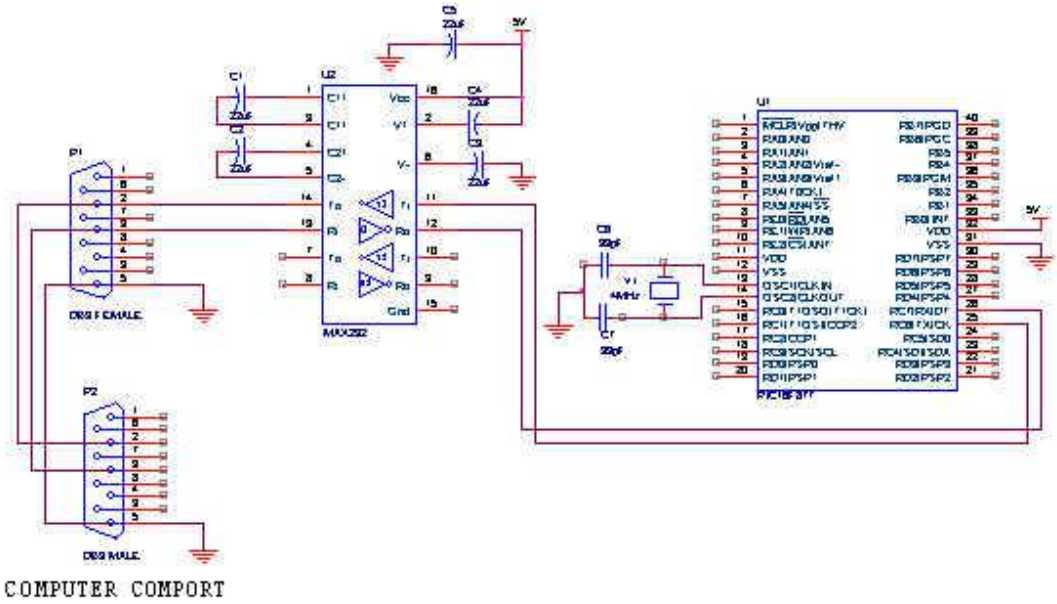
ภาคควบคุมสถานะเปิด – ปิดของสวิทช์

ชุดควบคุมสถานะ การเปิด - ปิดของสวิทช์จะใช้รีเลย์ 12 VDC โดยจะใช้หน้าสัมผัสของรีเลย์มาเป็นตัวกำหนดการ เปิด - ปิดของสวิทช์ โดยควบคุมด้วยชุดไมโครคอนโทรลเลอร์ จะทำหน้าที่ควบคุมสถานะ และ ส่งค่าสถานะ ให้กับคอมพิวเตอร์

เมื่อคำสั่งควบคุมสถานะถูกส่งมาจากเครื่องคอมพิวเตอร์มาที่ไอซี PIC 16F877 ไอซีจะทำการประมวลผลโดยตรวจสอบ หากบิตที่ 9 – 16 เป็นเลข “3” จะส่งสัญญาณออกที่ขา 39(P0.0) เพื่อทำการตัดกระแสที่จ่ายให้กับทรานซิสเตอร์Q1 มีผลทำให้รีเลย์ตัวที่ 1อยู่ในสถานะ ปิด หากบิตที่ 9 – 16 เป็นเลข “4” จะส่งสัญญาณออกที่ขา 39(P0.0) เพื่อทำการส่งกระแสที่จ่ายให้กับทรานซิสเตอร์Q1 มีผลทำให้รีเลย์ตัวที่ 1อยู่ในสถานะ เปิด และ หากบิตที่ 17 – 24 เป็นเลข “3” จะส่งสัญญาณออกที่ขา 38(P0.1) เพื่อทำการตัดกระแสที่จ่ายให้กับทรานซิสเตอร์Q2 มีผลทำให้รีเลย์ตัวที่ 2อยู่ในสถานะ ปิด หากบิตที่ 17 – 24เป็นเลข “4” จะส่งสัญญาณออกที่ขา 38(P0.1) เพื่อทำการส่งกระแสที่จ่ายให้กับทรานซิสเตอร์Q2 มีผลทำให้รีเลย์ตัวที่ 2อยู่ในสถานะเปิดเมื่อแพ็คเก็จคำสั่งเมื่อแพ็คเก็จคำสั่งตรวจสอบ สถานะ(C01) ถูกส่งมาจากเครื่องคอมพิวเตอร์มาที่ไอซี16F877 ไอซีจะทำการประมวลผลโดยตรวจสอบสถานะปัจจุบันของสวิทช์โดยชุดไมโครคอนโทรเลอร์จะนำสัญญาณที่รับมาเข้ารหัสเพื่อส่งค่าข้อมูลเป็นแพ็คเก็จโดยจะมีการเข้ารหัสเก็จข้อมูล 24 บิตโดยข้อมูล 8 บิตแรก จะเป็นเฮดเดอร์ (Header) เป็นตัวอักษร”C” และข้อมูลสถานะ (Data) เป็นเลขฐาน 16 จำนวน 16 บิต หาก สวิตช์ 1 อยู่ในสถานะปิด ข้อมูลสถานะ 8 บิตแรกจะเป็นเลข”1” และสถานะเปิด ข้อมูลสถานะ 8 บิตแรกจะเป็นเลข “2” หาก สวิตช์ 2 อยู่ในสถานะ ปิด ข้อมูลสถานะ 8 บิตหลังจะเป็นเลข”1” และ สถานะเปิด ข้อมูลสถานะ 8 บิตหลังจะเป็นเลข “2” ชุดไมโครคอนโทรเลอร์จะนำสัญญาณที่รับมาเข้ารหัสนี้ให้กับชุดแปลงสัญญาณตามมาตรฐาน RS232 เพื่อส่งข้อมูลเข้าเครื่องคอมพิวเตอร์ต่อไป ดังแสดงในรูปที่17 แสดงวงจรควบคุม

Asynchronous serial interface ระหว่าง PC กับ PIC16F877

เป็นเรื่องของการใช้งาน serial communication แบบ RS232 ระหว่าง PIC กับ computer ก่อนอื่นดูรูปแบบการ interface ก่อนว่ามันจะมีรูปแบบอย่างไรในการทดลองดังรูปที่ 12



ภาพที่ 13 แสดง Asynchronous serial interface ระหว่าง PC กับ PIC16F877ส่วน serial interface

ในการเชื่อมต่อระหว่าง PIC กับ COMPUTER จะมี 2 ข้อที่ต้องพึงระวังก็คือ ขา RX ของ PIC จะต่อกับขา TX ของ COMPUTER และขา TX ของ PIC จะต่อกับขา RX ของ COMPUTER โดยผ่าน MAX232 เพราะว่า สัญญาณ ที่ PIC สร้างออกมาจะเป็นแบบ 0 ถึง 5 V ดังนั้นต้องให้สัญญาณดังกล่าว ผ่าน MAX232 เพื่อแปลงเป็นสัญญาณแบบ RS232 ส่วนขา GND ของ PIC ก็ต่อกับขา GND ของ COM PORT ของ COMPUTER

โปรโตคอลในการติดต่อระหว่างอุปกรณ์ต่าง ๆ

ในการรับและส่งข้อมูลระหว่างเครื่องคอมพิวเตอร์และไมโครคอนโทรเลอร์จะกระทำผ่านทาง Serial port ซึ่งเป็นการส่งเป็นเฟรมข้อมูล โดยการส่งจะส่งที่ละ 5 bytes ทั้งที่ส่งจากเครื่องคอมพิวเตอร์ไปไมโครคอนโทรเลอร์และที่ส่งจากไมโครคอนโทรเลอร์ไปเครื่องคอมพิวเตอร์

รูปแบบการส่งจากไมโครคอนโทรเลอร์ไปเครื่องคอมพิวเตอร์

เราจะทำการกำหนดข้อมูลที่จะส่งออกไป 5 bytes นั้นโดยการแบ่งออกเป็น block และกำหนดให้แต่ละ block นั้นมีขนาด 1 byte จากนั้นก็จะส่งข้อมูลออกไปตามแล้วแต่คำสั่งที่ได้กำหนดไว้ในการเลือกเมนูในการส่ง แล้วจึงรอการตอบกลับมาจากเครื่องคอมพิวเตอร์ เวลารับข้อมูลเข้ามาก็จะรับเป็นเฟรมทั้งหมด 5 bytesจากนั้นจะทำการตัดเอาเฉพาะข้อมูลที่ต้องการแสดงผลที่หน้าจอของไมโครคอนโทรเลอร์

รูปแบบการส่งจากเครื่องคอมพิวเตอร์ไปไมโครคอนโทรเลอร์

เมื่อมีการส่งข้อมูลมายังเครื่องคอมพิวเตอร์ เราจะทำการแบ่งข้อมูลต่างๆออกเป็นส่วนๆ โดยตอนแรกเราจะทำการตรวจสอบ Byte แรกว่ามีค่าตรงกับเท่าไหร่ ถ้าตรงกับค่าไหนที่เราได้กำหนดไว้ก็จะทำตามขั้นตอนของชุดคำสั่งนั้นๆ เช่น ถ้า byte แรกที่รับมามีค่าตรงกับค่า 01 เราก็จะรู้ได้ว่ามีการสั่งงาน และเราจะนำเอาข้อมูลแต่ละ byte มาทำการเก็บลงในตารางฐานข้อมูล ซึ่งเราจะทำการสร้างฐานข้อมูล 2 ตารางเพื่อสำหรับไว้เก็บข้อมูลที่ได้รับมาจากไมโครคอนโทรเลอร์ ในการจัดการเกี่ยวกับการติดต่อ การนำข้อมูลไปเก็บไว้ในฐานข้อมูลเราจะใช้ visual basic เป็นตัวจัดการรวมทั้งการส่งข้อมูลไปยังไมโครคอนโทรเลอร์ด้วย

โปรโตคอลที่ใช้ติดต่อกับโมเด็ม

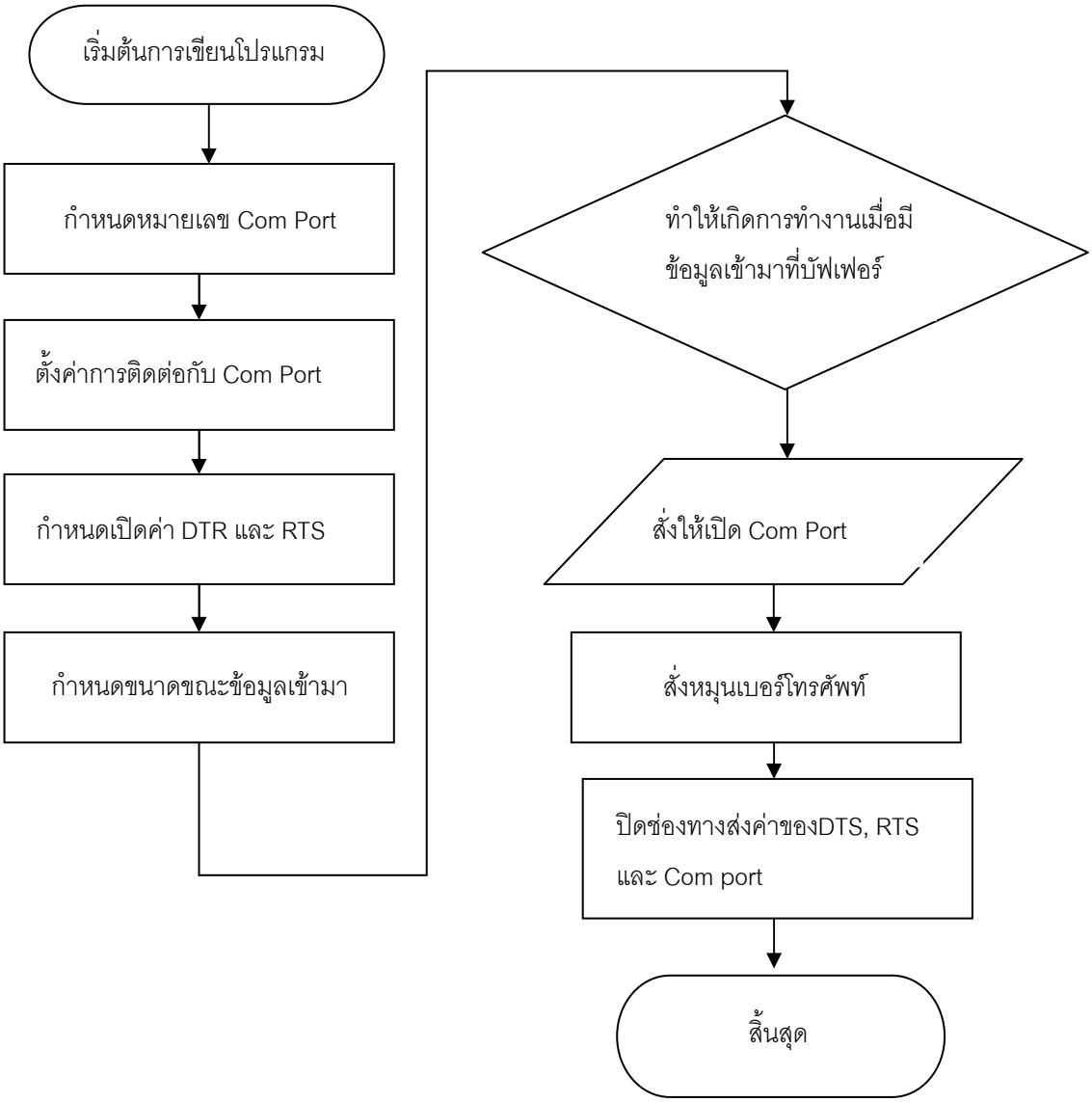
สำหรับการที่เราจะติดต่อกับโมเด็มได้นั้นจะต้องมีคำสั่งตามมาตรฐานโรงงานที่ผลิตโมเด็มได้ตกลงและ กำหนดขึ้นมาใช้ เป็นโปรโตคอล(Protocal) ซึ่งเรียกกันว่า AT Command เพื่อที่จะสามารถส่งคำสั่ง สำหรับการควบคุมหรือรับข้อมูลจากโมเด็ม อาจจะมีบ้างคำสั่งที่อาจต่างกัน ทั้งนี้สามารถดูรูปแบบคำสั่งได้จากคู่มือของโมเด็มแต่ละรุ่นที่เราซื้อมาได้ สำหรับคำสั่งที่จะส่งไปควบคุมติดต่อที่โมเด็มที่เราจะเขียนเป็นโปรแกรมต่อไปนี้ มี AT Command ที่ต้องใช้

ตารางที่ 1 คำสั่ง AT Command สำหรับการ Dial-Up

AT Command	**การทำงานของคำสั่ง**
ATZ	รีเซทโมเด็ม
ATHn	ตั้งค่าโมเด็มใน Hook Mode n คือ 0 = off ,1= on
ATDTn	ส่งหมายเลขโทรศัพท์ให้โมเด็มเพื่อที่จะต่อกับสัญญาณปลายทางที่เรียก n คือหมายเลขโทรศัพท์

