

สำนักหอสมุดกลาง พระจอมเกล้าลาดกระบัง



ภาควิชาครุศาสตร์วิศวกรรม

คณะครุศาสตร์อุตสาหกรรม

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ใบรับรองปริญญาโท

ชื่อหัวข้อ เครื่องตรวจสอบบุคคลเข้าหน่วยงาน

PERSONAL CHECK IN SYSTEM

ชื่อนักศึกษา

- 1. นายกิตติธัช สุติศา รหัสประจำตัว 41031501
- 2. นายมนัส ตลับเงิน รหัสประจำตัว 41031522
- 3. นางสาววันวิสาข์ พูลคล้าย รหัสประจำตัว 41031529

หลักสูตร ครุศาสตร์อุตสาหกรรมบัณฑิต สาขาวิชา เทคโนโลยีการวัดคุมทางอุตสาหกรรม

อาจารย์ที่ปรึกษา ผศ.วิสุทธิ อธิพรธรรม

อาจารย์ที่ปรึกษาร่วม อาจารย์โกศล ตราชู

คณะกรรมการสอบปริญญาโท	ลายมือชื่อ
1. ผศ.วิสุทธิ อธิพรธรรม	
2. อาจารย์สุรพงษ์ สิริพงษ์ดี	
3. อาจารย์อำพล ทองระอา	
4. อาจารย์สุชิน อางหาญ	
5. อาจารย์สุระชัย พิมพ์สวัสดิ์	

วัน/เดือน/ปีที่สอบ วันศุกร์ที่ 12 พฤษภาคม พ.ศ. 2543 เวลา 12.00 น.

สถานที่สอบ ห้อง ค.310 คณะครุศาสตร์อุตสาหกรรม สจล.

ภาควิชารับรองแล้ว

ลงนาม.....

(ผศ.วิสุทธิ อธิพรธรรม)

หัวหน้าภาควิชาครุศาสตร์วิศวกรรม

วันที่ 23 เดือน ๕ พ.ศ. ๒๕๔๓



เลขหมู่.....
เลขทะเบียน..... 37174
วัน, เดือน, ปี = 5 ก.ย. 2543

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ทางธุรกิจหรือการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ปริญญานิพนธ์

เครื่องตรวจสอบบุคคลเข้าหน่วยงาน
PERSONAL CHECK IN SYSTEM



ปริญญานิพนธ์ฉบับนี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรครุศาสตรบัณฑิต
สาขาวิชาเทคโนโลยีการวัดคุมทางอุตสาหกรรม
ภาควิชาครุศาสตร์วิศวกรรม คณะครุศาสตร์อุตสาหกรรม
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2542

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ปริญญานิพนธ์

เรื่อง เครื่องตรวจสอบบุคคลเข้าสู่หน่วยงาน

Personal Check in System

วัตถุประสงค์ของการทำปริญญานิพนธ์

1. เพื่อศึกษาการติดต่อสื่อสารระหว่างไมโครคอนโทรลเลอร์กับคอมพิวเตอร์ โดยผ่าน RS-232
2. เพื่อออกแบบเครื่องตรวจสอบบุคคลเข้าสู่หน่วยงาน
3. เพื่อสร้างเครื่องตรวจสอบบุคคลเข้าสู่หน่วยงาน
4. เพื่อทดสอบการทำงานของเครื่องตรวจสอบบุคคลเข้าสู่หน่วยงาน
5. เพื่อศึกษาหลักการเขียนโปรแกรม และการใช้งานไมโครคอนโทรลเลอร์ 89C51

ประโยชน์ที่คาดว่าจะได้รับ

1. ได้ความรู้ ความเข้าใจเรื่องการติดต่อสื่อสารระหว่างไมโครคอนโทรลเลอร์กับหน่วยความจำ โดยผ่าน RS-232
2. ได้วงจรต้นแบบของเครื่องตรวจสอบบุคคลเข้าสู่หน่วยงาน
3. ได้เครื่องตรวจสอบบุคคลเข้าสู่หน่วยงานที่ใช้งานได้จริง
4. ได้เครื่องตรวจสอบบุคคลเข้าสู่หน่วยงานที่สามารถนำไปประยุกต์ใช้งานอื่นๆ ที่มีลักษณะใกล้เคียง
5. เข้าใจถึงหลักการเขียนโปรแกรม และการใช้งานไมโครคอนโทรลเลอร์ 89C51

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ชื่อหัวข้อ	เครื่องตรวจสอบบุคคลเข้าหน่วยงาน
นักศึกษา	นายกิตติรัช สุติศา นายมนัส ตลับเงิน นางสาววันวิสาข์ พูลคล้าย
อาจารย์ที่ปรึกษา	ผศ. วิสุทธิ อธิพรธรรม
อาจารย์ที่ปรึกษาร่วม	อาจารย์โกศล ตราชู
หลักสูตร	ครุศาสตร์อุตสาหกรรมบัณฑิต
สาขาวิชา	เทคโนโลยีการวัดคุมทางอุตสาหกรรม
ปีการศึกษา	2542

บทคัดย่อ

ปริญญานิพนธ์ฉบับนี้นำเสนอเครื่องตรวจสอบบุคคลเข้าหน่วยงาน ซึ่งลักษณะของระบบประกอบด้วยหน่วยประมวลผล 2 ส่วนใหญ่ คือ ส่วนประมวลผลของวงจรควบคุม ซึ่งจะเชื่อมโยงกับการอ่านข้อมูลจากบาร์โค้ด หรือคีย์บอร์ด และส่วนประมวลผลของวงจรบันทึกข้อมูลและตรวจสอบข้อมูล ซึ่งหน่วยประมวลผลของทั้งสองส่วนนี้จะทำการติดต่อสื่อสารกันแบบอนุกรม RS-232 ทำให้ขณะใช้งานไม่จำเป็นต้องต่อกับไมโครคอมพิวเตอร์ จึงทำให้สะดวกในการใช้งานและเคลื่อนย้าย อีกทั้งสามารถนำเครื่องตรวจสอบบุคคลเข้าหน่วยนี้ไปประยุกต์ใช้งานกับหน่วยงานอื่นๆ ได้

II

Thesis Title	Personal Check in System
Students	Mr.Kittitouch Suteeca Mr.Manus Talubneon MissWanwisa Poolklai
Advisor	Assist. Prof. Wisuit Atipornatum
Co – Advisor	Mr.Koson Trachoo
Education Level	Bachelor of Science in Industrial Education
Program in	Industrial Instrument Technology
Academic Year	1999

ABSTRACT

This thesis presents the project of personal check in system. The project consists of a control circuit for read data from barcode or keyboard and check-save data circuit. It is communicates data with RS-232 serie type. It is not use microcomputer in running time. It is easy to use and portable and apply in other job.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กิตติกรรมประกาศ

ปริญญาานิพนธ์ฉบับนี้ ได้สำเร็จเป็นรูปเล่มได้ด้วยความช่วยเหลือ ช่วยเหลือ ตลอดทั้ง คำแนะนำแนวความคิดต่าง ๆ อีกทั้งทางด้านสถานที่ เครื่องมือ และอุปกรณ์ต่างๆ จากท่านอาจารย์ ที่ปรึกษาปริญญาานิพนธ์ทั้งสองท่าน ตลอดจะอาจารย์ประจำภาควิชาครุศาสตร์วิศวกรรมทุกท่าน และที่สำคัญที่สุด คือ คณะผู้จัดทำที่ร่วมมือกันทำงาน ฝ่าฝืนอุปสรรคต่างๆ ด้วยความมานะจน สำเร็จ

คณะผู้จัดทำขอกราบขอบพระคุณท่านบุพการี ที่ได้ให้โอกาส ให้ขวัญกำลังใจ และให้การ สนับสนุนในการศึกษามาโดยตลอด รวมทั้งคณาจารย์ และเจ้าหน้าที่ภาควิชาครุศาสตร์วิศวกรรม คณะครุศาสตร์อุตสาหกรรม สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง ที่ได้กรุณา ให้งบประมาณสนับสนุน และความสะดวกในการทำงานครั้งนี้ จึงทำให้ปริญญาานิพนธ์เล่มนี้สำเร็จ ได้ด้วยดี



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ

เรื่อง	หน้า
บทคัดย่อภาษาไทย	I
บทคัดย่อภาษาอังกฤษ	II
กิตติกรรมประกาศ	III
สารบัญ	IV
สารบัญตาราง	VII
สารบัญรูป	VIII
บทที่ 1 บทนำ	1
1.1 ความเป็นมา	1
1.2 ชี้ความสามารถของโครงการ	1
1.3 เนื้อหาโดยสังเขป	1
บทที่ 2 ทฤษฎีและหลักการ	3
2.1 กล่าวนำ	3
2.2 การสื่อสารข้อมูลแบบขนาน	3
2.3 การสื่อสารข้อมูลแบบอนุกรม	4
2.3.1 การสื่อสารข้อมูลแบบอนุกรมตามมาตรฐาน RS-232C	4
2.4 บาร์โค้ด	7
2.5 ชนิดของตัวบาร์โค้ด	8
2.5.1 ไดโอดเปล่งแสง (LED)	9
2.5.2 แสงอินฟราเรด (IR)	9
2.5.3 แสงเลเซอร์ (Laser)	9
2.5.4 ไฟเบอร์ออปติก	9
2.6 การทำงานของบาร์โค้ด	10
2.7 ผลของช่องรับแสง	12
2.8 รูปแบบของรหัสของบาร์โค้ด	13
2.8.1 รหัส 3 ใน 9 หรือรหัส 39	13

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ (ต่อ)

เรื่อง	หน้า
2.8.2 รหัสแท่ง 2 ใน 5	15
2.8.3 รหัสแบบ Codebar	20
2.8.4 รหัสสากล UPC (Universal Product Code)	21
2.8.5 รหัสตัวเลขของยุโรป EAN (European Article Numbering)	22
2.9 LCD โมดูล	25
2.10 การอินเตอร์เฟสกับจอแสดงผล LCD และคีย์บอร์ด	30
2.11 การอินเตอร์เฟสกับจอแสดงผล LCD โมดูล	30
2.12 การเชื่อมต่อและควบคุม	30
2.13 การอินเตอร์เฟสกับคีย์บอร์ด	32
2.13.1 แบบเชื่อมต่อสวิตช์โดยตรงกับพอร์ต	32
2.13.2 แบบเชื่อมต่อสวิตช์แบบเมตริกซ์	33
2.13.3 แบบเชื่อมต่อสวิตช์ผ่านชิปไดรฟ์จิสเตอร์	35
2.13.4 แบบเชื่อมต่อสวิตช์แบบมัลติเพล็กซ์	35
2.13.5 แบบเชื่อมต่อสวิตช์เข้ากับระบบบัสข้อมูล	36
2.14 หน่วยความจำข้อมูล	36
2.14.1 สแตติกแรม (Static RAM)	36
2.14.2 ไดนามิกแรม (Dynamic RAM)	37
2.14.3 การอ่านและเขียนข้อมูลกับหน่วยความจำ	38
2.14.4 การเชื่อมต่อหน่วยความจำกับ MCS-51	39
บทที่ 3 การออกแบบ การสร้าง และการทำงาน	45
3.1 กล่าวนำ	45
3.2 การออกแบบวงจร	46
3.2.1 ส่วนวงจรควบคุม	46
3.2.2 วงจรบันทึกข้อมูลและตรวจสอบข้อมูล	52

สารบัญ (ต่อ)

เรื่อง	หน้า
บทที่ 4 การทดลองและผลการทดลอง	54
4.1 การทดลองส่วนของวงจรควบคุมจากบอร์ด ANTC51M	54
4.1.1 การทดลองส่วนของการแสดงผล	54
4.1.2 การทดลองส่วนของการรับข้อมูลจากเครื่องอ่านบาร์โค้ด	56
4.1.3 การทดลองส่วนรับข้อมูลจากคีย์บอร์ด	57
4.2 การทดลองส่วนของการบันทึกข้อมูลและการตรวจสอบข้อมูล	58
4.3 การทดลองส่วนของการสื่อสารข้อมูล	59
4.3.1 การทดลองการสื่อสารข้อมูลระหว่างวงจรควบคุมกับคอมพิวเตอร์	59
4.3.2 การทดลองรวมส่วนของการสื่อสารข้อมูลผ่าน RS-232 ของวงจรควบคุมกับวงจรบันทึกข้อมูลและตรวจสอบข้อมูล	60
4.3.3 การทดลองเครื่องตรวจสอบบุคคลเข้าหน่วยงาน	61
บทที่ 5 บทสรุป ปัญหา แนวทางแก้ไข และการพัฒนา	68
5.1 สรุป	68
5.2 ปัญหา และแนวทางการแก้ปัญหา	68
5.3 แนวทางการแก้ไข และการพัฒนา	70
ภาคผนวก ก รูปต้นแบบเครื่องตรวจสอบบุคคลเข้าหน่วยงาน	71
ภาคผนวก ข โปรแกรมการทำงาน	73
ภาคผนวก ค วงจรการทำงาน	118
ภาคผนวก ง คู่มือการใช้งาน	121
บรรณานุกรม	127
ประวัติผู้แต่ง	128

VII

สารบัญตาราง

ตาราง	หน้า
ตารางที่ 2.1 การเปรียบเทียบมาตรฐานการสื่อสารข้อมูลของ EIA	6
ตารางที่ 2.2 การเปรียบเทียบการสื่อสารแบบขนานกับแบบอนุกรม	7
ตารางที่ 2.3 ชนิดของตัวอ่านบาร์โค้ด	9
ตารางที่ 2.4 ผลของช่องแสงที่มีต่อแรงดันทางเอาต์พุต	11
ตารางที่ 2.5 โครงสร้างตัวอักษรในรหัส	14
ตารางที่ 2.6 ชุดอักขระของบาร์โค้ดรหัสแทรก 2 ใน 5	16
ตารางที่ 2.7 รูปแบบบาร์โค้ด codebar และค่าระบบตัวเลขฐานสอง	20
ตารางที่ 2.8 ระบบจำนวนของตัวอักขระ	21
ตารางที่ 2.9 ตัวเลขของรหัสสากล	21
ตารางที่ 2.10 ขาสัญญาณของ LCD module	27
ตารางที่ 2.11 ชุดคำสั่งควบคุมข้อความ	29
ตารางที่ 2.12 ค่าลอจิกของสัญญาณ	41

VIII

สารบัญรูป

รูปภาพ	หน้า
รูปที่ 2.1 โครงสร้างของการสื่อสารข้อมูลแบบขนาน	4
รูปที่ 2.2 โครงสร้างของการสื่อสารข้อมูลแบบอนุกรมตามมาตรฐาน RS-232C	5
รูปที่ 2.3 ระดับแรงดันทางเอาต์พุตของตัวอ่านบาร์โค้ด Vws คือ แรงดันเมื่อผ่านแถบขาว และVbs คือ แรงดันเมื่อผ่านแถบลำ	10
รูปที่ 2.4 รูปแบบของการเข้ารหัส IA	12
รูปที่ 2.5 อักขระในบาร์โค้ด 3 ใน 9	13
รูปที่ 2.6 รูปแบบของบาร์โค้ดรหัสแทรก 2 ใน 5	15
รูปที่ 2.7 ผังการจัดวางบาร์โค้ดรหัสแทรก 2 ใน 5	17
รูปที่ 2.8 บาร์โค้ดสมบูรณ์ของจำนวน 0123 ในรูปแบบรหัสแทรก 2 ใน 5	18
รูปที่ 2.9 การเข้ารหัส Codebar “A37859B”	18
รูปที่ 2.10 รูปแบบการเข้ารหัส UPC	19
รูปที่ 2.11 (ก) ผังการจัดวางข้อมูลจริงของบาร์โค้ดรหัส UPC (ข) ฟิลด์ซ้ายและฟิลด์ขวาของข้อมูลตัวอักษร	23 23
รูปที่ 2.12 รูปบาร์โค้ดรหัส EAN	24
รูปที่ 2.13 การต่อแบบ Memory map	26
รูปที่ 2.14 การต่อแบบ I/O port	28
รูปที่ 2.15 การเชื่อมต่อระหว่าง LCD โมดูลและบอร์ด MCS-51	32
รูปที่ 2.16 การเชื่อมต่อสวิตช์แบบต่างๆ	33
รูปที่ 2.17 การเก็บข้อมูลแต่ละบิตของสแตติกแรม	37
รูปที่ 2.18 โครงสร้างภายในสแตติกแรม	37
รูปที่ 2.19 กลุ่มสัญญาณที่ใช้ในการติดต่อหน่วยความจำ	38
รูปที่ 2.20 สัญญาณในการอ่านข้อมูล	39
รูปที่ 2.21 แสดงสัญญาณในการเขียนข้อมูล	39
รูปที่ 2.22 สายต่อวงจรทางพอร์ต 0 และ พอร์ต 2	40

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูป (ต่อ)

รูปภาพ	หน้า
รูปที่ 2.23 การต่อ MCS-51 กับหน่วยความจำ	40
รูปที่ 2.24 การจัดเนื้อที่หน่วยความจำ	42
รูปที่ 2.25 แผนผังเวลาการอ่านหน่วยความจำข้อมูลภายนอก	42
รูปที่ 2.26 แผนผังเวลาการเขียนหน่วยความจำข้อมูลภายนอก	43
รูปที่ 2.27 การเชื่อมต่อหน่วยความจำข้อมูลภายนอก	43
รูปที่ 2.28 การต่อหน่วยความจำข้อมูลภายนอกกับ MCS-51	44
รูปที่ 3.1 โครงสร้างของเครื่องตรวจสอบบุคคลเข้าหน่วยงาน	45
รูปที่ 3.2 รูปวงจรควบคุมจาก ANT-C51M	47
รูปที่ 3.3 ชุดเครื่องอ่านบาร์โค้ด	48
รูปที่ 3.4 วงจรรับข้อมูลจากบาร์โค้ด	49
รูปที่ 3.5 คีย์เมทริกซ์สวิตช์	50
รูปที่ 3.6 ส่วนรับข้อมูลเมทริกซ์สวิตช์	51
รูปที่ 3.7 ลักษณะการแสดงผลของ LCD	51
รูปที่ 3.8 วงจรบันทึกข้อมูลและตรวจสอบข้อมูล	53
รูปที่ 4.1 บอร์ดวงจรควบคุมที่สามารถทำงานได้	55
รูปที่ 4.2 การทดลองนำข้อมูลแสดงผลที่ LCD	55
รูปที่ 4.3 การต่อวงจรควบคุมอุปกรณ์แสดงผลและเครื่องอ่านบาร์โค้ด	56
รูปที่ 4.4 ผลของอุปกรณ์แสดงผลเมื่อมีการกดคีย์บอร์ด	57
รูปที่ 4.5 วงจรบันทึกข้อมูลและการตรวจสอบข้อมูล	58
รูปที่ 4.6 การทดลองวงจรสื่อสารข้อมูลของวงจรควบคุมกับคอมพิวเตอร์	59
รูปที่ 4.7 การทดลองสื่อสารผ่าน RS-232 ของวงจรควบคุมกับวงจรบันทึกข้อมูล	61
รูปที่ ก.1 แผงวงจรรวมเครื่องตรวจสอบบุคคลเข้าหน่วยงาน	72
รูปที่ ก.2 รูปสำเร็จของเครื่องตรวจสอบบุคคลเข้าหน่วยงาน	72
รูปที่ ข.1 แผนผังการทำงานของเครื่องตรวจสอบบุคคลเข้าหน่วยงาน	111
รูปที่ ค.1 วงจรควบคุมจาก ANT-C51M	119
รูปที่ ค.2 วงจรบันทึกข้อมูลและตรวจสอบข้อมูล	120

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 1

บทนำ

1.1 ความเป็นมา

เนื่องจากปัจจุบันนี้ คณะครุศาสตร์อุตสาหกรรมมีนักศึกษา คณาจารย์ เจ้าหน้าที่และบุคลากรภายนอกเข้า และออกในขณะทั้งในเวลาและนอกเวลาราชการ ทำให้ยามรักษาความปลอดภัยดูแลไม่ทั่วถึง เกิดสิ่งของสูญหาย และหาบุคคลผู้รับผิดชอบไม่ได้

ดังนั้น เพื่อเป็นการป้องกันสิ่งเหล่านั้นไม่ให้เกิดขึ้นอีก หรือเพื่อเป็นการลดความเสี่ยงที่เกิดจากผู้ที่ไม่ประสงค์ต่อทรัพย์สินทั้งของคณะ จึงได้เกิดแนวคิดในการสร้างเครื่องตรวจสอบบุคคลเข้าหน่วยงานนี้ขึ้นมา เพื่อช่วยรักษาความปลอดภัยในทรัพย์สินของคณะ อีกทั้งยังช่วยในการสืบสวนหาบุคคลที่ต้องสงสัยได้ง่ายขึ้น และสามารถนำเสนอข้อมูลที่บันทึกไว้ออกมายังอุปกรณ์แสดงผลได้

1.2 ขีดความสามารถของโครงการ

1. สามารถป้อนรหัสด้วยบาร์โค้ด หรือป้อนรหัสด้วยคีย์ตัวเลข ได้
2. สามารถแสดงสถานะของผู้ใช้ได้
3. สามารถบันทึกเวลา วัน เดือน ปี ที่เข้าและออก และรหัสของผู้ใช้ได้
4. สามารถแสดงเวลา วัน เดือน ปี ได้

1.3 เนื้อหาโดยสังเขป

ปฏิญานิพนธ์ฉบับนี้มีเนื้อหาที่ทำการเสนอทั้งหมด 5 บท ดังต่อไปนี้

บทที่ 2 ทฤษฎี และหลักการ เป็นเนื้อหาที่เกี่ยวกับทฤษฎี และหลักการของการสื่อสารแบบอนุกรมและขนาน , หลักการทำงานของบาร์โค้ด , การสร้างรหัสบาร์โค้ด , ชนิดของบาร์โค้ด , การแสดงผลโดยจอ LCD , การติดต่อกับคีย์บอร์ด และหน่วยความจำ

บทที่ 3 การออกแบบ การสร้าง และการทำงาน กล่าวถึงขั้นตอนในการออกแบบฮาร์ดแวร์ และซอฟต์แวร์ รวมทั้งการนำส่วนต่างๆ มาทำการอินเทอร์เฟสกัน เพื่อให้สามารถทำงานร่วมกันได้

บทที่ 4 การทดลอง และผลการทดลอง เป็นส่วนที่ทำการทดสอบการทำงานของเครื่องตรวจสอบบุคคลเข้าหน่วยงาน โดยแบ่งเป็น 3 ส่วนใหญ่ๆ คือส่วนของวงจรควบคุม , ส่วนเอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปโฆษณาตามการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ของวงจรมันทีก็ข้อมูล และตรวจสอบข้อมูล และส่วนของการติดต่อสื่อสารข้อมูล ตลอดจนผลการทดลองของแต่ละส่วนว่าตรงตามวัตถุประสงค์ที่ตั้งไว้หรือไม่

บทที่ 5 บทสรุป ปัญหา แนวทางแก้ไขและพัฒนา กล่าวสรุปผลการทำโครงการทดลองจนปัญหาที่เกิดขึ้นระหว่างการดำเนินงาน ตลอดจนแนวทางแก้ไขปัญหาที่เกิดขึ้น รวมทั้งข้อเสนอแนะ เพื่อเป็นแนวทางในการพัฒนาเครื่องตรวจสอบบุคคลเข้าหน่วยงาน ให้สามารถนำเครื่องตรวจสอบบุคคลเข้าหน่วยงาน ไปประยุกต์ใช้งานได้

ภาคผนวก ก รูปต้นแบบเครื่องตรวจสอบบุคคลเข้าหน่วยงาน

ภาคผนวก ข โปรแกรมการทำงาน

ภาคผนวก ค วงจรการทำงาน

ภาคผนวก ง คู่มือการใช้งาน



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 2

ทฤษฎีและหลักการ

2.1 กล่าวนำ

เมื่อก้าวถึงเรื่องการสื่อสารเกี่ยวกับระบบไมโครคอนโทรลเลอร์ ส่วนใหญ่จะคุ้นเคยและรู้จัก RS - 232 เป็นอย่างดี ซึ่งมักเรียกว่า การสื่อสารแบบอนุกรม (Serial communication) ปัจจุบันระบบคอมพิวเตอร์ใช้งานธุรกิจทั่วไป (PC) จะมีพอร์ต RS-232 มาให้ เพื่อใช้ในการติดต่อสื่อสารแบบอนุกรม และยังสามารถต่อเข้ากับ Modem เพื่อการรับส่งข้อมูลทางโทรศัพท์ นอกจากนี้ยังสามารถต่อเข้ากับอุปกรณ์ใช้งานประเภทต่างๆ ได้อีก เช่น ใช้กับ Mouse และสามารถใช้งานกับไมโครคอนโทรลเลอร์ เป็นต้น

RS - 232 เป็นที่นิยมใช้งานกันมาก เนื่องจาก

1. ความเรียบง่าย หมายถึง การใช้จำนวนสายเพียง 3 เส้น เท่านั้น (เรียกว่า การต่อแบบ Local connecting) สามารถสื่อสารได้แบบ Full duplex ถึงแม้จะกล่าวกันว่า การส่งแบบอนุกรมจะช้ากว่าการส่งแบบขนาน ซึ่งก็เป็นจริงอยู่ตามหลักการ แต่ความเร็วของอนุกรมยังเร็วมากในแง่ของการใช้งานจริง ความเรียบง่ายของจำนวนสาย ซึ่งเป็นข้อดีของ RS - 232 เป็นอย่างมาก ซึ่งทำให้ทุกอย่างที่ตามมาเรียบง่ายไปด้วย ไม่ว่าจะเป็นขั้วต่อ ราคาของสาย และความยาวของสายที่ต่อได้ไกลกว่าแบบขนานด้วย

2. สามารถทำการสื่อสารแบบ FULL DUPLEX หมายถึง ความสามารถในการสื่อสารแบบโต้ตอบกันได้ในเวลาเดียวกัน ซึ่งลักษณะนี้จะตอบสนองการทำงาน ในแบบที่เรียกว่า Interactive ได้เป็นอย่างดี เช่น การใช้ภาษา BASIC-52 ของชิพเบอร์ 8052 อีกทั้งยังรองรับการใช้งานอื่นๆ ในทำนองเดียวกันได้

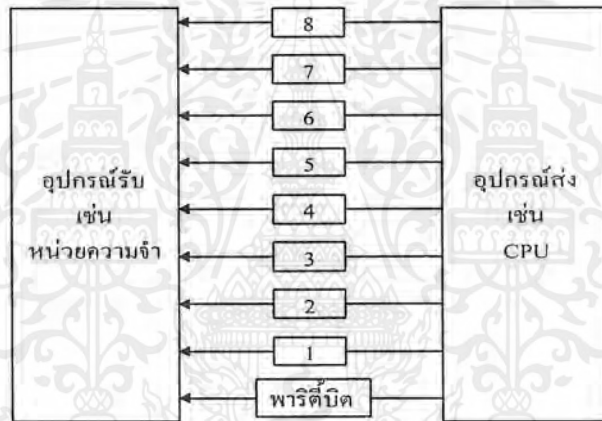
3. มีความยืดหยุ่นสูง เนื่องจาก RS-232 สามารถกำหนดพารามิเตอร์ต่างๆ ได้ในการสื่อสารได้มากมาย เช่น Baud rate , Parity , Data bit เป็นต้น จึงทำให้มีความยืดหยุ่นสูง เช่น สมมุติว่าเกิดปัญหาเรื่องความยาวของสายคือสายของการติดต่อสื่อสารสั้นไป ดังนั้นสามารถลด Baud rate ลงเพื่อให้ได้ความยาวของสายมากขึ้น เพื่อเป็นการประหยัดเวลาในการติดต่อสื่อสาร

2.2 การสื่อสารข้อมูลแบบขนาน

ลักษณะของการสื่อสารข้อมูลแบบขนานนั้น จะเป็นการสื่อสารข้อมูลที่ข้อมูลจะรับ-ส่ง โดยผ่านสาย หรือช่องสัญญาณพร้อมกันหลายๆ เส้น ดังแสดงในรูปที่ 2.1 โดยที่จำนวนของ

สัญญาณจะมีจำนวนไม่แน่นอน ขึ้นอยู่กับโครงสร้างการประมวลผลข้อมูลของระบบนั้นๆ ข้อดีของการสื่อสารข้อมูลแบบนี้ คือ สามารถสื่อสารข้อมูลกันได้ในระยะเวลาดังๆ แต่ ก็มีข้อเสีย คือ จะสิ้นเปลืองสายสัญญาณเป็นจำนวนมาก และถ้ายังใช้ในการสื่อสารข้อมูลในระยะทางไกลๆ นอกจากจะสิ้นเปลืองค่าใช้จ่ายจำนวนมากแล้ว ยังทำให้สัญญาณถูกลดทอนลงไปด้วย

ดังนั้น โดยทั่วไปแล้วการสื่อสารข้อมูลแบบขนานนิยมนำไปใช้กับการสื่อสารข้อมูลในระยะเวลาดังๆ ที่ต้องการสื่อสารข้อมูลด้วยอัตราเร็ว เช่น การเชื่อมต่อของสัญญาณระหว่างหน่วยประมวลผลกลางกับอุปกรณ์รอบข้าง หรือการสื่อสารข้อมูลของเครื่องมือวัดและอุปกรณ์ต่างๆ กับเครื่องไมโครคอมพิวเตอร์



รูปที่ 2.1 โครงสร้างของการสื่อสารข้อมูลแบบขนาน

2.3 การสื่อสารข้อมูลแบบอนุกรม

2.3.1 การสื่อสารข้อมูลแบบอนุกรมตามมาตรฐาน RS-232C

การสื่อสารข้อมูลแบบอนุกรมที่มีการใช้งานอยู่ในปัจจุบัน ได้มีการกำหนดมาตรฐานในการรับ-ส่งข้อมูลไว้หลายแบบด้วยกัน ที่ได้รับความนิยมนำมาใช้งานอย่างกว้างขวางคือ การสื่อสารข้อมูลแบบอนุกรมตามมาตรฐาน RS-232C

ทั้งนี้เนื่องจากความต้องการในการสื่อสารข้อมูลผ่านทางเครือข่ายโทรศัพท์ที่มีมากขึ้นเรื่อยๆ ดังนั้น จึงได้มีการกำหนดมาตรฐานที่เรียกว่า RS-232C ขึ้น เพื่อใช้เป็นมาตรฐานแก่อุปกรณ์ที่ถูก

ผลิตขึ้นจากบริษัทต่างๆ ในสหรัฐอเมริกา Bell System operating telephone companies เป็นบริษัทหลักบริษัทแรกที่เป็นผู้ผลิตและติดตั้งระบบสื่อสารข้อมูล และเป็นผู้ผลิตอุปกรณ์ต่างๆ ที่ใช้ในการอินเตอร์เฟซอุปกรณ์ดิจิทัลกับเครือข่ายโทรศัพท์รายใหญ่ อุปกรณ์นี้คือ Bell modem ซึ่งถูกพัฒนาโดย Bell Laboratories และถูกใช้เป็นมาตรฐานในงานอุตสาหกรรมจนถึงปัจจุบันนี้ ดังนั้น ความต้องการข้อมูลเกี่ยวกับข้อกำหนดในการอินเตอร์เฟซกับโมเด็มจึงมีเพิ่มขึ้นเรื่อยๆ เพื่อตอบสนองต่อความต้องการนี้ EIA(ELECTRONIC INDUSTRIAL ASSOCIATE), Bell System และผู้ผลิตโมเด็มรายอื่นๆ จึงได้ร่วมมือกันตั้งมาตรฐาน RS-232C ขึ้น

มาตรฐาน RS-232C ได้ถูกตีพิมพ์โดย EIA ในปี ค.ศ. 1969 โดยตัวอักษร RS แทน "Recommended Standard" 232 แทนหมายเลขของมาตรฐาน ส่วนอักษร C แสดงให้เห็นว่ามาตรฐานนั้นได้รับการแก้ไขกี่ครั้ง

การที่มาตรฐาน RS - 232C นี้เป็นที่นิยมใช้ เนื่องจากเป็นระบบการสื่อสารข้อมูลที่ใช้ในเครื่องไมโครคอมพิวเตอร์ IBM PC ซึ่งเป็นคอมพิวเตอร์ที่มีใช้อย่างแพร่หลายตั้งแต่อดีตมาจนถึงปัจจุบัน

มาตรฐาน RS-232C จะมีโครงสร้างการสื่อสารเป็นแบบจุดต่อจุดเท่านั้น โดยมีลักษณะคุณสมบัติทางไฟฟ้า และทางกายภาพ ดังแสดงในตารางที่ 2.1 และรูปที่ 2.2



รูปที่ 2.2 โครงสร้างของการสื่อสารข้อมูลแบบอนุกรมตามมาตรฐาน RS-232C

ตารางที่ 2.1 การเปรียบเทียบมาตรฐานการสื่อสารข้อมูลของ EIA

พารามิเตอร์	RS - 232A	RS - 423A	RS - 422A	RS - 485
โหมดการทำงาน	Single-ended	Single-ended	Differential	Differential
จำนวนของตัวรับและตัวส่งที่ยอมรับได้	1 ตัวส่ง 1 ตัวรับ	1 ตัวส่ง 10 ตัวรับ	1 ตัวส่ง 10 ตัวรับ	32 ตัวส่ง 32 ตัวรับ
ความยาวของคู่สายสัญญาณรับส่งข้อมูล	50 ฟุต	4000 ฟุต	4000 ฟุต	4000 ฟุต
อัตราการส่งข้อมูลสูงสุด (bit / sec)	20 k	100 k	10 M	10 M
แรงดันไฟฟ้าโหมคร่วมสูงสุด	± 2.5 V	± 6 V	+ 6 V -2.5 V	+12 V -7 V
Driver output	ต่ำสุด ± 5 V สูงสุด ± 15 V	ต่ำสุด ± 3.6 V สูงสุด ± 6 V	ต่ำสุด ± 2 V	ต่ำสุด ± 1.5 V
Driver Load (w)	3 k ถึง 7 k	ต่ำสุด 450	ต่ำสุด 100	ต่ำสุด 60
Driver slew rate	30 V/ μ S สูงสุด		NA	NA
กระแสสูงสุดเมื่อเอาต์พุตลัดวงจร	500 mA ลัดวงจรกับ Vcc หรือ GND	150 mA ลัดวงจรกับ GND	150 mA ลัดวงจรกับ GND	150 mA ลัดวงจรกับ GND 250 mA ลัดวงจรกับ 8V หรือ 12V
ค่าความต้านทานเอาต์พุตของตัวส่ง (w)	NA - power ON 300 - power off	NA - power ON 60 k - power off	NA - power ON 60 k - power off	120 k power on, off
ค่าความต้านทานอินพุตของตัวรับ (w)	3 k ถึง 7 k	4 k	4 k	12 k
ความไวตัวรับ	± 3 V	± 200 mV	± 200 m V	± 200 mV

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 2.2 การเปรียบเทียบการสื่อสารแบบขนานกับแบบอนุกรม

รูปแบบการสื่อสารข้อมูล	แบบขนาน	แบบอนุกรม
1. ระยะทาง	ปกติจะน้อยกว่า 100 ฟุต	ส่งได้ตั้งแต่ระยะทางสั้น ๆ ไปจนถึงระยะทางเป็นไมล์
2. ความเร็ว	อัตราความเร็วสูงมาก ในระยะที่ไม่ไกลนัก กำหนดได้เป็น บิต/วินาที	อัตราความเร็วของข้อมูลที่ใช้กันทั่วไป คือ ช่วง 0 ถึง 2 ล้านบิต/วินาที
3. ระดับของสัญญาณ	ในการอินเตอร์เฟสจะใช้ระดับของสัญญาณที่ใช้กับอุปกรณ์ (TTL) คือ สัญญาณลอจิก 1 และ 0 จะแทนด้วยระดับแรงดัน 5 V และ 0 V ตามลำดับ	ใช้มาตรฐานของ EIA RS-232C คือ มีระดับสัญญาณไฟฟ้าขนาด 12V หรืออาจจะใช้มาตรฐาน 20 mA current loop หรืออาจจะใช้ระดับสัญญาณ (TTL) ก็ได้
4. ความผิดพลาดของสัญญาณ	ถ้าส่งในระยะทางที่ไกล ความผิดพลาดของข้อมูลจะเกิดขึ้นง่าย	การผิดพลาดของสัญญาณจะมีน้อยลง
5. ค่าใช้จ่าย	ถ้าส่งในระยะทางที่ไกลๆ จะสิ้นเปลืองค่าใช้จ่ายมาก เพราะต้องใช้สายส่งสัญญาณหลายเส้น	สิ้นเปลืองน้อยกว่าหลายเท่า ถึงแม้ว่าจะใช้อุปกรณ์เปลี่ยนสัญญาณข้อมูลจากแบบขนานไปเป็นแบบอนุกรมแล้วส่งผ่านสายส่งใช้อุปกรณ์ ในการแปลงสัญญาณกลับมาเป็นขนานอีก ก็ยังลงทุนน้อยกว่า

2.4 บาร์โค้ด

ในระบบที่มีการปฏิบัติการงานอัตโนมัติที่มีจำนวนมากๆ เครื่องจักรจะถูกนำมาใช้เพื่ออำนวยความสะดวก โดยการเปลี่ยนรูปแบบข้อมูล จากแบบเดิมที่มนุษย์เข้าใจมาเป็นรูปแบบของรหัสแทน ซึ่งรหัสนี้อาจใช้แทนตัวแปรเดียว หรือหลาย ๆ ตัวก็ได้ สำหรับงานที่แตกต่างกันไป

บาร์โค้ด คือ รหัสที่ใช้แทนสิ่งเหล่านั้น ในรูปของเครื่องจักรที่อ่านรหัสแถบสีดำ และช่องว่างสีขาวในอัตราส่วนที่กำหนด ซึ่งจะแทนอักขระแต่ละตัว โดยมีเซนเซอร์เป็นตัวอ่านความหมายจากแถบนั้นออกมา เพื่อประมวลผลต่อไปในขั้นตอนของสัญญาณทางไฟฟ้า กรรมวิธีในการทำงานเอกสารนี้เป็นเอกสารที่ส่งวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

นี้อาจเปรียบเทียบกับกระบวนการทำงานในร่างกายมนุษย์ คือ มีสายตาเป็นตัวตรวจจับ และสมองเป็นตัวประมวลผล หรือสั่งงาน

บาร์โค้ดจัดเป็นรูปแบบการใช้งานที่ง่ายที่สุด รวมทั้งราคา และความน่าเชื่อถือได้ นับว่าเหมาะสมที่สุดที่จะใช้งานกับระบบข้อมูลของคอมพิวเตอร์ ตัวอย่างการนำบาร์โค้ดไปใช้งานและช่วยงานได้มากคือ ระบบไปรษณีย์อัตโนมัติ โดยการนำไปใช้คัดเลือกรับของจดหมาย และปลายทางที่จะส่งไป

ชนิดของตัวตรวจจับบาร์โค้ดต่างๆ ไปแบ่งได้ 2 ประเภท คือ ชนิดมือถือ และแบบที่ตั้งอยู่กับที่ สำหรับแบบมือถือนั้น ผู้ปฏิบัติงานฝึกหัดเพียงเล็กน้อยสามารถที่จะทำงานได้ และสามารถที่จะทำงานได้รวดเร็ว และถูกต้องกว่าการใช้งานใช้คนป้อนข้อมูลมาก อีกทั้งการเปลี่ยนแปลงรูปแบบบาร์โค้ดในผลิตภัณฑ์นั้นก็ง่ายมาก มีข้อมูลที่น่าสนใจ เปรียบเทียบให้เห็นถึงความผิดพลาดซึ่งเกิดจากการใช้บาร์โค้ดจะมีแค่ 1 ใน 10,000 ในขณะที่หากใช้คนป้อนข้อมูลความผิดพลาดจะสูงถึง 1 ใน 300 อัตราการผิดพลาดที่ได้จากการใช้บาร์โค้ดสามารถลดลงได้โดยใช้วิธีการตรวจสอบตัวเลข และเทคนิคการป้องกันข้อมูลในรูปแบบอื่นร่วมด้วย

ความแตกต่างของรูปแบบต่างๆ ของบาร์โค้ดทุกวันนี้มีอยู่มากมาย ขึ้นอยู่กับความเหมาะสมในงานแต่ละชนิดไป โดยจะขึ้นอยู่กับตัวถอดรหัสของบาร์โค้ด ซึ่งจะต้องตรงกับชนิดของบาร์โค้ด นอกเหนือจากที่จะต้องมีการมีระบบของแหล่งกำเนิดแสง และตัวอ่านตามปกติแล้ว การแสดงผลจะแสดงเอาต์พุตออกทางตัวแสดงผล LED หรือต่อไปยังอินพุตของระบบคอมพิวเตอร์ โดยการต่อผ่านทางพอร์ต RS-232 มาตรฐาน

การใช้แสงเลเซอร์ในการอ่านบาร์โค้ดเพิ่งจะมีมาได้เมื่อต้นทศวรรษที่ 70 โดยใช้งานร่วมกับระบบไมโครโปรเซสเซอร์ ซึ่งการประมวลผลจากตัวอ่านนี้หากว่ามีความสามารถในการอ่านรูปแบบรหัสของบาร์โค้ดที่แตกต่างกันจะมีความยืดหยุ่นในการใช้งานมากกว่า โดยเฉพาะงานที่ต้องใช้รูปแบบรหัสมากกว่า 1 รูปแบบในสายงานการผลิต ยกตัวอย่างในอุตสาหกรรมเวชภัณฑ์จะใช้รหัส UPC เป็นหลัก ในขณะที่ใช้รหัส 39 สำหรับการใช้งานรูปแบบใหม่ๆ

