ภาควิชาครุศาสตร์วิศวกรรม

คณะครุศาสตร์อุตสาหกรรม

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ใบรับรองปริญญานิพนธ์

ชื่อหัวข้อ เครื่องสั่งอาหารระบบอัตโนมัติ

Automatic Food Order Machine

ชื่อนักศึกษา

1. นางสาวเกษร	เดชสุวรรณ	รหัสประจำตัว	45035331
2. นายจาตุรงค์	ศรโสภา	รหัสประจำตัว	45035334
3. นายทศพล	เพ็ชรภิมล	รหัสประจำตัว	45035341
4. นายเอกราช	ศรีสุข	รหัสประจำตัว	45035372

หลักสูตร ครุศาสตร์อุตสาหกรรมบัณฑิต **สาขาวิชา** อิเล็กทรอนิกส์และคอมพิวเตอร์

อาจารย์ที่ปรึกษา ผศ.กิติพงศ์ มะโน

อาจารย์ที่ปรึกษาร่วม อาจารย์สุรพงษ์ สิริพงศ์ดี

คณะกรรมการสอบปริญญานิพนธ์		ลายมือชื่อ
1. อาจารย์พงษ์เกียรติ	เชษฐพิทักษ์สกุล	
2. ผศ.กิติพงศ์	มะโน	
3. อาจารย์สุรพงษ์	สิริพงศ์ดี	
4. อาจารย์ประเสริฐ	เดนพันค้อ	
5. อาจารย์พิชญ์สินี	มงคลขจิต	

วัน/เดือน/ปีที่สอบ วันศุกร์ที่ 14 พฤศจิกายน พ.ศ. 2546 เวลา 16:00 น.

สถานที่สอบ ห้อง ค.311 คณะครุศาสตร์อุตสาหกรรม สจล.

ภาควิชารับรองแล้ว

ลงนาม.....

(นายสุรสิทธิ์ ราตรี)

หัวหน้าภาควิชาครุศาสตร์วิศวกรรม

วันที่ 31 เดือน พ.ค. พ.ศ. 47

<BT4610142>

เครื่องสั่งอาหารระบบอัตโนมัติ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปริญญานิพนธ์

เครื่องสั่งอาหารระบบอัตโนมัติ

AUTOMATIC FOOD ORDER MACHINES

T051870

นางสาวเกษร เดชสุวรรณ
นายจาตุรงค์ ศรโสภา
นายทศพล เพ็ชรภิมล
นายเอกราช ศรีสุข

ปริญญานิพนธ์ฉบับนี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรครุศาสตร์อุตสาหกรรมบัณฑิต
สาขาวิชาอิเล็กทรอนิกส์และคอมพิวเตอร์
ภาควิชาครุศาสตร์วิศวกรรม คณะครุศาสตร์อุตสาหกรรม
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
ปีการศึกษา 2546

เลขหมู่.....
เลขทะเบียน 51870
วัน,เดือน,ปี - 3 ส.ค. 2547
.b.....
.i.....

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ปริญญานิพนธ์

เรื่อง เครื่องสั่งอาหารระบบอัตโนมัติ

Automatic Food Order Machines

วัตถุประสงค์

1. เพื่อศึกษาการทำงานของไมโครคอนโทรลเลอร์, โปรแกรมวิชวลเบสิก, ฐานข้อมูล และระบบการทำงานของเครื่องส่งและเครื่องรับ
2. เพื่อออกแบบเครื่องสั่งอาหารระบบอัตโนมัติ
3. เพื่อสร้างเครื่องสั่งอาหารระบบอัตโนมัติ
4. เพื่อทดสอบการทำงานของเครื่องสั่งอาหารระบบอัตโนมัติ
5. เพื่อนำเครื่องสั่งอาหารระบบอัตโนมัติไปใช้งาน

ประโยชน์ที่คาดว่าจะได้รับ

1. ทำให้เข้าใจระบบการทำงานของไมโครคอนโทรลเลอร์ โปรแกรมวิชวลเบสิกเกี่ยวกับฐานข้อมูล ระบบการทำงานของเครื่องรับ เครื่องส่ง พอร์ตอนุกรมและสามารถนำไปประยุกต์ใช้งานได้
2. ได้โครงสร้างเครื่องสั่งอาหารระบบอัตโนมัติ
3. ได้เครื่องสั่งอาหารระบบอัตโนมัติ
4. ได้ผลการทดสอบของเครื่องสั่งอาหารระบบอัตโนมัติ
5. นำเครื่องสั่งอาหารระบบอัตโนมัติไปใช้งานจริงได้
6. สามารถนำระบบของเครื่องสั่งอาหารระบบอัตโนมัติไปพัฒนาเป็นระบบอื่นๆ ได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ชื่อหัวข้อ	เครื่องสั่งอาหารระบบอัตโนมัติ
นักศึกษา	นางสาวเกษร เดชสุวรรณ
	นายจาตุรงค์ ศรโสภา
	นายทศพล เพ็ชรภิมล
	นายเอกราช ศรีสุข
อาจารย์ที่ปรึกษา	ผศ.กิติพงค์ มะโน
อาจารย์ที่ปรึกษาร่วม	อาจารย์สุรพงษ์ สิริพงศ์ดี
หลักสูตร	ครุศาสตร์อุตสาหกรรมบัณฑิต
สาขาวิชา	อิเล็กทรอนิกส์และคอมพิวเตอร์
ปีการศึกษา	2546

บทคัดย่อ

ปริญญานิพนธ์ฉบับนี้นำเสนอการออกแบบและการสร้างเครื่องสั่งอาหารระบบอัตโนมัติ โดยระบบนี้ประกอบด้วยส่วนสำคัญ 2 ส่วน คือ ส่วนภาคส่งใช้ไมโครคอนโทรลเลอร์ AT89C52 ทำหน้าที่แสดงผล ด้วยจอแอลซีดีและส่งข้อมูลไปยังภาครับ ส่วนภาครับใช้ไมโครคอนโทรลเลอร์ AT89C52 ทำหน้าที่รับข้อมูลมาประมวลผลในฐานข้อมูลด้วยคอมพิวเตอร์ และพิมพ์ผลด้วยเครื่องพิมพ์ สามารถนำผลการพิมพ์ใช้สั่งอาหารได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Thesis Title	Automatic Food Order Machines
Students	Miss Kasorn Dachsuwan
	Mr.Jaturang Sornsobha
	Mr.Totsapol Phetphimol
	Mr.Akaraj Srisuk
Advisor	Asst.Prof.Kitipong Mano
Co-Advisor	Mr.Surapong Siripongdee
Education Level	Bachelor of Science in Industrial Education
Program in	Industrial Instrument Technology
Academic Year	2003

ABSTRACT

This thesis presents a design and implementation of Automatic Food Order Machines. The system consists of important. The transmitter use Microcontroller AT89C52 with LCD display and send data to the receiver.The receives use Microcontroller AT89C52 too.Which receivers the order and process by computer to print out.The order for the chief.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กิตติกรรมประกาศ

ปริญญานิพนธ์ฉบับนี้สามารถลุล่วงไปด้วยดี เนื่องมาจากความร่วมมือของสมาชิกภายในกลุ่มทุกท่าน ขอขอบคุณ ผู้ช่วยศาสตราจารย์กิติพงค์ มะโน อาจารย์สุระพงษ์ สิริพงศ์ดี ภาควิชาครุศาสตร์วิศวกรรม คณะครุศาสตร์อุตสาหกรรม และอาจารย์ประจำภาควิชาครุศาสตร์วิศวกรรมทุกท่านที่ให้ความอนุเคราะห์ เครื่องมือและอุปกรณ์ รวมทั้งยังให้คำแนะนำ แนวความคิด แนวความรู้ต่างๆ และแนวทางการแก้ปัญหาในการจัดทำปริญญานิพนธ์ ขอขอบคุณห้องสมุดคณะครุศาสตร์อุตสาหกรรม ห้องสมุดคณะวิศวกรรมศาสตร์ และหอสมุดกลาง ที่ช่วยอำนวยความสะดวกและเอื้อเฟื้อสถานที่ในการค้นคว้าหาข้อมูล สุดท้ายที่ควรระลึกถึงอย่างยิ่ง คือ บิดาและมารดาที่เป็นผู้ให้การสนับสนุนด้านการศึกษาและเป็นผู้ให้กำลังใจด้วยดีตลอดมาตั้งแต่อดีตจนถึงปัจจุบัน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ

เรื่อง	หน้า
บทคัดย่อภาษาไทย	I
บทคัดย่อภาษาอังกฤษ	II
กิตติกรรมประกาศ	III
สารบัญ	IV
สารบัญตาราง	VII
สารบัญรูป	VIII
บทที่ 1 บทนำ	1
1.1 ความเป็นมาและความสำคัญ	1
1.2 ขีดความสามารถของโครงงาน	1
1.3 เนื้อหาโดยสังเขป	2
บทที่ 2 ทฤษฎีและหลักการ	3
2.1 กล่าวนำ	3
2.2 ไมโครคอนโทรลเลอร์เบอร์ 8051	3
2.2.1 คุณสมบัติของไมโครคอนโทรลเลอร์ 8051	4
2.2.2 โครงสร้างของไมโครคอนโทรลเลอร์ 8051	4
2.3 ความรู้เบื้องต้นเกี่ยวพอร์ตอนุกรม	6
2.3.1 การสื่อสารข้อมูลแบบซิงโครนัส	6
2.3.2 การสื่อสารข้อมูลแบบอะซิงโครนัส	7
2.4 พอร์ตอนุกรม RS-232	9
2.4.1 คอนเน็กเตอร์สำหรับพอร์ตอนุกรม RS-232	9
2.4.2 การเชื่อมต่อคอนเน็กเตอร์สำหรับพอร์ตอนุกรม RS-232	10
2.4.3 รีจีสเตอร์พอร์ตอนุกรม RS-232	13
2.5 ฐานข้อมูล	14
2.5.1 องค์ประกอบของฐานข้อมูล	15
2.5.2 ประโยชน์จากการประมวลผลด้วยฐานข้อมูล	15
2.6 พื้นฐานการสื่อสารผ่านพอร์ตอนุกรมด้วยวิชวลเบสิก	16
2.6.1 การรับและส่งข้อมูลโดยคอนโทรล MSComm	18

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ (ต่อ)

เรื่อง	หน้า
2.6.2 การใช้งานคอนโทรล MSComm	18
บทที่ 3 การออกแบบ การสร้าง และการทำงาน	20
3.1 กล่าวนำ	20
3.2 เครื่องส่ง	20
3.3 เครื่องรับ	23
3.4 ฐานข้อมูล	24
3.4.1 การออกแบบฐานข้อมูล	24
3.4.2 การออกแบบโปรแกรมเพิ่มรายการอาหาร	25
3.4.3 การออกแบบโปรแกรมรับรายการอาหาร	25
3.4.4 การออกแบบโปรแกรมพิมพ์ใบเสร็จรับเงิน	26
3.4.5 ขั้นตอนการออกแบบฐานข้อมูล	27
บทที่ 4 การทดลองและผลการทดลอง	36
4.1 บทนำ	36
4.2 ส่วนการแสดงผล	36
4.2.1 การทดลอง	36
4.2.2 ผลการทดลอง	36
4.3 การติดต่อระหว่างเครื่องส่งกับพอร์ตอนุกรม	37
4.3.1 การทดลอง	37
4.3.2 ผลการทดลอง	37
4.4 การติดต่อระหว่างเครื่องรับกับเครื่องส่ง	38
4.4.1 การทดลอง	38
4.4.2 ผลการทดลอง	38
4.5 การเพิ่มรายการอาหาร	38
4.5.1 การทดลอง	38
4.5.2 ผลการทดลอง	38
4.6 การค้นหารายการอาหาร	40
4.6.1 การทดลอง	40
4.6.2 ผลการทดลอง	40

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ (ต่อ)

เรื่อง	หน้า
4.7 เครื่องสั่งอาหารระบบอัตโนมัติ	42
4.7.1 การทดลอง	42
4.7.2 ผลการทดลอง	42
บทที่ 5 บทสรุป	44
5.1 สรุป	44
5.2 ปัญหาและแนวทางแก้ไข	44
5.3 แนวทางการพัฒนา	45
บรรณานุกรม	46
ภาคผนวก ก เครื่องต้นแบบ	47
ภาคผนวก ข วงจรและแผ่นพิมพ์	50
ภาคผนวก ค รายการอุปกรณ์	56
ภาคผนวก ง แผนผังการทำงานและรหัสต้นฉบับของโปรแกรม	61
ภาคผนวก จ คู่มือการใช้งาน	80
ภาคผนวก ฉ รายละเอียดและคุณสมบัติของอุปกรณ์	86
ประวัติผู้แต่ง	108

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญตาราง

ตารางที่	หน้า
2.1 ไมโครคอนโทรลเลอร์ตระกูล 8051	3
2.2 การจัดขาสัญญาณของพอร์ตอนุกรมแบบต่างๆ และหน้าที่การทำงาน	11
4.1 การทดลองส่วนการแสดงผล	36
4.2 การทดลองส่งข้อมูลโดยติดต่อผ่านพอร์ตอนุกรม	37
ค.1 รายการอุปกรณ์ของวงจรแสดงผล	57
ค.1 (ต่อ)รายการอุปกรณ์ของวงจรแสดงผล	58
ค.2 รายการอุปกรณ์ของวงจรเครื่องส่ง	58
ค.2 (ต่อ)รายการอุปกรณ์ของวงจรเครื่องส่ง	59
ค.3 รายการอุปกรณ์ของวงจรเครื่องรับ	60

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูป

รูปที่	หน้า
2.1 โครงสร้างของ 8051	5
2.2 ไดอะแกรมเวลาของการสื่อสารข้อมูลแบบซิงโครนัส	7
2.3 รูปแบบของข้อมูลแบบอะซิงโครนัส	8
2.4 คอนเน็กเตอร์อนุกรม	10
2.5 การต่ออนุกรมภายนอกเข้ากับคอมพิวเตอร์ผ่านทางพอร์ตอนุกรมแบบต่างๆ	12
3.1 ผังงานของเครื่องสั่งอาหารระบบอัตโนมัติ	20
3.2 วงจรเครื่องส่งสัญญาณ	21
3.3 วงจรส่วนการแสดงผล	22
3.4 วงจรเครื่องรับ	23
3.5 ผังการทำงานของโปรแกรมในฐานข้อมูล	24
3.6 ผังการทำงานของโปรแกรมเพิ่มรายการอาหาร	25
3.7 ผังการทำงานของโปรแกรมรับรายการอาหาร	26
3.8 ผังการทำงานของโปรแกรมพิมพ์ใบเสร็จ	27
3.9 การเปิดโปรแกรม Visual Basic 6.0	28
3.10 การสร้างฟอร์มเพื่อติดต่อฐานข้อมูล	29
3.11 การสร้างฟอร์มหลักในการใช้งาน	29
3.12 การสร้างฟอร์มเพิ่มรายการอาหารและค้นหารายการอาหาร	30
3.12 (ต่อ) การสร้างฟอร์มเพิ่มรายการอาหารและค้นหารายการอาหาร	31
3.13 การสร้างฟอร์มรับรายการอาหารที่มีการสั่ง	32
3.14 การสร้างฟอร์มที่ใช้ในการพิมพ์ใบเสร็จ	33
3.15 ฟอร์มหลักที่ใช้งานฐานข้อมูล	33
3.16 ฟอร์มเพิ่มรายการอาหารและค้นหารายการอาหาร	34
3.17 ฟอร์มรับรายการอาหารที่มีการสั่ง	34
3.18 ฟอร์มที่ใช้ในการพิมพ์ใบเสร็จ	35
3.19 รูปแบบใบเสร็จ	35
4.1 เมนูใช้ในการเพิ่มรายการอาหาร	39
4.2 การเพิ่มรายการอาหารในเมนู	39

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูป (ต่อ)

รูปที่	หน้า
4.3 เมนูใช้ในการค้นหารายการอาหาร	41
4.4 การค้นหารายการอาหาร	41
4.5 การพิมพ์ใบเสร็จรับเงิน	43
4.6 ใบเสร็จรับเงิน	43
ก.1 ภาพด้านหน้าของเครื่องส่ง	48
ก.2 ภาพด้านหน้าของเครื่องรับ	48
ก.3 ภาพด้านข้างของเครื่องรับ	49
ก.4 ภาพการเชื่อมต่อระหว่างเครื่องลูกข่ายกับเครื่องรับ	49
ข.1 วงจรเครื่องรับ	51
ข.2 แผ่นวงจรพิมพ์วงจรเครื่องรับ	51
ข.3 ตำแหน่งการวางอุปกรณ์แผ่นวงจรพิมพ์วงจรเครื่องรับ	52
ข.4 แผ่นวงจรพิมพ์ของวงจร MAX 232	52
ข.5 ตำแหน่งการวางอุปกรณ์วงจรพิมพ์ของวงจร MAX 232	52
ข.6 วงจรเครื่องส่ง	53
ข.7 วงจรส่วนการแสดงผล	54
ข.8 แผ่นวงจรพิมพ์วงจรเครื่องส่งและส่วนการแสดงผล	54
ข.9 ตำแหน่งการวางอุปกรณ์แผ่นวงจรพิมพ์วงจรเครื่องส่งและส่วนการแสดงผล	55
ง.1 ผังงานของโปรแกรมควบคุมการติดต่อกับพอร์ตอนุกรม	62
จ.1 ส่วนประกอบและปุ่มควบคุมของเครื่องสั่งอาหารระบบอัตโนมัติ	82

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 1
บทนำ

1.1 ความเป็นมาและความสำคัญ

ปัจจุบันธุรกิจร้านอาหารเป็นธุรกิจที่มีการแข่งขันกันมาก อันเนื่องมาจากเป็นธุรกิจที่สามารถสร้างกำไรได้ง่ายโดยไม่ต้องใช้เวลานาน เป็นแรงจูงใจให้คนที่มีเงินทุนผันตัวเองมาทำธุรกิจนี้กันมาก ดังนั้นร้านอาหารแต่ละร้านจำเป็นจะต้องหาวิธีการต่างๆ ขึ้นมาใช้ ในการสร้างแรงจูงใจลูกค้าให้มาใช้บริการของทางร้าน ปัญหาต่างๆ ที่เกิดขึ้นในร้านอาหารก็มีอยู่มากมาย โดยเฉพาะร้านอาหารที่มีขนาดใหญ่ มีพื้นที่กว้าง ทำให้ไม่สะดวกและล่าช้าในการสั่งอาหาร เนื่องจาก บริกรต้องใช้เวลาในการไปสั่งอาหาร กระบวนการสั่งอาหารบริกรต้องจดรายการอาหาร ต้องรอรับอาหารตามที่ลูกค้าต้องการ กรณีที่มีลูกค้าเข้ามาใช้บริการเป็นจำนวนมาก จำนวนบริกรมีไม่เพียงพอกับจำนวนของลูกค้าทำให้บริการไม่ทั่วถึง อาจทำให้ลูกค้าไม่พอใจและไม่ประทับใจในการบริการ ส่งผลให้ทางร้านขาดรายได้ จากปัญหาที่เกิดขึ้นในการสั่งอาหาร ทำให้ทางร้านอาหารต่างๆ ใช้วิธีการแก้ปัญหา โดยการเพิ่มจำนวนบุคลากร อบรมให้พนักงานมีประสิทธิภาพในการบริการ จัดให้มีบรรยากาศที่ดี จัดสภาพร้านให้ถูกสุขลักษณะ ปรับปรุงรสชาติของอาหาร เพิ่มโปรโมชั่นใหม่ๆ ให้แก่ลูกค้า

วิธีแก้ปัญหาดังกล่าว ทำให้สิ้นเปลืองต้นทุน เสียเวลา ยุ่งยากในการอบรมและจัดหาพนักงาน เพื่อแก้ปัญหาที่เกิดขึ้นกับร้านอาหารจึงเกิดแนวคิดในการสร้างโครงงานนี้ขึ้นมา เพื่อลดปัญหาต่างๆ ให้แก่ร้านอาหารและสร้างความประทับใจแก่ผู้มาใช้บริการของร้านอาหาร

1.2 ขีดความสามารถของโครงงาน

1. เครื่องสั่งอาหารสามารถบรรจุรายการอาหารได้ 30 รายการ
2. มีการรับส่งข้อมูลจากเครื่องสั่งอาหารไปยังเครื่องรับรายการอาหาร
3. มีระบบฐานข้อมูลประมวลผลทางสถิติเกี่ยวกับการจำหน่ายอาหารบนเครื่องคอมพิวเตอร์ส่วนบุคคล
4. สามารถพิมพ์ใบสั่งอาหารได้
5. สามารถพิมพ์บิลค่าอาหารได้
6. ทำเครื่องส่ง 4 เครื่อง
7. โปรแกรมเครื่องรับสามารถรองรับการทำงานได้ 16 โต๊ะ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

8. เครื่องส่งสามารถเปลี่ยนเบอร์โต๊ะได้

1.3 เนื้อหาโดยสังเขป

ปริญญานิพนธ์ฉบับนี้มีเนื้อหาทั้งหมด 5 บท ดังต่อไปนี้

บทที่ 1 บทนำ ซึ่งมีเนื้อหาเกี่ยวกับ ความเป็นมาและความสำคัญที่ทำให้เกิดโครงงานนี้ขึ้นมา รวมทั้งยังกล่าวถึงวัตถุประสงค์ ขอบเขตและประโยชน์ของการทำปริญญานิพนธ์ในครั้งนี้

บทที่ 2 ทฤษฎีและหลักการ กล่าวถึงเนื้อหาที่นำมาอ้างอิง ใช้เป็นแนวทางในการออกแบบและสร้างเครื่องสั่งอาหารระบบอัตโนมัติ

บทที่ 3 การออกแบบและการสร้าง กล่าวถึงเนื้อหาโดยละเอียดตั้งแต่ขั้นตอนในการออกแบบ วงจรในส่วนของตัวรับและตัวส่ง การนำส่วนต่างๆ มาประกอบและทำงานร่วมกันเพื่อให้ทำงานร่วมกันอย่างประสิทธิภาพ

บทที่ 4 การทดลองและผลการทดลอง เป็นการนำเสนอในส่วนของการทดลองและผลการทดลองโดยแบ่งการทดลองออกเป็นส่วนๆ ตามการออกแบบ พร้อมบันทึกการทดลองในแต่ละส่วน

บทที่ 5 บทสรุป ปัญหาแนวทางแก้ไขและพัฒนา เป็นการสรุปเกี่ยวกับความสามารถประสิทธิภาพการทำงานของเครื่องสั่งอาหารระบบอัตโนมัติและกล่าวถึงปัญหาที่เกิดนับตั้งแต่การเริ่มสร้างโครงงานจนกระทั่งโครงงานเสร็จสมบูรณ์ ตลอดจนแนวทางแก้ไขปัญหา พร้อมทั้งเสนอแนวทางการพัฒนาระบบของเครื่องสั่งอาหารระบบอัตโนมัติให้สามารถนำไปใช้งานได้และปรับปรุงให้ประสิทธิภาพมากยิ่งขึ้น

ภาคผนวก ก แสดงภาพเครื่องต้นแบบ การติดตั้ง การเชื่อมต่อกับอุปกรณ์อื่นๆ ขณะใช้งานจริง

ภาคผนวก ข ประกอบด้วยแผนผังรายละเอียดของวงจรและแผ่นวงจรพิมพ์

ภาคผนวก ค แสดงรายการอุปกรณ์ที่ใช้งานในแต่ละวงจร

ภาคผนวก ง แสดงแผนผังการทำงานและรหัสต้นฉบับของโปรแกรมทั้งหมดที่สร้างขึ้นเพื่อประกอบการทำงานของโครงงาน

ภาคผนวก จ เป็นคู่มือการใช้งานของเครื่องสั่งอาหารระบบอัตโนมัติ

ภาคผนวก ฉ แสดงรายละเอียดและคุณสมบัติของอุปกรณ์สำคัญที่ใช้ในโครงงาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 2
ทฤษฎีและหลักการ

2.1 กล่าวนำ

เนื้อหาของปริญญานิพนธ์ในบทนี้เป็นทฤษฎีและหลักการ ที่นำมาใช้ประกอบในโครงงาน โดยจะประกอบด้วยทฤษฎี หลักการที่เกี่ยวข้องกับคุณสมบัติ การทำงานของไมโครคอนโทรลเลอร์ MCS-51, พอร์ตอนุกรม และฐานข้อมูล

2.2 ไมโครคอนโทรลเลอร์เบอร์ 8051

ไมโครคอนโทรลเลอร์ในตระกูล MCS-51ที่มีขนาด 8 บิต ทุกๆ เบอร์จะมีสถาปัตยกรรมพื้นฐานที่เหมือนกัน เพื่อความเหมาะสมในการนำเอาไปใช้งานตามความต้องการต่างๆ แต่เดิม 8051 ถูกสร้างด้วยวิธี HMOS I แต่ในปัจจุบันได้สร้างด้วยวิธี HMOS II จึงมีชื่อเป็น 8051AH ไมโครคอนโทรลเลอร์ตระกูล 51 นั้น ถึงแม้ว่าจะมีหลายเบอร์แต่เราก็จะเรียกว่าเป็น “8051” ซึ่งหมายถึงไมโครคอนโทรลเลอร์ตระกูล 51 ส่วนเบอร์ 8032 และ 8052 มีหน่วยความจำภายในเพิ่มขึ้นและมีวงจรนับ/จับเวลา ขนาด 16 บิต เพิ่มขึ้นมาดังตารางที่ 2.1

ตารางที่ 2.1 ไมโครคอนโทรลเลอร์ตระกูล 8051

เบอร์	หน่วยความจำภายใน		จำนวนไทเมอร์/เคาน์เตอร์	อินเตอร์รัพต์หมายเลข
	เก็บโปรแกรม	เก็บข้อมูล		
8052H	8K x 8 ROM	256 x 8 ROM	3 x 16 Bit	6
8051H	4K x 8 ROM	128 x 8 ROM	2 x 16 Bit	5
8051	4K x 8 ROM	256 x 8 ROM	2 x 16 Bit	5
8032AH	ไม่มี	128 x 8 ROM	3 x 16 Bit	6
8031AH	ไม่มี	128 x 8 ROM	2 x 16 Bit	5
8031	ไม่มี	128 x 8 ROM	2 x 16 Bit	5
8751H	4K x 8 EPROM	128 x 8 ROM	2 x 16 Bit	5
80751H-12	4K x 8 EPROM	128 x 8 ROM	2 x 16 Bit	5

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.2.1 คุณสมบัติของไมโครคอนโทรลเลอร์ตระกูล MCS-51

1) ต้องการแหล่งจ่ายไฟ +5 โวลต์ ชุดเดียว

2) มีหน่วยความจำโปรแกรม (Program Memory) ขนาด 4 กิโลไบต์ สำหรับเบอร์ 8051 และ 8031 สำหรับเบอร์ 8052 มีหน่วยความจำถึง 8 กิโลไบต์

3) มีหน่วยความจำสำหรับเก็บข้อมูล (Data Memory) ขนาด 128 ไบต์ สำหรับเบอร์ 8052 ขึ้นไปมีถึง 256 ไบต์

4) มีหน่วยความจำสำหรับเก็บโปรแกรมและข้อมูลแยกจากกันอย่างละ 64 กิโลไบต์

5) มีไทเมอร์เคาน์เตอร์ ขนาด 16 บิต 2 ชุด (สำหรับเบอร์ 8052 มี 3 ชุด) ทำงานได้ 4 โหมด

6) รับอินเตอร์รัพท์ได้ 6 แหล่ง 5 เวกเตอร์ สำหรับเบอร์ 8052 ขึ้นไปมี 8 แหล่ง 6 เวกเตอร์

7) มีพอร์ตรับส่งข้อมูลอนุกรม (UART) 2 พอร์ตแบบ Full Duplex เลือกรูปได้ 4 โหมด

8) มีคำสั่งในการทำ AND, OR หรือ Complement ได้ทั้งแบบ 8 บิตและ 1 บิต

9) มีวงจรออสซิลเลเตอร์ภายใน

2.2.2 โครงสร้างของไมโครคอนโทรลเลอร์ตระกูล MCS-51

ภายใน 8051 จะประกอบด้วยเกตชนิดต่างๆ เช่น วงจรถอดรหัสคำสั่ง, วงจรสัญญาณนาฬิกา เป็นต้น โครงสร้างภายในของ 8051 จะประกอบด้วยส่วนย่อยๆ ดังรูปที่ 2.1 โครงสร้างของ 8051 จะประกอบด้วย 3 ส่วน หลักๆ ดังนี้

1) ซีพียู (Central Processing Unit) ส่วนนี้ทำหน้าที่สร้างสัญญาณควบคุมการติดต่อกับส่วนอื่นๆ เรียกว่า วงจรควบคุม (Control Unit) สัญญาณที่สร้างจากวงจรควบคุม ได้แก่ สัญญาณสำหรับการติดต่อกับหน่วยความจำ, อุปกรณ์รับข้อมูลเข้าหรือส่งข้อมูลออก ซึ่งส่วนควบคุมการขัดจังหวะ และส่วนควบคุมบัสก็เป็นส่วนหนึ่งของวงจรควบคุมด้วย การสร้างสัญญาณวงจรควบคุมจากซีพียูนี้ จะสร้างสัญญาณโดยการถอดรหัสจากคำสั่งที่มีการกำหนดไว้และสัญญาณที่สร้างขึ้นมาจะอ้างอิงกับสัญญาณนาฬิกา ที่สร้างขึ้นจากวงจรออสซิลเลเตอร์ เพื่อให้ทุกๆ ส่วนทำงานประสานกันอย่างถูกต้อง ในซีพียูยังประกอบด้วยส่วนประมวลผล (Arithmetic Logic Unit) ทำหน้าที่ประมวลผลข้อมูล เช่น การบวก, การลบ, การคูณหรือการหาร ข้อมูลแล้วนำผลลัพธ์ไปเก็บไว้ในหน่วยความจำที่ต้องการ

2) หน่วยความจำ (Memory) มีไว้สำหรับจดจำข้อมูล ซึ่งในการนำข้อมูลเข้าและออกจากหน่วยความจำ จำเป็นต้องรู้ตำแหน่งของหน่วยความจำ (Address) ในการนำข้อมูลเข้าไปเก็บในหน่วยความจำเรียกว่า การเขียนข้อมูลและการนำเข้าข้อมูลออกจากหน่วยความจำ เรียกว่า การอ่านข้อมูล ในไมโครคอนโทรลเลอร์ 8051 ข้อมูลในแต่ละตำแหน่งจะมีขนาด 8 บิต ดังนั้นแต่ละ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

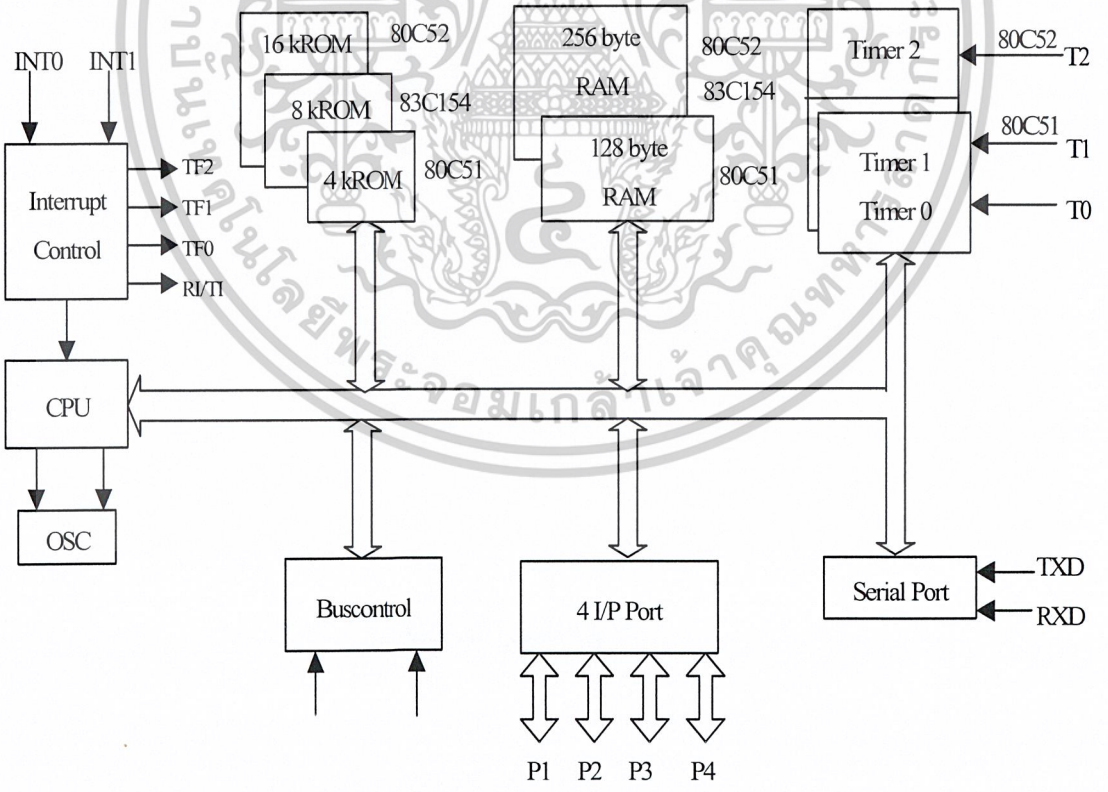
ตำแหน่งของหน่วยความจำสามารถเก็บข้อมูลมีค่าได้ระหว่าง 00000000_2 ถึง 11111111_2 หรือ 00H ถึง 0FF ในการติดต่อหน่วยความจำจะต้องมีสัญญาณ 3 กลุ่ม คือ

2.1) ตำแหน่งที่ต้องการติดต่อกับหน่วยความจำ ซึ่งในไมโครคอนโทรลเลอร์ 8051จะมีหน่วยความจำโปรแกรมและหน่วยความจำข้อมูลที่มีขนาดสูงสุดชนิดละ 65536 ตำแหน่ง (64 กิโลไบต์) ดังนั้นการอ้างตำแหน่งของหน่วยความจำจะต้องใช้เส้นแสดงตำแหน่งของหน่วยความจำ จะต้องใช้เส้นแสดงตำแหน่งในเลขฐานสอง ทั้งหมด 16 เส้น (2^{16} เท่ากับ 65536)

2.2) ข้อมูลที่อ่านหรือเขียนกับหน่วยความจำในตำแหน่งที่เราต้องการ

2.3) สัญญาณควบคุมจะส่งไปยังหน่วยความจำ เพื่อจะบอกกับหน่วยความจำว่าต้องอ่านหรือเขียนข้อมูล โดยวงจรถอดรหัสคำสั่ง จะทำการสร้างสัญญาณควบคุมจากคำสั่งที่อ่านเข้ามาจากหน่วยความจำโปรแกรม

3) อุปกรณ์อินพุต/เอาต์พุต (Input/Output Device) เป็นส่วนใช้ส่งข้อมูลเข้าหรือนำข้อมูลออกจากไมโครคอนโทรลเลอร์ 8051 ทำให้สามารถติดต่อกับอุปกรณ์ภายใน อุปกรณ์อินพุตเอาต์พุตได้แก่ อินพุต/เอาต์พุตพอร์ตแบบขนาน วงจรนับเวลา/จับเวลา 0 วงจรนับเวลา/จับเวลา 1 พอร์ตสื่อสารอนุกรม



รูปที่ 2.1 โครงสร้างของ 8051

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.1) พอร์ตขนานเป็นที่สำหรับใช้รับส่งข้อมูลซึ่งเป็นสัญญาณดิจิตอลเข้าหรือออกจากตัวไมโครคอนโทรลเลอร์ 8051 มีทั้งหมด 4 พอร์ต โดยแต่พอร์ตจะรับส่งข้อมูลได้ 8 บิต มีพอร์ต P0, P1, P2 และ P3 บางพอร์ตใช้งานได้มากกว่า 1 อย่าง

3.2) วงจรนับเวลา/จับเวลา 0 และวงจรนับเวลา/จับเวลา 1 เป็นวงจรที่สามารถทำการนับจำนวนไซเคิลของสัญญาณที่ต่อจากภายนอกของไมโครคอนโทรลเลอร์ 8051 หรือจำนวนของสัญญาณนาฬิกาภายในไมโครคอนโทรลเลอร์ 8051 ก็ได้ สามารถตั้งค่าเริ่มต้นของการนับและการค่าการนับได้โดยซีพียู

3.3) พอร์ตอนุกรม ซีพียูจะอ่านข้อมูล พอร์ตอนุกรมเป็นแบบ 8 บิต แต่ละข้อมูลจะถูกส่งออกมาจากไมโครคอนโทรลเลอร์ 8051 เรียงไปทีละบิตออกจากขา TXD ในการรับส่งข้อมูลก็จะรับเข้ามาทีละบิตทางขา RXD และจัดเรียงใหม่เป็น 8 บิต เพื่อให้ซีพียูอ่านไปใช้งานต่อไปในไมโครคอนโทรลเลอร์ 8051 มีพอร์ตใช้งานได้หลายแบบทำให้สะดวกแก่การนำไปใช้งานต่างๆ ได้มากมาย การนำพอร์ตไปใช้งานจะต้องเขียนโปรแกรมขึ้นมาควบคุม

2.3 ความรู้เบื้องต้นเกี่ยวกับพอร์ตอนุกรม

การเคลื่อนย้ายข้อมูลจากคอมพิวเตอร์ไปยังอุปกรณ์ต่อพ่วงภายนอกหรือคอมพิวเตอร์ด้วยกันมีกัน 2 รูปแบบ คือ รับส่งข้อมูลแบบขนานและรับส่งข้อมูลแบบอนุกรม

การรับส่งข้อมูลแบบขนานเป็นการรับและส่งข้อมูลครั้งละ 4 ถึง 8 บิตในเวลาเดียวกัน ทำให้การรับและส่งข้อมูลมีความเร็วสูง แต่จำนวนสายที่ใช้ในการถ่ายทอดข้อมูลมีมากเท่ากับจำนวนบิตของข้อมูลที่ทำการถ่ายทอด นอกจากนั้นยังมีสายที่ใช้สำหรับควบคุมและตรวจสอบการรับส่งข้อมูลด้วย ซึ่งอาจต้องใช้สายมากเป็น 2 เท่าของจำนวนบิตข้อมูล

ในขณะที่การรับส่งข้อมูลแบบอนุกรมจะเป็นการรับส่งข้อมูลครั้งละ 1 บิต โดยมีรูปแบบการรับส่งที่เป็นมาตรฐาน ต้องมีการตรวจสอบความพร้อมในการรับและส่งข้อมูลของตัวส่งและตัวรับ การรับส่งข้อมูลแบบอนุกรมมีข้อดีในเรื่องจำนวนสายสัญญาณที่น้อยมากและไม่แปรผันตามจำนวนบิตของข้อมูล ระยะทางในการรับส่งข้อมูลสูงกว่าแบบขนานมาก