จากตารางที่ 1 แสดงคำสั่งที่สำคัญที่ใช้ในการเชื่อมต่อกับโมเด็ม

ขั้นตอนเขียนโปรแกรมเพื่อติดต่อกับโมเด็ม



ภาพที่ 14 แสดงขั้นตอนการเขียนโปรแกรมเพื่อติดต่อกับโมเด็ม

ลำดับการเขียนโปรแกรม

1. กำหนดหมายเลข Com Port ที่ใช้ติดต่อกับโมเด็ม (Detect)กับ Com Port

MSComm1.ComPort=numPort

2. ตั้งค่าการติดต่อกับ Com Port คือ อัตราการส่งข้อมูล

(Baud Rate), Parity, Data (จำนวนบิต), Stop Bit

MSComm1.Settings="28800, N, 8, 1"

3. กำหนดเปิดค่าของ DTR และ RTS

MSComm1.DTREnable = True

MSComm1.RTSEnable = True

4. กำหนดขนาดขณะที่ข้อมูลเข้ามา ให้ไปอ่านข้อมูลทั้งหมดที่อยู่ในบัฟเฟอร์

MSComm1.InputLen=1

5. ทำให้เกิดการทำงานด้วย Event-driven เมื่อมีข้อมูลเข้ามาที่บัฟเฟอร์รับข้อมูล จึงทำให้เกิด

CommEvent ใน OnComm Event

MSComm1.Rthreshold=1

6. สั่งให้เปิด Com port

MSComm1.OpenPort=True

7. สั่งหมุนเบอร์โทรศัพท์ (Phone Dial-Up) ด้วย Command AT ตามด้วยหมายเลขเบอร์โทรศัพท์ และเชื่อมต่อด้วยฟังก์ชั่นเพื่อส่งคำสั่ง Enter ทั้งนี้ในเครื่องที่ใช้เชื่อมประโยค จะเป็นเครื่องหมายบวก (+) หรือเครื่องหมาย & ก็ได้ ส่วนฟังก์ชั่นที่ทำหน้านี้เหมือน Chr$(13) กันมีดังนี้

+ Chr$(10) + vbCr & Chr$(10) & Chr$(13) & vbCrLf

+ Chr$(13) + vbCrLf & vbCr & Chr$(13)

คำสั่งตัวอย่างในการติดต่อกับModem

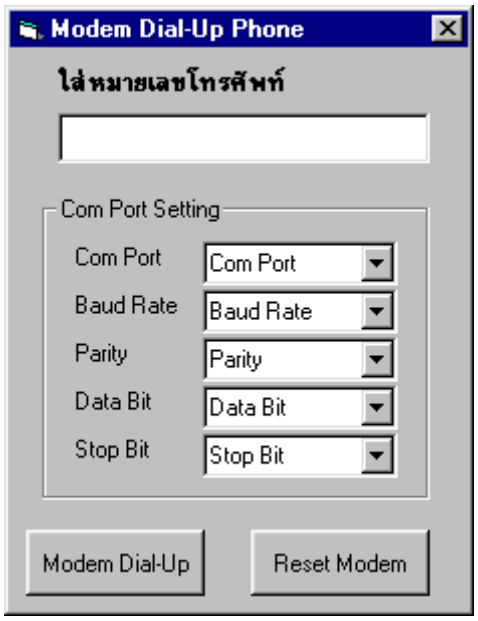
MSComm1.Output=”ATDT1234567"+Chr(13)

8. ปิดช่องทางส่งค่าของ DTS ,RTS และ Com port

MSComm1.DTREnable = True

```
MSComm1.RTSEnable = True
MSComm1.OpenPort=False
```

ตัวอย่างโปรแกรมติดต่อ Modem



โปรแกรม สั่งหมุนเบอร์โทรศัพท์

```
Sub Modem_Dial_Up()
  Dim valStartTime
' กำหนดค่าต่างๆเพื่อติดต่อ Port กับ โมเด็ม
  'MSComm1.CommPort = 1
  MSComm1.Settings ="28800,N,8,1"
  If Not MSComm1.PortOpen Then MSComm1.PortOpen = True
  MSComm1.DTREnable = True
  MSComm1.RTSEnable = True
  MSComm1.RThreshold = 1
  MSComm1.InputLen = 1
' ส่ง AT Command เพื่อติดต่อกับโมเด็มและหมุนเบอร์โทรศัพท์
   MSComm1.Output = "ATH1" & Chr$(13) ' สั่งเปิด Hook Mode
   MSComm1.Output = "ATDT" & txtPhoneNum & Chr$(13) ' สั่งหมุนเบอร์โทรศัพท์
```

```
'หน่วงเวลารอสัญญาณที่ส่งไปให้โมเด็มทำงาน
  valStartTime = Timer
  While Timer < valStartTime + 5    ' หน่วงเวลา 5 วินาที
    DoEvents
  Wend
  MSComm1.Output = "ATH0" & Chr$(13) ' สั่งปิด Hook Mode
  MSComm1.DTREnable = False
  MSComm1.RTSEnable = False
  MSComm1.PortOpen = False
End Sub
```

โปรแกรม Reset Modem

```
Sub Reset_Modem()
'กำหนดค่าต่างๆเพื่อติดต่อ Port กับ โมเด็ม
  MSComm1.CommPort =1
  MSComm1.Settings = "28800,N,8,1"
  If Not MSComm1.PortOpen Then MSComm1.PortOpen = True
  MSComm1.DTREnable = True
  MSComm1.RTSEnable = True
  MSComm1.RThreshold = 1
  MSComm1.InputLen = 1
'สั่งให้ Reset โมเด็ม และยกเลิกการติดต่อ
  MSComm1.Output = Chr$(13) & "ATZ" & Chr$(13) ‘คำสั่งให้ Reset โมเด็ม
  MSComm1.Output = "ATH0" & Chr$(13)  ' คำสั่งให้ Hook Mode เป็น Off
  MSComm1.DTREnable = False
  MSComm1.RTSEnable = False
  MSComm1.PortOpen = False
End Sub
```

Visual Basic เขียนโปรแกรมส่งไฟล์ข้อมูลผ่าน Serial Port

โดยจะส่งผ่านทาง Com Port สิ่งแรกที่ต้องมาทำความเข้าใจเกี่ยวกับการเข้าถึงไฟล์ข้อมูลก่อน ใน Visual Basic สามารถเข้าถึงไฟล์ข้อมูลได้ 3 ลักษณะคือ

1. Sequential File หรือ ไฟล์ข้อมูลแบบเรียงลำดับต่อเนื่อง

มีลักษณะการบันทึก และอ่านไฟล์ข้อมูล ที่เราป้อนลงในเครื่องคอมพิวเตอร์เป็นลำดับจากต้นไฟล์จนถึงท้ายไฟล์ข้อมูล(First In First Out) ซึ่งข้อมูลที่นำไปเก็บจะต่อเนื่องกันโดยไม่มีช่องว่าง ทั้งนี้แม้ว่า แต่ละ เรกคอร์ด (Record) จะมีความยาวข้อมูลไม่เท่ากัน
ข้อเสีย ของการบันทึกข้อมูลของไฟล์ประเภทนี้คือเมื่อจะใช้ข้อมูลทุกครั้งจากไฟล์จะต้องเริ่มใช้ตัวแรก เรียงลำดับกันไป หากจะเรียกเจาะจงข้อมูลไม่สามารถทำได้เป็นผลให้ทำงานช้าจึงไม่เหมาะกับการเก็บ ไฟล์ข้อมูลจำนวนเยอะๆ ครับ

2. Random File หรือ ไฟล์แบบสุ่ม

วิธีการจัดการข้อมูลของ Random File นี้สามารถจะเข้าถึงข้อมูลได้รวดเร็วเพราะไม่ต้องมี การเรียงข้อมูลเหมือนอย่าง Sequential File ซึ่งเราสามารถเจาะจงใช้ข้อมูลได้โดยตรงยัง Record นั้นๆ แต่มีข้อดีก็ต้องข้อเสียเล็กน้อยคือการใช้งานจะซับซ้อนกว่า Sequential File

3. Binary File ระบบการจัดการไฟล์แบบนี้ จะทำให้เราสามารถรับส่งข้อมูลเป็นไปอย่างสมบูรณ์ เพราะไฟล์ประเภทนี้จะสามารถแทนข้อมูลในลักษณะใดๆก็ได้ และยังสามารถกำหนดรูปแบบโครง สร้างข้อมูล ตัวแปรและความยาวในการบันทึกข้อมูล ทั้งนี้ขนาดความจุของไฟล์ก็มีขนาดเล็ก

เริ่มด้วยการเปิดไฟล์ข้อมูล

Open pathname for mode [Access access] [lock] as [#] filenumber [Len=reclength]
Pathname คือ ชื่อไฟล์ที่จะใช้งาน
Mode คือ คำสงวนที่ใช้ในเลือกโหมดของไฟล์มีคำว่า Append, Binary, Input, Output, or Random ถ้าไม่ได้กำหนด ค่าปกติจะตั้งไว้สำหรับ โหมด Random
Access คือ คำสงวนใช้ในการกระทำเพื่อเข้าถึงไฟล์ข้อมูลมีคำว่า Read, Write, or Read Write

Lock คือ คำสงวนที่ใช้กำหนดคุณสมบัติของไฟล์ข้อมูล มีคำว่า Shared, Lock Read, Lock Write, and Lock Read Write

Filenumber คือ หมายเลขของไฟล์ข้อมูลที่เราใช้ เปิด อยู่ในช่วง 1-511 ทั้งนี้เราสามารถหาหมายเลขไฟล์ที่ว่างด้วยคำสั่ง FreeFile

Reclength คือ ความยาวในการบันทึกข้อมูลกำหนดได้ถึง 32,767 bytes

ตัวอย่าง

Open "TESTFILE" For Input As #1

Open "TESTFILE" For Binary Access Write As #1

Open "TESTFILE" For Output Shared As #1

ทุกครั้งที่เปิดด้วยคำสั่ง Open แล้วเมื่อไม่มีการใช้การอ่านและเขียนข้อมูลแล้ว

จะต้องปิดไฟล์ข้อมูลด้วยคำสั่ง Close #filenumber

Close #1

การอ่านและเขียนไฟล์ข้อมูล

คำสั่ง Get เรานำมาใช้เพื่อที่จะอ่านข้อมูลจากไฟล์ที่เราเปิดขึ้นมาเก็บไว้ในตัวแปร โดยมีรูปแบบการใช้งานดังนี้

Get [#] filenumber, [renumber], varname

คำสั่ง Put ก็จะนำมาใช้เพื่อที่จะเขียนข้อมูลจากตัวแปรลงสู่ไฟล์ที่เราเปิดขึ้นมา โดยมีรูปแบบการใช้งานดังนี้

Put [#] filenumber, [renumber], varname

Filenumber คือ หมายเลขไฟล์ที่เราเปิดขึ้นใช้อยู่ปัจจุบันในโปรแกรม

Recnumber คือ หมายเลขลำดับที่บันทึกข้อมูลในการเปิดไฟล์แบบ Random mode หรือ หมายเลขที่ของ Byte ในการเปิดไฟล์แบบ Binary mode เพื่อที่จะเริ่มการอ่านหรือการเขียนข้อมูล

Varname คือ ตัวแปรสำหรับเก็บบันทึกข้อมูลที่อ่านได้จากไฟล์

การตรวจสอบสถานะต่างๆของไฟล์

คำสั่ง Seek ที่ต้องนำมาใช้ก็เพื่อให้การเก็บบันทึกข้อมูลลงไฟล์ให้เป็นไปอย่างต่อเนื่อง โดยจะคืนค่าตำแหน่ง ปัจจุบันในการอ่านและเขียนข้อมูลของไฟล์ที่เราเปิดใช้อยู่ ซึ่งมีค่าตั้งแต่ 1 ถึง 2,147,483,647 ส่วนรูปแบบการใช้มีดังนี้

Seek (filenumber)

คำสั่ง LOF จะนำมาใช้เพื่อตรวจสอบขนาดของข้อมูลในไฟล์ที่เราเปิดใช้งานโดยมีรูปแบบดังนี้ LOF (filenumber)

คำสั่งEOFสำหรับตรวจสอบสถานะของข้อมูลว่าสิ้นสุดขอบเขตไฟล์นั้นๆหรือยังถ้าใช่จะคืนค่าเป็นลักษณะ บูลีน (Boolean) เท่ากับ True ถ้ายังไม่สิ้นสุดจะคืนค่า false มีรูปแบบการใช้ดังนี้ EOF (filenumber) คือ หมายเลขไฟล์ที่เราเปิดขึ้นใช้อยู่ปัจจุบันในโปรแกรม

การโปรแกรมเพื่อการรับส่งข้อมูลผ่าน Portการส่งไฟล์ข้อมูล

1. กำหนดตัวแปรเพื่อใช้งานในรูทีนนี้
2. หาหมายเลขไฟล์ที่ว่างเพื่อเปิดใช้งานด้วยคำสั่ง Free File แล้วเก็บไว้ในตัวแปร no Send
3. ใช้คำสั่งเปิดไฟล์แบบ ไบนาลี่ เพื่ออ่านข้อมูล
4. อ่านข้อมูลจากไฟล์ที่เปิด โดยให้ข้อมูลแต่ละชุด มีจำนวนเท่ากับหรือไม่เกินขนาดของ OutBufferSize คือ การกำหนด Buffer ในการส่งข้อมูลออก ใน MSComm1 ซึ่งจะเป็น Buffer สำหรับส่งข้อมูลออกผ่านทาง com port
    - เมื่ออ่านข้อมูลจากไฟล์เป็น Block ข้อมูลเรียบร้อยแล้วก็จะถูกส่งด้วยคำสั่ง MSComm1.Output
    - รอจนกว่าข้อมูลจะถูกส่งออกไปจนหมดจาก Buffer โดยใช้การตรวจสอบด้วยคำสั่ง MSComm1.OutBufferCount
5. ปิดไฟล์ข้อมูลตามหมายเลขที่เปิดขึ้นมาใช้

การรับไฟล์ข้อมูล

1. ทำการหาไฟล์ที่ว่างและเปิดไฟล์ที่เราจะเก็บข้อมูลที่รับเข้ามาลักษณะของการบันทึกไฟล์ ข้อมูลใหม่
2. ตรวจสอบข้อผิดพลาดขณะเปิดไฟล์

3. ใช้คำสั่ง Seek หาข้อมูลปัจจุบันเพื่อเพิ่มต่อข้อมูลของไฟล์ที่เปิดรับข้อมูลอยู่
คำสั่งส่วนนี้จะอยู่ในโพซีเดอร์แบบ Menu หรือ Command Button ซึ่งเป็นสัญลักษณ์ในการควบคุม ก็ได้