2.5 ชนิดของตัวบาร์โค้ด

ตัวบาร์โค้ดพื้นฐานมี 4 ชนิดดังแสดงในตารางที่ 2.3 ซึ่งจะบอกตั้งแต่ราคาต่อหน่วย และลักษณะการทำงาน สำหรับรายละเอียดของตัวอ่านแต่ละชนิด มีดังต่อไปนี้

ตารางที่ 2.3 ชนิดของตัวอ่านบาร์โค้ด

ชนิดของตัวอ่านบาร์โค้ด	แหล่งกำเนิดแสง	ราคาต่อหน่วย
LED	ไดโอดเปล่งแสงสีแดง	ต่ำ
IR	แสงย่านอินฟราเรด	ปานกลาง
แสงแบนด์แคบ	แสงเลเซอร์	สูงสุด
ไฟเบอร์ออปติก	ใช้แสงจากสภาพแวดล้อมภายนอก	สูง

2.5.1 ไดโอดเปล่งแสง (LED)

ตัวอ่านแบบไดโอดเปล่งแสงจะมีราคาต่อหน่วยต่ำ แต่การทำงานนั้นอาจถูกรบกวนโดยแสงสว่างจากสภาพแวดล้อมได้ การใช้งานตัวอ่านต้องสัมผัสกับวัตถุที่จะอ่าน และฉลากบาร์โค้ด หากเกิดความสกปรกจะทำให้ประสิทธิภาพในการอ่านข้อมูลลดลง โดยทั่วไปแล้วความลึกในการฉายแสง (depth of field) จะมีค่าอยู่ในระดับ 0.075 นิ้ว

2.5.2 แสงอินฟราเรด (IR)

ตัวอ่านชนิดนี้จะคล้ายกับไดโอดเปล่งแสง แต่มีข้อดีกว่า คือ มีผลรบกวนจากแสงสว่างจากสภาพแวดล้อมรอบ ๆ น้อยมาก และไม่มีปัญหาในการอ่านเนื่องมาจากความสกปรกของผิวบาร์โค้ด

2.5.3 แสงเลเซอร์ (Laser)

ปัจจุบันนี้ตัวอ่านที่ใช้แสงเลเซอร์จะมีราคาแพงที่สุดในบรรดาตัวอ่านบาร์โค้ด มีความลึกในการฉายแสงอยู่ในระดับ 3 นิ้ว ซึ่งหมายความว่าสามารถใช้งานอ่านบาร์โค้ดที่ระยะไกลออกไปได้ ตัวอ่านไม่จำเป็นต้องติดกับผิวป้ายบาร์โค้ด โดยทั่วไปจะใช้แสงที่มีความยาวคลื่นประมาณ 750 นาโนเมตร ซึ่งเป็นแสงที่อยู่ในช่วงที่สายตาไม่สามารถมองเห็นได้ แต่การทำงานจะมี LED สีแดงช่วยเล็งหาเป้าหมายขณะทำงาน

2.5.4 ไฟเบอร์ออปติก

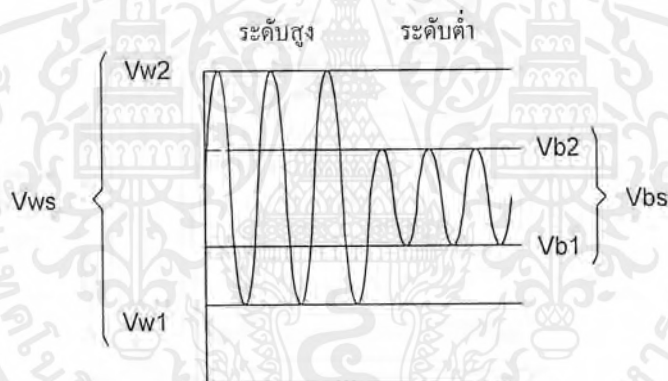
ใช้แสงสว่างจากสิ่งแวดล้อมมีราคาต่อหน่วยอยู่ในระดับค่อนข้างสูง แต่น้อยกว่าแบบเลเซอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.6 การทำงานของบาร์โค้ด

ตัวอ่านบาร์โค้ด หรือสแกนเนอร์เป็นอุปกรณ์ที่ใช้ในการตรวจจับบาร์โค้ด ซึ่งจะให้อาต์พุตเป็นสถานะต่ำเมื่อพบแถบสีดำ และมีสถานะสูงเมื่อพบสีขาว รูปแบบของสัญญาณ ดังรูปที่ 2.3 ซึ่งทำหน้าที่เหมือนตาในระบบบาร์โค้ด โดยการเปลี่ยนแถบเส้นขาวดำที่เห็นให้เป็นสัญญาณทางไฟฟ้า ส่วนประกอบหลัก ๆ ที่ต้องนำมาพิจารณาในการตัดสินใจเลือกชนิดของตัวอ่านบาร์โค้ด คือ

1. รูปแบบของสัญญาณทางเอาต์พุตที่ต้องการว่าเป็นแอนะล็อกหรือดิจิทัล
2. ชนิดของตัวอ่าน หรือชนิดของแสงที่ใช้อ่าน
3. ตัวอ่านเป็นชนิดที่ต้องสัมผัส หรือไม่สัมผัสกับผิวของวัตถุ
4. ตัวอ่านอยู่กับที่ หรือสามารถเคลื่อนย้ายได้
5. สิ่งแวดล้อมในบริเวณที่ใช้งานว่ามีสภาพแสงรบกวนต่อการทำงานหรือไม่



รูปที่ 2.3 ระดับแรงดันทางเอาต์พุตของตัวอ่านบาร์โค้ด V_{ws} คือ แรงดันเมื่อผ่านแถบขาว และ V_{bs} คือ แรงดันเมื่อผ่านแถบดำ

การเลือกตัวอ่านที่เหมาะสมนั้นขึ้นอยู่กับความเข้ากันได้ทั้งหมดของการประยุกต์ใช้งาน เช่น รูปแบบของป้าย หรือฉลาก ตัวถอดรหัส (decoder) และระบบทั้งหมด

การตัดสินใจว่าจะเลือกเอาต์พุตให้เป็นแอนะล็อก หรือดิจิทัลจะต้องพิจารณาด้านความต้องการทางอินพุตของส่วนถอดรหัสข้อมูล และชนิดของแสงที่ใช้อ่านจะขึ้นอยู่กับแสงของสภาพแวดล้อมภายนอกประเภทของการนำไปใช้งาน และชนิดของป้ายบาร์โค้ดที่มีตัวอ่านเป็นอินฟราเรดสามารถอ่านป้ายที่มีความสกปรก ซึ่งความสกปรกนี้จะพบได้บ่อยในบริเวณการบรรจุหีบห่อสินค้า

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ชนิดของหมึกที่ใช้พิมพ์ป้ายบาร์โค้ดจะต้องสัมพันธ์กับแสงที่ใช้ในตัวอ่านด้วยส่วนการเลือกตัวอ่านเป็นชนิดอยู่หนึ่ง หรือเคลื่อนที่ รวมทั้งสัมพันธ์กับผิว หรือไม่ขึ้นอยู่กับประเภทของการใช้งาน

หลังจากได้ข้อมูลออกจากสแกนเนอร์ หรือตัวอ่านบาร์โค้ดแล้ว สัญญาณจะถูกส่งต่อมายังส่วนประมวลผลข้อมูลเพื่อแปลความหมาย โดยทำการเปรียบเทียบสัญญาณทางไฟฟ้าที่ได้จากการอ่านสัญญาณบาร์ และสเปซ สิ่งที่ใช้พิจารณาการเลือกตัวถอดรหัสแปลผลสำหรับบาร์โค้ด คือ

1. ความเข้ากันได้กับชนิดของตัวอ่าน
2. สัญญาณทางเอาต์พุตที่ต้องการ
3. ต้องการชนิดที่สามารถเคลื่อนย้ายได้หรือไม่
4. แสดงผลเพียงอย่างเดียว หรือจะให้พิมพ์ข้อมูลออกมาด้วย
5. ต้องการคีย์ในการป้อนข้อมูลหรือไม่
6. ความสามารถในการถอดรหัสบาร์โค้ดได้หลายรูปแบบหรือไม่
7. ปัจจัยการใช้งานทางด้านสภาพแวดล้อม

ตารางที่ 2.4 ผลของช่องแสงที่มีต่อแรงดันทางเอาต์พุต

ขนาดช่องแสง (นิ้ว)	ค่าต่ำสุด (มิลลิโวลต์)	ค่าปกติ (มิลลิโวลต์)
แรงดันจากแถบกว้าง		
0.004	100	150
0.006	200	300
0.008	400	600
0.010	620	930
0.012	900	1350
แรงดันจากแถบแคบ		
0.004	50	90
0.006	100	180
0.008	200	360
0.010	310	558
0.012	450	810

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.7 ผลของช่องรับแสง

ตัวอ่านบาร์โค้ดจะมีช่องแสงเป็นทางให้แสงสะท้อนจากผิวบาร์โค้ดผ่านส่วนตรวจจับ และช่องแสงนี้ต้องมีขนาดเล็กกว่าความกว้างของแถบบาร์โค้ด แรงดันที่เอาต์พุตจะขึ้นอยู่กับขนาดของช่องแสงดัง แสดงในตารางที่ 2.4

ค่าพารามิเตอร์ที่สำคัญอีกค่าหนึ่งของตัวอ่านบาร์โค้ด คือ ค่าของดัชนีความละเอียด (Resolution Index หรือ RI) อัตราส่วนของสัญญาณแถบแคบ VD (Narrowbar Signal) หารด้วยค่าของสัญญาณแถบกว้าง VS (Widebar Signal) โดยที่ค่าของสัญญาณแถบแคบ VD คือ ระดับของสัญญาณไฟฟ้าที่อ่านผ่านแถบที่แคบที่สุดของบาร์โค้ด และสัญญาณแถบกว้าง VS คือ ค่าความแตกต่างระหว่างขนาดของสัญญาณที่ได้จากบริเวณแถบดำ และแถบขาวของป้ายบาร์โค้ดสัมพันธ์กับสัญญาณที่สร้าง โดยความกว้างของบาร์ และสเปซหาได้โดยใช้สมการต่อไปนี้

$$RI = \frac{VD}{VS} \times 100\%$$

เมื่อ

$$VS = V_{w2} - V_{w1}$$

$$VD = V_{n1} - V_{n1}$$

ช่วงสิ้นสุด อักขระสุดท้าย | A อักขระสุดท้าย ช่วงสิ้นสุด



รูปที่ 2.4 รูปแบบของการเข้ารหัส IA

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ซึ่งการวัดอ้างอิงค่าตามมาตรฐาน NBS (National Bureau of Standard) สัญญาณที่วัดได้จากตัวอ่านบาร์โค้ดจะขึ้นอยู่กับความเร็วในการสแกนผ่านไปบนป้ายเวลาขาขึ้น (Rise Time) และเวลาขาลง (Fall Time) ของสัญญาณที่อ่านจะอยู่ระหว่าง 10 เพอร์เซ็นต์ และ 90 เพอร์เซ็นต์ ของสัญญาณภายในเวลา 40 ไมโครวินาที

2.8 รูปแบบของรหัสของบาร์โค้ด

ความแตกต่างของรหัสตัวอักษรบาร์โค้ดที่ได้รับการพัฒนาขึ้นมา ใช้งานในทุกวันนี้มีลักษณะรูปแบบมากมาย ซึ่งจะเปลี่ยนแปลงไปขึ้นอยู่กับรูปแบบการตรวจสอบความผิดพลาด ความหนาแน่นในการพิมพ์ข้อมูล ตัวอักษรตอนนี้ ชนิดของตัวอักษรที่ใช้งาน ไม่ว่าจะเป็นตัวอักษรหรือว่าตัวเลข ซึ่งสามารถนำมาเข้ารหัส และประยุกต์ใช้งานจริงได้ รูปแบบของรหัสบาร์โค้ดที่ใช้กันในทุกวันนี้มี 5 แบบ หลัก ๆ ดังนี้

2.8.1 รหัส 3 ใน 9 หรือรหัส 39

รหัส 39 ประกอบด้วยส่วนประกอบแถบกว้าง 3 ส่วน ซึ่งเป็นแถบทึบหรือบาร์ และแถบว่างหรือสเปซ จากทั้งหมด 9 ส่วน ดังแสดงในรูปที่ 2.4 และรูปที่ 2.5 ซึ่งในบาร์โค้ดจะประกอบด้วย

1. ช่วงแถบว่างที่อยู่แต่ละด้านของบาร์โค้ด
2. ส่วนแสดงการเริ่มต้น และหยุดของตัวอักษร
3. ข้อมูลของตัวอักษร

อักษร 1 =	10001	0100
อักษร 4 =	00101	0100
บาร์	1 0 0 0 1	
สเปซ	0 1 0 0 0	

อักษร 1



บาร์	0 0 1 0 1
สเปซ	0 1 0 0 0

อักษร 4



รูปที่ 2.5 อักษรในบาร์โค้ด 3 ใน 9

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปแบบโครงสร้างของรหัส 39

ในรหัสแบบ 39 ความกว้างของแถบบาร์ และสเปซ จะอยู่ในรูปแบบของตัวเลขฐานสอง โดยแถบที่แคบจะแทนด้วยเลข 0 และแถบกว้างจะแทนด้วยเลข 1 ดังนั้น รหัส 3 ใน 9 ข้อมูล 1 พิลด์ จะประกอบด้วยแถบกว้าง 3 แถบ จึงมีเลขฐานสองค่า 1 อยู่ 3 ตัว และที่เหลือจะเป็นค่า 0 อยู่ 6 ตัว รหัสของแถบบาร์โค้ดจะประกอบด้วยรหัสเริ่มต้นทางด้านซ้ายสุด และรหัสหยุดที่ทางขวาสุดของขอบเขตระหว่างแถบที่แสดงการเริ่มต้นและแถบหยุดจะเป็นส่วนบรรจุข้อมูล ซึ่งสามารถบรรจุสูงสุดได้ถึง 32 ตัวอักษร แต่จะขึ้นอยู่กับความสามารถของอุปกรณ์ที่ใช้ร่วมด้วย

ตารางที่ 2.5 โครงสร้างตัวอักษรในรหัส

ตัวอักษร	รูปแบบ	บาร์	สเปซ	ตัวอักษร	รูปแบบ	บาร์	สเปซ
1	■ ■ ■ ■ ■	10001	0100	M	■ ■ ■ ■ ■	11000	0001
2	■ ■ ■ ■ ■	01001	0100	N	■ ■ ■ ■ ■	00101	0001
3	■ ■ ■ ■ ■	11000	0100	O	■ ■ ■ ■ ■	10100	0001
4	■ ■ ■ ■ ■	00101	0100	P	■ ■ ■ ■ ■	01100	0001
5	■ ■ ■ ■ ■	10100	0100	Q	■ ■ ■ ■ ■	00011	0001
6	■ ■ ■ ■ ■	01100	0100	R	■ ■ ■ ■ ■	10010	0001
7	■ ■ ■ ■ ■	00011	0100	S	■ ■ ■ ■ ■	01010	0001
8	■ ■ ■ ■ ■	10010	0100	T	■ ■ ■ ■ ■	00110	0001
9	■ ■ ■ ■ ■	01010	0100	U	■ ■ ■ ■ ■	10001	1000
0	■ ■ ■ ■ ■	00110	0100	V	■ ■ ■ ■ ■	01001	1000
A	■ ■ ■ ■ ■	10001	0010	W	■ ■ ■ ■ ■	11000	1000
B	■ ■ ■ ■ ■	01001	0010	X	■ ■ ■ ■ ■	00101	1000
C	■ ■ ■ ■ ■	11000	0010	Y	■ ■ ■ ■ ■	10100	1000
D	■ ■ ■ ■ ■	00101	0010	Z	■ ■ ■ ■ ■	01100	1000
E	■ ■ ■ ■ ■	10100	0010	,	■ ■ ■ ■ ■	00011	1000
F	■ ■ ■ ■ ■	01100	0010	.	■ ■ ■ ■ ■	10010	1000
G	■ ■ ■ ■ ■	00011	0010	SPACE	■ ■ ■ ■ ■	01010	1000
H	■ ■ ■ ■ ■	10010	0010	@	■ ■ ■ ■ ■	00110	1000
I	■ ■ ■ ■ ■	01010	0010	\$	■ ■ ■ ■ ■	00000	1110
J	■ ■ ■ ■ ■	00110	0010	/	■ ■ ■ ■ ■	00000	1101
K	■ ■ ■ ■ ■	10001	0001	+	■ ■ ■ ■ ■	00000	1011
L	■ ■ ■ ■ ■	00001	0001	%	■ ■ ■ ■ ■	00000	0111

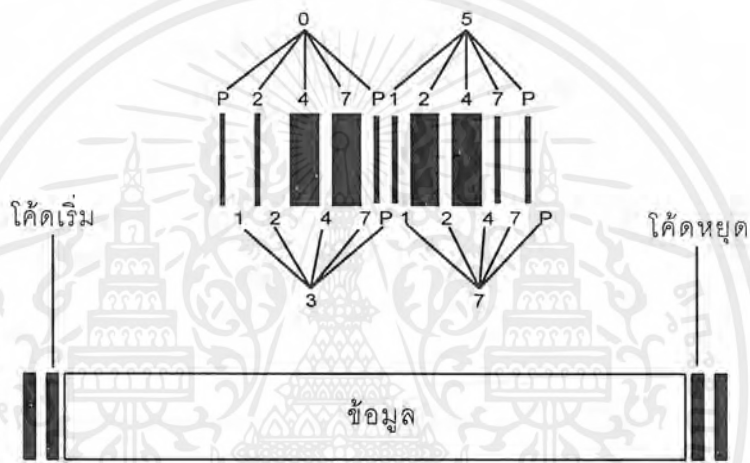
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ผังการจัดการข้อมูลสตริงในรหัส 39

/ เริ่มต้น ___ / ข้อมูล ฟิลด์ข้อมูล ___ / ค่าผิดพลาด ตรวจสอบตัวอักษร ___ / ___ หยุด /
(32 ตัวอักษร) (1 ตัวอักษร - สำรองไว้)

ผังการจัดการข้อมูลสตริงในรหัสแทรก 2 ใน 5

เริ่มต้น ___ / ___ ฟิลด์ข้อมูล ___ / ___ หยุด
(ความยาวขึ้นอยู่กับข้อมูล)



รูปที่ 2.6 รูปแบบของบาร์โค้ดรหัสแทรก 2 ใน 5

ตัวอย่างของข้อมูลบาร์โค้ด สำหรับเลข 1 และเลข 4 เปรียบเทียบให้ดู ดังแสดงในรูปที่ 2.6 ตัวอักษรทั้งหมดของรหัส 39 แสดงไว้ในตารางที่ 2.5 แถบและช่องว่าง หรือว่าบาร์และสเปซแต่ละอันสามารถเลือกได้ ซึ่งจะแคบ หรือกว้างขึ้นอยู่กับวิธีการแปลงรหัส ตัวอักษรแต่ละตัวประกอบด้วย แถบกว้าง 3 แถบ และแถบแคบ 6 แถบ ตัวเลข 1 ใช้แทนส่วนกว้าง และเลข 0 แทนส่วนแคบ ตัวอักษรอื่นๆ แบ่งแยกโดยช่องว่างระหว่างตัวอักษร

2.8.2 รหัสแทรก 2 ใน 5

เฉพาะข้อมูลตัวเลขเท่านั้นที่สามารถเข้ารหัสแบบแทรก 2 ใน 5 ความหนาแน่นของข้อมูลสูงสุด คือ 18 ตัวอักษรต่อนิ้ว รหัสอาจผิดพลาดได้หากไม่เข้ารหัสเป็นตัวเลขคู่ของตัวอักษรเข้ารหัส ในสัญลักษณ์ตัวอักษรตัวแรกของคู่แทนโดยบาร์ และตัวอักษรตัวที่ 2 แทนโดยสเปซ รหัสแทรก 2 ใน 5 ในอุตสาหกรรมผลิตยา, ในร้านอาหาร และอุตสาหกรรมผลิตสิ่งพิมพ์ เป็นต้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 2.6 ชุดอักขระของบาร์โค้ดรหัสแทรก 2 ใน 5

ตัวเลข (ฐานสิบ)	9 รหัส 2 ใน 5 (ตัดแปลงจาก BCD)	ค่าไบนารี
0	00110	6
1	10001	17
2	01001	9
3	11001	25
4	00101	5
5	10100	20
6	01100	12
7	00011	3
8	10010	18
9	01010	10
อักขระเริ่มต้น	00	0
อักขระหยุด	10	2

คุณสมบัติของรหัสแทรก 2 ใน 5 หรือ บาร์โค้ด USS-I 2/5

- ชนิดของตัวอักขระ : ตัวเลข
 ความยาวของข้อมูล : เปลี่ยนแปลงได้ แต่ต้องเป็นจำนวนคู่
 การถอดรหัส : ได้ทั้งสองทิศทาง (Bi-Directional)
 ความหนาแน่นของข้อมูล : สูงสุด 18 ตัวอักขระต่อนิ้ว
 ตัวอักขระพิเศษ : มีความแตกต่างกับรูปของการเริ่มและการหยุด

คุณสมบัติของรหัส Codebar

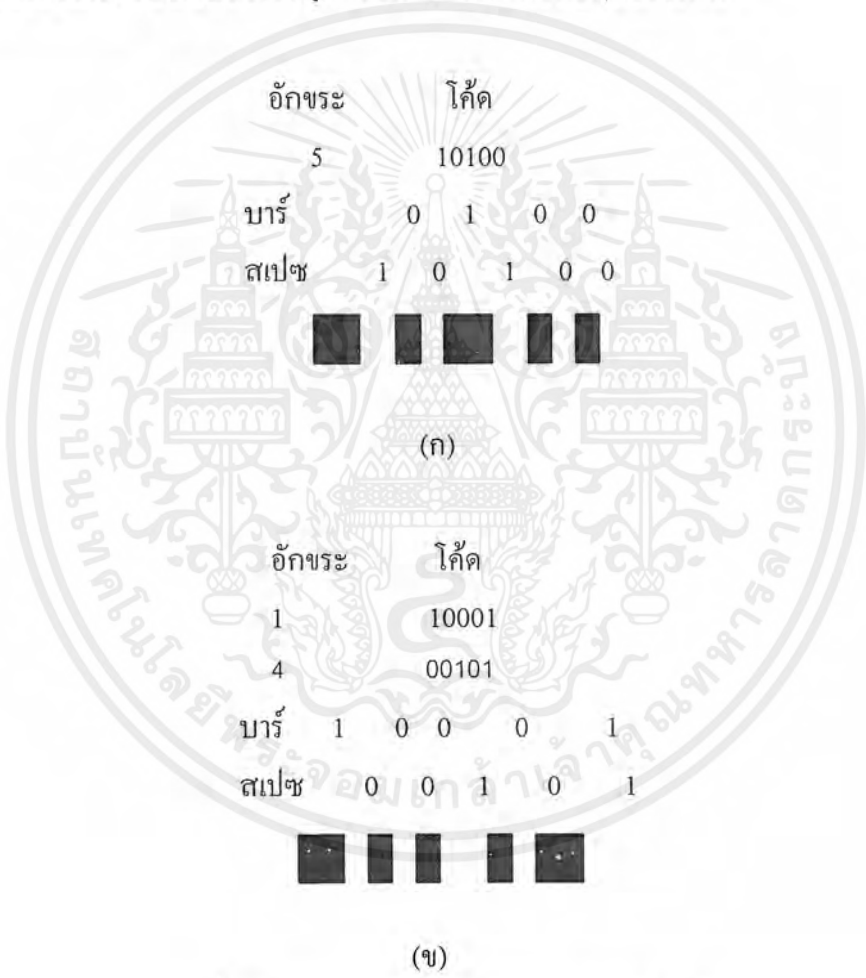
- ตัวอักขระ : ตัวเลข 0 ถึง 9 รวมทั้งตัวอักขระพิเศษอีก 6 ตัว คือ \$, - , . , / ,
 และ + รวมทั้งตัวอักขระเริ่มต้น หรือหยุดอีก 4 ตัว คือ A, B, C
 และ D

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
 ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- ความยาวของชุดข้อมูล : เปลี่ยนแปลงได้
- การถอดรหัส : ได้ทั้งสองทิศทาง
- ความหนาแน่นข้อมูล : สูงสุด 12.8 ตัวอักษรต่อนิ้ว

รูปแบบโครงสร้างของรหัส 2 ใน 5

รูปแบบสำหรับตัวอักษรจะประกอบด้วยบาร์ และสเปซสลับกันไป แทนค่าตัวเลขฐานสิบ ในรูปของตัวเลขฐานสอง 5 บิต (4 บิตแสดงค่าตัวเลข และ 1 บิตพาริตี) รูปแบบของบาร์โค้ดจะ ประกอบด้วยส่วนเริ่มต้น ส่วนของข้อมูล และส่วนแสดงการสิ้นสุดของรหัส



รูปที่ 2.7 ผังการจัดวางบาร์โค้ดรหัสแทรก 2 ใน 5

การเข้ารหัสตัวเลขต้องประกอบด้วยจำนวนคู่เสมอ ตัวเลขที่เข้ารหัสแล้ว ดังรูปที่ 2.7 สัญลักษณ์ของบาร์โค้ดบ่อยครั้งจะประกอบด้วยตัวอักษรที่มนุษย์สามารถอ่านได้อย่างง่ายๆ ซึ่งทำให้ง่ายต่อการตรวจสอบเมื่อไม่สามารถอ่านบาร์โค้ดได้ โดยตัวเลขจะเขียนไว้ด้านบน หรือด้านล่างของบาร์โค้ดเสมอ ความยาวของแถบบาร์โค้ดสามารถคำนวณโดยสมการที่ 2.1

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

$$L = (P (4N + 6) + 6 + N) X + 2Q \tag{2.1}$$

ซึ่ง

P คือ จำนวนของคู่อักขระ

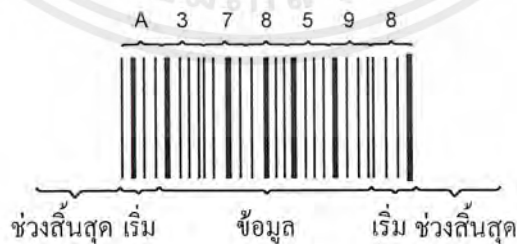
N คือ อัตราส่วนของแถบกว้างต่อแถบแคบ

X คือ ขนาดความกว้างของแถบแคบ

Q คือ ความกว้างของแถบแสดงการสิ้นสุดรหัส



รูปที่ 2.8 บาร์โค้ดผสมบาร์ของจำนวน 0123 ในรูปแบบรหัสแทรก 2 ใน 5



รูปที่ 2.9 การเข้ารหัส Codebar “A37859B”

ส่วนประกอบของข้อมูล ดังแสดงในรูปที่ 2.7 (ก) แสดงส่วนประกอบของรหัสแบบแทรก

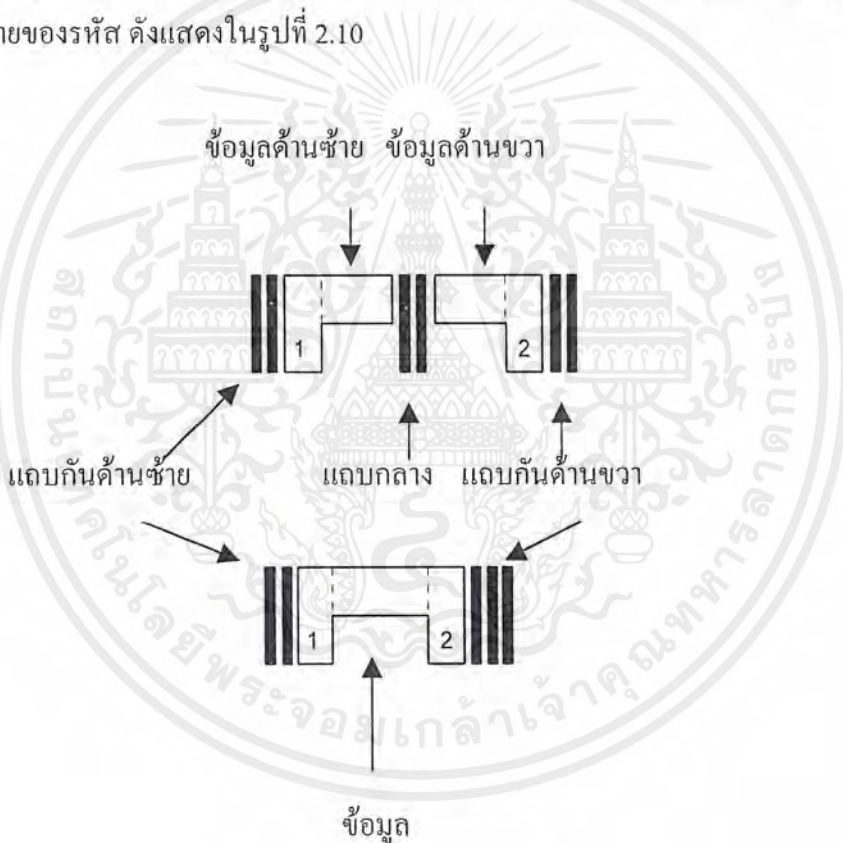
2 ใน 5 โดยอ่านค่าจากแถบได้ 5 และจากช่องว่างก็ได้ 5 โดยดูจากแถบหรือช่องว่าง กว้างแทนด้วย 1

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

และในทางตรงกันข้าม แถบหรือช่องว่างแคบแทนด้วย 0 ส่วนในรูปที่ 2.7 (จ) แสดงการอ่านรหัส จากแถบได้ 1 และจากช่องว่างได้ 4

ส่วนประกอบของบาร์โค้ด

ส่วนแสดงการเริ่มต้นและหยุดของบาร์โค้ด แสดงในรูปที่ 2.10 ส่วนเริ่มต้นจะอยู่ทางซ้าย ของข้อมูลทั้งหมด ประกอบด้วย 4 ส่วนแคบๆ โดยสลับกันระหว่างบาร์ และสเปซ ส่วนแถบแสดง การหยุดจะอยู่ทางด้านขวาของข้อมูลทั้งหมด ประกอบด้วยแถบใหญ่แล้วตามด้วยแถบเล็ก โดยมี ช่องว่างแคบ ๆ สลับกันไว้ และในส่วนประกอบของบาร์โค้ดจะมีส่วนที่แสดงการสิ้นสุดของรหัสที่ อยู่ปิดหัวท้ายของรหัส ดังแสดงในรูปที่ 2.10



รูปที่ 2.10 รูปแบบการเข้ารหัส UPC

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.8.3 รหัสแบบ Codebar

Codebar สามารถใช้กับข้อมูลตัวเลขและตัวอักษรพิเศษอีก 6 ตัวคือ \$, -, : , / , . , และ + และตัวอักษร 4 ตัว ที่แสดงการเริ่มต้นหรือหยุด คือ A, B, C และ D สัญลักษณ์ของ Codebar ใช้สำหรับเปลี่ยนแปลงความยาวของข้อมูล ซึ่งคุณสมบัติต่างๆ ได้ให้ไว้ในตารางที่ 2.7 ส่วนประกอบของข้อมูลแต่ละตัวจะแสดงให้ในรูปที่ 2.11

Codebar แต่ละตัวประกอบด้วยขอบเขตแสดงการสิ้นสุด, ส่วนแสดงการเริ่มต้น หรือหยุด และส่วนของข้อมูล ซึ่งข้อมูลสามารถเปลี่ยนแปลงความยาวได้ถึง 32 ตัวอักษร ตัวอย่างดังแสดงในรูปที่ 2.12 ตัวอักษรแต่ละตัวแทนโดยส่วนประกอบ 7 ส่วน มี 4 บาร์ และ 3 สเปซ ระหว่างแถบตัวอักษร แสดงว่าเริ่มต้น หรือหยุดมี 4 ตัว สามารถใช้เป็นตัวเริ่มต้นหรือหยุดได้ ส่วนประกอบการเพิ่มข้อมูล สัญลักษณ์ภายในตัวอักษรแบ่งโดยช่องว่างระหว่างตัวอักษรประกอบด้วยส่วนของช่องว่างแคบๆ 1 ช่อง ส่วนประกอบเบื้องต้นสำหรับตัวอักษร Codebar ทั้งหมด แสดงไว้ในตารางที่ 2.7 ซึ่งส่วนประกอบกว้างแทนไบนารี 1 และส่วนประกอบแคบแทนไบนารี 0 แต่ละตัวอักษรสามารถแทนโดยไบนารีขนาด 7 บิต เท่านั้น

ตารางที่ 2.7 รูปแบบบาร์โค้ด codebar และค่าระบบตัวเลขฐานสอง

ตัวอักษรที่เข้ารหัส	ค่าในรูปของเลขฐานสอง	แพตเทิร์นของบาร์และสเปซ
0	0000011	
1	0000110	
2	0001001	
3	1100000	
4	0010010	
5	1000010	
6	0100001	
7	0100100	
8	0110000	
9	1001000	
.	0001100	
\$	0011000	
:	1000101	
/	1010001	
-	1010100	
+	0010101	
A	0011010	
B	0101001	
C	0001011	
D	0001110	

2.8.4 รหัสสากล UPC (Universal Product Code)

บาร์โค้ด UPC ชุดอักขระประกอบด้วยตัวเลข และอีก 3 ส่วนพิเศษ คือ ส่วนเริ่มต้น, ส่วนหยุดและตัวอักขระตัวอักษรแต่ละตัวสร้างขึ้นโดย 4 ส่วน คือ 2 บาร์ และ 2 สเปซ มี 2 ตัวอย่างข้อมูลพื้นฐานดังแสดงในรูปที่ 2.10

ตารางที่ 2.8 ระบบจำนวนของตัวอักขระ

ตัวอักขระ	การใช้งาน
0	รหัส UPC ปกติ
2	รายการสุ่มน้ำหนักจำพวกพืชผล และเนื้อสัตว์
3	รายการเกี่ยวกับสุขภาพ เช่น รหัสสากลของยา
4	การปฏิบัติการในร้านที่ไม่ใช่รายการอาหารพร้อมรหัสตัวเลขเพื่อตรวจสอบและป้องกัน รวมทั้งการใช้งานในรูปแบบที่มีจำกัด
5	สำหรับใช้กับคูปอง
อื่น ๆ	สำรองไว้ใช้งาน

ตารางที่ 2.9 ตัวเลขของรหัสสากล

ตัวอักขระ	อักขระฟิลด์ซ้าย (พาริตีคู่)	อักขระฟิลด์ขวา (พาริตีคู่)
1	0001101	1110010
2	0011001	1100110
3	0010011	1000010
4	0111101	1011100
5	0100011	1001110
6	0110001	1010000
7	0101111	1000100
8	0111011	1001000
9	0001011	1110100

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รหัสสากล UPC จะใช้เพื่อเข้ารหัสรายการต่างๆ ในบัตรประจำตัวในหน่วยงานต่างๆ และใช้ในระบบการบรรจุหีบห่อสินค้าอุปโภคบริโภคในสหรัฐฯ โดยที่ประมุขของ UPC ซึ่งไม่เป็นสากลเท่าไรนัก ระบบตัวอักษรเฉพาะ และชนิดพิเศษใช้ของข้อมูล การจัดการของระบบตัวเลข ดังแสดงในตารางที่ 2.8 ตัวเลขของรหัสสากลที่แสดงให้เห็นในตารางที่ 2.9 ข้อมูลตัวอย่างสำหรับตัวเลข 5 มี 2 แบบ ได้แสดงไว้ในรูปที่ 2.10 ซึ่งแถบทึบแทนค่า 1 และช่องว่างแทนค่า 0 ในสัญลักษณ์ของบาร์โค้ดรหัสประเภทนี้มีความผิดพลาดต่ำ จึงได้มีการพัฒนาไปประยุกต์ใช้กับเครดิตการ์ด

2.8.5 รหัสตัวเลขของยุโรป EAN (European Article Numbering)

ลักษณะรหัสตัวเลขของยุโรปที่ใช้ในทวีปยุโรป จะใช้คู่กับรหัส UPC เป็นรหัสที่ใช้สำหรับข้อมูลตัวเลข ตัวอย่างดังแสดงในรูปที่ 2.12 ซึ่งเป็นรหัส EAN-13

การพิมพ์ฉลากบาร์โค้ด

ในอุตสาหกรรมผลิตบาร์โค้ดโดยทั่วไป ป้ายบาร์โค้ดที่ใช้จะใช้การพิมพ์โดยคอมพิวเตอร์ โดยเครื่องพิมพ์ดอตเมตริกซ์ ซึ่งจะคำนึงถึงชนิดของรหัสที่ใช้เป็นสำคัญ บาร์โค้ดจะพิมพ์โดยมาตรฐานที่จะทำให้ผิดพลาดน้อยที่สุดโดยผู้ใช้บาร์โค้ดโดยทั่วไปมีระบบการตรวจสอบ 4 ตัวแปรที่สำคัญคือ รูปแบบการวางป้าย รูปแบบการเข้ารหัส คุณภาพความเข้ม (Contrast) ของการพิมพ์ และความกว้างของแถบ และช่องว่างของบาร์โค้ด การเปลี่ยนแปลงในความกว้างของแถบและช่องว่างบนบาร์โค้ดที่กำหนดจะเกิดจากกรรมวิธีในการพิมพ์บาร์โค้ด การสร้างป้ายบาร์โค้ดจะขึ้นอยู่กับความต้องการของผู้ใช้ ป้ายบาร์โค้ดสามารถเลือกได้มีทั้งที่พิมพ์ไว้แล้ว และระบบที่สามารถนำไปสร้างบาร์โค้ดใหม่ได้เองตามต้องการ

ป้ายบาร์โค้ดที่มีใช้งานกันทุกวันนี้แบ่งได้เป็น 2 รูปแบบตามลักษณะการพิมพ์ คือบาร์โค้ดที่พิมพ์สำเร็จไว้แล้ว และป้ายบาร์โค้ดที่ต้องพิมพ์ใช้เอง ซึ่งแต่ละแบบมีข้อดีข้อเสียอยู่ในตัวเอง

ข้อดี ของการใช้ป้ายสำเร็จ คือ

1. สามารถที่จะพิมพ์บาร์โค้ดที่มีความสามารถหนาแน่นข้อมูลสูงๆ ได้
2. มีความเชื่อถือได้ของการพิมพ์สูง เพราะส่วนใหญ่มักจะพิมพ์โดยระบบออฟเซต
3. สามารถที่จะพิมพ์บนวัสดุอื่น นอกเหนือจากการพิมพ์บนกระดาษได้
4. ไม่ต้องยุ่งยากหาเครื่องพิมพ์
5. สามารถที่จะสร้างบาร์โค้ดติด เพื่อการบรรจุในผลิตภัณฑ์ที่แตกต่างกันได้

ข้อเสีย

1. ราคาต่อหน่วยจะสูงกว่า
2. จะต้องเตรียมข้อมูลของป้ายไว้ล่วงหน้าก่อน

เริ่ม---/--- NSC---/---ซ้าย---/---กลาง---/---ขวา---/---ตรวจสอบ---/---หยุด
 101 0 0 01010 5 ตรวจสอบ 101

(ก)

	ฟิลต์ซ้าย, พาริตีคี่			ฟิลต์ขวา, พาริตีคู่		
บาร์	1	1	1	บาร์	1	1 1 1
สเปซ	0	0 0 0	สเปซ	0 0	0	

(ข)

รูปที่ 2.11 (ก) ผังการจัดวางข้อมูลสตริงของบาร์โค้ดรหัส UPC

(ข) ฟิลต์ซ้ายและฟิลต์ขวาของข้อมูลตัวอักษร

การพิมพ์บาร์โค้ดโดยเครื่องพิมพ์ สามารถที่จะพิมพ์นอกสถานที่ได้ จากเครื่องพิมพ์หลายรูปแบบ ถึงแม้ว่าจะมีคุณภาพของบาร์โค้ดด้อยกว่าการใช้ป้ายสำเร็จ แต่มีความยืดหยุ่นในการใช้งานสูง เครื่องพิมพ์ใช้ได้หลายแบบ คือ

เทอร์มอลพรินเตอร์

1. ราคาของป้ายขึ้นกับชนิดของกระดาษที่ใช้
2. คุณภาพการพิมพ์อยู่ในระดับปานกลาง
3. ความคงทนของป้ายปานกลาง
4. ราคาของระบบมีตั้งแต่ราคาถูกถึงปานกลาง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ดอตเมตริกพริ้นเตอร์

1. ใช้กระดาษชนิดราคาถูก
2. ความหนาแน่นของการพิมพ์ปานกลาง
3. พิมพ์บาร์โค้ดได้เต็มรูปแบบ
4. ราคาของระบบอยู่ในช่วงกว้าง
5. คุณภาพการพิมพ์ไม่แน่นอน

การพิมพ์ตัวอักษรเต็ม (Full Character Impact Printing)

1. ราคากระดาษต่ำ
2. อุปกรณ์มีราคาสูง
3. สามารถพิมพ์ป้ายที่มีความหนาแน่นสูงได้
4. ป้ายอาจหลุดได้ง่าย

เลเซอร์พริ้นเตอร์

1. อุปกรณ์มีราคาสูง
2. พิมพ์บนพื้นผิวได้หลายชนิด
3. คุณภาพการพิมพ์สูง



รูปที่ 2.12 รูปบาร์โค้ดรหัส EAN

จะเห็นว่ารูปแบบของบาร์โค้ดปัจจุบันมีหลายชนิด ขึ้นอยู่กับประเภทของงาน ซึ่งนิยมใช้กันมาก โดยเฉพาะในระบบงานการบริการต่างๆ ที่ต้องการความรวดเร็ว และความถูกต้องในการทำงานสูง ซึ่งหากใช้บาร์โค้ดแล้ว จะช่วยให้มีความยืดหยุ่นในการทำงาน นับเป็นการนำเทคโนโลยีมาใช้อำนวยความสะดวกในอีกรูปแบบหนึ่ง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.9 LCD โมดูล