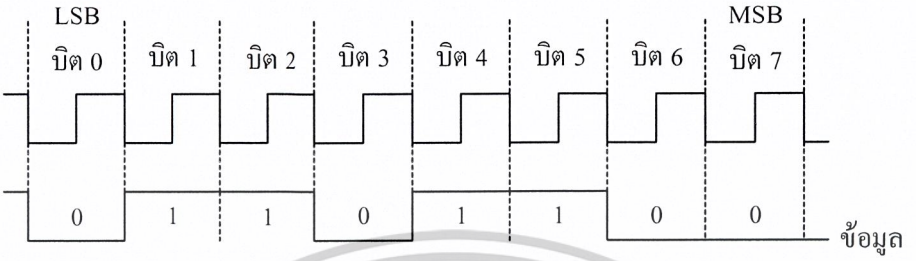
การสื่อสารแบบอนุกรมแบ่งออกเป็น 2 แบบ คือ การสื่อสารอนุกรมแบบซิงโครนัสและการสื่อสารอนุกรมแบบอะซิงโครนัส

2.3.1 การสื่อสารข้อมูลแบบซิงโครนัส

การสื่อสารแบบซิงโครนัสจะมีสัญญาณนาฬิการ่วมอยู่กับการรับและส่งสัญญาณด้วย ตัวอย่างการส่งข้อมูลแบบซิงโครนัส คือ คีย์บอร์ดของคอมพิวเตอร์ ซึ่งสายเส้นหนึ่งจะเป็นสายของ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สัญญาณนาฬิกา ส่วนสายอีกเส้นจะเป็นสายของข้อมูล ดังนั้นการติดต่อกันแบบซิงโครนัสนี้จะต้องใช้สายในการเชื่อมต่ออย่างน้อยที่สุด 3 เส้น คือ สัญญาณนาฬิกา, ข้อมูลและกราวด์ รูปที่ 2.1 แสดงให้เห็นถึงไดอะแกรมเวลาของการสื่อสารข้อมูลแบบซิงโครนัส



รูปที่ 2.2 ไดอะแกรมเวลาของการสื่อสารข้อมูลแบบซิงโครนัส

2.3.2 การสื่อสารข้อมูลแบบอะซิงโครนัส

การสื่อสารข้อมูลแบบอะซิงโครนัส คือ การรับและส่งข้อมูลโดยไม่จำเป็นต้องมีสัญญาณนาฬิการ่วมด้วย แต่จะใช้การกำหนดค่าอัตราความเร็วในการรับและส่งข้อมูลให้มีค่าเท่ากัน ซึ่งเรียกอัตราเร็วนี้ว่า อัตราบอดหรือบอดเรต (Baud Rate) มีหน่วยเป็น บิตต่อวินาที

รูปแบบของข้อมูลที่ใช้ในการรับส่งแบบอะซิงโครนัสประกอบด้วย 4 ส่วนด้วยกัน คือ

1) บิตเริ่มต้น

2) บิตข้อมูลแบบอนุกรม มีขนาด 5, 6, 7 หรือ 8 บิต

3) บิตตรวจสอบพาริตี้ (Parity Bit) มีขนาด 1 บิต หรือไม่มีบิต

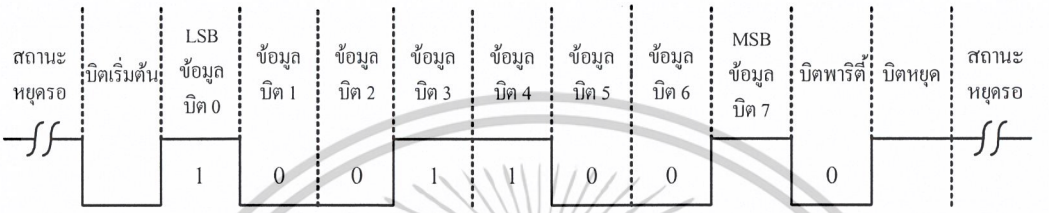
4) บิตปิดท้ายหรือบิตหยุด (Stop Bit) มีขนาด 1, 1.5, หรือ 2 บิต

รูปที่ 2.3 แสดงรูปแบบของข้อมูลอนุกรมแบบอะซิงโครนัส เมื่อไม่มีการส่งข้อมูลขา DATA จะมีสถานะลอจิก “1” เรียกสถานะนี้ว่า สถานะหยุดรอ (Waiting Stage) การเริ่มต้นส่งข้อมูลจะเริ่มจากการให้ขา DATA มีลอจิก “0” ด้วยช่วงระยะเวลา 1 บิต เรียกบิตนี้ว่าบิตเริ่มต้น (Start Bit) จากนั้นบิตข้อมูลจะถูกส่งออกไปโดยเริ่มจากบิตที่มีนัยสำคัญต่ำสุดหรือบิต LSB ก่อน ซึ่งข้อมูลที่ต้องการส่งอาจมีจำนวน 5, 6, 7 หรือ 8 บิตก็ได้ จากนั้นตามด้วยบิตพาริตี้ (Parity Bit) ซึ่งใช้ในการตรวจสอบความผิดพลาดที่เกิดขึ้นจากการส่งข้อมูล บิตสุดท้ายที่จะส่ง คือ บิตปิดท้ายหรือบิตหยุด (Stop Bit) โดยจะเป็นการทำให้ขา DATA มีสถานะลอจิก “1” อีกครั้งด้วยระยะเวลาอย่างน้อย 1 บิต 1.5 บิต หรือ 2 บิต เพื่อเป็นการแสดงว่าสิ้นสุดข้อมูลแล้ว

อัตราความเร็วในการรับและส่งข้อมูลของการรับส่งข้อมูลแบบอะซิงโครนัสหรืออัตราบอดหรือบอดเรตที่ใช้สำหรับพอร์ตอนุกรม RS-232 มีด้วยกันหลายค่า ได้แก่ 110, 150, 300, 600,

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1200, 2400, 4800, 9600 และ 19200 บิตต่อวินาที โดยมีค่ามากขึ้นตามเทคโนโลยีของคอมพิวเตอร์ เนื่องจากบอดเรต คือ ค่าของจำนวนบิตที่สามารถส่งได้ใน 1 วินาที สมมติว่า ข้อมูลอนุกรมมีขนาด 8 บิต ไม่มีการตรวจสอบพาริตี้ มีบิตเริ่มต้น 1 บิต และบิตปิดท้าย 1 บิต ความยาวของข้อมูล 1 ไบต์ จะมีความยาวเท่ากับ 10 บิต ถ้าใช้บอดเรตในการส่งข้อมูลเท่ากับ 9,600 บิตต่อวินาที ก็จะสามารถส่งข้อมูลได้ด้วยความเร็ว 960 ไบต์ต่อวินาที



รูปที่ 2.3 รูปแบบของข้อมูลแบบอะซิงโครนัส

การตรวจสอบพาริตี้สามารถกำหนดเป็นแบบคี่ (Odd), แบบคู่ (Even) หรือไม่มีการตรวจสอบพาริตี้ก็ได้ พาริตี้คี่หรือพาริตี้คู่แสดงถึงจำนวนลอจิก “1” ทั้งหมดภายในข้อมูลที่ส่งไป 1 ไบต์รวมพาริตี้ว่ามีจำนวนเป็นเลขคู่หรือเลขคี่ ยกตัวอย่างข้อมูลที่จะทำการส่งมีขนาด 8 บิต มีค่าเท่ากับ 99H หรือ 10011001B จะเห็นว่าข้อมูลในไบต์มีจำนวนลอจิก “1’’ จำนวน 4 ตัวซึ่งเป็นเลขคู่ ดังนั้นถ้ากำหนดค่าพาริตี้เป็นคู่ ค่าของพาริตี้บิตจะต้องมีลอจิกเป็น “0” แต่ถ้ากำหนดพาริตี้เป็นคี่ ค่าของบิตพาริตี้จะต้องเป็น “1” เพื่อให้ข้อมูล 1 ไบต์รวมทั้งบิตพาริตี้เป็นคี่

บิตพาริตี้ถูกสร้างขึ้นจากภาคส่งข้อมูลของ UART (Universal Asynchronous Receiver Transmitter) ซึ่งทางภาครับจะต้องกำหนดคุณสมบัติการตรวจสอบพาริตี้ที่ตรงกันเอาไว้ว่าจะตรวจสอบพาริตี้คี่หรือพาริตี้คู่ โดยการนับจำนวนลอจิก “1” ทั้งหมดรวมทั้งบิตพาริตี้ด้วย ถ้ากำหนดพาริตี้ไว้เป็นคู่แต่อ่านค่าตัวเลขในการนับออกมาได้ตัวเลขเป็นคี่ ทางภาครับจะแสดงข้อผิดพลาดออกมาให้ผู้ใช้ทราบ กระบวนการดังกล่าวเป็นวิธีการตรวจสอบความผิดพลาดที่เกิดขึ้นในการรับส่งข้อมูลที่ง่ายที่สุด แต่มันสามารถตรวจสอบได้เมื่อมีบิตข้อมูลที่ทำการรับส่งผิดพลาดเพียงบิตเดียวเท่านั้น ถ้าข้อมูลที่ทำการส่งมีบิตที่ผิดพลาดมากกว่า 1 บิต การตรวจสอบด้วยวิธีนี้จะไม่ได้ผล สำหรับการตั้งพาริตี้บิตเป็น NONE นั้นทั้งภาครับและภาคส่ง จะไม่มีการตรวจสอบพาริตี้

คอมพิวเตอร์ในรุ่น AT เกือบทั้งหมดจะใช้ไอซี UART เบอร์ 16450 และ 16550 ส่วนคอมพิวเตอร์ในรุ่น XT ใช้ไอซี UART เบอร์ 8250 ไอซี UART เหล่านี้มีระดับแรงดันของลอจิกเป็นแบบทีทีแอล (+5 โวลต์) แต่เพื่อให้แรงดันเป็นไปตามมาตรฐาน RS-232 และเพื่อให้การรับส่งข้อมูล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สามารถทำได้ที่ระยะไกลมากยิ่งขึ้น ระดับแรงดันทีทีแอลจะถูกแปลงเป็นระดับแรงดันที่สูงขึ้น โดยลอจิก “0” จะมีระดับแรงดัน –3โวลต์ ถึง –12โวลต์ และลอจิก “1” มีระดับแรงดัน +3โวลต์ ถึง +12 โวลต์

2.4 พอร์ตอนุกรม RS-232

มาตรฐานการเชื่อมต่อแบบอนุกรม RS-232 เป็นมาตรฐานอุตสาหกรรมที่ออกแบบมาเพื่อใช้ในการส่งข้อมูลอนุกรมแบบอะซิงโครนัส 2 ทิศทาง โดยมาตรฐาน RS-232 ในอดีตนั้นออกแบบมาเพื่อการส่งผ่านข้อมูลจากคอมพิวเตอร์ไปยังโมเด็มเพียงอย่างเดียว เพื่อที่จะนำข้อมูลจากโมเด็มนี้ส่งผ่านสายโทรศัพท์ไปยังคอมพิวเตอร์อีกชุดหนึ่งซึ่งอยู่ห่างไกล โดยสมาคมอุตสาหกรรมอิเล็กทรอนิกส์ (Electronic Industries Association : EIA) ได้วางมาตรฐานที่มีชื่อเรียกกันว่า EIA RS232 มาตรฐานนี้ในช่วงแรกจะใช้คอนเน็กเตอร์เป็นแบบ DB-25 โดยกำหนดความยาวสูงสุดของสายสัญญาณไว้ที่ 50 ฟุต มีระดับสัญญาณตั้งแต่ –3โวลต์ ถึง –12โวลต์ แสดงว่ามีข้อมูล (mark) และ +3โวลต์ ถึง +12โวลต์ แสดงว่าเป็นช่องว่าง (space)

มาตรฐาน RS-232 ถูกใช้ในการกำหนดรูปแบบการสื่อสารข้อมูลกันระหว่างอุปกรณ์เชื่อมต่อข้อมูล (Data Terminal Equipment : DTE) กับวงจรข้อมูลปลายทาง (Data Circuit Terminating DCE) อุปกรณ์ DTE จะต้องเป็นอุปกรณ์ที่มีการประมวลผลในตัวเช่น ไมโครคอนโทรลเลอร์หรือไมโครคอมพิวเตอร์ซึ่งมีความสามารถในการสร้างบิตข้อมูลแบบอนุกรมได้ ส่วนอุปกรณ์ DCE ทำหน้าที่เป็นเพียงตัวรับข้อมูลที่ส่งมาจาก DTE เท่านั้น

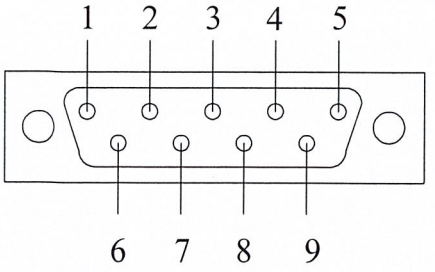
ข้อแตกต่างของอุปกรณ์ DTE และอุปกรณ์ DCE อย่างหนึ่งที่เห็นได้ชัด คือ คอนเน็กเตอร์ของ DTE จะเป็นตัวผู้ ส่วนคอนเน็กเตอร์ของ DCE จะเป็นตัวเมีย ซึ่งพอร์ตอนุกรมของคอมพิวเตอร์ที่ใช้กันอยู่ทั่วไปจะเป็นแบบ DTE ส่วนคอนเน็กเตอร์ที่อยู่ที่โมเด็มจะเป็นแบบ DCE

สำหรับการใช้งานคอมพิวเตอร์ พอร์ตอนุกรม RS-232 ถูกใช้เพื่อเชื่อมต่อกับโมเด็ม, เมาส์ และเครื่องพิมพ์ที่สามารถติดต่อทางพอร์ตอนุกรมได้

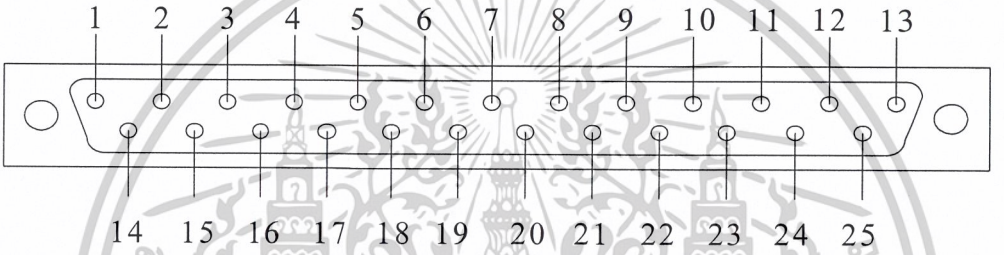
2.4.1 คอนเน็กเตอร์สำหรับพอร์ตอนุกรม RS-232

มาตรฐานการเชื่อมต่อแบบ RS232 จะใช้คอนเน็กเตอร์แบบ DB-25 ตัวผู้ หรือ DB-9 ตัวผู้ ซึ่งคอนเน็กเตอร์แบบ DB-25 จะมีขาต่อใช้งานเพียง 9 เส้น เช่นเดียวกับคอนเน็กเตอร์แบบ DB-9 เนื่องจากขาอื่นๆ ที่เคยมีใช้งานมาในอดีตไม่ค่อยมีความสำคัญมากนักจึงถูกยกเลิกไป โดยแสดงรูปร่างและตำแหน่งขาในรูปที่ 2.4

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



(ก) คอนเน็กเตอร์อนุกรม 9 ขาหรือแบบ DB-9 (มองจากด้านหลังคอมพิวเตอร์)



(ข) คอนเน็กเตอร์อนุกรม 25 ขาหรือแบบ DB-25 (มองจากด้านหลังคอมพิวเตอร์)

รูปที่ 2.4 คอนเน็กเตอร์อนุกรม

2.4.2 การเชื่อมต่อคอนเน็กเตอร์สำหรับพอร์ตอนุกรม RS-232

สำหรับการเชื่อมต่อสายระหว่างคอมพิวเตอร์กับอุปกรณ์ภายนอกแสดงดังในรูปที่ 2.5 ลูกศรในรูป แสดงถึงทิศทางของข้อมูล การเชื่อมต่อในรูปที่ 2.5 (ก) เป็นการเชื่อมต่อแบบ NULL MODEM หรือการเชื่อมต่อโดยตรงโดยไม่ต้องผ่านโมเด็ม ส่วนการเชื่อมต่อในรูปที่ 2.5 (ข) เป็นการเชื่อมต่อโดยใช้สัญญาณน้อยที่สุดเพียง 3 เส้น โดยเส้นหนึ่งสำหรับส่งข้อมูล อีกเส้นสำหรับรับข้อมูลและเส้นสุดท้ายเป็นกราวด์

ขา Data Carrier Detect : DCD หรืออาจเรียกว่า Carrier Detect : CD ขานี้จะแอกตีฟเมื่อมีการส่งสัญญาณพาห์จากอุปกรณ์สื่อสารข้อมูล เช่น โมเด็ม สำหรับใช้งานปกติ ขานี้จะไม่ถูกนำมาใช้งานมากนัก

ขา Recceive Data : RD หรือ RXD ขานี้ใช้เพื่อรับสัญญาณอนุกรมเข้ามายังคอมพิวเตอร์ โดยจะนำข้อมูลที่อ่านได้ไปเก็บไว้ในรีจิสเตอร์บัฟเฟอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 2.2 การจัดขาสัญญาณของพอร์ตอนุกรมในแบบต่างๆ และหน้าที่การทำงาน

คอนเน็คเตอร์ DB-9	คอนเน็คเตอร์ DB-25	ชื่อของสายสัญญาณ	ชนิดของสายสัญญาณ
1	8	Data Carrier Detect : DCD	อินพุต
2	3	Received Data : RXD	อินพุต
3	2	Transmitted Data : TXD	เอาต์พุต
4	20	Data Terminal Ready : DTR	เอาต์พุต
5	7	Single Ground : GND	-
6	6	Data Set Ready : DSR	อินพุต
7	4	Request To Send : RTS	เอาต์พุต
8	5	Clear To Send : CTS	อินพุต
9	22	Ring Indicator : RI	อินพุต

ขา Transmitted Data : TD หรือ TXD ขานี้ใช้เพื่อส่งข้อมูลอนุกรมออกจากคอมพิวเตอร์ โดยการนำข้อมูลที่เก็บอยู่ในบัฟเฟอร์สำหรับส่งข้อมูลส่งออกไป

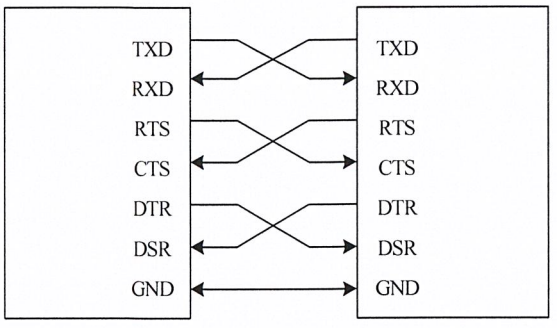
ขา Data Terminal Ready : DTR เป็นขาเอาต์พุตที่ใช้สำหรับส่งสัญญาณออกจากคอมพิวเตอร์เพื่อให้อุปกรณ์ปลายทางรับรู้ว่าต้องการติดต่อกับอุปกรณ์ปลายทาง โดยขา DTR นี้จะต้องเชื่อมต่อกับขา DSR ของอุปกรณ์ปลายทาง และขา DTR ของอุปกรณ์ปลายทางจะต้องเชื่อมต่อกับขา DSR ของคอมพิวเตอร์และถ้าใช้การเชื่อมต่อแบบ 3 สาย ต้องเชื่อมต่อขา DTR และ DSR ของพอร์ตอนุกรมเข้าด้วยกัน และจะต้องต่อเชื่อมเข้ากับขา DCD ด้วยในกรณีที่โปรแกรมสื่อสารที่ใช้มีการตรวจจับสัญญาณพาห์

ขา Signal Ground : GND เป็นขากราวด์ของสัญญาณ

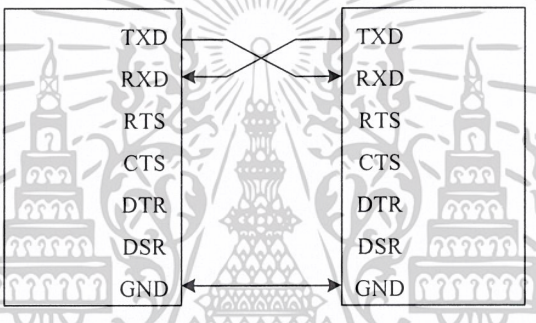
ขา Data Set Ready : DSR ขานี้จะใช้ควบคู่กับขา DTR เพื่อตรวจสอบการเชื่อมต่อกันระหว่างคอมพิวเตอร์กับอุปกรณ์ปลายทาง ซึ่งขา DSR นี้จะเป็นขาสำหรับรับข้อมูลจากภายนอก

ขา Request To Send : RTS เป็นขาเอาต์พุตสำหรับส่งสัญญาณร้องขอให้อุปกรณ์ปลายทางส่งข้อมูลมาให้คอมพิวเตอร์โดยขาที่รับสัญญาณ RTS ก็คือขา CTS ซึ่งในกรณีที่มีการเชื่อมต่อแบบ 3 สาย จะต้องเชื่อมต่อขา RTS และ CTS เข้าด้วยกัน เพื่อให้การรับและส่งข้อมูลสามารถเกิดขึ้นได้ตลอดเวลา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



(ก) การต่ออุปกรณ์ภายนอกเข้ากับคอมพิวเตอร์แบบ Null Modem



(ข) การต่ออุปกรณ์ภายนอกเข้ากับคอมพิวเตอร์แบบ RS-232
ในลักษณะที่ใช้สายสัญญาณน้อยที่สุดเพียง 3 เส้น

รูปที่ 2.5 การต่ออุปกรณ์ภายนอกเข้ากับคอมพิวเตอร์ผ่านทางพอร์ตอนุกรมในรูปแบบต่างๆ

ขา Clear To Send : CTS เป็นขาอินพุตทำหน้าที่รอรับสัญญาณที่ส่งเข้ามา เมื่อมีการส่งสัญญาณเข้ามาที่ขานี้ ข้อมูลที่ขา TXD จะถูกส่งออกไป ขานี้จะใช้เพื่อตรวจสอบอุปกรณ์ต่อพ่วงว่าพร้อมที่จะรับข้อมูลแล้วหรือยัง

ขา Ring Indicator : RI ใช้แสดงสถานะสัญญาณเรียกจากสายโทรศัพท์ ปกติในการสื่อสารโดยทั่วไปสายนี้จะไม่ถูกใช้งาน จะใช้งานก็ต่อเมื่อมีการเชื่อมต่อกับโมเด็มแล้วยังมีความต้องการตรวจสอบสัญญาณเรียกจากสายโทรศัพท์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.4.3 รีจีสเตอร์ของพอร์ตอนุกรม RS-232

UART (Universal Asynchronous Reseiver Transmitter) เป็นอุปกรณ์ที่ทำหน้าที่รับและส่งข้อมูลแบบอะซิงโครนัส สำหรับการสื่อสารแบบอนุกรมบนคอมพิวเตอร์ถือว่า UART เป็นหัวใจสำคัญของการสื่อสารแบบอนุกรม

หน้าที่หลักของ UART คือ ทำหน้าที่แปลงข้อมูลที่อยู่ในรูปแบบขนานจากคอมพิวเตอร์ให้อยู่ในรูปแบบอนุกรมแบบอะซิงโครนัส แล้วส่งออกไปและทำหน้าที่แปลงสัญญาณอนุกรมแบบอะซิงโครนัสที่ป้อนเข้ามา UART ให้เป็นขนานก่อนที่จะส่งเข้าสู่คอมพิวเตอร์ ซึ่งนอกจาก UART จะส่งข้อมูลไปยังคอมพิวเตอร์แล้ว ยังแจกแจงข้อมูลอื่นๆ ให้คอมพิวเตอร์รับทราบด้วย เช่น อัตราเร็วในการรับส่งข้อมูล (บอดเรต), รูปแบบการส่งข้อมูล, ความผิดพลาดที่เกิดขึ้นระหว่างการถ่ายทอดข้อมูล (ผิดพลาดจากพารีตี้, เฟรมข้อมูล, โอเวอร์รัน) เป็นต้น

ภายใน UART จะมีส่วนของวงจรสร้างบอดเรตแบบโปรแกรมได้ (programmable buadrate generator) โดยการกำหนดค่าตัวหารให้กับสัญญาณนาฬิกาของ UART โดยตัวหารนี้มีขนาด 16 บิต ดังนั้นจึงสามารถกำหนดตัวหารอยู่ในช่วง 1 – 65,535 ซึ่ง UART สามารถรับส่งข้อมูลได้ทั้งแบบฮาล์ฟดูเพล็กซ์ (half duplex) และฟูลดูเพล็กซ์ (full duplex) โดยการส่งแบบฮาล์ฟดูเพล็กซ์เป็นการส่งข้อมูลแบบทิศทางเดียว ส่วนการส่งข้อมูลแบบฟูลดูเพล็กซ์นั้นสามารถรับและส่งข้อมูลได้ในคราวเดียวกัน

ในเครื่องคอมพิวเตอร์โดยทั่วไปมี UART ที่ใช้งานกันอยู่ 2 เบอร์ คือ 8250 ซึ่งเป็น UART มาตรฐานที่มีใช้กันมายาวนาน UART เบอร์นี้จะมีบัพเฟอร์สำหรับรับและส่งข้อมูลตำแหน่งเดียวกัน ทำให้การรับและส่งข้อมูลถูกจำกัดความเร็วอยู่ที่ 57.6 กิโลบิตต่อวินาทีเท่านั้น แต่ UART เบอร์นี้ถือว่าเป็นต้นแบบของ UART ที่ใช้กันในคอมพิวเตอร์ คอมพิวเตอร์ทุกๆ รุ่นจะต้องสนับสนุนการทำงานตามรูปแบบของ UART เบอร์นี้

UART อีกเบอร์หนึ่ง คือ 16450 มีความสามารถรับส่งข้อมูลได้ที่ความเร็ว 115200 บิตต่อวินาที และเพิ่มรีจีสเตอร์สำหรับพักข้อมูลสำหรับ UART นอกจากนั้นยังเพิ่มส่วนของชิฟต์รีจีสเตอร์แบบ FIFO (First In First Out) ขนาด 16 ไบต์เข้าไป ทำให้สามารถสนับสนุนความเร็วในการรับส่งข้อมูลที่ 256 กิโลบิตต่อวินาทีได้ โดยคอมพิวเตอร์ในปัจจุบันใช้ UART เบอร์นี้หรือใหม่กว่า เช่น เบอร์ TL164750 ซึ่งมีรีจีสเตอร์แบบ FIFO ขนาด 64 ไบต์ ทำงานได้ที่ระดับแรงดัน +5 โวลต์ และ +3 โวลต์ มีโหมดประหยัดพลังงาน สามารถรับส่งข้อมูลได้ที่ความเร็ว 1 เมกะบิตต่อวินาที เมื่อใช้สัญญาณนาฬิกา 16 เมกะเฮิร์ต

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ความเร็ว ในการส่งข้อมูลที่มากมายของ UART เบอร์ใหม่ๆ ก็ไม่ได้ช่วยการรับส่งข้อมูลของคอมพิวเตอร์เร็วขึ้น เนื่องจากว่าคอมพิวเตอร์ยังใช้ความถี่ของสัญญาณนาฬิกาในการแปลงข้อมูลเพียง 1.8432 MHz

เครื่องคอมพิวเตอร์โดยทั่วไปสามารถต่อพอร์ตอนุกรมสูงสุดได้ 4 พอร์ต มีชื่อเรียกว่าเป็น COM1, COM2, COM3 และ COM4 ซึ่งพอร์ตอนุกรมแต่ละตัวต่างก็ใช้งาน UART ภายในคอมพิวเตอร์ในการติดต่อกับอุปกรณ์ภายนอกเช่นเดียวกัน ในแผนผังการทำงานภายในพอร์ตอนุกรมประกอบด้วยรีจีสเตอร์ 8 บิต 8 ตัว ที่ใช้งานร่วมกับ UART แอดเดรสของรีจีสเตอร์ภายในพอร์ตอนุกรมสามารถคำนวณได้จากค่ารีจีสเตอร์พื้นฐานของพอร์ตอนุกรม ตัวอย่าง พอร์ตอนุกรม COM1 มีแอดเดรสอยู่ที่ 3F8H ตำแหน่งของรีจีสเตอร์ต่างๆ จะเป็นตำแหน่งที่บวกเข้าไปกับค่า 3F8H โดยมีรีจีสเตอร์ที่ใช้งานพอร์ตอนุกรมมีดังนี้

00H เป็นรีจีสเตอร์บัฟเฟอร์สำหรับเก็บข้อมูลที่รับเข้ามาหรือเตรียมข้อมูลก่อนที่จะส่งออก

01H รีจีสเตอร์เอ็นเอเบิลการอินเตอร์รัพต์ ใช้เซตโหมดการอินเตอร์รัพต์ของพอร์ตอนุกรม

02H รีจีสเตอร์แสดงโหมดการอินเตอร์รัพต์ ใช้เพื่อตรวจสอบโหมดของการอินเตอร์รัพต์

03H รีจีสเตอร์สำหรับกำหนดรูปแบบของข้อมูล

04H รีจีสเตอร์ควบคุมโมเด็ม ใช้ตรวจสอบบิตสำหรับติดต่อกับโมเด็ม เช่น RTS หรือ DTR

05H รีจีสเตอร์แสดงสถานะการรับและการส่งข้อมูลแบบอนุกรม

06H รีจีสเตอร์แสดงสถานะของโมเด็ม ซึ่งจะแสดงสถานะของขา DCD, RI, DSR และ CTS

06H รีจีสเตอร์สำหรับการเก็บข้อมูลชั่วคราว

2.5 ฐานข้อมูล

ฐานข้อมูล (Database) คือ วิธีการจัดเก็บข้อมูลที่สัมพันธ์กันอย่างมีระเบียบ ซึ่งจะทำให้ง่ายต่อการใช้งานและค้นหาข้อมูล ซึ่งฐานข้อมูลที่คนส่วนใหญ่คุ้นเคย คือ ฐานข้อมูลเชิงสัมพันธ์ เป็นรูปแบบการจัดการเก็บข้อมูลในฐานข้อมูลที่สัมพันธ์กัน โดยมองข้อมูลในลักษณะของตารางต่างๆ ที่มีความสัมพันธ์กัน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.5.1 องค์ประกอบของฐานข้อมูล

1) แอพพลิเคชั่นฐานข้อมูล

แอพพลิเคชั่นฐานข้อมูล เป็นแอพพลิเคชั่นที่สร้างไว้ให้ผู้ใช้งานสามารถติดต่อกับฐานข้อมูลได้อย่างสะดวกซึ่งมีรูปแบบการติดต่อกับฐานข้อมูลแบบเมนูหรือกราฟฟิก โดยไม่จำเป็นต้องมีความรู้เกี่ยวกับฐานข้อมูลเลยก็สามารถเรียกใช้งานฐานข้อมูลได้

2) ระบบการจัดการฐานข้อมูล

ระบบการจัดการฐานข้อมูล เป็นซอฟต์แวร์ที่ทำหน้าที่จัดการฐานข้อมูลในฐานข้อมูล ทั้งการจัดเก็บ การแสดงผล การค้นหา การสำรองข้อมูล ฯลฯ โดยเป็นเครื่องมือในการทำงานของผู้บริหารฐานข้อมูล และเป็นตัวกลางที่เชื่อมผ่านระหว่างแอพพลิเคชั่นฐานข้อมูลที่สร้างขึ้นกับตัวข้อมูลในฐานข้อมูล ตัวอย่าง DBMS เช่น Microsoft Access, FoxPro, SQLServer, Oracle, Informix, DB2 เป็นต้น

3) ดาต้าเบสเซิร์ฟเวอร์

ดาต้าเบสเซิร์ฟ เป็นคอมพิวเตอร์ที่คอยให้บริการในการจัดการฐานข้อมูล ซึ่งก็คือคอมพิวเตอร์ที่ระบบจัดการฐานข้อมูลทำงานอยู่นั้นเอง เพราะฉะนั้นจึงมักจะเป็นคอมพิวเตอร์ที่มีประสิทธิภาพสูงกว่าคอมพิวเตอร์ที่ใช้งานทั่วไป

4) ข้อมูล

ข้อมูล คือ ตัวเนื้อของข้อมูลที่เราใช้งาน ซึ่งจะถูกเก็บในหน่วยความจำของดาต้าเบสเซิร์ฟเวอร์ โดยจะถูกเรียกมาใช้งานจากระบบจัดการฐานข้อมูล

5) ผู้บริหารฐานข้อมูล

ผู้บริหารฐานข้อมูล เป็นคนที่ทำหน้าที่ดูแลข้อมูลในฐานข้อมูลผ่านระบบจัดการฐานข้อมูล ซึ่งจะควบคุมให้การทำงานเป็นไปอย่างราบรื่น นอกจากนี้ยังทำหน้าที่กำหนดผู้ที่จะมีสิทธิ์ใช้งานฐานข้อมูล กำหนดในเรื่องความปลอดภัยของการใช้งาน พร้อมทั้งดูแลดาต้าเบสเซิร์ฟเวอร์ให้ทำงานอย่างเป็นปกติด้วย

2.5.2 ประโยชน์จากการประมวลผลด้วยฐานข้อมูล

1) ลดความซ้ำซ้อนของข้อมูล

2) สามารถหลีกเลี่ยงความขัดแย้งของข้อมูลได้ในระดับหนึ่ง

3) สามารถใช้ข้อมูลร่วมกันได้

4) สามารถควบคุมความเป็นมาตรฐานได้

5) สามารถจัดหาระบบความปลอดภัยที่รัดกุมได้

6) สามารถควบคุมความคงสภาพของข้อมูลได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

7) สามารถสร้างสมดุลในความขัดแย้งของความต้องการได้

8) เกิดความเป็นอิสระของข้อมูล

2.6 พื้นฐานการสื่อสารผ่านทางพอร์ตสื่อสารแบบอนุกรมโดย Visual Basic

ในการพัฒนาแอปพลิเคชั่นที่สามารถทำการสื่อสารและแลกเปลี่ยนข้อมูล ผ่านทางพอร์ตสื่อสารแบบอนุกรม (Serial Port) ได้นั้นจำเป็นที่จะต้องมีความรู้พื้นฐานในเรื่องการสื่อสารข้อมูล แต่ในการพัฒนาแอพพลิเคชัน เพื่อใช้ในการสื่อสารผ่านทางพอร์ตสื่อสารอนุกรมโดยใช้ Visual Basic นั้น ผู้พัฒนาสามารถใช้คอนโทรลสื่อสารที่ Visual Basic ได้เตรียมไว้ให้เพื่อใช้ในการพัฒนาแอพพลิเคชันสื่อสารข้อมูล ซึ่งคอนโทรลที่จะสามารถใช้ในการพัฒนา คือ คอนโทรล MSComm เป็นคอนโทรล ที่มีความสามารถในการติดต่อสื่อสารข้อมูลผ่านทางพอร์ตอนุกรมได้เป็นอย่างดีมีฟังก์ชันที่ใช้ในการติดต่อสื่อสารกับพอร์ตอนุกรมอย่างครบถ้วน สามารถใช้ในการควบคุมการสื่อสาร การดักจับเหตุการณ์ที่เกิดขึ้นในระหว่างการสื่อสาร ผ่านทางพอร์ตอนุกรมเพื่อพัฒนาแอพพลิเคชันที่จะใช้ในการควบคุมเหตุการณ์นั้นๆ

พื้นฐานในการสื่อสารโดยผ่านทางพอร์ตอนุกรม คือ เครื่องคอมพิวเตอร์จะมีพอร์ตอนุกรมที่ใช้ในการเชื่อมต่อกับอุปกรณ์ภายนอก และหัวใจสำคัญในการสื่อสารข้อมูลโดยใช้พอร์ตอนุกรมก็คือ กระบวนการในการเปลี่ยนแปลงข้อมูลหรือการจัดเรียงข้อมูล เมื่อมีการแลกเปลี่ยนข้อมูลระหว่างข้อมูลที่รับหรือส่งผ่านทางพอร์ตอนุกรมกับซีพียู กล่าวคือ ในการส่งข้อมูลจากซีพียูมายังพอร์ตอนุกรมนี้ จะทำการส่งมาในรูปแบบของข้อมูลที่มีหน่วยเป็นไบต์ แต่เมื่อมาถึงพอร์ตอนุกรมแล้วจะต้องมีกระบวนการในการแปลงข้อมูลนั้นให้อยู่ในรูปของบิต ซึ่งในการส่งข้อมูลออกหรือว่าการรับข้อมูลเข้าของพอร์ตอนุกรมนั้น จะทำการส่งข้อมูลในรูปแบบทีละบิตเป็นลำดับเรียงกันไป ในการรับข้อมูลก็เช่นเดียวกัน จะต้องมีกระบวนการในการแปลงลำดับของบิตที่รับเข้ามาผ่านทางพอร์ตอนุกรมให้เป็นไบต์เพื่อส่งไปให้กับซีพียูต่อไป

ในการพัฒนาแอพพลิเคชันที่ใช้ในการสื่อสารผ่านพอร์ตอนุกรมโดยใช้คอนโทรล MSComm นี้ จะมีความสะดวกเป็นอย่างมาก เพราะว่าในขั้นตอนการพัฒนาแอพพลิเคชัน ผู้พัฒนาไม่จำเป็นที่จะต้องสนใจหรือต้องทำการพัฒนาส่วนการแปลงข้อมูลหรือการจัดเรียงข้อมูลเพื่อใช้ในการติดต่อสื่อสารผ่านพอร์ตอนุกรม เพราะว่าการทำงานในส่วนนี้จะจัดการโดยคอนโทรล MSComm ทั้งหมด สิ่งที่ผู้พัฒนาจะต้องสนใจและทำการพัฒนา คือ จะทำการติดต่อกับคอนโทรล MSComm นี้อย่างไร คอนโทรลนี้มีฟังก์ชันอะไรให้ใช้งานได้บ้าง

ในการตั้งค่าพื้นฐานเพื่อใช้ในการเปิดช่องทางการติดต่อกับพอร์ตอนุกรมของคอนโทรล MSComm จะต้องมีการตั้งค่าพื้นฐาน 3 ส่วนด้วยกัน คือ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สำนักหอสมุดกลาง พระจอมเกล้าลาดกระบัง

Properties

Commport เป็นการตั้งค่าเพื่อเลือกหมายเลขของพอร์ตอนุกรมที่จะใช้งาน โดยทั่วไปแล้วจะมีค่าเป็น 1 , 2 , 3 และ 4 ซึ่งหมายถึง com1 , com2 , com 3 และ com4 ตามลำดับ

Settings เป็นการตั้งค่าเพื่อกำหนดค่าพื้นฐานต่างๆที่จะใช้เป็นมาตรฐานในการสื่อสารข้อมูลผ่านทางพอร์ตอนุกรม เช่น ค่าความเร็วของช่องทางเดินของข้อมูลการเซ็ตบิตเพื่อใช้เช็คความถูกต้องของข้อมูลจำนวนชุดของบิตที่จะส่ง การเลือกใช้บิตหยุด เป็นต้น โดยจะทำการส่งค่าเป็นแบบข้อความ เช่น “9600, N, 8, 1” เป็นต้น