ตัวอย่างการใช้งานโปรแกรมรับส่งไฟล์ข้อมูล

เราจะประกาศตัวแปร NumRev ไว้ที่ส่วนหัวของโปรแกรม เพื่อเก็บค่าของ FreeFile

```
Private Sub Command1_Click ()
NumRev = FreeFile
Open “D:\ test.dat” For Binary Access Write As numRev
  If Err Then
     MsgBox Error$, 48
     Close numRev
     NumRev = 0
     Exit Sub
     Else
     Seek numRev, LOF (numRev) + 1
  End If
End Sub
```

สำหรับการรับไฟล์ข้อมูลที่ถูกส่งมานั้น เราจะต้องมาเขียนโปรแกรมในเหตุการณ์(Event) ของ MSComm เมื่อรับข้อมูลเข้ามา ซึ่งใช้ Event ชื่อ comEvReceive ในโปรซีเดอร์
Sub MSComm1_OnComm () ทั้งนี้เราจะใช้เงื่อนไข Select Case เข้ามาใช้ตรวจสอบเมื่อเกิด Events ของการรับข้อมูล
ซึ่งชุดคำสั่งส่วนนี้จะนำไปไว้ในโพซีเดอร์ MSComm1_OnComm ()

```
Private Static Sub MSComm1_OnComm ()
   Select Case MSComm1.CommEvent
   Case comEvReceive
   Dim buffRev As Variant
   buffRev = MSComm1.Input
      If numRev Then
         i = 3
```

```
        Do
            Err = 0
            'เก็บข้อมูลลงไฟล์ที่เปิดไว้แล้ว
            Put numRev, bffRev
            If Err Then
                i = MsgBox (Error$, 21)
                If i = 1 Then
                    Close numRev 'ปิดไฟล์ข้อมูล
                End If
            End If
        Loop While i <> 3
    End If
End Select
```

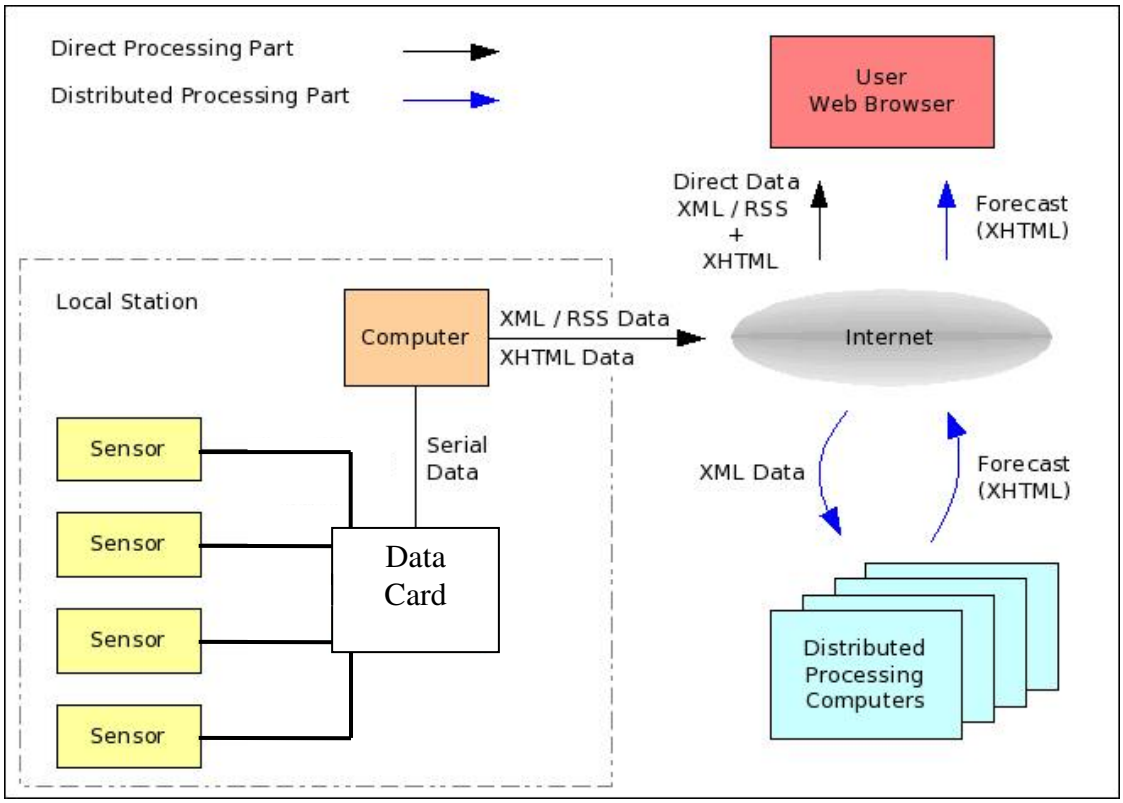
อุปกรณ์และวิธีการ

อุปกรณ์

1. เครื่องคอมพิวเตอร์ Notebook Pentium 1.6 GHz, หน่วยความจำหลัก 512 MB
2. แหล่งจ่ายไฟ 12 V 500mA
3. โปรแกรม Visual Basic 6, ภาษา C
4. บอร์ดไมโครคอนโทรลเลอร์ PIC 16F877
5. หลอดไฟ, สายไฟ, โมเด็ม

วิธีการ

ในงานวิจัยนี้จะเริ่มจากการศึกษาการทำงานของไมโครคอนโทรลเลอร์ จากนั้นจึงทำการศึกษาวงจรควบคุมอุปกรณ์ไฟฟ้าในที่นี้คือหลอดไฟ โดยใช้ไมโครคอนโทรลเลอร์เป็นตัวควบคุมแล้วศึกษาวิธีการเชื่อมต่อระหว่างวงจรควบคุมอุปกรณ์ไฟฟ้ากับคอมพิวเตอร์โดยผ่านทางซีเรียลพอร์ตโดยใช้โปรแกรม Visual Basic 6 ในการควบคุมการทำงานจากนั้นจึงศึกษาวิธีการสร้างเวปไซด์ที่ใช้ในการสั่งการควบคุมแล้วจึงทำการศึกษาวิธีการเชื่อมต่อระหว่างเวปไซด์กับคอมพิวเตอร์ แล้วทำการทดลองและสรุปผลหลังจากนั้นนำผลที่ได้ส่งข้อมูลไปยังเวปไซด์เพื่อทำการทดสอบกับคอนโทรลเลอร์บน Visual Basic 6 เพื่อดูว่าสัญญาณที่ได้นี้สามารถทำงานได้ตามที่ต้องการ



ภาพที่ 15 แสดงหลักการทำงานของระบบ

จากภาพที่ 15 แสดงหลักการสั่งงานและควบคุมของระบบโดยรวม

1. ไมโครคอนโทรลเลอร์

1.1 การกำหนดค่าเริ่มต้นก่อนการกำหนดหน้าที่การทำงานของอุปกรณ์

เริ่มต้นเลือกโปรแกรมที่ใช้ในการ Simulate เพื่อทดสอบผลของสัญญาณ โดยจะใช้ภาษาซี ในการเขียนโปรแกรมควบคุมลงไมโครคอนโทรลเลอร์ในส่วนแรกต้องเข้าใจและทราบโดยคร่าวๆ ว่าโปรแกรมที่จะเขียนขึ้นนั้นจะประกอบด้วยวงจรอะไรบ้าง เช่น ประกอบด้วยวงจรควบคุมหลอดไฟ วงจรควบคุมอุณหภูมิ วงจรตรวจเช็คการทำงาน เพื่อที่จะได้สามารถคำนวณได้คร่าวๆ ว่าจะใช้อินพุต, เอาพุตจำนวนเท่าไรในการเขียนโปรแกรมควบคุมที่ได้มาลงอยู่ในคอนโทรเลอร์นี้

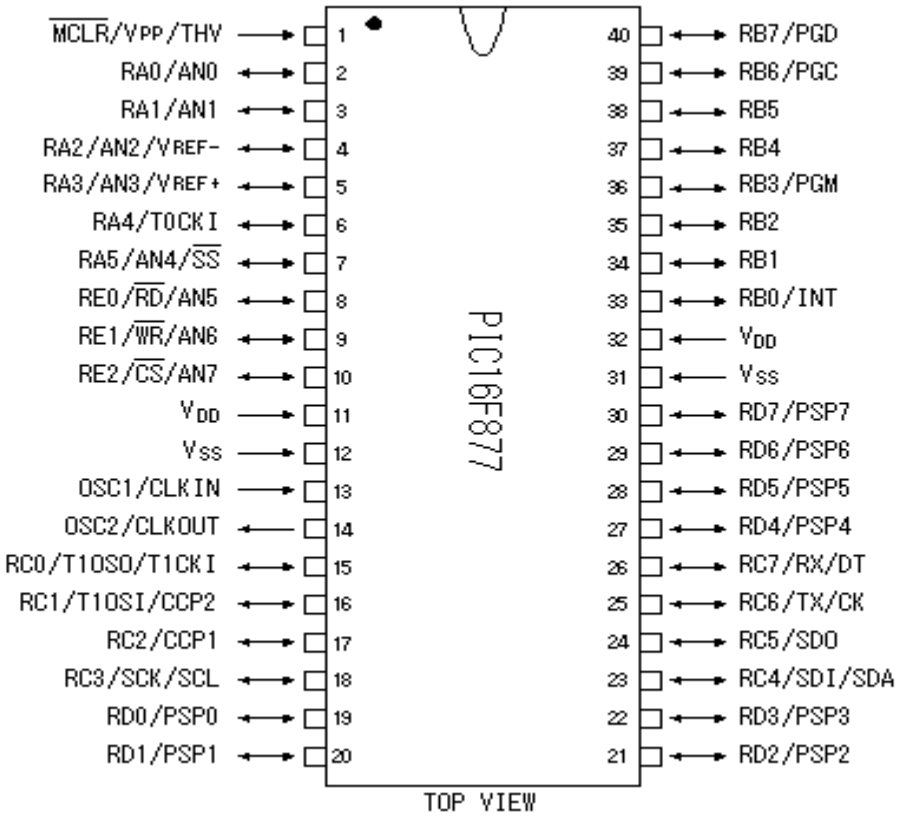
อีกเหตุผลหนึ่งคือเพื่อให้ง่ายต่อการออกแบบและเขียนโปรแกรมดังนั้นในการเขียนโปรแกรมทดสอบต้องทำการกำหนดอินพุต, เอาต์พุตตามที่คอนโทรลเลอร์นั้นสามารถสร้างได้ จึงสามารถจะทดสอบโปรแกรมนั้นได้

1.2 การกำหนดอินพุต, เอาต์พุต

ตามที่ได้กล่าวข้างต้นว่า วงจรที่ใช้ในการควบคุมอุปกรณ์ไฟฟ้าจะต้องทำการกำหนดให้อยู่ในรูปแบบควบคุมที่แน่นอน เนื่องจากการกำหนดขาของไมโครคอลโทรลเลอร์นั้นต้องดูหลักการทำงานของแต่ละขาของไมโครคอนโทรลเลอร์ว่าแต่ละขาทำหน้าที่อะไรและแตกต่างอย่างไรจึงมีความจำเป็นในการกำหนดขาอินพุตและเอาต์พุตให้ตรงตามจุดประสงค์

การออกแบบภาคควบคุมสถานะเปิด – ปิดของสวิทช์

ชุดควบคุมสถานะ การเปิด - ปิดของสวิทช์จะใช้รีเลย์ 12 VDC โดยจะใช้หน้าสัมผัสของรีเลย์มาเป็นตัวกำหนดการ เปิด - ปิดของสวิทช์ โดยควบคุมด้วยชุดไมโครคอนโทรลเลอร์ จะทำหน้าที่ควบคุมสถานะ และ ส่งค่าสถานะ ให้กับคอมพิวเตอร์



ภาพที่ 16 แสดงหน้าที่ของขาควบคุมไมโครคอนโทรเลอร์

จากภาพที่ 16 แสดงขาควบคุมของไมโครคอนโทรลเลอร์ในงานวิจัยนี้ได้มีการกำหนดอินพุตไว้จำนวน 5 ชุดและเอาพุตจำนวน 5 ชุด และภาคตรวจวัดอุณหภูมิจำนวน 1 ชุด

ชุดควบคุมการเปิดปิดไฟได้กำหนดให้เป็นขา 35, 37, 38, 39, 40 ขาละ 1 ชุดควบคุม ชุดควบคุมการตรวจวัดกระแส(เอาต์พุต)ได้กำหนดให้เป็นขา 22, 27,28,29,30 ต่อเชื่อมอยู่กับULN 2003 และไปขยายกระแสโดยผ่านชุดรีเลย์ (เนื่องจากกระแสที่ได้มีค่าที่น้อยมากจึงต้องเพิ่มวงจรขยายกระแส)

1.3 ชุดตรวจสอบกระแส Current Transformer (CT)

Current transformer (CT)

CT ไม่เหมือน power transformer ทั้งหมดแต่ใช้หลักการ electromagnetic induction เหมือนกัน ลักษณะการใช้งานต่างกัน ใน power transformer กระแสไหลผ่านขดลวด primary จะมีความสัมพันธ์กับกระแสด้าน secondary ซึ่งเป็นไปตาม load แต่ CT มีขดลวด primary ต่ออนุกรม (series) กับ line เพื่อวัดกระแสที่ไหลผ่าน หรือกล่าวได้ว่า กระแสในขดลวด primary จะไม่ขึ้นกับ load ที่ต่ออยู่ อาจแบ่งประเภทของ CT ได้เป็นสองชนิดตามการใช้งาน

- CT ที่ใช้วัดกระแส โดยนำอุปกรณ์ instrument เช่น metering system ต่างๆ คือ energy meter, current indicating meter มาต่อเข้าที่ด้าน secondary เรียกว่า metering current transformer
- CT ที่ใช้กับระบบป้องกัน (protective equipment) เช่น trip coil, relay ซึ่งเรียกว่า protective current transformer

หน้าที่ของ CT คือ แปลงกระแสสูงค่าหนึ่ง เป็นกระแสอีกค่าหนึ่งที่ต่ำลง ตามมาตรฐานกำหนด rated current ของขดลวด secondary ไว้ที่ 5A และ 1A เพื่อเป็นมาตรฐานในการผลิตอุปกรณ์ หรือ เครื่องมือที่นำมาต่อเข้าที่ secondary terminal กรณีใช้งานกับไฟแรงสูง จำเป็นต้องมีฉนวนที่ สามารถทนต่อแรงดันใช้งานและแรงดันผิดปกติที่อาจเกิดขึ้นในระบบ แต่หากไม่คำนึงถึงฉนวน (insulation) สิ่งสำคัญของ CT ที่ต้องมี คือ