ปัจจุบัน LCD เป็นที่นิยมกันเป็นอย่างมาก สำหรับการแสดงผลในเครื่องมือเครื่องใช้ต่างๆ ทั้งนี้เนื่องจากความเหมาะสมหลายๆ ด้าน เช่น การใช้กระแสต่ำ สามารถแสดงผลได้ทั้งตัวอักษรและตัวเลข หรือแสดงกราฟฟิก (เฉพาะรุ่น) จะมีปัญหาในด้านวงจร ซึ่งมีระบบการทำงานที่ซับซ้อนและหาอุปกรณ์ค่อนข้างยาก แต่ขณะนี้ผู้ผลิต LCD จะทำรุ่นที่เป็น LCD จะผลิตเป็น Module ออกมาคือ เป็น module ที่มีตัว LCD และวงจรควบคุมให้พร้อม (เรียกว่า LCM) ซึ่งทำให้ผู้ใช้สามารถต่อเข้ากับระบบไมโครคอนโทรลเลอร์ได้ง่าย และสะดวกกับการเขียนโปรแกรม รวมทั้งมีจำหน่ายกันอย่างกว้างขวาง และมีราคาที่เหมาะสมทำให้ผู้ใช้ทางด้านไมโครคอนโทรลเลอร์หันมาใช้แผงที่แสดงด้วย LCD Module กันมากขึ้น

LCD Module มีหลายรุ่นและคุณสมบัติแตกต่างกันไป ซึ่งแบ่งได้เป็นสองกลุ่มหลัก คือ แบบ Dot matrix และ Graphic โดยแบบ Dot matrix จะแสดงตัวอักษรขนาด 5 x 8 Dot และมีจำนวนตัวอักษรและบรรทัดแตกต่างกันในแต่ละรุ่น ส่วนแบบ Graphic จะสามารถแสดงผลได้แบบ Bit map คือ สร้างเป็นเป็นภาพใดๆ ก็ได้ตามต้องการ แนวทางในการใช้งานของทั้ง 2 แบบมีลักษณะใกล้เคียงกัน การใช้งานทั่วไปมักจะเป็นแบบ Dot matrix มากกว่า เนื่องจากราคาถูกกว่า และเพียงพอต่องานส่วนใหญ่ คุณสมบัติของ Dot matrix LCD module สามารถสรุปเป็นข้อๆ ได้ดังนี้

1. มีให้เลือกหลายรุ่นตามความต้องการใช้งาน โดยมีตัวอักษรและบรรทัดแตกต่างกันไป
2. ตัวอักษรแสดงด้วย Dot matrix ขนาด 5 x 8 Dot
3. สามารถต่อเข้ากับระบบไมโครคอนโทรลเลอร์ ได้ 2 ลักษณะ คือ แบบ Memory map และแบบผ่าน 8255 port ซึ่งจะใช้ขาสัญญาณทั้งหมด 14 PIN (ขั้วต่อ 16 PIN)
4. การใช้งานง่าย และสะดวก ระบบไมโครคอนโทรลเลอร์เพียงแค่ส่งข้อมูลให้กับ LCD module ข้อความจะปรากฏบนแผงแสดง และจะค้างไว้ตลอด ทำให้ไม่ต้องเสียเวลาของระบบ
5. มีคำสั่งพิเศษไว้สำหรับอำนวยความสะดวก เช่น CLEAR , DISPLAY , HOME , CURSOR ,ON OFF CURSOR,BLANK CRARACTER และอื่นๆ อีก
6. สามารถแสดงผลเป็นตัวอักษรภาษาอังกฤษ และตัวเลขได้ 160 ตัว และสัญลักษณ์พิเศษอีก 32 ตัว รวมทั้งสามารถกำหนดตัวอักษรได้อีก 8 ตัว
7. กินกระแสน้อย และมีน้ำหนักเบา รวมทั้งทำงานด้วยไฟเลี้ยงระดับ 5 โวลท์

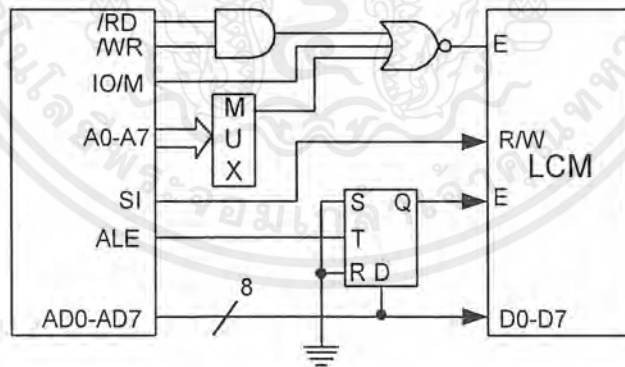
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การต่อเข้ากับระบบไมโครคอนโทรลเลอร์

LCD Module จะต่อเข้ากับระบบไมโครคอนโทรลเลอร์ ได้ 2 ลักษณะดังกล่าวไปแล้ว ซึ่งทั้งสองแบบมีข้อดีข้อเสียแตกต่างกันไป โดยแต่ละแบบจะมีหลักการดังนี้

การต่อแบบ Memory map

1. สามารถต่อเข้ากับ Chip เบอร์ต่างๆไปได้ เช่น 8051 หรือ Z80 โดยจะจะทำให้ระบบไมโครคอนโทรลเลอร์มองเห็น LCD module ในลักษณะของ Memory ได้ทันที
2. ผู้ใช้สามารถเขียนและอ่านข้อมูลจาก LCD module ได้ ทำให้มองเสมือนว่าเป็น Memory buffer ไปในตัว
3. เนื่องจากสามารถอ่านข้อมูลกลับได้ จึงทำให้สามารถตรวจสอบ Flag ความพร้อมในขณะที่ LCD module กำลังทำงานได้
4. ใช้ได้กับบอร์ดที่มี LCD bus มาให้พร้อมเท่านั้น
5. ทำให้กินพื้นที่หน่วยความจำไปส่วนหนึ่ง และต้องการ Decode ละเอียดพอสมควร
6. การจัดขั้วสัญญาณจะต้องเป็นไปตามแบบของ Chip



รูปที่ 2.13 การต่อแบบ Memory map

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 2.10 ขาสัญญาณของ LCD module

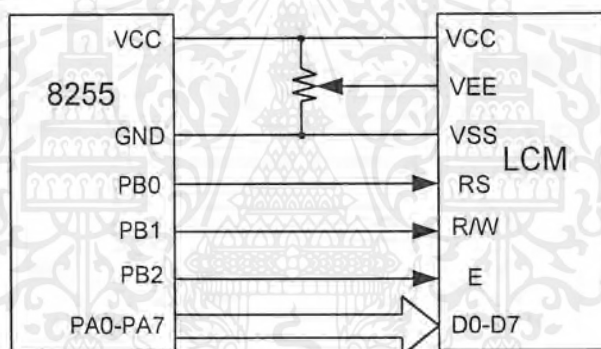
PIN	SYMBOL	LEVEL	FUNCTION
1	Vss	-	0 V GND
2	Vcc	-	+5 V Power Supply
3	Vee	-	+V For Liquid Crystal Drive
4	RS	H/L	Register Select H: Data Input L: Instruction Input
5	R/W	H/L	H: Data Read L: Data Write
6	E	H	Enable Signal (L->H)
7	DB 0	H/L	Data Bus Bit 0
8	DB 1	H/L	Data Bus Bit 1
9	DB 2	H/L	Data Bus Bit 2
10	DB 3	H/L	Data Bus Bit 3
11	DB 4	H/L	Data Bus Bit 4
12	DB 5	H/L	Data Bus Bit 5
13	DB 6	H/L	Data Bus Bit 6
14	DB7	H/L	Data Bus Bit 7

7. สามารถต่อเข้ากับ Chip เบอร์ต่างๆไปได้เช่น 8051 หรือ Z80 โดยจะทำให้ระบบไมโครคอนโทรลเลอร์ มองเห็น LCD module ในลักษณะของ Memory ได้ทันที
8. ผู้ใช้สามารถเขียนและอ่านข้อมูลจาก LCD module ได้ ทำให้มองเสมือนว่าเป็น Memory buffer ไปในตัว
9. เนื่องจากสามารถอ่านข้อมูลกลับได้ จึงทำให้สามารถตรวจสอบ Flag ความพร้อมในขณะที่ LCD module กำลังทำงานได้
10. ใช้ได้กับบอร์ดที่มี LCD bus มาให้พร้อมเท่านั้น
11. ทำให้กินพื้นที่หน่วยความจำไปส่วนหนึ่ง และต้องการ Decode ละเอียดพอสมควร
12. การจัดขาสัญญาณจะต้องเป็นไปตามแบบของ Chip

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การต่อแบบ I/O port

1. สามารถต่อเข้ากับ I/O port ใดๆก็ได้ โดยใช้สายสัญญาณจำนวน 11 เส้น และใช้โปรแกรมเป็นตัวสร้างสัญญาณขึ้นมาให้ตรงกับข้อกำหนดของ LCD module
2. ผู้ใช้จะเขียนข้อมูลให้ LCD module ได้โดยตรง ซึ่งผู้ใช้ควรกำหนด memory ส่วนหนึ่งเสมือน Buffer ให้กับ LCD module
3. ไม่สามารถอ่านข้อมูลกลับได้ จึงต้องใช้การหน่วงเวลาของระบบเอง เพื่อรอให้ LCD module กระทำกระบวนการต่างๆ
4. ใช้กับบอร์ดต่างๆไป ที่มีพอร์ต
5. ไม่เปลืองส่วนของ Memory ในการใช้งาน
6. การจัดหาสัญญาณกระทำได้อย่างอิสระ



รูปที่ 2.14 การต่อแบบ I/O port

ชุดคำสั่งควบคุมและการแสดงข้อความ

ขาสัญญาณ VEE มีไว้สำหรับกำหนดความเข้มของตัวอักษร โดยถ้าต่อกับ GND จะมีความเข้มสูงสุด แต่ถ้าต่อกับ VCC จะมีความเข้ต่ำสุด ปกติ LCD รุ่นธรรมดา อาจจะทำกับ GND ไว้เลยก็ได้ ไม่ต้องใส่ VR ให้สั่นเปลือง แต่ถ้าเป็นรุ่น STN (มมมองกว้าง) ให้ใช้ $R=2k\Omega$ ต่อลง GND อีกที เพื่อให้ความเข้มเหมาะสม การเขียนข้อมูลหรืออ่านข้อมูลกับ LCD module ก็คือการกำหนดคุณสมบัติต่างๆ ในการใช้งานของ LCD ตามชุดคำสั่งควบคุม และรวมไปถึงการเขียนข้อมูลที่เป็นข้อความ เพื่อให้ปรากฏบนแผงด้วย โดยมีรายละเอียดตามตารางต่อไปนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 2.11 ชุดคำสั่งควบคุมข้อความ

INSTRUCTION	RS	R/W	DATA BIT								EXE. TIME
			7	6	5	4	3	2	1	0	
CLAER DISPLAY	0	0	0	0	0	0	0	0	0	1	1640
CURSOR AT HOME	0	0	0	0	0	0	0	0	0	*	1640
ENTRY MODE SET	0	0	0	0	0	0	0	1	I/D	S	40
DISPLAY ON/OFF	0	0	0	0	0	0	1	D	C	B	40
DISPLAY SHIFT	0	0	0	0	0	1	S/C	R/L	*	*	40
FUNCTION SET	0	0	0	0	1	DL	N	F	*	*	40
SET CGRAM ADD.	0	0	0	1	CGRAM ADDRESS						40
SET DDRAM ADD.	0	0	1	DDRAM ADDRESS						40	
BUSY, ADD READ	0	1	BF	ADDRESS						0	
CGRAM,DDRAM WR	0	1	WRITE DATA						40		
CGRAM,DDRAM RD	1	1	READ DATA						40		

1. การเขียนข้อมูลให้กับ LCD module จะแบ่งออกเป็น 2 ลักษณะ คือ Instruction และ Data โดยจะกำหนดด้วยขาสัญญาณ RS คือ ถ้า RS=0 จะหมายถึงส่งสัญญาณควบคุม (instruction) หรืออ่านค่า Flag สภาพการทำงานของ LCD module และถ้า RS=1 จะหมายถึงการเขียนค่าหรืออ่าน Data กับ LCD module
2. หลักการเขียนข้อมูลให้ LCD คือ เมื่อการเขียนข้อมูลไปแล้ว ตัว LCD module จะใช้เวลาในการทำงานชั่วขณะหนึ่ง ซึ่งในระบบไมโครคอนโทรลเลอร์ สามารถตรวจสอบได้จาก Busy Flag (BF) และถ้าเรียบร้อยแล้วจึงจะสามารถเขียนข้อมูลอันต่อไปได้กรณีที่การต่อวงจรเป็นแบบ I/O port คือ ไม่สามารถอ่านข้อมูลย้อนกลับได้ ระบบไมโครคอนโทรลเลอร์ก็ต้องใช้วิธีหน่วงเวลาแทน
3. การเขียนข้อมูลให้กับ LCD module นี้ สามารถทำได้ทั้ง 4 Bit และ 8 Bit โดยในกรณี 4 Bit นี้ จะใช้สายสัญญาณ Data เพียง 4 เส้นเท่านั้น คือ DB4 - DB7(ใช้สำหรับไมโครคอนโทรลเลอร์ แบบ 4 Bit หรือ เพื่อการประหยัดสาย) การเขียนข้อมูลจะ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- กระทำเหมือนกับ 8 Bit เขียนเพียง 2 ครั้ง คือ DB4-DB7 ก่อนแล้วตามด้วย DB0-DB3 และจะต้องกำหนดคุณสมบัติตามค่า DL ในคำสั่ง Function set ด้วย
4. DDRAM (Display Data Ram) คือหน่วยความจำใน LCD module ที่เป็น Buffer ของรหัส ASCII ใดๆ ลงไปในหน่วยความจำนี้ จะปรากฏที่แผงตัวอักษรทันที
 5. CGRAM (Character Generator Ram) คือ หน่วยความจำใน LCD module สำหรับเก็บภาพตัวอักษรที่ผู้ใช้สามารถสร้างเองได้ (8 ตัว) โดยอาจสร้าง Address ได้ทั้งหมด 64 byte คือ 8 ตัวอักษร คูณ 8 แถว

2.10 การอินเตอร์เฟสกับจอแสดงผล LCD และคีย์บอร์ด

การอินเตอร์เฟสและการเขียนโปรแกรมควบคุมการทำงานของจอแสดงผล LCD กับ MCS-51 ซึ่งเกิดบอร์ดคอมพิวเตอร์ ซึ่งในปัจจุบันจอแสดงผล LCD ถูกนำมาใช้งานมากขึ้น อันเนื่องจากการควบคุมที่ง่าย, แสดงผลตัวอักษรได้สวยงามชัดเจน และราคาที่คุ้มกับประสิทธิภาพที่ได้รับ และกล่าวถึงเทคนิคการอินเตอร์เฟสกับคีย์บอร์ด

2.11 การอินเตอร์เฟสกับจอแสดงผล LCD โมดูล

ในระบบที่มีการใช้งานไมโครคอนโทรลเลอร์ทำงานแบบระบบโดดเดี่ยว (Stand Alone Microcontroller Applications) บ่อยครั้งที่จำเป็นต้องติดตั้งอุปกรณ์แสดงผลด้วย เพื่อแสดงข้อความ, ตัวเลข, ผลการวัดค่า หรือข้อมูลต่าง ๆ ในการสื่อสารกับผู้ใช้งานได้ และอุปกรณ์หนึ่งที่ถูกนำมาใช้งานกันอย่างกว้างขวางคือจอแสดงผล LCD โมดูล ตัวอย่างที่เห็นกันได้ง่ายคือ นำมาใช้ในการแสดงผลแบบเมนู เพื่อให้ผู้ใช้งานกดคีย์ในการเลือกรูปแบบการทำงานต่อไป เป็นต้น

2.12 การเชื่อมต่อและควบคุม

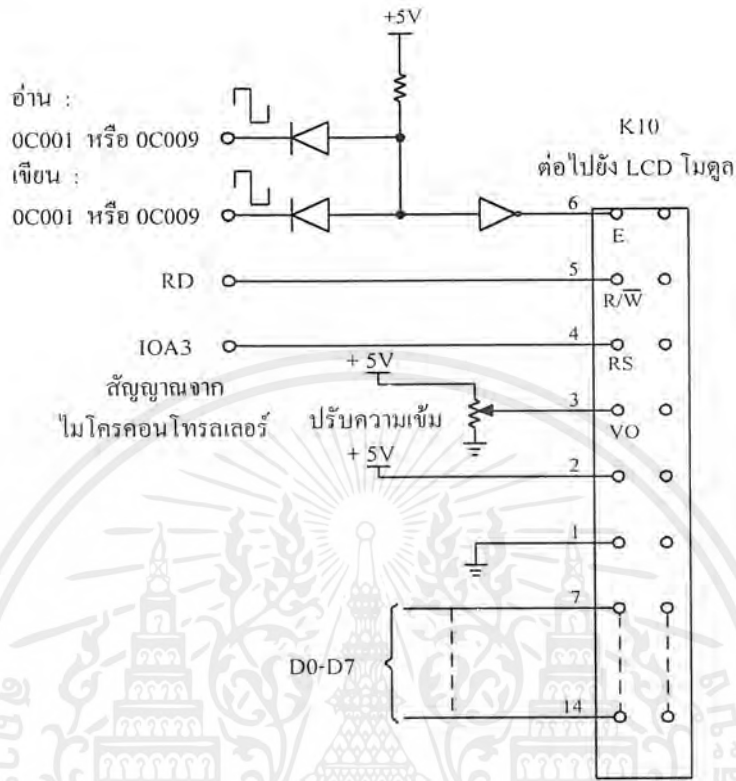
การเชื่อมต่อทางไฟฟ้าระหว่างบอร์ด MCS-51 และจอแสดงผล LCD โมดูลใช้ลวดเชื่อมต่อหรือสายไฟขนาด 14 เส้น จอแสดงผล LCD โมดูลกำหนดให้ใช้รุ่นใดก็ได้ที่ใช้ไอซีคอนโทรลเลอร์อ้างอิงเป็นเบอร์ HD44780 หรือเบอร์อื่นที่เทียบเท่า ตัวอย่างเช่น รุ่น H2570 LM016L และ LM1612A เป็นต้น ขาเชื่อมต่อของ LCD โมดูลของแต่ละยี่ห้อแต่ละแบบไม่มีมาตรฐานที่แน่นอนเหมือนกัน การเชื่อมต่อขาต่างๆ จึงต้องอาศัยคู่มือของ LCD โมดูลรุ่นนั้นๆ ที่จัดซื้อมาอ้างอิงเป็นหลักตำแหน่งขาต่างๆ ที่เชื่อมต่อไปยัง LCD โมดูลที่คอนเน็คเตอร์ K_{10} บนบอร์ดตั้ง

แสดงในรูปที่ 2.15 อาศัยรายละเอียดจากรูปนี้เพื่อเชื่อมต่อไปยัง LCD โมดูล และอ้างอิงในการตรวจสอบวงจรได้

การควบคุมการทำงานของ LCD โมดูลอาศัยสัญญาณอินพุตซึ่งจะถูกส่งไปยัง LCD โมดูล หลังจากที่มีการกำหนดรูปแบบของการทำงานว่าอยู่ในโหมดการอ่าน หรือการเขียนไปยังตำแหน่งแอดเดรส 0C009H หรือ 0C001H จากรูปที่ 2.15 เห็นได้ว่าขาสัญญาณเลือกโหมดการอ่านหรือเขียน (R/W) ต่อบางขาโดยตรงกับขา RD ของ MCS-51 บอร์ด เมื่อระบุโหมดการอ่านหรือเขียนแล้วต้องกำหนดการทำงานของ LCD โมดูลด้วยว่าในโหมดการอ่าน หรือเขียนนั้นอยู่ในรูปของข้อมูลตัวอักษร หรือข้อมูลคำสั่ง การกำหนดรูปแบบของข้อมูลทำได้ที่ขาสัญญาณ RS (Register Select) ของ LCD โมดูล ซึ่งเชื่อมต่อโดยตรงกับขา IOA₃ ของไมโครคอนโทรลเลอร์โดยถ้า RS = IOA₃ = 1 และติดต่อกับแอดเดรส 0C009H เป็นโหมดการอ่าน หรือเขียนข้อมูลอักขระ และถ้า RS = IOA₃ = 0 และติดต่อกับแอดเดรส 0C001H จะเป็นโหมดการอ่านหรือเขียนข้อมูลคำสั่งควบคุมภายใน LCD โมดูล

การส่งผ่านข้อมูลทั้งตัวอักษร และคำสั่งควบคุมถูกป้อนเข้ากับบัสข้อมูล 2 ทิศทางของระบบจริงๆ แล้ว การใช้งานจอแสดงผล LCD โมดูลสามารถทำงานได้ทั้งในโหมด 4 บิต (D₀-D₃) หรือโหมด 8 บิต (D₀-D₇) ก็ได้ แต่เนื่องจากในระบบที่เราศึกษาอยู่เป็นไมโครคอนโทรลเลอร์ที่ทำงานแบบ 8 บิต ดังนั้น การเชื่อมต่อ LCD โมดูล จึงต่อใช้งานแบบ 8 บิต กับบัสข้อมูลได้พอดี และสามารถเขียนโปรแกรมให้ทำงานในโหมดส่งผ่านข้อมูล 8 บิตกับจอ LCD โมดูลได้ทันที อย่างไรก็ตาม ในบางกรณีผู้อ่านอาจต้องศึกษาเพิ่มเติมการใช้งาน LCD โมดูล ในโหมดส่งผ่านข้อมูล 4 บิต ด้วย เช่น ถ้ามีพอร์ตเหลือใช้งานเพียงพอร์ตเดียว แต่จำเป็นต้องมีการใช้งานพอร์ตนี้กับระบบส่วนอื่นๆ ด้วย การเลือกใช้งาน LCD โมดูล ให้เป็นโหมดส่งผ่านข้อมูล 4 บิตนี้ จึงเป็นวิธีช่วยลดจำนวนสายสัญญาณของพอร์ตที่มีจำกัดให้เหลือใช้ได้มากขึ้น

จากรูปที่ 2.15 ที่ตำแหน่งขา 3 ของ K₁₀ ถูกต่อเข้ากับความต้านทานปรับค่าได้เพื่อปรับแรงดัน V₀ ที่ป้อนให้กับจอแสดงผล LCD โมดูล แรงดันที่จุดนี้มีผลต่อการปรับความเข้ม (Contrast) เพื่อให้ผู้ใช้สามารถปรับความเข้มได้เหมาะสมกับสภาพแวดล้อมในขณะนั้น ทั้งความเข้มของแสงจากภายนอก และมุมมองของผู้ใช้งาน ก่อนการเชื่อมต่อ LCD โมดูลกับ K₁₀ ควรแน่ใจว่าตำแหน่งขาต่าง ๆ ในการเชื่อมต่อทั้งหมดถูกต้องแล้วจึงทำการเชื่อมต่อจริง ถ้ายังไม่แน่ใจควรตรวจสอบจากคู่มือให้แน่ใจเสียก่อนเสมอ เพราะไม่เช่นนั้นอาจทำให้เกิดความเสียหายขึ้นได้



รูปที่ 2.15 การเชื่อมต่อระหว่าง LCD โมดูลและบอร์ด MCS-51

2.13 การอินเตอร์เฟสกับคีย์บอร์ด

โครงการที่ถูกสร้างขึ้นส่วนมากจำเป็นต้องติดต่อกับอุปกรณ์อินพุต เช่น อุปกรณ์ตรวจจับสัญญาณไฟฟ้าต่างๆ , คีย์บอร์ด เป็นต้น โดยเฉพาะคีย์บอร์ดที่ใช้ติดต่อกับผู้ใช้งานในการควบคุมเครื่อง การอินเตอร์เฟสกับคีย์บอร์ด 6 รูปแบบที่ใช้งานกันอยู่ทั่วไป สำหรับตรวจจับสถานะการเปิดและปิดของสวิทช์ในที่นี้ยกตัวอย่างเฉพาะวงจรการเชื่อมต่อสวิทช์ในแต่ละรูปแบบเป็นหลัก เพื่อให้สามารถมองเห็นแนวทางการเขียนโปรแกรมได้ ซึ่งแต่ละรูปแบบการเชื่อมต่อสวิทช์จะมีข้อดีและข้อเสียแตกต่างกันไป ดังแสดงในรูปที่ 2.16

2.13.1 แบบเชื่อมต่อสวิทช์โดยตรงกับพอร์ต

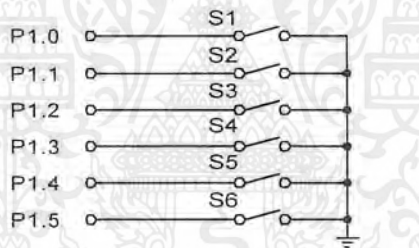
รูปที่ 2.16 (ก) แสดงการเชื่อมต่อสวิทช์โดยตรงกับพอร์ตซึ่งเป็นแบบที่ง่ายที่สุด การกดสวิทช์แต่ละตัวจะทำให้ขาพอร์ตนั้นถูกต่อลงกราวด์โดยตรง ซึ่งไม่จำเป็นต้องมีวงจรจำกัดกระแสก็ได้ เพราะว่าภายในตัวไมโครคอนโทรลเลอร์มีความต้านทานพูลอัพต่ออยู่แล้วการเขียนโปรแกรมทำได้ง่ายมากโดยการใช้คำสั่งตรวจสอบสถานะของแต่ละบิตในพอร์ต เช่น คำสั่ง JB หรือ JNB

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

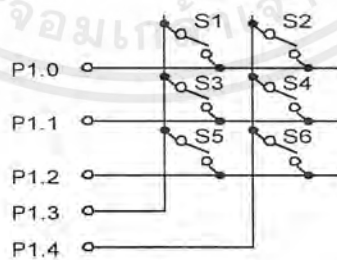
ข้อเสียของวงจรแบบนี้คือสิ้นเปลืองจำนวนขาพอร์ตจำนวนมาก ถ้าต้องการใช้สวิตช์มากเท่าใดก็ ต้องขยายพอร์ตให้มากตามไปด้วย ซึ่งเป็นเรื่องยุ่งยาก เช่น ถ้าต้องการออกแบบเป็นคีย์บอร์ดสำหรับ ป้อนตัวอักษรขนาด 60 คีย์ ผู้ออกแบบต้องจัดสร้างขาพอร์ตให้ได้ถึง 60 ขาพอร์ต ซึ่งแทบจะเป็นไป ได้ยากและสิ้นเปลืองอุปกรณ์มาก

2.13.2 แบบเชื่อมต่อสวิตช์แบบเมตริกซ์

รูปที่ 2.16 (ข) แสดงการเชื่อมต่อสวิตช์แบบเมตริกซ์ สวิตช์แต่ละตัวจะถูกเชื่อมต่อกันแบบ แถว และคอลัมน์ในรูปแบบของเมตริกซ์ การตรวจสอบการกดคีย์ใดบนคีย์บอร์ดทำได้โดยการป้อน ค่าตรวจสอบค่าหนึ่งไปยังด้านคอลัมน์ และตรวจสอบการเปลี่ยนแปลงที่เกิดขึ้นทางด้านแถว หรือ กล่าวได้ว่าตำแหน่งของการกดคีย์ได้จากการเขียน โปรแกรมสแกนคีย์ด้านแถว ข้อดีของการเชื่อม ต่อคีย์บอร์ด คือ สามารถใช้สวิตช์ได้มากขึ้น ในขณะที่สิ้นเปลืองขาพอร์ตจำนวนน้อย เช่น ใช้ขา พอร์ตเพียง 16 ขาพอร์ต สามารถต่อคีย์บอร์ดได้ถึง 64 คีย์ ข้อเสียของวงจรนี้ คือ ไม่สามารถรับการ กดคีย์พร้อมกันได้ และต้องเขียน โปรแกรมในการตรวจสอบคีย์ที่ยุ่งยากซับซ้อน



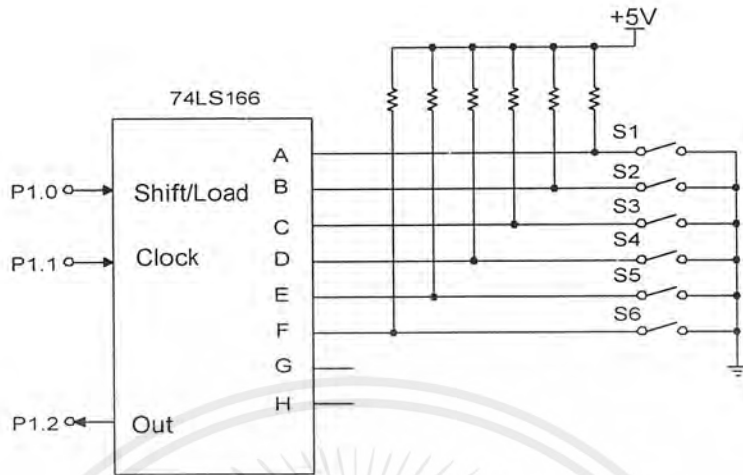
(ข) การเชื่อมต่อสวิตช์แบบเมตริกซ์



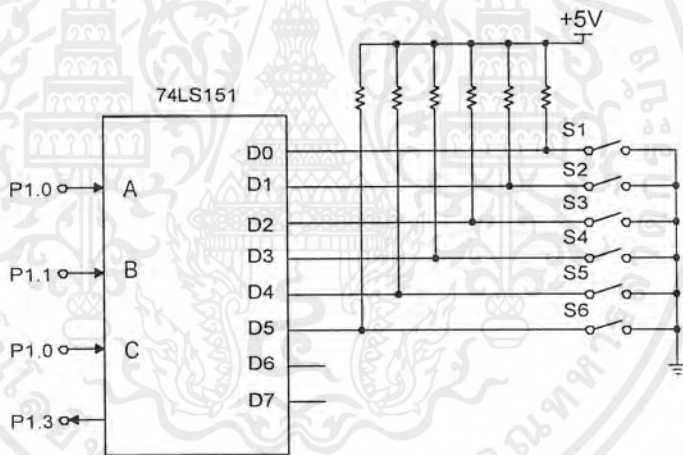
(ค) การเชื่อมต่อสวิตช์แบบเมตริกซ์

รูปที่ 2.16 การเชื่อมต่อสวิตช์แบบต่างๆ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



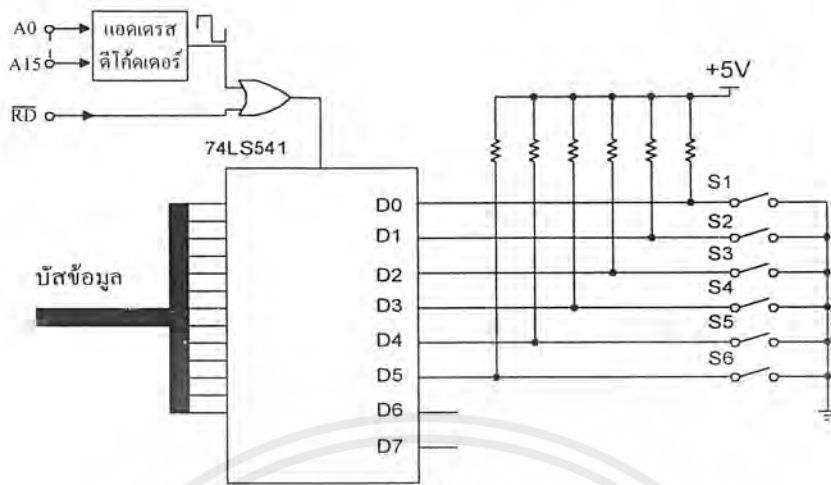
(ค) การเชื่อมต่อสวิตช์ผ่านรีจิสเตอร์



(จ) การเชื่อมต่อสวิตช์แบบมัลติเพล็กซ์

รูปที่ 2.16 (ต่อ) การเชื่อมต่อสวิตช์แบบต่างๆ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



(จ) การเชื่อมต่อสวิตช์เข้ากับระบบบัสข้อมูล

รูปที่ 2.16 (ต่อ) การเชื่อมต่อสวิตช์แบบต่างๆ

2.13.3 แบบเชื่อมต่อสวิตช์ผ่านชิพรีจิสเตอร์

รูปที่ 2.16 (ค) แสดงการเชื่อมต่อสวิตช์ผ่านชิพรีจิสเตอร์ จะเห็นได้ว่าวงจรนี้ต้องการขาพอร์ตน้อยมากเพียง 3 ขาพอร์ต โดยใช้ขาพอร์ตหนึ่งกำหนดโหมดที่ขา SHIFT/LOAD ให้อยู่ในสถานะโหนด เพื่ออ่านสถานะของสวิตช์ทุกตัวเข้าสู่ชิพรีจิสเตอร์ ขึ้นต่อไป ก็เปลี่ยนโหมดการทำงานให้อยู่ในสถานะชิพข้อมูล และให้กำเนิดพัลส์จำนวน 8 พัลส์ เพื่อทำการชิพข้อมูลสถานะของสวิตช์ผ่านขาพอร์ต $P_{1,2}$ ไปใช้งาน นั่นคือใช้ขาพอร์ตเพียง 3 ขาพอร์ตเท่านั้นก็ทำงานได้แล้ว รวมทั้งการเขียนโปรแกรมควบคุมยังสามารถทำได้ง่าย ถ้าต้องการเพิ่มจำนวนก็มากขึ้นทำได้โดยเพิ่มชิพรีจิสเตอร์มากขึ้นให้เท่าเทียมกัน อย่างไรก็ตาม ข้อเสียที่เกิดขึ้น คือใช้เวลาในการอ่านสถานะของคีย์ทั้งหมดเป็นเวลานานจนกระทั่งชิพข้อมูลได้ครบตามจำนวนคีย์

2.13.4 แบบเชื่อมต่อสวิตช์แบบมัลติเพล็กซ์

วงจรการเชื่อมต่อแบบนี้แสดงในรูปที่ 2.16 (ง) ซึ่งต้องการขาพอร์ตจำนวน 4 ขาพอร์ต พอร์ต $P_{1,0}$, $P_{1,1}$ และ $P_{1,2}$ ทำหน้าที่ควบคุมไอซีมัลติเพล็กซ์เซอร์ เพื่อเลือกสวิตช์ S_1-S_6 ที่ต้องการจะติดต่อด้วยสถานะของคีย์ที่ถูกเลือกจะถูกส่งกลับไปยัง ไมโครคอนโทรลเลอร์ผ่านขาพอร์ต $P_{1,3}$

2.13.5 แบบเชื่อมต่อสวิทช์เข้ากับระบบบัสข้อมูล

วงจรการเชื่อมต่อแบบนี้แสดงในรูปที่ 2.16 (จ) ซึ่งทำการเชื่อมต่อสวิทช์เข้ากับระบบบัสข้อมูล เพื่อทำการอ่านสถานะของสวิทช์เข้าสู่ระบบบัสข้อมูลของไมโครคอนโทรลเลอร์ การเชื่อมต่อแบบนี้มักถูกนำมาใช้เมื่อไม่มีขาพอร์ตเหลือไว้ใช้งานเลย การติดต่อกับคีย์บอร์ดทำได้โดยการอ้างตำแหน่งแอดเดรสไปยังวงจรดีโค้ดเดอร์ และทำการเอกติฟิคาสัญญาณ RD เพื่อส่งสัญญาณไปอินาเบิลไปยังไอซีบัฟเฟอร์ส่งผ่านสถานะของคีย์ต่างๆ เข้าสู่ระบบบัสข้อมูล ถึงแม้ว่าถ้าพิจารณาว่าการเขียนโปรแกรมจะเห็นได้ว่าไม่ยากเลย แต่ในทางปฏิบัตินั้น การต่อวงจรแบบนี้จะทำให้เกิดความยุ่งยากในทางฮาร์ดแวร์มาก เพราะจะต้องเชื่อมต่อวงจรเข้ากับทั้งระบบบัสแอดเดรสและบัสข้อมูล

2.14 หน่วยความจำข้อมูล

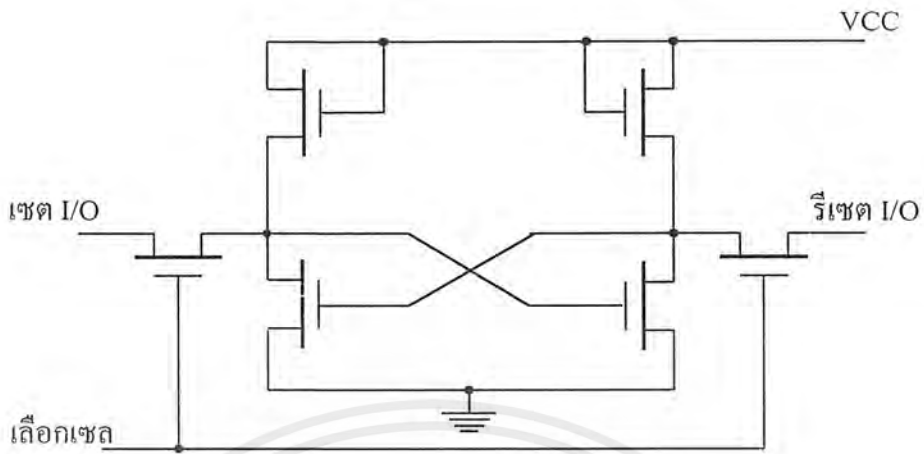
หน่วยความจำชนิดนี้ CPU สามารถอ่านและเขียนข้อมูลได้ เรียกว่า RAM (Random Access Memory) ซึ่งเป็นหน่วยความจำที่ใช้เก็บข้อมูลจากการประมวลผลของ CPU ข้อมูลใน RAM จะคงอยู่ตลอดเวลาที่มีแหล่งจ่ายไฟต่ออยู่กับหน่วยความจำ ตัวหน่วยความจำ RAM นี้ แบ่งออกเป็นสองชนิดใหญ่ๆ คือ

สแตติกแรม (Static RAM)

ไดนามิกแรม (Dynamic RAM)

2.14.1 สแตติกแรม (Static RAM)

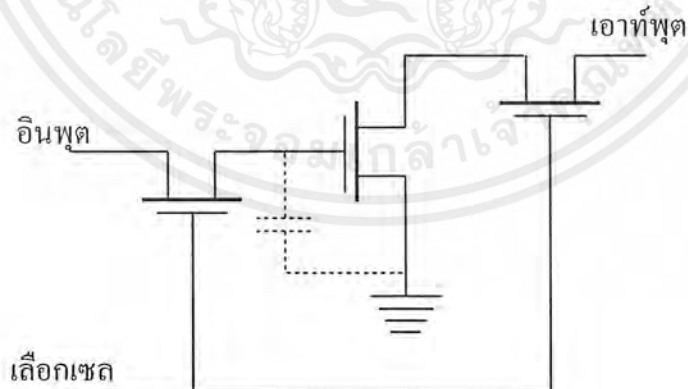
เป็นหน่วยความจำ RAM ที่สามารถอ่านและเขียนข้อมูลได้ และข้อมูลจะยังคงอยู่ตลอดถ้ามีไฟเลี้ยง โครงสร้างภายในในการเก็บข้อมูลแต่ละบิตจะสร้างเป็นฟลิปฟลอป ทำให้การเก็บข้อมูลแต่ละบิตจะต้องสร้างจากรานซิสเตอร์ออกมาหลายตัว ดังรูปที่ 2.17 แสดงโครงสร้างของสแตติกแรมในการเก็บข้อมูล 1 บิต



รูปที่ 2.17 การเก็บข้อมูลแต่ละบิตของสแตติกแรม

2.14.2 ไดนามิกแรม (Dynamic RAM)

เป็นหน่วยความจำที่ข้อมูลจะคงอยู่ตลอดไปถ้ามีไฟเลี้ยง แต่ต้องมีการกระตุ้นหรือการเขียนข้อมูลซ้ำ (Refresh) ตลอดเวลาด้วยวงจรพิเศษ เนื่องจากโครงสร้างภายในของไดนามิกแรมจะสร้างตัวเก็บประจุ ดังนั้นในการเก็บข้อมูล 1 บิต จึงต้องมีการเขียนซ้ำเพื่อให้ประจุคงอยู่ เนื่องจากโครงสร้างภายในเป็นแบบตัวเก็บประจุทำให้ในการเก็บข้อมูล 1 บิต จะสร้างทรานซิสเตอร์ไม่กี่ตัว ซึ่งจะทำให้ความจุของข้อมูลต่อชิพสูงกว่าแบบสแตติกแรม โครงสร้างภายในแสดงดังรูปที่ 2.18

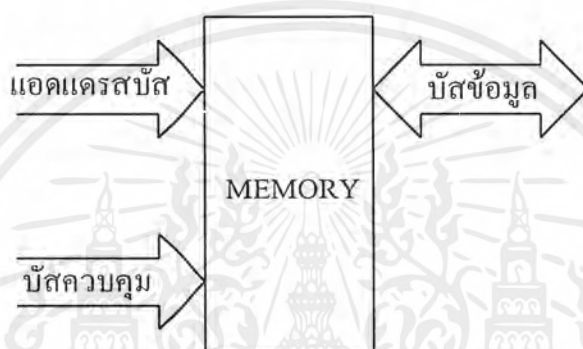


รูปที่ 2.18 โครงสร้างภายในสแตติกแรม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.14.3 การอ่านและเขียนข้อมูลกับหน่วยความจำ