PortOpen เป็นการตั้งค่าในการเปิดหรือการปิดพอร์ตอนุกรม ที่จะใช้ในการสื่อสารข้อมูลค่าต่าง ๆ ที่ใช้ในการตั้งค่าให้กับคุณสมบัติ Settings จะมีอยู่ด้วยกัน 4 ส่วน ได้แก่

1) Baud Rate

2) Parity Values

3) Valid Data Bit Values

4) Stop Bit Values

ซึ่งสามารถตั้งค่าต่าง ๆ ที่เป็นไปได้ของคุณสมบัติแต่ละตัวดังนี้ คือ

1) Baud Rate จะมีค่าได้ดังนี้ คือ 110, 300, 600, 1200, 2400, 9600 (เป็นค่ามาตรฐาน), 14400, 28800, 38400, 56000, 128000, 256000

2) Parity Values จะมีค่าได้ดังนี้ คือ

E	Even
M	Mark
N	None (เป็นค่ามาตรฐาน)
O	Odd
S	Space

3) Valid Data Bit Values จะมีค่าได้ดังนี้ คือ 4, 5, 6, 7, 8 (เป็นค่ามาตรฐาน)

4) Stop Bit Values จะมีค่าได้ดังนี้ คือ 1 (เป็นค่ามาตรฐาน), 1.5, 2

ถ้าต้องการเปิดพอร์ตอนุกรมเพื่อใช้ในการสื่อสารข้อมูลผ่านทางพอร์ต Com 2 และต้องการให้มี Baud Rate เป็น 9600 ไม่มีการเช็คค่า Parity Bit มี Valid Data Bit เป็น 8 และ Stop Bit เป็น 1 จะต้องทำการตั้งคุณสมบัติต่างๆ ของคอนโทรล MSComm ดังนี้ คือ

```
“Open the Serial Port (Com 2)
MSComm1.CommPort = 2
MSComm1.Setting = “9600, 8, 1”
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้
51870

```
MSComm1.PortOpen = True
```

เมื่อทำการเปิดพอร์ตอนุกรมเรียบร้อยแล้วก็จะสามารถทำการส่งหรือรับข้อมูลผ่านทางพอร์ตอนุกรมที่เปิดได้ โดยการใช้เมธอดในการรับส่งข้อมูลที่คอนโทรล MSComm ได้จัดเตรียมไว้

2.6.1 การรับและส่งข้อมูลผ่านพอร์ตอนุกรมโดยคอนโทรล MSComm

ในการรับหรือส่งข้อมูลผ่านทางพอร์ตอนุกรมนี้สามารถที่จะทำได้ก็ต่อเมื่อได้มีการเปิดการเชื่อมต่อกับพอร์ตอนุกรมเป็นที่เรียบร้อยแล้วเท่านั้น โดยในการรับหรือส่งข้อมูลโดยการใช้คอนโทรล MSComm นี้จะรับส่งผ่านบัฟเฟอร์ ซึ่งคอนโทรลสื่อสารนี้ได้เตรียมไว้ให้ โดยจะแยกการรับและการส่งข้อมูลผ่านทางบัฟเฟอร์ของแต่ละส่วน โดยจะประกอบด้วย

1) บัฟเฟอร์ที่ใช้ในการส่งข้อมูล (The Transmit Buffer)

2) บัฟเฟอร์ที่ใช้ในการรับข้อมูล (The Receive Buffer)

ในการรับหรือส่งข้อมูลผ่านทางบัฟเฟอร์นี้จะสามารถรับหรือส่งได้ทั้งข้อมูลที่เป็นข้อความหรือข้อมูลที่เป็นไบนารี เช่น รูปภาพหรือไฟล์วีดีโอ เป็นต้น โดยในการรับหรือส่งข้อมูลโดยใช้คอนโทรล MSComm จะมีลักษณะการใช้งานดังนี้

การรับข้อมูลเพื่อใช้แสดงในคอนโทรลเท็กซ์ จะใช้เมธอด Input ดังนี้

```
TextDisplay.Text = MSComm1.Input
```

การส่งข้อมูลออกไปยังพอร์ตอนุกรม จะใช้เมธอด Output ดังนี้