- primary winding
- magnetic core
- secondary winding
- burden

primary winding เป็นขดลวดที่ต่ออนุกรม (series) กับ line หรือ primary circuit แบ่ง เป็นชนิด มีรอบเดียว single-turn primary winding ได้แก่ ring-type หรือ through-type ใช้ line conductor/cable, หรือ busbar คล้องหรือสวมให้ผ่านช่อง (window) ของ core ที่มีขดลวด secondary พันอยู่จึงถือเป็นรอบเดียว และ ชนิดที่มีหลายรอบ multi-turn primary winding หรือ wound primary

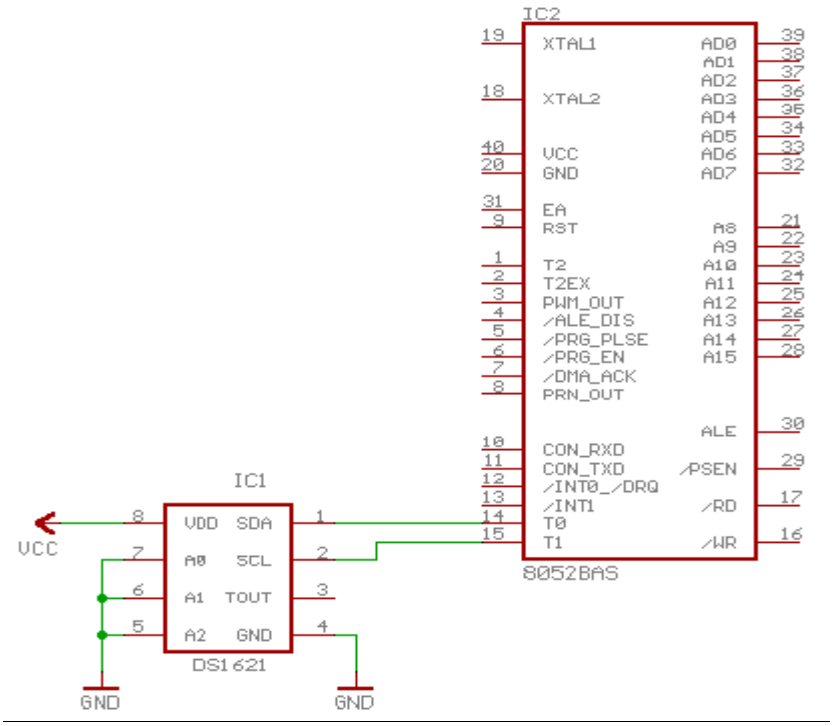
current transformer ขดลวดของ primary มีหนึ่งรอบจะดีกว่าหลายรอบ กล่าวคือ ผลต่อ แรงทางกล ที่กระทำกับ conductor ของ primary ในขณะมีกระแสลัดวงจรไหลผ่าน และ ความร้อน ที่เกิดขึ้น จากกระแสสูง (dynamic and thermal stresses) magnetic core เป็นแกนเหล็กที่ให้ induced flux ไหล คุณสมบัติของ CT ที่สำคัญคงเป็น เรื่องความละเอียดถูกต้องแม่นยำ และ ความเที่ยงตรง (accuracy) ของ CT คุณสมบัตินี้ขึ้นอยู่กับชนิดของ material ที่ใช้ทำ core และโครงสร้างของ core จึงมักใช้ magnetic alloys secondary winding เป็นขดลวดชุดที่สองที่พันบนแกนเพื่อลดกระแสให้ต่ำลง สามารถนำอุปกรณ์ต่างๆ มาต่อเพื่อวัดค่าได้ คุณสมบัติของ CT ขึ้นอยู่กับ flux density ใน core เป็น สำคัญ flux ที่เกิดขึ้นจะขึ้นอยู่กับ impedance ทั้งหมดใน secondary circuit ส่วนหนึ่งคือ impedance ของขดลวด secondary บางกรณีจะมีค่ามากกว่า impedance ของอุปกรณ์ หรือ burden ที่ต่อเข้า burden เป็น impedance ของอุปกรณ์ ที่ต่อทาง secondary เช่น relay, เครื่องมือวัดหรืออุปกรณ์อื่นๆ รวมทั้งสายที่ต่อระหว่างอุปกรณ์กับ secondary terminal รวมทั้งหมด คือ burden ของ CT อาจมี หน่วยเป็น VA หรือ ohm

1.4 การออกแบบชุดควบคุมอุณหภูมิ

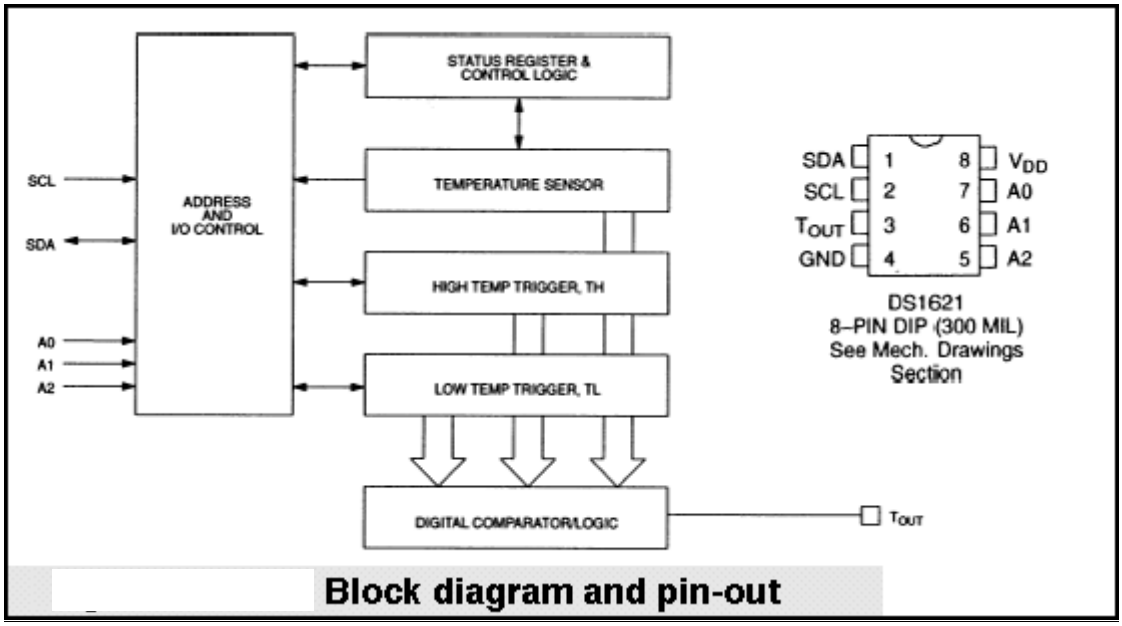
หลักการทำงานของวงจร

วงจร ดิจิตอลเทอร์โมมิเตอร์ การต่อใช้งานไอซีตรวจจับอุณหภูมิเบอร์ DS1820 มีเพียง 3 ขา คือ ขา DQ ซึ่งเป็นขาเชื่อมต่อกับระบบบัส ขาต่อไฟเลี้ยงภายนอกและขา GND เมื่อ ไอซี DS1820 ตรวจจับอุณหภูมิได้ จะนำค่าที่ได้นี้มาเก็บไว้ในหน่วยความจำ ไอซี DS1820 สามารถให้ข้อมูลของ อุณหภูมิได้ 16 bit เพื่อส่งค่านี้ให้กับ Microcontroller เบอร์ PIC16F877 คำนวณค่าของอุณหภูมิ แล้ว แปลงออกมาเป็นข้อมูลเลขฐาน 10 จึงสามารถแสดงความละเอียดของค่าอุณหภูมิได้ถึง 0.5 องศา เซลเซียส โดยมีย่านวัดอุณหภูมิตั้งแต่ -55 ถึง +125 องศาเซลเซียส

การกำหนดขาและหลักการในการทำงานของชุดควบคุมอุณหภูมิจะใช้อินพุต เป็น ไอซี DS1820 โดยใช้หลักการของภาคตรวจวัดอุณหภูมิเป็นตัวเช็คอุณหภูมิของสถานที่แห่งนั้น โดยจะ ใช้เป็นเซ็นเซอร์ไอซีเบอร์ DS1820 เป็นตัวตรวจวัดอุณหภูมิ เหตุผลที่เลือกเซ็นเซอร์เบอร์ DS1820 ก็ เพราะว่ามีฟังก์ชั่นการทำงานที่ไม่ยุ่งยาก และค่าผิดพลาดอยู่ที่ +- 0.5 องศาเซลเซียส



ภาพที่ 17 แสดงการกำหนดขาของภาคตรวจวัดอุณหภูมิ



ภาพที่ 18 แสดง Block diagram and in-out ของ DS1621

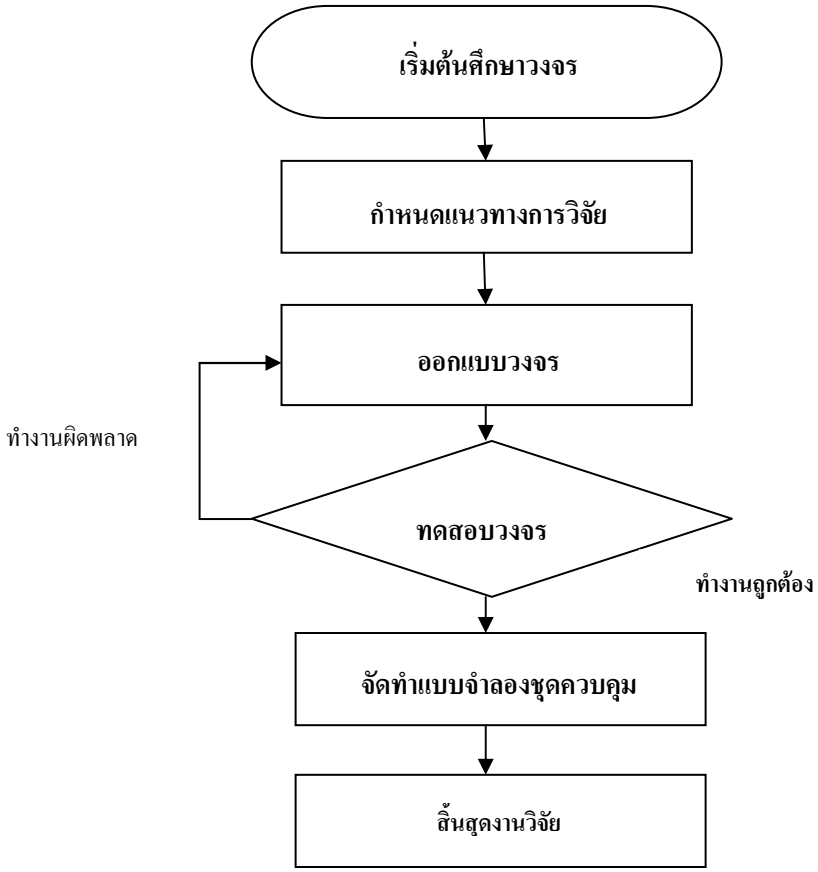
ภาคตรวจวัดอุณหภูมิสามารถนำไปประยุกต์ใช้ในการตรวจวัดอุณหภูมิอุปกรณ์อื่นๆได้อีกหลายชนิด เช่น ใช้ในการตรวจวัดอุณหภูมิในหม้อต้มน้ำในโรงงานอุตสาหกรรม เป็นต้น แต่ในงานวิจัยนี้ใช้ในการตรวจวัดอุณหภูมิภายในห้อง

การเชื่อมต่อระหว่างไมโครคอนโทรลเลอร์ กับคอมพิวเตอร์

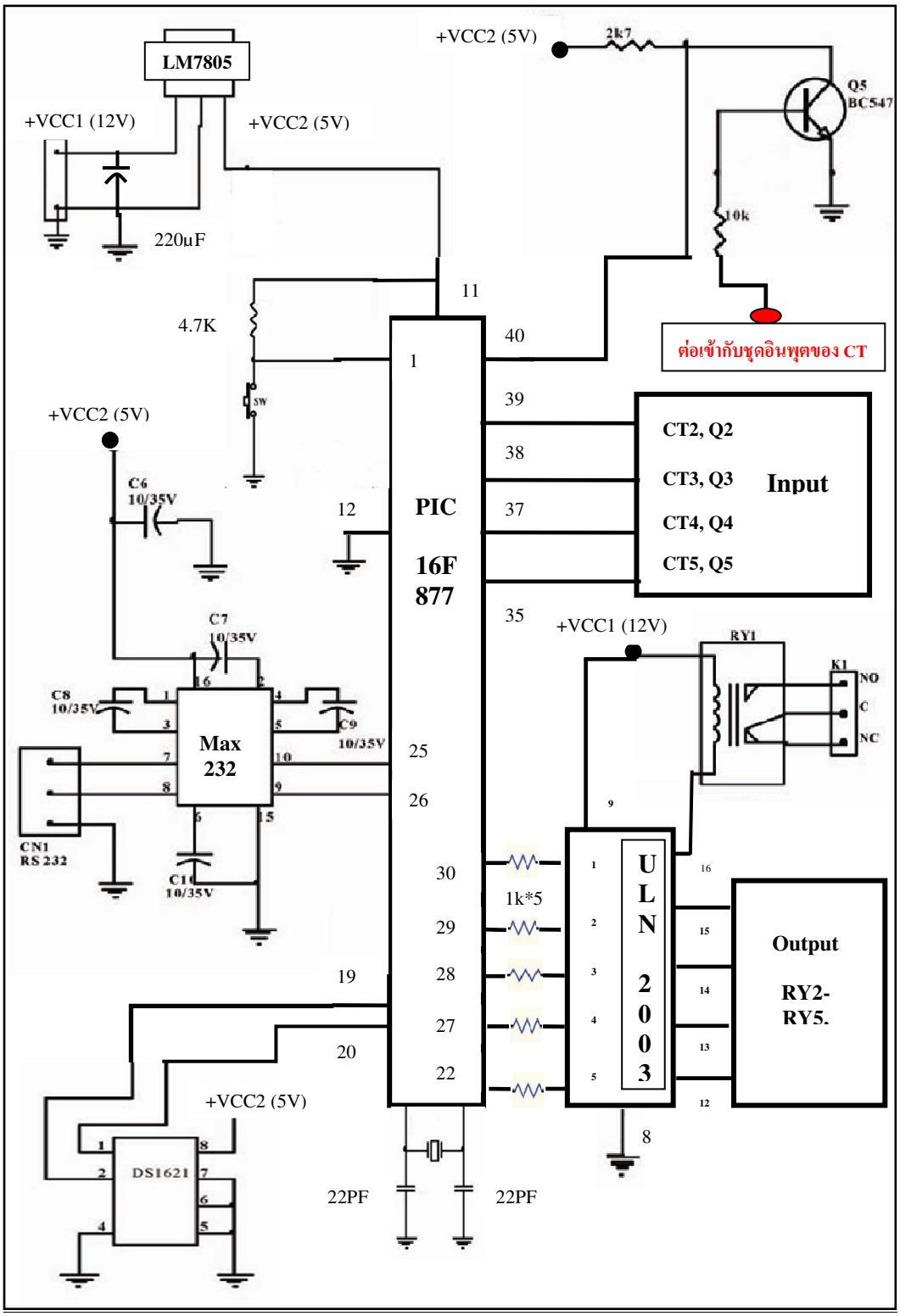
Hyper Terminal เป็นโปรแกรมที่มีอยู่ในวินโดว์ ถ้าไม่มีสามารถทำได้ดังนี้ โดยไปที่ Control Panel > Add/Remove Programs เลือกแท็ป Window setup ดับเบิ้ลคลิกที่ Communication คลิกถูกหน้า HyperTerminal กด Ok เริ่มใช้ คลิก