ไอซีหน่วยความจำโดยทั่วไปจะประกอบด้วยกลุ่มสัญญาณ 3 กลุ่ม คือ แอดเดรสบัส, บัสข้อมูล และบัสควบคุม โดยแอดเดรสบัสจะใช้ในการอ้างตำแหน่ง ถ้าแอดเดรสบัส 10 เส้น คือ A0-A9 สามารถอ้างได้ 2^{10} หรือ 1 กิโลไบต์ บัสข้อมูลจะเป็นตัวบอกว่าในแต่ละตำแหน่งจะเก็บข้อมูลได้ที่บิต เช่น ถ้ามีบัสข้อมูล 8 เส้น คือ D0-D7 สามารถเก็บข้อมูลได้ตำแหน่งละ 1 ไบต์ ส่วนบัสควบคุมจะใช้ในการอ่านเขียนหน่วยความจำ ระบบทั้ง 3 ระบบ แสดงได้ดังรูปที่ 2.19



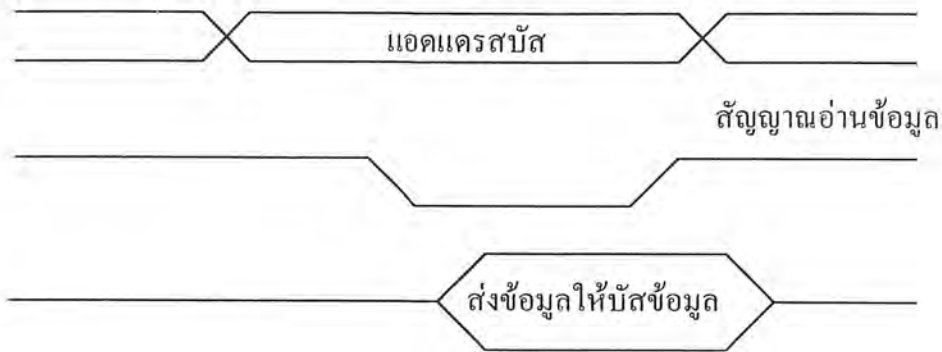
รูปที่ 2.19 กลุ่มสัญญาณที่ใช้ในการติดต่อหน่วยความจำ

โดยทั่วไป ไอซีหน่วยความจำจะมีขา CE (Chip Enable) ไว้สำหรับเลือกไอซีตัวที่ต้องการใช้ทำงาน ในกรณีที่มีไอซีหน่วยความจำหลายๆ ตัว ขา OE (OUTPUT Enable) จะใช้ต่อกับไอซีถอดรหัสสำหรับอ่านข้อมูล ถ้าขานี้แอกทีฟ ข้อมูลที่อยู่ในไอซีหน่วยความจำจะถูกส่งออกมาทางบัสข้อมูลได้

การอ่านข้อมูลจากหน่วยความจำ มีลำดับขั้นดังนี้

1. ส่งตำแหน่งที่จะอ่านไปก่อนทางแอดเดรสบัส
2. ส่งสัญญาณควบคุมว่าต้องการจะอ่าน
3. ข้อมูลภายในไอซีหน่วยความจำจะถูกส่งออกมาทางบัสข้อมูล

การอ่านข้อมูลจากหน่วยความจำสามารถเขียนเป็นแผนผังเวลาได้ดังรูปที่ 2.20



รูปที่ 2.20 สัญญาณในการอ่านข้อมูล

การเขียนข้อมูลลงหน่วยความจำ มีลำดับขั้นดังนี้

1. ส่งตำแหน่งที่จะเขียนข้อมูลออกไปก่อนทางแอดแครสบัส
2. ส่งข้อมูลที่จะเขียนไปทางบัสข้อมูล
3. ส่งสัญญาณเขียนข้อมูล

การเขียนข้อมูลให้หน่วยความจำสามารถเขียนได้เป็นแผนผังเวลาดังรูปที่ 2.21

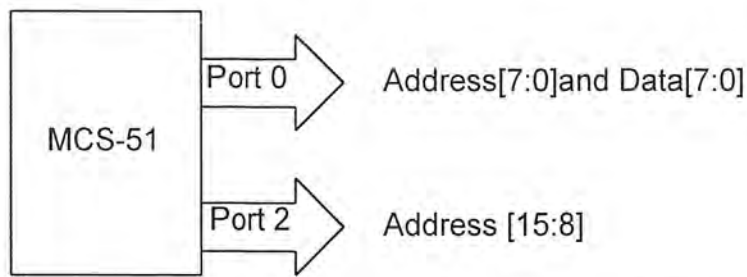


รูปที่ 2.21 แสดงสัญญาณในการเขียนข้อมูล

2.14.4 การเชื่อมต่อหน่วยความจำกับ MCS-51

สัญญาณที่ใช้ในการติดต่อกับหน่วยความจำประกอบด้วยแอดแครสบัส บัสข้อมูล และบัสควบคุม ใน MCS-51 สามารถติดต่อกับหน่วยความจำภายนอกได้ 64 กิโลไบต์ โดยใช้สายสัญญาณแอดแครสบัส 16 เส้นส่งออกมาทางพอร์ต 0 และ พอร์ต 2 จะใช้ Multiplexed ระหว่างแอดแครสบัสและบัสข้อมูล โดยแอดแครสบัสจะเป็นบิตตำแหน่ง A0-A7 ดังรูปที่ 2.22

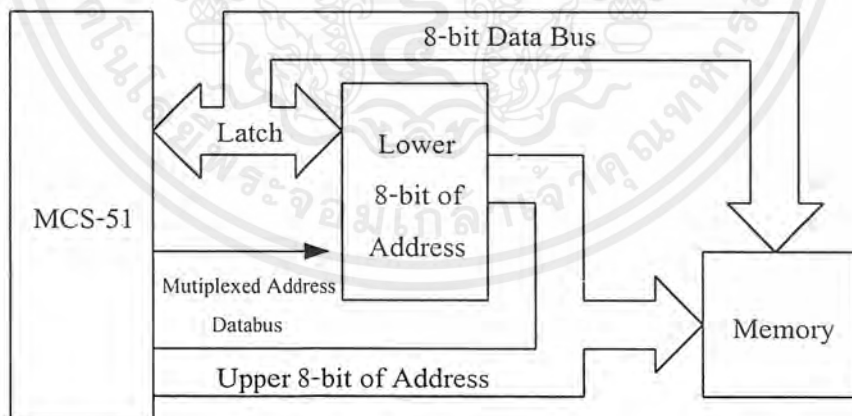
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.22 สายต่อวงจรทางพอร์ต 0 และ พอร์ต 2

ในการเชื่อมต่อหน่วยความจำต้องมีอุปกรณ์ภายนอกมา Latch สัญญาณแอดเดรสบัส ที่ได้จากพอร์ต 0 เพื่อที่จะใช้พอร์ต 0 เป็นบัสข้อมูลต่อไป ในรูปที่ 2.26 จะแสดงหน่วยความจำประเภท ROM และ RAM กับ MCS-51 โดยมีอุปกรณ์ภายนอกมา Latch ค่าแอดเดรสบัสไบต์ต่ำเอาไว้ เรียกว่า “Address Latching”

อุปกรณ์ที่นิยมใช้ได้แก่ ไอซี TTL เบอร์ 74LS373 โดยใช้สัญญาณที่ใช้ Latch คือ สัญญาณ ALE จาก MCS-51



รูปที่ 2.23 การต่อ MCS-51 กับหน่วยความจำ

ในการอ่านหน่วยความจำโปรแกรม และการอ่านเขียนหน่วยความจำข้อมูล การเชื่อมต่อขาแอดเดรสบัสและบัสข้อมูลเหมือนกัน ถ้าหน่วยความจำทั้งสองอยู่ในตำแหน่งเดียวกัน แต่จะแยกกัน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

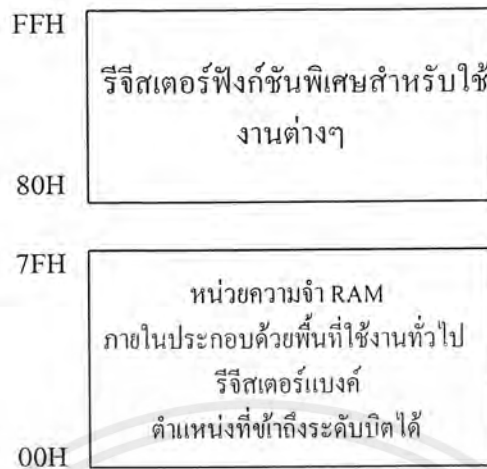
ด้วยสัญญาณควบคุมจากแผนผังเวลา การติดต่อกับหน่วยความจำถ้าเป็นการติดต่อกับ ROM จะใช้สัญญาณ PSEN แต่ถ้าติดต่อกับ RAM จะใช้สัญญาณ RD และ WR ซึ่งแสดงค่าลอจิกของสัญญาณดังนี้

ตารางที่ 2.12 ค่าลอจิกของสัญญาณ

	PSEN	RD	WR
อ่านหน่วยความจำโปรแกรม	0	1	1
อ่านข้อมูลภายนอก	1	0	1
เขียนข้อมูลภายนอก	1	1	0

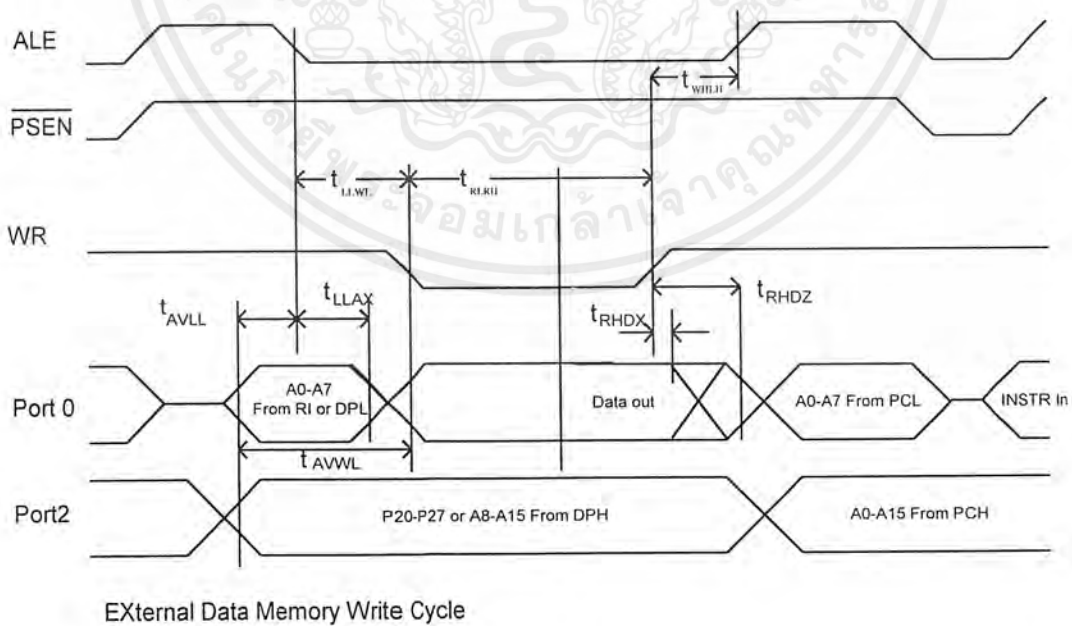
การเชื่อมต่อกับหน่วยความจำข้อมูล

โดยปกติแล้วหน่วยความจำข้อมูลจะใช้สำหรับเก็บข้อมูลที่ได้จากการประมวลผล หน่วยความจำข้อมูลจะเป็นหน่วยความจำที่สามารถอ่านและเขียนข้อมูลได้ โดยจะใช้หน่วยความจำประเภท RAM MCS-51 สามารถอ้างหน่วยความจำข้อมูลได้ 64 กิโลไบต์ โดยแบ่งออกเป็นหน่วยความจำข้อมูลภายใน (Internal Data Memory) และหน่วยความจำข้อมูลภายนอก (External Data Memory) โดยหน่วยความจำข้อมูลภายในเป็นหน่วยความจำข้อมูลที่มีอยู่ใน MCS-51 เอง ซึ่งการจัดเนื้อที่หน่วยความจำข้อมูลภายในแสดงได้ดังรูปที่ 2.24 การติดต่อกับหน่วยความจำข้อมูลภายในจะติดต่อกับคำสั่ง MOV หน่วยความจำข้อมูลภายในนี้จะประกอบด้วยส่วนที่เป็นรีจิสเตอร์ และพื้นที่สำหรับผู้พัฒนาโปรแกรมที่จะใช้งานทั่วไป มีขนาดเพียง 128 หรือ 256 ไบต์ เท่านั้น สำหรับ MCS-51 แต่ละเบอร์ โดยปกติหน่วยความจำข้อมูลภายในจะไม่พอใช้จึงจำเป็นต้องต่อหน่วยความจำข้อมูลภายนอกเพิ่มขึ้น สำหรับการอ่าน และเขียนหน่วยความจำข้อมูลภายนอก แสดงเป็นลำดับสัญญาณเวลาได้ดังรูปที่ 2.25



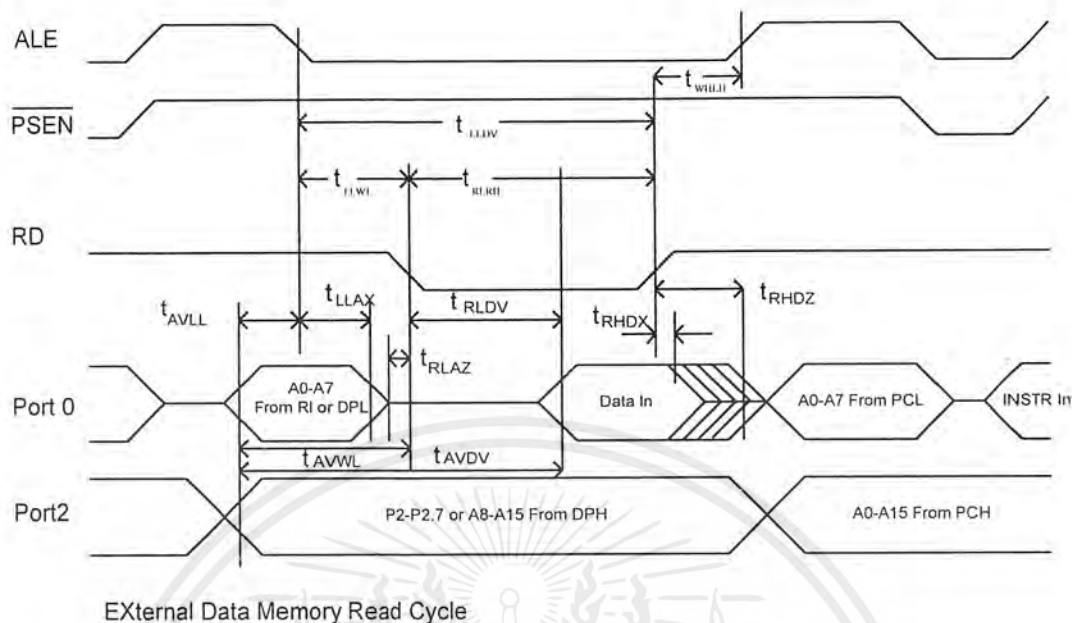
รูปที่ 2.24 การจัดเนื้อที่หน่วยความจำ

จากรูปจะเห็นว่า ถ้า MCS-51 ต้องการอ่านข้อมูลจากหน่วยความจำภายนอก จะใช้สัญญาณ RD และถ้าต้องการเขียนข้อมูลให้กับหน่วยความจำข้อมูลภายนอกจะใช้สัญญาณ WR ในขณะที่สัญญาณ PSEN ไม่แอกติฟ ดังนั้น การเชื่อมต่อกับหน่วยความจำข้อมูลภายนอกแสดงได้ดังรูปที่ 2.26



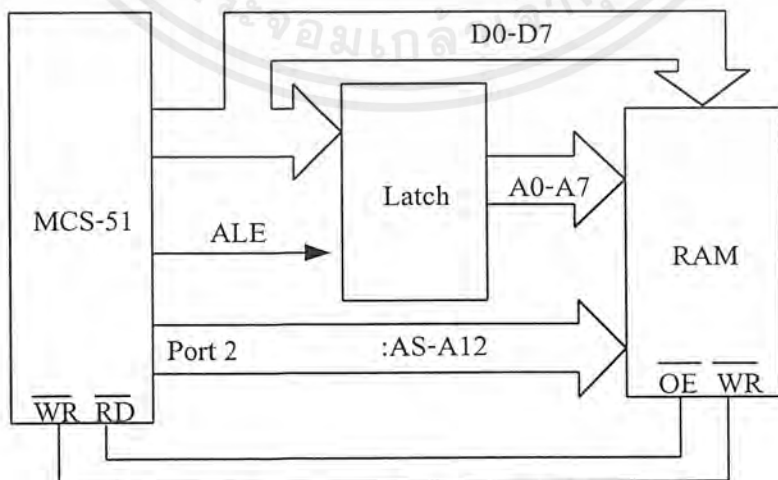
รูปที่ 2.25 แผนผังเวลาการอ่านหน่วยความจำข้อมูลภายนอก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.26 แผนผังเวลาการเขียนหน่วยความจำข้อมูลภายนอก

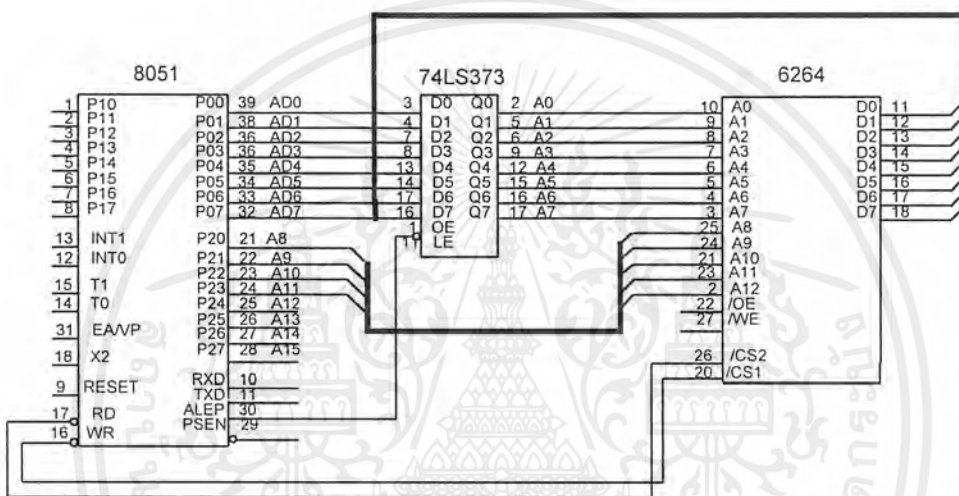
จากรูปจะเห็นว่า ถ้า MCS-51 ต้องการอ่านข้อมูลจากหน่วยความจำภายนอก จะใช้สัญญาณ RD และถ้าต้องการเขียนข้อมูลให้กับหน่วยความจำข้อมูลภายนอกจะใช้สัญญาณ WR ในขณะที่สัญญาณ PSEN ไม่แอคทีฟ ดังนั้น การเชื่อมต่อกับหน่วยความจำข้อมูลภายนอกแสดงได้ดังรูปที่ 2.27



รูปที่ 2.27 การเชื่อมต่อหน่วยความจำข้อมูลภายนอก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การเชื่อมต่อหน่วยความจำข้อมูลภายนอกกับ MCS-51 จะเป็นทำนองเดียวกับการเชื่อมต่อหน่วยความจำโปรแกรมภายนอก ไอซีหน่วยความจำที่จะใช้จะใช้แบบสแตติก (Static RAM) จากแผนผังเวลา จะเห็นว่าเมื่อ MCS-51 ต้องการติดต่อกับหน่วยความจำข้อมูลภายนอกสัญญาณที่ต้องใช้ คือ พอร์ต 0 และ พอร์ต 2 สำหรับเป็นแอดแควเรสส์ สัญญาณ ALE สำหรับ Latch สัญญาณพอร์ต 0 สัญญาณ RD และสัญญาณ WR สำหรับอ่านเขียนข้อมูลลงหน่วยความจำข้อมูลภายนอกกับ MCS-51 อาจแสดงดังรูปที่ 2.28



รูปที่ 2.28 การต่อหน่วยความจำข้อมูลภายนอกกับ MCS-51

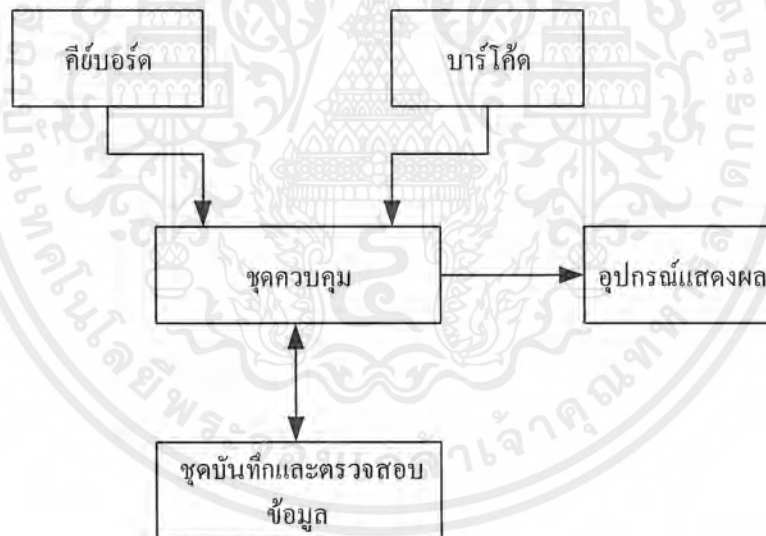
หากต้องการเชื่อมต่อหน่วยความจำข้อมูลภายนอกหลายๆ ตัว กับ MCS-51 จะต้องมียังจรถดครหัสเพื่อเลือกให้หน่วยความจำทำงานในลักษณะเดียวกับการเชื่อมต่อกับหน่วยความจำโปรแกรม

บทที่ 3

การออกแบบ การสร้าง และการทำงาน

3.1 กล่าวนำ

ในการออกแบบเครื่องตรวจสอบบุคคลเข้าหน่วยงานจะประกอบด้วยสองส่วน คือ ส่วนของวงจรควบคุม และส่วนของวงจรับันทึกข้อมูลและตรวจสอบข้อมูล ซึ่งส่วนของวงจรควบคุมนั้น ประกอบด้วยส่วนของวงจรับข้อมูลจากบาร์โค้ด วงจรับข้อมูลจากคีย์บอร์ด วงจรแสดงผล และวงจรการติดต่อสื่อสารแบบอนุกรม RS-232 ในส่วนของวงจรับบันทึกข้อมูลและตรวจสอบข้อมูลนั้นประกอบด้วยวงจรถือสารข้อมูลแบบอนุกรม RS-232 และวงจรของหน่วยความจำ ซึ่งได้แสดงโครงสร้างการทำงานของเครื่องตรวจสอบบุคคลเข้าหน่วยงาน ดังรูปที่ 3.1 ซึ่งมีการทำงานดังนี้



รูปที่ 3.1 โครงสร้างของเครื่องตรวจสอบบุคคลเข้าหน่วยงาน

ส่วนของวงจรับข้อมูลจากเครื่องอ่านบาร์โค้ด วงจรแสดงผล และส่วนของวงจรับข้อมูลจากคีย์บอร์ดจะต่ออยู่กับไมโครคอนโทรลเลอร์ของวงจรควบคุม ซึ่งไมโครคอนโทรลเลอร์นี้จะเป็นส่วนที่รับข้อมูลบาร์โค้ดจากบัตรเข้ามา และส่งข้อมูลที่อ่านได้ไปยังวงจรแสดงผล เมื่อผู้ที่ต้องการผ่านเข้า – ออกภายในหน่วยงานจะต้องทำการรูดบัตรที่มีรหัสบาร์โค้ดผ่านตัวบาร์โค้ด หรือส่วนรับข้อมูลจากคีย์บอร์ดซึ่งจะรับรหัสประจำตัวเป็นตัวเลขบนคีย์บอร์ด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

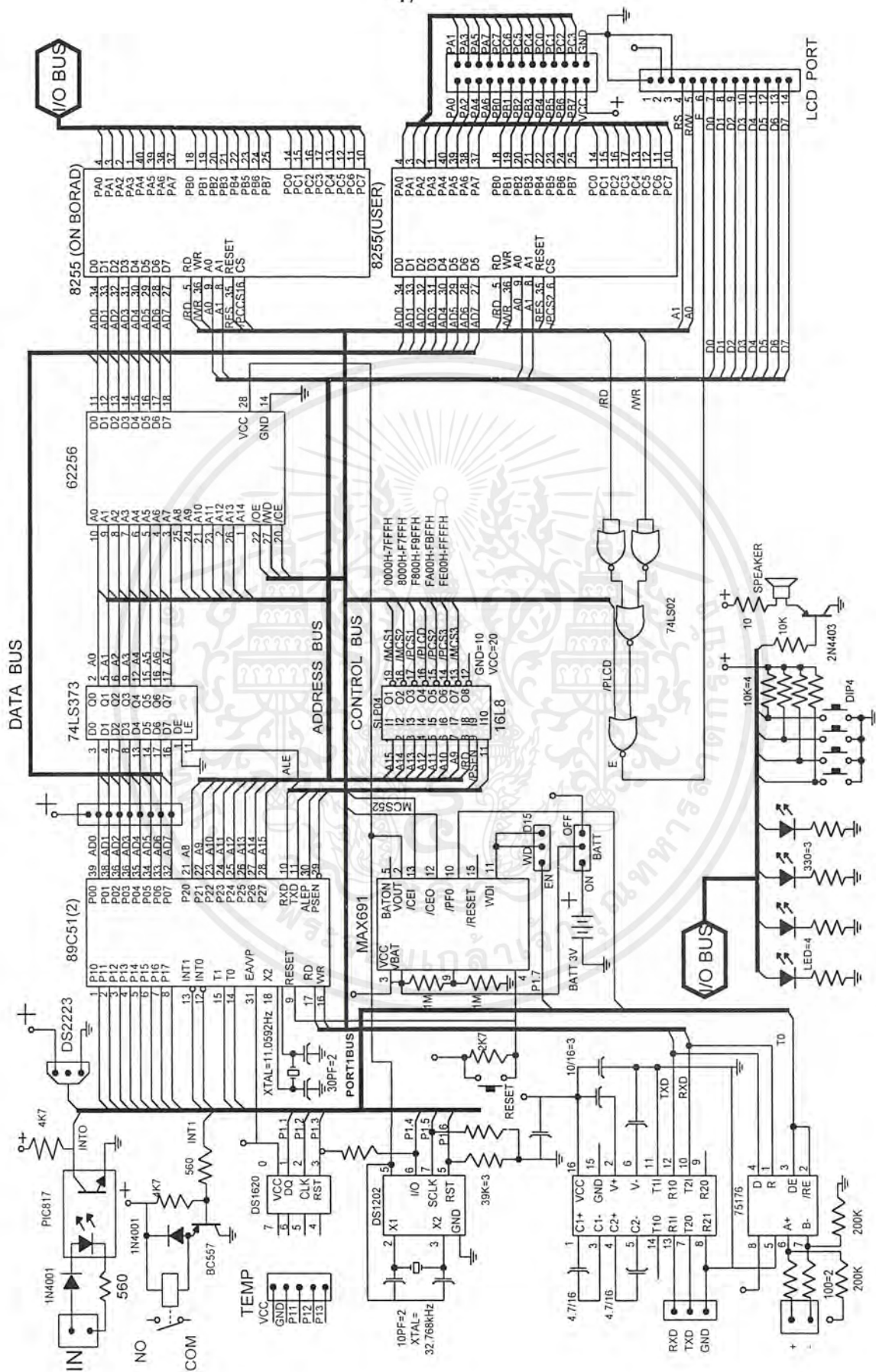
จากนั้นไมโครคอนโทรลเลอร์จะรับรหัสประจำตัวมาจากเครื่องอ่านบาร์โค้ด หรือจาก คีย์บอร์ด เพื่อทำการประมวลผลข้อมูลที่รับเข้ามา แล้วส่งสัญญาณไปแสดงผลที่อุปกรณ์แสดงผล ซึ่งก็คือ รหัสประจำตัว และพร้อมกันนั้นจะส่งข้อมูลที่ทำให้การประมวลผลได้ผ่านพอร์ตสื่อสารแบบอนุกรม RS-232 ไปยังวงจรมันทีกข้อมูลและตรวจสอบข้อมูล ซึ่งไมโครคอนโทรลเลอร์ของวงจรมันทีกข้อมูลและตรวจสอบข้อมูลนั้นจะทำการรับข้อมูลแล้วนำข้อมูลที่อ่านได้ไปแสดงที่อุปกรณ์แสดงผล และทำการตรวจสอบว่าข้อมูลที่รับมานั้นเป็นข้อมูลที่มีอยู่ในหน่วยงานหรือไม่ ถ้าเป็นบุคคลที่อยู่ในหน่วยงานก็จะแสดงผลที่อุปกรณ์แสดงผลว่าสามารถผ่านเข้า - ออกในหน่วยงานได้ และพร้อมกับทำการบันทึกวันและเวลาที่ทำการเข้า - ออก ในหน่วยงาน แต่ถ้าตรวจสอบแล้วพบว่าข้อมูลที่อ่านได้ไม่มีอยู่ในหน่วยงาน ก็จะทำการแสดงผลที่อุปกรณ์แสดงผลว่าไม่สามารถทำการผ่านเข้า - ออกในหน่วยงานได้

3.2 การออกแบบวงจร

การออกแบบเครื่องตรวจสอบบุคคลเข้าหน่วยงานจะแยกเป็นสองส่วนใหญ่ ๆ คือ ส่วนของฮาร์ดแวร์ และส่วนของซอฟต์แวร์ ซึ่งในส่วนของฮาร์ดแวร์นั้นประกอบด้วยสามส่วน คือ วงจรควบคุม , ส่วนของวงจรมันทีกข้อมูลและตรวจสอบข้อมูล และส่วนวงจรติดต่อสื่อสาร ซึ่งมีการทำงานดังนี้

3.2.1 ส่วนวงจรควบคุม

ส่วนของวงจรควบคุมจะทำหน้าที่ควบคุมการรับและส่งข้อมูล การแสดงผล รวมทั้งการประมวลผล เพื่อทำการรับข้อมูลและส่งข้อมูลระหว่างไมโครคอนโทรลเลอร์ของวงจรควบคุมกับไมโครคอนโทรลเลอร์ของวงจรมันทีกข้อมูลและตรวจสอบข้อมูล ซึ่งในวงจรควบคุมนี้ได้ใช้บอร์ด ANT-C51M ซึ่งมีไอซีไมโครคอนโทรลเลอร์เบอร์ 89C51 เป็นตัวประมวลผลข้อมูลที่รับเข้ามาทางเครื่องอ่านบาร์โค้ด และคีย์บอร์ด เมื่อไมโครคอนโทรลเลอร์ทำการประมวลผลข้อมูลแล้วจะนำข้อมูลที่ไปแสดงผลออกทางอุปกรณ์แสดงผล พร้อมกันกับส่งผ่านพอร์ตสื่อสารอนุกรม RS-232 เพื่อไปตรวจสอบข้อมูลที่ได้ว่าตรงกับฐานข้อมูลของหน่วยงานที่ทำการบันทึกไว้หรือไม่ และเมื่อไมโครคอนโทรลเลอร์ของวงจรมันทีกข้อมูลและตรวจสอบข้อมูลได้ประมวลผลเสร็จก็จะส่งข้อมูลมาที่ส่วนของวงจรควบคุม เพื่อทำการแสดงผลออกอุปกรณ์แสดงผล LCD ว่าสามารถทำการเข้า - ออกหน่วยงานได้หรือไม่ แล้วจะกลับไปรอรับข้อมูลต่อไป



รูปที่ 3.2 วงจรควบคุมจาก ANT-C51M

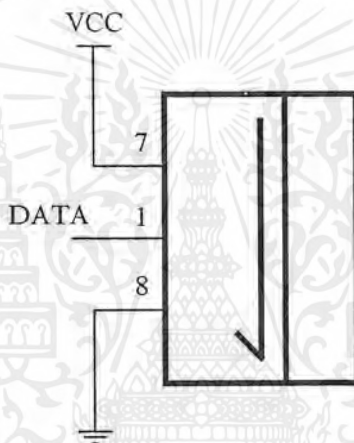
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่นอนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ส่วนของวงจรรับข้อมูล

ในส่วนของวงจรรับข้อมูลทำหน้าที่รับข้อมูลเพื่อส่งข้อมูลให้แก่ไมโครคอนโทรลเลอร์ เพื่อนำไปประมวลผล ซึ่งสามารถแบ่งได้เป็นสองส่วน คือ

ส่วนรับข้อมูลจากเครื่องอ่านบาร์โค้ด

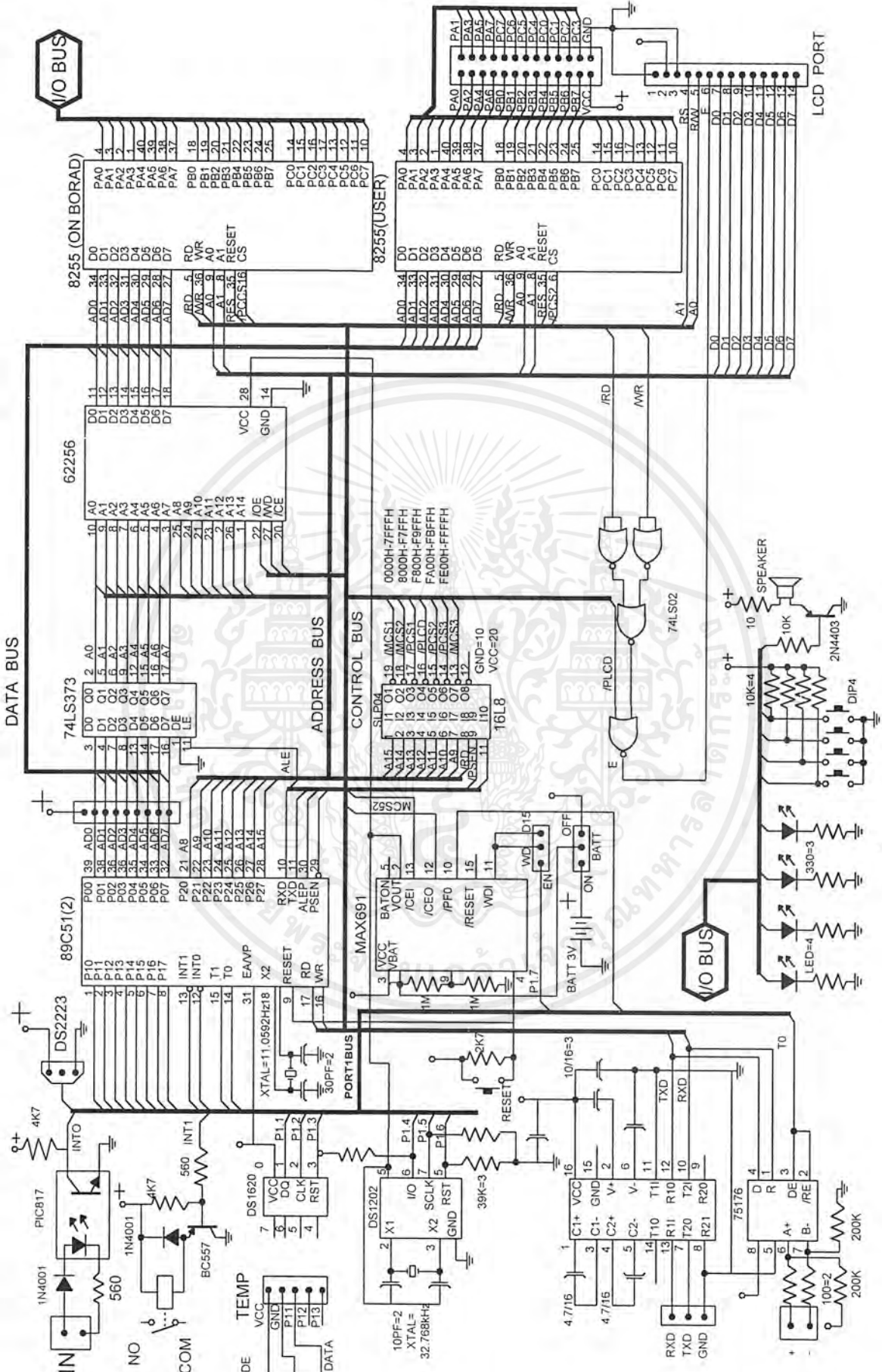
ในส่วนของวงจรรับข้อมูลจากเครื่องอ่านบาร์โค้ดทำหน้าที่ในการรับข้อมูลจากการรูดบัตรนักศึกษา จากนั้นทำการส่งข้อมูลให้แก่พอร์ตอินพุตของไอซี 89C51 เพื่อนำไปประมวลผลแสดงในรูปที่ 3.3



รูปที่ 3.3 ชุดเครื่องอ่านบาร์โค้ด

ในการออกแบบส่วนของการอ่านข้อมูลรหัสบาร์โค้ดนั้นได้นำบาร์โค้ดไปต่อร่วมกับวงจรควบคุม ซึ่งในส่วนของวงจรควบคุมประกอบด้วยไอซีไมโครคอนโทรลเลอร์ 89C51 หน่วยความจำ ROM และ RAM เพื่อใช้ในการเก็บข้อมูล

เมื่อมีผู้ต้องการใช้เครื่องตรวจสอบบุคคลเข้าหน่วยงาน ผู้ใช้สามารถกระทำได้โดยการนำบัตรนักศึกษาที่มีรหัสบาร์โค้ด ซึ่งเป็นรหัสประจำตัวของผู้ถือบัตร ในการเข้าใช้เครื่องได้นั้นรหัสของบัตรผู้นั้นต้องมีรหัสบันทึกไว้ในฐานข้อมูลเรียบร้อยแล้ว เมื่อทำการรูดบัตรนักศึกษาผ่านเครื่องอ่านบาร์โค้ด รหัสข้อมูลจากบัตรจะถูกอ่านข้อมูลออกมา และข้อมูลจะถูกส่งออกที่ขา DATA ของบาร์โค้ด รหัสข้อมูลที่ออกมาจากเอาต์พุตของเครื่องอ่านบาร์โค้ด จะส่งข้อมูลไปยังพอร์ตอินพุต P1.0 ของไอซีไมโครคอนโทรลเลอร์ 89C51 โดยที่ไมโครคอนโทรลเลอร์จะทำหน้าที่แปลงข้อมูลที่อ่านได้จากบาร์โค้ดเป็นรหัส ASCII และทำการประมวลผลต่อไปแสดงดังรูปที่ 3.4

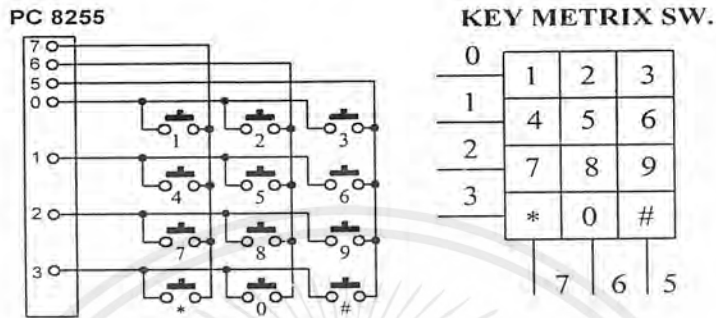


รูปที่ 3.4 วงจรรับข้อมูลจากบาร์โค้ด

เอกสารนี้เป็นเอกสารที่สงวนไว้ใช้สำหรับใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ส่วนของวงจรรับข้อมูลคีย์บอร์ด

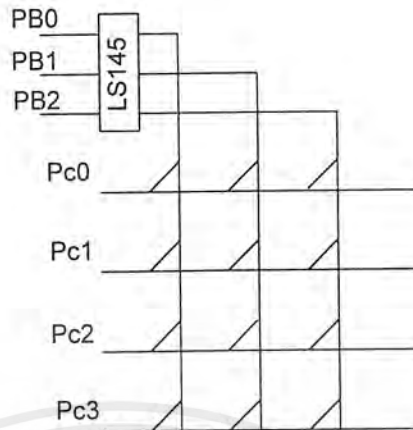
ในส่วนของวงจรรับข้อมูลคีย์บอร์ด จะทำหน้าที่ในการรับสัญญาณข้อมูลจากการกดรหัสของผู้ที่ต้องการใช้เครื่องคอมพิวเตอร์บนคีย์บอร์ด



รูปที่ 3.5 คีย์เมทริกซ์สวิตช์

คีย์บอร์ดจะประกอบด้วย 12 ปุ่มกด คือ ตัวเลข 0-9 และเครื่องหมาย 2 เครื่องหมาย คือ * และ # โดยแบ่งการเดินสายออกเป็นทางแนวนอน และแนวตั้งของแต่ละปุ่มกด ดังรูปที่ 3.5 คีย์บอร์ดจะให้ผู้ใช้กดรหัส โดยการกดแต่ละครั้งจะได้เอาต์พุตทางแนวตั้งและแนวนอนอย่างละค่า โดยที่เอาต์พุตจะให้รหัสที่ไม่ซ้ำกัน 12 ค่า และจะมีความต้านทานต่อแบบ Pull up ที่ 4 บิตบนของพอร์ต C

การทำงานของคีย์บอร์ดจะเริ่มจากการรับข้อมูลทาง PB0-PB1 ของพอร์ต B ซึ่งต่อกับไอซี 74LS145 และ PC0-PC3 ของพอร์ต C ซึ่งจะรอรับข้อมูลเมื่อมีผู้ใช้กดคีย์บอร์ดตัวใดตัวหนึ่งบนปุ่มกด รหัสข้อมูลจะถูกส่งไปให้อินพุตของไอซี 8255 ที่พอร์ต B และพอร์ต C ดังรูปที่ 3.6