```
MSComm1.Output = “This is a text string”
```

เมื่อทราบการตั้งค่าต่าง ๆ เพื่อใช้ในงานคอนโทรลแล้ว ในการพัฒนาแอพพลิเคชันเพื่อใช้ในการติดต่อกับพอร์ตอนุกรมก็จะสามารถทำได้ง่าย สิ่งที่ผู้ทำการพัฒนาแอพพลิเคชันสื่อสารข้อมูลต้องสนใจ คือ การจัดการรับข้อมูลเข้าผ่านพอร์ตอนุกรมเพื่อจะนำมาใช้ในการประมวลผลหรือส่งข้อมูลที่เป็นผลลัพธ์จากการประมวลผลผ่านทางพอร์ตอนุกรมเพื่อนำไปใช้งานต่อไปเท่านั้น

2.6.2 การใช้งานคอนโทรล MSComm

คอนโทรล MSComm เป็นคอนโทรลตัวหนึ่งที่ช่วยติดต่อกับพอร์ตอนุกรม (Serial Port) ซึ่งผู้อ่านทำการรับ-ส่งข้อมูลผ่านทางพอร์ตอนุกรมได้ด้วยคอนโทรลนี้ เช่น การติดต่อโดยตรงกับบอร์ดอิเล็กทรอนิกส์ เป็นต้น ซึ่งคอนโทรล MSComm ที่มากับ Visual Basic จะเป็นคอนโทรลที่ทำงานโดยมีการตอบสนองต่อเหตุการณ์แบบ even-driven นั่นก็คือ คอนโทรลจะทำหน้าที่ตรวจสอบการเกิดขึ้นหรือการร้องขอให้เกิดเหตุการณ์ต่างๆ กับพอร์ตอนุกรมโดยอัตโนมัติ และจะมีการแจ้งเตือนให้ได้ทราบผ่านทางโพรซีเยอร์เหตุการณ์ เช่นเดียวกับคอนโทรลทั่วไปของ Visual Basic

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

คอนโทรล MSComm จะมีหน้าที่มาตรฐานหลักๆ สำหรับการสื่อสารผ่านพอร์ตอนุกรม 3 ประการ ดังต่อไปนี้

1) หมุนหมายเลขติดต่อกับโทรศัพท์ปลายทางที่กำหนด

2) ตรวจสอบการเข้ามาของข้อมูลยังพอร์ตอนุกรมโดยอัตโนมัติ

3) ส่งข้อมูลตามที่กำหนดจากโปรแกรมไปยังพอร์ตอนุกรม

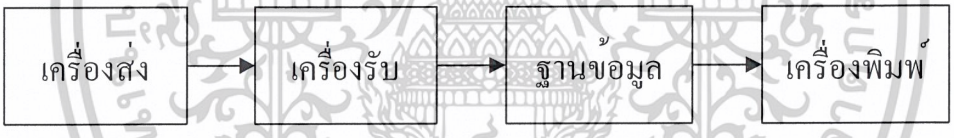
ในความเป็นจริงคอนโทรล MSComm ไม่ได้ทำหน้าที่ติดต่อกับพอร์ตอนุกรมโดยตรง แต่มันจะทำหน้าที่เรียกใช้ฟังก์ชันวินโดวส์ API ซึ่งวินโดวส์จะทำการส่งหรือรับข้อมูลผ่านทางพอร์ตอนุกรมโดยอาศัยไดรเวอร์ Comm.drv จากนั้นก็จะส่งผ่านข้อมูลที่ถูกจัดรูปแบบตามมาตรฐานการสื่อสาร (ทั้งนี้ขึ้นอยู่กับอุปกรณ์ที่ต่อเข้ากับพอร์ตอนุกรม) ให้กับดีไวซ์ไดรเวอร์อีกต่อหนึ่ง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 3
การออกแบบ การสร้าง และการทำงาน

3.1 กล่าวนำ

จากหลักการพื้นฐาน และทฤษฎีต่างๆ ที่เกี่ยวข้องกับไมโครคอนโทรลเลอร์ 8051, ฐานข้อมูล, โปรแกรม Visual Basic และพอร์ตอนุกรม RS-232 สามารถทำการออกแบบและสร้างวงจรการทำงานของเครื่องสั่งอาหารระบบอัตโนมัติ โดยมีขั้นตอนการออกแบบ การสร้าง และการทำงานโดยผังการทำงานของเครื่องสั่งอาหารระบบอัตโนมัติในรูปที่ 3.1 สามารถแบ่งการทำงานออกเป็น 4 ส่วน ซึ่งมีลักษณะการทำงานดังนี้ คือ เครื่องส่งจะส่งรหัสของอาหารและจำนวนของอาหารที่ต้องการไปยังเครื่องรับ เครื่องรับจะตรวจเช็คและยืนยันการส่งข้อมูลที่ได้ไปยังเครื่องส่งและเครื่องรับจะนำข้อมูลที่ได้ส่งไปเก็บไว้ในฐานข้อมูลโดยผ่านพอร์ตอนุกรม RS-232 ภายในฐานข้อมูลโปรแกรมจะประมวลผลข้อมูลที่ได้เป็นใบเสร็จรับเงินส่งข้อมูลที่ประมวลผลได้ไปยังเครื่องพิมพ์ เครื่องพิมพ์จะพิมพ์ใบเสร็จรับเงินมาแสดง



รูปที่ 3.1 ผังงานของเครื่องสั่งอาหารระบบอัตโนมัติ

3.2 เครื่องส่ง

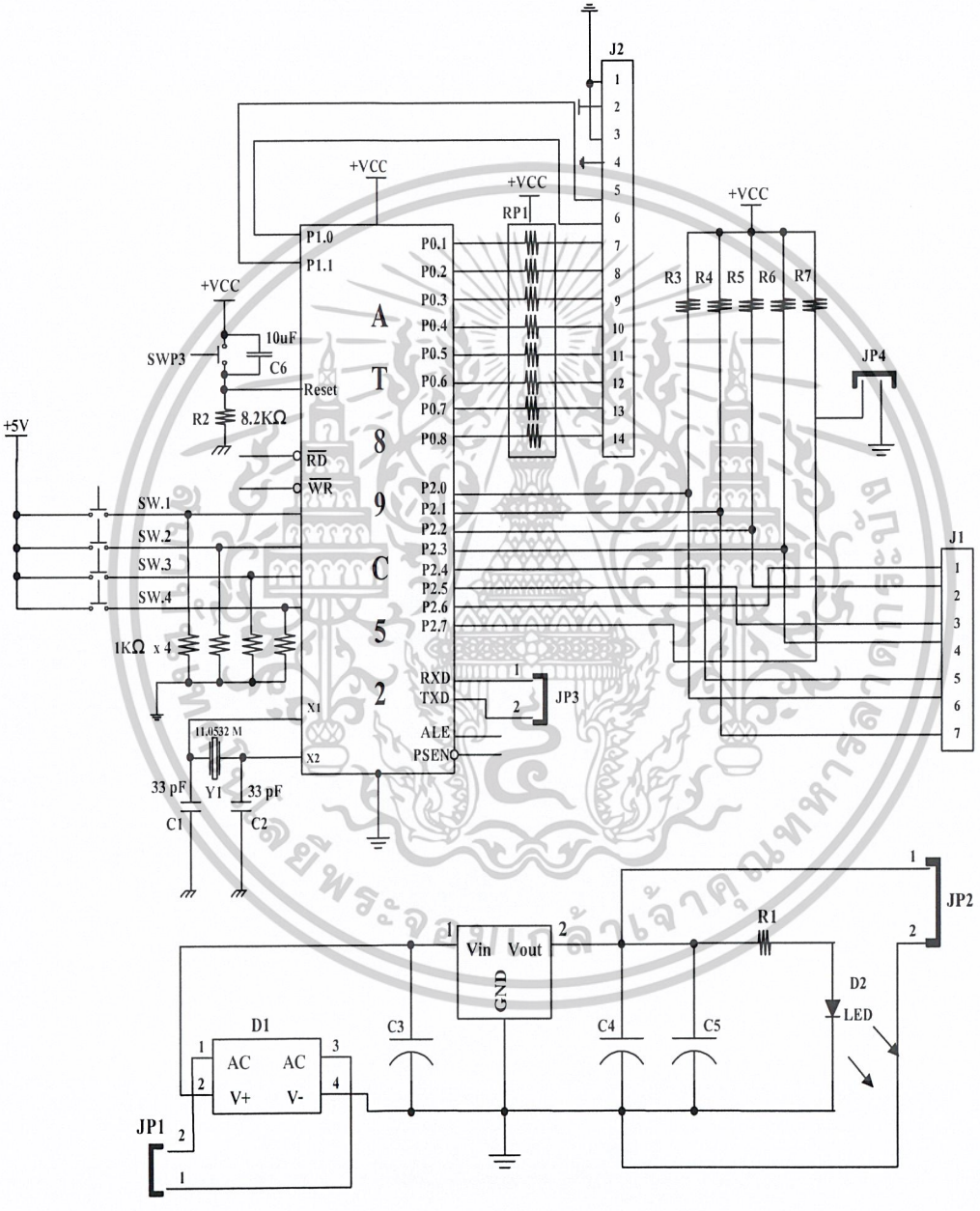
เครื่องส่งจะทำหน้าที่นำรหัสรายการอาหารและจำนวนอาหารที่ป้อนเข้ามาไปเก็บไว้ในหน่วยความจำข้อมูลแล้วนำข้อมูลนั้นส่งไปยังเครื่องรับ ถ้าข้อมูลถูกส่งเรียบร้อยแล้วจะแสดงข้อความ Send Success

3.2.1 การออกแบบวงจรเครื่องส่ง

วงจรเครื่องส่งจะใช้ CPU AT89C52 ทำหน้าที่ควบคุมการแสดงผลทางจอแอลซีดี, การรับข้อมูลจาก Dip Switch, การรับข้อมูลจากคีย์แพด และการส่งข้อมูลแบบอนุกรม ซึ่งไมโครคอนโทรลเลอร์จะมีคริสตอล X1 พร้อมตัวเก็บประจุ C1 และ C2 เป็นตัวกำหนดฐานเวลาโดยพอร์ต P3.4 – P3.7 จะเชื่อมต่อกับ Dip Switch และต่อตัวต้านทานพูลดาวน์ไว้ เพื่อกำหนด

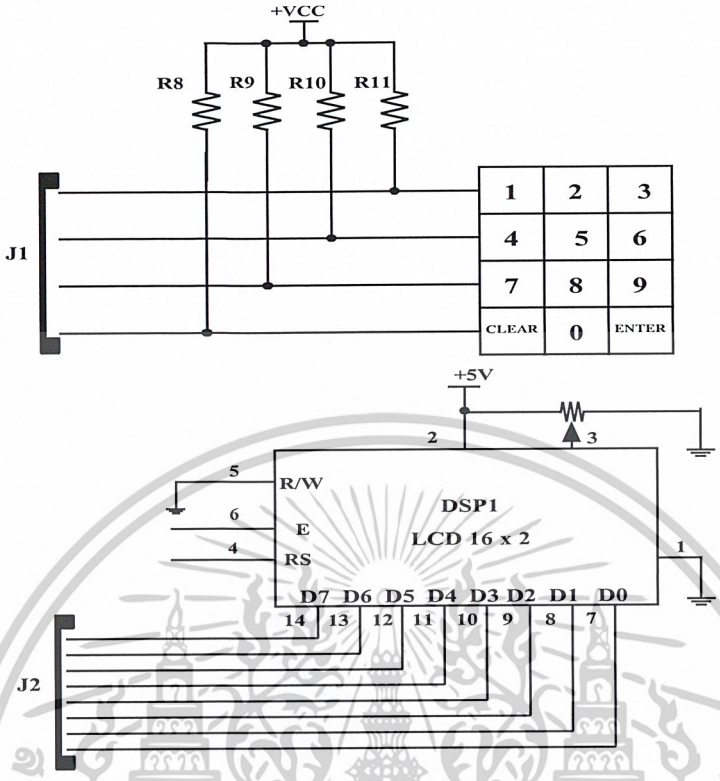
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สภาวะเบื้องต้นที่ไม่มีการเลื่อน Dip Switch หากไม่มีการเลื่อน Dip Switch ค่าของพอร์ต P2.4 – P2.7 จะมีค่าเป็น “0” หากมีการเลื่อน Dip Switch ค่าของพอร์ต P2.4 – P2.7 ก็จะได้ค่าของข้อมูลตามลักษณะการเลื่อน Dip Switch ค่าที่ได้จากการเลื่อน Dip Switch จะใช้เป็นค่าในการกำหนดหมายเลขของเครื่องส่ง



รูปที่ 3.2 วงจรเครื่องส่ง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.3 วงจรส่วนการแสดงผล

1) การออกแบบส่วนการแสดงผล

ในรูปที่ 3.3 การออกแบบในส่วนการแสดงผลซึ่งใช้พอร์ต 3 ของไมโครคอนโทรลเลอร์เชื่อมต่อกับคีย์แพดทั้ง 7 เส้น คือ สายคอลัมน์ 3 เส้น และสายทางโรว์หรือแถวอีก 4 เส้นโดยพอร์ต P2.0 – P2.3 จะต่อตัวต้านทานพูลอัพไว้ เพื่อกำหนดสภาวะเริ่มต้นที่ไม่มีการกดคีย์โดยไมโครคอนโทรลเลอร์จะทำการส่งข้อมูล “0” ไปยัง P2.6, P2.4 และ P2.5 ตามลำดับ ทุกครั้งที่มีการส่งข้อมูลไปยังสายคอลัมน์ของคีย์แพดไมโครคอนโทรลเลอร์จะอ่านค่าที่ P2.0 – P2.3 เข้ามาด้วย หากไม่มีการกดค่าของ P2.0 – P2.3 ก็จะไม่เป็น 1111 อีกต่อไป เป็นการแจ้งให้ทราบว่ามีการกดคีย์แพดขึ้นแล้ว ต่อจากนั้นไมโครคอนโทรลเลอร์ก็จะทำการค้นหาค่าของคีย์นั้นไปเปิดตารางข้อมูลเพื่อจะได้หมายเลขของคีย์ที่กดอย่างแท้จริง จากนั้นไมโครคอนโทรลเลอร์จะนำข้อมูลที่ได้เก็บไว้ในหน่วยความจำข้อมูลภายใน พร้อมกับนำข้อมูลไปแสดงผลที่จอแอลซีดีโมดุลโดยจะต่อขา D0 – D7 เข้ากับ P0.0 – P0.7 เพื่อเป็นทางผ่านของข้อมูลระหว่างแอลซีดีกับไมโครคอนโทรลเลอร์ พอร์ต P1.0 จะต่อเข้ากับขา RS ของแอลซีดีโมดุลเพื่อใช้ในการแยกชนิดของข้อมูลที่ทำประมวลผล ถ้าขานี้เป็น “0” ข้อมูลที่ส่งมาจะเป็นคำสั่ง แต่ถ้าขานี้เป็น “1” ข้อมูลที่ส่งมาจะเป็นข้อมูลสำหรับ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

แสดงผล ส่วน P1.1 จะต่อกับขา E ของแอลซีดีโมดุล เพื่อสร้างพัลส์อินาเบิลให้แอลซีดีโมดุลให้ทำงาน

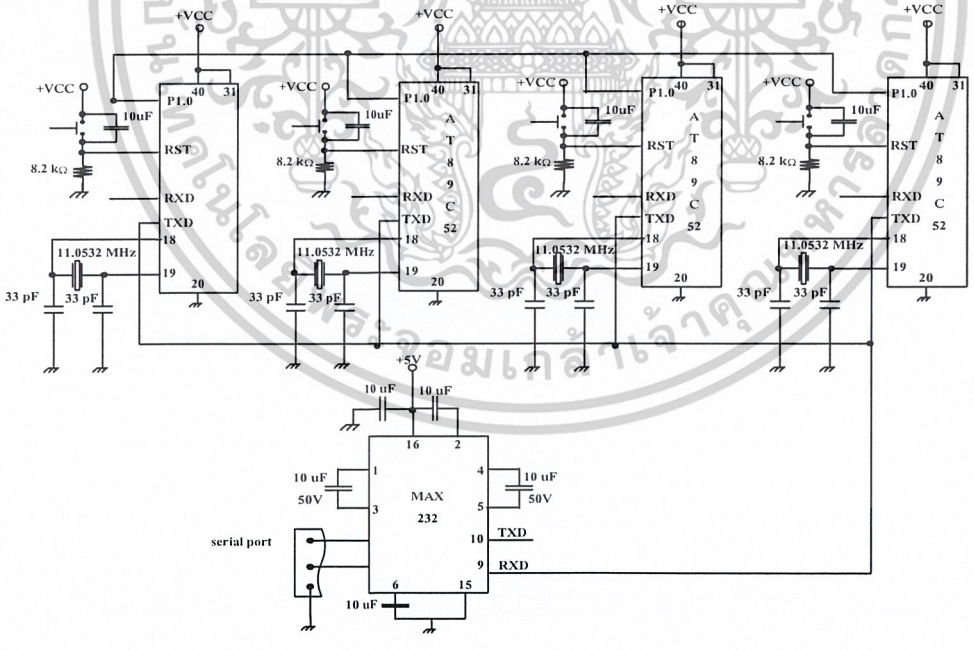
3.3 เครื่องรับ

เครื่องรับทำหน้าที่รับข้อมูลที่เครื่องส่งเข้ามา เพื่อตรวจเช็คข้อมูลที่ส่งเข้ามาว่าถูกต้องหรือไม่ ถ้าข้อมูลที่ส่งมาถูกต้องเครื่องรับจะส่งข้อความ Send Success เพื่อยืนยันความถูกต้องไปยังเครื่องส่ง และจะส่งรหัสรายการอาหารที่ได้รับผ่านพอร์ตอนุกรม RS – 232 ไปเก็บไว้ในฐานข้อมูล

3.3.1 การออกแบบวงจรเครื่องรับ

วงจรเครื่องรับจะใช้ ไมโครคอนโทรลเลอร์ AT89C52 ทั้งหมด 16 ตัว เพื่อทำหน้าที่รับส่งข้อมูลจากเครื่องส่งโดยขา RXD ของไมโครคอนโทรลเลอร์ AT89C52 จะต่อกับขา TXD ไมโครคอนโทรลเลอร์ AT89C52 ของเครื่องส่งและขา TXD ของไมโครคอนโทรลเลอร์ AT89C52 จะต่อกับขา RXD ของไมโครคอนโทรลเลอร์ AT89C52 ทางเครื่องส่งเช่นกัน

IC MAX 232 ทำหน้าที่ปรับระดับสัญญาณให้เหมาะสมกับมาตรฐาน RS – 232 ก่อนที่จะทำการเชื่อมต่อกับคอมพิวเตอร์ โดยขา RXD ของ MAX 232 จะต้องต่อเข้ากับขา TXD ของไมโครคอนโทรลเลอร์ AT89C52 ทั้ง 16 ตัวในเครื่องรับ



รูปที่ 3.4 วงจรเครื่องรับ

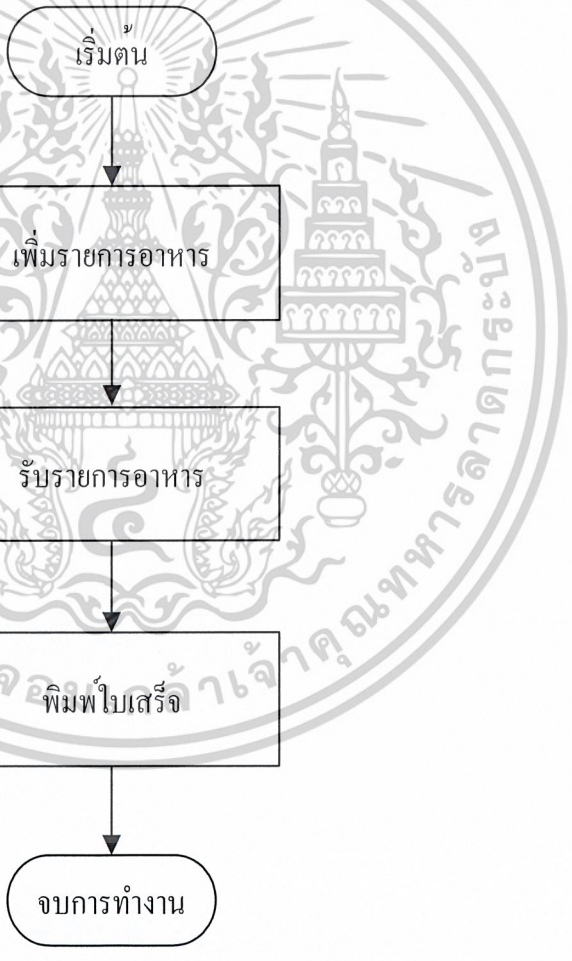
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.4 ฐานข้อมูล

ภายในฐานข้อมูลโปรแกรมจะประมวลผลรหัสรายการอาหารที่รับเข้ามาแสดงเป็นรายการอาหารมาแสดง ซึ่งจะแสดงเป็นรายการอาหารที่สั่งและจำนวนเงินค่าอาหารทั้งหมดที่สั่ง ซึ่งโปรแกรมจะส่งข้อมูลให้แสดงทางเครื่องพิมพ์

3.4.1 การออกแบบฐานข้อมูล

การออกแบบฐานข้อมูลต้องศึกษาโปรแกรมการเรียกใช้ระบบฐานข้อมูล เพื่อจัดเก็บข้อมูลรายการต่างๆ โปรแกรมจะแบ่งเป็น 3 ส่วน คือ ส่วนที่หนึ่งการเพิ่มรายการอาหาร ส่วนที่สองการรับรายการอาหาร และส่วนที่สามการพิมพ์ใบเสร็จรับเงิน ซึ่งทั้ง 3 ส่วนจะมีหลักการทำงานดังรูปที่ 3.5

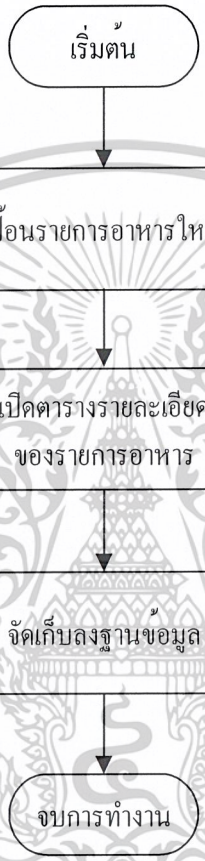


รูปที่ 3.5 ผังการทำงานของโปรแกรมในฐานข้อมูล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.4.2 การออกแบบโปรแกรมเพิ่มรายการอาหาร

จากรูปที่ 3.6 เป็นการทำงานของโปรแกรมเพิ่มรายการอาหาร เมื่อลูกค้าต้องการสั่งอาหารเพิ่มเติมให้ดูรายละเอียดรายการอาหารในเมนูอาหารแล้วป้อนรหัสรายการอาหารที่ต้องการเพิ่มเติม โปรแกรมจะนำข้อมูลนั้นไปเก็บไว้ในฐานข้อมูล

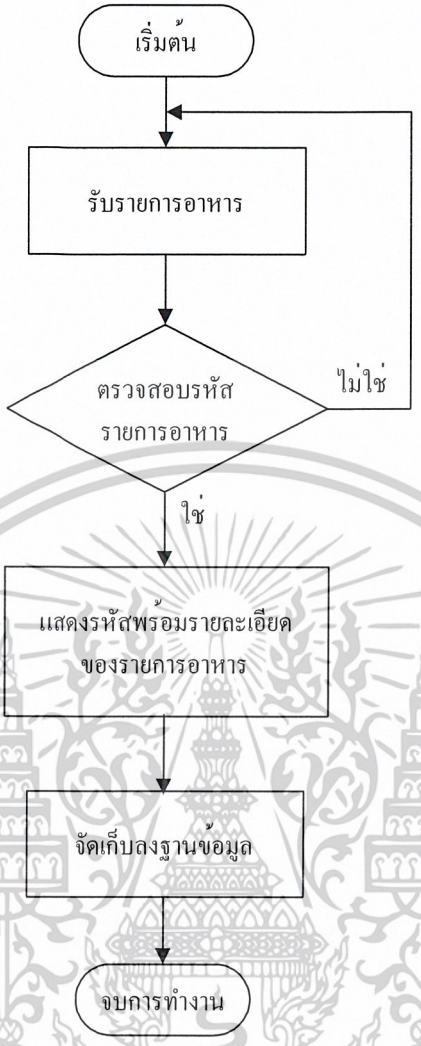


รูปที่ 3.6 ผังการทำงานของโปรแกรมเพิ่มรายการอาหาร

3.4.3 การออกแบบโปรแกรมรับรายการอาหาร

จากรูปที่ 3.7 เป็นผังการทำงานของโปรแกรมรับรายการอาหาร ซึ่งโปรแกรมจะรับรหัสรายการอาหารเข้ามาตรวจสอบว่ารหัสที่รับเข้ามาถูกต้องหรือไม่ ถ้ารหัสที่รับเข้ามาถูกต้องก็โปรแกรมจะแสดงรายละเอียดของรายการอาหารที่สั่ง และจัดเก็บรายการอาหารนั้นลงฐานข้อมูล ถ้ารหัสที่รับเข้ามาไม่ถูกต้องโปรแกรมก็จะแจ้งเตือนให้ป้อนรหัสที่ถูกต้องลงไป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

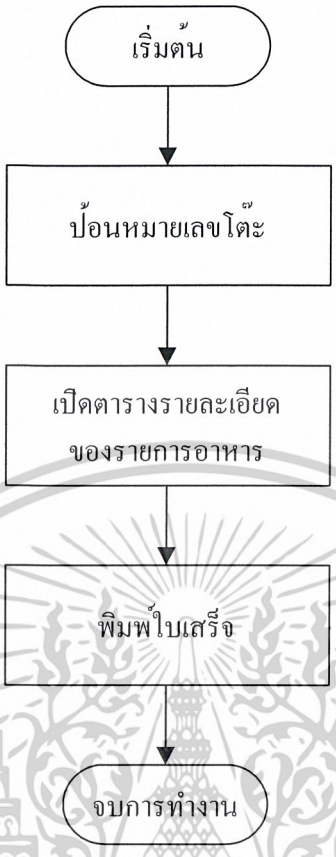


รูปที่ 3.7 ผังการทำงานโปรแกรมรับรายการอาหาร

3.4.4 การออกแบบโปรแกรมพิมพ์ใบเสร็จรับเงิน

จากรูปที่ 3.8 เป็นผังการทำงานของโปรแกรมพิมพ์ใบเสร็จรับเงิน ซึ่งเมื่อต้องการทราบว่าลูกค้าสั่งอาหารชนิดใดบ้างและเป็นจำนวนเงินเท่าใด ให้ป้อนหมายเลขโต๊ะโปรแกรมจะเปิดตารางรายละเอียดของรายการอาหาร และแสดงเป็นใบเสร็จทางเครื่องพิมพ์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



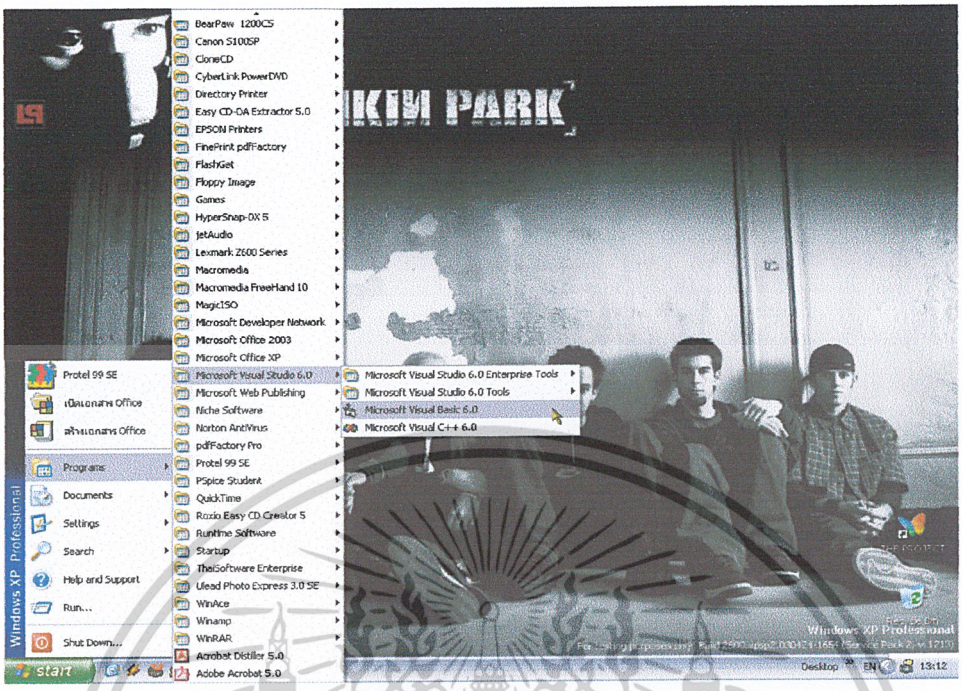
รูปที่ 3.8 ผังการทำงานของโปรแกรมพิมพ์ใบเสร็จ

3.4.5 ขั้นตอนการออกแบบฐานข้อมูล

จากลักษณะการทำงานทั้งหมดของโปรแกรมในฐานข้อมูลทั้ง 3 ส่วนมีขั้นตอนการสร้างโปรแกรมการใช้งาน ดังนี้

1) ขั้นตอนที่ 1 เริ่มต้นจากการเปิดโปรแกรมวิชวลเบสิกเวอร์ชัน 6 ที่เมนู Start, Program, Microsoft Visual Basic 6 ขึ้นมาจะพบกับลักษณะการทำงานที่เรียกว่า IDE (Integrated Development Environment) คือ หน้าต่างที่รวบรวมเครื่องมือ ข้อมูลต่างๆ ที่ใช้งานต่าง ๆ ไว้ในหน้าจอเดียว ซึ่งทำให้เราเลือกใช้งานได้ หลังจากนั้น เลือกคำสั่ง New Project จะได้ฟอร์มที่ใช้สำหรับวางออบเจ็กต์ต่างๆ เพื่อใช้ในการติดต่อกับฐานข้อมูล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.9 การเปิดโปรแกรม Microsoft Visual Basic 6

2) ขั้นตอนที่ 2 เป็นการออกแบบการสร้างเมนูรูปภาพเพื่อความสะดวกในการติดต่อกับผู้ใช้งานยิ่งขึ้น ในขั้นตอนนี้ต้องจัดเตรียมรูปภาพเพื่อใช้ในการสร้าง โดยการสร้างให้เลือกที่ออบเจ็กต์ Image List เพื่อเลือกรูปภาพที่จะใช้ติดกับผู้ใช้โปรแกรม จากนั้นเลือกออบเจ็กต์ Toolbar เพื่อใช้กำหนดคุณสมบัติเกี่ยวกับเมนูรูปภาพที่เตรียมไว้ ซึ่งการเลือกรูปภาพจะต้องสอดคล้องกับเมนูที่ได้ออกแบบ และสร้างไว้ในขั้นตอนที่ 1 เมนูรูปภาพที่เลือกนำมาออกแบบมีดังนี้

2.1) เพิ่มรายการอาหาร ใช้สำหรับการเพิ่มรายการอาหารลงในฐานข้อมูล

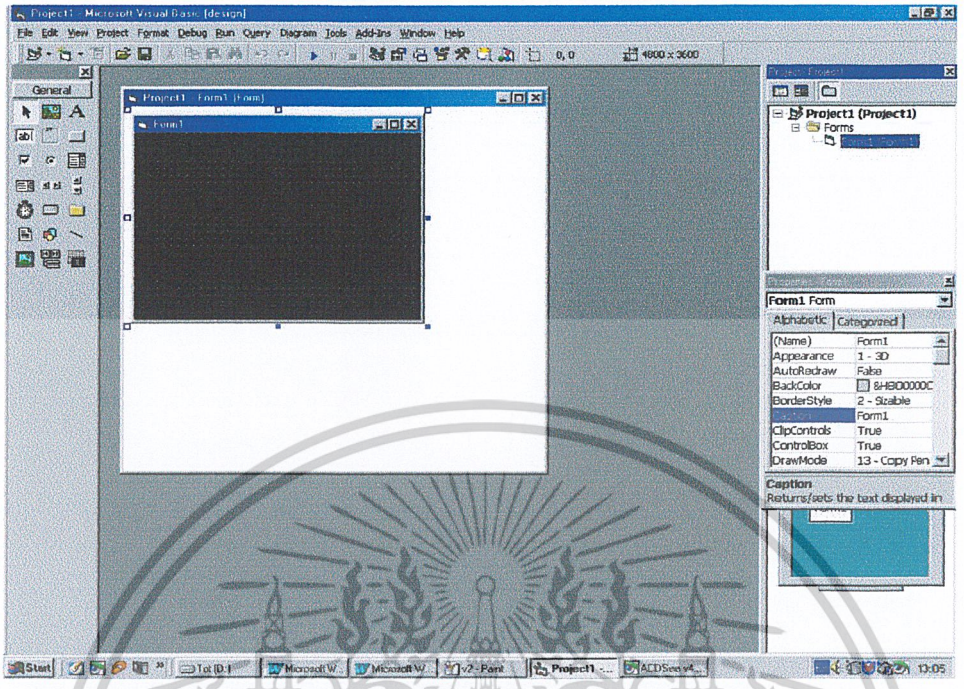
2.2) รายการสั่งอาหาร ใช้สำหรับรับรายการอาหารที่ทำการสั่งเข้ามา

2.3) พิมพ์ใบเสร็จ ใช้สำหรับสั่งพิมพ์ใบเสร็จรายการอาหาร

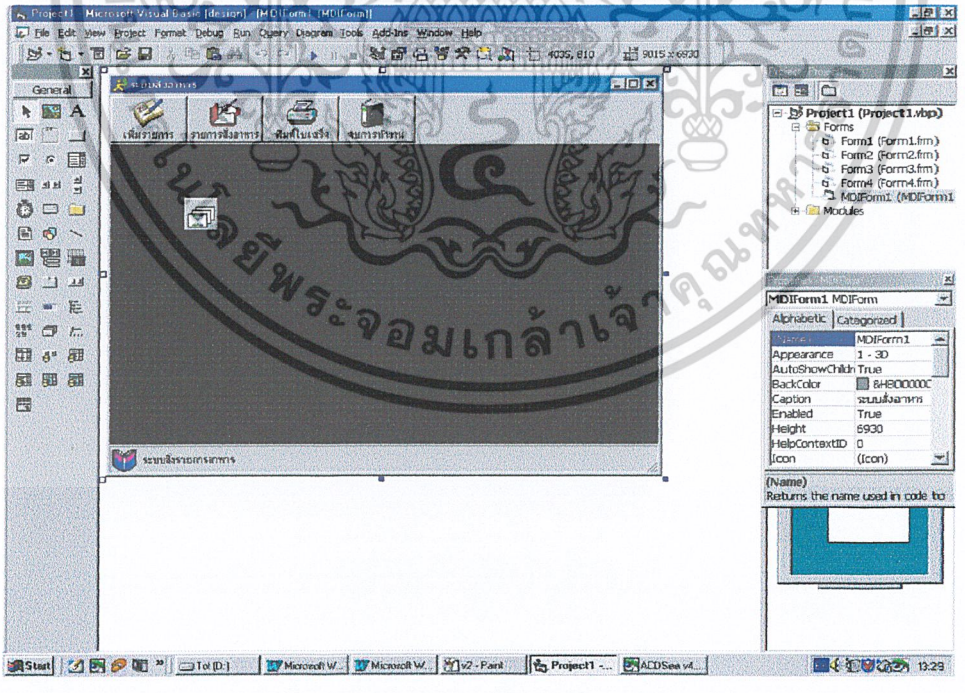
2.4) จบการทำงาน ใช้สำหรับปิดโปรแกรมเมื่อเสร็จสิ้นการทำงาน

เมื่อผู้ใช้เปิดโปรแกรมขึ้นมาใช้งานและต้องการเลือกใช้คำสั่งต่าง ๆ สามารถเลือกที่เมนูรูปภาพได้ตามต้องการ เมื่อเสร็จสิ้นกระบวนการแล้วจะแสดงผลได้ดังรูปที่ 3.11

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.10 การสร้างฟอร์มเพื่อติดต่อกับฐานข้อมูล



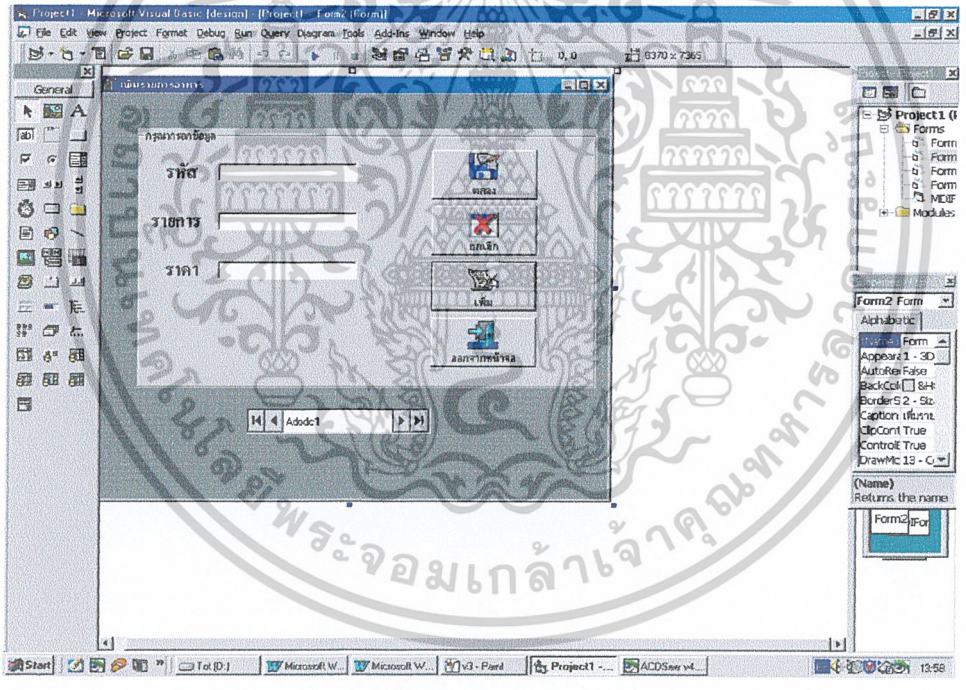
รูปที่ 3.11 การสร้างฟอร์มหลักในการใช้งาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3) ขั้นตอนที่ 3 เป็นการออกแบบการสร้างฟอร์มที่ใช้ในการเพิ่มจำนวนรายการอาหารเข้าไปในฐานข้อมูล และฟอร์มการค้นหารายการอาหาร เพื่อสะดวกในการเพิ่มจำนวนรายการอาหารและค้นหารายการอาหารที่ต้องการ ซึ่งสามารถสร้างฟอร์มได้ดังรูปที่ 3.12

ซึ่งภายในฟอร์มเพิ่มรายการอาหารและค้นหารายการอาหารนั้นจะมีปุ่มต่างๆ ซึ่งมีหน้าที่การทำงานดังนี้

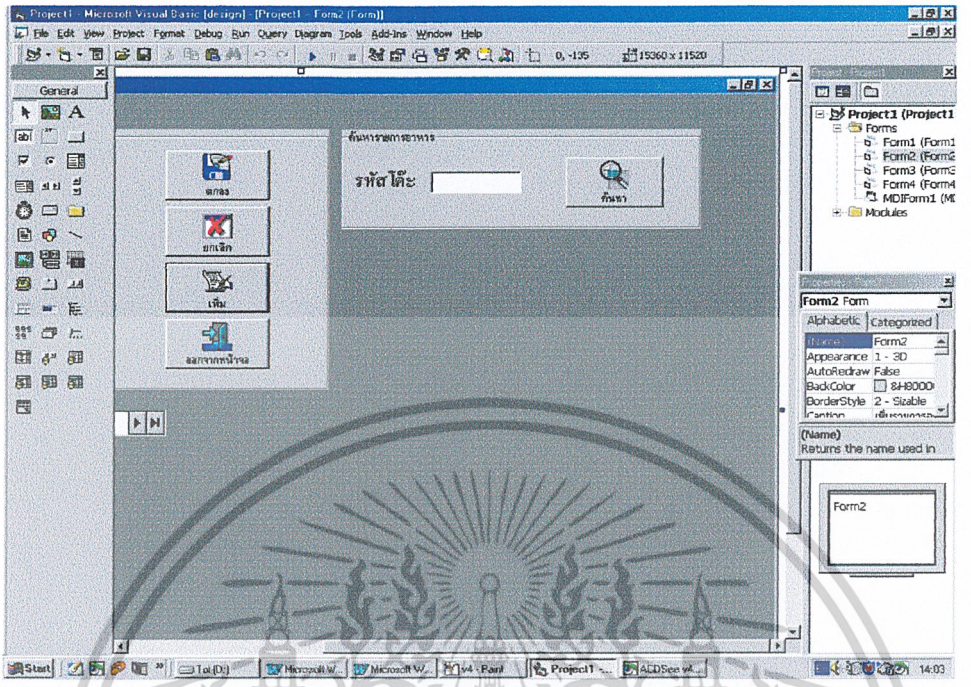
3.1) เพิ่ม	ใช้กดเมื่อต้องการเพิ่มรายการอาหารลงในฐานข้อมูล
3.2) ตกลง	ใช้กดเพื่อเป็นการยืนยันที่จะบันทึกข้อมูล
3.3) ยกเลิก	ใช้กดเพื่อเป็นการยกเลิกการบันทึกข้อมูล
3.4) ออกจากหน้าจอ	ใช้กดเพื่อเป็นการออกไปสู่หน้าจอหลัก
3.5) ค้นหา	ใช้กดเพื่อทำการค้นหารายการอาหารตามรหัสที่ต้องการ



(ก) ฟอร์มการเพิ่มรายการอาหาร

รูปที่ 3.12 การสร้างฟอร์มเพิ่มรายการและค้นหารายการอาหาร

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



(ข) แสดงฟอร์มการค้นหารายการอาหาร

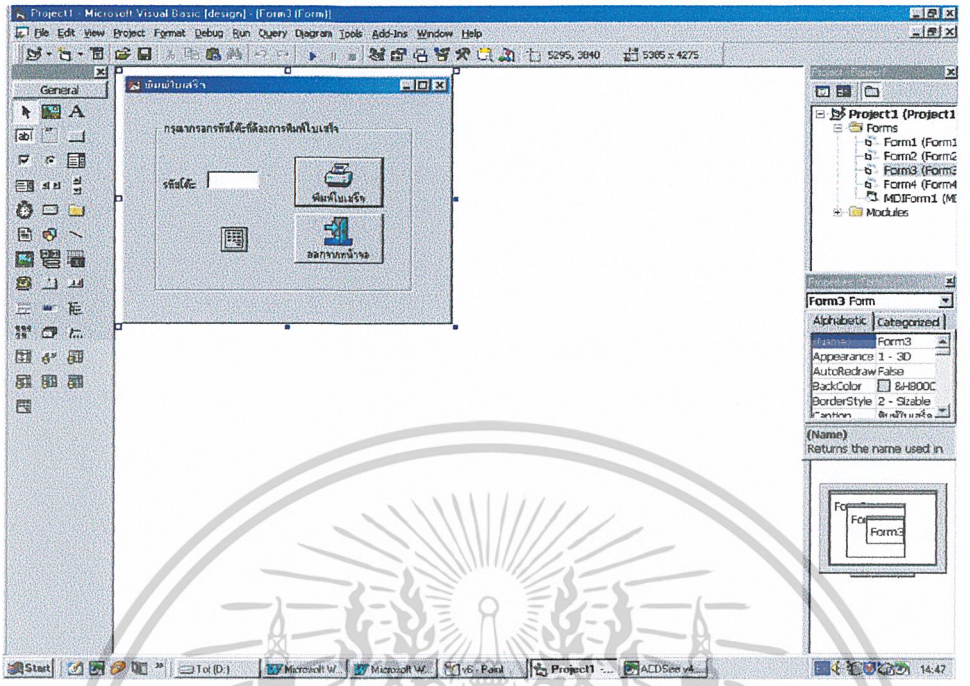
รูปที่ 3.12 (ต่อ) การสร้างฟอร์มเพิ่มรายการและค้นหารายการอาหาร

4) ขั้นตอนที่ 4 เป็นการออกแบบการสร้างฟอร์มที่ใช้ในการรับรายการอาหารที่มีการสั่งเข้ามา และแสดงรายการที่มีการสั่งเข้ามา การสร้างฟอร์มทำได้ดังรูปที่ 3.13

ซึ่งภายในฟอร์มรับรายการอาหารนั้นจะมีปุ่มต่างๆ ซึ่งมีหน้าที่การทำงานดังนี้

4.1) ลบรายการอาหาร	ใช้เพื่อลบรายการอาหารใน List View
4.2) ลบข้อมูล	ใช้เพื่อลบรายการอาหารในฐานข้อมูล
4.3) ออกจากหน้าจอ	ใช้กดเพื่อเป็นการออกไปสู่หน้าจอหลัก
4.4) List View	ใช้ในการรับรายการอาหารที่มีการสั่งเข้า

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.13 การสร้างฟอร์มรับรายการอาหารที่มีการสั่ง

5) ขั้นตอนที่ 5 เป็นการออกแบบการสร้างฟอร์มที่ใช้ในการพิมพ์ใบเสร็จของรายการอาหารที่มีการสั่งของแต่ละโต๊ะ ซึ่งการใช้งานนั้นสามารถใช้ได้โดยการเลือกโต๊ะที่ต้องการพิมพ์ใบเสร็จ แล้วกดปุ่มพิมพ์ใบเสร็จโปรแกรมก็ทำการติดต่อกับ Crystal Reports ที่สร้างรูปแบบใบเสร็จไว้เรียบร้อยแล้ว การออกแบบการสร้างฟอร์มที่ใช้ในการพิมพ์ใบเสร็จนั้นสามารถทำได้ดังรูปที่ 3.14

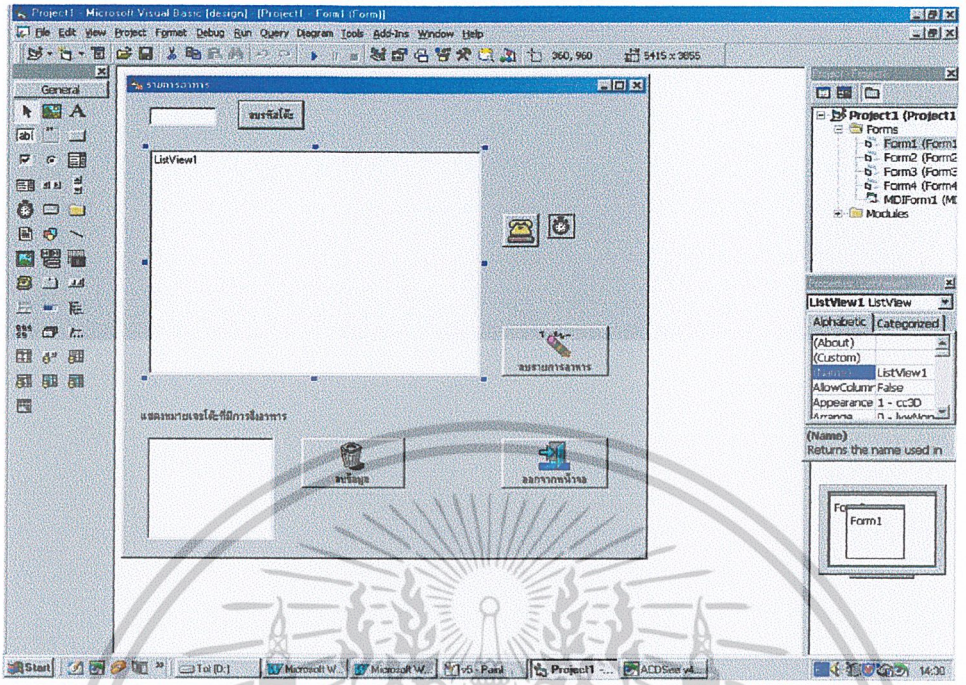
ซึ่งภายในฟอร์มพิมพ์ใบเสร็จรายการอาหารนั้นจะมีปุ่มต่างๆ ซึ่งมีหน้าที่การทำงานดังนี้

5.1) พิมพ์ใบเสร็จ ใช้เพื่อพิมพ์ใบเสร็จรายการอาหารในแต่ละโต๊ะ

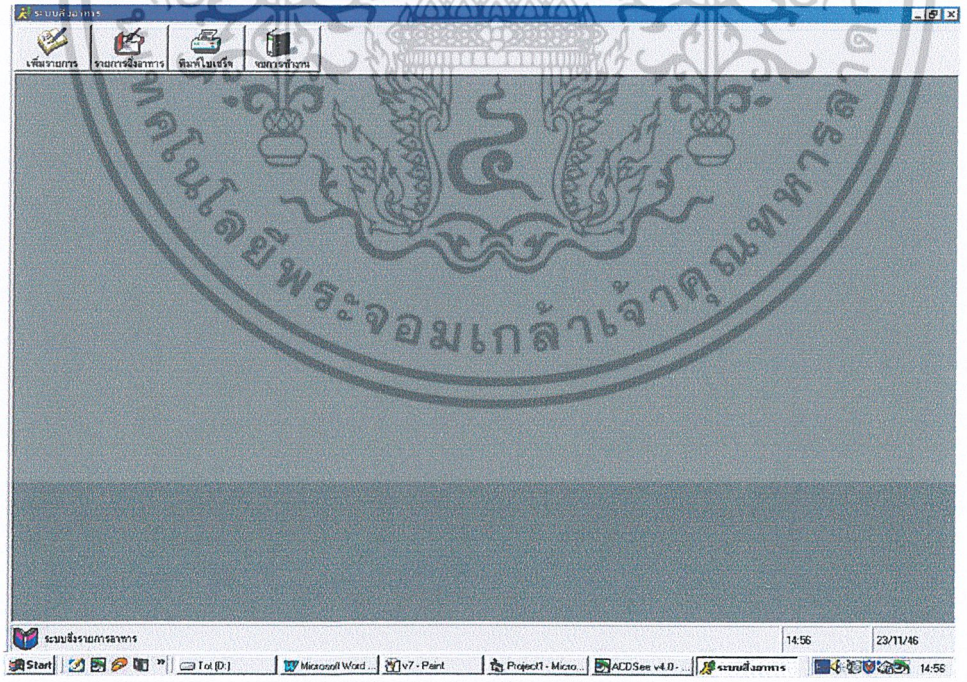
5.2) ออกจากหน้าจอ ใช้กดเพื่อเป็นการออกไปสู่หน้าจอหลัก

จะได้ฟอร์มต่างๆ ดังแสดงไว้ในดังรูปที่ 3.15 ถึง รูปที่ 3.19

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

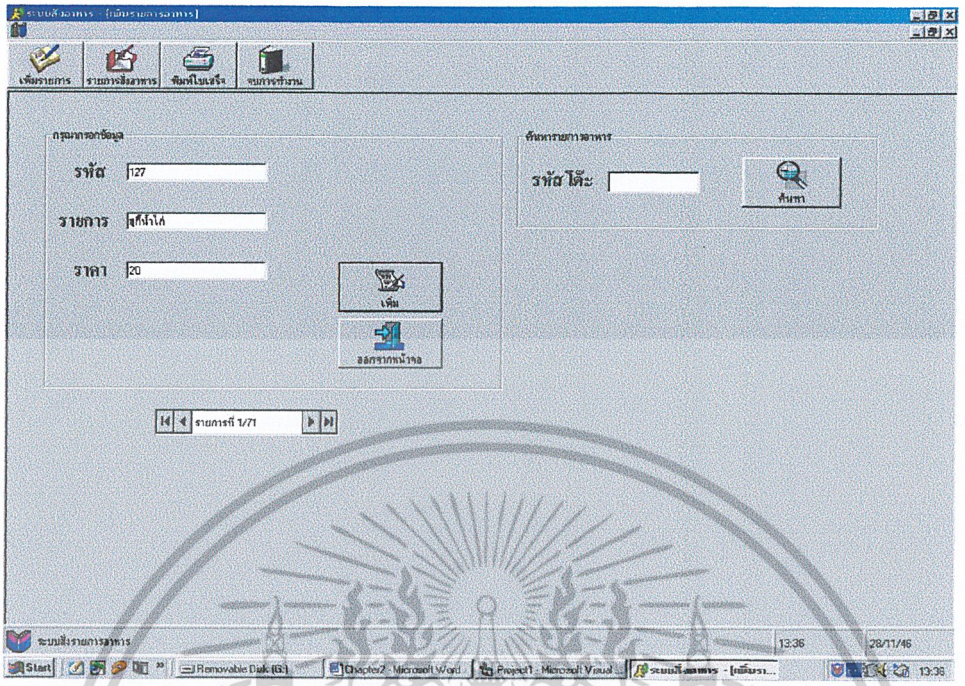


รูปที่ 3.14 การออกแบบการสร้างฟอร์มที่ใช้ในการพิมพ์ใบเสร็จ

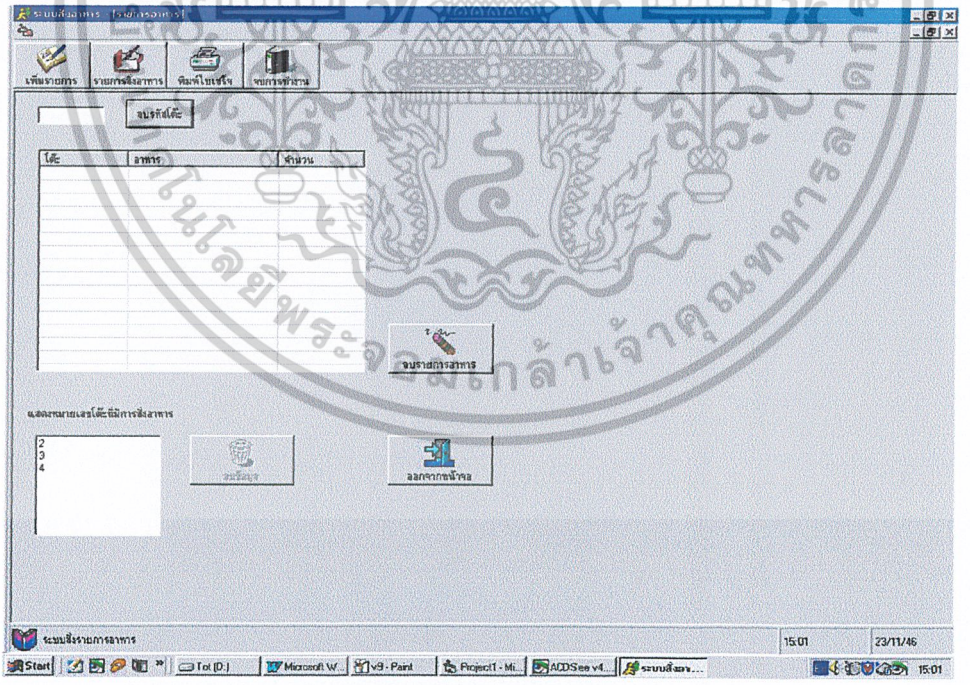


รูปที่ 3.15 ฟอร์มหลักที่ใช้งานในฐานข้อมูล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

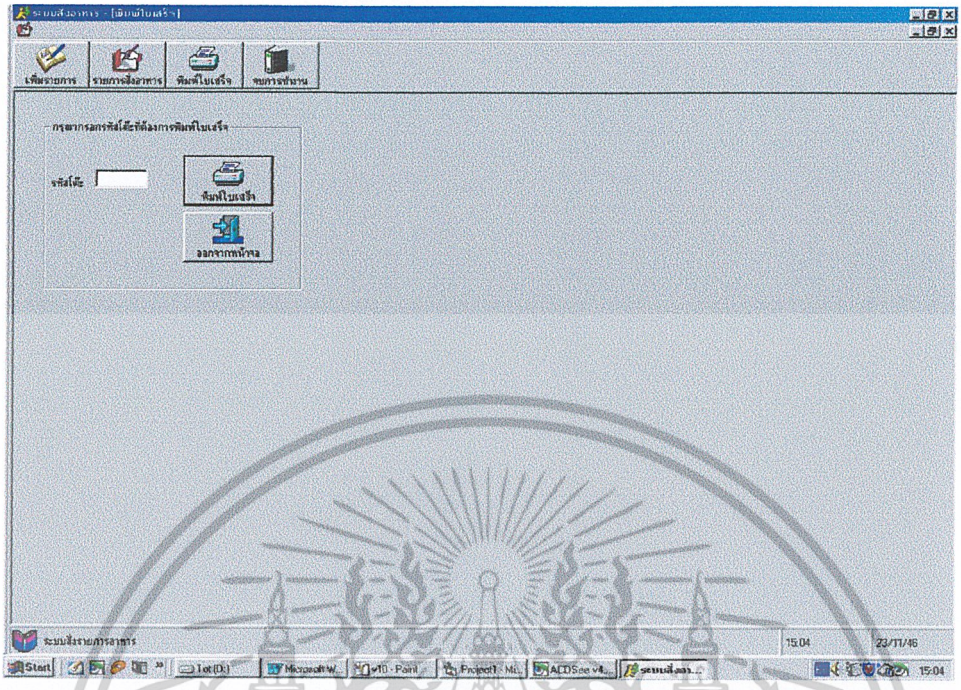


รูปที่ 3.16 ฟอร์มเพิ่มรายการและค้นหารายการอาหาร

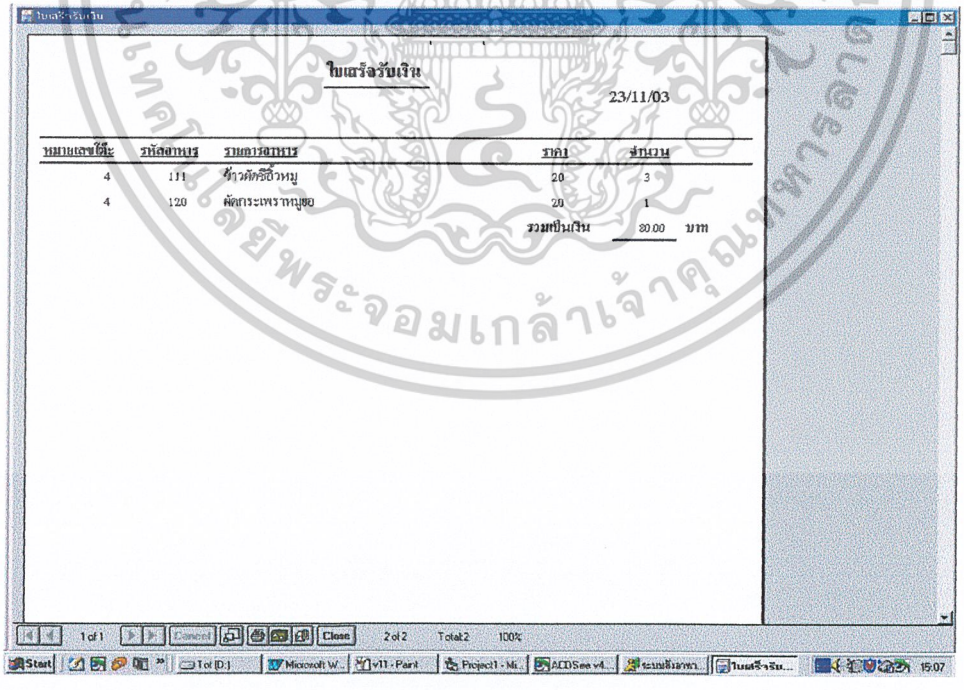


รูปที่ 3.17 ฟอร์มรับรายการอาหารเมื่อมีการสั่งรายการอาหาร

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.18 ฟอร์มที่ใช้ในการพิมพ์ใบเสร็จ



รูปที่ 3.19 รูปแบบใบเสร็จ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 4

การทดลองและผลการทดลอง

4.1 บทนำ

เพื่อให้การทดลองและการตรวจสอบการทำงานของระบบง่ายขึ้น จึงแบ่งภาคการทำงานของระบบออกเป็นส่วนต่างๆ คือ ส่วนการแสดงผล, ส่วนวงจรเครื่องส่ง, ส่วนวงจรเครื่องรับ, การส่งและการอ่านข้อมูลผ่านพอร์ตอนุกรม, การส่งรายการอาหาร และการใช้เครื่องส่งอาหารระบบอัตโนมัติ ทำการทดลองได้ผลการทดลองดังต่อไปนี้

4.2 ส่วนการแสดงผล

ในส่วนการแสดงผลนี้ สามารถทำการทดลองและได้ผลการทดลองดังต่อไปนี้

4.2.1 การทดลอง

ทำการทดลองนำรหัสหลายๆ ตัวป้อนเข้าไปในส่วนการแสดงผลที่ควบคุมด้วยโปรแกรมการแสดงผล โดยรหัสที่ป้อนในการทดลองนี้ได้สมมติขึ้น เพื่อทดลองโปรแกรมควบคุมของระบบ

4.2.2 ผลการทดลอง

จากตารางที่ 4.1 การทดลองส่วนการแสดงผล จะเห็นได้ว่าจากการทดลองนำรหัสหลายๆ ตัวป้อนเข้าไปการทดลองที่ได้ คือ รหัสที่ป้อนกับรหัสที่แสดงมีค่าเหมือนกัน คือ ผลการทดลองที่ได้ถูกต้องตรงตามความต้องการ

ตารางที่ 4.1 การทดลองส่วนการแสดงผล

ครั้งที่	รหัสที่ป้อน	รหัสที่แสดง
1	1101	1101
2	1302	1302
3	1120	1120
4	2310	2310
5	1144	1144

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.3 การติดต่อระหว่างเครื่องส่งกับพอร์ตอนุกรม

ในส่วนการติดต่อระหว่างเครื่องส่งข้อมูลกับพอร์ตอนุกรม สามารถทำการทดลองและได้ผลการทดลองดังนี้

4.3.1 การทดลอง

ทำการทดลองโดยต่อเครื่องส่งเข้ากับคอมพิวเตอร์ แล้วทำการป้อนรหัสรายการอาหารใน

4.3.2 ผลการทดลอง

ทดลองการส่งข้อมูลโดยติดต่อผ่านพอร์ตอนุกรม ซึ่งวิธีการทดลองทำได้โดยต่อเครื่องส่งเข้ากับคอมพิวเตอร์ และป้อนรหัสรายการอาหาร ซึ่งแสดงผลการทดลองได้จาก Hyper Terminal ดังตารางที่ 4.2

ตารางที่ 4.2 การทดลองส่งข้อมูลโดยติดต่อผ่านพอร์ตอนุกรม