1. Start > programs > Accessories > Communication > HyperTerminal
2. กำหนดชื่อและรูปไอคอนกด OK จะได้
3. กำหนด Port ที่จะติดต่อกับไมโครคอนโทรลเลอร์ (ปกติเป็น COM1 หรือ COM2)
4. กำหนด Bit per second ให้ตรงกับที่กำหนดในโปรแกรม Ok

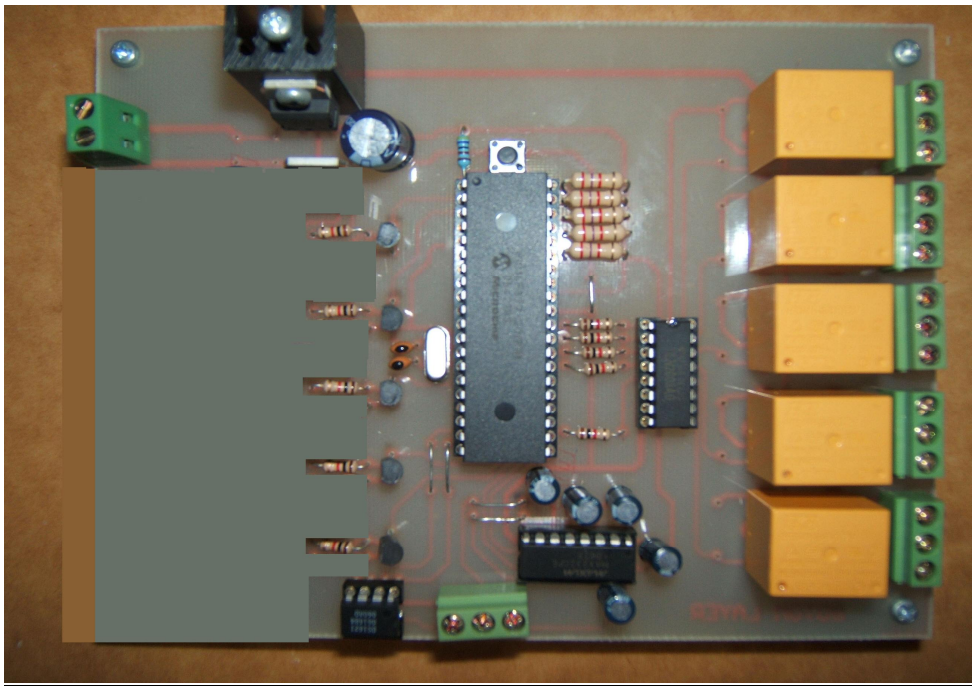
จากนั้นทำการศึกษา,กำหนดแนวทางการวิจัย,ออกแบบวงจรและกำหนดคุณสมบัติ



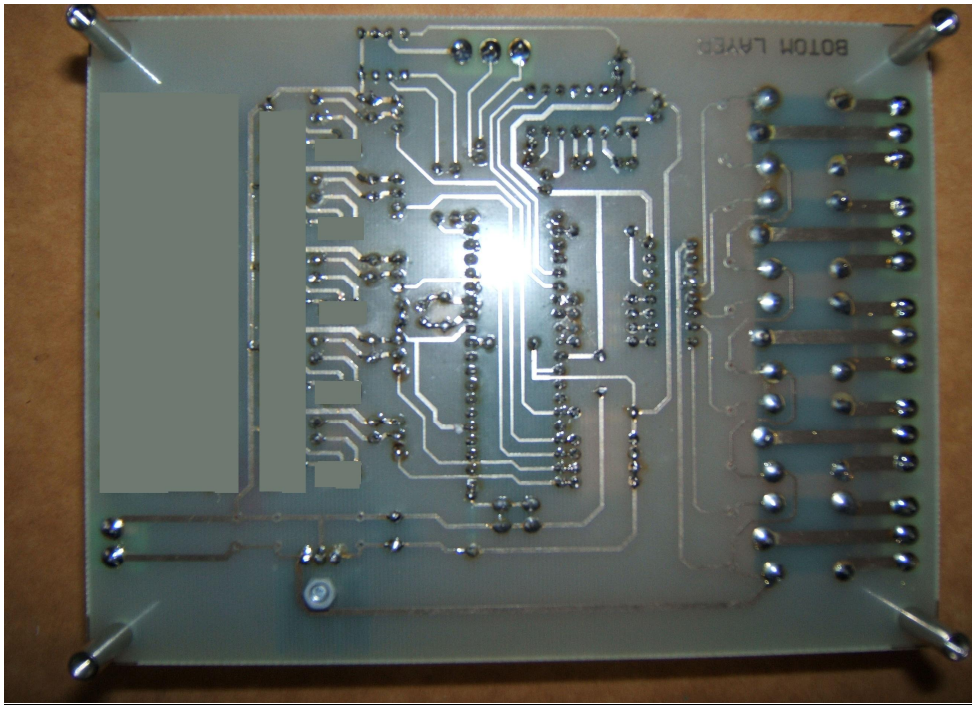
ภาพที่ 19 กำหนดแนวทางการวิจัย



ภาพที่ 20 แสดงวงจรควบคุมที่ออกแบบ

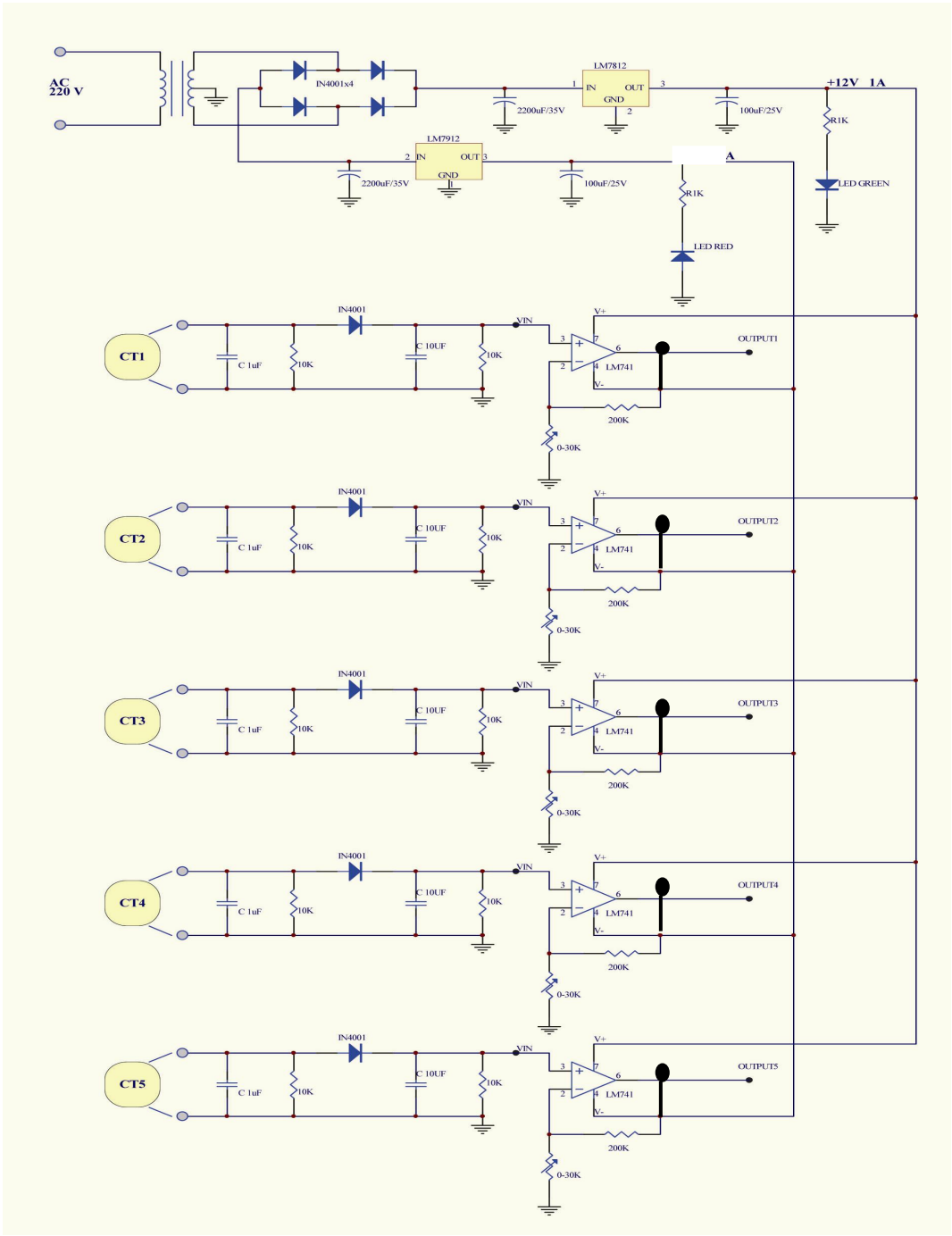


ก.



ข.

ภาพที่ 21 แสดงการต่ออุปกรณ์ลงแผ่นวงจรควบคุม(ก,ข)



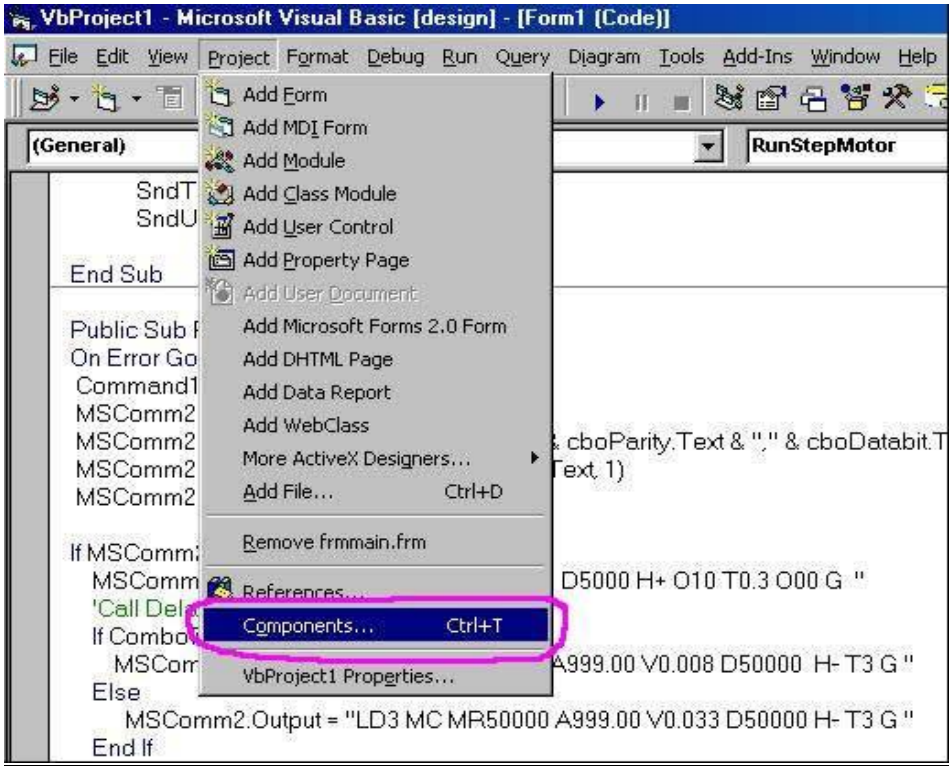
จาก CT1-CT5 ขา 6 ของออปแอมป์ LM741 จะต่อกับขา B ของทรานซิสเตอร์ เบอร์ BC547 ในชุดวงจรควบคุม

ภาพที่ 22 แสดงวงจรชุดอินพุต

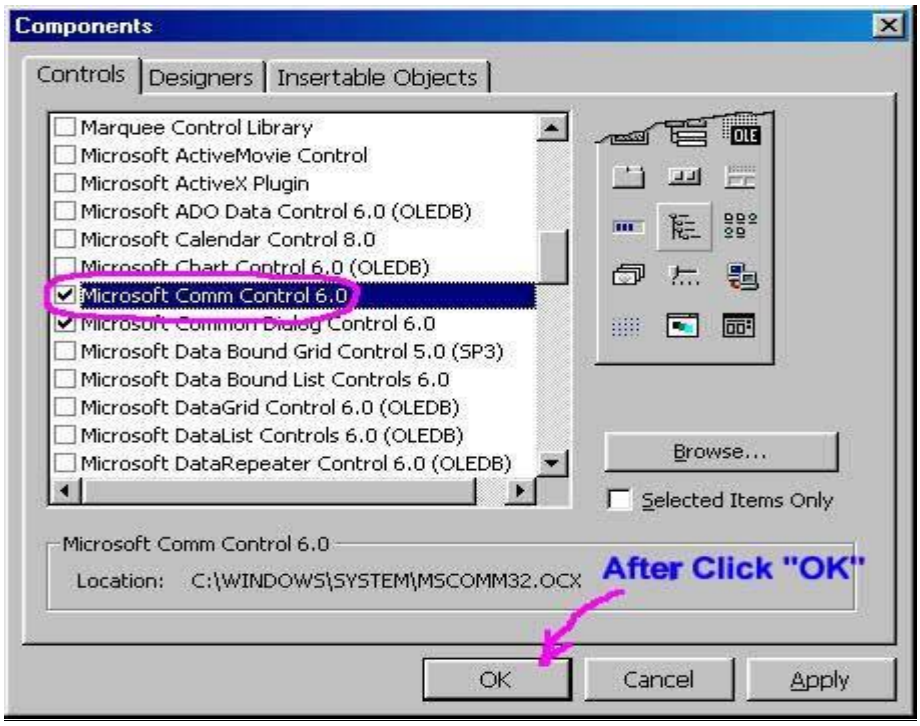
Visual Basic โปรแกรมผ่าน I/O Serial Port