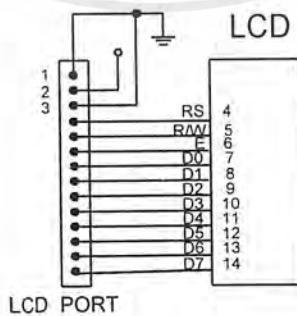


รูปที่ 3.6 ส่วนรับข้อมูลเมทริกซ์สวิตช์

ส่วนของการแสดงผล

ในส่วนของการแสดงผลจะใช้อุปกรณ์แสดงผล คือ จอ LCD ทำหน้าที่ในการแสดงผลข้อมูลที่ได้จากประมวลผลของไมโครคอนโทรลเลอร์ที่ได้รับข้อมูลมาจากเครื่องอ่านบาร์โค้ด หรือ คีย์บอร์ด โดยที่จอแสดงผล LCD บรรทัดที่ 1 จะแสดงรหัสประจำตัวของผู้ใช้และจะแสดงสถานะของผู้ใช้ว่าผ่านหรือไม่ผ่าน

การทำงานของวงจรแสดงผล เริ่มจากการส่งข้อมูลกำหนดการอ่านหรือเขียนแล้วจึงกำหนดการอ่านหรือเขียนนั้นเป็นข้อมูล หรือคำสั่ง (RS) ส่งข้อมูลออกไปทางขา Data (D0-D7) เป็นการทำงานแบบ 8 บิต การแสดงผลบรรทัดที่ 1 และ 2 ต้องกำหนดบัสแอดเดรส ซึ่งเป็นตำแหน่งในการแสดงผลบรรทัดที่ 1 จะอยู่ที่แอดเดรส 80h บรรทัดที่ 2 จะอยู่ที่แอดเดรส C0h



รูปที่ 3.7 ลักษณะการแสดงผลของ LCD

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.2.2 วงจรบันทึกข้อมูลและตรวจสอบข้อมูล

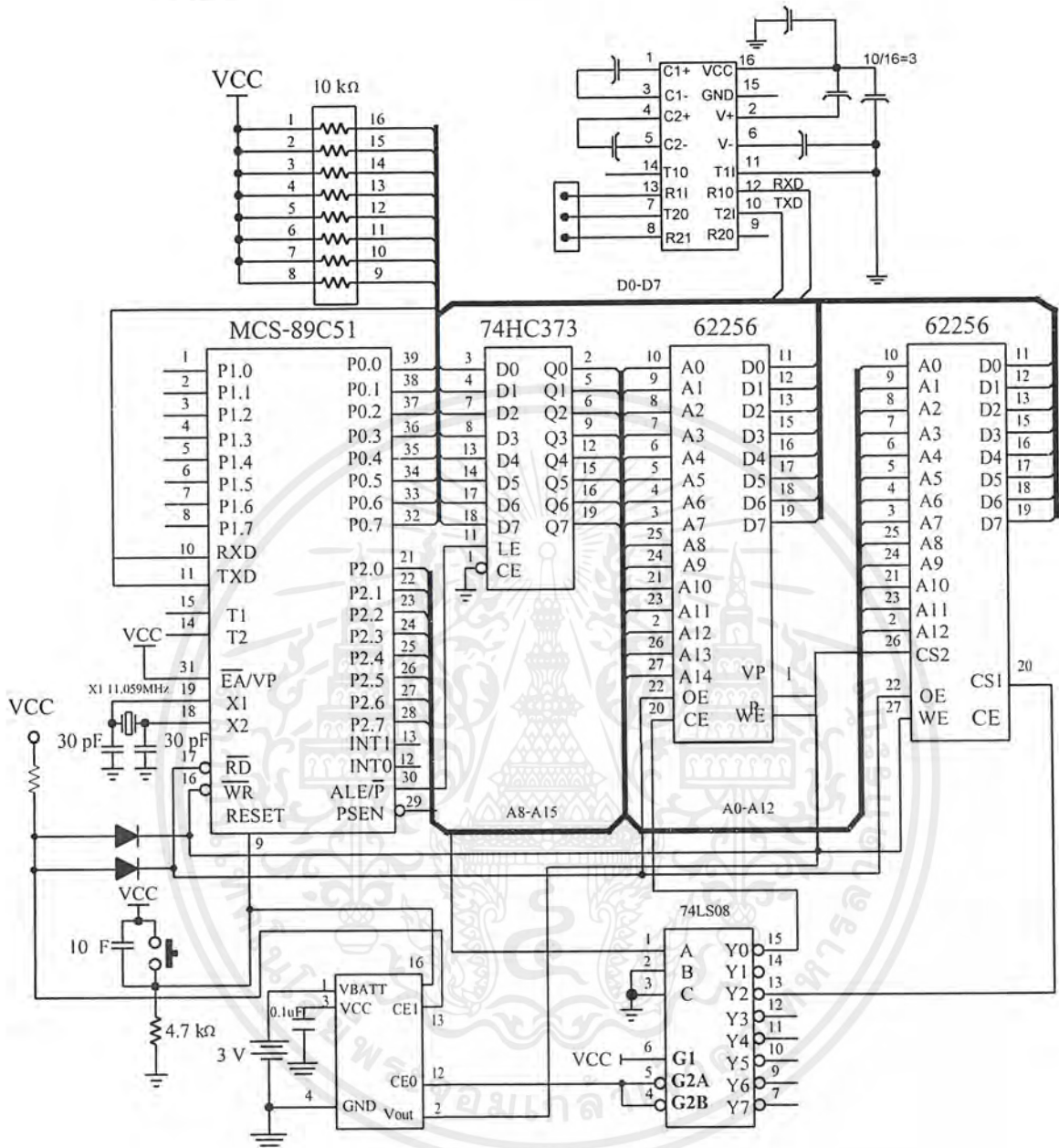
เป็นวงจรที่ทำหน้าที่บันทึกและตรวจสอบข้อมูล รวมถึงการส่งข้อมูลผ่านพอร์ตสื่อสารอนุกรมแบบ RS – 232 ระหว่างไมโครคอนโทรลเลอร์ของชุดควบคุม และไมโครคอนโทรลเลอร์ของชุดบันทึกข้อมูลและตรวจสอบข้อมูล ซึ่งภายในชุดบันทึกข้อมูลและตรวจสอบข้อมูลนี้ประกอบไปด้วยส่วน 2 ส่วน คือ

ส่วนของการตรวจสอบข้อมูล ประกอบด้วย ไอซีไมโครคอนโทรลเลอร์ 89C51 ซึ่งเป็นตัวประมวลผลข้อมูลที่ได้รับเข้ามาทางพอร์ตอนุกรมเมื่อทำการประมวลผลข้อมูลว่าข้อมูลที่ได้มีในฐานข้อมูลของหน่วยงานหรือไม่ แล้วจึงส่งผลที่ได้ไปปรากฏที่อุปกรณ์แสดงผล

ส่วนของการบันทึกข้อมูล ในส่วนนี้แบ่งออกเป็น 2 ส่วนด้วยกันคือ

ส่วนการบันทึกฐานข้อมูล จะนำรหัสที่ได้รับมาบันทึกเป็นฐานข้อมูล ซึ่งรหัสแต่ละรหัสนั้นประกอบด้วยอักขระจำนวน 10 อักขระ ในแต่ละอักขระมีจำนวน 8 บิต ดังนั้นในรหัสหนึ่งๆ จะมีทั้งหมด 80 บิต และเครื่องตรวจสอบหน่วยงานเครื่องนี้จะทำการเก็บรหัสเพื่อเป็นฐานข้อมูลไว้ในหน่วยความจำที่ตำแหน่ง 0000H – 099C0H ซึ่งมีจำนวน 2,560 บิต ดังนั้นในส่วนนี้สามารถนำรหัสมาทำการบันทึกฐานข้อมูลได้สูงสุด คือ 256 รหัส

ส่วนการบันทึกข้อมูลการเข้า – ออก ประกอบด้วย รหัส วัน-เดือน-ปี และเวลาที่ผ่านเข้า – ออกในหน่วยงาน ซึ่งข้อมูลที่ต้องทำการบันทึกนี้มีอักขระจำนวน 22 อักขระ แต่ละอักขระมี 8 บิต ในการบันทึกข้อมูลการเข้า – ออก ของรหัสหนึ่งๆ (จำนวน 176 บิต) ไว้ในหน่วยความจำที่ตำแหน่ง 3000H – 0FFFFH ซึ่งสามารถทำการบันทึกข้อมูลการเข้า – ออกในหน่วยงานได้สูงสุด คือ 2,792 รหัส



รูปที่ 3.8 วงจรบันทึกข้อมูลและตรวจสอบข้อมูล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 4

การทดลองและผลการทดลอง

จากการทดลองและการตรวจสอบการทำงานของระบบได้แบ่งการทดลองออกเป็น 3 ส่วน คือ ส่วนที่หนึ่งเป็นการทดลองส่วนของวงจรควบคุมจากบอร์ด ANT-C51M ส่วนที่สองเป็นส่วนของการทดลองการบันทึกข้อมูล และการตรวจสอบข้อมูล ส่วนที่สาม คือ ส่วนของการสื่อสารข้อมูล

4.1 การทดลองส่วนของวงจรควบคุมจากบอร์ด ANT-C51M

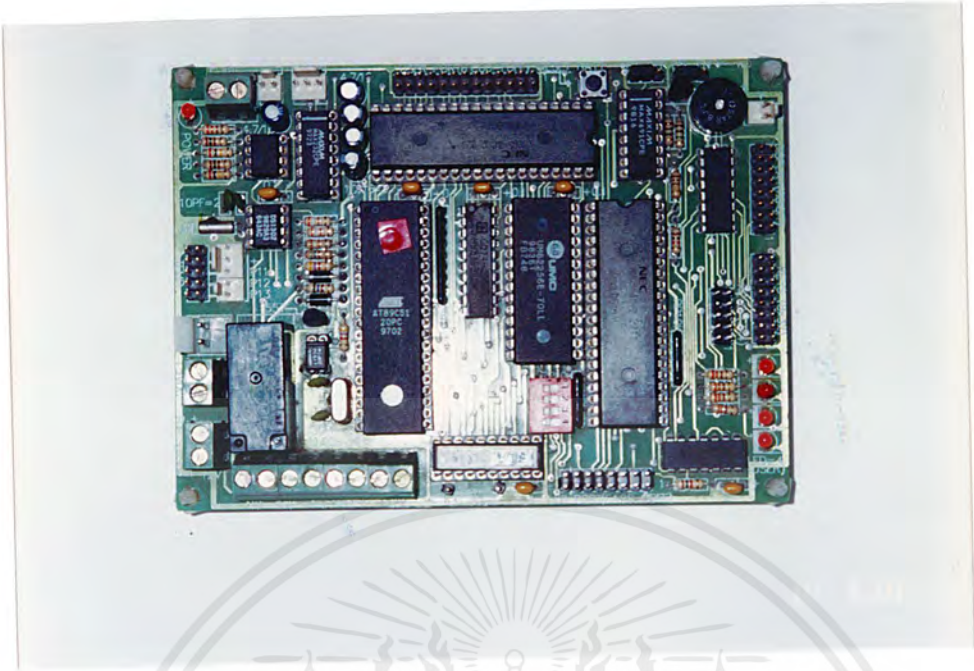
4.1.1 การทดลองส่วนของการแสดงผล

ลำดับขั้นการทดลอง

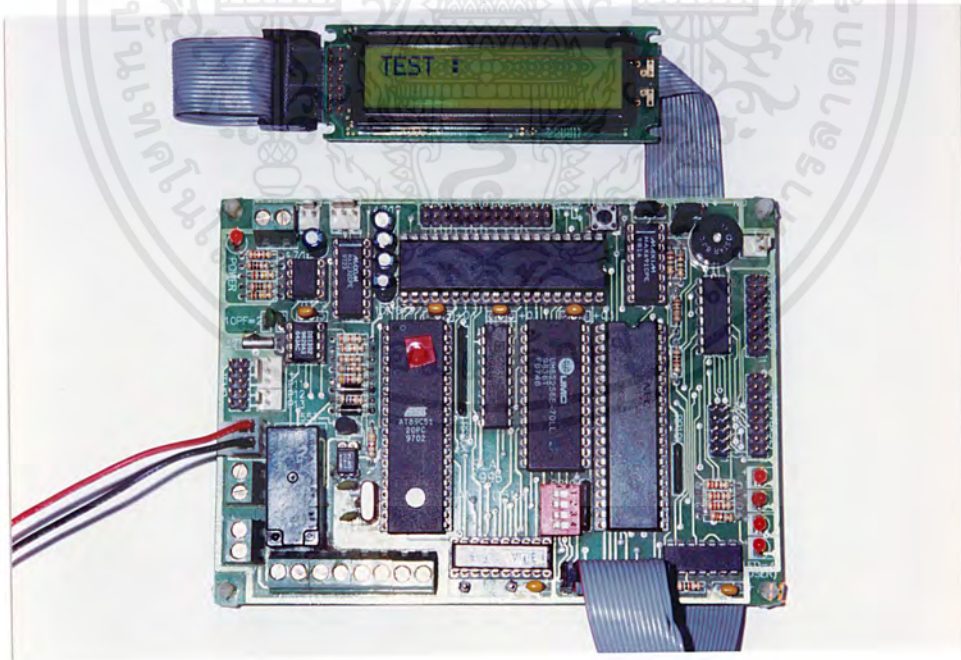
1. ประกอบวงจรควบคุม และอุปกรณ์แสดงผลตามรูปที่ 4.2
2. ตรวจสอบความเรียบร้อยของวงจร
3. เขียนโปรแกรมแสดงผลออกที่อุปกรณ์แสดงผล เพื่อทดสอบการทำงานของอุปกรณ์แสดงผลที่ต่อกับบอร์ด ANT-C51M ที่พอร์ต LCD PORT (FA00-FBFF)
4. ทำการจ่ายไฟให้กับวงจร เพื่อทดลอง โปรแกรมแสดงผล
5. ทำการกดสวิตช์รีเซ็ต เพื่อสังเกตการแสดงผลที่อุปกรณ์แสดงผล

ผลการทดลอง

จากการทดลองส่วนของวงจรควบคุมและอุปกรณ์แสดงผล ได้ทำการทดลองเขียนโปรแกรมแสดงผลออกทางพอร์ต LCD PORT (FA00-FBFF) ของบอร์ด ANT-C51M โดยเขียนโปรแกรมสั่งให้ตัวอักษร TEST ไปปรากฏที่อุปกรณ์แสดงผล จากการทดลองวงจรควบคุมสามารถส่งตัวอักษร TEST ออกที่ไปปรากฏที่อุปกรณ์แสดงผลตามที่ได้โปรแกรมได้



รูปที่ 4.1 บอร์ดวงจรควบคุมที่สามารถทำงานได้



รูปที่ 4.2 การทดลองนำข้อมูลแสดงผลที่ LCD

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

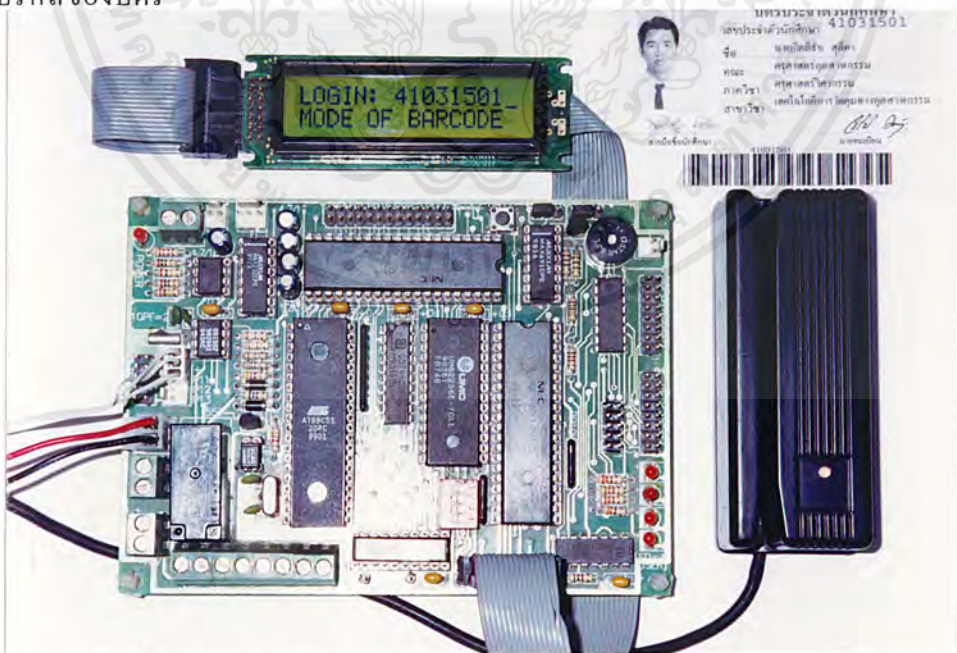
4.1.2 การทดลองส่วนของการรับข้อมูลจากเครื่องอ่านบาร์โค้ด

ลำดับขั้นการทดลอง

1. นำวงจรควบคุมที่ได้ทำการทดลองในหัวข้อ 4.2 มาทำการทดลองอ่านข้อมูลบาร์โค้ดจากเครื่องอ่านบาร์โค้ดดังรูปที่ 4.3 โดยนำขา DATA ของเครื่องอ่านบาร์โค้ดต่อที่พอร์ต P1.1 ของไอซี ไมโครคอนโทรลเลอร์ 89C51 และจ่ายไฟให้กับบาร์โค้ด ดังรูปวงจรที่ 3.4
2. ตรวจสอบความเรียบร้อยของวงจร
3. เขียนโปรแกรมที่ใช้ในการอ่านบาร์โค้ด และแสดงผลที่อุปกรณ์แสดงผล
4. ทำการจ่ายไฟให้กับวงจรควบคุม เพื่อทดลองโปรแกรมที่ใช้ในการอ่านบาร์โค้ด และแสดงผลที่อุปกรณ์แสดงผล
5. กดสวิทช์รีเซต
6. นำบัตรที่มีรหัสบาร์โค้ดมาทำการรูดบัตรผ่านเครื่องอ่านบาร์โค้ด สังเกตการแสดงผล

ผลการทดลอง

จากการทดลองส่วนของการรับข้อมูลจากเครื่องอ่านบาร์โค้ด พบว่าเมื่อทำการรูดบัตรแล้ว ไมโครคอนโทรลเลอร์ 89C51 จะอ่านข้อมูลจากบาร์โค้ด และแสดงผลออกที่อุปกรณ์แสดงผลได้ ตรงกับรหัสของบัตร



รูปที่ 4.3 การต่อวงจรควบคุม อุปกรณ์แสดงผลและเครื่องอ่านบาร์โค้ด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

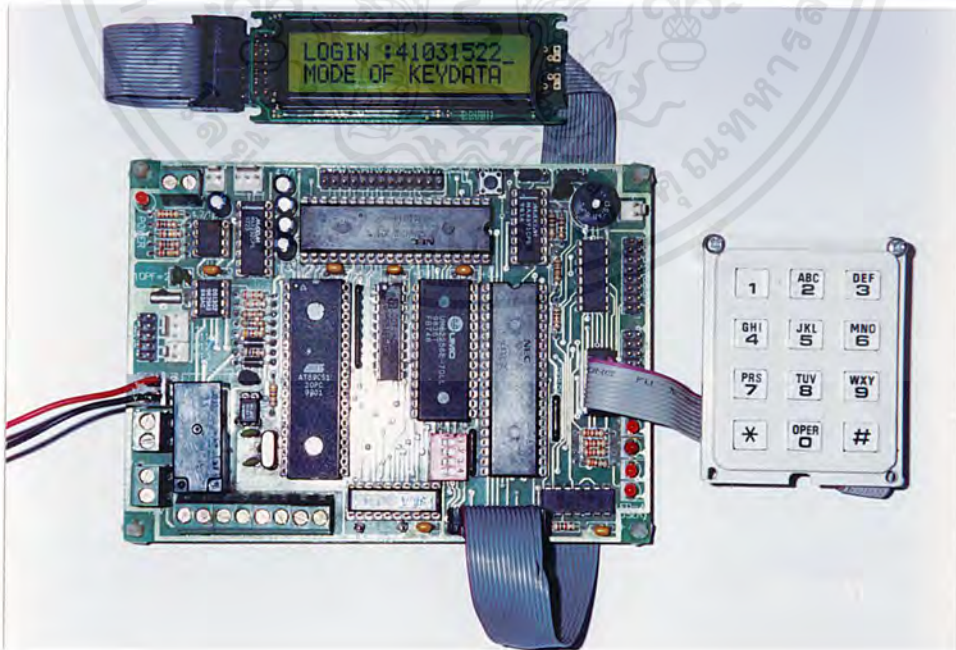
4.1.3 การทดลองส่วนรับข้อมูลจากคีย์บอร์ด

ลำดับขั้นตอนการทดลอง

1. นำคีย์บอร์ดต่อรวมกับวงจรควบคุมในการทดลองที่ 4.2 ที่พอร์ต KEY (3x4)
2. ตรวจสอบความเรียบร้อยของวงจร
3. เขียนโปรแกรมอ่านข้อมูลจากการกดคีย์บอร์ดและแสดงผลออกที่อุปกรณ์แสดงผล โดยกำหนดค่าคีย์แต่ละคีย์ และให้มีค่าตามรหัส BCD 8421
4. ทำการจ่ายไฟให้กับวงจรควบคุม เพื่อทดลองโปรแกรมอ่านข้อมูลจากการกดคีย์บอร์ด สวิตซ์
5. กดสวิตซ์รีเซต
6. ทำการกดคีย์บอร์ด สังเกตการแสดงผลที่อุปกรณ์แสดงผลว่าได้ผลตรงกับค่าที่กดไปหรือไม่

ผลการทดลอง

จากการทดลองเขียนโปรแกรมรับค่าจากคีย์บอร์ด เมื่อทำการกดคีย์บอร์ดจะพบว่าอุปกรณ์แสดงผลปรากฏค่าประจำคีย์บอร์ดได้ตรงกันดังนี้



รูปที่ 4.4 ผลอุปกรณ์แสดงผล เมื่อมีการกดค่าคีย์บอร์ด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

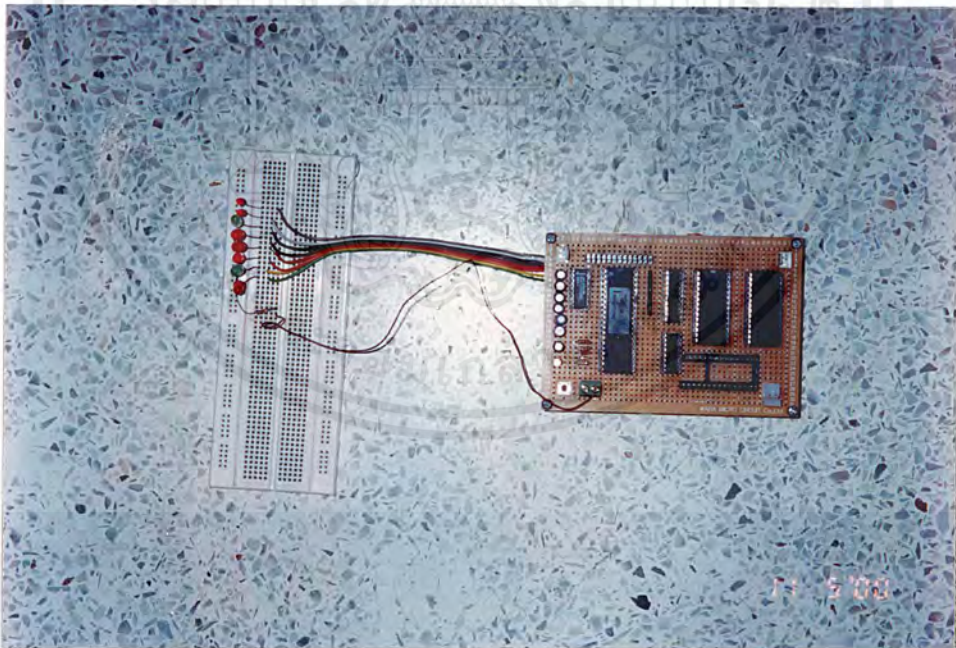
4.2 การทดลองส่วนของการบันทึกข้อมูลและการตรวจสอบข้อมูล

ลำดับขั้นการทดลอง

1. นำวงจรของการบันทึกข้อมูล และการตรวจสอบข้อมูลตามรูปที่ 3.9 ที่ขา P1.0-P1.7 ของไมโครคอนโทรลเลอร์ 89C51 มาต่อกับหลอด LED จำนวน 8 ดวง
2. ตรวจสอบความเรียบร้อยของวงจร
3. เขียนโปรแกรมส่งค่าไปเก็บไว้ในหน่วยความจำ
4. ทำการจ่ายไฟให้กับวงจร สังเกตผลที่เกิดขึ้นกับหลอด LED ทั้ง 8 ดวง

ผลการทดลอง

จากการทดลองส่งค่าออกไมโครคอนโทรลเลอร์ทางพอร์ต P1.0-P1.7 นั้น พบว่าจะทำให้หลอด LED ทั้ง 8 ดวงนั้นสว่างตามค่าที่ส่งไป แสดงว่าการบันทึกข้อมูลและการตรวจสอบข้อมูลของวงจรนี้ใช้งานได้



รูปที่ 4.5 วงจรบันทึกข้อมูลและการตรวจสอบข้อมูล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.3 การทดลองส่วนของการสื่อสารข้อมูล

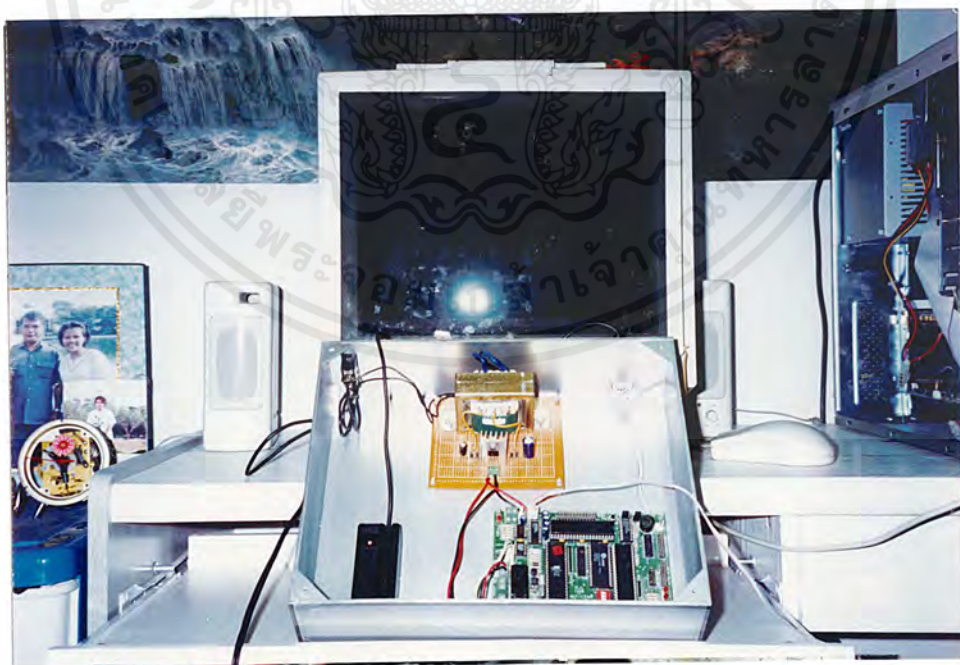
4.3.1 การทดลองการสื่อสารข้อมูลระหว่างวงจรควบคุมกับคอมพิวเตอร์

ลำดับขั้นการทดลอง

1. นำวงจรควบคุมที่พอร์ตของการสื่อสารข้อมูล RS-232 ต่อเข้ากับคอมพิวเตอร์ที่พอร์ต DB9 ดังรูปที่ 4.6
2. ตรวจสอบความเรียบร้อยของวงจร
3. เขียนโปรแกรมส่งค่าไปยังคอมพิวเตอร์โดยใช้ร่วมกับโปรแกรม Xtalk
4. ทำการจ่ายไฟให้กับวงจรควบคุมและทำการรันโปรแกรม Xtalk
5. สังเกตผลที่เกิดขึ้น

ผลการทดลอง

จากการทดลองใช้คอมพิวเตอร์ในการสื่อสารข้อมูลกันผ่านวงจรสื่อสารข้อมูลโดยใช้โปรแกรม Xtalk นั้น ในการทดลองพบว่าสามารถที่จะส่งข้อมูลที่มีอยู่ไปแสดงที่จอคอมพิวเตอร์ได้ แสดงว่าสามารถทำการรับ-ส่งข้อมูลได้เช่นกัน



รูปที่ 4.6 การทดลองวงจรสื่อสารข้อมูลของวงจรควบคุมกับคอมพิวเตอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.3.2 การทดลองรวมส่วนของการสื่อสารข้อมูลผ่าน RS-232 ของวงจรควบคุม กับวงจรบันทึกข้อมูลและตรวจสอบข้อมูล

ลำดับขั้นการทดลอง

1. ประกอบวงจรตามรูปที่ 4.7
2. ตรวจสอบความเรียบร้อยของวงจร
3. เขียนโปรแกรมรับ-ส่งข้อมูลระหว่างวงจรควบคุมกับวงจรบันทึกข้อมูลและตรวจสอบข้อมูล
4. ทำการจ่ายไฟให้วงจรและทำการรันโปรแกรม
5. กดสวิทช์รีเซต
6. ทำการทดลองรับ-ส่ง ข้อมูลจากวงจรตามรูปที่ 4.7 และนำออกไปแสดงผลที่อุปกรณ์แสดงผล

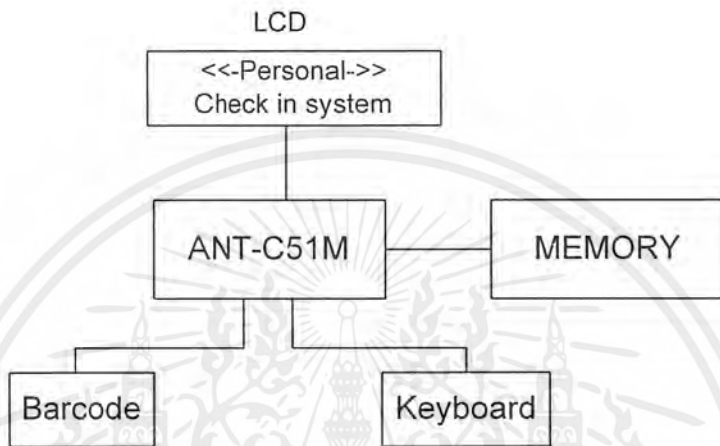
ผลการทดลอง

จากการทดลองวงจรรับ-ส่งข้อมูล และอุปกรณ์แสดงผลเข้าด้วยกัน พบว่าอุปกรณ์รับข้อมูลได้แก่ เครื่องอ่านบาร์โค้ด และเมทริกซ์สวิทช์ และส่วนของอุปกรณ์แสดงผล เมื่อรวมวงจรเข้าด้วยกันแล้วในส่วนของการควบคุมควมต้องมีการปรับปรุง เพื่อให้วงจรแต่ละวงจรสามารถทำงานร่วมกันได้

เมื่อรันโปรแกรมกด สวิทช์รีเซต นำบัตรที่มีรหัสบาร์โค้ดมาจุดในทิศทางขึ้นหรือลงก็ได้ เมื่อทำการรูดบัตรถ้าสามารถอ่านได้จะเกิดเสียงขึ้นที่บัสเซอร์ แต่ถ้าไม่สามารถอ่านได้ก็จะไม่มีเสียงเกิดขึ้น กรณีที่ทำการรูดบัตรแล้วถ้าสามารถอ่านข้อมูลได้ ข้อมูล หรือเลขรหัสในบาร์โค้ดจะปรากฏผลออกที่อุปกรณ์แสดงผล แล้วสักครู่จะแสดงผลว่าข้อมูลของรหัสนี้มีอยู่หรือไม่ ถ้ามีข้อมูลอยู่ อุปกรณ์แสดงผลจะแสดงว่าให้เข้า-ออกหน่วยงานได้ พร้อมกับบันทึกเวลาการเข้า-ออกลงในหน่วยความจำด้วย แต่ถ้าไม่มีข้อมูลของรหัสนี้อยู่ อุปกรณ์แสดงผลจะแสดงผลว่าไม่มีข้อมูล จึงไม่สามารถทำการเข้า-ออกหน่วยงานได้ หลังจากนั้นจะเคลียร์หน้าจอเพื่อรอรับรหัสต่อไป และในการรับข้อมูลจากคีย์เมทริกซ์สวิทช์เมื่อกดเมทริกซ์สวิทช์ที่ * ก็จะเข้าสู่เมนูของการรอรับรหัส จากนั้นจึงทำการกดรหัสต่าง ๆ เมื่อกดรหัสครบแล้วให้ทำการกดยืนยันรหัสนั้นๆ ด้วยการกด * ถ้ารหัสนี้มีข้อมูลอยู่ อุปกรณ์แสดงผลจะทำการแสดงผลว่ามีข้อมูลอยู่สามารถทำการเข้า-ออกหน่วยงานได้ พร้อมกับบันทึกวันเวลาการเข้า-ออกหน่วยงานด้วย แต่ถ้ารหัสนี้ไม่มีในข้อมูลอุปกรณ์แสดงผลจะ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

แสดงผลว่าไม่มีข้อมูลและห้ามเข้า-ออกในหน่วยงาน แล้วสักครู่จะเคลียร์หน้าจอเพื่อรอรับการอ่านบาร์โค้ดต่อไป



รูปที่ 4.7 การทดลองสื่อสารผ่าน RS-232 ของวงจรควบคุมกับวงจรบันทึกข้อมูล

4.3.3 การทดลองเครื่องตรวจสอบบุคคลเข้าหน่วยงาน

การทดลองเครื่องตรวจสอบบุคคลเข้าหน่วยงานได้แบ่งเป็น 8 โหมดการทำงานด้วยกัน คือ การทดลองในโหมดบาร์โค้ด

ลำดับขั้นการทดลอง

1. เปิดเครื่องตรวจสอบบุคคลเข้าหน่วยงาน
2. เลือกโหมดบาร์โค้ดด้วยการกดคีย์ 0 สังเกตผลที่เกิดขึ้น
3. ทำการรูดบัตรบาร์โค้ด สังเกตผลที่เกิดขึ้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ผลการทดลอง

จากการทดลอง เมื่อทำการเปิดเครื่องแล้วพบว่า

1. เมื่อทำการกดคีย์ 0 แล้ว จะอยู่ในโหมดบาร์โค้ด ซึ่งหน้าจอแสดงผลจะปรากฏดังนี้

Login:_
Mode of bar code

2. ทำการรูดบัตรหน้าจอจะปรากฏหมายเลขจากบาร์โค้ด

Login:41031501
Mode of bar code

ซึ่งแสดงว่าเครื่องตรวจสอบบุคคลเข้าหน่วยงานสามารถทำการอ่านรหัสจากบาร์โค้ดได้ แล้วจึงนำรหัสที่อ่านได้ไปตรวจสอบข้อมูลและบันทึกข้อมูลต่อไป ซึ่งการแสดงผลสถานะของผู้ใช้เมื่อมีการรูดบัตรมี 2 แบบ คือ

1. “ผ่าน” หน้าจอจะแสดงคำว่า “YOU MEMBER”
2. “ไม่ผ่าน” หน้าจอจะปรากฏคำว่า “YOU NO MEMBER”

การทดลองในโหมดคีย์บอร์ด

ลำดับขั้นการทดลอง

1. เปิดเครื่องตรวจสอบบุคคลเข้าหน่วยงาน
2. เลือกโหมดคีย์บอร์ดด้วยการกดคีย์ * สังเกตผลที่เกิดขึ้น
3. ทำการกรอกรหัสด้วยคีย์บอร์ด เมื่อกรอกรหัสครบแล้วให้ทำการกดคีย์ * (Enter) อีกครั้ง สังเกตผลที่เกิดขึ้น

ผลการทดลอง

จากการทดลอง เมื่อทำการเปิดเครื่องแล้วพบว่า

1. เมื่อทำการกดคีย์ * จอแสดงผลจะปรากฏอยู่ในโหมดคีย์บอร์ดดังนี้

Login:_
Mode of keyboard

2. ทำการกรอกรหัสลงไปแล้วตามด้วยการกดคีย์ *

Login:41031501
Mode of key

ซึ่งแสดงว่าเครื่องตรวจสอบบุคคลเข้าหน่วยงานสามารถทำการรับรหัสจากคีย์บอร์ดได้แล้วจึงนำรหัสที่อ่านได้ไปตรวจสอบข้อมูลและบันทึกข้อมูลต่อไป ซึ่งการแสดงผลสถานะของผู้กรอกรหัสมี 2 แบบ คือ

1. “ผ่าน” หน้าจอจะแสดงคำว่า “YOU MEMBER”
2. “ไม่ผ่าน” หน้าจอจะปรากฏคำว่า “YOU NO MEMBER”

การทดลองในโหมดบันทึกฐานข้อมูล

ลำดับขั้นการทดลอง

1. เปิดเครื่องตรวจสอบบุคคลเข้าหน่วยงาน
2. เลือกโหมดบันทึกฐานข้อมูลด้วยการกดคีย์ 9 สังเกตผลที่เกิดขึ้น
3. ทำการกรอกรหัสด้วยคีย์บอร์ด เมื่อกรอกรหัสครบแล้วให้ทำการกดคีย์ * (Enter) อีกครั้ง สังเกตผลที่เกิดขึ้น

ผลการทดลอง

จากการทดลอง เมื่อทำการเปิดเครื่องแล้วพบว่า

1. เมื่อทำการกดคีย์ 9 จอแสดงผลจะปรากฏอยู่ในโหมดคีย์บอร์ดดังนี้

Login:_
WR data to Ram

2. ทำการกรอกรหัสลงไปให้ครบ 8 ตัว

Login:41031501
WR datd to Ram

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3. ทำการกดคีย์ * (Enter) แล้วจอแสดงผลจะมี 2 แบบ คือ

3.1 เมื่อทำการบันทึกฐานข้อมูลได้ จะปรากฏดังนี้

OK
put data in ram

3.2 เมื่อไม่สามารถทำการบันทึกฐานข้อมูลได้เนื่องจากในฐานข้อมูลมีรหัสนี้อยู่แล้วจะปรากฏดังนี้

database
to repeat itself

ซึ่งแสดงว่าเครื่องตรวจสอบบุคคลเข้าหน่วยงานสามารถทำการบันทึกฐานข้อมูลได้
การทดลองในโหมดการดูการลงทะเบียนฐานข้อมูล

ลำดับขั้นการทดลอง

1. เปิดเครื่องตรวจสอบบุคคลเข้าหน่วยงาน
2. เลือกโหมดการดูการลงทะเบียนฐานข้อมูลด้วยการกดคีย์ 5 สังเกตผลที่เกิดขึ้น
3. ทำการกดปุ่ม 2 , 8 และ 5 สังเกตผลที่เกิดขึ้น

ผลการทดลอง

จากการทดลอง เมื่อทำการเปิดเครื่องแล้วพบว่า

1. ทำการกดปุ่ม 5 หน้าจอจะแสดงดังนี้

WAIT

2. รอสักครู่หน้าจอจะแสดงฐานข้อมูลที่มีอยู่

41031501
Exit list push 5

3. สามารถดูข้อมูลถัดไปได้โดยการกด 2 (UP) หรือ 8 (DOWN) หรือต้องการออกจากการทำงานกด 5

การทดลองในโหมดการดูทะเบียนการเข้า-ออก

ลำดับขั้นตอนการทดลอง

1. เปิดเครื่องตรวจสอบบุคคลเข้าสู่หน่วยงาน
2. เลือกโหมดดูทะเบียนการเข้าออกด้วยการกดคีย์ 4 สังเกตผลที่เกิดขึ้น
3. ทำการกดปุ่ม 2 , 8 และ 5 สังเกตผลที่เกิดขึ้น

ผลการทดลอง

1. ทำการกด 4 หน้าจอจะแสดง

WAIT
List time in-out

2. รอสักครู่หน้าจอจะแสดงทะเบียนการเข้า-ออกที่มีอยู่

รหัส	วัน	เดือน	ปี
40131501	15	05	00
1850	150500	1852	
เวลาเข้า	วัน	เดือน	ปี
			เวลาออก

สามารถข้อมูลถัดไปได้โดยการกด 2 (UP) หรือ 8 (DOWN) หรือต้องการออกจากการทำงานกด 5

การทดลองในโหมดการลบบันทึกฐานข้อมูลทั้งหมด

ลำดับขั้นตอนการทดลอง

1. เปิดเครื่องตรวจสอบบุคคลเข้าสู่หน่วยงาน
2. เลือกโหมดลบบันทึกฐานข้อมูลทั้งหมดด้วยการกดคีย์ 4 สังเกตผลที่เกิดขึ้น
3. ทำการลบบันทึกฐานข้อมูลทั้งหมด
4. ตรวจสอบฐานข้อมูลที่มีอยู่

ผลการทดลอง

1. ทำการกด 7 หน้าจอจะแสดง

Del all database
* yes / # no

2. กด * เพื่อลบฐานข้อมูลทั้งหมด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3. ทำการดูฐานข้อมูลหน้าจะแสดง

No data base

การทดลองในโหมดการลบบันทึกฐานข้อมูลรายบุคคล

ลำดับขั้นการทดลอง

1. เปิดเครื่องตรวจสอบบุคคลเข้าหน่วยงาน
2. ทำการบันทึกฐานข้อมูลลงไป
3. เลือกโหมดลบบันทึกฐานข้อมูลรายบุคคลด้วยการกดคีย์ 8 สังเกตผลที่เกิดขึ้น
4. กดรหัสที่ต้องการลบลงไป
5. ตรวจสอบฐานข้อมูลที่มีอยู่

ผลการทดลอง

1. บันทึกรหัส 41031501 ลงไป
2. ทำการกด 8 หน้าจะแสดง

Login: _
Do you want to del?