ครั้งที่	รหัสที่ป้อน	รหัสแสดงที่หน้าจอ LCD	รหัสแสดงที่โปรแกรม
1	1101	1101	011101
2	1302	1302	011302
3	1120	1120	011120
4	2110	2110	012110
5	1144	1144	011144
6	4152	4152	014152
7	1120	1120	011120
8	5131	5131	015131

จากผลการทดลองสังเกตในช่องรหัสแสดงที่โปรแกรม ซึ่งจะมีรหัส 01 เพิ่มขึ้นมาด้านหน้า คือ หมายเลขโต๊ะ ซึ่งหมายเลขโต๊ะจะส่งมาจากเครื่องลูกข่ายในแต่ละเครื่อง ที่มีหมายเลขประจำเครื่องไว้เรียบร้อยแล้ว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.4 การติดต่อระหว่างเครื่องรับกับเครื่องส่ง

ในส่วนการติดต่อระหว่างเครื่องรับกับเครื่องส่ง สามารถทำการทดลองและผลที่ได้จากการทดลองเป็นดังนี้

4.4.1 การทดลอง

ทดลองโปรแกรมติดต่อระหว่างเครื่องส่งกับพอร์ตอนุกรม และทำการทดลองติดต่อระหว่างเครื่องรับกับเครื่องส่ง

4.4.2 ผลการทดลอง

หลังจากที่ได้ทดลองการติดต่อระหว่างเครื่องส่งกับพอร์ตอนุกรมเรียบร้อยแล้ว จากนั้นก็ทำการทดลองการติดต่อระหว่างเครื่องรับกับเครื่องส่ง ซึ่งสามารถทำได้โดยการต่อเครื่องส่งเข้ากับเครื่องรับแล้วทำการป้อนรหัสรายการอาหารเข้าไป และดูผลการทดลองที่ได้จากโปรแกรม Hyper Terminal ซึ่งผลการทดลองในส่วนนี้จะเหมือนกับส่วนแรก ดังตารางที่ 4.1

4.5 การเพิ่มรายการอาหาร

ในส่วนการเพิ่มรายการอาหารได้ทำการทดลองและได้ผลการทดลองดังนี้

4.5.1 การทดลอง

1) กดปุ่มเพิ่มรายการอาหาร หน้าต่างการเพิ่มรายการอาหารจะมีปุ่ม OK และ CANCLE แสดงขึ้นมา ดังรูปที่ 4.1

2) กรอกรหัสอาหาร รายการอาหาร และราคาอาหารที่ต้องการสั่งเพิ่ม

3) เมื่อกรอกข้อมูลครบ ถ้าต้องการบันทึกข้อมูลลงฐานข้อมูลให้ทำการกดปุ่ม OK

4) เมื่อกรอกข้อมูลครบ ถ้าต้องการยกเลิกข้อมูลให้กดปุ่ม CANCLE

5) เมื่อกดปุ่ม OK ข้อมูลก็จะบันทึกลงฐานข้อมูล

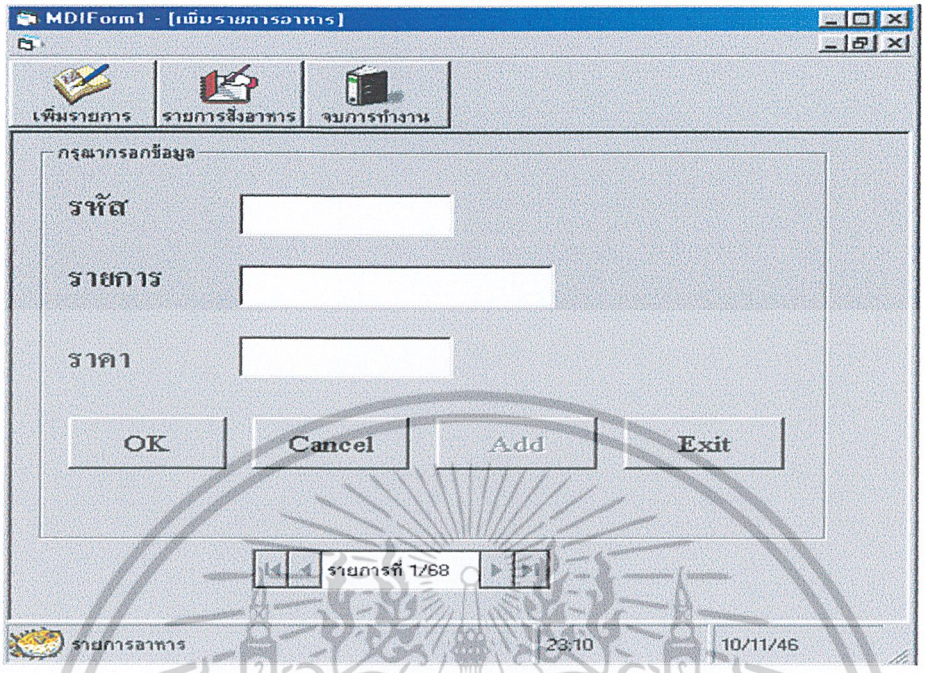
6) เมื่อกดปุ่ม CANCLE ข้อมูลที่กรอกเข้าไปก็จะถูกยกเลิกไป

7) จากข้อ 3 ถึงข้อ 7 จะแสดงดังรูปที่ 4.2

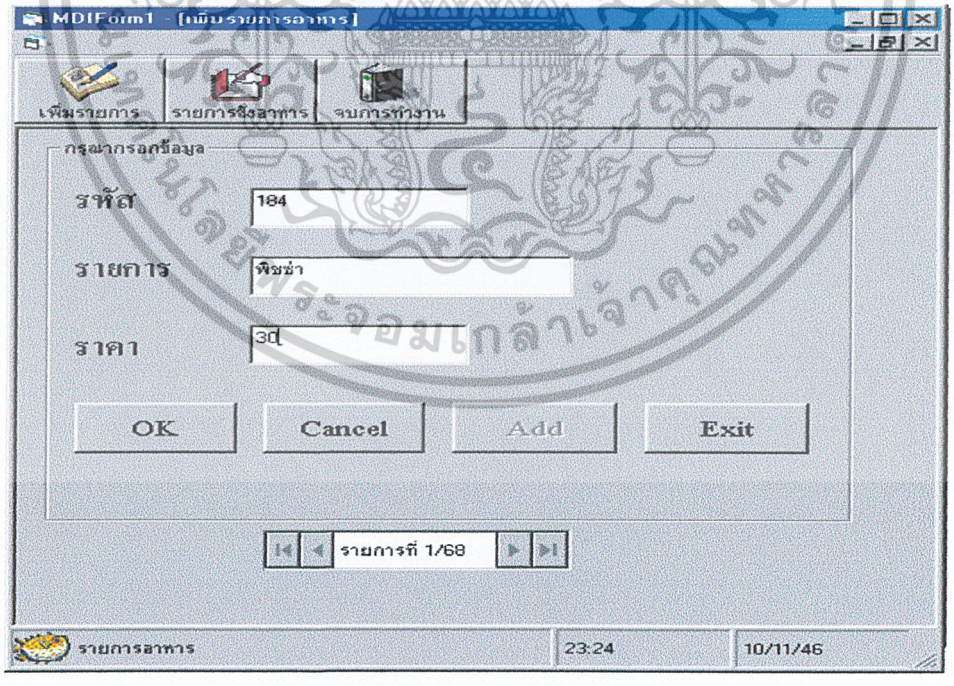
4.5.2 ผลการทดลอง

เมื่อทำการเพิ่มข้อมูลและกรอกข้อมูลครบแล้ว กดปุ่ม OK ข้อมูลก็จะบันทึกลงฐานข้อมูล ถ้ากรอกข้อมูลครบและกดปุ่ม CANCLE ข้อมูลก็จะถูกยกเลิกไป และถ้าหากกรอกข้อมูลไม่ครบ โปรแกรมก็จะแจ้งเตือนว่ายังกรอกข้อมูลไม่ครบ ผลการทดลองที่ได้เป็นไปตามที่กำหนด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.1 เมนูใช้ในการเพิ่มรายการอาหาร



รูปที่ 4.2 การเพิ่มรายการอาหารในเมนู

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.6 การค้นหารายการอาหาร

จากการทดลองค้นหารายการอาหาร ได้ทำการทดลองและได้ผลการทดลองดังนี้

4.6.1 การทดลอง

ทดลองโดยการค้นหารายการอาหารที่มีอยู่ในฐานข้อมูล โดยการกรอกรหัสรายการอาหารตามที่ต้องการแล้วกดปุ่มค้นหา

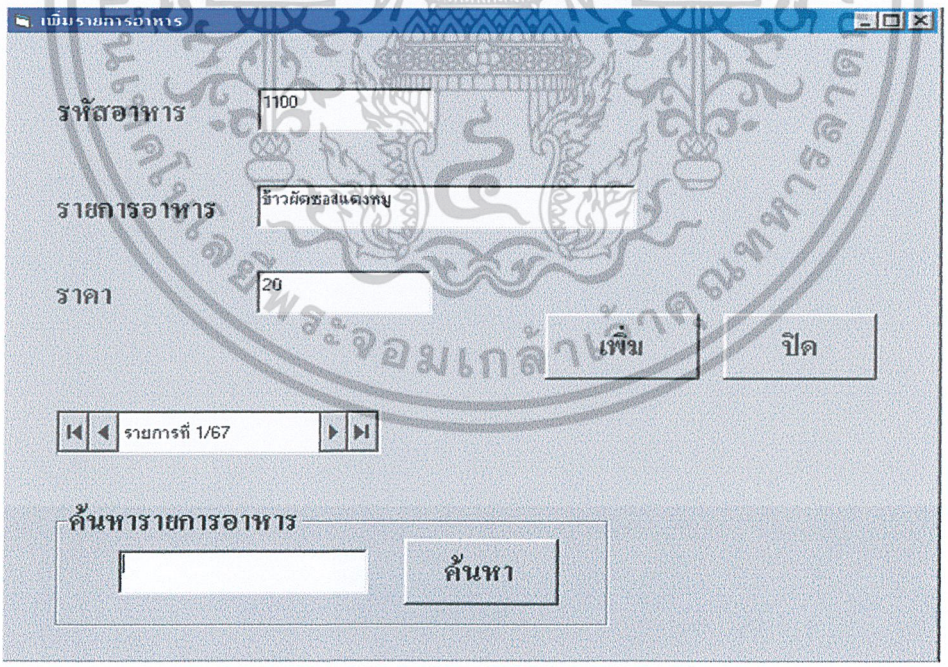
4.6.2 ผลการทดลอง

จากการทดลองการติดต่อกับฐานข้อมูล โดยการค้นหารายการอาหารที่มีอยู่ในฐานข้อมูล โดยเมื่อทำการสร้างฟอร์มและติดต่อกับฐานข้อมูลได้แล้ว ก็ทำการทดลองโดยการกรอกรหัสรายการอาหารตามที่ต้องการ และกดปุ่มค้นหา จากการทดลองได้ผลการทดลองดังรูปที่ 4.3 และรูปที่ 4.4

วิธีการทดลองทำได้ดังนี้

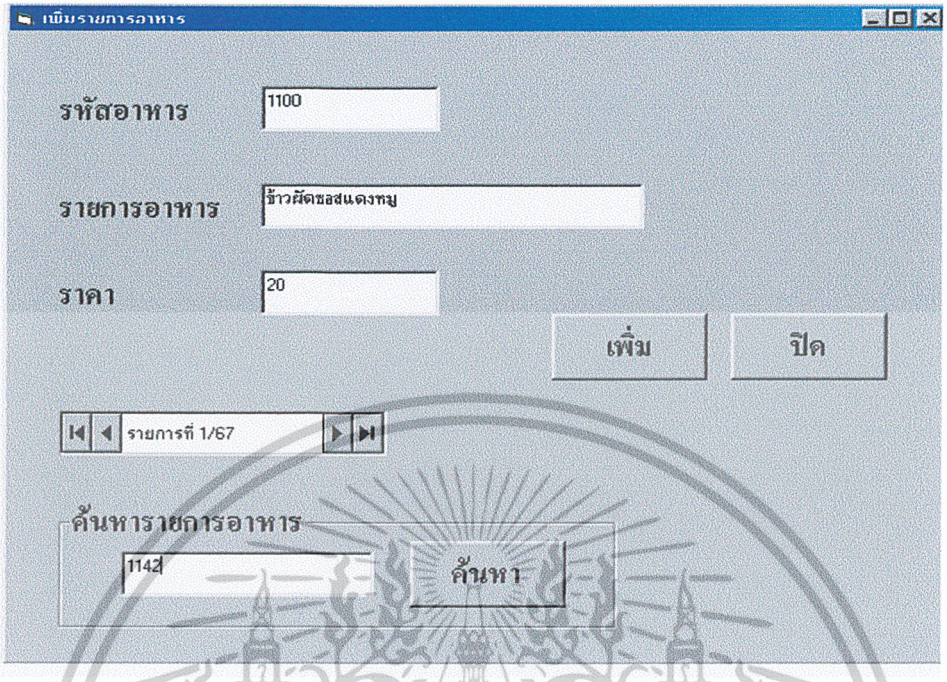
1) เปิดฟอร์มเพิ่มรายการอาหาร

2) กรอกรหัสรายการอาหารตามที่ต้องการ



รูปที่ 4.3 เมนูใช้ในการค้นหารายการอาหาร

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.4 การค้นหารายการอาหาร

3) กดปุ่มค้นหารายการอาหารที่ต้องการ รายการจะแสดงขึ้นมา ถ้ารหัสไม่ตรงกับข้อมูลที่มีอยู่ในฐานข้อมูลหรือไม่ตรงกับรายการอาหารที่มีอยู่ จะมีการแจ้งผลการค้นหาว่าไม่มีข้อมูลที่ต้องการ และถ้าไม่มีการกรอกรหัสรายการอาหารก็จะมีการแจ้งเตือนว่ายังไม่มีการกรอกข้อมูล ผลการทดลองจะแสดงได้ดังตารางที่ 4.3

ตารางที่ 4.3 ผลการทดลองป้อนรหัสรายการอาหาร

รหัสที่ป้อน	ผลการทดลอง		
	รหัส	รายการอาหาร	ราคา
100	100	ข้าวผัดซอสแดงไก่	20
101	101	ข้าวผัดซอสแดงหมู	20
102	102	ข้าวผัดซอสแดงทะเล	20
103	103	ข้าวผัดแหนม	20
104	104	ข้าวผัดหมู	20

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 4.3 (ต่อ) ผลการทดลองป้อนรหัสรายการอาหาร

รหัสที่ป้อน	ผลการทดลอง		
	รหัส	รายการอาหาร	ราคา
105	105	ข้าวผัดไก่	20
106	106	ข้าวผัดทะเล	20
107	107	ข้าวผัดหอยลาย	20
108	108	ข้าวผัดไข่	20
109	109	ข้าวผัดปู	20
110	110	ข้าวผัดปลากระป๋อง	20
111	111	ข้าวผัดซีอิ้วหมู	20
112	112	ข้าวผัดซีอิ้วไก่	20
113	113	ข้าวผัดซีอิ้วทะเล	20
114	114	ข้าวหมูหวาน	20
115	115	ข้าวไก่หวาน	20
116	116	ข้าวกุ้งหวาน	20

4.7 เครื่องสั่งอาหารระบบอัตโนมัติ

การทดลองและผลการทดลองในส่วนเครื่องสั่งอาหารอัตโนมัติ ทำได้ดังนี้

4.7.1 การทดลอง

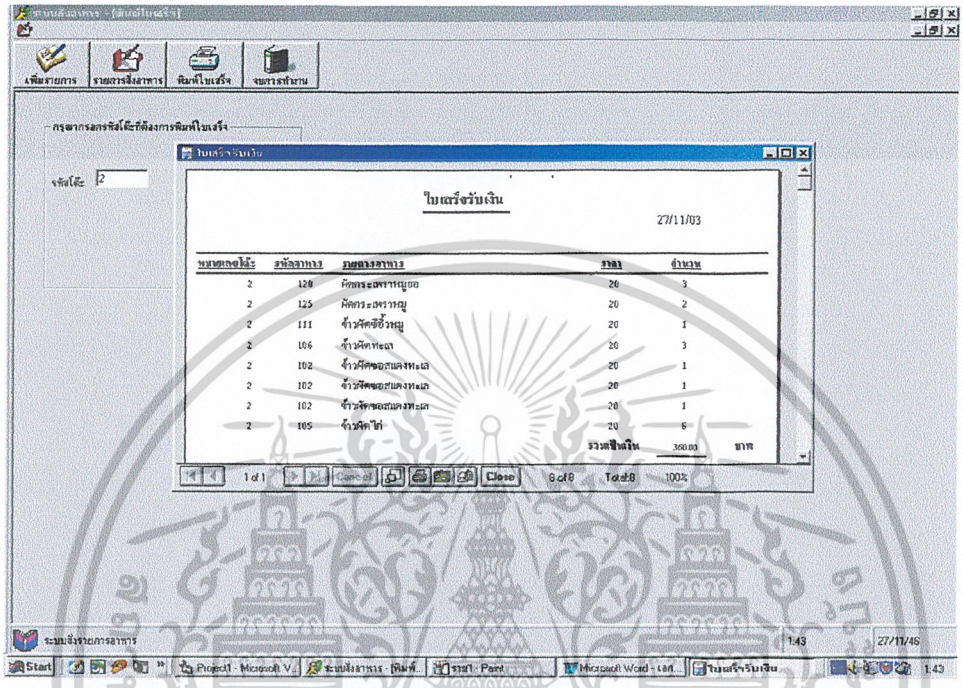
เมื่อทำการทดลองส่วนต่างๆ เสร็จ จากนั้นทำการเชื่อมต่อส่วนต่างๆ ในระบบเข้าด้วยกันให้เรียบร้อย โดยต่อเครื่องส่งทั้งหมดเข้ากับเครื่องรับ ในส่วนเครื่องรับต่อเข้าคอมพิวเตอร์โดยผ่านทางพอร์ตอนุกรม เมื่อต่อระบบสมบูรณ์แล้วทำการเปิดโปรแกรมรับรายการอาหารขึ้นมา เลือกเมนูสั่งรายการอาหาร จากนั้นทดลองสั่งอาหารตามคู่มือรายการอาหาร โดยการกดรหัสอาหารตามที่ต้องการ

4.7.2 ผลการทดลอง

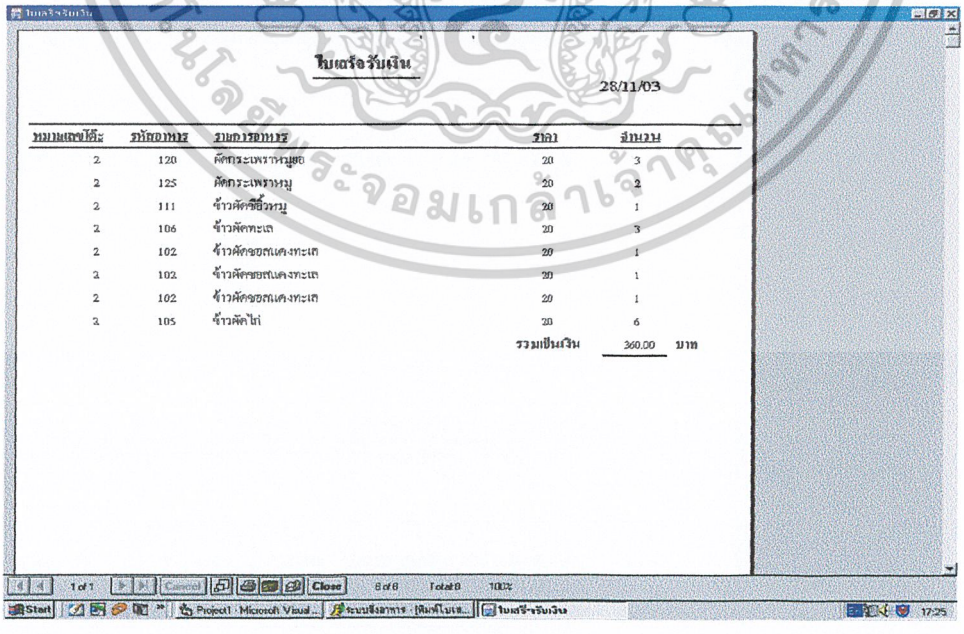
เมื่อทำการส่งรายการอาหารเรียบร้อยแล้ว รายการอาหารที่สั่งก็จะไปปรากฏที่หน้าต่างเมนูสั่งรายการอาหาร และสามารถพิมพ์ออกมาเป็นใบเสร็จรับเงินได้ ซึ่งผลการทดลองที่ได้จาก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ใบเสร็จรับเงินมีหมายเลขโต๊ะ, รายการ และจำนวนเงินทั้งหมดตรงกับที่ได้สั่งไว้ ซึ่งแสดงให้เห็นว่าเครื่องสั่งอาหารระบบอัตโนมัติมีความเที่ยงตรงในการทำงาน



รูปที่ 4.5 การพิมพ์ ใบเสร็จรับเงิน



รูปที่ 4.6 ใบเสร็จรับเงิน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 5
บทสรุป

5.1 สรุป

เครื่องสั่งอาหารระบบอัตโนมัติ ประกอบไปด้วย 4 ส่วน คือ เครื่องส่งรายการอาหาร เครื่องรับรายการอาหาร ฐานข้อมูล และเครื่องพิมพ์ ซึ่งในการใช้งานนั้นสามารถทำได้โดยกดรหัสอาหารตามต้องการที่เครื่องส่ง ซึ่งรหัสอาหารที่สั่ง จะถูกเก็บไว้ในหน่วยความจำจนกว่าจะมีการกดปุ่มยืนยันการสั่งอาหาร ระบบก็จะส่งรหัสรายการอาหารไปแสดงที่จอคอมพิวเตอร์ และส่งข้อมูลที่ได้ไปยังเครื่องพิมพ์ เพื่อพิมพ์ใบเสร็จ เครื่องสั่งอาหารระบบอัตโนมัตินี้ ทำให้เราได้เข้าใจการทำงานของระบบการส่ง การรับข้อมูล การเขียนโปรแกรมในฐานข้อมูล การเขียนโปรแกรมควบคุมการส่งและการรับข้อมูล ผ่านพอร์ตอนุกรมอนุกรมมาตรฐาน RS-232 เข้าใจการทำงานในด้านเครื่องส่งและเครื่องรับที่ใช้ไมโครคอนโทรลเลอร์ AT89C52 ควบคุม เครื่องสั่งอาหารระบบอัตโนมัติที่สร้างขึ้นในโครงงานนี้เป็นเครื่องต้นแบบในการใช้งาน เหมาะกับนักศึกษา และผู้สนใจทั่วไปได้ศึกษา เพื่อนำระบบไปพัฒนาให้ใช้งานได้ดีกว่านี้หรือนำระบบเครื่องสั่งอาหารระบบอัตโนมัตินี้ไปพัฒนาเป็นระบบต่างๆ ได้

5.2 ปัญหาและแนวทางการแก้ไข

จากการดำเนินการสร้างและได้ทดสอบการทำงานของเครื่องสั่งอาหารระบบอัตโนมัติ พบว่ามีปัญหาเกิดขึ้นหลายประการ ซึ่งสามารถสรุปได้ดังนี้

1) ปัญหา การเขียนโปรแกรมติดต่อกับฐานข้อมูลไม่สามารถทำการติดต่อได้

แนวทางการแก้ไข ทำการตรวจสอบในส่วนของการติดต่อฐานข้อมูลว่า ได้มีการตั้งค่าในการติดต่อกับฐานข้อมูลเรียบร้อยแล้วหรือยัง

2) ปัญหา เครื่องส่งไม่สามารถส่งข้อถูกไปยังเครื่องรับได้

แนวทางแก้ไข ตรวจสอบ สายที่ใช้เชื่อมต่อระหว่างเครื่องส่งกับเครื่องรับและปุ่ม ON/OFF ของเครื่องรับ

3) ปัญหา เมื่อเปิดปุ่ม ON/OFF ของเครื่องส่งจอ LCD ไม่แสดงข้อความและเคอร์เซอร์

แนวทางแก้ไข ตรวจสอบแหล่งจ่ายแบตเตอรี่ 9 โวลต์ภายในวงจรเครื่องส่ง

4) ปัญหา บนหน้าจอเครื่องส่งไม่แสดงข้อความ Send Success หลังจากที่กดปุ่ม Send

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

แนวทางแก้ไข ตรวจสอบสวิตซ์ที่เครื่องรับ, สายที่ใช้เชื่อมต่อระหว่างเครื่องส่งกับเครื่องรับ

5) ปัญหา การออกแบบวงจรต่างๆ ทำได้ล่าช้ากว่าที่กำหนดไว้ เนื่องจากเดิมจะใช้ไมโครคอนโทรลเลอร์เบอร์ rfPIC12C509 เป็นตัวส่งสัญญาณและใช้เบอร์ rfPICRXD0420 เป็นตัวรับสัญญาณ เมื่อทำการสั่งซื้อ IC ทั้งสองตัวนี้ ปรากฏว่า IC ขาดตลาด จึงได้เปลี่ยนมาใช้ IC เบอร์ 12F675 เป็นตัวส่งและตัวรับยังคงใช้ IC ตัวเดิม หลังจากนั้น IC เบอร์ 12F675 ได้ส่งมาถึงก่อนกำหนดการสอบปลายภาคประมาณหนึ่งสัปดาห์ ทำให้ไม่มีเวลาในการศึกษารายละเอียดของ IC เบอร์ 12F675 มากนัก

แนวทางแก้ไข อาจารย์ที่ปรึกษาแนะนำให้เปลี่ยนมาทำเครื่องสั่งอาหารระบบอัตโนมัติแบบมีสายในการรับส่งข้อมูลแทนแบบไร้สาย

5.3 แนวทางการพัฒนา

โครงงานที่สร้างขึ้นนี้สามารถทำงานได้ตามขีดความสามารถที่กำหนดเอาไว้แล้ว แต่อย่างไรก็ตามยังสามารถที่จะเพิ่มขีดความสามารถในการทำงานของโครงงานนี้อีกดังต่อไปนี้

1) ควรออกแบบระบบการรับส่งให้ได้ระยะทางที่ไกลกว่าระบบนี้ เนื่องจากระบบนี้จำกัดการส่งภายในห้องอาหารที่มีขนาดไม่กว้างมากนัก

2) ควรออกแบบตัวเครื่องให้มีขนาดที่เล็กและกะทัดรัดมากกว่านี้

3) ควรพัฒนาโปรแกรม Visual Basic ที่ใช้ควบคุมให้มีการรับส่งข้อมูลที่มีความแน่นอนและควรพัฒนาหน้าต่างที่ใช้ในการรับข้อมูลให้มีความสวยงามมากขึ้น

4) ควรพัฒนาโปรแกรมให้มีความกระชับและมีคุณภาพมากกว่านี้ โดยควรตัดในส่วนที่ไม่จำเป็นในการใช้งานทิ้งไป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บรรณานุกรม

ฉันทวุฒิ พืชผล และคณะ. **Visual Basic 6.** กรุงเทพฯ : โปรวิชั่น. 2542

ณัฐพล วงค์สุนทรชัย และคณะ. **ปฏิบัติการไมโครคอนโทรลเลอร์ PIC12F675.** กรุงเทพฯ : อินโนเวฟเอ็ก เพอริเมนต์. 2521

วุฒิพงศ์ พงศ์สุวรรณ และคณะ. **Visual Basic Version 6.** กรุงเทพฯ : แอดวานซ์ มีเดีย ซัพพลายส์. 2543

วรพจน์ กรแก้ววัฒนกุล. **เรียนรู้ปฏิบัติการไมโครคอนโทรลเลอร์ MCS - 51.** กรุงเทพฯ : อินโนเวตีฟเอ็กเพอริเมนต์. 2540

สัจจะ จรัสรุ่งรวีวร. **คู่มือการเขียนโปรแกรมและใช้งาน Visual Basic 6.** นนทบุรี : อินโฟเพรส. 2542

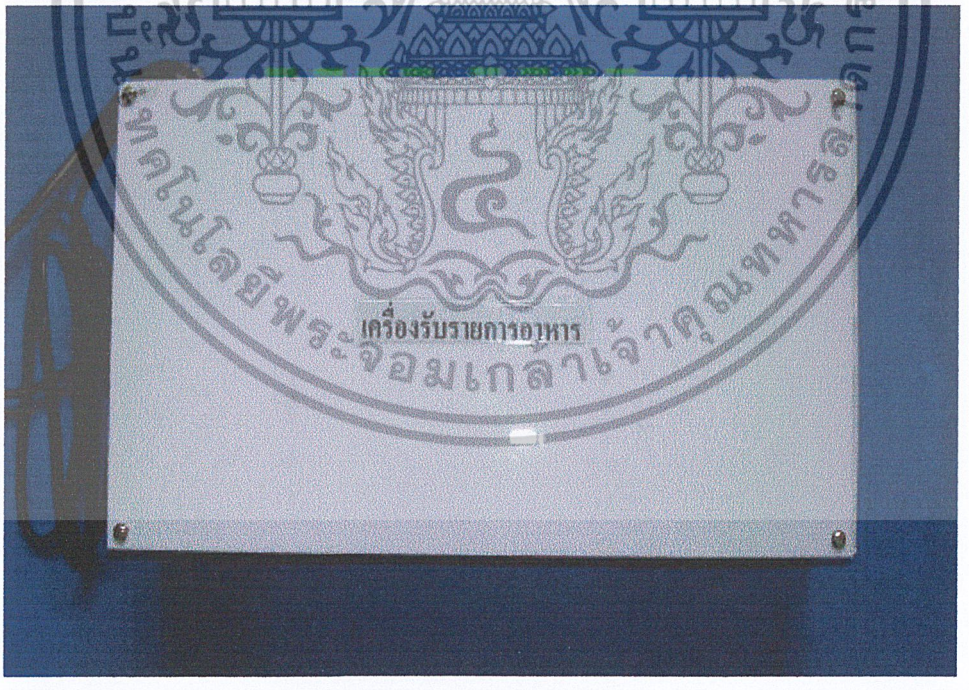
อรรถพล บุญโภคา และคณะ. **เรียนรู้และปฏิบัติการเชื่อมต่อคอมพิวเตอร์กับอุปกรณ์ภายนอกผ่านพอร์ตอนุกรม.** พิมพ์ครั้งที่ 2. กรุงเทพฯ : อินโนเวตีฟ เอ็กเพอริเมนต์. 2542

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก ก
เครื่องต้นแบบ

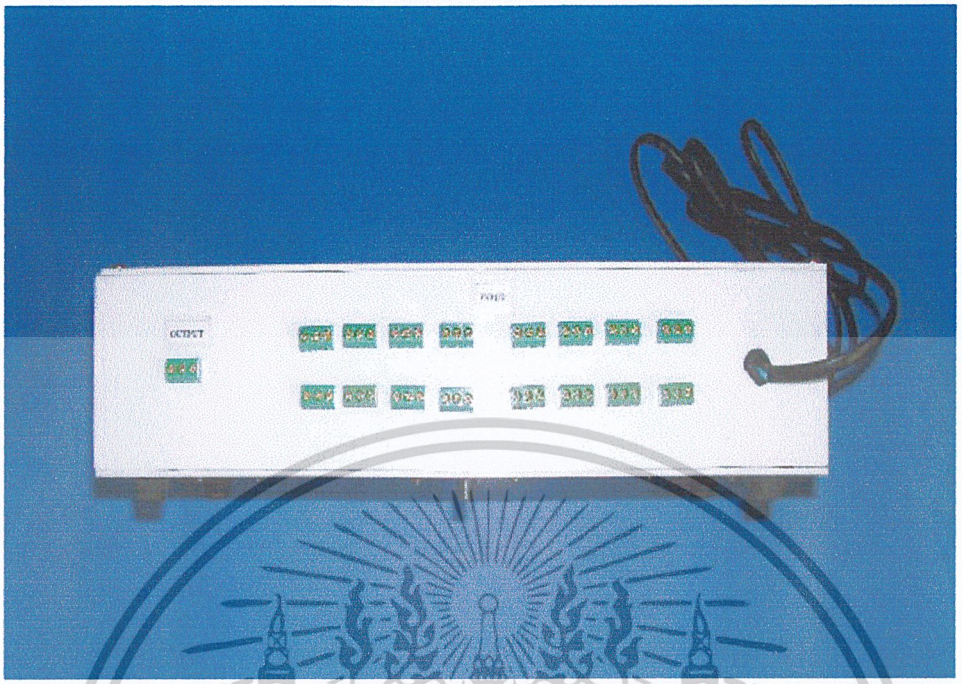
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ ก.1 ภาพด้านหน้าของเครื่องส่ง

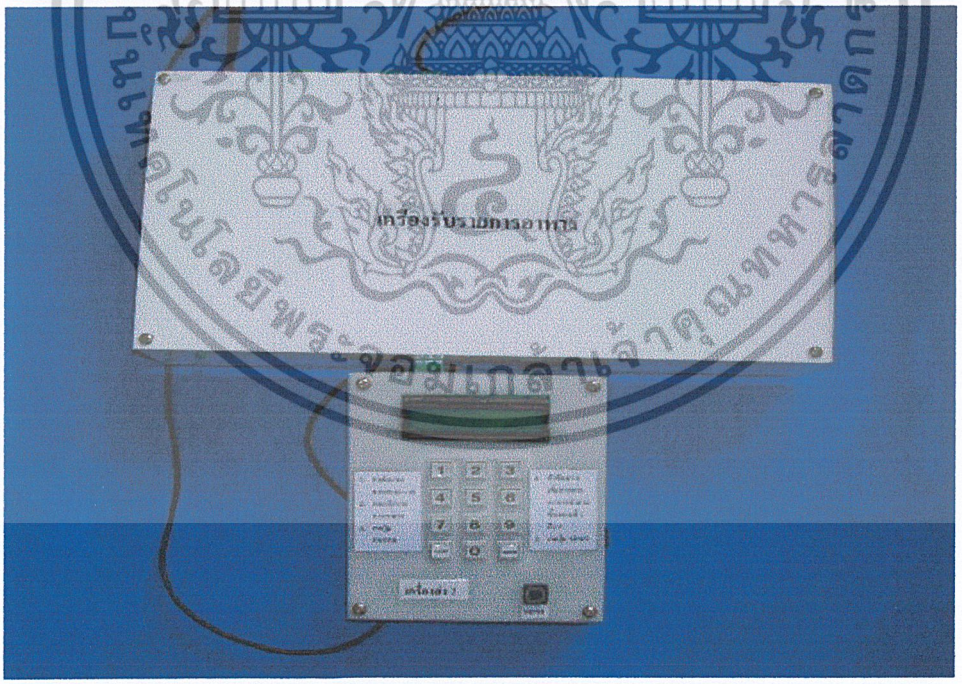


รูปที่ ก.2 ภาพด้านหน้าของเครื่องรับ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ ก.3 ภาพด้านข้างของเครื่องรับ



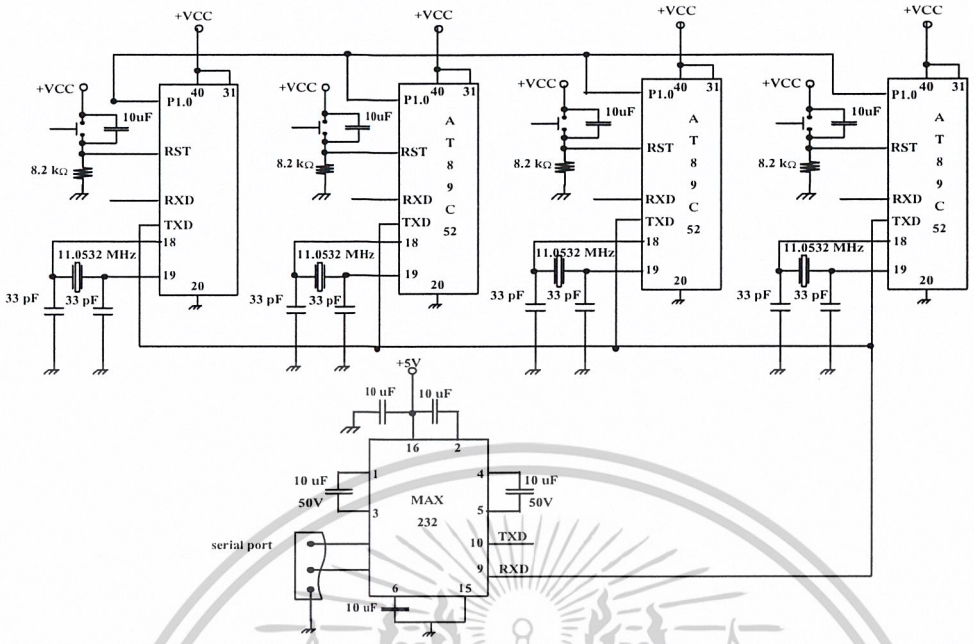
รูปที่ ก.4 ภาพการเชื่อมต่อระหว่างเครื่องส่งกับเครื่องรับ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

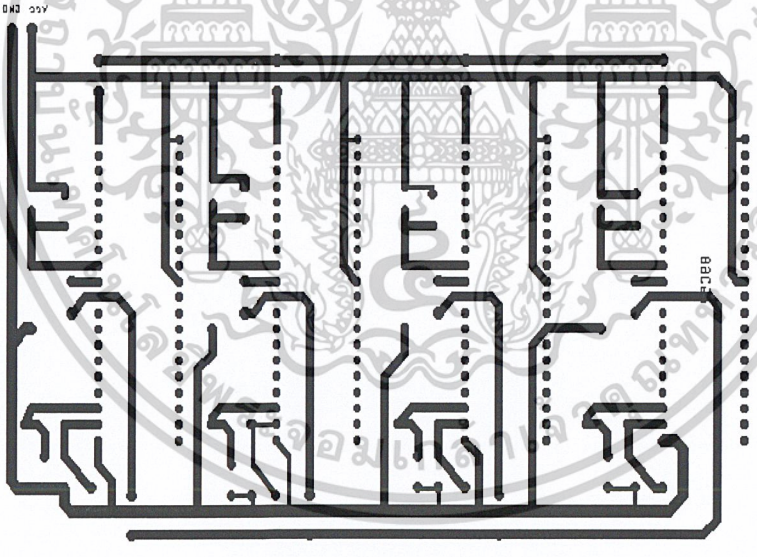
ภาคผนวก ข

วงจรและแผ่นวงจรพิมพ์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

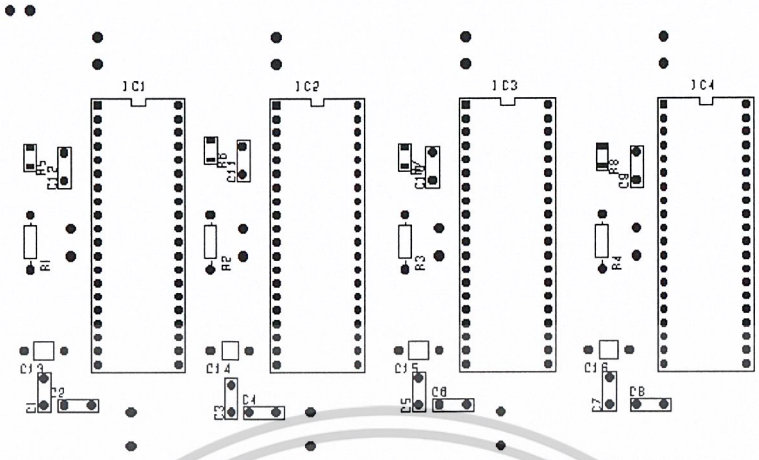


รูปที่ ข.1 วงจรเครื่องรับ

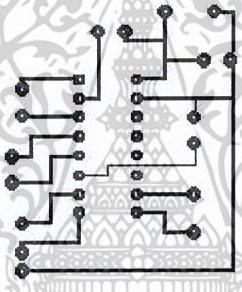


รูปที่ ข.2 แผ่นวงจรพิมพ์วงจรเครื่องรับ

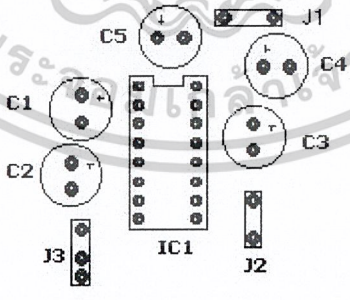
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ ข.3 ตำแหน่งการวางอุปกรณ์แผ่นวงจรพิมพ์วงจรเครื่องรับ

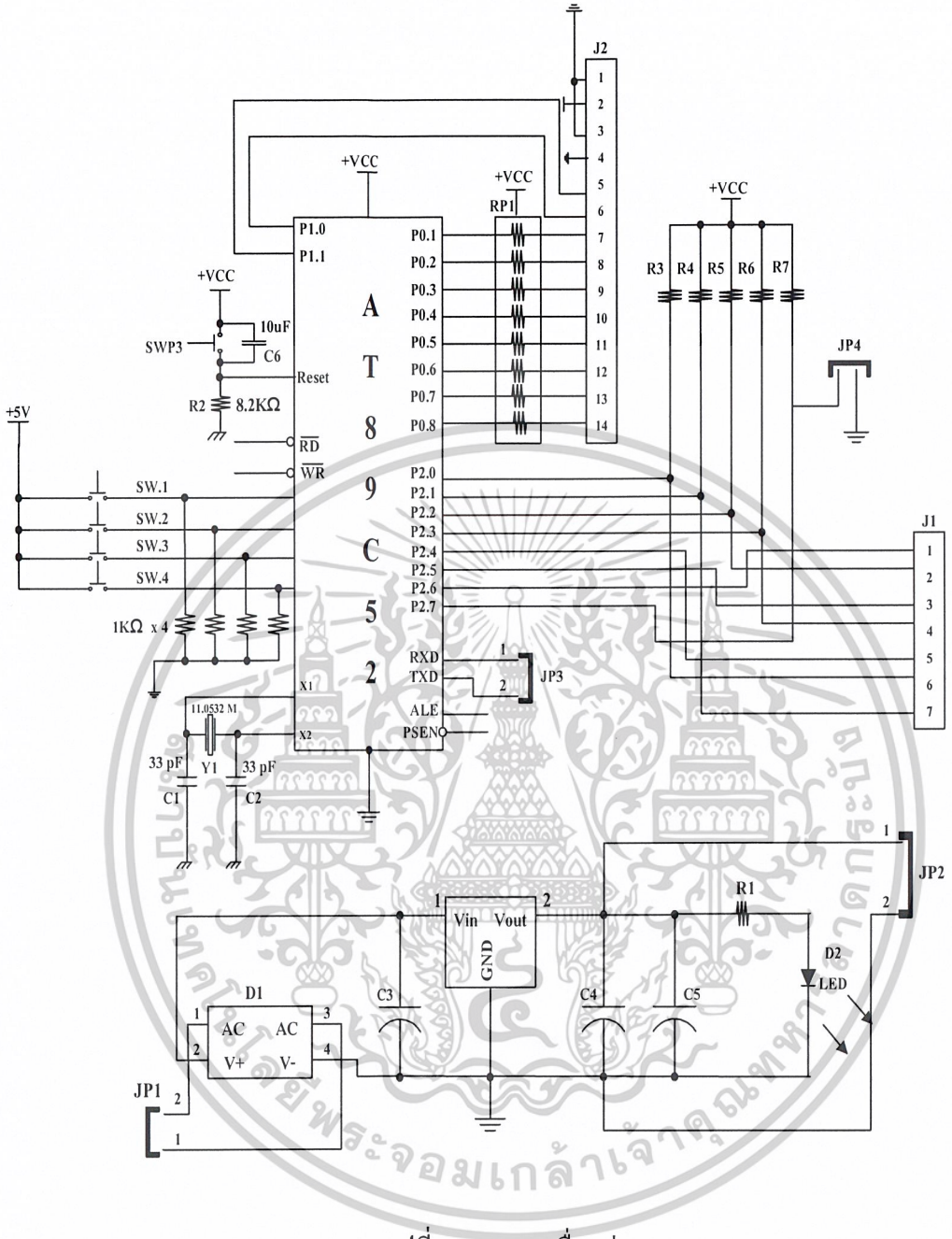


รูปที่ ข.4 แผ่นวงจรพิมพ์ของวงจร MAX 232



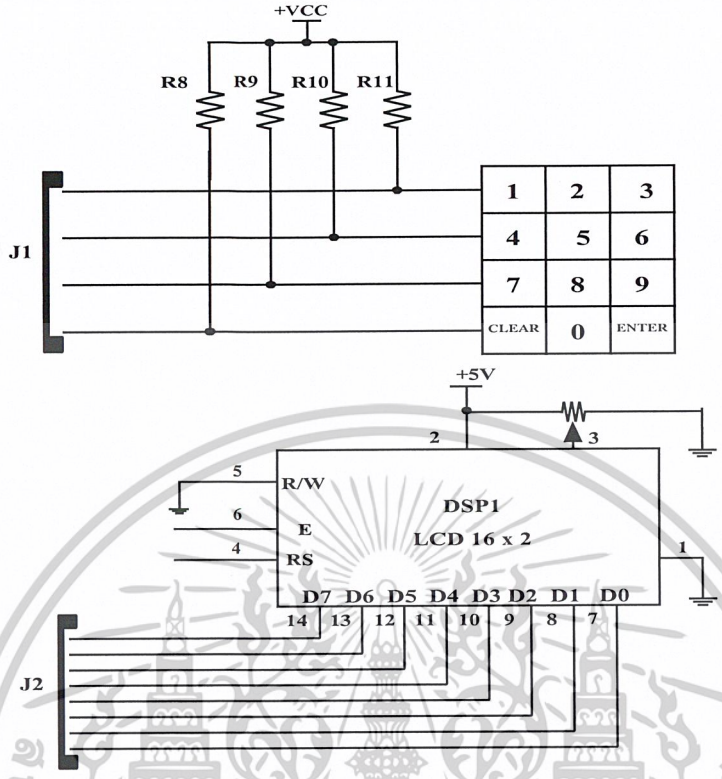
รูปที่ ข.5 ตำแหน่งการวางอุปกรณ์แผ่นวงจรพิมพ์ของวงจร MAX 232

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

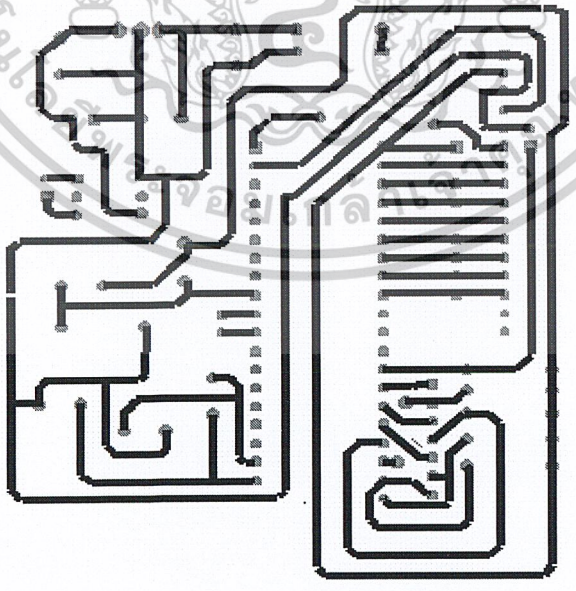


รูปที่ ข.6 วงจรเครื่องส่ง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

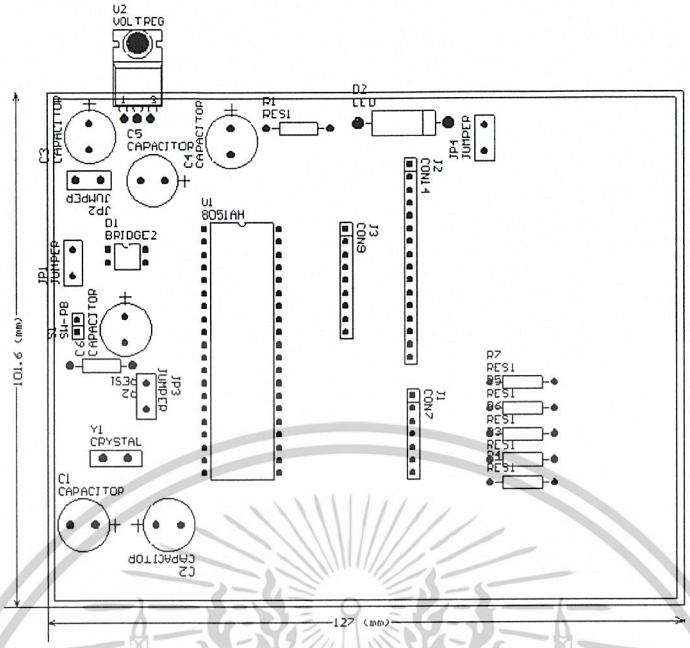


รูปที่ ข.7 วงจรส่วนการแสดงผล



รูปที่ ข.8 แผ่นวงจรพิมพ์วงจรเครื่องส่งและส่วนการแสดงผล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ ข.9 ตำแหน่งการวางอุปกรณ์แผ่นวงจรเครื่องส่งและส่วนการแสดงผล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก ค
รายการอุปกรณ์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ ค.1 รายการอุปกรณ์ของวงจรแสดงผล

ชื่ออุปกรณ์	รายละเอียด	จำนวน
วงจรรวม		
IC	AT89C52	4 ตัว
IC	7805	2 ตัว
อุปกรณ์สารกึ่งตัวนำ		
Crystal	11.0592 MHz	4 ตัว
Diode Bridge	BD104G	4 ตัว
LED	สีแดง	4 ตัว
ตัวเก็บประจุ		
C1,C2	33 PF 50V เซรามิก	8 ตัว
C3,C4	0.1 µF 50V เซรามิก	8 ตัว
C5	330 µF 25V อิเล็กโตรไลต์	4 ตัว
C6	10 µF 25V อิเล็กโตรไลต์	4 ตัว
C7	0.1 µF 50V ไมล่า	4 ตัว
ตัวความต้านทาน		
R1	1 KΩ 5% 1/4 W	8 ตัว
R2	8.2 KΩ 5% 1/4 W	4 ตัว
R3, R4, R5, R6, R7 (R- pack)	10 KΩ 5% 1/4 W	20 ตัว
R8	10 KΩ 5% 1/4 W	4 ตัว
อุปกรณ์อื่นๆ		
L1	120 H	1 ตัว
ซ็อกเก็ตไอซี	40 ขา	4 ตัว
แบตเตอรี่	9 V	4 ตัว
SW1	สวิตซ์กด	1 ตัว
จอLCD	มี back light	4 ตัว
แผ่นวงจรพิมพ์		4 แผ่น
สายไฟ		1 เส้น
สายแพ		1 เส้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ ค.1 (ต่อ) รายการอุปกรณ์ของวงจรแสดงผล

ชื่ออุปกรณ์	รายละเอียด	จำนวน
อุปกรณ์อื่นๆ		
Connector ตัวผู้	2 ขา แบบขาตรง	8 อัน
Connector ตัวผู้	4 ขา แบบขาตรง	4 อัน
Connector ตัวผู้	7 ขา แบบขาตรง	4 อัน
Connector ตัวผู้	14 ขา แบบขาตรง	4 อัน

ตารางที่ ค.2 รายการอุปกรณ์ของวงจรเครื่องส่ง

ชื่ออุปกรณ์	รายละเอียด	จำนวน
วงจรรวม		
IC เร็กกูเลเตอร์	7805	4 ตัว
อุปกรณ์สารกึ่งตัวนำ		
XTAL	11.0592 MHz	4 ตัว
Diode Bridge	BD104G	4 ตัว
LED	สีแดงและเขียว	4 ตัว
ตัวเก็บประจุ		
C1 – C2	0.1 µF ไมล่า	8 ตัว
C3 – C10	33 PF 25V เซรามิก	8 ตัว
C11 - C14	10 µF 25V อิเล็กโตรไลท์	4 ตัว
C15	330 µF 25V อิเล็กโตรไลท์	4 ตัว
ตัวความต้านทาน		
R1, R3, R4, R5	1 KΩ 5% 1/4 W	4 ตัว
R6, R7, R9, R10	1 KΩ 5% 1/4 W	2 ตัว
R (R- pack)	10 KΩ 5% 1/4 W	4 ตัว
R11, R12, R13, R14	8.2 KΩ 5% 1/4 W	4 ตัว
R15	330 Ω 5% 1/4 W	4 ตัว
R16 – R36	10 KΩ 5% 1/4 W	20 ตัว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ ค.2 (ต่อ) รายการอุปกรณ์ของวงจรเครื่องส่ง

ชื่ออุปกรณ์	รายละเอียด	จำนวน
อุปกรณ์อื่นๆ		
สายแพ	-	4 เส้น
SW	สวิตซ์กดติด – ปล่อยดับ ขนาดเล็ก	4 ตัว
คอนเน็กเตอร์ตัวเมีย	ขนาด 10 ขา	4 ตัว
คอนเน็กเตอร์ตัวเมีย	ขนาด 7 ขา	4 ตัว
คอนเน็กเตอร์ตัวเมีย	ขนาด 4 ขา	4 ตัว
คอนเน็กเตอร์ตัวเมีย	ขนาด 2 ขา	8 ตัว
คอนเน็กเตอร์ตัวผู้	ขนาด 10 ขา	4 ตัว
คอนเน็กเตอร์ตัวผู้	ขนาด 7 ขา	4 ตัว
คอนเน็กเตอร์ตัวผู้	ขนาด 4 ขา	4 ตัว
คอนเน็กเตอร์ตัวผู้	ขนาด 2 ขา	8 ตัว
คีย์แพด	ขนาด 3 x 4	4 ตัว
แบตเตอรี่	3 V	1 ก้อน
SW1,SW2	สวิตซ์กดติด – ปล่อยดับ ขนาดใหญ่	2 ตัว
จอLCD	มี back light 2 บรรทัด	4 ตัว
แผ่นวงจรพิมพ์	-	1 แผ่น
แบตเตอรี่	9 V	4 ก้อน
ซ็อกเก็ตไอซี	40 ขา	4 ตัว
เทอร์มินอล	3 ขา สีเขียว	4 ตัว
Dip Switch	สวิตซ์เลื่อน	4 ตัว
สวิตซ์	ON/OFF	4 ตัว
Terminal	3 อินพุต สีเขียว	16 ตัว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

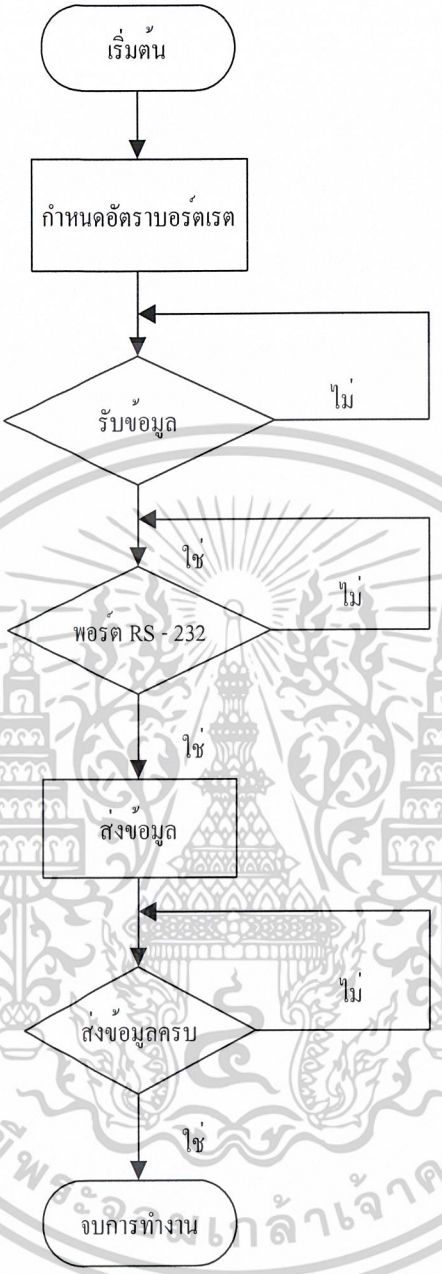
ตารางที่ ค.3 รายการอุปกรณ์ของวงจรเครื่องรับ

ชื่ออุปกรณ์	รายละเอียด	จำนวน
วงจรรวม		
IC	7805	1 ตัว
IC	AT89C52	16 ตัว
อุปกรณ์สารกึ่งตัวนำ		
Diode Bridge	BD104G	1 ตัว
LED	สีเขียว	1 ตัว
ตัวเก็บประจุ		
C6 – C11	10 µF 50V อิเล็กโตรไลท์	29 ตัว
C12 – C20	33 PF เซรามิก	32 ตัว
C21	1000 µF 25V	1 ตัว
ตัวความต้านทาน		
R1	100 KΩ 5% 1/4 W	1 ตัว
R2	390 Ω 5% 1/4 W	1 ตัว
R3	10 KΩ 5% 1/4 W	1 ตัว
R4,R5	470 Ω 5% 1/4 W	1 ตัว
R6	1 KΩ 5% 1/4 W	1 ตัว
อุปกรณ์อื่นๆ		
L1,L2	82 nH	2 ตัว
L3	22 nH	1 ตัว
SW1,SW2	สวิตช์กดขนาดเล็ก	16 ตัว
จอLCD	มี back light	1 เส้น
แผ่นวงจรพิมพ์	-	1 แผ่น
สายแพ	ขนาด 40 เส้น	1 เส้น
Socket IC	40 ขา	16 ตัว
หม้อแปลง	Input 220 VAC Output 9-0-9	1 ตัว
สายไฟ	AC 220	1 เส้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก ง
แผนผังการทำงานและรหัสต้นฉบับของโปรแกรม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ ง.1 ผังงานของโปรแกรมควบคุมการติดต่อกับพอร์ตอนุกรม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โปรแกรมควบคุมการการทำงานของเครื่องส่ง