สามารถทำได้โดยใช้ VB Control ที่ชื่อว่า MSComm โดยที่คุณต้อง กำหนด Custom Control เข้าไปที่ เมนู Project--->Components แล้วเลือกที่ช่อง MSComm ก็จะปรากฏ เป็นรูปไอคอนโทรศัพท์สีเหลือง ให้คลิกที่ไอคอนลากนำมาไว้บน Form ใน Project ของโปรแกรมโดยสามารถทำตามวิธีที่กล่าวมา ได้ดังรูป

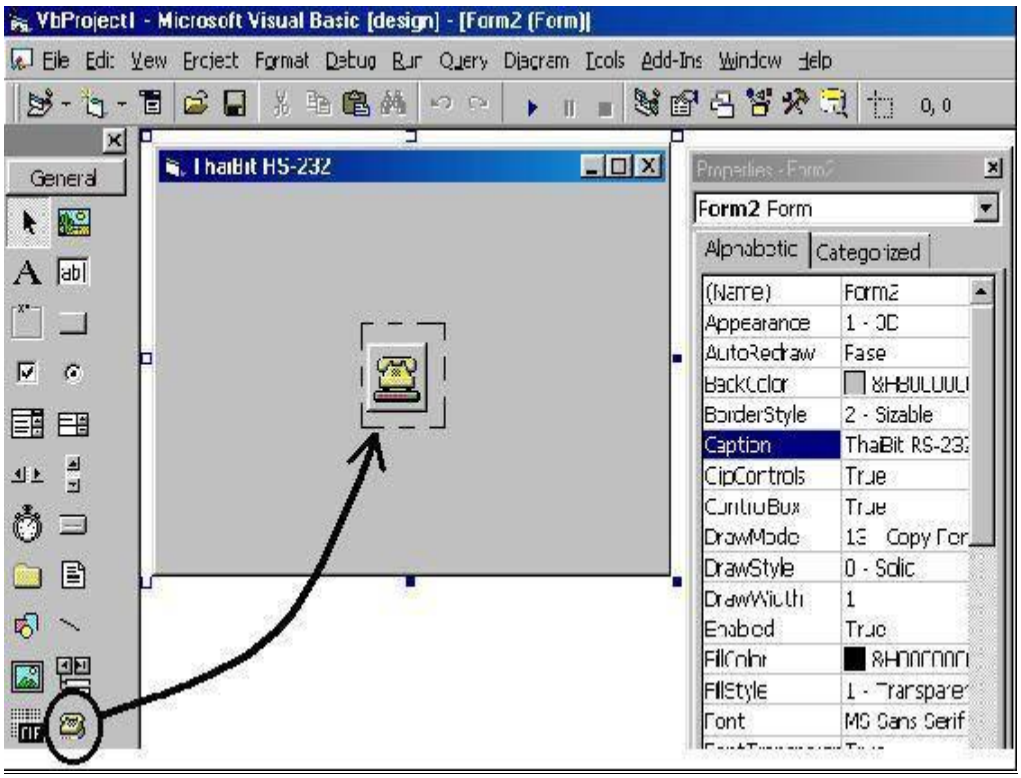
1. ขั้นที่ตอนแรก เลือกที่เมนูบาร์ด้านบนของโปรแกรม Visual Basic ดังรูปด้านล่าง



2. ขั้นที่สอง เลือกชื่อ Control ชื่อ Microsoft Comm Control 6.0 ดังรูปด้านล่าง



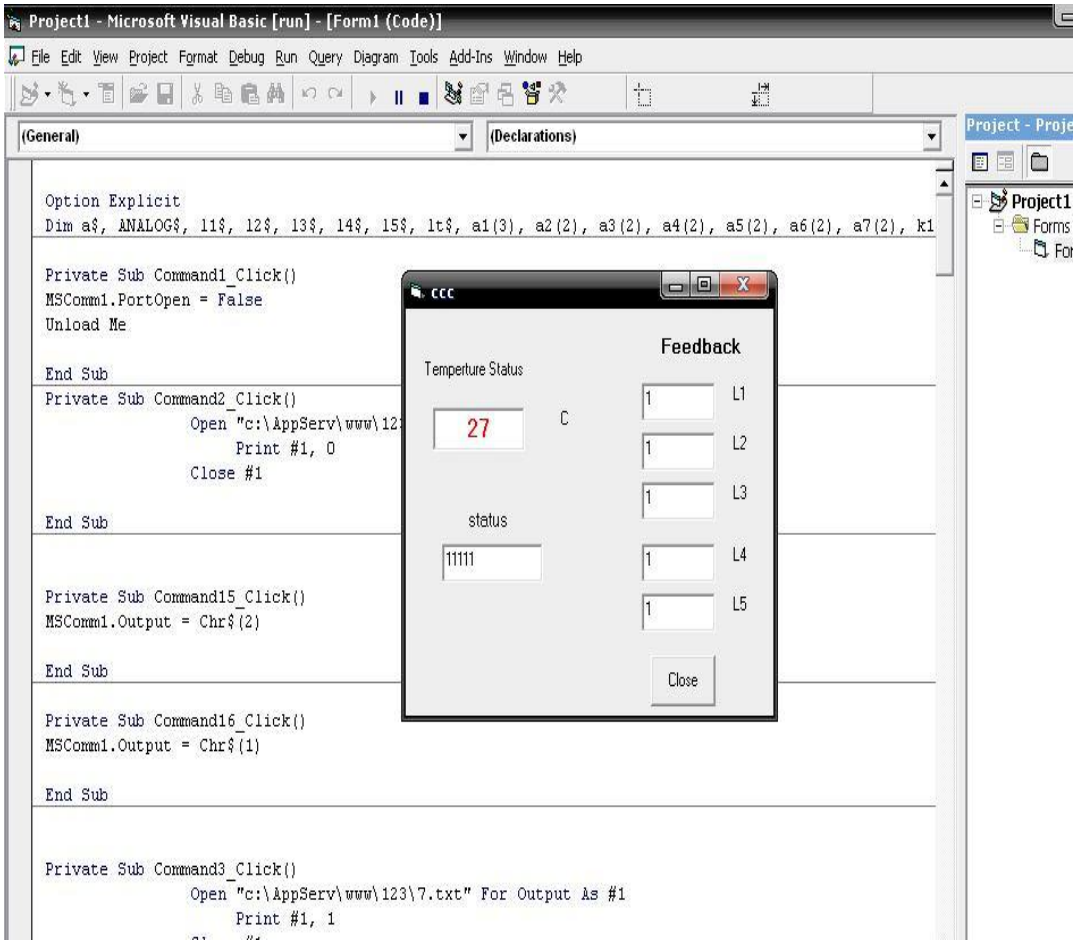
3. ขั้นที่สาม ลากControlชื่อMicrosoft Comm จากToolBox มาไว้บนFormดังรูปด้านล่าง



จากนั้นทำการสร้างฟอร์มในVisual Basic และเขียนโปรแกรมการทำงาน

หลักการเขียนโปรแกรมเพื่อควบคุมอุปกรณ์ไฟฟ้า

หลักกการเขียนต้องให้โปรแกรม Visual Basic เป็นตัวรับและส่งค่าเพื่อควบคุมอุปกรณ์ไฟฟ้า ดังนั้นจึงต้องเขียนโปรแกรมเพื่อให้Visual Basicติดต่อกับไมโครคอลโทรลเลอร์และนำค่าที่ได้มาเก็บไว้ในฐานข้อมูล



ภาพที่ 23 แสดงเมื่อทำการสั่ง RUN ในโปรแกรม Visual Basic

การเขียนโปรแกรมติดต่อกับ Serial Port สามารถทำได้ 2 วิธี คือ

1. การติดต่อแบบอินเตอร์รัพต์

ขบวนการอินเตอร์รัพท์ อุปกรณ์รอบข้างเกือบทุกชิ้นจะต้องปฏิบัติงานอยู่เพื่อส่งสัญญาณไปให้แก่ซีพียูเสมอ ถ้าอุปกรณ์นั้นพร้อมที่จะรับส่งจะส่งเป็นรหัสแอสกี้ เราจะเขียนโปรแกรมอินเตอร์รัพต์ เมื่อที่ข้อมูลเข้ามาก็จะทำให้มี CommEvent กับ OnComm Event

2.การติดต่อแบบโพลลิ่ง

ในระบบพีซี การโพลมีบ้างที่ใช้การส่งผ่านข้อมูลระหว่าง Terminal กับ CPU กรณีข้อมูลเป็นประเภทไบท์ที่ส่งจากคีย์บอร์ด โดยวิธีการนี้จะตรวจสอบ คีย์บอร์ดว่ามีข้อมูลส่งมาหรือเปล่า โดยจะตรวจสอบตลอดเวลา การทำงานกับข้อมูลที่รับเข้ามาจะตรวจสอบด้วยความเร็วที่สูงกว่าอัตราความเร็วข้อมูลที่ส่งเข้ามาทาง คีย์บอร์ด การที่ CPU ส่งสัญญาณออกไปตรวจสอบพบว่ามีข้อมูลที่ต้องส่งเข้ามา เรียกว่า"Wet Poll" ซึ่งจะเสียช่วงเวลา 90 เปอร์เซ็นต์ คาบเวลาที่เสียไปนั้น เราเลี่ยงไปใช้เทคนิค การโพลแบบ "Round Robin" แทน แต่ในVBเราจะใช้การตรวจสอบข้อมูลที่มาจาก Serial Port ตลอด โดยจะใช้ Control Timer เข้ามาช่วยในการเขียนโปรแกรมซึ่งสามารถตรวจสอบได้ถึงระดับ 1 มิลลิวินาที หรือจะใช้ Do....Loop ก็ได้

ในตัวคอนโทรล MSComm มี Event ที่ใช้เพียง Event เดียวเท่านั้นเอง ก็คือ OnComm Event ซึ่งจะใช้ในการติดต่อแบบอินเตอร์รัพต์ การเขียนโปรแกรมติดต่อ Serial Port แบบธรรมดาจะใช้ comEvent เพียง comEvReceive, comEvSend ถ้าเป็นการติดต่อสื่อสารแบบ โมเด็มจะใช้ comEvent หลายตัวในการตรวจสอบสัญญาณ

เมื่อต่ออุปกรณ์ต่างๆเข้ากับไมโครคอนโทรลเลอร์และทำการเขียนโปรแกรมควบคุมการสั่งงานผ่านโปรแกรมVisual Basicอุปกรณ์ไฟฟ้าสามารถทำงานตามการสั่งงานได้อย่างถูกต้องและสามารถตรวจสอบผลของการสั่งงานได้

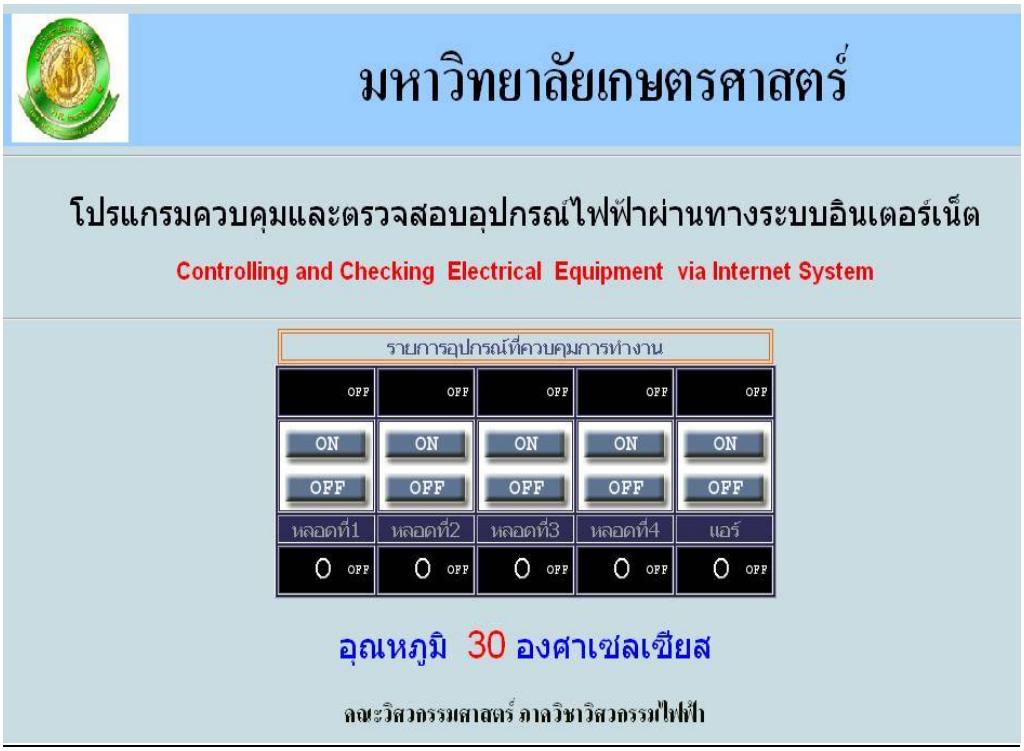
การออกแบบเวปไซด์ที่ใช้ควบคุมการทำงาน

การสร้างเวปไซด์ที่ใช้ที่ใช้ควบคุมการทำงานจะใช้โปรแกรม AppSev ในการออกแบบคุณสมบัติต่างๆในการสั่งงานที่หน้าเวปไซด์ โดยในเวปไซด์จะทำการควบคุมสวิทย์ on-off จำนวน 5 ชุด ประกอบไปด้วยการควบคุมหลอดไฟฟ้าจำนวน 4 ชุด และเครื่องปรับอากาศ 1 ชุด และแสดงสถานะอุณหภูมิห้องขณะนั้น และมีฟังก์ชั่นแสดงผลการสั่งงานของอุปกรณ์ว่ามีผลการทำงานตามการสั่งงานจริงหรือไม่โดยให้แสดงสถานะเป็น 1, 0 เป็น 1 แสดงว่ามีการสั่งงานอุปกรณ์นั้นได้มีการทำงานจริง เป็น 0 แสดงว่าการสั่งงานอุปกรณ์นั้นไม่ทำงานตามการสั่งงาน

โดยหลักการทำงานนั้นเวปไซด์จะไปทำหน้าที่ในการไปอัพเดทข้อมูลและสั่งงานในฐานข้อมูลในVisual Basic ซึ่ง Visual Basic จะทำการอัพเดทค่าและสั่งงานไปยังไมโครคอนโทรลเลอร์ และจะนำผลที่ได้มาแสดงในหน้าหลักของเวปไซด์

เวปไซด์จะสามารถทำการสั่งงานอุปกรณ์ไฟฟ้าได้ก็ต่อเมื่อจะต้องทำการ Run โปรแกรม Visual Basic ก่อนเพื่อที่จะได้สามารถติดต่อและสั่งงานกับคอนโทรลเลอร์ได้ ดังนั้น Main Computer จะต้องมีการทำงานอยู่ตลอดเวลา เพื่อเวปไซด์จะได้สามารถทำการควบคุมอุปกรณ์ไฟฟ้าได้ตลอดเวลา