3. ทำการกดรหัส 41031501 แล้วตามด้วย *
4. ตรวจสอบฐานข้อมูลปรากฏว่าไม่พบรหัส 41031501

การทดลองในโหมดการลบบันทึกเวลาทั้งหมด

ลำดับขั้นการทดลอง

1. ทำการเปิดเครื่องตรวจสอบบุคคลเข้าสู่หน่วยงาน
2. เลือกโหมดลบบันทึกเวลาทั้งหมดด้วยการกดคีย์ 6 สังเกตผลที่เกิดขึ้น
3. ลบบันทึกเวลาทั้งหมด
4. ทำการตรวจสอบบันทึกเวลาในหน่วยความจำ

ผลการทดลอง

1. ทำการกด 6 หน้าจอจะแสดง

Del data record
* yes / # no

2. กด * เพื่อทำการลบบันทึกเวลา
3. ทำการตรวจสอบข้อมูลปรากฏว่าไม่พบการบันทึกเวลา

No data in RAM



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 5

บทสรุป ปัญหา แนวทางแก้ไข และการพัฒนา

5.1 สรุป

ปริญญานิพนธ์ฉบับนี้เสนอผลงานเกี่ยวกับเครื่องตรวจสอบบุคคลเข้าหน่วยงาน เพื่อตรวจสอบการเข้า-ออกของบุคคลในหน่วยงาน ว่ามีบุคคลใดได้เข้า-ออกในหน่วยงานบ้าง อีกทั้งยังเป็นการช่วยในการรักษาความปลอดภัยภายในหน่วยงานนั้นๆ ซึ่งการเข้า-ออกในหน่วยงาน บุคคลภายในหน่วยงานต้องมีข้อมูลหรือรหัสบันทึกเก็บไว้ในฐานข้อมูล โดยบุคคลในหน่วยงานต้องใช้บัตรพนักงานป้อนรหัสซึ่งเป็นรหัสบาร์โค้ด หรือป้อนรหัสด้วยการกดรหัสของตนบนคีย์บอร์ด เมื่อมีบุคคลทำการป้อนรหัสที่หน้าประตู อุปกรณ์แสดงผลจะแสดงรหัสของบุคคลนั้น และจะแสดงผลว่ารหัสที่ทำการป้อนเข้ามานั้นมีข้อมูลในฐานข้อมูลหรือไม่ ถ้าในฐานข้อมูลของหน่วยงานไม่มีรหัสของบุคคลนั้นๆ อุปกรณ์แสดงผลจะแสดงให้ทราบว่าบุคคลนี้ไม่สามารถเข้า-ออกในหน่วยงานได้ ถ้าในฐานข้อมูลของหน่วยงานมีรหัสของบุคคลนี้อยู่ อุปกรณ์แสดงผลจะแสดงให้ทราบว่าบุคคลนี้สามารถเข้า-ออกในหน่วยงานได้ และจะทำการบันทึกข้อมูลการเข้า-ออกในหน่วยงานนี้ไว้เพื่อเป็นข้อมูลในการรักษาความปลอดภัยในหน่วยงานนั้นๆ ซึ่งเครื่องตรวจสอบบุคคลเข้าหน่วยงานนี้สามารถที่จะนำไปใช้งาน หรือนำเครื่องตรวจสอบบุคคลเข้าหน่วยงานไปประยุกต์ใช้งานในหน่วยงานต่างๆ ได้

5.2 ปัญหา และแนวทางการแก้ปัญหา

จากการทดลองเครื่องตรวจสอบบุคคลเข้าหน่วยงาน ได้ทำการทดลองอุปกรณ์ทีละส่วน แล้วนำทุกส่วนมาประกอบรวมกันเป็นวงจรสมบูรณ์ ในการทดลองแต่ละส่วนนั้นสามารถสรุปปัญหา แนวทางการแก้ปัญหาในการทดลองเป็นข้อๆ ได้ดังนี้

1. ในการทดลองส่วนของบอร์ดควบคุม ซึ่งเป็นบอร์ดของ บริษัท คิลารีเสิร์ช จำกัด รุ่น ANT - C51M คู่มือที่ทางบริษัทให้มานั้นตำแหน่งพอร์ตที่กำหนดไว้ไม่ตรงกับวงจรบนบอร์ด ทำให้ในการทดลองเกี่ยวกับเมทริกซ์-สวิตช์นั้นมีการผิดพลาดเกิดขึ้น
แก้ไขโดยตรวจสอบวงจร โดยการตรวจสอบหาตำแหน่งพอร์ตจากวงจรบนบอร์ดว่ากำหนดไว้ที่ตำแหน่งใด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2. ในการทดลองในส่วนของ การสื่อสารข้อมูล สายของพอร์ตอนุกรม RS-232 เกิดการเสียขึ้นระหว่างการทดลองทำให้เกิดการล่าช้าในการทำงาน
แก้ไข โดยการจัดมาใหม่และซื้อสำรองไว้ซึ่งอาจมีการเสียอีก
3. ในการทดลองส่วนของเครื่องอ่านบาร์โค้ดเครื่องอ่านบาร์โค้ดนั้นอ่านได้แต่บัตรนักศึกษา รุ่นเก่า แต่ไม่สามารถอ่านบัตรนักศึกษา รุ่นใหม่ได้
แก้ไข โดยได้ทำการปรับค่าตัวต้านทานแบบเปลี่ยนค่าได้ของเครื่องอ่านบาร์โค้ด
4. ในการทดลองส่วนของ การอ่านข้อมูลจากบัตรบาร์โค้ด ในการรูดบัตรบาร์โค้ดนั้น ถ้า รูดเร็วเกินไป หรือช้าเกินไป จะไม่สามารถอ่านข้อมูลได้
แก้ไข โดยรูดบัตรบาร์โค้ด ในความเร็วที่สม่ำเสมอไม่เร็วหรือช้าเกินไป
5. ในการทดลองส่วนของบอร์ด ANT-C51M นั้น ไม่สามารถใช้เครื่องอัดโปรแกรมรุ่น SPI-LOAD ได้
แก้ไข โดยจัดซื้อเครื่องอัด โปรแกรมรุ่น PRO-100 มาใช้งานแทน
6. ในการทดลองรวม โปรแกรมย่อยแต่ละ โปรแกรมรวมกันนั้น เมื่อทำการทดลองจ่ายไฟ ให้กับเครื่องตรวจสอบบุคคลเข้าหน่วยงานแล้ว พบว่าไม่สามารถทำการกดคีย์บอร์ดได้ การแสดงผลไม่สามารถแสดงผลได้
แก้ไข โดยตรวจสอบตัวโปรแกรมใหม่ และเปลี่ยนตัวแปรที่ใช้ในแต่ละส่วนไม่ให้ซ้ำกัน
7. ในส่วนของวงจรการบันทึกข้อมูลและตรวจสอบข้อมูล ไม่สามารถทำการบันทึกและ ตรวจสอบข้อมูลได้
แก้ไข โดย ตรวจสอบวงจรและทำการต่อวงจรตรงส่วนของการดีโคเดอร์ ใหม่
8. ในส่วนของไฟสำรองของวงจรบันทึกข้อมูลและตรวจสอบข้อมูล ไม่สามารถทำการจ่ายไฟ ให้กับหน่วยความจำได้ หน่วยความจำไม่สามารถเก็บข้อมูลที่ทำการบันทึกลงไปได้
แก้ไข โดยตรวจสอบวงจรและทำการปรึกษากับอาจารย์ที่ปรึกษา แล้วทำการแก้ไขวงจรไฟสำรองใหม่

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5.3 แนวทางการแก้ไข และการพัฒนา

เครื่องตรวจสอบบุคคลเข้าหน่วยงาน สามารถทำงานได้ตามขีดความสามารถในวัตถุประสงค์ที่กำหนดไว้ แต่โครงการนี้ยังสามารถเพิ่มขีดความสามารถในการทำงานได้อีก คือ

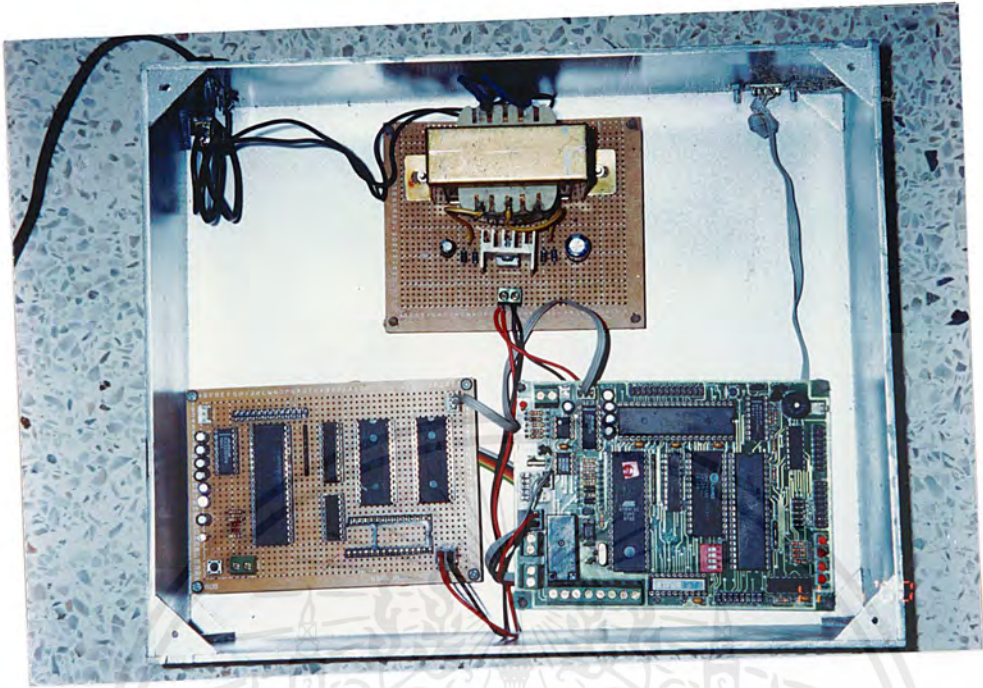
1. ในส่วนของโปรแกรมควบคุมระบบ สามารถที่จะเพิ่มรหัสผู้ที่ใช้เครื่องคอมพิวเตอร์ได้
2. ส่วนของการบันทึกข้อมูลการเข้า – ออกในหน่วยงาน สามารถที่จะสั่งพิมพ์ข้อมูลออกมาดูได้
3. ในส่วนของการบันทึกข้อมูลและฐานข้อมูลนั้น ระบบยังสามารถที่จะทำการประยุกต์ให้สามารถบันทึกข้อมูล และฐานข้อมูลให้มากกว่านี้ได้โดยการเพิ่มหน่วยความจำอีก



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ ก.1 แผงวงจรรวมเครื่องตรวจสอบบุคคลเข้าหน่วยงาน



รูปที่ ก.2 รูปสำเร็จของเครื่องตรวจสอบบุคคลเข้าหน่วยงาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โปรแกรมการทำงาน

```

CONA      EQU    0f800H          ;ORG 0000H
CONB      EQU    0f801H
CONC      EQU    0f802H          DS 8
CONP      EQU    0f803H          ;USER AREA (REGIER BANK-0)
                                SYSSTK:    DS 24

LCDWRC    EQU    0FA00H          ;SYEM ACK
LCDRDC    EQU    0FA01H          SYSFAG:    DS 1
LCDWRD    EQU    0FA02H          ;**** BIT ADDRESSABLE ****
LCDRDD    EQU    0FA03H          LCDBUF:    DS 32
                                ;LCD BUFFER

USRA      EQU    0FC00H          TIMEBUF:    DS 5
USRB      EQU    0FC01H          ;DD MM YY HH MM
USRC      EQU    0FC02H          TIMEOUT:    DS 12
USRP      EQU    0FC03H          SAVE_LOGIN: DS 8
                                SAVE_TIMEIN: DS 10
                                SAVE_TIMEOUT: DS 10

BARBIT    EQU    P1.1           ;SAVE_TIMEOUT: DS 10
;DATA FROM BARCODE READER      BAROUT:     DS 1
RTCDAT    EQU    P1.4           ;BAR CODE DIGIT BUFFER (LENGTH+1)
RTCCLK    EQU    P1.5           BARREV:     DS 1
RTCRST    EQU    P1.6           ;SAME AS BAROUT
BARBUF    EQU    8000H          KEYBUF:     DS 1
;RAW-DATA BUFFER (LENGTH+2)*10+1
                                ENDINT:

BYTE      ;***** MAIN *****
BRAT96    EQU    0FDH          MAIN:       ORG 0000H
;9600      RES:                MOV R2,#80H
SCANPF    EQU    00H          ;SCAN      ;POWER UP DELAY
- PRESS FLAG
RES1:     MOV R3,#0
SCANOL    EQU    01H          ;SCAN      DJNZ R3,$
- ONE LOOP FLAG              DJNZ R2,RES1
                                MOV SP,#SYSSTK

;*****INTERNAL RAM*****

```

```

INIT:      MOV  A,#89H          RESTAB1:   DB  "<<-personal->> "
;SET CONTROL (CONSOLE PORT)          DB  "check in system!"

      MOV  DPTR,#CONP
      MOVX @DPTR,A          FIND_KEY:    LCALL SCAN_KEY
      CLR  RTCCLK          CJNE  A,#0',MMM
      CLR  RTCRST          LCALL UBEEP

INIT_SER:  MOV  SCON,#52H      start_bar:  MOV  A,#80H
;TIMER1 MODE2          LCALL LCDWI
      MOV  TMOD,#20H          MOV  DPTR,#BF
;SERIAL 8 BIT UART MODE          LCALL LCDLDX
      MOV  TH1,#BRAT96          LCALL LCDOUT
;9600 BPS          MOV  R2,#02
      MOV  TL1,#BRAT96          CALL  DTSEC
      SETB TR1          SJMP  TT

;TIMER1 ON          BF:      DB  "Login : "
      MOV  R0,#08H          DB  "Mode Of Barcode!"
;CLEAR INT-RAM (ALL)          MMM:    LJMP  LOOP
      MOV  R2,#ENDINT-          *****
08H          TT:
RES4:      MOV  @R0,#0          MOV  A,#87H
      INC  R0          LCALL LCDWI
      DJNZ R2,RES4          L555:  CALL  BARIN
      CALL LCDSET          CALL  BARCV
      JC   MAINU          JC   MAINU
MAIN1:     MOV  A,#80H          MOV  A,#05H      ;send
      LCALL LCDWI          head 05 for check
MM1:      MOV          CALL  SBYTE
DPTR,#RESTAB1          MOV  R0,#BAROUT
      CALL  LCDLDX          ;SEND DATA TO SERIAL
      CALL  LCDOUT          MAIN2:   MOV  A,@R0
      MOV  R2,#2          CJNE  A,#0DH,MAIN4
      LCALL DTSEC
      LJMP  FIND_KEY          MACH:   LCALL UBEEP

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

CALL NEXT_11                                CALL SEND_TIME_SER
LCALL DELBC                                MEMBER:  LCALL RBYTE
CLR RI                                       NEXT_22: CJNE  A,#07H,YOU_OUT
ljmp main1                                  ;WAIT 07H for login
MOV DPTR,#YOU_IN
MAINU:  SJMP MAINE                          CALL LCDLDX
;*****                                  CALL LCDOUT
MAIN4:  LCALL SBYTE                        MOV R2,#10H
LCALL LCDWD                                CALL DTSEC
INC R0                                       LJMP OUTPUT11
LJMP MAIN2                                YOU_IN:  DB " welcome "
MAINE:  MOV R2,#80H                        DB "to ind.education"
;READ ERROR                                YOU_OUT: CJNE  A,#08H,MEMBER
MOV R3,#0                                    ;wait 08h for logout
LJMP L555                                    MOV DPTR,#SHOW_OUT
NEXT_11: LCALL RBYTE                       LCALL LCDLDX
CJNE                                        LCALL LCDOUT
A,#06H,NEXT_33_1 ; WAIT                    MOV R2,#02
06H IF YOU ARE MEMBER                      LCALL DTSEC
MOV                                           LJMP OUTPUT11
DPTR,#PUTDATA11                            SHOW_OUT: DB " logout "
LCALL LCDLDX                                DB " Good Bye "
LCALL LCDOUT
MOV R2,#10H                                NEXT_33: CJNE  A,#09H,NEXT_111
LCALL DTSEC                                ;WAIT 09H YOU NO MEMBER
SJMP                                        MOV DPTR,#ERROR11
DATE_TIME_BAR                              LCALL LCDLDX
PUTDATA11: DB " you member "                LCALL LCDOUT
DB " please wait "                          MOV R2,#10H
NEXT_33_1: LJMP NEXT_33                    LCALL DTSEC
;*****                                    SJMP OUTPUT11
DATE_TIME_BAR:                              ERROR11: DB " You no Member! "
CALL                                         DB "** No Database **"
SEND_TIME_LCD                              OUTPUT11: CALL DELAY

```

```

                RET
NEXT_111:      LJMP NEXT_11
,*****
ST54:         CALL TIMERD
                MOV A,TIMEBUF
                CALL HTOA
                MOV TIMEOUT,R2
                MOV
TIMEOUT+1H,R3
                MOV A,TIMEBUF+1
                CALL HTOA
                MOV
TIMEOUT+2H,R2
                MOV
TIMEOUT+3H,R3
                MOV A,TIMEBUF+2
                CALL HTOA
                MOV
TIMEOUT+4H,R2
                MOV
TIMEOUT+5H,R3
                MOV A,TIMEBUF+3
                CALL HTOA
                MOV
TIMEOUT+6H,R2
                MOV
TIMEOUT+7H,R3
                MOV A,TIMEBUF+4
                CALL HTOA
                MOV
TIMEOUT+8H,R2
                MOV TIMEOUT+9H,R3
                MOV A,TIMEBUF+5
                CALL HTOA
                MOV TIMEOUT+0AH,R2
                MOV TIMEOUT+0BH,R3
                RET
,*****
RTCRCR:      CLR RTCCLK
                CALL RTCDL
                SETB RTCRST
                CALL RTCDL
                MOV A,R2
                CALL RTCWRC
                MOV R4,#8
                CLR A
RTCRCR1:     CLR RTCCLK
                CALL RTCDL
                MOV C,RTCDA
                RRC A
                SETB RTCCLK
                CALL RTCDL
                DJNZ R4,RTCRCR1
                MOV R3,A
                CLR RTCRST
                CALL RTCDL
                RET
,*****
RTCWR:      CLR RTCCLK
                CALL RTCDL
                SETB RTCRST
                CALL RTCDL
                MOV A,R2
                CALL RTCWRC
                MOV A,R3

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

CALL RTCWRC                                MOV R2,#83H
CLR RTCRST                                  CALL RTCRD
CALL RTCDL                                  MOV TIMEBUF+4,R3
RET                                          MOV R2,#81H
RTCWRC: MOV R4,#8                            CALL RTCRD
RTCWRC1: RRC A                               MOV TIMEBUF+5,R3
MOV RTCDAT,C                               RET
SETB RTCCLK                                ;*****
CALL RTCDL                                TIMEWR: MOV R2,#8EH
CLR RTCCLK                                  MOV R3,#0
CALL RTCDL                                  CALL RTCWR
DJNZ R4,RTCWRC1                             MOV R2,#86H
RET                                          MOV R3,TIMEBUF
RTCDDL: push 5                               CALL RTCWR
MOV R5,#4                                    MOV R2,#88H
DJNZ R5,$                                    MOV R3,TIMEBUF+1
pop 5                                        CALL RTCWR
RET                                          MOV R2,#8CH
;*****                                    MOV R3,TIMEBUF+2
TIMERD: MOV R2,#87H                          CALL RTCWR
CALL RTCRD                                  MOV R2,#8AH
MOV TIMEBUF,R3                              MOV R3,TIMEBUF+6
MOV R2,#89H                                CALL RTCWR
CALL RTCRD
MOV TIMEBUF+1,R3
MOV R2,#8DH                                MOV R2,#84H
CALL RTCRD                                  MOV R3,TIMEBUF+3
MOV TIMEBUF+2,R3                            CALL RTCWR
MOV R2,#8BH                                MOV R2,#82H
CALL RTCRD                                  MOV R3,TIMEBUF+4
MOV TIMEBUF+6,R3                            CALL RTCWR
MOV R2,#85H                                MOV R2,#80H
CALL RTCRD                                  MOV R3,TIMEBUF+5
MOV TIMEBUF+3,R3                            CALL RTCWR

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้拿去ใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

MOV R2,#8EH          DELBC: PUSH 6
MOV R3,#80H          PUSH 7
CALL RTCWR           MOV R6,#45H
RET                  DELBC1:MOV R7,#7FH
;*****
HTOA: PUSH ACC       DJNZ R7,$
SWAP A              DJNZ R6,DELBC1
CALL HTOAS          POP 7
MOV R2,A            POP 6
POP ACC             RET
CALL HTOAS          ;***** UBEEP SUB *****
MOV R3,A            UBEEP: PUSH 2
RET                  PUSH 3
HTOAS: ANL A,#0FH   PUSH 4
CJNE A,#0AH,$+3    PUSH 5
JNC HTOAS1          MOV R2,#15H ;USER BEEP
ORL A,#30H          MOV R3,#70H
RET                  CALL SOUND
HTOAS1: SUBB A,#9   POP 5
ORL A,#40H          POP 4
RET                  POP 3
;***** SBYTE SUB *****
SBYTE: JNB TI,$     POP 2
CLR TI              RET
MOV SBUF,A
RET
;***** DELAY *****
RBYTE: JNB RI,$     DTSEC: MOV R3,#0
CLR RI              DTSEC1: MOV R4,#0
MOV A,SBUF          DJNZ R4,$
RET                  NOP
NOP
;***** DELAY SUB *****
DUNZ R3,DTSEC1

```

```

DJNZ R2,DTSEC                                JNZ SOUNDX1
RET                                            RET

;***** SOUND SUB *****
; SOUND GENERATOR
; IN = R2 FREQUENCY
; = R3 LENGTH
; REG = A,R2,R3,R4,R5,DPTR

SOUNDY: DJNZ R4,SOUNDY1
;LENGTH COUNT DOWN
MOV R4,#80H
DJNZ R3,SOUNDY1
MOV R5,#1
SOUNDY1: RET

SOUND: MOV R5,#0 ;END
FLAG ;***** BARCV SUB *****
MOV R4,#80H ; CONVERT RAW-DATA TO ASCII
;DELAY CONANT ; IN = BARBUF
SOUND1: CALL SOUNDS ; OUT = BAROUT,CY
CJNE R5,#1,SOUND1 ; REG = A,B,R0,R2,R3,DPTR
RET

SOUNDS: MOV DPTR,#CONB BARCV: MOV DPTR,#BARBUF
;OUT 1 (PORTA BIT7) MOVX A,@DPTR
MOVX A,@DPTR MOV B,#2
SETB ACC.3 MUL AB
MOVX @DPTR,A ;STD. COMPARE
CALL SOUNDX MOV DPTR,#BARBUF+1 ;CHECK
MOV DPTR,#CONB ;OUT 0 FORWARD OR REVERSE
MOVX A,@DPTR MOVX A,@DPTR
CLR ACC.3 CJNE A,B,$+3
MOVX @DPTR,A JC BARRV
CALL SOUNDX MOV DPTR,#BARBUF ;****
RET FORWARD ****
MOV R0,#BAROUT

SOUNDX: MOV A,R2 BARCV1: MOV R2,#9 ;DIGIT LOOP
;FREQUENCY DELAY MOV R3,#0
SOUNDX1: CALL SOUNDY BARCV2: MOVX A,@DPTR
DEC A CJNE A,#0FFH,BARCV3

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

SJMP BARCVE	SJMP BARRVE
BARCV3: CJNE A,B,\$+3	BARRV3: CJNE A,B,\$+3
CPL C	CPL C
MOV A,R3	MOV A,R3
RLC A	RRC A
MOV R3,A	MOV R3,A
INC DPTR	INC DPTR
DJNZ R2,BARCV2	DJNZ R2,BARRV2
INC DPTR ;SKIP 0	INC DPTR ;SKIP 0 BETWEEN
BETWEEN DIGIT	DIGIT
CALL BARTB	MOV A,R3
JC BARCVE	RLC A
	MOV R3,A
CJNE A,#'',BARCV4	CALL BARTB
CJNE R0,#BAROUT,BARCV35	JC BARRVE
SJMP BARCV1 ;IS * START	
BARCV35: MOV @R0,#0DH	CJNE A,#'',BARRV4
CLR C	CJNE R0,#BARREV,BARRV35
RET ;EXIT OK	SJMP BARRV1 ;IS * START
BARCV4: MOV @R0,A	BARRV35: MOV R1,#BAROUT ;LOAD
INC R0	BARREV TO BAROUT
SJMP BARCV1	DEC R0
BARCVE: SETB C	BARRV37: CJNE R0,#BARREV-1,BARRV38
RET ;EXIT ERROR	MOV @R1,#0DH
	CLR C
BARRV: MOV DPTR,#BARBUF ;****	RET ;EXIT OK
REVERSE ****	
MOV R0,#BARREV	BARRV38: MOV A,@R0
	MOV @R1,A
BARRV1: MOV R2,#9 ;DIGIT	DEC R0
LOOP	INC R1
MOV R3,#0	SJMP BARRV37
BARRV2: MOVX A,@DPTR	
CJNE A,#0FFH,BARRV3	BARRV4: MOV @R0,A

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

INC R0
SJMP BARRV1
BARRVE: SETB C
RET ;EXIT ERROR
;***** BARTB SUB *****
; BARCODE LOOKUP TABLE
; IN = R3,CY (CODE39)
; OUT = A,CY (ASCII)
; REG = A
BARTB: PUSH DPH
PUSH DPL
MOV DPTR,#BARTBX
JC BARTB1
MOV DPTR,#BARTBY
BARTB1: CLR A ;TABLE
MOVC A,@A+DPTR
CJNE A,#0FFH,BARTB2
SJMP BARTBE
BARTB2: CLR C ;COMPARE
SUBB A,R3
JZ BARTB5
INC DPTR
INC DPTR
SJMP BARTB1
BARTB5: INC DPTR ;COMPARE OK
CLR A
MOVC A,@A+DPTR
CLR C
POP DPL
POP DPH
RET ;EXIT OK
BARTBX: DB 00100001B,'1' ;BIT 9 = 1
TABLE
DB 01100000B,'3'
DB 00110000B,'5'
DB 00100100B,'8'
DB 00001001B,'A'
DB 01001000B,'C'
DB 00011000B,'E'
DB 00001100B,'H'
DB 00000011B,'K'
DB 01000010B,'M'
DB 00010010B,'O'
DB 00000110B,'R'
DB 10000001B,'U'
DB 11000000B,'W'
DB 10010000B,'Y'
DB 10000100B,'.'
DB 0FFH
BARTBY: DB 00110100B,'0' ;BIT 9 = 0
TABLE
DB 01100001B,'2'
DB 00110001B,'4'
DB 01110000B,'6'
DB 00100101B,'7'
BARTBE: SETB C ;EXIT
ERROR
POP DPL
POP DPH
RET

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

DB 01100100B,'9'
DB 01001001B,'B'
DB 00011001B,'D'
DB 01011000B,'F'
DB 00001101B,'G'
DB 01001100B,'I'
DB 00011100B,'J'
DB 01000011B,'L'
DB 00010011B,'N'
DB 01010010B,'P'
DB 00000111B,'Q'
DB 01000110B,'S'
DB 00010110B,'T'
DB 11000001B,'V'
DB 10010001B,'X'
DB 11010000B,'Z'
DB 10000101B,'-'
DB 11000100B,'.'
DB 10010100B,'*'
DB 10101000B,'$'
DB 10100010B,'/'
DB 10001010B,'+'
DB 00101010B,'%'
```

```

BARIN1: CLR A ;BACK BAR
BARIN2: CALL BARDL
INC A
JZ BARIN5
JB BARBIT,BARIN2
MOVX @DPTR,A
INC DPTR

CLR A ;WHITE BAR
BARIN3: CALL BARDL
INC A
JZ BARIN5
JNB BARBIT,BARIN3
MOVX @DPTR,A
INC DPTR
SJMP BARIN1

BARIN5: MOV A,#0FFH
MOVX @DPTR,A
RET

BARDL: MOV R2,#01H ;DELAY SUB.
DJNZ R2,$
RET

;***** BARIN SUB *****
; READ BAR CODE TO BARBUF
; OUT = BATBUF
; REG = A,R2,DPTR
BARIN: call st54
MOV DPTR,#BARBUF
JB BARBIT,$ ;WAIT FOR
FIRST PULSE
JNB BARBIT,$
```

```

;*****LCD INITIALIZE*****
LCDSET:
MOV R2,#02H
LCALL DTSEC
MOV A,#00111000B
LCALL LCDWI
MOV A,#00001110B
LCALL LCDWI
MOV A,#01H
```

```

LCALL LCDWI                                POP DPL
MOV R2,#02H                                POP DPH
LCALL DTSEC                                POP ACC
RET                                          RET

;***** LCDWI SUB *****                ;***** LCDLDX SUB *****
; LCD WRITE INRUCTION (RS=0)              ; LOAD PMEM TO LCDBUF (ALL.LINE)
; IN = A                                  ; IN = DPTR
; REG = A                                  ; REG = A,R0,R2,DPTR
LCDWI:  PUSH ACC                            LCDLDX: PUSH 0
        PUSH DPH                            PUSH 2
        PUSH DPL                            MOV R2,#32      ;LOAD ALL
        MOV DPTR,#LCDWRC                    MOV R0,#LCDBUF
        MOVX @DPTR,A                        LCDLDX1: CLR A
        MOV DPTR,#LCDRDC                    MOV C A,@A+DPTR
LCDWI1: MOVX A,@DPTR                        MOV @R0,A
;WAIT FOR BF=0                             INC DPTR
        JB ACC.7,LCDWI1                     INC R0
        POP DPL                             DJNZ R2,LCDLDX1
        POP DPH                             POP 2
        POP ACC                             POP 0
        RET                                  RET

;***** LCDWD SUB *****                LCDLD1:  PUSH 0
; LCD WRITE DATA (RS=1)                  PUSH 2
; IN = A                                  MOV R2,#16      ;LOAD LINE-1
; REG = A                                  MOV R0,#LCDBUF
LCDWD:  PUSH ACC                            SJMP LCDLDX1
        PUSH DPH                            LCDLD2:  PUSH 0
        PUSH DPL                             PUSH 2
        MOV DPTR,#LCDWRD                    MOV R2,#16      ;LOAD LINE-2
        MOVX @DPTR,A                        MOV R0,#LCDBUF+16
        MOV DPTR,#LCDRDC                    SJMP LCDLDX1
LCDWD1: MOVX A,@DPTR                        ;***** LCDOUT SUB *****
;WAIT FOR BF=0                             ; LOAD LCDBUF TO LCD-MODULE (DMC202)
        JB ACC.7,LCDWD1                     ; IN = LCDBUF

```

```

; REG = A,R0,R2
LCDOUT: PUSH 0
        PUSH 2
        PUSH ACC
        MOV R0,#LCDBUF ;LCD
BUFFER
        MOV A,#80H ;SET
ADDRESS LINE 1
        CALL LCDOUTS
        MOV A,#0C0H ;SET
ADDRESS LINE 2
        CALL LCDOUTS
        POP ACC
        POP 2
        POP 0
        RET
LCDOUTS: CALL LCDWI
;LOAD ONE LINE
        MOV R2,#16 ;16 CHAR.
LCDOUTS1: MOV A,@R0
        CALL LCDWD ;WRITE
DATA
        INC R0
        DJNZ R2,LCDOUTS1
        RET
LIST: ;LCALL DELAY
        CJNE A,#'5',PUT_DATA
        LCALL UBEEP
        LCALL LIST1
        LJMP MAIN1
PUT_DATA: ;LCALL DELAY
        CJNE A,#'9',DEL_ONE
        LCALL UBEEP
        LCALL PUT_DATA1
        LJMP MAIN1
DEL_ONE: ;LCALL DELAY
        CJNE A,#'8',DEL_ALL
        LCALL UBEEP
        LCALL DEL_ONE1
        LJMP MAIN1
DEL_ALL: ;LCALL DELAY
        CJNE A,#'7',DEL_LIST
        LCALL UBEEP
        LCALL DEL_ALL1
        LJMP MAIN1
DEL_LIST: CJNE A,#'6',LIST_TIMEOUT
        LCALL UBEEP
        LCALL DEL_LIST1
        LJMP MAIN1
LIST_TIMEOUT:CJNE A,#'4',LOOPBACK
        LCALL UBEEP
        LCALL LIST_TIME_INOUT
        LJMP MAIN1
;*****
LOOP: ;LCALL SCAN_KEY
        ;LCALL DELAY
        CJNE A,#'',LIST
        LCALL UBEEP
        LCALL RST
        LJMP MAIN1
LOOPBACK: LJMP FIND_KEY
;*****
RST: ;MOV R5,#08H ;<-----
        MOV A,#80H
        LCALL LCDWI
        MOV DPTR,#MAINTY

```

```

LCALL LCDLDX                lcall ubEEP
LCALL LCDOUT                MOV @R1,#0DH
MOV R2,#02H
CALL DTSEC                  MOV A,#05H ;HEAD TO
LJMP CHECK                  CHECK
MAINTY: DB "login : "      LCALL SBYTE
DB "mode of keyboard"     lcall delay
M1: MOV R1,#KEYBUF
M2: MOV A,@R1
CHECK: MOV A,#87H          CJNE A,#0DH,M4
LCALL LCDWI                LJMP NEXT_1
mov r5,#09h
MOV R1,#KEYBUF            M4: LCALL SBYTE
CHECK1: LCALL SCAN_KEY     INC R1
MOV @R1,A                 SJMP M2
djnz r5,wd1               WD: LCALL LCDWD
jmp oo1                   LCALL DELAY
wd1: CJNE A,#*,menu        INC R1
mov dptr,#lee_8           SJMP CHECK1
lcall lcdldx
lcall lcdout
mov r2,#10h
lcall dtsec
jmp rst
lee_8: db " you login "    MOV DPTR,#CHECKDATA
db "less 8 character"     LCALL LCDLDX
menu: cjne a,#*,wd        LCALL LCDOUT
ret                        MOV R2,#10H
;jmp main1                LCALL DTSEC
;CALL UBEEP               SJMP DATE_TIME
oo1: lcall scan_key       CHECKDATA: DB " you member "
cjne a,#*,oo              DB " please wait "
ret
;jmp main1
oo: cjne a,#*,oo1         ;*****
NEXT_CHECK_1:LJMP NEXT_CHECK1
;*****
DATE_TIME: ;MOV A,#0C0H

```

```

;CALL LCDWI                LCALL DTSEC
CALL SEND_TIME_LCD        SJMP OUTCHECK
CALL SEND_TIME_SER        CHECKERROR: DB " login error "
INOUT_KEY: LCALL RBYTE    DB "system error "
NEXT_CHECK: CJNE          OUTCHECK: CALL DELAY
A,#07H,CHECKOUT_KEY ;wait 07h for
login                      RET
MOV DPTR,#DATA_IN_KEY    NEXT_1_1: LJMP NEXT_1
CALL LCDLDX              .*****
CALL LCDOUT              SEND_TIME_LCD:
MOV R2,#05H              MOV A,#0C0H
CALL DTSEC               CALL LCDWI
LJMP OUTCHECK            MOV LCDBUF+16,TIMEOUT
DATA_IN_KEY:DB " welcome " MOV LCDBUF+17,TIMEOUT+1H
DB "To Ind.Education"    MOV LCDBUF+18,# '/'
CHECKOUT_KEY:CJNE        MOV LCDBUF+19,TIMEOUT+2H
A,#08H,INOUT_KEY ;wait 08h for MOV LCDBUF+20,TIMEOUT+3H
logout                    MOV LCDBUF+21,# '/'
MOV DPTR,#DATA_OUT_KEY  MOV LCDBUF+22,TIMEOUT+4H
LCALL LCDLDX            MOV LCDBUF+23,TIMEOUT+5H
LCALL LCDOUT           MOV LCDBUF+24,# ' '
MOV R2,#05H            MOV LCDBUF+25,# ' '
LCALL DTSEC           MOV LCDBUF+26,# ' '
LJMP OUTCHECK         MOV LCDBUF+27,TIMEOUT+6H
DATA_OUT_KEY:          MOV LCDBUF+28,TIMEOUT+7H
DB " logout "         MOV LCDBUF+29,# ':'
DB " Good Bye "      MOV LCDBUF+30,TIMEOUT+8H
NEXT_CHECK1:CJNE      MOV LCDBUF+31,TIMEOUT+9H
A,#09H,NEXT_1_1 ;wait 09h for no
member                  CALL LCDOUT
MOV DPTR,#CHECKERROR  CALL DELAY
LCALL LCDLDX          MOV R2,#10H
LCALL LCDOUT          CALL DTSEC
MOV R2,#05H          RET
.*****

```

```

SEND_TIME_SER:                MOV    @R1,A
    PUSH  5                    djnz  r5,wp1
    MOV   R5,#10              jmp   op1
    MOV   A,#0AH              wp1:  CJNE  A,#**',menu_p
;HEAD SEND TIME 0AH          ;LCALL UBEEP
    CALL  SBYTE               mov   dptr,#put_less
    MOV   R1,#TIMEOUT        lcall lcdldx
SEND_TIME: MOV   A,@R1        lcall lcdout
    LCALL SBYTE               mov   r2,#10h
    DJNZ  R5,SEND_NEW_TIME    lcall dtsec
    LCALL UBEEP               jmp   put_data1
    POP   5                    put_less: db " database "
    RET                        db "less 8 character"
SEND_NEW_TIME: INC  R1
    SJMP  SEND_TIME          menu_p:  cjne  a,#**',wp
;***** PUT DATA *****    ret
PUT_DATA1: MOV   A,#80H      op1:   lcall scan_key ;<----
    CALL  LCDWI              cjne  a,#**'.op
    MOV                                ret
    DPTR,#TABLE_PUT_DATA    ;jmp   main1
    CALL  LCDLDX            op:    cjne  a,#**',op1
    CALL  LCDOUT            lcall ubeep
    MOV   R2,#05H          MOV   @R1,#0DH
    CALL  DTSEC            MOV   A,#01H ;HEAD TO
    SJMP  NEXT_PUT_DATA    PUT DATABASE
TABLE_PUT_DATA:            LCALL SBYTE
    DB "Login : "          lcall delay
    DB "Wr data to ram "  MP1:  MOV   R1,#KEYBUF
NEXT_PUT_DATA:            MP2:  MOV   A,@R1
    MOV   A,#87H          CJNE  A,#0DH,MP4 ;<----
    CALL  LCDWI            LJMP  NEXT_PUT_DATA2
    mov   r5,#09h        MP4:  LCALL SBYTE
    MOV   R1,#KEYBUF      INC   R1
CHECKPUT: LCALL SCAN_KEY    SJMP  MP2

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านธุรกิจ
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

WP:    CALL LCDWD                CJNE A,#04H,NEXT_PUT_DATA2
      CALL DELAY                ;WAIT 04H
      INC R1                    MOV DPTR,#TABLE_ERROR_PUT
      SJMP CHECKPUT            CALL LCDLDX
NEXT_PUT_DATA2:                CALL LCDOUT
      CALL RBYTE                MOV R2,#10H
      CJNE                      CALL DTSEC
A,#02H,NEXT_PUT_DATA3        ;WAIT    LJMP put_data1
02H                                TABLE_ERROR_PUT:
      MOV DPTR,#TABLE_OK_PUT    DB " error "
      CALL LCDLDX                DB "non write to ram"
      CALL LCDOUT
      MOV R2,#10H                ;***** LIST DATA *****
      CALL DTSEC                LIST1: MOV A,#80H
      LJMP put_data1            CALL LCDWI
TABLE_OK_PUT:                  MOV DPTR,#DATAL
      DB " OK "                  LCALL LCDLDX
      DB "put data to ram "     LCALL LCDOUT
      MOV R2,#02
NEXT_PUT_DATA3:                LCALL DTSEC
      CJNE                      SJMP N_1
A,#03H,NEXT_PUT_DATA4        ;WAIT    DATAL: DB " WAIT "
03H                                DB " "
      MOV DPTR,#TABLE_SUM_PUT    N_1: MOV A,#29H                ;HEAD OF
      CALL LCDLDX                LIST DATABASE
      CALL LCDOUT                LCALL SBYTE
      MOV R2,#10H                GG: MOV R1,#KEYBUF
      CALL DTSEC                MOV R7,#08H
      LJMP put_data1            BACK: LCALL RBYTE
TABLE_SUM_PUT:                  CJNE A,#2DH,SHOW                ;IF NO
      DB " database "            DATA BASE (2DH)
      DB "to repeat itself"     MOV A,#80H
NEXT_PUT_DATA4:                LCALL LCDWI
      MOV DPTR,#SHOW_NO_DATA

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปเผยแพร่โดยไม่ขออนุญาต
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

LCALL LCDLDX                                CJNE A,#'2',DOWN
LCALL LCDOUT                                LCALL UBEEP
MOV R2,#05H                                MOV A,#2AH ;UP
LCALL DTSEC                                LCALL SBYTE
LJMP MAIN1                                LJMP GG

SHOW_NO_DATA:                               DOWN: CJNE A,#'8',CANCLE
DB " NONE "                                LCALL UBEEP
DB " DATABASE "                            MOV A,#2BH ;DOWN
SHOW: MOV @R1,A                             LCALL SBYTE
INC R1                                     LJMP GG
DJNZ R7,BACK                               CANCLE: CJNE A,#'5',WAIT_PUSH
MOV A,#80H                                LCALL UBEEP
LCALL LCDWI                                MOV A,#2CH ;CANCLE
MOV LCDBUF+1,#' '                          LCALL SBYTE
MOV LCDBUF+2,#' '                          CALL DELAY
MOV LCDBUF+4,KEYBUF                        RET
MOV LCDBUF+5,KEYBUF+1H                    ;*****
MOV LCDBUF+6,KEYBUF+2H                    DEL_ONE1: MOV A,#80H
MOV LCDBUF+7,KEYBUF+3H                    CALL LCDWI
MOV LCDBUF+8,KEYBUF+4H                    MOV DPTR,#TABLE_DEL_ONE
MOV LCDBUF+9,KEYBUF+5H                    CALL LCDLDX
MOV LCDBUF+10,KEYBUF+6H                   CALL LCDOUT
MOV LCDBUF+11,KEYBUF+7H                   MOV R2,#02H
MOV LCDBUF+15,#' '                        CALL DTSEC
LCALL LCDOUT                                SJMP NEXT_DEL_ONE
MOV A,#0C0H                                TABLE_DEL_ONE:
CALL LCDWI                                DB "login : "
MOV DPTR,#SHOW_LIST                       DB "Do you want del?"
CALL LCDLD2                                NEXT_DEL_ONE:
CALL LCDOUT                                MOV A,#87H
SJMP WAIT_PUSH                             CALL LCDWI
SHOW_LIST: DB "Exit list push 5"          mov r5,#09h
WAIT_PUSH:                                MOV R1,#KEYBUF
CALL SCAN_KEY                              CHECK_DEL_ONE:

```

```

CALL SCAN_KEY                WD_ONE: CALL LCDWD
MOV @R1,A                    CALL DELAY
djnz r5,wn1                 INC R1
jmp on1                     SJMP CHECK_DEL_ONE
wn1: CJNE A,#'',menu_n      ; <-
---
MOV DPTR,#LESS_THAN        CALL RBYTE
LCALL LCDLDX                CJNE A,#21H,DEL_ONE_NON
LCALL LCDOUT                ;WAIT 21H
MOV R2,#10H                MOV DPTR,#OK_DEL_ONE
LCALL DTSEC                CALL LCDLDX
ljmp del_one1              CALL LCDOUT
LESS_THAN: DB " login delete "  MOV R2,#10H
DB "less 8 character"        CALL DTSEC
menu_n:  cjne a,'#',wd_one    LJMP del_one1
ret                          OK_DEL_ONE: DB " OK "
on1:  lcall scan_key         DB " DELETE "
cjne a,'#',on              DEL_ONE_NON:CJNE
ret                          A,#22H,DEL_ONE_ERROR ;WAIT 22H
on:  cjne a,'#',on1         MOV DPTR,#NON_DEL_ONE
lcall ubeep                CALL LCDLDX
MOV @R1,#0DH              CALL LCDOUT
MOV A,#20H                ;HEAD    MOV R2,#10H
TO DELETE ONE             CALL DTSEC
LCALL SBYTE                LJMP del_one1
lcall delay                NON_DEL_ONE:DB " NONE "
MPD1: MOV R1,#KEYBUF        DB " DATABASE "
MPD2: MOV A,@R1             DEL_ONE_ERROR:
CJNE A,#0DH,MPD4          ; <-    CJNE A,#23H,NEXT_DEL_ONE1
---                          ;WAIT 23H
LJMP NEXT_DEL_ONE1        MOV DPTR,#ERROR_DEL_ONE
MPD4: LCALL SBYTE          CALL LCDLDX
INC R1                    CALL LCDOUT
SJMP MPD2                 MOV R2,#10H
                          CALL DTSEC

```

```

LJMP del_one1                                CALL DTSEC
ERROR_DEL_ONE:                                RET
DB " ERROR "                                  OK_DEL_ALL: DB " <<<<- OK ->>> "
DB " NONE DELETE. "                          DB "DEL DATABASE ALL"
,*****
OUTDEL_ONE: CALL DELAY                        DEL_LIST1: MOV A,#80H
RET                                            LCALL LCDWI
,*****                                       MOV DPTR,#DATA_DEL_LIST
DEL_ALL1: MOV A,#80H                          LCALL LCDLDX
CALL LCDWI                                    LCALL LCDOUT
MOV DPTR,#TABLE_DEL_ALL                      MOV R2,#02
CALL LCDLDX                                  LCALL DTSEC
CALL LCDOUT                                  SJMP NEXT_DEL_LIST
MOV R2,#02H                                  DATA_DEL_LIST:
CALL DTSEC                                   DB "Del recode all? "
SJMP NEXT_DEL_ALL                           DB " * Yes / # No "
TABLE_DEL_ALL:DB "Del database all"          NEXT_DEL_LIST:LCALL SCAN_KEY
DB " * yes / # no "                          cjne a,#'#',ch_star
NEXT_DEL_ALL:                                ret
CALL SCAN_KEY                               ch_star: CJNE A,#'*,NEXT_DEL_LIST
cjne a,#'#',c_star                          lcall ubeep
ret                                           MOV A,#27H ;--->HEAD
c_star: CJNE A,#'*,NEXT_DEL_ALL             OF DEL LIST
LCALL UBEEP                                  LCALL SBYTE
MOV A,#24H ;HEAD                            WAIT_28H: LCALL RBYTE
OF DEL_ALL                                  CJNE A,#28H,WAIT_28H ;--->
CALL SBYTE                                  WAIT 28H
REV_NEW: CALL RBYTE                          MOV A,#80H
CJNE A,#25H,REV_NEW ;-                      CALL LCDWI
->WAIT 25H                                   MOV DPTR,#TABLE_DEL_LIST
MOV DPTR,#OK_DEL_ALL                        LCALL LCDLDX
CALL LCDLDX                                  LCALL LCDOUT
CALL LCDOUT                                  MOV R2,#10H
MOV R2,#10H                                  LCALL DTSEC

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

RET
TABLE_DEL_LIST:          AGAIN_LOGIN1:LCALL RBYTE
    DB " <<-- OK -->> "    REV_LOGIN: MOV @R1,A
    DB " Del recode all "    DJNZ R5,AGAIN_LOGIN
;*****
;*****
LIST_TIME_INOUT:MOV A,#80H    LCALL UBEEP
    LCALL LCDWI            SJMP REV_TIMEIN
    MOV                    AGAIN_LOGIN:INC R1
                                JMP AGAIN_LOGIN1
DPTR,#DATA_DATE_TIME
    LCALL LCDLDX          REV_TIMEIN: MOV A,#2EH
    LCALL LCDOUT         LCALL SBYTE
    MOV R2,#02           MOV R5,#0AH
    LCALL DTSEC          MOV R1,#SAVE_TIMEIN
    SJMP N_DT           AGAIN_TIME1:LCALL RBYTE ;<-
DATA_DATE_TIME:DB "***** wait *****"
    DB "list recode all "    MOV @R1,A
                                DJNZ R5,AGAIN_TIMEIN
N_DT: MOV A,#0FFH        LCALL UBEEP
    LCALL SBYTE ;-->SEND    SJMP REV_TIMEOUT
HEAD
N_DT1: MOV R5,#08 ;-----    AGAIN_TIMEIN:
                                INC R1
                                JMP AGAIN_TIME1
---
    MOV R1,#SAVE_LOGIN
    LCALL RBYTE ;<--REV    REV_TIMEOUT:MOV A,#2EH
LOGIN
    LCALL SBYTE
    CJNE A,#0FEH,REV_LOGIN    MOV R5,#0AH
    MOV DPTR,#NO_DATA_DT     MOV R1,#SAVE_TIMEOUT
    LCALL LCDLDX            AGAIN_TIMEO:LCALL RBYTE
    LCALL LCDOUT           CJNE A,#2FH,SAVE_T ;-->
    MOV R2,#10H            WAIT 2FH
    LCALL DTSEC           MOV A,#' '
    RET                    DDTT: MOV @R1,A
NO_DATA_DT: DB " none "    DJNZ R5,DDT
    db " recode data "    LCALL UBEEP

```

```

LJMP SHOW_DATE_TIME
DDT: INC R1
S JMP DDTT
SAVE_T: MOV @R1,A
DJNZ R5,AGAIN_TIMEOUT
LCALL UBEEP
LJMP SHOW_DATE_TIME
AGAIN_TIMEOUT:INC R1
S JMP AGAIN_TIMEO
SHOW_DATE_TIME:
MOV A,#01H
LCALL LCDWI
SHOW_LOGIN: MOV A,#80H
LCALL LCDWI
MOV LCDBUF,SAVE_LOGIN
MOV
LCDBUF+1,SAVE_LOGIN+1H
MOV
LCDBUF+2,SAVE_LOGIN+2H
MOV
LCDBUF+3,SAVE_LOGIN+3H
MOV
LCDBUF+4,SAVE_LOGIN+4H
MOV
LCDBUF+5,SAVE_LOGIN+5H
MOV
LCDBUF+6,SAVE_LOGIN+6H
MOV
LCDBUF+7,SAVE_LOGIN+7H
MOV LCDBUF+8,#' '
MOV LCDBUF+9,SAVE_TIMEIN
MOV
LCDBUF+10,SAVE_TIMEIN+1H
MOV LCDBUF+11,SAVE_TIMEIN+2H
MOV LCDBUF+12,SAVE_TIMEIN+3H
MOV LCDBUF+13,SAVE_TIMEIN+4H
MOV LCDBUF+14,SAVE_TIMEIN+5H
MOV LCDBUF+15,#' '
MOV LCDBUF+16,SAVE_TIMEIN+6H
MOV LCDBUF+17,SAVE_TIMEIN+7H
MOV LCDBUF+18,SAVE_TIMEIN+8H
MOV LCDBUF+19,SAVE_TIMEIN+9H
MOV LCDBUF+20,#' '
MOV LCDBUF+21,SAVE_TIMEOUT
MOV
LCDBUF+22,SAVE_TIMEOUT+1H
MOV
LCDBUF+23,SAVE_TIMEOUT+2H
MOV
LCDBUF+24,SAVE_TIMEOUT+3H
MOV
LCDBUF+25,SAVE_TIMEOUT+4H
MOV
LCDBUF+26,SAVE_TIMEOUT+5H
MOV LCDBUF+27,#' '
MOV
LCDBUF+28,SAVE_TIMEOUT+6H
MOV
LCDBUF+29,SAVE_TIMEOUT+7H
MOV
LCDBUF+30,SAVE_TIMEOUT+8H
MOV
LCDBUF+31,SAVE_TIMEOUT+9H
LCALL LCDOUT
MOV R2,#02
CALL DTSEC

```

WAIT_DATAKEY:

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

CALL SCAN_KEY                                LJMP NEXT
CJNE A,#'2',DOWNLIST                          KEY4: CJNE A,#0DH,KEY7
LCALL UBEEP                                  MOV A,#'4'
MOV A,#2AH ;UP                               LJMP NEXT
LCALL SBYTE                                KEY7: CJNE A,#0BH,KEY10
LJMP N_DT1                                  MOV A,#'7'
DOWNLIST: CJNE A,#'8',CANCLELIST             LJMP NEXT
LCALL UBEEP                                KEY10: CJNE A,#07H,C1
MOV A,#2BH ;DOWN                            MOV A,#''
LCALL SBYTE                                LJMP NEXT
LJMP N_DT1
CANCLELIST: CJNE A,#'5',WAIT_DATAKEY        C1: CALL ST54
A,#'5',WAIT_DATAKEY                          INC R6
LCALL UBEEP                                MOV DPTR,#CONB
MOV A,#2CH ;CANCLE                          MOV A,R6
LCALL SBYTE                                MOVX @DPTR,A
CALL DELAY                                  MOV DPTR,#CONC
RET                                          MOVX A,@DPTR
ANL A,#0FH
*****
SCAN_KEY:                                    CJNE A,#0FH,KEY2
SCAN: CALL ST54                              SJMP C2
MOV R6,#00H                                KEY2: CJNE A,#0EH,KEY5
MOV DPTR,#CONB                              MOV A,#'2'
MOV A,R6                                    SJMP NEXT
MOVX @DPTR,A                                KEY5: CJNE A,#0DH,KEY8
MOV A,#'5'                                  MOV A,#'5'
SJMP NEXT
C0: MOV DPTR,#CONC                          KEY8: CJNE A,#0BH,KEY0
MOVX A,@DPTR                                MOV A,#'8'
ANL A,#0FH                                  SJMP NEXT
CJNE A,#0FH,KEY1                            KEY0: CJNE A,#07H,C2
SJMP C1                                      MOV A,#'0'
KEY1: CJNE A,#0EH,KEY4                      SJMP NEXT
MOV A,#'1'

```

```

C2:   CALL ST54                      DJNZ R7,DELAY2
      INC R6                          DJNZ R6,DELAY1
      MOV DPTR,#CONB                 POP 7
      MOV A,R6                       POP 7
      MOVX @DPTR,A                   RET
      MOV DPTR,#CONC                 END
      MOVX A,@DPTR                    ORG 0000H
      ANL A,#0FH                      DS 8
      CJNE A,#0FH,KEY3                CODEBUF: DS 8
      LJMP SCAN                       TIMEBUF: DS 0ah
KEY3: CJNE A,#0EH,KEY6                MEMBER: DS 1
      MOV A,#'3'                       C_MEM: DS 1
      SJMP NEXT                       MEMKEY: DS 1
KEY6: CJNE A,#0DH,KEY9                StepMem: DS 1
      MOV A,#'6'                       EndAddH: DS 1
      SJMP NEXT                       EndAddL: DS 1
KEY9: CJNE A,#0BH,KEY12               DISPTIMEH: DS 1
      MOV A,#'9'                       DISPTIMEL: DS 1
      SJMP NEXT                       ENDRAM:
KEY12: CJNE A,#07H,NO_PRESS
      mov a,##'                         FEQU EQU 00H
      sjmp next                         FDEL EQU 01H
NO_PRESS:
      LJMP SCAN_KEY                    Q_SAVE EQU 01H;'s'
                                          Q_DISP EQU 29H;'p'
NEXT:  RET                              Q_SENT_TIME EQU 0AH;'t'
                                          Q_USE_IN EQU 05H;'i'
;*****
DELAY: PUSH 6                          Q_REV_COM EQU 2EH;'c'
      PUSH 7                            Q_DEL_1USER EQU 20H;'g'
      MOV R6,#00H                       Q_DEL_AUSER EQU 24H;'a'
DELAY1: MOV R7,#00H                    Q_DEL_TIME EQU 27H;'r'
DELAY2: NOP                             Q_DISP_TIME EQU 0FFH;'w'
      NOP                               A_SAVE_COMP EQU 02H

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้拿去ใช้ประโยชน์ทางการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

A_SAVE_NOT EQU 03H ORG 0000H
A_DEL_COMP EQU 21H JMP MAIN
A_DEL_NOT EQU 22H
A_NOT_MEM EQU 09H MAIN:
A_DELALL_COMP EQU 25H mov sp,#50h
A_NOT_EROR EQU 0FFH MOV TMOD,#20H
A_TIME_EROR EQU 0FEH MOV SCON,#52H
A_TIME_IN EQU 07H MOV TH1,#0FDH
A_TIME_OUT EQU 08H SETB TR1
A_TIME_COMP EQU 06H
A_NO_SAVET EQU 0FEH ; mov member,#0h
A_NO_TIMEOUT EQU 2FH ; call mem_to_ram
A_DISP_NO_MEM EQU 2DH
A_DELTIME_COMP EQU 28H ; mov EndAddL,#00h
; mov EndAddH,#30h
KEYUP EQU 2BH;'u' ; call end_to_ram
KEYDOWN EQU 2AH;'d'
KEYOUT EQU 2CH;'q' mov p1,#0h
DELCODE EQU 01H ; mov dptr,#2345h
NORCODE EQU 02H ; mov a,dph
INCODE EQU 03H ; call to_serial
SAVE_TO EQU 06H ; mov a,dpl
; call to_serial
StartAddH EQU 30H
StartAddL EQU 00H
; mov a,#31h
RAMMEM EQU 2FFFH ; call sbyte
RAMENDL EQU 2FFEh ruk00: ; call rbyte
RAMENDH EQU 2FFDH ; mov p1,a
; inc a
RECLONG EQU 22 ; call sbyte
TIMELONG EQU 10 ; jmp ruk00
SUB_MAIN:

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ทางการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

; mov a,#02h          NOT_DISPTIME: CJNE
; call sbyte          A,#Q_DEL_1USER,NOT_DEL1USE
; jmp sub_main        CALL MAIN_DEL_MEM
                     JMP OUT_INT

CALL RBYTE

NOT_DEL1USE: CJNE
MOV MEMKEY,A        A,#Q_DEL_AUSER,NOT_DELAUSER
CJNE A,#Q_SAVE,NOT_SAVE CALL MAIN_DELALL_MEM
                     JMP OUT_INT

CALL MAIN_SAVE_CODE

; mov p1,#55h        NOT_DELAUSER: CJNE
JMP OUT_INT          A,#Q_DEL_TIME,NOT_DELTIME
NOT_SAVE:           CALL MAIN_DEL_TIME
CJNE A,#Q_DISP,NOT_DISP
CALL RAM_TO_MEM     NOT_DELTIME:
CALL MAIN_DISP_MEM  OUT_INT:
;Display Code of Member JMP SUB_MAIN
JMP OUT_INT

;*****
NOT_DISP: CJNE
A,#Q_USE_IN,NOT_USE table: db
CALL RAM_TO_MEM    30h,31h,32h,33h,34h,35h,36h,37h,38h,39h,41h
CALL RAM_TO_END    ,42h,43h,44h,45h,46h
CALL MAIN_SAVE_TIME
JMP OUT_INT

to_serial:
NOT_USE: CJNE      push dph
A,#Q_DISP_TIME,NOT_DISPTIME push dpl
CALL RAM_TO_MEM   push acc
CALL RAM_TO_END   swap a
CALL MAIN_DISP_TIME anl a,#0fh
                     mov dptr,#table
                     movc a,@a+dptr
                     call sbyte

```

```

pop acc                                MOV DPL,A
anl a,#0fh                             push dph
mov dptr,#table                         push dpl
movc a,@a+dptr                          CALL SUB_COMP
call sbyte                              pop dpl
pop dpl                                  pop dph
pop dph                                  JB FEQU,C_CODE_EQ
                                           INC R7
ret                                       MOV A,R7
;*****
;Sub is resive code from serail to code C_NO_M:
buffer C_CODE_EQ:
; RET
REV_CODE:
MOV R0,#CODEBUF ;*****
MOV R2,#08H SUB_COMP: MOV R2,#08H
REV_LOOP: CALL RBYTE INC DPTR
MOV @R0,A MOV R0,#CODEBUF
INC R0 L_SUB_C: MOVX A,@DPTR
DJNZ R2,REV_LOOP MOV R3,A
RET MOV A,@R0
;*****
;Compare Code new with Memer Code INC DPTR
INC R0
DJNZ R2,L_SUB_C
COMP_CODE: SETB FEQU
MOV R3,MEMBER S_COMP_E: RET
CJNE R3,#00H,C_HAVE_M
JMP C_NO_M ;*****
C_HAVE_M: MOV R7,#01H
L_COMP_1: MOV A,R7
MOV B,#0AH
MUL AB
MOV DPH,B
MOV DPH,#STARTADDH

```

```

MOV DPL,#STARTADDL
MOV A,#00H
MDELT_LOOP: MOVX @DPTR,A
INC DPTR
MOV A,DPH
CJNE
A,#0FFH,MDELT_LOOP
MOV A,DPL
CJNE
A,#0FFH,MDELT_LOOP
MOV A,#A_DELTIME_COMP
CALL SBYTE
CALL RETURN_NOR
RET
;*****
;*****
RETURN_NOR:
MOV C_MEM,#01H
RN_LOOP: MOV A,C_MEM
CALL ADD_MEM_2DPTR
MOVX A,@DPTR
CJNE A,#DELCODE,RETURN
JMP RN_JUMP
RETURN: MOV A,#NORCODE
MOVX @DPTR,A
RN_JUMP: INC C_MEM
MOV A,MEMBER
CJNE A,C_MEM,RN_LOOP
RET
;*****
;*****
RAM_TO_MEM: MOV
DPTR,#RAMMEM
MOVX A,@DPTR
MOV MEMBER,A
RET
;*****
;*****
MEM_TO_RAM: MOV DPTR,#RAMMEM
MOV A,MEMBER
MOVX @DPTR,A
RET
;*****
;*****
RAM_TO_END: MOV DPTR,#RAMENDL
MOVX A,@DPTR
MOV EndAddL,A
MOV DPTR,#RAMENDH
MOVX A,@DPTR
MOV EndAddH,A
RET
;*****
;*****
END_TO_RAM: MOV DPTR,#RAMENDL
MOV A,EndAddL
MOVX @DPTR,A
MOV DPTR,#RAMENDH
MOV A,EndAddH
MOVX @DPTR,A
RET
;*****
;*****
RBYTE: JNB RI,$
CLR RI
MOV A,SBUF
RET
;*****
;*****
SBYTE: JNB TI,$
CLR TI
MOV SBUF,A

```

เอกสารนี้เป็นเอกสารสงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น โปรดดูว่าต้นฉบับใหม่ไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

RET                                     CALL SBYTE
                                         RET
;*****
;Give address end of member to dptr
ADD_END_MEM: MOV  A, MEMBER
              MOV  B, #0AH
              MUL  AB
              MOV  DPH, B
              MOV  DPL, A
              RET
                                         CODE_OVER:
                                         ; mov  a,dph
                                         ; call to_serial
                                         ; mov  a,dpl
                                         ; call to_serial
                                         MOV  R2, #09H
                                         SAD_LOOP: INC  DPTR
                                         DJNZ R2, SAD_LOOP
;*****
;Save code from serial to RAM
;
MAIN_SAVE_CODE: CALL RAM_TO_MEM
RAM_TO_MEM
              MOVX A, @DPTR
              MOV  A, MEMBER
              CJNE A, #00H, MEE_MEM
              MOV  MEMBER, #01H
              MOVX @DPTR, A
MEE_MEM:     CALL REV_CODE
              CLR  FEQU
              CALL COMP_CODE
              JB  FEQU, CODE_OVER
              CALL SAVE_CODE
              CALL SET_USE_NOR
              INC  MEMBER
              CALL MEM_TO_RAM
;SAVE TO RAM
SAVE_DEL_COMP: MOV  A, #A_SAVE_COMP
              MOVX @DPTR, A
              RET
                                         ;*****
                                         SET_USE_NOR: ;INC  DPTR
                                         MOV  A, #NORCODE
                                         MOVX @DPTR, A
                                         RET
;*****

```

```

SAVE_CODE: CALL MOV A,#A_DISP_NO_MEM
ADD_END_MEM CALL SBYTE
MOV A,MEMBER RET
; INC A
MOVX @DPTR,A DISP_MEE_MEM: MOV C_MEM,#01H
; call to_serial
; mov a,dph CALL DISP_C_CODE
; call to_serial CJNE A,#DELCODE,LOOP_MDM
; mov a,dpl JMP INC_DELCODE
; call to_serial
INC DPTR LOOP_MDM: CALL DISP_MEM_SEN
CALL SUB_SAVE ; mov a,c_mem
RET ; call to_serial
CALL MOV_BUF2S
; save code from Buffer_Code to RAM KEY_NDF: CALL RBYTE
; MOV R4,A
; CJNE R4,#KEYUP,KEY_NU
SUB_SAVE: MOV R0,#CODEBUF INC_DELCODE:
MOV R2,#08H ; MOV A,C_MEM
L_SUB_S: MOV A,@R0 mov a,member
MOVX @DPTR,A dec a
INC R0 cjne a,c_mem,up_not_top
INC DPTR ; CJNE A,MEMBER,UP_NOT_TOP
DJNZ R2,L_SUB_S MOV C_MEM,#01H
RET
CALL DISP_C_CODE
;Sub Program main is display code of ; jmp loop_mdm
memer
; JMP INC_DELCODE
MAIN_DISP_MEM: UP_NOT_TOP: INC C_MEM
MOV A,MEMBER CALL DISP_C_CODE
CJNE CJNE A,#DELCODE,LOOP_MDM
A,#00H,DISP_MEE_MEM JMP INC_DELCODE

```

```

KEY_NU:    CJNE                MOV  @R0,A
R4,#KEYDOWN,KEY_ND          INC  DPTR
DEC_DELCODE: MOV  A,C_MEM      INC  R0
                        CJNE  A,#01H,DOWN_NOT_B
                        MOV   C_MEM,MEMBER      DJNZ  R2,LOOP_DMS
                        dec   c_mem             RET

;*****
CALL  DISP_C_CODE          MOV_BUF2S: MOV  R0,#CODEBUF
CJNE                MOV  R2,#08H
A,#DELCODE,LOOP_MDM      LOOP_MB2S: MOV  A,@R0
                        JMP   DEC_DELCODE      CALL  SBYTE
                        INC   R0
DOWN_NOT_B: DEC  C_MEM      DJNZ  R2,LOOP_MB2S
CALL  DISP_C_CODE      RET
CJNE                ;##### new
A,#DELCODE,LOOP_MDM      #####
;jmp  loop_mdm
JMP  DEC_DELCODE      ;*****
KEY_ND:                ;Delete one member

CJNE                MAIN_DEL_MEM:
R4,#KEYOUT,KEY_NDF      CALL  REV_CODE
RET                    CLR  FEQU
                        CALL  COMP_CODE
;*****
DISP_MEM_SEN: MOV  A,C_MEM      JNB  FEQU,D_NOT_M
                        MOV  A,R7
                        CALL  ADD_CHE_MEM
                        MOV  A,#DELCODE
                        MOVX  @DPTR,A
                        MOV  A,#A_DEL_COMP
                        CALL  SBYTE
                        MOV  R2,#08H          RET
                        MOV  R0,#CODEBUF      D_NOT_M: MOV  A,#A_DEL_NOT
LOOP_DMS:  MOVX  A,@DPTR      CALL  SBYTE

```

```

RET
;*****
;????????? --> ;Delete all member Code
ADD_MEM_2DPTR: ;
MOV B,#0AH MAIN_DELALL_MEM:
MUL AB MOV MEMBER,#00H
MOV DPH,B CALL MEM_TO_RAM ;**
MOV DPL,A MOV A,#A_DELALL_COMP
MOV R2,#9 CALL SBYTE
AM2_LOOP: INC DPTR RET
DJNZ R2,AM2_LOOP
RET ;*****
;Main save time is user login/logout to RAM
;*****
DISP_C_CODE: MAIN_SAVE_TIME:
MOV A,C_MEM CALL REV_CODE
CALL ADD_MEM_2DPTR CLR FEQU
MOVX A,@DPTR CALL COMP_CODE
RET JNB FEQU,MST_NOT_MEM
MOV StepMem,R7
;????????????? <-- ; mov p1,stepmem
;*****
;Return Address of Check Byte of Member
MOV A,StepMem
CALL ADD_CHE_MEM
MOVX A,@DPTR
ADD_CHE_MEM: MOV B,#0AH CJNE
MUL AB A,#DELCODE,MST_NOT_DEL
MOV DPH,B MST_NOT_MEM: MOV A,#A_NOT_MEM
MOV DPL,A CALL SBYTE
MOV R2,#09H JMP MST_GET_OUT
LSDM: INC DPTR MST_NOT_DEL:
DJNZ R2,LSDM push acc
RET mov a,#a_time_comp

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไมออนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

call sbyte                CALL SAVE_TIME_OUT
pop acc                   CALL SAVE_CODE_OUT

CJNE                      ; mov p1,#0f0h
A,#NORCODE,MST_NOT_NOR   mov a,#a_time_out
CALL RBYTE               call sbyte
CJNE                      JMP MST_GET_OUT
A,#Q_SENT_TIME,MST_NOT_SEN

CALL REV_TIME            MST_EROR: MOV A,#A_TIME_EROR
CALL SAVE_CODE_TIME      CALL SBYTE
CALL SAVE_TIME_IN        MST_NOT_SEN:
CALL INC_ADD_TIME        MST_GET_OUT:
CALL END_TO_RAM          MST_NOT_IN:
;                          RET
CALL SAVE_CODE_IN        ;
;                          *****
; mov p1,#0fh            ;
mov a,#a_time_in         REV_TIME:
call sbyte               MOV R0,#TIMEBUF
;future add              MOV R2,#TIMELONG
;                          RET_LOOP: CALL RBYTE
JMP MST_GET_OUT          MOV @R0,A
MST_NOT_NOR: CJNE        INC R0
A,#INCODE,MST_NOT_IN    DJNZ R2,RET_LOOP
CALL RBYTE               RET
CJNE
A,#Q_SENT_TIME,MST_NOT_SEN ; *****
CALL REV_TIME            ;
;                          SAVE_CODE_TIME:
CALL ADD_TIME_OUT        MOV DPH,EndAddH
;                          MOV DPL,EndAddL
mov p1,dpl               MOV A,StepMem
;                          MOVX @DPTR,A
CJNE A,#00,MST_EROR     RET

```

```

MOV A,#SAVE_TO
;*****
SAVE_TIME_IN:
MOV DPH,ENDADDH
MOV DPL,ENDADDL
INC DPTR
; mov a,dph
; call to_serial
; mov a,dpl
; call to_serial
CALL SUB_SAVE_TIME
RET
;*****
SUB_SAVE_TIME:
MOV R0,#TIMEBUF
MOV R2,#TIMELONG
SST_LOOP: MOV A,@R0
MOVX @DPTR,A
INC R0
INC DPTR
DJNZ R2,SST_LOOP
RET
;*****
SATO_LOOP: INC DPTR
DJNZ R2,SATO_LOOP
RET
;*****
INC_ADD_TIME:
MOV DPH,EndAddH
MOV DPL,EndAddL
MOV R2,#RECLONG
IAT_LOOP: INC DPTR
DJNZ R2,IAT_LOOP
MOV EndAddH,DPH
MOV EndAddL,DPL
RET
;*****
ADD_CHECK_TIME:
CALL STARTADD_TO
;Get start Address of save time Out to DPTR
STARTADD_TO:
MOV DPH,EndAddH
MOV DPL,EndAddL
MOV R2,#TIMELONG+1
SATO_LOOP: INC DPTR
DJNZ R2,SATO_LOOP
RET
;*****
ADD_TIME_OUT:
CALL END_TO_DPTR
ATO_LOOP: CALL ATO_DEC_DPTR

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

MOVX A,@DPTR                                JMP ADD_JMP
CJNE                                         DPL_NOT_0: DEC R6
A,StepMem,ATO_NOT_EQ                         ADD_JMP: DJNZ R2,ADD_LOOP1
MOV A,#00H                                    MOV DPL,R6
RET                                           MOV DPH,R5
ATO_NOT_EQ: CALL                             RET
COMP_START_DPTR                             ;*****
CJNE A,#00H,ATO_LOOP                         END_TO_DPTR:
MOV A,#0FFH                                    MOV DPH,EndAddH
RET                                           MOV DPL,EndAddL
                                           RET
;*****                                     ;*****
SAVE_CODE_IN:                               COMP_START_DPTR:
MOV A,StepMem                                MOV A,DPH
CALL ADD_CHE_MEM                            CJNE
MOV A,#INCODE                               A,#STARTADDL,CSD_NOT_EQ
MOVX @DPTR,A                                MOV A,DPL
RET                                           CJNE
;*****                                     A,#STARTADDH,CSD_NOT_EQ
SAVE_CODE_OUT:                              MOV A,#00H
MOV A,StepMem                                RET
CALL ADD_CHE_MEM                            CSD_NOT_EQ: MOV A,#0FFH
MOV A,#NORCODE                               RET
MOVX @DPTR,A
RET                                           ;*****
;*****                                     ; new 09/05/00
ATO_DEC_DPTR:                               MAIN_DISP_TIME:
MOV R2,#RECLONG                              CALL RAM_TO_END
MOV R5,DPH                                    MOV DISPTIMEH,EndAddH
MOV R6,DPL                                    MOV DISPTIMEL,EndAddL
ADD_LOOP1: CJNE                              CALL COMP_ADD_TIME
R6,#00H,DPL_NOT_0                            CJNE A,#0H,MDT_SAVE
DEC R5
MOV R6,#0FFH                                MOV A,#A_NO_SAVET

```

```

CALL SBYTE                                MDT_ND:    CJNE
RET                                         R4,#KEYOUT,MDT_JUMP
                                           RET

MDT_SAVE:
CALL DEC_DISPTIME                          ;*****
MDT_LOOP:  CALL
MDT_SEN_CODE                                COMP_ADD_TIME:
CALL RBYTE                                  MOV  A,DISPTIMEL
CJNE A,#Q_REV_COM,$                        CJNE A,#StartAddL,CAT_NOT_EQ
CALL SEN_TIME_IN                           MOV  A,DISPTIMEH
CALL RBYTE                                  CJNE A,#StartAddH,CAT_NOT_EQ
CJNE A,#Q_REV_COM,$                        MOV  A,#00H
CALL CHECK_TO                               RET
CJNE                                         CAT_NOT_EQ: MOV  A,#0FFH
A,#SAVE_TO,NO_TIMEOUT                      RET
CALL SEN_TIME_OUT                          ;*****
JMP  MDT_JUMP

NO_TIMEOUT: MOV                            DEC_DISPTIME:
A,#A_NO_TIMEOUT                            MOV  R2,#RECLONG
CALL SBYTE                                  DD_LOOP:  MOV  A,DISPTIMEL
                                           CJNE A,#00H,DD_JUMP1
MDT_JUMP:  CALL RBYTE                      MOV  DISPTIMEL,#0FFH
MOV  R4,A                                   DEC  DISPTIMEH
CJNE R4,#KEYUP,MDT_NU                      JMP  DD_JUMP
CALL MDT_KEYUP                             DD_JUMP1: DEC  DISPTIMEL
JMP  MDT_LOOP                              DD_JUMP:  DJNZ R2,DD_LOOP

MDT_NU:   CJNE                               RET
R4,#KEYDOWN,MDT_ND                          ;*****
CALL MDT_KEYDOWN
JMP  MDT_LOOP                                MDT_SEN_CODE:
                                           MOV  DPH,DISPTIMEH
                                           MOV  DPL,DISPTIMEL

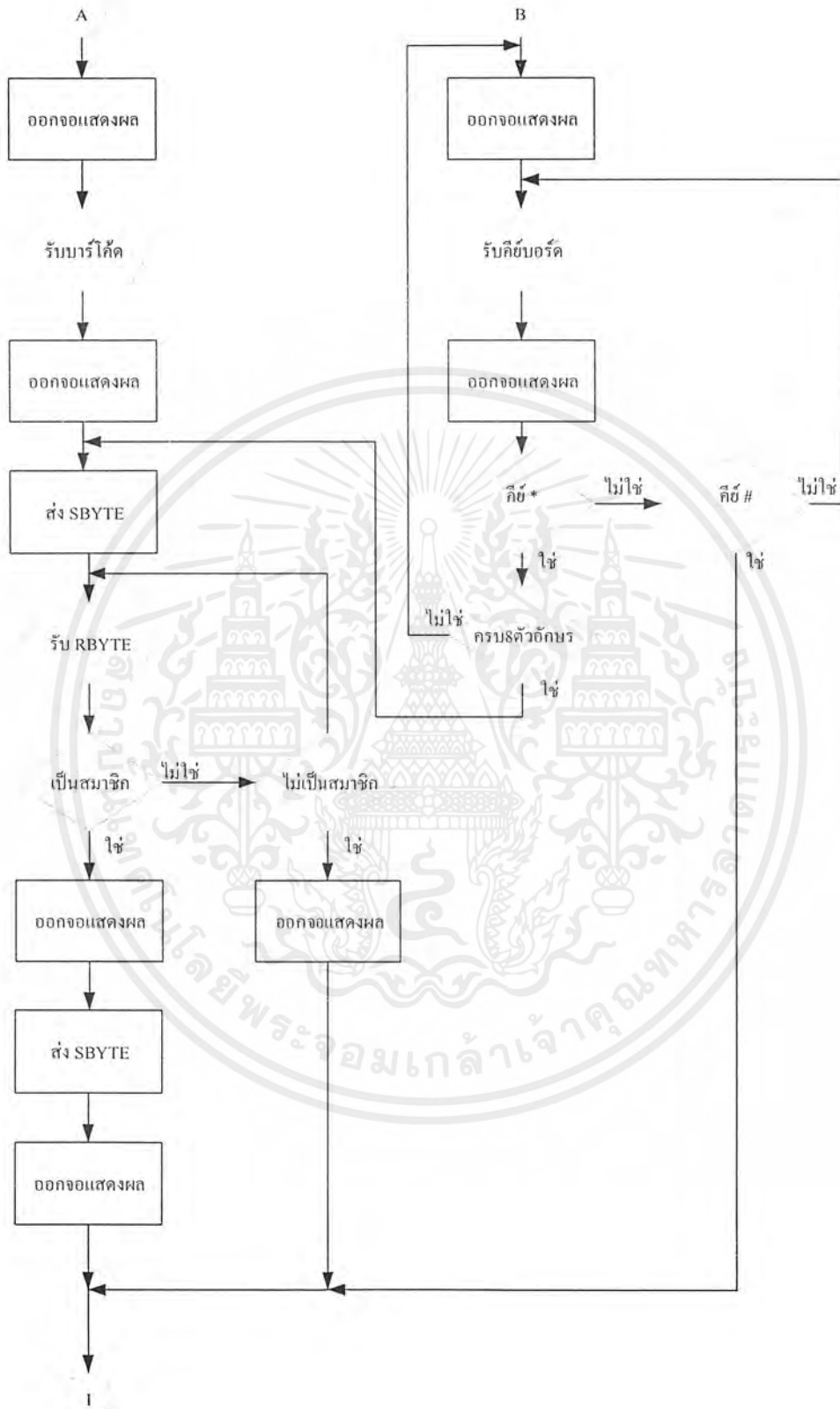
```



```

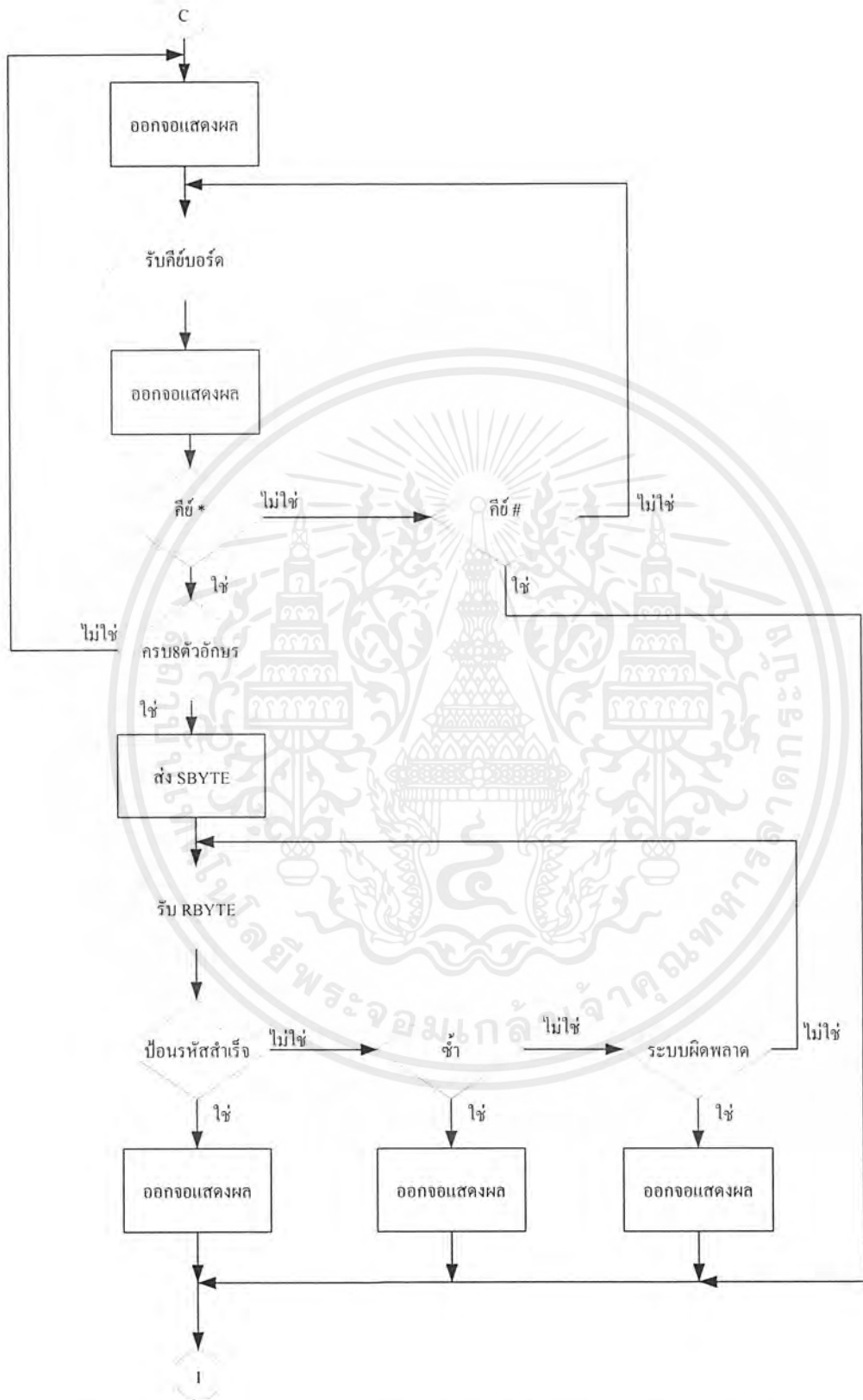
    RET
;*****
MDT_KEYDOWN:
    CALL INC_DISPTIME
    CALL DIS_VS_END
    CJNE A,#00H,MDTK_JUMP
    MOV  DISPTIMEL,#StartAddL
    MOV
DISPTIMEH,#StartAddH
MDTK_JUMP:  RET
;*****
INC_DISPTIME:
    MOV  R2,#RECLONG
    MOV  DPH,DISPTIMEH
    MOV  DPL,DISPTIMEL
INCD_LOOP: INC  DPTR
            DJNZ R2,INCD_LOOP
            MOV  DISPTIMEH,DPH
            MOV  DISPTIMEL,DPL
            RET
;*****
DIS_VS_END:
    MOV  A,DISPTIMEL
    CJNE
A,EndAddL,DVE_NOT_EQ
    MOV  A,DISPTIMEH
    CJNE
A,EndAddH,DVE_NOT_EQ
    MOV  A,#00H
    RET
DVE_NOT_EQ: MOV  A,#0FFH
            RET
            END