```
LCD_EN          BIT         P1.0
LCD_RS          BIT         P1.1
LCD_ADDR        EQU         020H
LCD_DATA        EQU         021H
DATA            EQU         022H
SET_ADDR        EQU         023H
BUFFER          EQU         024H
BUFFER1         EQU         025H
COUNT           EQU         026H
CHECK           EQU         027H
ROW             EQU         028H
COUNT_SCAN      EQU         029H
COUNT_EDIT      EQU         02AH
FLAG            EQU         02BH
CODE            EQU         02CH

                ORG         0000H

                MOV         P0,#00H
                MOV         P1,#02H
                MOV         P2,#0FFH
                MOV         FLAG,#00H
                MOV         COUNT,#00H
                MOV         BUFFER,#30H
                MOV         CHECK,#00H
                MOV         COUNT_EDIT,#00H
                MOV         SET_ADDR,#030H
                CLR         LCD_EN
                CLR         LCD_RS
MAIN:           ACALL       INIT_LCD
LOOP:           MOV         LCD_ADDR,#00FH
                ACALL       SET_ADDR_LCD
                MOV         DPTR,#STRING_2
                ACALL       WRLINE_LCD
                MOV         R4,#25
                ACALL       LOOP_LCD_L_SHF
                ACALL       LCD_CLR
                MOV         LCD_ADDR,#000H
                ACALL       SET_ADDR_LCD
                MOV         DPTR,#STRING_3
                ACALL       WRLINE_LCD
                MOV         R4,#13
                ACALL       LOOP_LCD_L_SHF
                ACALL       DELAY_1s
                ACALL       DELAY_1s
                ACALL       DELAY_1s
                ACALL       LCD_CLR
                ACALL       DELAY_1s
                ACALL       LCD_CLR
LOOP_KEY:       ACALL       LCD_BLINK
                MOV         P2,#0FFH
SEND_DATA:      MOV         A,#0FFH
                INC         COUNT
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                    RET

SEND_DATA:          MOV         A,#0FFH
                    CJNE        A,FLAG,NO_ENTER_CODE
                    MOV         FLAG,#00H
                    LCALL       LCD_CLR
                    LCALL       PLEASE_WAIT
                    MOV         BUFFER1,#30H
                    MOV         TMOD,#020H
                    MOV         SCON,#042H
                    MOV         TH1,#-24
                    SETB        TR1

LOOP2:              MOV         R0,BUFFER1
                    MOV         A,@R0
                    ACALL       SEND_DATA1
                    INC         BUFFER1
                    MOV         A,BUFFER1
                    CJNE        A,BUFFER,LOOP2
                    MOV         A,#'E'
                    LCALL       SEND_DATA1
                    MOV         TMOD,#020H
                    MOV         SCON,#050H
                    MOV         TH1,#-24
                    SETB        TR1

LOOP3:              JNB         RI,$
                    CLR         RI
                    MOV         A,SBUF
                    CJNE        A,#'A',LOOP3
                    LCALL       SUCCESS
                    ACALL       LCD_CLR
                    MOV         COUNT,#00H
                    MOV         CHECK,#00H
                    MOV         BUFFER,#30H
                    LJMP        LOOP_KEY
SEND_DATA1:         JNB         TI,$
                    CLR         TI
                    MOV         SBUF,A
                    ACALL       DELAY_100ms
                    RET

NO_ENTER_CODE:      LCALL       LCD_CLR
                    MOV         LCD_ADDR,#000H
                    ACALL       SET_ADDR_LCD
                    MOV         DPTR,#STRING_1
                    ACALL       WRLINE_LCD
                    ACALL       DELAY_1s
                    ACALL       LCD_CLR
                    LJMP        LOOP_KEY
PLEASE_WAIT:        MOV         LCD_ADDR,#000H
                    MOV         DPTR,#STRING_5
                    ACALL       WRLINE_LCD
                    RET
SUCCESS:            MOV         LCD_ADDR,#000H
                    ACALL       SET_ADDR_LCD

                    MOV         A,#04H
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                    CJNE        A,CHECK,PRESS_KEY
                    LJMP        PRESS_KEY_1
PRESS_KEY:
                    MOV         ROW,#00H
                    CLR         P2.0
                    ACALL       SCAN_KEY
                    MOV         ROW,#03H
                    SETB        P2.0
                    CLR         P2.1
                    ACALL       SCAN_KEY
                    MOV         ROW,#06H
                    SETB        P2.1
                    CLR         P2.2
                    ACALL       SCAN_KEY
                    SETB        P2.2
                    CLR         P2.3
                    JNB         P2.5,NUMBER0
                    JNB         P2.6,CLEAR
                    SETB        P2.3
                    MOV         A,CHECK
                    CJNE        A,#00H,LOOP_KEY
                    JNB         P2.7,SEND_DATA
                    AJMP        LOOP_KEY

PRESS_KEY_1:        CLR         P2.3
                    JNB         P2.4,ENTER
                    JNB         P2.6,CLEAR
                    SETB        P2.3
                    LJMP        LOOP_KEY

NUMBER0:            MOV         LCD_DATA,#'0'
                    ACALL       WRCHAR_LCD
                    ACALL       DELAY_1s
                    ACALL       WRITE_REG
                    AJMP        LOOP_KEY

CLEAR:              MOV         A,#00H
                    CJNE        A,COUNT,CLEAR1
                    LJMP        CLEAR2
CLEAR1:             DEC         BUFFER
                    DJNZ        COUNT,CLEAR1

CLEAR2:             ACALL       LCD_CLR
                    MOV         COUNT,#00H
                    MOV         CHECK,#00H
                    LJMP        LOOP_KEY

ENTER:              LCALL       LCD_CLR
                    MOV         FLAG,#0FFH
                    MOV         COUNT,#00H
                    MOV         CHECK,#00H
                    LJMP        LOOP_KEY

WRITE_REG:          MOV         R0,BUFFER
                    MOV         @R0,LCD_DATA
                    INC         BUFFER
                    INC         CHECK
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                    MOV         DPTR,#STRING_4
                    ACALL       WRLINE_LCD
                    ACALL       DELAY_1s
                    ACALL       LCD_OFF
                    ACALL       DELAY_1s
                    ACALL       LCD_ON
                    ACALL       DELAY_1s
                    ACALL       LCD_OFF
                    ACALL       DELAY_1s
                    ACALL       LCD_ON
                    ACALL       DELAY_1s
                    RET

LOOP_LCD_L_SHF:     ACALL       LCD_LSHF
                    ACALL       DELAY_100ms
                    ACALL       DELAY_100ms
                    DJNZ        R4,LOOP_LCD_L_SHF
                    RET

SCAN_KEY:           MOV         COUNT_SCAN,#02H
                    JNB         P2.4,PRESS
                    DEC         COUNT_SCAN
                    JNB         P2.5,PRESS
                    DEC         COUNT_SCAN
                    JNB         P2.6,PRESS
                    RET

PRESS:              MOV         A,COUNT_SCAN
                    ADD         A,ROW
                    LCALL       SHOW_LCD
                    RET

SHOW_LCD:           MOV         DPTR,#STRING_6
                    MOVC        A,@A+DPTR
                    MOV         LCD_DATA,A
                    ACALL       WRCHAR_LCD
                    ACALL       DELAY_400ms
                    ACALL       WRITE_REG
                    RET
;LCD Initialize
INIT_LCD:           ACALL       DELAY_100ms         ; Delay
                    CLR         LCD_RS              ; Clear LCD_RS Pin
                    MOV         P0,#00111000B       ; 8 Bit Mode
                    ACALL       LCD_CLK             ; Pulse LCD Clock
                    ACALL       DELAY_10ms          ; Delay
                    MOV         P0,#00111000B       ; 8bit Mode
                    ACALL       LCD_CLK             ; Pulse LCD Clock
                    ACALL       LCD_OFF             ; Display Off
                    ACALL       LCD_CLR             ; Clear Display
                    MOV         P0,#00000110B       ; Entry Mode
                    ACALL       LCD_CLK             ; Pulse LCD Clock
                    ACALL       LCD_HOME            ; Return Home
                                                      Display

; LCD Clear Display
LCD_CLR:            CLR         LCD_RS              ; Clear LCD_RS Pin
                    MOV         P0,#00000001B       ; Display Clear
                    ACALL       LCD_CLK             ; Pulse LCD Clock
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                    RET
;LCD Return Home
LCD_HOME:           CLR         LCD_RS              ; Clear LCD_RS Pin
                    MOV         P0,#00000010B       ; Return Home
                    ACALL       LCD_CLK             ; Pulse LCD Clock
                    RET

; LCD Display Off
LCD_OFF:            CLR         LCD_RS              ; Clear LCD_RS Pin
                    MOV         P0,#00001000B       ; Display Off
                    ACALL       LCD_CLK             ; Pulse LCD Clock
                    RET
; LCD Clk
LCD_CLK:            SETB        LCD_EN              ; Pulse Clock to
LCD_EN
                    ACALL       LCD_DELAY
                    CLR         LCD_EN
                    ACALL       LCD_DELAY
                    RET
; LCD Display On
LCD_ON:             CLR         LCD_R               ; Clear LCD_RS Pin
                    MOV         P0,#00001100B       ; Display On
                    ACALL       LCD_CLK             ; Pulse LCD Clock
                    RET
; LCD Cursor On
LCD_BLINK:          CLR         LCD_R               ; Clear LCD_RS Pin
                    MOV         P0,#00001111B       ; Display Cursor
and Blink
                    ACALL       LCD_CLK             ; Pulse LCD Clock
                    RET
; LCD Left Shift Display
LCD_LSHF:           CLR         LCD_RS              ; Clear LCD_RS Pin
                    MOV         P0,#00011000B       ; Left Shift Display
                    ACALL       LCD_CLK             ; Pulse LCD Clock
                    RET

; LCD Right Shift Display
LCD_RSHF:           CLR         LCD_RS              ; Clear LCD_RS Pin
                    MOV         P0,#00011100B       ; Right Shift
Display
                    ACALL       LCD_CLK             ; Pulse LCD Clock
                    RET

;Set LCD Address
;I/P:               LCD_ADDR
SET_ADDR_LCD:       CLR         LCD_RS      ; Clear LCD_RS Pin
                    MOV         A,LCD_ADDR  ; Move LCD_ADDR to ACC.
                    SETB        ACC.7       ; Set bit ACC.7
                    MOV         P0,A        ; Move to DATABUS
                    ACALL       LCD_CLK     ; Pulse LCD Clock
                    RET

;Write Character To Register
SEND_TO_REG:        MOV         R0,SET_ADDR

; Program           : ENTER DATA
; Description       : Key Matrix And LCD Display Show
; Define Port&Pin Name
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                    MOV         @R0,DATA
                    INC         SET_ADDR
                    RET

; Write Character to show LCD
; I/P:              LCD_DATA
WRCHAR_LCD:         SETB        LCD_RS      ; Set LCD_RS Pin
                    MOV         P0,LCD_DATA ; Move LCD_DATA to
DATABUS
                    ACALL       LCD_CLK     ; Pulse LCD Clock
                    ACALL       LCD_ON      ; Display On
                    RET

; Write Line of 8 Character from ROM
; I/P:              DPTR : Locate ROM Address
WRLINE_LCD:         MOV         R1,#0       ; Clear loop counter
WRLINE_LCD_1        SETB        LCD_RS      ; Set LCD_RS Pin
                    CLR         A           ; Clear ACC.
                    MOVC        A,@A+DPTR
; Move data from @DPTR to ACC.
                    MOV         P0,A        ; Move ACC. to DATABUS
                    ACALL       LCD_CLK     ; Pulse LCD Clock
                    INC         DPTR        ; Increase Pointer
                    INC         R1          ; Increase loop counter
                    CJNE        R1,#28,WRLINE_LCD_1
; Do until 8 times
                    ACALL       LCD_ON      ; Display On
                    RET

; Dummy Delay time LCD_DELAY, 10m, 100m, 1s
LCD_DELAY:          MOV         R7,#003     ; Do 2 times
LCD_DELAY_1:        MOV         R6,#0E6H    ; Each loop = 1 ms
LCD_DELAY_2:        NOP
                    NOP
                    DJNZ        R6,LCD_DELAY_2
                    DJNZ        R7,LCD_DELAY_1
                    RET

DELAY_10ms:         MOV         R7,#010     ; Do 10 times
DELAY_10ms_1:       MOV         R6,#0E6H    ; Each loop = 1 ms
DELAY_10ms_2:       NOP
                    NOP
                    DJNZ        R6,DELAY_10ms_2
                    DJNZ        R7,DELAY_10ms_1
                    RET

DELAY_100ms:        MOV         R7,#100     ; Do 100 times
DELAY_100ms_1:      MOV         R6,#0E6H    ; Each loop = 1 ms
DELAY_100ms_2:      NOP
                    NOP
                    DJNZ        R6,DELAY_100ms_2
                    DJNZ        R7,DELAY_100ms_1
                    RET

                    DJNZ        R6,DELAY_1s_1
                    RET
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
DELAY_400ms:        MOV         R6,#40      ; Do 100 times
DELAY_400ms_1:      ACALL       DELAY_10ms
                    DJNZ        R6,DELAY_400ms_1
                    RET

DELAY_1s:           MOV         R6,#100     ; Do 100 times
DELAY_1s_1:         ACALL       DELAY_10ms
;Define Constant < Store in Flash EEPROM Program Memory >
;01234567
STRING_1:           DB          'NO Code!'
STRING_2:           DB          'AUTOMATIC WIRELESS FOOD O'
STRING_3:           DB          'WIRELESS FOOD ORDER MACHINES'
STRING_4:           DB          'Send Success'
STRING_5:           DB          'Please Wait!'
STRING_6:           DB          '12346789'
                    END
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โปรแกรมควบคุมการรับข้อมูล

```
; Define User Register
LCD_ADDR        EQU     020H        ; For keep LCD Address
LCD_DATA        EQU     021H        ; For keep LCD Data
DATA            EQU     022H
SET_ADDR        EQU     023H
BUFFER          EQU     024H
BUFFER1         EQU     027H
COUNT           EQU     025H
CHECK           EQU     026H
TABLE           EQU     027H
; Main Program.

                ORG     0000H       ; Reset Vector
START:          MOV     P2,#00H
                MOV     P0,#00H
                MOV     P1,#0FFH

LOOP1:          MOV     P1,#0FFH
                MOV     COUNT,#00H
                MOV     BUFFER,#30H
                MOV     TMOD,#020H  ; T1 8 Bit Auto,T0 16 Bit
                MOV     SCON,#050H  ; Mode1 RX Enable
                MOV     TH1,#-24
                SETB    TR1         ; Start Timer1
RECEIVE_DATA:   JNB     RI,RECEIVE_DATA
                CLR     RI
                MOV     A,SBUF
                MOV     R0,BUFFER
                MOV     @R0,A
                CJNE    A,#'E',RECEIVE_DATA1

LOOP:           JB      P1.0,SEND_DATA
                LCALL   DELAY_1s
                LCALL   DELAY_1s
                LCALL   DELAY_1s
                LJMP    LOOP
RECEIVE_DATA1:  INC     BUFFER
                LJMP    RECEIVE_DATA

SEND_DATA:      MOV     P1,#00000000B
                MOV     BUFFER,#30H
                MOV     TABLE,#00H
                MOV     TMOD,#020H  ; T1 8 Bit Auto,T0 16 Bit
                MOV     SCON,#042H  ; Mode1 RX Disable
                MOV     TH1,#-24
                SETB    TR1         ; Start Timer1
                MOV     R0,BUFFER
SND_CODE        MOV     A,#'0'
                JNB     TI,$
                CLR     TI
                MOV     SBUF,A
                MOV     A,#'1'
                JNB     TI,$
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                    CLR         TI
                    MOV         SBUF,A
SND_DATA1:          MOV         A,@R0
                    JNB         TI,$
                    CLR         TI
                    MOV         SBUF,A
                    INC         BUFFER
                    MOV         R0,BUFFER
                    MOV         A,@R0
                    CJNE        A,#'E',CHECK_CODE
                    MOV         A,#'E'
                    JNB         TI,$
                    CLR         TI
                    MOV         SBUF,A
                    LCALL       DELAY_1s
                    MOV         A,#'A'
                    JNB         TI,$
                    CLR         TI
                    MOV         SBUF,A
                    LCALL       DELAY_1s
                    LCALL       DELAY_1s
                    LCALL       DELAY_1s
                    LCALL       DELAY_1s
                    LCALL       DELAY_1s
                    LCALL       DELAY_1s
                    LCALL       DELAY_1s
                    LCALL       DELAY_1s
                    LCALL       DELAY_1s
                    LCALL       DELAY_1s
                    LCALL       DELAY_1s
                    LCALL       DELAY_1s
                    LCALL       DELAY_1s
                    LCALL       DELAY_1s
                    LCALL       DELAY_1s
                    LJMP        LOOP1
CHECK_CODE:         INC         TABLE
                    MOV         A,TABLE
                    CJNE        A,#04H,SND_DATA1
                    MOV         TABLE,#00H
                    MOV         A,#'E'
                    JNB         TI,$
                    CLR         TI
                    MOV         SBUF,A
                    LCALL       DELAY_1s
                    LCALL       DELAY_1s
                    LCALL       DELAY_1s
                    LJMP        SND_CODE

; Dummy Delay time LCD_DELAY, 10m, 100m, 1s
LCD_DELAY:          MOV         R7,#002     ; Do 2 times
LCD_DELAY_1:        MOV         R6,#0E6H    ; Each loop = 1 ms
LCD_DELAY_2:        NOP
                    NOP
                    DJNZ        R6,LCD_DELAY_2
                    DJNZ        R7,LCD_DELAY_1
                    RET
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
DELAY_10ms:     MOV     R7,#010H    ; Do 10 times
DELAY_10ms_1:   MOV     R6,#0E6H    ; Each loop = 1 ms
DELAY_10ms_2:   NOP
                NOP
                DJNZ    R6,DELAY_10ms_2
                DJNZ    R7,DELAY_10ms_1
                RET

DELAY_100ms:    MOV     R7,#100     ; Do 100 times
DELAY_100ms_1:  MOV     R6,#0E6H    ; Each loop = 1 ms
DELAY_100ms_2:  NOP
                NOP
                DJNZ    R6,DELAY_100ms_2
                DJNZ    R7,DELAY_100ms_1
                RET
DELAY_1s:       MOV     R5,#070     ; Do 100 times
DELAY_1s_1:     ACALL   DELAY_10ms
                DJNZ    R5,DELAY_1s_1
                RET
                END
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โปรแกรมฟอร์มรับรายการอาหาร

```
Option Explicit
Private Sub Command12_Click(Index As Integer)
Text1.Text = ""
Text1.SetFocus
End Sub
Private Sub Form_Load( )
Command3.Enabled = False
MSComm1.CommPort = 1
MSComm1.Settings = "1200,n,8,1"
If Not MSComm1.PortOpen Then
MSComm1.PortOpen = True
End If
Connect
Call ListTable
End Sub

Private Sub Command3_Click( )
Dim SQL As String
  If MsgBox ("คุณต้องการลบข้อมูล ใช่หรือไม่?"), vbYesNo + vbQuestion +
vbDefaultButton2, "คำยืนยัน") = vbYes Then
Beep
SQL = "delete from order1 where Id =" & CInt(ChList)
DB.Execute (SQL)
DB.Close
Connect
List1.Clear
Call ListTable
Command3.Enabled = False
Text1.SetFocus
Text1.Text = ""
Else: Command3.Enabled = False
Text1.SetFocus
Text1.Text = ""
End If

End Sub

Private Sub List1_Click( )
Dim choose As Integer
Command3.Enabled = True
ChList = List1.List(choose)
Text1.SetFocus
Text1.Text = ""
End Sub

Private Sub ListView1_GotFocus( )
Text1.Text = ""
Text1.SetFocus
End Sub
Private Sub Timer1_Timer()
If MSComm1.InBufferCount Then
'Text1.Text = Text1.Text & Chr(Asc(MSComm1.Input))
Text1.Text = Text1.Text & MSComm1.Input
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
Food = Text1.Text
If Len(Food) = 7 Then
Call Getdata
End If
End If
End Sub

Private Sub Getdata( )
jcount = Len(Food)
ReDim strFood(jcount)
For i = 0 To jcount - 1
strFood(i) = Left(Food, i + 1)
strFood(i) = Right(strFood(i), 1)
Next
strTable = strFood(0) & strFood(1)
strCount = strFood(2)
For i = 3 To jcount - 2
strMenu = strMenu & strFood(i)
Next
Text1.Text = ""
Food = ""
Call Data
End Sub

Private Sub Data()
    Set RS = New ADODB.Recordset
    RS.Open "Select * from menu where Id=" & CInt(strMenu), DB,
adOpenStatic, adLockOptimistic
    If Not RS.EOF Then
    RS.MoveFirst
    For i = 1 To RS.RecordCount
ListView1.ListItems.Add i, , strTable
ListView1.ListItems(i).ListSubItems.Add 1, , RS.Fields("name")
ListView1.ListItems(i).ListSubItems.Add 2, , strCount
RS.MoveNext
    Next
    End If
    ListView1.SetFocus
    RS.Close
    '- - - - - - - - -- - - - - -- -
    On Error GoTo 0
    Dim SQL As String
    SQL = "Insert into Order1 values(" & strTable & "," & strMenu &
"," & strCount & ",0)"
    DB.Execute (SQL)
    strMenu = ""
End Sub

Private Sub ListTable()
    Set RS = New ADODB.Recordset
    RS.Open "Select DISTINCT ID from order1", DB, adOpenStatic,
adLockOptimistic
    If Not RS.EOF Then
    RS.MoveFirst
    For i = 0 To RS.RecordCount - 1
    List1.AddItem RS.Fields("ID"), i
    RS.MoveNext
    List1.Refresh
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    Next
    End If
    RS.Close
End Sub

Private Sub CmdExit_Click()
  If MsgBox("คุณต้องการออกจากหน้านี้ ใช่หรือไม่?", vbYesNo + vbQuestion +
vbDefaultButton2, "คำยืนยัน") = vbYes Then
            Beep
            Unload Me
Text1.Text = ""
Text1.SetFocus
Else:
Text1.Text = ""
Text1.SetFocus
             End If
End Sub
Private Sub Command1_Click()
  If MsgBox("คุณต้องการลบข้อมูล ใช่หรือไม่?", vbYesNo + vbQuestion +
vbDefaultButton2, "คำยืนยัน") = vbYes Then
Beep
ListView1.ListItems.Clear
Text1.Text = ""
Text1.SetFocus
Else
Text1.Text = ""
Text1.SetFocus
End If
End Sub
Option Explicit
Public DB As ADODB.Connection
Public RS As ADODB.Recordset
Public RS1 As ADODB.Recordset
Public Food, strFood(), strTable, strCount, strMenu As String
Public i, j, jcount As Integer
Public ChList As String
Public Sub Connect()
Set DB = New ADODB.Connection
DB.ConnectionString = "Provider=Microsoft.Jet.OLEDB.4.0;" & _
"Persist Security Info=False;Data Source=" & App.Path &
"\food.mdb"
DB.Open
End Sub
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โปรแกรมฟอร์มเพิ่มรายการและค้นหารายการอาหาร

```
Option Explicit
Private Sub Adodc1_WillMove(ByVal adReason As
ADODB.EventReasonEnum, adStatus As ADODB.EventStatusEnum, ByVal
pRecordset As ADODB.Recordset)
Dim CurrentRecord As Integer
Dim AllRecord As String
With Adodc1.Recordset
CurrentRecord = .AbsolutePosition
AllRecord = .RecordCount
End With
Adodc1.Caption = "รายการที่" & CurrentRecord & "/" & AllRecord

CmdAdd.Enabled = True
CmdOK.Visible = False
CmdCancel.Visible = False
End Sub

Private Sub CmdAdd_Click ( )
Adodc1.Recordset.AddNew
CmdAdd.Enabled = False
Adodc1.Enabled = False
CmdOK.Visible = True
CmdCancel.Visible = True
End Sub

Private Sub CmdCancel_Click ( )
Adodc1.Recordset.CancelUpdate
CmdAdd.Enabled = True
CmdOK.Visible = False
CmdCancel.Visible = False
Adodc1.Enabled = True
End Sub

Private Sub CmdOK_Click( )
If (Text1.Text = "") Or (Text2.Text = "") Or (Text3.Text = "") Then
MsgBox "คุณยังใส่ข้อมูลไม่ครบ!", vbOKOnly, "ข้อมูลผิดพลาด"
Else
Adodc1.Enabled = True
End If
End Sub

Private Sub CmdExit_Click( )
  If MsgBox("คุณต้องการออกจากหน้านี้ ใช่หรือไม่?", vbYesNo + vbQuestion +
vbDefaultButton2, "คำยืนยัน") = vbYes Then
             Beep
             Unload Me
             End If
End Sub

Private Sub CmdSearch_Click( )
Dim userCriteria As String
If (Text4.Enabled = True) And (Text4 <> "") Then
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
userCriteria = " ID like '" & Text4.Text & "'"
     With Adodc1.Recordset
          .MoveFirst
          .Find userCriteria, adSearchForward
             Beep
     End With
          Text4.Enabled = True
          Text4.Text = ""
    Else
          MsgBox "คุณยังไม่ได้ใส่ข้อมูล!", vbYes, "คำเตือน"
           Text4.Enabled = True
      End If
If Adodc1.Recordset.EOF Then
MsgBox "ไม่พบข้อมูล", vbYes, "ผลการค้นหา"
End If
End Sub

Private Sub Form_Load( )
CmdOK.Visible = False
CmdCancel.Visible = False
End Sub

Private Sub Text3_KeyPress (KeyAscii As Integer)
If KeyAscii = 13 Then
Call CmdOK_Click
End If
End Sub

Private Sub Text4_Keypress (KeyAscii As Integer)
If KeyAscii = 13 Then
Call CmdSearch_Click
End If
End Sub
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โปรแกรมฟอร์มพิมพ์ใบเสร็จรายการอาหาร

```
Option Explicit

Private Sub CmdExit_Click ( )
If MsgBox ("คุณต้องการออกจากหน้านี้ ใช่หรือไม่? ", vbYesNo + vbQuestion +
vbDefaultButton2, "คำยืนยัน") = vbYes Then
      Beep
      Unload Me
End If
End Sub

Private Sub Command4_Click( )
On Error GoTo ProError
Dim str As String
str = "{Order1.ID}= " & txtBegin.Text
str = str & "and {Order1.Check}= " & False
CrystalReport1.SelectionFormula = str
CrystalReport1.ReportFileName = "C:\VBPort1\food.rpt"
CrystalReport1.Action = 1
txtBegin.SetFocus
ProExit:
Exit Sub
ProError:
MsgBox Err.Number & vbCrLf & Err.Description
Resume ProExit
txtBegin.SetFocus
End Sub

Private Sub txtBegin_KeyPress (KeyAscii As Integer)
If KeyAscii = 13 Then
Call Command4_Click
End If End Sub
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โปรแกรมฟอร์มหลัก