ภาพที่ 24 แสดงหน้าเวปไซด์ขณะอุปกรณ์ไฟฟ้ามีการทำงาน



ภาพที่ 25 แสดงหน้าเวปไซด์ขณะสั่งหยุดการทำงานของอุปกรณ์ไฟฟ้า

ภาพที่ 26 แสดงแบบจำลองโรงงานและอุปกรณ์ไฟฟ้าที่ควบคุม

ผลและวิจารณ์

ผล

จากการวัดค่าอุณหภูมิ อุณหภูมิขณะวัด คือ 29 องศาเซลเซียส เมื่อทำการทดสอบโดยการเก็บค่าเมื่อมีอุณหภูมิที่แตกต่างต่างกันในที่นี้ใช้วิธีการทดสอบในอุณหภูมิของน้ำที่ 70 องศาเซลเซียส และเก็บค่าเมื่อน้ำมีอุณหภูมิลดลงเรื่อยโดยเซ็นเซอร์อุณหภูมิจะลดลงทีละ 1 องศาเซลเซียสจนถึง 35 องศาเซลเซียสซึ่งเป็นอุณหภูมิคงที่ และจากการทดสอบหลอดไฟแบบไส้ขนาด 23W, 25W, 25W, 60W, 100W ตามลำดับ ได้ผลการทดลองดังนี้

ตารางที่ 2 แสดงผลการทดลองสั่งงานอุปกรณ์ไฟฟ้า ก

ขนาดหลอดไฟ	**กระแสไฟฟ้าจากการคำนวณ I=P/V (แอมแปร์)**	**กระแสที่ได้จากการวัดโดย Current Transformers (CT) (แอมแปร์)**	**ผลของการสั่งงานที่สามารถเช็คได้จาก CT ส่งค่าเป็น 1,0**
23 W หลอดฟลูออเรสเซ็นต์	0.105	0.06	ON (1)
25 W หลอดไส้ Superluck	0.114	0.06	ON (1)
25 W หลอดไส้ Softone	0.114	0.06	ON (1)
60 W หลอดไส้ Daylight Blue	0.273	0.17	ON (1)
100 W หลอดไส้ Clear bright sparkling	0.455	0.32	ON (1)

ตารางที่ 3 แสดงผลการทดลองสั่งงานอุปกรณ์ไฟฟ้า ข

ขนาดหลอดไฟ	แรงดัน VDC ที่ได้จากวัด CT (Vin) (โวลต์)	แรงดัน Vin ที่นำไปขยายแรงดัน (Op amp) (โวลต์)	ผลของการนำแรงดันไปทริ๊กซ์ที่ขา B ทรานซิสเตอร์ (ขา C ของทรานซิสเตอร์มีไฟ 5V) (V)
23 W หลอดฟลูออเรสเซ็นต์	0.09	3.52	0.01 ทำงาน
25 W หลอดไส้ Superluck	0.04	4.11	0.01 ทำงาน
25 W หลอดไส้ Softone	0.05	4.11	0.01 ทำงาน
60 W หลอดไส้ Daylight Blue	0.20	4.65	0.01 ทำงาน
100 W หลอดไส้ Clear bright sparkling	0.41	4.95	0.01 ทำงาน

วิจารณ์

จากผลการทดลองหลอดไฟขนาดต่างๆเมื่อทำการสั่งงานที่โปรแกรมหลอดไฟจะทำงานตามวงจรที่ได้ออกแบบไว้และจะสามารถเช็คผลของการสั่งงานได้จาก Current transformers (CT) ถ้ามีกระแสไหลผ่าน CT แสดงว่าผลของการสั่งงานหลอดไฟฟ้านั้นทำงานตามการสั่งงาน (สถานะ ON)

สรุปและข้อเสนอแนะ

สรุป

ผู้จัดทำได้พัฒนาโปรแกรมและฮาร์ดแวร์โดยเทียบผลการคำนวณกระแสและแรงดันจากตัวอย่างในเอกสารอ้างอิงหลาย ๆ เล่ม แล้วจึงนำมาผนวกเขียนเป็นวงจรและโปรแกรม ตลอดจนได้พัฒนาให้ใช้งานง่าย จากการทดสอบโปรแกรมมีข้อผิดพลาดน้อยมาก จึงเหมาะที่จะนำไปควบคุมอุปกรณ์ไฟฟ้า และยังสามารถนำไปประยุกต์ใช้กับการเรียนการสอนได้เป็นอย่างดี นอกจากจะสามารถควบคุมการทำงานของอุปกรณ์ไฟฟ้าได้ผู้จัดทำยังได้แยก Source Code ของโปรแกรมออกเป็นส่วน ๆ โดยส่วนใหญ่เขียนอยู่ในรูปของ Function ซึ่ง Function เหล่านี้สามารถนำไปประยุกต์ใช้งานร่วมกับโปรแกรมอื่น ๆ ได้ ตามความเหมาะสม

จากผลการคำนวณโดยใช้โปรแกรมที่ผู้จัดทำสร้างขึ้น และนำผลที่ได้ไปเทียบกับผลการและโปรแกรมยังสามารถนำไปพัฒนาต่อหรือแก้ไข Source Code ได้ เพราะตัวโปรแกรมยังไม่ได้ Compile และยังสามารถพัฒนาและควบคุมอุปกรณ์ไฟฟ้าได้อีกจำนวนมาก

ข้อเสนอแนะ

หากต้องการให้มีความแม่นยำในการควบคุมและตรวจสอบอุปกรณ์ไฟฟ้าได้มีประสิทธิภาพมากยิ่งขึ้นควรจะมีการหาวิธีตรวจสอบผลของการสั่งงานควบคุมอุปกรณ์ไฟฟ้าได้หลายๆวิธี

เอกสารและสิ่งอ้างอิง

กอบเกียรติ กาญจนาพงศ์กุล. ไมโครคอนโทรลเลอร์ ไอซีชิพที่น่าใช้งาน. **คอมพิวเตอร์อิเลคทรอนิคส์ เวิลด์** (177): 30-37.

ชลิต วณิชยานันต์ คุณวุฒิ, 2545 .**การใช้คอมพิวเตอร์ควบคุมอุปกรณ์ภายนอกผ่านพอร์ตอนุกรมและพอร์ตขนาน โดยโปรแกรมวิสชวล เบสิก.** สำนักวิจัยและบริการวิชาการ สถาบันราชภัฏบ้านสมเด็จเจ้าพระยา

ธีรวัฒน์ ประกอบผล. 2540. **ดิจิตอลอิเล็กทรอนิกส์** .แมคกรอ-ฮิล อินเตอร์เนชั่นแนล เอ็นเตอร์ไพร์ส, อิงค์ , กรุงเทพมหานคร

พจน์ พุทธา, วชิระ กังวานคุณากร, 2543 .**เครื่องคุมอุปกรณ์ไฟฟ้าด้วยเครื่องคอมพิวเตอร์ผ่านสายไฟกระแสสลับ.** เทคโนโลยีอิเล็กทรอนิกส์ (โทรคมนาคม) สถาบันเทคโนโลยีพระจอมเกล้าพระนครเหนือ

พ.อ.เจนวิทย์ เหลืองอร่าม,ปิยวิทย์ เหลืองอร่าม.**การเขียนโปรแกรม VISUAL BASIC 6.0** .บริษัทธรรมสาร จำกัด.กรุงเทพฯ, 2546.

สัลยุทธ์ สว่างวรรณ. 2547. **สถาปัตยกรรมคอมพิวเตอร์**. เอช.เอ็น. กรุ๊ป จำกัด, กรุงเทพฯ. แปลจาก W. Stallings. Computer Organization and Architecture. Pearson Education Indochina, Prentice hall

อภิชาต ภู่พลับ.**เริ่มต้นเขียนโปรแกรมติดต่อและควบคุมฮาร์ดแวร์ด้วย Visual Basic**. อินโฟเพลส ดิเวลลอปเปอร์.กรุงเทพฯ, 2546.

Chen, Y. 2004. **Hardware Acceleration of Security Application using Reconfigurable System-on- Chip.** Available source: http://www.itee.uq.edu.au/~chenyi/project/confirmation%20seminar%20-%20yi%20chen.ppt

Copyright 1995 by Dallas Semiconductor Corporation. All Rights Reserved. For **important information regarding patents and other intellectual property rights**, please refer to Dallas Semiconductor data books.

Ross Bannatyne and Greg Viot. 1998. **Introduction to Microcontrollers-Part 1.** IEEE (1998) 350-360

ภาคผนวก

โปรแกรมที่ใช้ในการควบคุม

โปรแกรม Visual Basic ที่ใช้ในการควบคุม

```
Option Explicit
Dim a$, ANALOG$, l1$, l2$, l3$, l4$, l5$, lt$, a1(3), a2(2), a3(2), a4(2), a5(2), a6(2), a7(2), k1

Private Sub Command1_Click()
MSComm1.PortOpen = False
Unload Me

End Sub
Private Sub Command2_Click()
        Open "c:\AppServ\www\123\1.txt" For Output As #1
           Print #1, 0
        Close #1

End Sub

Private Sub Command15_Click()
MSComm1.Output = Chr$(2)

End Sub

Private Sub Command16_Click()
MSComm1.Output = Chr$(1)

End Sub
```

```
Private Sub Command3_Click()
        Open "c:\AppServ\www\123\7.txt" For Output As #1
           Print #1, 1
        Close #1
End Sub

Private Sub Command4_Click()
        Open "c:\AppServ\www\123\7.txt" For Output As #1
           Print #1, 0
        Close #1
End Sub

Private Sub Command5_Click()
        Open "c:\AppServ\www\123\2.txt" For Output As #1
           Print #1, 1
        Close #1
End Sub

Private Sub Command6_Click()
        Open "c:\AppServ\www\123\2.txt" For Output As #1
           Print #1, 0
        Close #1
End Sub

Private Sub Command7_Click()
        Open "c:\AppServ\www\123\4.txt" For Output As #1
           Print #1, 1
        Close #1

End Sub
```

```
Private Sub Command8_Click()
        Open "c:\AppServ\www\123\4.txt" For Output As #1
           Print #1, 0
        Close #1

End Sub

Private Sub Command9_Click()
        Open "c:\AppServ\www\123\5.txt" For Output As #1
           Print #1, 1
        Close #1

End Sub

Private Sub Form_Load()
MSComm1.PortOpen = True
End Sub

Private Sub MSComm1_OnComm()
ANALOG$ = MSComm1.Input

 Text8.Text = ANALOG$

End Sub


'  MSComm1.Output = Chr$(34)
```

```
Private Sub Command10_Click()
        Open "c:\AppServ\www\123\5.txt" For Output As #1
           Print #1, 0
        Close #1

End Sub

Private Sub Command11_Click()
        Open "c:\AppServ\www\123\3.txt" For Output As #1
           Print #1, 0
        Close #1
End Sub

Private Sub Command12_Click()
        Open "c:\AppServ\www\123\3.txt" For Output As #1
           Print #1, 1
        Close #1
End Sub

Private Sub Command13_Click()
        Open "c:\AppServ\www\123\6.txt" For Output As #1
           Print #1, 1
        Close #1
End Sub

Private Sub Command14_Click()
        Open "c:\AppServ\www\123\6.txt" For Output As #1
           Print #1, 0
        Close #1
```

```
End Sub

Private Sub Timer1_Timer()

'Mytime = #5:04:23 PM#
'Mystr = Format(Mytime, "h:m:s")
'Mystr = Time
    k1 = Mid$(ANALOG$, 1, 1)
 '   Text7.Text = ANALOG$
    If k1 = 2 Then
       Text1.Text = Mid$(ANALOG$, 2, 3)
   Else
         Text2.Text = Mid$(ANALOG$, 2, 1)
         Text3.Text = Mid$(ANALOG$, 3, 1)
         Text4.Text = Mid$(ANALOG$, 5, 1)
         Text5.Text = Mid$(ANALOG$, 4, 1)
         Text6.Text = Mid$(ANALOG$, 1, 1)

   End If

End Sub

Private Sub Timer2_Timer()

            Open "c:\AppServ\www\123\tmp.txt" For Output As #1
               Print #1, "" & Format(Val(Text1.Text), "#"); ""
            Close #1
```

```
Open "c:\AppServ\www\123\s1.txt" For Output As #1

Print #1, "" & Text2.Text; ""
Close #1

  Open "c:\AppServ\www\123\s2.txt" For Output As #1
   Print #1, "" & Text3.Text; ""
Close #1

  Open "c:\AppServ\www\123\s3.txt" For Output As #1
   Print #1, "" & Text4.Text; ""
Close #1

 Open "c:\AppServ\www\123\s4.txt" For Output As #1
   Print #1, "" & Text5.Text; ""
Close #1

 Open "c:\AppServ\www\123\s5.txt" For Output As #1
   Print #1, "" & Text6.Text; ""
Close #1



Open "c:\AppServ\www\123\1.txt" For Input As #1
Do While Not EOF(1)
Line Input #1, l1$
Loop
Close #1
a1(0) = l1$

Open "c:\AppServ\www\123\2.txt" For Input As #1
```

```
Do While Not EOF(1)

Line Input #1, l2$
Loop
Close #1
 a2(0) = l2$

Open "c:\AppServ\www\123\3.txt" For Input As #1
Do While Not EOF(1)
Line Input #1, l3$
Loop
Close #1
 a3(0) = l3$


Open "c:\AppServ\www\123\4.txt" For Input As #1
Do While Not EOF(1)
Line Input #1, l4$
Loop
Close #1
 a4(0) = l4$

Open "c:\AppServ\www\123\5.txt" For Input As #1
Do While Not EOF(1)
Line Input #1, l5$
Loop
Close #1
 a5(0) = l5$

 lt$ = l1$ & l2$ & l3$ & l4$ & l5$
 Open "c:\AppServ\www\123\text2.txt" For Output As #1
```

```
            Print #1, "" & lt$; ""


  Close #1
       '  Timer3.Enabled = True
End Sub

'              Open "c:\AppServ\www\123\2.txt" For Input As #1
 '            Do While Not EOF(1)
  '           Line Input #1, a$
   '          Loop
    '          Close #1
'             If a$ = "" Then
 '              a$ = 0
  '          End If
   '          If a$ = 1 Then
    '             Command5_Click
     '       Else
      '           Command6_Click
       '      End If

Private Sub delay(i)
  Dim x, q
  q = 0
    For x = 1 To i
          q = q + 1
    Next x
End Sub

Private Sub Timer3_Timer()
            Dim dd$, lout, out1$
```

```
Open "c:\AppServ\www\123\text2.txt" For Input As #1

Do While Not EOF(1)
Line Input #1, dd$
Loop
Close #1

l1$ = Mid$(dd$, 1, 1)
l2$ = Mid$(dd$, 2, 1)
l3$ = Mid$(dd$, 3, 1)
l4$ = Mid$(dd$, 4, 1)
l5$ = Mid$(dd$, 5, 1)

If a1(1) <> a1(0) Then
If l1$ = "1" Then
MSComm1.Output = Chr$(11)
Else
MSComm1.Output = Chr$(10)
End If
delay (100)
End If

If a2(1) <> a2(0) Then
If l2$ = "1" Then
MSComm1.Output = Chr$(21)
Else
MSComm1.Output = Chr$(20)
End If
delay (100)
End If
```

```
If a3(1) <> a3(0) Then

If l3$ = "1" Then
    MSComm1.Output = Chr$(31)
  Else
    MSComm1.Output = Chr$(30)
  End If
 delay (100)
End If

If a4(1) <> a4(0) Then
  If l4$ = "1" Then
    MSComm1.Output = Chr$(41)
  Else
    MSComm1.Output = Chr$(40)
  End If
 delay (100)
End If

If a5(1) <> a5(0) Then
  If l5$ = "1" Then
    MSComm1.Output = Chr$(51)
  Else
    MSComm1.Output = Chr$(50)
  End If
 delay (100)
End If

 delay (100)
a1(1) = a1(0)
a2(1) = a2(0)
```

```
            a3(1) = a3(0)


            a4(1) = a4(0)
            a5(1) = a5(0)
        '   Timer3.Enabled = False
End Sub
```