```



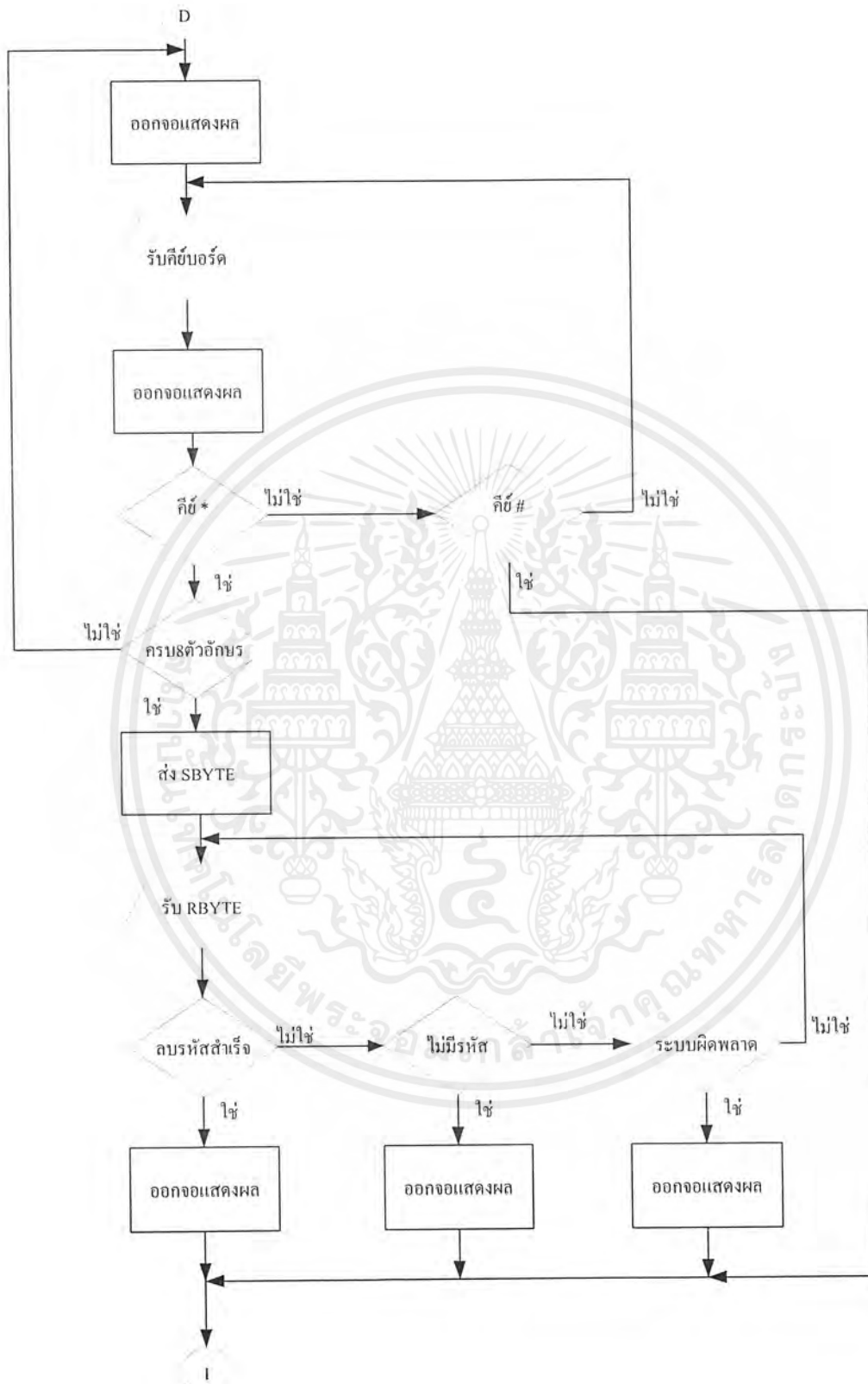
รูปที่ ข.1 แผนผังการทำงานของเครื่องตรวจสอบบุคคลเข้าหน่วยงาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



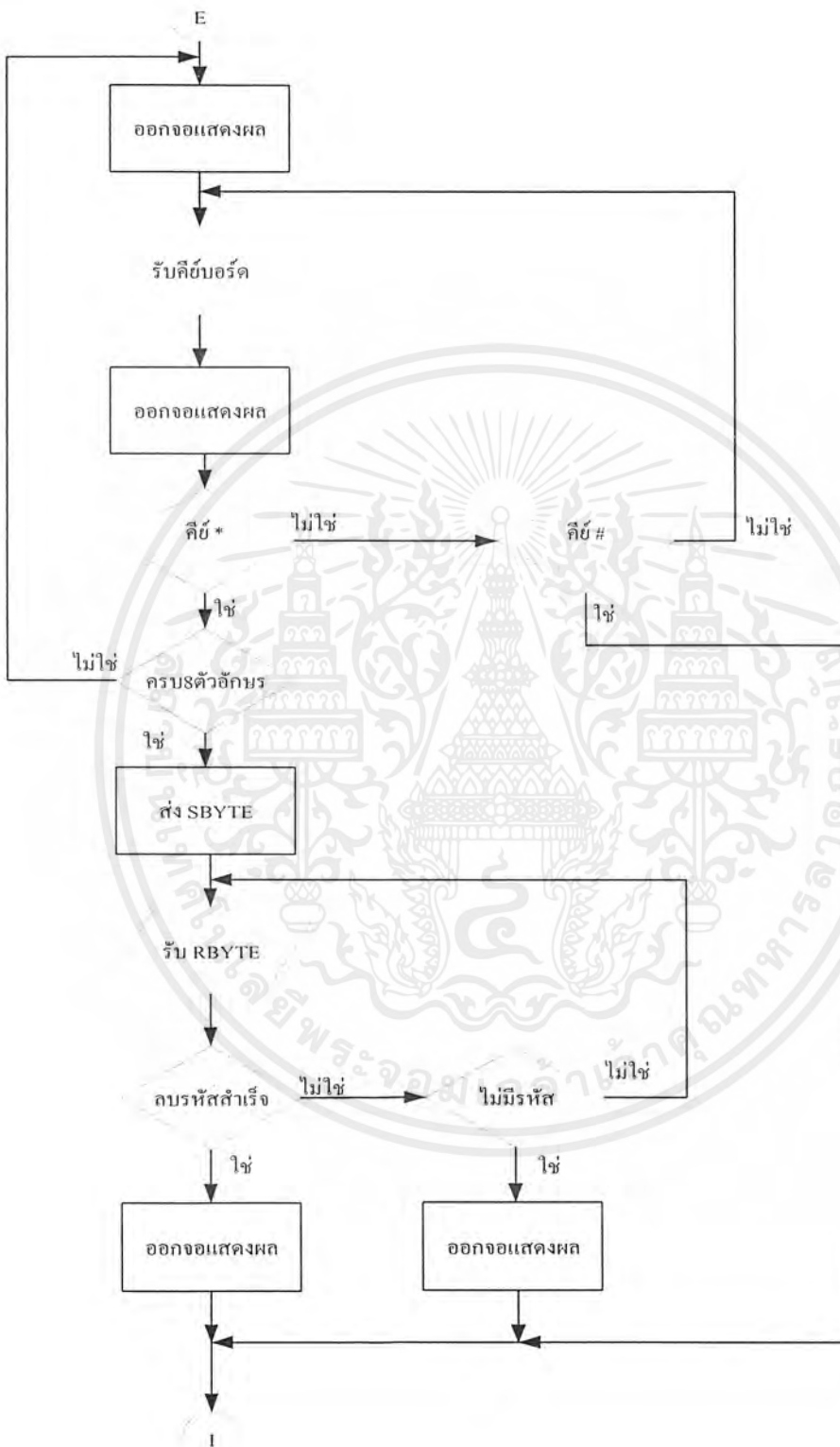
รูปที่ ข.1 (ต่อ) แผนผังการทำงานของเครื่องตรวจสอบบุคคลเข้าหน่วยงาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



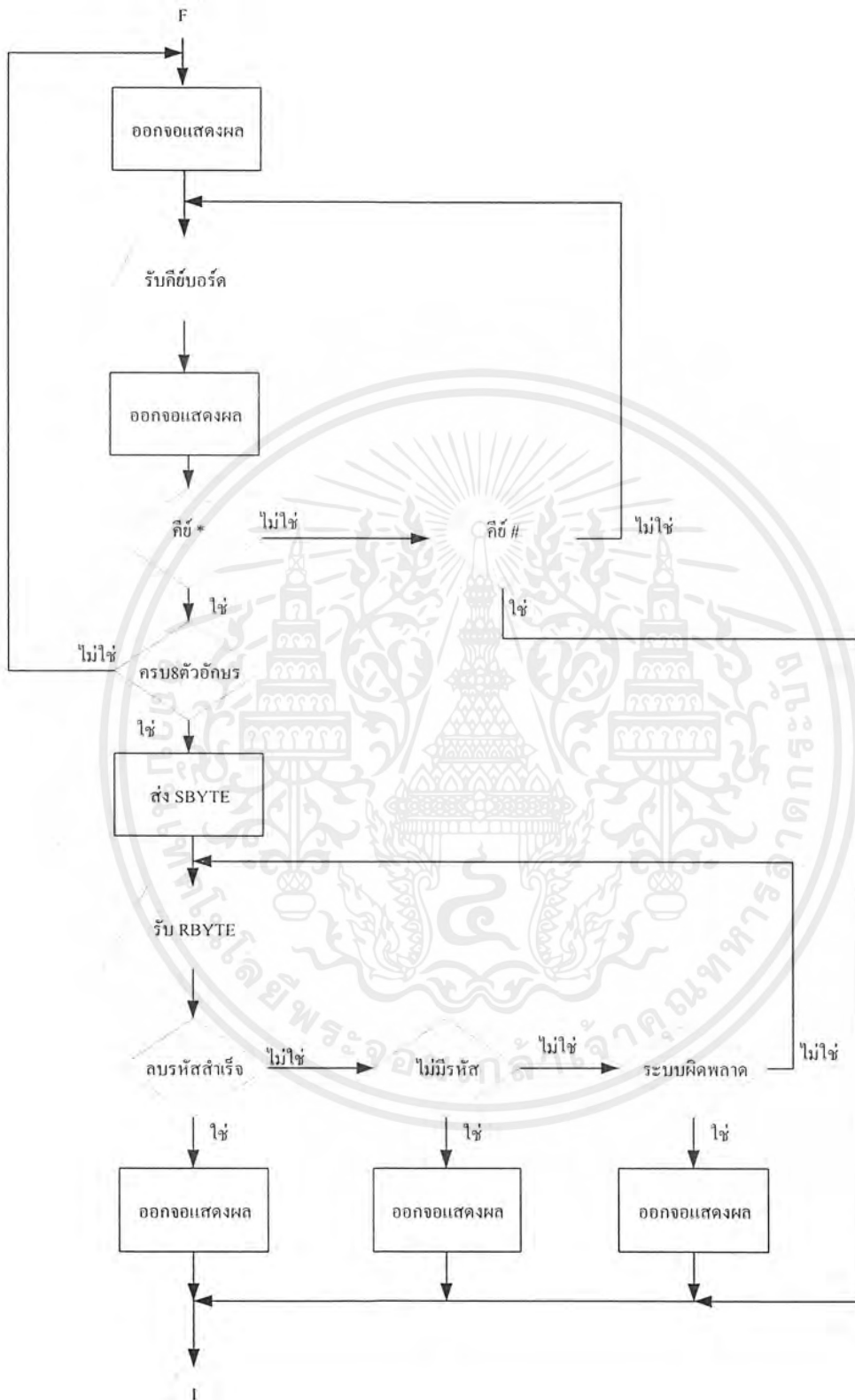
รูปที่ ข.1 (ต่อ) แผนผังการทำงานของเครื่องตรวจสอบบุคคลเข้าหน่วยงาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



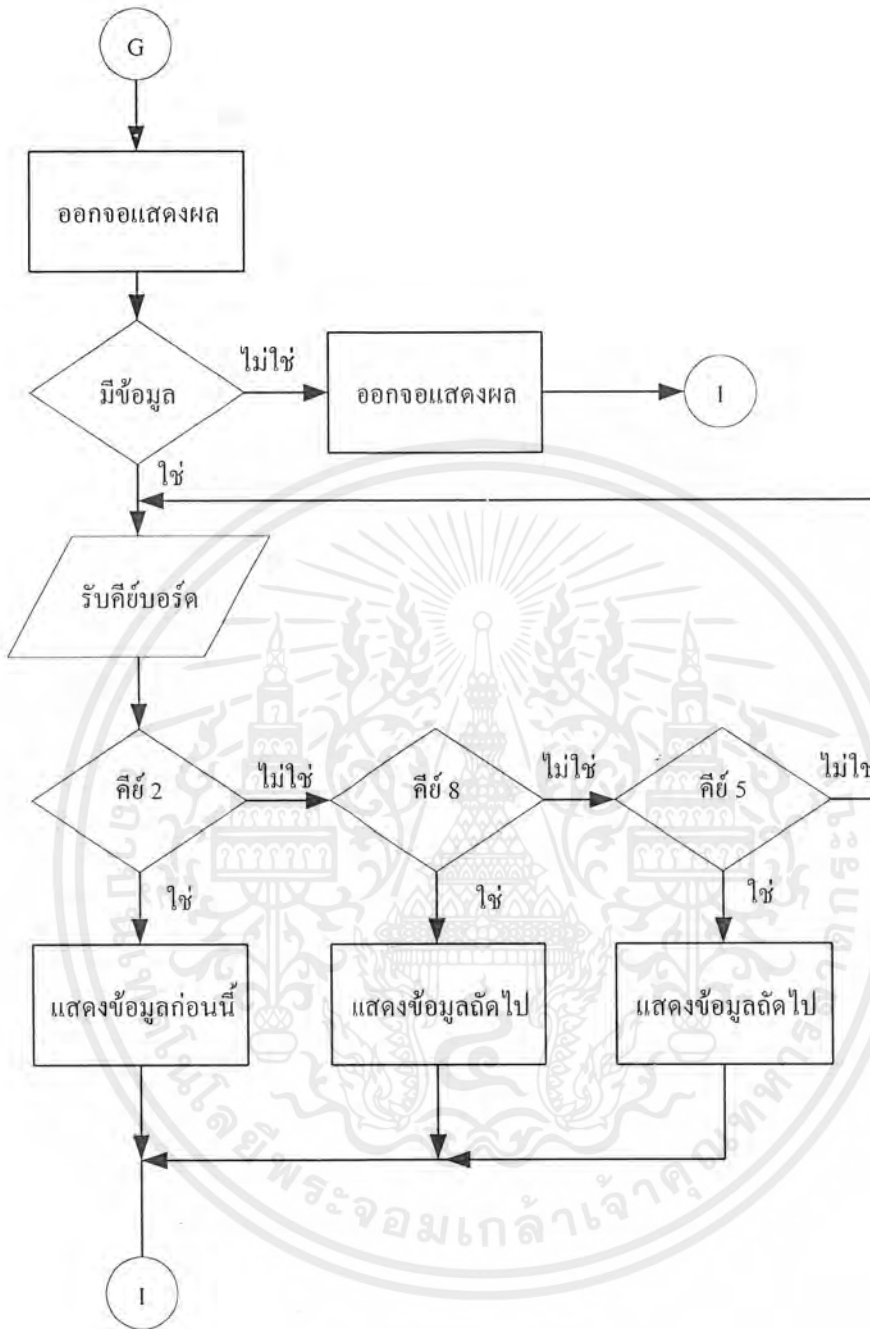
รูปที่ ข.1 (ต่อ) แผนผังการทำงานของเครื่องตรวจสอบบุคคลเข้าหน่วยงาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



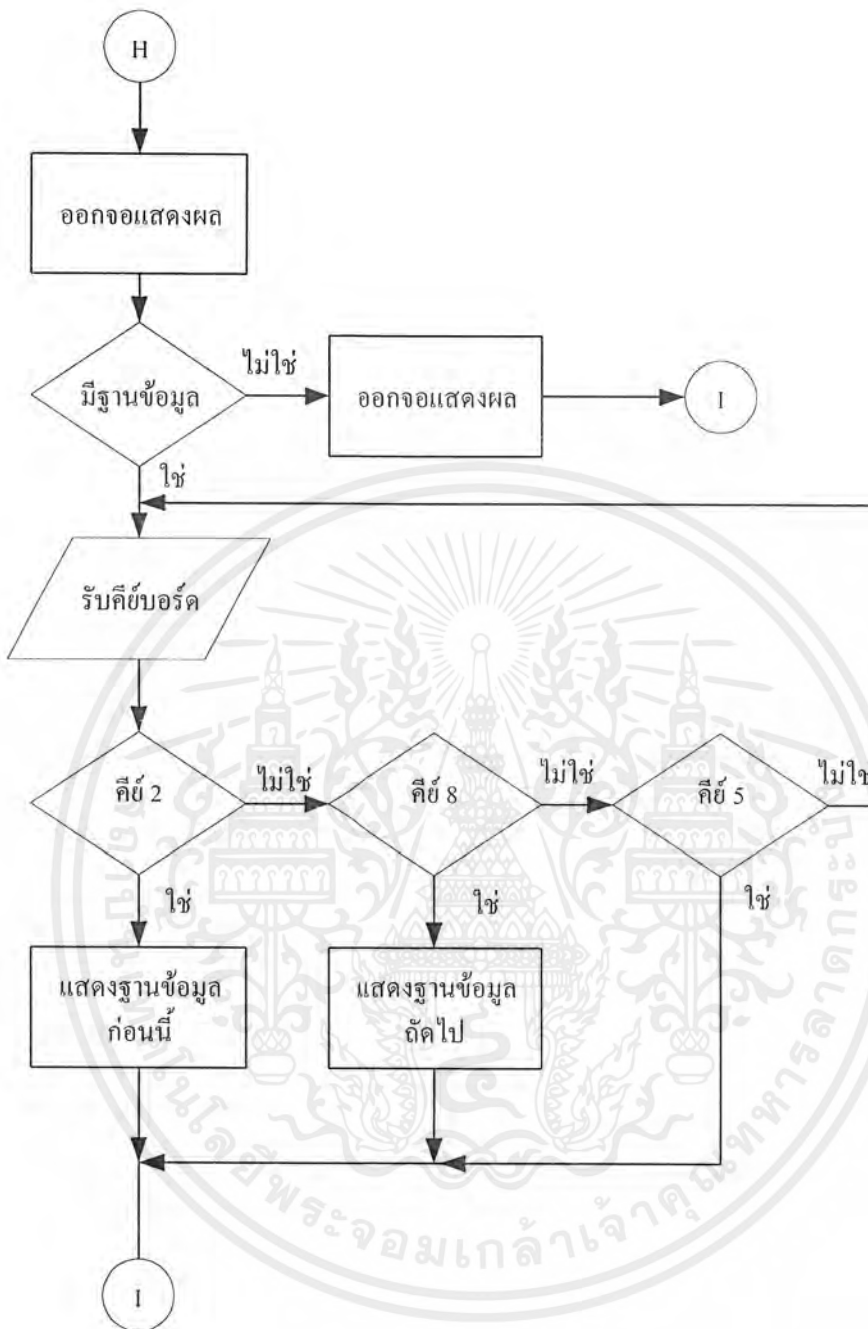
รูปที่ ข.1 (ต่อ) แผนผังการทำงานของเครื่องตรวจสอบบุคคลเข้าหน่วยงาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ ข.1 (ต่อ) แผนผังการทำงานของเครื่องตรวจสอบบุคคลเข้าหน่วยงาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

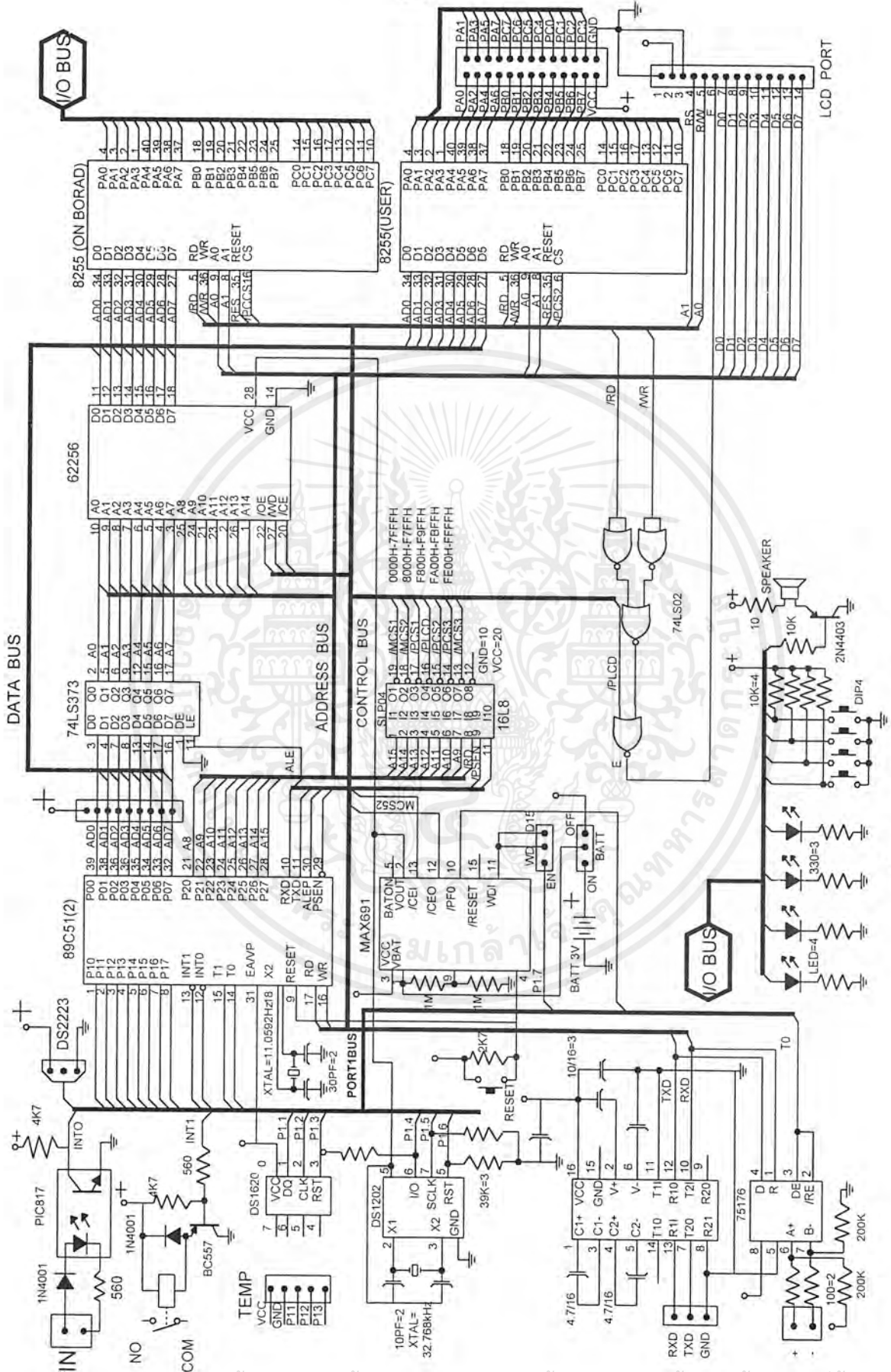


รูปที่ ข.1 (ต่อ) แผนผังการทำงานของเครื่องตรวจสอบบุคคลเข้าหน่วยงาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

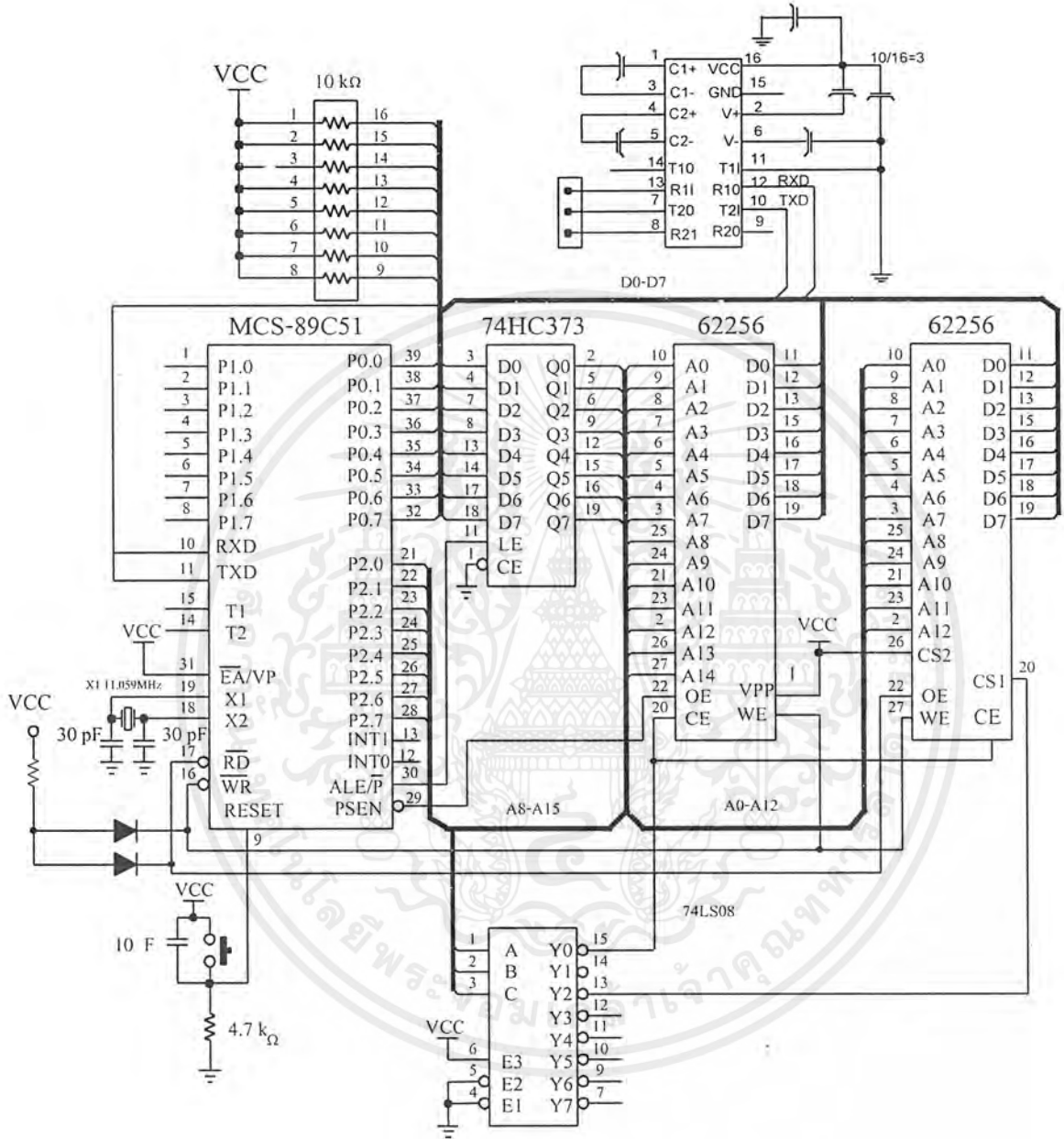


เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ ค.1 วงจรควบคุมจาก ANT-C51M

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



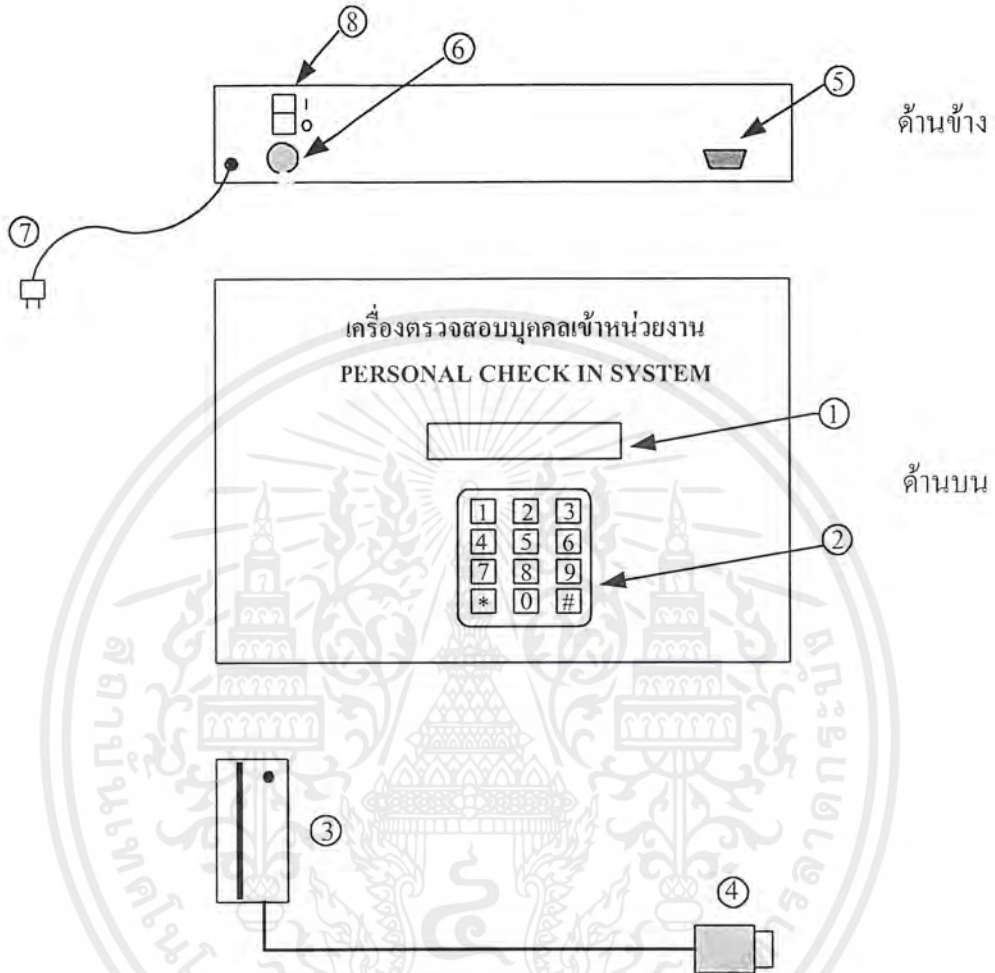
รูปที่ ค.2 วงจรบันทึกข้อมูลและตรวจสอบข้อมูล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ส่วนประกอบ



หมายเลข	ความหมาย
1	จอแสดงผล (LCD)
2	คีย์บอร์ด
3	เครื่องอ่านบาร์โค้ด
4	พอร์ตบาร์โค้ด (ตัวผู้)
5	พอร์ตบาร์โค้ด (ตัวเมีย)
6	ฟิวส์
7	เต้าเสียบ 220 V 50Hz
8	สวิตช์เปิด-ปิด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การใช้งาน

การใช้งานแบ่งออกเป็น 2 แบบคือ

1. การใช้งานทั่วไป
2. การใช้งานส่วน โปรแกรม

1. การใช้งานทั่วไป

เมื่อต้องการใช้เครื่องตรวจสอบบุคคลเข้าหน่วยงานจะใช้งานได้ 2 แบบคือ

1.1 ใช้บาร์โค้ด

เมื่อต้องการป้อนข้อมูลโดยใช้บาร์โค้ดมีลำดับขั้นตอนดังนี้

1. กด 0 หน้าจอจะปรากฏ

Login:_
Mode of bar code

2. ทำการรูดบัตรหน้าจอจะปรากฏหมายเลขจากบาร์โค้ด

Login:41031501
Mode of bar code

1.2 ใช้คีย์บอร์ด

เมื่อต้องการป้อนข้อมูลโดยใช้คีย์บอร์ดมีลำดับขั้นตอนดังนี้

1. กด * หน้าจอจะปรากฏ

Login:_
Mode of keyboard

2. ทำการกรอกรหัสลงไปแล้วตามด้วย * ปิดท้าย

การแสดงผลสถานะของผู้ใช้

การแสดงผลสถานะของผู้ใช้มี 2 แบบคือ

1. “ผ่าน” หน้าจอจะแสดงคำว่า “YOU MEMBER”
2. “ไม่ผ่าน” หน้าจอจะปรากฏคำว่า “YOU NO MEMBER”

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2. การใช้งานส่วนโปรแกรม

การใช้งานส่วนโปรแกรมแบ่งออกเป็น 6 แบบคือ

- 2.1 การบันทึกฐานข้อมูล
- 2.2 การดูการลงทะเบียนฐานข้อมูล
- 2.3 การดูทะเบียนการเข้า-ออก
- 2.4 การลบบันทึกฐานข้อมูลทั้งหมด
- 2.5 การลบบันทึกฐานข้อมูลรายบุคคล
- 2.6 การลบบันทึกเวลาทั้งหมด

2.1 การบันทึกฐานข้อมูล มีขั้นตอนดังนี้

1. ทำการกดปุ่ม 9 หน้าจอจะแสดงดังนี้

Login: _
WR data to Ram

2. ทำการป้อนข้อมูล อย่างน้อย 8 ตัวอักษร โดยคีย์บอร์ดแล้วปิดท้ายด้วย * เมื่อเครื่องทำการบันทึกเสร็จเรียบร้อยหน้าจอจะแสดง

OK
put data in ram

2.2 การดูการลงทะเบียนฐานข้อมูล มีขั้นตอนดังนี้

1. ทำการกดปุ่ม 5 หน้าจอจะแสดงดังนี้

WAIT

2. รอสักครู่หน้าจอจะแสดงฐานข้อมูลที่มีอยู่

41031501
Exit list push 5

สามารถดูข้อมูลถัดไปได้โดยการกด 2 (UP) หรือ 8 (DOWN) หรือต้องการออกจากการทำงานกด 5

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.3 การดูทะเบียนการเข้า-ออก มีขั้นตอนดังนี้

1. ทำการกด 4 หน้าจอจะแสดง

WAIT
List time in-out

2. รอสักครู่หน้าจอจะแสดงทะเบียนการเข้า-ออกที่มีอยู่

รหัส	วัน	เดือน	ปี
40131501	15	05	00
1850	150500	1852	
เวลาเข้า	วัน	เดือน	ปี

สามารถดูข้อมูลถัดไปได้โดยการกด 2 (UP) หรือ 8 (DOWN) หรือต้องการออกจากการทำงานกด 5

2.4 การลบบันทึกฐานข้อมูลทั้งหมด

1. ทำการกด 7 หน้าจอจะแสดง

Del all database
* yes / # no

2. กด * เมื่อต้องการลบบันทึกฐานข้อมูลทั้งหมด กด # เมื่อไม่ต้องการลบบันทึกฐานข้อมูลทั้งหมด

2.5 การลบบันทึกฐานข้อมูลรายบุคคล

1. ทำการกด 8 หน้าจอจะแสดง

Login: _
Do you want to del?

2. ทำการกรกรหัสจำนวน 8 ตัวอักษรที่ต้องการลบออกแล้วตามด้วย *

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.6 การลบบันทึกเวลาทั้งหมด

1. ทำการกด 6 หน้าจอจะแสดง

Del data record * yes / # no

2. กด * เมื่อต้องการลบบันทึกเวลาทั้งหมด กด # เมื่อไม่ต้องการลบบันทึกเวลาทั้งหมด



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บรรณานุกรม

ทวีศักดิ์ ไพศาลภาณุมาศ. สารระล่ำคัญของบาร์โค้ด. เซมิคอนดัคเตอร์อิเล็กทรอนิกส์ ฉบับที่ 145
ประจำเดือน ธันวาคม-มกราคม : กรุงเทพมหานคร, 2538.

ไพรัช ธีชชัยพงษ์. ทฤษฎีและการใช้งานไมโครโปรเซสเซอร์และไมโครคอมพิวเตอร์. บริษัทซีเอ็ด
ยูเคชั่น จำกัด (มหาชน) : กรุงเทพมหานคร, 2521.

สุนทร วิฑูสุรพจน์. การใช้งานไมโครคอนโทรลเลอร์ ตระกูล 8051. บริษัทซีเอ็ดยูเคชั่น จำกัด
(มหาชน) : กรุงเทพมหานคร, 2537.

Microcontroller AT89C51 Device Data, 1987.



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ประวัติผู้แต่ง



ชื่อผู้ทำปฏิญานិพนธ์

นาย กิตติรัช สุติศา

วันเดือนปีเกิด

23 ธันวาคม 2520

สถานที่เกิด

โรงพยาบาลมหาราชจังหวัดเชียงใหม่

ภูมิลำเนาเดิม

จังหวัดเชียงใหม่

ที่อยู่ปัจจุบัน

23/1 ซอย ป่าห้า ถนน ห้วยแก้ว อำเภอ เมือง

จังหวัด เชียงใหม่ 50200

โทรศัพท์

(053) 224-790 , 213-620

ประวัติการศึกษา

ประถมศึกษา

โรงเรียนปรี้นรอยแยลส์ วิทยาลัย

มัธยมศึกษาตอนต้น

โรงเรียนสาธิตมหาวิทยาลัยเชียงใหม่

ประกาศนียบัตรวิชาชีพ(ปวช.)

วิทยาลัยเทคนิคเชียงใหม่

ประกาศนียบัตรวิชาชีพชั้นสูง(ปวส.)

สถาบันเทคโนโลยีราชมงคลวิทยาเขตภาคพายัพ

ปริญญาตรี

สาขาวิชาเทคโนโลยีการวัดคุมทางอุตสาหกรรม

ภาควิชาครุศาสตร์วิศวกรรม

คณะครุศาสตร์อุตสาหกรรม

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหาร

ลาดกระบัง

คติพจน์

อย่าปล่อยให้เวลาผ่านไปโดยไร้ประโยชน์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ประวัติผู้แต่ง



ชื่อผู้ทำปฏิญานิพนธ์	วันวิสาข์ พูลคล้าย
วันเดือนปีเกิด	1 มิถุนายน 2520
สถานที่เกิด	โรงพยาบาลพุทธชินราช จ. พิษณุโลก
ภูมิลำเนาเดิม	จังหวัดพิจิตร
ที่อยู่ปัจจุบัน	34 หมู่ 8 ตำบล หัวดง อำเภอ เมือง จังหวัดพิจิตร 66170
โทรศัพท์	056- 674223 ,697099 หรือ 054 – 350403

ประวัติการศึกษา

ประถมศึกษา	โรงเรียนบ้านหัวดง
มัธยมศึกษาตอนต้น	โรงเรียนตะพานหิน
ประกาศนียบัตรวิชาชีพ(ปวช.)	วิทยาลัยเทคนิคพิจิตร
ประกาศนียบัตรวิชาชีพชั้นสูง(ปวส.)	สถาบันเทคโนโลยีราชมงคลวิทยาเขตภาคพายัพ
ปริญญาตรี	สาขาวิชาเทคโนโลยีการวัดคุมทางอุตสาหกรรม ภาควิชาครุศาสตร์วิศวกรรม คณะครุศาสตร์อุตสาหกรรม สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหาร ลาดกระบัง
คติพจน์	พอใจในสิ่งที่ตนมี

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ประวัติผู้แต่ง



ชื่อผู้ทำปฏิญานិพนธ์	นายมนัส ตลับเงิน
วันเดือนปีเกิด	3 กรกฎาคม 2520
สถานที่เกิด	โรงพยาบาลสระบุรี
ภูมิลำเนาเดิม	จังหวัดสระบุรี
ที่อยู่ปัจจุบัน	53 หมู่ 2 ตำบล บางโฉมด อำเภอ บ้านหมอ จังหวัด สระบุรี
โทรศัพท์	036-300887

ประวัติการศึกษา

ประถมศึกษา	โรงเรียนวัดสุนทรเทพมุณี มิตรภาพที่ 188
มัธยมศึกษาตอนต้น	โรงเรียนบ้านหมอ พัฒนานุกูล
ประกาศนียบัตรวิชาชีพ(ปวช.)	วิทยาลัยเทคนิคสระบุรี
ประกาศนียบัตรวิชาชีพชั้นสูง(ปวส.)	วิทยาลัยเทคนิคท่าหลวงซีเมนต์ไทยอนุสรณ์
ปริญญาตรี	สาขาวิชาเทคโนโลยีการวัดคุมทางอุตสาหกรรม ภาควิชาครุศาสตร์วิศวกรรม คณะครุศาสตร์อุตสาหกรรม สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหาร ลาดกระบัง
คติพจน์	พิจารณาตนเอง ก่อนถึงเห็นคนอื่นผิด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้