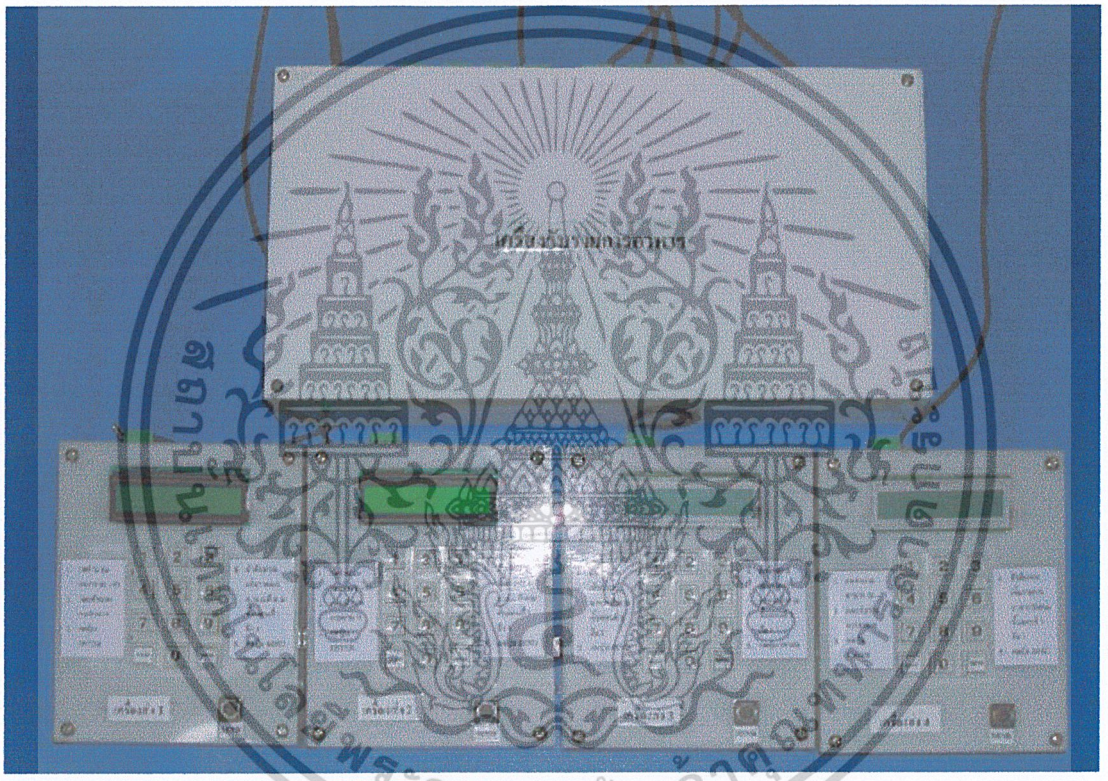
```
option Explicit

Private Sub MDIForm_UnLoad(Cancel As Integer)
If MsgBox("จบการทำงาน", vbOKOnly + vbQuestion, "คำยืนยัน") = vbOK Then
       Beep
       Unload Me
End If
End Sub

Private Sub Toolbar1_ButtonClick(ByVal Button As
MSComctlLib.Button)
Select Case Button.Key
Case "End"
        Unload Form1
        Unload Form2
        Unload Form3
        If MsgBox("คุณต้องการออกจากโปรแกรมนี้ ใช่หรือไม่? ", vbYesNo + vbQuestion
+ vbDefaultButton2, "คำยืนยัน") = vbYes Then
        End
        End If
Case "Add"
        Form2.Show
        Form2.SetFocus
        Unload Form1
        Unload Form3
Case "Open"
        Form1.Show
        Form1.SetFocus
        Unload Form2
        Unload Form3
Case "Print"
        Form3.Show
        Form3.SetFocus
        Unload Form1
        Unload Form2
        End Select
End Sub
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก จ
คู่มือการใช้งาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

คู่มือการใช้งาน
เครื่องสั่งอาหารระบบอัตโนมัติ



ภาควิชาครุศาสตร์วิศวกรรม

คณะครุศาสตร์อุตสาหกรรม

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

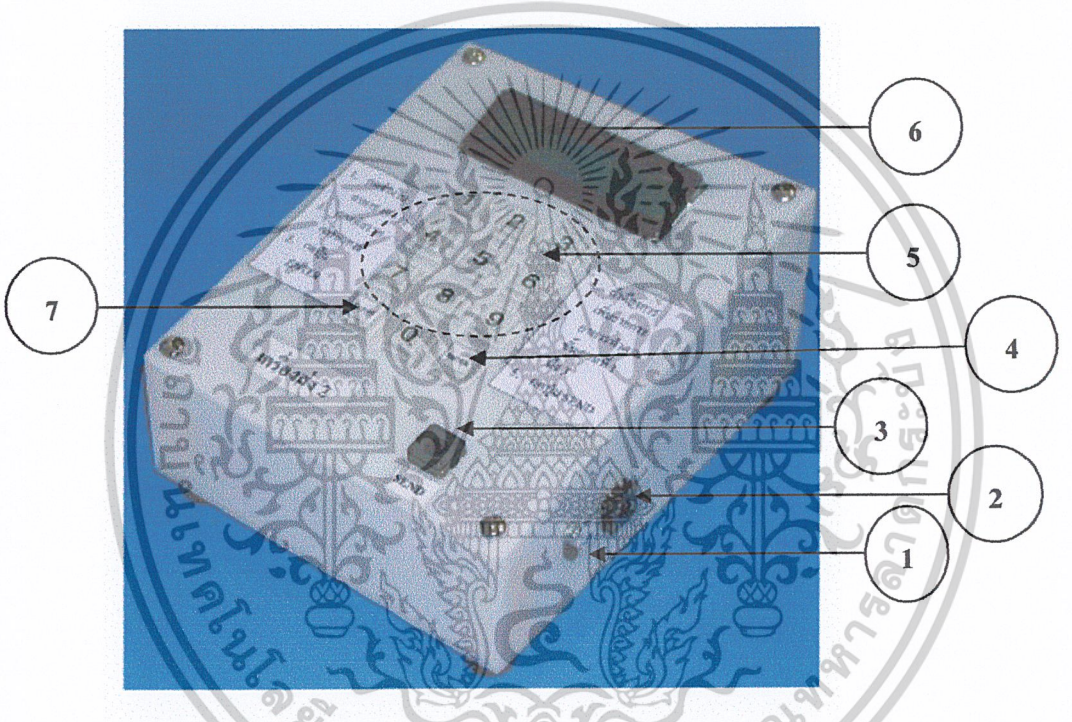
ปีการศึกษา 2546

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1. คำแนะนำเบื้องต้น

ก่อนที่จะลงมือใช้งานเครื่องสั่งอาหารระบบอัตโนมัติ ควรทำการศึกษาการใช้งานจากคู่มือให้เข้าใจเพื่อการใช้งานที่ถูกต้อง เพื่อเป็นการป้องกันการเสียหายที่อาจเกิดขึ้นกับเครื่องสั่งอาหารระบบอัตโนมัติ

2. ส่วนประกอบและปุ่มควบคุม



รูปที่ จ.1 ส่วนประกอบและปุ่มควบคุมของเครื่องสั่งอาหารระบบอัตโนมัติ

จากรูปที่ จ.1 มีรายละเอียดดังต่อไปนี้

หมายเลขที่ 1 สวิตช์เปิด–ปิดเครื่อง

หมายเลขที่ 2 สวิตช์รีเซ็ต

หมายเลขที่ 3 ปุ่ม SEND

หมายเลขที่ 4 ปุ่ม ENTER

หมายเลขที่ 5 ปุ่มกดเลือกรหัสอาหาร

หมายเลขที่ 6 จอแสดงผล

หมายเลขที่ 7 ปุ่ม CLEAR

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3. การติดตั้งและใช้งาน

3.1 ต่อเครื่องส่งเข้ากับเครื่องรับและต่อเข้ากับเครื่องคอมพิวเตอร์ก่อนใช้งาน

3.2 เลื่อนสวิตซ์ ON/OFF ไปที่ตำแหน่ง ON

3.3 รอให้หน้าจอเป็นเคอร์เซอร์

3.4 เลือกเมนูอาหารตามที่ต้องการแล้วจำรหัสรายการอาหารที่ต้องการสั่งไว้

3.5 กดจำนวนอาหารที่ท่านสั่ง ซึ่งจำนวนอาหารที่ท่านสั่งจะเป็นรหัสหนึ่งหลัก เช่น 1, 2, 3 เป็นต้น โดยเมื่อทำการกดจำนวนอาหารแล้วจะต้องกดรหัสอาหารต่อท้ายด้วย

3.6 กดรหัสอาหารตามที่ท่านต้องการ โดยรหัสอาหารนั้นก็จะเป็นรหัสจำนวนสามหลัก เช่น 100, 101, 102 เป็นต้น

3.7 โดยรหัสที่ท่านต้องกดก็จะเป็นรหัสสี่หลักด้วยกัน ตัวอย่างเช่น 1125 แสดงว่าท่านสั่งอาหารจำนวน 1 ที่ โดยอาหารที่ท่านสั่งคือรหัส 125 เป็นต้น

3.8 เมื่อกดรหัสครบสี่หลักแล้วให้ทำการกดปุ่ม Enter จากนั้นให้กดรหัสรายการอาหารตามที่ท่านต้องการต่อไปแล้วกดปุ่ม Enter จนกว่าจะครบจำนวนที่ท่านต้องการสั่ง

3.9 เมื่อทำการกดรหัสรายการอาหารต่างๆ เป็นที่เรียบร้อยแล้วก็ทำการปุ่ม Send เพื่อส่งข้อมูลไปยังคอมพิวเตอร์ที่แคชเชียร์

3.10 หน้าจอจะแสดงข้อความ Please Wait เพื่อแสดงให้รู้ว่ากำลังรอการส่งรายการอาหาร

3.11 เมื่อข้อมูลส่งไปยังฐานข้อมูลหน้าจอจะแสดงข้อความ Send Success เพื่อแสดงให้รู้ว่าขั้นตอนการสั่งรายการอาหารเสร็จสิ้นแล้ว

3.12 กรณีที่ไม่ต้องการรายการอาหารนั้นให้กดปุ่ม Clear

3.13 กรณีที่ไม่ต้องการรายการอาหารนั้นแต่ได้กดปุ่ม Enter แล้ว ให้กดปุ่ม Clear เช่นกัน

3.14 เมื่อเสร็จสิ้นการสั่งรายการอาหารให้เลื่อนสวิตซ์ ON/OFF ไปที่ตำแหน่ง OFF

4. การแก้ปัญหาเบื้องต้น

เมื่อประสบปัญหาในการใช้งานเครื่องสั่งอาหารระบบอัตโนมัติ สามารถตรวจสอบแนวทางไขปัญหาเบื้องต้นได้ จากตารางข้างล่างนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

อาการ	สาเหตุและ/หรือวิธีแก้ไข
จอไม่ติด, ไม่มีข้อความวิ่ง,ไม่มีเคอร์เซอร์กระพริบ	ตรวจสอบแบตเตอรี่ 9 โวลต์,ไม่ได้เปิดสวิตซ์เครื่องส่ง
บนหน้าจอแสดงไม่แสดงข้อความ Send Success หลังจากที่กดปุ่ม Send	ตรวจสอบสวิตซ์ที่เครื่องรับ, ตรวจสอบสายที่เชื่อมต่อระหว่างเครื่องส่งกับเครื่องรับ, ไม่เสียบปลั๊กเครื่องรับ
บนหน้าจอแสดงแสดงข้อความ Please Wait เป็นเวลานาน หลังจากที่กดปุ่ม Send	ตรวจสอบข้อมูลที่แคซเซียร์ว่าข้อมูลถูกส่งไปหรือยัง ถ้ายังไม่ได้ส่งให้กดปุ่ม Reset และป้อนข้อมูลส่งไปอีกครั้ง

5. การดูแลและข้อควรระวัง

5.1 การดูแลรักษา

- เลื่อนสวิตซ์ไปที่ OFF ทุกครั้งหลังจากใช้งานเสร็จ
- หมั่นตรวจสภาพแบตเตอรี่เครื่องส่งให้พร้อมใช้งานอยู่เสมอ
- ตรวจสอบสายที่ใช้ในการเชื่อมต่อระหว่างเครื่องส่งกับเครื่องรับก่อนใช้งาน

5.2 ข้อควรระวัง

- ควรระมัดระวังเครื่องส่งที่วางอยู่บนโต๊ะอาหารอาจเกิดอุบัติเหตุจากการรับประทานอาหาร
- เมื่อไม่ใช้งานเครื่องรับควรถอดปลั๊กออกทุกครั้ง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

6. ข้อมูลจำเพาะ

<u>เครื่องส่ง</u>

คุณสมบัติ	รายละเอียด
การป้อนข้อมูล	ใช้คีย์เมตริกซ์ขนาด 3 x 4 กว้าง 5.5 ซม. ยาว 8 ซม.
ส่วนแสดงผล	จอแสดงแบบผลึกเหลว ขนาด 18 ตัวอักษร 2 บรรทัด สูง 0.5 ซม. กว้าง 2.5 ซม. ยาว 7 ซม.
การส่งข้อมูล	ส่งข้อมูลแบบอนุกรม ขนาด 8 บิต
แหล่งจ่ายพลังงาน	แบบแบตเตอรี่ไฟกระแสตรง ขนาด 9 โวลต์
ขนาดเครื่อง	กว้าง 12 ซม. ยาว 15 ซม. สูง 6 ซม.

<u>เครื่องรับ</u>

คุณสมบัติ	รายละเอียด
การรับข้อมูล	รับได้ทั้งหมด 16 เครื่อง รับข้อมูลแบบอนุกรม ขนาด 8 บิต
แหล่งจ่ายพลังงาน	ไฟฟ้ากระแสสลับ 220 โวลต์ ความถี่ 50-60 เฮิรตซ์
การส่งข้อมูล	ส่งข้อมูลแบบอนุกรม ขนาด 8 บิต
การส่งข้อมูลไปยังฐานข้อมูล	ส่งข้อมูลด้วยพอร์ตอนุกรม ขนาด 8 บิต
ขนาดเครื่อง	กว้าง 18 ซม. ยาว 34 ซม. สูง 7 ซม.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก ฉ
รายละเอียดและคุณสมบัติของอุปกรณ์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Features

- Compatible with MCS-51™ Products
- 8K Bytes of In-System Reprogrammable Flash Memory
- Endurance: 1,000 Write/Erase Cycles
- Fully Static Operation: 0 Hz to 24 MHz
- Three-level Program Memory Lock
- 256 x 8-bit Internal RAM
- 32 Programmable I/O Lines
- Three 16-bit Timer/Counters
- Eight Interrupt Sources
- Programmable Serial Channel
- Low-power Idle and Power-down Modes

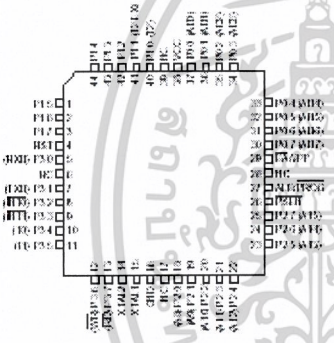
Description

The AT89C52 is a low-power, high-performance CMOS 8-bit microcomputer with 8K bytes of Flash programmable and erasable read only memory (PEROM). The device is manufactured using Atmel's high-density nonvolatile memory technology and is compatible with the industry-standard 80C51 and 80C52 instruction set and pinout. The on-chip Flash allows the program memory to be reprogrammed in-system or by a conventional nonvolatile memory programmer. By combining a versatile 8-bit CPU with Flash on a monolithic chip, the Atmel AT89C52 is a powerful microcomputer which provides a highly-flexible and cost-effective solution to many embedded control applications.

Pin Configurations

PDIP

PQFP/TQFP



PLCC

8-bit Microcontroller with 8K Bytes Flash

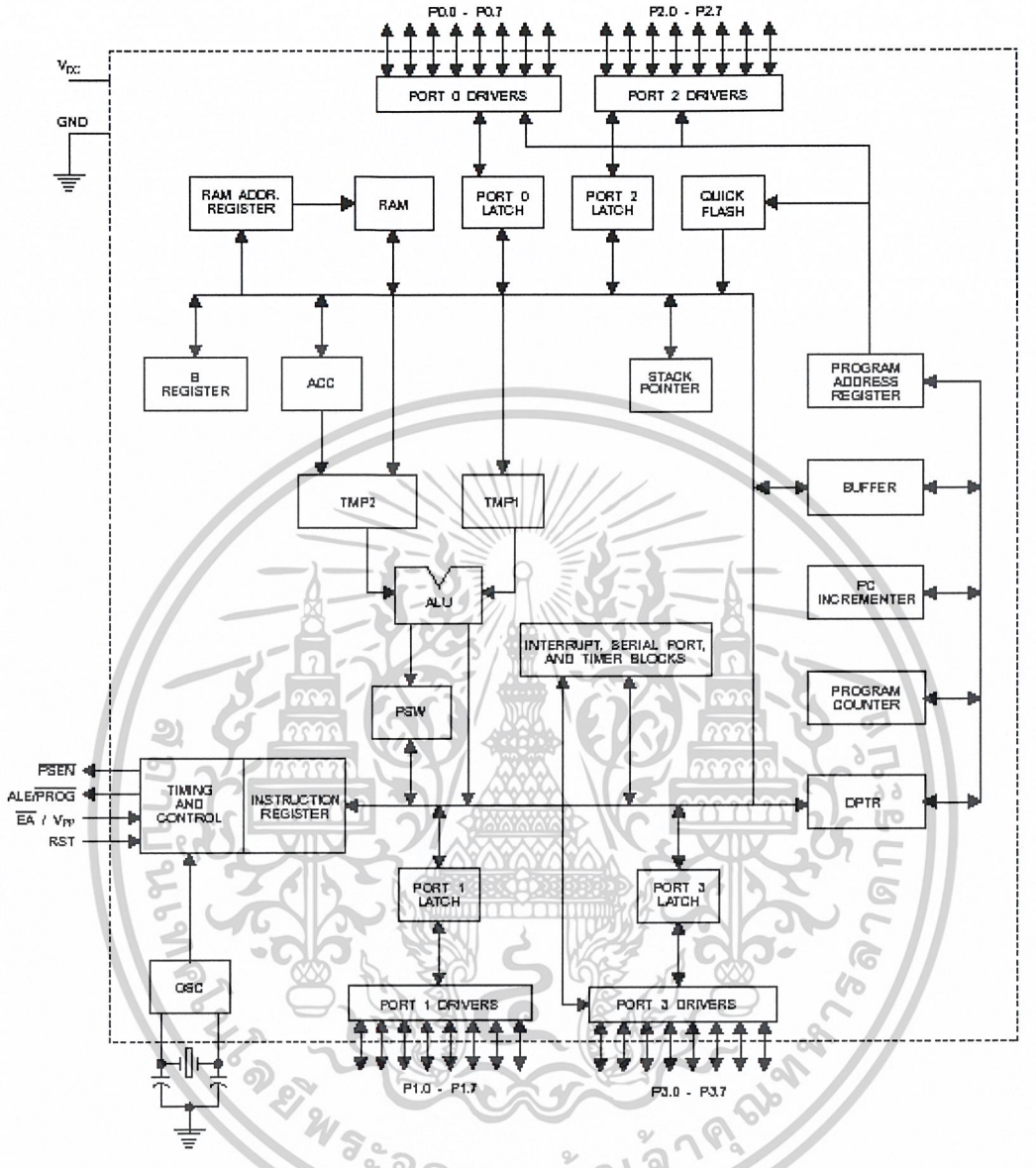
AT89C52

Not Recommended for New Designs. Use AT89S52.

Rev. 0313H–02/00

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Block Diagram



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

The AT89C52 provides the following standard features: 8K bytes of Flash, 256 bytes of RAM, 32 I/O lines, three 16-bit timer/counters, a six-vector two-level interrupt architecture, a full-duplex serial port, on-chip oscillator, and clock circuitry. In addition, the AT89C52 is designed with static logic for operation down to zero frequency and supports two software selectable power saving modes. The Idle Mode stops the CPU while allowing the RAM, timer/counters, serial port, and interrupt system to continue functioning. The Power-down mode saves the RAM contents but freezes the oscillator, disabling all other chip functions until the next hardware reset.

Pin Description

VCC
Supply voltage.

GND
Ground.

Port 0

Port 0 is an 8-bit open drain bi-directional I/O port. As an output port, each pin can sink eight TTL inputs. When 1s are written to port 0 pins, the pins can be used as high-impedance inputs.

Port 0 can also be configured to be the multiplexed low-order address/data bus during accesses to external program and data memory. In this mode, P0 has internal pullups.

Port 0 also receives the code bytes during Flash programming and outputs the code bytes during program verification. External pullups are required during program verification.

Port 1

Port 1 is an 8-bit bi-directional I/O port with internal pullups. The Port 1 output buffers can sink/source four TTL inputs. When 1s are written to Port 1 pins, they are pulled high by the internal pullups and can be used as inputs. As inputs, Port 1 pins that are externally being pulled low will source current (I_{IL}) because of the internal pullups.

In addition, P1.0 and P1.1 can be configured to be the timer/counter 2 external count input (P1.0/T2) and the timer/counter 2 trigger input (P1.1/T2EX), respectively, as shown in the following table.
Port 1 also receives the low-order address bytes during Flash programming and verification.

Port Pin	Alternate Functions
P1.0	T2 (external count input to Timer/Counter 2), clock-out
P1.1	T2EX (Timer/Counter 2 capture/reload trigger and direction control)

Port 2

Port 2 is an 8-bit bi-directional I/O port with internal pullups. The Port 2 output buffers can sink/source four TTL inputs. When 1s are written to Port 2 pins, they are pulled high by the internal pullups and can be used as inputs. As inputs, Port 2 pins that are externally being pulled low will source current (I_{IL}) because of the internal pullups.

Port 2 emits the high-order address byte during fetches from external program memory and during accesses to external data memory that use 16-bit addresses (MOVX @ DPTR). In this application, Port 2 uses strong internal pullups when emitting 1s. During accesses to external data memory that use 8-bit addresses (MOVX @ RI), Port 2 emits the contents of the P2 Special Function Register.

Port 2 also receives the high-order address bits and some control signals during Flash programming and verification.

Port 3

Port 3 is an 8-bit bi-directional I/O port with internal pullups. The Port 3 output buffers can sink/source four TTL inputs. When 1s are written to Port 3 pins, they are pulled high by the internal pullups and can be used as inputs. As inputs, Port 3 pins that are externally being pulled low will source current (I_{IL}) because of the pullups.

Port 3 also serves the functions of various special features of the AT89C51, as shown in the following table.

Port 3 also receives some control signals for Flash programming and verification.

Port Pin	Alternate Functions
P3.0	RXD (serial input port)
P3.1	TXD (serial output port)
P3.2	$\overline{INT0}$ (external interrupt 0)
P3.3	$\overline{INT1}$ (external interrupt 1)
P3.4	T0 (timer 0 external input)
P3.5	T1 (timer 1 external input)
P3.6	$\overline{WR}$ (external data memory write strobe)
P3.7	$\overline{RD}$ (external data memory read strobe)

RST

Reset input. A high on this pin for two machine cycles while the oscillator is running resets the device.

ALE/$\overline{PROG}$

Address Latch Enable is an output pulse for latching the low byte of the address during accesses to external memory. This pin is also the program pulse input ($\overline{PROG}$) during Flash programming.

In normal operation, ALE is emitted at a constant rate of 1/6 the oscillator frequency and may be used for external

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

timing or clocking purposes. Note, however, that one ALE pulse is skipped during each access to external data memory.

If desired, ALE operation can be disabled by setting bit 0 of SFR location 8EH. With the bit set, ALE is active only during a MOVX or MOVC instruction. Otherwise, the pin is weakly pulled high. Setting the ALE-disable bit has no effect if the microcontroller is in external execution mode.

PSEN

Program Store Enable is the read strobe to external program memory.

When the AT89C52 is executing code from external program memory, PSEN is activated twice each machine cycle, except that two PSEN activations are skipped during each access to external data memory.

EA/VPP

External Access Enable. EA must be strapped to GND in order to enable the device to fetch code from external program memory locations starting at 0000H up to FFFFH. Note, however, that if lock bit 1 is programmed, EA will be internally latched on reset.

EA should be strapped to V_{CC} for internal program executions.

This pin also receives the 12-volt programming enable voltage (V_{PP}) during Flash programming when 12-volt programming is selected.

XTAL1

Input to the inverting oscillator amplifier and input to the internal clock operating circuit.

XTAL2

Output from the inverting oscillator amplifier.

Table 1. AT89C52 SFR Map and Reset Values

0F8H									0FFH
0F0H	B 00000000								0F7H
0E8H									0EFH
0E0H	ACC 00000000								0E7H
0D8H									0DFH
0D0H	PSW 00000000								0D7H
0C8H	T2CON 00000000	T2MOD XXXXXX00	RCAP2L 00000000	RCAP2H 00000000	TL2 00000000	TH2 00000000			0CFH
0C0H									0C7H
0B8H	IP XX000000								0BFH
0B0H	P3 11111111								0B7H
0A8H	IE 0X000000								0AFH
0A0H	P2 11111111								0A7H
98H	SCON 00000000	SBUF XXXXXXXX							9FH
90H	P1 11111111								97H
88H	TCON 00000000	TMOD 00000000	TL0 00000000	TL1 00000000	TH0 00000000	TH1 00000000			8FH
80H	P0 11111111	SP 00000111	DPL 00000000	DPH 00000000				PCON 0XXX0000	87H

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Special Function Registers

A map of the on-chip memory area called the Special Function Register (SFR) space is shown in Table 1.

Note that not all of the addresses are occupied, and unoccupied addresses may not be implemented on the chip. Read accesses to these addresses will in general return random data, and write accesses will have an indeterminate effect.

User software should not write 1s to these unlisted locations, since they may be used in future products to invoke new features. In that case, the reset or inactive values of the new bits will always be 0.

Timer 2 Registers Control and status bits are contained in registers T2CON (shown in Table 2) and T2MOD (shown in Table 4) for Timer 2. The register pair (RCAP2H, RCAP2L) are the Capture/Reload registers for Timer 2 in 16-bit capture mode or 16-bit auto-reload mode.

Interrupt Registers The individual interrupt enable bits are in the IE register. Two priorities can be set for each of the six interrupt sources in the IP register.r

Table 2. T2CON – Timer/Counter 2 Control Registe

T2CON Address = 0C8H — Reset Value = 0000 0000B

Bit Addressable

Bit	TF2	EXF2	RCLK	TCLK	EXEN2	TR2	C/$\overline{T2}$	CP/$\overline{RL2}$
	7	6	5	4	3	2	1	0

Symbol	Function
TF2	Timer 2 overflow flag set by a Timer 2 overflow and must be cleared by software. TF2 will not be set when either RCLK = 1 or TCLK = 1.
EXF2	Timer 2 external flag set when either a capture or reload is caused by a negative transition on T2EX and EXEN2 = 1. When Timer 2 interrupt is enabled, EXF2 = 1 will cause the CPU to vector to the Timer 2 interrupt routine. EXF2 must be cleared by software. EXF2 does not cause an interrupt in up/down counter mode (DCEN = 1).
RCLK	Receive clock enable. When set, causes the serial port to use Timer 2 overflow pulses for its receive clock in serial port Modes 1 and 3. RCLK = 0 causes Timer 1 overflow to be used for the receive clock.
TCLK	Transmit clock enable. When set, causes the serial port to use Timer 2 overflow pulses for its transmit clock in serial port Modes 1 and 3. TCLK = 0 causes Timer 1 overflows to be used for the transmit clock.
EXEN2	Timer 2 external enable. When set, allows a capture or reload to occur as a result of a negative transition on T2EX if Timer 2 is not being used to clock the serial port. EXEN2 = 0 causes Timer 2 to ignore events at T2EX.
TR2	Start/Stop control for Timer 2. TR2 = 1 starts the timer.
C/$\overline{T2}$	Timer or counter select for Timer 2. C/$\overline{T2}$ = 0 for timer function. C/$\overline{T2}$ = 1 for external event counter (falling edge triggered).
CP/$\overline{RL2}$	Capture/Reload select. CP/$\overline{RL2}$ = 1 causes captures to occur on negative transitions at T2EX if EXEN2 = 1. CP/$\overline{RL2}$ = 0 causes automatic reloads to occur when Timer 2 overflows or negative transitions occur at T2EX when EXEN2 = 1. When either RCLK or TCLK = 1, this bit is ignored and the timer is forced to auto-reload on Timer 2 overflow.

Data Memory

The AT89C52 implements 256 bytes of on-chip RAM. The upper 128 bytes occupy a parallel address space to the Special Function Registers. That means the upper 128 bytes have the same addresses as the SFR space but are physically separate from SFR space.

When an instruction accesses an internal location above address 7FH, the address mode used in the instruction specifies whether the CPU accesses the upper 128 bytes of RAM or the SFR space. Instructions that use direct addressing access SFR space.

For example, the following direct addressing instruction accesses the SFR at location 0A0H (which is P2).

```
MOV 0A0H, #data
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Instructions that use indirect addressing access the upper 128 bytes of RAM. For example, the following indirect addressing instruction, where R0 contains 0A0H, accesses the data byte at address 0A0H, rather than P2 (whose address is 0A0H).

```
MOV @R0, #data
```

Note that stack operations are examples of indirect addressing, so the upper 128 bytes of data RAM are available as stack space.

Timer 0 and 1

Timer 0 and Timer 1 in the AT89C52 operate the same way as Timer 0 and Timer 1 in the AT89C51.

Timer 2

Timer 2 is a 16-bit Timer/Counter that can operate as either a timer or an event counter. The type of operation is selected by bit C/$\overline{T2}$ in the SFR T2CON (shown in Table 2). Timer 2 has three operating modes: capture, auto-reload (up or down counting), and baud rate generator. The modes are selected by bits in T2CON, as shown in Table 3.

Timer 2 consists of two 8-bit registers, TH2 and TL2. In the Timer function, the TL2 register is incremented every machine cycle. Since a machine cycle consists of 12 oscillator periods, the count rate is 1/12 of the oscillator frequency.

Table 3. Timer 2 Operating Modes

RCLK +TCLK	CP/$\overline{RL2}$	TR2	MODE
0	0	1	16-bit Auto-reload
0	1	1	16-bit Capture
1	X	1	Baud Rate Generator
X	X	0	(Off)

In the Counter function, the register is incremented in response to a 1-to-0 transition at its corresponding external input pin, T2. In this function, the external input is sampled during S5P2 of every machine cycle. When the samples show a high in one cycle and a low in the next cycle, the count is incremented. The new count value appears in the register during S3P1 of the cycle following the one in which the transition was detected. Since two machine cycles (24 oscillator periods) are required to recognize a 1-to-0 transition, the maximum count rate is 1/24 of the oscillator frequency. To ensure that a given level is sampled at least once before it changes, the level should be held for at least one full machine cycle.

Capture Mode

In the capture mode, two options are selected by bit EXEN2 in T2CON. If EXEN2 = 0, Timer 2 is a 16-bit timer or counter which upon overflow sets bit TF2 in T2CON. This bit can then be used to generate an interrupt. If EXEN2 = 1, Timer 2 performs the same operation, but a 1-to-0 transition at external input T2EX also causes the current value in TH2 and TL2 to be captured into RCAP2H and RCAP2L, respectively. In addition, the transition at T2EX causes bit EXF2 in T2CON to be set. The EXF2 bit, like TF2, can generate an interrupt. The capture mode is illustrated in Figure 1.

Auto-reload (Up or Down Counter)

Timer 2 can be programmed to count up or down when configured in its 16-bit auto-reload mode. This feature is invoked by the DCEN (Down Counter Enable) bit located in the SFR T2MOD (see Table 4). Upon reset, the DCEN bit is set to 0 so that timer 2 will default to count up. When DCEN is set, Timer 2 can count up or down, depending on the value of the T2EX pin.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Figure 1. Timer in Capture Mode

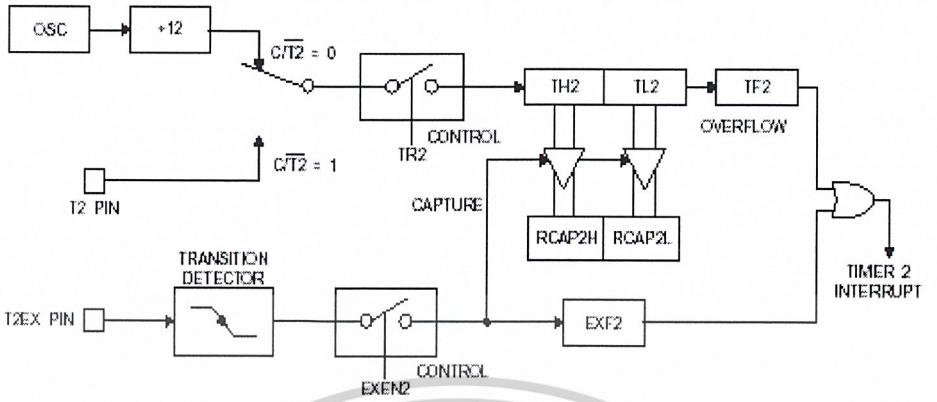


Figure 2 shows Timer 2 automatically counting up when DCEN = 0. In this mode, two options are selected by bit EXEN2 in T2CON. If EXEN2 = 0, Timer 2 counts up to 0FFFFH and then sets the TF2 bit upon overflow. The overflow also causes the timer registers to be reloaded with the 16-bit value in RCAP2H and RCAP2L. The values in Timer in Capture ModeRCAP2H and RCAP2L are preset by software. If EXEN2 = 1, a 16-bit reload can be triggered either by an overflow or by a 1-to-0 transition at external input T2EX. This transition also sets the EXF2 bit. Both the TF2 and EXF2 bits can generate an interrupt if enabled.

Setting the DCEN bit enables Timer 2 to count up or down, as shown in Figure 3. In this mode, the T2EX pin controls the direction of the count. A logic 1 at T2EX makes Timer 2 count up. The timer will overflow at 0FFFFH and set the TF2 bit. This overflow also causes the 16-bit value in RCAP2H and RCAP2L to be reloaded into the timer registers, TH2 and TL2, respectively.

A logic 0 at T2EX makes Timer 2 count down. The timer underflows when TH2 and TL2 equal the values stored in RCAP2H and RCAP2L. The underflow sets the TF2 bit and causes 0FFFFH to be reloaded into the timer registers.

The EXF2 bit toggles whenever Timer 2 overflows or underflows and can be used as a 17th bit of resolution. In this operating mode, EXF2 does not flag an interrupt.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Baud Rate Generator

Timer 2 is selected as the baud rate generator by setting TCLK and/or RCLK in T2CON (Table 2). Note that the baud rates for transmit and receive can be different if Timer 2 is used for the receiver or transmitter and Timer 1 is used for the other function. Setting RCLK and/or TCLK puts Timer 2 into its baud rate generator mode, as shown in Figure 4.

The baud rate generator mode is similar to the auto-reload mode, in that a rollover in TH2 causes the Timer 2 registers to be reloaded with the 16-bit value in registers RCAP2H and RCAP2L, which are preset by software.

The baud rates in Modes 1 and 3 are determined by Timer 2's overflow rate according to the following equation.

$$\text{Modes 1 and 3 Baud Rates} = \frac{\text{Timer 2 Overflow Rate}}{16}$$

The Timer can be configured for either timer or counter operation. In most applications, it is configured for timer operation ($C/\overline{T2} = 0$). The timer operation is different for Timer 2 when it is used as a baud rate generator. Normally, as a timer, it increments every machine cycle (at 1/12 the oscillator frequency). As a baud rate generator, however, it increments every state time (at 1/2 the oscillator frequency). The baud rate formula is given below.

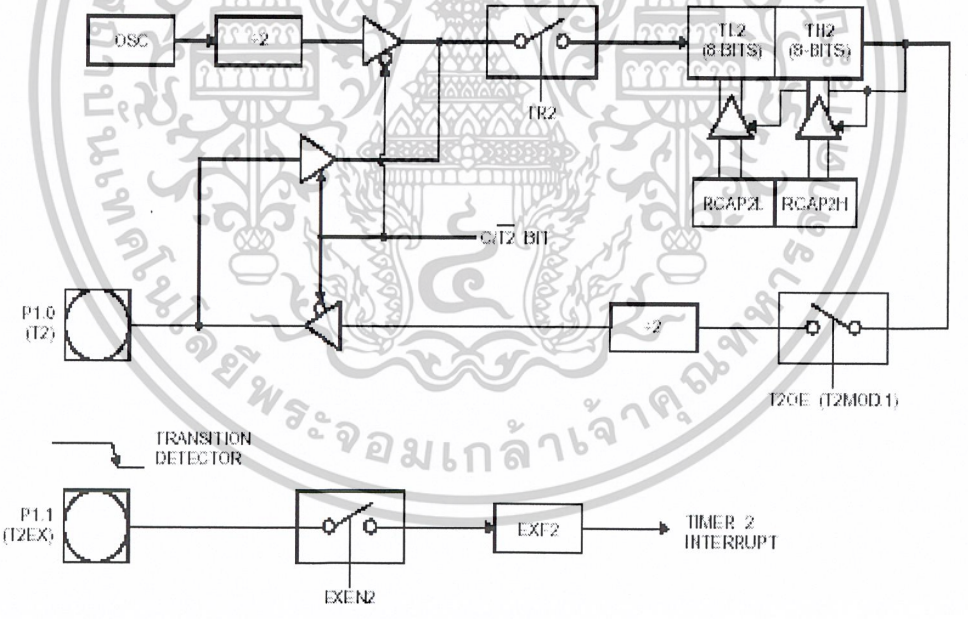
$$\frac{\text{Modes 1 and 3}}{\text{Baud Rate}} = \frac{\text{Oscillator Frequency}}{32 \times [65536 - (\text{RCAP2H,RCAP2L})]}$$

where (RCAP2H, RCAP2L) is the content of RCAP2H and RCAP2L taken as a 16-bit unsigned integer.

Timer 2 as a baud rate generator is shown in Figure 4. This figure is valid only if RCLK or TCLK = 1 in T2CON. Note that a rollover in TH2 does not set TF2 and will not generate an interrupt. Note too, that if EXEN2 is set, a 1-to-0 transition in T2EX will set EXF2 but will not cause a reload from (RCAP2H, RCAP2L) to (TH2, TL2). Thus when Timer 2 is in use as a baud rate generator, T2EX can be used as an extra external interrupt.

Note that when Timer 2 is running (TR2 = 1) as a timer in the baud rate generator mode, TH2 or TL2 should not be read from or written to. Under these conditions, the Timer is incremented every state time, and the results of a read or write may not be accurate. The RCAP2 registers may be read but should not be written to, because a write might overlap a reload and cause write and/or reload errors. The timer should be turned off (clear TR2) before accessing the Timer 2 or RCAP2 registers.

Figure 5. Timer 2 in Clock-out Mode



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Oscillator Characteristics

XTAL1 and XTAL2 are the input and output, respectively, of an inverting amplifier that can be configured for use as an on-chip oscillator, as shown in Figure 7. Either a quartz crystal or ceramic resonator may be used. To drive the device from an external clock source, XTAL2 should be left unconnected while XTAL1 is driven, as shown in Figure 8. There are no requirements on the duty cycle of the external clock signal, since the input to the internal clocking circuitry is through a divide-by-two flip-flop, but minimum and maximum voltage high and low time specifications must be observed.

Idle Mode

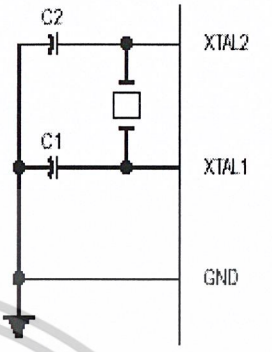
In idle mode, the CPU puts itself to sleep while all the on-chip peripherals remain active. The mode is invoked by software. The content of the on-chip RAM and all the special functions registers remain unchanged during this mode. The idle mode can be terminated by any enabled interrupt or by a hardware reset.

Note that when idle mode is terminated by a hardware reset, the device normally resumes program execution from where it left off, up to two machine cycles before the internal reset algorithm takes control. On-chip hardware inhibits access to internal RAM in this event, but access to the port pins is not inhibited. To eliminate the possibility of an unexpected write to a port pin when idle mode is terminated by a reset, the instruction following the one that invokes idle mode should not write to a port pin or to external memory.

Power-down Mode

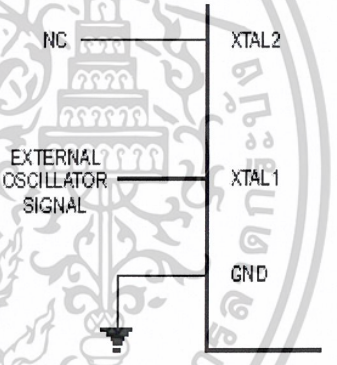
In the power-down mode, the oscillator is stopped, and the instruction that invokes power-down is the last instruction executed. The on-chip RAM and Special Function Registers retain their values until the power-down mode is terminated. The only exit from power-down is a hardware reset. Reset redefines the SFRs but does not change the on-chip RAM. The reset should not be activated before V_{CC} is restored to its normal operating level and must be held active long enough to allow the oscillator to restart and stabilize.

Figure 7. Oscillator Connections



Note: C1, C2 = 30 pF ± 10 pF for Crystals
= 40 pF ± 10 pF for Ceramic Resonators

Figure 8. External Clock Drive Configuration



Status of External Pins During Idle and Powe-down Modes

Mode	Program Memory	ALE	$\overline{PSEN}$	PORT0	PORT1	PORT2	PORT3
Idle	Internal	1	1	Data	Data	Data	Data
Idle	External	1	1	Float	Data	Address	Data
Power-down	Internal	0	0	Data	Data	Data	Data
Power-down	External	0	0	Float	Data	Data	Data

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

DC Characteristics

The values shown in this table are valid for T_A = -40°C to 85°C and V_{CC} = 5.0V ±20%, unless otherwise noted.

Symbol	Parameter	Condition	Min	Max	Units
V_{IL}	Input Low-voltage	(Except $\overline{EA}$)	-0.5	0.2 V_{CC}-0.1	V
V_{IL1}	Input Low-voltage ($\overline{EA}$)		-0.5	0.2 V_{CC}-0.3	V
V_{IH}	Input High-voltage	(Except XTAL1, RST)	0.2 V_{CC}+0.9	V_{CC}+0.5	V
V_{IH1}	Input High-voltage	(XTAL1, RST)	0.7 V_{CC}	V_{CC}+0.5	V
V_{OL}	Output Low-voltage[1] (Ports 1,2,3)	I_{OL} = 1.6 mA		0.45	V
V_{OL1}	Output Low-voltage[1] (Port 0, ALE, $\overline{PSEN}$)	I_{OL} = 3.2 mA		0.45	V
V_{OH}	Output High-voltage (Ports 1,2,3, ALE, $\overline{PSEN}$)	I_{OH} = -60 µA, V_{CC} = 5V ± 10%	2.4		V
		I_{OH} = -25 µA	0.75 V_{CC}		V
		I_{OH} = -10 µA	0.9 V_{CC}		V
V_{OH1}	Output High-voltage (Port 0 in External Bus Mode)	I_{OH} = -800 µA, V_{CC} = 5V ±10%	2.4		V
		I_{OH} = -300 µA	0.75 V_{CC}		V
		I_{OH} = -80 µA	0.9 V_{CC}		V
I_{IL}	Logical 0 Input Current (Ports 1,2,3)	V_{IN} = 0.45V		-50	µA
I_{TL}	Logical 1 to 0 Transition Current (Ports 1,2,3)	V_{IN} = 2V, V_{CC} = 5V ± 10%		-650	µA
I_{LI}	Input Leakage Current (Port 0, $\overline{EA}$)	0.45 < V_{IN} < V_{CC}		±10	µA
RRST	Reset Pulldown Resistor		50	300	KΩ
C_{IO}	Pin Capacitance	Test Freq. = 1 MHz, T_A = 25° C		10	pF
I_{CC}	Power Supply Current	Active Mode, 12 MHz		25	mA
		Idle Mode, 12 MHz		6.5	mA
	Power-down Mode[2]	V_{CC} = 6V		100	µA
		V_{CC} = 3V		40	µA

Notes: 1. Under steady state (non-transient) conditions, I_{OL} must be externally limited as follows:
Maximum I_{OL} per port pin: 10 mA
Maximum I_{OL} per 8-bit port:
Port 0: 26 mA Ports 1, 2, 3: 15 mA
Maximum total I_{OL} for all output pins: 71 mA
If I_{OL} exceeds the test condition, V_{OL} may exceed the related specification. Pins are not guaranteed to sink current greater than the listed test conditions.
2. Minimum V_{CC} for Power-down is 2V.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

AC Characteristics

Under operating conditions, load capacitance for Port 0, ALE/$\overline{PROG}$, and $\overline{PSEN}$ = 100 pF; load capacitance for all other outputs = 80 pF.