โปรแกรม Appsev ที่ใช้ในการสร้างเวปไซด์

```
<title>Controling Electrical Instrument by Internet System</title>
<table width=100%  class=MsoTableGrid border=0 cellspacing=0 cellpadding=0
 style='background:#99CCFF;border-collapse:collapse;mso-yfti-tbllook:480;
 mso-padding-alt:0cm 5.4pt 0cm 5.4pt'>
 <tr style='mso-yfti-irow:0;mso-yfti-firstrow:yes;mso-yfti-lastrow:yes'>
  <td width=100% valign=top   style='width:102.3pt;padding:0cm 5.4pt 0cm 5.4pt'>
     <img width=108 height=105  src="image/ku.gif" </p>
  </td>
  <td width=100% style='width:610.7pt;padding:0cm 5.4pt 0cm 5.4pt'>
  <p class=MsoNormal align=center style='text-align:center'><b><span lang=TH
  style='font-size:48.0pt;font-family:"Angsana New";mso-ascii-font-family:"Times New Roman";
  mso-hansi-font-family:"Times New Roman"'>มหาวิทยาลัยเกษตรศาสตร์</span></b><b><span
  style='font-size:48.0pt'><o:p></o:p></span></b></p>


  </td>
 </tr>
</table>
<?php

$fileread2 = fopen("C:/AppServ/www/123/tmp.txt","r");
$tmp1 = fread($fileread2,5);
```

```
fclose($fileread2);

$fileread2 = fopen("C:/AppServ/www/123/tmp2.txt","r");
$tmp2 = fread($fileread2,5);
fclose($fileread2);

////////////////////////////////feedback//////////////////////////////////////

$fileread2 = fopen("C:/AppServ/www/123/s1.txt","r");
$r1 = fread($fileread2,1);
if($r1==1){

    $im11 = "image/1.gif";
  }
else
  {

 $im11 = "image/0.gif";

  }
fclose($fileread2);

$fileread2 = fopen("C:/AppServ/www/123/s2.txt","r");
$r1 = fread($fileread2,1);
if($r1==1){

    $im21 = "image/1.gif";
  }
else
  {
```

```
 $im21 = "image/0.gif";

   }
fclose($fileread2);

$fileread2 = fopen("C:/AppServ/www/123/s3.txt","r");
$r1 = fread($fileread2,1);
if($r1==1){

    $im31 = "image/1.gif";
  }
else
  {

 $im31 = "image/0.gif";

   }
fclose($fileread2);

$fileread2 = fopen("C:/AppServ/www/123/s4.txt","r");
$r1 = fread($fileread2,1);
if($r1==1){

    $im41 = "image/1.gif";
  }
else
  {

 $im41 = "image/0.gif";
```

```
  }
fclose($fileread2);
$fileread2 = fopen("C:/AppServ/www/123/s5.txt","r");
$r1 = fread($fileread2,1);
if($r1==1){

    $im51 = "image/1.gif";
  }
else
  {

   $im51 = "image/0.gif";
   }
fclose($fileread2);

////////////////////////////////on-off/////////////////////////////////////////////

$fileread2 = fopen("C:/AppServ/www/123/1.txt","r");
$r1 = fread($fileread2,1);
if($r1==1){
    $im1 = "image/on.gif";

  }
else
  {
   $im1 = "image/off.gif";

   }
fclose($fileread2);
$fileread2 = fopen("C:/AppServ/www/123/2.txt","r");
```

```
$r1 = fread($fileread2,1);
if($r1==1){
     $im2 = "image/on.gif";
 }
else
   {
    $im2 = "image/off.gif";

    }
fclose($fileread2);

$fileread2 = fopen("C:/AppServ/www/123/3.txt","r");
$r1 = fread($fileread2,1);
if($r1==1){
     $im3 = "image/on.gif";

   }
else
   {
    $im3 = "image/off.gif";

    }
fclose($fileread2);

$fileread2 = fopen("C:/AppServ/www/123/4.txt","r");
$r1 = fread($fileread2,1);
if($r1==1){
     $im4 = "image/on.gif";

   }
```

```
else
  {
   $im4 = "image/off.gif";

   }
fclose($fileread2);

$fileread2 = fopen("C:/AppServ/www/123/5.txt","r");
$r1 = fread($fileread2,1);
if($r1==1){
    $im5 = "image/on.gif";

  }
else
  {
   $im5 = "image/off.gif";

   }
fclose($fileread2);
echo"<body bgcolor=\"#C9DCE2\" text=\"black\" link=\"blue\" vlink=\"purple\" alink=\"red\">";
?>
<?php
echo " <hr>"
."<br>"
."<div align=\"center\"><font face=\"DSN PreeCha,Arial\"><span style=\"font-size:24pt;\">
โปรแกรมควบคุมและตรวจสอบอุปกรณ์ไฟฟ้าผ่านทางระบบอินเตอร์เน็ต</span></font></div>
<p align=center><span style=\"font-size:19pt; mso-spacerun: yes\"> <span style=\"font-size:15pt;
mso-fareast-font-family:; font-family: Arial, Helvetica, sans-serif; font-weight: bold;\"></font>
<font color=\"red\">Controlling and Checking  Electrical  Equipment  via
Internet System</span></span></p>"
```

```
 . "          <hr>"
// . "          <table width=\"873\" border=\"0\" align=\"right\">"
// . "            <tr>"
// . "              <td width=\"867\">"
// . "                <marquee>"
// . "                <span class=\"style1\"><font face=\"DSN PreeCha,Arial\"><span
class=\"style5\">โปรแกรมควบคุมอุปกรณ์ไฟฟ้าผ่านระบบอินเทอร์เน็ต <span style=\"font-
size:14px; mso-spacerun: yes\"><span style=\"mso-fareast-font-family:; font-family: Arial,
Helvetica, sans-serif; font-weight: bold;\">CONTROLING  ELECTRICAL 
INSTRUMENTS  BY INTERNET SYSTEM </span><span style=\"mso-fareast-font-
family:; font-family: Arial, Helvetica, sans-serif; font-weight: bold;\">มหาวิทยาลัยเทคโนโลยี
พระจอมเกล้าธนบุรี</span><span style=\"font-size:14px; mso-spacerun: yes; color:
#FF6600;\"><span style=\"mso-fareast-font-family:; font-family: Arial, Helvetica, sans-serif;
font-weight: bold;\">"
// . "                </marquee>"
// . "              </span><span style=\"font-size:14px; mso-spacerun: yes\"><span style=\"mso-
fareast-font-family:; font-family: Arial, Helvetica, sans-serif; font-weight: bold;\">
</span></span></span></span></span></span></td>"
// . "            </tr>"
// . "          </table>"
//  . "<br>"
//  . "<p> </p>"
// . ""
 . "     <table width=\"458\" border=\"1\" align=\"center\" bordercolor=\"#FF6600\">"
 . "       <tr>"
 . "         <td width=\"700\"><div align=\"center\" class=\"style6\"><b><font
color=\"#515180\">รายการอุปกรณ์ที่ควบคุมการทำงาน</font></b></div></td>"
 . "       </tr>"
 . "     </table>     "
 . "<table border=\"1\" width=\"408\" align=\"center\" bgcolor=\"#2C2C57\">"
```

```
 . "    <tr>"
 . "        <td width=\"86\">"
 . "            <p align=\"center\"><img src=\"$im1\" width=\"86\" height=\"36\"
border=\"0\"></p>"
 . "        </td>"
 . "        <td width=\"86\">"
 . "            <p align=\"center\"><img src=\"$im2\" width=\"86\" height=\"36\"
border=\"0\"></p>"
 . "        </td>"
 . "        <td width=\"86\">"
 . "            <p align=\"center\"><img src=\"$im3\" width=\"86\" height=\"36\"
border=\"0\"></p>"
 . "        </td>"
 . "        <td width=\"86\">"
 . "            <p align=\"center\"><img src=\"$im4\" width=\"86\" height=\"36\"
border=\"0\"></p>"
 . "        </td>"
 . "        <td width=\"86\">"
 . "            <p align=\"center\"><img src=\"$im5\" width=\"86\" height=\"36\"
border=\"0\"></p>"
 . "        </td>" . "    </tr>"
 . "    <tr>"
 . "        <td width=\"86\">"
 . "            <p align=\"center\"><a href=\"11.php\"><img src=\"image/on1.gif\" width=\"86\"
height=\"36\" border=\"0\"></a><a href=\"10.php\"><img src=\"image/off1.gif\" width=\"86\"
height=\"36\" border=\"0\"></a></p>"
 . "        </td>"
 . "        <td width=\"86\">"
```

```
 . "           <p align=\"center\"><a href=\"21.php\"><img src=\"image/on1.gif\" width=\"86\"
height=\"36\" border=\"0\"></a><a href=\"20.php\"><img src=\"image/off1.gif\" width=\"86\"
height=\"36\" border=\"0\"></a></p>"
 . "       </td>"
 . "       <td width=\"86\">"
 . "           <p align=\"center\"><a href=\"31.php\"><img src=\"image/on1.gif\" width=\"86\"
height=\"36\" border=\"0\"></a><a href=\"30.php\"><img src=\"image/off1.gif\" width=\"86\"
height=\"36\" border=\"0\"></a></p>"
 . "       </td>"
 . "       <td width=\"86\">"
 . "           <p align=\"center\"><a href=\"41.php\"><img src=\"image/on1.gif\" width=\"86\"
height=\"36\" border=\"0\"></a><a href=\"40.php\"><img src=\"image/off1.gif\" width=\"86\"
height=\"36\" border=\"0\"></a></p>"
 . "       </td>"
. "       <td width=\"86\">"
 . "           <p align=\"center\"><a href=\"51.php\"><img src=\"image/on1.gif\" width=\"86\"
height=\"36\" border=\"0\"></a><a href=\"50.php\"><img src=\"image/off1.gif\" width=\"86\"
height=\"36\" border=\"0\"></a></p>"
 . "       </td>"

 . "   </tr>"

. "   <tr>"
 . "       <td width=\"86\">"
 . "           <p align=\"center\"><font color=\"#99999999\"><b>หลอดที่1</p>"
 . "       </td>"
 . "       <td width=\"86\">"
 . "           <p align=\"center\"><font color=\"#99999999\"><b>หลอดที่2</p>"
 . "       </td>"
 . "       <td width=\"86\">"
```

```
. "           <p align=\"center\"><font color=\"#99999999\"><b>หลอดที่3</p>"
 . "       </td>"
  . "       <td width=\"86\">"
 . "           <p align=\"center\"><font color=\"#99999999\"><b>หลอดที่4</p>"
 . "       </td>"
 . "       <td width=\"86\">"
 . "           <p align=\"center\"><font color=\"#99999999\"><b>แอร์</p>"
 . "       </td>"

 . "   </tr>"

 . "   <tr>"
 . "       <td width=\"86\">"
 . "           <p align=\"center\"><img src=\"$im51\" width=\"86\" height=\"36\"
border=\"0\"></p>"
 . "       </td>"
 . "       <td width=\"86\">"
 . "           <p align=\"center\"><img src=\"$im41\" width=\"86\" height=\"36\"
border=\"0\"></p>"
 . "       </td>"
 . "       <td width=\"86\">"
 . "           <p align=\"center\"><img src=\"$im21\" width=\"86\" height=\"36\"
border=\"0\"></p>"
 . "       </td>"
 . "       <td width=\"86\">"
 . "           <p align=\"center\"><img src=\"$im31\" width=\"86\" height=\"36\"
border=\"0\"></p>"
 . "       </td>"
```

```
  . "       <td width=\"86\">"

  . "           <p align=\"center\"><img src=\"$im11\" width=\"86\" height=\"36\"
border=\"0\"></p>"
  . "       </td>" . "    </tr>"

  . "</table>"
  //. "<p> </p>"

  ."<br>"

  ."<div align=\"center\"><font face=\"DSN PreeCha,Arial\"><span style=\"font-
size:24pt;\"><font color=\"blue\">อุณหภูมิ  </font> <font color=\"red\">  $tmp1 </font>
<font color=\"blue\">องศาเซลเซียส  </span></font></div>  </span>"

  . "</body>"
 ."";
?>
<td width=100% style='width:610.7pt;padding:0cm 5.4pt 0cm 5.4pt'>
 <p class=MsoNormal align=center style='text-align:center'><b><span lang=TH
  style='font-size:20.0pt;font-family:"Angsana New";mso-ascii-font-family:"Times New Roman";
  mso-hansi-font-family:"Times New Roman"'></font> <font color=\"blue\">คณะ
วิศวกรรมศาสตร์ ภาควิชาวิศวกรรมไฟฟ้า</span></b><b><span
  style='font-size:18.0pt'><o:p></o:p></span></b></p>
```

ประวัติการศึกษา และการทำงาน

ชื่อ –นามสกุล	นายธนกร ดุจเพ็ญ
วัน เดือน ปี ที่เกิด	28 กันยายน 2524
สถานที่เกิด	อำเภอเมือง จังหวัดบุรีรัมย์
ประวัติการศึกษา	มัธยมศึกษาโรงเรียนบุรีรัมย์พิทยาคม สายวิทย์-คณิต ปริญญาตรีมหาวิทยาลัยภาคตะวันออกเฉียงเหนือ คณะวิศวกรรมศาสตร์ ภาควิชาวิศวกรรมไฟฟ้า สาขาอิเล็กทรอนิกส์
ตำแหน่งหน้าที่การงานปัจจุบัน	-
สถานที่ทำงานปัจจุบัน	-
ผลงานดีเด่นและรางวัลทางวิชาการ	-
ทุนการศึกษาที่ได้รับ	-