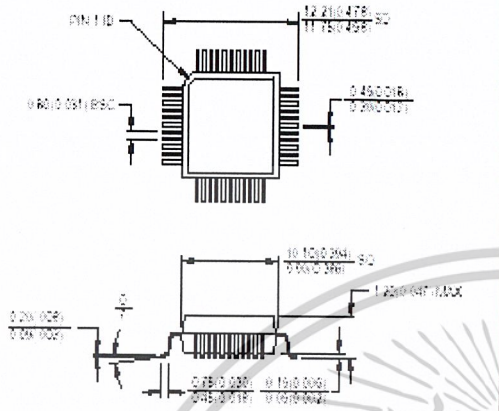
External Program and Data Memory Characteristics

Symbol	Parameter	12 MHz Oscillator		Variable Oscillator		Units
		Min	Max	Min	Max	
$1/t_{CLCL}$	Oscillator Frequency			0	24	MHz
t_{LHLL}	ALE Pulse Width	127		$2t_{CLCL}$-40		ns
t_{AVLL}	Address Valid to ALE Low	43		t_{CLCL}-13		ns
t_{LLAX}	Address Hold After ALE Low	48		t_{CLCL}-20		ns
t_{LLIV}	ALE Low to Valid Instruction In		233		$4t_{CLCL}$-65	ns
t_{LLPL}	ALE Low to $\overline{PSEN}$ Low	43		t_{CLCL}-13		ns
t_{PLPH}	$\overline{PSEN}$ Pulse Width	205		$3t_{CLCL}$-20		ns
t_{PLIV}	$\overline{PSEN}$ Low to Valid Instruction In		145		$3t_{CLCL}$-45	ns
t_{PXIX}	Input Instruction Hold after $\overline{PSEN}$	0		0		ns
t_{PXIZ}	Input Instruction Float after $\overline{PSEN}$		59		t_{CLCL}-10	ns
t_{PXAV}	$\overline{PSEN}$ to Address Valid	75		t_{CLCL}-8		ns
t_{AVIV}	Address to Valid Instruction In		312		$5t_{CLCL}$-55	ns
t_{PLAZ}	$\overline{PSEN}$ Low to Address Float		10		10	ns
t_{RLRH}	$\overline{RD}$ Pulse Width	400		$6t_{CLCL}$-100		ns
t_{WLWH}	$\overline{WR}$ Pulse Width	400		$6t_{CLCL}$-100		ns
t_{RLDV}	$\overline{RD}$ Low to Valid Data In		252		$5t_{CLCL}$-90	ns
t_{RHDX}	Data Hold After $\overline{RD}$	0		0		ns
t_{RHDZ}	Data Float After $\overline{RD}$		97		$2t_{CLCL}$-28	ns
t_{LLDV}	ALE Low to Valid Data In		517		$8t_{CLCL}$-150	ns
t_{AVDV}	Address to Valid Data In		585		$9t_{CLCL}$-165	ns
t_{LLWL}	ALE Low to $\overline{RD}$ or $\overline{WR}$ Low	200	300	$3t_{CLCL}$-50	$3t_{CLCL}$+50	ns
t_{AVWL}	Address to $\overline{RD}$ or $\overline{WR}$ Low	203		$4t_{CLCL}$-75		ns
t_{QVWX}	Data Valid to $\overline{WR}$ Transition	23		t_{CLCL}-20		ns
t_{QVWH}	Data Valid to $\overline{WR}$ High	433		$7t_{CLCL}$-120		ns
t_{WHQX}	Data Hold After $\overline{WR}$	33		t_{CLCL}-20		ns
t_{RLAZ}	$\overline{RD}$ Low to Address Float		0		0	ns
t_{WHLH}	$\overline{RD}$ or $\overline{WR}$ High to ALE High	43	123	t_{CLCL}-20	t_{CLCL}+25	ns

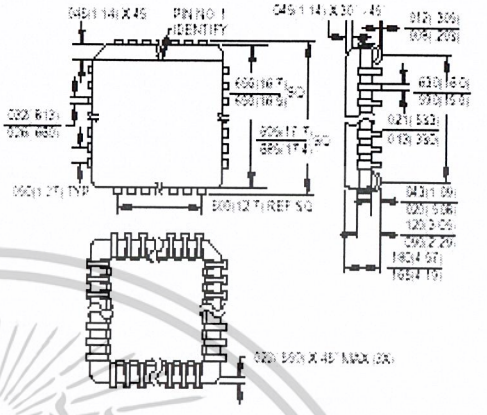
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Packaging Information

44A, 44-lead, Thin (1.0 mm) Plastic Gull Wing Quad Flatpack (TQFP)
Dimensions in Millimeters and (Inches)*
JEDEC STANDARD MS-026 ACB

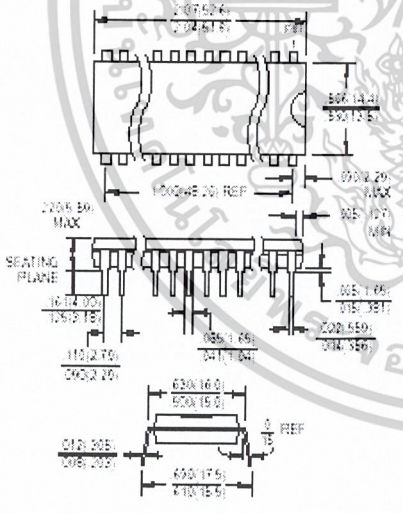


44J, 44-lead, Plastic J-leaded Chip Carrier (PLCC)
Dimensions in Inches and (Millimeters)
JEDEC STANDARD MS-018 AC

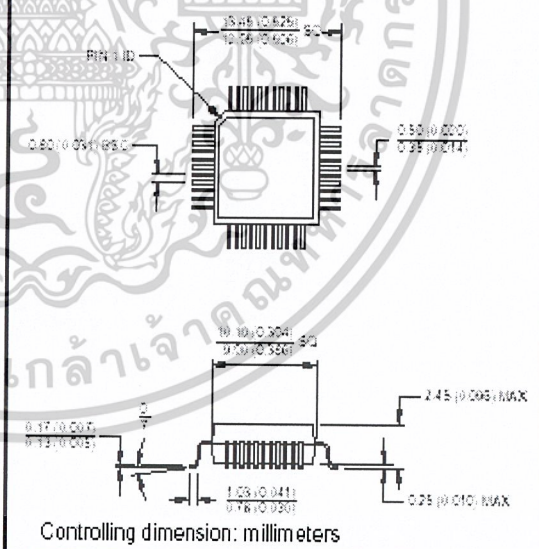


Controlling dimension: millimeters

40P6, 40-lead, 0.600" Wide, Plastic Dual Inline Package (PDIP)
Dimensions in Inches and (Millimeters)



44Q, 44-lead, Plastic Quad Flat Package (PQFP)
Dimensions in Millimeters and (Inches)*
JEDEC STANDARD MS-022 AB

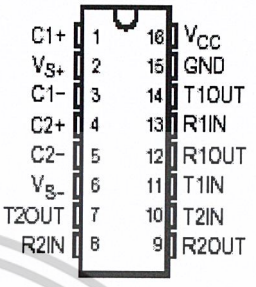


Controlling dimension: millimeters

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- Meet or Exceed TIA/EIA-232-F and ITU Recommendation V.28
- Operate With Single 5-V Power Supply
- Operate Up To 120 kbit/s
- Two Drivers and Two Receivers
- ±30-V Input Levels
- Low Supply Current . . . 8 mA Typical
- Designed to Be Interchangeable With Maxim MAX232
- ESD Protection Exceeds JESD 22
  - 2000-V Human-Body Model (A114-A)
- Applications
  - TIA/EIA-232-F, Battery-Powered Systems, Terminals, Modems, and Computers

MAX232 . . . D, DW, N, OR NS PACKAGE
MAX232I . . . D, DW, OR N PACKAGE
(TOP VIEW)



description/ordering information

The MAX232 is a dual driver/receiver that includes a capacitive voltage generator to supply TIA/EIA-232-F voltage levels from a single 5-V supply. Each receiver converts TIA/EIA-232-F inputs to 5-V TTL/CMOS levels. These receivers have a typical threshold of 1.3 V, a typical hysteresis of 0.5 V, and can accept ±30-V inputs. Each driver converts TTL/CMOS input levels into TIA/EIA-232-F levels. The driver, receiver, and voltage-generator functions are available as cells in the Texas Instruments LinASIC™ library.

ORDERING INFORMATION

T_A	PACKAGE†		ORDERABLE PART NUMBER	TOP-SIDE MARKING
0°C to 70°C	PDIP (N)	Tube of 25	MAX232N	MAX232N
	SOIC (D)	Tube of 40	MAX232D	MAX232
		Reel of 2500	MAX232DR	
	SOIC (DW)	Tube of 40	MAX232DW	MAX232
		Reel of 2000	MAX232DWR	
	SOP (NS)	Reel of 2000	MAX232NSR	MAX232
−40°C to 85°C	PDIP (N)	Tube of 25	MAX232IN	MAX232IN
	SOIC (D)	Tube of 40	MAX232ID	MAX232I
		Reel of 2500	MAX232IDR	
	SOIC (DW)	Tube of 40	MAX232IDW	MAX232I
		Reel of 2000	MAX232IDWR	

† Package drawings, standard packing quantities, thermal data, symbolization, and PCB design guidelines are available at www.ti.com/sc/package.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Function Tables

EACH DRIVER

INPUT TIN	OUTPUT TOUT
L	H
H	L

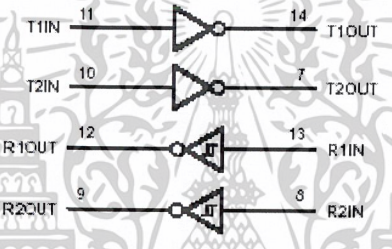
H = high level, L = low level

EACH RECEIVER

INPUT RIN	OUTPUT ROUT
L	H
H	L

H = high level, L = low level

logic diagram (positive logic)



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

absolute maximum ratings over operating free-air temperature range (unless otherwise noted)†

Input supply voltage range, V_{CC} (see Note 1) ..... −0.3 V to 6 V
Positive output supply voltage range, V_{S+} ..... V_{CC} − 0.3 V to 15 V
Negative output supply voltage range, V_{S-} ..... −0.3 V to −15 V
Input voltage range, V_I: Driver ..... −0.3 V to V_{CC} + 0.3 V
Receiver ..... ±30 V
Output voltage range, V_O: T1OUT, T2OUT ..... V_{S-} − 0.3 V to V_{S+} + 0.3 V
R1OUT, R2OUT ..... −0.3 V to V_{CC} + 0.3 V
Short-circuit duration: T1OUT, T2OUT ..... Unlimited
Package thermal impedance, θ_{JA} (see Notes 2 and 3): D package ..... 73°C/W
DW package ..... 57°C/W
N package ..... 67°C/W
NS package ..... 64°C/W
Operating virtual junction temperature, T_J ..... 150°C
Storage temperature range, T_{stg} ..... −65°C to 150°C

† Stresses beyond those listed under "absolute maximum ratings" may cause permanent damage to the device. These are stress ratings only, and functional operation of the device at these or any other conditions beyond those indicated under "recommended operating conditions" is not implied. Exposure to absolute-maximum-rated conditions for extended periods may affect device reliability.

NOTES: 1. All voltages are with respect to network GND.
2. Maximum power dissipation is a function of $T_J(max)$, θ_{JA}, and T_A. The maximum allowable power dissipation at any allowable ambient temperature is $P_D = (T_J(max) - T_A)/\theta_{JA}$. Operating at the absolute maximum T_J of 150°C can affect reliability.
3. The package thermal impedance is calculated in accordance with JESD 51-7.

recommended operating conditions

			MIN	NOM	MAX	UNIT
V_{CC}	Supply voltage		4.5	5	5.5	V
V_{IH}	High-level input voltage (T1IN,T2IN)		2			V
V_{IL}	Low-level input voltage (T1IN, T2IN)				0.8	V
R1IN, R2IN	Receiver input voltage				±30	V
T_A	Operating free-air temperature	MAX232	0		70	°C
		MAX232I	−40		85	

electrical characteristics over recommended ranges of supply voltage and operating free-air temperature (unless otherwise noted) (see Note 4 and Figure 4)

PARAMETER		TEST CONDITIONS	MIN	TYP‡	MAX	UNIT
I_{CC}	Supply current	V_{CC} = 5.5 V, All outputs open, T_A = 25°C		8	10	mA

‡ All typical values are at V_{CC} = 5 V and T_A = 25°C.
NOTE 4: Test conditions are C1–C4 = 1 µF at V_{CC} = 5 V ± 0.5 V.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

DRIVER SECTION

electrical characteristics over recommended ranges of supply voltage and operating free-air temperature range (see Note 4)

PARAMETER			TEST CONDITIONS		MIN	TYP†	MAX	UNIT
V_{OH}	High-level output voltage	T1OUT, T2OUT	R_L = 3 kΩ to GND		5	7		V
V_{OL}	Low-level output voltage‡	T1OUT, T2OUT	R_L = 3 kΩ to GND			-7	-5	V
r_o	Output resistance	T1OUT, T2OUT	V_{S+} = V_{S-} = 0,	V_O = ±2 V	300			Ω
I_{OS}§	Short-circuit output current	T1OUT, T2OUT	V_{CC} = 5.5 V,	V_O = 0		±10		mA
I_{IS}	Short-circuit input current	T1IN, T2IN	V_I = 0				200	µA

† All typical values are at V_{CC} = 5 V, T_A = 25°C.
‡ The algebraic convention, in which the least-positive (most negative) value is designated minimum, is used in this data sheet for logic voltage levels only.
§ Not more than one output should be shorted at a time.
NOTE 4: Test conditions are C1–C4 = 1 µF at V_{CC} = 5 V ± 0.5 V.

switching characteristics, V_{CC} = 5 V, T_A = 25°C (see Note 4)

PARAMETER		TEST CONDITIONS	MIN	TYP	MAX	UNIT
SR	Driver slew rate	R_L = 3 kΩ to 7 kΩ, See Figure 2			30	V/µs
SR(t)	Driver transition region slew rate	See Figure 3		3		V/µs
	Data rate	One TOUT switching		120		kbit/s

NOTE 4: Test conditions are C1–C4 = 1 µF at V_{CC} = 5 V ± 0.5 V.

RECEIVER SECTION

electrical characteristics over recommended ranges of supply voltage and operating free-air temperature range (see Note 4)

PARAMETER			TEST CONDITIONS		MIN	TYP†	MAX	UNIT
V_{OH}	High-level output voltage	R1OUT, R2OUT	I_{OH} = -1 mA		3.5			V
V_{OL}	Low-level output voltage‡	R1OUT, R2OUT	I_{OL} = 3.2 mA				0.4	V
V_{IT+}	Receiver positive-going input threshold voltage	R1IN, R2IN	V_{CC} = 5 V,	T_A = 25°C		1.7	2.4	V
V_{IT-}	Receiver negative-going input threshold voltage	R1IN, R2IN	V_{CC} = 5 V,	T_A = 25°C	0.8	1.2		V
V_{hys}	Input hysteresis voltage	R1IN, R2IN	V_{CC} = 5 V		0.2	0.5	1	V
r_I	Receiver input resistance	R1IN, R2IN	V_{CC} = 5,	T_A = 25°C	3	5	7	kΩ

† All typical values are at V_{CC} = 5 V, T_A = 25°C.
‡ The algebraic convention, in which the least-positive (most negative) value is designated minimum, is used in this data sheet for logic voltage levels only.
NOTE 4: Test conditions are C1–C4 = 1 µF at V_{CC} = 5 V ± 0.5 V.

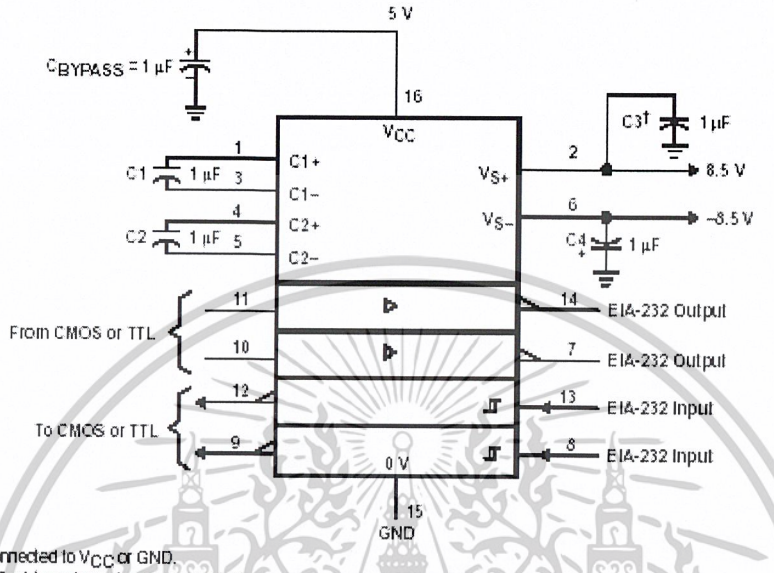
switching characteristics, V_{CC} = 5 V, T_A = 25°C (see Note 4 and Figure 1)

PARAMETER		TYP	UNIT
$t_{PLH(R)}$	Receiver propagation delay time, low- to high-level output	500	ns
$t_{PHL(R)}$	Receiver propagation delay time, high- to low-level output	500	ns

NOTE 4: Test conditions are C1–C4 = 1 µF at V_{CC} = 5 V ± 0.5 V.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

APPLICATION INFORMATION



† C3 can be connected to V_{CC} or GND.

NOTES: A. Resistor values shown are nominal.
B. Nonpolarized ceramic capacitors are acceptable. If polarized tantalum or electrolytic capacitors are used, they should be connected as shown.

Figure 4. Typical Operating Circuit

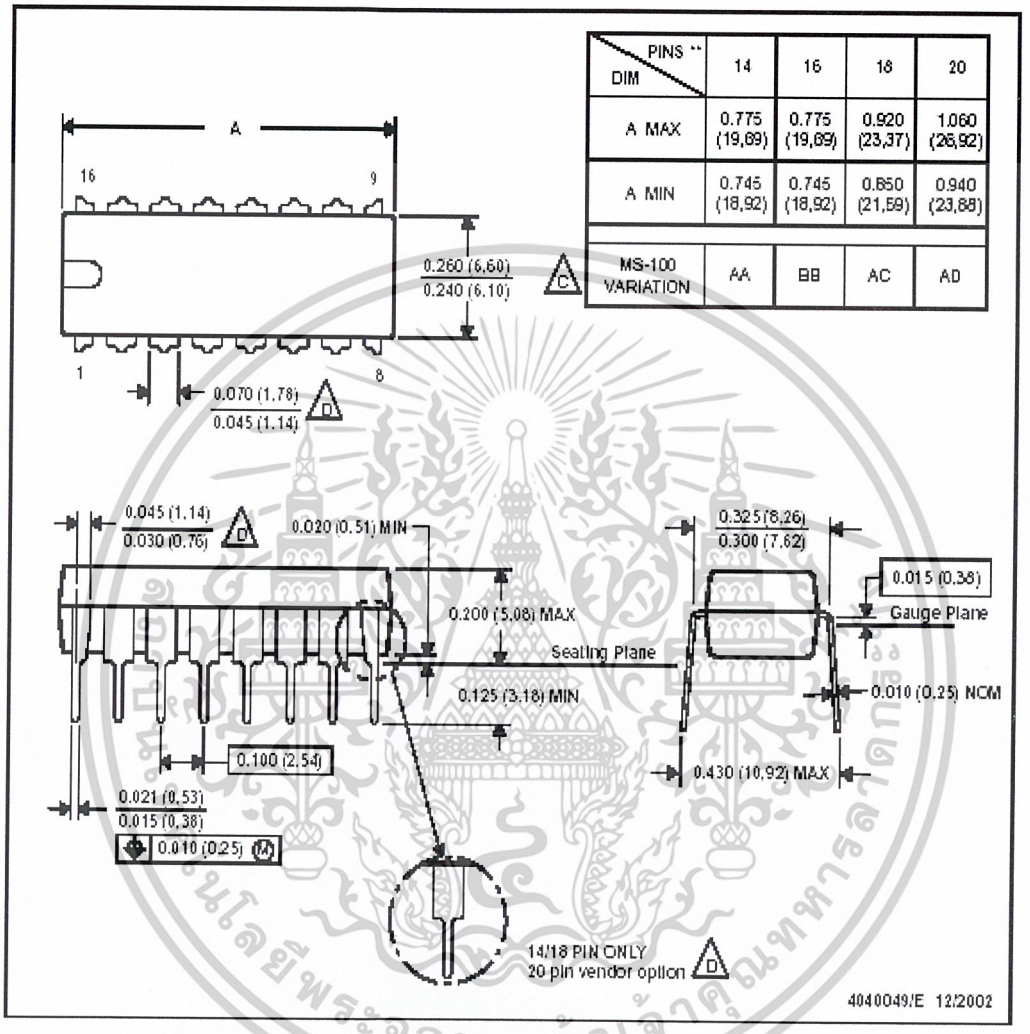
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

MECHANICAL

N (R-PDIP-T**)

PLASTIC DUAL-IN-LINE PACKAGE

16 PINS SHOWN



DIM \ PINS **	14	16	18	20
A MAX	0.775 (19,69)	0.775 (19,69)	0.920 (23,37)	1.060 (26,92)
A MIN	0.745 (18,92)	0.745 (18,92)	0.850 (21,59)	0.940 (23,88)
MS-100 VARIATION	AA	BB	AC	AD

NOTES: A. All linear dimensions are in inches (millimeters).
B. This drawing is subject to change without notice.
C. Falls within JEDEC MS-001, except 18 and 20 pin minimum body length (Dim A).
D. The 20 pin end lead shoulder width is a vendor option, either half or full width.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

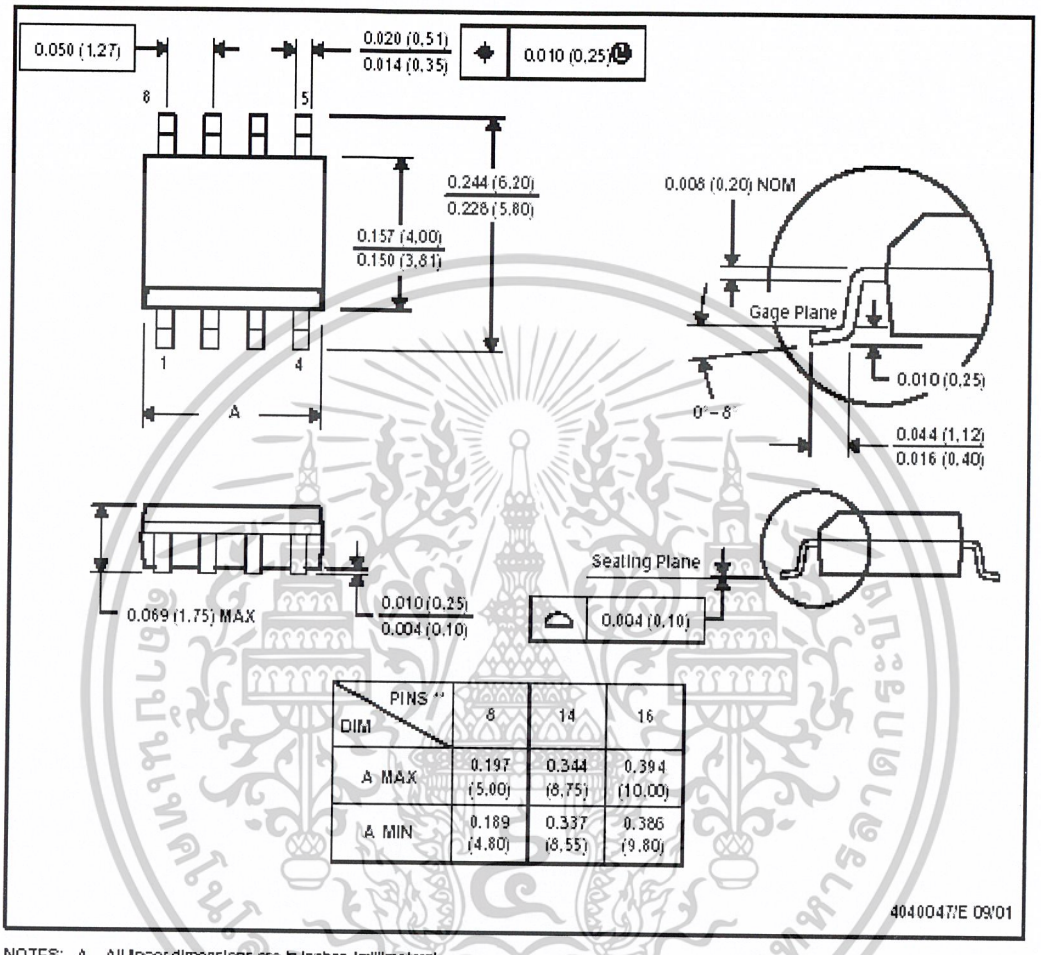
MECHANICAL DATA

MSOI002B – JANUARY 1995 – REVISED SEPTEMBER 2001

D (R-PDSO-G**)

PLASTIC SMALL-OUTLINE PACKAGE

8 PINS SHOWN



DIM \ PINS **	8	14	16
A MAX	0.197 (5,00)	0.344 (8,75)	0.394 (10,00)
A MIN	0.189 (4,80)	0.337 (8,55)	0.386 (9,80)

NOTES: A. All linear dimensions are in inches (millimeters).
B. This drawing is subject to change without notice.
C. Body dimensions do not include mold flash or protrusion, not to exceed 0.006 (0,15).
D. Falls within JEDEC MS-012

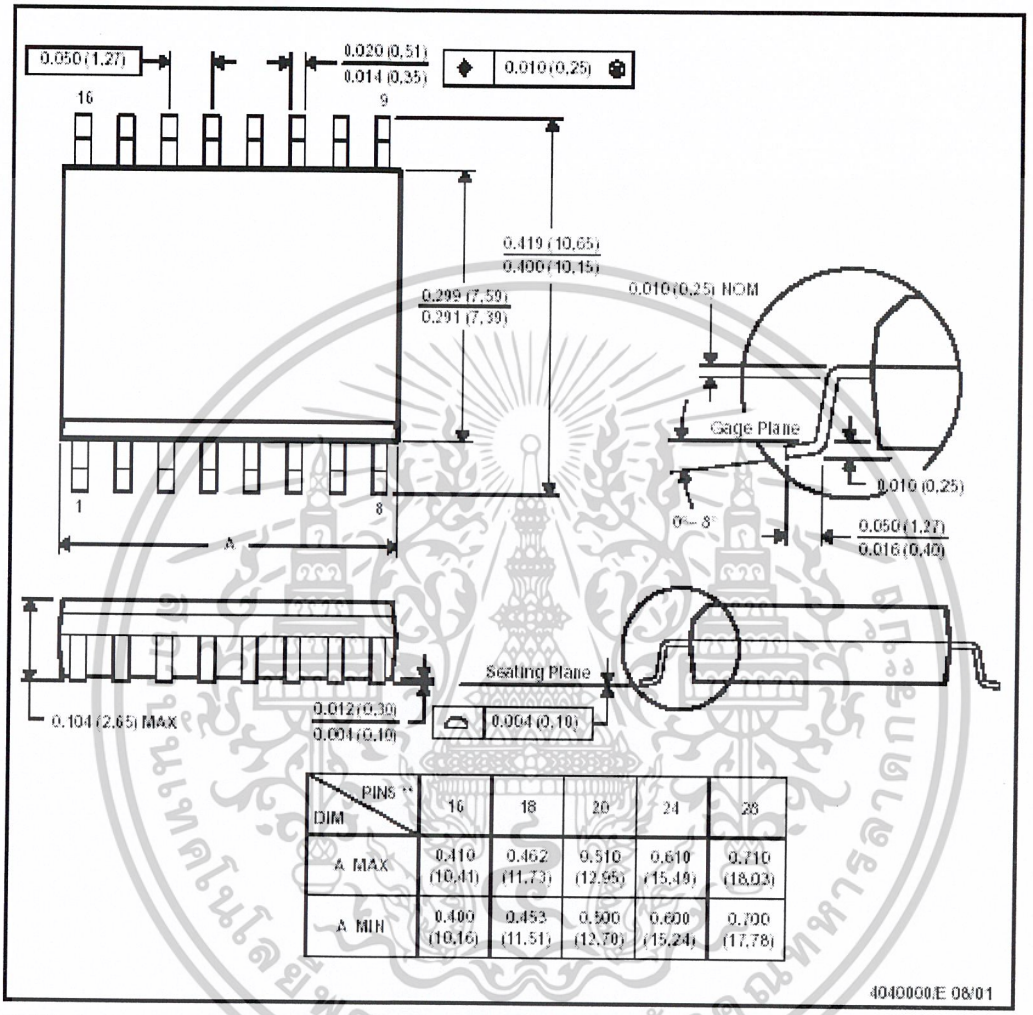
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

MSOI002E – JANUARY 1995 – REVISED SEPTEMBER 2001

DW (R-PDSO-G**)

PLASTIC SMALL-OUTLINE PACKAGE

16 PINS SHOWN



DIM \ PINS **	16	18	20	24	28
A MAX	0,410 (10,41)	0,462 (11,73)	0,510 (12,95)	0,610 (15,49)	0,710 (18,03)
A MIN	0,400 (10,16)	0,453 (11,51)	0,500 (12,70)	0,600 (15,24)	0,700 (17,78)

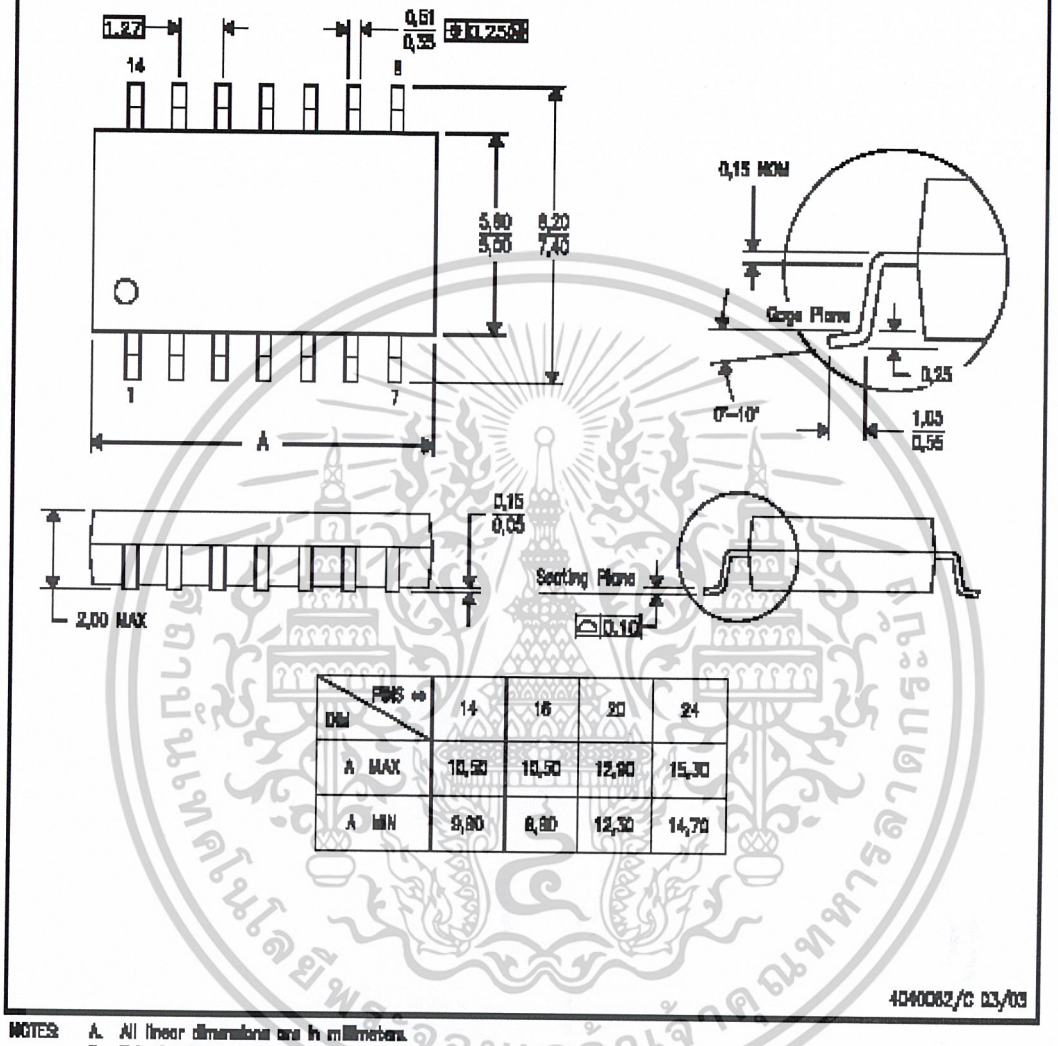
NOTES: A. All linear dimensions are in inches (millimeters).
B. This drawing is subject to change without notice.
C. Body dimensions do not include mold flash or protrusion not to exceed 0.006 (0,15).
D. Falls within JEDEC MS-013

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

MECHANICAL DATA

N8 (R-PDSO-G)** **PLASTIC SMALL-OUTLINE PACKAGE**

14-PIN SHOWN



DIM \ PINS **	14	16	20	24
A MAX	10,50	10,50	12,90	15,30
A MIN	9,90	9,90	12,30	14,70

NOTES:
A. All linear dimensions are in millimeters.
B. This drawing is subject to change without notice.
C. Body dimensions do not include mold flash or protrusion, not to exceed 0,15.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ประวัติผู้แต่ง

ชื่อ-สกุล นางสาวเกษร เดชสุวรรณ

วัน เดือน ปีเกิด 8 พฤศจิกายน พ.ศ. 2524

ภูมิลำเนาเดิม 18 หมู่ 2 ตำบลไสหมาก อำเภอเชียรใหญ่
จังหวัดนครศรีธรรมราช 80190

ประวัติการศึกษา

ประถมศึกษา โรงเรียนวัดทาบทอง จังหวัดนครศรีธรรมราช

มัธยมศึกษาตอนต้น โรงเรียนธัญญาวดีศึกษา จังหวัดนครศรีธรรมราช

ประกาศนียบัตรวิชาชีพ วิทยาลัยเทคนิคยะลา

ประกาศนียบัตรวิชาชีพชั้นสูง วิทยาลัยเทคนิคยะลา

ปริญญาตรี สาขาวิชาอิเล็กทรอนิกส์และคอมพิวเตอร์
ภาควิชาครุศาสตร์วิศวกรรม
คณะครุศาสตร์อุตสาหกรรม สจล.

คติพจน์ อย่าเชื่อในสิ่งที่เห็นทั้งหมด แม้เพียงครึ่งเดียวของสิ่งที่ได้ยิน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ประวัติผู้แต่ง

ชื่อ-สกุล นายจาตุรงค์ ศรโสภา

วัน เดือน ปีเกิด 15 พฤศจิกายน พ.ศ. 2523

ภูมิลำเนา 81/1 หมู่ 1 ตำบลจำปา อำเภอท่าเรือ จังหวัดพระนครศรีอยุธยา 13130

ประวัติการศึกษา

ประถมศึกษา	โรงเรียนแสนโกสิก จังหวัดพระนครศรีอยุธยา
มัธยมศึกษาตอนต้น	โรงเรียนท่าเรือนิตยานุกูล จังหวัดพระนครศรีอยุธยา
ประกาศนียบัตรวิชาชีพ	วิทยาลัยเทคนิคพระนครศรีอยุธยา
ประกาศนียบัตรวิชาชีพชั้นสูง	วิทยาลัยเทคนิคพระนครศรีอยุธยา
ปริญญาตรี	สาขาวิชาอิเล็กทรอนิกส์และคอมพิวเตอร์ ภาควิชาครุศาสตร์วิศวกรรม คณะครุศาสตร์อุตสาหกรรม สจล.

คติพจน์ หากต้องการทำให้งานสำเร็จลุล่วงไปด้วยดี จงลงมือทำเอง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ประวัติผู้แต่ง

ชื่อ-สกุล นายทศพล เพ็ชรภิมล

วัน เดือน ปีเกิด 24 ธันวาคม พ.ศ. 2524

ภูมิลำเนา 32 หมู่ 6 ตำบลดอน อำเภอปะนาเระ จังหวัดปัตตานี 94130

ประวัติการศึกษา

ประถมศึกษา โรงเรียนบ้านดอน (นุ้ยนิธยาคาร) จังหวัดปัตตานี

มัธยมศึกษาตอนต้น โรงเรียนวุฒิชัยวิทยา จังหวัดปัตตานี

ประกาศนียบัตรวิชาชีพ วิทยาลัยเทคนิคยะลา

ประกาศนียบัตรวิชาชีพชั้นสูง วิทยาลัยเทคนิคยะลา

ปริญญาตรี สาขาวิชาอิเล็กทรอนิกส์และคอมพิวเตอร์
ภาควิชาครุศาสตร์วิศวกรรม
คณะครุศาสตร์อุตสาหกรรม สจล.

คติพจน์ การผัดวันประกันพรุ่งทำให้เราเสียเวลาไปโดยเปล่าประโยชน์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ประวัติผู้แต่ง

ชื่อ-สกุล นายเอกราช ศรีสุข

วัน เดือน ปีเกิด 27 ธันวาคม พ.ศ. 2524

ภูมิลำเนา 72 หมู่ 3 ถนนบ้าน โพธิ์ ตำบลทับเที่ยง อำเภอเมือง จังหวัดตรัง 92000

ประวัติการศึกษา

ประถมศึกษา โรงเรียนอนุบาลตรัง จังหวัดตรัง

มัธยมศึกษาตอนต้น โรงเรียนบูรณะรำลึก จังหวัดตรัง

ประกาศนียบัตรวิชาชีพ วิทยาลัยเทคนิคตรัง

ประกาศนียบัตรวิชาชีพชั้นสูง วิทยาลัยเทคนิคตรัง

ปริญญาตรี สาขาวิชาอิเล็กทรอนิกส์และคอมพิวเตอร์
ภาควิชาครุศาสตร์วิศวกรรม
คณะครุศาสตร์อุตสาหกรรม สจล.

คติพจน์ ทำวันนี้ให้ดีที่สุด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้